DOMAIN NAME SYSTEM (DNS) SECURITY REQUIREMENTS GUIDE (SRG) TECHNOLOGY OVERVIEW

Version 2, Release 4

23 October 2015

Developed by DISA for the DoD

Trademark Information

Names, products, and services referenced within this document may be the trade names, trademarks, or service marks of their respective owners. References to commercial vendors and their products or services are provided strictly as a convenience to our users, and do not constitute or imply endorsement by DISA of any non-Federal entity, event, product, service, or enterprise.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

1. INTRODUCTION

1.1 Executive Summary

This Domain Name System (DNS) Security Requirements Guide (SRG) provides the technical security policies and requirements for applying security concepts to systems.

This document details DoD DNS security practices and procedures applicable to all DoD name servers, including authoritative and recursive servers. The requirements are relevant to all name servers connected to the DoDIN.

The DNS SRG is being developed based upon the Application Core SRG requirements. In addition, the NIST Special Publication (SP) 800-81 rev2 is used as a basis for DNS deployment best practices, encryption algorithms, guidelines on using DNS Security Extensions (DNSSEC) digital signatures for DNS query/response and TSIG (hash-based Transaction SIGnature) for authenticating zone updates.

This SRG does not address the DNS configuration of DNS clients (i.e., the workstations, servers, and network devices that query name servers). Each of these DNS resolver clients' security posture should be validated with the STIG for the underlying technology or operating system.

It is assumed the base platform, on which the DNS server software is installed, is STIG-compliant.

1.1.1 Security Requirements Guides (SRGs)

SRGs are collections of requirements applicable to a given technology family. SRGs represent an intermediate step between Control Correlation Identifiers (CCIs) and Security Technical Implementation Guides (STIGs). CCIs represent discrete, measurable, and actionable items sourced from Information Assurance (IA) controls defined in a policy, such as the National Institute of Standards and Technology (NIST) Special Publication (SP) 800-53. STIGs provide product-specific information for validating and attaining compliance with requirements defined in the SRG for that product's technology area.

There are four core SRGs: Application, Network, Operating System, and Policy. Each addresses the applicable CCIs in the context of the technology family. Subordinate to the core SRGs, there are Technology SRGs developed to address the technologies at a more granular level.

This [Technology] SRG is based on the [Parent SRG]. This [Technology] SRG contains general check and fix information that can be utilized for products for which STIGs do not exist.

The STIGs based on this SRG will provide the product-specific technical implementation guidance for that product. The STIG will contain the specific check and fix information for the product it covers.

SRG Hierarchy example:

Application SRG
|__Database SRG
|__MS SQL Server 2005 STIG

The SRG relationship and structure provides the ability to identify requirements that may be considered not applicable for a given technology family and provide appropriate justification. It also provides the structure to identify variations in specific values based on the technology family. These variations will be captured once and will propagate down to the Technology SRGs and then to the STIGs. This will eliminate the need for each product-specific STIG to address items that are not applicable.

1.1.2 SRG Naming Standards

In an effort to establish consistency across the SRGs, a naming standard for the Group Title and STIGIDs has been established.

Technology SRG Naming Standards

For Technology SRG Group Title and STIGIDs the following applies:

{Core SRG value}-{*Technology SRG*}-{5- or *6-digit numeric sequence number*}

Examples:

SRG-NET-000001-RTR-000001
SRG-APP-000001-COL-000001
SRG-NET-000001-VVSM-00001
SRG-OS-000001-UNIX-000001

Checks/Fixes will be included at this level in a general form. These checks and fixes will apply for any STIGs that are created for products that do not have product-specific check and fix guidance.

1.2 Authority

DoD Instruction (DoDI) 8500.01 requires that "all IT that receives, processes, stores, displays, or transmits DoD information will be […] configured […] consistent with applicable DoD cybersecurity policies, standards, and architectures" and tasks that Defense Information Systems Agency (DISA) "develops and maintains control correlation identifiers (CCIs), security requirements guides (SRGs), security technical implementation guides (STIGs), and mobile code risk categories and usage guides that implement and are consistent with DoD cybersecurity policies, standards, architectures, security controls, and validation procedures, with the support of the NSA/CSS, using input from stakeholders, and using automation whenever possible." This document is provided under the authority of DoDI 8500.01.

Although the use of the principles and guidelines in these SRGs/STIGs provide an environment that contributes to the security requirements of DoD systems, applicable NIST SP 800-53 cybersecurity controls need to be applied to all systems and architectures based on the Committee on National Security Systems (CNSS) Instruction (CNSSI) 1253.

1.2.1 Relationship to STIGs

The SRG defines the requirements for various technology families, and the STIGs are the technical implementation guidelines for specific products. A single SRG/STIG is not all-inclusive for a given system, which may include, but is not limited to: Database, Web Server, and Domain Name System (DNS) SRGs/STIGs. For a given system, compliance with all (multiple) SRGs/STIGs applicable to a system is required.

1.3 Vulnerability Severity Category Code Definitions

Severity Category Codes (referred to as CAT) are a measure of vulnerabilities used to assess a facility or system security posture. Each security policy specified in this document is assigned a Severity Category Code of CAT I, II, or III.

Table 1-1: Vulnerability Severity Category Code Definitions

	DISA Category Code Guidelines
CAT I	Any vulnerability, the exploitation of which will, **directly and immediately** result in loss of Confidentiality, Availability, or Integrity.
CAT II	Any vulnerability, the exploitation of which **has a potential** to result in loss of Confidentiality, Availability, or Integrity.
CAT III	Any vulnerability, the existence of which **degrades measures** to protect against loss of Confidentiality, Availability, or Integrity.

1.4 SRG and STIG Distribution

Parties within the DoD and Federal Government's computing environments can obtain the applicable STIG from the Information Assurance Support Environment (IASE) website. This site contains the latest copies of any STIGs, SRGs, and other related security information. The address for the IASE site is http://iase.disa.mil/.

