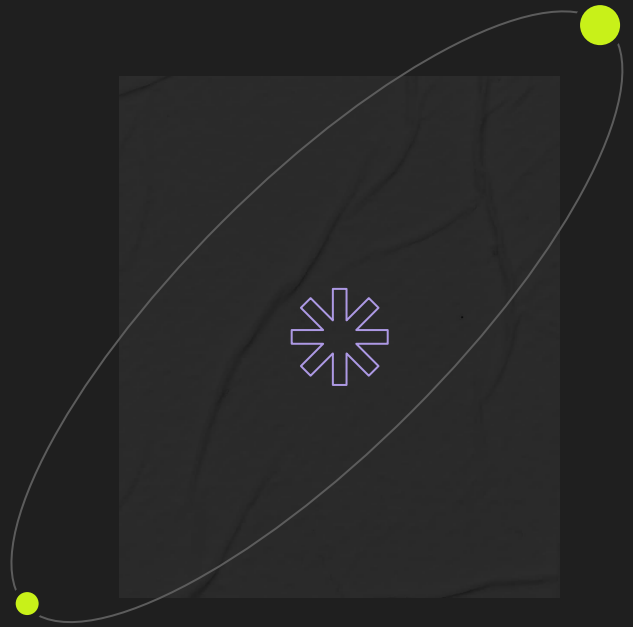
Cyber Security Fundamentals

Muhammad Amar Primus Firdaus
11230940000067

Content Of Cyber Security Fundamentals

Network and Security Concepts		
1	Information Assurance Fundamentals	Authentication, Authorization, Nonrepudiation, Confidentiality, Integrity, Availability.
2	Basic Cryptography	Information about basic cryptography
3	Symmetric Encryption	Example of Simple Symmetric Encryption with Exclusive OR (XOR), Improving upon stream Ciphers with Block Ciphers
4	Public Key Encryption	Information about Public Key Encryption
5	The Domain Name System (DNS)	Security and the DNS
6	Firewalls	History Lesson, What's in a Name ?, Packet-Filtering Firewalls, Stateful Firewalls, Application gateway Firewalls, Conclusions.
7	Virtualization	In the Beginning, The Virtualization Menu, Full Virtualization, Getting a Helping Hand from the Processor, If All Else Fails, Break It to Fix It, Use What You Have, Doing It the Hard Way, Biting the Hand That Feeds, Conclusion
8	Radio-Frequency Identification	Identify What?, Security and Privacy Concerns

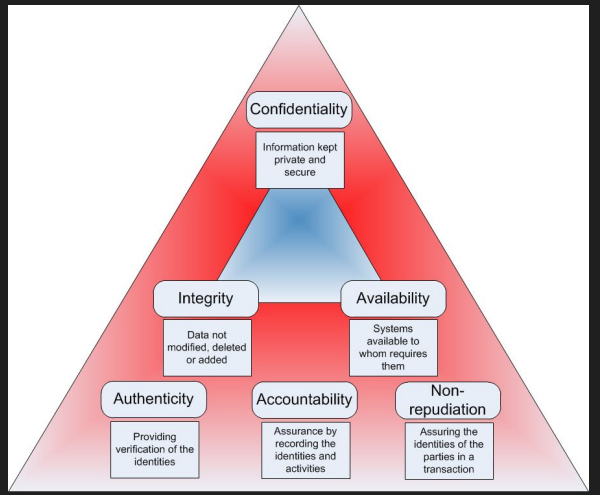
01 Information Assurance Fundamentals



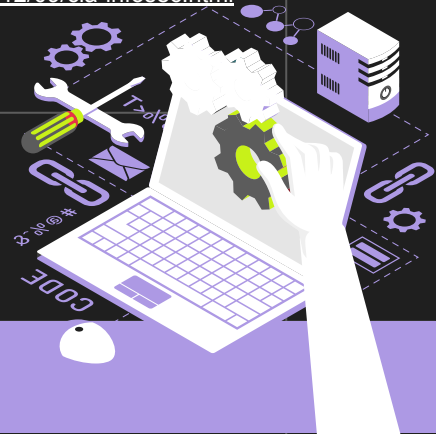
Authentication, Authorization, Nonrepudiation, Confidentiality, Integrity, Availability.

Information Assurance Fundamental

Information security professionals are dedicated to ensuring the protection of these principles for every system they protect. There are three key concepts that security professionals must understand in order to properly apply CIA principles, namely Authentication, authorization, and nonrepudiation. system designers can use to maintain system security with respect to confidentiality, integrity, and availability. There are 6 important elements in information security namely Authentication, Authorization, Nonrepudiation, Confidentiality, Integrity, Availability.



https://geraintw.blogspot.com/2012/09/cia-infosec.html



01

Authentication

Authenticationas a “security measure designed to establish the validity of a transmission, message, or originator, or a means of verifying an individual’s authorization to receive specific categories of information.” authentication is important to any secure system, as it is the key to verifying the source of a message or that an individual is whom he or she claims.

