UNIT 1

CONTENTS

UNIT 1

1.1 HACKING IMPACTS

- At the risk of stating the obvious, hacking—computer crime—can result in massive financial losses for companies, governments, and individuals alike. The costs associated with computer crime can manifest themselves in various ways, which may range from the obscure to a clear hit to the bottom line.
- Digital assets where costs from hackers can manifest themselves fall into four major categories: resources, information, time, and reputation.

1. **Resources**: Resources refer to computer-related services that perform actions or tasks on behalf of a user. These services can include core services, object code, or disk space. If these resources are controlled, utilized, or disabled by an unauthorized entity, it could result in the inability to capture revenue for a company or have an impact on an important process, resulting in the failure to meet expected objectives.

2. **Information**: Information can represent a significant cost if destroyed or altered without authorization. **Loss** of data is relatively easy to measure, but disclosure and integrity are more challenging. Data can be affected in several ways, including loss, disclosure, and integrity. Loss of data takes time to collect or produce and has value, while **disclosure** represents the traditional fear of hacking, which is proprietary information theft. Integrity is ensuring that information is accurate and complete, and if data is manipulated, it could result in loss to the owner.

3. **Time**: The loss of time can lead to costs in the form of payroll, not meeting critical deadlines, or an unavailable E-commerce site that would normally produce thousands of dollars in revenue if it were available. Anything that consumes time also consumes money, and expenditures for recovering from an incident can represent the greatest form of financial loss.

4. **Brand and Reputation**: Brand and reputation are critical to many companies. If a company's reputation is damaged due to a security breach, it can have a significant impact on its bottom line. This can lead to loss of customers, investors, and business partners. Companies with recognizable brands are particularly vulnerable to reputational damage.

1.1a THE HACKER

- The term "hacker" has evolved over time to describe computer criminals, but historically it referred to those who explored the inner workings of computers for fun and a challenge. To effectively evaluate the risks of an organization, it is important for business and security consultants to understand the nuances of the hacker community, including their social status, drivers, and targets.
- Testing the defensive capabilities of an organization against different types of threats, including internal attacks from employees, is essential. This approach should also be applied to external threats from unknown sources on the internet. In the following sections, we will explore the types of hackers, their techniques, and what organizations can expect from them to better plan their security tests.

Type Of Hacker

- Hackers are a diverse group, including individuals of various races, religions, and ages. A common misconception is that they are uneducated, unprofessional, and have nothing better to do than cause trouble. While some hackers may have questionable ethics, many are law-abiding citizens who are attracted to the anonymity provided by the internet.
- Unlike other criminals, many hackers would be horrified if they had to confront their victims face to face or witness the consequences of their actions. Hackers view computers as tools and rely on

impersonal acts. The challenge and desire for satisfaction are common motives among hackers. While they may not physically harm anyone, their actions can cause significant harm to computer systems and networks.

- There are several types of hackers, but we can reduce this to three basic characteristics that we can use to categorize the enemy:

1. Script kiddies
2. Hackers
3. Über hacker

Script Kiddies

- A "script kiddie" is a wannabe hacker who uses tools made by more skilled hackers to commit malicious acts. They can cause varying degrees of damage and are categorized into unstructured, structured, and determined groups.

a. **Unstructured:** Pranksters and recreational hackers fall into this group, causing limited but destructive damage. Internal employees performing recreational hacking pose a great threat to organizations.

b. **Structured:** The right tool in the wrong hands, combined with opportunistic behavior, can have measurable results. Distributed Denial of Service attacks (DDoS) are an example of this. With comprehensive toolsets freely available on the Internet, script kiddies armed with such tools are a concern.

c. **Determination:** The persistence of a determined script kiddie increases the probability of success. The shotgun approach may not be the best tactic, but their determination can pose more of a problem than accomplished hackers.