1.5 Document Revisions

Comments or proposed revisions to this document should be sent via email to the following address: disa.stig_spt@mail.mil. DISA will coordinate all change requests with the relevant DoD organizations before inclusion in this document. Approved changes will be made in accordance with the DISA maintenance release schedule.

1.6 Other Considerations

DISA accepts no liability for the consequences of applying specific configuration settings made on the basis of the SRGs/STIGs. It must be noted that the configurations settings specified should be evaluated in a local, representative test environment before implementation in a production environment, especially within large user populations. The extensive variety of environments makes it impossible to test these configuration settings for all potential software configurations.

For some production environments, failure to test before implementation may lead to a loss of required functionality. Evaluating the risks and benefits to a system's particular circumstances and requirements is the system owner's responsibility. The evaluated risks resulting from not applying specified configuration settings must be approved by the responsible Authorizing Official. Furthermore, DISA implies no warranty that the application of all specified configurations will make a system 100% secure.

Security guidance is provided for the Department of Defense. While other agencies and organizations are free to use it, care must be given to ensure that all applicable security guidance is applied both at the device hardening level as well as the architectural level due to the fact that some of the settings may not be able to be configured in environments outside the DoD architecture.

2. ASSESSMENT CONSIDERATIONS

2.1 NIST SP 800-53 Requirements

All applicable baseline technical NIST SP 800-53 requirements and security best practice requirements are included in this SRG.

CNSSI 1253 defines the required controls for DoD systems, based on confidentiality, integrity, and availability (baseline) of the given information system. In all cases, CNSSI 1253, along with required baselines, will serve as the policy requirement for any given asset or information system.

2.2 General Procedures

This SRG has procedures that are intended to provide appropriate evaluation and remediation functions for a typically configured system. These procedures are not product specific and are intended for use when a product-specific STIG is not available.

The SRG contains specific wording in the check and fix sections to indicate the scope of each requirement as it pertains to the technology-level STIG.

The term “DNS application” is the software specific to DNS server (e.g., Berkeley Internet Name Domain (BIND)) not inclusive of other platform elements.

The term “underlying platform” consists of the hardware, OS, and supporting applications distributed with the OS and possibly third-party software on which the DNS application runs.

The term “DNS implementation” indicates a particular DNS server product from a vendor. This includes the DNS application and, depending on the design, may include the underlying platform or specific configuration documentation and tools to properly install and run one or more DNS servers.

The term “DNS server” is an installed and configured instance of a DNS implementation. The term “DNS system” refers to one or more DNS servers running on a particular platform and the underlying platform’s OS and tools.

2.3 Additional References

Additional information regarding deploying, configuring and implementing a secure DNS can be found in NIST SP 800-1 rev2, from which much of this SRG was written. SP 800-81 rev2 provides more detailed guidance on implementing DNSSEC, specific to different DNS software products.

In addition, the DoD Network Information Center website has an overview of DNS (https://www.nic.mil/webmenu/docfiles/DNS_Overview.pdf) and a general FAQ regarding deployment of DNSSEC in the DoD (https://www.nic.mil/webmenu/docfiles/dnssec_faq.html).

Another source of information is the DNSSEC website, found at http://www.dnssec.net.

3. CONCEPTS AND TERMINOLOGY CONVENTIONS

3.1 Zones

Most users of IP-based applications are very familiar with domains and their use, but zones are actually the more relevant building blocks of the naming structure in DNS. A zone is a logical group of network devices and may be an entire domain, a domain with all of its sub-domains, or a portion of a domain.

The rationale for the existence of zones is that they make it easier to distribute the name database. Just as it would not be practical for the entire Internet DNS to reside on a single server, it also does not make sense to assign a unique name server for each lower-level domain. Zones allow network architects and administrators to combine domains in ways that optimize the management of a given portion of the name space.

3.2 Name Servers

A name server's primary function is to respond to client queries for information from the DNS. Although name servers can be configured in a wide variety of ways, there are essentially two types of name servers:

- Authoritative: Those that store zone records for one or more DNS zones.
  - An authoritative name server answers authoritatively when asked for information about records in one of its zone files. It consults its internal data to answer incoming requests rather than trying to pass them on to another server.
- Caching: Those that provide responses based on a series of queries for records stored on other name servers or from a cache of responses built by using previous queries.
  - A caching name server caches the queries it resolves on behalf of clients. Most computers on the Internet do not search for and query authoritative name servers directly but instead are configured to query from one to three caching name servers that do this work on their behalf. Caching name servers are also referred to as recursive or resolving name servers.

Importantly, a name server can be simultaneously both an authoritative and a caching name server; however, this is normally not considered a secure configuration. When an authoritative name server is also a caching server, both authoritative and external records are stored in the cache. This configuration means that if an adversary is able to corrupt or poison the cache through external queries, the adversary may be able to modify the name server's responses to legitimate queries for authoritative records. When the authoritative function is separated from the resolving function, as current best practice dictates, this risk is eliminated. While it is possible, with appropriate controls and expert administrative skills, to securely run a dual authoritative/caching name server, it is impossible to adequately describe such configurations in a manner that can be objectively validated. For this reason, this SRG generally prohibits "dual mode" (simultaneously authoritative and recursive) configurations.

However, administrators should note that caching name servers are permitted (and may be required) to be authoritative for several specific zones and names (e.g., localhost, 127.in-addr-arpa, etc.) for the purposes of preventing such private/internal queries from leaking to the Internet. Caching servers may also be configured to authoritatively deny the existence of any name/class/type Resource Record set (RRSet).

3.2.1 Authoritative Name Servers

For fault tolerance and continuity of operations, most zones have two or more authoritative servers. The authoritative server where the master copy of the zone data is maintained, is called the master (primary) name server, hereafter simply called the master name server. It loads the zone contents from a local zone file that the DNS database administrator creates and edits.

Each master for a zone may have one or more associated slave (secondary) servers, hereafter called slaves. A slave server is also authoritative for that zone but gets its updates from the master server using a replication process known as a zone transfer. Typically, each zone has only one master from which all slave name servers obtain updates. There are, however, multiple configuration options for any DNS architecture.