There are many methods available to authenticate a person. In each method, the authenticator issues a challenge that a person must answer. This challenge normally comprises requesting a piece of information that only authentic users can supply. These pieces of information normally fall into the three classifications known as ***factors of authentication***.

FACTOR	EXAMPLES
Something You Know	Information the system assumes others do not know; this information may be secret, like a password or PIN code, or simply a piece of information that most people do not know, such as a user’s mother’s maiden name.
Something You Have	Something the user possesses that only he or she holds; a Radio Frequency ID (RFID) badge, One-Time-Password (OTP) generating Token, or a physical key
Something You Are	A person’s fingerprint, voice print, or retinal scan—factors known as biometrics

Source : Page 2

02

Authorization

The NIAG defines ***Authorization*** as “access privileges granted to a user, program, or process.”. authentication relates to verifying identities, authorization focuses on determining what a user has permission to do. After a secure system authenticates users, it must also decide what privileges they have. For instance, an online banking application will authenticate a user based on his or her credentials, but it must then determine the accounts to which that user has access. Additionally, the system determines what actions the user can take regarding those accounts, such as viewing balances and making transfers.

https://www.freepik.com/icon/authorization_7465669

03

Nonrepudiation

Nonrepudiation is defined by NIAG as "the assurance that the sender of the sender of data is provided with proof of delivery and the recipient is given proof of the identity of the sender, so that neither can deny that they have processed the data ."To fulfil the requirements secure systems usually rely on asymmetric cryptography (or asymmetric (or public key) cryptography). While symmetric key systems use a single key to encrypt and decrypt data, asymmetric systems use key pairs. These systems use one (private) key to sign the data and use another (public) key to verify the data. (public) key to verify the data. If the same key can sign and verify the content of a message, the sender can claim that anyone with access to that key can easily verify the data. who has access to that key can easily forge it.

https://stock.adobe.com/id/search/images?k=non-repudiation

04

Confidentiality

The term ***confidentiality*** is familiar to most people, even those not in the security industry. The NIAG defines confidentiality as “assurance that information is not disclosed to unauthorized individuals, processes, or devices . ”Asurring that unauthorized parties do not have access to a piece of information is a complex task. It is easiest to understand when broken down into three major steps. First, the information must have protections capable of preventing some users from accessing it. Second, limitations must be in place to restrict access to the information to only those who have the authorization to view it. Third, an authentication system must be in place to verify the identity of those with access to the data

https://www.amberlo.io/blog/law-business/ways-to-prioritize-client-confidentiality/

05

Integrity

In the information security realm, ***integrity*** normally refers to data integrity, or ensuring that stored data are accurate and contain no unauthorized modifications. The National Information Assurance Glossary (NIAG) defines integrity as follows: Quality of an IS (Information System) reflecting the logical correctness and reliability of the operating system; the logical completeness of the hardware and software implementing the protection mechanisms; and the consistency of the data structures and occurrence of the stored data. Note that, in a formal security mode, integrity is interpreted more narrowly to mean protection against unauthorized modification or destruction of information

https://cybersecurityglossary.com/integrity/

06

Availibility

The NIAG defines ***availability*** as “timely, reliable access to data and information services for authorized users.” Information systems must be accessible to users for these systems to provide any value. If a system is down or responding too slowly, it cannot provide the service it should. Attacks on availability are somewhat different from those on integrity and confidentiality. The best-known attack on availability is a denial of service (DoS) attack. A DoS can come in many forms, but each form disrupts a system in a way that prevents legitimate users from accessing it. One form of DoS is resource exhaustion, whereby an attacker overloads a system to the point that it no longer responds to legitimate requests. The resources in question may be memory, central processing unit (CPU) time, network bandwidth, and/or any other component that an attacker can influence.

https://cybersecurityglossary.com/integrity/

02

Basic Cryptography

Information About Basic of Cryptography

What is Cryptography ?

The English word ***cryptography*** derives from Greek and translates roughly to “hidden writing.” For thousands of years, groups who wanted to communicate in secret developed methods to write their messages in a way that only the intended recipient could read. In the information age, almost all communication is subject to some sort of eavesdropping, and as a result cryptography has advanced rapidly. Understanding how cryptography works is important for anyone who wants to be sure that their data and communications are safe from intruders.

https://blog.1password.com/what-is-public-key-cryptography/

History of Cryptography

The ancient Egyptians began the first known practice of writing secret messages, using nonstandard hieroglyphs to convey secret messages as early as 1900 bc. Since that time, people have developed many methods of hiding the content of a message. These methods are known as ciphers.The most famous classical cipher is the substitution cipher. Substitution ciphers work by substituting each letter in the alphabet with another one when writing a message. Cryptography is driven by the constant struggle between people who want to keep messages secret and those who work to uncover their meanings. Substitution ciphers are very vulnerable to cryptanalysis, the practice of breaking codes.

https://www.kaspersky.com/resource-center/definitions/what-is-cryptography

Cryptography Modern

Since the invention of modern electronic computers, cryptography has changed significantly. We no longer write messages on paper pads or speak them character by character into a microphone but transmit them electronically as binary data. The increase in computing power also gives cryptanalysts powerful new tools for analyzing encrypted data for patterns. These developments have led to new algorithms and techniques for hiding data. The next section provides some detail about modern cryptography and how the principles of classical cryptography are applied to digital systems.

