1

INTRODUCTION

WHY ATTACK DNS?

The Domain Name System (DNS) is fundamental to the proper operation of virtually all Internet Protocol (IP) network applications, from web browsing to email, multimedia applications, and more. Every time you type a web address, send an email or access an IP application, you use DNS. DNS provides the lookup service to translate the website name you entered, for example, to its corresponding IP address that your computer needs to communicate via the Internet.

This lookup service is more commonly referred to as a *name resolution* process, whereby a worldwide web "www" address is resolved to its IP address. And a given web page may require several DNS lookups. If you view the source of a random web page, for example, count the number of link, hypertext reference (href), and source (src) tags that contain a unique domain name. Each of these stimulate your browser to perform a DNS lookup to fetch the referenced image, file or script, and perhaps pre-fetch links. And each time you click a link to navigate to a new page, the process repeats with successive DNS lookups required to fully render the destination page.

Email too relies on DNS for email delivery, enabling you to send email using the familiar user@destination syntax, where DNS identifies the destination's IP address for transmission of the email. And DNS goes well beyond web or email address resolution. Virtually every application on your computer, tablet, smartphone, security

DNS Security Management, First Edition. Michael Dooley and Timothy Rooney.

cameras, thermostats, and other “things” that access the Internet require DNS for proper operation. Without DNS, navigating and accessing Internet applications would be all but impossible.

Network Disruption

An outage or an attack that renders the DNS service unavailable or which manipulates the integrity of the data contained within DNS can effectively bring a network down from an end user perspective. Even if network connectivity exists, unless you already know the IP address of the site to which you’d like to connect and enter it into the browser address field, you’ll be unable to connect, and you won’t see any linked images or content.

Such an event of the unavailability of DNS will likely spur a flurry of old fashioned phone calls to your support desk or call center to politely report the problem. IP network administrators generally desire to minimize such calls to the support center, polite or otherwise, given that it forces those supporting the network to drop what they’re doing and resolve the issue with the added pressure of visibility across the wider IT or Operations organization.

DNS as a Backdoor

Just as DNS is the first step in allowing users to connect to websites, it is likewise usable by bad actors to connect to internal targets within your enterprise and external command and control centers for updates and directives to perform nefarious tasks. Given the necessity of DNS, DNS traffic is generally permitted to flow freely through networks, exposing networks to attacks that leverage this freedom of communications for lookups or for tunneling of data out of the organization.

Thus, attacking DNS could not only effectively bring down a network from users’ perspectives, leveraging DNS could enable attackers to communicate to malware-infected devices within the network to initiate internal attacks, to exfiltrate sensitive information, or to perform other malicious activity. Malware-infected devices may be enlisted to serve as remote robots or *bots* under the control of an attacker. A collection of such bots is referred to as a *botnet*. A botnet enables an attacker to enlist an army of devices potentially installed around the world to perform software programmable actions.

By its very nature, the global Internet DNS system serves as a distributed data repository containing domain names (e.g., for websites) and corresponding IP address information. The distributed nature of DNS applies not only to the global geographic distribution of DNS servers, but to the distribution of administration of the information published within respective domains of this repository. DNS has proven extremely effective and scalable in practice and most people take DNS for granted given this and its historical reliability. However, its essential function and decentralized architecture serve to attract attackers seeking to exploit the architecture and rich data store for sinister activities.

While DNS is the first step in IP communications, many enterprise security strategies trivialize or startlingly even ignore its role in communications and therefore its susceptibility to attacks on this vital network service or on the network itself. Most security strategies and solutions focus on filtering "in-band" communication flow in order to detect and mitigate cyber attacks. However, as we shall see, filtering DNS traffic can support a broader network security plan in providing additional information for use in identifying and troubleshooting attack incidents. This book is intended to provide details regarding the criticality of DNS, its vulnerabilities, and strategies you can implement to better secure your DNS infrastructure, which will in turn better secure your overall network.

