Net-Centric Implementation

Part 4: Node Guidance

v3.2.0

26 October 2010

Net-Centric Enterprise Solutions for Interoperability (NESI) is a collaborative activity of the USN PEO for C4I and Space, the USAF Electronic Systems Center, and the Defense Information Systems Agency.

Approved for public release; distribution is unlimited.

SSIC: 3093.4

NESI-X Subdocument generated using: **View, Part 4: Node Guidance**
Generated: **Wed Nov 10 09:56:40 PST 2010**
NESI-X Version: **v2.2.0 build 6206 - 2010/11/09 17:35**

Table of Contents

Perspectives 5

NESI Executive Summary 5

Part 4: Node Guidance 7

Node Decomposition 9

Node Architectural Elements 10

Node Operating Environments 18

Core Environments 19

Edge Environments 20

General Responsibilities 22

Nodes as Stakeholders 23

Net-Centric Information Engineering 24

Internal Component Environment 25

Integration of Legacy Systems 26

Coordination of Node and Enterprise Services 27

Coordination of Internal Components 28

Security and Management 29

Enterprise Security 31

Cryptography 32

Integrity 34

Computing Infrastructure Integrity 38

Network Infrastructure Integrity 41

User Environment Integrity 49

Data, Application and Service Integrity 50

Identity Management 51

Public Key Infrastructure 52

Authorization and Access Control 54

Confidentiality 56

Black Core 57

Network Information Assurance 58

Trusted Guards 59
Enterprise Management 60
Node Computing Infrastructure 69
Virtual Machines 70
Web Client Platform 71
Browser 72
Common Access Card (CAC) Reader 73
Web Infrastructure 74
Domain Directories 76
Instrumentation for Metrics 77
Time-Critical Operations 78
Remote Management 83
Host Information Assurance 85
User Environment 86
Browser 72
Remote KVM Switch Connectivity 88
Processes 90
Services 92
Core Enterprise Services (CES) 95
Overarching CES Issues 97
CES Definitions and Status 98
CES and Intermittent Availability 99
Cross-Domain Interoperation 100
Net-Ready Key Performance Parameter (NR-KPP) 101
Information Assurance (IA) 102
Net-Centric Operations and Warfare Reference Model (NCOW RM) 103
Key Interface Profile (KIP) 104
Integrated Architectures 105
NCES Directory Services 106
Service Discovery 107
NCES Federated Search 108
Collaboration Services 109

Text Conferencing 110
Service Enablers 112
Service Discovery 107
Information Exchange Patterns 113
Service Optimization and Scalability 114
Utility Services 116
Node Data Strategy 118
Node Transport 120
Physical and Data Link Layers 121
Network Layer 123
Internet Protocol (IP) 124
IPv4 to IPv6 Transition 126
IP Routing and Routers 128
Integration of Non-IP Transports 131
Transport Layer 132
Subnets and Overlay Networks 133
Broadcast, Multicast, and Anycast 135
Virtual Private Networks (VPN) 136
Ad Hoc Networks 137
Network Services 138
Domain Name System (DNS) 140
Dynamic Host Configuration Protocol (DHCP) 142
Network Time Service 145
Application Layer Protocols 147
Mobility 149
Traffic Management 151
Planning Network Services 152
Architectural Approaches to Traffic Management 153
Traffic Engineering 155
Guidance and Best Practice Details 171
Glossary 322
References 355

P1117: NESI Executive Summary

Net-Centric Enterprise Solutions for Interoperability (***NESI***) provides actionable guidance for acquiring net-centric solutions that meet DoD **Network Centric Warfare** goals. The concepts in various directives, policies and mandates, such as those included in the References section of this perspective, are the basis of NESI guidance. The NESI *Net-Centric Implementation* documentation does the following: addresses architecture, design and implementation; provides compliance checklists; and includes a collaboration environment with a repository.

NESI is a body of architectural and engineering knowledge that helps guide the design, implementation, maintenance, evolution, and use of **Information Technology** (**IT**) in net-centric solutions for military application. NESI provides specific technical recommendations that a DoD organization can use as references. NESI serves in many areas as a reference set of compliant instantiations of DoD directives, policies and mandates.

NESI is derived from a studied examination of enterprise-level needs and from the collective practical experience of recent and on-going program-level implementations. NESI is based on current and emergent technologies and describes the practical experience of system developers within the context of a minimal top-down technical framework. NESI guidance strives to be consistent with commercial best practices in the area of enterprise computing and IT.

NESI applies to all phases of the acquisition process as defined in DoD Directive 5000.1 [R1164] and DoD Instruction 5000.2; [R1165] NESI provides explicit guidance for implementing net-centricity in new acquisitions and for migrating legacy systems to greater degrees of net-centricity.

NESI subsumes a number of references and directives; in particular, the Air Force C2 Enterprise Technical Reference Architecture (C2ERA) and the Navy Reusable Applications Integration and Development Standards (RAPIDS). Initial authority for NESI is per the Memorandum of Agreement between Commander, Space and Naval Warfare Systems Command (SPAWAR); Navy Program Executive Officer, C4I & Space (now PEO C4I); and the United States Air Force Electronic Systems Center (ESC), dated 22 December 2003, Subject: Cooperation Agreement for Net-Centric Solutions for Interoperability (NESI). The Defense Information Systems Agency (DISA) formally joined the NESI effort in 2006.

Perspectives	NESI ***Perspectives*** describe a topic and encompass related, more specific Perspectives or encapsulate a set of Guidance and Best Practice details, Examples, References, and Glossary entries that pertain to the topic.
Guidance	NESI ***Guidance*** is in the form of atomic, succinct, absolute and definitive Statements related to one or more Perspectives. Each Guidance Statement is linked to Guidance Details which provide Rationale, relationships with other Guidance or Best Practices, and Evaluation Criteria with one or more Tests, Procedures and Examples which facilitate validation of using the Guidance through observation, measurement or other means. Guidance Statements are intended to be binding in nature, especially if used as part of a Statement of Work (SOW) or performance specification.
Best Practices	NESI ***Best Practices*** are advisory in nature to assist program or project managers and personnel. Best Practice Details can have all the same parts as NESI Guidance. The use of NESI Best Practices are at the discretion of the program or project manager.
Examples	NESI ***Examples*** illustrate key aspects of Perspectives, Guidance, or Best Practices.
Glossary	NESI ***Glossary*** entries provide terms, acronyms, and definitions used in the context of NESI Perspectives, Guidance and Best Practices.
References	NESI ***References*** identify directives, instructions, books, Web sites, and other sources of information useful for planning or execution.

Releasability Statement

NESI *Net-Centric Implementation* v3.2 is cleared for public release by competent authority in accordance with DoD Directive 5230.9; [R1232] *Distribution Statement A: Approved for public release; distribution is unlimited* applies to the documentation set. Obtain electronic copies of this document at http://nesipublic.spawar.navy.mil.

Vendor Neutrality

NESI documentation sometimes refers to specific vendors and their products in the context of examples and lists. However, NESI is vendor-neutral. Mentioning a vendor or product is not intended as an endorsement, nor is a lack of mention intended as a lack of endorsement. Code examples typically use open-source products since NESI is built on the open-source philosophy. NESI accepts inputs from multiple sources so the examples tend to reflect contributor preferences. Any products described in examples are not necessarily the best choice for every circumstance. Users are encouraged to analyze specific project requirements and choose tools accordingly. There is no need to obtain, or ask contractors to obtain, the tools that appear as examples in this guide. Any lists of products or vendors are intended only as examples, not as a list of recommended or mandated options.

Disclaimer

Every effort has been made to make NESI documentation as complete and accurate as possible. Even with frequent updates, this documentation may not always immediately reflect the latest technology or guidance. Also, references and links to external material are as accurate as possible; however, they are subject to change or may have additional access requirements such as Public Key Infrastructure (PKI) certificates, Common Access Card (CAC) for user identification, and user account registration.

Contributions and Comments

NESI is an open project that involves the entire development community. Anyone is welcome to contribute comments, corrections, or relevant knowledge to the guides via the Change Request tab on the NESI Public site, http://nesipublic.spawar.navy.mil, or via the following email address: nesi@spawar.navy.mil.

P1130: Part 4: Node Guidance

Part 4: Node Guidance is the fourth of six parts of the *NESI Net-Centric Implementation* documentation set. Part 4 provides a group of ***Perspectives*** which organize and present Node information, encapsulating pertinent Guidance and Best Practices. For more complete introductory information see the NESI Executive Summary [P1117] perspective and Part 1: Overview [P1286].

A **Node** is a collection of **components** (i.e., **systems**, **applications**, **services** and other Nodes) which results from the alignment of organizations, technologies, process, or capabilities. Potential alignment attributes include operational environment, management, acquisition, mission, technological, sustainment, spatial, or temporal. A Node enables a common strategy for realizing net-centricity and interoperability. A Node can represent an abstract concept of possibly ill-defined size (e.g., a type or class) as well as a more concrete concept (e.g., a specific ship or aircraft) with a defined set of components.

> ***Note:*** *The use of the capitalized term* ***Node*** *in NESI Part 4, alone or preceded by the term* ***NESI*** *(i.e.,* ***NESI Node****) differentiates the specific usage as defined in this perspective from the more general term node. A Node might be nested; such cases would likely introduce additional complexities that would require extra management attention and coordination.*

Nodes presumably are managed actively. Either the Node or a component within the Node (i.e., a system that is acting as executive agent for a capability) can provide the shared capabilities necessary to support net-centric interoperability. The Node Decomposition [P1343] perspective is useful in identifying the shared capabilities the Node manages.

Nodes and components may combine to create a composite capability that is more flexible and agile; more necessary or appropriate components, services or Nodes may replace unnecessary or inappropriate ones. Factors such as physical environments and employment concepts directly influence the scope of a Node, and boundaries can vary widely.

> ***Note:*** *Consider, as a notional example, whether to categorize an individual soldier as a Node. While soldiers are increasingly outfitted with sensors and computing devices, it is unlikely (in the near term) that an individual soldier could host the requisite capabilities needed to ensure compliance with, for instance, the DoD* ***Information Assurance*** *(****IA****) Strategy including intrusion detection,* ***firewalls****, and such. Rather, a collection of soldiers such as an infantry battalion would be connected to a field command center that provides the requisite infrastructure. This does not preclude an individual soldier from being directly addressable on the* ***Global Information Grid*** *(****GIG****), able to conduct information exchanges on a global scale. It simply means that requisite infrastructure is likely to be shared with others rather than isolated to an individual soldier. Likewise, nothing precludes a soldier from being a full Node should technology enable the soldier to carry all the requisite infrastructure elements.*

Node Interaction Patterns

A Node appears programmatically as a set of common capabilities, and aligned budgets and schedules shared among all its components. A Node appears technically as a set of bound, modular architectural components that a complex of **structured identifiers** references. Every asset, resource, data, service or application hosted in a Node can be referenced through identifiers and bindings. Exactly which interoperable formats and protocols apply depends on the Node interactions. When deployed, there are three models that describe Node interactions throughout the enterprise.

- ***intraNode*** - interactions between components within a Node. Interoperability and usage agreements are a local Node matter; these agreements do not necessarily involve open standards and are resolved within the program or within the relevant family of programs by aligning their contracts or by the mission Combatant Command (COCOM) aligning the identifiers and binding configurations in the field.

- ***interNode*** - interactions between Nodes. InterNode interactions rely on inter-operation between their underlying infrastructures as well as compatible mission data formats and service interfaces. Interoperability requires a shared set of open standards and, potentially, intermediate gateways and their services such as a Transport router or PKI Certificate Authority server. Compatible mission interactions may require gateways to provide functionality ranging from simple translation through complex security related filtering. Per guidance from the Office of Management and Budget (OMB) and DoD [R1181] to use net-centric, open international standards in conjunction with the top-level standards themselves, selection of net-centric standards for a Node is the delegated responsibility of the relevant family of programs. Subsequent application of the dynamic,

operational aspects of these standards in the field is the delegated responsibility of the Node COCOM for the mission, again in accordance with established international standards and DoD policies. Candidate standards that meet requirements and are from the **DISR**-approved open, international set should have preference to enhance long-term sustainability and coalition interoperability.

- ***extraNode*** - interactions between a Node and other entities in the ***GIGSpace*** or among non-Nodal entities in the GIGSpace (the GIGSpace covers other things that are not Nodal in nature but part of and important to the GIG, such as e-mail services, Voice over IP services, and other common but externally developed services). ExtraNode interactions require infrastructure interoperable with the larger GIG infrastructure. Accomplish this through DISR-approved open, international standards (i.e., IPv6 for Transport, XML for Data, etc.) and intermediate gateway systems and services. These interactions are completely dictated by the policies and mandates of net-centricity with the GIG.

Detailed Perspectives

The following perspectives present a detailed discussion of NESI Node guidance. In cases which may interconnect with the larger GIG, content is consistent with the DISA GIG interoperability guidance and profiles.

- Node Decomposition [P1343]
- General Responsibilities [P1131]
- Security and Management [P1331]
- Node Computing Infrastructure [P1153]
- User Environment [P1341]
- Processes [P1342]
- Services [P1164]
- Node Data Strategy [P1329]
- Node Transport [P1138]

The guidance and best practices in these perspectives is primarily for those in a position to influence decisions regarding infrastructure and services provided by the Node for shared use by the systems within the Node. With respect to the GIG, the principal question addressed is how should a Node implement the shared infrastructure necessary to achieve the DoD vision of broad integration and interoperability across the GIG, on behalf of systems within the Node, and in accordance with DoD policy and direction?

The guidance associated with these perspectives is applicable to information systems, such as those for command and control or intelligence. It may also be applicable, in part or whole, to other classes of systems or variants, such as embedded, real-time or tactical edge systems.

The guidance also considers multiple operating environments including but not limited to fixed, deployed, mobile air/land/sea Nodes or other instance-specific implementations. Characterizations of those environments, along with the analysis of pertinent use cases for the Node's intended missions, are key tools in the correct selection and application of guidance in the framework (see the Node Operational Environments [P1345] perspective.)

P1343: Node Decomposition

Node decomposition helps key program personnel (including managers, architects and engineers) to map program requirements to architectural infrastructure elements, operational environments, and requirements for net-centricity and interoperability. This activity helps identify relevant guidance and best practices to enable net-centricity and interoperability. Node decomposition also helps identify key interfaces and standards that tie together the architecture. The products resulting from this decomposition help guide program personnel to their particular area of focus and to the corresponding detailed technical guidance. Node decomposition also can contribute to a program's capability or requirements traceability matrix and facilitate work breakdown.

Mission capabilities of a Node are composed of **components** and **services** that support specific operational capabilities. Node decomposition helps identify those services that the Node provides and those that the **Global Information Grid** (**GIG**) infrastructure provides. Node decomposition helps organize the parts of a Node's architecture for which the Node's program managers, system engineers and their program partners have responsibility, and the external GIG infrastructure services for which DISA has responsibility (see the Coordination of Node and Enterprise Services [P1136] perspective and the *Defense Information Systems Network [DISN]: Policy and Responsibilities*, CJCSI 6211.02).

Program managers and architects must ensure not only integration and interoperability within a Node and between Nodes, but also between Nodes and the enterprise-level GIG infrastructure services. Ensuring specifiable, measurable, and testable infrastructure interoperability is a prerequisite first to proper Node composition and subsequently to proper **Enterprise** composition.

Key program personnel must determine shared services and attributes as well as which interfaces are internal and which are external to the Node. Internal interfaces with interoperability requirements generally are those related to Node composition of the architectural categories, especially Enterprise Management [P1330] and Enterprise Security [P1332]. These last two have small components (agents and security controls) embedded in each architectural category that interoperates with overall Node Management and Security systems. Interfaces require interoperability testing early in the life cycle to reduce integration risk and provide regression baselines.

Node decomposition will help identify and select the relevant interfaces and operational attributes for the target Node. The decomposition enables program managers and architects to identify intra-, inter- and extraNode interfaces based on operational environment characterizations during requirement analyses, use case analyses, and other systems engineering activities. These characterizations help with drilling down and identifying appropriate NESI guidance. Decomposition helps organize responsibilities for the development of components in a Node's architecture. Some of these components are the responsibility of the Node's program managers and system engineers, some are the responsibility of partner programs, and some components may have shared responsibility. Finally, some are external GIG infrastructure services for which DISA has responsibility.

This approach supports the convergence of interoperability solutions both within and across Nodes without unnecessarily constraining the Node's program and its architects' ability to address particular circumstances. Node decomposition is based on and consistent with other similar DoD and industry efforts. It integrates the various frameworks for the architectural categories, such as the **Internet Engineering Task Force** (**IETF**) **TCP/IP** transport/networking frameworks, the XML data framework, the various frameworks for computing infrastructure and the **Public Key Infrastructure** security framework.

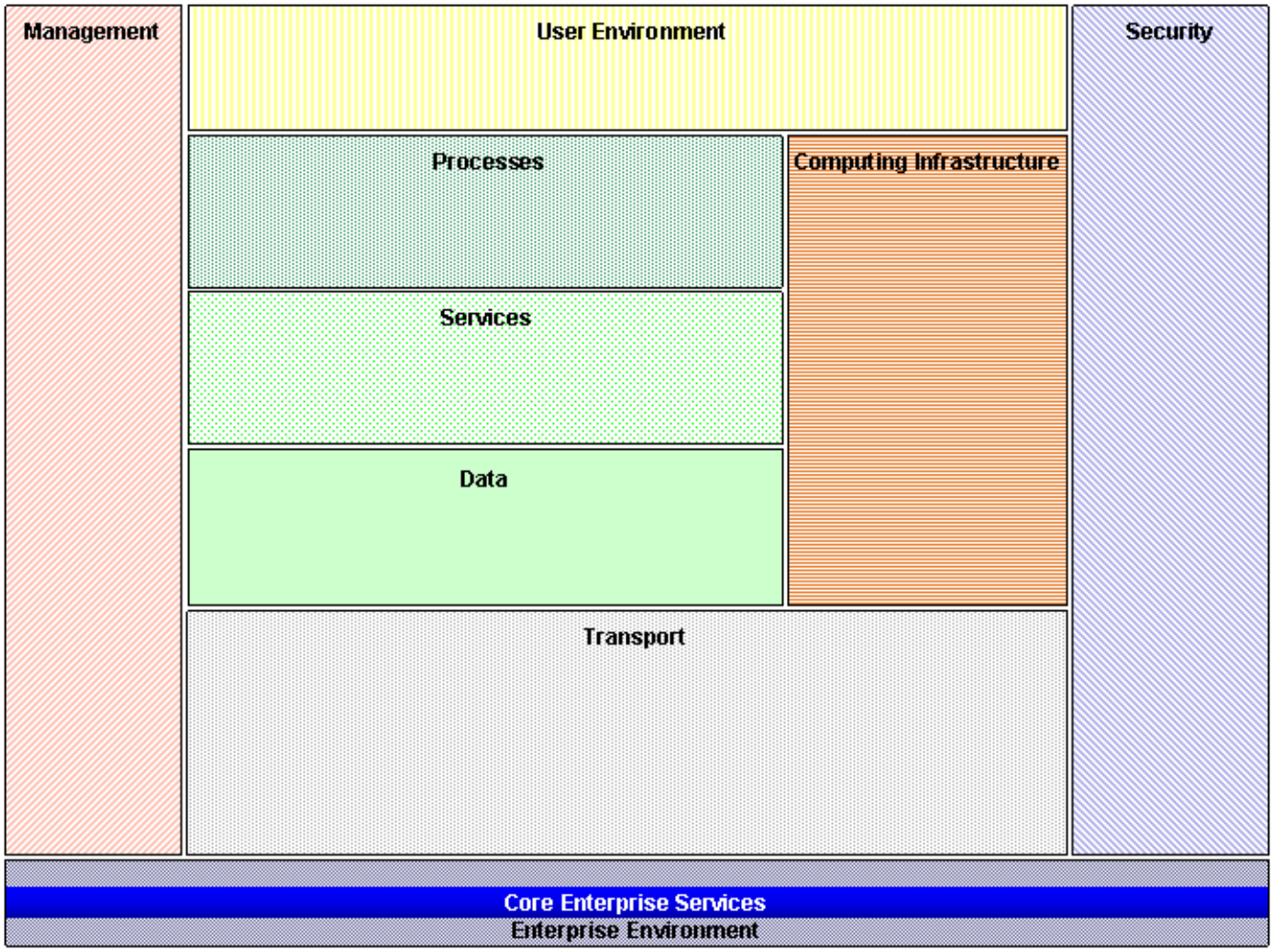
Detailed Perspectives

The two main subsections of Node decomposition cover the key concepts necessary to characterize a Node:

- Node Architectural Elements [P1344]
- Node Operational Environments [P1345]

P1344: Node Architectural Elements

Designing and constructing a **Node** can include an iterative process of functionally decomposing the Node into subordinate services and components, an iterative process of selecting and assembling services and components into larger architectural constructs, or a combination of the two. The approach of assembling services and components becomes especially important for **Enterprises** which are dynamic and adaptable (i.e., require replacing, augmenting or upgrading modules in an iterative manner). The following image (I1228: *Node Decomposition Categories*) shows a set of Node architectural categories which equate to principal sections of the *Part 4: Node Guidance* set of perspectives.



I1228: Node Decomposition Categories

The NESI Node decomposition process separates the architectural categories into subordinate elements as the subsections and additional decomposition images in this perspective illustrate.

The architectural categories are also the subject of additional Part 4 perspectives containing relevant content, guidance, standards, and architectural elements. In some cases, these architectural categories and elements highlight potential shared capabilities within a Node. The exact selection of architectural elements may vary depending on the target Node's system requirements.

> ***Note:*** *An infrastructure capability, in general terms, is a Node architectural construct that multiple consumers share. Sharing utility resources (including facilities) necessary for multiple missions or that are otherwise scarce or under-utilized enhances operational efficiency. Ensuring multiple resources have consistent protection through a uniformly assured infrastructure reduces potential security gaps.*
> *Shared infrastructure implies that there are common interfaces among all the users of that infrastructure. Such common interfaces are standardized at least as de facto standards within that user community. Shared*

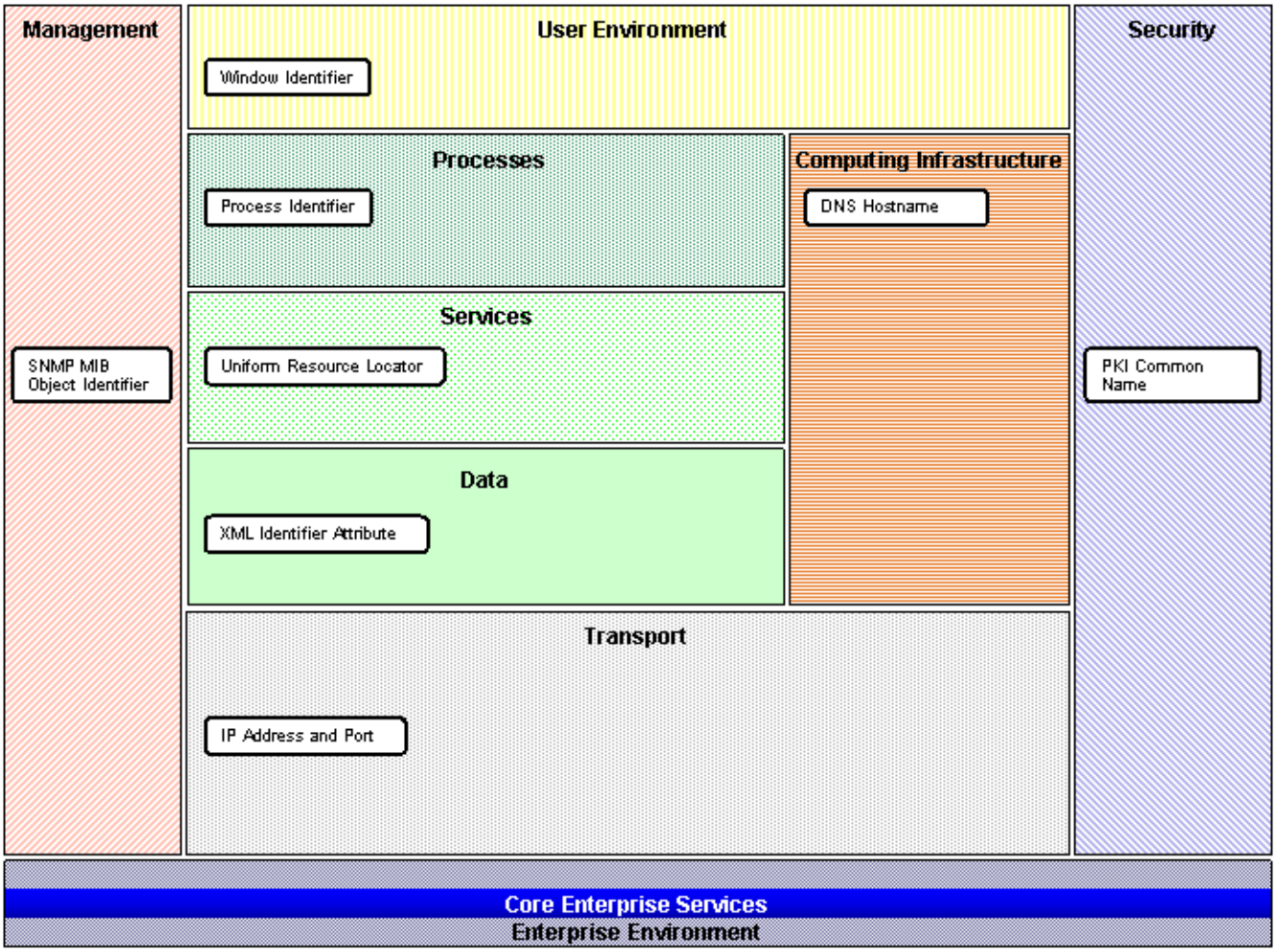
infrastructure resources require policies that deconflict, optimize, specialize and otherwise mediate among its consumers. Performance or security considerations generally drive sharing policies.

Guidance in the Part 4: Node Guidance [P1130] perspectives is aligned with the Node decomposition categories. Mapping the Node decomposition to Node design in the context of operational environment characteristics can result in a function matrix that can point to relevant guidance and best practices for implementing interoperable Node shared capabilities. Generally, the service and transport environments provide the key functional components necessary to adapt capabilities across the various operational environments.

The following subsections contain brief descriptions of the structured identifiers, bindings and architectural categories with links to more in-depth perspectives and guidance.

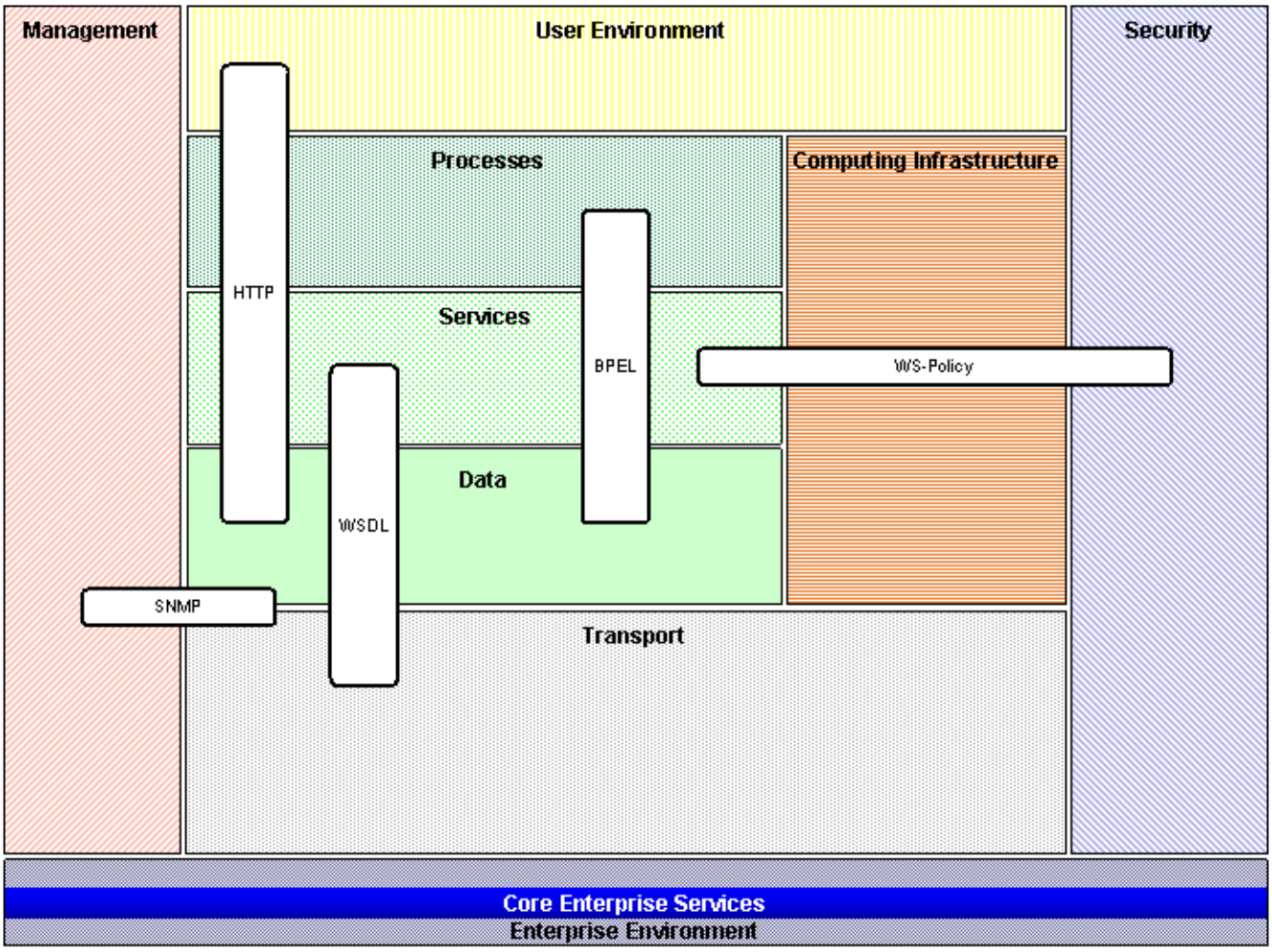
Structured Identifiers and Bindings

Structured identifiers provide a standards-based method of identifying endpoints within an architecture. Components deployed into one or more decomposition categories are addressable by their associated structured identifiers. These identifiers often specify the type and location of the resource. Open standards help to "bind" or connect components within and across decomposition categories. Example structured identifiers for various Node decomposition categories are in the small boxes in the following diagram (I1238: *Example Structured Identifiers*).



I1238: Example Structured Identifiers

Each of a Node's architectural categories has an infrastructure, and constructing a Node includes connecting these functional infrastructure components together with standardized bindings as the figure below (I1237: *Example Bindings*) shows. Open, standardized bindings ensure interoperability of Node components, enabling the integration or binding of alternative or replacement modules with a minimum of effort.

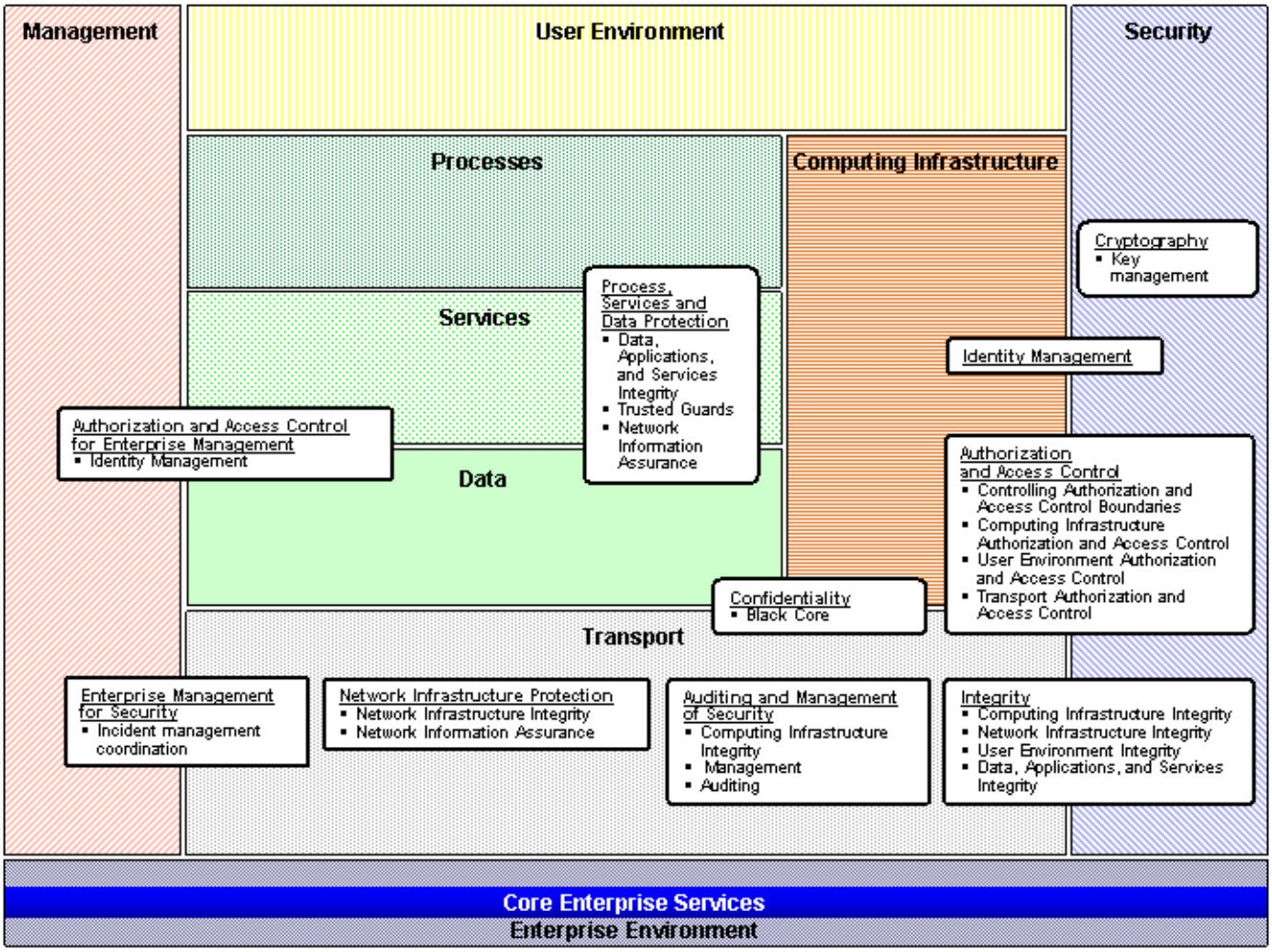


I1237: Example Bindings

Security and Management

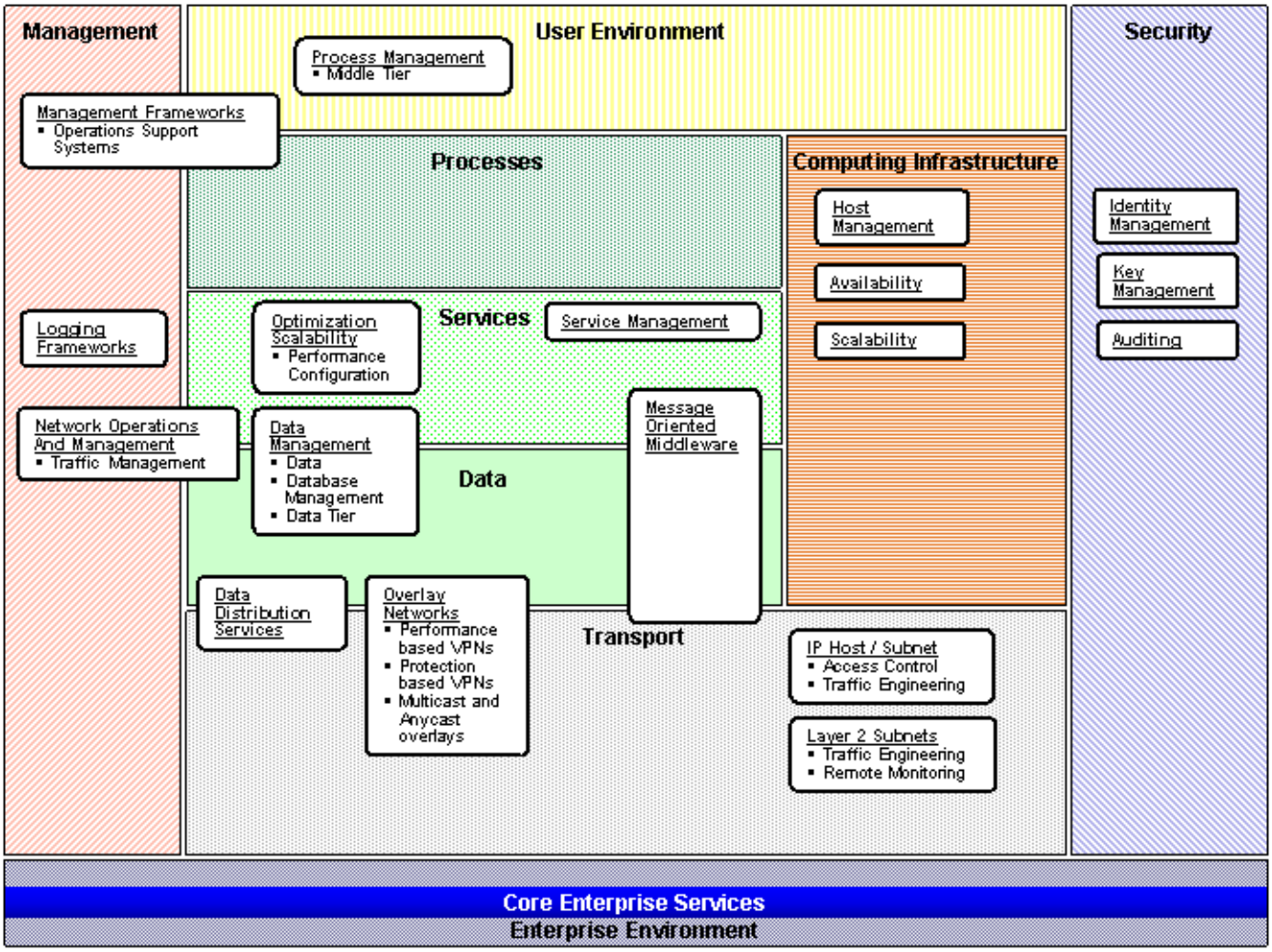
Architectural elements that deal with Security and Management fall in two classes: enterprise management, and management sub-components embedded in the architectural elements, such as transport management agents or computing infrastructure security control devices. For details on the relationship between security and management see the Security and Management [P1331] perspective.

Enterprise security concentrates on two aspects of protection: the local integration of information assurance into Node components and the larger enterprise security engineering often known as Mission Assurance (illustrated by the architectural elements in I1229: *Enterprise Security*). For details see the Enterprise Security [P1332] perspective.



I1229: Enterprise Security

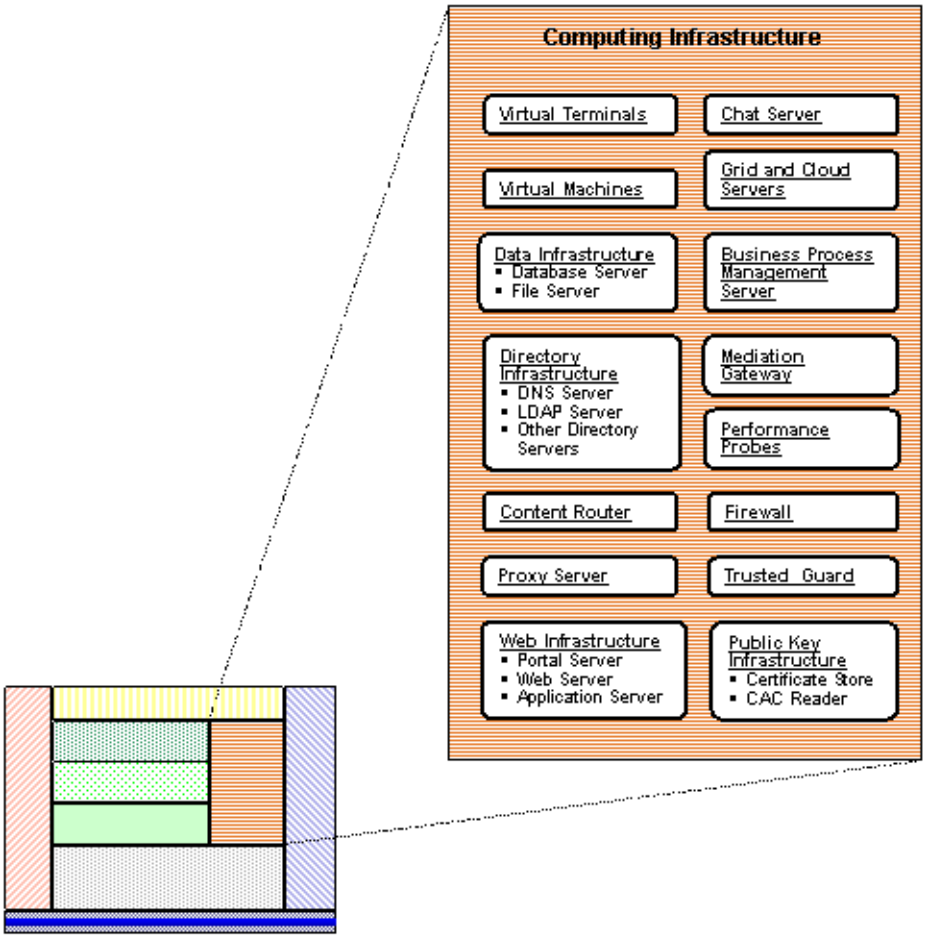
Like Enterprise Security, Enterprise Management concentrates on two aspects of performance: the local integration of management agents and the larger enterprise management systems that coordinate all the Node's components and activities. The small boxes in the Management section of Figure I1230: *Enterprise Management* represent enterprise management tools and operations support systems. Embedding agents that do performance monitoring, configuration, etc., can aid component management integration; the remaining boxes in Figure I1230 represent topics of particular interest in Transport, Computing Infrastructure, etc. For details see the Enterprise Management [P1330] perspective.



I1230: Enterprise Management

Node Computing Infrastructure

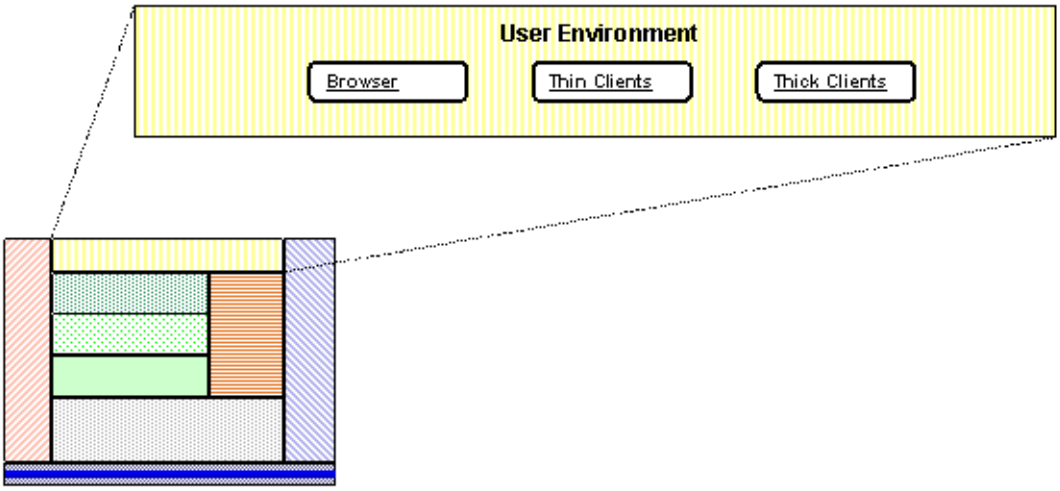
Like security and management, computing infrastructure includes both global and environment-specific classes as the architectural elements in the Computing Infrastructure decomposition illustrate (I231: *Node Computing Infrastructure*). For details see the Node Computing Infrastructure [P1153] perspective.



I1231: Node Computing Infrastructure

User Environment

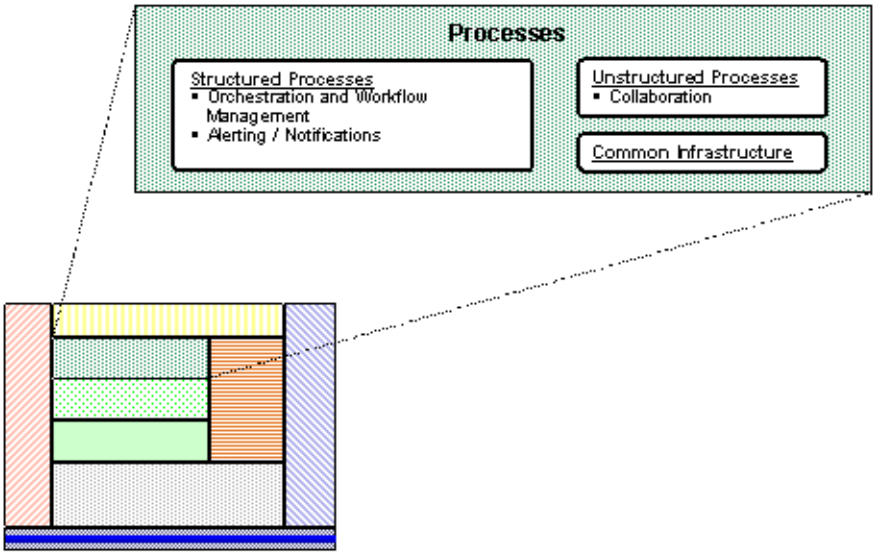
The User Environment comprises those architectural elements that are directly related to handling interaction with the users of the Node. For details see the User Environment [P1341] perspective.



I1232: User Environment

Processes

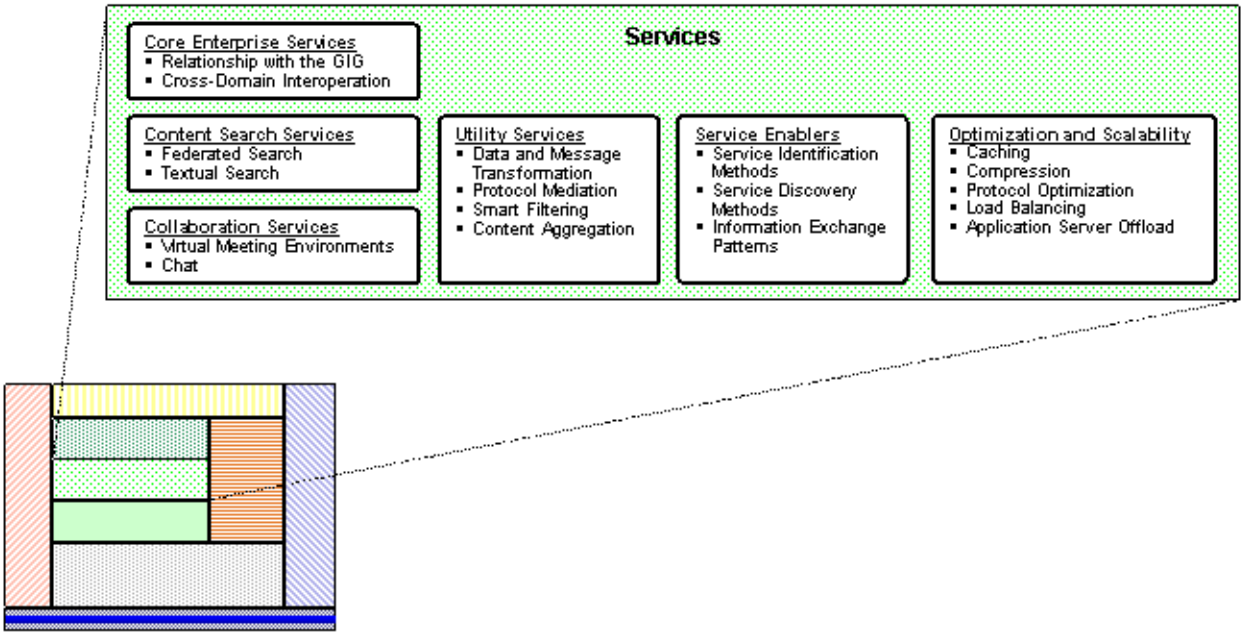
Mission capabilities of a Node usually include both structured and unstructured processes that support specific operational workflows. Nodes applying the **Service-Oriented Architecture** (**SOA**) architectural style to implement mission capabilities do so through the composition of potentially independent and distributed services. Architectural elements in this category provide infrastructure support for composing the necessary elements to execute a Node's mission capabilities. For details see the Processes [P1342] perspective.



I1233: Processes

Services

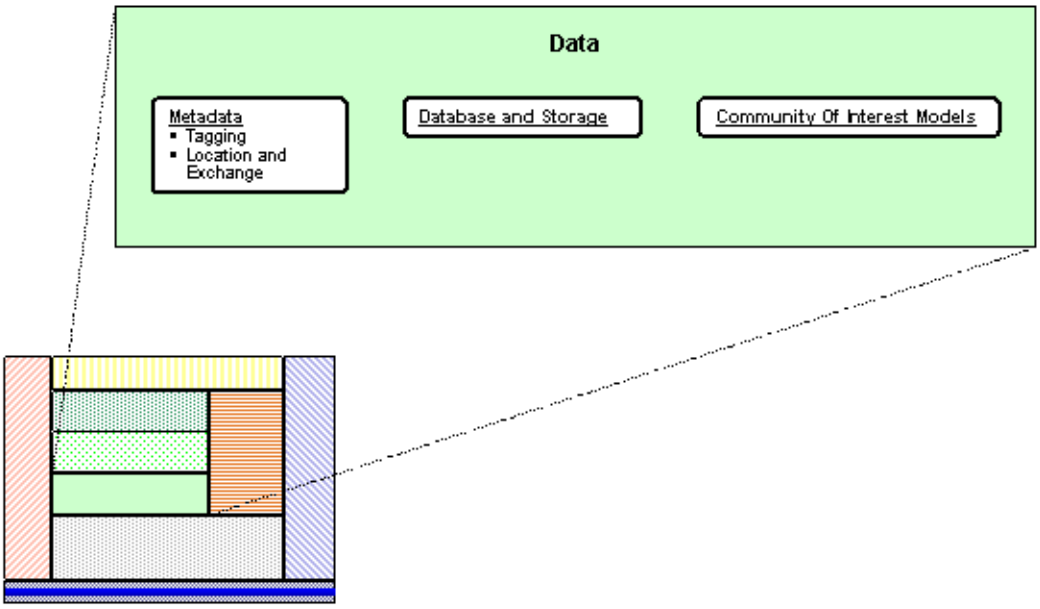
Architectural elements in the Services category provide infrastructure support for developing, hosting and managing modular components used to compose mission capability, including utilization of services hosted outside the Node. For details see the Services [P1164] perspective.



I1234: Services

Data

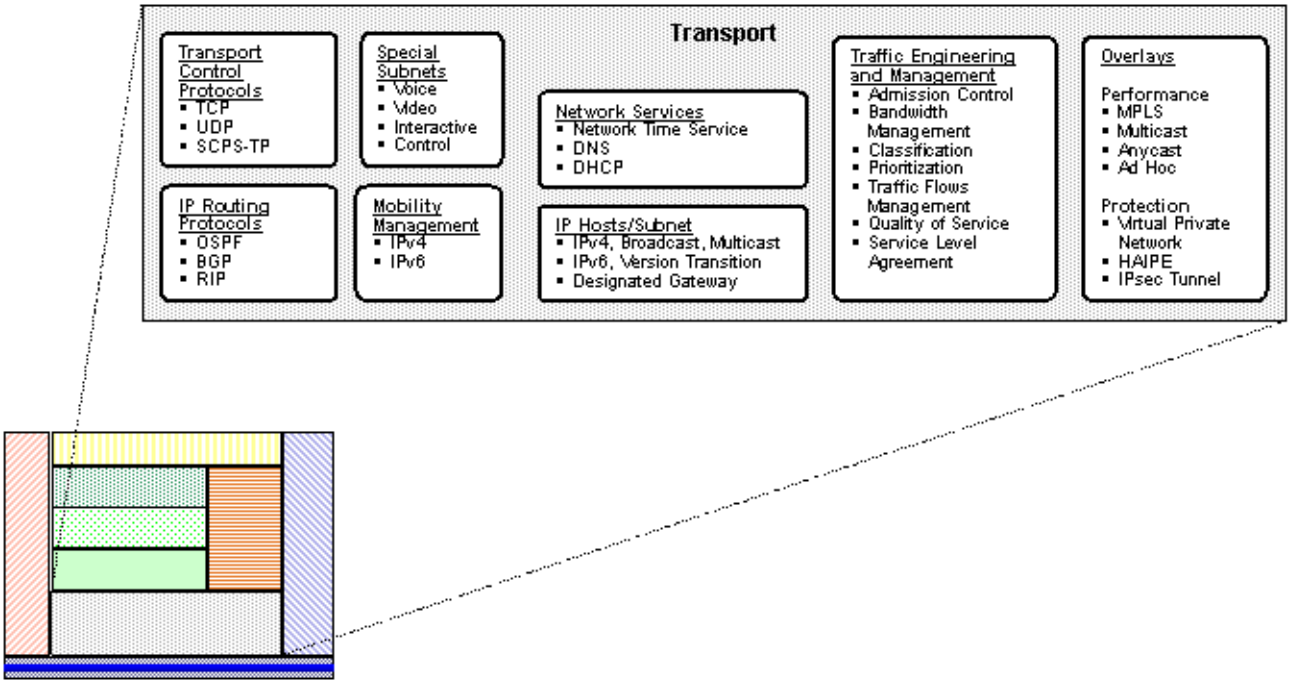
Architectural elements in the Data category provide infrastructure support for the storage, definition and manipulation of data required for fulfilling mission capabilities. For details, see the Node Data Strategy [P1329] perspective.



I1235: Node Data Strategy

Transport

Architectural elements in the Node Transport category provide transport of data both internal and external to the Node. For details, see the Node Transport [P1138] perspective.



I1236: Node Transport

P1345: Node Operating Environments

The characterization of operating environments is a tool to help identify and shape Node boundaries. Along with the analysis of pertinent use cases for the Node's intended missions, these characterizations support Node decomposition. Operating environments are typically in two broad categories: ***Core*** and ***Edge*** (or ***Tactical Edge***). The characterizations help identify the potential target environments for a Node.

While most NESI guidance generally applies across Core Environments, particular constraints presented within edge environments may require tailoring NESI guidance to address these unique constraints.

Defining Criteria

Operational environment categorization can be according to criteria such as network connectivity characteristics, storage, and processor availability which in turn map well to various components in the decomposition. This perspective presents criteria useful for categorization of operational environments with respect to network, system, physical environment, operational, and security. The criteria are intended to define the environment in which the Node operates.

The matrix in the following image (I1226) shows example criteria for the commonly identified types of environments (see also the Core Environments [P1346] and Edge Environments [P1347] perspectives) to help determine how to characterize the operating environment as the first step in decomposing a Node. The value assignments mapping the criteria to particular edge environments are based on collection and analysis of use cases across DoD components.

Example Criteria		GIG Core	Tactical Fixed Center	Tactical Mobile Center	Mobile Platform	Dismounted User
LAN	Connectivity	Well Connected	Well Connected	Well Connected	Intermittently Connected	Mostly Disconnected
	Latency	Low	Low	Low	Medium Low	Medium Low
	Bandwidth	High	High	High	Medium Low	Medium Low
	Reliability	Reliable	Reliable	Reliable	Reliable	Reliable
	Predictablity	Predictable	Predictable	Somewhat Predictable	Somewhat Predictable	Unpredictable
WAN	Connectivity	Well Connected	Well Connected	Well Connected	Intermittently Connected	Mostly Disconnected
	Latency	Low	Low	Medium Low	Medium Low	Virtually Unlimited
	Bandwidth	High	High	Medium Low	Medium Low	Virtually None
	Reliability	Reliable	Reliable	Somewhat Reliable	Unreliable	Unreliable
	Predictablity	Predictable	Predictable	Predictable	Somewhat Predictable	Somewhat Predictable
System	Standard User Interface	Desktop or Laptop	Desktop or Laptop	Desktop or Laptop	Laptop, Tablet or Handlheld	Laptop, Tablet or Handlheld
	Processing	Services or Workstations	Services or Workstations	Services or Workstations	Workstation or Handheld	Workstation or Handheld
	Storage	Large Data Storage Device	Large Data Storage Device	Large Data Storage Device	Single Hard Drives	Single Hard Drives
	Ruggedness	Few Considerations	Few Considerations	Few Considerations	Many Considerations	Many Considerations
	Size	Not Limited	Not Limited	Not Limited	Somewhat Limited	Very Limited
	Weight	Not Limited	Not Limited	Not Limited	Somewhat Limited	Very Limited
	Power	Grid or Macro Generator	Grid or Macro Generator	Generator or Batteries	Generator or Batteries	Batteries
Environment	HVAC	HVAC	HVAC	None	None	None
	Lighting	Controlled	Controlled	Controlled	Variable	Variable
	Hazards	Few	Few	Few	Many	Many
Operational	Reparability	Spares Available	Spares Available	Some Spares	No Spares	No Spares
	Decision Timeliness	Somewhat Limited	Somewhat Limited	Somewhat Limited	Very Limited	Very Limited
	Content	Intermediate or Complex	Intermediate or Complex	Intermediate or Complex	Intermediate or Complex	Simplified
	System Training	Extensive to Intermediate	Extensive to Intermediate	Extensive to Intermediate	Intermediate to Minimal	Intermediate to Minimal
Security	Confidentiality	Insider Threat, Packet Sniffers	Insider Threat, Packet Sniffers	Transmission Interception	Transmission Interception	Capture
	Integrity	Viruses	Viruses	Transmission Errors	Transmission Errors	Spoofing
	Availability	Denial of Service	Denial of Service	Denial of Service	Jamming	Capture, Damage

I1226: Example Environment Defining Criteria

Detailed Perspectives

The following perspectives lay out the characteristics of operating environments and further define some of the commonly identified environments:

- Core Environments [P1346]
- Edge Environments [P1347]

P1346: Core Environments

Core Environments are the most advantaged in terms of available resources. A Core Environment typically has a relatively unlimited and continuous power source, relatively no space or weight constraints, and a relatively continuous high-bandwidth network connection. In the case of manned nodes, there also generally are multiple displays for each individual and access to a large shared display. Users may be working in shifts and thereby sharing their workstations asynchronously. User interface issues relevant to this environment include designing for large shared displays, supporting collaboration, and dealing with information overload. There may be other nodes that are not necessarily manned but serve as strategic data centers or redundant backup sites. Furthermore, failover and redundancy capabilities are characteristics of a Core Environment. From a network perspective, a Core Environment is well-connected to the rest of the **Global Information Grid** (**GIG**) and well-supported to execute missions.

While resources within Core Environments are comparable to those of Tactical Fixed Centers, a major difference is that Core Environments generally do not face the same operational risks as do Tactical Fixed Centers. Core environments can often take the available resources and stable environment for granted; Tactical Fixed Centers suffer the risk of losing resources or having other variances in their operating environment due to threats imposed by adversaries.

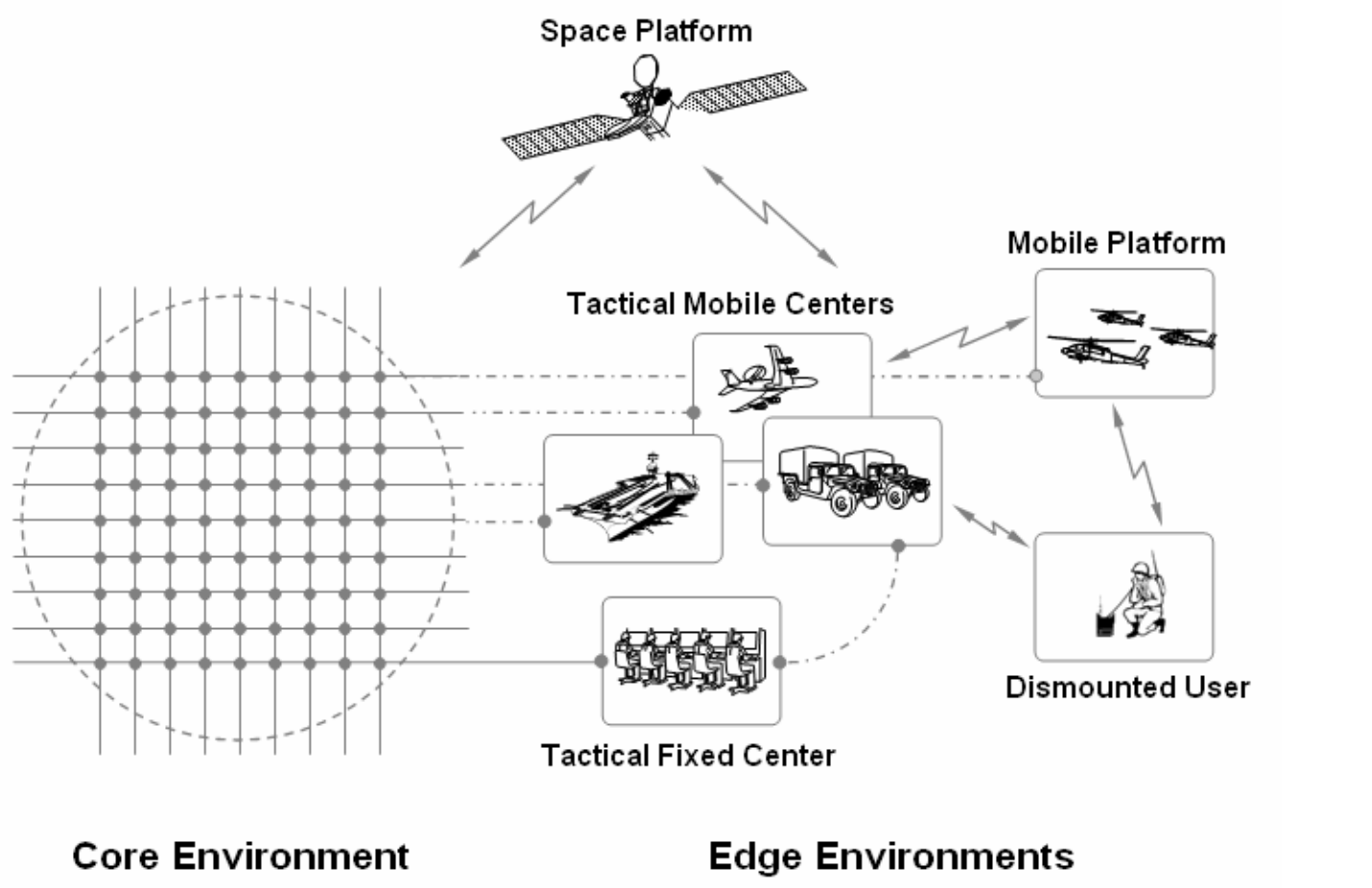
Furthermore, a Core Environment cannot be characterized purely by its physical geographic location. Air Operations Centers (AOCs), for example, have resources as described above. While there may be functional air operations centers within the continental United States, command centers also exist globally. Regardless of the location, both an AOC within the Continental United States (CONUS) and an AOC (e.g., the Combined Air Operations Center at Al Udeid, Qatar) are part of Core Environments as they share the luxuries of plentiful resources and stability of operations.

P1347: Edge Environments

Edge environments are best characterized by a set of environment classes rather than a single broad environment. These classes evolved in recognition of the complexity of the Edge Environments and the need to distinguish tactical users based on the defining criteria.

These classes are not meant to be constraining or be exclusive; these representative set of environments are an initial attempt to bound the problem (i.e., identify a typical set of edge environments to serve as an initial way to organize guidance). Furthermore, systems will not necessarily fall cleanly within only one environment or another. While categorization may provide an initial indication on what system guidance may or may not be applicable, it's likely that use cases will span multiple tactical edge environments.

Examples of Edge Environments are illustrated in I1227 and descriptions follow. Each class, going from a Tactical Fixed Center to a Dismounted User, represents a progressively less-connected and less-supported user. Space Platforms, with their extreme resource limitations, are included in Edge Environments.



I1227: Example Edge Operating Environments

Tactical Fixed Centers

Tactical Fixed Centers are the most advantaged edge environments in terms of the resources available to them. Tactical Fixed Centers are most similar to Core Environments in that they typically have a virtually unlimited and continuous power source, no space or weight constraints, and a continuous high-bandwidth network connection. They also generally have multiple displays for each individual and access to a large shared display. Users may be working in shifts and thereby sharing their workstations asynchronously. User interface issues relevant to this environment include designing for large shared displays, supporting collaboration, and dealing with information overload. These considerations are reflected in the framework by attributes such as hardware display and application content. From a network perspective, Tactical Fixed Centers are well-connected to the rest of the **Global Information Grid** (**GIG**) and well-supported to execute their missions. However, in contrast to the core

environments, Tactical Fixed Centers are often in forward theaters and typically face a greater number of threats and operational risks.

Tactical Mobile Centers

A Tactical Mobile Center could be in a small or large moving vehicle or in a shelter deployed to a particular forward location. If a Tactical Mobile Center is temporarily in a stationary position, network connections tend to be better and more space may be available. In general, Tactical Mobile Centers have more power than Mobile Platform or Dismounted environments, but less than in a Tactical Fixed Center. As with Tactical Fixed Centers, users may be working under controlled lighting, have multiple displays, and have keyboard and mouse inputs, but they may have slightly less decision time to act. However, the complexity of the data and collaboration issues are comparable to Tactical Fixed Centers.

Tactical Mobile Centers may also share characteristics with Mobile Platforms, including interfaces in vehicles, touch screen interfaces, or collaboration. While the internal local area network connections may be quite good, the connection to global wide area networks may have short periods without connectivity, have less bandwidth, and be slightly more latent than the global network connections in Tactical Fixed Centers. Tactical Mobile Centers have connectivity to the rest of the GIG, but it is less reliable. Therefore, applications supporting Mobile Center users would benefit greatly from event recognition and response capability with respect to periods of connectivity to the GIG or local network resources.

Mobile Platforms

Mobile Platform users are typically traveling in a vehicle (air, land, or sea) and operating in high pressure environments where they must make decisions quickly. They are usually connected with other users via ad hoc networks using radios, laptops or rugged touch-screen devices. These users may have varying levels of experience with the system in the field. The information they are able to send and receive may be a bit more complex than the dismounted user. They have advantages over the dismounted user because of the increased display space, the availability of a touch screen keyboard, and the additional vehicle-generated power supply. However, mobile platforms still have limited space, weight carrying capacity, and power and are subject to the environmental constraints of operating within a vehicle. These constraints are often underestimated. Mobile Platform users may have very limited connectivity to the GIG, but have reliable connectivity between elements on the local area network. The Tactical Mobile Center, with which Mobile Platform connectivity may also be intermittent, will likely serve as a proxy to the GIG. As with the Tactical Mobile Center, applications supporting Mobile Platform users would also benefit from event recognition and response capability with respect to periods of connectivity to the GIG or local network resources.

Dismounted Users

Dismounted Users are in the most disadvantaged of environments. They are typically traveling on foot so they must be able to carry the complete system including the power supply which is often extremely limited. These users may spend a great deal of time disconnected from the network. When connected, the network connection may have high latency and limited bandwidth. These users operate in pressure filled environments where decisions must be made in minutes or even seconds. The unpredictable and dynamic nature of the environment in which Dismounted Users operate may lead to unexpected system failure. These users tend to employ the most efficient means possible to obtain and provide information. A Dismounted User typically has a radio and may have access to a cell phone or personal digital assistant (PDA) device with a small visual display. Dismounted Users may only have connectivity to other local users and little or no connectivity to the rest of the GIG.

Space Platforms

Space platforms are extremely resource limited; size, weight and power (SWP) are at a premium. Space platforms are not conducive to major growth or changes because of access limitations once in space. Communications vary significantly depending on the application. Mission support to communications can vary and provide broad bandwidths while standard Command, Control and Telemetry communications are generally much more limited. Architectures and designs of space platforms focus on efficient use of resources. Generally there are no direct human interfaces on the platform and controls are from mission ground stations.

P1131: General Responsibilities

In addition to the specific requirements of a Node to support transport, common computing infrastructure, **Enterprise Services** and **Community of Interest (COI) Services** there are some general responsibilities that a Node must support in order to ensure that the final product can interact with the rest of the **Global Information Grid** (**GIG**). The responsibilities include the following:

- Nodes as Stakeholders [P1132]
- Net-Centric Information Engineering [P1133]
- Internal Component Environment [P1134]
- Integration of Legacy Systems [P1135]
- Coordination with External Enterprise [P1136]
- Coordination of Internal Components [P1137]

P1132: Nodes as Stakeholders

Formally represent a **Node** as a **stakeholder** in the acquisition and evolutionary activities of all the **Components** the Node will host. A Node's Component composition will change over time; maintain and identify all the known Components throughout the lifecycle of the Node. This action is fundamental to the provisioning of a shared infrastructure and the avoidance of functional duplication within the Node.

The necessity of a Node involvement as a stakeholder in its Components may not be obvious; it has a bearing on **Global Information Grid** (**GIG**) interoperability. Component independent planning and evolution is likely to result in the external exposure of inconsistencies or, worse, incomplete, inaccurate, or misunderstood data. Consider two systems within the Node that both ingest a particular type of data, but process it at different levels of fidelity, and are independently intending to publish the result to the rest of the GIG. This is an example of when a Node manager would want to work across the systems to ensure that the Node presents its collective capability clearly.

Guidance

- G1569: Maintain a comprehensive list of all of the **Components** that are part of the Node.
- G1570: Assume an active management role among the **Components** within the Node.

P1133: Net-Centric Information Engineering

Of particular concern for **Global Information Grid** (**GIG**) interoperability is the information contained in inter-nodal information exchanges. Information exchanges are typically the purview of the systems within the Node, rather than the Node itself, and the details are worked out by a **Community of Interest** (**COI**). But the Node infrastructure must be engineered to support information exchanges between various COIs. The COIs can require any number of Components to fulfill the mission. When a Component wishes to make its data available to the **enterprise**, there are different enterprise design patterns the Component can use. For example, the mechanism selected by a Component to exchange information may be publish-subscribe, broker, or client server. The Node infrastructure must support whichever enterprise design pattern mechanism is selected. Consequently, the Node has a stake in the Component design. Additionally, the Node has a stake in performance specifications provided in the **Service Level Agreements** (**SLA**). The Node must support the SLA contract with the Node's infrastructure.

Node management should designate COI representatives to track, advocate, and engineer information exchanges in support of the **DoD Net-Centric Data Strategy**. According to this strategy, "COI is the inclusive term used to describe collaborative groups of users who must exchange information in pursuit of their shared goals, interests, missions, or business processes and who therefore must have shared vocabulary for the information they exchange." The principal mechanism for recording COI agreements is the **DoD Metadata Registry** required by the DoD CIO *DoD Net-Centric Data Management Strategy: Metadata Registration* memo. There are registry implementations on the **Unclassified but Sensitive Internet Protocol Router Network** (**NIPRNet**), **Secret Internet Protocol Router Network** (**SIPRNet**), and **Joint Worldwide Intelligence Communications System** (**JWICS**).

The DoD Metadata Registry Web site (http://metadata.dod.mil) provides a search capability; there is also a **SOAP**-based interface to the Registry.

Guidance

- G1571: Maintain a comprehensive list of all the **Communities of Interest** (**COIs**) to which the **Components** of a Node belong.
- G1572: Include the Node as a party to any **Service Level Agreements** (**SLAs**) signed by any of the **components** of the Node.
- G1573: Define the enterprise design patterns that a Node supports.
- G1574: Define which enterprise design patterns a **Component** requires.
- G1575: Designate Node representatives to relevant **Communities of Interest** (**COIs**) in which Components of the Node participate.