03 Symmetric Encryption

Example of Simple Symmetric Encryption with Exclusive OR (XOR), Improving upon stream Ciphers with Block Ciphers

Symmetric Encryption

Symmetric encryption, by definition, requires both communication endpoints to know the same key in order to send and receive encrypted messages (see the picture). Symmetric encryption depends upon the secrecy of a key. Key exchanges or pre-shared keys present a challenge to keeping the encrypted text's confidentiality and are usually performed out of band using different protocols. Algorithms in this category are usually fast because their operations use cryptographic primitives. As previously discussed in Basic Cryptography we explained how the cryptographic primitive substitution works. Permutation, or altering the order, is another cryptographic primitive that many symmetric algorithms also use in practice.

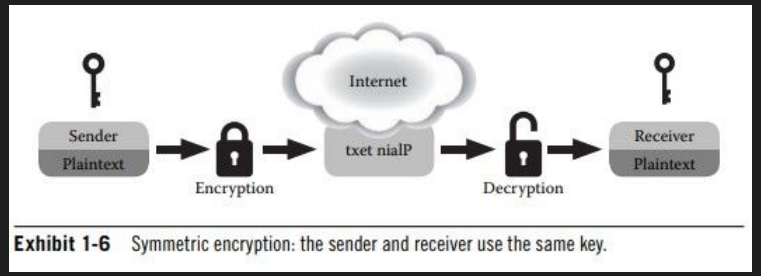


Exhibit 1-6 Symmetric encryption: the sender and receiver use the same key.

02

Example of simple symmetric Encryption with exclusive Or

(XOR) At its most basic level, symmetric encryption is similar to an exclusive OR (XOR) operation, which has the following truth table for input variables p and q:

P	Q	=	P XOR Q
True	True		False
True	False		True
False	True		True
False	False		False

The XOR operation is nearly the same as one would expect for OR, except when both p and q are true. The properties of XOR make it ideal for use in symmetric cryptography because one of the inputs (p) can act as the message and the other input (q) can act as the key. The recipient of an encrypted message (p XOR q) decrypts that message by performing the same XOR operation that the sender used to encrypt the original message (p).

P XOR Q	Q	=	(P XOR Q) XOR Q
False	True		True
True	False		True
True	True		False
False	False		False

Improving Upon Stream Ciphers with Block Ciphers

Block ciphers are more common in symmetric encryption algorithms because they operate on a block of data rather than each character (bit or byte). PRNG algorithms used in stream ciphers are typically time intensive. Block ciphers are the best choice for bulk data encryption. Stream ciphers remove patterns from ciphertext using PRNGs, but block ciphers use a more efficient method called cipher block chaining (CBC).

When using a block cipher in CBC mode, both a key and a random initialization vector (IV) convert blocks of plaintext into ciphertext. The initialization vector and plaintext go through an XOR operation, and the result is an input to the block cipher with the chosen key (see Exhibit 1-7). This ensures that the resulting ciphertext is different, even if the same key was used to encrypt the same plaintext, as long as the IV is different and sufficiently random with each execution of the algorithm.

The next block will be encrypted with the same key, but instead of using the original IV, CBC mode uses the ciphertext generated by the last function as the new IV. In this way, each block of ciphers text is chained to the last one. This mode has the drawback of data corruption at the beginning of the file, resulting in complete corruption of the entire file, but is effective against cryptanalysis.

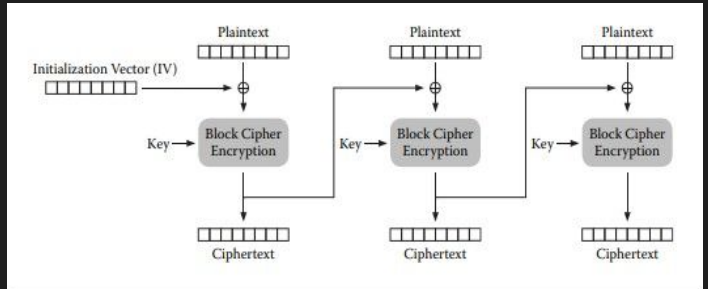


Exhibit 1-7 Cipher block chaining (CBC) mode encryption. *Source*: Cipher block chaining http://en.wikipedia.org/wiki/File:Cbc_encryption.png.