DNS BASIC OPERATION

Figure 1.1 illustrates the basic flow of a DNS query. Upon entry of the desired destination by name, www.example.com in this case, software called a *resolver* is invoked by the application, for example, web browser. This resolver software is typically included with the device operating system. If a connection had recently been made to this website, its IP address may already be stored in the *resolver cache*. The resolver cache helps improve resolution performance by temporarily keeping track of recently resolved name-to-IP address mappings. In such a case, the resolver may return the IP address immediately to the application to establish a connection without having to query a DNS server.

If no relevant information exists in the resolver cache the device will query its *recursive DNS server*. The role of the recursive server is to locate the answer to the

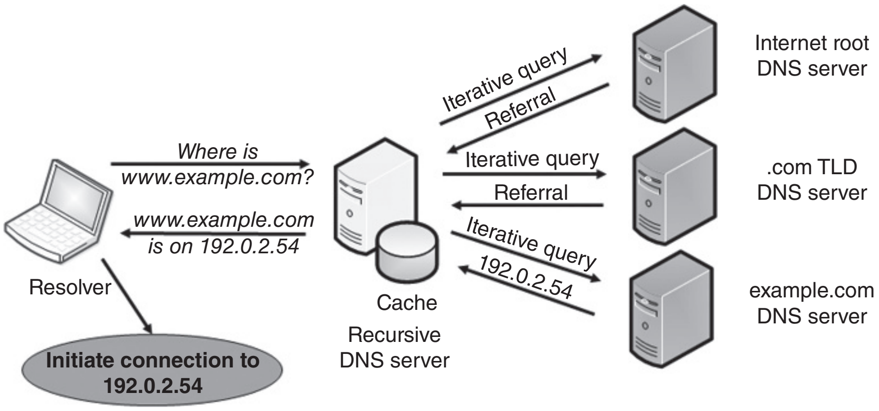

Figure 1.1. Basic DNS Resolution Flow

device's query. The recursive server is itself a resolver of the DNS query; we refer to the resolver on the originating device as a *stub resolver* as it initiates a query to its recursive server, and it relies solely on the recursive server to locate and return the answer. The stub resolver is configured with DNS server IP addresses to query as part of the IP network initialization process. For example, when a device boots up, it typically requests an IP address from a dynamic host configuration protocol (DHCP) server. The DHCP server can be configured to not only provide an IP address but the IP addresses of recursive DNS servers to which DNS queries should be directed. Use of DHCP in this manner facilitates mobility and efficiency as addresses can be shared and can be assigned based on the relevant point of connection to the IP network.

As we mentioned, the recursive DNS server's role is to resolve the query on behalf of the stub resolver. It performs this role using its own cache of previously resolved queries or by querying DNS servers on the Internet. The process of querying Internet DNS servers seeks to first locate a DNS server that is *authoritative* for the domain for which the query relates (example.com in this case) and then to query an authoritative server itself to obtain an answer that can be passed back to the client, thereby completing the resolution process. The location of the authoritative server is determined by querying Internet DNS servers that are responsible for the layers of the domain tree "above" or "to the right" of the domain in question. We'll discuss this process in more detail in Chapter 2. The recursive server caches the resolution information in order to respond more quickly to a similar query without having to re-seek the answer on the Internet.

To access your website, people need to know your web address, or technically your uniform resource locator or URL. And you need to publish this web address in DNS in the form of a *resource record* so browsers can locate your DNS servers and resolve your www address to your web server's IP address. Multiple, at least two, authoritative DNS servers must be deployed to provide services continuity in the event of a server outage. Generally, an administrator configures a *master* server that then replicates or transfers its domain information to one or more *slave* servers. We will discuss more details on this process and server roles in Chapter 2.

Basic DNS Data Sources and Flows

Figure 1.2 illustrates a subset of the various data stores for DNS data and corresponding data sources. The authoritative DNS servers must be configured to answer queries for domain name-to-IP address mappings for this domain for which they are authoritative. Depending on your DNS server vendor implementation, DNS configuration information may be supplied by editing text files, using a vendor graphical user interface (GUI) or deploying files from an IP address management (IPAM) system as shown in Figure 1.2. Each server generally relies on a configuration file and authoritative servers store DNS resolution information in zone files or a database. Some implementations utilize dynamic journal files to temporarily store DNS information