Best Practices

- BP1865: Provide sufficient program, project, or initiative **metadata** descriptions and automated support to enable **mediation** and translation of the data between **interfaces**.
- BP1866: Coordinate with end users to develop interoperable materiel in support of high-value mission capability.

P1134: Internal Component Environment

Nodes should provide an environment to support the development, integration, and testing of net-centric capabilities of their components. As Nodes themselves and the components within the Nodes move closer to the implementation of net-centric capabilities, it becomes increasingly important to provide a development, integration, and test environment to support those capabilities. This environment should allow for exercising the Node infrastructure and either hosting services locally within the Node or providing access to **Net-Centric Enterprise Services** (**NCES**). The particulars on how to do this depend on the characteristics of the Node. For example, mobile or deployed Nodes would provide environments substantially different than fixed land-based or permanent Nodes.

Specialized services will likely be hosted locally for Nodes in real-time, dynamic and mobile environments, such as those used for information exchange across the ***Joint Airborne Network***. An emerging trend in the commercial networking/IT industry is to realize high performance capabilities with a combination of hardware-based switches (e.g., **XML router**) and services (e.g., mediation). Commercial industry has experienced significant performance issues while running applications and services on the **Internet**, especially those that are XML-based.

When applicable, developers should be using the NCES piloted **Enterprise Services** offered by **DISA** for development, test, and integration at the earliest opportunity within the Node and component lifecycles. Potential causes of problems include security parameters, network configuration, and product inconsistencies.

Guidance

- G1576: Provide an environment to support the development, build, integration, and test of net-centric capabilities.
- G1577: Maintain an **Enterprise Service** schedule for interim and final **enterprise** capabilities within the Node.
- G1578: Define a schedule for **Components** that includes the use of the **Enterprise Services** defined within the Node's enterprise service schedule.
- G1579: Define which **Enterprise Services** the Node will host locally when the Node becomes operational.
- G1580: Define which **Enterprise Services** will be hosted over the **Global Information Grid** (**GIG**) when the Node becomes operational.

P1135: Integration of Legacy Systems

Nodes might contain systems or **applications** that are in the **Sustainment** lifecycle phase. These **Components** are often referred to as ***legacy*** systems or applications. Changing the internals of such Components to support net-centricity may be impractical and often has little return on investment. Usually, the decisions to brand a system or an application as a **legacy system** is made at a high level in conjunction with the operational user and acquisition communities. When the legacy functionality needs to be exposed as an interim solution internally to a Node or externally to the Node as a **proxy**, this often is accomplished using a service wrapper technique. Refer to ***Migration Patterns*** for a more detailed discussion about the service wrapping and other solutions that could be used to expose the legacy functionality.

Guidance

- G1581: Expose legacy functionality through the use of a service.

P1136: Coordination of Node and Enterprise Services

The **Net-Centric Enterprise Services** (**NCES**) capabilities under definition, development, or in pilot testing are complex and use leading edge technologies. Reflect the status, availability and deployment schedule for services in an integrated master schedule for the **Node** that shows planned dependencies of systems within the Node on these services. Given the rate of evolution and leading edge nature of some services, detail the coordination of efforts, including specific version numbers, workarounds, assumptions, constraints, configuration, and best practices. Note that these practices are applicable for coordination with both external and Node-provided **Enterprise Services**.

Guidance

- G1577: Maintain an **Enterprise Service** schedule for interim and final **enterprise** capabilities within the Node.
- G1578: Define a schedule for **Components** that includes the use of the **Enterprise Services** defined within the Node's enterprise service schedule.
- G1582: In Node **Enterprise Service** schedules, include version numbers of Enterprise Services interfaces being implemented.

Best Practices

- BP1865: Provide sufficient program, project, or initiative **metadata** descriptions and automated support to enable **mediation** and translation of the data between **interfaces**.

P1137: Coordination of Internal Components

The shared infrastructure provided by Nodes, for shared use by its member **components** cannot evolve independently of the components within the Node. Nodes may host a variety of components which may be members of multiple Nodes. Consequently, the development of components is likely to occur with differing timeframes and rates of evolution. This presents a coordination challenge for the Node managers.

Guidance

- G1583: Provide routine **Enterprise Services** schedule updates to every **component** of a Node.

P1331: Security and Management

Enterprise Management and Security are two distinct but closely related topics. Management and security functions provide underlying enablers to assure mission operations have both the performance and the protection they need. Furthermore, management functions must be secure and manageable and security functions must be both manageable and secure.

Security and Management overlap in two ways: functionally, through identity management and accountability, and technologically, through use of discovery and logging services.

The first key concept that ties mission operations, security, and management together is ***identity management***. The primary principle of identity management is the cross-functional composition of the standard identifier of any resource, whether human or machine-based. Consequently, many of the management functions assure performance by assigning identifiers (such as addresses) and the rest of the management functions focus on configuring capacity, authorizing usage, auditing and analyzing operations based on those identifiers. Likewise, many of the security functions assure protection by authenticating resources in order to assign part of the net-centric standard identifier (generally an encryption-based credential) and the rest of the security functions focus on authorizing, checking compliance, auditing and analyzing operations based on those identifiers.

The second key concept that ties mission operations, security and management together is ***accountability***. Accountability is built on both integrated sensors that track activity patterns and the enterprise-wide logging sub-systems that aggregate and roll-up the sensors' notifications, alerts or events. Accounting and performance management audits use performance-based activity patterns to trigger sensors and analyze logs, while security audits use protection-based patterns.

Many of the advantages of net-centric operations, Service-Oriented Architectures, etc., derive from their ability to exploit ***common infrastructure***: the risk-reduction from earlier and more intensive testing, in addition to the potential cost-sharing enabled by development amortized over a larger clientele base. As a result, much of Enterprise Engineering and Enterprise Management is managing the aggregation of diverse mission applications, data and services onto that shared infrastructure. Such aggregation management creates its own constructs, implemented as standardized, **structured identifiers**. For example, related groups of applications, data stores and services not only may share computing infrastructure servers using a common path identifier family within a Node, they generally share an intra-Node Local Area Network (LAN) that interconnects with other Nodes and the GIG as an IP subnet with a common address prefix. Drivers for infrastructure aggregation constructs can be either performance or protection or both. Increasingly, aggregation constructs not only have common computing and transport identifier structures that define the limits of Quality of Service (QoS), they also have a common cryptographic identifier structure used for **Information Assurance** perimeters. Taken together, such aggregation constructs not only help define the Node boundaries, but also the structure of the larger GIG within which the Node must operate.

The Enterprise Security [P1332] perspective concentrates on two aspects of protection: the local integration of information assurance into Node **components** and the larger Enterprise security engineering often known as Mission Assurance that coordinates all the components and activities. There are three main activities:

- ***Preserve Node Integrity and Confidentiality*** with integrated sensors and security controls.
- ***Assuring Security Interoperability*** by verifying that authentication and authorization interactions with security controls are interoperable at the intraNode, interNode and extraNode level.
- ***Accountability*** is provided through security sensor placement, configuration management, logging, and alert notifications.

The first task of Program Managers and Architects for Enterprise Security engineering should be to ensure the integration of security sensors and controls into components for proper boundary hardening. The second is to ensure that these sensors and security controls are interoperable with Node-level and enterprise security sub-systems, especially those used for authentication/authorization and the notification, logging and auditing necessary for accountability.

> ***Note:*** *Logging is the process of recording events and auditing is the process of reviewing events against policy. Ensuring proper logging requires a design that includes a logging infrastructure. Auditing requires more than infrastructure because it is the application of policy. Human review is the policy decision point (PDP) of last resort and therefore the responsibility of a Concept of Operations (CONOPS) and training, not program acquisition. This*

is true even if there are automated PDPs; the authoritative auditing process is still generally completed by a human who verifies the validity of the automated match hits.

The Enterprise Management [P1330] perspective concentrates on two aspects of performance: the local integration of management agents into Node components and the larger Enterprise Management that coordinates all the components and activities. There are three main performance activities:

- tuning node and infrastructure configurations in order to meet operational service level requirements
- assuring management interoperability by testing compliance between management agents and the enterprise management systems
- accountability through usage sensor placement, logging and their configuration.

The first task of Program Managers and Architects for Enterprise Management engineering should be to ensure the incorporation of monitoring sensors and configuration controls into components for proper monitoring and tuning. The second is to ensure that these sensors and configuration controls are interoperable with Node-level and enterprise management sub-systems used for remote management and the notification, logging and auditing necessary for accountability.

Good enterprise management uses the principles of ***Decomposition*** and ***Delegation*** to enable service support and delivery that can handle global scales and global diversity. Decomposition of the enterprise into standard managed modules enables the configuration and change management or release management necessary for supporting a highly diverse portfolio of service modules. Decomposition also enables the monitoring and adjusting of the portfolio's service provisioning and delivery through capacity management, financial analysis, etc.

Detailed Perspectives

- Enterprise Security [P1332]
- Enterprise Management [P1330]

P1332: Enterprise Security

Security is not a single idea, object, or task. The common phrase ***defense in depth*** is very apt in describing how to secure **information technology** (**IT**) environments. While the objective may be to impede adversaries completely, slowing them down is the more likely and practical outcome. Some examples include the following:

- Causing an adversary to expend more resources to accomplish the same task
- Generally creating more exposure to enable better detection and disruption of an adversary's activities

Multiple security boundaries provide protection depth. Some of these boundaries are physical, while others are information-based in nature (e.g., virtual technologies, social processes or extended-trust meta-data). A heterogeneous approach is necessary for everything in a Node that must be protected, in order not to expose a single point of failure. The "weakest link" adage is very applicable to net-centric operational security (OPSEC).

Enterprise Security includes the fundamental core or "capstone" concepts and guidance for Security that are necessary to understand the "Security Considerations" found in the other Node functional environment perspectives. For a further discussion of security concerns regarding accountability, logging and auditing see the Enterprise Management [P1330] perspective.

Detailed Perspectives

- Cryptography [P1333]
- Integrity [P1334]
- Identity Management [P1178]
- Authorization and Access Control [P1339]
- Confidentiality [P1340]
- Network Information Assurance [P1147]
- Trusted Guards [P1150]

Guidance

- G1301: Practice layered security.

P1333: Cryptography

Cryptography is a fundamental technique to support operations security (OPSEC) by enabling the following activities:

Ensuring Integrity (e.g., digital signatures): Digital signatures enable tamper detection and non-repudiation. A digital signature or digital signature scheme is a type of cryptography used to simulate the security properties of a handwritten signature on paper with all the benefits and more. Optionally, include a scanned copy of the written signature for completeness. They cannot be copied or as easily forged. Digital signature schemes normally provide two algorithms, one for signing which involves the user's secret or **private key** (the only key in symmetric schemes), and (in asymmetric schemes) one for verifying signatures which involves the user's **public key**. The output of the signature process is called the "digital signature."

Authenticating identity (e.g., keys) Authentication is the process of attempting to verify the digital identity of the sender of a communication such as a log in request. The sender being authenticated, often referred to as the principal, may be a person using a computer, a hardware device or a computer program. An anonymous credential, in contrast, only weakly establishes identity, together with a constrained right or status of the user or program.

Ensuring confidentiality: Encryption of the payload covers data, signatures, session keys, certificates for integrity, authentication, and authorization information.

Authorization (e.g., X.509 certificates, roles, and accounts): Perform authentication prior to authorization. Authenticated identities, even an anonymous identity, are necessary to perform successful authorization. Authorization grants the level of privileges (authorization) assigned to a particular authenticated identity. In most cases, anonymous or weak authenticated identities should have limited capabilities or level of authorization, such as read-only access to general access resources.

Cryptographic guidance requires a sensitivity/protection/performance trade off analysis. Factors to consider follow:

- shelf life of information (actionable, analysis)
- key and algorithm hardness
- key length and type (symmetry versus asymmetry)
- management procedure attack resistance and resilience
- cryptography overhead impact
- transport path bandwidth-delay product for handshaking and key distribution
- processor speed and memory for encryption/decryption algorithms
- storage space and access speed for encryption/decryption algorithms

Complexity of crypto management is defined by the following:

- key assignment and distribution
- authorization scope (delegation, transitive trust, revocation, etc.)
- accountability
- auditability

Guidance

- G1317: Ensure applications store **Certificates** for subscribers (the owner of the **Public Key** contained in the Certificate) when used in the context of signed and/or encrypted email.
- G1325: Encrypt **symmetric keys** when not in use.
- G1344: Encrypt sensitive data stored in configuration or resource files.
- G1371: Use the **National Institute of Standards and Technology** (**NIST**) ***Digital Signature Standard*** promulgated in the **Federal Information Processing Standards** Publication 186 (**FIPS** Pub 186-3 as of June 2009) for creating **Digital Signatures**.
- G1374: Individually **encrypt** sensitive **message** fragments intended for different intermediaries.

- G1376: Do not **encrypt** message fragments that are required for correct **SOAP** processing.
- G1378: Encrypt communication with **LDAP** repositories.
- G1381: Encrypt sensitive persistent data.

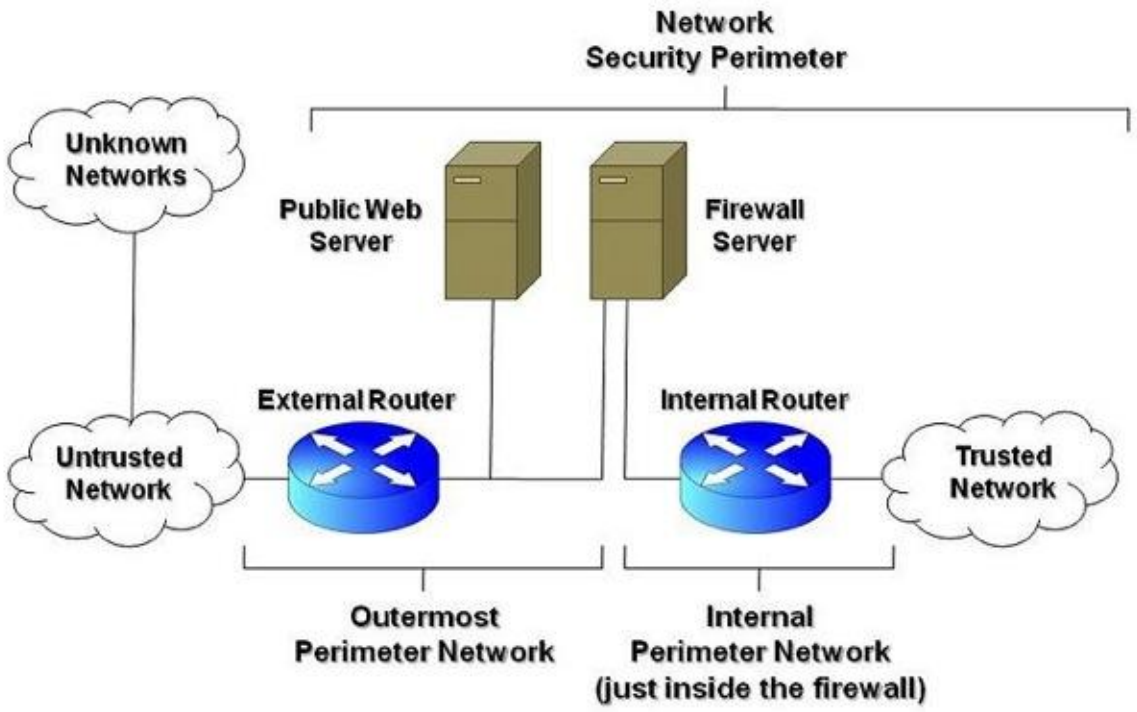
P1334: Integrity

Integrity of an **enterprise** consists of ensuring the overall integrity of its **systems** and the data they contain. External interfaces are the first line of defense, but defense-in-depth may require assurance controls on internal **Node** interfaces as well. A program's Capability Description Document (CDD) initially defines interfaces which the Node's architects formally specify. With proper safeguards and testing, interfaces can act as formal integrity boundaries.

Node and system architects ensure integrity by first specifying hardened boundaries and equipping them with sensors and security controls. Baseline vulnerability assessment information is also helpful. Vulnerability assessments should occur for every boundary interface that exposes and must protect data, applications and **services**. Evaluation of each interface will not only use net-centric metrics to indicate how well they make information available, but also by vulnerability metrics indicating how well they defend information within those boundaries. The following subsections and linked detailed perspectives cover the interface controls and security technologies that current **Information Assurance** (**IA**) guidance requires for each interface boundary. Not only do all boundary interfaces require interface controls, but the subsidiary boundary interfaces major architectural constructs provide require interface controls as well. Examples follow:

- computing infrastructure system boundaries and virtual machine boundaries
- transport network boundaries and subnetwork/overlay network/virtual network boundaries
- user environment boundaries and display or window boundaries
- management domain and sub-domain boundaries
- boundaries defined for the security technologies themselves, including subordinate **Certificate Authorities**
- data and service boundaries, including Web page frames, **applets** and **servlets**

The following diagram (I1239: *Example Two-Perimeter Network Security Design*) is an example of how to identify two such boundaries and their security control components. The diagram shows how to structure subsidiary boundaries in the Transport infrastructure in order to separate Nodes with different IA authorities and policies onto separate **Global Information Grid** (**GIG**) intra-networks, such as those found in joint operations. At the same time, by appropriate placement of transport routers and guards, the two services can interconnect and interoperate to coordinate their joint operations. This architectural structuring, because it is based on open standards, allows each service to select and standup its own implementation of the architecture, with its own security policies, without preventing the interoperable flow of authorized joint coordination information.



I1239: Example Two-Perimeter Network Security Design

Key security concepts are in the following subsections and the linked detailed perspectives. The security activities can serve as guides or templates for a Node's Interface Control Document (ICD), as required by the **Security Technical Implementation Guides** (**STIGs**) and the *DoD Information Assurance Certification and Accreditation Process (DIACAP)*.[R1291] The intent of these activities is to help Node architects and program managers determine the best ways to identify and mitigate weaknesses in Nodes while maintaining net-centric interoperability.

The subsections and the linked detailed perspectives also provide recommendations about how to select and apply the relevant standards and technologies to provide security capabilities. The intent is to mitigate the exposure of weak link systems in Nodes while maintaining interoperability. Certain security activities, techniques and technologies are common to among Node components.

- ***Integrity:*** quality of an Information System (IS) reflecting the logical correctness and reliability of the operating system, the logical completeness of the hardware and software implementing the protection mechanisms, and the consistency of the data structures and occurrence of the stored data; formal security terminology often interprets integrity more narrowly to mean assurance that an entity has not been modified in an unauthorized manner or guarding against improper information modification or destruction and does not require system behavior that meets all operational goals and expectations. Many attacks modify expected behavior without modifying the responsible entity or information.
- ***Defense-in-depth:*** establishes variable barriers across multiple layers and missions of the organization; barriers in net-centric systems are generally in the form of network boundaries and their associated security controls.
- ***Boundary:*** physical or logical perimeter of a system; hardening techniques and technologies assure integrity and define security perimeters thanks to the embedding of security controls as boundary protection that prevents and detects malicious and other unauthorized communications.
- ***Standard vulnerability specifications and scorecards based on them:*** examples include the Common Vulnerability Enumeration (CVE; see http://cve.mitre.org/ or http://en.wikipedia.org/wiki/Common_Vulnerabilities_and_Exposures), the Common [Software] Weakness Enumeration (CWE; see http://

cwe.mitre.org/) and the Open Vulnerability Assessment Language (OVAL; see http://oval.mitre.org/); they help to evaluate the hardness of boundary interfaces, the adequacy of the embedded security sensors or controls, and the effectiveness of the enterprise security engineering policies and support systems.

Security Integration Activities

The following security-based activities integrate security and IA throughout a Node using the above concepts. Each concept has a variety of techniques and technologies, use of which varies according to the functional category and Node operational requirements. The following sections are divided first into the functional categories, and then into the major activities. Specific techniques and technologies for that functional category's security activities are then listed as sub-sub-sections or lists.

Boundary Creation

Boundary creation includes selection of security control technologies to embed in boundary interfaces for baseline integrity protection. The simplest form often does not provide access control, just interoperability and accountability and in military settings is used primarily when physical boundaries and access control are sufficient assurance of Node integrity. When installing or embedding security controls, ensure the target Component is in a state of known integrity, e.g., by booting with known media such as Original Equipment Manufacturer (OEM) media or "gold" disks (referring to a master disk that has known safe status, documented chain of custody media, etc.). Also ensure that the components in question have valid anti-tamper signatures for their storage media, current malware signature files and scanner engines and very recently successfully completed holistic scans. See the Network Infrastructure Integrity [P1336] perspective and the DISA Information Assurance Support Environment (IASE) Security Technical Implementation Guides (STIGS) and Supporting Documents Web site for additional information.

Access Control Integration

Access control integration employs security controls (including, for example, identity management subsystems, virus scanners and guards) designed to detect and deny unauthorized access and permit authorized access in an IS. This integration adds additional hardening as well as finer-grained control than the all or nothing access provided by simple boundary creation. However, interactions of these security controls with users and other principals, as well as with enterprise security systems, generate interoperability requirements and testing for the Node.

Quarantine Creation

Quarantine is the term which describes a special family of boundary-based damage control techniques and technologies that limit external compromises of systems to an in-Node isolation construct. These techniques often also provide a way to remedy identified deficiencies prior to re-enabling normal access to system resources. Also may provide additional boundary hardening to ensure the integrity of good Components missing necessary capabilities.

High Availability Integration

High availability integration is a configuration activity which assures with high probability that a system will be operational at any given time, and will recover quickly in the event of a failure. In general, a high-availability system has safeguards to prevent unscheduled outages from power failures, code defects, or hardware failures.

Management

In the security realm, management includes monitoring and configuring boundaries and their embedded security sensors/controls through use of enterprise security engineering support systems, operational policies and procedures.

Auditing

Most information systems have a logging facility and can log all "deny access" actions which would show intrusion attempts. Modern systems have an array of logging features that include the ability to set severity based on the data logged. An auditing schedule should be established to routinely inspect logs for signs of intrusion and probing.

Detailed Perspectives

- Computing Infrastructure Integrity [P1335]
- Network Infrastructure Integrity [P1336]
- User Environment Integrity [P1337]
- Data, Application and Service Integrity [P1338]

Guidance

- G1300: Secure all **endpoints**.
- G1301: Practice layered security.

Best Practices

- BP1868: Incorporate mechanisms to enhance Computing Infrastructure (CI) availability.

P1335: Computing Infrastructure Integrity

Increasingly, security integration and enterprise security for the computing infrastructure is growing beyond securing basic hardware, firmware and software boundaries to include activities that must deal with boundaries based on virtual machines and **services** that cross system and **Node** boundaries. However, none of these more dynamic boundaries are secure unless the underlying basic components have the necessary integrity and other security capabilities.

The primary computing infrastructure boundary is the information **system** component. Subsidiary constructs include the firmware, the operating system (OS), the file system data storage, and application execution contexts such as the user account.

Operating System Hardening

Security of the operating system relies on creating some common boundaries. Creating these boundaries often requires numerous procedures such as configuring system and network interface components properly or removing or disabling unused, undefended and unnecessary files and services, while ensuring that all of the applicable security patches are in place. The DISA **Security Technical Implementation Guide** (**STIG**) repository contains authoritative checklists for operating system hardening. In addition there are Department of Defense Information Assurance Vulnerability Alert (IAVA) and Information Assurance Vulnerability Management (IAVM) notifications for compliance.

Data Storage Encryption

Data encryption can happen in many different ways. One method involves providing encryption as part of the storage. Many newer operating systems and applications have built in support for data encryption at the file, directory/folder, and volume/disk level. Each level has a potential need for boundary creation; this requires weighing the trade offs. For example, encryption at the folder or disk/volume level does not require that users or applications provide individual file encryption; therefore, auxiliary files receive automatic encryption support. However, finer-grained control will consequently require additional development, testing and training.

Remote data storage architectures typically perform encryption at the physical storage endpoint. Ensure that data remains encrypted when transmitted over the network to the physical storage endpoint to assure end to end confidentiality.

For further information see the Data at Rest [P1360] perspective.

DRM Signing at the OS and Hardware Level

Various operating systems and applications like Windows Vista, Windows Server 2008 and the Linux kernel 2.6.12 and later use Trusted Platform Module (TPM). TPM supports capabilities such as Windows BitLocker full-drive encryption technology as well as Digital Rights Management (DRM) and software licenses. A TPM microchip is embedded on the computer's (or other device's) motherboard and stores unique system identifiers along with the decryption keys. Certain systems may provide the TPM as part of the standard build.

Parity Checking

Beyond the standard use of parity checking performed with memory or communications there are also applications that make use of parity checking for the whole computer system such as Bit9. This is an example of one approach that can check a whole system for tampering to better protect against unanticipated (zero day) exploits, unauthorized software installations, etc. This process could be coded into proprietary software and or included into a program's Statement of Work (SOW), etc.

Virus Scanning

Viruses are a significant interface independent cross-boundary threat that requires constant monitoring. Some security control computing practices can help to mitigate the risk of virus infections and reduce the possibility of inadvertently triggering or spreading viruses and will help defend against malicious code attacks. Virus scanners are security controls and act as gatekeepers at boundaries. However, they do not require interoperability with other components or Nodes, except for enterprise security. Consequently, they do

not traditionally fall under the main capabilities associated with boundary gate-keeping, authorization or authentication.

Components should also enable baseline holistic scans of the whole system to prevent some of the stealthier viruses that can hide from any scan that is initialized while the system is already up and running.

Finally schedule anti-virus software to check in regularly with the master server that provides the signature and application updates.

For additional details see the Host Information Assurance [P1161] perspective.

Spyware and Malware Scanning

Spyware is a significant interface independent cross-boundary threat that requires repeated monitoring. In addition to enabling direct attacks, spyware is also a potential entry point for viruses. Enabling good security control placement can defend against malicious code attacks by limiting the risk of spyware infections, inadvertent triggering of, or spreading, spyware and related viruses.

Spyware security control programs share many best practices with related virus security control placement. Ensure that any spyware security control programs do not "step" on security control antivirus software and vice versa.

For additional details see the Host Information Assurance [P1161] perspective.

Computing Infrastructure Quarantine Support

Providing computing infrastructure quarantines is generally bundled with software security sensors and controls that detect unwanted or compromised software. With dynamic, service-oriented configurations, it is likewise important to have some type of spyware security sensor/control that can detect and remove or quarantine those unwanted "helper" components that repeatedly attempt to install themselves in a configuration. Quarantine is also a capability that is used by other security sensors/control components like malware scanners and analyzers.

High Availability

For more detailed guidance of highly available Computing Infrastructure, see BP1868 and DoD Instruction 8500.2, *Information Assurance (IA) Implementation*, [R1198] especially for Mission Assurance Category (MAC) I systems and networks. The following subsections summarize important concepts.

Data Backup and Recovery

Nodes should provide frameworks to support backup and recovery of data. Backup logs support auditing of activities.

Enable operations personnel to destroy backup media physically during disposal to prevent unauthorized reading of the media contents. Employ the "two person" rule to dispose of media; maintain meticulous tracking logs, available in hard copy as well as electronically, of all backup media.

Verify encryption of all data on removable media is with a level of encryption appropriate for the level of data protection required by policy.

Fault Tolerance

Critical components, ones on which other components are dependent such as enterprise services and infrastructure components, must not become weak links that significantly cripple the Node's operations. Their high availability ensures the continuity of operation. A precept of high availability architectures is that they are fault tolerant and/or redundant, starting with the hardware components. If a primary component fails, the secondary component takes over in a process that is seamless to the application running on the server. As such, fault-tolerant systems "operate through" a component failure without loss of data or application state.

In addition, fault tolerant/redundancy includes software-based failover clustering, in which a hardware or software failure on one server causes the workload to be shifted by the Computing Infrastructure to a second server.

Computing Infrastructure Configuration Rollback and Recovery

Nodes should provide frameworks to support backup and recovery of Node provisioning information to support configuration and change management activities. Nodes should make this framework available to Components to enable coordinated configuration and change management activities across all the Components in the Node.

Management

Management activities specific to the security realm have a heavy emphasis on managing cryptographic components of the computing infrastructure, especially those that provide key management.

Key Management

Key backup and recovery is especially important in data storage encryption to prevent loss of otherwise long-lifetime data. For example, if a disk is encrypted and then moved to another machine (because the original machine had a hardware failure), without good key backup and recovery, the data could be inaccessible. Designated key recovery agents should be kept to a minimum in order to expose fewer keys to cryptographic attack and provides a higher level of assurance that encrypted data will not be decrypted inappropriately. Refer to the National Institute of Standards and Technology Special Publication 800-57, *Recommendation for Key Management - Part 2: Best Practices for Key Management Organization (*NIST SP800-57-Part2) and the Key Management [P1041] perspective in NESI *Part 5* for additional information.

Auditing and Logging

Most information systems have a logging facility and can log all "deny access" actions which would show intrusion attempts. Modern information systems have an array of logging features that include the ability to set severity based on the data logged. An auditing schedule should be established to routinely inspect logs for signs of intrusion and probing.

Guidance

- G1622: Implement **commercial off-the-shelf** (**COTS**) software that protects against malicious code on each operating system in the Node in accordance with the Desktop Application **Security Technical Implementation Guide** (**STIG**).
- G1623: Implement personal **firewall** software on computers used for remote connectivity in accordance with the Desktop Applications, Network, and Enclave **Security Technical Implementation Guides** (**STIGs**).

Best Practices

- BP1707: Configure and locate elements of the Node Web infrastructure in accordance with the Web Server **Security Technical Implementation Guide** (**STIG**).
- BP1708: Configure and locate elements of the Node Web infrastructure in accordance with the Desktop Applications **Security Technical Implementation Guide** (**STIG**).
- BP1709: Configure and locate elements of the Node Web infrastructure in accordance with the Network **Security Technical Implementation Guide** (**STIG**).
- BP1868: Incorporate mechanisms to enhance Computing Infrastructure (CI) availability.

P1336: Network Infrastructure Integrity

Network integrity is based on network boundaries and constructs that may not be as familiar to the average person as information system boundaries and constructs. Network boundaries and constructs are often the domain of network architects and operations rather than end users, and they are often not confined to a tangible system but distributed among multiple end systems, routers and switches. Network virtualization, for example, is a routine application of these principles. In many ways, however, network constraints are very much like computing infrastructure: there are hardware and software constructs whose boundaries must be hardened as a pre-requisite to securing more dynamic constructs such as **virtual private networks** (**VPNs**) and secure sessions.

Boundary Creation

Boundaries in Transport networks are a function of the physical, link and network layer technologies and are reflected in the address structures and the bindings. Aligning these Transport functional boundaries with **Information Assurance** (**IA**) boundaries and positioning the appropriate security controls is the subject of the following discussion.

The boundary between a host or router system and its local network is its network stack (or stacks, in routers); to be visible and reachable the boundary must have an **IP** address. Security controls at this boundary are primarily a function of hardening the system hardware and software, including the network stack.

Hardening a system is a combination of assuring initial integrity of the system and its default configuration through certification and accreditation processes such as the *DoD Information Assurance Certification and Accreditation Process (DIACAP)*.[R1291] Ongoing vulnerability management must follow, especially as system software changes and configurations are adapted to local requirements and policies.

The Network **Security Technical Implementation Guide** (**STIG**) on the DISA Information Assurance Support Environment (IASE) Web site provides guidance for the boundary between the **Node's** internal network and external networks. A summary and list of examples of what is in the Network STIG follows; see the Network Information Assurance [P1147] perspective for additional details.

Router Security Considerations

There are many things to consider when determining how to secure a router or other type of network device. They all involve using the router to support the appropriate placement of security sensors and controls to harden the various Transport boundaries. They also may require associated enterprise security components to manage the policies so deployed and enforced.

Patches and Updates

Subscribe to alert services provided by the manufacturers of any networking hardware so that they are up to date with both security issues and service patches. As vulnerabilities are found, and they inevitably will be found, good vendors make patches available quickly and announce these updates through e-mail or on their Web sites. Always test the updates before implementing them in a production environment.

Protocols

Denials of service attacks often take advantage of protocol-level vulnerabilities, for example, by flooding the network. To counter this type of attack, add Node security controls and policies.

- use ingress and egress filtering
- screen Internet Control Message Protocol (ICMP) traffic from the internal network
- block trace route
- control broadcast traffic
- block other unnecessary traffic

Ingress and Egress Filtering

Spoofed packets (packets with fake or hijacked addresses) are indicative of probes, attacks, and other activities by a knowledgeable attacker. Network boundary devices should verify both incoming and outgoing packet addresses. While this does not protect the Node from a denial of service attack, it does keep such attacks from originating from the Node's network and if other networks apply the same verification, the Node's network could be saved from a denial of service attack.

This type of filtering also enables the originator to be easily traced to its true source since the attacker would have to use a valid, and legitimately reachable, source address. For more information, see the Internet Engineering Task Force (IETF) *Network Ingress Filtering: Defeating Denial of Service Attacks Which Employ IP Source Address Spoofing* Request for Comment (RFC 2827).

ICMP Traffic

The Internet Control Message Protocol (ICMP) is a stateless protocol that uses the Internet Protocol (IP) and allows verification of host availability information from one host to another. It often is used for Enterprise Management performance testing and fault isolation. However, providing a security control that can block ICMP traffic at the outer perimeter router will protect the Node from cascading ping floods and other denial of service attacks.

Trace Route

Trace route is a means to collect network topology information. It detects devices en route to a destination system and is very useful in determining whether Node and mission data is traveling along optimal routes. Its implementation varies for each manufacturer; some use a ping with differing time to live (TTL) values while others use a **User Datagram Protocol** (**UDP**) datagram. Enabling policies that block ICMP messages can control the variable ping, while the UDP datagram may require an **access control list** (**ACL**) type policy to block it. By enabling the deployment of blocking policies of this type, security controls prevent an attacker from learning details about the Node's network.

Broadcast Traffic

Directed broadcast traffic can be used to discover and enumerate hosts on a network and as a vehicle for a denial of service attack. For example, by blocking specific source addresses, security controls prevent malicious echo requests from causing cascading ping floods.

Unnecessary Traffic

Incoming traffic from the **Internet** to the boundary router is from unknown, untrusted users who require access to the Node's **Web servers**. The users are accessing a specific list of IP addresses and port numbers and can be restricted to access no other port numbers or IP addresses. Using access control lists (security controls available on most routers) only traffic for the desired combination of addresses and ports can pass through the boundary router; an assumption is that any other addresses are potentially hostile. Port numbers in this example are not related to ports on a switch which are the physical sockets into which the Ethernet cables are plugged. Here, the reference is to the IP addressing system, where the IP address is extended with a **TCP** or UDP port number. For example a Web server is frequently on port **`80`**; the full address of the Web service on a server with an IP address of **`192.168.0.1`** would be **`192.168.0.1:80`**. Cisco routers and switches use a proprietary Cisco Discovery Protocol (CDP) to discover information about their neighbors such as model numbers and operating system revision level. However, this is a security weakness as a malicious user could gain the same information. Disable CDP definitely on the boundary router and possibly on the internal routers and switches, dependent upon whether they are required for management software.

Administrative Access

Consider where router access will occur for administration purposes. Security controls enforce policies which determine which interfaces and ports allow an administration connection, and from which network or host will perform the administration; restrict access to those specific locations. Disable unused interfaces and consider static routes to enhance security. Also consider disabling Web-based router configuration. Control physical access to routers.

Do not leave an Internet-facing administration interface available without encryption and countermeasures to prevent hijacking. In addition, apply strong password policies, and use an administration access control system.

Perform router auditing and monitor router logs, and monitor for intrusion detection.

Password Policies

Add a password to the administrator account; many systems are hacked into just because the administrator has left the password blank. Secondly, use complex passwords. Brute force password software can launch more than just dictionary attacks and can discover common passwords where a letter is replaced by a number. Similarly, the Simple Network Management Protocol (SNMP) is probably required for management purposes; although SNMP security is not at all strong, do add passwords (community string) when configuring it. SNMP v3 provides much improved security. Use an administration access control system rather than embedding the administrator's name in the configuration.

Unused Interfaces

Only required interfaces should be enabled on the router. An unused interface is not monitored or controlled, and it is probably not updated. This might expose the Node to unknown attacks on those interfaces. Usually the Telecommunications network (Telnet) protocol is used for administrative access so limit the number of Telnet sessions available and use a time-out to ensure that the session closes if unused for a set time.

Static Routes

Static routes prevent specially formed packets from changing routing tables on the Node's router(s). An attacker might try to change routes by simulating a routing protocol message to cause denial of service or to forward requests to a rogue server. By using static routes, an administrative interface must first be compromised to make routing changes. However, remember that static routes are static; if a link fails the routers will not switch over automatically to use an alternate route, and static routes may need complex configuration.

Web-Based Configuration

If an inbuilt Web server is an optional method for configuration access, as well as a command line mode, disable the Web service as it is probably prone to many **TCP/IP** security weaknesses.

Services

On a deployed router, every open port is associated with a listening service. To reduce the attack potential, default services that are not required should be shut down. Examples include the Bootstrap Protocol (bootps) and Finger, which are rarely required. Enterprise security tools and personnel should also scan the routers to detect which ports are open.

Intrusion Detection

With restrictions in place at the router to prevent TCP/IP attacks, the router should be able to identify when an attack is taking place and notify a system administrator of the attack. Attackers learn what the Node's security priorities are and attempt to work around them. An **intrusion detection system** (**IDS**) can show where the perpetrator is attempting attacks.

Physical Access

Most routers are vulnerable if the attacker can get physical access to the device since they usually have a back-door access method to overwrite the existing configuration so lock the routers away in a room with restricted access.

Switch Security Considerations

There are many things to consider when determining how to secure a switch or other type of link-local network device. As in network devices like routers, they support the appropriate placement of security sensors and controls

to harden the various local area transport boundaries. They also may require associated enterprise security components to manage the policies so deployed and enforced.

Patches and Updates

Install and test patches and updates as soon as they are available on identical hardware and software located in a testing environment. If possible, include real data that has been "sanitized" in the data stores of any system selected for patching, testing or testing patches. For example, a copy of a real DB may be used, with all sensitive information stripped from it.

VLAN Boundaries

Virtual local area networks (VLANs) allow Node architects to separate network segments and apply access control based on security rules. A VLAN without ACLs provides a first level of security, limiting access to members of the same VLAN. However inter-VLAN traffic is usually required and this is provided by the router routing traffic between the IP subnets and this can be controlled by the use of ACLs. ACLs between VLANs restrict the flow of traffic between different segments of the network. This filtering is typically a simple static packet filter, as opposed to stateful packet inspection or application-layer proxying, which many dedicated firewall devices perform. Using ACLs between VLANs provides an intermediate level of protection by blocking internal intrusions from within the enterprise while intrusions from outside are already blocked by the boundary network. In addition to firewall filtering, VLAN ACLs can also be implemented for an additional layer of security. The disadvantage of implementing ACLs on the VLANs is that they may have an impact on performance and must be configured correctly and efficiently.

Administration Access

Consider where the switch access for administration purposes will occur. Security controls enforce policies which determine which interfaces and ports an administration connection is allowed into, and from which network or host the administration is to be performed. Restrict access to those specific locations. Disable unused interface, and consider static routes to enhance security. Consider disabling Web-based router configuration. In addition, control physical access to routers.

Do not leave an Internet-facing administration interface available without encryption and countermeasures to prevent hijacking. In addition, apply strong password policies, and use an administration access control system.

Perform security auditing, monitor router logs, and monitor for intrusion detection.

Unused Ports

Disable unused Ethernet ports on switches to prevent an unauthorized person with physical access from plugging into an unused port.

Services

Make sure that all unused services are disabled. Also disable Trivial File Transfer Protocol (TFTP), remove Internet-facing administration points, and configure ACLs to limit administrative access.

Encryption

Although not traditionally implemented at the switch, data encryption over the wire ensures that sniffed packets are useless in cases where a monitor is placed on the same switched segment or where the switch is compromised, allowing sniffing across segments.

Internet Boundaries: Subnets

Many administrators use the natural 8-bit boundary in the 16 bits of a class B host ID as the subnet boundary. Subnetting hides the details of internal network organization to external users. Subnets without additional security controls to restrict access are not a good security preventative measure, however simple subnets enable logical and guidance-mandated placement of such controls and will help better manage network performance.

Trusted Guards

Trusted guards are accredited to pass information between two networks at different security levels, such as between SECRET General Service (GENSER) and TOP SECRET Sensitive Compartmented Information (TS SCI), according to well defined rules and other controls.

For additional information see the Trusted Guards [P1150] perspective.

Demilitarized Zone (DMZ)

In computer security a DMZ, based on military usage of the term but more appropriately known as a demarcation zone or perimeter network, is a physical or logical sub network that contains and exposes an organization's external services to a larger, untrusted network, usually the Internet. The purpose of a DMZ is to add an additional layer of security to an organization's LAN, VLAN or subnet; an external attacker only has access to equipment in the DMZ, rather than the whole network.

Firewalls

Firewalls are a form of security sensor and access control package that are embedded at network boundaries between Nodes or between a Node and the larger **Global Information Grid** (**GIG**). They harden the boundaries of and protect the transport network architecture construct known as the **intranet**. Without firewalls, an intranet is only a performance-based grouping of local subnets linked by routers and switches.

Restrict Internet Access to Authorized Sources

Only allow source addresses from the IP network numbers assigned to trusted segments behind the Node's firewall(s), including DMZ networks. This includes primary and secondary network numbers, and subnets that are routed to the Internet through the Node's firewall (including addresses reserved for VPN clients). Apply appropriate subnet masks to trusted networks, i.e., masks that are sufficiently long to identify only that fragment of the IP network number used by Node traffic. For example, if the Node architecture specifies the use of an IETF RFC 1918 (*Address Allocation for Private Internets*) private address from the Class B number **`172.16.0.0`**, and policy only assigns numbers from **`172.16.1.x`**, the configurations should use **`255.255.255.0`** (or **`/24`**), not **`255.255.0.0`** (or **`/16`**) as the subnet mask. Block broadcasts from traversing the firewall's interfaces. While most broadcasts will not pass across **LAN** segments, take measures to ensure this is especially true for Internet-bound packets (or packets destined for any untrusted segment). Prevent traffic from any RFC 1918 private addresses from being forwarded over an Internet access circuit. While Internet service providers (ISPs) block incoming traffic containing private addresses, relying on an external ISP to process traffic according to Node-local policy may not ensure enforcement with any accountability. Block outbound traffic from VLAN workgroups or entire network segments that have no business establishing client connections to Internet servers. If the Node has internal servers that have no business establishing client connections to Internet servers, block all outbound traffic from such systems. An example might be an intranet server that relies entirely on internally provided services (**DNS**, mail, time, etc.) and uses no applications that require Internet access.

Restrict Internet-Accessible Services (Destinations)

Allow outbound connections only to those services the Node's security and acceptable use policies allow for client hosts. Wherever possible, only allow clients to access authorized services from authorized servers. Allow access to service ports Node-internal servers must use to operate correctly, and only allow Node-internal servers access to these services. If the Node operates local mail servers, make certain that only these servers establish outbound **Simple Mail Transfer Protocol** (**SMTP**) connections. (If such measures had been practiced, the Sobig worm, which installed its own SMTP mailing engine, would not have spread so rapidly.) If the Node operates an **HTTP** proxy, or a proxy system that performs some form of Web **URL** or content filtering, only allow outbound proxy connections through the Node firewall. If the Node provides DNS internally, or uses a split DNS, use internal servers as forwarders for the Node-internal trusted network, and only allow outbound DNS requests from the Node's DNS servers so configured. Unless the Node's firewall is participating in routing, block routing protocols at the Node firewall. This is important for large enterprises with multiple firewalls and Internet access routers as well as small operational facilities with broadband connections that use a firewall to exchange and negotiate PPP over Ethernet (PPPoE). Allow any authorized services that make use of unique ports for remote desktop, subscription, licensing channels (e.g., GoToMyPC, BackWeb, and Microsoft). Allow access to these services from hosts that are authorized to use them. Certain network and security vendors use

unique ports for proprietary (and secure) management access. Permit these, but only from hosts used by the administrators of such equipment.

Follow the guidance provided in the STIG for **Domain Name System** (**DNS**) implementations.

Overlay Network Boundaries

Common examples of overlay network constructs include **virtual private networks** (**VPNs**), and content-based networks (including the localized ones known as DMZs) based on port and protocol firewalls or deep-inspection guards. For further details on subnets and VPNs see the Subnets and Overlay Networks [P1351] perspective.

Performance VPN Access Control

Use a hardened virtual private network (VPN) server to allocate IP address leases and Multi-Protocol Label Switching (MPLS) labels to remote access clients. Use strong authentication to VPN servers.

Protection VPNs

Do not use pre-shared keys. Pre-shared key authentication is a relatively weak authentication method. In addition, pre-shared keys are stored in plaintext. Pre-shared key authentication often is provided for interoperability purposes and to adhere to IP Security (IPsec) standards.

Use the advanced encryption standard (AES) for stronger encryption.

For computers connected to the Internet, do not send the name of the **Certificate Authority** (**CA**) with certificate requests. When using certificate authentication to establish trust between IPsec peers, each IPsec peer sends to the other peer a list of trusted root CAs from which it accepts a certificate for authentication. Each of these CA names is sent as a certificate request payload (CRP), and it must be sent before trust is established. Although transmitting this list aids in connectivity by facilitating the selection of a CA, it can expose sensitive information about the trust relationships of a computer, such as the name of the company that owns the computer and the domain membership of the computer (if an internal public key infrastructure is being used), to an attacker. Therefore, to secure computers that are connected to the Internet, enable the option to exclude the CA name from the certificate request.

For computers connected to the Internet, do not use Kerberos as an authentication method. When using Kerberos V5 authentication during main mode negotiation, each IPsec peer sends its computer identity in unencrypted format to the other peer. The computer identity is unencrypted until encryption of the entire identity payload takes place during the authentication phase of the main mode negotiation. An attacker can send an Internet Key Exchange (IKE) packet that causes the responding IPsec peer to expose its computer identity and domain membership. Use certificate authentication to secure computers that are connected to the Internet.

Do not allow unsecured communication for computers connected to the Internet. If it is Node policy to configure a filter action to negotiate Internet Protocol Security (IPsec), ensure that the following options are disabled in order to secure computers that are connected to the Internet:

- ***Accept unsecured communication, but always respond using IPsec.*** This option allows initial incoming unsecured traffic (for example, TCP SYN packets) but requires protection of outgoing traffic. Disable this option to prevent denial-of-service attacks.
- ***Allow unsecured communication with non-IPsec-aware computers.*** This option allows unsecured communications with computers that cannot negotiate the use of IPsec or process IPsec-secured communications; it is appropriate only in environments where IPsec-secured communication is not necessary.

Tactical and Other Non-IP Networks

Gateways and/or edge routers handle tactical data link local networks such as Link 16. As such they are sub-nets or overlay nets from the wider GIG point of view. Link local networks may require additional boundary protection such as **High Assurance Internet Protocol Encryption** (**HAIPE**), spread spectrum, etc. For further information see the Subnets and Overlay Networks [P1351], Black Core [P1152] and Design Tenet: Encryption and HAIPE [P1247] perspectives.

Content Proxy Networks

Use Domain Name System Security Extensions (DNSSEC) or equivalent directory services to define content routing topologies (Refer to IETF RFC 4033). Use strong authentication with and between proxy servers and message routers.

Use secure directory services such as StartTLS or SLDAP to define **Enterprise Service Bus** (**ESB**) routing topologies.

Overlay Firewalls

Use "black boxes" (like a Nokia IP2255 appliance running Check Point NG) or stripped and hardened dedicated computers as overlay firewalls. The latter choice could involve significantly more maintenance.

Overlay DMZ and Quarantine Zones

Deploy anti-virus gateways at Node network boundaries. In addition, deploy intrusion detection system (IDS), intrusion prevention system (IPS) and other security technologies on at least all outward facing gateways. Nodes should employee virus protection, enabled for both outbound and inbound traffic, at the gateways.

Other Security Concepts

Common DoD-required Transport security controls include the following.

Host, Application, and Network Based IDS/IPS

An intrusion prevention system is a computer security device (generally a software agent, but can be hardware based as well) that monitors network and/or system activities for malicious or unwanted behavior. It can react, in real-time, to block, prevent and or report those activities. The primary difference between an IDS and an IPS system is that IDS only reports where the IPS can take an active role in prevention as well as reporting the activity. The three generally accepted types of IDS/IPS agents are at the network, the operating system, and the application. They perform in one of several ways, like antivirus applications they can use a signature-based, anomaly-based, or hybrid mode to compare observed activity against behaviors that are indicative of potentially malicious outcomes.

Parity Checking

Beyond the standard use of parity checking performed with memory or communications there are also applications that make use of parity checking for the whole computer system. This process could be coded into Node proprietary software, into a Statement of Work (SOW) or Request for Comment (RFC), etc.

Quarantine Concepts and Context

In-Node Transport quarantines are often bundled with the security sensor and controls used to create the boundaries of network constructs such as a DMZ.

Quarantine Zone in DMZ

Most security professionals recognize that a good standard security practice is to implement a quarantine zone within or parallel to the primary DMZ. The main purpose of this is to verify specific installation, configuration and overall compliance with security policy mandates.

Highly Availabilty

Highly available networks require a combination of highly available hardware and software components and highly available distributed components such as routing topologies.

Fault Tolerant and Redundant Networks

Networks are critical Node infrastructure components whose high availability ensure the continuity of net-centric operation. High availability network systems start with the hardware components. If a primary router

fails, traffic may either be switched to an alternate "blade" or be rerouted through alternate network links without any action required on the part of other components.

Multi-Homed Hosts

Nodes should employ network multi-homing to enabling components to connect through alternate networks and not just relying a single network connection whenever mission critical resources, components, or services are not local or organic. Generally, a router or gateway on the external boundary of the Node can accomplish this; multi-homing requires assigning as many network addresses as there are networks employed, requiring management considerations.

Management

Capabilities necessary to Transport network management for enterprise security purposes include the usual two techniques and Component technologies:

Key Management

Refer to IETF RFC 4962, *Guidance for Authorization, Authentication and Accounting Key Management*, for information on network key management.

Auditing and Logging

Most routers have a logging facility and can log all deny actions which would show intrusion attempts. Modern routers have an array of logging features that include the ability to set severities based on the data logged. An auditing schedule should be established to routinely inspect logs for signs of intrusion and probing.

Guidance

- G1352: Use database clustering and redundant array of independent disks (RAID) for high availability of data.
- G1667: Implement **Virtual Private Networks** (**VPNs**) in accordance with the guidance provided in the Network **Security Technical Implementation Guide** (**STIG**).

P1337: User Environment Integrity

User environment boundaries and infrastructure constructs considered separately from the computing infrastructure only emerged with the rise of the **Internet**, the **World Wide Web**, net-centric operations and **service-oriented architectures**. These constructs and boundaries start with physical hardware; software and virtual constructs and boundaries are layered on top. Some of the more established user environment infrastructure constructs include displays and input devices (both real and virtual), client applications, Web browsers and, more recently, rendering engines.

Determining user environment boundaries tends to focus on those subsets of the computing infrastructure resources delegated to and dedicated to a particular user, service agent or process display.

Browser Hardening

Browser hardening is the process of identifying an acceptable Web enabled browser that will function properly with the necessary site accesses. Properly configure the browser to work with the antivirus, antiphishing, antispyware, and firewall solutions. Only download and install a browser from a trusted site and ensure that the digital hashes match before installation. Never run the browser as a "root" or "admin" user.

There are numerous browser **Information Assurance** (**IA**) plug-ins for application, data and services security. Users should either not be able to install additional plug-ins and controls or at least be restricted to approved and **PKI** digitally signed plug-ins and controls. Enable only the those plug-ins and controls that are really needed by the end users, such as Active X, Java controls, etc. Configure these mobile code controls per the DoD Mobile Code policy; see the Mobile Code [P1314] perspective for more information.

Mobile Device Protection

Adopt a multi-tier security approach to mobile security. Set policies to password-protect hand-helds, ensuring employees use strong passwords and personal identification numbers (PINs), and change them frequently to make it difficult for thieves to access confidential information. Protect mobile devices, boundary devices, with internal antivirus gateways, firewall, anti-SMS spam filters, and data encryption technologies. Install regular security updates to protect phones and corporate information from viruses and other malware. Organizations should provide this technology to their employees and teach them how to use it properly. Disable Bluetooth and wireless signals when they are not in use. Bluetooth headsets should be paired exclusively with one employee's handheld device. Regularly scan mobile devices and their information for viruses and other malware. Regularly scan mobile devices and their information for viruses and other malware. Many mobile devices have the capability to receive a "Self Destruct" order which scrambles the internal workings of the device (memory, flash BIOS, etc). This should be a consideration during acquisition and included in concepts of operations (CONOPS) and training.

High Availability Guidance

Employees should schedule regular backups for hand-helds just as they would for any other computer system.

P1338: Data, Application and Service Integrity

Data, application and service boundaries and constructs are virtual; they cannot be separated fully from the underlying computing and transport infrastructures. Generally, they sub-divide these infrastructures in order to prevent interference between, and maintain the integrity of, different mission or business operations. Although the actual boundaries and constructs are operational-specific and consequently a local matter, many of the techniques and technologies used are standard.

Boundary Creation

Formal boundaries in data, applications or services are generally created by application-layer interfaces. Examples include data models and schema, application programming interfaces (APIs) input and output argument datatypes and service protocol interfaces. Baseline hardening such boundaries through type- and range-checking or protocol error handling is a generally standard engineering practice.

Digital Signing

Digital Rights Management (DRM) signing (application) depends on a **Trusted Platform Module** (**TPM**) which is used with various operating systems and applications like Windows Vista, Windows Server 2008, and the Linux kernel 2.6.12 and later. It supports capabilities such as Windows BitLocker full-drive encryption technology as well as DRM and software licenses. A TPM microchip is embedded on the computer (or other device) motherboard and stores unique system identifiers along with the decryption keys.

Parity Checking

Beyond assuring integrity by parity checking data in memory or in communications, there are also utilities that make use of parity checking at the services or application level, enabling the "white-listing" of components for execution. White-listing components may more efficiently protect by detecting and preventing zero day exploits, unauthorized software installations, etc. Providing such a capability is a combination of concept of operations (CONOPS) and helper utilities (such as Parity from Bit9 or variant on the open source Tripwire such as Tripwire Enterprise from Tripwire Incorporated).

High Availability

Ensuring high availability of data, applications and services generally is the responsibility of the underlying functional environment infrastructure and not a separate capability. For example, see the *High Availability* subsection in the Computing Infrastructure Integrity [P1335] perspective.

Management

Managing data security, application or service-level security is generally the responsibility of the underlying functional environment infrastructure and is not a separate capability.

Guidance

- G1302: Validate all inputs.

P1178: Identity Management

Identity Management covers the spectrum of tools and processes that serve to represent and administer digital **identities** and manage access for those identities. Identity is an essential part of the **Core Enterprise Services** (**CES**) Security Services, but CES Increment 1 does not address Identity Management. Identities of **Global Information Grid** (**GIG**) entities, human and non-human (i.e., **services**), must be unique across the GIG. DoD **PKI** X.509 **certificates** reserve a field to contain identity data, but there are issues today with how that field is populated for certain types of users (e.g., coalition partners), and how to handle non-person entities.

While a universal solution for Identity Management is not yet defined, it is possible to make progress in the implementation of these services, particularly for Web applications and services with U.S. users having a **Common Access Card** (**CAC**) holding DoD PKI X.509 certificates.

Identity is not as well understood and defined for non-person entities, such as services that may be part of a long invocation chain that in turn is part of a workflow or is orchestrated to yield a specific answer to a service invocation. The definition of Web server credentialing, though, relies on the DNS name of the site for **identification**.

The **Net-Centric Enterprise Services** (**NCES**) and **Public Key Infrastructure** (PKI) Program Offices are working on the challenges of non-person Identity Management, and there is a request for information (RFI) to identify potential solutions.

Each identity credential technology varies in strength. The weakest methods are password-based and the strongest are combinations of biometrics and smart cards.

There are also differing strengths within each method. For instance, systems that require complex passwords are stronger than those that accept simple ones, and systems using retina or fingerprint readers are stronger than those that use finger length.

Components that are separate from the implementation of mission- or business-specific functionality often provide identity authentication management and authorization.

Detailed Perspective

- Public Key Infrastructure [P1179]

Guidance

- G1652: Use DoD **PKI** X.509 **certificates** for **servers**.

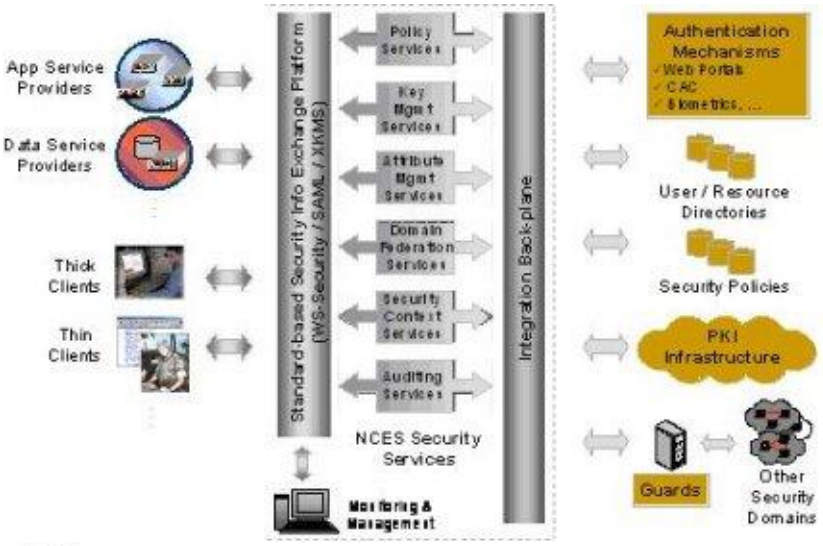
P1179: Public Key Infrastructure

Net-Centric Enterprise Services (**NCES**) Security Services rely heavily on **Public Key Infrastructure** (**PKI**) and **Public Key** (**PK**) Enabling (**PK-Enabling**). PKI provides an assured way for enabled applications to authenticate both intra-node and inter-node. PKI supports the concept of a single login across the **enterprise**, but legacy non-PK-enabled applications and services mean that username and password synchronization is also needed to support the single login concept; however, this is only practical in a limited sense (i.e., not the entire **Global Information Grid** or **GIG**). There remain some PKI implementation challenges, such as the implementation of the process for validating that an entity's **certificate** has not been revoked. Some commercial (**COTS**) products, including some Web Application Containers, do not support the use of the **Online Certificate Status Protocol** (**OCSP**) or do not provide a capability to do file-based checking of the older **Certificate Revocation List** (**CRL**). The U.S. Department of Defense, through the DISA NCES program, supplies Robust Certificate Validation System (RCVS) services for PKI certificates, including **Common Access Card** (**CAC**) credentials; for smart card reader information, see the Common Access Card (CAC) Reader [P1156] perspective. PKI certificate checking includes using OCSP and CRL; the **Joint Interoperability Test Command** (**JITC**) OCSP portal contains more detailed information. For additional PKI-information see the Technologies and Standards for Implementing Software Security [P1391]-related perspectives including Public Key Infrastructure (PKI) and PK Enable Applications [P1061], Key Management [P1041], Certificate Processing [P1009], Encryption Services [P1020], and Smart Card Logon [P1315].

Nodes having both DoD and **Intelligence Community** (**IC**) systems and networks will also face the fact that the DoD and IC have implemented separate PKIs (including the dependent Directory Services). In general, the DoD PKI operates on the collateral classification networks, and the IC PKI operates on classified **Sensitive Compartmented Information** (**SCI**) networks. Nodes may have to interface with multiple PKIs, therefore, depending on the systems and security levels at the Node. This presents some additional challenges when cross-domain interoperation is required, whether intra- or inter-node.

Nodes that have multinational or coalition personnel accessing the system will also encounter a challenge in obtaining CACs containing PKI certificates for these persons. The process is not well defined. As DoD moves further into the net-centric concepts, obtaining certificates for non-human entities in multinational or coalition systems will also be a challenge.

Authorization based on **attributes** corresponding to an entity is a practical way to implement authorization, provided that the enterprise can agree on the definitions of the attributes, policy, and a way of securely communicating and validating role membership. Unfortunately, attribute definitions and common security policy are not defined yet for the **Global Information Grid** (**GIG**), and Nodes are forced to use interim approaches, such as Windows **Active Directory** (**AD**) or **Node Information Services** (**NIS**) group memberships, and evolve to a uniform definition of GIG roles and policies. Federation has not been addressed sufficiently to provide specific guidance.



I1191

Guidance

- G1306: **Authenticate** the **identity** of **application** users.

P1339: Authorization and Access Control

Authentication and **identity management** are prerequisites for **authorization** and **access control**. Where authentication and identity management serve to determine "who" (i.e., person or machine) a subject is, authorization and access control determine what privileges a given subject (once identified or authenticated) is allowed for a given resource. In other words, authorization determines what a subject can do with a given resource.

Authorization may grant or deny privileges for resources based on a wider variety of criteria beyond the identity of a subject. Authorization may determine privileges by conditions which may or may not have anything to do with the attributes of the particular subject. For example, user and security roles, the time of day, and location may all be used along with or without the identity of a subject to make a determination for granting privileges.

Because authentication, authorization, and access control are so closely related in most real applications, it is often difficult to discuss them separately. Authentication only establishes the validity of a human or machine entity. Authorization establishes the privileges and span of control for entities, but checking those privileges may be a side effect of being allowed network or physical access rather than checking specific privileges. Access control implements explicit authorization as a combination of policy management components and embedded security control components such as Policy Decision Points (PDPs) and Policy Enforcement Points (PEPs) such as Access Control Sets (ACSs).

The following example is to clarify authorization and access control. Modern files systems are an implementation of authorization and access control. File and directory authorization grants privileges (such as read, write, or execute) to the subject which owns a given file or directory. Additionally, access control is based on the group(s) a subject belongs to in order to grant additional privileges to the subject for the use of a given file or directory.

Various techniques such as roles or attributes may be the basis for access control. (**RBAC**) and Attribute-Based Access Control (ABAC) are examples. For further information on authentication processes see the Design Tenet: Identity Management, Authentication, and Privileges [P1243] perspective. Role definitions are typically within a system boundary and occasionally within or between enclaves. Access control and security often use roles.

Doctrinal spans of control interacting with technical spans of control define net-centric boundaries within Nodes. The presumption in net-centric operations is that the infrastructure extends the span of control beyond the local system; therefore, the limits of the Node technologies define the boundaries.

Authorization policies, therefore, apply within a system and within a Node. Interoperability between Nodes or between a Node and other **Global Information Grid** (**GIG**) systems require federated authorization and protocol negotiations (such as **PKI Certificate Authority** chains and **SAML** transitive trust). In addition, policy may also need alignment through manual negotiation and coordinated configuration.

Restrict the use of administrative credentials in an organization. Administrators can view and modify the security policy settings on computers, network devices, user environments, etc. For this reason, and as a general security best practice, apply the Principle of Least Privilege [P1317] (see *Part 5: Developer Guidance*) throughout the Node.

Authorization and computing infrastructure access control occur at the following main standardized technical boundaries identified by process and storage identifiers: the local system; any virtual machine (VM); any cluster, grid and network file system; and any GIG utility computing grid or network file system.

Authorization and user environment access control occur at the following main standardized technical boundaries or user environment identifiers: the local user account, any virtual machine or browser sandbox.

Process logic access control, such as captured in a formal business process specification (e.g., WS-BPEL [R1347]), and service access control are generally dependent on security controls within **Web service** infrastructure boundaries. WS-Policy and SAML use XML boundaries, which generally map to data structures and process objects.

Authorization and access control can extend to the transport layer. Use features intended to ensure that a third party cannot intercept, read or alter data transmitted over a network.

For example, **SSL** allows for authenticating and controlling access to data over an **HTTP** connection using **credentials** (such as a client or server digital **certificate**). Access may be controlled for a given subject (such as a user or client system) or a group of subjects (for example all users belonging to a given certificate authority).

With SSL communication, any of the following authentication scenarios are possible:

- No SSL authentication (or null authentication): The server does not send a certificate and does not request a certificate from the client. From an SSL perspective, the server does not know who the remote client is, or accepts any certificate that the client may present.
- One-way SSL authentication: Either the server or the client, but not both, requires certificates. Server authentication, for example, is one-way authentication where the server sends its certificate to the client but does not request a certificate from the client. Alternatively, the server may require a certificate, but does not send one and the client does not require one.
- Two-way SSL authentication: This is client and server authentication, where the server sends a certificate required by the client and also requires the client to send a certificate.

Configuring SSL authentication in the server is independent of configuring SSL authentication in the client.

Guidance

- G1306: **Authenticate** the **identity** of **application** users.

P1340: Confidentiality

Confidentiality is the property of preventing disclosure of information to unauthorized individuals or systems. For example, a credit card transaction on the Internet requires transmitting the credit card number from the buyer to the merchant and from the merchant to a transaction processing network. The system attempts to enforce confidentiality by encrypting the card number during transmission, by limiting the places where it might appear (in databases, log files, backups, printed receipts, and so on), and by restricting access to the places where it is stored. If an unauthorized party obtains the card number in any way, a breach of confidentiality has occurred.

Breaches of confidentiality take many forms. Permitting someone to look over your shoulder at your computer screen while you have confidential data displayed on it could be a breach of confidentiality. If a laptop computer containing sensitive information about a company's employees is stolen or sold, it could result in a breach of confidentiality. Giving out confidential information over the telephone is a breach of confidentiality if the caller is not authorized to have the information.

Confidentiality is necessary (but not sufficient) for maintaining the privacy of the people whose personal information a system holds. Confidentiality and privacy control occurs in computing, network and user environment infrastructure.

- ***Computing Infrastructure*** confidentiality and privacy control occur within standardized technical boundaries such as the local system; a virtual machine (VM); a Node cluster, grid and network file system; and a **Global Information Grid** (**GIG**) utility computing grid and network file system. This requires protection (usually encryption) of both the virtual storage and virtual network protocols through secure transports.

- ***Network Infrastructure*** confidentiality and privacy control occur within the following standardized technical boundaries by offering either physical protection or payload encryption: the local area subnet or VLAN, the intranet subnets, any relevant overlay networks, and the GIG internet (e.g., **SIPRNet**).

- ***User Environment Infrastructure*** confidentiality and privacy control occur within the following standardized technical boundaries through access control privileges: the local user account and any virtual machine (VM) or browser sandbox.

- ***Data, applications and services*** confidentiality and privacy control occur within the following standardized technical boundaries or application identifiers: the local application or service invocation or session context, Web page context, or application field context.

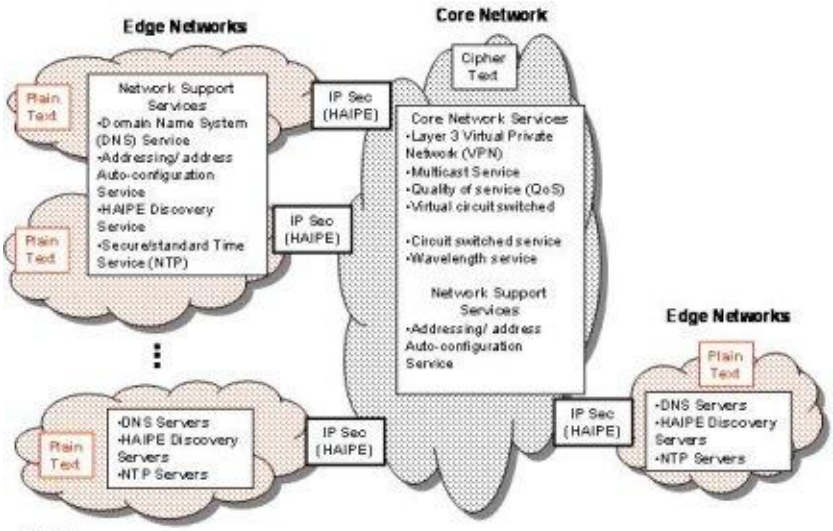
Detailed Perspective

Black Core [P1152]

P1152: Black Core

The DoD will be aggregating **Internet Protocol** (**IP**) packet traffic from multiple security enclaves onto network segments secured at the network layer in the protocol stacks; these segments, called the Black Core, are enabled through the use of **High Assurance Internet Protocol Encryption** (**HAIPE**) devices. Challenges to the implementation of HAIPE devices and the Black Core include organic support for the following: IP-based **quality of service** (**QoS**), dynamic unicast IP routing, support for dynamic **multicast** IP routing, support for mobility, and support for simultaneous **Internet Protocol Version 6** (**IPv6**) and **Internet Protocol Version 4** (**IPv4**) operation.

The Black Core is a concept fundamental to **Global Information Grid** (**GIG**) networking, but actionable guidance is still in its infancy. Interoperability with the Black Core will require active monitoring by the **Node's** management and program offices. The basic architecture of the Black Core is shown below. The Node typically provides one or more edge networks as shown in the diagram, along with the services indicated. The edge (Node) networks are sometimes referred to as **Plain Text** (**PT**) networks, while the Black Core is the **Cipher Text** (**CT**) network.



I1182

Best Practices

- BP1670: Plan for Black Core implementation in the local Node.
- BP1671: Consider Black Core transition whenever there is a significant Node network design or configuration decision to make in an effort to avoid costly downstream changes caused by Black Core transition.

P1147: Network Information Assurance

Implementation of the DoD **Information Assurance** (**IA**) Strategic Plan is required to comply with the DoD **Net-Ready Key Performance Parameter** (**NR-KPP**). Components that implement IA, however, can be a barrier to interoperability by default; proper implementation is critical. Furthermore, as net-centric applications and services emerge, so too will the need to dynamically configure the IA Components to permit net-centric operations. As an example, **access control** based on **Internet Protocol** (**IP**) address would not work, as the addresses of service users will not be known a priori when such services are dynamically discoverable.

The DoD provides requirements and extensive guidance for the implementation of information assurance at the DISA Information Assurance Support Environment (IASE) Web site. In particular, the Network **Security Technical Implementation Guide** (**STIG**) on the IASE Web site provides guidance for the network implementation, particularly the boundary between the Node's internal network and external networks. It identifies several IA systems, capabilities, and configurations as listed below and provides guidance for implementation of each.

Rather than repeating the contents of specific guidance in this document, readers should check the IASE Web site for current Network IA guidance on topics such as the following:

- External Network **Intrusion Detection System** (**IDS**), anomaly detection, or prevention device if required by the **Computer Network Defense Service Provider** (**CNDSP**)
- **Router** Security with **Access Control Lists**
- **Firewall** and application level **proxies** (may be separate device to proxy applications)
- Internal **Network Intrusion Detection** (**NID**) system
- DMZ, if applicable for publicly accessible services
- Split **Domain Name System** (**DNS**) architecture
- Domain Name System Security Extensions (DNSSEC) for higher level domain servers
- Secure devices and operating systems (i.e., **STIG** compliant)
- Ports and **protocols**

Furthermore, DoD **computer network defense** (**CND**) policies ***mandate all owners of DoD information systems and computer networks enter into a service relationship with a CNDS provider***.

Best Practices

- BP1701: Configure **Components** for **Information Assurance** (**IA**) in accordance with the Network **Security Technical Implementation Guide** (**STIG**).