Source : Page 15

04
Public Key Encryption

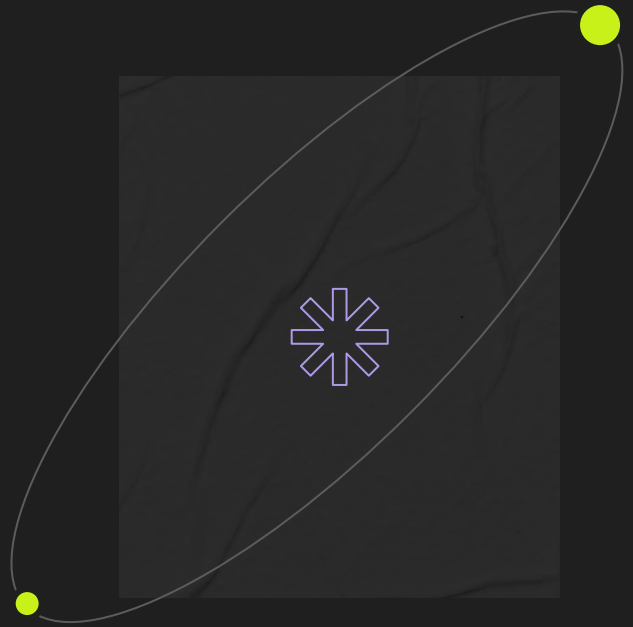
Information About Public Key Encryption

What is Public Key Encryption ?

Public key encryption represents a branch of cryptography for which the distinguishing attribute of the system is the use of two linked keys for encryption and decryption, rather than a single key. While a variety of public key encryption solutions have been proposed, with some implemented and standardized, each system shares one common attribute: each public key system uses one key, known as the public key, to encrypt data, and a second key, known as the private key, to decrypt the encrypted data. Public key encryption solves one of the major issues with symmetric key encryption, namely, the use of a shared key for both sides of the conversation.

Public key encryption is a new technology revolutionizing the field of cryptography. The encryption scheme allows parties to communicate over hostile communication channels with little risk of untrusted parties revealing the contents of their communication. The use of two keys—one public and one private—reduces the burden of establishing a shared secret prior to the initial communication. While the mathematics involved in public key encryption is complex, the result is an encryption system that is well suited for untrusted communication channels.

05
The Domain Name System



Security and The DNS

What is Domain Name System ?

DNS is a fundamental piece of the Internet architecture. Knowledge of how the DNS works is necessary to understand how attacks on the system can affect the Internet as a whole and how criminal infrastructure can take advantage of it. This section explains the fundamentals of the domain name system (DNS), which is an often overlooked component of the Web's infrastructure, yet is crucial for nearly every networked application. Many attacks, such as fast-flux and DNS application, take advantage of weaknesses in the DNS design that emphasize efficiency .

The primary goal that the designers of the DNS had in mind was scalability. This goal grew from the failure of the previous solution that required each user to download a multi thousand-line file named hosts.txt from a single server. To create a truly scalable system, the designers chose to create a hierarchy of "domains." At the top of the hierarchy is the "root" domain under which all other domains reside. Just below the root domain are top-level domains (TLD) that break up the major categories of domains such as .com, .gov, and the country code TLDs. Below the TLDs are second-level domains that organizations and individuals can register with the registry that manages that TLD. Below second-level domains are the third-level domains and so forth, with a maximum of 127 levels.

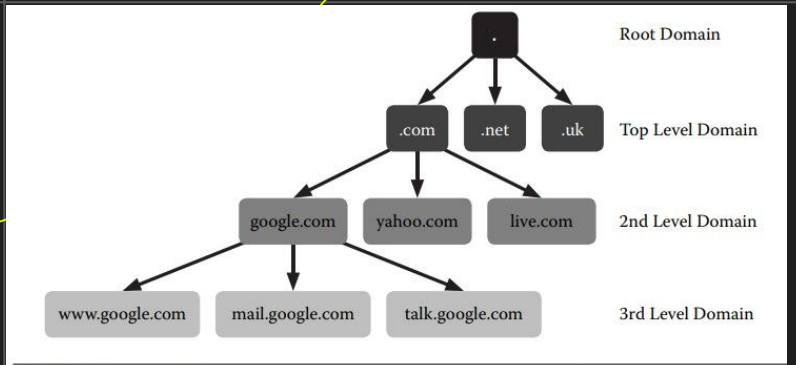


Exhibit 1-10 The hierarchical structure of the domain name system (DNS).

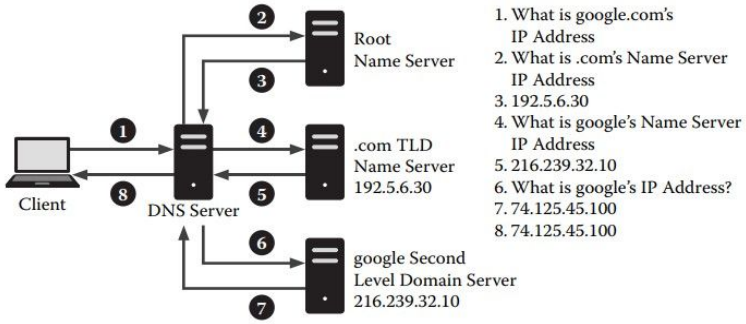


Exhibit 1-11 Resolution of google.com using a recursive DNS server.