3.2.2 Caching Name Servers

A caching name server can resolve client queries using one of two mechanisms:

- Forwarding
- Recursion

The forwarding feature provides the ability for the name server to simply forward the request or query on to another caching server without any additional effort. With recursion, the name server recursively searches for another name server that has authoritative name information to get the records that the client (resolver) requested.

In this document, servers that support forwarding are referred to as forwarding servers. Similarly, servers that support recursion are called recursive servers. Some servers can be configured to be recursive servers when responding to most queries yet also forward specific queries to other servers.

Caching servers store the results of the lookups they perform in memory. This process, known as caching, improves performance because frequently queried records are readily available. Caching also reduces network traffic since the zone's authoritative name servers do not need to be queried as often. Cached records are retained only for a limited period of time set by the record originator. Once this time expires, the records are discarded from the cache, and the caching server will attempt to retrieve them at next query from the original source. This process prevents obsolete data from remaining past its ‘use by’ date.

When a caching name server attempts to resolve a host name, it first searches its cache for the lowest level information in the DNS hierarchy it can find. For example, if trying to resolve

example.thisdomain.mil, the name server first checks if there is a Resource Record (RR) known as an Address (A) record for that name in the cache. If there is not an A record, then the name server checks whether there is a Name Server (NS) resource record for example.thisdomain.mil in the cache. If there is an NS record, then it sends a query to the authoritative server for example.thisdomain.mil. If there is not an NS record, then it searches for an authoritative name server for thisdomain.mil, and so on.

Upon system initialization, the caching name server has no previously stored information in its cache and thus must start with the root authoritative name servers. Knowledge of the root name servers is located in the "root hints", a file that is included in the installation of the name server software and can be updated if IP addresses of the root servers change.

3.3 Resolvers

While their configuration is not covered in this SRG specifically, another key component of DNS is the client resolver/stub resolver; this discussion is offered as a point of reference. Client/stub resolvers formulate and send DNS queries to name servers. Most resolvers are rather simple, as would be found on a typical desktop or server computer system that uses DNS only as a client. They merely send host names to caching name servers and wait for the response. The resolvers are typically configured with a list of two or three caching name servers, thereby ensuring there is a fallback in case the first name server is unavailable.

In the future, resolvers may perform more complex functions. For example, they may sign queries if the caching name server requires authentication of requests. They may also request that replies be signed and the signatures be validated. This functionality is not available in common commercial resolvers today.

4. DNS ARCHITECTURE AND GENERAL SECURITY REQUIREMENTS

DNS security can be addressed at many different levels ranging from enterprise architecture to operating system parameters. Some aspects of the DNS protocols require strict consistency across all DNS products and operating systems; otherwise, the wide variety of hosts on the Internet and within enterprise networks could not discover the host names and IP addresses they need to function properly. Other portions of the DNS, particularly with regard to security configuration, allow for differences among DNS software products.

4.1 Enterprise DNS Security Initiatives

DISA is currently involved in several initiatives designed to provide increased protection for DNS services within the NIPRNet. These initiatives are in various phases of deployment and are not, at the time of this writing, fully operational. This being the case, while there are no requirements in the SRG that will either inhibit or compromise these initiatives, there are also no requirements that directly address their use. Once the initiatives are fully operational, their use will become compulsory, and modifications to this SRG will be made accordingly.

4.1.1 Enterprise Recursive Service (ERS)

DNS queries to external computers are permitted from any computer on the NIPRNet. There were previously no controls in place to determine that the traffic traversing the Internet Access Points (IAPs) was in fact DNS traffic. This situation could allow malicious entities within DoD to transfer sensitive data anywhere in the world.

The Enterprise Recursive Service (ERS) is designed to provide increased protection for DNS services within the NIPRNet by allowing only ERS systems access to the Internet and ensuring that only legitimate DNS traffic utilizes port 53.

While most CC/S/A/FAs (Combatant Commands/Services/Defense Agencies/DoD Field Activities) on the NIPRNet host their own recursive name server, these recursive name servers are no longer permitted the capability of sending DNS queries to name servers on the Internet.

The use of ERS is mandatory, and the necessary configuration settings include implementing a default-deny policy at the Integrated Service Routers (ISRs) enforcing outbound UDP and TCP port 53 (DNS) traffic to be routed to the ERS constellations.

4.2 Name Server Operating System Platforms

Another critical component of DNS security is the security of the operating system (OS) platforms on which the DNS software runs. If it is not possible to secure the OS, then DNS itself cannot be secure. Accordingly, organizations must select an appropriate OS for its name servers, one that has a well-documented, secure configuration. DNS server software should only run on approved operating systems as defined by the appropriate OS STIG.

Even a securely configured operating system is vulnerable to the flaws of the programs and applications that run on it. To prevent DNS software from being subject to the vulnerabilities of other programs, it is best not to run other programs at all. At a minimum, run only those programs that are necessary for either OS or DNS support, including those required to comply with applicable STIGS. In other words, in a properly secured environment, a name server would not run on the same device that also provides users web, email, firewall, or database services.

4.3 Redundancy, Dispersal, and Availability

A critical component of securing an information system is ensuring its availability. The best way to ensure availability is to eliminate any single point of failure in the system itself and in the network architecture that supports it.

Fortunately, the inherent design of DNS supports a high-availability environment. Master and slave servers regularly communicate zone information, so if any name server is disabled at any time, another can immediately provide the same service. The task for the network architect is to ensure that a disaster or outage cannot simultaneously impact both the master and all of its slave servers.

The solution is to disperse name servers in such a way as to avoid single points of failure. At a minimum, authoritative name servers for the same zone should be on different network segments so that at least one name server is available in the event that a router or switch fails. This fault tolerance should also extend to wide area data communications lines. For example, if a site has multiple leased lines connecting the network on which the name server resides to a larger network, such as the NIPRNet, routing protocols should be configured in such a way that if one of the lines fails, another one will still be available to support the name server.