Hackers

- Hackers are highly skilled individuals who explore computers for education, challenge, and status among peers. They compete to gain recognition and power in the hacking community. Hackers possess exceptional skills and a refusal to accommodate traditional thinking, enabling them to form deductions and process information with ease.
- They can manipulate technology to make systems do what is needed by exploring unorthodox options. Hackers are not to be underestimated, and there are four types: malicious, solvers, hacktivist, and vigilante, each with unique characteristics. Despite these distinctions, all hackers share a strong desire for power and control over remote resources.

a. **Malicious hackers** are individuals who have the intent of causing damage or destruction to information systems. They may be writers of malware or gain access to sites and corrupt information. Their actions are usually based on some opinion of the target or a desire to gain a reputation. Destruction of systems and data may also be used to cover tracks or other attacks. Malicious hackers are especially concerning because they have the skill and no conscience for the ramifications of their actions.

b. **Solvers** are hackers who gain access to systems to solve a problem they or a friend may have. They may change or remove information to rectify a situation. Solvers may also hack to prove a point and rely on the concept that they hacked a site to prove an insecurity. However, their actions may still be illegal and cause harm to others. It is important to note that ethical hacking, with permission, can be performed to identify and address vulnerabilities in systems.

c. **Hacktivists** are hacking communities that band together for a common cause, such as anarchism, racism, animal rights, or environmental protection. However, advocacy hackers can be dangerous to certain businesses that support or represent antagonism, such as companies that perform testing on animals or mine for resources. Companies wishing to have an ethical hack performed on their networks should state what represents the greatest threat to their business so that testers can assume the mindset of the proposed attacker.

d. **Vigilantism** involves groups of individuals who surreptitiously attack the Internet's lower lifeforms, such as those involved in child pornography. However, this raises questions of law and ethics, as the FBI may investigate perpetrators of computer crime only to find out their target was a ring of child pornography dealers and be forced to arrest the vigilante-hacker trying to put lowlifes out of business. Law enforcement is concerned with vigilantes attacking systems because data used for prosecution can be lost during the attack, canceling the original intent of the vigilantes. It is important to note that hacking and infiltrating network systems is against the law and may cause harm to innocent parties.

Über Hacker

- The term "über" in German means "super," and an "über hacker" is someone with exceptional skills and experience in programming, systems, hardware, communications, and protocols. They are the elite and most feared hackers, capable of writing tools for other hackers and sought after by unscrupulous businesses and governments.
- Some remain hidden in legitimate professions, but others use their power for personal gain or espionage. There are two types of super hackers: those who extort and those who spy.

a. **Extortionists** Hackers are increasingly using information to blackmail individuals and organizations for monetary gain, creating a new breed of criminal known as extortionists. Financial institutions, online retailers, and gambling sites are common targets due to their access to cash and potential impact of reputation loss. Hackers gain unauthorized access to a bank's system, obtain personal information, and demand payment to keep quiet. Surprisingly, many companies comply with the demands, assuming the information won't be released. However, this only labels them as a "sucker" for other crime communities to target for free money. Unlike other hackers, über hackers focus on money rather than reputation. There are two types: hitmen and terrorists.

- ✓ **Hitman:** Über hackers in this category often work with criminal organizations in a mutually beneficial relationship. They are hired to gather information to control people and money in exchange for payment. For example, a hacker may plant incriminating evidence on a government official's computer to force them to act in favor of the organization. These individuals are essentially given an offer they can't refuse.
- ✓ **Terrorist:** While not as impactful as traditional acts of terrorism, there are many examples of terrorists using computers to carry out attacks. It is hoped that government entities like the NSA, FBI, and CIA are successful in countering these threats.

  Cyber terrorism takes many forms, from mild to severe, and targets governments, public systems, and organizations involved with technology or the community. The Chinese hack of several US government networks in response to the US naval spy plane damaged by a Chinese jet is an example of a targeted attack. While this book focuses on ethical hacking, it's important to acknowledge the potential harm caused by some factions' desire to cause damage.

b. **Espionage**

  Government and industrial espionage can have a significant impact on the success of organizations worldwide. While government espionage uses both people and technology, industrial espionage mainly focuses on stealing information to gain a competitive advantage. For instance, the theft of

research and testing data for a new drug could significantly benefit a competitor's ability to pursue or support a less successful study of the same subject.