Exhibit 1-10 shows how the hierarchical nature of the DNS leads to a tree-like structure consisting of domains and subdomains.

Exhibit 1-11 shows the steps required for a resolver to complete this process.

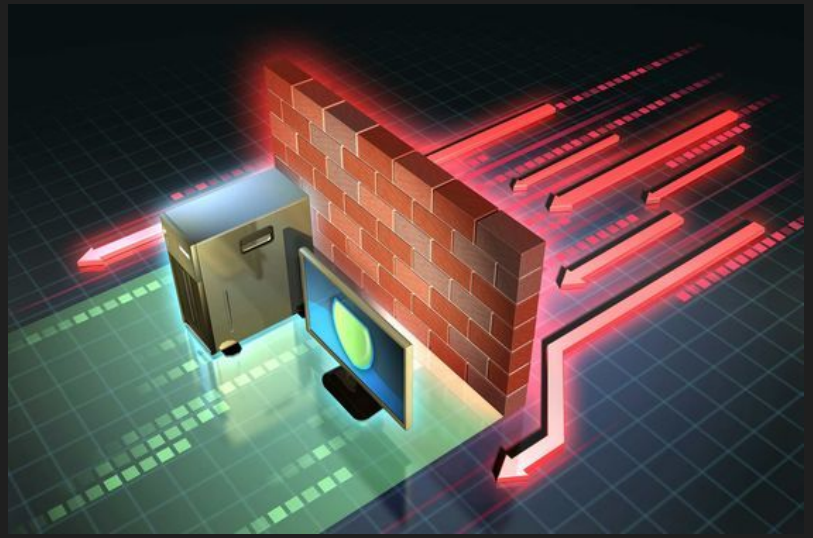
Security and The DNS

As a fundamental part of the modern Internet, the security of the DNS is important to all Internet users. In the previous discussion of how the DNS system works, it is important to note that no authentication of results ever occurred. This makes the system vulnerable to an attack known as DNS cache poisoning, wherein an attacker tricks a DNS server into accepting data from a non authoritative server and returns them to other resolvers.

Exhibit 1-12 A long phishing domain appearing to belong to CitiBank.

Exhibit 1-12 shows a five-level domain of online.citibank.com.n5mc.cn that may appear to belong to CitiBank but is actually a subdomain of n5mc.cn. Organizations that want to issue takedown requests for these domains need to understand how the DNS works so they can take the correct actions.

06
Firewalls



History Lesson, What's in a Name ?, Packet-Filtering Firewalls, Stateful Firewalls, Application gateway Firewalls, Conclusions.

01

History Lesson

The Internet turned forty years old in 2009, but the use of devices to separate one network from another, undesirable network, did not occur until the late 1980s.13 At the time, network administrators used network routers to prevent traffic from one network from interfering with the traffic of a neighboring network. Enhanced routers introduced in the 1990s included filtering rules. The designers of these routers designated the devices as security firewalls.

The next generation of security firewalls improved on these filter enabled routers. During the early 1990s, companies such as DEC, Check Point, and Bell Labs developed new features for firewalls. Check Point, for instance, eased the technical expertise requirements for configuring firewalls by providing user-friendly interfaces while at the same time providing administrators with new configuration options for refined rule sets.

02

What's in a Name ?

Firewalls are network devices or software that separates one trusted network from an untrusted network (e.g., the Internet) by means of rule-based filtering of network traffic.

While Exhibit 1-13 identifies the firewall as a separate physical device at the boundary between an untrusted and trusted network, in reality a firewall is merely software. This does not mean that physical, separate devices are not firewalls, but merely that these devices are simply computers running firewall software. Host-based firewalls have found their way into most operating systems. Windows XP and later versions have a built-in firewall called the Windows Firewall.

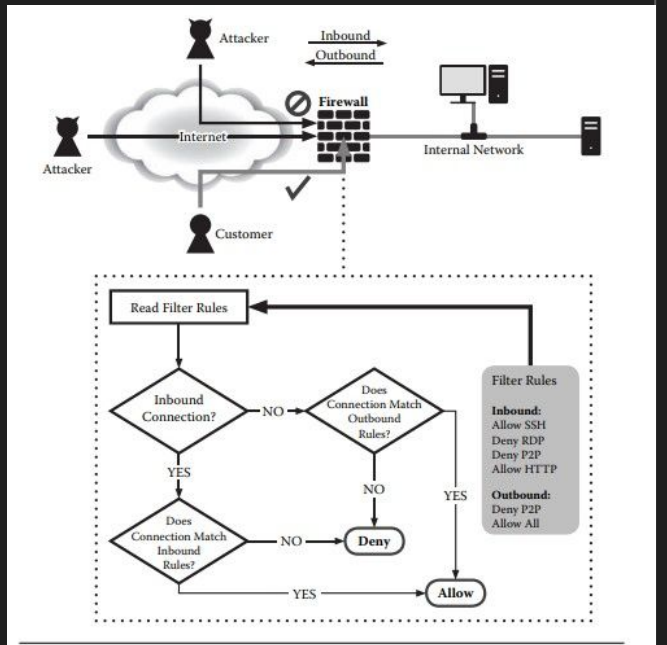


Exhibit 1-13 A basic firewalled network.