P1150: Trusted Guards

Trusted guards are accredited to pass information between two networks at different security levels, such as between ***SECRET General Service (GENSER)*** and ***TOP SECRET Sensitive Compartmented Information (TS SCI)*** level networks, according to well defined rules and other controls. Guard products only pass defined types of information (e.g., email, images, or formatted messages). A key challenge is how to implement net-centric operations across trusted guards in the presence of **CES** services. See the Cross-Domain Interoperation [P1169] perspective for additional information.

Best Practices

- BP1653: Do not build dedicated Node guard products.
- BP1654: Do not build dedicated **Component** guard products.
- BP1668: Acquire and configure approved guard products with the help of the Government program offices that acquire such guards.
- BP1669: Select **XML**-capable **trusted guards**.

P1330: Enterprise Management

Enterprise Management involves planning, organizing, staffing and governing an **enterprise**. A Node packages operational capabilities into standard technology-based components (see the NESI Node Decomposition [P1343] perspective). Each component, regardless of functional area, has management information associated with it that makes it manageable throughout its lifecycle while at the same time enabling their assembly into a Node within the lifecycle and operational context of that Node. This management information is available to authorized managers through management interfaces (to include paper and electronic means).

In addition to a technical Node decomposition viewpoint, there is a semi-standardized Lifecycle decomposition viewpoint that the business operations community of the enterprise management generates. The community that manages infrastructure service operations (often referred to as **NetOps**) further focuses on aspects of the Node and Lifecycle viewpoints in a more detailed activity decomposition view.

Thus, the following three viewpoints, each with applicable standards and governance, may apply when considering or decomposing enterprise management functions.

- ***Component*** - identifies guidance and necessary interfaces to manage the Node components throughout the lifecycle
- ***Lifecycle*** - identifies guidance about configuration management, change management and responsibility handoffs
- ***Operational Activity*** - identifies detailed guidance for the deployment and operational support phase of the lifecycle

Component Viewpoint

Three basic principles help describe Enterprise Management:

- ***Decomposition*** breaks the enterprise down into modules for management purposes
- ***Delegation*** assigns the responsibility for managing each module to a representative management agent; delegation of responsibility may be applied through a tiered approach such that hierarchies of management agents may aggregate, collate and correlate management information reported by more localized management agents
- ***Decision authorization*** specifies where, when and which policies and human oversight affect Node and component operations, including machine-to-machine operations; decision authority in machine-to-machine operations rests in policy decision points and policy enforcement points, which may be in separate component modules (often the manager and agent, respectively) or co-located due to performance or security constraints

Standards, in addition to the above principles, play an important role in enterprise management. For interoperability and enterprise management purposes, each type of managed module must identify itself and publish a standardized version of the management information and operations it makes available to enterprise management systems. For example, a managing component may interact remotely with the modules it is responsible for managing. In this case, each module will reside on a network and use standard transport interfaces and management protocols such as the ***Simple Network Management Protocol*** (***SNMP***). To enable management functions, each instance of a managed module must have a **Uniform Resource Identifier** (**URI**) that enables deploying, provisioning, monitoring and adjusting in accordance with the enterprise's policies and protocols. Management URIs are usually defined as part of the data standard's protocol. For example, STD 62 (IETF RFC 3418) uses SNMP **URLs** for management URIs.

Lifecycle Viewpoint

Traditionally, lifecycle decomposition is a procedural decomposition of change management. Since responsibility and authority for controlling change is a jealously guarded right of every organization, no matter how small, a standard lifecycle decomposition must enable customized and tailored components while simultaneously establishing minimum acceptance and interoperability criteria of those components.

Historically, coordinated change management between organizations (including acceptance and interoperability testing) was either not necessary due to independent organizations without interaction or routinely was built-in as a unified command or other overarching higher authority that aligned subordinates. In either case, the result was a single change management process: either a relatively simple local process, or highly political deconfliction interactions between high level leaders. Consequently, there were no successful open international standards because they poorly replicated existing processes at a higher overhead cost.

This situation is changing; with the rise of software and its inherent dynamic and complex configuration management, developers felt the need for more formal standards of acceptance and interoperability, one amenable to industrialized production methods. There have been several attempts to promulgate these standards, such as the International Organization for Standardization "Quality management systems - Guidelines for configuration management" (ISO 10007, 1 June 2003) and "IEEE Standard for Software Configuration Management Plans" (IEEE Std 828). However, due to the extreme diversity of software products, classic methods of industrialized production have not brought many of the anticipated cost reductions, and none has seen widespread adoption as a unified standard. A number of common concepts, constructs, and procedures are emerging and do show up in various standards optimized for a particular Node decomposition area. They first appeared in the Transport area, in the **Internet** standards venue, when the **Internet Engineering Task Force** (**IETF**) standards body insisted on cross-organization interoperability as the primary criterion for acceptance of any proposed standard.

The U.S. Government describes the lifecycle procedural breakdown with a major emphasis on acquisition and a minor one on operation. Thus, Lifecycle decomposition is driven by two basic principles:

- a spiral of change in which distinct organizational roles hand-off responsibility for change management of a system, component or Node
- a minimal set of process constructs: management roles, protocols and data that serve to coordinate the handoffs and provide continuity throughout the spiral

Additionally, two things drive the elaboration, refinement or extension of these principles:

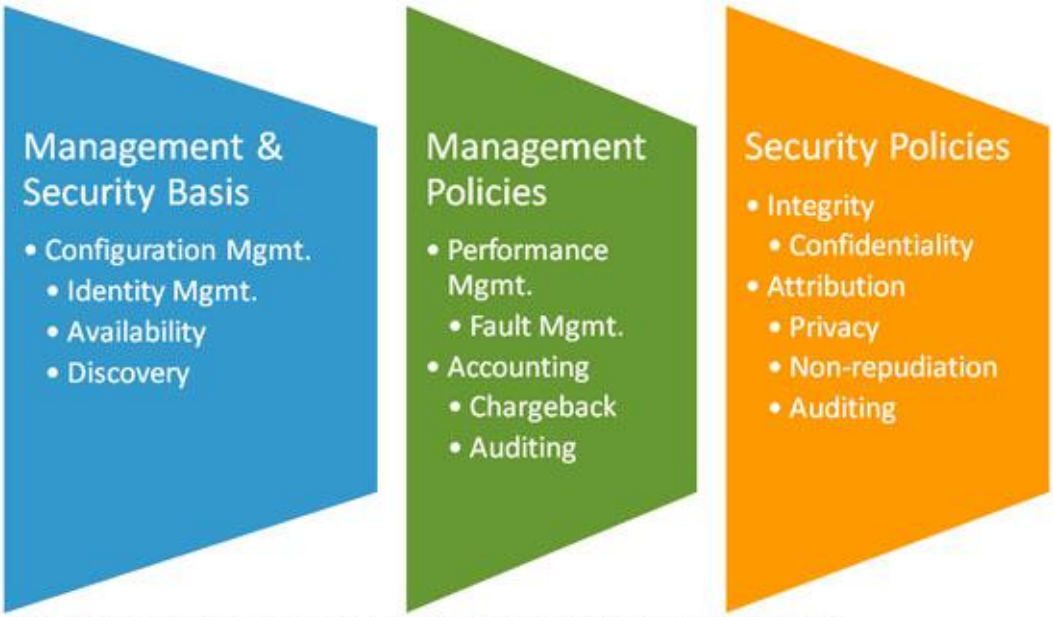
- the resources available to the organization for refinement of process constructs
- the resources required for development and integration of replacement technology in accordance with the refined process

Operational Activity Viewpoint

A refinement of the Component and Lifecycle Viewpoint decompositions into the Transport, Networks and Telecommunications functional areas generated the initial Service Operations Activity Viewpoint decomposition. Five areas defined the original decomposition: ***Fault***, ***Configuration***, ***Accounting***, ***Performance*** and ***Security***; thus, this decomposition is known by the acronym ***FCAPS***. The standards body which has evolved into the **International Telecommunication Union** first developed this reference framework for telecommunications management, captured by the ISO X.700 family of standards which is now part of the ITU-T Recommendation series M.3000.

The five activity areas were originally seen as independent; as the standard developed, it became evident that they were sufficiently inter-dependent that a single protocol was sufficient to cover all five areas. The main differences among the activity areas were how human oversight and policies were included. Configuration Management (to include Identity Management, Availability and Discovery) is the foundation layer for both Management and Security; the relevant policies are simply statements of the acceptable bounds of existence (what is in the configuration inventory) and efficacy (which types, versions, and default behavior options).

The split between Management and Security derives from the different types of operational and organizational policy drivers: efficiency and assurance. Management of efficiency drives Performance Management plus its extension, Fault Management, and its organizational policy management, Accounting (to include Chargeback and Auditing). Security (responsibility for assurance) drives Integrity plus its extension Confidentiality, and its organizational policy management Attribution (and its extensions Privacy, Non-Repudiation and Auditing). Management (efficiency) and Security (assurance) policies are generally captured in a relevant profile or other policy construct.



I1240: Operational Activity Decomposition of Enterprise Management

Note that such profiles must be appropriate to the Node Operating Environments [P1345] in which they are deployed, in accordance with operational guidance, with the following characterizations:

- Configuration includes component type, count and rate of change for making effective selections over the whole portfolio and lifecycle (i.e., development, production and deployment, operations and support).
- Management includes component resource availability, expected capacity and rate of consumption and expected level of tolerable inefficiencies.
- Security includes components' tolerance of change (especially unexpected, unauthorized and enterprise management changes) and assurance of sufficient efficiencies to provide resource reserves necessary for resilience and expected levels of interference and threats

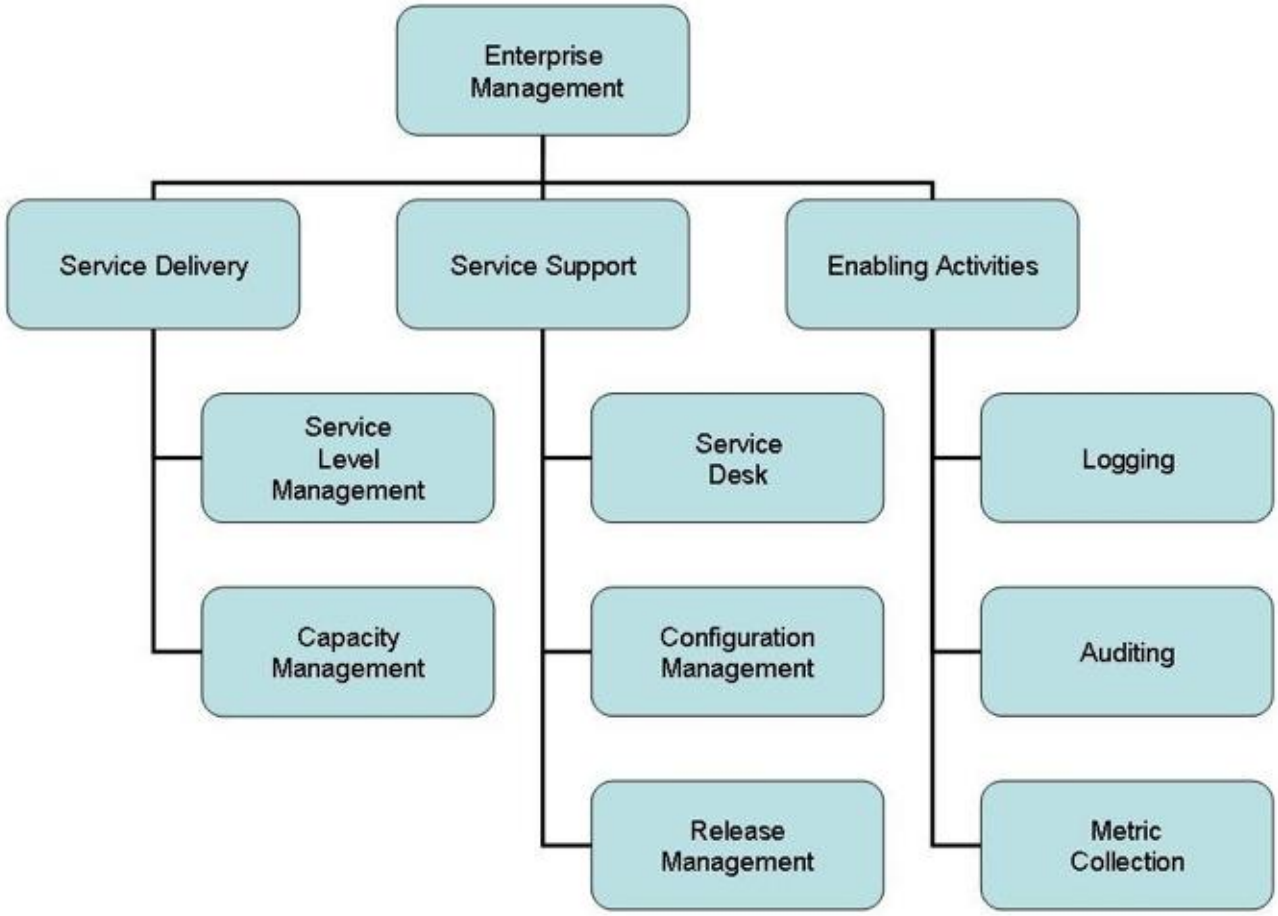
Enterprise Management Decomposition Example

The *Example Enterprise Management Decomposition* diagram below (I1219) illustrates a decomposition of enterprise management into service delivery, service support and enabling (supporting) activities.

- ***Service Delivery*** monitors and reconfigures the provisioned capabilities and capacities according to dynamic policy needs
- ***Service Support*** covers the selection, identifier assignment, deployment, default provisioning, and default configuration of managed entities in accordance with the enterprise's planned operations and policies
- ***Enabling Activities*** support both service delivery and service support

Service organizations may stand up a variety of functional teams that focus on planning and deployment, provisioning, configuration and report analysis, and monitoring and incident handling, with manager systems equipped for information fusion, operations coordination, analyses, report generation, planning and policy creation.

Beyond the simpler task of maintaining status information such as link status or service up/down status, enterprise management may include complex service arrangements involving multiple, orchestrated services. Additionally, coordinated help-desk support and reporting are needed. The DoD NetOps concept is addressing some of these topics.



I1219: Example Enterprise Management Decomposition

The following subsections describe in more detail the Service Delivery, Service Support and Enabling Activities modules in the *Example Enterprise Management Decomposition* diagram (I1219).

Service Delivery

Service Level Management

Service Level Agreements (**SLAs**) specify performance requirements, measures of effectiveness, reporting, cost, and recourse in a contractual agreement between service providers and consumers.

Capacity Management

This aspect of service delivery manages the ability to provide services in order to meet the level of performance specified in SLAs. Faults of various kinds can disrupt service delivery capacity and thus require active management.

Fault Management

Fault management constitutes the activities of identifying, analyzing and handling faults; in other words, recognizing when performance is so out of expected or relied upon range that policy dictates reaction. Performance metrics collected as part of the Enabling Activities provide data to support analysis. Fault management is the process of defining threshold policy constructs that cover unacceptable behavior (refer to the Enterprise Security [P1332]-related perspectives for additional information), starting with unacceptable performance. The two (unacceptable behavior and unacceptable performance) overlap in their common need for enabling technologies such as standardized threshold policy constructs, event logging and in those policies surrounding availability when poor performance can constitute a denial of service attack.

Service Support

Service Desk

A Service Desk contributes to service support by monitoring and responding to situations which impact performance, integrity, faults, accounting and attribution aspects of service delivery.

Incident Management

An incident is any event which is not part of the standard operation of the service and which causes, or may cause, an interruption or a reduction of the quality of the service.

The objective of incident management is to restore normal operations as quickly as possible with the least possible impact on either the business or the user at a cost-effective price.

Inputs for incident management mostly come from users, but inputs can have other sources as well such as management information or detection systems. The outputs of the process include change requests, resolved and closed incidents, management information, and communication to the customer.

Problem Management

Problem Management is the process responsible for managing the life cycle of all problems. The primary objectives of problem management are to prevent incidents from happening and to minimize the impact of incidents that cannot be prevented.

Configuration Management

Configuration Management relies on the persistent and continually updated storage of information about the elements that an organization uses in the provision and management of its **information technology** (**IT**) operations and management. Classically, this information base is implemented as a database; hence, the Information Technology Infrastructure Library (ITIL) term ***Configuration Management Database*** (***CMDB***). This is more than just an asset register, as it usually contains information that relates to the maintenance, movement, and problems experienced with Configuration Items (CIs).
The CMDB also holds a much wider range of information about items upon which the organization's operations and management depend to include the following:

- ***Hardware***
- ***Software***
- ***Documentation***
- ***Personnel***

Configuration Management essentially consists of four tasks:

- ***Identification*** - the specification, identification of all IT components and their inclusion in the CMDB
- ***Control*** - the management of each CI, specifying who is authorized to change it
- ***Status*** - the recording of the status of all CIs in the CMDB and the maintenance of this information
- ***Verification*** - the reviews and audits to ensure the information contained in the CMDB is accurate

Without the definition of all configuration items that provide an organization's operations and management, it can be very difficult to identify which items are used for which services. This could result in critical configuration items being stolen, moved or misplaced, affecting the availability of the services dependent upon them. It could also result in using unauthorized items in the provision of operations and management.

Note: *Configuration Management (CM) does not require a database, which is a particular architectural choice. CM in network and Web environments is often done with either directory service registries or search-based discovery services, and the results are not necessarily stored in a database.*

Assets and Resources

The essence of configuration management is to inventory and identify a Node's technology and information component assets and group them into recognized operational assets. Assets come in many types and each service concentrates on those in support of particular concept of operations (CONOPS). The Air

Force, for example, recognizes the following asset categories (refer to Air Force Doctrine Document 2, *Operations and Organization*, 3 April 1997 and AFDD 2-5.1, *Electronic Warfare*, 5 November 2002):

- Command and Control (C2) and Force Protection
- Intelligence, Surveillance and Reconnaissance (ISR)
- Inter- and Intra-theater Air Mobility
- Air and Space
- Electro-magnetic Spectrum Control

Information System assets are less obviously traceable; however DoD Directive 3020.40, *Defense Critical Infrastructure Program (DCIP)*, 19 August 2009 specifies any distinguishable network entity that provides a service or capability as an infrastructure asset.

Hardware historically has been the basis for managing assets (as materiel or facilities). Increasingly, however, infrastructure and mission software and services are becoming distinguishable assets and defining them as infrastructure simply because their network hardware address distinguishes them is increasingly insufficient. Net-centric operations and service-oriented approaches have demonstrated the limits of treating software in much the same way as hardware and treating the shrink-wrapped package as the asset instead of the capability the software provides.

Identifiers

Uniform Resource Identifiers (URIs) are a basic pre-requisite to Node manageability. Identifiers often provide more than a distinguishing attribute; they often overload the identifier with metadata about the named entity's functional decomposition (as in structured identifiers). Using a particular naming authority (for example, mailto), clarifies the requisite Transport and other Node decomposition infrastructures. For example, the mailto authority defines the user environment rendering of email messages, the computing infrastructure processing and storage data types, and optionally, the cryptographic infrastructure encoding information to expect.

Asset Types and Metadata

Overloading an identifier with all possible current and future metadata about an asset's type, especially when the asset types were produced under multiple authorities, proved infeasible and to the creation of an easily extendable standard framework for specifying standard management metadata, the Common Management Information Service (CMIS). This particular encoding was too processing intensive and essentially has been replaced by the simpler tabular encoding of the SNMP Structure of Management Information (SMI) approach. Subsequently, the ASN.1 protocol encoding of both CMIS and SMI became so optimized as to make it unmanageable by humans. This led to a proposal to use the more readable XML encoding of the Distributed Management Task Force (DMTF) Common Information Model (CIM) instead. The NetOps community deemed that the XML performance was too poor; BinaryXML encoding of the management information model and protocol is currently under discussion for both SNMP SMI and DMTF CIM protocols. Consequently, typed asset identifiers for software packages are still used in common practice. See the Java EE Deployment Descriptors [P1037] perspective for a detailed discussion and recommendations of one such use case. Attention to interoperability between computing infrastructure structured, type-encoded identifiers such as file extensions and Management identifiers such as XML strings will pay off in seamless management operations.

Asset Types and Unique IDs

All asset identifiers must provide the ability to distinguish an asset from any other asset within the management domain. Since the size and population of that management domain cannot be determined except in the field, asset identifier size requirements must be sufficiently large to provide a suitable namespace and mechanisms to extend that space if necessary. In addition, political authorities structure the global asset namespace, starting at the global level with the Internet Corporation for Assigned Names and Numbers (ICANN);[R1314] this is most evident in the allocation and assignment of unique instance IDs. Finally, asset management systems must be sized to cover and support the potential inventory types and total number of instances.

Versioning

Version identifiers are also necessary, given that assets may evolve over time without substantially changing capability or deployed role while changing in at least some sufficiently important particular. Unfortunately, this automatically sets up a potential conflict between component vendors who wish to highlight each improvement for marketing purposes and configuration and change management personnel who wish to minimize the amount of acceptance interoperability testing. The latter community has attempted to provide version numbering standards, but they are best practices and often limited to particular component types.

Change Control Management

Change control management uses a formal process to ensure that the introduction of changes to a system is in a controlled and coordinated manner. This process includes assessing all changes for risks and assessing the potential business impacts should a change produce undesired results.

If change control management procedures are not effective, unauthorized changes to operations and management may result. This could have major business impacts, including financial loss, customer loss, market loss, litigation, and in the worse case scenario, even collapse of the business that the operations and management are there to support.

In addition to change management of versioned releases and their patches, the configuration change management community distinguishes between deployment and provisioning, in order to separate the processes centered around hardware acquisition and physical configuration from the processes centered around enabling, activating and other software-based configuration changes, respectively.

Deployment

Deployment generally refers to those management activities, processes and data concerned with acquisition, especially capital expenditure governance, and physical installation and configurations.

Provisioning

Provisioning generally refers to those management activities, processes and data concerned with allocation and assignment of infrastructure, shared or common resources, especially the accountability, charge back and customer management aspects, and virtual asset configurations.

Software Asset Management

Software Asset Management (SAM) is the practice of integrating people, processes and technology to allow software licenses and usage to be systematically tracked, evaluated and managed. The goal of SAM is to reduce IT expenditures, human resource overhead and risks inherent in owning and managing software assets.

SAM includes maintaining software license compliance; tracking the inventory and usage of software assets; and maintaining standard policies and procedures surrounding the definition, deployment, configuration, use and retirement of software assets. SAM represents the software component of IT asset management, but SAM also is intrinsically linked to hardware asset management by the concept that ineffective inventory hardware controls significantly inhibit efforts to control the software thereon.

Patch Management

Patch Management is an area of systems management that involves acquiring, testing, and installing multiple patches (code changes) to an administered system. Systems can include servers, routers, personal digital assistants (PDAs), etc. Patch management tasks include maintaining current knowledge of available patches, deciding what patches are appropriate for particular systems, ensuring that patches are installed properly, testing systems after installation, and documenting all associated procedures, such as specific required configurations.

Patches sometimes are ineffective and can cause more problems than they fix. System administrators can take simple steps, such as performing backups and testing patches on non-critical systems prior to installations, to avoid problems caused by unintended side effects of patches.

Release Management

Release Management is the process that encompasses the planning, design, build, configuration and testing of hardware and software releases to create a defined set of release components. Release activities also include the planning, preparation, scheduling, training, documentation, distribution and installation of the release to many users and locations. Release Management uses the controlling processes of Change and Configuration Management.

Enabling Activities

Logging

Log Management

A log is a record of the events occurring within an organization's systems and networks. Logs are composed of log entries; each entry contains information related to a specific event that has occurred within a system or network. Originally, logs primarily supported troubleshooting problems. Logs now serve many functions within organizations, such as optimizing system and network performance, recording the actions of users, and providing data useful for investigating malicious activity. Logs have evolved to contain information related to many different types of events occurring within networks and systems. Within an organization, many logs contain records related to computer security; common examples of these computer security logs are audit logs that track user authentication attempts and security device logs that record possible attacks.

Audit Log

An Audit Log is a record of transactions in an information system that provides verification of the activity of the system. The simplest audit trail is the transaction itself. For example, if a person's salary is increased, the change transaction includes the date, amount of raise and name of authorizing manager.

A more elaborate audit trail can be created when the system is verified for accuracy; for example, samples of processing results can be recorded at various stages. Item counts and hash totals verify that the system has processed all inputs.

An audit trail can include any activity whatsoever, but transactions that do not effect a change are often not recorded. For example, ad hoc searches and database look-ups may not be identified in an audit trail, and routine queries are typically exempt from auditing.

Auditing

Every operating system (OS) includes security features and vulnerabilities which vary from OS to OS and sometimes between versions of the same OS. The security features are designed in such a way that they can be turned on or off and set to high security or low security, depending on the purpose for which the user intends to use the OS. In most cases, the default settings are not designed for high security. It often is up to the user to enable the security features to the desired level of security for that installation.
The process of auditing OS security includes evaluating whether the security features have been enabled and the parameters have been set to values consistent with the security policy of the organization and verifying that all users of the system (user IDs) have appropriate privileges to the various resources and data held in the system.

Metric Collection

Collection of metrics is a prerequisite for good performance analysis. Metrics are a key component in enabling functionality for the modules in the Example Enterprise Management Decomposition figure (I1219) included in this perspective. Multiple open standards define common infrastructure metrics for many categories such as in the following examples:

- Transport metrics defined as part of a component's Management Information Base (MIB) counters, for example RFC 2863 interface counters
- Various specification benchmarks define computing infrastructure metrics

Performance Metrics

Node and component performance, both infrastructure and mission-oriented, have an impact on net-centric operations. In a dynamic environment, where information exchange sources may not be infrastructure service providers, infrastructure metrics can be a key factor in the selection of service and information sources. Performance metric metadata, when advertised externally and frequently updated, allow potential service users to compare and select an implementation that meets their performance requirements, such as a measurement of reliability. Metrics are needed also to determine if performance has been supplied according to more traditional Service Level Agreements and for common infrastructure operations management.

Standard instrumentation for the collection of performance metrics of Nodes and components is necessary for management interoperability. Metrics should be visible and accessible as part of component service registration and updated periodically. See the Instrumentation for Metrics [P1163] perspective for more detailed information.

Performance Parameters and Ranges

Performance metrics are constituted from a combination of the base parameter type and its nominal (native default) range of values, for example a process execution counter. Simply collecting and monitoring such metrics may be sufficient for simple performance management; such metrics are so common as to be the default in the management information constructs such as SNMP MIBs and the DMTF CIM. In larger or more complex systems, performance metrics may include policy constructs that define the expected and reasonable ranges of performance parameters and increment, for example, a high- or low-watermark counter when exceeded, to aid in future capacity planning and even immediate adaptation activities.

Fault Thresholds and Policies

When the nominal or expected range of a performance parameter is far exceeded or exceeded for an unduly long time, most components management information models include thresholds: policy constructs that define alert or alarm events. In addition, there is an enabling event and logging infrastructure that generates event messages, sends them to the appropriate management system for logging, correlation, analysis, and potentially triggers corrective or adaptive reactions.

Web Service Metrics

Descriptions of some sample metrics that may be appropriate for **Web services** are in the Instrumentation for Metrics [P1163] perspective.

Best Practices

- BP1688: For **Services Management**, use an interim solution based on standardized Simple Network Management Protocol (SNMP) agents or other locally provided instrumentation and external monitoring tools.

P1153: Node Computing Infrastructure

The computing infrastructure of a Node can include several major subsystems: processing hardware, operating systems and other process execution constructs as well as storage hardware, file systems and other data storage constructs. Historically, these subsystems were inseparable thanks to their shared backplane or bus; over time, faster transport infrastructures radically extended this older architecture model, enabling computing infrastructure modularization, standardization, and virtualization. One result is the Virtual Machine (VM) construct; another is the Computing Infrastructure Enterprise Service construct. As a result, interoperability is increasingly a primary concern. Furthermore, modularized, standardized and virtualized computing infrastructures create many non-obvious impacts on the management of a Node, especially performance and fault management; there are other non-obvious impacts on the security of a Node.

Computing Infrastructure Management

The initial concern of a modularized computing infrastructure is bootstrapping its assembly, starting with powering up a core subset of processing, storage, user interface and transport (backplane) hardware, firmware and core software. For a more detailed discussion, see the Remote Management [P1394] perspective.

Distributed Computing Infrastructure

Initially, computing infrastructure distribution used the Client and Server constructs; increasingly, computing infrastructure distribution virtualization is using Web infrastructure constructs. For more detailed discussions, see the Web Client Platform [P1154] and Web Infrastructure [P1157] perspectives.
In order to support the new net-centric constructs, discovery services such as directories and search engines are augmenting and even replacing file system namespaces as the primary discovery interface. For more detailed discussions see the Domain Directories [P1162] and Service Enablers [P1325] perspectives.

Security Considerations

Modularized computing infrastructure assemblies can be resilient; compromised components are easier to quarantine, and defense in depth is easier to specify and develop. Composite assemblies require configuration beyond what a simpler standalone component requires. In particular, see the Authorization and Access Control [P1339] and Identity Management [P1178] perspectives for more information on security aspects.

Management Considerations

For general management concerns see the Enterprise Management [P1330] set of perspectives. Modularized computing infrastructure assemblies can be robust (i.e., faulty components are as easy to swap out as security-compromised ones and provision of additional capacity in response to surges in demand is much easier to specify and develop), but composite assemblies require configuration beyond what a simpler standalone component requires. Discussions on making tradeoffs among performance, scalability, robustness and manageability are in several detailed perspectives including Time-Critical Operations [P1395], Traffic Management [P1356], Planning Network Services [P1357], Design Tenet: Differentiated Management of Quality-of-Service [P1265] and DDS Quality of Service [P1192]. Transport Quality of Service is particularly important where Local Area Networks or even the Internet have replaced the computer bus or backplane.

Detailed Perspectives

- Virtual Machines [P1390]
- Web Client Platform [P1154]
- Web Infrastructure [P1157]
- Domain Directories [P1162]
- Instrumentation and Metrics [P1163]
- Time-Critical Operations [P1395]
- Remote Management [P1394]
- Host Information Assurance [P1161]

P1390: Virtual Machines

Virtualization creates a simulated computer environment, a ***virtual machine***, for its guest software, including complete operating systems. The guest software executes as if it were running directly on the physical hardware, although managing access to physical system resources (such as the network access, display, keyboard, and disk storage) may be at a more restrictive level than access to the processor and system memory. Virtual machines have benefits for both servers and clients.

Virtualization provides many advantages over deploying servers on traditional hardware. By encapsulating a server in a virtual machine, administrators can more easily replicate and manage servers in a large hosting environment. In many cases, servers do not require 100% of their resources 100% of the time. This permits a server to share its physical resources across many different virtual machines at the same time, reducing the number of physical servers and thus power, space, and maintenance requirements in the data center. The virtualization software products are usually very modestly priced, and some are free. However, each virtual machine requires its own individually licensed operating system; this is not much of an issue for enterprises either having enterprise operating system licenses, or using free versions of the Linux operating system.

Virtualization also provides an advantage over the traditional server procurement model. Provisioning a new virtual server can occur very quickly, allowing a program to begin development work immediately.

Client machines can benefit from virtualization, too. Virtualization is a very effective way to have multiple different operating systems on one client host. A specific virtual machine with its operating system and unique applications may be selected at boot time (as is possible without virtualization). Alternatively, multiple virtual machines with different operating systems can be executing at the same time. In that case, data can be transferred among the different operating systems and the applications running on them. Because each virtual machine is isolated from the others (contemporary processor products provide hardware support for that), faults in the operating system or its application(s) cannot affect other virtual machines and their software. This allows for easier software development, and for testing the compatibility of new applications with current applications. Another benefit of virtual machines is that malware in one virtual machine cannot infect the other virtual machines.

Currently, there are emerging standards relating to virtualization that vendors are adopting which will support portability across various vendor products. One such emerging standard is the **Open Virtualization Format Specification** managed by the Distributed Management Task Force (DMTF),

Best Practices

- BP1904: Use **Open Virtualization Format** (**OVF**) for all virtual machines
- BP1905: Digitally sign all **Open Virtualization Format** (**OVF**) virtual machines.
- BP1906: Only run **Open Virtualization Format** (**OVF**) virtual machines with a valid digital signature from a trusted source.

P1154: Web Client Platform

Web clients (both desktops and **servers**) should be capable of accessing **Java Platform, Enterprise Edition** (**Java EE**) **services** and **.NET** services; service developers are free to choose the best technology for their service.

Two key elements of the standard frameworks follow:

- Browser [P1155]
- Common Access Card (CAC) Reader [P1156]

Guidance

- G1613: Prepare a **Node** to host new **Component services** developed by other Nodes or by the **enterprise** itself.

Best Practices

- BP1614: Plan a contingency response to the **Node** becoming a new **component service** within another Node.
- BP1672: Be prepared to integrate fully with the **Information Assurance** (**IA**) infrastructure.
- BP1673: Be prepared to integrate fully with the **Enterprise Management Services** (**EMS**) infrastructure.

P1155: Browser

Web browsers are fundamental to the DoD vision of net-centric information sharing and access to distributed **services**. Because **Global Information Grid** (**GIG**) interoperability partners may not be known a priori, Web browsers should support a wide breadth of browser technologies, such as **JavaScript**, Java **applets**, and **plug-ins**.

Configure Web browsers in accordance with the Web Server **Security Technical Implementation Guide** (**STIG**), Desktop Applications STIG, and Windows 2003/XP/2000 Addendum STIG.

Best Practices

- BP1615: Select **Web browsers** that support a wide breadth of current browser extension technologies.
- BP1674: Configure in accordance with the applicable **Security Technical Implementation Guides** (**STIGs**).

P1156: Common Access Card (CAC) Reader

Smart Cards provide greatly increased security for multiple applications. The usefulness of a smart card is based on its intrinsic portability and security. A typical smart card has the same dimensions as a standard credit card and appears to be very similar with the exception of a set of gold contacts. When inserted into a reader, these contacts provide power to a microprocessor located on the smart card; the smart card is thus able to store and process information, in particular cryptographic keys and algorithms for providing digital signatures and for use with other encryption. A major impediment to the widespread use of smart cards has been interoperability. Unfortunately, smart cards are currently not vendor interoperable and therefore must use specific software and smart card readers. This is an issue that the **National Institute of Standards and Technology** (**NIST**) **Information Technology Laboratory** (**ITL**) is addressing. Smart cards can enable **identification** for accessing computers; see the Smart Card Logon [P1315] perspective.

Guidance

- G1619: Configure **clients** with a **Common Access Card** (**CAC**) reader.

P1157: Web Infrastructure

A **Web** infrastructure allows software developers to deploy Web-enabled applications, **services** and other software in a Node. While many Web infrastructures exist, most software will converge on popular platforms or technologies (e.g., Apache; **Java Platform, Enterprise Edition**; **.NET**; etc.). The Node should provide common shared Web infrastructures for software deployments to minimize unnecessary duplication of these common environments. A common Web infrastructure will also allow Nodes to provide better integration with local **Information Assurance** (**IA**) and **Enterprise Management Services** (**EMS**) infrastructures as well as **CES** and **COI services** available both internally and externally to the Node.

Address the following three major elements of Web infrastructure at the Node.

Web Portal

A Web **portal** provides an environment for hosting small Web applications called **portlets**, and allows for content selection, arrangement and other visual preferences tailored to each user. Though not strictly essential for **Global Information Grid** (**GIG**) interoperability, it is reasonable that some GIG net-centric services and applications will provide portal-based Web applications that Nodes may want to host locally. To reduce issues of portability, Web portals provided by the Node should support widely accepted standards such as **JSR**-168 and **Web Services for Remote Portlets** (**WSRP**). However, because commercial products also provide non-portable proprietary interfaces, there is a risk of needing multiple Web portal products or reengineering the portlet to work on an existing Node portal. See the Web Portals [P1077] perspective in NESI Part 5 for additional information.

Web Server

Web server technology is becoming fundamental in making information visible and accessible to external Global Information Grid (GIG) users. The most significant barrier to interoperation is security. Making information accessible to a community of users as large as the GIG necessitates the implementation of **authentication** and **authorization** technology that is sufficient to prove a user's identity and that is scalable, respectively. Web servers should provide DoD **Public Key Infrastructure** (**PKI**) based authentication and role based authorization mapped to **certificate** attributes as described in the applicable **Security Technical Implementation Guides** (**STIGs**). Eventually, the container should integrate with the **Net-Centric Enterprise Services** (**NCES**) Security Service, when available. In the interim, base authorization on the **Electronic Data Interchange - Personnel Identifier** (**EDI-PI**) contained in the PKI certificate attributes. The use of the EDI-PI as the attribute on which to base authorization decisions is a matter of debate and ongoing engineering, as there are issues about the issuance of EDI-PI to certain user populations, such as coalition users. In the absence of an EDI-PI attribute, use other attributes for authorization decisions.

Web Application Container

Web application containers provide an environment for serving full, interactive application functionality and services on the Web. There are two major container technologies: **Java Platform, Enterprise Edition** (**Java EE**) and **.NET**. NESI expresses no preference regarding which of the two technologies to use; *NESI Part 5: Developer Guidance* addresses both (see, for example, Java EE Deployment Descriptors [P1037] and .NET Framework [P1086]).

The design and implementation of a Node's Web infrastructure should accommodate both Java EE and .NET. The rationale for this is that Nodes will likely have to host services locally and applications that were developed externally using either technology. Use Web services (**SOAP**, **XML**, etc.) to interoperate between Java EE and .NET applications or services. Such interoperation may be required, for example, when **orchestrating** Web services across Nodes as part of a Joint mission thread.

As is the case with Web servers, application containers should provide DoD Public Key Infrastructure (PKI) based authentication and role based authorization mapped to certificate attributes as described in the applicable STIGs. Eventually, the container should integrate with the NCES Security Service when available. In the interim, base authorization on the Electronic Data Interchange-Personnel Identifier contained in the PKI certificate attributes. The use of the EDI-PI as the attribute on which to base authorization decisions is a matter of debate and ongoing engineering, as there are issues about the issuance of EDI-PI to certain user populations, such as coalition users. In the absence of an EDI-PI attribute, use other attributes for authorization decisions.

The Web application container should be capable of processing Web services protocols in accordance with the **Web Services Interoperability** (**WS-I**) Basic Profile. The container should also support XML security protocols including XML Encryption, XML Signature, and XML Key Management. These protocols are used in protecting content within an XML document that may be passed among multiple orchestrated Web services.

Guidance

- G1621: Provide a Node Web infrastructure for all **Components** within the Node.

Best Practices

- BP1675: In the Node's Web infrastructure, support the technologies and standards used by the **CES** services under development as well as any technologies and standards used for **Community of Interest** (**COI**) services.
- BP1677: Consider using Web **proxy** servers and load balancers.
- BP1707: Configure and locate elements of the Node Web infrastructure in accordance with the Web Server **Security Technical Implementation Guide** (**STIG**).
- BP1708: Configure and locate elements of the Node Web infrastructure in accordance with the Desktop Applications **Security Technical Implementation Guide** (**STIG**).
- BP1709: Configure and locate elements of the Node Web infrastructure in accordance with the Network **Security Technical Implementation Guide** (**STIG**).
- BP1710: Support appropriate and widely accepted standards for Web **portals** provided by the Node.

P1162: Domain Directories

Within and across Nodes, directory technologies such as Microsoft **Active Directory** (**AD**) or OpenLDAP are tools for system, network, and security administration. Many options exist on how Nodes employ these tools; however, interoperability issues can arise between **Global Information Grid** (**GIG**) Nodes if sub-enterprises employ these tools differently (even within the same technology family, such as AD).

The ***DoD Active Directory Interoperability Working Group (DADIWG)*** is forming guidance on Active Directory implementation.

Implement Active Directory (AD), if used, in accordance with the recommendations of the DADIWG; also, periodically monitor the DADIWG Web site (use restricted to .gov and .mil domains) for the status of GIG implementation issues.

Best Practices

- BP1679: Implement a Node that uses **Active Directory** (**AD**) in accordance with the recommendations of the DoD Active Directory Interoperability Working Group (DADIWG).

P1163: Instrumentation for Metrics

Performance has an impact on net-centric operations. Instrumentation is a term frequently used in association with the generation, collection, and analysis of performance metrics. In a dynamic environment, where **services** and information exchange partners may be dynamic, metrics can be a key factor in the selection of services. Performance metrics that are advertised externally and frequently updated allow potential service users the ability to select an implementation that meets their performance requirements, such as a measurement of reliability. Metrics are normally also needed to ensure performance is provided according to more traditional **Service Level Agreements** (**SLAs**) and for operations management.

Instrument **component** services that a Node exposes to the **Global Information Grid** (**GIG**) to collect performance metrics. Metrics should be visible and accessible as part of the Component service registration and updated periodically. Appropriate GIG working groups have not yet defined standards for metrics but should do so at some point in the future .

Some sample metrics that may be appropriate for **Web services** are in the following table.

SLA Metric	***Metric Description***
Availability	How often is the service available for consumption?
Accessibility	How capable is the service of serving a client request now?
Performance	How long does it take for the service to respond?
Compliance	How fully does the service comply with stated standards?
Security	How safe and secure is it to interact with this service?
Energy Efficiency	How energy-efficient is this service for mobile applications?
Reliability	How often does the service fail to maintain its overall service quality?

Best Practices

- BP1680: Instrument **component** services that a Node exposes to the **Global Information Grid** (**GIG**) to collect performance metrics.
- BP1681: Make metrics for **component** services visible and accessible as part of the service registration and update the metrics periodically.
- BP1867: Use metrics to track responsiveness to user information sharing needs.

P1395: Time-Critical Operations

Network-based systems consist of physically dispersed nodes. Especially for network centric warfare, they have inherent dynamic uncertainties at all levels from the mission, to the mission environment, to the system, to the applications. One notable example is that the system is often overloaded and yet must do the best it can according to application- and situation-specific criteria. Another is that the network almost always has variable and unknown network latencies and bandwidths. A third is that node connectivity changes (e.g., as mobile nodes make and lose wireless contact with the network, and as natural obstacles and hostile damage partition the network). Physically dispersed network-based systems and their applications usually operate in time frames from about a second to minutes. This requires the use of powerful resource management techniques to deal with complex dynamic situations. The given time frames limit, but do not eliminate, the role of off-line scheduling and planning techniques (e.g., based on linear programming, searching).

Due to network latency, bandwidth limitations, processing delays, and other factors, things that happen at any given Node take some time to become visible elsewhere in the system. Since most networks are graphs, the communication of happenings may be along different inter-node routes having different latencies; thus, the order in which the affected Nodes observe events is often not the same as the order in which they actually happened. It is often necessary to provide some mechanism that preserves the actual order of certain events at the affected observer nodes. In non-time-critical systems, it is normally sufficient for the Nodes to agree on an order of events, whether or not it is the actual order.

Time-Critical vs. Real-Time

"Time-critical" is a generalization of the conventional term "real-time" to accommodate timeliness despite dynamic uncertainties, especially in but not limited to distributed (e.g., network-based, such as for network centric warfare) systems. Traditional real-time systems are deliberately focused on static, primarily centralized, small scale subsystems (e.g., signal processing, digital avionics flight control).

One of the dynamic aspects of non-trivial network-based systems is an underlying presumption that they can, and often do, partially fail. Presume that fault and failure detection and recovery are the normal operational case, not an abnormal exception case added onto the design at the end. In a time-critical system, this detection and recovery is also time-critical, which is a major technological challenge.

Most traditional real-time systems are highly physically centralized. If a real-time system has multiple Nodes, they typically share knowledge about the state of the system and manage the system through a shared global memory whose latency is very low compared to the rate at which the nodes change state. Increasingly, larger scale real-time systems are built with multiple processors on one or more blades in a chassis, using a processor interconnection whose magnitude and variance of latency on or among the blades are higher and sometimes more variable than those of shared global memory but still orders of magnitude less than those in network-based systems. It usually is possible to carry over conventional centralized real-time resource management to such "slightly decentralized" systems. Traditional real-time systems also limit themselves to the vicinity of microsecond to millisecond time frames. Those time frames are extremely uncommon in network centric warfare systems and lend themselves only to relatively easy real-time resource management concepts and techniques that do not scale up to time-critical dynamic network-based systems. Static presumptions of real-time systems assure that the system is under-loaded such that all its timeliness constraints can be satisfied, so long as those presumptions are correct. Most real-time systems are relatively monolithic functionally. Some are expected to fail monolithically; others have (usually low level) fault detection and recovery mechanisms, but most of these take place in real-time only to the extent that well-understood redundancy techniques are used to mask faults.

Moreover, the real-time system practitioner community (vendors, users) has little consensus on the concepts and terminology of the field resulting in various vague and often contradictory interpretations. This complicates the procurement of real-time systems having the desired properties. Even the real-time research community has a consensus on the formal definition of only one concept, the special case of "hard real-time," but it bears little resemblance to the varying uses of that term by practitioners, leaving the vast body of deployed real-time systems (which are the general case of soft real-time ones) without precise definitions. That hinders research progress on the majority of open problems in the design and implementation of practical real-time and time-critical systems, especially dynamic network-based ones.

Much of this perspective on time-critical systems also applies to the special case of traditional real-time ones.

Operation Timeliness

An operation is time-critical if it has a timeliness constraint that is one of its correctness criteria, not just a performance metric. A timeliness constraint consists of two parts:

- a constraint on the completion time of the operation (e.g., meet a deadline, miss a deadline by no more than 20%)
- a constraint on the predictability of the completion time of the operation (e.g., always meet the deadline, meet the deadline with probability greater than 0.85)

An operation is a non-time-critical one if it does not have a timeliness constraint. The performance of operations usually occurs at times which maximize the throughput of operations in the system (throughput is a system performance metric, not an operation metric). Non-time-critical systems, such as transaction processing systems, are transformational because they transform input data to output data.

An operation completion time constraint specifies the utility to the system gained or lost depending on when the operation completes. A deadline is the most familiar completion time constraint. This simple constraint is the least common one in actual practice, especially in network-based systems. The traditional real-time interpretation of a deadline is that utility is binary (e.g., meeting the deadline yields some application-specific positive utility and missing the deadline yields 0 or infinitely negative or application-specific negative utility). This interpretation is mistakenly referred to as a ***hard*** deadline, implying that it cannot be missed (otherwise it is referred to mistakenly as a ***soft*** deadline), but that decision is properly specified by the predictability part of a timeliness constraint. Deadlines are used more generally in the scheduling field (manufacturing, logistics, etc.) to include utility that depends on earliness (e.g., in some cases earlier is better, and in others too early is counterproductive) and tardiness (e.g., a little tardy is acceptable but too much is not). This richer and more expressive interpretation is necessary in time-critical network-based systems.

More general operation completion time constraints do not involve deadlines at all. The utility to the system of completing an operation may be expressed as a naturally derived (i.e., mission-, system-, or situation-specific) function of when it completes. A deadline is the special case of a downward step function. As time passes, the utility of completing any particular operation may increase or decrease as specified by that function. Such operations are scheduled to execute in an order that maximizes the expected total utility of completing them.

An operation is time-critical to the degree that its completion time is acceptable, and the predictability of that completion time is acceptable for correctness with respect to the application- and mission-specific purpose of the operation at that time. Note that the time frames (e.g., microseconds, hours) for the durations and completion times of operations are not relevant to how time-critical an operation is, only the acceptability of the completion time and predictability of completion time are relevant.

Operation timeliness is a correctness criterion for time-critical systems because these systems are reactive; they interact with (e.g., control) their physical environment, such as sensors and weapons. Unacceptable operation timeliness, either completion time or predictability of completion time, may cause incorrect results or constitute a failure of the operation and everything in the mission that depends on that operation being acceptably timely.

To contrast the timeliness constraints of conventional real-time systems with the general case of time-critical systems, disregard the plethora of inconsistent and imprecise concepts and terms in the real-time practitioner community in favor of the one precise term for which there is consensus in the real-time research community: hard real-time, which is exactly a special case of the general case of time-critical systems. One part of the timeliness constraint is real-time's usual binary-valued deadline operation completion time; a second part is that meeting the deadline is deterministic (given the usual caveats that determinism requires; see below). Clearly it seems easier (albeit perhaps artificially so) to presume that all operations have deadlines, than it is to extract the operation's actual completion time constraint from the nature of the physical environment with which it reacts. Likewise, it seems easier simply to require a priori knowledge that all deadlines will be met than to understand the actual predictability requirement. In traditional real-time systems where this timeliness constraint is natural and correct, specifying, designing, implementing and testing the system is relatively easier than for general time-constrained systems, especially network-based ones, but many experiences demonstrate that when the classical hard real-time timeliness constraints are applied arbitrarily to operations in dynamically uncertain time-critical systems, cost-effectiveness and success are diminished.

Consider the hypothetical missile defense scenario of sending course updates to a guided interceptor missile. The updates are not inherently periodic with deadlines, their natural occurrence and utility vary depending on application-specific factors such as how far away the hostile missile is from its intended target, how close the interceptor missile is to the hostile missile, the relative threat of the hostile missile (e.g., based on its destination),

etc. As the hostile and interceptor missiles get closer together, the interceptor course updates have more utility and need to be more frequent. In addition, the defense system is usually overloaded, so its resources, such as the computational and networking ones needed to track hostile and interceptor missiles and send course updates to interceptor missiles, should be used to satisfy the mission's measures of effectiveness, such as intercepting as many as possible of the most important (i.e., threatening) hostile missiles. Consequently, an interceptor missile may not follow the shortest direct path to the hostile missile; instead, as system resources are allocated dynamically to accommodate the changing situation, it may follow a zig-zag course. Casting this scenario artificially as a hard real-time one requires more resources and results in a brittle system, compared with managing it as a dynamic time-critical system.

Predictability

Most systems have performance metrics that users want to be predictable. Informally, something is predictable to the degree that it remains known in advance despite possible future changes to it.

There is a spectrum of predictability. The most predictable end-point of the spectrum is determinism. A property (e.g., an operation completion time) is deterministic if is known absolutely in advance despite possible future changes to it; thus determinism requires a static context. The least predictable end-point of the predictability spectrum is where absolutely nothing is known in advance about the future state of the property. Thus, there are not degrees of determinism, it is binary; there are degrees only of predictability.

The information used for prediction is inevitably incomplete and inaccurate and hence modeled in some way, and the prediction is performed according to the rules of the model. The model defines the metric for the predictability spectrum and the definition of the end-points. The most common formal models for making predictions under uncertainty use classical probability theory to represent and manipulate uncertainties both in the observed or presumed information and in the predictions. In those models, the metric is some measure of variability (e.g., coefficient of variation), the most predictable end-point is the deterministic distribution (zero coefficient of variation), and the least predictable end-point is the extreme mixture of exponentials distribution (arbitrarily high coefficient of variation). Other models use entropy, Bayesian and other belief-based formalisms, imprecise probabilities, fuzzy logic, etc.

Predictability of properties in a non-time-critical system (e.g., of throughput) is for performance measurement.

Predictability of operation completion times is a fundamental factor in time-critical systems because they are reactive, and hence is part of a time-critical operation's timeliness constraint.

Often it is necessary to trade the two parts (completion time and predictability of the completion time) of a timeliness constraint off against one another. The operation completion time may be predicted, but with lower probability, to be more satisfactory; or the operation completion time may be predicted, with a higher probability, to be less satisfactory. This is analogous to the choice of either putting money in a mutual fund that might, with some lower probability, provide a higher return on the investment or putting money in a savings account that will, with probability one, provide a lower return on the investment.

Most traditional real-time systems choose to predict deterministically in advance whether or not the operation's completion time will meet its deadline. Some real-time systems choose the stronger deterministic prediction of what the actual completion time will be.

In time-critical systems, operation completion times are in general necessarily predicted non-deterministically (e.g., probabilistically) except for the special case of hard real-time. Usually, higher predictability of operation completion time is preferred at the expense of less satisfactory operation completion time. Predictability (e.g., of operation completion times) in time-critical systems, especially network-based ones such as for network centric warfare, is very technically challenging due to the dynamic uncertainties and the requirement for predictions to be timely.

Time-Critical System

A system is time-critical to the degree that its time-critical operations satisfy their timeliness constraints acceptably well under the current circumstances. Degree, in this sense, is application- and situation-specific, and may be at least partly subjective; there is no consensus on a formalization. The primary complication is that in general a time-critical system has a multiplicity of asynchronous time-critical operations contending for various sequentially shared resources (e.g., processor cycles, network access, sensor timeline). Design and implementation of the special

case of real-time systems (especially hard real-time ones as defined above) usually avoids (or at least reduces as much as possible) that complication.

There are two means for achieving the goal of acceptably satisfying operation timeliness constraints.

The first means is to over-provision resources (e.g., processing capability that can take place on a time scale sufficiently faster than the time scale of the time-critical operations). Sufficient over-provisioning can compensate for whatever dynamic uncertainties (e.g., resource arrivals, dependencies, etc.) may be present in the system and its environment that could impede operations from being time-critical to an acceptable degree. The same applies for over-provision of other resources such as networking, etc. Most real-time systems use this approach.

Over-provisioning of resources is often infeasible, due to size, weight, and power limits or the costs of the resources; this is especially true in a tactical environment. Logistical factors such as when upgrades to systems are scheduled (currently at least several years apart in most military systems) also affect the ability to over-provision resources.

The second means is to provide explicit time constraint-based resolution of contention for shared resources, for example, deadline-based scheduling. The choice between, or combination of, these two means is a system- and application-specific cost/performance engineering decision. This means is often the more difficult engineering one, but can produce systems that are much more explicitly responsive to the applications' and missions' timeliness requirements and hence more easily adapt to the dynamic changes that are inevitable in network-based warfare systems.

Time-Critical Operating System

An operating system (OS) is a software system intended to abstract and manage the physical resources of a hardware platform (e.g., processors, memory, and network interfaces) in a manner that eases writing the application and other software above the OS. A time-critical OS, like all other time-critical systems, is time-critical to the degree that its time-critical operations satisfy their timeliness constraints acceptably well under the current circumstances. The intent is that the degree of the OS operations' time-criticality is sufficient to facilitate acceptably satisfying the operation timeliness constraints of the higher level system(s) using the OS.

A real-time OS is a special case of a time-critical OS, similar to how real-time operations and systems are special cases. Unfortunately, as with all other concepts and terms in the field of real-time, the practitioner community has widely disparate opinions on the definition of a real-time OS. What almost all these opinions have in common is that two OS properties, interrupt response and OS service latencies, have known least upper bounds (essentially deadlines). Some real-time OS vendors provide a characterization, such as a histogram, of these latencies for specified conditions. It is common for the practitioner community to erroneously assert that these latency bounds must be less than some small number of microseconds or milliseconds for the OS to qualify as being a real-time one. "Real-time" is not the same as "real fast." Extant real-time operating systems have a wide variety of other features and properties; one class of these is various subsets of the Real-Time Extension within the UNIX specification, which includes ANSI/IEEE standards POSIX 1003.1b-1993 and POSIX 1003.1i-1995 (see, for example, the IEEE Xplore Digital Library at http://ieeexplore.ieee.org/Xplore/dynhome.jsp or the Open Group discussion at http://www.opengroup.org/onlinepubs/009695299/mindex.html).

Because the real-time research community applies its consensus on a definition of hard real-time, it comes closer to consensus on a definition of a hard real-time OS (i.e., essentially one that supports hard real-time operations above the OS). Hence, the OS performs some resource management statically at run-time, and the remaining resource management is performed a priori (off-line). A variety of resource management techniques and additional features appear in various research real-time operating systems. Some provide for sporadic or even non-real-time tasks to execute in the background if they do not interfere with meeting the hard real-time deadlines. However, virtually all dynamic functionality found in non-real-time operating systems is necessarily omitted.

Currently, no COTS operating systems, and only a few COTS time-critical middleware products (e.g., compliant with the Real-Time CORBA standard) support even deadlines much less more realistic operation completion time constraints. The system or application designers or users are forced to convert the deadlines into the priority space provided by the run-time infrastructure. In time-critical systems, deadlines are physical times in the real world and thus independent of each other, i.e., changing a deadline does not necessarily require changing other deadlines. Priorities are relative to one another, so changing a priority assignment usually causes a cascade of other priority assignment changes. The conversion of deadlines to priorities has consistently proven to be a major source of error, debugging, testing, and modification costs in non-trivial time-critical systems, especially network-

based ones. An example counter to popular misunderstanding of time-critical and particularly real-time systems is the Microsoft Windows XP operating system. All operating systems include drivers for attached physical devices, and those drivers have real-time (sometimes hard real-time) operations. That is not conventionally considered to qualify the operating system as a real-time one because the drivers are almost always invisible to the applications. But an operating system (such as Windows XP) does qualify as a time-critical (and sometimes even a real-time) one when it can meet the goal of its time-critical operations satisfying their timeliness constraints acceptably well. That is possible by over-provision of resources, when the time scale of its applications' time-critical operations is sufficiently greater (for example, several orders of magnitude) than that of the operating system (i.e., the magnitudes and variances of the operating system's service latencies).

Best Practices

- BP1915: Model time-critical operations.
- BP1916: Resolve contention among resources in a consistent manner.
- BP1917: Use cyclic executive scheduling if a real-time system requires that all operations have a priori start and completion times.
- BP1918: Use deadline-based algorithms for scheduling operations with deadlines.
- BP1919: Design systems presuming degraded environments are the normal case.
- BP1920: Design systems to have timeliness as a core capability.
- BP1921: Design systems to have fault management as a core capability.
- BP1922: Design systems to have security as a core capability.

P1394: Remote Management

Minimizing the time required for conducting administration and configuration management of computing infrastructure is critical to reducing the total cost of ownership, reducing downtime, and maintaining high levels of availability in large computing environments. One concept that supports this goal is the ability to power-on or wake-up a component, such as a computer, remotely. Another concept supporting this goal is the ability to provision and cascade operating system images, or other bootable images, over a network. There exist various standards and approaches to support these concepts; two supporting approaches are ***Wake on LAN*** (***WOL***) and ***Preboot Execution Environment*** (***PXE***), respectively.

Remote Power Management

WOL provides the ability to power-on or wake-up a component from a reduced power state over a network remotely. This capability allows administrators to initiate maintenance on components that are in a powered off or in a reduced power state, therefore saving an administrator from physically having to visit each component.

WOL works at the hardware level and is platform and operating system independent. WOL monitors the network (at the data link layer) looking for a special "magic packet" which triggers the component hardware to initiate to a powered-on state.

In order for WOL to function, a component must support WOL at the hardware level (i.e., WOL support within the component system board, basic input/output system or BOIS, power supply, and network interface); most computers (and many other network-enabled components) manufactured today support WOL functionality. For WOL to function, enable WOL functionality within each component, for example by setting a configuration parameter within a computer BIOS.

For a further discussion on WOL see the AMD Magic Packet Technology white paper.

Remote Bootable Image

PXE is a specification which provides a standard way to boot a computer using executable images served over a network. PXE has many uses including booting disk-less computers, performing data backups, updating computer firmware, and managing patches. The PXE specification leverages several existing standards such as **Dynamic Host Configuration Protocol** (**DHCP**) for identity management and network admission control and Trivial File Transfer Protocol (TFTP) for remote software configuration management. PXE environments often rely on WOL to provide remote power management capability.

A PXE-enabled computer executes a bootable image served from the network using the following process. PXE client firmware (generally located on the computer network interface) discovers a local DHCP server by broadcasting on the local network a request to access to the network and discover additional PXE services. Once a DHCP server that supports the PXE extensions responds, the PXE-enabled computer receives configuration information necessary to start operations (boot) or a pointer to the PXE boot (configuration) server. From the PXE boot server the PXE-enabled computer will download the bootable image via TFTP and execute it.

PXE remote management requires all three standard utility services: DHCP, TFTP and PXE, although they are often implemented as a single standard DHCP/TFTP/PXE client and the management services are co-located on the same physical server.

For further information see the Preboot Execution Environment (PXE) Specification.

Security Considerations

Securing WOL requires securing physical access to the hardware (both computing and network components) to prevent unauthorized configuration changes. Furthermore, it is important to protect network access, since most WOL implementations do not provide any form of authentication. This includes protecting physical access to networks when possible and controlling network access to prevent unauthorized WOL unicast and subnet-directed broadcast packets from reaching components.

Securing PXE requires securing physical access to the hardware (both computing and network components), and securing the network and utility services to include DHCP, TFTP and PXE servers. Securing DHCP requires additional utility services such as Dynamic Domain Name System (DDNS).

The PXE specification does not have provisions for detecting unauthorized PXE boot servers or unauthorized PXE-enabled client computers on a network. This provides the opportunity that an unauthorized system on the network can boot images provided by a PXE boot server and also allows for someone to impersonate a PXE boot server and provide unauthorized boot images to PXE-enabled client computers on the network. Therefore, it is essential to have properly configured network safeguards such as proper router configuration, firewalls, and intrusion detection systems to prevent and detect unauthorized PXE related activities.

See the following perspectives for more detailed information on securing remote management capabilities:

- Authorization and Access Control [P1339]
- Application Layer Protocols [P1355]
- Host Information Assurance [P1161]

Management Considerations

For general remote management concerns see the Enterprise Management [P1330] perspective.

Best Practices

- BP1933: Control access to firmware with strong passwords.
- BP1934: Disable Preboot Execution Environment (PXE) capabilities when not required.
- BP1935: Disable Wake on LAN (WOL) capabilities when not required.
- BP1936: Physically secure hardware components.
- BP1937: Disable component remote management capabilities when secure remote management is not possible.

P1161: Host Information Assurance

Host **Information Assurance** (**IA**) protections are part of the DoD Information Assurance Strategic Plan, which in turn is a part of the **Net-Ready Key Performance Parameter** (**NR-KPP**) that gets assessed during the **Joint Capabilities Integration and Development System** (**JCIDS**) acquisition process. Failure to implement host information assurance protections could jeopardize the approval for a Node to operate on the **Global Information Grid** (**GIG**).

Guidance

- G1622: Implement **commercial off-the-shelf** (**COTS**) software that protects against malicious code on each operating system in the Node in accordance with the Desktop Application **Security Technical Implementation Guide** (**STIG**).
- G1623: Implement personal **firewall** software on computers used for remote connectivity in accordance with the Desktop Applications, Network, and Enclave **Security Technical Implementation Guides** (**STIGs**).
- G1624: Install anti-**spyware** software on all Windows Desktop computers.

Best Practices

- BP1868: Incorporate mechanisms to enhance Computing Infrastructure (CI) availability.

P1341: User Environment

The user environment comprises those functions that are directly related to handling interaction with the users of the Node. The most obvious functions are those that represent the primary user interface to the Node. A wide spectrum of interfaces can be found as a primary user interface - from faceplate switch status on a piece of hardware through highly distributed software components commonly known as ***clients***. In addition to the primary interface, whole classes of peripheral functions also fit into the user environment. For example, biometric security feedback functions are one such class of peripheral interfaces.

Much of this perspective focuses on those functions that represent primary user interfaces to net-centric software based systems. These primary user interfaces are built on a collection of hardware devices and associated software.

The hardware components of user interfaces can be roughly divided into two categories: input and output devices. The types of input and output hardware devices available span a broad range but a few merit detailed discussion. The most common primary user environment hardware infrastructure consists of a keyboard for text entry, a mouse for pointing and selection, and a display monitor for visual depiction of both textual and image (still and video) data. For details refer to the Remote KVM Switch Connectivity [P1393] perspective.

The software components of user interfaces also span a spectrum of types and architectural patterns. Two primary patterns and one special pattern merit discussion: the thick client, the thin client, and the special pattern of **Web browsers**.

Thick Clients

Thick Clients are software or hardware that provide full or near full functionality without depending on other non-local software or hardware to function. A thick client will reside fully on a computer with full interaction rights to an underlying host operating system and able to perform the entirety of its functionality without remote execution or dependence on non-local resources.

Thin Clients

Thin Clients are software or hardware which depend on other non-local software or hardware in order to provide the needed functionality. A thin client may provide a basic framework necessary to communicate and represent functionality that primarily executes on a non-local resource.

Browsers

In its simplest and original form, the Web browser is a thin client that initiates requests to a **Web server** and displays the information that the server returns (through the rendering of static **HTML** pages). As the Web browser has evolved and its functionality has increased to provide a more robust framework for richer capability that has facilitated, in addition to its simplest thin client operations, a transformation of the Web browser into a hybrid client which itself is capable of supporting both thin and hybrid clients that interact with complex services on the network. As such an adaptable and defacto basis for user interaction in the **Service-Oriented Architecture** (**SOA**) distributed paradigm, Web browsers merit specific attention and discussion

Security Considerations

See the Identity Management [P1178] and Authorization and Access Control [P1339] perspectives for assured user access concepts and considerations. See the Confidentiality [P1340] perspective for privacy concepts and considerations.

Management Considerations

For general management considerations see the Enterprise Management [P1330] perspective. For management concerns relating to human-computer interactions see the Human-Computer Interaction [P1032] perspective.

Detailed Perspectives

- Browser [P1155]

- Remote KVM Switch Connectivity [P1393]

P1393: Remote KVM Switch Connectivity

A Keyboard, Video, and Mouse (KVM) switch is a hardware device that allows a user to control multiple computers and other hardware (such as routers) using a single keyboard, video monitor and mouse. Many KVMs also support connecting audio and USB devices through the KVM switch. Some KVM switches can also function in reverse; that is, connecting a single PC to multiple monitors, keyboards, and mice. While not encountered often, this provides a means for the end user to control a single computer from two or more locations. Finally, some KVM switches support acting as a matrix switch where two or more sets of keyboards, video displays, and mice are able to control two or more systems simultaneously. Generally, there is support for control of up to 512 components from a single KVM.

KVM switches typically connect to the keyboard, video, and mouse ports through direct cabling. More recent net-centric KVM switches connect virtually through various KVM protocols over **TCP/IP**.

KVM switches are one means of providing remote user environments; additional means include standards-based remote access services such as Secure Shell (SSH), Virtual Network Computing (VNC), and Remote Desktop Services (RDS) via Remote Desktop Protocol (RDP).

Employing remote user environments over TCP/IP enjoys the advantage of not requiring co-located full-featured KVM hardware and operations personnel in every server farm; however, they do require co-located KVM network interface devices such as dongles. Remote user environments also enable operations personnel to be located beyond the nominal range limits of the hardwired KVM or terminal cables.

Operational environments, particularly as part of the Tactical Edge, may have limited space to support peripheral equipment such as multiple keyboards, monitors and mice. A KVM switch can save critical space in aircraft, tactical vehicles and ships, allowing an operator to use multiple computers with the same keyboard, monitor and mouse.

Security Considerations

KVM switches require securing like any other remote access. In addition, most hardware KVM solutions make use of basic input/output system (BIOS) and other firmware within the KVM switch and need appropriate hardening.

Using KVM switches in a multi-level security environment requires strict adherence to security regulations and procedures. For example, connecting an unclassified computer and a classified computer to the same KVM can easily lead to a situation in which an operator may think that the watch station keyboard, monitor and mouse are connected to the classified computer but in fact they are connected to the unclassified computer, resulting in the compromise of classified information.

See the following perspectives for more information on security aspects:

- Enterprise Security [P1332]
- Integrity [P1334]
- Authorization and Access Control [P1339]
- Application Layer Protocols [P1355]
- Identity Management [P1178]
- Internet Protocol [P1139]
- IP Routing and Routers [P1143]

Management Considerations

For general management concerns see the Enterprise Management [P1330] perspective.

Best Practices

- BP1933: Control access to firmware with strong passwords.
- BP1936: Physically secure hardware components.
- BP1937: Disable component remote management capabilities when secure remote management is not possible.

- BP1938: Disable Bluetooth functionality except when required.
- BP1939: Configure Bluetooth-enabled device pairs to use the strongest Bluetooth security mode supported by each device pair.
- BP1940: Use strong Bluetooth passwords.
- BP1941: Disable Bluetooth device discoverability except during required device pairing.

P1342: Processes

Processes can be ***structured*** or ***unstructured***; however, the distinction between structured and unstructured processes is often vague. Processes often contain both structured and unstructured elements, making it difficult to classify a process as one or the other.

Structured Processes

Structured processes tend to have a greater depth of standardization and a consequent potential for automation. This standardization may align around the process itself, information exchanges, or communication protocols. Standardization does not necessarily imply computer based process. For example, human-based processes using well defined paper forms may be highly structured.

Structured processes are operational information exchanges formally standardized as machine-to-machine protocols with formal machine-readable data encodings.

Advantages of structured processes include specification using standardized process execution languages such as Web-Services Business Process Execution Language (BPEL) [R1347] and technical specification using net-centric, interoperable interfaces using standard structured protocols such as **SOAP** and standard structured data encodings such as XML.

Unstructured Processes

Unstructured processes tend to follow processes and information exchange patterns that are not well defined. Examples may include voice communication, fax, chat, whiteboard collaboration, etc.

Unstructured processes support military and business operations that cannot predict upfront and in deep detail what kinds of data they will want to exchange. Unstructured processes are often suitable for initial setup, negotiating operational relationships, or defining operational problems. These situations usually do not require extensive machine-readable information flow specifications. Examples include voice teleconferencing and collaboration software.

Architectural Considerations

The distinction between structured and unstructured processes is sometimes important, because structured processes have patterns that are easier for software to implement in an interoperable fashion. This means there is a better chance of the processes being measurable, testable, and verifiable.

One way to automate structured processes within software with a focus on data interoperability, process interoperability, and service interoperability is to categorize operational mission data, activities and information flows as structured or unstructured; decompose the process, the data, the data flows, and the resulting set of services to support the process; and leverage standards to help implement and automate the process.

Include a deep, many layered and detailed structural specification of the information flow interfaces as part of the architecture or design of a structured process such that machines can interpret them. Furthermore, for tactical edge operations, consider how interruptions (both expected and unexpected) impact processes. Such interruptions may occur due to operational challenges that disrupt the underlying infrastructure or due to human supervision and compensation for exceptional conditions.

The following technologies and concepts are useful in implementing ***both*** structured and unstructured processes.

Collaboration

Systems and applications can achieve collaborative functions via architectures that are peer-to-peer, server-centric, or some hybrid of the two. By its very nature, collaboration requires systems including related infrastructure to be interoperable both within and across Nodes. Some common examples of applications that may require interoperable infrastructures include e-mail, chat, and videoconferencing. Collaboration generally involves unstructured or opaque data flows of natural language, although it can contain structured data as well such as raw image or sensor data. Even so, use of the underlying infrastructure may be highly structured, to assure timely performance, especially in interactive environments, and protection integrity, privacy, etc. In short, the

designers enable highly adaptable unstructured information flows by demanding the highest anticipated levels of performance, highest anticipated levels of protection (and consequently the most highly structured, even rigid capabilities available) of the underlying common infrastructure. Collaboration technologies enable the exchange of minimally structured information in interactive settings such as audio and visual while making relatively minimal demands on the underlying computing, transport, cryptographic and management infrastructures. This assures maximal agility in shifting data formats and process protocols, enabling maximal robustness and effectiveness at the lowest level of effort. Interoperability evaluation then becomes a matter of aligning the underlying infrastructure capabilities, both at design/acquisition time and by infrastructure operations. Aligning the human processes and data formats such as natural language is then a much simplified, on demand, mission-local management concern.

Orchestration and Workflow Management

All information systems enable business operations by arranging software and data objects into workflow patterns that support actual business processes. Increasingly, monolithic application(s) are no longer statically defining these workflow patterns; instead, these workflow patterns are configured and managed dynamically, persistently stored, and widely distributed using infrastructure components that generally perform business process management. These composite applications, based on a collection of services and data, often share common performance and protection requirements, which enable them to share a common infrastructure.

Orchestration refers to a specific technical mechanism that coordinates the software object execution (primarily services) and related data into the composite services or the composite applications that execute and manage workflow patterns. Orchestration can support static or dynamic/on-demand needs. Such explicit coordination often requires common performance and protection and therefore is a strong driver for common infrastructure. However, not all applications, data access or services will have identical performance and protection requirements because they are not part of the same mission. Consequently, common mission infrastructures are often separated from each other through hierarchical subdivisions such as separate servers or subnets or through separate virtual infrastructures. Using **structured identifiers** to ensure both deconfliction of technologies at the mission level and interoperability within the larger enterprise is key to effective, efficient and robust workflow orchestration and management.