Organizations should also be prepared for greater disasters, such as the destruction of a building, an entire campus, or in the case of a hurricane, an entire city. In situations in which all the hosts defined on an authoritative name server are located in the same building as the name server, then loss of DNS will not impact availability of service simply because the computing infrastructure is already down. On the other hand, if all the authoritative name servers for a zone reside in a single building, but hosts defined within the zone are located elsewhere, then the loss of the DNS will impact service. The loss of service occurs because users (and other infrastructure devices and servers) will not be able to resolve host names for servers/services that are otherwise still operational at an unaffected site.

Given that name servers can be dispersed across a network and still be physically located near one another, the DNS architecture should require reasonable geographic dispersal as well. Understandably, many small office sites will not have local name servers. However, if the host records for a particular site reside on a name server at a remote location, there should be a backup for that name server at an alternate location. If an organization does not have the resources for this level of dispersal, it can partner with another organization to have each organization's master name server serve as a slave for the other organization's zone. In this configuration, the name servers are both masters for some zones and slaves for others.

4.3.1 Network Related Availability

One of the most critical vulnerabilities of the DNS is that local queries for information can be affected by nonlocal communications failures. For example, an attempt to look up "local.thisdomain.mil" could be prevented by an inability to reach a root DNS or a .MIL server caused by a broken link, a routing failure, a misconfigured firewall, or other network-related issues. This can prevent even physically co-located systems from communicating with one another due to an inability to convert a configured DNS name into the IP address(es) they require to connect.

For the vast majority of applications, normal configurations of communication and server redundancy are sufficient to provide the level of service required. However, there may be circumstances where additional DNS survivability or continuity of operations will be required to meet mission-critical requirements.

4.3.2 Stub Zones

One technique to enhance survivability is "Stub Zones". Recursive resolvers can be configured with information about a specific zone, which allows the resolver to bypass normal hierarchical lookup and go directly to the zone. For example, a resolver configured with stub zone information about ".MIL" does not need to query any of the root servers about the location of .MIL name servers since that information will be maintained at/by the resolver.

The recursive resolver is configured with the name of the zone, the location of the master, and optionally a few slaves. The resolver then downloads the complete set of NS records for the zone as well as the A records for names in the NS records. These are updated and maintained on the same schedule as if the recursive resolver were a slave server for the zone.

For the purposes of this SRG, resolver operators should consider the installation of up to three classes of stub zones: first, a .MIL stub zone as described above to insulate against issues external to the DoD; second, a local stub zone covering the local network or the organizational enterprise network; third, one or more stub zones covering zones that point to systems that contain resources critical to the local mission.

Stub zones are not a panacea and do require additional administrative oversight to ensure configuration information does not become stale. Operators should balance the additional operational burden against the mission needs when deciding on the extent of stub zone implementation.

4.4 Authentication and Access Control

The general security objectives for all information systems are confidentiality, integrity, and availability. The primary objective of DNS authentication and access control is the integrity of DNS records; only authorized personnel must be able to create and modify resource records, and name servers should only accept updates from authoritative master servers for the relevant zones. Integrity is assured through authentication and access control features, largely provided by the

underlying operating system or Network Device Module (NDM), though firewalls also play a significant role in controlling DNS transactions on a network.

A secondary objective of DNS authentication and access control is the confidentiality of DNS records; only those with a legitimate need should be able to obtain the host names and IP addresses defined within a zone. Note that it can be difficult to define the extent of the legitimacy for a zone; however, both name server configuration and firewalls can support this objective. Nevertheless, an organization should never rely on restrictions to DNS host records to provide a significant safeguard to the hosts themselves since this would constitute a very thin veil of protection. The operating assumption should be that a determined adversary would be able to obtain an IP address of a host within the zone. With a defense-in-depth posture, other controls will prevent the adversary from being able to do much with this address if it were obtained.

4.4.1 Zone Updates

DNS resource records are created and modified through zone updates. This can occur either through updating of individual records on a master name server or through zone transfers from a master name server to one of its slave servers.

4.4.1.1 Updates of Individual Records

Individual resource records can be updated manually, where a DNS database administrator either edits the zone file directly or uses a tool to do so, or dynamically, where an automated process enters changes to the zone file without the intervention of the DNS database administrator.

The dynamic update capability has considerable appeal in an environment in which IP addresses change so frequently that it would be unacceptably burdensome or expensive to dedicate the time of a DNS database administrator to this function. This condition would likely be met at sites that rely on the Dynamic Host Configuration Protocol (DHCP) to assign IP addresses to client devices such as workstations, laptops, and IP telephones. It would also apply to sites that utilize frequently changing service records.

On the other hand, dynamic updates can pose a security risk if the proper security controls are not implemented. When dynamic updates are permitted without any mitigating controls, a host with network access to the name server can modify any zone record with an appropriately crafted dynamic update request.

A common solution is to require cryptographic authentication of all dynamic update requests, but not all DNS software supports this functionality. When it does not, dynamic updates must be prohibited. Name servers must support cryptographic authentication of dynamic update requests using TSIG (also referred to as Transaction Authentication), a symmetric key technology. Some DNS implementations utilize GSS-TSIG, which leverages a Kerberos-based cryptographic authentication infrastructure shared among both servers and clients.

4.4.1.2 Zone Transfers

A slave updates its zone information by requesting a zone transfer from its master server. In this transaction, the risk for the slave is that the response to its request is not in fact from its authorized master but rather from an adversary posing as the master. In this scenario, such an adversary would be able to modify and insert records into the slave's zone at will. To protect against this occurrence, the slave must be able to authenticate the master to provide assurance that any zone updates are valid.

The risk to the master server in this situation is that it would honor a request from a host that is not an authorized slave, but rather an adversary seeking information about the zone. To protect against this possibility, the master must first have knowledge of which machines are authorized slaves. Then the master must authenticate each slave when that server requests a zone transfer.