03

Packet-Filtering Firewalls

The most rudimentary of firewalls is the packet-filtering firewall. Packet-filtering firewalls work at the IP level of the network. Most routers integrate this type of firewall to perform basic filtering of packets based on an IP address. The principle behind packet-filtering firewalls is that the firewall bases the decision to allow a packet from one network into another network solely on the IP address of the source and destination of the packet.

Packet-filtering firewalls can also expand on the basic principle of IP-address-only filtering by looking at the Transmission Control Protocol (TCP) or User Diagram Protocol (UDP) source and destination ports. In this mode, the firewall operates in nearly the same fashion as the packet-filtering firewalls operating on the IP address. For a packet to pass through the firewall, the source IP and port and the destination IP and port must match at least one rule in the filter list. More advanced routers and even some higher-end switches offer this functionality.

Stateful Firewalls

Simple packet-filtering firewalls suffer from one significant downside: they do not take into consideration the state of a connection, only the endpoints of the connection. Stateful firewalls allow only properly established connections to traverse the firewall's borders. While packet filtering is still a key element of these firewalls, the firewall also pays attention to the state of the connection.

It is the ability to determine the order and state of a communication session that allows stateful firewalls to make faster determinations about incoming packets. Of course, it is important that these firewalls do not run out of memory from storing the state of stale connections. To avoid this problem, stateful firewalls will purge state information for sessions that have "gone quiet" for a significantly long period. Once a session has expired, the next packet originating from either host will result in the firewall verifying the packet against packet-filtering rules and the establishment of a new session

05

Application Gateway Firewalls

A classic example of an application gateway firewall is a Web proxy or e-mail-filtering proxy. A Web proxy, for instance, understands the proper HTTP protocol and will prevent an improperly constructed request from passing. Likewise, an e-mail filtering proxy will prevent certain emails from passing based on predefined conditions or heuristics (for example, if the email is spam). These proxies also prevent unknown protocols from passing through. For example, a properly configured HTTP proxy will not understand an SSH connection and will prevent the establishment of the connection (see Exhibit 1-14). This level of packet inspection cannot occur with either a packet-filtering or stateful firewall, as neither firewall type looks at the application layer of the network stack. By identifying improperly constructed packets for a given protocol, the application gateway firewalls may prevent some types of protocol-specific attacks; however, if a particular protocol's definition allows for such a vulnerability, the gateway will provide no protection.

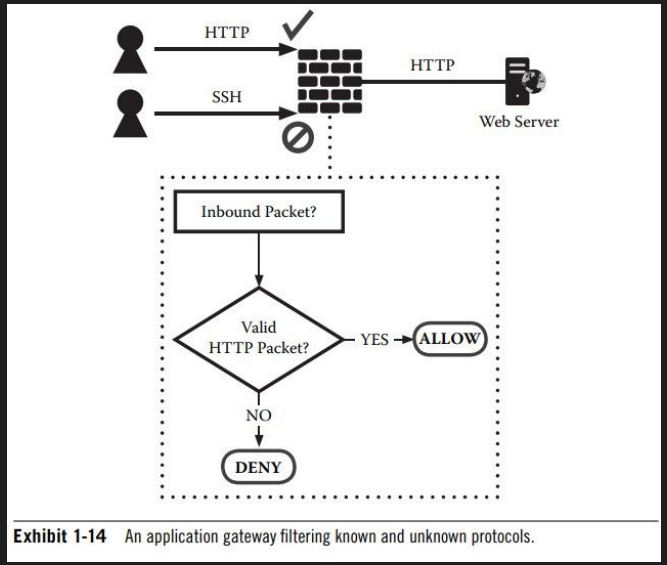


Exhibit 1-14 An application gateway filtering known and unknown protocols.

06

Conclusions

Firewalls come in a variety of forms, from simple packet filtering to the more complex proxy. The topic of firewalls is complex and extremely well documented. Authors from the IT security community have dedicated entire books to the subject of designing, administering, and implementing firewalls. To understand the importance of firewalls, the minute details of their operation can be avoided, but it is critical to understand the high-level concepts of their operation. Understanding the basics of how firewalls process traffic and how that processing prevents unwanted intrusions is the key to understanding the security of firewalls.

Like antivirus solutions, the impression that firewalls will stop all evils of the Internet is overstated at best. Firewalls provide a single layer of defense in the larger scheme of defense in depth. While firewalls can reduce the attack surface of a server by blocking unnecessary ports from the Internet at large, firewalls cannot protect resources that are vulnerable to specific vulnerabilities such as buffer overflows and privilege escalation attacks.