Alerting and Notifications

Alerts, notifications, traps, etc., can be utilized together with a structured process. They can also serve as an important intermediate interface type between unstructured process interfaces and emerging structured standard interfaces. They have the potential to be very structured but in practice they are not.

The terms event, trap, alert, and notification all commonly signify an asynchronous communication protocol pattern in a process. These messages classically do not expect or require acknowledgment by the receiver. The data in these messages classically are short and minimally structured, using opaque data types such as strings, integers, etc. They routinely signal that a threshold or trigger condition of interest to the receiving partner has occurred.

P1164: Services

The *DoD Net-Centric Services Strategy* (NCSS) [R1313] establishes **services** as the preferred means by which data producers and capability providers make their data assets and capabilities available across the Department of Defense (DoD) and beyond. The DoD vision is to establish a Net-Centric Environment (NCE), a framework for human and technical connectivity and interoperability. This environment allows DoD users and mission partners to share and protect information, to make informed decisions, and to leverage shared services and **Service-Oriented Architecture** (**SOA**) that have the following characteristics:

- Supported by the required use of a single set of standards, rules, and a common, shared secure infrastructure provided by the Defense Information Enterprise Mission Area (DIEMA)
- Populated with appropriately secure mission and business services provided and used by each mission area
- Governed by a cross-Mission Area board, chaired by the DoD **Chief Information Officer** (**CIO**)
- Managed by **Global Information Grid** (**GIG**) **Network Operations** (**NetOps**).

Service-Oriented Architecture (SOA) is an architectural style for describing an environment in terms of distinct shared mission and business functions and data exposed as carefully designed, available, secured and managed services. Such services, therefore, are often referred to as "mission" or "business services" and they usually reside in the application layer of the architecture (where the mission and business applications typically reside). Since each carries a distinct mission or business function, they serve as building blocks for key elements of mission or business functionality that can become mission threads and business flows.

Services built specifically for the purpose of creating accessibility for visible mission data and metadata, as part of the *DoD Net-Centric Data Strategy* [R1312] implementation, are also part of the enterprise. As described in the Node Data Strategy [P1329] perspective, some of those data services potentially may be used in operational environments as described above. This would depend on the specific need for the exposed data, maturity level of the service, service ownership, and other factors.

Carrying a business or mission value is not the only characteristic of a service upon which the SOA architectural style is built. One other characteristic of a service is implementation in a loosely coupled manner that, in some cases, would allow orchestrating the service into flows even at run time, creating services composed of other services, and changing the internal implementation of a service without affecting its interface. See the Service-Oriented Architecture [P1304] perspective in NESI Part 1: Overview [P1286] for a list of distinct characteristics that identify a service in SOA. See also NESI Part 3: Migration Guidance [P1198] for discussions on SOA migration of legacy systems and SOA maturity levels.

Another key component of the DoD services vision is the establishment of the enabling and execution environment for mission/business services. This support environment consists of the following:

- Infrastructure responsible for the reliable, timely and secure delivery of service execution results to the consumer
  - Hardware, Operating Systems
  - Networking [P1138]
  - Data storage
  - Middleware that may include Web Infrastructure [P1157], Message-Oriented Middleware [P1046], data servers (e.g., **RDBMS**), run-time service discovery, etc.; some of the middleware-related topics are also discussed in the Information Exchange Patterns [P1326] and Service Optimization and Scalability [P1327] perspectives
  - Utilities and functions responsible for resolving interoperability and integration issues for seamless services communications within or across management boundaries; see the Utility Services [P1328] perspective regarding commonly used techniques
  - Security and Management [P1331] measures implemented within all of the above elements and as specialized utility applications
- Services and functions, along with their underlying infrastructure, implemented at the community or enterprise level that provide collaboration tools, access to services-related **metadata** and thus enable service discovery and use, and technological support for enterprise governance of services. See the Core Enterprise Services [P1175] and related perspectives, especially NCES Directory Services [P1176]. Services are subject to enterprise governance.

> ***Note:*** *Many of the elements of the services-enabling environment participate in the governance structure and processes with participation increasing as the governance matures; however, in this NESI release, such governance of services currently is outside the scope of this perspective.*

DoD leadership has expanded the use of the term "service" beyond mission or business services, as often occurs in some commercial enterprises as well. This is due in part to the fact that the term was in use before the formalized notion of SOA evolved but more so because the benefit of applying principles of service orientation throughout the enterprise architecture enables a degree of uniformity in management of mission and business services plus utilities and infrastructure elements that support and enable them (often called "infrastructure services").

For example, any infrastructure environment utility or function (e.g., a protocol translation function), in good practice, should have defined the party responsible for it, its scope of use and deployment, its interface, rules of access, etc. This data about the utility could be expressed using the same description metadata standard (e.g., **Service Definition Framework** or **SDF**) that is used for a mission service; the utility could be visible and discoverable to the enterprise through the same catalogs and search engines, and a there can be a **Service Level Agreement** (**SLA**) established between the users of the utility and those who are responsible for it. This illustrates the applicability of SOA management approaches to service-enabling utilities and supporting infrastructure elements. The **NCES** enterprise utilities are examples of using the term "service" to describe support environment functions.

The main distinction between an infrastructure and a mission or business service is that an infrastructure service does not represent a primary, distinct mission or business function like a mission or business service does. An infrastructure service is not designed with the flexibility of a mission or business service to be orchestrated into an operational flow or thread. Instead, it might be a part of the underlining infrastructure necessary for mission threads and business processes to execute.

There is a distinction between a service as a service-oriented architecture element (e.g., a service that fetches a specific situational awareness data) and the technology selected to implement it (e.g., a Web service following WS-* specifications, an RSS, etc). Nodes must identify common standards for the modularization, distribution and interaction mechanisms. Service interfaces define the modular boundaries of the provider and consumer. They also serve as the framework for the interactions between provider and consumer **components** and their usage agreements.

> ***Note:*** *See the set of Web Services [P1078] perspectives in NESI Part 5 for guidance on implementing* ***Web services****.*

Node interaction includes intraNode, interNode and extraNode (the notion that helps understand service interoperability issues). Based on the scope of service use in relation to the Node boundaries and independently of the type of the service (e.g., mission/business or support environment), three groups of services include the following:

Enterprise Service (ES) - a service which has broad applicability/usage across multiple Nodes or across the GIG and typically involves or supports interNode interaction. For services supporting Node operations, loss of an enterprise service can have significant impact on data or process availability necessary for Nodes to operate. An important aspect of Enterprise Services is that their data and interface definitions are collaboratively developed and accepted across the Enterprise but not necessarily centrally governed.

Core Enterprise Service (CES) - a subset of the Enterprise Services where the service is ubiquitous across the Enterprise and, depending on the nature of a CES, the loss of it might have a severe impact on the availability of the necessary data and processes for Nodes and perhaps the GIG to operate. This critical impact potential necessitates that a central coordinating authority act as executive agent for the collaboratively developed and accepted data and interface definitions. The executive agent also probably executes some necessary "core" element of the infrastructure required to support a minimal set of capability in support of the CES.

Local (Internal) Service - a service that typically is mission- or application-specific or provides support to intraNode interaction and operation. This class of service is often designed as a means of distributed application integration; it may be used or reused in other Nodes but the data/interface definition ownership and stewardship responsibilities stay with the original Node, Component or Program.

It is possible that a "community" of Nodes may share services; the threshold at which these services become Enterprise Services is subjective and during that transition, services may have both internal and enterprise characteristics. Services may start out as local and then gather momentum in a community. When the **Community of Interest** (**COI**) advocates

standards for that service, it becomes a candidate for an Enterprise Service. ES-track standard services are so critical that the COI identifies an executive agent for coordinating the evolution of the service definition as well as operation of a minimal infrastructure to support interNode and extraNode interactions using that service. Reengineering of services may be necessary for the services to become suitable for enterprise use (see the Phases of SOA Adoption [P1238] perspective in Part 3: Migration Guidance [P1198]).

The loss of an operationally significant CES or ES does not necessarily imply an impact on a Node's internal operations or its ability to operate independent of the GIG. A local cache, proxy, or alternative source may actually service the request. See the Cross-Domain Interoperation [P1169] and CES and Intermittent Availability [P1168] perspectives for further information.

Access to Core Enterprise Services from Nodes or systems in tactical edge and other environments with either challenged infrastructure performance or extraordinary protection characteristics may also require support for caching, content-filtering, anonymizing, and mediation-proxy interoperability, especially between Core Enterprise Services and the local Node. See the perspectives Design Tenet: Inter-Network Connectivity [P1266], Integration of Legacy Systems [P1135], and CES and Intermittent Availability [P1168] for further information.

Service security is an integral part of securing nodes as well as the infrastructure. Services have two major component families, the "provider" components and the "consumer" components, each managed within the context of its local host. It is essential to harden both properly. Some of the technologies used in this process include but are not limited to: Kerberos, WS-Security, X.509, and **SAML**. See the Integrity [P1334] perspective for more information.

Provider components, such as servers, are often a tightly integrated combination of the local computing infrastructure management, the server host's transport layer port management, and the management model of and infrastructure for the application itself. The use of Web services also requires the management of local Web infrastructure providers.

Consumer components, such as clients and browsers, require computing infrastructure management, user environment management, consumer host transport layer port management, and the standardized end-to-end management of the application itself. Web service components also require management of the local Web infrastructure for consumers.

In addition to the management of the components, service management depends on the scope of the service in question. Some services, especially Network Services [P1353] and Application Layer Protocols [P1355] have such a large impact and their components are so widely distributed that responsibility for management is distributed throughout the enterprise. Such distributed management requires coordination among the providers and is generally standardized in terms of **structured identifier** allocation and assignment as well as synchronization protocols.

Enterprise services, on the other hand, generally have their provider as the primary responsible authority, but due to their wide use also have particular Service Optimization and Scalability [P1327], filtering, aggregation, and federation concerns (See the Utility Services [P1328] perspective for more information). Coordination of distributed management in these cases is often more a matter of federation, mirror-site synchronization and proxy deployment management.

Internal services with a mission focus have a primary responsible authority, the provider, but also require coordination with other partner mission services through orchestration and workflow management techniques and technologies.

One of the challenges in promoting an Internal Service to Enterprise Service is that the service may have to switch from internal, intra-Node infrastructures to standardized, interoperable inter-Node infrastructures. For example, many orchestration technologies require all partner Nodes either have common (shared) or interoperable transport and computing file system infrastructures. Three critical areas for interoperable infrastructures are identifier allocation and assignment, service discovery, and enterprise management monitoring and configuration of components.

Detailed Perspectives

- Core Enterprise Services (CES) [P1175]
- Service Enablers [P1325]
- Service Optimization and Scalability [P1327]
- Utility Services [P1328]

P1175: Core Enterprise Services (CES)

Core Enterprise Services (**CES**) require a centralized governing authority to select, develop and manage the services due to their enterprise-wide scope and importance (see the Services [P1164] perspective). In the DoD, both mandated and organic evolution will define the set of Core Enterprise Services for use across the network. While the exact nature of how CES evolve organically within the DoD is unclear, the *DoDNet-Centric Services Strategy* (NCSS) [R1313] obligates Nodes to employ a set of DoD Core Enterprise Services that are identified by the DoD ***Enterprise Information Environment Mission Area*** (EIEMA). These services provide a common information environment infrastructure for the purpose of making other services in the enterprise visible and accessible to anticipated and unanticipated users. The CES also enable interoperability across the **Global Information Grid** (**GIG**) and reduce duplication and unnecessary redundancy in the EIEMA portfolio. The EIEMA community will mandate the use of CES across the DoD as the services become available.

Within the DoD, DISA is responsible for defining and developing some of these capabilities through the **Net-Centric Enterprise Services** (**NCES**) program with the following mission:

- Provide executive life cycle management of enterprise capabilities to support the DoD transformation to net-centricity
- Provide executive oversight in planning and delivery of **Enterprise Service** (**ES**) support to mission performance across the Warfighter, Business, and Intelligence Missions Areas
- Provide the infrastructure to publish data/metadata artifacts and enable the DoD Net-Centric Data Strategy

There are four NCES Product Lines [R1259]:

- ***Collaboration*** - Communicate in real-time using voice, text, and video sessions. Supports collaboration between consumers and producers of information to ensure a common understanding and de-confliction of information. For more on Collaboration see the Collaboration Services [P1184] perspective.
- ***Content Discovery and Delivery*** (CD&D) - Enterprise-wide access to shared/stored data; improved situational awareness; ability for user to acquire more information, more quickly, with a smaller footprint. Federated Search is a type of an enterprise Content Discovery Service; for DoD CES implementation see the NCES Federated Search [P1182] perspective.
- ***User Access*** (Portal) - Tailorable user interface providing a window into NCES and access to its capabilities and information.
- ***Service-Oriented Architecture Foundation*** (SOAF) - Loosely-coupled set of services (security, registry, metadata, mediation, etc.) providing foundation for interoperable computing, including the following capabilities that are mapped to services:
  - Enterprise Service Management provides a toolset with a graphic user interface
    - collects standardized metrics for every monitored service through service component management standard interfaces
    - publishes or otherwise makes available collected metrics to authorized and authenticated consumers
    - enables authorized consumers to set behavioral policy thresholds for each metric
    - publishes or otherwise notifies authorized consumers when a metric goes outside a threshold.
    - publishes a catalog of the monitored services and any inter-dependencies and interactions among them, based on a combination of registered and discovered configurations, to authorized consumers
  - Mediation - capabilities for information transformation, service adaptation, and service orchestration (for a discussion about Transformation see the Utility Services [P1328] perspective)
  - Messaging - Messaging provides a federated, distributed, and fault-tolerant enterprise message bus
  - Metadata services - provide the ability for DoD Enterprise systems to discover and manage (publish, make visible, and access) various metadata artifacts critical to a system's and/or a person's ability to exchange and understand data components within the enterprise. They provide visibility of data representations and enable the development and management of data products to support mediation capabilities within the enterprise. The **DoD Metadata Registry** (**MDR**) stores metadata artifacts such as RDBMS schemas, XML schemas, Taxonomies, and XSL

Transforms. The MDR allows categorization of all of the metadata artifacts (and potentially, services, documents, and people) under one or more taxonomies

- People and Service Discovery - See NCES Directory Services [P1176] and Service Discovery [P1181] perspectives.
- Service Security - provides the support necessary to enable DoD net-centricity

For further information on service management, see the ***Management Considerations*** section of the Services [P1164] perspective.

For further information on service security, see the ***Security Considerations*** section of the Services [P1164] perspective.

Detailed Perspectives

- Overarching Issues [P1165]
- NCES Directory Services [P1176]
- Service Discovery [P1181]
- NCES Federated Search [P1182]
- Collaboration Services [P1184]

P1165: Overarching CES Issues

There are particular challenges in implementing and deploying **Core Enterprise Services** (**CES**), especially in a tactical edge environment. Availability of CES will be a continuing challenge until all services reach full maturity and operational status. Designating a CES liaison should help to monitor the availability of CES functionality and report on them back through the engineering processes of the Node and **components** within the Node. Conversely, the engineering processes for the Node should specifically include provisions for incremental implementation of the CES services.

Nodes operating at special classification levels should coordinate with other Nodes within the same level and with DISA to host CES services on the relevant networks.

Overarching Node application Enterprise Services issues include maturity, availability, disconnected operations, cross-domain security, and compliance. These elements equate to the following perspectives:

- Maturity: CES Definitions and Status [P1166]
- Disconnected operations: CES and Intermittent Availability [P1168]
- Cross-domain security: Cross-Domain Interoperation [P1169]
- Compliance: Net-Ready Key Performance Parameter (NR-KPP) [P1170]

Guidance

- G1576: Provide an environment to support the development, build, integration, and test of net-centric capabilities.
- G1577: Maintain an **Enterprise Service** schedule for interim and final **enterprise** capabilities within the Node.
- G1578: Define a schedule for **Components** that includes the use of the **Enterprise Services** defined within the Node's enterprise service schedule.
- G1626: Identify which **Core Enterprise Services** (**CES**) capabilities the Node **Components** require.
- G1627: Identify the priority of each **Core Enterprise Services** (**CES**) capability the Node **components** require.
- G1629: Identify which **Net-Centric Enterprise Services** (**NCES**) capabilities the Node requires during deployment.

Best Practices

- BP1649: Specifically include provisions for incremental implementation of the **CES** services.
- BP1650: Specifically include provisions for incremental implementation of the hosting Node's **CES** services for Node **Components**.
- BP1661: Engage with the **Net-Centric Enterprise Services** (**NCES**) program office to explore approaches for mobile use of the **Core Enterprise Services** (**CES**) services in mobile Nodes that rely on **Transmission Control Protocol/Internet Protocol** (**TCP/IP**) for inter-node communication.
- BP1675: In the Node's Web infrastructure, support the technologies and standards used by the **CES** services under development as well as any technologies and standards used for **Community of Interest** (**COI**) services.
- BP1683: Coordinate the Node schedule with the schedules of the **Core Enterprise Service** (**CES**) providers.
- BP1684: Coordinate the Node schedule with the **Component** schedules.
- BP1695: Designate a **Core Enterprise Services** (**CES**) liaison to monitor the availability of services.
- BP1697: Make the parallel development of **Core Enterprise Services** (**CES**) outside the control of the Node a part of the Node's risk management activities.

P1166: CES Definitions and Status

The **Core Enterprise Services** (**CES**) capabilities are in various states of maturity. Capabilities will be delivered in increments; CES Increment 1 capabilities, shown below, are scheduled for operation beginning in 2008 (source: https://ges.dod.mil/soa.htm; user authorization required).

Service Discovery	Provides a ***yellow pages***, categorized by DOD function, enabling users to advertise and locate capabilities available on the network
Service Security	Provides a layer of defense in depth that enables protection, defense, and integrity of the information environment
Identity Management	Provides the methodology and functions for maintaining information on people, consumers, and service providers. Supports the validation of identity authentication credentials
Service Management	Enables monitoring of DoD **Web services**. Provides reporting of service-level information to potential and current service consumers, program analysts, and program managers
Service Mediation	Allows disparate applications to work together across the **enterprise** by supporting the transformation of information from one format to another, and the correlation and fusion of data from diverse sources. Supports creation and implementation of process workflows across the enterprise
Machine-to-Machine Messaging	Provides reliable **machine-to-machine** message exchange across the enterprise
Metadata Services	Provides access to **Extensible Markup Language** (**XML**) data elements, taxonomy galleries, schemas, and validation and generation tools for DOD software developers
DoD Web Services Profile	Provides specifications and implementation guidelines to maximize interoperability across DOD Web service implementations

NCES Increments will be rolled out every 24-26 months. Consider the NCES increment schedule in scheduling Node evolution in coordination with systems within the Node.

Guidance

- G1301: Practice layered security.

P1168: CES and Intermittent Availability

Core Enterprise Services (**CES**) may be unavailable for several reasons, including loss of connectivity, actual service unavailability, or service rejection. There are two related challenges: how to handle lapses in the availability of CES services and how to align inter-Node and intra-Node solutions. The lack of availability of CES services must not disrupt intra-node availability of locally hosted services. While alignment of intra- and inter-node technical solutions is very desirable, the interface to locally hosted **Components** must not be dependent on the availability of CES services.

Specific guidance is largely dependent upon the specific Node operating environment and mission. There are some basic options for meeting these challenges:

- Locally host failover copies of certain CES services. Components that are dependent upon **Enterprise Services** for infrastructure functions, such as security, continue to operate after failing over to the local instances until **enterprise** accessibility is re-established. This approach requires replication of enterprise services data (the data used by the enterprise services) between the local failover services and the "master" enterprise services. It also requires development of failover behavior in the applications, services, and infrastructure.
- Develop Components to be adaptive, applying default rules and behaviors when Enterprise Services are inaccessible. This approach, along with the definition of the default rules and behaviors would depend on factors such as the sensitivity and importance of the information involved. For example, access control decisions might default to local capabilities such as **Active Directory** local user accounts. Or local caching might be used to retain the most recently known values for information such as previously discovered services.
- Employ separate external-facing and internal-facing implementations of published services so that external disruptions do not affect local accessibility. The external-facing copy of the service could use Enterprise Services, and the internal-facing copy could implement local Node behavior. As an example, the external-facing copy could implement **Public Key Infrastructure** (**PKI**) **authentication** and **authorization**, whereas the internal-facing copy could implement Active Directory security. The challenge in this approach is in the coordination of the external-facing and internal-facing copies of such services, such as to provide shared access to databases or replication of data between the external-facing and internal-facing implementations.

Nodes and Components will likely employ some combination of, or evolution of, the above options.

Uniformity and alignment between the technical mechanisms for accessing local services and Enterprise Services should be an objective. Where possible, the burden of providing such uniformity and alignment should rest on the Node infrastructure, rather than the individual Components within the Node, thus isolating the complexities and making them more manageable. Consider the necessity of using CES-provided **Software Developers Kits** (**SDKs**) and **Key Interface Profile** (**KIP**) compliance when formulating an approach; use of an approved SDK may drive separation of external-facing and internal-facing implementation described in the last option above. Finally, the immaturity of the CES services and the alignment of local and external services access, as a whole, should figure prominently in the risk management activities of the Node and Components within the Node.

Guidance

- G1630: Comply with the applicable **Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**) for implemented **Core Enterprise Services** (**CES**) in the Node.
- G1631: Expose **Core Enterprise Services** (**CES**) that comply with the applicable **Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**) in all Node services **proxies**.

Best Practices

- BP1651: Ensure **Node Components** have access to **Core Enterprise Services**.

P1169: Cross-Domain Interoperation

By and large, the implementation of net-centric concepts across security domains has not been defined. Trusted guards do not act as network **routers**; information to be transferred across a guard is delivered to the guard, processed, and then delivered to a defined endpoint on the other side if the rules are satisfied. The guard in the middle disrupts the normal pattern for use of the **CES** services.

In order for **services** to work through the trusted guards that interconnect different domains, there must be a well defined set of messages that can be passed through the guard to effect the conversation necessary to use the service and return results. This restriction, if built into the service's interface, could be unduly restrictive on the design of the interface.

It may be more practical for each such service to provide service proxies for use in the other security domains, and corresponding client proxies in the local domain. The server **proxy** and client proxy for the service might then communicate across the trusted guard in a private, high efficiency manner that the guard can process. But even this approach is restrictive in that the server proxies have to be installed in the other security domains, and this departs from some fundamentals of net-centric concepts such as dynamic **service discovery**.

Until such approaches are prototyped and explored more fully, Nodes should anticipate that services will not be capable of cross-domain invocation. Furthermore, for services that have utility in other security domains, implementer should consider providing copies of such services for hosting in the other domains, and use **XML** document transfers across the trusted guard to keep the copies in synchronization. This approach depends on many factors, and may not be suitable for all services.

Guidance

- G1613: Prepare a **Node** to host new **Component services** developed by other Nodes or by the **enterprise** itself.

Best Practices

- BP1614: Plan a contingency response to the **Node** becoming a new **component service** within another Node.
- BP1691: Use **Node** implemented **Service Discovery** (**SD**) to meet compartmentalization needs.
- BP1698: Plan for the event that **Component** services within a **Node** cannot be invoked across security domains.

P1170: Net-Ready Key Performance Parameter (NR-KPP)

The **Net-Ready Key Performance Parameter** (**NR-KPP**) provides a means to assess net-ready attributes required for both the technical exchange of information and the end-to-end operational effectiveness of that exchange. The NR-KPP replaces the Interoperability KPP, and incorporates net-centric concepts for achieving **Information Technology** (**IT**) and **National Security Systems** (**NSS**) interoperability and supportability. The NR-KPP assists Program Managers, the test community, and Milestone Decision Authorities in assessing and evaluating IT and NSS interoperability.

The NR-KPP assesses information needs, information timeliness, **information assurance** (**IA**), and net-ready attributes required for both the technical exchange of information and the end-to-end operational effectiveness of that exchange. The NR-KPP consists of verifiable performance measures and associated metrics required to evaluate the timely, accurate, and complete exchange and use of information to satisfy information needs for a given capability. Program managers will use the NR-KPP documented in **Capability Development Documents** (**CDD**) and **Capability Production Documents** (**CPD**) to analyze, identify, and describe IT and NSS interoperability needs in the **Information Support Plan** (**ISP**) and in the test strategies in the **Test and Evaluation Master Plan** (**TEMP**).

The Chairman of the Joint Chiefs of Staff Instruction (CJCSI) 6212.01E, *Interoperability and Supportability of Information Technology and National Security Systems*, 15 December 2008, [R1175] removed the **Net-Centric Operations and Warfare Reference Model** (**NCOW RM**), integrating the components of the former NCOW RM into other elements of the NR-KPP. The following five elements now comprise the NR-KPP:

- Compliant solution architecture
- Compliance with DOD Net-Centric Data and Services strategies ([R1172] and [R1313],respectively), including data and services exposure criteria
- Compliance with applicable GIG Technical Direction to include **DISR**-mandated IT Standards reflected in the **TV**-1 and implementation guidance of GIG Enterprise Service Profiles (GESPs) necessary to meet all operational requirements specified in the DoD Information Enterprise Architecture and solution architecture system/service views
- Verification of compliance with DOD IA requirements
- Compliance with supportability elements to include, spectrum analysis, Selective Availability Anti-Spoofing Module (SAASM) and the Joint Tactical Radio System (JTRS)

Detailed Perspectives

- Information Assurance (IA) [P1171]
- Net-Centric Operations and Warfare Reference Model (NCOW RM) [P1172]
- Key Interface Profile (KIP) [P1173]
- Integrated Architectures [P1174]

P1171: Information Assurance (IA)

Most Nodes, when delivering a capability to the warfighter or business domains, will use **Information Technology** (**IT**) to enable or deliver that capability. For those Nodes, developing a comprehensive and effective approach to **IA** is a fundamental requirement and is key in successfully achieving Node's objectives. The DoD defines IA as follows (see DoDD 8500.1 [R1197]):

Information Assurance (IA). Measures that protect and defend information and information systems by ensuring their availability, integrity, authentication, confidentiality, and non-repudiation. This includes providing for the restoration of information systems by incorporating protection, detection, and reaction capabilities.

DoD policy and implementing instructions on information assurance are in DoD Directive 8500.01 [R1197] and DoD Instruction 8500.2 [R1198]. Nodes and **Components** for programs should be familiar with statutory and regulatory requirements governing information assurance and understand the major tasks involved in developing an IA organization, defining IA requirements, incorporating IA in the Node's and Component architecture, developing an acquisition IA strategy (when required), conducting appropriate IA testing, and achieving IA certification and accreditation for the program.

Guidance

- G1632: Certify and accredit Nodes with all applicable DoD **Information Assurance** (**IA**) processes.
- G1633: Host only DoD **Information Assurance** (**IA**) certified and accredited **Components**.
- G1634: Certify and accredit **Components** with all applicable DoD **Information Assurance** (**IA**) processes.

P1172: Net-Centric Operations and Warfare Reference Model (NCOW RM)

The **Net-Centric Operations and Warfare Reference Model** (**NCOW RM**) represented strategies for transforming the **enterprise** information environment of the Department of Defense. It was an architecture-based description of activities, services, technologies, and concepts to enable a net-centric enterprise information environment for warfighting, business, and management operations throughout the DoD. Included in this description were activities and services required to establish, use, operate, and manage this net-centric enterprise information environment. Major activity blocks included the generic user-interface, the intelligent-assistant capabilities, the net-centric service (core, **Community of Interest**, and enterprise control) capabilities, the dynamically allocated communications, computing, and **storage** media resources, and the enterprise information environment management components. Also included was a description of a selected set of key standards and/or emerging technologies that would be needed as the NCOW capabilities of the **Global Information Grid** (**GIG**) were realized.

Transforming to a net-centric environment requires achieving four key attributes: reach, richness, agility, and assurance. The initial elements for achieving these attributes include the *DoDNet-Centric Services Strategy*[R1313], the *DoD Net-Centric Data Strategy*[R1172], and the *DoD Information Assurance (IA) Strategy*[R1345] to share information and capabilities. The NCOW RM incorporated these strategies as well as net-centric results produced by the Department's **Horizontal Fusion** (**HF**) pilot portfolio.
The NCOW RM provided the means and mechanisms for acquisition program managers to describe their transition from the current environment (described in ***GIG Architecture Version 1***) to the future environment (described in ***GIG Architecture Version 2***). In addition, the NCOW RM was a key tool during program oversight reviews for examining integrated architectures to determine the degree of net-centricity a program possessed and the degree to which a program could evolve to increased net-centricity. Compliance with the NCOW RM was one of the four elements that initially comprised the **Net-Ready Key Performance Parameter** (**NR-KPP**).

> ***Note:*** *The NCOW RM was a key compliance mechanism for evaluating DoD information technology capabilities and the NR-KPP in accordance with Chairman of the Joint Chiefs of Staff Instruction (CJCSI) 6212.01D, Interoperability and Supportability of Information Technology and National Security Systems, 8 March 2006. The 15 December 2008 revision to this instruction, CJCSI 6212.01E, removed the NCOW RM element of the NR-KPP, integrating the components of the former NCOW RM into other elements of the NR-KPP.*

Guidance

- G1636: Comply with the **Net-Centric Operations and Warfare Reference Model** (**NCOW RM**).

P1173: Key Interface Profile (KIP)

Note: *Chairman of the Joint Chiefs of Staff Instruction (CJCSI) 6212.01E[R1175], revised 15 December 2008, deletes the* ***Key Interface Profile*** *(****KIP****) element of the NR-KPP and replaces it with the "Technical Standards/ Interfaces" element. This revision further indicates that* ***Global Information Grid*** *(****GIG****) Enterprise Service Profiles (GESPs) are evolving to provide a net-centric oriented approach for managing interoperabilty across the GIG based on the definition and configuration control of key interfaces and enterprise services. The* ***Defense Acquisition University*** *(****DAU****) Interim Defense Acquisition Guidebook, Chapter 7, contains additional information.*

The following information is from an earlier version of the *Defense Acquisition Guidebook* (specifically, Chapter 7.3.4.2). A KIP is the set of documentation produced as a result of interface analysis which designates an interface as key; analyzes it to understand its architectural, interoperability, test and configuration management characteristics; and documents those characteristics in conjunction with solution sets for issues identified during the analysis. The profile consists of refined operational and systems view products, Interface Control Document/Specifications, Systems Engineering Plan, Configuration Management Plan, **Technical Standards View** (TV-1) with SV-TV Bridge, and procedures for standards conformance and interoperability testing. Relevant GIG KIPs, for a given capability, are documented in the **Capability Development Document** and **Capability Production Document**. Compliance with identified GIG KIPs are analyzed during the development of the **Information Support Plan** (**ISP**) and **Test and Evaluation Master Plan**, and assessed during **Defense Information Systems Agency Joint Interoperability Test Command** (**JITC**) joint interoperability certification testing. An interface is designated as a key interface when one or more the following criteria are met:

- The interface spans organizational boundaries.
- The interface is mission critical.
- The interface is difficult or complex to manage.
- There are capability, interoperability, or efficiency issues associated with the interface.
- The interface impacts multiple acquisition programs.

Program manager compliance with applicable GIG KIPs is demonstrated through inspection of **Joint Capabilities Integration and Development System** (**JCIDS**) documentation and test plans, and during JITC interoperability certification testing (see CJCSI 3170.01 and CJCSI 6212.01 for detailed discussions of the process).

Guidance

- G1630: Comply with the applicable **Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**) for implemented **Core Enterprise Services** (**CES**) in the Node.
- G1631: Expose **Core Enterprise Services** (**CES**) that comply with the applicable **Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**) in all Node services **proxies**.

Best Practices

- BP1685: For **Key Interface Profile** (**KIP**) specifications that are not available or insufficiently mature, implement a "best effort" by following the published intent of functionality and monitor or participate in the relevant specification development body.

P1174: Integrated Architectures

The **DoD Architecture Framework** (**DoDAF**), available via the ***General Public Documents*** Quick Link on the DoD Architecture Registry System Welcome Page, provides the rules, guidance, and product descriptions for developing and presenting architecture descriptions to ensure a common denominator for understanding, comparing, and integrating architectures. An integrated architecture consists of multiple views or perspectives (**Operational View** [**OV**], **Systems and Services View** [**SV**], **Technical Standards View** [**TV**] and **All-Views** [**AV**]) that facilitate integration and promote interoperability across capabilities and among related integrated architectures.

- The OV is a description of the tasks and activities, operational elements, and information exchanges required to accomplish DoD missions.
- The SV is a description, including graphics, of systems and interconnections providing for, or supporting, DoD functions.
- The TV is the minimal set of rules governing the arrangement, interaction, and interdependence of system parts or elements, whose purpose is to ensure that a conformant system satisfies a specified set of requirements. Technical Views include approved standards from the **DoD Information Technology Standards Registry** (**DISR**).[R1179]
- The AV products provide information pertinent to the entire architecture but do not represent a distinct view of the architecture. AV products set the scope and context of the architecture.

The **Global Information Grid** (**GIG**) architecture describes the basic, high level architecture in which Nodes reside. It is an integrated architecture consisting of the various DoDAF views. It provides a common lexicon and defines a basic infrastructure for the performance of information exchanges with other GIG Nodes using the **GIG Enterprise Services** (**GES**) that DISA is developing as part of the **Net-Centric Enterprise Services** (**NCES**) program.

Guidance

- G1635: Make Nodes that will be part of the **Global Information Grid** (**GIG**) consistent with the *GIG Integrated Architecture*.

P1176: NCES Directory Services

Secure inter-node interoperability relies heavily on the ability to lookup information about people and objects or devices across the breadth of the **Global Information Grid** (**GIG**). The technologies that support this form of discovery are known collectively as directory services. There are several standardized and layered directory services. The lower layer directory services primarily discover Internet Hosts on which data, applications, services and people's accounts reside.

The best known of the lower layer directory services is the **Domain Name System** (**DNS**). The lower layer directory services also include various host identification services such as the **Dynamic Host Configuration Protocol** (**DHCP**). The Network Services [P1353] perspective covers these services in more detail. More localized enterprise directory services include Windows directory services (such as Windows Internet Name Service or WINS) and Novell Directory Services (NDS). These services are confined within the local area network or virtual local overlay network and require the **Net-Centric Enterprise Services** (**NCES**) directory services to interoperate beyond the Node or its local infrastructure.

For performance and scalability reasons, core lower layer directories usually are constrained to critical services such as **Public Key Infrastructure** (**PKI**) support for email and people (such as administrative user email accounts) in addition to their primary function as a host identity registry.

The NCES service taxonomy includes NCES Directory Services under the scope of CES People Discovery as part of Service-Oriented Architecture Foundation product line (see [R1259]). NCES People Discovery provides services to publish and find, via LDAP-standard interfaces, available information on GIG users and connected devices. The Joint Enterprise Directory Services (JEDS) provides user information aggregated from a number of DoD repositories.

Nodes routinely use directory services today, such as Microsoft **Active Directory** and the DoD PKI Global Directory Service (GDS). Although implementations are widespread across the GIG, there is limited coordination and synchronization, creating pockets of information that must be unified. There are also substantial differences among implementations, including naming conventions. This situation is made more complex by the fact that these directories are typically also integral to a Node's security and system administration, supporting such basic functions as user login.

SOA Directory Services

A SOA-specific registry and directory service is **Universal Description Discovery and Integration** (**UDDI**). See the Service Discovery [P1181] perspective for detailed information.

Guidance

- G1625: Provide a **commercial off-the-shelf** Directory Service that all of the **components** of a Node can use.
- G1637: Make Node-implemented **directory services** comply with the directory services **Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**).
- G1638: Comply with the directory services **Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**) in Node directory services **proxies**.

Best Practices

- BP1686: Align Node interfaces to **Components** for directory services with the guidance being provided by the Joint Directory Services Working Group (JDSWG) and sub-working groups, including such guidance as naming conventions, federation, and synchronization.
- BP1687: Follow **Active Directory** naming conventions defined in the *Active Directory User Object Attributes Specification* as required by the DoD **CIO** memorandum titled *Microsoft Active Directory (AD) Services*.

P1181: Service Discovery

The ability to discover services is a major factor in the enablement of using and sharing services in the enterprise. The discovery concept relies on human- and machine-usable registries for maintaining metadata descriptions of information and services. The intent of these "service registries" is to provide all of the information required for an application developer to locate and use an appropriate service; for example, determine the features and functions the service provides, identify how to invoke the service, discover the supported **Quality of Service** (**QoS**), understand how to contact the service owners, and determine where the service resides. In the case of highly mature services (see the set of Migration Patterns [P1201] perspectives for SOA maturity discussions), Nodes and Components should also be able to discover dynamically where Component services and information reside in the **Global Information Grid** (**GIG**) and bind to those providers at runtime.

The DISA **Net-Centric Enterprise Services** (**NCES**) program provides such a registry/repository as part of the NCES SOA Foundation product line. NCES Service Discovery consists of a **commercial off-the-shelf** (**COTS**) **Universal Description, Discovery, and Integration** (**UDDI**) registry customized to provide service governance as well as enhanced end user access. Web services are also available to enable service publishing and service discovery at the application layer.

Nodes face several implementation choices regarding the alignment of **Component** and Node approaches to service discovery. Register Components that the Node exposes with the DISA-hosted registries so that the Component services are visible to other Nodes. Internal-facing services that are not likely to be of value beyond the boundary of a Node do not have to be discoverable, although it is a good practice. Implementing service discovery within a Node can support availability of Component services within the Node.

Guidance

- G1639: Describe **Components** exposed by the Node as specified by the **Service Definition Framework**
- G1640: Register **components** that a **Node** exposes as **SOAP** Web services with DoD-approved registries.
- G1641: Comply with the Service Discovery **Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**) in Node-implemented **Service Discovery** (**SD**).
- G1642: Comply with the **Service Discovery** (**SD**) **Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**) in Node Service Discovery **proxies**.

Best Practices

- BP1690: Use Node implemented **Service Discovery** (**SD**) for high availability.
- BP1691: Use **Node** implemented **Service Discovery** (**SD**) to meet compartmentalization needs.

P1182: NCES Federated Search

The DISA **Net-Centric Enterprise Services** (**NCES**) program description of Content Discovery states that Content Discovery provides a standard, vendor neutral approach for exposing metadata to the **Global Information Grid** (**GIG**). It consists of three components:

- ***Centralized Search*** - Web content crawled by Intelink
- ***Federated Search*** - Interface for submitting search queries and returning aggregated results
- ***Enterprise Catalog*** - Interface for information producers to update enterprise metadata catalogs

The capability allows searching across a set of Content Discovery Services and yielding an integrated result. The Federated Search service allows sending a query to a large set of disparate data providers, collecting the results generated by each, and presenting the results back to the user after de-duplicating, ranking, etc. This allows a user to submit a query from one place using one syntax and retrieve relevant data from many sources across DoD. This approach leverages existing data sources and production processes.

Federated Search implementation is a set of cooperating Web services. These services talk to each other using a common specification. The specification defines the communication of the query and the results from the query. It describes not only the meaning, but also the format of the data that services exchange.

The Federated Search service uses the **Defense Discovery Metadata Specification** (**DDMS**) to represent the concepts of a query as well as the resource result records, called meta cards, that a search result generates. Data providers match outgoing queries against the resource meta cards to generate search results. The DDMS ties the queries to the results using a common vocabulary.

The domain of the Federated Search service is limited to the provider sites the sponsoring organizations make available for the DoD enterprise. The Federated Search service does not provide visibility or access to private provider sites that do not participate in the Federated Search service. Each **Node** should implement Federated Search - **Registration Web Service** (**RWS**) and **Search Web Service** (**SWS**). Data producers use the RWS to register content sources; the SWS is searches for content from the registered sources.

Guidance

- G1643: Comply with the **Federated Search** - **Registration Web Service** (**RWS**) **Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**) in Node implemented Federated Search - Registration Web Service (RWS).
- G1644: Comply with the **Federated Search** - **Search Web Service** (**SWS**) **Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**) in Node implemented Federated Search - Search Web Service (SWS).
- G1645: Implement a local **Content Discovery Service** (**CDS**).
- G1646: Comply with the directory services **Global Information Grid** (**GIG**) **Key Interface Profiles (KIPs**) in Node **Federated Search** Services **proxies**.
- G1647: Provide access to the **Federated Search** Services.

Best Practices

- BP1648: Host the **Registration Web Service** (**RWS**) registration **portlet** in the Node.
- BP1865: Provide sufficient program, project, or initiative **metadata** descriptions and automated support to enable **mediation** and translation of the data between **interfaces**.

P1184: Collaboration Services

Collaboration tools provide a virtual meeting room environment for human interaction. The virtual environment enables multimedia collaboration (text, voice, and video) in multiple modes (person-to-person, open chat, restricted meeting, etc.) and application broadcasting and sharing.

A 2 February 2009 DoD **CIO** memo, *DoD Enterprise Services Designation*, describes the designated DoD **Enterprise Services**, including collaboration services. The **DISA Joint Interoperability Test Command** (**JITC**) has validated a suite of collaboration tools and standards called the **Defense Collaboration Tool Suite** (**DCTS**) for interoperability and operational use. The DCTS **Collaboration Management Office** (**CMO**) within DISA is responsible for fielding, sustaining, and managing the life cycle of DCTS. Collaboration products approved for interoperability are listed at http://jitc.fhu.disa.mil/washops/jtcd/dcts/status.html. Products certified for use on the **Secret Internet Protocol Router Network** (**SIPRNet**) are listed at http://jitc.fhu.disa.mil/washops/jtcd/dcts/projects.html.

Programs are not to implement chat services or renew licenses on existing services that overlap with approved DoD Enterprise Services without a waiver. Circumstances that may justify a waiver include challenging or hostile operational environments that have additional performance, including **quality of service** (**QoS**), requirements that the designated DoD Enterprise Services cannot adequately meet. If a program utilizes a locally developed or provided chat service, the NESI Text Conferencing [P1388] perspective provides applicable reference information and guidance. Any such locally developed or provided service should conform with standards registered within **Defense IT Standards Registry** (**DISR**), applicable **security technical implementation guides**, and products from JITC list.

Detailed Perspective

- Text Conferencing [P1388]

Best Practices

- BP1692: Determine which Collaboration Service vendor offering to employ in a disadvantaged environment or separate network.
- BP1693: Make sure that **collaboration** products used to satisfy urgent requirements are from the **JTIC** list.

P1388: Text Conferencing

Text conferencing, sometimes called ***on-line chat*** or simply ***chat***, is a synchronous text-based communication. The common English definition of chat implies something less than serious; however, on-line chat is a very serious and effective means of communication (i.e., collaborating) that can convey important, formal dialog between the participants. Information that flows between participants is not limited to simple text but can covey complex constructs that reflect information, knowledge, understanding and even wisdom. Recently, text communication has moved beyond human-to-human dialog and has become increasingly used to connect automated software agents to humans and other software agents

Text conferencing provides the ability to transmit plain text messages between inidividuals or groups of individuals in near-real-time. Some implimentations support structured messages that help the text conferencing infrstructure process and distribute the text as desired by the sender. Text conferencing implementations generally have the following qualities:

- Allow for the rapid dissemination of information
- Provide a history of communications useful for after action reviews or to catch up on missed messages
- Support filterable inbound message traffic
- Operate at the security level of the underlying network
- Are simple to use
- Require minimum bandwith and are easily compressed
- Reduce voice network traffic
- Overcome elctro-magnetic interferences
- Overcome line-of-sight of radio limitations
- Provide a means for finding, retrieving, and subscribing to changes in the presence status (e.g., "online" or "offline") of users

There are predominately two protocols that govern text communication: Internet Relay Chat (IRC), and Extensible Messaging and Presence Protocol (XMPP).

Internet Relay Chat (IRC)

Internet Relay Chat (IRC) is a form of near-real-time synchronous conferencing that is comprised of a network of IRC servers and IRC clients. The IRC network optimizes the routing of messages between clients by only transmitting a message once along any network link.

There are several types of software components that interact with IRC networks: ***user clients***, ***bouncers***, and ***bots***. IRC user clients simplify for human users the use of IRC messages, usually with an easy-to-use interface. IRC bouncers run on a server and act as persistent proxies for the user clients, supporting intermittent connectivity between the IRC server and the IRC user client. IRC bots often provide high-speed, automated IRC services such as registration and management. Bots can be in any number of languages since the IRC protocol acts as a standardized message based interface. Additionally, bots may execute in a user session to assist with common tasks.

Additional IRC Information Sources

- IETF RFC1459, Internet Relay Chat Protocol, May 1993
- IETF RFC2810, Internet Relay Chat: Architecture, April 2000

Extensible Messaging and Presence Protocol (XMPP)

The Extensible Messaging and Presence Protocol (XMPP) is an **eXtensible Markup Language** [**XML**] protocol for providing near-real-time synchronous text conferencing and presence information. XMPP- based text conferencing infrastructure is comprised of a network of XMPP servers and XMPP clients.

XMPP clients send XMPP XML messages to an XMPP server. The XMPP messages can be messages for other clients or commands that are to be processed by the XMPP servers. XMPP servers are tasked with maintaining

the ***presence*** of XMP clients (users) on the XMPP network. As XMPP clients join and leave the XMPP network, their presence is made available to other XMPP clients that have expressed interest in those XMPP clients.

XMPP gateways can link XMPP networks to other networks such as email (**SMTP**), Internet Relay Chat (IRC), Session Initiation Protocol (SIP) for Instant Messaging and Presence Leveraging Extensions (SIMPLE), and Short Message Service (SMS) as well as other legacy networks (see Application Layer Protocols [P1355] for additional information). XMPP only defines the concept of a gateway; the implementation of the gateways is outside the scope of XMPP.

XMPP relies on the use of the Jabber Indentifier (JID) which ties the identificaion of the XMPP client (user) to a domain (i.e., <**`node@domain/resource`**>. This scheme is similar to the methods used to deliver email but it is not similar to the method used by Internet Relay Chat (IRC) which has a limit of characters and is tied to the host name. This difference in structure and size of structured identifiers used to identify users can limit interoperability of user identifiers between XMPP and IRC systems.

The current base XMPP specifications are RFC 3920 and RFC 3921 (see the additional XMPP information sources below). However, the **Internet Engineering Task Force** (**IETF**) XMPP Working Group is revising these specifications to incorporate lessons learned from current implementation challenges.

Additional XMPP Information Sources

- XMPP Standards Foundation, http://xmpp.org
- IETF RFC3920, *Extensible Messaging and Presence Protocol (XMPP): Core*, October 2004
- IETF RFC3921, *Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence*, October 2004
- XEP0205, *Best Practices to Discourage Denial of Service Attacks*, Version 0.2, 10 July 2007

Best Practices

- BP1907: Use Internet Relay Chat (IRC) bots to provide network based IRC services.

P1325: Service Enablers

The following basic factors enable service use:

- service is identified by standard **structured identifier** such as a **Uniform Resource Identifier** (**URI**)
- service is advertised across the enterprise
- service is discoverable across the enterprise

In addition to these basic factors, give careful consideration to the following separate but related topics:

- ***Service Provider*** - service deployment, provisioning, service consumer relationship maintenance, change management
- ***Service Consumer*** - service selection, integration and interoperability, service provider relationship maintenance, change management
- ***Service Infrastructure*** - service advertisement and discovery scope management, isolation, aggregation, mirrors and proxies, capacity and mission assurance management, etc.

For interaction (including interNodal and extraNodal) with the **Global Information Grid** (**GIG**), the DISA **Net-Centric Enterprise Services** (**NCES**) program provides a **Core Enterprise Services** (**CES**) level implementation for some of these enablers (e.g., Discovery Services).

Service Management interoperability depends on management standards such as those from the Information Technology Infrastructure Library (ITIL), the Distributed Management Task Force (DMTF), the **International Telecommunications Union** (**ITU**) and the Telemanagement Forum's extended Telecommunications Operations Map (model).

> ***Note:*** *In the case of a composite service, register each of the services that comprise it and provide each service's own unique URI and description.*

Service Identification

URIs uniquely identify **HTTP**-based services, and their identifiers are managed in accordance with Command Structure, Doctrine and Commander's Intent.

Service Publication and Advertisement

Provide enough semantic information in service advertisements to allow perspective service consumers to determine whether the service is suitable for a particular application. The service consumer should not have to examine the service code to make this determination.

Each service provider registers and provides a public abstract interface of its services and data to include its transport and information assurance bindings.

For further information see the Service Discovery [P1181], Service Definition Framework [P1296], and Universal Description, Discovery, and Integration (UDDI) [P1075] perspectives.

Service Discovery

A service may be discoverable a number of ways: by searching a repository such as the **DoD Metadata Registry**, by searching a well-known service catalog technology such as multi-cast catalog or anycast catalog or by searching a **UDDI** directory service, or by using a generic search engine such as Google.

Detailed Perspectives

- Service Discovery [P1181]
- Information Exchange Patterns [P1326]

P1326: Information Exchange Patterns

Three fundamental information exchange patterns prevalent in DoD enterprise are request/response, publish/subscribe and streaming media. Different **Service Level Agreements** (**SLA**) and **Quality of Service** (**QoS**) requirements, especially in the area of transport infrastructure, distinguish these usage patterns. Consequently, they are sensitive to deployment at the Tactical Edge.

Request / Response

While considered a "classic" in client-server architectures, the request/response messaging exchange pattern is also fundamental to the **Service-Oriented Architecture** (**SOA**) style. A service Consumer sends a request message to a service Producer. The Producer processes the message and executes appropriate service operations based on the content of the message. Following the completion of these operations, a response message is returned to the Consumer. This response message may return the requested information or notification of an operation complete (or an exception).

While this pattern is typically implemented in a purely synchronous fashion (as in **Web service** calls over **HTTP**, where the requester holds a connection open and waits until the response is delivered or the timeout period expires), asynchronous implementations of the request/response pattern are also valid.

Publish / Subscribe

Publish/subscribe is a message exchange pattern in which clients address messages to a specific node in a content hierarchy, called a topic. Publishers and subscribers are generally anonymous and can publish or subscribe dynamically to the content hierarchy. The system takes care of distributing the messages arriving from a node's multiple publishers to its multiple subscribers.

This pattern usually is used to distribute events (e.g., notifications about changes in shared state in the architecture) to multiple interested parties as soon as the events become available. An event contains enough information for the subscriber to allow it to initiate an appropriate action, which could include invoking a service. For example, a service consumer interested in a particular remote data subscribes to RSS notifications about changes in or about that data (e.g., a change in data location). When the notification is received, the consumer requests a Web service using parameters provided in the notification and obtains the update. The event itself could be a result of the execution of a service or a result of processing of one or more other events.

This pattern typically is implemented in a loosely coupled asynchronous fashion. One of the main reasons for this is that at the time of the event the networking link with the consumer might be unavailable or the consumer could be down. This requires an intermediary in the form of a queue or other type of agent to store the event message until consumer is able to receive and process it. The degree of message persistence (and therefore the robustness of the system) varies among implementations.

For further information on this topic see the Processes [P1342] perspective.

Streaming and Isochronous Flows

There is a class of data flows such that the flow can be processed as a steady and continuous stream. Noted for their Quality of Service requirements, particularly their sensitivity to variance in inter-packet delay, this class of data includes voice, video and interactive services such as remote control and collaboration.

P1327: Service Optimization and Scalability

Optimization and scalability techniques generally improve application performance by increasing throughput and decreasing latency. Many tactical edge environments are characterized by low-bandwidth and intermittent communications, as well as other resource shortfalls. Optimization and scalability services make the best of challenged resources.

The subsections below describe several representative optimization/scalability techniques; many additional pertinent optimization/scalability techniques exist. Further, there are many varieties of each optimization/scalability technique in commercial industry as well as purpose-built renditions for the military domain, so definitions may vary among vendors.

Caches and compression are common technological threads in performance optimization. Caches are local temporary storage areas for when rapid or frequent access to data or objects is necessary, but they do not transform the data proper. Compression reduces the amount of data in a sequence of bits or bytes for concise transmission and then reconstructs it for access.

Caching

Caching is local storage of remote data designed to reduce unnecessary transfer of data. Caching may improve throughput and decreases latency by avoiding unnecessary trips across the network.

Object caching is very different than byte caching in that it is often protocol/application specific and is an all-or-nothing affair. If the cache contains the object, the user gets access to the object from a local store extremely quickly. Object caching can greatly reduce, almost to zero, the bandwidth and the latency of Web applications. The only transactions that cross the wide area network (WAN) are a quick check to ensure that the copy in cache is still current.

A typical design of application servers includes pools and caches of the internal container services objects that allow the architect to tune the server resources according to the application specifications for performance, scalability, and availability.

Compression

The goal of data compression is to represent an information source (e.g., a data file, a speech signal, an image, or a video signal) as accurately as possible using the fewest number of bits. Data compression is particularly useful in communications because it enables devices to transmit the same amount of data in fewer bits. There are a variety of data compression techniques, but only a few have been standardized.

The **International Telecommunications Union** (**ITU**) has defined a standard data compression technique for transmitting faxes (Group 3 standard) and a compression standard for data communications through modems (V.42bis). In addition, there are file compression formats, such as ARC and ZIP. Backup utilities, spreadsheet applications, and database management systems also use data compression. Certain types of data, such as bit-mapped graphics, can be compressed to a small fraction of their normal size.

Byte caching (sometimes referred to as dictionary or delta-based compression) is a combination technique that relies on a low-level cache of small, sub-application-object pieces of information to detect compressible, repetitive patterns in application cache traffic. It then symbolizes those patterns with a token, and sends the token in lieu of the bulky traffic; tokens typically are a byte or two and symbolize large blocks (e.g., 64KB). The cache on the far end matches the token with the original block of data, reconstitutes the traffic, and sends it on to the application or user (whichever is appropriate).

Protocol Optimization

Protocol optimization aims to reduce latency by removing inefficiencies in key protocols. For example, **TCP** and **HTTP** protocol optimization make Web traffic more efficient over the WAN by removing the unnecessary roundtrips that the protocols introduce as part of their set-up processes.

Load Balancing

Load balancing is a technique (usually performed by load balancers) to spread work among two or more computers, network links, central processing units (CPUs), hard drives, or other resources, in order to get optimal resource utilization, throughput, or response time. These tunable pools of infrastructure resources are managed by a combination of resource capacity metrics and load-balancing algorithm.

Typical industry standard load balancing algorithms available today include the following:

- Round Robin
- Least Connections
- Fastest Response Time
- Weighted Round Robin
- Weighted Least Connections
- Custom rating values assigned to individual servers in a pool, for example server ratings based on delay measurements provided by SNMP or other communication mechanism

Application Server Offload

Application server offload services scale applications by offloading processing tasks from the application servers to purpose built hardware and software devices. For example, compression computations consume CPU resources on servers. Many vendors offload those computations onto purpose-built hardware that performs compression at wire speeds.

P1328: Utility Services

Services use various common filtering, aggregation and data transformation techniques. The techniques in the following subsections are not an exhaustive set but they are of particular use for environments with constrained resources such as the tactical edge.

Smart Content Filtering

Smart filtering and aggregation services, in conjunction with **Quality of Service** (**QoS**) mechanisms, are needed at key information distribution nodes, such as airborne **command and control** (**C2**) centers (e.g., AWACS) at the tactical edge, to effectively and efficiently distribute information across the wide area network (WAN) and to/from end users on a priority basis.

Smart filtering services enable fine grain filtering based on the full content of each message. With such pinpoint filtering, users may receive just the information that they request (as long as they are authorized,) which minimizes bandwidth utilization. If smart filtering is coupled with QoS mechanisms, then the user will be able to receive just the information subscribed to on a priority basis.

Purpose-built content/message routers can provide full content monitoring and filtering on a per user and per application basis with real-time performance.

Content Aggregation

There are points in the network where information naturally aggregates as it moves towards its destination. For example, information from a squad of solders may flow through the vehicle's communication system. Further, information from a number of vehicles may flow through a battlefield node that intentionally is provisioned to have higher bandwidth and more reliable connectivity than other nodes. User generated packets are introduced to the network and move through the aggregation points, where information aggregation services are applied.

An example of an information aggregation service follows:

Rules in the aggregation point's router ingress interface identify the packets based on network service, protocol, destination, or some other unique factor. The router forwards the packets to a local application that places them into queue for that particular type of information. Periodically, with time intervals perhaps measured in 10s of seconds as dictated by mission need, the application takes the queue contents and builds an outbound packet. The constructed packet payload is the contents of the queue. It is then forwarded towards the destination using an appropriate transport protocol for the intended operational environment.

Transformation

Transformation includes translation between transport mechanisms or data formats as well as protocol mediation. Examples include the following:

- Conversion between two different message formats, such as two tactical data links (e.g., Link 16 and Variable Message Format or VMF)
- Conversion between two XML data formats

Standards such as **XSLT** enable transforming the XML content from one provider to another XML data mode that another consumer can use. The NCES Adapter Library translates information formats from popular standards to XML and translates from XML to other popular information format adapters (provided by the NCES Mediation Services product line). For more detail see the XSLT [P1106] perspective.

Compression

Compression has important applications in the areas of data transmission and data storage. The number of applications processing large volumes of data is increasing, while the proliferation of communication networks is resulting in greater transfer of data over communication links. Compressing data, both during transmission and while at rest, often leads to reduced costs associated with data transportation and storage.

Reducing the amount of data transmitted has the effect of increasing the capacity of the communication channel. This additional capacity may be used to transport additional data or in some cases allow for reduced queuing time for more critically important messages. The additional capacity also allows for additional error detection and/or correction data which increases robustness and reliability of the communication channel.

Similarly, compressing a file to half of its original size is equivalent to doubling the capacity of the storage medium. It may then become feasible to store the data at a higher, thus faster, level of the storage and reduce the load on the input/output channels of the computer system. The more that storage space is conserved, the more storage is available for other uses.

There are various algorithms for data compression. While, in principle, it is possible to use any general purpose compression algorithm on any type of data, many are unable to achieve significant compression on data that is not of the form for which they were designed to compress. The ability to compress depends on the inherent redundancy in the information to be compressed.

Compression algorithms fall into two categories, ***lossy*** and ***lossless***. Lossy algorithms reduce the size of the data through compression but lose fidelity in the process (often with the trade-off of increased compression of the data). On the other hand, lossless algorithms reduce the size of the data through compression techniques that result in no loss of fidelity or accuracy of the data. In other words, lossless algorithms allow for exact recreation of the data to its state before compression. Both categories of data compression are useful depending on the given requirements.

The selection of an appropriate compression algorithm for a given application depends on a number of parameters including redundancy within the data, noise within the data, tolerance to the loss of fine detail, available bandwidth, storage capacity, and the speed of the compression and decompression processes. Shannon's Theorem and subsequent algorithm standards relate all these factors; Shannon's Theorem also sets theoretical bounds on the possible compression available without introducing errors which would distort the content.

For example, a binary string of ones and zeros is generally not compressible unless there are long strings of repeated ones or zeros imbedded in it. Given simple redundancy at the bit level, run length encoding, which replaces the string by the symbol and the number of repeats, is possible. Alphabetic text in a human language has slightly more complicated redundancy and a lossless technique called Huffman coding is preferred. There are likewise specialized algorithms for video, audio, and graphics such as used in the following standards MPEG-2, Ogg Vorbis, and JPEG.

Best Practices

- BP1711: Use the **CES** Mediation Service, or a locally hosted copy, when **XML** document translation between **schemas** is a necessity.
- BP1712: Register developed mappings in the **DoD Metadata Registry**.

P1329: Node Data Strategy

One of the key differentiators in the net-centric paradigm is the treatment of data as a key architectural element with particular attention on how data interoperates among different **Components**, **Nodes** and **Systems** in a net-centric enterprise.

The DoD ***Net-Centric Data Strategy (NCDS)*** [R1172] lays out specific approaches to achieve net-centric goals to provide visible, accessible, understandable, trusted and governable data. Common approaches allow Components and Nodes to handle data across multiple technical and organizational boundaries.

The Relationship to the DoD Net-Centric Data Strategy [P1299] perspective in Part 1: Overview [P1286] briefly describes the relationship between NESI and the DoD NCDS. The Net-Centric Data Strategy (NCDS) [P1204] perspective in Part 3: Migration [P1198] and the Data [P1244] perspectives supporting the ***ASD(NII) Net-Centric Checklist*** Data Tenets (P1244, P1250, P1252, P1253, P1254, P1256, P1257 and P1258 in NESI Part 2: Traceability [P1288]) contain detailed information including Guidance and Best Practices.

NCDS emphasizes developing community-based (versus enterprise-wide) data interoperability standards through collaborative governance forums known as **Communities of Interest** (**COIs**). DoD Directive 8320.2, ***Data Sharing in a Net-Centric Department of Defense*** [R1217] provides COI guidance in the light of achieving net-centric enterprise data goals. The Communities of Interest [P1302] perspective in Part 1: Overview [P1286] discusses how a COI shares a common vocabulary to exchange information.

For more detailed code level implementation information, see the set of perspectives related to Data [P1012] in the Part 5: Developer Guidance [P1118].

Relationship Between Data and Services

The DoD NCDS includes using services as a means of making any visible data accessible by the community or enterprise users. Such services could provide access either to mission data or to metadata describing the data or access to other available services or to their inventories. For example, a COI or a Program may choose to implement a utility service to transform or translate data.

Role of Node Infrastructure

Node infrastructure plays a key role in implementing a net-centric data strategy. It provides persistent information for data, as well as for any **metadata** that describes the data or the services available to access the data. Mission data access is not necessarily the same as metadata access; explicitly call out each interface, one a mission service and the other an infrastructure service. In other words, XML schemas, catalogs, etc., often live on a different server than the mission content. Node infrastructure also provides technological means of delivering data from the source to the consumer; e.g., using **Web** or messaging infrastructure on top of the underlining network to provide the conduit. The infrastructure delivers data via options including unchanged or transformed, within the Node or across Node boundaries, within the community or for the wider enterprise. Node infrastructure also provides all the necessary support and measures for the implementation of data security, management, fault tolerance and diagnostics.

Security Considerations

For security considerations related to data at rest see the Data at Rest [P1360] perspective in Part 5: Developer Guidance [P1118]. For security considerations for data in transit, see the Black Core [P1152], Confidentiality [P1340], Design Tenet: Encryption and HAIPE [P1247], and Public Key Infrastructure (PKI) and PK Enable Applications [P1061] perspectives.

Management Considerations

The DoD ***Net-Centric Data Strategy*** and the DoD ***Defense Information Enterprise Architecture (DEIA)*** [R1335] both address data management. The guidance in these references establishes metadata and schema registries and repositories which specify the structure of the data in question. The guidance also provides the overall governance and management processes for the registration and deposition of metadata and schemas

that makes the data visible and discoverable through directory services. The Security and Management [P1331] perspective contains additional related considerations on this topic.

Data management may also require managing multiple data registries and repositories, including federated configurations. One approach combines a locally-centralized Node data registry and repository with search or syndicated publication of data records in other registries and repositories.

Effective net-centric data management makes data visible, discoverable and accessible. Open standards such as **Extensible Markup Language** (**XML**) and ***Structure of Management Information*** (***SMI***; see RFC 2578) prescribe using metadata for specifying ordinary metadata, in turn (i.e., meta-metadata). Ensuring such standardized meta-metadata is common across all components, applications and services, helps component designers and architects understand the schemas and ordinary metadata, aiding data reuse so encoded from other components and services. In addition to making data visible, discoverable and accessible, metadata can establish data provenance and freshness through Data Stewardship processes.

In addition to these primary net-centric capabilities, data management includes configuration of content discovery and syndication that make data visible and discoverable through search or publication services.

It is often not possible to decouple the management of mission data often from management of the local computing infrastructure. Such computing infrastructure includes the file system or database and any associated user environment. Consider management of the local Web infrastructure when using **Web services** to expose the data and provide access.

Storage infrastructure management may have a major impact on mission data, since data challenges at the tactical edge often involve both storage and access to storage infrastructure. Management of databases and storage area networks goes beyond configuration; it also includes the necessary performance and fault management, such as in the following examples.

- ***Caching/Proxies/Distributed Masters***: use of content distribution constructs to deploy data closer to its consumers selectively
- ***High-Speed Transactions***: use of high-performance data storage constructs with transactional semantics to ensure producers and consumers are correctly synchronized

P1138: Node Transport

A **Node** provides a transport infrastructure shared among the **components** within the Node, implements **Global Information Grid** (**GIG**) **Information Assurance** (**IA**) boundary protections, and is **Internet Protocol Version 6** (**IPv6**) capable. In some cases, guidance may seem rudimentary, but history demonstrates that configuration errors for such rudimentary aspects are often the cause of interoperability, integration, and IA issues.

Transport elements a Node provides are obviously essential in achieving net-centricity, but they also play a key role in minimizing interoperability issues.

Security Considerations

The **DISA Security Technical Implementation Guides** (**STIGs**; http://iase.disa.mil/stigs/stig/index.html) are applicable in several places throughout the NESI Part 4 Node Transport perspectives. The STIGs frequently change to include newly discovered vulnerabilities and as the current "state of the art" is refined. Consult the program-applicable STIGs and monitor them periodically for updates as a fundamental part of design activities.

For an overview of general security considerations, see the Enterprise Security [P1332] perspective. For additional detail, see the Data, Application and Service Integrity [P1338] perspective.

Management Considerations

For general management considerations, see the Security and Management [P1331] and Enterprise Management [P1330] perspectives. For additional detail, see the following perspectives:

- Design Tenet: Decentralized Operations and Management [P1276]
- Design Tenet: Enterprise Service Management [P1278]
- Design Tenet: Differentiated Management of Quality-of-Service [P1265]
- Traffic Management [P1356]

Detailed Perspectives

Transport elements that a Node provides are obviously essential in achieving net-centricity but also play a key role in minimizing interoperability issues. The following perspectives describe several Transport elements:

- Physical and Data Link Layers [P1348]
- Network layer [P1349]
- Transport Layer [P1350]
- Subnets and Overlay Networks [P1351]
- Network Services [P1353]
- Application Layer Protocols [P1355]
- Mobility [P1141]
- Traffic Management [P1356]

Guidance

- G1584: Provide a transport infrastructure that is shared among **components** within the Node.
- G1585: Provide a transport infrastructure for the Node that implements **Global Information Grid** (**GIG**) **Information Assurance** (**IA**) boundary protections.

Best Practices

- BP1704: Consult the applicable **Security Technical Implementation Guidance** (**STIG**) documents as a fundamental part of design activities, and monitor the STIGs periodically for updates.

P1348: Physical and Data Link Layers

As data flows to and from a computer (typically via Ethernet although there are other choices like asynchronous transfer mode or ATM; Sonet; and the IEEE 802.11 family) it moves through a modulator-demodulator device. This device structures the data into electronic signals that can be carried over physical communications media. This communication media may include copper wire, fiber optic cable, or wireless (such as microwaves, laser, or radio waves).

The data link layer is responsible for encoding bits into packets prior to transmission and then decoding the packets back into bits at the destination. Bits are the most basic unit of information in computing and communications. Packets are the fundamental unit of information transport in all modern computer networks, and increasingly in other communications networks as well.

The data link layer is also responsible for logical link control, media access control, hardware addressing, error detection and handling and defining physical layer standards. It provides reliable data transfer by transmitting packets with the necessary synchronization, error control and flow control.

The data link layer is divided into two sublayers: the media access control (MAC) layer and the logical link control (LLC) layer. The former controls how computers on the network gain access to the data and obtain permission to transmit it; the latter controls packet synchronization, flow control and error checking.

The data link layer is where most local area network (LAN) and wireless LAN technologies are defined. Popular technologies and protocols generally associated with this layer include the following.

- Ethernet
- Token Ring
- FDDI (fiber distributed data interface)
- ATM
- SLIP (serial line Internet protocol)
- PPP (point-to-point protocol)
- HDLC (high level data link control)
- ADCCP (advanced data communication control procedures).

Descriptions of a few of the possible standards and media follow.

IEEE 802 Standards

The services and protocols specified in IEEE 802 map to the lower two layers (Data Link and Physical) of the seven-layer Open Systems Interconnection (OSI) networking reference model. In fact, IEEE 802 splits the OSI Data Link Layer into two sub-layers named Logical Link Control (LLC) and Media Access Control:

- Data link layer
  - LLC Sublayer
  - MAC Sublayer
- Physical layer

Fiber Optic

Fiber optic related standards include the following.

- FDDI: ANSI X3T9.5 (Fiber Distributed Data Interface)
- SDH: ITU G.707 & G.708 SDH (Synchronous Digital Hierarchy; international form of SONET) SONET: Telcordia GR-253-CORE (Synchronous Optical Networking; Bell System form of SDH)
- ANSI T1.105-1991, *Digital Hierarchy - Optical Interface Rates and Formats Specification (SONET)*
- Fibre Channel: ANSI NCITS T11 (formerly X3T9.3) (mostly for storage area networks or SANs)

- GIG Ethernet: IEEE 802.3-2005 (also known as 802.3z; the fiber optic variants collectively are known as 10000BASE-X)

Tactical Data Links (TDL)

Joint Staff approved, standardized wireless/radio communications links suitable for transmission of digital information. Current practice is to characterize a tactical data link by its standardized message formats and transmission characteristics. TDLs interface two or more command and control or weapons systems via a single or multiple network architecture and multiple communications media for exchange of tactical information. Examples are Link 16 and **Situation Awareness Data Link** (**SADL**).

For more information see the Integration of Non-IP Transports [P1151] perspective.

SensorNets

A sensor network, or SensorNet is a network consisting of spatially distributed autonomous devices using sensors to monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants, at different locations cooperatively. More simply stated, it is a network where the source data is sensor data. SensorNets are often wireless networks. Wireless SensorNets can use any type of radio transmission on any protocol but most frequently employ **IP** data transfer.

Radio/Waveforms

IP network traffic can be conveyed over any radio. The legacy serial transmissions easily send and receive packets. Formatted radios such as Link-16 and others can also transfer packets but the packets must be "fit" into the format structure.

With the rise of software defined radios, the **NetOps** administrator or commander has the opportunity to select dynamically the kind of media communications technology most appropriate for use in the local sub-network infrastructure. This enables matching the Quality of Service (QoS) and **Information Assurance** goals to the underlying capabilities of the media communications.

A software defined radio (SDR) can receive or transmit signals in the radio frequency (RF) spectrum, but its signal-modulation methods depend on software loaded into the radio. Today, SDRs rely mainly on traditional circuits to process RF signals; but day by day, software gets closer to the antenna. A typical SDR comprises RF front-end circuits that connect to analog-to-digital converters (ADCs) on the receive side and digital-to-analog converters (DACs) on the transmit side. These converters connect to a signal processing subsystem that contains general-purpose or reconfigurable processors.

The processor software implements wireless standards, or "waveforms," such as Global System for Mobile communications (GSM), Code Division Multiple Access (CDMA) or the Single Channel Ground and Airborne Radio System (SINCGARS.) As long as the RF front-end circuits and the ADCs and DACs operate with a wide enough bandwidth, designers can modify the radio's capabilities simply by updating its software.

The **Joint Tactical Radio System** (**JTRS**) is a family of software-programmable tactical radios. They will provide combat personnel with voice, data, and video communications that are interoperable among all battlefield participants regardless of the branch of service.

In the case of a serial radio it will transfer packets at its designed channel data rates. So a 56,000 bits per second (56k bps) modem that is interfaced to a 56k bps radio or telephone line channel will transfer data at 56k bps. In the case of formatted radios this is not necessarily true. For example a user of a time slotted radio who has only one time slot every 12 seconds will have available the data rate in the time slot in bps divided by 12. Thus, these types of radios will change network performance.