One way an organization can help prevent bogus requests for zone transfers from the master server is to hide the existence of the name server. When this technique is employed, the master name server is called a "stealth master". Hiding the stealth master involves removing its NS record from all related zone files and also ensuring that it does not appear as the master name server in any Start of Authority (SOA) records. Slaves are still configured to request zones from the master, but no hosts other than the slaves can learn the master's IP address without access to the software configuration files, which should be adequately protected. The stealth master architecture also has the advantage of improving performance because the master server is not burdened by client queries and exists only for zone updates and transfers.

4.4.2 Query Restrictions for Caching Servers

The default configuration in nearly all DNS server software is to permit any client to access any record. There are two reasons why this configuration would be considered weak security. First, some of the records may reveal information about how an internal network is configured and therefore should not be shared with external clients. Adversaries could potentially use this information to plan an attack on the internal network. Second, processing queries for any client could allow an adversary with access to the name server to exploit known or unknown flaws in the query process. The exploits might disable the name server, degrade its performance, modify the record cache, or provide the adversary with elevated privileges.

4.4.2.1 Operational Security

DNS administrators must review the contents of their zones annually, at a minimum, for content or aggregation of content that may provide an adversary information that can potentially compromise operational security. This specifically includes names that provide an outsider some indication about the function of the referenced system unless the function is obvious in the context of other standard DNS information (e.g., naming a DNS server as dns.zone.mil or an SMTP mail server as mail.zone.mil is not an Operations Security (OPSEC) violation, given that the functions of these servers are easily identifiable during DNS queries). The DNS administrator is the final adjudicator of the sensitivity of DNS information, in concert with the OPSEC processes of the organization, but should make a conscious decision to include such

information based on operational need. NIST guidance includes specific guidelines that Host Information (HINFO), Responsible Person (RP), and Location Information (LOC) records not be included in the zone.

4.4.2.2 Restrictions on Recursion

A potential vulnerability of DNS is that an attacker can poison a name server's cache by sending queries that will cause the server to obtain host-to-IP address mappings from bogus name servers that respond with incorrect information. Once a name server has been poisoned, legitimate clients may be directed to non-existent hosts (which constitutes a denial of service) or, worse, hosts that masquerade as legitimate ones to obtain passwords or other sensitive data.

To guard against poisoning, name servers authoritative for .mil domains should be separated functionally from name servers that resolve queries on behalf of internal clients. Organizations may achieve this separation by dedicating machines to each function or, if possible, by running two instances of the name server software on the same machine: one for the authoritative function and the other for the resolving function. In this design, each name server process may be bound to a different IP address or network interface to implement the required segregation.

In addition to enforcing this separation, organizations must ensure that the caching servers only accept queries from known supported clients, as they are much less likely to attempt poisoning attacks than unknown external hosts. In most cases, the hosts inside an enclave constitute the list of known supported clients. However, in some cases, the caching name server may be expected to serve clients distributed over a wide area, which is acceptable as long as the name server's audience is limited in some fashion. In no case should an internal caching name server accept queries from any Internet host.

4.4.3 Restrictions on Forwarding

The forwarding of DNS queries by a caching name server is a configuration option that tells a name server to send some or all of its queries to another caching name server instead of attempting to answer these queries with normal recursion. This type of configuration is used to change the default, recursive behavior of caching servers. This has the benefit of sharing larger caches within an organization, cutting down on upstream utilization and network usage. It also allows for more simplified, centralized management of distributed recursive service for organizations that have many smaller locations, each with its own recursive DNS server.

There are generally two different types of forwarding. The first type is forwarding all queries that arrive at one server to a set of other servers. The second type is forwarding all queries relating to a given domain to a set of other servers. Forwarding all queries is generally used to optimize the service for administration, link usage, or maintenance of other security requirements. Domain-based forwarding allows certain domains to be hidden from or changed to the clients of the caching DNS server or to change the contents of those domains.

A side effect of forwarding is that if the link between the forwarding server and the server to which queries are being forwarded is broken, DNS resolution will not work for the domain or

domains being forwarded to the remote server. Query forwarding also allows the administrators of the remote server to change the DNS responses that are received by the clients of the forwarding servers. DNS forwarding can also add delays in response times.

Organizations need to carefully configure any forwarding that is being used by their caching name servers. They should only configure "forwarding of all queries" to servers within the DoD. Systems configured to use domain-based forwarding should not forward queries for mission-critical domains to any servers that are not under the control of the US Government.

4.4.4 Firewalls and DNS

Firewalls are an important component of a defense-in-depth protection strategy for DNS. One design consideration is the location of name servers relative to the location of firewalls within a network topology.

An authoritative name server for internal zones, handling queries from internal hosts, must never be on the external (or untrusted) side of a firewall. This would make the DNS server an untrusted device by definition. Furthermore, placing the server external to the firewall would render it vulnerable to attack because it would not have the benefit of firewall protection.

The use of firewalls should be in addition to hardening the DNS server system. Neither the firewall protection nor the hardening of the server should be taken as providing absolute protection for the name server.

Similarly, a name server that handles queries from external hosts (outside the protected Enclave and visible to the Internet community) must never be on an internal network.

Externally accessible name servers must reside within a perimeter network, a DMZ, where firewalls can monitor and block traffic to and from the name servers and the outside world, while also ensuring that external hosts do not directly communicate with internal hosts.

On the other hand, if the name server provides authoritative information exclusively for internal hosts, then it should reside on the internal side of the firewall; it cannot be reached from the outside.

An example of protection from the Internet community is the Enterprise Recursive Service (ERS), which is intended to isolate the inner workings of the DoD DNS from the Internet by deploying proxy servers and enterprise recursive servers; ERS enables the authentication of transactions between DNS servers by deploying DNSSEC.

A firewall administrator will need to configure the firewall to support DNS transactions (UDP and TCP 53). The administrator should ensure that this traffic is limited to authorized name servers; in particular, that inbound Port 53 requests to other hosts must be prohibited. The firewall administrator may also need to configure the firewall to support secure shell (TCP 22) and control messages for the desired Remote Name Daemon Control (RNDC) port or an acceptable alternative means of remote administration of the name server. RNDC should only be

utilized over a secure encrypted communication path. The exact specification of these rules is beyond the scope of this SRG.