1.1b SOCIOLOGY

- The social framework of hackers and their community. It highlights that despite the common perception of hackers as isolated individuals, they exist within social groups that provide support, training, and expertise. The society of hackers is driven by technology, secrecy, and anonymity, which play a fundamental role in their community.
- Benedict Anderson's concept of the imagined community is used to describe how hackers, who may never meet each other, are bound together in allegiance to a common cause. The sharing of information is essential to the community as a whole, but secrecy is also necessary because hacking is illegal.
- Anonymity provides fluidity of membership and acceptance, with new members often joining and groups creating their own rules without a leader. The Internet has become an instrument fueling the diversity and communal actions of hackers, enabling them to share information and ultimately form an imagined community.

a. Motives

- The difficulty in categorizing the motives behind cybercriminal behavior, as it is a complex and ever-changing issue influenced by social factors, mental capacity, and attitude. The author uses an anecdote about a child cutting their sibling's hair out of curiosity to demonstrate the potential risks associated with curiosity.
- Two hackers' motivations are then explored, with Maelstrom driven by the thrill of evading authority and peer recognition, while Kevin Mitnick was motivated by a desire to gain a better understanding of computer systems and their interactions. The paragraph highlights that understanding cybercriminal motivations is challenging due to the many variables involved.
- This paragraph outlines six fundamental drivers for hackers:

1. **Addiction to computers**: Many hackers feel compelled to hack due to their addiction and obsession with computers, which offer a controllable environment that poses intellectual challenges.
2. **Curiosity of the possible**: Hackers are motivated by the unknown of the target and their own abilities, and are driven by the strong stimulant of curiosity to discover opportunities.
3. **Excitement**: Some hackers hack for the excitement, which can be based on the anonymity the Internet provides and the freedom to be whoever they want to be.
4. **Social status**: Gaining acceptance into the community or establishing alpha roles within a smaller group can be a critical encouragement for successfully attacking or vandalizing a system.
5. **Power**: Taking control of a system is a thrill not easily duplicated or attained in their normal lifestyles, and is a formidable motivator for hackers.
6. **Betterment of society**: Some hackers believe they are helping the general public by exposing security holes and vulnerabilities, leading to the resulting fix reducing future attacks. However, this can also be a risky venture if hackers turn into consultants.

1.2 THE HACKER FRAMEWORK

- A framework is a structured and measurable collection of tasks arranged in a hierarchy to achieve the relationships between tasks and methods. The book presents a customizable framework for penetration testing that aligns with the overall security program and achieves its intended value. It provides

operational structure to ensure tasks are performed at the appropriate time and sequence, considering the limitations and constraints that may affect the test's value.

- The simplified figure 3.1 illustrates the primary phases of the framework, with each point representing a task or value element. Selecting the right elements within a framework is crucial to meet specific test goals. The author compares this to racquetball, where technique and tools are essential, but the value ultimately rests on the tester and framework being followed.
- The paragraph emphasizes the importance of evaluating which elements are needed to meet the goals of the test, as the impact of omitting certain elements can vary, affecting the overall value in the security program. Different characteristics of the testing process have varying degrees of intensity and requirements, which may influence other areas of the test and its overall value.