P1349: Network Layer

The network layer is the third layer of seven in the Open Systems Interconnection (OSI) model [R1256] and the third layer of five in the **TCP/IP** model. These reference models are stacked architectures which allow separation of functions and thus make it easier from the software point of view to insert, replace, and separate software functional modules. In all of the models, the network layer responds to service requests from the transport layer and issues service requests to the data link layer.

In essence, the network layer is responsible for end-to-end (source-to-destination) packet delivery, whereas the data link layer is responsible for node-to-node (hop-to-hop) frame delivery.

The network layer provides the functional and procedural means of transferring variable length data sequences from a source to a destination via one or more networks while maintaining the quality of service and error control functions.

Detailed Perspectives

- Internet Protocol [P1139]
- IP Routing and Routers [P1143]
- Integration of Non-IP Transports [P1151]

P1139: Internet Protocol (IP)

The commercial **Internet** and U.S. Department of Defense (DoD) networks are built upon the **Internet Protocol** (**IP**). Today, these networks are based on version 4 of this protocol (IPv4). The primary motivation for embracing the next generation of IP (version 6 or IPv6) is due to the explosive growth of the Internet. The **Assistant Secretary of Defense for Networks and Information Integration**, **ASD(NII)**, has a goal which includes adapting Internet and **World Wide Web** constructs and standards with enhancements for mobility, surety, and military unique features (e.g., precedence, preemption) as one of nine *Net-Centric Attributes* [R1180]. IP is among the most fundamental of protocols needed for **Global Information Grid** (**GIG**) interoperability. There are, however, a number of interoperability challenges emerging as DoD usage of IP networking continues to expand.

IPv4

IPv4, the first widely deployed version of the Internet Protocol, currently is the dominant network layer protocol on the Internet and, apart from IPv6, it is the only standard internetwork-layer protocol used on the Internet. The **Internet Engineering Task Force** (**IETF**) described IPv4 in a September 1981 Request for Comments (IETF RFC 791). DoD also standardized IPv6 as MIL-STD-1777 dated 12 August 1983 (canceled 5 December 1995).

IPv4 is a data-oriented protocol for use on packet switched internetworks (e.g., Ethernet). It is a best effort protocol in that it does not guarantee delivery. IPv4 also does not make any guarantees on the correctness of the data; this may result in duplicated packets or packets delivered out of order. An upper layer protocol (e.g., **TCP** or, in part, **UDP**) needs to address these aspects.

Broadcast, Multicast

In computer networking, broadcasting refers to transmitting a packet that (conceptually) every device on the network will receive. In practice, the scope of the broadcast is limited to a broadcast domain. IPv4 supports broadcast, but IPv6 does not include it in the newer standard.

Multicast is the delivery of information to a group of destinations simultaneously using the most efficient strategy to deliver the messages over each link of the network only once, creating copies only when the links to the destinations split. As opposed to broadcast, multicast only sends information to a limited set of destinations.

IPv6

The Internet has been growing at an exponential rate, roughly doubling in size every year. Devices connected to the Internet are assigned globally unique addresses, and the available address space is rapidly becoming exhausted. IPv4 uses 32-bit addresses, constraining the number of unique addresses available as public Internet addresses; an IPv4 address shortage is inevitable. The IETF, to solve the address shortage problem and to provide other IP improvements, embarked on developing IPv6 to replace IPv4 after a long dual use transition period. IPv6 is already widely used in Asia, and manufacturers sell dual stack routers which process both IPv4 and IPv6 stacks.

IPv6 development supports the continued growth of the Internet by using 128-bit addresses to provide essentially unlimited address space. In addition, other improvements were made relative to IPv4, based on a generation of experience. Some of these other improvements are listed below:

- ***Streamlined processing within routers*** - The IPv6 protocol has a simplified header and the larger address allows summarizing routes in a hierarchical manner. This can dramatically reduce the size of routing tables and improve the performance of routers. IPv6 tries to make it easier to build very fast routers. IPv6 has no header checksum for routers to update, has no fragmentation in routers, has no options in the basic IPv6 header, and has a 64-bit word size.

- ***More efficient multicast support*** - All IPv6 implementations must support multicast. In addition, an added capability limits the scope of multicast transmissions. The addition of anycast addresses to IPv6 is a major development because anycast messages go only to one member of a defined group of multiple addresses, rather than to each member.

- ***Native mobility support*** - IPv6 has increased support for mobility and ad hoc networking, which is lacking or limited in IPv4. The IPv6 protocol provides an improved version of Mobile IP, which allows mobile computers to

connect to the network at different locations without disrupting communications (elimination of "triangle routing" for mobile IP).

- ***Mandatory security features*** - All IPv6 implementations must support the IP Security (IPsec) features for data integrity and confidentiality (end-to-end, IP-layer authentication and encryption are possible). IPsec is available but optional for IPv4.

- ***Autoconfiguration*** - It is possible to configure the IP addresses and other network-related parameters automatically with or without separate servers. While IPv4 does have **Dynamic Host Configuration Protocol** (**DHCP**), some applications, such as IP Telephony, cannot operate through DHCP and DHCP is not scalable.

- ***Improved Neighbor Discovery*** - The IPv6 Neighbor Discovery (ND) provides a number of significant improvements over the IPv4 Address Resolution Protocol (ARP). ARP worked as a link-layer protocol using network broadcasts which link-layer bridges forward. For large subnets, ARP sometimes creates "broadcast storms" crowding out all useful network traffic for some period of time. Also ARP is insecure; there is no way to verify that a machine responding to an ARP query really is the correct machine; the result is that it is easy to steal traffic destined to another machine. ND on the other hand runs over IPv6 using multicasting, which is media independent. It is possible to constrain ND to where it is needed so as not to create broadcast storms. ND can work with IP Security to get authenticity and/or confidentiality guarantees.

- ***Hierarchical Addressing and Route Summarization*** - The IPv6 addressing structure differs significantly from IPv4. IPv6 supports improved hierarchical addressing with route summarization, address renumbering and multi-homed sites. These features have the potential to simplify network configurations and reconfigurations. Route summarization permits routers to exchange much less reachability information over the network, reducing router overhead traffic. This is of obvious benefit for tactical RF links. IPv4 already realizes some benefits of route summarization through a combination of Classless Interdomain Routing (CIDR) and hierarchical network assignments. IPv6 hierarchical addressing may require considerable adaption for mobile, multi-hop networks that involve movement across subnets. A more detailed analysis is needed to assess the value of hierarchical addressing in IPv6 for DoD mobile networks and RF subnets.

Additional IPv6 Information Sources

The following IETF Request For Comments documents represent a few of the RFCs available via the IETF RFC Index (created on 14 March 2009; http://www.ietf.org/iesg/1rfc_index.txt).

- RFC 4291, Draft Standard, *IP Version 6 Addressing Architecture*, February 2006
- RFC 3587, Informational, *IPv6 Global Unicast Address Format*, August 2003
- RFC 2375, Informational, *IPv6 Multicast Address Assignments*, July 1998
- RFC 2460, Draft Standard, *Internet Protocol, Version 6 (IPv6) Specification*, December 1998
- RFC 4861, Draft Standard, *Neighbor Discovery for IP version 6 (IPv6)*, September 2007
- RFC 4862, Draft Standard, *IPv6 Stateless Address Autoconfiguration*, September 2007
- RFC 4443, Draft Standard, *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*, March 2006

Detailed Perspective

The following perspective addresses transition from IPv4 to IPv6:

- IPv4 to IPv6 Transition [P1140]

Guidance

- G1600: Obtain **Internet Protocol Version 6** (**IPv6**) addresses to use for DoD IP addressable resources from **DISA**.

P1140: IPv4 to IPv6 Transition

A 9 June 2003 **ASD(NII)**/DoD **CIO** memo, *Internet Protocol Version 6 (IPv6)*, [R1190] was the first in a series of memos addressing DoD transition to **IPv6** and establishing IPv6 as the next generation network protocol for DoD. The transition goal originally was Government FY 2008; however, transition planning is still under way. The DoD IPv6 Transition Office in the **Defense Information Systems Agency** (**DISA**) is responsible for master transition plan development, acquiring **Internet Protocol** (**IP**) addresses, providing necessary infrastructure and technical guidance, and ensuring the use of unified solutions across DoD to minimize cost and interoperability issues. DoD components are developing component transition plans and are providing guidance and governance to programs. There are Milestone Objectives (MOs) outlined for the gradual and controlled transition of the **enterprise**. Currently only those systems approved as MO1 pilots are allowed to switch to IPv6 in operational environments.

To enable this transition, as of 1 October 2003 all **Global Information Grid** (**GIG**) assets being developed, procured, or acquired shall be IPv6 capable (while retaining compatibility with IPv4). The ***DoD IPv6 Working Group*** is coordinating IPv6 implementation issues through formal standards bodies. A list of the standard IPv6 specifications approved for use in DoD networks so that they become "IPv6 capable" is in the **Defense IT Standards Registry** (**DISR**).

> ***Note:*** *The Civilian Agency Acquisition Council and the Defense Acquisition Regulations Council issued a ruling effective 10 December 2009 amending the Federal Acquisition Regulation (FAR) to require including IPv6 compliant products in all new information technology (IT) acquisitions using an Internet Protocol according to the Federal Register Volume 74, Number 236 (see http://edocket.access.gpo.gov/2009/pdf/E9-28931.pdf)*

The IPv6 Working Group tasks include preparing an IPv6 transition plan for the Node infrastructure as well as the transport users within the Node in coordination with the **Component** and DoD transition plan; the Node IPv6 transition plan is subject to review and approval by the appropriate IPv6 transition authority. Coordination is essential to ensure that the intermediate network infrastructures are IPv6 capable in the planned timeframe, and similarly for other-end network infrastructures for known system interfaces. The Node's IPv6 transition plan should consider applicable DoD Component IPv6 transition plans, IPv6 working group products, and interoperability testing. The net-centric concepts of loose coupling and discoverable services may be impacted by the transition to IPv6 if services begin depending on IPv6-specific features. Identify services which utilize IPv6 features and which may perform differently if accessed via an **Internet Protocol Version 4** (**IPv4**) infrastructure.

IPv6 transition has an impact on many transport infrastructure components. The IPv6 Transition Plan for a Node should include transition of all impacted network elements including the **Domain Name System** (**DNS**), routing, security, and dynamic address assignment.

The transition between today's IPv4 Internet and a future IPv6-based one will be a long process during which both protocol versions will coexist. The **Internet Engineering Task Force** (**IETF**) created the NGTrans Working Group (now concluded) to identify IPv6 transition issues and propose technical solutions to achieve it. Ongoing IPv6 operations standards, tools, techniques and best practices derived from both this work and experience with the 6bone testbed (also now retired) are the responsibility of the V6Ops Working Group.

No single general rule applies to the IPv4 to IPv6 transition process. In some cases, moving directly to IPv6 will be the answer. For instance IPv6 could be pushed by a political decision to extend the number of IP addresses to sustain the economic growth of a country. Another example is the large-scale deployment of a new IP architecture (such as mobile or home networking) to provide disruptive applications and innovative services.

Other transition plans will enable a gradual interoperability between IPv4 and IPv6 as transition evolves. Here, Internet Service Providers (ISPs) and enterprises will prefer to preserve the heavy investments made to deploy IPv4 networks.

Some studies foresee that the transition period will last between today and 2030-2040. At that time, IPv4 networks should have totally disappeared.

The NGTrans Working Group defined three main transition techniques.

- ***Dual-stack network***. The **dual stacking** approach requires hosts and routers to implement both IPv4 and IPv6 protocols. This enables networks to support both IPv4 and IPv6 services and applications during the transition period in which IPv6 services emerge and IPv6 applications become available. At the present time, the dual-stack approach is a fundamental mechanism for introducing IPv6 in existing IPv4 architectures and will remain heavily used in the

near future. The drawback is that an IPv4 address must be available for every dual-stack machine. This is unfortunate, since IPv6 was developed precisely due to the scarcity of IPv4 addresses.

- ***Tunneling***. **Tunneling** enables the interconnection of IP clouds. For instance, a tunnel can interconnect separate IPv6 networks through a native IPv4 service. A border router encapsulates IPv6 packets before transportation across an IPv4 network and decapsulates the packets at the border of the receiving IPv6 network. Tunnel configuration can be static, dynamic, or implicit (6to4, 6over4). The Tunnel Broker (TB) approach automatically can manage tunnel requests coming from the users and ease the configuration process. The Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) is a recent technique to avoid manual tunnel configuration. In later stages of transition, tunnels will also interconnect remaining IPv4 clouds through the IPv6 infrastructure.

- ***Translation mechanism***. Translation is necessary when an IPv6 only host has to communicate with an IPv4 host. At the least, the IP header requires translation, but the translation will be more complex if the application processes IP addresses; in fact such translation inherits most of the problems of IPv4 network address translators. Application-Level Gateways (ALGs) translate embedded IP addresses, recompute checksums, etc. Stateless IP/ICMP Translation (SIIT) and Network Address Translation-Protocol Translation (NAT-PT) are the associated translation techniques. A blend of translation and the dual stack model, known as Dual Stack Transition Mechanism (DSTM), addresses the case where insufficient IPv4 addresses are available. Like tunneling techniques, translation implementation can be in border routers and hosts.

There are many ways to "mix and match" this complex set of coexistence and transition techniques.

Guidance

- G1586: Provide a transport infrastructure for the Node that is **Internet Protocol Version 6** (**IPv6**) capable in accordance with the appropriate governing transition plan.
- G1587: Prepare an **Internet Protocol Version 6** (**IPv6**) transition plan for the Node.
- G1588: Coordinate an **Internet Protocol Version 6** (**IPv6**) transition plan for a Node with the **Components** that comprise the Node.
- G1589: Address issues in the appropriate governing **Internet Protocol Version 6** (**IPv6**) transition plan as part of the IPv6 Transition Plan for a Node.
- G1590: Include transition of all the impacted elements of the network as part of the **Internet Protocol Version 6** (**IPv6**) Transition Plan for a Node.
- G1591: Prepare IPv6 Working Group products as part of the **Internet Protocol Version 6** (**IPv6**) transition plan for a Node.
- G1592: Include interoperability testing in the plan as part of the **Internet Protocol Version 6** (**IPv6**) transition plan for a Node.
- G1599: Simultaneously support **Internet Protocol Version 4** (**IPv4**) and **Internet Protocol Version 6** (**IPv6**) in the Node's **Domain Name System** (**DNS**) service.
- G1600: Obtain **Internet Protocol Version 6** (**IPv6**) addresses to use for DoD IP addressable resources from **DISA**.

Best Practices

- BP1705: Design **Domain Name System** (**DNS**) infrastructure in accordance with appropriate governing **Internet Protocol Version 6** (**IPv6**) Transition Office requirements.
- BP1923: Employ an operating system that supports simultaneously **IPv4** and **IPv6**.

P1143: IP Routing and Routers

Routers not only provide the main connection to the **Global Information Grid** (**GIG**), but they also are a first line of **computer network defense**. These complex devices provide security filtering, address management, network management, and time synchronization. A ***GIG Router Working Group (GRWG)*** is addressing implementation issues.

Components should be able to operate in a heterogeneous environment. The presence of **Internet Protocol Version 4** (**IPv4**) and **Internet Protocol Version 6** (**IPv6**) packets and services in a dual-stack environment should not cause a degradation of application performance.

Routing capabilities in real-time, dynamic and mobile environments, such as at the tactical edge, are still in their infancy. A variety of working groups, such as the GRWG and the Office of the Secretary of Defense ***Joint Airborne Network (JAN) Working Group***, continue to define, prototype and refine routing capabilities.

Routing is an umbrella term for the set of protocols that determine the path that data follows in order to travel across multiple networks from a source to a destination. Data routing from source to destination is through a series of routers and across one or more networks.

Routing protocols enable a router to build up a forwarding table that correlates final destinations with next hop addresses. Routing protocols specify a set of messages routers exchange; the message contents allow a router to inform its peers about the **IP** routes it knows and allow that knowledge to spread throughout the network.

An IP network administered by a single authority is called an autonomous system (AS); such a network could run an Interior Gateway Protocol (IGP). However, multiple autonomous systems also need to interconnect and exchange routes among themselves to create a larger network not administered by any single authority; the public **Internet** is an example. In this case selecting routes to add to the IP forwarding table requires great flexibility; for example, path length may not be meaningful if part of that path has links with costs set by a different AS using different criteria. More important are administrative policies like the selection of preferred transit networks with which to partner. The Border Gateway Protocol (BGP) serves this environment. It allows each AS to select which other AS are the preferred choices to inject routes into its network.

When BGP routers propagate an IP route to another AS, they include the entire list of AS that have propagated the route to them, from the AS that originated the route to the current AS propagating it further. This is called the path vector and BGP is a path vector protocol. Having the entire list of AS that have propagated the route allows a BGP router to decide if the route uses its preferred transit AS or goes through an AS to avoid whenever possible. This is greater flexibility than offered by a shortest path IGP. Note that IP networking requires loop-free paths but not necessarily shortest paths; the BGP path vector guarantees loop-free paths.

Example routing protocols follow.

Open Shortest Path First (OSPF) Protocol

The OSPF protocol is a hierarchical interior gateway protocol (IGP) for routing in Internet Protocol, using a link-state in the individual areas that make up the hierarchy. The protocol uses a computation based on Dijkstra's algorithm to calculate the shortest path tree inside each area. OSPF is the primary means of routing in the Internet. It does not respond well to rapidly changing node connectivity and as such is not considered to be suitable for mobile, wireless military networks.

The following **Internet Engineering Task Force** (**IETF**) Requests For Comments (RFCs) provide additional information concerning OSPF:

- RFC 2328, Standard, *OSPF Version 2*, April 1998, for unicast routing
- RFC 3101, Proposed Standard, *OSPF Not-So-Stubby Area (NSSA) Option*, January 2003
- RFC 1793, Proposed Standard Extending OSPF to Support Demand Circuits, April 1995; updated by RFC 3883, *Proposed Standard, Detecting Inactive Neighbors over OSPF Demand Circuits (DC)*, October 2004
- RFC 5340, Proposed Standard, *OSPF for IPv6*, July 2008
- RFC 3137, Informational, *OSPF Stub Router Advertisement*, June 2001

- RFC 3630, Proposed Standard, *Traffic Engineering (TE) Extensions to OSPF Version 2*, September 2003; updated by RFC4203, Proposed Standard, *OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)*, October 2005
- RFC 1584, Historic, *Multicast ExtensionstoOSPF*, March 1994
- RFC 1585, Informational, *MOSPF: Analysis and Experience*, March 1994

Border Gateway Protocol (BGP)

BGP is the standard protocol for routing between autonomous system (AS) domains. It works by maintaining a table of IP networks or "prefixes" which designate network reachability among autonomous systems. It relies on **Transmission Control Protocol** (**TCP**) sessions between BGP peers and does not have an automatic neighbor discovery capability. As the number of AS domains increases, BGP may take longer to converge than OSPF after a routing change occurs.

The following IETF RFCs provide additional BGP information:

- RFC 4271, Draft Standard, *Border Gateway Protocol 4 (BGP-4)*, January 2006
- RFC 1772, Draft Standard, *Application of Border Gateway Protocol In the Internet*, March 1995
- RFC 4760, Draft Standard, *Multiprotocol Extensions for BGP-4*, January 2007
- RFC 3107, Proposed Standard, *Carrying Label Information in BGP-4*, May 2001
- RFC 5065, Draft Standard, *Autonomous System Configurations for BGP*, August 2007
- RFC 2439, Proposed Standard, *BGP Route Flap Damping*, November 1998
- RFC 4659, Proposed Standard, *BGP-MPLS IP Virtual Private Network (VPN) Extension for IPv6 VPN*, September 2006
- RFC 4797, Informational, *Use of Provider Edge to Provider Edge (PE-PE) Generic Routing Encapsulation (GRE) or IP in BGP/MPLS IP Virtual Private Networks*, Jan 2007
- RFC 4456, Draft Standard, *BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)*, April 2006
- RFC 4384, Best Current Practice, *BGP Communities for Data Collection*, February 2006

Routing Information Protocol (RIP)

RIP sends routing-update messages at regular intervals and when the network topology changes. When a router receives a routing update that includes changes to an entry, it updates its routing table to reflect the new route. The metric value for the path increases by 1, and the sender is the next hop. RIP routers maintain only the best route (the route with the lowest metric value) to a destination. After updating its routing table, the router immediately begins transmitting routing updates to inform other network routers of the change. These updates are sent independently of the regularly scheduled updates that RIP routers send.

Intermediate System - Intermediate System Protocol

The IS-IS protocol is one of a family of IP routing protocols. IS-IS is an Interior Gateway Protocol (IGP) for the Internet, used to distribute IP routing information throughout a single Autonomous System (AS) in an IP network.

IS-IS is a link-state routing protocol, which means that the routers exchange topology information with their nearest neighbors. The topology information is flooded throughout the AS, so that every router within the AS has a complete picture of the topology of the AS. This picture is then used to calculate end-to-end paths through the AS, normally using a variant of the Dijkstra algorithm. Therefore, in a link-state routing protocol, the next hop address to which data is forwarded is determined by choosing the best end-to-end path to the eventual destination.

Additional information sources include the following:

- IETF RFC 1142, Informational, *OSI IS-IS Intra-domain Routing Protocol*, February 1990
- IS-IS Protocol: Intermediate System - Intermediate System, http://www.dataconnection.com/iprouting/isisprotocol.htm

Internet Control Message Protocol (ICMP)

ICMP is a network layer Internet protocol that provides message packets to report errors and other information regarding IP packet processing back to the source. IETF has documented ICMP in RFC 792, *Internet Control Message Protocol*, September 1981.

ICMP generates several kinds of useful messages, including Destination Unreachable, Echo Request and Reply, Redirect, Time Exceeded, Router Advertisement, and Router Solicitation. If an ICMP message cannot be delivered, the message is not retransmitted to avoid an endless flood of ICMP messages.

ICMP Router-Discovery Protocol (IDRP)

IDRP uses Router Advertisement and Router Solicitation messages to discover the addresses of routers on directly attached subnets. Each router periodically multicasts Router Advertisement messages from each of its interfaces. Hosts then discover addresses of routers on directly attached subnets by listening for these messages. Hosts can use Router-Solicitation messages to request immediate advertisements rather than waiting for unsolicited messages.

IRDP offers several advantages over other methods of discovering addresses of neighboring routers. Primarily, it does not require hosts to recognize routing protocols, nor does it require manual configuration by an administrator.

Guidance

- G1601: Use configurable **routers** to provide dynamic **Internet Protocol** (**IP**) address management using the **Dynamic Host Configuration Protocol** (**DHCP**).
- G1602: Use configurable **routers** to provide static **Internet Protocol** (**IP**) addresses.
- G1604: Use configurable **routers** to provide time synchronization services using **Network Time Protocol** (**NTP**).
- G1605: Use configurable **routers** to provide **multicast** addressing.
- G1606: Manage **routers** remotely from within the **Node**.
- G1607: Configure routers according to **National Security Agency** (**NSA**) Router Security Configuration guidance.

Best Practices

- BP1699: Configure **routers** in accordance with the Network **Security Technical Implementation Guide** (**STIG**).
- BP1700: Configure **routers** in accordance with Enclave **Security Technical Implementation Guide** (**STIG**).

P1151: Integration of Non-IP Transports

Systems that are not **Internet Protocol** (**IP**) networked, such as aircraft data links (**Link-16**, **SADL**, etc.), should implement IP gateways to interoperate with the **Global Information Grid** (**GIG**) until IP is supported natively. Most such systems already have plans for transition to IP networking, and gateways are an interim measure.

Implement these gateways as **services** in accordance with ***NESI Part 5: Developer Guidance***. This does not mean that the service would be limited to request/reply or other such usage patterns. In fact, for high-frequency data, such as track reporting, a function of the service could be to set up an out-of-band communication with a subscriber.

Guidance

- G1611: Implement **Internet Protocol** (**IP**) gateways to interoperate with the **Global Information Grid** (**GIG**) until IP is supported natively for **Components** that are not IP networked.

P1350: Transport Layer

The Transport Layer traditionally is the fourth layer of the Open Systems Interconnection (OSI) Reference Model. It provides transparent transfer of data between end systems using the services of the network layer (e.g., **Internet Protocol** or **IP**) below to move packets of data between the two communicating systems.

Transmission Control Protocol (TCP)

TCP, one of the core protocols of the IP suite, provides guaranteed delivery of messages when required. TCP divides messages into packets which are acknowledged back to the sending computer. If a packet is not acknowledged TCP retransmits the package. There are many current variants of TCP; the most common is called TCP Reno. Others like TCP Westwood, TCP Peach, TCP Vegas, TCP Real, etc., address issues that TCP has with network congestion. Using TCP, programs on networked computers can create connections to one another, over which they can send data. The protocol guarantees that data the source sends will be received in the same order without any missing packets.

In addition to variants of TCP, extensions to TCP exist to optimize performance in networks with issues such as packet loss and high latency. These issues cause poor network performance when using TCP (due to issues with the TCP cumulative acknowledgment algorithm in this environment). One such extension is TCP Selective Acknowledgment (TCP SACK). TCP SACK is useful for networks where high packet loss is probable (or when packets arrive out of order), such as with mobile networks. TCP SACK attempts to increase network throughput by following a process of selective acknowledgment where the data receiver informs the sender about all segments that have arrived successfully. Thus, the sender may retransmit only the undelivered segments.

For further discussion of mobility considerations see the Mobility [P1141] perspective.

User Datagram Protocol (UDP)

UDP is a connectionless transport layer protocol that belongs to the Internet Protocol family. UDP is basically an interface between IP and upper-layer processes. Unlike TCP, UDP adds no reliability, flow-control, or error-recovery functions. However, UDP consumes less network overhead than TCP.

UDP is useful in situations where the reliability mechanisms of TCP are not necessary, such as in cases where a higher-layer protocol might provide error and flow control.

Space Communications Protocol Specifications (SCPS)

The Space Communications Protocol Specifications (SCPS) are a collection of communications protocols the Consultative Committee on Space Data Systems (CCSDS) developed to provide reliable communications in space environments. SCPS include file transfer, transport, security, and network protocols. For more information on these recommended standards, see the CCSDS Blue Books: Recommended Standards Web page.

- *Space Communications Protocol Specification (SCPS)-File Protocol (SCPS-FP)*, CCSDS 717.0-B-1, May 1999 [under consideration for removal from the CCSDS library due to lack of use at present]; ISO 15894
- *Space Communications Protocol Specification (SCPS)-Transport Protocol (SCPS-TP)*, CCSDS 714.0-B-2, October 2006; ISO 15893
- *Space Communications Protocol Specification (SCPS)-Security Protocol (SCPS-SP)*, CCSDS713.5-B-1, May 1999; ISO 15892
- *Space Communications Protocol Specification (SCPS)-Network Protocol (SCPS-NP)*, CCSDS 713.0-B-1, May 1999; ISO 1589

SCPS protocol suite development supports space channels where the round trip delay is high and the error rate can be higher than that seen on the wires and fibers used in ground networks employing **TCP/IP**. TCP has great difficulty with high error rates and high round trip delays. As a result, attempts to use alternatives including SCPS-TP commonly occur. However, using a substitute protocol creates accountability issues as it must tell the source that a message was delivered when it was not and it then takes responsibility for delivery. If ultimate delivery fails, the source does not get a final delivery notification; it gets a failure message and the sender must take an alternate action that is unexpected. Imagine tracking a time critical target, sending orders, and later finding out the orders were not delivered. For further information about the SCPS protocol suite see http://www.scps.org/.

P1351: Subnets and Overlay Networks

Subnets and overlay networks are both building blocks by which net-centric applications, data and **services** bind transport network resources to their particular needs.

The sections below cover some of the standard transport binding address-constructs, binding techniques and operational rationales used by applications, data, and services when binding to the transport infrastructure.

Subnets

Subnets are the original technique by which Internet host systems were grouped "close" together for performance and "within" security perimeters. Nodes on a subnet often also use a single media technology optimized for their local area, a **Local Area Network** (**LAN**).

Subnets are a way of structuring the network by grouping all systems that share a single local area media such as a broadband LAN, a wireless data link or fiber bundle that share a single subnet mask (**IPv4**) or prefix (**IPv6**).

A designated router represents each subnet in the larger **Global Information Grid** (**GIG**). This router is responsible for both tracking changes in the immediate global network topology and ensuring that local changes do not concern the larger GIG unless absolutely necessary.

Media Access Control (MAC) addressing and designated routers both can change as systems start up, move and shutdown; a key to successful network performance is ensuring that both addressing and router election are correct and efficient.

Subnet membership helps to ensure both information distribution performance and protection; sometimes there is a desire to extend the use of subnets beyond the normal range of a particular media. This can be accomplished through use of link layer device such as repeater or bridge, which like routers forward traffic but unlike routers do not concern themselves with the topology of the larger GIG or **IP** addresses.

Link layer devices may also serve as sub-sub-nets known as virtual local area networks or VLANs when, instead of extending the range of the local media, they partition a single local media such as broadband for performance or protection purposes. Subnets are also important for larger GIG resiliency because they enable multi-homing in which a local area network connects to the larger GIG through more than one subnet address space, represented by more than one designated router. These alternate connections create a mesh of alternate paths for traffic to use, enabling both failover capability and load-sharing.

Overlay Networks

Overlay networks are a virtual extension of the subnet concept, but instead of blocks of IP addresses they use other network identifier constructs. Formally, an overlay network is a virtual network built on top of another network. Nodes in the overlay are connected by virtual or logical links, each of which may run on top of many lower layer links in the underlying networks. Overlay networks can be created at any layer in the Transport stack, but their network location identifiers usually bind to an IP address. SPINES (see http://www.spines.org/) is an example open source general purpose overlay that can be readily tailored for various applications from the Distributed Systems and Networks lab at Johns Hopkins University.

Virtual Private Network (VPN) Overlay Networks

- ***MPLS VPNs*** - MultiProtocol Label Switching (MPLS) VPNs use special short-hand labels to create overlay networks that conform to more sophisticated forwarding policies than the default IP routing metrics. They are especially useful in limiting the variability of delay or choice of intermediate networks.
- ***IPSec VPNs*** - Internet Protocol Security (IPSec) VPNs use cryptography to tunnel sensitive information exchanges through less-trusted intermediate networks.

For further VPN content, see the Virtual Private Networks [P1149] perspective.

Content Delivery Overlay Networks

Content Delivery Overlay Networks are used for replication and synchronization; a content delivery network (CDN) is a multicast-address network that extremely efficiently distributes web content, especially for load-sharing or content with high QoS requirements such streaming audio, video, and Internet television (IPTV) programming. CDNs are, in the strictest sense, Network Layer Overlay Network because they are based on multicast addressing that is maintained by multicast-capable routers.

Application Layer Overlay Networks

The following techniques are example application layer overlay networks.

- ***P2P Overlays*** - Peer-to-peer networks are typically used for connecting nodes via largely ad hoc connections set up and labeled for each information flow of interest. These are used to build a distribution topology based on application layer protocols that advertise local availability of content. For further information on P2P concepts see http://en.wikipedia.org/wiki/Peer-to-peer.
- ***Content Routers*** - Message Router overlays match content (often represented as XML) needs to content suppliers, often through deep packet inspection that then generate the information flow labels, which are then used to select appropriate Network layer routes. In some implementations, content router(s) can distribute the content needs of all subscribers (e.g. applications and users) across the network and can optimally push the matching content to each subscriber upon publication.
- ***Disruption Tolerant Networking*** - DTN overlays use proxies to stand in for content suppliers and consumers whose network layer connectivity may be intermittent or changing. Information flow labels are assigned to either the current "best" network layer route or a temporary buffering server if one is not available. For an example of an application of DTN, see the *Disruption Tolerant Networking for Marine Corps CONDOR* paper from the Military Communications Conference, 2005 (MILCOM 2005).

Detailed Perspectives

- Broadcast, Multicast and Anycast [P1146]
- Virtual Private Networks [P1149]
- Ad Hoc Networks [P1352]

P1146: Broadcast, Multicast, and Anycast

Broadcast, **Multicast**, and Anycast are bandwidth optimizations techniques for content dissemination; they are all used to send packets of information from a source simultaneously to multiple destinations unlike Unicast which routes information from a source to a singe destination.

Broadcast

Broadcast delivers data to all addresses on a media; for example the various wired (802.3/Ethernet) and wireless (802.11/WiFi) broadcast mechanisms that use special addresses on which all host systems must receive messages. Broadcast implemetation may be at the link layer or at the network layer (available in **Internet Protocol Version 4**, or **IPv4**, but not **IPv6**) or higher layers.

Multicast

IP Multicast is the delivery of information to a group of destinations simultaneously using the most efficient strategy to deliver the messages over each link of the network only once, creating copies only when the links to the destinations split. Multicast currently supports various groups throughout the DoD to provide capabilities such as collaboration and alerting; the use of multicast addressing is growing. Multicast capability is being engineered actively into the **Global Information Grid** (**GIG**). Careful planning is still required, however, until multicast becomes ubiquitous across the entire GIG.

Anycast

Anycast (included as part of the formal IPv6 specification but implemented as external extensions to the IPv4 specification) is a network addressing and routing scheme to route data to the next router or next group of routers in a network. A combination of Anycast and Multicast can create the functionality of Broadcast in an IPv6 network.

Guidance

- G1601: Use configurable **routers** to provide dynamic **Internet Protocol** (**IP**) address management using the **Dynamic Host Configuration Protocol** (**DHCP**).
- G1610: Configure the **Dynamic Host Configuration Protocol** (**DHCP**) services to assign **multicast** addresses.

Best Practices

- BP1706: Design node networks, including the selection of **Components** and configuration, to support **multicasting** even if not currently used.

P1149: Virtual Private Networks (VPN)

Virtual Private Networks (**VPNs**) create a private "**tunnel**" within a network by encrypting traffic between specified end points. If a **Node** requires a VPN, implement it in accordance with the guidance provided in the Network **Security Technical Implementation Guide** (**STIG**). Do not place services and information intended to be broadly accessible to other **Global Information Grid** (**GIG**) Nodes behind a VPN because they will be reachable by only the Nodes that are part of the VPN.

A VPN is a private network overlaid on top of a public network (usually the **Internet**) to connect remote sites or users together. Instead of using a dedicated, real-world connection such as a leased line, a VPN uses "virtual" connections routed through the Internet from a private network (such as a company's intranet) to an authorized remote site or user (such as a company's employee that does not otherwise have direct access to the company's intranet).

The VPN overlay approach extends the subnetwork concept of using address assignment to run logical links over local media networks. Overlay VPN logical links run on top of any kind of network: local media, IP network or another overlay network. Such overlay nets and VPNs are usually optimized for performance or protection or both.

VPNs sometime use standards such as **High Assurance Internet Protocol Encryption** (**HAIPE**) and Internet Protocol Security (IPsec) for security.

Guidance

- G1667: Implement **Virtual Private Networks** (**VPNs**) in accordance with the guidance provided in the Network **Security Technical Implementation Guide** (**STIG**).

Best Practices

- BP1702: Do not place services and information intended to be broadly accessible to other nodes behind a **Virtual Private Network** (**VPN**).

P1352: Ad Hoc Networks

A wireless ad hoc network is a decentralized wireless network containing two or more participants. In some ad hoc networks, participants are willing to forward data for other participants, as in the case of Internet Connection Sharing or Mobile Ad Hoc Network (MANET). Sometimes ad hoc networks (including MANET), determine dynamically which participants forward data based on the network connectivity. This is in contrast to wired networks, in which routers perform the task of routing, and managed wireless networks, in which a special node known as an access point manages communication among other nodes.

Commercial routing protocols, such as Open Shortest Path First (OSPF) and Border Gateway Protocol (BGP), are designed and optimized for fixed infrastructures. The frequency of the message intervals to locate neighbor nodes and exchange routing tables is too low to keep up with the dynamic and mobile network state in a mobile environment or other similar unstable environments. An **Internet Protocol** (**IP**) routing protocol for mobile environments needs to interoperate with standard routing technology, detect and adapt to recurring link failures and mobility with minimal overhead and route data over the platform's multiple links to maximize throughput and reliability. For each of these requirements, the academic and research communities have done related work in the areas of MANET, multipath routing, and wireless extensions to common routing protocols. Continued research is needed to determine the best protocol settings to use (link metrics, hello intervals, dead intervals, etc.) and how to modify/extend the standard protocols to meet the requirements for mobile environments.

A MANET is a wireless ad hoc network of mobile routers (and associated hosts) connected by wireless links, the union of which form an arbitrary topology. The routers are free to move randomly and organize themselves arbitrarily; thus, the network's wireless topology may change rapidly and unpredictably.

Individual mobile networks implement their own internal MANET routing protocols which are transparent to IP (i.e., Open Systems Interconnection [OSI] Layer 3) and do not extend across mobile network boundaries. However, these mobile networks can interface with other networks using standard routing protocols, such as the OSPF protocol and BGP.

Additional Information

The following book and Internet Engineering Task Force (IETF) Requests for Comments (RFCs) provide additional information:

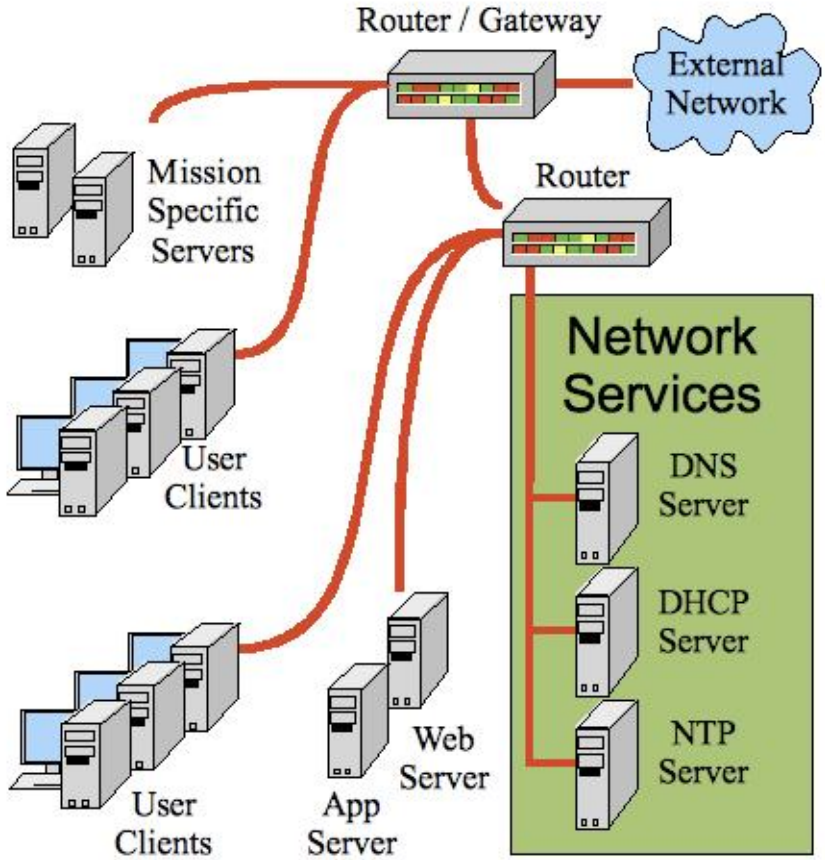
- C K Toh, *Ad Hoc Mobile Wireless Networks*, Prentice Hall Publishers, 2002.
- IETF RFC 3561, *Experimental Ad Hoc On Demand Distance Vector (AODV)*, July 2003
- IETF RFC 3684, *Topology Dissemination Based on Reverse-Path Forwarding (TBRPF)*, February 2004
- IETF RFC 4728, *Experimental Dynamic Source Routing (DSR)*, Feb 2007
- IETF RFC 3626, *Experimental Optimized Link State Routing (OLSR)*, Oct 2003

P1353: Network Services

Network services are a special category of **services** available over **Internet Protocol** (**IP**) networks to network clients (hosts) that network administrators generally manage and maintain. When network clients request to join a network, they receive various configuration parameters that enable and facilitate the use of the network. The configuration parameter distribution can be manual (i.e., via paper) or via automated protocols. Regardless of the distribution mechanism, the network client must be configured accordingly.

Network service servers predominately provide services that are generic and local in nature. For example, the local network generally provides the time service. Some newer network services have replaced older versions (i.e., **Network Time Protocol** [**NTP**] time services have replaced Time Server services, and **Domain Name System** [**DNS**] has replaced the Name Server). Any service could theoretically be categorized as a network service; however, network services generally provide a service that is important for the integrity or security of the network and the safety of its clients.

Most network services are simply represented by the name of the service and an IP address. One major exception is the **Dynamic Host Configuration Protocol** (**DHCP**) server which is responsible for providing automated distribution of the configuration parameters. Access to this server is via a special broadcast message (**`DHCPDISCOVER`**) requesting membership onto the network. Most DHCP Clients know how to obtain from the DHCP Server the list of IP addresses that provide time using the DHCP options numbers.



I1220: Common Network Services

The following table list some of the more common configuration parameters that DHCP services provide as defined by the Internet Engineering Task Force Network Working Group in RFC 2132, *DHCPOptions and BOOTP Vendor Extensions*:

Configuration Parameter	Description

DNS Servers	The DNS option specifies a list of Domain Name System name servers available to the client; list servers in order of preference
NTP Servers	The NTP option specifies a list of IP addresses indicating NTP servers available to the client. Servers should be listed in order of preference
Trivial File Transport Protocol (TFTP) Server	The TFTP option identifies a TFTP server when using the "sname" field for DHCP options in the DHCP header

Detailed Perspectives

- Doman Name System [P1142]
- Dynamic Host Configuration Protocol [P1354]
- Network Time Service [P1144]

P1142: Domain Name System (DNS)

The **Domain Name System** (**DNS**) stores the relationships of host **Internet Protocol** (**IP**) address and their corresponding domain names in the equivalent of a distributed database (used here as a simplistic concept). The most import role of the DNS is to map IP addresses to human friendly domain names and back again. For example, where **`nesi.spawar.navy.mil`** may map to an **Internet Protocol Version 4** (**IPv4**) address of **`128.49.49.225`**, the **Internet Protocol Version 6** (**IPv6**) address might be **`1080::34:0:417A`**. For more information on DNS see the Internet Engineering Task Force (IETF) *Domain Names - Concepts and Facilities* Standard (RFC 1034). DNS also performs other essential functions, such as reverse lookups (obtaining host names from IP addresses, which can be important for security) and email configuration (special DNS **Mail eXchange (MX) Records** indicate the **server** used to receive email for a host). These capabilities are fundamental to net-centric operations and are essential for other computing, network, and **Enterprise Services**.

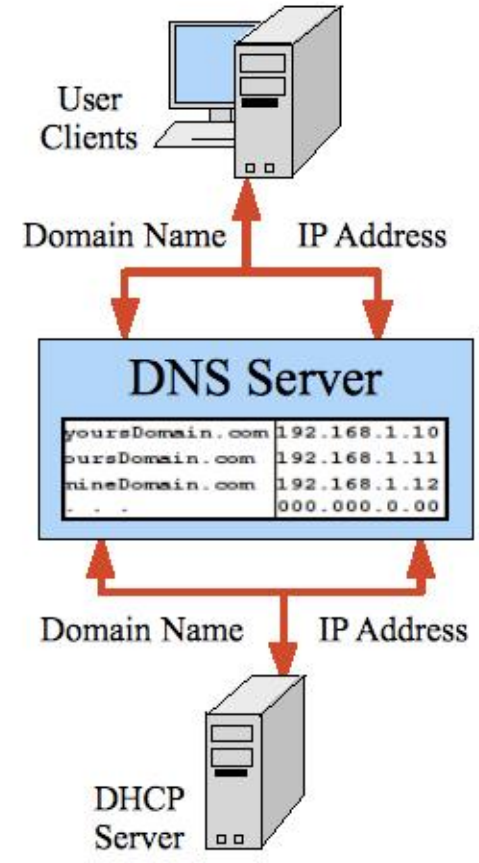
The DNS namespace is hierarchical. At each level in the hierarchy, the namespace can be divided into sub-namespaces called zones, which are delegated to other authoritative servers and which can be divided and delegated to other authoritative servers, and so on.

Each Node should implement DNS to manage hostname/address resolution within the Node, rather than use hard coded IP addresses, and use the DNS Mail eXchange (MX) Record capabilities to configure electronic mail delivery to the Node.

The DNS implementation should reflect the guidance provided in the *Domain Name System Security Technical Implementation Guide*. This **STIG** addresses implementation options such as the choice of basic DNS server types (primary, secondary, caching-only), use of a split-DNS design, location of servers in the network and relationship to other network entities, secure administration, security of zone transfers, and initial configuration.

Consider operational performance constraints, such as narrow bandwidth and intermittent connectivity, in designing the DNS for a **Node**. It may be desirable, for instance, to implement a caching-only DNS server for constrained environments.

The following image (I1221) shows a client requesting a domain name resolution as well as a **Dynamic Host Configuration Protocol** (**DHCP**) server updating DNS records.



I1221: DNS

Guidance

- G1595: Implement **Domain Name System** (**DNS**) to manage hostname/address resolution within the Node.
- G1596: Use **Domain Name System** (**DNS**) **Mail eXchange (MX) Record** capabilities to configure electronic mail delivery to the Node.
- G1598: Allow dynamic **Domain Name System** (**DNS**) updates to the Node's internal DNS service by local **Dynamic Host Configuration Protocol** (**DHCP**) **server(s)**.
- G1599: Simultaneously support **Internet Protocol Version 4** (**IPv4**) and **Internet Protocol Version 6** (**IPv6**) in the Node's **Domain Name System** (**DNS**) service.
- G1600: Obtain **Internet Protocol Version 6** (**IPv6**) addresses to use for DoD IP addressable resources from **DISA**.
- G1662: Follow the guidance provided in the **Security Technical Implementation Guide** (**STIG**) for **Domain Name System** (**DNS**) implementations.

Best Practices

- BP1597: Consider operational performance constraints in the design of the Node's **Domain Name System** (**DNS**).
- BP1663: Design a **Domain Name System** (**DNS**) in coordination with the appropriate governing **Internet Protocol Version 6** (**IPv6**) Transformation Office.
- BP1705: Design **Domain Name System** (**DNS**) infrastructure in accordance with appropriate governing **Internet Protocol Version 6** (**IPv6**) Transition Office requirements.

P1354: Dynamic Host Configuration Protocol (DHCP)

The **Dynamic Host Configuration Protocol** (**DHCP**) automates the network configuration of network devices (i.e., hosts) connected to **Internet Protocol** (**IP**) based networks. DHCP is built on the client-server model. A DHCP server allocates and manages IP addresses and delivers IP network configuration parameters (such as the default gateway, DNS servers, and other servers including time) to DHCP clients. DHCP consists of two major components:

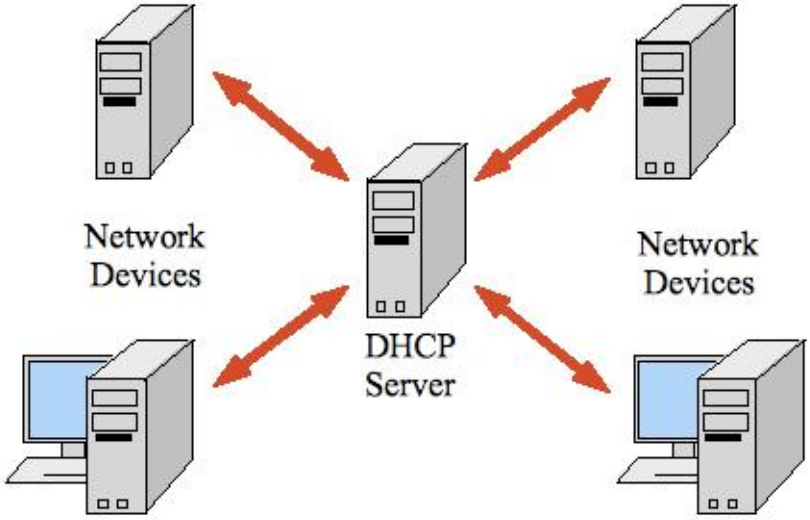
- A protocol for requesting and delivering to a DHCP client specific configuration parameters from a DHCP server
- A mechanism for managing and allocating IP addresses to DHCP clients

DHCP clients discover DHCP servers using a broadcast message rather than finding the DHCP servers in a directory. If there are multiple DHCP servers that hear the broadcast, they each can make an offer to the DHCP client to provide DHCP services. The client then chooses one of the offers; this provides a starting point for discovering all the other network services on the network.

DHCP provides three modes for allocating IP addresses. The best-known mode is ***dynamic***, in which the client receives a "lease" on an IP address for a period of time. Depending on the stability of the network, this could range from hours (a wireless network at an airport) to months (for desktops in a wired lab). At any time before the lease expires, the DHCP client can request renewal of the lease on the current IP address. A properly-functioning client will use the renewal mechanism to maintain the same IP address throughout its connection to a single network; otherwise, it may risk losing its lease while still connected, thus disrupting network connectivity while it renegotiates with the server for its original or a new IP address.

The two other modes for allocation of IP addresses are ***automatic*** (also known as DHCP Reservation), in which the address is permanently assigned to a client, and ***manual***, in which the address is selected by the client (manually by the user or any other means) and the DHCP protocol messages are used to inform the server that the address has been allocated.

Use of the automatic and manual methods generally is in situations which require finer-grained control over IP address (typical of tight firewall setups, although typically a firewall will allow access to the range of IP addresses that the DHCP server can allocate dynamically).

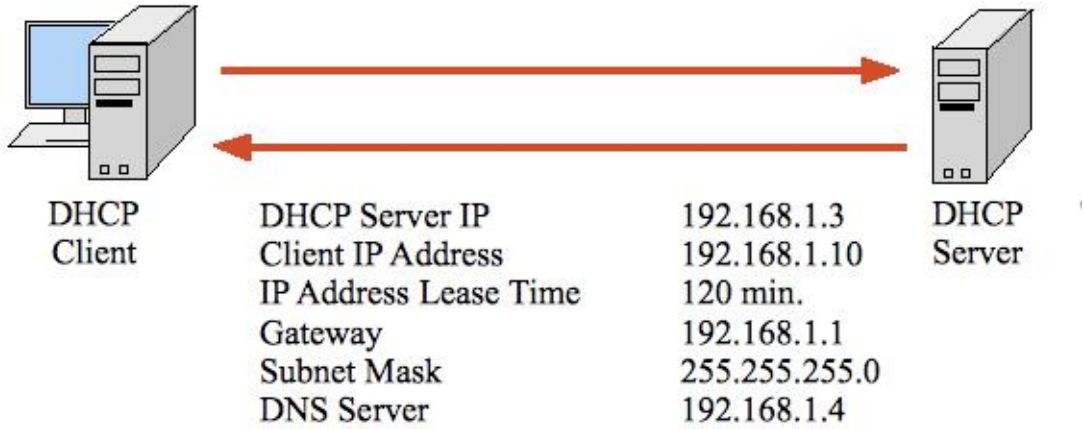


I1223: Example DHCP Interaction

From a DHCP perspective, there are only two kinds of entities: DHCP Clients (network devices or hosts) and DHCP Servers.

DHCP Clients

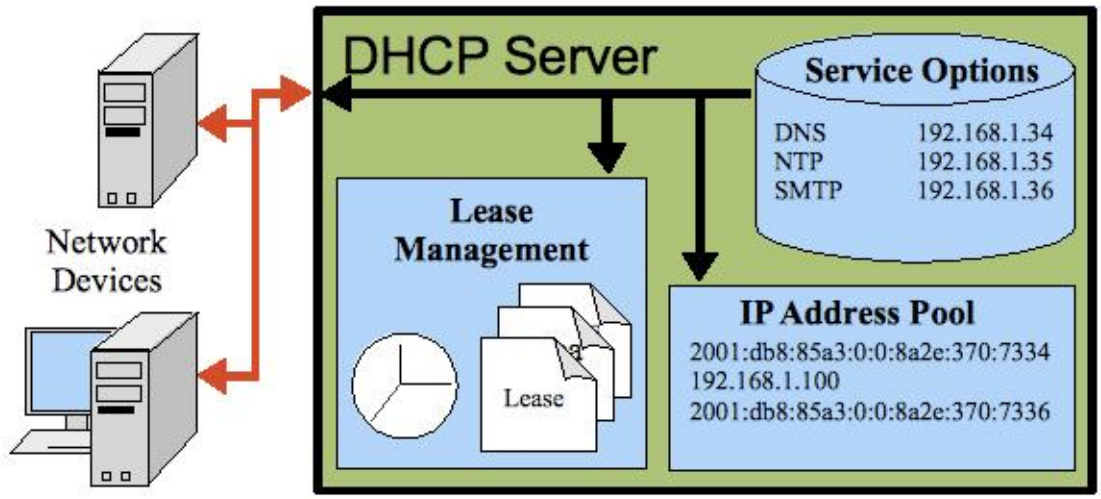
DHCP clients, sometimes referred to as network devices or hosts, use the network to contact the DHCP Servers to obtain an IP address and the configuration parameters required to use that connection. Once configured, the DHCP client then obtains the IP addresses of the network services (i.e., **Domain Name System** [**DNS**] server, **Network Time Protocol** [**NTP**] server, etc.) required to accomplish necessary tasks. All IP addresses a DHCP server provides are only leased to the DHCP client; the client needs to be able to recover when the DHCP server revokes the IP addresses the server allocated to the client.



I1224: Example DHCP Interaction

DHCP Servers

DHCP servers dynamically allocate IP addresses to DHCP clients dynamically and manage the leases of those addresses. In addition, the DHCP server can provide the DHCP client with the IP addresses of the various network services available on the network the DHCP Server manages. When leases expire, the DHCP Server attempts to reallocate the previous address to the same client. If the client is registered in the Domain Name System, DHCP will register any new addresses back to the DNS Server.



I1225: DHCP Server

Guidance

- G1598: Allow dynamic **Domain Name System** (**DNS**) updates to the Node's internal DNS service by local **Dynamic Host Configuration Protocol** (**DHCP**) **server(s)**.

- G1601: Use configurable **routers** to provide dynamic **Internet Protocol** (**IP**) address management using the **Dynamic Host Configuration Protocol** (**DHCP**).
- G1610: Configure the **Dynamic Host Configuration Protocol** (**DHCP**) services to assign **multicast** addresses.

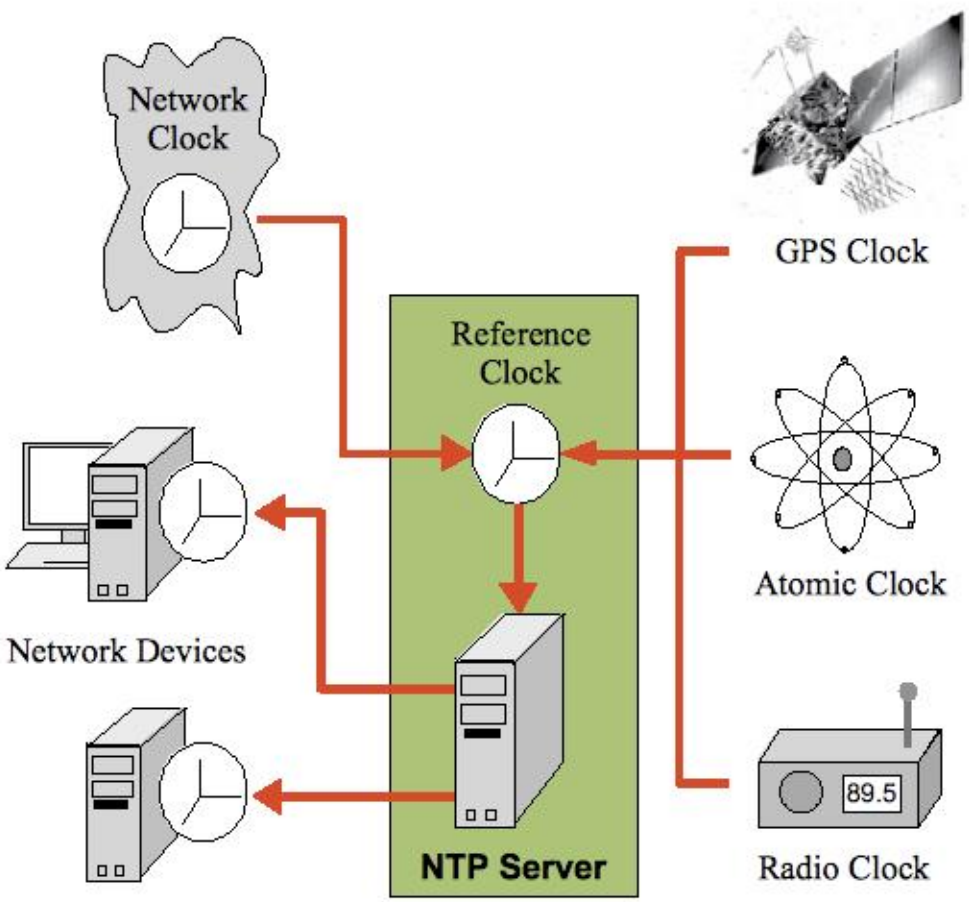
P1144: Network Time Service

Net-centric operations and security depend on date and time synchronization. Many **protocols** rely upon synchronized time to function properly, particularly security protocols. Mission **Component** logic and the usefulness of data can also suffer if there is not a common understanding and synchronization of time across the **enterprise**.

The most important and widely-used protocol for distributing and synchronizing time is the **Network Time Protocol** (**NTP**), though other less-popular or outdated time protocols remain in use.

To enable time synchronization, an NTP server reads the actual time from a reference clock and distributes this information to its clients using a computer network. The time server may be a local network time server or an internet time server. The time reference for a time server could be another time server on the network or the Internet, a connected radio clock or an atomic clock. The most common true time source is a **Global Positioning System** (**GPS**) or GPS master clock. Time servers are sometimes multi-purpose network servers, dedicated network servers, or dedicated devices. All a dedicated time server does is provide accurate time.

As an example, the U.S. Naval Observatory [http://www.usno.navy.mil] provides ***Stratum 1*** or top-level time service to Continental U.S. (CONUS) Nodes from servers at tick.usno.navy.mil and tock.usno.navy.mil. Stratum 1 time servers act as "wholesale" sources and supply time synchronization data to more local Stratum 2 "retail" time servers, which in turn provide time services to individual local systems.



I1222: Network Time Service

Guidance

- G1604: Use configurable **routers** to provide time synchronization services using **Network Time Protocol** (**NTP**).
- G1608: Obtain reference time from a standard globally synchronized time source.

- G1609: Arrange for a backup time source.

P1355: Application Layer Protocols

Internet Protocol (**IP**) networking originally developed as an environment supporting reliable transfer of digital data among a community of users. The transport infrastructure does not categorize **services**, because from the transport viewpoint it does not matter; services and Internet Engineering Task Force (IETF) "STD 66" (RFC 3986, *Uniform Resource Identifier (URI): General Syntax*) service authorities (such as **HTTP** for the Web, **FTP** for file transfer, and **SMTP** for e-mail) are just ports and associated service protocols. However, the categorization of a number of such services uses their transport port and protocol due to transport performance (**QoS**) and security reasons as well as IETF governance of many of the standards.

The user community rapidly found uses best achieved by a special protocol or protocol set that they could share in common. Some of these application layer protocols are in the following subsection.

Widely-Employed Application Layer Protocols

The Internet Protocol suite includes many application layer protocols that represent a wide variety of applications, including the following:

- ***File Transfer Protocol (FTP)*** is a network protocol used to transfer data from one computer to another through a network such as the Internet. FTP supports exchanging and manipulating files over a TCP computer network. A FTP client may connect to an FTP server to manipulate files on that server. There are many FTP client and server programs available for different operating systems, making FTP a popular choice for exchanging files independent of the operating systems involved.
- ***Simple Network Management Protocol (SNMP)*** forms part of the Internet Protocol suite as defined by the Internet Engineering Task Force. Network management systems use SNMP to monitor network-attached devices for conditions that warrant administrative attention. SNMP consists of a set of standards for network management, including an Application Layer protocol, a database schema, and a set of data objects.
- ***Telnet*** (a contraction of ***Tel***ecommunication ***net***work) is a network protocol used on **Internet** or **local area network** (**LAN**) connections. The term telnet also refers to software which implements the client part of the protocol. Telnet clients are available for virtually all platforms. Most network equipment and operating systems with a TCP/IP stack support some kind of Telnet service server for their remote configuration.
- ***X Windows*** is a windowing system that implements the X display protocol and provides windowing on bitmap displays. It provides the standard toolkit and protocol with which to build graphical user interfaces (GUIs) on most Unix-like operating systems and OpenVMS. The X Windows system has been ported to many other contemporary general purpose operating systems.
- ***Network File System (NFS)*** is a network file system protocol which allows a user on a client computer to access files over a network as easily as if the network devices were attached to its local disks.
- ***Simple Mail Transfer Protocol (SMTP)*** is a standard for electronic mail (e-mail) transmissions across the Internet. While electronic mail server software uses SMTP to send and receive mail messages, user-level client mail applications typically only use SMTP for sending messages to a mail server for relaying. For receiving messages, client applications usually use either the Post Office Protocol (POP) or the Internet Message Access Protocol (IMAP) to access their mail box accounts on a mail server.
- ***Hypertext Transfer Protocol (HTTP)*** is an application-level protocol for distributed, collaborative, hypermedia information systems. HTTP is a generic, stateless, protocol which can be used for many tasks beyond its use for hypertext, such as name servers and distributed object management systems, through extension of its request methods, error codes and headers.
- ***Secure Shell (SSH)*** is a network protocol that allows data exchange using a secure channel between two networked devices. SSH was designed as a replacement for TELNET and other insecure remote shells which sent information, notably passwords, in plaintext, leaving them open to interception.
- ***Session Initiation Protocol (SIP)*** is a signalling protocol, widely used for setting up and tearing down multimedia communication sessions such as voice and video calls over the Internet. Other feasible application examples include video conferencing, streaming multimedia distribution, instant messaging, presence information and online games. The protocol can be used for creating, modifying and terminating two-party

(unicast) or multiparty (**multicast**) sessions consisting of one or several media streams. The modification can involve changing addresses or ports, inviting more participants, adding or deleting media streams, etc.

P1141: Mobility

There have been significant advances in **Transmission Control Protocol/Internet Protocol** (**TCP/IP**) connectivity to mobile **Nodes**, such as airplanes, ships, and battlefield units; however, some significant challenges remain. In particular, it is unclear to what extent mobile Nodes can utilize **Enterprise Services**, particularly the DISA **Core Enterprise Services** (**CES**), directly. The characteristics of the link are likely to be extremely variable, including high frequency of topology changes, intermittent connectivity, higher than typical packet loss, low bandwidth, or high latency. Such characteristics are generally problematic for anything but the simplest of enterprise services. Components that use these services need to adapt in real-time to the presence or absence of the service and to the potentially intermittent performance of enterprise services. Consequently, these components must be able to handle the failover and recover from enterprise service errors and gaps.

Managers of mobile Nodes that rely on the **Internet Protocol** (**IP**) for inter-Node communication should engage with the DISA **Net-Centric Enterprise Services** (**NCES**) Program Office [R1259] to explore approaches for mobile use of the CES services. Alternatives might include development of specialized **Software Developers Kits** (**SDKs**) that implement the required adaptive behavior or use of service **proxies** within the Node that could failover gracefully.

Many of the transport elements listed above may require extensions to account for the Node's intended mobile environment. For example, today's commercial routing protocols are not intended for the extent of dynamic and mobile behavior encountered in tactical military environments.

Another example is that **TCP** performance over satellite links is generally poor due to delays and blockages inherent to satellite links. Consider TCP extensions and other transport protocols developed to mitigate this risk for high bandwidth, high latency satellite communications.

Mobile IP is a standard that allows users with mobile devices whose IP addresses are associated with one network to stay connected when moving to a network with a different IP address. When a user leaves the network with which his device is associated (home network) and enters the domain of a foreign network, the foreign network uses the Mobile IP protocol to inform the home network of a care-of address to which to send all packets for the user's device.

Nodes can be mobile or deployable as well as fixed. Mobile networks, by their very nature, are untethered and usually reliant upon radio frequency (RF) transmissions. An inherent challenge to address is that of ensuring uninterrupted **Global Information Grid** (**GIG**) interoperability as the underlying network changes dynamically.

> ***Note:*** *A goal of mobile or deployable Nodes is that they can plug into different locations in the GIG without loss of interoperability.*

Mobile IPv4

A mobile node can have two addresses:

- a permanent home address
- a care-of address associated with the network the mobile node is visiting

There are two kinds of entities in Mobile IP:

- a home agent stores information about mobile nodes whose permanent address is in the home agent's network
- a foreign agent stores information about mobile nodes visiting its network; foreign agents also advertise care-of addresses which Mobile IP uses

A node wanting to communicate with the mobile node sends packets to the home address of the mobile node. The home agent intercepts these packets and, using a table, tunnels the packets to the mobile node's care-of address with a new IP header while preserving the original IP header. Decapsulation at the end of the tunnel removes the added IP header from the packets prior to delivery to the mobile node.

When acting as a sender, a mobile node simply sends packets directly to the other communicating node through the foreign agent.

Mobile IPv6

A key benefit of Mobile IPv6 as opposed to Mobile IPv4 is that even though the mobile node changes locations and addresses, the existing connections through which the mobile node is communicating are maintained. To accomplish this, connections to mobile nodes are with a specific address always assigned to the mobile node and through which the mobile node is always reachable. Mobile IPv6 provides Transport layer connection survivability when a node moves from one link to another.

Best Practices

- BP1594: Examine the use of **Transmission Control Protocol** (**TCP**) extensions and other transport protocols that have been designed to mitigate risk for high bandwidth, high latency satellite communications.

P1356: Traffic Management

Network traffic management uses the principles of Traffic Engineering and **Quality of Service** (**QoS**) to optimize the network by dynamically analyzing, predicting and regulating the behavior of the network in transmitting data. Although traffic engineering originated in the telecommunications industry, the principles have been applied successfully to all kinds of communications networks including **local area networks** (**LANs**), wide area networks (WANs), cellular telephone networks and the **Internet**.

A major objective of traffic management is to optimize network performance to meet a wide variety of mission objectives. To accomplish this, traffic management must maximize the timely transport of traffic while simultaneously minimizing traffic loss, traffic exposure to compromise (particularly denial of service attacks) and operations/maintenance costs

Striking this balance between effective, secure and efficient Transport requires engineering embedded sensor and control points and engineering enterprise operations support systems that integrate network situation information and coordinate performance management operations

Good traffic management applied to network infrastructure enhances performance metrics, such as bandwidth, delay and interference, by defining administrative policies in accordance with commanders' intentions that govern traffic admission, aggregation, response to congestion, error handling, etc. Poor choices in such policies result in traffic delay, loss, and interference; however, good choices result in timely, responsive, robust information flows.

A way to avoid congestion, for example, is matching capacity to usage or usage to capacity. The matching process may occur either before access, as part of planning, or during usage spikes/troughs as an adaptive mechanism. Planning allows network service consumers to request a baseline service contract with the service provider. Specify the service consumer's requirements for bandwidth and other performance metrics as part of a **Service Level Agreement** (**SLA**). The network service determines if there is enough bandwidth available to fulfill the request. If there is enough capacity, the bandwidth is allocated to the consumer. If there is not enough capacity, the service consumer is rejected or capacity is added to the network.

In an ideal world, with proper network planning, networks should never be congested or suffer interference. However, the reality is that networks do have congestion either from fulfilling unplanned network service requests (i.e., load) or as a result of a degraded network. Congestion is only one performance tradeoff failure; another involves interference and noise which interact with congestion. Interference causes congestion due to error correction and retransmission, and congestion causes interference due to interactions inside of shared resources. The network traffic can respond to these conditions through various traffic engineering principles such as restricting or buffering network capacity.

Quality of service is a defined level of performance that adapts to the environment in which it is operating. The user of the information may be request the required QoS. The level of QoS provided is based on the request, the available capabilities of the provider, and the priority of the user.

Class of Service (CoS) is a queuing discipline. The CoS algorithm compares fields of packets or CoS tags to classify packets in different priority queues by grouping similar types of traffic and treating each type as a class with its own level of service. Class of service is simpler to manage that quality of service. Class of service is often more coarse-grained in traffic control where quality of service is more fine-grained.

The two taken together are a means for the user to specify the level of performance that he desires and the network engineer to attempt to provide that service. QoS is derived from a capability in Asynchronous Transfer Mode (ATM) where bandwidth is allocated and QoS can be guaranteed. QoS in IP networks is not guaranteed. It is an attempt by the IP network to provide service similar to ATM service.

Detailed Perspectives

The following perspectives provide more detailed information.

- Planning Network Services [P1357]
- Architectural Approaches to Traffic Management [P1358]
- Traffic Engineering [P1359]

P1357: Planning Network Services

Network planning is essential for meeting a desired network level of service. Planning can be static, off-line well in advance of the actual usage, or it can be dynamic in response to service consumer's requests. The network service balances the consumer's resource request against the available network resources and, if possible, reserves the network resources for the consumer.

To accomplish the planning and administration of the network, traffic engineering abstracts the network as a service governed by a service contract. As with most contracts, there are two independent types of parties (with at least one of each type) involved: service provider and service consumer. **Service Level Agreement** (**SLA**) parameters define the terms and conditions of a network service. The SLA parameters capture the levels of availability, serviceability, performance, operation or other service attributes as reflected in performance metrics. The SLA parameters are expressed as one or more Service Level Objectives (SLOs) which must be measurable, repeatable, attainable, controllable within measured bounds, and mutually acceptable.

Network **Quality of Service** (**QoS**) provides an assessment of "excellence" of the network service. The assessment is for each of the SLA parameters. Each SLA parameter assessment represents an aggregate of the compliance measures for the individual SLOs.

SLA Parameter	Explanation	SLO Example
Availability	Constraints on when the service can be used by the provider or when it is needed by the consumer	Network shall be available 99.9% of the time in delivering traffic to and from IP endpoints
Accessibility	Enablers or barriers to use of a service as specified by the provider or for facilities for overcoming the barrier by the consumer	Network shall support IPv4 and IPv6 traffic
Performance	Sustainable rate of providing the service or the demand for capacity from the consumer	Network latency shall be 40 milliseconds or less between IP endpoints
Compliance	Assurance of the quality of the product provided by the producer or required by the consumer	Network shall comply with IPv6
Security	Risk to the provider in servicing consumer or to the consumer in using the provider's service	Network shall support a minimum of a 1024-bit cryptographic keys
Efficiency	Cost of servicing a consumers request or using the producer's product	Networks shall support a network packet sizes from 512 to 16,384 bytes
Reliability	Assurance consistency of the product by the producer or the expectation of consistency of the product by the consumer	Network IP Packet loss shall not exceed 0.1% based on the arithmetic mean of the aggregate monthly measurement between IP endpoints
Provenance	Assurance of the origin and history of the product by the producer or the expectation of the origin and history of the product by the consumer	Network traffic shall only be on wired networks

P1358: Architectural Approaches to Traffic Management

The following standards-based **Quality of Service** (**QoS**) approaches to Traffic Management are two examples of those used both on commercial enterprise **intranets** and in the DoD. The Differentiated Services (DiffServ) architecture enables course-grain deconfliction and priority labeling of traffic in accordance with a business model or commander's intent. The Integrated Services (IntServ) architecture enables fine-grain traffic deconfliction and prioritization, but the extra control comes at a price: higher operational costs, greater network operational complexity, and overall network brittleness.