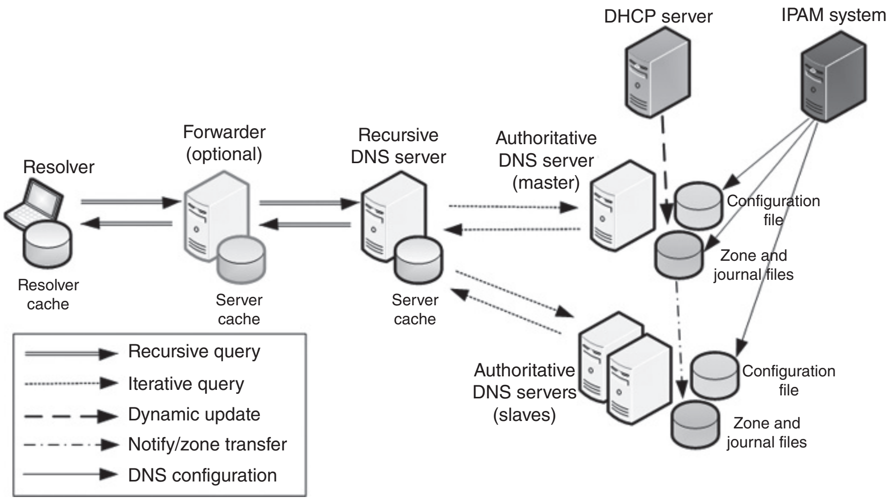

Figure 1.2. DNS Query Flow and Data Sources

updates prior to committing to zone files in an effort to improve performance. All vendor implementations feature the ability to update DNS resolution information on a given “master” server which will then replicate this information to other authoritative servers to provide redundancy of this information.

Figure 1.2 also illustrates various DNS message types that are used to query for and configure DNS information. The recursive and iterative query types enable the resolution of DNS data, while dynamic updates and zone transfers enable the dynamic updating and replication of resolution information, respectively. DNS configuration information includes the parameters of operation for the DNS server daemon as well as published resolution data within zone files. We’ll describe these in more detail in Chapters 2 and 3, but for now, you may observe that there are several independent data sources that may configure your DNS information as you permit.

DNS Trust Model

The DNS trust model refers to how DNS information flows among these components of the DNS system. In general, information received by other components in the system is trusted though various forms of validation and authentication can improve trustworthiness as we shall discuss later.

From the client resolver perspective, the client trusts its resolver cache and the recursive server to provide answers to DNS queries. Should either trusted data source be corrupted, the resolver could inadvertently redirect the user application to an inappropriate destination. For example, a user, thinking he or she is connecting to his or her bank, may inadvertently be connected to an imposter site in an attempt by an attacker to collect authentication credentials or financial information.

The recursive server trusts its cache and the various DNS servers it queries, whether internally on the enterprise network or externally on the Internet. It relies not only on accurate responses from authoritative DNS servers but on other domain servers which provide referrals to locate DNS servers authoritative for the domain in question. Referral answers are generally provided by the Internet root servers as well as top level domain ("TLD," e.g., .com, .edu, .net, etc.) servers as shown in Figure 1.1. Referrals may also be provided by other servers operated internally, externally, or by DNS hosting providers to walk down the hierarchical domain tree to locate the authoritative DNS servers. Corruption of recursive server DNS information, whether referral or resolution data, could have broader impacts affecting many clients given the caching of seemingly accurate resolution data. Each user attempting to connect to a website whose resolution information has been poisoned may be provided such falsified information from the recursive server cache.

Authoritative DNS servers are so called given that they are purportedly operated by or on behalf of the operator of a given DNS domain who is responsible for the information published on these servers. Resolvers attempting to resolve hostnames within the domain of the authoritative server trust the server to respond with accurate information, where *accurate* means *as published by the domain administrator*. Information published within authoritative DNS servers originates from a variety of sources as shown in Figure 1.2, including manually edited text files, inter-server transfers and updates, and/or use of IPAM solutions. Inter-server transfers refer to master–slave replication, while updates may originate from other DNS servers, DHCP servers, other systems, or even end user devices if permitted by administrators. Corruption of authoritative server configuration information impacts all Internet users attempting to connect with resources within the corresponding domain.