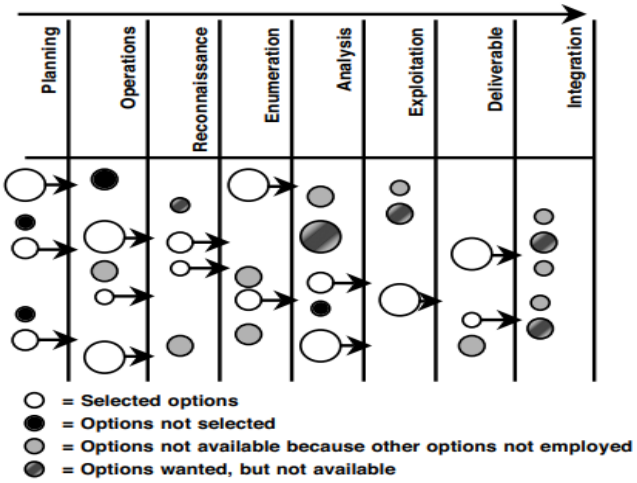

FIGURE 3.1 Determining the Impact on Value Based on Selected Options

a. **Planning the test:** A successful penetration testing project is crucial for evaluating existing business demands, processes, security policies, culture, laws, regulations, best practices, and industry requirements. These inputs help make decisions on the scope and scale of the test, which directly impacts the deliverables and integration of results into the security program. Formulating a controlled attack is essential, where security policies, programs, posture, and risk all guide the outcome of the test. The company's focus on security, core business needs, challenges, and expectations sets the stage for the entire engagement.

b. **Sound operations**: refer to the underlying actions necessary for performing a penetration testing engagement regardless of the scope. Addressing key questions related to logistics, limitations, and evaluation of testers is critical for ensuring a controlled attack, achieving desired results, and mitigating potential risks.

c. The **reconnaissance** phase involves searching for publicly available information that can assist in an attack, and poor planning or a lack of understanding of limitations can lead to poor results. The relationship between the framework, tasks, and methods of the test becomes clear during the reconnaissance phase.

d. During **enumeration**, readily available information is obtained directly from the target's systems, applications, and networks. The line between a passive and active attack begins to blur, and appropriate expectations must be set to avoid legal consequences. Port scanning is a crucial tool for compiling a list of information obtained from systems by manipulating the basic communication setup between two networked systems using the TCP/IP protocol.

Overall, a successful penetration testing project requires planning, sound operations, and attention to detail during the reconnaissance and enumeration phases to achieve the desired results while mitigating risks.

Port scanning is a technique used to detect which computers respond to connection requests, by using the three-way handshake of the TCP protocol.

1. Computer A sends a message called a "SYN" (Synchronize) to Computer B.
2. Computer B acknowledges that message with a "SYN+ACK" (SYN with an Acknowledgement) to Computer A.
3. Computer A sends back an acknowledgement—"ACK."

In the reconnaissance phase of an attack plan, information gathering and analysis are critical. The collected data is evaluated to determine whether more reconnaissance is necessary or to build a matrix for the next phase, vulnerability analysis.

Logical deduction skills are crucial in this phase, as they are used to manipulate the rigid computer environment and formulate hypotheses for attacking security controls. Enumeration involves inventorying all collected information to build logical threads to circumvent network, system, or application security controls.

e. **Vulnerability Analysis** Vulnerability Analysis phase, collected information is compared to known vulnerabilities to identify potential attack points. Information from various sources, such as the internet and target companies, is analyzed for possible exposures. The internet provides a wealth of information about known vulnerabilities, incidents, service packs, updates, and available hacker tools that can help identify potential attack points. The architecture of the target can be easily associated with internet information, making it a useful resource for vulnerability analysis.
f. **Exploitation** During the exploitation phase of a penetration test, testers use gathered information to attack the target system or application. Planning is critical, and testers follow a pattern to break the process into threads and groups with specific tasks. The goal is to achieve the objectives within the specified scope and time frame. Success is evaluated at every point in the operation. The tester must translate planning into an attack that meets the goals. Exploiting systems and applications can be easy or complicated, but the tester follows a pattern to achieve the objective.