Differentiated Services

DiffServ is a networking architecture that specifies a simple, scalable, coarse-grained mechanism for classifying network traffic, managing network traffic, and providing Quality of Service (QoS) guarantees on modern IP networks. As such, it allows senior commanders to prioritize traffic over shared infrastructure according to technology and mission needs by separating it into classes and trading-off resource allocation according to class. DiffServ can, for example, provide low-latency, guaranteed service (GS) to critical network traffic such as voice or video while providing simple best-effort traffic guarantees to non-critical services such as Web traffic or file transfers. DiffServ exhibits good scaling properties. However, in the absence of additional conditioning mechanisms, DiffServ provides only preferential, differentiated levels of service and not guarantees.

Traffic flows into a DiffServ policy domain through its ingress boundary router, which then classifies and marks it with the appropriate DiffServ Code Point (DSCP) marking. From that ingress router on, the traffic is routed along its path through internal routers, which condition the traffic stream in accordance with the policies specified by the Traffic Conditioning Agreement (TCA) associated with that DSCP marking. All traffic leaving a Diffserv domain does so through an egress boundary router, which acts as the limit of the policy and the commander's span of control. For end to end traffic policy compliance, the ultimate client endpoint router should also be the egress router.

The following **Internet Engineering Task Force** (**IETF**) Requests for Comments (RFCs) provide additional information:

- RFC 2474, Standards Track, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*, December 1998
- RFC 2475, Informational, *An Architecture for Differentiated Service*, December 1998
- RFC 4124, Proposed Standard, Protocol *Extensions for Support of Diffserv-aware MPLS Traffic Engineering*, Jun 2005.
- RFC 4125, Experimental, *Maximum Allocation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*, Jun 2005.
- RFC 4594, Informational, *Configuration Guidelines for DiffServ Service Classes*, Aug 2006.
- RFC 3270, Proposed Standard, *Multi-Protocol Label Switching (MPLS) Support of Differentiated Services*, May 2002.

Integrated Services

IntServ is an architecture that specifies the elements to guarantee quality of service (QoS) on networks. IntServ can, for example, allow video and sound to reach the receiver without interruption. IntServ specifies a fine-grained QoS system, which is often contrasted with a DiffServ coarse-grained control system. The idea of IntServ is that every router in the system implements IntServ, and every application that requires some kind of guarantees has to make an individual reservation. "Flow Specs" describe what the reservation is for, while "RSVP" (in this usage, Resource ReSerVation Protocol) is the underlying mechanism to signal it across the network.

IntServ is based on a network traffic engineering model that primarily serves the real-time flow of **IP** packets along a network path of IP nodes between two endpoints (i.e., end-to-end). IntServ accomplishes this by reserving a portion of the network bandwidth to the flow of IP packets along the designated network path. The packets flowing within the reserved bandwidth behave deterministically along the path. Packets that are not apportioned to a dedicated portion of the bandwidth remain highly non-deterministic. In other words, the packets under the control of IntServe flow under a reserved apportionment of the bandwidth. The IETF first proposed the IntServ model in

1993 as RFC 1663 primarily to support real-time teleconferencing, remote seminars, telescience and distributed simulation services.

In an IntServe architecture, a data flow starts with a request from a potential consumer (i.e., requestor) of a data stream (i.e., broadcast). How the consumer discovers the source of the broadcast is outside the scope IntServe. The consumer makes a reservation request to its router. The router then passes the request up stream to all the routers in the path to the broadcaster. If there are multiple consumers of the broadcast, the reservations are merged as they move upstream to help reduce network traffic. As the router can service the reservation, the broadcast starts to flow from the broadcaster to the consumer. If a router is already servicing a broadcast request at or above the requested data rate from another consumer, the reservation request does not need to go up stream any further and the broadcast can start flowing to the consumer from that router.

Note: *Broadcasts can be separated into various layers, with each layer representing a particular quality range. For example, a 20Kbps low quality audio layer may be encoded separately from the high quality enhancement of the audio. Additionally, the video aspect of the broadcast can be encoded into yet more layers.*

Hosts on the **Internet** use the Resource Reservation Protocol to request a QoS level on the network on behalf of an application data flow. Routers use RSVP to deliver QoS requests to other routers along the path(s) of the data flow. The impacts of using RSVP over the black core must be understood and accounted for as more information about the black core becomes available.

The following IETF RFCs provide additional information:

- RFC 2205, Proposed Standard, *Resource ReSerVation Protocol RSVP -- Version 1 Functional Specification*, September 1997.
- RFC 2207, Proposed Standard, *RSVP Extensions for IPSEC Data Flows*, September 1997.
- RFC 2998, Informational. *A Framework for Integrated Services Operation over Diffserv Networks*, Nov. 2000.
- RFC 1633, Informational, *Integrated Services in the Internet Architecture: an Overview*, Jun 1994,

QoS-Based Routing

QoS-based routing is a mechanism under which paths for flows are determined based on some knowledge of resource availability in the network as well as the QoS requirement of flows. These protocols search for routes with sufficient resources for the QoS requirements. QoS-based routing also has potential to address tactical edge environments; however, the overhead of QoS routing protocols is very high for bandwidth-limited mobile ad hoc networks (MANETs).

The following IETF RFCs provide additional information:

- RFC 2386, Informational A Framework for QoS-based Routing in the Internet, Aug 1998.
- RFC 2676, Experimental QoS Routing Mechanisms and OSPF Extensions, Aug. 1999.
- RFC 3583, Informational Requirements of a Quality of Service (QoS) Solution for Mobile IP, Sep 2003.

P1359: Traffic Engineering

Traffic engineering is a method of optimizing the performance of a network by dynamically analyzing, predicting and regulating the behavior of data transmitted over that network. Traffic engineering uses statistical techniques such as queuing theory to predict and engineer the behavior of telecommunications networks such as telephone networks or the **Internet**. The crucial observation in traffic engineering is that in large systems the law of large numbers can help make the aggregate properties of a system over a long period of time much more predictable than the behavior of individual parts of the system. The queueing theory originally developed for circuit-switched networks is applicable to packet-switched networks.

Traffic Classification

Packet classifiers select **Internet Protocol** (**IP**) packets in a traffic stream based upon the content of some portion of the packet header. In essence, classifiers "steer" packets matching some specified rule to an element of a traffic conditioner for further processing. Classifiers must be configured by some management procedure in accordance with the appropriate Traffic Conditioning Agreement (TCA).

In the Differentiated Services (DiffServ) architecture, two basic types of classifiers exist. The first is a multifield (MF) classifier, which examines multiple fields in the IP datagram header to determine the service class to which a packet belongs. The second is a behavior aggregate (BA) classifier, which examines a single field in an IP datagram header and assigns the packet to a service class based on what it finds.

Behavior Aggregate (BA) Classifier

The BA classifier classifies IP packets based solely on the Differntiated Services Code Point (DSCP). Specific DSCP values are used as the selector for per-hop behavior (PHB).

Multi-Field (MF) Classifier

The MF classifier is used when the BA classifier is insufficient to classify a packet. The MF classifier selects IP packets based on the value of a combination of one or more IP header fields (i.e., source address, destination address, Differntiated Services field, protocol ID, source port, destination port numbers, and DSCP).

> ***Note:*** *Sometimes the packets are fragmented from each other upstream in the packet stream. When an MF classifier uses the contents of transport-layer header fields, it may not consistently classify subsequent packet fragments. A possible solution is to maintain a fragmentation state; however, this is not a general solution due to the possibility of upstream fragment re-ordering or divergent routing paths.*

Traffic Conditioning

Traffic conditioning can involve the metering, shaping, policing and/or re-marking of packets to ensure that traffic conforms to the rules specified in the Traffic Conditioning Agreement and in accordance with the domain's service provisioning policy. The extent of traffic conditioning required is dependent on the specifics of the service offering. Conditioning might be simple DSCP re-marking or very complex policing and shaping operations.

Classifiers select a traffic stream and then direct packets to a logical instance of a traffic conditioner. A meter might measure the traffic stream against a traffic profile. The state of the meter with respect to a particular packet (e.g., whether it is in-profile or out-of-profile) may be part of the traffic marking, dropping, or shaping actions.

> ***Note:*** *A traffic conditioner may not necessarily contain all four conditioning operations (metering, shaping, policing, re-marking). For example, if there is no traffic profile in effect, packets may only be subject to the classifier and marker operations.*

Representative traffic engineering building blocks follow.

Bandwidth Management

Bandwidth management is the process of measuring and controlling the communications (traffic, packets) on a network link to avoid filling the link to capacity or overfilling the link, which would result in network congestion and poor performance. More sophisticated bandwidth management techniques use a macro approach that manages traffic on a per user rather than a per application basis. This frees the network provider from having constantly to identify what clients/customers are doing and avoids some of the legal concerns and public outcry about providers dictating what customers can do. This approach acknowledges that on Internet Service Provider (ISP) type networks, "fairness" is a per client issue. By managing per client, no single user can use more bandwidth than the user's allocation, no matter what application the user may be running or how many users are on the user's endpoint.

Admission Control

Admission control is a mechanism that estimates the level of QoS that a new user session will need and whether sufficient bandwidth is available. If bandwidth is available, the session is admitted. Admission control is a network **Quality of Service** (**QoS**) procedure. Admission control determines how bandwidth and latency are allocated to streams with various requirements. An application that wishes to use the network to transport traffic with QoS must first request a connection, which involves informing the network about the characteristics of the traffic and the QoS the application requires. This information is stored in a traffic contract. The network judges whether it has enough resources available to accept the connection and then either accepts or rejects the connection request. Admission control is useful in situations where a certain number of connections (phone conversations, for example) may all share a link, while an even greater number of connections causes significant degradation in all connections to the point of making them all useless such as in congestive collapse.

Prioritization

Prioritization is a mechanism to give important network traffic precedence over unimportant network traffic. Prioritization is also called class of service (CoS) since traffic is classed into categories such as high, medium, and low (or gold, silver, and bronze, etc.), and the lower the priority, the more "drop eligible" is a packet.

Rate Limiting

Rate limiting is the process of restricting a classified packet flow or a source interface to a rate that is less than the physical rate of the port. Rate limiting enforces data rates below the physical line rate of a port for an IP interface, a classified packet flow, or a Layer 2 interface. It allows limiting the total bandwidth one class of traffic uses and making it available for other classes. Some implementations allow hierarchies of rate limits with preferential access among them.

Delay Management

Delay Management is a capability to control traffic in order to optimize or guarantee performance, low latency, and/or bandwidth by delaying packets. Delay and latency are similar terms that refer to the amount of time it takes to transmit a bit from source to destination. One way to view latency is how long a system holds on to a packet. That system may be a single device like a router, or a complete communication system including routers and links (derived from the Linctionary.com Delay, Latency, and Jitter entry, http://www.linktionary.com/d/delay.html). Traffic shapers delay some or all of the packets in a traffic stream in order to bring the stream into compliance with a traffic profile.

IP QoS manages delay of packets through a router. However, in wireless environments, such as an airborne network, the transmission time over a line-of-sight link is likely to dominate delays. In such cases, delay management through the router will be important mostly for queuing outgoing packets on the radio link.

Drop Management

Drop management is a capability to alleviate congestion by dropping packets when necessary or appropriate. Drop management includes mechanisms such as admission control (drop all traffic before queuing), pre-emption (drop all traffic henceforth), active queue management (for example Random Early Detection (RED), and Weighted RED which drops selected traffic packets. Refer to the Internet Engineering Task Force (IETF) *Recommendations on Queue Management and Congestion Avoidance in the Internet* Request for Comment (RFC 2309).

G1300

Secure all endpoints.

Rationale:

Something is only as secure as its weakest link. Therefore, all access points in an application should be secured. An endpoint is defined as an entry or an exit point of an application. Any access point can be vulnerable to attacks. For instance, if an application file reads configuration settings from a properties file, that file can be corrupted or incorrectly configured. This can cause incorrect behavior in the application. Also if component, **module** or application provides remote access or is part of any inter-process communications, these areas are vulnerable to attacks. For instance, if the application provides an external socket interface, does it validate commands being sent by the client?

Referenced By:

NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / Naval Open Architecture / Maintainability
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Policy / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Integrity
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Integrity

Evaluation Criteria:

1) Test:

Does the application handle invalid configuration, provide appropriate defaults, and protect sensitive data?

Procedure:

Check application processing of data files (configuration files, properties files, preferences, XML, etc.).

Example:

None.

2) Test:

Does the application properly handle security when dealing with externally accessible API(s) and external ports?

Procedure:

Verify sensitive data is protected, and verify all network base protocols validate commands and values.

Example:

None.

G1301

Practice layered security.

Rationale:

An application with layered security provides more protection against attacks. Combining multiple layers of security defenses can provide additional protection when one layer is broken.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Policies and Processes for Implementing Software Security / Secure Coding and Implementation Practices / Practice Defense in Depth
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Policies and Processes for Implementing Software Security / Secure Coding and Implementation Practices / Practice Defense in Depth
NESI / Part 5: Developer Guidance / Software Security / Policies and Processes for Implementing Software Security / Secure Coding and Implementation Practices / Practice Defense in Depth
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Layering and Modularity
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Other Design Tenets
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES Definitions and Status
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES Definitions and Status
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES Definitions and Status
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES Definitions and Status
NESI / Part 2: Traceability / Naval Open Architecture / Maintainability
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Integrity
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Integrity

Evaluation Criteria:

1) Test:

Do internal and external API(s) perform security checks?

Procedure:

Make sure layers of API(s) starting from externally accessible API(s) down through the layers of internally accessible API(s) provide sufficient security checks. For example, does each layer of the API perform data validation? If internal API is calling remote services, is the data sufficiently protected from snoopers (e.g., use of secure sockets)?

Example:

None

2) Test:

Does the application handle security when processing data files?

Procedure:

Embed all application specific resources such as graphics, internal application configuration files such as internationalization properties/resources, XML files as part of a signed application deployment file (.jar, .exe, etc.).

Example:

None

G1302

Validate all inputs.

Rationale:

Do not limit input validation to the presentation tier; rather, all external APIs should validate inputs prior to use. This is just one aspect of defense in depth which can prevent many attacks including SQL Injection, Cross-Site Scripting, Buffer Overflows, and Denial of Service.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Policies and Processes for Implementing Software Security / Secure Coding and Implementation Practices / Validate Input
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Policies and Processes for Implementing Software Security / Secure Coding and Implementation Practices / Validate Input
NESI / Part 5: Developer Guidance / Software Security / Policies and Processes for Implementing Software Security / Secure Coding and Implementation Practices / Validate Input
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Integrity / Data, Application and Service Integrity
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Integrity / Data, Application and Service Integrity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Other Design Tenets
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Policy / Design Tenet: Identity Management, Authentication, and Privileges

Evaluation Criteria:

1) Test:

Does the application provide proper handling for null input?

Procedure:

Check application handling of null values.

Example:

None

2) Test:

Does the application use prefix or postfix validation (asserts) to verify input parameters?

Procedure:

Check application range validation of externally accessible API(s).

Example:

None

G1306

Authenticate the identity of application users.

Rationale:

This ensures there is some traceability and also provides the first in a multilayer security system.

> ***Note:*** *This guidance is derived from the DoD Class 3 PKI Public Key-Enabled Application Requirements Document, Version 1.0, 13 July 2000.*

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Identity Management / Public Key Infrastructure
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Identity Management / Public Key Infrastructure
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Authorization and Access Control
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Authorization and Access Control
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Policy / Design Tenet: Identity Management, Authentication, and Privileges

Evaluation Criteria:

1) Test:

Does the application authenticate with another service (**LDAP**, database or simple password)?

Procedure:

Inspect application code to ensure that the user is authenticated against an LDAP, database or simple password service.

Example:

None

2) Test:

Does the application require user certificates?

Procedure:

Ensure the application is setup to require client side certificates. This can be done easily by using a machine without any DoD client certificates installed and attempting to access the application.

Example:

None

G1317

Ensure applications store Certificates for subscribers (the owner of the Public Key contained in the Certificate) when used in the context of signed and/or encrypted email.

Rationale:

This will allow other parties to use the public key to encrypt messages sent to the application.

> ***Note:*** *This guidance is derived from the DoD Class 3 PKI Public Key-Enabled Application Requirements Document. Section (4.5), Version 1.0, July 13, 2000.*

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Cryptography
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Cryptography
NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Technologies and Standards for Implementing Software Security / Key Management
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Technologies and Standards for Implementing Software Security / Key Management
NESI / Part 5: Developer Guidance / Software Security / Technologies and Standards for Implementing Software Security / Key Management
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Policy / Design Tenet: Identity Management, Authentication, and Privileges

Evaluation Criteria:

1) Test:

Is the public key available from the Directory Server application?

Procedure:

See if it is possible to extract the public key certificate from the Directory Server application.

Example:

None

G1325

Encrypt symmetric keys when not in use.

Rationale:

Symmetric keys enable both sides of the conversation to have knowledge of the key for encryption. It can not be given out freely, which means if it is going to be stored for repeated use, it should be encrypted first before storage.

> ***Note:*** *This guidance is derived from the DoD Class 3 PKI Public Key-Enabled Application Requirements Document, Version 1.0, 13 July 2000.*

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Technologies and Standards for Implementing Software Security / Encryption Services
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Technologies and Standards for Implementing Software Security / Encryption Services
NESI / Part 5: Developer Guidance / Software Security / Technologies and Standards for Implementing Software Security / Encryption Services
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Cryptography
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Cryptography
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / Naval Open Architecture / Maintainability
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Policy / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Encryption and HAIPE

Evaluation Criteria:

1) Test:

Does the application encrypt symmetric keys when not in use?

Procedure:

Check that the application encrypts symmetric keys during storage.

Example:

None.

G1344

Encrypt sensitive data stored in configuration or resource files.

Rationale:

Sensitive data used for application configuration files (XML), user profiles, or resource files should be protected from tampering. The sensitive data should be encrypted and or a message **digest** or checksum should be calculated to check for tampering. Application should handle generation, accessing and storing data to these files.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Technologies and Standards for Implementing Software Security / Application Resource Security
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Technologies and Standards for Implementing Software Security / Application Resource Security
NESI / Part 5: Developer Guidance / Software Security / Technologies and Standards for Implementing Software Security / Application Resource Security
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Cryptography
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Cryptography
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Policy / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Encryption and HAIPE

Evaluation Criteria:

1) Test:

Is sensitive data in configuration files and user profiles?

Procedure:

Check properties files, XML configuration files or user profiles for sensitive data in the clear.

Check for an application to edit, and creation of the file.

Example:

None.

G1352

Use database clustering and redundant array of independent disks (RAID) for high availability of data.

Rationale:

Database clusters combined with RAID technology (e.g., data striping and mirroring) can help ensure continued operation of a system that suffers hardware or software failure.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Technologies and Standards for Implementing Software Security / RDBMS Security
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Technologies and Standards for Implementing Software Security / RDBMS Security
NESI / Part 5: Developer Guidance / Software Security / Technologies and Standards for Implementing Software Security / RDBMS Security
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Availability
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Scalability
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Integrity / Network Infrastructure Integrity
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Integrity / Network Infrastructure Integrity
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / Naval Open Architecture / Maintainability

Evaluation Criteria:

1) Test:

Is the system designed to support high availability?

Procedure:

Check for the existence of a cluster and/or failover capability.

Check for the existence of RAID data storage for the database.

Example:

None.

G1371

Use the National Institute of Standards and Technology (NIST) *Digital Signature Standard* promulgated in the Federal Information Processing Standards Publication 186 (FIPS Pub 186-3 as of June 2009) for creating Digital Signatures.

Rationale:

Using the FIPS Pub 186-3 ***Digital Signature Standard*** enables interoperability of **Digital Signature Algorithms**.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Cryptography
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Cryptography
NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Technologies and Standards for Implementing Software Security / XML Digital Signatures
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Technologies and Standards for Implementing Software Security / XML Digital Signatures
NESI / Part 5: Developer Guidance / Software Security / Technologies and Standards for Implementing Software Security / XML Digital Signatures
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Technologies and Standards for Implementing Software Security / SOAP Security
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Technologies and Standards for Implementing Software Security / SOAP Security
NESI / Part 5: Developer Guidance / Software Security / Technologies and Standards for Implementing Software Security / SOAP Security
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Policy / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Encryption and HAIPE

Evaluation Criteria:

1) Test:

Does the Web service user generate signatures using the FIPS 186-3 ***Digital Signature Standard***?

Procedure:

Generate a test message and check it for compliance with the FIPS 186-3 ***Digital Signature Standard***.

Example:

None

2) Test:

Does the Web service provider generate signatures using the FIPS 186-3 ***Digital Signature Standard***?

Procedure:

Generate a test message and check it for compliance with the FIPS 186-3 ***Digital Signature Standard***.

Example:

None

G1374

Individually encrypt sensitive message fragments intended for different intermediaries.

Rationale:

Individually encrypting message fragments allows targeting individual fragments at different intermediaries along the message path to the final destination.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Cryptography
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Cryptography
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Technologies and Standards for Implementing Software Security / SOAP Security
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Technologies and Standards for Implementing Software Security / SOAP Security
NESI / Part 5: Developer Guidance / Software Security / Technologies and Standards for Implementing Software Security / SOAP Security
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Policy / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Encryption and HAIPE

Evaluation Criteria:

1) Test:

Are sensitive fragments of the message encrypted?

Procedure:

Observe messages that are sent to see if the sensitive fragments of the message are encrypted.

Example:

None

G1376

Do not encrypt message fragments that are required for correct SOAP processing.

Rationale:

It is possible to encrypt the entire SOAP message, various portions of the SOAP message or the contents of the data transported within the SOAP message. Encrypting the entire SOAP message requires that any intermediate processing of the SOAP message includes decryption of the entire message.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Cryptography
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Cryptography
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Other Design Tenets
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Technologies and Standards for Implementing Software Security / SOAP Security
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Technologies and Standards for Implementing Software Security / SOAP Security
NESI / Part 5: Developer Guidance / Software Security / Technologies and Standards for Implementing Software Security / SOAP Security
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Encryption and HAIPE

Evaluation Criteria:

1) Test:

Does the Web service user encrypt the entire message?

Procedure:

Generate a test message and check it to make sure the XML tags are not encrypted.

Example:

None

2) Test:

Does the Web service provider encrypt the entire message?

Procedure:

Generate a test message and check it to make sure the XML tags are not encrypted.

Example:

None

G1378

Encrypt communication with LDAP repositories.

Rationale:

Encryption of communication to LDAP servers helps prevent disclosure of data during transmission.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Technologies and Standards for Implementing Software Security / LDAP Security
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Technologies and Standards for Implementing Software Security / LDAP Security
NESI / Part 5: Developer Guidance / Software Security / Technologies and Standards for Implementing Software Security / LDAP Security
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Cryptography
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Cryptography
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / Naval Open Architecture / Maintainability
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Policy / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Encryption and HAIPE

Evaluation Criteria:

1) Test:

Are connections to LDAP repositories encrypted?

Procedure:

Verify that connections to LDAP repository use Transport Layer Security (TLS) or Secure Sockets Layer (SSL).

Example:

G1381

Encrypt sensitive persistent data.

Rationale:

When data is persisted, there is always a chance that the security of the system that stores the data may be compromised. To minimize the risk, all sensitive data such as passwords and personal information should be encrypted when it is persisted.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Cryptography
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Cryptography
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Encryption and HAIPE
NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Policies and Processes for Implementing Software Security / Data at Rest
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Policies and Processes for Implementing Software Security / Data at Rest
NESI / Part 5: Developer Guidance / Software Security / Policies and Processes for Implementing Software Security / Data at Rest

Evaluation Criteria:

1) Test:

Is all sensitive data that is persisted encrypted?

Procedure:

Look at all data stores and check for encrypted passwords and other sensitive data..

Example:

G1569

Maintain a comprehensive list of all of the Components that are part of the Node.

Rationale:

Throughout the lifecycle of a Node (from design to instantiation), this action is fundamental to the provisioning of a shared infrastructure and the avoidance of functional duplication within the Node. This activity has a direct impact on the design and implementation requirements during acquisition.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Enterprise Service Management
NESI / Part 4: Node Guidance / General Responsibilities / Nodes as Stakeholders
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Service-Oriented Architecture (SOA)
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability
NESI / Part 2: Traceability / Naval Open Architecture / Reusability

Evaluation Criteria:

1) Test:

Is there a list of Components that comprise the Node?

Procedure:

Examine the documents (for example, the Node's design requirements) and look for a list of Components.

Example:

None.

G1570

Assume an active management role among the Components within the Node.

Rationale:

Involvement of the Node as a stakeholder in its Components (from design to instantiation) has a bearing on **Global Information Grid** (**GIG**) interoperability. Strong coordination among a Node's Components will likely avoid the external exposure of inconsistencies or, worse, incomplete, inaccurate, or misunderstood data.

Referenced By:

NESI / Part 4: Node Guidance / General Responsibilities / Nodes as Stakeholders
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability

Evaluation Criteria:

1) Test:

Do the Components of the Node set forth requirements in their [appropriate acquisition document] for coordinating with the Node.

Procedure:

Check the [appropriate acquisition document] of the Components and determine if the Node is listed as a stakeholder or if there are requirements for coordinating with the Node.

Example:

A Component's **Capability Development Document** (**CDD**) may state a requirement for participating in a Node which could satisfy this requirement.

2) Test:

Do the Components of the Node list the Node as a primary stakeholder in their [appropriate acquisition document]?

Procedure:

Check the [appropriate acquisition document] of the Components and determine if the Node is listed as a stakeholder or if there are requirements for coordinating with the Node.

Example:

A Component's **Capability Development Document** (**CDD**) may state a requirement for participating in a Node which could satisfy this requirement.

G1571

Maintain a comprehensive list of all the Communities of Interest (COIs) to which the Components of a Node belong.

Rationale:

The Node infrastructure must be engineered to support the information exchange between **Communities of Interests** (**COIs**). If a comprehensive list of COIs is not created and maintained then the infrastructure may no longer be adequate and may continue to make provisions for COIs that are no longer a part of the Node.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Net-Centric Information Engineering
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Net-Centric Information Engineering
NESI / Part 4: Node Guidance / General Responsibilities / Net-Centric Information Engineering
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Data / Design Tenet: Be Responsive to User Needs

Evaluation Criteria:

1) Test:

Do the Node's Components have representation registered within the DoD Metadata Registry as members of the Communities of Interest (COIs)?

Procedure:

Examine the DoD Metadata Registry for members of the Node organization that are members of the pertinent COIs.

Example:

None.

G1572

Include the Node as a party to any Service Level Agreements (SLAs) signed by any of the components of the Node.

Rationale:

The Node has a stake in performance specifications provided in the **Service Level Agreements** (**SLA**). Since the SLA is a contract that commits the application service provider to a required level of service. The Node must be able to support that level of service with its infrastructure.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Availability
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Net-Centric Information Engineering
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Net-Centric Information Engineering
NESI / Part 4: Node Guidance / General Responsibilities / Net-Centric Information Engineering
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Scalability

Evaluation Criteria:

1) Test:

Does the Node have copies of all Service Level Agreements (SLAs) signed by its Components?

Procedure:

Compare the Service Level Agreements (SLAs) against the service Components supported by the Node.

Example:

None.

G1573

Define the enterprise design patterns that a Node supports.

Rationale:

The Node infrastructure must be engineered to support information exchanges between various **Communities of Interest** (**COIs**). The COIs can require any number of **Components** to fulfill the COIs mission, When a Component wishes to make its data available over the **enterprise**, there are different enterprise design pattern which can be used. For example, the mechanism selected by a Component to exchange information may be publish-subscribe, broker, or client server. The Node infrastructure must support whichever enterprise design pattern mechanism is selected.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Net-Centric Information Engineering
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Net-Centric Information Engineering
NESI / Part 4: Node Guidance / General Responsibilities / Net-Centric Information Engineering
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Open Architecture
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Service-Oriented Architecture (SOA)

Evaluation Criteria:

1) Test:

Does the Node document which types of enterprise design patterns it supports?

Procedure:

Look through the Node documents for a list of enterprise design patterns it supports.

Example:

None.

G1574

Define which enterprise design patterns a Component requires.

Rationale:

A Component should document which enterprise design patterns it intends to capitalize on to meet its mission. For example, a client interested in using a client-server weather service, could have problems if the weather service is a real-time publish-subscribe service. This action clarifies for the Node which enterprise design patterns are required by its Components and provides direction for which patterns to support at the Node level.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Net-Centric Information Engineering
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Net-Centric Information Engineering
NESI / Part 4: Node Guidance / General Responsibilities / Net-Centric Information Engineering
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Open Architecture
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Service-Oriented Architecture (SOA)

Evaluation Criteria:

1) Test:

Does the Component indicate which type of enterprise design pattern it will use?

Procedure:

Look through the Component documentation and that defines what type of enterprise design pattern it uses.

Example:

None.

G1575

Designate Node representatives to relevant Communities of Interest (COIs) in which Components of the Node participate.

Rationale:

COI is the inclusive term used to describe collaborative groups of users who must exchange information in pursuit of their shared goals, interests, missions, or business processes and who therefore must have shared vocabulary for the information they exchange. The principal mechanism for recording COI agreements is the **DoD Metadata Registry** required by the DoD CIO Memorandum *DoD Net-Centric Data Management Strategy: Metadata Registration*. There are registry implementations on the **Unclassified but Sensitive Internet Protocol Router Network** (**NIPRNet**), **Secret Internet Protocol Router Network** (**SIPRNet**), and **Joint Worldwide Intelligence Communications System** (**JWICS**).

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Net-Centric Information Engineering
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Net-Centric Information Engineering
NESI / Part 4: Node Guidance / General Responsibilities / Net-Centric Information Engineering
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Data / Design Tenet: Be Responsive to User Needs

Evaluation Criteria:

1) Test:

Does the Node have representation registered within the Metadata Registry as members of the **Communities of Interest** (**COIs**)?

Procedure:

Examine the **DoD Metadata Registry** for members of the Node organization that are members of the pertinent COIs.

Example:

None.

G1576

Provide an environment to support the development, build, integration, and test of net-centric capabilities.

Rationale:

Nodes should provide an environment to support the development, integration, and testing of net-centric capabilities of its **Components**. As Nodes themselves and the Components within the Nodes move closer to the implementation of net-centric capabilities, it becomes increasingly important to provide a development, integration, and test environment to support those capabilities. This environment should allow for the exercise not just the Node infrastructure, but also either host locally within the Node, or provide access to, **Net-Centric Enterprise Services** (**NCES**) piloted services. The particulars on how this is done depend on the characteristics of the Node. For example, mobile or deployed Nodes would provide environments substantially different than fixed land-based or permanent Nodes.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Joint Net-Centric Capabilities
NESI / Part 4: Node Guidance / General Responsibilities / Internal Component Environment
NESI / Part 2: Traceability / Naval Open Architecture / Maintainability

Evaluation Criteria:

1) Test:

Are there instructions on how to develop, build, integrate or test Components within the Node?

Procedure:

Look for user guides or installation instructions that cover the Node environment.

Example:

None.

G1577

Maintain an Enterprise Service schedule for interim and final enterprise capabilities within the Node.

Rationale:

The current state of **Enterprise Services** is in flux. Developing **Components** that rely on those services can create a circular problem for development. An enterprise service schedule for interim and final capabilities will help elevate the co-dependencies of the Component lifecycle from the Node lifecycle.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / General Responsibilities / Coordination of Node and Enterprise Services
NESI / Part 4: Node Guidance / General Responsibilities / Internal Component Environment
NESI / Part 2: Traceability / Naval Open Architecture / Maintainability

Evaluation Criteria:

1) Test:

Is there an enterprise service schedule or roadmap that covers interim and final capabilities of the Node?

Procedure:

Look for the existence of the schedule or a roadmap for the Node.

Example:

None.

G1578

Define a schedule for Components that includes the use of the Enterprise Services defined within the Node's enterprise service schedule.

Rationale:

The exercise of matching those **Enterprise Services** required by the **Component** to those provided by the Node can help identify and gaps in the Node's functionality. By tying the Component's enterprise services to the Node's **enterprise** schedule, critical paths may be identified in the Node's schedule.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / General Responsibilities / Coordination of Node and Enterprise Services
NESI / Part 4: Node Guidance / General Responsibilities / Internal Component Environment
NESI / Part 2: Traceability / Naval Open Architecture / Maintainability

Evaluation Criteria:

1) Test:

Does the Component have an enterprise service schedule or roadmap that shows the progression of enterprise service usage by interim and final capabilities of the Component?

Procedure:

Look for the existence of the schedule or a roadmap for the Component.

Example:

None.

G1579

Define which Enterprise Services the Node will host locally when the Node becomes operational.

Rationale:

Locally defined **Enterprise Services** are inherently faster and less susceptible to network failures and traffic than local services. If a **Component** requires performance based or critical enterprise services that the Node will only provide as a **proxy**, then development, building, integration and testing should be done to the local enterprise service specification. If the Node developed enterprise service will not be ready until near the end of the Component's schedule, take steps to minimize risk.

Referenced By:

NESI / Part 4: Node Guidance / General Responsibilities / Internal Component Environment
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Service-Oriented Architecture (SOA)

Evaluation Criteria:

1) Test:

Does the Node specification identify which Enterprise Services will be locally defined within the Node?

Procedure:

Review the Node specification for a list of Enterprise Services that will be locally defined within the Node.

Example:

None.

G1580

Define which Enterprise Services will be hosted over the Global Information Grid (GIG) when the Node becomes operational.

Rationale:

Enterprise Services that are defined using **proxies** should have interfaces that follow the standards defined by the enterprise service provider. Therefore, the access to the **server** should be fairly stable and almost static in nature with few changes. These are services that should be in the critical path of a Component's mission.

Referenced By:

NESI / Part 4: Node Guidance / General Responsibilities / Internal Component Environment
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Service-Oriented Architecture (SOA)

Evaluation Criteria:

1) Test:

Does the Node specification identify which Enterprise Services will be defined using proxies?

Procedure:

Review the Node specification for a list of Enterprise Services that will be defined using proxies.

Example:

None.

G1581

Expose legacy functionality through the use of a service.

Rationale:

Nodes might contain **legacy systems** or **applications** that are in the **Sustainment** lifecycle phase. These **components** are often referred to as ***legacy*** systems or applications. If a Node needs to expose functionality or data from the legacy component, changing the internals of such components to support net-centricity is often impractical with little return on investment. In these cases, it is often desirable to offer a reasonable interim solution by exposing the functionality through the use of well known patterns (such as a **facade design pattern**).

Referenced By:

NESI / Part 4: Node Guidance / General Responsibilities / Integration of Legacy Systems
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Open Architecture
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Service-Oriented Architecture (SOA)
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability

Evaluation Criteria:

1) Test:

Does the Node use **facade design patterns** such as the wrapper or adapter pattern to expose the functionality of legacy systems or applications?

Procedure:

Make sure that all the Components that are exposed to the internal Node Components or to the external network (with the Node as a proxy) use a facade design pattern such as wrapper or adapter.

Example:

None.

G1582

In Node Enterprise Service schedules, include version numbers of Enterprise Services interfaces being implemented.

Rationale:

Given the complexity, varied implementation timing, and leading edge nature of **Enterprise Services**, the **orchestration** of efforts is essential for the successful integration of the Node's Components. The dependencies captured by such a schedule should clearly show what capabilities will be available and when during the Node's lifecycle.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Network Connectivity
NESI / Part 4: Node Guidance / General Responsibilities / Coordination of Node and Enterprise Services
NESI / Part 2: Traceability / Naval Open Architecture / Maintainability

Evaluation Criteria:

1) Test:

Are Enterprise Services interface versions provided on the enterprise service schedule for the Node?

Procedure:

Review the Enterprise Services schedule published for the Node and make sure the schedule provides necessary details including specific version numbers, workarounds, assumptions, constraints and configuration limitations that are interwoven into the schedule.

Example:

An Enterprise Service might be releasing a new version during the lifecycle of the Node's development; which version's functionality will be available when is essential for the successful integration of the Node's Components.

2) Test:

Are Enterprise Services interface versions provided on the enterprise service schedule for the Component?

Procedure:

Review the Enterprise Services schedule published for the Component and make sure the schedule provides necessary details including specific version numbers, workarounds, assumptions, constraints and configuration limitations that are interwoven into the schedule.

Example:

An Enterprise Service might be releasing a new version during the lifecycle of the Node's development; which version's functionality will be available when is essential so the Component can utilize the appropriate available capabilities.

G1583

Provide routine Enterprise Services schedule updates to every component of a Node.

Rationale:

A fundamental justification for the existence of nodes is to ensure it provides a shared infrastructure for its components. If that infrastructure evolves independently of the components, then they may be developed at timeframes and rates of evolution that differ from the capabilities of the available shared infrastructure. In addition, components may be members of multiple Nodes, providing an additional coordination challenge. Regular updates to the components of the master schedule will assist in managing this challenge.

Referenced By:

NESI / Part 4: Node Guidance / General Responsibilities / Coordination of Internal Components
NESI / Part 2: Traceability / Naval Open Architecture / Maintainability

Evaluation Criteria:

1) Test:

Are there multiple iterations of the Enterprise Services schedule developed over time and is the most recent update timely?

Procedure:

Check for version numbering and release dates of the Enterprise Services schedule. Ensure that a reasonably recent update is available.

Example:

None.

G1584

Provide a transport infrastructure that is shared among components within the Node.

Rationale:

Transport elements provided by the Node are a means for the Node to implement **Global Information Grid** (**GIG**) **Information Assurance** (**IA**) boundary protections, bind Components together, and satisfy other enterprise requirements. As transport elements are an essential piece of the net-centric puzzle, they also play a key role in minimizing interoperability issues. A Node's provisioning of the shared transport and related guidance is a key aspect of its existence.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport
NESI / Part 4: Node Guidance / Node Transport
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Transport Goal

Evaluation Criteria:

1) Test:

Does the Node's design provide for a transport infrastructure?

Procedure:

Review the Node's infrastructure design and ensure that the Node provides the necessary transport elements for shared use by its Components.

Example:

None.

2) Test:

Are the Node's Components using the Node provisioned transport infrastructure?

Procedure:

Review the design of the Node's Components (see G1569) and ensure that they all utilize the common transport infrastructure of inter-Nodal communication.

Example:

None.

G1585

Provide a transport infrastructure for the Node that implements Global Information Grid (GIG) Information Assurance (IA) boundary protections.

Rationale:

The **Global Information Grid** (**GIG**) is intended to be the ***outside world*** for all the components within the Node. In order to protect the components within the Node from the outside world and to protect the outside world from the Node, the Node should control the **IA** Boundary.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Net-Centric IA Posture and Continuity of Operations
NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport
NESI / Part 4: Node Guidance / Node Transport
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Transport Goal

Evaluation Criteria:

1) Test:

Is there an IA device in the acquisition list?

Procedure:

Look for an IA device within the parts list for the Node.

Example:

None.

2) Test:

Is the IA device configured to meet security requirements?

Procedure:

Check the Node's IA installation guide and look for procedures that describe how to configure the IA device for the Nodes particular needs.

Example:

None.

G1586

Provide a transport infrastructure for the Node that is Internet Protocol Version 6 (IPv6) capable in accordance with the appropriate governing transition plan.

Rationale:

During the transition period in the DoD community (FY06-FY15) networks, services and applications will be in a mixed environment. All Critical **Key Performance Parameters** (**KPPs**) must be able to operate in an **Internet Protocol Version 4** (**IPv4**) only network, an **Internet Protocol Version 6** (**IPv6**) only network, and a dual-stack network.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Transport Goal
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: IPv6

Evaluation Criteria:

1) Test:

Does the Node provide a transport infrastructure that is Internet Protocol Version 6 (IPv6) capable?

Procedure:

Verify that the Node transport infrastructure supports IPv6 such that Node Components are able to complete all critical functions utilizing only IPv6 on the network (with no use of IPv6 over IPv4 tunneling).

Example:

None.

G1587

Prepare an Internet Protocol Version 6 (IPv6) transition plan for the Node.

Rationale:

The transition from **Internet Protocol Version 4** (**IPv4**) to **Internet Protocol Version 6** (**IPv6**) is non-trivial and requires a great deal of coordination and effort on the part of everyone involved. The transition plan helps to minimize the potential disastrous side effects of the transition.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: IPv6

Evaluation Criteria:

1) Test:

Is there an Internet Protocol Version 6 (IPv6) transition plan for the Node?

Procedure:

Look for an Internet Protocol Version 6 (IPv6) transition plan document.

Example:

None.

G1588

Coordinate an Internet Protocol Version 6 (IPv6) transition plan for a Node with the Components that comprise the Node.

Rationale:

The effects of the transition from **Internet Protocol Version 4** (**IPv4**) to **Internet Protocol Version 6** (**IPv6**) is isolated in the Node infrastructure but can have impacts on all the **Components** that comprise the Node. The transition Plan should cover a "window" that allows all the Components to operate in either IPv4 or IPv6 (i.e., **Dual Stack Mode**) to make the transition.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: IPv6

Evaluation Criteria:

1) Test:

Does the plan allow for a **Dual Stack** environment at least during some transition period?

Procedure:

Look for a part of the transition plan that addresses **Dual Stack** mode of operation.

Example:

None.

G1589

Address issues in the appropriate governing Internet Protocol Version 6 (IPv6) transition plan as part of the IPv6 Transition Plan for a Node.

Rationale:

DoD has mandated that each service create an **IPv6** transformation office to manage the transition to IPv6. Node transition plans must be aligned and in conformance with the appropriate governing office's plans or criteria.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: IPv6

Evaluation Criteria:

1) Test:

Does the Node's IPv6 Transition Plan have a section that addresses specific criteria established by the appropriate governing IPv6 transition office or plan?

Procedure:

Review the IPv6 plan for a section or specific criteria that address the appropriate items from the appropriate governing plan or is approved by the appropriate governing office.

Example:

The Air Force IPv6 Transition Office requires each program to develop a plan with approval by the transition office (in lieu of aligning with a central plan). To check an Air Force Node's alignment, look to see that the Node's IPv6 transition plan is approved by the appropriate authority.

G1590

Include transition of all the impacted elements of the network as part of the Internet Protocol Version 6 (IPv6) Transition Plan for a Node.

Rationale:

Internet Protocol Version 6 (**IPv6**) transition has an impact on many transport infrastructure **Components**. The Node's IPv6 Transition Plan should include transition of all impacted network elements including **DNS**, routing, security, and dynamic address assignment. The *DoD IPv6 Network Engineer's Guidebook* (Draft) and the *DoD IPv6 Application Engineer's Guidebook* (Draft) provide guidance for transition of impacted Components.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: IPv6

Evaluation Criteria:

1) Test:

Does the Internet Protocol Version 6 (IPv6) Transition Plan address the impact of the transition to IPv6 on the Domain Name Service (DNS)?

Procedure:

Review the plan and look for a section dedicated to the Domain Name Service (DNS). At a minimum, it should indicate that there is no impact.

Example:

None.

2) Test:

Does the Internet Protocol Version 6 (IPv6) Transition Plan address the impact of the transition to IPv6 on routing?

Procedure:

Review the plan and look for a section dedicated to routing. At a minimum, it should indicate that there is no impact.

Example:

None.

3) Test:

Does the Internet Protocol Version 6 (IPv6) Transition Plan address the impact of the transition to IPv6 on security?

Procedure:

Review the plan and look for a section dedicated to security. At a minimum, it should indicate that there is no impact.

Example:

None.

4) Test:

Does the Internet Protocol Version 6 (IPv6) Transition Plan address the impact of the transition to IPv6 on dynamic address assignment?

Procedure:

Review the plan and look for a section dedicated to dynamic address assignment. At a minimum, it should indicate that there is no impact.

Example:

None.

G1591

Prepare IPv6 Working Group products as part of the Internet Protocol Version 6 (IPv6) transition plan for a Node.

Rationale:

The **Internet Protocol Version 6** (**IPv6**) Working Group has prescribed various products that can aid in the planning for the transition from **Internet Protocol Version 4** (**IPv4**) to IPv6. The Node's Transition Plan should prepare these products to ensure that all the required activities are addressed.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: IPv6

Evaluation Criteria:

1) Test:

Are the Internet Protocol Version 6 (IPv6) Working Group products in the Node's Transition Plan?

Procedure:

Look for the Working Group products in the Node's Transition Plan.

Example:

None.

G1592

Include interoperability testing in the plan as part of the Internet Protocol Version 6 (IPv6) transition plan for a Node.

Rationale:

During the **DoD** transition period, a mixed **IPv4/IPv6** environment will exist. Interoperability testing with both standards will ensure the Node can fully function during the transition period with all other Nodes.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: IPv6

Evaluation Criteria:

1) Test:

Does the Node's IPv6 transition plan address interoperability testing in a mixed environment?

Procedure:

Review the transition plan and verify that a test plan exists that specifically addresses interoperability testing in a mixed IP environment.

Example:

None.

G1595

Implement Domain Name System (DNS) to manage hostname/address resolution within the Node.

Rationale:

Using **Domain Name System** (**DNS**) obviates the need for hard-coding **Internet Protocol** (**IP**) addresses within the Node. In addition, DNS servers local to the Node allow for stable access of replicated entries from outside the Node.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 4: Node Guidance / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: IPv6

Evaluation Criteria:

1) Test:

Are there any hard coded Internet Protocol (IP) addresses within the source code or data files?

Procedure:

Look at the source code, properties files and descriptor files for the occurrence of Internet Protocol Version 4 (IPv4) or Internet Protocol Version 6 (IPv6) Internet Protocol (IP) addresses.

Example:

None.

2) Test:

Is there a Domain Name System (DNS) server in the Node acquisition list?

Procedure:

Look for a Domain Name System (DNS) server within the parts list for the Node.

Example:

None.

G1596

Use Domain Name System (DNS) Mail eXchange (MX) Record capabilities to configure electronic mail delivery to the Node.

Rationale:

Utilizing the **Domain Name System** (**DNS**) **Mail eXchange (MX) record** capability will avoid the need to hard code delivery routes and instructions within a Node's email system and buffers it from physical changes made to email delivery points and routes outside of the Node. The DNS MX record is a standard and commonly accepted mechanism for resolving email delivery routes and addresses across the Internet.
Internet Engineering Task Force (**IETF**) Request for Comments (RFC) 2821 of April 2001 established rules for MX record usage.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 4: Node Guidance / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure

Evaluation Criteria:

1) Test:

Are there **Mail eXchange (MX) Records** defined within the **Domain Name System** (**DNS**)?

Procedure:

Look at the Domain Name System (DNS) records for Mail eXchange (MX) Records.

Example:

None.

G1598

Allow dynamic Domain Name System (DNS) updates to the Node's internal DNS service by local Dynamic Host Configuration Protocol (DHCP) server(s).

Rationale:

There are two basic methods for assigning of **Internet Protocol** (**IP**) addresses within a network: static and dynamic. Static addresses are assigned to a particular system and never change. Dynamic Internet Protocol (IP) addresses are issued for a variable length of time: the ***DCHP lease time***. **Dynamic Host Configuration Protocol** (**DHCP**) is the principle mechanism used to assign and manage dynamic IP addresses. If the DHCP servers are allowed to update the **Domain Name System** (**DNS**), then the number of static addresses required by the system can be drastically reduced with preference being given to requesting services by domain name rather than IP address.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 4: Node Guidance / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Dynamic Host Configuration Protocol (DHCP)
NESI / Part 4: Node Guidance / Node Transport / Network Services / Dynamic Host Configuration Protocol (DHCP)
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure

Evaluation Criteria:

1) Test:

Does the Domain Name System (DNS) server in the Node acquisition list support updates from Dynamic Host Configuration Protocol (DHCP) Servers?

Procedure:

Review the Domain Name System (DNS) server specification to confirm that it supports such operations.

Example:

None.

G1599

Simultaneously support Internet Protocol Version 4 (IPv4) and Internet Protocol Version 6 (IPv6) in the Node's Domain Name System (DNS) service.

Rationale:

During the transition period in the DoD community (FY06-FY15) networks, services and applications will be in a mixed environment. The Domain Name System (DNS) returns different address records depending on the Internet Protocol (IP) environment: A records for IPv4 or AAAA records for IPv6. A DNS must be able to support both.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 4: Node Guidance / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: IPv6

Evaluation Criteria:

1) Test:

Does the Domain Name System (DNS) server support both A and AAAA records?

Procedure:

Review the Domain Name System (DNS) specification to confirm that it supports both A and AAAA records.

Example:

None.

G1600

Obtain Internet Protocol Version 6 (IPv6) addresses to use for DoD IP addressable resources from DISA.

Rationale:

All the **Internet Protocol** (**IP**) addresses in use on a DoD network must be from an appropriate clearing house in order to maintain control and accountability on the network. **DISA** is the clearing house for all DoD addresses.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 4: Node Guidance / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Internet Protocol (IP)
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Internet Protocol (IP)
NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: IPv6

Evaluation Criteria:

1) Test:

Is there a proper entry in the Military Network Information Center (MILNIC) for every IP address assigned to the system?

Procedure:

Verify an adequate address allocation has been made in the Military Network Information Center (MILNIC) for the system.

Example:

None.

G1601

Use configurable routers to provide dynamic Internet Protocol (IP) address management using the Dynamic Host Configuration Protocol (DHCP).

Rationale:

There are two basic methods for assigning of **Internet Protocol** (**IP**) addresses within a network: static and dynamic. Static addresses are assigned to a particular system and never change. Dynamic IP addresses are issued for a variable length of time: the ***DCHP lease time***. The **Dynamic Host Configuration Protocol** (**DHCP**) is the principle mechanism used to assign and manage dynamic IP addresses.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 4: Node Guidance / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Subnets and Overlay Networks / Broadcast, Multicast, and Anycast
NESI / Part 4: Node Guidance / Node Transport / Subnets and Overlay Networks / Broadcast, Multicast, and Anycast
NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Dynamic Host Configuration Protocol (DHCP)
NESI / Part 4: Node Guidance / Node Transport / Network Services / Dynamic Host Configuration Protocol (DHCP)
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Network Connectivity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Inter-Network Connectivity

Evaluation Criteria:

1) Test:

Does the router in the Node acquisition list support Dynamic Host Configuration Protocol (DHCP)?

Procedure:

Review the router specification to confirm that it supports such operations.

Example:

None.

G1602

Use configurable routers to provide static Internet Protocol (IP) addresses.

Rationale:

Some network **Components** such as the **routers** themselves and other security related services must reside on static **Internet Protocol** (**IP**) addresses. Serious comprises in the network can arise if these services are allowed to be dynamic.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 4: Node Guidance / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Network Connectivity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Inter-Network Connectivity

Evaluation Criteria:

1) Test:

Does the **router** in the Node acquisition list support static **Internet Protocol** (**IP**) addressing?

Procedure:

Review the router specification to confirm that it supports such operations.

Example:

None.

G1604

Use configurable routers to provide time synchronization services using Network Time Protocol (NTP).

Rationale:

Over time, most computer clocks drift. **Network Time Protocol** (**NTP**) is one way to ensure that a computer clock stays accurate. Unfortunately, in order to stay synchronized, a network connection needs to be maintained. In environments that have limited bandwidth or poor **quality of service** (**QoS**) this can become a major issue.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 4: Node Guidance / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Network Time Service
NESI / Part 4: Node Guidance / Node Transport / Network Services / Network Time Service
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Network Connectivity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Inter-Network Connectivity

Evaluation Criteria:

1) Test:

Does the **router** in the Node acquisition list support NTP Service?

Procedure:

Review the routers specification to confirm that it supports such operations.

Example:

None.

G1605

Use configurable routers to provide multicast addressing.

Rationale:

Multicast addresses identify interfaces that allow a packet to be sent to all the addresses registered for the multicast service. This allows network to easily support applications such as **collaboration**, audio and video.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 4: Node Guidance / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Network Connectivity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Inter-Network Connectivity

Evaluation Criteria:

1) Test:

Does the **router** in the Node acquisition list support NTP Service?

Procedure:

Review the router specification to confirm that it supports such operations.

Example:

None.

G1606

Manage routers remotely from within the Node.

Rationale:

Router manufactures routinely provide tools to enable remote, over the network, router configuration and management in addition to a local console within the **Node**. These tools can speed and centralize the administration of the routers in a Node.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 4: Node Guidance / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Decentralized Operations and Management
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Network Connectivity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Inter-Network Connectivity

Evaluation Criteria:

1) Test:

Does the **router** in the Node acquisition list support remote management?

Procedure:

Review the router specification to confirm that it supports such operations.

Example:

None.

G1607

Configure routers according to National Security Agency (NSA) Router Security Configuration guidance.

Rationale:

The *Router Security Configuration Guide* provides technical guidance intended to help network administrators and security officers improve the security of their networks. It contains principles and guidance for secure configuration of **Internet Protocol** (**IP**) routers, with detailed instructions for Cisco System routers. The information presented can be used to control access, help resist attacks, shield other network **Components**, and help protect the integrity and confidentiality of network traffic.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 4: Node Guidance / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Network Connectivity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Concurrent Transport of Information Flows
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Inter-Network Connectivity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Encryption and HAIPE

Evaluation Criteria:

1) Test:

Is the **Router** Security Checklist complete and up to date?

Procedure:

Check for the occurrence of the checklist; there should be a copy for every time the checklist has been completed. The checklist should indicate the date, time and results of the checklist with recommendation actions.

Example:

Router Security Checklist
This security checklist is designed to help review router security configuration and remind a user of any security areas that might be missed.

- Router security policy written, approved, distributed.
- Router IOS version checked and up to date.
- Router configuration kept off-line, backed up, access to it limited.
- Router configuration is well-documented, commented.
- Router users and passwords configured and maintained.
- Password encryption in use, enable secret in use.
- Enable secret difficult to guess, knowledge of it strictly limited. (if not, change the enable secret immediately)
- Access restrictions imposed on Console, Aux, VTYs.
- Unneeded network servers and facilities disabled.
- Necessary network services configured correctly (e.g. DNS)
- Unused interfaces and VTYs shut down or disabled.
- Risky interface services disabled.

- Port and protocol needs of the network identified and checked.
- Access lists limit traffic to identified ports and protocols.
- Access lists block reserved and inappropriate addresses.
- Static routes configured where necessary.
- Routing protocols configured to use integrity mechanisms.
- Logging enabled and log recipient hosts identified and configured.
- Router's time of day set accurately, maintained with NTP.
- Logging set to include consistent time information.
- Logs checked, reviewed, archived in accordance with local policy.
- SNMP disabled or enabled with good community strings and ACLs.

G1608

Obtain reference time from a standard globally synchronized time source.

Rationale:

Currently, Network Time Service is not a homogeneous service across the **Global Information Grid** (**GIG**). Security directives prevent **IP**-based time synchronization across **firewall** boundaries (e.g., AFI 33-115, 16). An example of a precise globally synchronized time source is a **Global Positioning System** (**GPS**) system.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Network Time Service
NESI / Part 4: Node Guidance / Node Transport / Network Services / Network Time Service
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Network Connectivity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Inter-Network Connectivity

Evaluation Criteria:

1) Test:

Does the acquisition list include a precise globally synchronized time source such as a **Global Positioning System** (**GPS**)?

Procedure:

Review the acquisition list for a precise globally synchronized time source such as a **Global Positioning System** (**GPS**) that can provide accurately synchronized time.

Example:

None.

G1609

Arrange for a backup time source.

Rationale:

Use one or more backup time sources. The most common type of backup time sources are crystal oscillators. The physical characteristics of the piezoelectric quartz crystal produce electrical oscillations at an extremely accurate frequency which can be used to mark time.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Network Time Service
NESI / Part 4: Node Guidance / Node Transport / Network Services / Network Time Service
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Network Connectivity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Inter-Network Connectivity

Evaluation Criteria:

1) Test:

Does the acquisition list include a backup time source?

Procedure:

Review the acquisition list for a backup time system that can be used to synchronize time accurately.

Example:

Crystal oscillator examples include cesium or rubidium. The following table shows crystal oscillator types:

MCXO	microcomputer-compensated crystal oscillator
OCVCXO	oven-controlled voltage-controlled crystal oscillator
OCXO	oven-controlled crystal oscillator
RbXO	rubidium crystal oscillators (RbXO)
TCVCXO	temperature-compensated-voltage controlled crystal oscillator
TCXO	temperature-compensated crystal oscillator
VCXO	voltage-controlled crystal oscillator

G1610

Configure the Dynamic Host Configuration Protocol (DHCP) services to assign multicast addresses.

Rationale:

When **Dynamic Host Configuration Protocol** (**DHCP**) services assign temporary **Internet Protocol** (**IP**) addresses to clients, the clients may wish to participate in a **multicast** service. Therefore, the DHCP service must support the assignment of multicast addresses as part of normal operations.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Subnets and Overlay Networks / Broadcast, Multicast, and Anycast
NESI / Part 4: Node Guidance / Node Transport / Subnets and Overlay Networks / Broadcast, Multicast, and Anycast
NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Dynamic Host Configuration Protocol (DHCP)
NESI / Part 4: Node Guidance / Node Transport / Network Services / Dynamic Host Configuration Protocol (DHCP)
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Network Connectivity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Inter-Network Connectivity

Evaluation Criteria:

1) Test:

Does the **router** in the Node acquisition list support the assignment of **multicast** Internet Protocol (**IP**) addresses as part of the normal **Dynamic Host Configuration Protocol** (DHCP) service?

Procedure:

Review the **router** specification to confirm that it supports such operations.

Example:

None.

G1611

Implement Internet Protocol (IP) gateways to interoperate with the Global Information Grid (GIG) until IP is supported natively for Components that are not IP networked.

Rationale:

Component systems such as aircraft data links (**Link-16**, **SADL**, etc), should implement **Transmission Control Protocol/Internet Protocol** (**TCP/IP**) gateways to interoperate with the **Global Information Grid** (**GIG**) until TCP/IP is supported natively. This acts as an interim step that can be used to bridge the **Internet Protocol** (**IP**) divide.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Packet Switched Infrastructure
NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Integration of Non-IP Transports
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Integration of Non-IP Transports

Evaluation Criteria:

1) Test:

Are **Internet Protocol** (**IP**) and non-IP networks connected via gateways?

Procedure:

Verify IP and non-IP networks are connected via one or more gateways.

Example:

1. Identify gateways between IP and non-IP networks within DoDAF diagrams.

2. Verify successful data translation between IP and non-IP networks via a gateway such as verifying track data transmission between a Link 16 equipped user and a GIG edge IP router.

G1613

Prepare a Node to host new Component services developed by other Nodes or by the enterprise itself.

Rationale:

A key aspect of an open systems approach to interoperability is **modular design** which is also a basic tenet of good development practice. Modularity will support the dynamic redeployment of a **Component** into different Nodes that requires the capabilities of the Component thus promoting broader interoperability between different Nodes and Components. Where possible, Nodes should adopt standards based, platform independent frameworks that facilitate ***pluggable*** deployment capabilities for Components so it can leverage the capabilities developed elsewhere.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Cross-Security-Domains Exchange
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Client Platform

Evaluation Criteria:

1) Test:

Does the Node support the elements of a modern component based framework such as **Java Platform, Enterprise Edition** (**Java EE**), **.NET** or **CORBA**?

Procedure:

Look for the existence of Java Platform, Enterprise Edition (Java EE), .NET or CORBA frameworks with in the Node's Component list or in its delivered software.

Example:

None.

G1619

Configure clients with a Common Access Card (CAC) reader.

Rationale:

DoD Instruction 8520.2, *Public Key Infrastructure (PKI) and Public Key (PK) Enabling* [R1206], defines **Common Access Card** (**CAC**) applicability and scope, in part, as follows:

This Instruction applies to:... 2.4. All DoD unclassified and classified information systems including networks (e.g., Non-secure Internet Protocol (IP) Router Network , Secret Internet Protocol Router Network, Web servers, and e-mail systems. Excluded are Sensitive Compartmented Information, and information systems operated within the Department of Defense that fall under the authority of the Director of Central Intelligence Directive (DCID) 6/3 (reference (h)).

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Policy / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Client Platform / Common Access Card (CAC) Reader

Evaluation Criteria:

1) Test:

Do all the **client** and **server** hardware come equipped with **Common Access Card** (CAC) Readers?

Procedure:

Review the hardware list and verify that all hardware comes with or has external CAC readers.

Example:

None.

G1621

Provide a Node Web infrastructure for all Components within the Node.

Rationale:

A Web application infrastructure includes those elements which allow an application developer to deploy an application at a Node without regard to how the application will display results to an end user, execute or be deployed. By providing open access to a common Web infrastructure, Components are relieved of having to implement their own divergent Web infrastructure, thereby promoting increased interoperability and reusability.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Infrastructure

Evaluation Criteria:

1) Test:

Does the Node acquisition list include duplicate Web application infrastructure elements that are not provided by the Node?

Procedure:

Review the acquisition list for Web application infrastructure elements (Web Portal, Web Server and Web Application Containers). If duplicates are found or not provided by Node, address the issue with the appropriate stakeholders.

Example:

None.

G1622

Implement commercial off-the-shelf (COTS) software that protects against malicious code on each operating system in the Node in accordance with the Desktop Application Security Technical Implementation Guide (STIG).

Rationale:

The viral and worm assault on computing resources is major concern but is not strictly limited to DoD hardware and operating systems. It has become a ubiquitous, wide spread problem that spreads destruction indiscriminately. Since the problem is not strictly a DoD problem, **commercial off-the-shelf** (**COTS**) solutions are always being updated to meet the current threats and are essential in protecting the assets. All hardware platforms should employ virus and worm detection and removal software that is routinely run (especially on hardware the runs Microsoft products).

Note: *For purposes of this guidance, anti virus software includes related update and maintenance capabilities typically available with such packages.*

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Host Information Assurance
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Integrity / Computing Infrastructure Integrity
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Integrity / Computing Infrastructure Integrity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Other Design Tenets

Evaluation Criteria:

1) Test:

Do all hardware devices listed in the Node acquisition list have COTS licensed virus and worm detection software?

Procedure:

Review the Node acquisition list and make sure there is one license for each piece of computer hardware.

Example:

None.

2) Test:

Do all hardware devices listed in the Node acquisition list have COTS virus and worm detection software installed?

Procedure:

Review the prerequisites in the installation manual for virus and worm software.

Example:

None.

G1623

Implement personal firewall software on computers used for remote connectivity in accordance with the Desktop Applications, Network, and Enclave Security Technical Implementation Guides (STIGs).

Rationale:

All hardware that is plugged into a network is subject to attack by hackers. In addition to hardware **firewalls** that may be in place, every piece of hardware should be protected by a software firewall. This is especially important for forward deployed computers that may not have an external firewalls on the local network. Personal firewalls continuously monitor the activity on the local computer network interface and detect possible hostile attacks. The user has the discretion to block hostile attacks permanently or for a particular occasion. Since this problem is not restricted to DoD assets, **commercial off-the-shelf** (**COTS**) products are continuously updated to meet the latest threats and are essential in meeting these threats.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Decentralized Operations and Management
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Host Information Assurance
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Integrity / Computing Infrastructure Integrity
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Integrity / Computing Infrastructure Integrity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Inter-Network Connectivity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Other Design Tenets

Evaluation Criteria:

1) Test:

Do all the hardware devices listed in the Node acquisition list have COTS software firewall licensed software?

Procedure:

Review the Node acquisition list and make sure there is one license for each piece of computer hardware.

Example:

None.

2) Test:

Do all hardware devises listed in the Node acquisition list have COTS **firewall** software installed and is it enabled?

Procedure:

Review the prerequisites in the installation manual for firewall software.

Example:

None.

G1624

Install anti-spyware software on all Windows Desktop computers.

Rationale:

Spyware is a category of malicious software that can impact system operation in ways similar to virus and other intrusions. Extending the principles of protection against viruses and other intrusions to spyware is an essential activity to ensure stable system operation and security. Anti-spyware software is required on all Windows computers per the Windows Desktop Application **Security Technical Implementation Guide** (**STIG**).

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Host Information Assurance
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Other Design Tenets

Evaluation Criteria:

1) Test:

Do all the Windows Desktop computers listed in the Node acquisition list have COTS software anti-spyware licensed software?

Procedure:

Review the Node acquisition list and make sure there is one license for each piece of computer hardware.

Example:

None.

2) Test:

Do all Windows Desktops listed in the Node acquisition list have COTS anti-spyware software installed and is it enabled?

Procedure:

Review the prerequisites in the installation manual for anti-spyware software.

Example:

None.

G1625

Provide a commercial off-the-shelf Directory Service that all of the components of a Node can use.

Rationale:

A Directory Service is a service that stores information about objects on a computer network. Common objects stored by a Directory Service include network users, common resources (such as shares and printers), authentication and authorization information.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / NCES Directory Services

Evaluation Criteria:

1) Test:

Is an Open Source directory service going to be used?

Procedure:

Review the prerequisites in the installation manual for open source directory service software.

Example:

None.

2) Test:

Is there a COTS directory service listed in the Node acquisition list?

Procedure:

Review the Node acquisition list and make sure there is one license for a directory service.

Example:

None.

G1626

Identify which Core Enterprise Services (CES) capabilities the Node Components require.

Rationale:

A Node needs to determine the set of **Core Enterprise Services** (**CES**) its **components** will require in order to ensure efficient prioritization of activities and resources to provide those services. **NCES** has defined a set of common capabilities that help categorize types of services that may be required by a Node's components. Identification of the capabilities the components require will help the Node determine which services to implement.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Open Architecture

Evaluation Criteria:

1) Test:

Does the list of components that comprise the Node indicate which CES capabilities are required to deploy each Component?

Procedure:

Review the list of components and verify that they have indicated which CES capabilities are required to support the component.

Example:

None.

G1627

Identify the priority of each Core Enterprise Services (CES) capability the Node components require.

Rationale:

Identifying the priority of capabilities required by the Node's **Components** will assist the Node in allocation of scarce resources towards the delivery of **CES** in the Node and minimize risks during deployment of Components within the Node. Some capabilities are ***essential*** at getting a component Deployed at a Node. Some are essential for a particular component increment. With this information the Node can construct a schedule that supports the transition and evolution of the current federation of systems to the **Global Information Grid** (**GIG**) vision.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Open Architecture

Evaluation Criteria:

1) Test:

Does the list of components that comprise the Node indicate the priority of the CES capabilities either relative to each other or as of a date?

Procedure:

Review the list of components and verify that they have indicated what the priority of the CES capabilities either relative to each other or as of a date.

Example:

None.

G1629

Identify which Net-Centric Enterprise Services (NCES) capabilities the Node requires during deployment.

Rationale:

Relying on a high-bandwidth **Transmission Control Protocol/Internet Protocol** (**TCP/IP**) network connection is not a reality for many deployed Nodes. These Nodes will have to develop many of their own **CES** capabilities for use by their member **components** while deployed. When the Node is not deployed, it may rely on proxies to the **Net-Centric Enterprise Services** (**NCES**) services.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Joint Net-Centric Capabilities
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Open Architecture

Evaluation Criteria:

1) Test:

Does the Node have a list of **Net-Centric Enterprise Services** (**NCES**) capabilities that it depends on while deployed?

Procedure:

Review the Node's documents for a list of Net-Centric Enterprise Services (NCES) capabilities required by the Node while deployed.

Example:

None.

G1630

Comply with the applicable Global Information Grid (GIG) Key Interface Profiles (KIPs) for implemented Core Enterprise Services (CES) in the Node.

Rationale:

When a **CES** is implemented locally, use the **Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**) developed by **DISA** as the authoritative definition of the interfaces. This allows a **component** that is hosted by one Node to be hosted on another Node with a minimal impact.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Key Interface Profile (KIP)
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Key Interface Profile (KIP)
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Key Interface Profile (KIP)
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Key Interface Profile (KIP)
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES and Intermittent Availability
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES and Intermittent Availability
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES and Intermittent Availability
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES and Intermittent Availability
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Open Architecture

Evaluation Criteria:

1) Test:

Do all **CES** used locally within the Node implement the applicable **Global Information Grid** (**GIG**) **Key Interface Profile** (**KIP**)?

Procedure:

Verify that the interfaces for Core Enterprise Services (CES) implement Global Information Grid (GIG) Key Interface Profiles (KIPs) for that CES.

Example:

None.

G1631

Expose Core Enterprise Services (CES) that comply with the applicable Global Information Grid (GIG) Key Interface Profiles (KIPs) in all Node services proxies.

Rationale:

A Node may expose or control access to **Global Information Grid** (**GIG**) **CES** by using **proxies**. This allows a **Component** that is hosted by one Node to be hosted on another Node with a minimal impact.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Key Interface Profile (KIP)
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Key Interface Profile (KIP)
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Key Interface Profile (KIP)
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Key Interface Profile (KIP)
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES and Intermittent Availability
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES and Intermittent Availability
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES and Intermittent Availability
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES and Intermittent Availability
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Open Architecture

Evaluation Criteria:

1) Test:

Do all **CES proxies** locally defined within the Node expose CES using the applicable **Global Information Grid** (**GIG**) **Key Interface Profile** (**KIP**)?

Procedure:

Verify that the interfaces for CES proxies follow Key Interface Profiles (KIPs) for that Global Information Grid (GIG) KIP.

Example:

None.

G1632

Certify and accredit Nodes with all applicable DoD Information Assurance (IA) processes.

Rationale:

Nodes are part of the DoD **Global Information Grid** (**GIG**) and are consequently required to have DoD **Information Assurance** (**IA**) certification and accreditation. Details for certification and accreditation are specified in DoD Directive 8500.1 [R1197], DoD Instruction 8500.2 [R1198], DoD Directive 8580.1 [R1199], and DoD Instruction 8510.01 [R1291]. Satisfaction of these requirements results in IA compliance verification of the Node.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Net-Centric IA Posture and Continuity of Operations
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Information Assurance (IA)
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Information Assurance (IA)
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Information Assurance (IA)
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Information Assurance (IA)
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Other Design Tenets

Evaluation Criteria:

1) Test:

Does the Node have DoD **Information Assurance** (**IA**) certification and accreditation?

Procedure:

Ask to examine the certification and accreditation reports.

Example:

None.

G1633

Host only DoD Information Assurance (IA) certified and accredited Components.

Rationale:

Nodes that expose the external Node users to non-certified or non-accredited **Components** represent a risk to the stability of the entire Node network and can introduce interoperability issues between Nodes (and related Components).

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Net-Centric IA Posture and Continuity of Operations
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Information Assurance (IA)
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Information Assurance (IA)
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Information Assurance (IA)
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Information Assurance (IA)
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Other Design Tenets

Evaluation Criteria:

1) Test:

Does the Node have a plan to scan all Components on a routine basis?

Procedure:

Look for a plan and examine the results of the scan.

Example:

None.

G1634

Certify and accredit Components with all applicable DoD Information Assurance (IA) processes.

Rationale:

Each **Component** could theoretically be deployed on any Node. Therefore, it is the responsibility of the Component to be DoD **Information Assurance** (**IA**) certified and accredited.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Net-Centric IA Posture and Continuity of Operations
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Information Assurance (IA)
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Information Assurance (IA)
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Information Assurance (IA)
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Information Assurance (IA)
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Other Design Tenets

Evaluation Criteria:

1) Test:

Are all the **Components** DoD **Information Assurance** (**IA**) certified and accredited?

Procedure:

Examine the certification and accreditation reports.

Example:

None.

G1635

Make Nodes that will be part of the Global Information Grid (GIG) consistent with the *GIG Integrated Architecture*.

Rationale:

The **Global Information Grid** (**GIG**) architecture describes the basic, high level architecture in which Nodes reside. It is an integrated architecture consisting of the various **DoDAF** views. It provides a common lexicon and defines a basic infrastructure for the performance of information exchanges with other GIG Nodes using the **GIG Enterprise Services** (**GES**) and the **Net-Centric Enterprise Services** (**NCES**). The GIG Integrated Architecture is available via the DoD Architecture Repository System (DARS), https://dars1.army.mil/ [user account and PKI certificate required for access].