07 Virtualization

In the Beginning, The Virtualization Menu, Full Virtualization, Getting a Helping Hand from the Processor, If All Else Fails, Break It to Fix It, Use What You Have, Doing It the Hard Way, Biting the Hand That Feeds, Conclusion

01

In The Beginning

Virtualization at its most fundamental level is the simulation or emulation of a real product inside a virtual environment. Recent efforts by many companies to capitalize on the wave of cloud computing have given the term new importance in the IT and business communities, but the term virtualization is older than most realize. In the 1960s, researchers at the IBM Thomas J. Watson Research Center in Yorktown, New York, created the M44/44X Project. The M44/44X Project consisted of a single IBM 7044 (M44) mainframe that simulated multiple 7044s (44X). The M44/44X Project was the first to use the term virtual machine (VM) to describe simulating or emulating a computer inside another computer using hardware and software.

For decades, the use of virtual machines inside mainframes has been common practice. The use of these virtual machines gives mainframes the ability to act not as a single machine but as multiple machines acting simultaneously. Each virtual machine is capable of running its operating system independent of the other virtual machines running on the same physical machine. In this sense, the mainframe effectively turns one machine into multiple machines. Mainframes only represent the beginning of the virtualization technology and are by no means the only systems that provide the service.

02

The Virtualization Menu

Platform virtualization is a broad category that contains several variations on a theme. The most predominant platform virtualization techniques 19 include full virtualization, hardware-assisted virtualization, paravirtualization, and operating system virtualization. Each of these techniques accomplishes the task of virtualization in different ways, but each results in a single machine performing the function of multiple machines working at the same time.

Virtualization systems consist of several key components: a VMM, physical hardware, virtual hardware, virtual operating systems, and a host (or real) operating system. Exhibit 1-15 illustrates the relationship of these components. The key component, the component that makes virtualization possible, is the VMM.

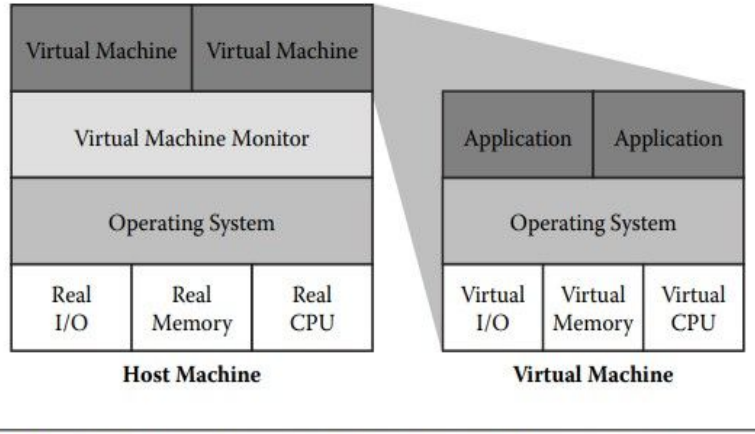


Exhibit 1-15 The relationship between virtual machines and a host machine.

03

Full Virtualization

Full virtualization aims to provide a realistic, accurate virtual representation of the real hardware. However, x86-based architecture faces issues due to the different levels of privilege offered by the x86 family of processors, known as rings. Ring-0, the most privileged code level, houses the kernel of the operating system and allows code to manipulate sensitive components freely. Virtual Machine Managers (VMMs) use full virtualization to execute code in the virtual machine in the same manner as a physical machine, ensuring it doesn't interfere with the host machine or other VMs. Virtual machine applications like VMware use the host machine's processor to execute instructions requested by the virtual machine, resulting in faster virtual machines and less processor time and resources.

04

Getting a Helping Hand for The Processor

Hardware manufacturers are increasingly interested in hardware-assisted virtualization, as virtualization technology matures from a software perspective. Intel and AMD have released newer x86-based processors with processor extensions, which provide solutions to privileged x86 instructions that the Virtualization Memory (VMM) cannot virtualize. These extensions provide an even more privileged layer than ring-0, allowing the VMM to operate in a new root mode privilege level below ring-0. When the virtual machine's operating system executes an instruction that would cause instability in the host machine's operating system, the hardware passes the request to the VMM, which resides in a separate processor space. This allows the virtual machine's operating system to run unobstructed, reducing overhead, while the host processor ensures the virtual machine's operating system does not impede the host operating system.

05

If All Else Fails, Break It to Fix It

Paravirtualization is a solution to the nonvirtualizable instruction problem in x86 processors, developed before hardware-assisted virtualization technologies were introduced. It allows the virtual machine's operating system to run in ring-0 after modifying the system to restrict dangerous x86 instructions. Paravirtualization breaks instructions that cause instability in the host machine and replaces them with calls to the Virtual Machine Manager (VMM), resulting in a virtual machine's operating system and applications running in the intended rings. However, paravirtualization requires modification to the virtual machine's operating system, which is difficult for closed-source operating systems. Most paravirtualization-based virtual machines run modified Linux operating systems.