DNS Administrator Scope

As a DNS administrator, you'll generally need to be concerned first with your internal users or customers attempting to resolve domain names within your internal infrastructure and those on the Internet. For access to your internal systems, you'll need to configure authoritative DNS servers with the domain name to IP address mappings for those systems for which internal users need access. We'll refer to this naming and address mappings for internal infrastructure as your *internal namespace*.

To enable your users to access Internet websites, you'll need to manage recursive servers which your users can query to locate external authoritative servers from which to seek query answers.

You will also need to provide external Internet and extranet users with name to address mappings for your Internet reachable systems such as websites and email servers. Note that this *external namespace* will likely be a subset of your internal namespace, though ideally totally independent. We'll discuss approaches to serving these constituencies beginning with respective DNS server deployment approaches in Chapter 5.

SECURITY CONTEXT AND OVERVIEW

The practice of network security essentially boils down to the management of risks against a network. Risks may consist not only of malicious attacks but also include natural or man-made disasters, poor architecture design, unintended side effects of legitimate actions, and user error. Development of a security plan requires enumeration of risks, identification of vulnerabilities which may be exploited to affect the risk, characterization of the likelihood of each risk, determination of the impact the risk presents to the organization and defining controls to constrain the risk impact for each. Application of controls seeks to mitigate the risk to eliminate, or more likely yield a lower level of residual risk that is more tolerable to the organization.

We will apply the National Institute of Standards and Technologies (NIST) Cybersecurity Framework (1) as the context within which we discuss security approaches and strategies. The cybersecurity framework has emerged as a de facto standard worldwide. While no security framework can be "one size fits all," it provides a taxonomy and methodology for organizations to characterize their current and desired (planned) cybersecurity status, to prioritize initiatives to enable improvement of their current status toward their desired state and to communicate among stakeholders about cybersecurity risk. The framework provides guidance for organizations to perform risk assessments and to plan to manage risk in light of each individual organization's vulnerabilities, threats, and risk tolerance.

The framework relies on existing security standards including COBIT 5 (2), ISA 62443 (3), ISO/IEC 27000 (4), NIST SP 800-53 Rev4 (5) among others. It references specific sections of these supporting standards within each of the major framework activities. As such, the framework essentially provides a common overlay among these various standards to define a language for expressing and managing cybersecurity risk.

Cybersecurity Framework Overview

NIST's cybersecurity framework seeks to facilitate communications within and external to an organization when conveying security goals, maturity status, improvement plans, and risks. The framework is comprised of three major components:

- The *framework core* defines security activities and desired outcomes for the lifecycle of an organization's management of cybersecurity risk. The core includes detailed references to existing standards to enable common cross-standard categorization of activities. The core defines these activities across five functions.

  - Identify – deals with what systems, assets, data, and capabilities require protection
  - Protect – implement safeguards to limit the impact of a security event
  - Detect – identification of incidents
  - Respond – deals with security event management, containing incident impacts
  - Recover – resilience and restoration capabilities

  Each function has a set of defined categories and subcategories which we will explore later in this chapter.

- The *framework profile* defines the mechanism for assessing and communicating the current level of security implementation as well as the desired or planned level of implementation. The profile applies business constraints and priorities, as well as risk tolerance to the framework core functions to characterize a particular implementation scenario.
- The *framework implementation tiers* define four gradations of maturity level of security implementations, ranging from informal and reactive to proactive, agile and communicative.
  - Tier 1 – Partial – Informal, ad hoc, reactive risk management practices with limited organizational level risk awareness and little to no external participation with other entities.
  - Tier 2 – Risk Informed – Management approved with widely established organization-wide risk awareness but with informal and limited organization-wide risk management practices and informal external participation.
  - Tier 3 – Repeatable – Risk management practices are formally approved as policy with defined processes and procedures which are regularly updated based on changes in business requirements as well as the threat and technology landscape. Personnel are trained and the organization collaborates with external partners in response to events.
  - Tier 4 – Adaptive – Organization-wide approach to managing cybersecurity risk where practices are adapted to the changing cybersecurity landscape in a timely manner. The organization manages risk and shares information with partners.