   The tester makes two fundamental determinations: whether the test results meet **expectations** and whether there are any **technical issues** that could impact the engagement. Any unexpected results should be closely monitored to avoid negative impacts on the target or exceeding the test's scope. The exploitation phase is also an opportunity for the tester to discuss test tactics rather than solely focusing on exploitation tactics.

g. **Final Analysis** The final analysis phase is critical to ensure the success of a penetration testing engagement. During this phase, vulnerabilities discovered during the test are categorized to determine their level of exposure and support the creation of a deliverable and mitigation plan. It also provides an opportunity to identify other opportunities and build a bigger picture of the target's security posture. The analysis process involves both interpretation and empirical results, and it's crucial to provide enough evidence to classify vulnerabilities correctly. This helps communicate the results in a clear and useful manner, making it easier to highlight vulnerabilities that represent a substantial risk and their remediation.
h. **Deliverable** The various types of deliverables in penetration testing, from basic vulnerability lists to comprehensive reports, and notes that the choice of deliverable is influenced by the organization being tested. The author also acknowledges that the trend of commoditized ethical hacking has led to some reports being created to scare executives. However, the author emphasizes the importance of good deliverables that provide insightful commentary and measurable levels of risk to drive valuable security improvements. The chapter on deliverables in the book provides more

information on what a good deliverable should include and sound practices for creating exceptional ones.

i. **Integration** The proposed integration process for using the results of a penetration test to their full advantage. It notes that this assumes that the test found something and followed previous phases, and that the deliverable communicates all necessary information for supporting integration.

   The deliverable can be combined with other existing materials such as a risk analysis, security policy, previous test results, and information associated with a security program to enhance mitigation. This integration process can help organizations prioritize and address vulnerabilities, and improve their overall security posture.

   The **first area is mitigation**, which involves addressing any vulnerabilities or threats that were identified during the test. This can range from simple fixes to complex solutions, and may include testing, piloting, implementing, and validating changes to systems.

   The **second area is defense planning**, which involves addressing insecurities in a strategic manner. This includes addressing networks, systems, applications, and policies to ensure sound security practices are employed to minimize the impact of future or undetected vulnerabilities. Defense planning establishes a foundation of security to grow on and ensure long-term success.

   The **third area is incident management**, which is the core element of security. This involves the ability to detect, respond, and recover from an attack, and is an essential part of any security program. The penetration test provides an opportunity to learn about weaknesses and attractive avenues of attack, critical points in the network that may need more attention than others, and aids in formulating an incident response plan.

1.3 INFORMATION SECURITY MODELS

- The concept of "defense-in-depth" in information security. It refers to the practice of implementing multiple layers of security controls that complement each other to provide effective protection against threats. The analogy of a bank is used to illustrate the concept, where multiple security controls such as guards, safes, alarms, security cameras, and locked doors work together to create a formidable defense.
- The paragraph emphasizes the importance of having unique and complementary security controls to increase the odds of detecting and thwarting an attack. However, duplicate controls or those with the same function should be considered a redundancy rather than an additional layer of defense. The paragraph also notes that the interpretation of defense-in-depth can vary, and having multiple firewalls in sequence may not necessarily be considered a true example of defense-in-depth.
- The application of the principle of defense-in-depth to the framework for performing a penetration test. It suggests that building a layered model can help in explaining the act of hacking within the detailed framework. Two models are introduced - one is about the different levels or layers where security controls can be employed, while the other is about a security architecture that helps companies classify different aspects of security. The author believes that these models will help in understanding the framework of a value-based penetration test.
- These models are combined and demonstrated in Figure 4.1. The defense-in-depth model is defined in four layers:
  1. Computer security
  2. Network security
  3. Service security
  4. Application security.