06

Use What You Have

Operating system-assisted virtualization is a technique that provides an application with the illusion of a dedicated operating system, unlike other virtualization techniques that offer a virtual machine capable of supporting ring-0 instructions. This technique is common on Linux and Unix-based systems via chroot, FreeVPS, and FreeBSD Jail. It allows a single operating system instance to run multiple applications in isolation while still providing them with necessary operating system resources, such as disk and network access.

07

Doing It The Hard Way

Emulators, like virtualization systems, operate on the same principles as virtualization systems, but without the requirement for the host machine to match the same basic architecture. Emulators emulate all aspects of the virtual machine's hardware, translating the virtual machine's instructions into instructions that can run on the host machine. Emulators can run radically different architectures than the host machine's, such as x86 architectures running older Apple Macintosh operating systems. However, this power comes with a significant overhead, resulting in a performance penalty. Emulators can also run virtual machines of the same architecture as the host machine, such as VMware, which can emulate the x86 architecture including the CPU within a virtual machine. This provides a more realistic virtual environment without relying on translation of certain ring-0 instructions.

08

Biting The Hand That Feeds

Virtualization of infrastructure resources can reduce the need for physical servers but also introduces risks. Virtualization systems aim to provide strict boundaries between the host system and virtual machines running on it. However, malicious actors can breach these boundaries. As virtualization systems gain popularity, attackers focus on weaknesses within these systems. The operating system and applications of a virtual machine run on the host system, and when the virtual machine accesses physical resources, the separation between the two can crumble.

09

Conclusion

Virtualization offers numerous benefits, including server consolidation and program isolation. With advancements in computing hardware, it is becoming more widely adopted in the IT community. Virtualization is a key component of new cloud-computing technologies, but its growth is not yet reaching its peak. Understanding the risks associated with virtualization is crucial, as transparency between virtual and host machines increases the risk of data exposure and system compromise. Classifying data and virtual machine types can reduce this exposure.

08
Radio-Frequency Identification

Identify What?, Security and Privacy Concerns

01

Identify What ?

RFID tags, such as the electronic product code (EPC), are passive RFID tags that store more information than Universal Product Codes (UPC). EPCs are the equivalent of barcodes and are often integrated into stickers. They contain a 96-bit number representing the product's manufacturer, type, and serial number. EPC codes can store over 600 billion unique serial numbers, making them useful for identifying specific product types. Organizations can place EPCs on any product or group of products they want to track. In 2005, Wal-Mart Stores mandated suppliers to tag shipments with RFID tags. Libraries have also begun using EPC tags to expedite book check-ins and check-outs. RFID tags are being used by organizations and governments to identify more than just household products, with many deploying RFID-equipped ID cards for access to buildings and systems.

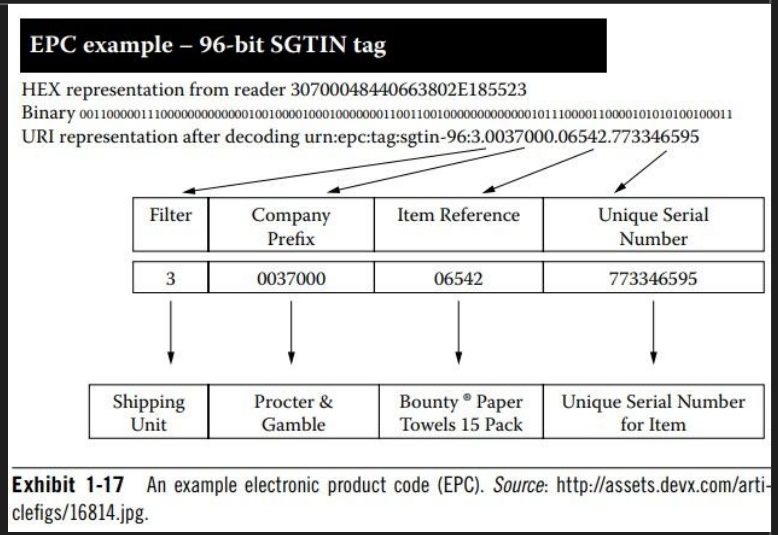


Exhibit 1-17 An example electronic product code (EPC). *Source*: http://assets.devx.com/articlefigs/16814.jpg.

02

Security and Privacy Concern

In 2005, researchers at Johns Hopkins University broke the encryption used by millions of RFID-enhanced car keys and Exxon Speedpass RFID payment system. These devices use RFID technology as a lock-picking prevention mechanism, and if the proper RFID tag is not in proximity of the reader, the car will not start. Speedpass allows Exxon customers to link a credit card to their keychain tokens to make purchases from Exxon gas stations. However, these devices only use 64-bit encryption to protect the tags. At DEFCON 17, Chris Paget, a security researcher, debunked the myth that readers can only interrogate RFID tags at short ranges. One ID card Paget researched is the U.S. enhanced driver's license (EDL), which can be easily read at distances of over twenty feet and contain no encryption. Attackers can easily clone these cards to steal victims' IDs without their knowledge. RFID tags also have ramifications for personal privacy, as attackers can read them without the user's knowledge.

Thank You