4.5 Logging

DNS software administrators require DNS transaction logs for a wide variety of reasons, including troubleshooting, intrusion detection, and forensics. These logs should be appropriately secured, having file permissions that restrict unauthorized changes or viewing, and archived, being appropriately backed-up and stored in order for them to be examined at a future time. Numerous software products are available to aid the DNS software administrator in examining these transaction logs.

4.6 Zone Files

Many implementations of DNS store zone information in text files, while others store this information in databases primarily accessed through GUI utilities. As both methods are acceptable, regardless of the format, there are basic practices that DNS database administrators should follow when managing the zones for which they have been assigned responsibility. DNS administrators also have the responsibility to ensure that DNS data is escrowed in order to cover the possibility of catastrophic failure. This data must be included in the disaster recovery plan.

4.6.1 Change and Ownership Documentation

A zone file should contain adequate documentation that would allow an IAO or newly assigned administrator to quickly learn the scope and structure of that zone. In particular, each record (or related set of records, such as a group of desktops) should be accompanied by a notation of the date the record was created, modified, or validated and record the owner's name, title, and organizational affiliation. The owner of a record is an individual with the authority to request that the record be modified or deleted.

This information will help administrators and auditors verify that the zone records are current and that only authorized personnel modify them. If records are not current, there is the potential that an adversary could simulate the activities of a retired host in order to capture logon credentials and other information. For example, presume a user has a bookmark for a retired web server in his browser. If the record for the server is not removed from DNS, a perpetrator could standup another server to mimic the behavior of the retired server, which users may still attempt to access because they may not have deleted or updated the bookmark for that server.

4.6.2 Zone-Spanning Records and Glue

If a name server were able to claim authority for a resource record in a domain for which it was not authoritative, this would pose a security risk. In this environment, an adversary could use illicit control of a name server to impact IP address resolution beyond the scope of that name server (i.e., by claiming authority for records outside of that server's zones). Fortunately, most DNS implementations do not allow this behavior. Nevertheless, the best way to eliminate this

risk is to eliminate from the zone files any records for hosts in another zone. The key exceptions to this rule involve glue for NS records and CNAME records for legacy resolution support and usage in support of DoD-approved commercial cloud service providers.

Glue is a term used for a situation in which A records for a delegated zone's name servers appear in the zone file of the parent zone. This is illustrated in the following hypothetical excerpt from the zone file for thisdomain.mil:

```
example    IN    NS    ns1.example.thisdomain.mil
           IN    NS    ns2.example.thisdomain.mil

ns1.example.thisdomain.mil.    IN    A    132.40.11.2
ns2.example.thisdomain.mil.    IN    A    132.40.15.3
```

In this instance, the zone file for thisdomain.mil is authoritative for ns1 and ns2 in example.thisdomain.mil, a different zone. Yet if the zone file did not include A records for ns1 and ns2, any client seeking records in example.thisdomain.mil would be unable to reach that zone. Hence, the glue connecting example.thisdomain.mil and thisdomain.mil is a necessary exception to the idea that one zone should not contain authoritative records for another zone.

Another situation in which canonical names from one zone might appear in the zone file of a different zone is in the case of aliases. Imagine the fictitious "example" branch office closed and all of its resources were migrated to the Systems Management Center (SMC) in Oklahoma City. In this situation, the DNS database administrator might retire the example.thisdomain.mil domain but still keep its zone operational until the user community has learned of the changes. The DNS database administrator can replace the A records in example.thisdomain.mil domain with CNAME records for the new servers in okc.thisdomain.mil as is shown in the following hypothetical excerpt from the example.thisdomain.mil zone file:

```
$ORIGIN example.thisdomain.mil
jupiter          IN    CNAME      jupiter.okc.thisdomain.mil.
saturn           IN    CNAME      saturn.okc.thisdomain.mil.
```

In general, zone-spanning aliases should be temporary (e.g., to facilitate a migration). When a host name is an alias for a record in another zone, an adversary has two points of attack: the zone in which the alias is defined and the zone authoritative for the alias's canonical name. This configuration also reduces the speed of client resolution because it requires a second look-up after obtaining the canonical name.

Note: There are certain situations where longer-lived CNAME records are appropriate. Specifically, these services have higher than normal availability requirements and/or are using caching services or are in support of commercial cloud provided services.

4.6.3 Improper NS Records and Lame Delegation

An NS record should map a domain name to an active name server authoritative for that domain. Unfortunately, in poorly configured zone files, these NS records may refer to machines that are no longer in operation or ones that do not provide name services. In some cases, they may provide authoritative name service but for different zones than the one intended in the NS record. This latter case is called lame delegation.

Poorly constructed NS records pose a security risk because they create conditions under which an adversary might be able to provide the missing authoritative name services that are improperly specified in the zone file. The adversary could issue bogus responses to queries that clients would accept because they learned of the adversary's name server from a valid authoritative name server, one that need not be compromised for this attack to be successful.

The list of slave servers must remain current within 72 hours of any changes to the zone architecture that would affect the list of slaves. If a slave server has been retired or is not operational but remains on the list, then an adversary might have a greater opportunity to impersonate that slave without detection, rather than if the slave was actually online. For example, the adversary may be able to spoof the retired slave's IP address without an IP address conflict, which would not be likely to occur if the true slave were active.

4.6.4 Root Hints

Caching name servers require a set of servers to begin the query to look up data for a resolver and then ultimately to cache that data. In order to bootstrap caching servers, information referring them to the root servers is provided since the root zone is the starting point for all zones. Modern versions of most DNS server software have this information hard-coded into the program. However, to ensure that the data is current and correct, and for possible future custom configurations, it is best to configure a "root.hints" zone into a caching name server.