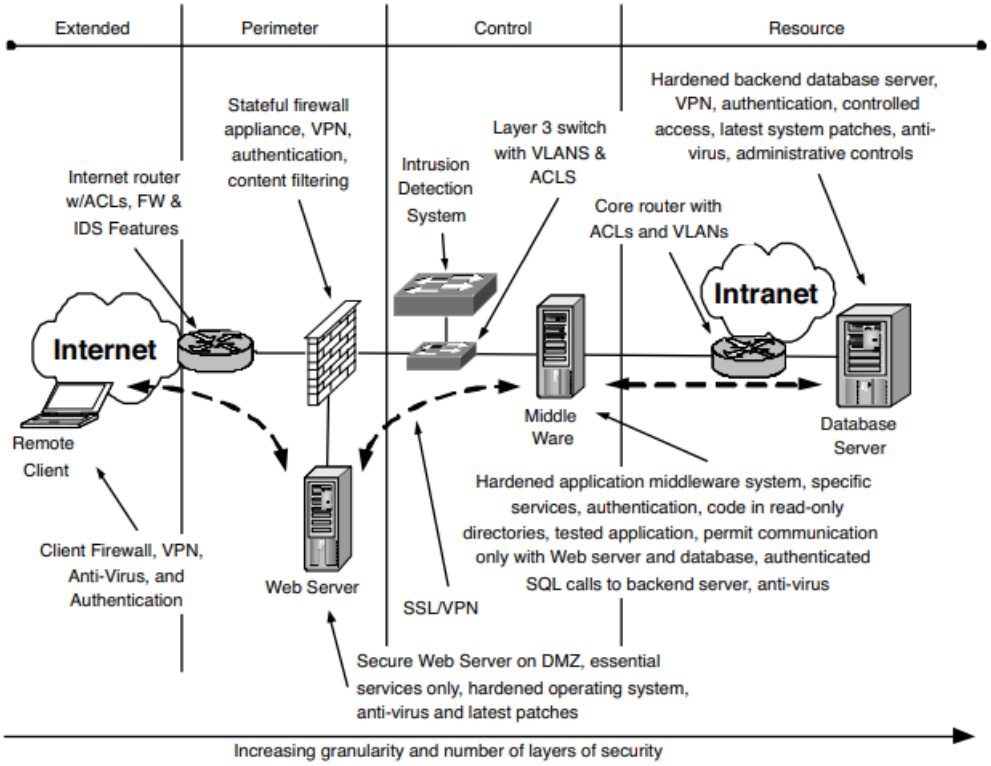

FIGURE 4.1 Defense-in-Depth within a Security Architecture

1.3a COMPUTER SECURITY

- Computer security encompasses a wide range of subjects, such as access control, user account management, software and database security, change control, and development. The operating system acts as the primary interface between hardware, software, and users, making it a crucial component of system security. Popular operating systems include Microsoft's Windows and UNIX, which has various versions such as Linux, BSD, Solaris, and AIX.
- In the past, computers were large and centralized, accessed by dumb terminals through a serial interface, and had relatively simple security controls. User access was limited to mandatory access control (MAC) and shared access to the central computer was tightly controlled by administrators to prevent unauthorized access to files.

Harden A System

- Determining what steps are necessary to harden a system can be very frustrating. There are numerous sources of various tactics for securing a system. There are sample configurations and tools that can be used to configure Microsoft and UNIX systems. The following are some common characteristics of hardening a system.

Physically Secure It

- Many systems are vulnerable to direct access. For this reason, you should make certain that only authorized people can physically access the box. Everything from turning the system off to installing malicious software can be performed on a system if you can get to it.
- Common practices are as follows:
  - Install case locks on publicly accessible systems, such as workstations.
  - Place critical systems in a locked cabinet (cage) in a controlled facility.

- Avoid the use, or installation, of removable media support such as floppies, CD-drives, and removable hard drives.
- Disable or remove support for external access ports, such as UBS ports, COM ports, and keyboard support when applicable.
- Set up a BIOS password to reduce the exposure of someone rebooting the system and making changes to the system.
- Disable the power switch or use a lockable switch.
- Ensure power supplies are secured and redundant. It is one thing to hit the power button; it is another to just unplug it.
- Provide suitable operating conditions such as raised floors and environmental controls.
- Control access to the computer room.