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Integrated Architectures
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Integrated Architectures
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Integrated Architectures
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Integrated Architectures
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Service-Oriented Architecture (SOA)
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability

Evaluation Criteria:

1) Test:

Are there **DoDAF** integrated architecture products defined for the Node that are consistent with the **GIG** Integrated Architecture?

Procedure:

Look for the occurrence of **Operational View** (**OV**), **Systems and Services View** (**SV**), **Technical Standards View** (**TV**) and **All Views** (**AV**).

Example:

None.

G1636

Comply with the Net-Centric Operations and Warfare Reference Model (NCOW RM).

Rationale:

> ***Note:*** *CJCSI 6212.01E removed the NCOW RM element of the Net-Ready Key Performance Parameter (NR-KPP), integrating the components of the former NCOW RM into other elements of the NR-KPP.*

The **Net-Centric Operations and Warfare Reference Model** (**NCOW RM**) focused on achieving net-centricity. Compliance with the NCOW RM translated to articulating how each Node approached and implemented net-centric features. Compliance did not require separate documentation; rather, it required that a Node address, within existing architecture, analysis, and program architecture documentation, the issues identified by using the model, and further, make explicit the path to net-centricity the program is taking.

Node compliance with the NCOW RM is demonstrated through inspection and analysis:

- Use of NCOW RM definitions and vocabulary;
- Incorporation of NCOW RM **Operational View** (**OV**) capabilities and services in the materiel solution;
- Incorporation of NCOW RM **Technical Standards View Information Technology** (**IT**) and **National Security Systems** (**NSS**) standards in the **TV** products developed for the materiel solution.

Compliance with the NCOW RM initially was a critical component of compliance with the **Net-Ready Key Performance Parameter** (**NR-KPP**).

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Net-Centric Operations and Warfare Reference Model (NCOW RM)
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Net-Centric Operations and Warfare Reference Model (NCOW RM)
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Net-Centric Operations and Warfare Reference Model (NCOW RM)
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Net-Centric Operations and Warfare Reference Model (NCOW RM)
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Service-Oriented Architecture (SOA)
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability

Evaluation Criteria:

1) Test:

Have the instructions in Chairman of the Joint Chiefs of Staff Instruction (CJCSI) 3170.01 been used to check the Node for Net-Centric Operations and Warfare Reference Model (NCOW RM) compliance?

Procedure:

Check Node documentation.

Example:

2) Test:

Have the instructions in Chairman of the Joint Chiefs of Staff Instruction (CJCSI) 6212.01 been used to check the Node for Net-Centric Operations and Warfare Reference Model (NCOW RM) compliance?

Procedure:

Check Node documentation.

Example:

3) Test:

Have the instructions in the Defense Acquisition University (DAU) Guidebook section 7.2.6 been used to check the Node for NCOW RM compliance?

Procedure:

Check Node documentation.

Example:

G1637

Make Node-implemented directory services comply with the directory services Global Information Grid (GIG) Key Interface Profiles (KIPs).

Rationale:

When directory services are implemented locally, use the **Global Information Grid** (**GIG**) **KIPs** developed by DISA as the authoritative definition of the interfaces. This allows a **Component** that is hosted by one Node to be hosted on another node with a minimal impact.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Service-Oriented Architecture (SOA)
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability

Evaluation Criteria:

1) Test:

Do all directory services used locally within the Node implement the applicable **Global Information Grid** (**GIG**) **Key Interface Profile** (**KIP**)?

Procedure:

Verify that the interfaces for directory services implement Global Information Grid (GIG) Key Interface Profiles (KIPs) for that directory services.

Example:

None.

G1638

Comply with the directory services Global Information Grid (GIG) Key Interface Profiles (KIPs) in Node directory services proxies.

Rationale:

A Node may expose or control access to **Global Information Grid** (**GIG**) directory services by using **proxies**. This allows a **Component** that is hosted by one Node to be hosted on another node with a minimal impact.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Service-Oriented Architecture (SOA)
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability

Evaluation Criteria:

1) Test:

Do all directory services **proxies** locally defined within the Node expose directory services using the applicable **Global Information Grid** (**GIG**) **Key Interface Profile** (**KIP**)?

Procedure:

Verify that the interfaces for directory services proxies follow Key Interface Profiles (KIPs) for that Global Information Grid (GIG) KIPs.

Example:

None.

G1639

Describe Components exposed by the Node as specified by the Service Definition Framework

Rationale:

The construction of registry entries is specified by the **Service Definition Framework** (**SDF**) documented in Net-Centric Implementation Directives (NCIDs) S300. The common Service Definition Framework that serves as the basis for adequately describing the offered **Component** service from both a provider's and consumer's perspective. It describes the contract between the Component service provider and the Component service consumer, and serves as the basis for a **Service Level Agreement** (**SLA**). The common service definition framework consists of elements that include interface, service level, security and implementation information.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Enterprise Service Management
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Visibility - Registered / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Visibility - Discoverable / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Registered / Service Enablers / Service Discovery
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 4: Node Guidance / Services / Service Enablers / Service Discovery

Evaluation Criteria:

1) Test:

Is there a **Service Definition Framework** (**SDF**) available for each of the Components' Services exposed through the Node?

Procedure:

Look for a Service Definition Framework (SDF) for each Component service exposed through the Node.

Example:

None

G1640

Register components that a Node exposes as SOAP Web services with DoD-approved registries.

Rationale:

Register Web services in accordance with DoD governance including the Chairman of the Joint Chiefs of Staff Instruction *Interoperability and Supportability of Information Technology and National Security Systems*, CJCSI 6212.01E,[R1175] and the **DoD Metadata Registry** processes and procedures. An appropriate way to publish and discover components that a Node exposes as SOAP Web services is to use the DoD Metadata Registry which DISA manages. The DISA **Net-Centric Enterprise Services** (**NCES**) Program includes a Service Discovery capability which enables publishing service information to the Enterprise Service Registry and the DoD Metadata Registry. DISA registry information for SOAP services uses the **Universal Description, Discovery, Integration** (**UDDI**) standard.[R1280]

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Visibility - Registered / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Visibility - Discoverable / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Registered / Service Enablers / Service Discovery
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 4: Node Guidance / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability

Evaluation Criteria:

1) Test:

Are the components that a Node exposes as SOAP Web services registered in the DISA-managed **DoD Metadata Registry**?

Procedure:

Use the DISA NCES Service Discovery search service to look for the components that a Node exposes as SOAP Web services in the DoD Metadate Registry.

Example:

None.

G1641

Comply with the Service Discovery Global Information Grid (GIG) Key Interface Profiles (KIPs) in Node-implemented Service Discovery (SD).

Rationale:

When a **Service Discovery** (**SD**) is implemented locally, use the **Global Information Grid** (**GIG**) KIPs developed by DISA as the authoritative definition of the interfaces. This allows a **Component** that is hosted by one Node to be hosted on another node with a minimal impact.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Visibility - Registered / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Visibility - Discoverable / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Registered / Service Enablers / Service Discovery
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 4: Node Guidance / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Service-Oriented Architecture (SOA)
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability

Evaluation Criteria:

1) Test:

Does the **Service Discovery** (**SD**) used locally within the Node implement the applicable **Global Information Grid** (**GIG**) **Key Interface Profile** (**KIP**)?

Procedure:

Verify that the interfaces for Service Discovery (SD) implement Global Information Grid (GIG) Key Interface Profiles (KIPs) for that Service Discovery.

Example:

None.

G1642

Comply with the Service Discovery (SD) Global Information Grid (GIG) Key Interface Profiles (KIPs) in Node Service Discovery proxies.

Rationale:

A Node may expose or control access to **Global Information Grid** (**GIG**) **Service Discovery** (**SD**) by using **proxies**. This allows a **Component** that is hosted by one Node to be hosted on another node with a minimal impact.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Visibility - Registered / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Visibility - Discoverable / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Registered / Service Enablers / Service Discovery
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 4: Node Guidance / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Service-Oriented Architecture (SOA)
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability

Evaluation Criteria:

1) Test:

Do the **Service Discovery** (**SD**) **proxies** locally defined within the Node expose Service Discovery using the applicable **Global Information Grid** (**GIG**) **Key Interface Profile** (**KIP**)?

Procedure:

Verify that the interfaces for Service Discovery (SD) proxies follow KIPs for that Global Information Grid (GIG) Key Interface Profiles (KIPs).

Example:

None.

G1643

Comply with the Federated Search - Registration Web Service (RWS) Global Information Grid (GIG) Key Interface Profiles (KIPs) in Node implemented Federated Search - Registration Web Service (RWS).

Rationale:

When a **Federated Search** - **Registration Web Service** (**RWS**) is implemented locally, use the **Global Information Grid** (**GIG**) KIPs developed by **DISA** as the authoritative definition of the interfaces. This allows a **Component** that is hosted by one Node to be hosted on another node with a minimal impact.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Accessibility - Operational / NCES Federated Search
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / NCES Federated Search

Evaluation Criteria:

1) Test:

Does a **Federated Search** - **Registration Web Service** (**RWS**) used locally within the Node implement the applicable **Global Information Grid** (**GIG**) **Key Interface Profile** (**KIP**)?

Procedure:

Verify that the interfaces for Federated Search - Registration Web Service (RWS) implement Global Information Grid (GIG) Key Interface Profiles (KIPs) for that Federated Search - Registration Web Service (RWS).

Example:

None.

G1644

Comply with the Federated Search - Search Web Service (SWS) Global Information Grid (GIG) Key Interface Profiles (KIPs) in Node implemented Federated Search - Search Web Service (SWS).

Rationale:

When a **Federated Search** - **Search Web Service** (**SWS**) is implemented locally, use the **Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**) developed by **DISA** as the authoritative definition of the interfaces. This allows a **Component** that is hosted by one Node to be hosted on another node with a minimal impact.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Accessibility - Operational / NCES Federated Search
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability

Evaluation Criteria:

1) Test:

Does **Federated Search** - **Search Web Service** (**SWS**) used locally within the Node implement the applicable **Global Information Grid** (**GIG**) **Key Interface Profile** (**KIP**)?

Procedure:

Verify that the interfaces for Federated Search - Search Web Service (SWS) implement Global Information Grid (GIG) Key Interface Profiles (KIPs) for that Federated Search - Search Web Service (SWS).

Example:

None.

G1645

Implement a local Content Discovery Service (CDS).

Rationale:

The node should implement the **Content Discovery Service** (**CDS**) as part of the node infrastructure to be shared among the **Components** hosted at the Node. A CDS will allow other Nodes and Components to find content within the node. The systems within the Node normally provide the content.

> ***Note:*** *If a Node is frequently disconnected, has intermittent connectivity, or is otherwise isolated, then hosting a local CDS might not be a practical solution for external content discovery and more effective means for internal discovery may be applicable.*

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Accessibility - Operational / NCES Federated Search
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability

Evaluation Criteria:

1) Test:

Does the Node implement the **Content Discovery Service** (**CDS**) **Global Information Grid** (**GIG**) **Key Interface Profile** (**KIP**)?

Procedure:

Look for an implementation at the Node of the Content Discovery Service (CDS) Global Information Grid (GIG) Key Interface Profiles (KIPs).

Example:

None.

G1646

Comply with the directory services Global Information Grid (GIG) Key Interface Profiles (KIPs) in Node Federated Search Services proxies.

Rationale:

A Node may expose or control access to **Global Information Grid** (**GIG**) **Federated Search** Services by using **proxies**. This allows a **Component** that is hosted by one Node to be hosted on another node with a minimal impact.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Accessibility - Operational / NCES Federated Search
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / Naval Open Architecture / Interoperability

Evaluation Criteria:

1) Test:

Do all **Federated Search** Services **proxies** locally defined within the Node expose Federated Search Services using the applicable **Global Information Grid KIP**?

Procedure:

Verify that the interfaces for Federated Search Services proxies follow KIPs for that Global Information Grid (GIG) Key Interface Profiles (KIPs).

Example:

None.

G1647

Provide access to the Federated Search Services.

Rationale:

Content Discovery Service can search across a set of Content Discovery Services and yield an integrated result. The current approach to providing this service is to harness an existing capability termed ***Federated Search*** developed under the **Horizontal Fusion** (**HF**) program. The capability utilizes the **DoD Discovery Metadata Specification** (**DDMS**).

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Accessibility - Operational / NCES Federated Search
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Data / Design Tenet: Provide Data Management
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Visibility / Design Tenet: Provide Data Management

Evaluation Criteria:

1) Test:

Does the Node provide access to the **Federated Search** Service **Global Information Grid** (**GIG**) **Key Interface Profile** (**KIP**)?

Procedure:

Look for a proxy or an implementation that provides access to the ***Federated Search***

Example:

None.

G1652

Use DoD PKI X.509 certificates for servers.

Rationale:

Using a DoD PKI X.509 **server certificate** identifies the server as being trusted by the DoD and guarantees that the server's identity is legitimate.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Policy / Design Tenet: Identity Management, Authentication, and Privileges
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Identity Management
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Identity Management

Evaluation Criteria:

1) Test:

Is the server certificate a valid DoD PKI X.509 certificate that is non-expired?

Procedure:

Open the server certificate and check that it is trusted by a trusted DoD root certificate.

Example:

None.

G1662

Follow the guidance provided in the Security Technical Implementation Guide (STIG) for Domain Name System (DNS) implementations.

Rationale:

As a fundamental common service on **IP**-based networks, **DNS** is often a focal point for network attackers. Following the **STIG** ensures alignment with DoD identified security practices and configurations. The STIG addresses implementation options such as the choice of basic DNS server types (primary, secondary, caching-only), use of a split-DNS design, location of servers in the network and relationship to other network components, secure administration, security of zone transfers, and initial configuration.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 4: Node Guidance / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Other Design Tenets

Evaluation Criteria:

1) Test:

Do the Node's **DNS** services follow the **STIG** for DNS implementations?

Procedure:

Compare Node DNS services configuration with those recommended by the STIG.

Example:

None.

G1667

Implement Virtual Private Networks (VPNs) in accordance with the guidance provided in the Network Security Technical Implementation Guide (STIG).

Rationale:

Virtual Private Networks provide a means for Node access to users outside the security enclave. To Network **STIG** provides recommendations on how to configure VPNs for secure access.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Subnets and Overlay Networks / Virtual Private Networks (VPN)
NESI / Part 4: Node Guidance / Node Transport / Subnets and Overlay Networks / Virtual Private Networks (VPN)
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Integrity / Network Infrastructure Integrity
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Integrity / Network Infrastructure Integrity
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Other Design Tenets

Evaluation Criteria:

1) Test:

Does the configuration of the Node's **VPN** servers follow the recommendations of the Network **STIG**?

Procedure:

Check VPN server configuration against recommended configurations in the Network STIG.

Example:

None.

BP1594

Examine the use of Transmission Control Protocol (TCP) extensions and other transport protocols that have been designed to mitigate risk for high bandwidth, high latency satellite communications.

Rationale:

TCP performance over satellite links is generally poor due to delays and blockages inherent to satellite links. TCP extensions (e.g., **IETF** RFC 1323) and other transport protocols that have been developed to mitigate this risk should be considered for high bandwidth, high latency satellite communications.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Mobility
NESI / Part 4: Node Guidance / Node Transport / Mobility
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Transport Goal

Evaluation Criteria:

1) Test:

If the system is involved in high bandwidth, high latency satellite communications, does the Node design address TCP performance?

Procedure:

Determine if parts of the system involve high bandwidth, high latency satellite communications and if so, look for a TCP extension.

Example:

None.

BP1597

Consider operational performance constraints in the design of the Node's Domain Name System (DNS).

Rationale:

Operational performance constraints such as narrow band width or intermittent service can have a large impact in how the **Domain Name System** (**DNS**) **server** is configured and consequently on the DNS chosen to support the Node.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 4: Node Guidance / Node Transport / Network Services / Domain Name System (DNS)

Evaluation Criteria:

1) Test:

Have the operational performance constraints been delineated and used to justify the **Domain Name System** (**DNS**) used by the Node?

Procedure:

Review the acquisition documents looking for justifications for the selection of the Domain Name System (DNS).

Example:

None.

BP1614

Plan a contingency response to the Node becoming a new component service within another Node.

Rationale:

While the complexities of nested Nodes are currently not addressed within *NESI Part 4*, nested Nodes are a possibility; thus, Nodes should be prepared to interact in such an environment. Review, in order to do contingence planning, the guidance for Nodes in Part 4; analyze the operational tradespace and the impact on the Node architecture, on infrastructure interoperability, and on any relevant service standards. Prepare the Node for such interactions by encouraging the proper definition of key interfaces and capabilities and creating a distinction between Nodal infrastructure and component capabilities. These distinctions would allow a Node, for example, to supplant its own infrastructure with those of its new parent Node (either directly or via proxies).

> ***Note:*** *The purpose of this practice is not necessarily to encourage nested Nodes, but to ensure that Nodes apply appropriate open* ***modular designs*** *both externally and internally to ensure greater interoperability in a variety of environments.*

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Cross-Security-Domains Exchange
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Client Platform

Evaluation Criteria:

1) Test:

Does the Node use standardized interfaces to obtain the services of routine activities?

Procedure:

Look for alignment and adherence to guidance of NESI Part 4 and open systems approaches.

Example:

None.

BP1615

Select Web browsers that support a wide breadth of current browser extension technologies.

Rationale:

Web browsers are a key application for allowing users to capitalize on the DoD vision of net-centric information sharing and access to distributed services. In order to ensure maximum interoperability with available services that may not be known a priori, browsers should support current standards and capabilities such as **JavaScript**, Java **applets**, and **plug-ins**.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Client Platform / Browser
NESI / Part 4: Node Guidance / User Environment / Browser

Evaluation Criteria:

1) Test:

Does the **Web browser** support commonly accepted browser technologies such as **plug-ins**, **APIs** and scripting languages?

Procedure:

Review the list of tested Web browsers and make sure they support plug-ins, APIs and scripting languages.

Example:

None.

BP1648

Host the Registration Web Service (RWS) registration portlet in the Node.

Rationale:

The process of registering a Node's **Component** service with the **Registration Web Service** (**RWS**) can be quite complicated. By providing access to the registration **portlet** the chances of obtaining a registration and of having valid data in the registration are greatly increased.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Accessibility - Operational / NCES Federated Search
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / NCES Federated Search

Evaluation Criteria:

1) Test:

Is the **Registration Web Service** (**RWS**) registration **portlet** hosted on the local Node?

Procedure:

Look for the Registration Web Service (RWS) registration portlet implementation.

Example:

None.

BP1649

Specifically include provisions for incremental implementation of the CES services.

Rationale:

The states of the individual services that comprise the **CES** are at different level of maturity. Consequently, an incremental approach allows Node development to continue in parallel with the CES functionality.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues

Evaluation Criteria:

1) Test:

Is there an incremental development approach?

Procedure:

Review the Node's schedule for incremental development.

Example:

None.

BP1650

Specifically include provisions for incremental implementation of the hosting Node's CES services for Node Components.

Rationale:

The states of the individual services that comprise the **CES** are at different levels of maturity. Consequently, an incremental approach allows **Component** development to continue in parallel with the Node and CES functionality.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues

Evaluation Criteria:

1) Test:

Is there an incremental development approach?

Procedure:

Review the schedule for Components for incremental development.

Example:

None.

BP1651

Ensure Node Components have access to Core Enterprise Services.

Rationale:

The burden of aligning to standard **CES** functionality and providing the functionality uniformly rests on the **Node** infrastructure, rather than the **components** within the Node. This isolates the components from the CES complexity and enhances portability and interoperability of the components. The access to CES may come from either from the standardized local Node infrastructure or through **Global Information Grid** (**GIG**) infrastructure.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Network Connectivity
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES and Intermittent Availability
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES and Intermittent Availability
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES and Intermittent Availability
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / CES and Intermittent Availability

Evaluation Criteria:

1) Test:

Do any **component** systems, applications or services implement any of the server side **CES Global Information Grid** (**GIG**) **Key Interface Profiles** (**KIPs**)?

Procedure:

Review the component systems, applications or services code for implementations of the server side CES Global Information Grid (GIG) Key Interface Profiles (KIPs).

Example:

None.

BP1653

Do not build dedicated Node guard products.

Rationale:

Current national policy dictates that a high-assurance guard or similar technology must be used whenever connecting networked security domains (i.e., ***SECRET US*** to ***SECRET REL*** or ***SIPRNET*** to ***NIPRNET***). Every single instantiation of every single guard needs to be approved by the appropriate authority. There are no type accreditations. Adding a new guard technique will likely incur additional scrutiny of the program as well as significant technical and schedule risks. The preferred approach is to use an already approved guard to mitigate risk.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Trusted Guards
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Trusted Guards

BP1654

Do not build dedicated Component guard products.

Rationale:

Current national policy dictates that a high-assurance guard or similar technology must be used whenever connecting networked security domains (i.e., ***SECRET US*** to ***SECRET REL*** or ***SIPRNET*** TO ***NIPRNET***). Every single instantiation of every single guard needs to be approved by the appropriate authority. There are no type accreditations. Adding a new guard technique will likely incur additional scrutiny of the program as well as significant and technical and schedule risks. The preferred approach is to use an already approved guard to mitigate risk.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Trusted Guards
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Trusted Guards

BP1661

Engage with the Net-Centric Enterprise Services (NCES) program office to explore approaches for mobile use of the Core Enterprise Services (CES) services in mobile Nodes that rely on Transmission Control Protocol/Internet Protocol (TCP/IP) for inter-node communication.

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Joint Net-Centric Capabilities

BP1663

Design a Domain Name System (DNS) in coordination with the appropriate governing Internet Protocol Version 6 (IPv6) Transformation Office.

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 4: Node Guidance / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: IPv6

BP1668

Acquire and configure approved guard products with the help of the Government program offices that acquire such guards.

Rationale:

Leveraging the certification documentation, expertise and existing relationships with the **National Security Agency** (**NSA**) and other pertinent authorities will streamline acquisition of approved guards.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Trusted Guards
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Trusted Guards

BP1669

Select XML-capable trusted guards.

Rationale:

As **XML** is a fundamental transfer format for data in interoperable net-centric environments, **trusted guards** should be capable of transferring XML data to facilitate cross-domain interoperability.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Trusted Guards
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Trusted Guards
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Cross-Security-Domains Exchange

BP1670

Plan for Black Core implementation in the local Node.

Rationale:

Node designers and operations personnel must implement and deploy encryptors or encryption support at enclave borders that can interoperate with partner Nodes and enclaves. See also the Black Core [P1152] and Confidentiality [P1340] perspectives.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Concurrent Transport of Information Flows
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Confidentiality / Black Core
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Confidentiality / Black Core

BP1671

Consider Black Core transition whenever there is a significant Node network design or configuration decision to make in an effort to avoid costly downstream changes caused by Black Core transition.

Rationale:

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Concurrent Transport of Information Flows
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Confidentiality / Black Core
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Confidentiality / Black Core

BP1672

Be prepared to integrate fully with the Information Assurance (IA) infrastructure.

Rationale:

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Net-Centric IA Posture and Continuity of Operations
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Client Platform

BP1673

Be prepared to integrate fully with the Enterprise Management Services (EMS) infrastructure.

Rationale:

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Client Platform

BP1674

Configure in accordance with the applicable Security Technical Implementation Guides (STIGs).

Rationale:

Configuring Web browsers using applicable STIGs reduces security vulnerabilities. The STIGs related to Web browsers include Web Server STIG, Desktop Applications STIG, and Windows 2003/XP/2000 Addendum STIG.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Client Platform / Browser
NESI / Part 4: Node Guidance / User Environment / Browser

BP1675

In the Node's Web infrastructure, support the technologies and standards used by the CES services under development as well as any technologies and standards used for Community of Interest (COI) services.

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Infrastructure

BP1677

Consider using Web proxy servers and load balancers.

Rationale:

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Infrastructure

BP1679

Implement a Node that uses Active Directory (AD) in accordance with the recommendations of the DoD Active Directory Interoperability Working Group (DADIWG).

Rationale:

The purpose of DoD Active Directory Interoperability Working Group (DADIWG) specification is to define a DoD naming convention for users with the objective of promoting more efficient data synchronization to support email communications for the Joint environment and to prepare **Active Directory** to support more sophisticated DoD-wide directory and discovery services. This specification develops consistent naming conventions - naming formats, content, and supporting data values, for a baseline set of attributes for Active Directory User Objects.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Domain Directories

BP1680

Instrument component services that a Node exposes to the Global Information Grid (GIG) to collect performance metrics.

Rationale:

In a dynamic environment, where services and information exchange partners may be dynamic, metrics can be a key factor in the selection of services. Performance metrics that are advertised externally and frequently updated allow potential service users the ability to select an implementation that meets their performance requirements, such as a measurement of reliability.

Standards for metrics are expected to be defined in the Net-Centric Implementation Directives (NCID) S500 document that is not yet available. Some draft metrics that may be appropriate for web services are given in the following table:

SLA Metric	***Metric Description***
Availability	How often is the service available for consumption?
Accessibility	How capable is the service of serving a client request now?
Performance	How long does it take for the service to respond?
Compliance	How fully does the service comply with stated standards?
Security	How safe and secure is it to interact with this service?
Energy Efficiency	How energy-efficient is this service for mobile applications?
Reliability	How often does the service fail to maintain its overall service quality?

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Instrumentation for Metrics

BP1681

Make metrics for component services visible and accessible as part of the service registration and update the metrics periodically.

Rationale:

Metrics are normally also needed to ensure performance is provided according to more traditional **Service Level Agreements** (**SLAs**) and for operations management.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Instrumentation for Metrics
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Joint Net-Centric Capabilities

BP1683

Coordinate the Node schedule with the schedules of the Core Enterprise Service (CES) providers.

Rationale:

An unavoidable consequence of the Node architecture is that Core Enterprise Services (CES) are evolving in parallel with the development of the Nodes themselves. If the schedule for a Node is not coordinated with those of the CES providers, newly deployed CES capabilities may not support Node capabilities under development.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues

Evaluation Criteria:

1) Test:

Is there a Node roadmap that maps to the **Core Enterprise Services** (**CES**) schedules?

Procedure:

Look for a document that cross-references the Centric Enterprise Service schedules of capabilities to the Node's schedule.

Example:

None.

BP1684

Coordinate the Node schedule with the Component schedules.

Rationale:

All schedules are subject to slippage or modifications due to changing priorities. Changes in the development schedule for a Node"s capabilities can have an impact on the schedules of Node **components**.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues

BP1685

For Key Interface Profile (KIP) specifications that are not available or insufficiently mature, implement a "best effort" by following the published intent of functionality and monitor or participate in the relevant specification development body.

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Key Interface Profile (KIP)
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Key Interface Profile (KIP)
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Key Interface Profile (KIP)
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / Net-Ready Key Performance Parameter (NR-KPP) / Key Interface Profile (KIP)

BP1686

Align Node interfaces to Components for directory services with the guidance being provided by the Joint Directory Services Working Group (JDSWG) and sub-working groups, including such guidance as naming conventions, federation, and synchronization.

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Joint Net-Centric Capabilities

BP1687

Follow Active Directory naming conventions defined in the *Active Directory User Object Attributes Specification* as required by the DoD CIO memorandum titled *Microsoft Active Directory (AD) Services*.

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / NCES Directory Services
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / NCES Directory Services

BP1688

For Services Management, use an interim solution based on standardized Simple Network Management Protocol (SNMP) agents or other locally provided instrumentation and external monitoring tools.

Rationale:

An interim solution, until such time an enterprise instrumentation capability is available, will provide potential service consumers with real world historical performance metrics as well ensure support for negotiated **service level agreements** (**SLAs**). Example standards for performance instrumentation that enable enterprise-wide management include the Simple Network Management Protocol (SNMP, especially the Remote Network Monitoring or RMON specification), and Distributed Management Task Force (DMTF) standards.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Enterprise Management
NESI / Part 4: Node Guidance / Security and Management / Enterprise Management

BP1690

Use Node implemented Service Discovery (SD) for high availability.

Rationale:

One of the main reasons to develop a local Node **Service Discovery** (**SD**) Service is to support high availability.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Visibility - Registered / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Visibility - Discoverable / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Registered / Service Enablers / Service Discovery
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 4: Node Guidance / Services / Service Enablers / Service Discovery

BP1691

Use Node implemented Service Discovery (SD) to meet compartmentalization needs.

Rationale:

For pilot implementations that are not reachable, such as might be the case in a higher classified environment, the Nodes should coordinate among themselves and DISA to provide pilot and full service implementations that are reachable.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Visibility - Registered / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Visibility - Discoverable / Service Enablers / Service Discovery
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Accessibility - Registered / Service Enablers / Service Discovery
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Service Discovery
NESI / Part 4: Node Guidance / Services / Service Enablers / Service Discovery
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Cross-Security-Domains Exchange

BP1692

Determine which Collaboration Service vendor offering to employ in a disadvantaged environment or separate network.

Rationale:

Monitor progress on fielding the NCES Collaboration Service. Performance or administration reasons may dictate hosting a collaboration solution at the Node.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Collaboration Services
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Collaboration Services
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Collaboration Services
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Collaboration Services

BP1693

Make sure that collaboration products used to satisfy urgent requirements are from the JTIC list.

Rationale:

See http://jitc.fhu.disa.mil/washops/jtcd/dcts/status.html and, for products certified for use on SIPRNET, http://jitc.fhu.disa.mil/washops/jtcd/dcts/projects.html), until the **Net-Centric Enterprise Services** (**NCES**) Collaboration Service is available.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Collaboration Services
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Collaboration Services
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Collaboration Services
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Collaboration Services

BP1695

Designate a Core Enterprise Services (CES) liaison to monitor the availability of services.

Rationale:

The **CES** liaison is an important role for keeping the Node and **component** engineering processes synchronized with CES providers such as **Net-Centric Enterprise Services** (**NCES**).

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues

BP1697

Make the parallel development of Core Enterprise Services (CES) outside the control of the Node a part of the Node's risk management activities.

Rationale:

Since the development of the **CES** is external to the development of the Node, there is an interdependency between the Node and the CES. The Node needs to consider this as an increase in the risk to the Node development. This risk needs to be communicated back to the CES management and development teams.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues

BP1698

Plan for the event that Component services within a Node cannot be invoked across security domains.

Rationale:

Until such approaches are prototyped and explored more fully, Nodes should anticipate that services will not be capable of cross-domain invocation.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Overarching CES Issues / Cross-Domain Interoperation
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Cross-Security-Domains Exchange

BP1699

Configure routers in accordance with the Network Security Technical Implementation Guide (STIG).

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 4: Node Guidance / Node Transport / Network Layer / IP Routing and Routers

BP1700

Configure routers in accordance with Enclave Security Technical Implementation Guide (STIG).

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / IP Routing and Routers
NESI / Part 4: Node Guidance / Node Transport / Network Layer / IP Routing and Routers

BP1701

Configure Components for Information Assurance (IA) in accordance with the Network Security Technical Implementation Guide (STIG).

Rationale:

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Information Assurance/Security / Design Tenet: Net-Centric IA Posture and Continuity of Operations
NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Network Information Assurance
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Network Information Assurance
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Network Information Assurance
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Network Information Assurance

BP1702

Do not place services and information intended to be broadly accessible to other nodes behind a Virtual Private Network (VPN).

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Subnets and Overlay Networks / Virtual Private Networks (VPN)
NESI / Part 4: Node Guidance / Node Transport / Subnets and Overlay Networks / Virtual Private Networks (VPN)

BP1704

Consult the applicable Security Technical Implementation Guidance (STIG) documents as a fundamental part of design activities, and monitor the STIGs periodically for updates.

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport
NESI / Part 4: Node Guidance / Node Transport

BP1705

Design Domain Name System (DNS) infrastructure in accordance with appropriate governing Internet Protocol Version 6 (IPv6) Transition Office requirements.

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 4: Node Guidance / Node Transport / Network Services / Domain Name System (DNS)
NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: IPv6

BP1706

Design node networks, including the selection of Components and configuration, to support multicasting even if not currently used.

Rationale:

The use of multicasting is growing within the DoD and multicast capability is being actively engineered into the **Global Information Grid** (**GIG**).

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Subnets and Overlay Networks / Broadcast, Multicast, and Anycast
NESI / Part 4: Node Guidance / Node Transport / Subnets and Overlay Networks / Broadcast, Multicast, and Anycast

BP1707

Configure and locate elements of the Node Web infrastructure in accordance with the Web Server Security Technical Implementation Guide (STIG).

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Integrity / Computing Infrastructure Integrity
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Integrity / Computing Infrastructure Integrity
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Infrastructure

BP1708

Configure and locate elements of the Node Web infrastructure in accordance with the Desktop Applications Security Technical Implementation Guide (STIG).

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Integrity / Computing Infrastructure Integrity
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Integrity / Computing Infrastructure Integrity
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Infrastructure

BP1709

Configure and locate elements of the Node Web infrastructure in accordance with the Network Security Technical Implementation Guide (STIG).

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Integrity / Computing Infrastructure Integrity
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Integrity / Computing Infrastructure Integrity
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Infrastructure

BP1710

Support appropriate and widely accepted standards for Web portals provided by the Node.

Rationale:

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Web Infrastructure

BP1711

Use the CES Mediation Service, or a locally hosted copy, when XML document translation between schemas is a necessity.

Rationale:

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Utility Services
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Utility Services
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Utility Services
NESI / Part 4: Node Guidance / Services / Utility Services

BP1712

Register developed mappings in the DoD Metadata Registry.

Rationale:

Registration of transformation, mediation and other utility service mappings in the **DoD Metadata Registry** makes them available to the wider DoD community and eases coordination with the services producing information which the utility services transform or mediate.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Joint Technical Architecture [now DISR]
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Utility Services
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Utility Services
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Utility Services
NESI / Part 4: Node Guidance / Services / Utility Services

BP1865

Provide sufficient program, project, or initiative metadata descriptions and automated support to enable mediation and translation of the data between interfaces.

Rationale:

Information exchanges should support known and unanticipated users. The program or project should initiate sufficient metadata descriptions and provide automated support to enable mediation and translation of data between interfaces. All of the data that can and should be shared externally beyond the programmatic bounds of your program should be defined well enough in metadata descriptions and translation of the data between interfaces should be automated.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Data / Design Tenet: Make Data Visible
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Visibility / Design Tenet: Make Data Visible
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Accessibility - Operational / NCES Federated Search
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / NCES Federated Search
NESI / Part 4: Node Guidance / General Responsibilities / Coordination of Node and Enterprise Services
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Visibility / Net-Centric Data Strategy (NCDS)
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Accessibility - Policy / Net-Centric Data Strategy (NCDS)
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Accessibility - Operational / Net-Centric Data Strategy (NCDS)
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Understandability / Net-Centric Data Strategy (NCDS)
NESI / Part 3: Migration Guidance / Net-Centric Data Strategy (NCDS)
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Net-Centric Information Engineering
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Net-Centric Information Engineering
NESI / Part 4: Node Guidance / General Responsibilities / Net-Centric Information Engineering
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Data / Metadata
NESI / Part 2: Traceability / DISR Service Areas / Data Management Services / Data / Metadata
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Understandability / Metadata
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Understandability - Registered / Metadata
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Service Exposure Verification Tracking Sheet / Service Understandability - COI Data Models / Metadata
NESI / Part 5: Developer Guidance / Data / Metadata
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Data / Design Tenet: Provide Data Management
NESI / Part 2: Traceability / Exposure Verification Tracking Sheets / Data Exposure Verification Tracking Sheet / Data Visibility / Design Tenet: Provide Data Management
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Data / Design Tenet: Make Data Interoperable

Evaluation Criteria:

1) Test:

Evaluation of interfaces and applicable mediation/translations to access that the program, project, or initiative has sufficient metadata descriptions and automated support to enable mediation and translation of the data between interfaces. Data is XML wrapped for exchange and configured to support standard transactions with headers, trailers and bodies.

Procedure:

Evaluate the degree to which data is XML wrapped for exchange and configured to support standard transactions with headers, trailers and bodies.

Evaluation of the DoD Metadata Registry entries to assess sufficient metadata descriptions and automated support the enables mediation and translation of the data between interfaces.

Example:

XML wrapped data are intend for exchange, that is configured in terms of standard transactions with headers, trailers and bodies.

BP1866

Coordinate with end users to develop interoperable materiel in support of high-value mission capability.

Rationale:

System providers acquire the materiel portion of mission capabilities that include all aspects of DOTMLP-F. An assessment by the community regarding the value of information or services provides useful direction in support of managing a mission area's portfolio of services. User feedback mechanisms provide a means of capturing and reporting user satisfaction and give portfolio managers decision-making information to steer investments, developments, and improvements. As service consumers gain access to information more quickly in the operational environment, command structures will inevitably change the manner in which IT investments are made. Service and information providers in a mission area should work together to define the processes for using the user feedback for service and information improvements because these processes are specific to a portfolio of capabilities in the Enterprise.

Referenced By:

NESI / Part 3: Migration Guidance / Migration Patterns / SOA-Enabled Migration Starting Point
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Net-Centric Information Engineering
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Net-Centric Information Engineering
NESI / Part 4: Node Guidance / General Responsibilities / Net-Centric Information Engineering
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Transport / Design Tenet: Joint Net-Centric Capabilities
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Data / Design Tenet: Make Data Interoperable

Evaluation Criteria:

1) Test:

Processes exist that allow a consumer to

1. request changes in the format (syntax or semantic) of the visible data asset;
2. report a problem with a data asset;
3. request additional data from the data provider

Procedure:

Evaluation of the process a consumer would follow to

1. request changes in the format (syntax or semantic) of the visible data asset;
2. report a problem with a data asset;
3. request additional data from the data provider.

Example:

An end-to-end output management strategy, across multiple business sites and/or the enterprise.

A distributed and extensible database which make information accessible to authorized users across the enterprise.

BP1867

Use metrics to track responsiveness to user information sharing needs.

Rationale:

Information sharing metrics are defined to measure and track implementation of the net-centric approaches. Measurement techniques should be developed to ensure that metrics are captured in a useful and consistent manner. Metrics should be tagged with **DDMS**-compliant metadata and provided to the NCE to promote awareness of data management successes and areas requiring improvement.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Instrumentation for Metrics
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Data / Design Tenet: Be Responsive to User Needs

Evaluation Criteria:

1) Test:

Does the program, project or initiative have metrics for determining responsiveness to user needs?

Procedure:

Evaluate the metrics being used to determine responsiveness to user data needs. If YES, describe; If NO, explain and identify a time frame for when the program, project, or initiative will have metrics for determining responsiveness to user needs; or specify NOT APPLICABLE and explain.

Example:

Examples of data metrics include percentage of Web-enabled components, progress toward service-enabling identified key functional components, and percentage of tagged community data.

BP1868

Incorporate mechanisms to enhance Computing Infrastructure (CI) availability.

Rationale:

Computing Infrastructure (CI) must be survivable, resilient, redundant, and reliable in the presence of attacks, failures, accidents, and natural or man-made disasters. A robust CI must incorporate survivability, resiliency, redundancy, and reliability to ensure operational availability in support of information sharing in DoD, as well as externally with federal agencies, state and local governments, allies, and coalition partners. In the context of the CI, the measure of reliability is included as a critical element in ensuring high mean time between failures (MTBF).

Survivable: Survivability ensures that CI systems, subsystems, equipment, processes, procedures, or CI-related doctrine, organization, training, materiel, leadership, personnel, facilities (DOTMLPF) continue to fulfill critical mission requirements in the presence of attacks, failures, accidents, and natural or man-made disasters.

Resilient: Incorporation of resiliency into CI ensures the ability to automatically recover from, or adjust to, attacks, failures, or accidents. Fault tolerance is a key example of resilience that measures the ability to respond gracefully to an unexpected CI system, subsystem, process, or procedure failure.

Redundant: Incorporation of automatic redundancy into the CI ensures that alternative devices are available to perform the required system functionality if a primary device fails. Redundancy also ensures that system data remains accessible and corruption free when CI components fail.

Reliable: Reliable OS platforms, other software infrastructure, and hardware components are critical to ensuring that operators can depend on their ability to support system functions and applications. Bandwidth conservation mechanisms minimize latency and jitter, as well as the instability that comes from running processors and networks with high loads. Processing efficiency mechanisms, such as efficient software implementation techniques, allow applications to meet performance and latency requirements. Typically, reliability is measured in mean time between user failures (MTBUF). MTBF of CI components is one factor affecting the overall system MTBF.

A Continuity of Operations Plan (COOP) and disaster recovery planning are also key to ensuring a robust CI. The DoD Dictionary of Military Terms defines COOP as "the degree or state of being continuous in the conduct of functions, tasks, or duties necessary to accomplish a military action or mission in carrying out the national military strategy." It includes the functions and duties of the commander, as well as the supporting functions and duties performed by the staff and others acting under the authority and direction of the commander.

Referenced By:

NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Availability
NESI / Part 2: Traceability / ASD(NII): Net-Centric Guidance / Services / Design Tenet: Enterprise Service Management
NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Policies and Processes for Implementing Software Security
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Policies and Processes for Implementing Software Security
NESI / Part 5: Developer Guidance / Software Security / Policies and Processes for Implementing Software Security
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Host Information Assurance
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Integrity / Computing Infrastructure Integrity
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Integrity / Computing Infrastructure Integrity
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Enterprise Security / Integrity
NESI / Part 4: Node Guidance / Security and Management / Enterprise Security / Integrity

Evaluation Criteria:

1) Test:

Does the program or initiative have a Continuity of Operations Plan (COOP) plan?

Procedure:

Verify existence of COOP.

Example:

Continuity of Operations Plans and Disaster Recovery Plans that include preparatory measures, response actions, and restoration activities planned or taken to ensure continuation of critical functions to maintain effectiveness, readiness, and survivability.

Technologies that allow, self-correcting mechanisms to be implemented (e.g., automatic recovery without manual intervention).

Clustering of servers, incorporation of relative addressing schemata (e.g., **DNS**), site mirroring, and provisioning of geographically distributed CI functionality are examples of fail-over implementations.

BP1904

Use Open Virtualization Format (OVF) for all virtual machines

Rationale:

OVF is an industry-supported standard for describing virtual machines. Using OVF allows virtual machines to execute within any environment that supports the format. This reduces vendor dependencies.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Virtual Machines

Evaluation Criteria:

1) Test:

Does the VM descriptor conform to the requirements of OVF?

Procedure:

Look for virtual machines instances and ensure that each instance is described by an OVF compliant descriptor.

Example:

BP1905

Digitally sign all Open Virtualization Format (OVF) virtual machines.

Rationale:

By digitally signing the virtual machine, systems that run the virtual machine will be able to detect any changes that may affect security or proper execution of the virtual machine. Additionally, signing the virtual machine with a **PKI** certificate (such as a DoD certificate) verifies the virtual machine came from a trusted source.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Virtual Machines

Evaluation Criteria:

1) Test:

Do all OVF virtual machine descriptors contain a valid digital signature?

Procedure:

Check the digital signature in each .ovf file to verify that it was signed using a valid trusted PKI certificate.

Example:

BP1906

Only run Open Virtualization Format (OVF) virtual machines with a valid digital signature from a trusted source.

Rationale:

Only executing virtual machines from trusted sources containing a valid signature ensures that the virtual machine is trusted and has not been modified.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Virtual Machines

Evaluation Criteria:

1) Test:

Has the virtual machine's integrity been verified prior to execution?

Procedure:

Confirm the digital signature and pedigree of any **PKI** certificates prior to execution of the virtual machine.

Example:

BP1907

Use Internet Relay Chat (IRC) bots to provide network based IRC services.

Rationale:

Internet Relay Chat (IRC) bots are stand-alone, independent programs, that connect to IRC Servers as clients. IRC bots commonly provide services in an IRC system; for example, keeping chat channels open, protecting chat channels, and recording messages for users who are currently off-line.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Text Conferencing
NESI / Part 2: Traceability / DISR Service Areas / Data Interchange Services / Services / Core Enterprise Services (CES) / Collaboration Services / Text Conferencing
NESI / Part 2: Traceability / DISR Service Areas / Distributed Computing Services / Services / Core Enterprise Services (CES) / Collaboration Services / Text Conferencing
NESI / Part 2: Traceability / DISR Service Areas / Environment Management / Services / Core Enterprise Services (CES) / Collaboration Services / Text Conferencing
NESI / Part 4: Node Guidance / Services / Core Enterprise Services (CES) / Collaboration Services / Text Conferencing

BP1915

Model time-critical operations.

Rationale:

Modeling will help understand time-critical requirements and constraints and provider capabilities for fulfilling them whether writing a new software system, making substantive changes to an existing system, or migrating an existing software system to a different language. Modeling is often a cost effective approach to explore design criteria as opposed to prototypes and experimentation. In addition to analysis, models may be able to generate code automatically. Modeling and Analysis of Real-Time and Embedded Systems (MARTE) Uniform Modeling Language 2 (UML2) profile is an example standard applicable to time-critical modeling.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Time-Critical Operations

Evaluation Criteria:

1) Test:

Do models exist for time-critical operations?

Procedure:

Verify that models exists (either in the system proper or in the design documentation) for time-critical operations.

Example:

Time-critical modeling represented as a MARTE UML profile.

BP1916

Resolve contention among resources in a consistent manner.

Rationale:

Inconsistent approaches to resolving contention for sequentially shared resources may result in potentially disastrous timing anomalies. This is especially true for time-critical systems. Often multiple operations contend for one or more sequentially shared software and hardware resources such as locks, processors, networks, secondary storage, etc. The order of resolving this contention (i.e., the order in which the contending operations are granted access to a resource) impacts properties of the system, such as timeliness, throughput, power consumption, security, etc. Therefore, choose contention-resolving methods that are not just optimal for a desired property in a single resource but for all resources that contribute to that property across the system. Otherwise, undesirable system-wide anomalies may occur, ranging from hard-to-isolate intermittent faults through unnecessary processing up to system hangs.

For example, if system timeliness is the system property of most interest, using Earliest Deadline First (EDF) technique for scheduling the processor before scheduling the contending operations may be appropriate. In that case, dequeue other operations for access to a mutual exclusion (mutex) synchronizer by EDF (not, as is common practice, first-come-first-served). Likewise with any other critical path or time-sensitive resources, access to which determines system timeliness. Otherwise, resource access eligibility inversion will occur (e.g., deadlines may be missed or ignored due to inconsistencies in resolving access to resources).

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Time-Critical Operations

Evaluation Criteria:

1) Test:

For each key system performance property identified (i.e. timeliness, throughput, power consumption), are critical resource types and contention-resolution methods identified?

Procedure:

Identify critical resource types and contention-resolution methods for each identified key system performance property.

Example:

None.

2) Test:

Has analysis of the combinations of resource contention-resolution methods shown those combinations to maintain the associated key system performance properties consistently ?

Procedure:

Verify documentation exists which describes the analysis of the combinations of resource contention-resolution methods used and shows those combinations consistently maintain the associated key system performance properties?

Example:

None.

BP1917

Use cyclic executive scheduling if a real-time system requires that all operations have a priori start and completion times.

Rationale:

Rate monotonic analysis will not satisfy a requirement to meet a priori deadlines. The best approach to fulfilling this requirement is called a cyclic executive; a single master thread executes each operation in a predefined sequence, beginning and ending each operation at a predefined time. Cyclic executives can be very tricky to get right or to change, since changing one operation's timing can affect the timing of many other operations. Cyclic executives also can eliminate the need to multiple asynchronously concurrent operations, which can be notoriously difficult to use in larger scale systems.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Time-Critical Operations

BP1918

Use deadline-based algorithms for scheduling operations with deadlines.

Rationale:

Well known, tested, and provable algorithms such as Earliest Deadline First (EDF) will meet all deadlines if that is possible, given its presumptions. One of those presumptions is that the system load does not exceed 100%. In soft real-time systems, EDF minimizes the maximum operation latency. No extant COTS real-time operating system provides EDF scheduling, but it can be implemented at the application level.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Time-Critical Operations

BP1919

Design systems presuming degraded environments are the normal case.

Rationale:

A design that presumes faults and failures are an unusual occurrence, and thus treats them as special cases, will not be able to detect and recover from faults and failures as effectively and in as timely a manner as if faults and failures are expected to be the normal case.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Time-Critical Operations

BP1920

Design systems to have timeliness as a core capability.

Rationale:

Adding non-functional capabilities, such as timeliness, fault management, and security, to a designed or implemented system usually is not cost-effective, if possible to do at all. Those capabilities are integral to the operation and thus significantly affect the design and implementation from the beginning of the initial modeling.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Time-Critical Operations

BP1921

Design systems to have fault management as a core capability.

Rationale:

Adding non-functional capabilities, such as timeliness, fault management, and security, to a designed or implemented system usually is not cost-effective, if possible to do at all. Those capabilities are integral to the operation and thus significantly affect the design and implementation from the beginning of the initial modeling.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Time-Critical Operations

BP1922

Design systems to have security as a core capability.

Rationale:

Adding non-functional capabilities, such as timeliness, fault management, and security, to a designed or implemented system usually is not cost-effective, if possible to do at all. Those capabilities are integral to the operation and thus significantly affect the design and implementation from the beginning of the initial modeling.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Operating System Services / Software Security / Policies and Processes for Implementing Software Security / Secure Coding and Implementation Practices / Practice Defense in Depth
NESI / Part 2: Traceability / DISR Service Areas / Security Services / Software Security / Policies and Processes for Implementing Software Security / Secure Coding and Implementation Practices / Practice Defense in Depth
NESI / Part 5: Developer Guidance / Software Security / Policies and Processes for Implementing Software Security / Secure Coding and Implementation Practices / Practice Defense in Depth
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Time-Critical Operations

BP1923

Employ an operating system that supports simultaneously IPv4 and IPv6.

Rationale:

In order to support applications that require both **IPv4** and **IPv6** communications, the operating system must also support both IPv4 and IPv6 simultaneously.

Referenced By:

NESI / Part 2: Traceability / DISR Service Areas / Communications Applications / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition
NESI / Part 4: Node Guidance / Node Transport / Network Layer / Internet Protocol (IP) / IPv4 to IPv6 Transition

Evaluation Criteria:

1) Test:

Does the operating system support dual stack IPv4 and IPv6?

Procedure:

Check the operating system's IP configuration for dual IPv4 and IPv6 configurations.

Example:

None

BP1933

Control access to firmware with strong passwords.

Rationale:

Protecting firmware access helps limit firmware (including the basic input/output system or BIOS) configuration changes to those with authority to do so. Using strong passwords that are sufficiently random, long, and contain special characters (when allowed by the system) helps prevent unauthorized access by guessing or brute force trial of passwords.

Referenced By:

NESI / Part 4: Node Guidance / User Environment / Remote KVM Switch Connectivity
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Remote Management

BP1934

Disable Preboot Execution Environment (PXE) capabilities when not required.

Rationale:

It is difficult to detect unauthorized PXE boot servers or unauthorized PXE-enabled client computers on a network. This provides the opportunity that an unauthorized system on the network can boot images provided by a PXE boot server and also allows for someone to impersonate a PXE boot server and provide unauthorized boot images to PXE-enable client computers on the network. Disabling PXE capabilities (for example disabling PXI within a basic input/output system or BIOS configuration) when not required helps mitigate this risk.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Remote Management

BP1935

Disable Wake on LAN (WOL) capabilities when not required.

Rationale:

Most WOL implementations do not provide any form of authentication. This potentially allows unauthorized powering up of a system in a powered off state, possibly allowing a further attack against the system. Disabling WOL capabilities (for example, disabling WOL within a basic input/output system or BIOS configuration) when not required helps mitigate this risk.

Referenced By:

NESI / Part 4: Node Guidance / Node Computing Infrastructure / Remote Management

BP1936

Physically secure hardware components.

Rationale:

Physically securing hardware components (by preventing physical access to the hardware or implementing techniques to prevent tampering with the hardware) helps prevent unauthorized physical changes (including configuration changes) to the hardware as well as software contained within the hardware.

Referenced By:

NESI / Part 4: Node Guidance / User Environment / Remote KVM Switch Connectivity
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Remote Management

BP1937

Disable component remote management capabilities when secure remote management is not possible.

Rationale:

Many components configured with remote management capabilities enable remote management by default. This presents an opportunity to compromise the component remotely. Furthermore, in high risk environments, it may not be feasible or possible to manage a component remotely in a secure manner. Disabling remote management capabilities when not required, especially when remote management is not secure, helps mitigate this risk.

Referenced By:

NESI / Part 4: Node Guidance / User Environment / Remote KVM Switch Connectivity
NESI / Part 4: Node Guidance / Node Computing Infrastructure / Remote Management

BP1938

Disable Bluetooth functionality except when required.

Rationale:

Like any network capability, Bluetooth provides an opportunity to compromise a component. Many components have Bluetooth enabled by default. Disabling Bluetooth capabilities when those capabilities are not required minimizes the risk of compromise through Bluetooth.

Referenced By:

NESI / Part 4: Node Guidance / User Environment / Remote KVM Switch Connectivity

BP1939

Configure Bluetooth-enabled device pairs to use the strongest Bluetooth security mode supported by each device pair.

Rationale:

Newer versions of the Bluetooth specification provide additional security protections to prevent unauthorized access to Bluetooth communications. Using the strongest security mode that both paired devices support minimizes the risk of unauthorized data access or device control.

Referenced By:

NESI / Part 4: Node Guidance / User Environment / Remote KVM Switch Connectivity

BP1940

Use strong Bluetooth passwords.

Rationale:

Using strong passwords that are sufficiently random, long, and contain special characters (when allowed by the system) helps prevent unauthorized access by guessing or brute force trial of passwords.

Referenced By:

NESI / Part 4: Node Guidance / User Environment / Remote KVM Switch Connectivity

BP1941

Disable Bluetooth device discoverability except during required device pairing.

Rationale:

Disabling Bluetooth device discoverability except during required device pairing prevents visibility to other Bluetooth devices except when needed. This helps to minimize the risk associated with Bluetooth.

Referenced By:

NESI / Part 4: Node Guidance / User Environment / Remote KVM Switch Connectivity

Glossary

.NET Framework		The .NET Framework is an integral Windows component that supports building and running the next generation of applications and XML Web services. The .NET Framework has two main components: the common language runtime and the .NET Framework class library. (Source: MSDN ***.NET Framework Conceptual Overview***, http://msdn.microsoft.com/en-us/library/zw4w595w.aspx)
Access Control		Limiting access to information system resources only to authorized users, programs, processes, or other systems. (Source: *National Information Assurance (IA) Glossary*, CNSSI 4009, revised June 2006) ***Note:*** *See also the following:* • ***Access Control List*** *(**ACL**) [GL1889]* • ***Discretionary Access Control*** *(**DAC**) [GL1197]* • ***Role-Based Access Control*** *(**RBAC**) [GL1643]*
Access Control List	ACL	In computer security, ACL is a concept used to enforce privilege separation. It is a means of determining the appropriate access rights to a given object depending on certain aspects of the process that is making the request, principally the process's user identity. In networking, ACL refers to a list of ports and services that are available on a host, each with a list of hosts and/or networks permitted to use the service. Both individual servers as well as routers can have access lists. Access lists are used to control both inbound and outbound traffic, and in this context they are similar to firewalls. (Source: http://en.wikipedia.org/wiki/Access_control_list)
Active Directory	AD	An implementation of Lightweight Directory Access Protocol (LDAP) directory services by Microsoft for use in Windows environments; allows administrators to assign enterprise-wide policies, deploy programs to many computers, and apply critical updates to an entire organization. An Active Directory stores information and settings relating to an organization in a central, organized, accessible database. Active Directory networks can vary from a small installation with a few hundred objects, to a large installation with millions of objects. (Source: http://en.wikipedia.org/wiki/Active_Directory)
All Views	AV	The DoDAF All-Views (AV) products provide information pertinent to the entire architecture but do not represent a distinct view of the architecture. AV products set the scope and context of the architecture. The scope includes the subject area and timeframe for the architecture. The setting in which the architecture exists comprises the interrelated conditions that compose the context for the architecture. These conditions include doctrine; tactics, techniques, and procedures; relevant goals and vision statements; concepts of operations; scenarios; and

		environmental conditions. (Source: *DoDAF* v1.5 Volume 1: Definitions and Guidelines, 23 April 2007)

American Standard Code for Information Interchange	ASCII	ASCII is a character set and a character encoding based on the Roman alphabet as used in modern English. ASCII codes represent text in computers, in other communications equipment, and in control devices that work with text. Most often, nowadays, character encoding has an ASCII-like base.
		
		ASCII defines the following printable characters, presented here in numerical order of their ASCII value:
		
		`!"#$%'()*+,-./0123456789:; ?`
		`@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`
		`abcdefghijklmnopqrstuvwxyz{|}~(`
		
		(Source: http://en.wikipedia.org/wiki/ASCII)

Applet		A J2EE component that typically executes in a Web browser. Applets can also execute in a variety of other applications or devices that support the applet programming model. (Source: *J2EE 1.4 Glossary*, http://java.sun.com/j2ee/1.4/docs/glossary.html)
Application		An application is a software program that performs a specific function directly for a user, with or without requiring extraordinary authority or privileges such as system-level control and monitoring, administrative or "super user" rights, or root-level access. (Source: derived from Committee on National Security Systems Instruction 4009, *National Information Assurance Glossary* [R1339])
Application Programming Interface	API	A special type of interface that specifies the calling conventions with which one component may access the resources and services provided by another component. APIs are defined by sets of procedures or function-invocation specifications. An API is a special case of an interface.
Assistant Secretary of Defense for Networks and Information Integration	ASD (NII)	(Source: http://www.dod.mil/nii/)
Attribute		A distinct characteristic of an object. Real-world object attributes are often specified in terms of their physical traits, such as size, shape, weight, and color. Cyberspace object attributes might describe size, type of encoding, and network address. (Source: ***Web Services for Remote Portlets Specification***, ***Appendix A: Glossary***; http://www.oasis-open.org/committees/download.php/3343/oasis-200304-wsrp-specification-1.0.pdf)
Authentication		The process that verifies the identity of a user, device, or other entity in a computer system, usually as a prerequisite to allowing access to resources in a system. The Java servlet specification requires three types of authentication (basic, form-based, and mutual) and supports

		digest authentication. (Source: *J2EE 1.4 Glossary*, http://java.sun.com/j2ee/1.4/docs/glossary.html)
Authorization		The process by which access to a method or resource is determined. Authorization depends on the determination of whether the principal associated with a request through authentication is in a given security role. A security role is a logical grouping of users defined by the person who assembles the application. A deployer maps security roles to security identities. Security identities may be principals or groups in the operational environment. (Source: *J2EE 1.4 Glossary*, http://java.sun.com/j2ee/1.4/docs/glossary.html)
Browser		Short for **Web browser**, a software application used to locate and display Web pages. (Source: http://www.webopedia.com/TERM/b/browser.html)
Capability Development Document	CDD	Provides operational performance attributes, including supportability, for the acquisition community to design the proposed system. Includes key performance parameters (KPP) and other parameters that guide the development, demonstration, and testing of the current increment. Outlines the overall strategy for developing full capability. (Source: http://www.dau.mil/pubs/glossary/12th_Glossary_2005.pdf)
Capability Production Document	CPD	Addresses the production attributes and quantities specific to a single increment of an acquisition program. Supersedes threshold and objective performance values of the CDD. (Source: http://www.dau.mil/pubs/glossary/12th_Glossary_2005.pdf)
Certificate	CERT	A certificate which uses a digital signature to bind together a public key with an identity information such as the name of a person or an organization, their address, and so forth. The certificate can be used to verify that a public key belongs to an individual. (Source: http://en.wikipedia.org/wiki/Certificate_%28cryptography%29)
Certificate Authority	CA	A trusted organization which issues digital public key certificates for use by other parties. It is an example of a trusted third party. CAs are characteristic of many public key infrastructure (PKI) schemes. (Source: http://en.wikipedia.org/wiki/Certificate_authority)
Certificate Revocation List	CRL	A list of certificates (more accurately, their serial numbers) which have been revoked, are no longer valid, and should not be relied upon by any system user. (Source: http://en.wikipedia.org/wiki/Certificate_Revocation_List)
Chief Information Officer	CIO	Job title for a manager responsible for **Information Technology** (IT) within an organization; often reports to the chief executive officer or chief financial officer. For information on the Assistant Secretary of Defense for Networks and Information Integration (ASD/NII)/DoD CIO see DoDD 5144.1 of 2 May 2005. (Source: http://en.wikipedia.org/wiki/Chief_Information_Officer)

Cipher Text	CT	Data that has been **encrypted**. Cipher text is unreadable until it has been converted into Plain Text (PT) (decrypted) with a key. (Source: http://www.webopedia.com/TERM/C/cipher_text.html)
Client		A system entity that accesses a Web service. (Source: http://www.oasis-open.org/committees/download.php/3343/oasis-200304-wsrp-specification-1.0.pdf)
COI Service		See **Community of Interest Service**.
Collaboration		Portal members can communicate synchronously through chat or messaging, or asynchronously through threaded discussion, blogs, and email digests (forums).
Collaboration Management Office	CMO	DISA organization responsible for fielding, sustaining and managing the life cycle of the **Defense Collaboration Tool Suite** (**DCTS**).
Command and Control	C2	(DoD) The exercise of authority and direction by a properly designated commander over assigned and attached forces in the accomplishment of the mission. Command and control functions are performed through an arrangement of personnel, equipment, communications, facilities, and procedures employed by a commander in planning, directing, coordinating, and controlling forces and operations in the accomplishment of the mission. (Source: DoD, ***Department of Defense Dictionary of Military and Associated Terms***, JP 1-02, 12 April 2001 as amended through 17 October 2008)
Commercial Off-The-Shelf	COTS	A term for systems that are manufactured commercially, and may be tailored for specific uses. (Source: http://en.wikipedia.org/wiki/Commercial_off-the-shelf)
Common Access Card	CAC	A DoD-wide smart card used as the identification card for active duty Uniformed Services personnel (to include the Selected Reserve), DoD civilian employees, eligible contractor personnel, and eligible foreign nationals; the primary platform for the Public Key Infrastructure (PKI) authentication token used to access DoD computer networks and systems in the unclassified environment and, where authorized by governing security directives, the classified environment; and the principal card enabling physical access to buildings, facilities, installations, and controlled spaces as described in DoD Directive 8190.3, "Smart Card Technology," 31 August 2002. ***Note:*** *The Defense Manpower Data Center (DMDC) Common Access Card site (http://www.dmdc.osd.mil/smartcard) contains additional information, reports and developer support concerning the DoD CAC implementation.* (Source: DoD Instruction 8520.2, 1 April 2004, [R1206] Enclosure (2) Definitions, page 13)

Common Object Request Broker Architecture	CORBA	CORBA "wraps" code written in another language into a bundle containing additional information on the capabilities of the code inside, and explaining how to call it. The resulting wrapped objects can then be called from other programs (or CORBA objects) over the network. The CORBA specification defines APIs, communication protocol, and object/ service information models to enable heterogeneous applications written in various languages running on various platforms to interoperate. (Source: http://en.wikipedia.org/wiki/CORBA)
Community of Interest	COI	A COI is a collaborative group of users that must exchange information in pursuit of its shared goals, interests, missions, or business processes and therefore must have shared vocabulary for the information it exchanges. (Source: DoDD 8320.02, 2 December 2004, *Data Sharing in a Net-Centric Department of Defense*)
Community of Interest Service		A service that may be offered to the enterprise, but is owned and operated by a **Community of Interest** to provide or support a well-defined set of mission functions and associated information.
Component		One of the parts that make up a system. A component may be hardware or software and may be subdivided into other components. Note the terms ***module***, ***component***, and ***unit*** are often used interchangeably or defined to be sub-elements of one another in different ways depending on the context. The relationship of these terms is not yet standardized. (Source: IEEE Std 610.12-1990) ***Note:*** *See **system component** and **software component**.*
Computer Network Defense	CND	Defensive measures to protect and defend information, computers, and networks from disruption, denial, degradation, or destruction. (Source: DoD, Department of Defense Dictionary of Military and Associated Terms, JP 1-02, 12 April 2001 as amended through 17 October 2008)
Computer Network Defense Service Provider	CNDSP	Those organizations responsible for delivering protection, detection and response services to its users. CNDS providers must provide for the coordination service support of a CNDS/CA. CNDS is commonly provided by a Computer Emergency or Incident Response Team (CERT/ CIRT) and may be associated with a Network Operations (NetOps) and Security Center (NOSC). (Source: DoD Directive O-8530.1, ***Computer Network Defense (CND)***,[R1191] 8 January 2001, Enclosure 2 Definitions, p. 12)
Confidentiality		The property that data is not made available to unauthorized individuals, entities, or processes.
Content Discovery Service	CDS	**Net-Centric Enterprise Services** (**NCES**) service that provided a Federated Search capability.
Core Enterprise Services	CES	Core Enterprise Services (CES) are a small set of **services** provided by the Enterprise Information Environment Mission Area (EIEMA). Some of the CES services will be centrally provided on behalf of the DoD

		while others might involve local provisioning. For locally provisioned services, EIEMA provides guidance to ensure consistent implementation throughout the DoD. (Source: *DoD Net-Centric Services Strategy*, Section 3.1 [R1313])
Credentials		The information describing the security attributes of a principal. (Source: *J2EE 1.4 Glossary*, http://java.sun.com/j2ee/1.4/docs/glossary.html)
Data Distribution Service for Real-Time Systems	DDS	DDS is a recently-adopted OMG standard that is the first open international middleware standard directly addressing publish-subscribe communications for real-time and embedded systems. DDS introduces a virtual Global Data Space where applications can share information by simply reading and writing data-objects addressed by means of an application-defined name (Topic) and a key. DDS features fine and extensive control of QoS parameters, including reliability, bandwidth, delivery deadlines, and resource limits. DDS also supports the construction of local object models on top of the Global Data Space. (Source: OMG Data Distribution Portal, http://portals.omg.org/dds)
Defense Acquisition University	DAU	The mission of the DAU is to provide practitioner training, career management, and services to enable the DoD Acquisition, Technology and Logistics (AT&L) community to make smart business decisions and deliver timely and affordable capabilities to the warfighter. (Source: http://www.dau.mil/about-dau/docs/mission_vision.ppt)
Defense Collaboration Tool Suite	DCTS	A flexible, integrated set of applications providing interoperable, synchronous, and asynchronous collaboration capability to the Department of Defense Agencies, Combatant Commands, and Military Services.
Defense Information System Network	DISN	The Defense Information System Network (DISN) provides long-haul information transfer services for all Department of Defense activities. It is an information transfer utility designed to provide dedicated point-to-point, switched voice and data, imagery, and video teleconferencing services. (Source: *DoD Dictionary of Military Terms*, http://www.dtic.mil/doctrine/dod_dictionary/; accessed 2 November 2010)
Defense Information Systems Agency	DISA	Combat support agency responsible for planning, engineering, acquiring, fielding, and supporting global net-centric solutions to serve the needs of the President, Vice President, the Secretary of Defense, and other DoD Components, under all conditions of peace and war. (Source: http://www.disa.mil/main/about/missman.html)
Defense IT Standards Registry	DISR	The DoD IT Standards Registry (DISR) is an online repository (http://disronline.disa.mil) for a minimal set of primarily commercial IT standards formerly captured in the Joint Technical Architecture (JTA), Version 6.0. These standards are used as the "building codes" for all systems being procured in the Department of Defense. Use of these building codes facilitates interoperability among systems and integration of new systems into the Global Information Grid (GIG). In addition, the DISR provides the capability to build profiles of standards that programs will use to deliver net-centric capabilities. (Source: http://akss.dau.mil/dag/GuideBook/IG_c7.2.4.2.asp)

Department of Defense	DoD	The Department of Defense is America's oldest and largest government agency. The DoD mission is to provide the military forces needed to deter war and to protect the security of the United States. (Source: adapted from *DoD 101, An Introductory Overview of the Department of Defense*; http://www.defenselink.mil/pubs/dod101/; accessed 30 April 2009)
Digest		A cryptographic checksum of an octet stream.
Digital Signature		A value computed with a cryptographic algorithm and bound to data in such a way that intended recipients of the data can use the signature to verify that the data has not been altered and/or has originated from the signer of the message, providing message integrity and authentication. The signature can be computed and verified with symmetric key algorithms, where the same key is used for signing and verifying, or with asymmetric key algorithms, where different keys are used for signing and verifying (a private and public key pair are used).
Digital Signature Algorithm	DSA	The ***Digital Signature Algorithm (DSA)*** is a United States Federal Government standard for digital signatures. It was proposed by the **National Institute of Standards and Technology** (**NIST**) in August 1991 for use in their Digital Signature Standard (DSS), specified in **Federal Information Processing Standard** (**FIPS**) 186, adopted in 1993. A minor revision was issued in 1996 as FIPS 186-1. The standard was expanded further in 2000 as FIPS 186-2 and again in 2009 as FIPS 186-3. (Source: http://en.wikipedia.org/wiki/Digital_Signature_Algorithm; accessed 7 September 2010)
Directory Service		A directory service organizes computerized content and runs on a directory server computer. It is not to be confused with the directory itself, which is the database that holds the information about objects that are to be managed by the directory service. The directory service is the interface to the directory and provides access to the data that is contained in that directory. It acts as a central authority that can securely authenticate resources and manage identities and relationships between them. (Source: http://en.wikipedia.org/wiki/Directory_service)
Discretionary Access Control	DAC	Means of restricting access to objects based on the identity and need-to-know of users and/or groups to which the object belongs. Controls are discretionary in the sense that a subject with a certain access permission is capable of passing that permission (directly or indirectly) to any other subject. (Source: National Information Assurance (IA) Glossary, CNSSI 4009, revised June 2006)
Document Object Model	DOM	The Document Object Model is a platform- and language-neutral interface that will allow programs and scripts to access and update the content, structure and style of documents dynamically. (Source: http://www.w3.org/DOM/)
Document Type Definition	DTD	The XML document type declaration contains or points to markup declarations that provide a grammar for a class of documents. This grammar is known as a document type definition, or DTD. The document

		type declaration can point to an external subset (a special kind of external entity) containing markup declarations, or can contain the markup declarations directly in an internal subset, or can do both. The DTD for a document consists of both subsets taken together. (Source: W3C *Extensible Markup Language (XML) 1.0*, Fifth Edition [R1121])
DoD Architecture Framework	DoDAF	The DoD Architecture Framework (DoDAF) Version 2.0 is the prescribed framework for all Department architectures, and represents a substantial shift in approach. It places emphasis upon a disciplined process of defining the purpose, scope and information requirements of the architecture up-front, followed by collection of dat in accordance with a standard vocabulary. Data collected through the architectural process is delivered to the customer in either standard models or "Fit for Purpose" presentations. (Source DoD CIO promulgation memo, *The Department of Defense Architecture Framework (DoDAF) Version 2.0*, 28 May 2009; see the ASD(NII)/DoD CIO *Enterprise Architecture & Standards* site at http://cio-nii.defense.gov/policy/eas.shtml)
DoD Discovery Metadata Specification	DDMS	The DoD Discovery Metadata Specification (DDMS) defines discovery metadata elements for resources posted to community and organizational shared spaces. (Source: http://metadata.dod.mil/mdr/irs/DDMS/)
DoD Metadata Registry		As part of the overall **DoD Net-Centric Data Strategy**, the DoD CIO established the DoD Metadata Registry (http://metadata.dod.mil) and a related metadata registration process for the collection, storage and dissemination of structural metadata information resources (schemas, data elements, attributes, document type definitions, style-sheets, data structures, etc.). This Web-based repository is designed to also act as a clearinghouse through which industry and government coordination on metadata technology and related metadata issues can be advanced. As OASD's Executive Agent, **DISA** maintains and operates the ***DoD Metadata Registry and Clearinghouse*** under the direction and oversight of **OASD(NII)**. (Source: DoD Metadata Registry v6.0 Web site, https://metadata.dod.mil/mdr/about.htm)
DoD Net-Centric Data Strategy	NCDS	This Strategy lays the foundation for realizing the benefits of net-centricity by identifying data goals and approaches for achieving those goals. To realize the vision for net-centric data, two primary objectives must be emphasized: (1) increasing the data that is available to communities or the Enterprise and (2) ensuring that data is usable by both anticipated and unanticipated users and applications. (Source: *Department of Defense Net-Centric Data Strategy*, DoD CIO, 9 May 2003, http://www.defenselink.mil/cio-nii/docs/Net-Centric-Data-Strategy-2003-05-092.pdf)
Domain Name System	DNS	The Domain Name System stores information about hostnames and domain names in a type of distributed database on networks, such as the Internet. Of the many types of information that can be stored, most importantly it provides a physical location (IP address) for each domain name, and lists the mail exchange servers accepting email for each domain. The DNS provides a vital service on the Internet as it allows the transmission of technical information in a user-friendly way. While

		computers and network hardware work with IP addresses to perform tasks such as addressing and routing, humans generally find it easier to work with hostnames and domain names (such as ***www.example.com***) in URLs and email addresses. The DNS therefore mediates between the needs and preferences of humans and of software.
Dual Stacking		Incorporating both IPv4 and IPv6 support in routers and computers.
Dynamic Host Configuration Protocol	DHCP	A protocol for assigning dynamic **Internet Protocol** (IP) addresses to devices on a network; DHCP a device can have a different IP address every time it connects to the network. (Source: http://www.webopedia.com/TERM/D/DHCP.html)
Electronic Data Interchange Personnel Identifier	EDI-PI	A unique number assigned to each recipient of a **Common Access Card** (CAC), which is issued by the United States **Department of Defense** through the Defense Enrollment Eligibility Reporting System (DEERS). (Source: http://en.wikipedia.org/wiki/Electronic_Data_Interchange_Personal_Identifier)
Encryption		Encryption is the process of obscuring information to make it unreadable without special knowledge. While encryption has been used to protect communications for centuries, only organizations and individuals with an extraordinary need for secrecy have made use of it. In the mid-1970s, strong encryption emerged from the sole preserve of secretive government agencies into the public domain, and is now employed in protecting widely-used systems, such as Internet e-commerce, mobile telephone networks and bank automatic teller machines. (Source: http://en.wikipedia.org/wiki/Encryption)
Endpoint		The URL or location of the Web service on the internet.
Enterprise		An organization considered as an entity or system that includes interdependent resources (e.g., people, organizations, and technology) that must coordinate functions and share information in support of a common mission or a set of related missions. In the computer industry, the term is often used to describe any large organization that utilizes computers. An intranet, for example, is a good example of an enterprise computing system. (Source: http://www.webopedia.com/TERM/e/enterprise.html)
Enterprise Management Service	EMS	Enterprise Management Services (EMS) which are often used internal to a node, using a variety of **COTS** tools, which are fundamental to execution of **Service Level Agreements** (SLAs).
Enterprise Service		A service that provides capabilities to the enterprise. See also **Core Enterprise Service** and **Community of Interest Service**.
Enterprise Service Bus	ESB	An architectural style that provides distributed invocation, mediation, and end-to-end management and security of services and service interactions to support the larger architectural style known as Service Oriented Architecture (SOA)

		Note: See the Enterprise Service Bus [P1389] in Part 5 for additional information.
eXtensible Markup Language	XML	A markup language defines tags (markup) to identify the content, data, and text in XML documents. It differs from **HTML**, the markup language most often used to present information on the Internet. HTML has fixed tags that deal mainly with style or presentation. An XML document must undergo a transformation into a language with style tags under the control of a style sheet before it can be presented by a browser or other presentation mechanism. Two types of style sheets used with XML are CSS and XSL. Typically, XML is transformed into HTML for presentation. Although tags can be defined as needed in the generation of an XML document, you can use a **document type definition** (**DTD**) to define the elements allowed in a particular type of document. A document can be compared by using the rules in the DTD to determine its validity and to locate particular elements in the document. A Web services application's J2EE deployment descriptors are expressed in XML with schemas defining allowed elements. Programs for processing XML documents use SAX or **DOM** APIs. (Source: http://java.sun.com/j2ee/1.4/docs/glossary.html)
Facade Design Pattern		An object that provides a simplified interface to a larger body of code, such as a class library. (Source: http://en.wikipedia.org/wiki/Facade_pattern)
Federal Information Processing Standard	FIPS	Under the Information Technology Management Reform Act (Public Law 104-106), the Secretary of Commerce approves standards and guidelines that are developed by the **National Institute of Standards and Technology** (**NIST**) for Federal computer systems. These standards and guidelines are issued by NIST as Federal Information Processing Standards (FIPS) for use government-wide. NIST develops FIPS when there are compelling Federal government requirements such as for security and interoperability and there are no acceptable industry standards or solutions. (Source: http://www.itl.nist.gov/fipspubs/geninfo.htm)
Federated Search		Implementation of a computer program that allows users to access multiple data sources with a single query string located within a single interface. (Source: http://en.wikipedia.org/wiki/Federated_search)
File Transfer Protocol	FTP	FTP transfers files to and from a remote network. The protocol includes the ftp command (local machine) and the in.ftpd daemon (remote machine). FTP enables a user to specify the name of the remote host and file transfer command options on the local host's command line. The in.ftpd daemon on the remote host then handles the requests from the local host. Unlike RCP, FTP works even when the remote computer does not run a UNIX-based operating system. A user must log in to the remote computer to make an FTB connection unless it has been set up to allow anonymous FTP. (Source: http://www.sun.com/products-n-solutions/hardware/docs/html/817-6210-10/glossary.html)
Firewall		A piece of hardware and/or software which functions in a networked environment to prevent some communications forbidden by the security policy, analogous to the function of firewalls in building construction.