  The implementation tiers enable an organization to apply the rigor of a selected maturity level to their target profile definition to align risk management practices to the particular organization's security practices, threat environment, regulatory requirements, business objectives, and organizational constraints.

Framework Implementation

Implementation of the cybersecurity framework entails interaction and feedback among three major organizational tiers.

- Executive level – with a focus on organizational and business risk, the executive level sets out business priorities, risk tolerance, and security budget to those in the business/process level.
- Business/Process level – in consideration of business priorities, risk tolerance, and budget, this level focuses on critical infrastructure risk management, defining a cybersecurity framework target profile for the organization based on these inputs and the current profile, allocating budget accordingly to closing gaps between these profiles. This level feeds back to the Executive level any changes in current and future risk based on security threats and technologies and provides implementation directives to the Implementation and Operations level.
- Implementation/Operations level – responsible for framework profile implementation and risk management tactics. Feedback to the business/process level includes implementation progress, issues, and changes in assets, vulnerabilities, and threats.

The cybersecurity framework document leverages this three-tier organizational structure and identifies the following basic steps in defining a cybersecurity plan:

1. The first step starts at the Executive level to identify your business and organizational priorities and objectives and risk tolerance in order to scope out in priority order the set of assets and systems within the network to focus on.
2. Within the selected scope, the second step entails the organization enumerating affected systems and assets, regulatory and legal requirements, risk tolerance, and corresponding threats and vulnerabilities associated with the scoped systems and assets.
3. This step consists of defining the current status of cybersecurity implementation. Using the framework core, you can identify your level of compliance and discipline in implementing each function category and subcategory. The resulting analysis becomes your Current Profile defining a snapshot of your organization's alignment with the framework.
4. A risk assessment should then be conducted to enumerate risks in terms of asset vulnerabilities, potential threats and respective likelihood, and the potential network and business impact of each threat.
5. The fifth step entails defining the desired cybersecurity activities and outcomes by defining the Target Profile. Using the framework core along with business-specific categories and subcategories, desired outcomes can be enumerated.

6. Comparing the Target Profile with the Current Profile, one may define the gaps which need to be addressed to evolve from the current status to the desired state. Based on the cost to implement gap closure for each category and subcategory in light of its corresponding security priority, the business can determine whether to invest in closing that gap based on the corresponding value to the business. This helps prioritize which gaps will be addressed initially, which can be addressed later, and at what cost for each from a capital, expense, and resource perspective.
7. The final step consists of formally defining and implementing an action plan to address the prioritized gaps. As implementation ensues within the Implementation/Operations level, and snapshots of current or in-progress status may be communicated by updating the Current Profile.

The Current and Target profiles enable communication within or outside an organization of its current and planned cybersecurity implementation state, respectively. The broad use of this common framework facilitates communication among these entities and stakeholders using well-defined terms.

Scoping DNS Once your executive team identifies and prioritizes DNS as within the scope of priority for applying security controls, your business team needs to define the corresponding set of affected DNS components. Table 1.1 illustrates an example scoping of basic business priorities that affect DNS to corresponding affected DNS components which could be considered for application of security controls.

TABLE 1.1 DNS Scope Examples

Broad DNS Scope	Affected DNS Components
Accurately resolving the organization's published namespace on the Internet	• Authoritative DNS servers and/or your external DNS hosting provider configured to resolve your namespace
Accurately resolving the organization's namespace for internal users	• Device stub resolvers • Recursive DNS servers configured to resolve DNS queries from internal device resolvers • Authoritative DNS servers configured to resolve your namespace for internal resolvers
Accurately resolving Internet domain names for legitimate internal user access to the Internet	• Device stub resolvers • Recursive DNS servers configured to resolve DNS queries from internal device resolvers

Current Profile Once the organization defines the scope as including one or all of these broad areas, a *Current Profile* should be developed regarding the current security level of the associated DNS components. Consider each of the categories and subcategories as it applies to your current DNS management and security policies and procedures. As with the cybersecurity framework itself, you may have additional processes or desired outcomes for consideration in your implementation.