When authoritative servers are sent queries for zones that they are not authoritative for, and they are configured as a non-caching server (as recommended), they can either be configured to return a referral to the root servers or to refuse to answer the query. The recommendation is to configure authoritative servers to refuse to answer queries for any zones for which they are not authoritative. This is more efficient for the server and allows it to spend more of its resources doing what its intended purpose is: answering authoritatively for its zone.

The security risk is that an adversary could change the root hints and direct the caching name server to a bogus root server. At that point, every query response from that name server is suspect, which would give the adversary substantial control over the network communication of the name server's clients. In nearly all cases, the Internet root hints are part of the installation package of the DNS software. Fortunately, the Internet root servers rarely change their addresses, so DNS software administrators will infrequently need to modify or update the root hints for servers meant to be resolving from the NIPRNet/Internet. The root hints file should contain all valid root servers, and on NIPRNet both the G and H root servers are required, at a minimum, since those servers are operated by the DoD. All DNS servers, particularly appliance-type devices, must have their default settings verified to incorporate the correct root servers and

must be reconfigured if necessary. DNS appliance devices that cannot be reconfigured as stated above may not be used.

DNS software administrators should update or verify the root hints periodically (annually should suffice) to ensure that they have current records. There are several methods for obtaining the most current root hints. It is recommended that the DNS software administrator remove the root hints file on an authoritative name server in order for it to resolve only those records for which it is authoritative; all other queries should be refused.

Bootstrapping a new caching server on the SIPRNet is problematic because the bundled root hints file will not provide appropriate information to allow the server to resolve the URL above properly. The preference would be to retrieve a copy from another known SIPRNet DNS server, using secure copy (scp) to its IP address or an out-of-band transfer method. If this is not possible, the SIPRNet Support Center (SSC) can provide the location or IP address from which to retrieve it upon request.

4.6.5 IPv6

A successful transition to IPv6 maintains the compatibility with the current installed base IPv4 infrastructure. Most nodes throughout the DoD will need both protocols to function properly for some time in the future. With this in mind, DNS can operate independent of the specific IP protocol version. DNS can answer queries to IPv6 (AAAA) requests over IPv4, or it can also answer queries to IPv4 (A) requests over IPv6.

Several technical implementations exist for transitioning IPv4 and IPv6 traffic. These mechanisms include:

- Dual IP Layer (also known as Dual Stack): This technique provides complete support for both protocols.
- Tunneling of IPv6 over IPv4: Point-to-point tunnels are created by encapsulating IPv6 packets within IPv4 headers.

Sites will decide which technique is appropriate to their specific needs. Other techniques may also become available in the future.

Resolving IPv6 addresses in DNS during the transition will require additional constraints to be considered. An inadvertent self-induced denial of service can easily occur due to records being inserted into the DNS zone. The recommendation is that AAAA records for a host should not be added to a DNS zone until the following conditions have been met:

- The address must be assigned to an interface on a host.
- The address must be configured/enabled on the host interface.
- The interface is on a link which is connected to the IPv6 infrastructure.
- In addition, if the AAAA record is added for the host, instead of for each application running on the host, all the applications on the host should be IPv6-enabled prior to adding the AAAA record.

DNS is only responsible for resolving a domain name to an IP address. Applications and operating systems are responsible for processing the IPv6 or IPv4 record that may be returned. With this in mind, a denial of service could easily be implemented for an application that is not IPv6-aware. The DNS registration of AAAA records for a host provides no knowledge of which applications on that host are IPv6-aware. If applications on such a host are not IPv6-aware, then other systems may attempt connections to that application using IPv6, and such connections will eventually time out, and the initiator will then typically retry using IPv4 or fail to connect altogether. Unless all applications on a given host are IPv6-aware, an AAAA record should not be applied to the same name as the one used for the IPv4 address on that host.

4.6.5.1 IPv6 Features

IP Version 6 is a redesign of the well-known IP Version 4. Some of the new features of IPv6 include the following:

- Increased address space: IPv6 uses 128-bit addresses versus 32-bit addresses in IPv4.
- Auto configuration for IP addresses and gateways.
- IPsec is designed into the protocol stack but must still be configured and enabled.
- Improved mobility and Quality of Service (QoS) support.

4.6.5.2 IPv6 Resource Records

Similar to the IPv4 resource record, IP Version 6 addresses are stored using AAAA (Quad-A) resource records. The format of the AAAA resource record is very similar to the IPv4 record as shown in the example below:

```
jupiter     IN    AAAA     2001:dc9::1
saturn      IN    AAAA     4321:1:2:3:4:5:678:90ab
```

The reverse lookup zones for IPv6 addresses still utilize PTR (pointer) records to resolve IP addresses to domain names. An example of the reverse record for the host ‘saturn’ is provided.

```
b.a.0.9.8.7.6.0.5.0.0.0.4.0.0.0.3.0.0.0.2.0.0.0.1.0.0.0.1.2.3.4    IN    PTR    saturn
```

If this appears to be confusing, tools can be found on the Internet to assist in the building of reverse zones for IPv6.

5. STANDARD OPERATING PROCEDURES FOR DNS

To secure DNS, a secure baseline configuration must be implemented and must be managed securely from that point forward. Monitoring DNS transactions and planning for contingencies is also crucial to a secure DNS infrastructure. This section covers these operational aspects of DNS security.

A recommended reference is RFC 2870 (Root Name Server Operational Requirements). This RFC lists mandates for root servers, but its guidance is of interest to anyone with responsibility for a name server at any level in the DNS hierarchy. The IAM at each facility should review this document and ensure all DNS administrators at the site review it as well.

Standard Operating Procedures impact individuals in a variety of DNS support roles, including security management, systems administration (business continuity and backups), DNS software administration (cryptographic key management), and DNS database administration. Each of these roles is addressed in the subsections that follow.

5.1 Security Management Responsibilities

Security management involves the tracking of security-relevant personnel assignments, physical access control, and business continuity. The ISSM will either have direct responsibilities for these areas or work closely with those that do.