GIG Enterprise Service	GES	A service that provides capabilities for use in the DoD enterprise. GIG Enterprise Services are the combination of Core Enterprise Services and Community of Interest Services. Also referred to as Global Enterprise Services.
Global Command and Control System	GCCS	GCCS-J is the DOD joint C2 system of record for achieving full spectrum dominance. It enhances information superiority and supports the operational concepts of full-dimensional protection and precision engagement. GCCS-J is the principal foundation for dominant battlespace awareness, providing an integrated, near real-time picture of the battlespace necessary to conduct joint and multinational operations. It fuses select C2 capabilities into a comprehensive, interoperable system by exchanging imagery, intelligence, status of forces, and planning information. GCCS-J offers vital connectivity to the systems the joint warfighter uses to plan, execute, and manage military operations. GCCS-J is a Command, Control, Communications, Computer, and Intelligence (C4I) system, consisting of hardware, software, procedures, standards, and interfaces that provide a robust, seamless C2 capability. The system uses the Defense Information Systems Network (DISN) and must work over tactical communication systems to ensure connectivity with deployed forces in the tactical environment. (Source: http://www.disa.mil/gccs-j/) ***Note:*** *Other variants include GCCS-M to support Maritime operations.*
Global Information Grid	GIG	Globally interconnected, end-to-end set of information capabilities, associated processes, and personnel for collecting, processing, storing, disseminating, and managing information on demand to warfighters, policy makers, and support personnel. The GIG includes all owned and leased communications and computing systems and services, software (including applications), data, security services, and other associated services necessary to achieve Information Superiority. It also includes National Security Systems (NSS) as defined in section 5142 of the Clinger-Cohen Act of 1996. The GIG supports all DoD, National Security, and related Intelligence Community (IC) missions and functions (strategic, operational, tactical, and business) in war and in peace. The GIG provides capabilities from all operating locations (bases, posts, camps, stations, facilities, mobile platforms, and deployed sites). The GIG provides interfaces to coalition, allied, and non-DoD users and systems.
Global Positioning System	GPS	A satellite constellation that provides highly accurate position, velocity, and time navigation information to users. (Source: JP 1-02,)
High Assurance Internet Protocol Encryption	HAIPE	DoD version of Internet Protocol (IP) security (IPsec) protocol. (Source: http://en.wikipedia.org/wiki/HAIPE)
High Availability		Data tier availability can be affected by hardware failure, power outages, data errors, user errors, programmer errors, OS errors, and RDBMS errors. Various hardware and software methods help mitigate availability issues. The more reliable a system needs to be, the more it costs.

		Consequently, defining availability to meet requirements is essential to controlling costs.
Horizontal Fusion	HF	Horizontal Fusion (HF) is a direct response to Secretary of Defense Donald H. Rumsfeld's vision of Force Transformation. It demonstrates the ability to use lightweight automation to replace system mass with superior access to information based on a coherent architecture for an arbitrary future. Horizontal Fusion acts as a catalyst by implementing and demonstrating technologies and techniques that significantly advance the process of information-sharing in a an evolving net-centric environment. (Source: http://horizontalfusion.dtic.mil/vision/)
Hypertext Markup Language	HTML	A markup language for hypertext documents on the Internet. HTML supports embedding images, sounds, video streams, form fields, references to other objects with URLs, and basic text formatting. (Source: http://java.sun.com/j2ee/1.4/docs/glossary.html)
Hypertext Transfer Protocol	HTTP	The Internet protocol used to retrieve hypertext objects from remote hosts. HTTP messages consist of requests from client to server and responses from server to client. (Source: http://java.sun.com/j2ee/1.4/docs/glossary.html)
Identification	ID	An act or process that presents an identifier to a system so that the system can recognize a system entity (e.g., user, process, or device) and distinguish that entity from all others. (Source: Committee on National Security Systems Instruction (CNSSI) No. 4009, National Information Assurance (IA) Glossary, 26 April 2010; http://www.cnss.gov/Assets/pdf/cnssi_4009.pdf)
Identity		The set of attribute values (i.e., characteristics) by which an entity is recognizable and that, within the scope of an identity manager's responsibility, is sufficient to distinguish that entity from any other entity. (Source: Committee on National Security Systems Instruction (CNSSI) No. 4009, National Information Assurance (IA) Glossary, 26 April 2010; http://www.cnss.gov/Assets/pdf/cnssi_4009.pdf)
Identity Management		Provides the methodology and functions for maintaining information on people, consumers, and service providers. Supports the validation of identity authentication credentials.
Information Assurance	IA	Measures that protect and defend information and information systems by ensuring their availability, integrity, authentication, confidentiality, and non-repudiation. This includes providing for restoration of information systems by incorporating protection, detection, and reaction capabilities. (Source: DoD Directive 8500.1, *Information Assurance (IA)*, http://www.dtic.mil/whs/directives/corres/pdf/850001p.pdf)
Information Support Plan	ISP	The identification and documentation of information needs, infrastructure support, IT and NSS interface requirements and dependencies focusing on net-centric, interoperability, supportability and sufficiency concerns. (Source: DoD Instruction 4630.8, 30 June 2004, [R1168] Enclosure 2, Definitions)

Information Technology	IT	Any equipment or interconnected system or subsystem of equipment, that is used in the automatic acquisition, storage, manipulation, management, movement, control, display, switching, interchange, transmission, or reception of data or information. Information technology includes computers, ancillary equipment, software, firmware, and similar procedures, services (including support services), and related resources. Information technology does not include any equipment that is acquired by a federal contractor incidental to a federal contract. (Source: CJCSI 6212.01E, [R1175] Glossary page GL-14)
Information Technology Laboratory	ITL	The ITL at the **National Institute of Standards and Technology** (NIST) has the broad mission of supporting U.S. industry, government, and academia with measurements and standards that enable new computational methods for scientific inquiry, assure IT innovations for maintaining global leadership, and re-engineer complex societal systems and processes through insertion of advanced Information Technology (IT). (Source: http://www.itl.nist.gov/itl-what_itl_does.html)
Integrity		The property whereby an entity has not been modified in an unauthorized manner. (Source: CNSS Instruction No. 4009, 26 April 2010, *National Information Assurance (IA) Glossary* [R1339])
Intelligence Community	IC	A federation of executive branch agencies and organizations that conduct intelligence activities necessary for conduct of foreign relations and protection of national security. (Source: http://www.intelligence.gov/)
Interface		The functional and physical characteristics required to exist at a common boundary or connection between systems or items. (Source: *Defense Standardization Program (DSP) Policies and Procedures*, DoD 4120.24-M, March 2000) A Key Interface is a common boundary shared between system modules that provides access to critical data, information, materiel, or services; and/or is of high interest due to rapid technological change, a high rate of failure, or costliness of connected modules. (Source: *A Modular Open Systems Approach (MOSA) to Acquisition*, Version 2.0, September 2004; http://www.acq.osd.mil/osjtf/mosapart.html)
International Telecommunication Union	ITU	United Nations agency for information and communication technologies. (Source: http://www.itu.int/net/about/index.aspx)
Internet		The Internet, or simply the Net, is the publicly available worldwide system of interconnected computer networks that transmit data by packet switching using a standardized Internet Protocol (IP) and many other protocols. It is made up of thousands of smaller commercial, academic, and government networks. It carries various information and services, such as electronic mail, online chat and the interlinked web pages and other documents of the World Wide Web. Because this is by far the largest, most extensive internet (with a lower case i) in the world, it is simply called the Internet (with a capital I). (Source: http://en.wikipedia.org/wiki/Internet)

Internet Engineering Task Force	IETF	The Internet Engineering Task Force (IETF) is a large open international community of network designers, operators, vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet. It is open to any interested individual. (Source: http://www.ietf.org/overview.html)
Internet Protocol	IP	Data packets routed across network, not switched via dedicated circuits.
Internet Protocol Version 4	IPv4	Version 4 of the Internet Protocol (IP). It was the first version of the Internet Protocol to be widely deployed, and forms the basis for most of the current Internet (as of 2004). It is described in IETF RFC 791, which was first published in September, 1981. IPv4 uses 32-bit addresses, limiting it to 4,294,967,296 unique addresses, many of which are reserved for special purposes such as local networks or **multicast** addresses. This reduces the number of addresses that can be allocated as public Internet addresses. As the number of addresses available is consumed, an IPv4 address shortage appears to be inevitable in the long run. This limitation has helped stimulate the push towards IPv6, which is currently in the early stages of deployment, and may eventually replace IPv4. (Source: http://en.wikipedia.org/wiki/IPv4)
Internet Protocol Version 6	IPv6	Version 6 of the Internet Protocol; it was initially called IP Next Generation (IPng) when it was picked as the winner in the IETF's IPng selection process. IPv6 is intended to replace the previous standard, IPv4, which only supports up to about 4 billion (4×10^9) addresses. IPv6 supports up to about 3.4×10^{38} (340 undecillion) addresses. This is the equivalent of 4.3×10^{20} (430 quintillion) addresses per square inch (6.7×10^{17} (670 quadrillion) addresses/mm^2)of the Earth's surface. It is expected that IPv4 will be supported until at least 2025, to allow time for bugs and system errors to be corrected. (Source: http://en.wikipedia.org/wiki/Ipv6)
Intranet		An intranet is a local area network (LAN) used internally in an organization to facilitate communication and access to information that is sometimes access-restricted. Sometimes the term refers only to the most visible service, the internal web site. The same concepts and technologies of the Internet such as clients and servers running on the Internet protocol suite are used to build an intranet. HTTP and other internet protocols are commonly used as well, especially FTP and email. There is often an attempt to use internet technologies to provide new interfaces with corporate "legacy" data and information systems. (Source: http://en.wikipedia.org/wiki/Intranet)
Intrusion Detection System	IDS	An IDS inspects all inbound and outbound network activity and identifies suspicious patterns that may indicate a network or system attack from someone attempting to break into or compromise a system. (Source: http://www.webopedia.com/TERM/i/intrusion_detection_system.html)
Java 2 Platform, Enterprise Edition	J2EE	The J2EE environment is the standard for developing component-based multi-tier enterprise applications. The J2EE platform consists of a set of services, application programming interfaces (APIs), and protocols that provide the functionality for developing multitiered, Web-based applications. Features include Web services support and development tools. Sun Microsystems has simplified the name of the Java platform

		for the enterprise; the "2" is dropped from the name, as well as the dot number so the next version of the Java platform for the enterprise is **Java Platform, Enterprise Edition** 5 or Java EE 5.(Source: http://java.sun.com/j2ee/1.4/docs/glossary.html)
Java Message Service	JMS	An API for invoking operations on enterprise messaging systems. (Source: http://java.sun.com/j2ee/1.4/docs/glossary.html)
Java Platform, Enterprise Edition	Java EE	Java Platform, Enterprise Edition (Java EE) is the industry standard for developing portable, robust, scalable and secure server-side Java applications. Building on the solid foundation of the Java Platform, Standard Edition (Java SE), Java EE provides Web services, component model, management, and communications APIs that make it the industry standard for implementing enterprise-class service-oriented architecture (SOA) and next-generation Web applications. Sun Microsystems has simplified the name of the Java platform for the enterprise. Formerly, the platform was known as Java 2 Platform, Enterprise Edition (**J2EE**), and specific versions had "dot numbers" such as J2EE 1.4. The "2" is dropped from the name, as well as the dot number so the next version of the Java platform for the enterprise is Java Platform, Enterprise Edition 5 or Java EE 5. (Source: http://java.sun.com/javaee/)
JavaScript		The Netscape-developed object scripting language used in millions of web pages and server applications worldwide. Contrary to popular misconception, JavaScript is not "Interpretive Java." Rather, it is a dynamic scripting language that supports prototype-based object construction.
JavaServer Pages	JSP	An extensible Web technology that uses static data, JSP elements, and server-side Java objects to generate dynamic content for a client. Typically the static data is HTML or XML elements, and in many cases the client is a Web browser. (Source: http://java.sun.com/j2ee/1.4/docs/glossary.html)
Java Specification Request	JSR	Java Specification Requests (JSRs) are the actual descriptions of proposed and final specifications for the Java platform. At any one time there are numerous JSRs moving through the review and approval process. (Source: JSR Overview, http://jcp.org/en/jsr/overview)
Joint Capabilities Integration and Development System	JCIDS	The JCIDS procedures support the Chairman of the Joint Chiefs of Staff and the Joint Requirements Oversight Council (JROC) in identifying and assessing joint military capability needs. (Source: CJCSI 3170.01G, 1 March 2009, *Joint Capabilities Integration and Development System*)
Joint Interoperability Test Command	JITC	JITC provides a full-range of agile and cost-effective test, evaluation, and certification services to support rapid acquisition and fielding of global net-centric warfighting capabilities. (Source: http://jitc.fhu.disa.mil/mission.html)

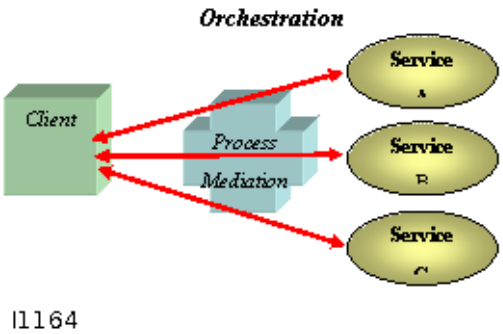
Joint Tactical Radio System	JTRS	JTRS is a family of interoperable, affordable software defined radios which provide secure, wireless networking communications capabilities for Joint forces. (Source: JTRS JPEO, http://jpeojtrs.mil/)
Joint Worldwide Intelligence Communications System	JWICS	The sensitive compartmented information portion of the **Defense Information Systems Network**. It incorporates advanced networking technologies that permit point-to-point or multipoint information exchange involving voice, text, graphics, data, and video teleconferencing. (Source: *DoD Diitionary of Military Terms*, http://www.dtic.mil/doctrine/dod_dictionary/; accessed 2 November 2010)
Key Interface Profile	KIP	An operational functionality, systems functionality and technical specifications description of the Key Interface. The profile consists of refined Operational and Systems Views, interface control specifications, Technical View with SV-TV Bridge, and referenced procedures for KIP compliance. The key interface profile is the technical specification that governs access to the **GIG**. (Source: CJCSI 6212.01D, 8 March 2006, Glossary page GL-14) ***Note:*** *CJCSI 6212.01E[R1175], 15 December 2008, deletes the "Key Interface Profile" element of the NR-KPP and replaces it with the "Technical Standards/Interfaces" element. This revision further indicates that Global Information Grid (GIG) Enterprise Service Profiles (GESPs) are evolving to provide a net-centric oriented approach for managing interoperabilty across the GIG based on the definition and configuration control of key interfaces and enterprise services.*
Key Performance Parameters	KPP	Those attributes or characteristics of a system that are considered critical or essential to the development of an effective military capability and those attributes that make a significant contribution to the key characteristics as defined in the Joint Operations Concepts. KPPs are validated by the Joint Requirements Oversight Council (JROC) for JROC Interest documents, and by the DOD component for Joint Integration or Independent documents. Capability development and capability production document KPPs are included verbatim in the acquisition program baseline. (Source: CJCSI 3170.01F[R1173], *Joint Capabilities and Development System*, 1 May 2007, Glossary page GL-14)
Least-Common-Denominator Data Access Mechanism		When one application is able to obtain data provided by another by removing arbitrary implementation barriers to data exchange.
Legacy System		An existing computer system or application program which continues to be used because the user (typically an organization) does not want to replace or redesign it. (Source: http://en.wikipedia.org/wiki/Legacy_system)
Light Directory Access Protocol	LDAP	The Lightweight Directory Access Protocol (LDAP) is an **Internet** protocol for accessing distributed directory services that act in accordance with X.500 data and service models. (Source: **Internet Engineering Task Force** Request for Comments 4510, ***Lightweight Directory Access Protocol (LDAP): Technical Specification Road Map***, http://tools.ietf.org/html/rfc4510)

Link-16	TADIL-J	Tactical Data Information Link (TADIL) primarily designed for use by Command and Control (C2) and Air-to-Air assets; uses the Joint Tactical Data Link (TADIL-J) message format. (Source: http://aatc.aztucs.ang.af.mil/aatcinfo.htm)
Local Area Network	LAN	A group of interconnected computer and support devices. (Source: http://www.sun.com/products-n-solutions/hardware/docs/html/817-6210-10/glossary.html)
Machine-to-Machine Messaging		Provides reliable machine-to-machine message exchange across the **enterprise**.
Mediation		A set of negotiated agreements for interacting between components that enable those components to work together to perform a task. These agreements are defined through common interfaces and data interchange specifications. Mediation services provide multiple methods for integrating data sources and services: Transformation: When a client requests data from a service in a particular format, a transformer retrieves and reformats the data before returning it to the client Aggregation: A mediator service may collect data derived from multiple sources, thus making many services appear to be one Adaptation: When a client cannot communicate directly with a service, an adapter provides service mediation (can be transport protocol as well as data format) when services need to communicate point-to-point Orchestration: Co-ordination of events in a process; orchestration directs and manages the on-demand assembly of multiple component services to create a composite application or business process Choreography: When a client request spawns a chain of events or service requests that do not rely on a central coordinator, a Choreographed Web Service knows when to execute other services and with which other services to interact; WS-CDL is an example of a business process management workflow language that implements choreography
Message		A self-contained unit of information exchanged between a producer and one or more consumers. Software commonly uses messages to communicate synchronously or asynchronously between service producers and consumers. Some examples of software messaging are **SOAP** messages, e-mail messages, **Data Distribution Service** (**DDS**) messages, and **Java Message Service** (**JMS**) messages.

Metadata		Data about the data, that is, the description of the data resources, its characteristics, location, usage, and so on. Metadata is used to identify, describe, and define user data.
Modular Design		Characterized by (1) Functional partitioning into discrete scalable, reusable modules consisting of isolated, self-contained functional elements; (2) Rigorous use of well-defined modular interfaces, including object-oriented descriptions of module functionality; (3) Ease of change to achieve technology transparency and, to the extent possible, make use of industry standards for key interfaces.
Module		(1) A program unit that is discrete and identifiable with respect to compiling, combining with other units, and loading; for example, the input to, or output from, an assembler, compiler, linkage editor, or executive routine. (2) A logically separable part of a program. Note: The terms ***module***, ***component***, and ***unit*** are often used interchangeably or defined to be sub-elements of one another in different ways depending upon the context. The relationship of these terms is not yet standardized. See also **component**. (Source: IEEE Std 610.12-1990)
Multicast		The delivery of information to a group of destinations simultaneously using the most efficient strategy to deliver the messages over each link of the network only once and only create copies when the links to the destinations split. (Source: http://en.wikipedia.org/wiki/Multicast)
MX Record		An MX record or Mail exchanger record is a type of resource record in the **Domain Name System** (DNS) specifying how **Internet** e-mail should be routed. MX records point to the servers that should receive an e-mail, and their priority relative to each other. (Source: http://en.wikipedia.org/wiki/MX_Record)
National Institute of Standards and Technology	NIST	Non-regulatory federal agency within the U.S. Commerce Department's Technology Administration with a mission to promote U.S. innovation and industrial competitiveness by advancing measurement science, standards, and technology in ways that enhance economic security and improve our quality of life. (Source: http://www.nist.gov/public_affairs/general2.htm)
National Security Agency	NSA	America's cryptologic organization; it coordinates, directs, and performs highly specialized activities to protect U.S. government information systems and produce foreign signals intelligence information. (Source: http://www.nsa.gov/about/index.cfm)
National Security Systems	NSS	Telecommunications and information systems, operated by the Department of Defense, the functions, operation, or use of which involves: (1) intelligence activities; (2) cryptologic activities related to national security; (3) the command and control of military forces; (4) equipment that is an integral part of a weapon or weapons systems; or (5) is critical to the direct fulfillment of military or intelligence missions. Subsection (5) in the preceding sentence does not include procurement of automatic data processing equipment or services to be used for routine administrative and business applications (including payroll,

		finance, logistics, and personnel management applications). (Source: CJCSI 3170.01F, 1 May 2007, page GL-16)
Net-Centric Enterprise Services	NCES	The NCES program provides enterprise-level Information Technology (IT) services and infrastructure components, also called Core Enterprise Services, for the Department of Defense (DoD) Global Information Grid (GIG).
Net-Centric Operations and Warfare Reference Model	NCOW RM	The NCOW RM described the activities required to establish, use, operate, and manage the net-centric enterprise information environment to include the generic user interface, the intelligent-assistant capabilities, the net-centric service capabilities (core services, **Community of Interest (COI) services**, and environment control services), and the enterprise management components. It also described a selected set of key standards that would be needed as the NCOW capabilities of the **Global Information Grid** (GIG) were realized. The NCOW RM represented the objective end-state for the GIG: a service-oriented, inter-networked, information infrastructure in which users request and receive services that enable operational capabilities across the range of military operations; **DoD** business operations; and Department-wide enterprise management operations. The NCOW RM was a key compliance mechanism for evaluating DoD information technology capabilities and the **Net-Ready Key Performance Parameter** in accordance with Chairman of the Joint Chiefs of Staff Instruction (CJCSI) 6212.01D, *Interoperability and Supportability of Information Technology and National Security Systems*, 8 March 2006. The 15 December 2008 revision to this instruction, CJCSI 6212.01E, removed the NCOW RM element of the Net-Ready Key Performance Parameter (NR-KPP), integrating the components of the former NCOW RM into other elements of the NR-KPP. (Source: CJCSI 6212.01E [R1175])
Net-Ready Key Performance Parameter	NR-KPP	The NR-KPP is a key parameter stating a system's information needs, information timeliness, information assurance (IA), and net-ready attributes required for both the technical exchange of information needs, information timeliness, IA, and net-ready attributes required for both the technical exchange of information and the operational effectiveness of that exchange. The NR-KPP consists of information required to evaluate the timely, accurate, and complete exchange and use of information to satisfy information needs for a given capability. ***Note:*** *The 15 December 2008 revision of the Chairman Joint Chief of Staff Instruction for Interoperability and Supportability of Information Technology and National Security Systems (CJCSI 6212.01E) removed the* ***NCOW RM*** *element of the NR-KPP, integrating its components into the other elements of the NR-KPP.* The NR-KPP is composed of the following five elements: • Compliant solution architecture • Compliance with DOD Net-Centric Data [R1172] and Services [R1313] strategies, including data and services exposure criteria • Compliance with applicable GIG Technical Direction to include **DISR**-mandated IT Standards reflected in the TV-1 and implementation guidance of GIG Enterprise Service Profiles (GESPs) necessary to meet all operational requirements specified in the DOD Information

		Enterprise Architecture and solution architecture system/service views • Verification of compliance with DOD IA requirements • Compliance with supportability elements to include, spectrum analysis, Selective Availability Anti-Spoofing Module (SAASM), and the Joint Tactical Radio System (JTRS) (Source: CJCSI 6212.01E [R1175])
Network Centric Warfare	NCW	NCW is an information superiority-enabled concept of operations that generates increased combat power by networking sensors, decision makers, and shooters to achieve shared awareness, increased speed of command, higher tempo of operations, greater lethality, increased survivability, and a degree of selfsynchronization. In essence, NCW translates information superiority into combat power by effectively linking knowledgeable entities in the battlespace. (Source: *Network Centric Warfare: Developing and Leveraging Information Superiority*. David S. Alberts, John J. Garstka and Frederick P. Stien. DoD Command and Control Research Program Publication Series, available at http://www.dodccrp.org/files/Alberts_NCW.pdf)
Network Intrusion Detection	NID	Attempt to detect malicious activity such as denial of service attacks, port-scans or even attempts to crack into computers by monitoring network traffic. (Source: http://en.wikipedia.org/wiki/Network_intrusion-detection_system)
Network Operations	NetOps	An organizational, procedural, and technological construct for ensuring information and decision superiority at the strategic, operational, and tactical levels of warfare as well as within DoD business operations. NetOps is an operational approach, which addresses the interdependency and integration of Information Assurance/Computer Network Defense (IA/CND), Systems and Network Management (S&NM), and Content Staging (CS) capabilities. NetOps consists of the organizations, tactics, techniques, procedures, functionalities, and technologies required to plan, administer, and monitor use of the GIG infrastructure and the end-to-end information flows of the GIG; and to respond to threats, outages, and other operational impact. NetOps ensures mission requirements are properly considered in GIG operational decision-making. NetOps enables the GIG to provide its users with information they need, when and where they need it, with appropriate protection. NetOps is essential for successful execution of net-centric warfare and other net-centric operations in support of national security objectives.
Network Time Protocol	NTP	Protocol for synchronizing the clocks of computer systems over packet-switched, variable-latency data networks. NTP uses **User Datagram Protocol** (UDP) port 123 as its transport layer. It is designed particularly to resist the effects of variable latency. (Source: http://en.wikipedia.org/wiki/Network_Time_Protocol)
Node		In general network usage, a node is a processing location such as a computer or some other device. Every node has a unique network address, sometimes called a Data Link Control (DLC) address or Media Access Control (MAC) address. (Source: http://www.webopedia.com/TERM/n/node.html)

		A NESI Node is a collection of integrated components (i.e., systems, applications, services and other Nodes) that are bound together spatially and/or temporally to meet the needs of a particular mission. It is conceptual in nature and can not be defined in terms of a concrete set of components or size. The membership of a component within a particular Node is not exclusive and a Component can be part of multiple Nodes.
Node Information Services	NIS	
Online Certificate Status Protocol	OCSP	Online Certificate Status Protocol is a method for determining the revocation status of an X.509 digital certificate using means other than **CRLs**. It is described in RFC 2560 and is on the Internet standards track. OCSP messages are encoded in ASN.1 and usually communicated over **HTTP**. OCSP's request/response nature leads to OCSP servers being termed as OCSP responders.
Open Virtualization Format	OVF	An open, secure, portable, efficient and extensible format for the packaging and distribution of software to be run in virtual machines. (Source: Distributed Management Task Force *Open Virtualization Format Specification*, http://www.dmtf.org/standards/published_documents/DSP0243_1.0.0.pdf) Source: Distributed Management Task Force Context: Virtualization
Operational View	OV	The OV is a description of the tasks and activities, operational elements, and information exchanges required to accomplish DoD missions. DoD missions include both warfighting missions and business processes. The OV contains graphical and textual products that comprise an identification of the operational nodes and elements, assigned tasks and activities, and information flows required between nodes. It defines the types of information exchanged, the frequency of exchange, which tasks and activities are supported by the information exchanges, and the nature of information exchanges. (Source: *DoDAF* v1.5 Volume I: Definitions and Guidelines, 23 April 2007)
Orchestration		Co-ordination of events in a process; orchestration directs and manages the on-demand assembly of multiple component services to create a composite application or business process. (Source: http://looselycoupled.com/glossary/orchestration) 

		Note: *See* ***Mediation****.*
Plain Text	PT	Textual data in **ASCII** format. Plain text is the most portable format because it is supported by nearly every application on every machine. It is quite limited, however, because it cannot contain any formatting commands. In cryptography, plain text refers to any message that is not encrypted. (Source: http://www.webopedia.com/TERM/p/plain_text.html)
Plug-In		A hardware or software module that adds a specific feature or service to a larger system. (Source: http://www.webopedia.com/TERM/p/plug_in.html)
Portal		A Web portal is a **Web site** that provides a starting point, gateway, or portal to other resources on the **Internet** or an intranet. Intranet portals are also known as "enterprise information portals" (EIP). Examples of existing portals are Yahoo, Excite, Lycos, Altavista, Infoseek, and Hotbot. (Source: http://en.wikipedia.org/wiki/web_portal)
Portlet		Portlets are pluggable user interface software components that are managed and displayed in a **Web** portal. Portlets produce fragments of markup code that are aggregated into a portal page. Typically, following the desktop metaphor, a portal page is displayed as a collection of non-overlapping portlet windows, where each portlet window displays a portlet. Hence a portlet (or collection of portlets) resembles a Web-based application that is hosted in a portal. Portlets may be implemented using various specifications such as the **Web Services for Remote Portlets** (**WSRP**) protocol or the Java Portlet Specification. (Source: adapted from Wikipedia ***Portlet***, http://en.wikipedia.org/wiki/Portlets, accessed 13 September 2010)
Private Key		The private key is one of a pair of keys that are generated as part of asymmetric key cryptography. The private key is kept secret; the **public key** can be shared openly with others.
Protocol		An agreed-upon format for transmitting data between two devices. The protocol determines the type of error checking to be used, data compression method, if any, how the sending device will indicate that it has finished sending a message, and how the receiving device will indicate that it has received a message. (Source: http://www.webopedia.com/TERM/p/protocol.html)
Proxy		A **server** that sits between a client application, such as a **Web browser**, and a real server. It intercepts all requests to the real server to see if it can fulfill the requests itself. If not, it forwards the request to the real server. Proxy servers have two main purposes: improve performance and filter requests. (Source: http://www.webopedia.com/TERM/p/proxy_server.html)
Public Key	PK	See **Public Key Cryptography**.

Public Key Cryptography		Public key cryptography, also known as asymmetric cryptography, is a form of cryptography in which a user has a pair of cryptographic keys - a public key and a private key. The private key is kept secret, while the public key may be widely distributed. The keys are related mathematically, but the private key cannot be practically derived from the public key. A message encrypted with the public key can be decrypted only with the corresponding private key. (Source: http://en.wikipedia.org/wiki/Public_key)
Public Key Enabling	PK-Enabling	The incorporation of the use of certificates for security services such as authentication, confidentiality, data integrity, and nonrepudiation. PK-Enabling involves replacing existing or creating new user authentication systems using certificates instead of other technologies, such as userid and password or **Internet Protocol** filtering; implementing public key technology to digitally sign, in a legally enforceable manner, transactions and documents; or using public key technology, generally in conjunction with standard symmetric encryption technology, to encrypt information at rest and/or in transit. (Source: DoD Instruction 8520.2, *Public Key Infrastructure (PKI) and Public Key (PK) Enabling*, 1 April 2004 [R1206])
Public Key Infrastructure	PKI	The framework and services that provide for the generation, production, distribution, control, accounting and destruction of public key certificates. Components include the personnel, policies, processes, server platforms, software, and workstations used for the purpose of administering certificates and public-private key pairs, including the ability to issue, maintain, recover, and revoke public key certificates. (Source: CNSS Instruction No. 4009, 26 April 2010, *National Information Assurance (IA) Glossary* [R1339])
Publish/Subscribe Messaging System		A messaging system in which clients address messages to a specific node in a content hierarchy, called a topic. Publishers and subscribers are generally anonymous and can dynamically publish or subscribe to the content hierarchy. The system takes care of distributing the messages arriving from a node's multiple publishers to its multiple subscribers. Messages are generally not persistent and will only be received by subscribers who are listening at the time the message is sent. A special case known as a "durable subscription" allows subscribers to receive messages sent while the subscribers are not active. (Source: http://java.sun.com/j2ee/1.4/docs/glossary.html)
Quality of Service	QoS	Data timeliness, accuracy, completeness, integrity, and ease of use. Refers to the probability of the network meeting a given traffic contract. In many cases is used informally to refer to the probability of a packet passing between two points in the network. (Source: http://en.wikipedia.org/wiki/Quality_of_service) -OR- A defined level of performance that adapts to the environment in which it is operating. QoS may be requested by the user of the information. The level of QoS provided is based on the request, the available capabilities of the provider, and the priority of the user.

Registration Web Service	RWS	**Horizontal Fusion** (**HF**) **service** used by data producers to register content sources.
Relational Database Management System	RDBMS	A database management system (DBMS) that is based on the relational model or that presents the data to the user as relations. A collection of tables, each table consisting of a set of rows and columns, can satisfy this property. RDBMSs also provide relational operators to manipulate the data in tabular form. (Source: http://en.wikipedia.org/wiki/RDBMS)
Representational State Transfer	REST	The Representational State Transfer (REST) architectural style for distributed hypermedia systems was originally defined by Roy Fielding in his Ph.D. dissertation, *Architectural Styles and#the Design of Network-based Software Architectures*. One of the authors of the later **HTTP** protocol specifications, he defined a minimalist, stateless-protocol approach to coordinating a service's client and server across a network. RESTful designs adhere to the following constraints: • Client-Server • Stateless • Cacheable • Layered System • Uniform interface Optionally, RESTful designs may also support a sixth constraint: • Code-on-Demand Originally intended for Web hypermedia, the general approach has since been extended to services layered on other protocols and data formats. (Souce: Fielding, Roy Thomas. Architectural Styles and the Design of Network-based Software Architectures. Doctoral dissertation, University of California, Irvine, 2000; http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm)
Role-Based Access Control	RBAC	With RBAC, security is managed at a level that corresponds closely to the organization's structure. Each user is assigned one or more roles, and each role is assigned one or more privileges that are permitted to users in that role. Security administration with RBAC consists of determining the operations that must be executed by persons in particular jobs, and assigning employees to the proper roles. Complexities introduced by mutually exclusive roles or role hierarchies are handled by the RBAC software, making security administration easier. (Source: National Institute of Standards and Technology Computer Security Resource Center, http://csrc.nist.gov/groups/SNS/rbac/)
Router		A device that forwards data packets along networks. A router is connected to at least two networks, commonly two **local area networks** (LANs) or wide area networks (WANs) or a LAN and its Internet Service Provider's network. Routers are located at gateways, the places where two or more networks connect. (Source: http://www.webopedia.com/TERM/r/router.html)

Schema		A diagrammatic representation, an outline, or a model. In relation to data management, a schema can represent any generic model or structure that deals with the organization, format, structure, or relationship of data. Some examples of schemas are (1) a database table and relational structure, (2) a **document type definition** (DTD), (3) a data structure used to pass information between systems, and (4) an **XML schema document** (XSD) that represents a data structure and related information encoded as XML. Schemas typically do not contain information specific to a particular instance of data (Source: DoD 8320.02-G, 12 April 2006, *Guidance for Implementing Net-Centric Data Sharing*)
Search Web Service	SWS	**Horizontal Fusion** (**HF**) **service** used to search for content from registered sources.
Secret Internet Protocol Router Network	SIPRNet	SIPRNet is DoD's largest interoperable command and control data network, supporting the **Global Command and Control System** (GCCS), the Defense Message System (DMS), collaborative planning and numerous other classified warfighter applications. Direct connection data rates range from 56 kbps to 155 Mbps. Remote dial-up services are available up to 19.2 kbps. (Source: http://www.disa.mil/services/data.html)
Secret Key		The asymmetric key cryptography approach generates two keys, a public key and a private key. The **private key** is often referred to as the secret key.
Secure Sockets Layer	SSL	A protocol for transmitting private documents via the Internet. SSL uses a cryptographic system employing two keys to encrypt data: a **public key** known to everyone and a **private** or **secret key** known only to the recipient of the message. (Source: http://www.webopedia.com/TERM/S/SSL.html)
Security Assertion Markup Language	SAML	The Security Assertion Markup Language (SAML) is a set of specifications describing security assertions that are encoded in **XML**, profiles for attaching the assertions to various protocols and frameworks, the request/response protocol used to obtain the assertions, and bindings of this protocol to various transfer protocols (for example, **SOAP** and **HTTP**). (Source: *Glossary for the OASIS Security Assertion Markup Language (SAML) V2.0*, http://docs.oasis-open.org/security/saml/v2.0/saml-glossary-2.0-os.pdf)
Security Technical Implementation Guide	STIG	Configuration standards for DoD **IA** and IA-enabled devices/systems. (Source: http://iase.disa.mil/stigs/index.html)
Sensitive Compartmented Information	SCI	Classified information concerning or derived from intelligence sources, methods, or analytical processes, that is required to be handled within formal access control systems established by the Director of Central Intelligence (DCI). (Source: DoD Directive 8520.1, 20 December 2001, *Protection of Sensitive Compartmented Information (SCI)*, Page 2, Section 3.3)

Server		A computer software application that carries out some task (i.e., provides a service) on behalf of yet another piece of software called a **client**.
Service		A service is an autonomous encapsulation of some business or mission functionality. The service concept includes the notion of service providers and service consumers interacting via well-defined reusable interfaces. ***Note:*** *See the Service-Oriented Architecture [P1304] perspective in Part 1 for additional information concerning services including implementation characteristics.*
Service Access Point	SAP	A SAP provides all of the information necessary for a user to access and consume a service including the logical and physical location of the service on the net.
Service Definition Framework	SDF	An SDF provides a common frame of reference for service users, customers, developers, providers, and managers. Its structure and methodology enable full definition of the **Service Access Points** (**SAPs**) for a service. ***Note:*** *See P1296 [P1296]:* ***Service Definition Framework*** *for additional information.*
Service Discovery	SD	Provides a ***yellow pages***, categorized by **DoD** function, enabling users to advertise and locate capabilities available on the network.
Service Level Agreement	SLA	A contractual vehicle between a service provider and a service consumer. It specifies performance requirements, measures of effectiveness, reporting, cost, and recourse. It usually defines repair turnaround times for users.
Service Management		Enables monitoring of DoD **Web services**. Provides reporting of service-level information to potential and current service consumers, program analysts, and program managers.
Service-Oriented Architecture	SOA	NESI describes SOA as an architectural style used to design, develop, and deploy information technology (IT) systems based on decomposing functionality into services with well-defined interfaces. ***Note:*** *See the Service-Oriented Architecture [P1304] perspective in Part 1 for additional information.*
Servlet		A Java program that extends the functionality of a Web server, generating dynamic content and interacting with Web applications using a request-response paradigm. (Source: http://java.sun.com/j2ee/1.4/docs/glossary.html)

Simple Mail Transfer Protocol	SMTP	The objective of the Simple Mail Transfer Protocol (SMTP) is to transfer mail reliably and efficiently. SMTP is independent of the particular transmission subsystem and requires only a reliable ordered data stream channel. (Source: Internet Engineering Task Force (IETF) Request for Comments (RFC) 5321 available at http://tools.ietf.org/html/rfc5321)
Simple Unstructured Data		Simple Unstructured Data has uncomplicated data structure but not all requisite **metadata** is provided.
Situation Awareness Data Link	SADL	An Enhanced Position Location and Reporting System (EPLRS) radio modified for use in an aircraft. SADL and EPLRS radios are used to establish a common secure tactical data link network. (Source: http://aatc.aztucs.ang.af.mil/aatcinfo.htm)
Smart Card		A credit card-size device, normally for carrying and use by personnel, that contains one or more integrated circuits and also may employ one or more of the following technologies: magnetic stripe, bar codes (linear and two-dimensional), non-contact and radio frequency transmitters, biometric information, encryption and authentication, or photo identification. (Source: DoD Directive 8190.3, *Smart Card Technology*, 31 August 2003, Page 2, Section 3.2)
SOAP		SOAP Version 1.2 is a lightweight protocol intended for exchanging structured information in a decentralized, distributed environment. It uses XML technologies to define an extensible messaging framework providing a message construct that can be exchanged over a variety of underlying protocols. The framework has been designed to be independent of any particular programming model and other implementation specific semantics. (Source: SOAP Version 1.2 Second Edition, http://www.w3.org/TR/soap12-part1/#intro) ***Note:*** *The World Wide Web Consortium (W3C) changed the name of this protocol from* ***Simple Object Access Protocol 1.1 (SOAP)*** *to* ***SOAP Version 1.2*** *in the current version.*
Software Component		A software component is a software system element offering a predefined service and able to communicate with other components. It is a unit of independent deployment and versioning, encapsulated, multiple-use, non-context-specific and composeable with other components. (Source: http://en.wikipedia.org/wiki/Software_component#Software_component)
Software Developers Kit	SDK	A set of development tools that allows a software engineer to create applications for a certain software package, software framework, hardware platform, computer system, operating system, and so on. It may be as simple as an application programming interface in the form of some files to interface to a particular programming language, or as complex as sophisticated hardware to communicate with a certain embedded system. Common tools include debugging aids and other utilities. SDKs frequently include sample code, technical notes, and other supporting documentation to clarify points from the primary reference material. (Source: http://en.wikipedia.org/wiki/SDK)

Spyware		Any software that covertly gathers user information through the user's **Internet** connection without the user's knowledge, usually for advertising purposes. (Source: http://www.webopedia.com/TERM/s/spyware.html)
Stakeholder		An enterprise, organization, or individual having an interest or a stake in the outcome of the engineering of a system. (Source: EIA-632, Annex A)
Storage		Provides physical and virtual places to host and retain data for purposes such as content staging, continuity of operations, or archival.
Structured Identifier		Identifiers are labels which serve as references to the identity of resources, assets, **nodes**, **components**, and other entities. Ideally, identifiers should quickly answer at least one of the following common questions about the entity: who, what, where, when and which. Identifiers include, for example, names (for user environment usage), addresses (for transport usage), pathnames (for computing infrastructure usage), cryptographic keys (for security/**IA** usage) and above all, **Uniform Resource Identifiers** or **URIs** (for management, applications and services). Not all identifiers are structured; however, a benefit of structured identifiers is that they are useful for component software and hardware to understand and parse progressively the data expressed within the identifier. Progressive understanding of a standardized structured identifier is a form of negotiation that enables different entities either to interoperate correctly or to conclude efficiently that interoperation is not possible, even when the entities have never communicated before. For example, structured identifiers commonly identify the type and instance of an entity. Structuring an identifier into type portions and instance portions enables it to answer quickly and efficiently both what type of interactions are possible and with which instance of that type. Another common practice is for structured identifiers to express the hierarchical relationship between entities. Examples of structured identifiers expressing a hierarchical relationship include domain names such as **`nesipublic.spawar.navy.mil`** or the familiar telephone number hierarchy of country code, area code, exchange and line. The hierarchical structure in those cases indicates that there is a governance authority hierarchy whose top level delegates authority to the lower ones. Examples of useful standards for interoperable net-centric structured identifiers include the following: • (**IETF**) Request for Comments (RFC) 3986, *Uniform Resource Identifier: Generic Syntax*, http://tools.ietf.org/html/rfc3986 • IETF RFC 1035, *Domain Names - Implementation and Specification*, http://tools.ietf.org/html/rfc1035 • *Multi-Purpose Internet Mail Extensions (MIME) Media Types*, http://www.iana.org/assignments/media-types/ • *XML Path Language (XPath) Version 1.0*, http://www.w3.org/TR/xpath
Sustainment		One of the two major efforts (with disposal) of the Operations and Support phase of a DoD acquisition program. Sustainment includes supply, maintenance, transportation, sustaining engineering, data

		management, configuration management, manpower, personnel, training, habitability, survivability, environment, safety (including explosives safety), occupational health, protection of critical program information, anti-tamper provisions, and **Information Technology** (**IT**), including **National Security Systems** (**NSS**), supportability and interoperability functions. (Source: DoD Instruction 5000.2, 12 May 2003, *Operation of the Defense Acquisition System*, Section 3.9.2)
Symmetric Key Algorithm		Encryption algorithm where the same key is used for both encrypting and decrypting a message.
System		A system is a construct or collection of different elements that together produce results not obtainable by the elements alone. The elements, or parts, can include people, hardware, software, facilities, policies, and documents; that is, all things required to produce systems-level results. The results include system level qualities, properties, characteristics, functions, behavior and performance. The value added by the system as a whole, beyond that contributed independently by the parts, is primarily created by the relationship among the parts; that is, how they are interconnected (Rechtin, 2000). (Source: International Council on Systems Enginering, *A consensus of the INCOSE Fellows*, http://www.incose.org/practice/fellowsconsensus.aspx)
System Component		A basic part of a system. System components may be personnel, hardware, software, facilities, data, material, services, and/or techniques that satisfy one or more requirements in the lowest levels of the functional architecture. System components may be subsystems and/or configuration items. ***Note:*** *See* ***component****.*
Systems and Services View	SV	The SV is a set of graphical and textual products that describes systems and interconnections providing for, or supporting, DoD functions. DoD functions include both warfighting and business functions. The SV associates systems resources to the Operational View (OV). These systems resources support the operational activities and facilitate the exchange of information among operational nodes. (Source: DoDAF v1.5 Volume I: Definitions and Guidelines, 23 April 2007)
Technical Standards View	TV	The TV is the minimal set of rules governing the arrangement, interaction, and interdependence of system parts or elements. Its purpose is to ensure that a system satisfies a specified set of operational requirements. The TV provides the technical systems implementation guidelines upon which engineering specifications are based, common building blocks are established, and product lines are developed. The TV includes a collection of the technical standards, implementation conventions, standards options, rules, and criteria organized into profile(s) that govern systems and system elements for a given architecture. (Source: *DoDAF* v1.5 Volume 1: Definitions and Guidelines, 23 April 2007)
Test and Evaluation Master Plan	TEMP	Describes all planned testing, including measures to evaluate the performance of the system during test periods, an integrated test schedule, and resource requirements.

Transmission Control Protocol	TCP	One of the core protocols of the Internet protocol suite. Using TCP, programs on networked computers can create connections to one another, over which they can send data. The protocol guarantees that data sent by one endpoint will be received in the same order by the other, without any pieces missing. It also distinguishes data for different applications (such as a Web server and an email server) on the same computer. (Source: http://en.wikipedia.org/wiki/Transmission_Control_Protocol)
Transmission Control Protocol/Internet Protocol	TCP/IP	TCP is a connection-oriented, end-to-end reliable protocol designed to fit into a layered hierarchy of protocols which support multi-network applications. The TCP provides for reliable inter-process communication between pairs of processes in host computers attached to distinct but interconnected computer communication networks. (Source: **Internet Engineering Task Force** Request for Comments 793, ***Transmission Control Protocol: DARPA Internet Program Protocol***, September 1981, http://tools.ietf.org/rfc/rfc0793.txt)
Trusted Guard		Accredited to pass information between two networks at different security levels according to well defined rules and other controls. Guard products only pass defined types of information (e.g., email, images, or formatted messages). A key challenge is how to implement net-centric operations across trusted guards in the presence of **CES** services.
Trusted Platform Module	TPM	The TPM is a microcontroller that stores keys, passwords and digital certificates. It typically is affixed to the motherboard of computers. It potentially can be used in any computing device that requires these functions. The nature of this hardware chip ensures that the information stored there is made more secure from external software attack and physical theft. The TPM standard is a product of the Trusted Computing Group consortium. Source: Encryption of Sensitive Unclassified Data at Test on Mobile Computing Devices and Removable Storage Media R1330: DoD Memorandum , **Encryption of Sensitive Unclassified Data at Rest on Mobile Computing Devices and Removable Storage Media** Chief Information Officer . [http://iase.disa.mil/policy-guidance/dod-dar-tpm-decree07-03-07.pdf]
Tunneling		Transporting IPv6 traffic through IPv4 networks by encapsulating IPv6 packet in IPv4 and vice-versa.
Unclassified but Sensitive Internet Protocol Router Network	NIPRNet	The Unclassified but Sensitive Internet Protocol (IP) Router Network (NIPRNet) is a global long-haul IP based network to support unclassified IP data communications services for combat support applications to the Department of Defense (DoD), Joint Chiefs of Staff (JS), Military Departments (MILDEPS), and Combatant Commands (COCOM). NIPRNet provides seamless interoperability IP services to customers with access data rates ranging from 56KB to 1.0GB via direct connections to a NIPRNet router, remote dial-up services (56KB), services to the Tactical community via ITSDN/STEP sites, and access to the Internet. (Source: http://www.disa.mil/services/data.html)

Uniform Resource Identifier	URI	An encoded address that represents any Web resource, such as an **HTML** document, image, video clip, or program. As opposed to a **URL** or a **URN**, which are concrete entities, a URI is an abstract superclass. (Source: http://publib.boulder.ibm.com/infocenter/adiehelp/index.jsp?topic=/com.ibm.wsinted.glossary.doc/topics/glossary.html)
Uniform Resource Locator	URL	A sequence of characters that represents information resources on a computer or in a network such as the Internet. This sequence of characters includes (1) the abbreviated name of the protocol used to access the information resource and (2) the information used by the protocol to locate the information resource.(Source: http://publib.boulder.ibm.com/infocenter/adiehelp/index.jsp?topic=/com.ibm.wsinted.glossary.doc/topics/glossary.html)
Uniform Resource Name	URN	A name that uniquely identifies a **Web service** to a **client**. (Source: http://publib.boulder.ibm.com/infocenter/adiehelp/index.jsp?topic=/com.ibm.wsinted.glossary.doc/topics/glossary.html)
Universal Description, Discovery, and Integration	UDDI	An industry initiative to create a platform-independent, open framework for describing services, discovering businesses, and integrating business services using the Internet, as well as a registry. It is being developed by a vendor consortium. (Source: http://java.sun.com/j2ee/1.4/docs/glossary.html)
User Datagram Protocol	UDP	A connectionless protocol that, like **TCP**, runs on top of **Internet Protocol** (IP) networks. Unlike **Transmission Control Protocol/ Internet Protocol** (TCP/IP), UDP/IP provides very few error recovery services, offering instead a direct way to send and receive datagrams over an IP network. It's used primarily for broadcasting messages over a network. (Source: http://www.webopedia.com/TERM/U/User_Datagram_Protocol.html)
Virtual Private Network	VPN	A network that is constructed by using public wires to connect nodes. For example, there are a number of systems that enable the creation of networks using the Internet as the medium for transporting data. These systems use encryption and other security mechanisms to ensure that only authorized users can access the network and that the data cannot be intercepted. (Source: http://www.webopedia.com/TERM/V/VPN.html)
Web Browser		A client program that initiates requests to a **Web server** and displays the information that the server returns. (Source: http://publib.boulder.ibm.com/infocenter/adiehelp/index.jsp?topic=/com.ibm.wsinted.glossary.doc/topics/glossary.html)
Web Container		A container that implements the Web-component contract of the **J2EE** architecture. This contract specifies a runtime environment for Web components that includes security, concurrency, life-cycle management, transaction, deployment, and other services. A Web container provides the same services as a **JSP** container as well as a federated view of the J2EE platform **APIs**. A Web container is provided by a Web or J2EE server. (Source: http://java.sun.com/j2ee/1.4/docs/glossary.html)

Web Server		Software that provides services to access the Internet, an intranet, or an extranet. A Web server hosts **Web sites**, provides support for HTTP and other protocols, and executes server-side programs (such as Common Gateway Interface (CGI) scripts or servlets) that perform certain functions. In the **J2EE** architecture, a Web server provides services to a **Web container**. For example, a Web container typically relies on a Web server to provide **HTTP** message handling. The J2EE architecture assumes that a Web container is hosted by a Web server from the same vendor, so it does not specify the contract between these two entities. A Web server can host one or more Web containers. (Source: http://www.oracle.com/technetwork/java/javaee/index-jsp-139417.html)
Web Service		A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format. Web service implementation can use any number of technologies and standards including **SOAP** messages and **REST**. (Source: http://www.w3.org/TR/ws-gloss/)
Web Services for Interactive Applications	WSIA	
Web Services for Remote Portlets	WSRP	The WSRP specification defines a **Web service** interface for interacting with interactive presentation-oriented Web services. It has been produced through the joint efforts of the Web Services for Interactive Applications (**WSIA**) and Web Services for Remote Portals (WSRP) OASIS Technical Committees. Scenarios that motivate WSRP/ WSIA functionality include (1) **portal** servers providing **portlets** as presentation-oriented Web services that can be used by aggregation engines; (2) portal servers consuming presentation-oriented Web services provided by portal or non-portal content providers and integrating them into a portal framework. (Source: http://www.oasis-open.org/committees/download.php/3343/oasis-200304-wsrp-specification-1.0.pdf)
Web Services Interoperability Organization	WS-I	WS-I is an open industry organization chartered to promote Web services interoperability across platforms, operating systems and programming languages. The organization's diverse community of Web services leaders helps customers to develop interoperable Web services by providing guidance, recommended practices and supporting resources. (Source: http://www.ws-i.org/about/Default.aspx)
Web Site		A Web site, website, or WWW site (often shortened to just "site") is a collection of Web pages (i.e., HTML/XHTML documents accessible via **HTTP** on the Internet). All publicly accessible Web sites in existence comprise the World Wide Web. The pages of a Web site are accessed from a common root URL, the homepage, and usually reside on the same physical server. The URLs of the pages organize them into a hierarchy, although the hyperlinks between them control how the reader perceives the overall structure and how the traffic flows between the different parts of the site. (Source: http://en.wikipedia.org/wiki/web_site)
World Wide Web	WWW	The World Wide Web ("WWW," or simply "Web") is an information space in which items of interest, referred to as resources, are identified by

		global identifiers called **Uniform Resource Identifiers** (**URI**). The term is often mistakenly used as a synonym for the **Internet**, but the web is actually a service that operates over the Internet. (Source: http://en.wikipedia.org/wiki/World_Wide_web)
World Wide Web Consortium	W3C	The World Wide Web Consortium (W3C) is an international consortium where Member organizations, a full-time staff, and the public work together to develop Web standards. W3C's mission is to lead the World Wide Web to its full potential by developing protocols and guidelines that ensure long-term growth for the Web. (Source: http://www.w3.org/Consortium/)
XML Schema Definition	XSD	A language proposed by the **W3C** XML Schema Working Group for use in defining schemas. Schemas are useful for enforcing structure and/or constraining the types of data that can be used validly within other XML documents. XML Schema Definition refers to the fully specified and currently recommended standard for use in authoring XML schemas. Because the XSD specification was only recently finalized, support for it was only made available with the release of MSXML 4.0. It carries out the same basic tasks as DTD, but with more power and flexibility. Unlike DTD, which requires its own language and syntax, XSD uses XML syntax for its language. XSD closely resembles and extends the capabilities of XDR. Unlike XDR, which was implemented and made available by Microsoft in MSXML 2.0 and later releases, the W3C now recommends the use of XSD as a standard for defining XML schemas. (Source: http://msdn2.microsoft.com/en-us/library/ms256452.aspx)
XSL Transformations	XSLT	A language to express the transformation of XML documents into other XML documents. (Source: W3C Glossary)

References

R1121	W3C , **Extensible Markup Language (XML) 1.0 (Fifth Edition)** . [http://www.w3.org/TR/2008/REC-xml-20081126/]
R1164	DoD Directive 5000.01, *The Defense Acquisition System*, 12 May 2003 (certified current as of 20 November 2007); http://www.dtic.mil/whs/directives/corres/pdf/500001p.pdf.
R1165	DoD Instruction 5000.02, *Operation of the Defense Acquisition System*, 8 December 2008; http://www.dtic.mil/whs/directives/corres/pdf/500002p.pdf.
R1167	DoD Directive 4630.05, *Interoperability and Supportability of Information Technology (IT) and National Security Systems (NSS)*, 05 May 2004 (certified current as of 23 April 2007); http://www.dtic.mil/whs/directives/corres/pdf/463005p.pdf.
R1168	DoD Instruction 4630.8, *Procedures for Interoperability and Supportability of Information Technology (IT) and National Security Systems (NSS)*, 30 June 2004; http://www.dtic.mil/whs/directives/corres/pdf/463008p.pdf.
R1170	DoD CIO , **Global Information Grid Architectural Vision** . [http://cio-nii.defense.gov/docs/gigarchvision.pdf]
R1171	DoD Deputy CIO None , **DoD Architecture Framework (DoDAF)** . [http://cio-nii.defense.gov/sites/dodaf20/]
R1172	*DoD Net-Centric Data Strategy*, DoD Chief Information Officer, 9 May 2003, http://www.defenselink.mil/cio-nii/docs/Net-Centric-Data-Strategy-2003-05-092.pdf
R1173	CJCSI 3170.01G, *Joint Capabilities Integration and Development System*, 01 March 2009; http://www.dtic.mil/cjcs_directives/cdata/unlimit/3170_01.pdf.
R1174	CJCSM 3170.01C, *Operation of the Joint Capabilities Integration and Development System*, 01 May 2007; http://www.dtic.mil/cjcs_directives/cdata/unlimit/m317001.pdf.
R1175	CJCSI 6212.01E, *Interoperability and Supportability of Information Technology and National Security Systems*, 15 December 2008; http://www.dtic.mil/cjcs_directives/cdata/unlimit/6212_01.pdf.
R1176	*Net-Centric Operations and Warfare Reference Model (NCOW RM)*, v1.1, 17 November 2005. ***Note:*** *CJCSI 6212.01E removed the NCOW RM element of the Net-Ready Key Performance Parameter (NR-KPP), integrating the components of the former NCOW RM into other elements of the NR-KPP.*
R1177	*Net-Centric Checklist*, V2.1.3, Office of the Assistant Secretary of Defense for Networks and Information Integration/Department of Defense Chief Information Officer, 12 May 2004; http://www.defenselink.mil/cio-nii/docs/NetCentric_Checklist_v2-1-3_.pdf.
R1178	*A Modular Open Systems Approach (MOSA) to Acquisition*, Version 2.0, September 2004; http://www.acq.osd.mil/osjtf/mosapart.html.
R1179	, **DoD IT Standards Registry (DISR)** . [https://disronline.disa.mil]

	Note: Valid DoD PKI Certificate reqired to access site; non-registered users can view or download the current DISR release (sorted by several different categories) from the Reports & Archives page (see DISR site for link).
R1180	*Net-Centric Attributes List*, Office of the Assistant Secretary of Defense for Networks and Information Integration/Department of Defense Chief Information Officer, 2 February 2007; http://www.defenselink.mil/cio-nii/docs/NetCentricAttributesOfficial.pdf .
R1181	*Global Information Grid (GIG) Key Interface Profiles (KIPs) Framework* (DRAFT), Version 0.95, 7 October 2005.
R1190	DoD CIO Memorandum , **Internet Protocol Version 6 (IPv6) Interim Transition Guidance** . [http://www.intelink.gov/inteldocs/view.php?fDocumentId=153740] Note: Intelink access may require using a .mil domain connection or DoD Common Access Card.
R1191	DoD Directive O-8530.1, *Computer Network Defense*
R1192	DoD Instruction O-8530.2, *Support to Computer Network Defense Services (CNDS)*
R1194	DoD Directive 5000.01, Enclosure 1, Paragraph E1.9, Information Assurance Acquisition managers shall address information assurance requirements for all weapon systems; Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance systems; and information technology programs that depend on external information sources or provide information to other DoD systems. DoD policy for information assurance of information technology, including NSS, appears in DoD Directive 8500.01.
R1197	DoD Directive 8500.01E, *Information Assurance (IA)*, 24 October 2004 (certified current as of 23 April 2007). This directive establishes policy and assigns responsibilities under 10 U.S.C. 2224 to achieve Department of Defense information assurance (IA) through a defense-in-depth approach that integrates the capabilities of personnel, operations, and technology, and supports the evolution to net-centric warfare.
R1198	DoD Instruction 8500.2, **Information Assurance (IA) Implementation** . [http://www.dtic.mil/whs/directives/corres/pdf/850002p.pdf]
R1199	DoD Instruction 8580.1, *Information Assurance (IA) in the Defense Acquisition System* ***This instruction implements policy, assigns responsibilities, and prescribes procedures necessary to integrate Information Assurance (IA) into the Defense Acquisition System; describes required and recommended levels of IA activities relative to the acquisition of systems and services; describes the essential elements of an Acquisition IA Strategy, its applicability, and prescribes an Acquisition IA Strategy submission and review process.***
R1205	DoD ASD(NII)/CIO , **The Department of Defense Internet Protocol Version 6 Transition Plan** . [https://www.us.army.mil/suite/doc/14185947] Note that access requires DoD CAC and DKO account.
R1206	DoD Instruction 8520.2; 1 April 2004; *Public Key Infrastructure (PKI) and Public Key (PK) Enabling*; http://www.dtic.mil/whs/directives/corres/pdf/852002p.pdf
R1217	DoD 8320.02-G, 12 April 2006, *Guidance for Implementing Net-Centric Data Sharing*; http://www.dtic.mil/whs/directives/corres/pdf/832002g.pdf

R1232	DoD Directive 5230.09, *Clearance of DoD Information for Public Release*, 22 August 2008
R1256	International Organization for Standardization (ISO) Open Systems Interconnection Basic Reference Model (OSI Model)
R1258	Assistant Secretary of Defense for Networks and Information Integration, Memorandum; *Joint Net-Centric Capabilities*, 15 July 2003
R1259	Defense Information Systems Agency, Net-Centric Enterprise Services (NCES) Program Management Office, http://www.disa.mil/nces/index.html
R1280	Universal Description, Discovery, and Integration (**UDDI**), http://www.oasis-open.org/committees/uddi-spec
R1291	DoD Instruction 8510.01, DoD Information Assurance Certification and Accreditation Process (DIACAP), 28 November 2007; available at http://www.dtic.mil/whs/directives/corres/pdf/851001p.pdf (superseded DoD Instruction 5200.40, DITSCAP)
R1308	OAISIS None , **Reference Architecture for Service Oriented Architecture Version 1.0 Public Review Draft 1** . [http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/soa-ra-pr-01.pdf]
R1312	*DoD Net-Centric Data Strategy*, DoD Chief Information Officer, 9 May 2003; http://www.defenselink.mil/cio-nii/docs/Net-Centric-Data-Strategy-2003-05-092.pdf
R1313	*DoD Net-Centric Services Strategy*, DoD CIO, 4 May 2007, http://www.defenselink.mil/cio-nii/docs/Services_Strategy.pdf
R1314	Internet Corporation for Assigned Names and Numbers, http://www.icann.org/
R1315	, **Server Load Balancing** Bourke, Tony . [http://oreilly.com/catalog/9780596000509/?CMP=OTC-KW7501011010&ATT=serverload]
R1316	Blue Coat *Web Applications* (Optimization Content partial source http://www.bluecoat.com/solutions/enterprise/controlperformance/webapplications)
R1317	NIST, *Cryptographic Algorithms and Key Sizes for Personal Identity Verification*, http://csrc.nist.gov/publications/nistpubs/800-78-1/SP-800-78-1_final2.pdf
R1318	NIST, *A Comparison of the Security Requirements for Cryptographics Modules in FIPS 140-1 and FIPS 140-2*, http://csrc.nist.gov/publications/nistpubs/800-29/sp800-29.pdf
R1319	*Dynamic Host Configuration Protocol*, RFC 2131, http://tools.ietf.org/html/rfc2131, March 1997, Internet Engineering Task Force (IETF) Network Working Group
R1320	*Domain Names - Implementation and Specification*, RFC 1035, http://tools.ietf.org/html/rfc1035, November 1987, Internet Engineering Task Force (IETF) Network Working Group
R1321	*DNS Extensions to Support IP Version 6*, RFC 3596, http://tools.ietf.org/html/rfc3596, October 2003, Internet Engineering Task Force (IETF) Network Working Group
R1322	*Network Time Protocol (Version 3) Specification, Implementation and Analysis*, RFC 1305, http:// www.ietf.org/rfc/rfc1305.txt, March 1992, Internet Engineering Task Force (IETF) Network Working Group

R1323	*Internet Time Synchronization: The Network Time Protocol*, RFC 1129, http://www.ietf.org/rfc/rfc1129.pdf, October 1989, Internet Engineering Task Force (IETF) Network Working Group
R1324	*Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI*, RFC 4330, http://tools.ietf.org/rfc/rfc4330.txt, January 2006, Internet Engineering Task Force (IETF) Network Working Group
R1325	Dynamic Updates in the Domain Name System (DNS UPDATE), RFC 2136, http://tools.ietf.org/html/rfc2136, April 1997, Internet Engineering Task Force (IETF) Network Working Group
R1326	*The DHCP Handbook*, ISBN: 1-57870-137-6, 1999, Ralph Droms, Ted Lemon, The Mcmillan Technical Publishing, Indianapolis, IN, USA
R1327	*DHCP Options and BOOTP Vendor Extensions*, RFC 2132, http://tools.ietf.org/html/rfc2132, March 1997
R1328	*DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*, http://tools.ietf.org/html/rfc3646, RFC 3646, December 2003
R1329	DoD None , **Ports, Protocols, and Services Management (PPSM)** DISA . [http://iase.disa.mil/ports/index.html]
R1330	DoD Memorandum , **Encryption of Sensitive Unclassified Data at Rest on Mobile Computing Devices and Removable Storage Media** Chief Information Officer . [http://iase.disa.mil/policy-guidance/dod-dar-tpm-decree07-03-07.pdf]
R1335	DoD Deputy CIO None , **Defense Enterprise Information Architecture** . [http://www.defenselink.mil/cio-nii/sites/diea/overview.html]
R1336	The Internet Society (ISOC) , **The Transition to IPv6** None . [http://www.isoc.org/briefings/006/isocbriefing06.pdf]
R1337	Internet Engineering Task Force (IETF) Network Working Group , **Terminology for Policy-Based Management RFC3198** . [http://tools.ietf.org/html/rfc3198]
R1339	Committee on National Security Systems (CNSS) Instruction 4009, **National Information Assurance (IA) Glossary** . [http://www.cnss.gov/Assets/pdf/cnssi_4009.pdf]
R1341	DoD Directive 8000.01, **Management of the Department of Defense Information Enterprise** ASD(NII)/DoD CIO . [http://www.dtic.mil/whs/directives/corres/pdf/800001p.pdf]
R1342	DoD CIO None , **Department of Defense Global Information Grid Architectural Vision** . [http://cio-nii.defense.gov/docs/GIGArchVision.pdf]
R1343	DoD None , **Net-Centric Environment Joint Functional Concept** . [http://www.dtic.mil/futurejointwarfare/concepts/netcentric_jfc.pdf]
R1344	DoD Directive 5144.1, **Assistant Secretary of Defense for Networks and Information Integration/DoD Chief Information Officer (ASD(NII)/DoD CIO)** . [http://www.dtic.mil/whs/directives/corres/pdf/514401p.pdf]
R1345	ASD(NII)/DoD CIO None , **Deputy Assistant Secretary of Defense for Cyber, Identity, and Information Assurance Strategy** . [http://cio-nii.defense.gov/docs/DoD_IA_Strategic_Plan.pdf]

R1347	OASIS , **Web Services Business Process Execution Language (WS-BPEL) Version 2.0** . [http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.pdf] OASIS Standard; accessed 6 August 2010 via http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html.
R1352	Intelink , **Approved for Use - GTG-Online | KnowledgeTree** . [https://www.intelink.gov/inteldocs/browse.php?fFolderId=69793] Web page (accessed 8 September 2010) contains links to GIG Technical Profiles (GTPs) approved for use; accessing Intelink from outside the **`.mil`** or **`.gov`** domains may require user registration.