Risk Assessment The next major step in the process comprises the risk assessment for affected DNS components. This step entails enumeration of each possible threat event. A threat event is an event that upon occurrence could impact the network and business detrimentally. Threat events may include events beyond security-related threats such as natural or man-made disasters so you may wish to consider all possible threats to secure your network and DNS in particular.

For each identified threat, consider the likelihood of the threat event occurring as well as the impact on your network and business should the threat event occur. The likelihood of a given threat event may be estimated by considering known vulnerabilities that may be exploited by an attacker to instigate the threat event. It's useful to plot each threat on a graph where the *x*-axis relates the relative impact of the threat while the *y*-axis reflects its relative likelihood. The relative impact could be estimated in terms of resource unavailability or downtime, end user or customer dissatisfaction, and/or lost revenue. Plotting risks in this manner can help you prioritize for which risks more urgent remediation is required.

As you may conclude from Figure 1.3, Risk #4 (R4) has a relatively high likelihood and high impact. This risk should likely be mitigated with the highest priority. Risk #2 of slightly lower impact and less likelihood should be next. Even though Risk #1 has a higher likelihood than Risk #2, its impact is substantially less. By applying controls, the goal is to shift each unmitigated risk down and to the left to render a lower overall residual risk to the organization.

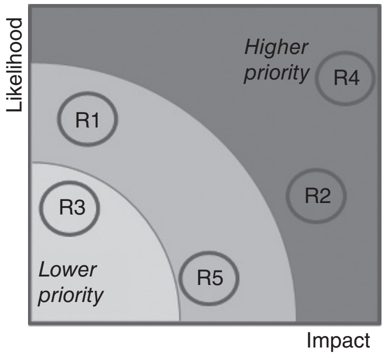

Figure 1.3. Risk Likelihood-Impact Plot

To add some structure to the process of assessing risk on a per system level, NIST published Federal Information Processing Standards (FIPS) Publication 199 (6). This publication defines standards for categorizing information and information systems based on the potential impact on the organization should certain threat events occur for use in assessing risk to an organization. Categorization is performed based on three security objectives for information and information systems.

- Confidentiality – the protection of information from unauthorized disclosure
- Integrity – protection against unauthorized modification or destruction of information
- Availability – protection against disruption of access to or use of information or system

FIPS Publication 199 defines three levels of impact on an organization for each of these objectives as follows:

- Low impact – expected to have a limited effect on the organization's operations, assets, or individuals; for example, loss of confidentiality, integrity, or availability could degrade an organization's capability though with noticeably reduced effectiveness. It could also result in minor damage to some or all of the organization's assets, minor financial loss and/or minor harm to individuals.
- Moderate impact – expected to have serious adverse impact on the organization's operations, assets, or individuals; for example, loss of confidentiality, integrity, or availability could cause significant degradation of an organization's capability though with substantially reduced effectiveness. It could also result in significant damage to the organization's assets, significant financial loss, and/or significant but not life-threatening harm to individuals.
- High impact – expected to have severe or catastrophic impact on the organization's operations, assets, or individuals; for example, loss of confidentiality, integrity, or availability could cause severe degradation of an organization's capability including the inability to perform one or more of its primary functions. It could also result in major damage to the organization's assets, major financial loss, and/or severe or catastrophic and life-threatening harm to individuals.

Categorization of each of the three objectives as low, moderate, or high is performed on various types of information at rest (e.g., within a file on a server) or in motion (e.g., within an IP packet traversing your network) through a network and on information systems themselves (e.g., servers, laptops, etc.). Examples of information types might be published DNS zone information or DNS query transaction information. The security categorization (SC) for the types and systems within your organization is represented as a tuple as illustrated in the following example:

$$\mathbf{SC}_{\text{info type/system}} = \{(\textbf{confidentiality}, \text{LOW}), (\textbf{integrity}, \text{HIGH}), (\textbf{availability}, \text{MODERATE})\}$$

For DNS, generally the highest requirement for most organizations is integrity, protecting DNS data from unauthorized changes. After all, users are relying on DNS data to enable them to connect to an intended destination. High availability likewise is paramount so that the resolution process and data is available. Confidentiality is typically lower in relative priority since DNS data generally is public information. However, many organizations publish a set of DNS information for resolution only for all or certain internal users and prohibit access for external users. In this scenario, for this type of information, confidentiality might be considered moderate if not high.