5.1.1 Business Continuity

Business continuity involves planning for contingencies in which an entire site is lost due to a terrorist attack, fire, or severe natural disaster. In general, the strategy must be to move production-computing services to another location prepared to handle them. To accomplish this task, hardware, software, and data must be present or easily obtainable at the other location. Any such planning must include DNS if it is to be effective.

In the case of DNS, a well-designed architecture might have already placed an operating name server (e.g., a zone slave) at the disaster recovery site. If cost or other constraints preclude this, the ISSO must, at a minimum, ensure that an off-site copy of zone information exists (preferably in a digital format but in a hard copy format otherwise) to prevent complete loss of records in the event of a disaster.

A key component of business continuity is continuity of electrical power. For root servers, the RFC requires power continuity of at least 48 hours, which essentially necessitates on-site electric generators. Of course, not all sites have this capability or are able to procure it. Nevertheless, if all the name servers supporting a zone lose power, users may be unable to reach the hosts defined in the zone because they cannot resolve the host's name to its IP address. Therefore, name servers should have Uninterruptible Power Supply (UPS) or alternative power source similar to the hosts that they support.

5.1.1 Backup

Fortunately, the DNS architecture is such that there should always be a hot backup of zone information present whenever the master name server is unavailable for any reason (i.e., the authoritative slave server maintains a copy of the zone files on the master). This built-in redundancy, however, does not extend to configuration files and logs. Therefore, name servers should be backed up to external media on a regular basis.

At some locations, an automated enterprise backup system supports many servers. In this case, name servers can simply be added to the enterprise system. At other locations, backups must be performed manually, placing a considerably higher burden on administrators. In circumstances in which zone and configuration information is static, remaining the same for several months at a time, it would make little sense to conduct full daily backups. Backups should occur as frequently as needed to capture changes on the name server.

5.1.2 Cryptographic Key Supersession

Similar to user account passwords, cryptographic keys such as TSIG/DNSSEC keys must be changed periodically to minimize the probability that they will be compromised. If there is a known compromise of a TSIG/DNSSEC key, then it needs to be replaced immediately. One of the most important aspects of key supersession is the method that will be used to transfer newly generated keys. Possibilities, in rough order of preference, are as follows.

- SSH
- Encrypted email using DoD PKI certificates
- Secure fax (STU-III)
- Regular mail (using the expedited mailing service holding the current GSA contract for "small package overnight delivery service")
- Hand courier

The operational details of key supersession are beyond the scope of this SRG.

5.2 DNS Database Administration Responsibilities

One of the simplest changes to DNS may also be one of its greatest potential vulnerabilities, specifically, adding a host name and its associated IP address to the zone file. Without a rigorous process for adding and modifying resource records, an attacker can social engineer the system (i.e., manipulate human processes rather than circumvent technical controls). For example, an attacker might be able to simply call the DNS database administrator on the phone and successfully request that the IP address of an email server be changed to a rogue server under the attacker's control. Resource records can be modified to compromise security in countless ways.

To best assure the integrity of zone files, requests to change the DNS records should be carefully managed, and the records should be checked periodically to ensure their validity. For example, when equipment is retired, SAs often fail to remove the associated host from the DNS. Without

periodic checks, an attacker may use a retired host IP address to obtain valuable information from another user who was unaware of the change.

The details of the procedures to add or modify resource records are beyond the scope of this SRG. The requirement is that there be a written procedure in place that meets certain basic criteria.

6. DNSSEC

DNSSEC is a promising technology for authentication and integrity of DNS data. A powerful feature of DNSSEC is the ability to sign record sets to ensure their integrity and authenticity throughout the DNS infrastructure and not just between the authoritative name server and its zone partner or local client. The advantages of this feature become apparent when DoD users wish to securely validate records from other organizations, including commercial vendors, business partners, and other Government agencies. For example, suppose a user wants assurance that information obtained from what it believes to be thisdomain.mil servers are, in fact, actually thisdomain.mil servers and not an adversary's servers masquerading as thisdomain.mil servers. With DNSSEC and its use of public key cryptography, organizations can configure a trust anchor for a higher-level zone public key (e.g., the "root" or ".mil" zone) and can then use a chain of trust through DNSSEC validation to verify keys and signatures at lower levels of the DNS hierarchy. In the above example, if thisdomain.mil signs its zone and then the thisdomain.mil signing key is itself signed by .mil, then a user who trusts .mil can verify the trust relationship (and therefore the thisdomain.mil zone data), and the desired assurance is achieved. It should be noted, the client stub-resolver must be DNSSEC-aware for the authentication to be valid for end-to-end protection. Currently, most client resolvers are not DNSSEC-aware.

The objective is to ensure that DNS data from DoD and partner organizations can be verified not just when it is initially obtained from the authoritative server, but also at any later time when obtained via a caching server. However, one of the resource costs of DNSSEC will be creating the infrastructure (policies, processes, and tools) that will be needed to support DNSSEC key management, since neither the DoD PKI nor COMSEC material systems nor any future DoD Key Management Infrastructure (KMI) can be repurposed to perform DNSSEC key management.

DNSSEC implementation should be thoroughly tested in a lab environment prior to a production deployment. Some immediate items of concern for the administrator transitioning to DNSSEC include disk space for zone records and network performance. A DNSSEC zone can be seven times larger than a non-DNSSEC zone. In addition, the message sizes associated with DNSSEC increase with the additional signatures and will go beyond the payload of the UDP protocol of 512 octets or less, causing truncation. In some cases, this will require the use of TCP, which can cause significant overhead and delays.

The DoD-wide configuring and enabling of DNSSEC is currently in process. At present, the .mil top-level domain (TLD) is signed and registered with the Internet Assigned Numbers Authority (IANA).

FRAGO1 to TASKORD 11-0410-2 specified that all CC/S/FAs must implement DNSSEC on their respective second-level .mil domain by 1 May 2013. DNSSEC for all lower level .mil subdomains was directed to be implemented by 3 Jun 2014. This requirement is for Unclassified networks only; Classified networks are exempt from the DNSSEC requirements which may be marked N/A for such systems.