Target Profile and Security Planning Your risk assessment will provide valuable input when prioritizing security initiatives in your security plan. We've included a sample of a DNS-specific framework core in Appendix A as a starting point. You can use our example framework core or create your own. Creating a target profile using the framework core allows you to define the desired outcomes for each of the defined categories along with those you may choose to add in. The differences between your target profile and your current profile define your to-do list of tasks, implementations, and process improvements necessary to transition from your current security implementation state to your desired target state.

To mitigate a given risk, a control or set of controls may be implemented to minimize the likelihood and/or impact of a given risk. Your risk assessment results will enable you to prioritize resources and efforts to apply controls in order to mitigate higher impact and higher likelihood threat events. A *control* is an implementation of technology, processes, and/or people resources that is intended to reduce such risks. Generally, a residual risk remains, which if excessive, may behoove you to apply additional controls.

In general, the application of multiple controls yields a *defense in depth* security approach that provides multiple lines of defense for a given threat event. Should an attacker penetrate one control, another is provided to hamper the further progress of the attack. When considering a given host, for example, a DNS server, a defense in depth approach entails securing each of the layers defined in Table 1.2.

Several aspects of this defense in depth strategy are common across several elements of your network, for example, all servers require strong credentials and all remote administrator access must be encrypted. Such *common controls* provide consistent protection and should apply to your DNS servers as well.

DNS-specific controls such as those example attributes outlined above provide added protection. The NIST framework core implicitly recommends a defense in depth strategy. NIST has also published a DNS-specific guide for secure DNS deployment (7). This useful guide describes DNS-specific controls with a particular focus on securing the integrity of DNS data. This guide provides thorough procedures for securing a BIND DNS server, including configuration and management of DNS security extensions (DNSSEC). We'll refer to this guide as well throughout this book where appropriate.

TABLE 1.2 Defense in Depth Layers

Data at rest	Data residing on the host, e.g., a file on a hard drive, thumb drive, or database	Configuration files, zone files, resource records, cached data
Data in transit	Data sent or received by the host	DNS queries and responses, DNS updates, zone transfers, configuration updates
Application	Reputability of each application running on the host	ISC BIND, Unbound, NSD, PowerDNS, Knot DNS, etc., i.e., your deployed DNS server application(s)
Host hardware and operating system	Reputability of the hardware manufacturer, software (e.g., BIOS) manufacturer, kernel and operating system hardening tactics	DNS server hardware, kernel, and operating system
Internal network	Internal firewalls, host firewalls, malware presence within internal infrastructure	Permissible ports and protocols for DNS, DNS ACLs, and port ACLs
Network perimeter	Boundary between trusted and untrusted environments	Permissible ports and protocols for DNS traffic traversal
External network	Internet-based vulnerabilities	Inbound purported DNS traffic; external DNS hosting providers; domain registrar(s)
Physical security	Building/datacenter/computer access, access control, property removal policies	DNS server physical security
Operations	Adherence to security policies by people, processes, and technologies; policy verification and enforcement	DNS configuration and transaction audits; training, holistic security awareness

Your security plan should define specific control implementations designed to mitigate specific threat events. Because each planned implementation will require organizational resources with respect to personnel involvement and perhaps capital and/or expense, you will generally need to prioritize and/or implement the plan in stages over time as resources permit. Application of the NIST cybersecurity framework provides structure and common language for efficiently conveying security status and goals. It also facilitates prioritization of security gaps to enable staging of the implementation of DNS and network security controls.

WHAT'S NEXT

Chapters 2 and 3 provide an overview of how DNS works with details regarding the DNS protocol, respectively. Chapter 4 introduces major security-related threats to and vulnerabilities of the DNS system. Chapters 5–11 delve more deeply into each vulnerability and defines detection and mitigation strategies accordingly. Chapter 12 discusses an overall security management architecture in terms of monitoring and maintaining your security approach and in defining response policies to security incidents. Chapter 13 discusses particular uses of DNS within broader security initiatives such as anti-spam and certificate validation.