Installing the Operating System

- During the installation of an operating system it is typical to know the role that system will play in the company. When concerned about the security of the system, there are several practices to start you on the right foot.
- Setup practices include the following:

  - Determine if there is a company-approved configuration or system image that is relevant to the role of the system. This means selecting a configuration that is appropriate for the intended use of the system, such as a different configuration for a web server that will be used for internet services compared to an internal development system.

  - Install the operating system from scratch, rather than updating an existing one, to avoid inheriting vulnerabilities, viruses, or poor configurations that may exist on the existing system.

  - Select an appropriate file system format that reflects the needs of the computer, keeping in mind that non-secure file systems such as FAT (File Allocation Table) are rarely used due to security concerns.

  - If given the option, avoid installing any services by default during the installation of operating systems such as Microsoft, RedHat, Solaris, and BSD, among others. Instead, enable services as needed to reduce the likelihood of exposing unused system elements to frivolous attacks.

  - Enable interfaces only when necessary to complete the installation, to avoid exposing the system to unnecessary interaction before it is properly configured. For example, connecting to a different system on the network to collect an application for installation may be required to load a specific module, but this should only be done when necessary.

Get It Running

- At this point you have a half-baked system somewhere between security and doing what you need it to do. However, you're still not ready to start piling on applications. There are some tasks to ensure the system is prepared for more serious hardening.
- Cleanup practices include the following:

  - The first step after installation is to check and configure the system to restart in the expected manner. This involves reviewing the init.d file in UNIX or the startup configuration in Windows to ensure that nothing has been added or removed during the installation process.

  - Create an administrative account, rather than using the root account for managing the system on a day-to-day basis. This provides an additional layer of security, as administrators can use the "SU"

(super user) command to perform specific tasks as needed. By configuring the system to disallow people from logging in as root, greater control over the system can be maintained.

- Disabling unnecessary services is an important step in securing the system. Even if they were not installed during the initial setup, some services may have been added automatically or may be hidden. Disabling services that are not needed can help reduce the attack surface of the system.

- Determine application dependencies to avoid rogue processes when cleaning the system or removing extraneous services and applications. Some applications may be installed to support system administration, and it is important to evaluate their relationships to avoid inadvertently removing necessary components or introducing security vulnerabilities.

Set System Policies

- Now that the operating system is installed and specific services are running, there are administrative configurations that need to be implemented to support moving into a functional role.
- Common administration setup is as follows:

  - **Set up password policies**: This involves establishing guidelines for creating and managing passwords for all user accounts on the system. Password policies typically specify minimum length requirements, complexity rules (such as requiring a mix of upper and lower case letters, numbers, and special characters), and expiration intervals.

  - **Establish an audit function:** This involves enabling logging and auditing features on the system to track and record all system changes, events, and activities. This helps administrators detect and investigate security incidents, as well as troubleshoot technical issues.

  - **Construct directory structure and file permissions:** This involves creating the necessary folders and directories on the system to organize data and applications, and configuring access control settings to ensure that only authorized users can access and modify files and directories. This is an important step in securing the system and protecting sensitive data from unauthorized access or modification.

Accessing the System

- Assuming the box will be accessible over a network, the next phase is to control the type of remote access for users, services, and applications.
- The network setup consists of the following:
  - Implement access control lists restricting only the protocols that are going to be used on the system.
  - Make protocol stack changes. For example, change the number of permitted open connections or shorten the wait time associated with half open connections.
  - Configure the system to accept or deny remote login and remote procedure calls that are associated with execution of remote applications.

Cleanup

- Before installing applications and other things that will affect the security of a system the next step is very important and many still don't do it: applying patches. By the time you get the CDs for installing an operating system there are undoubtedly patches for it.
- There are three types of patches:
  - ***Functionality*** - A patch that fixes or enhances a certain function of the system. For example, how memory is handled, performance of network connections, or adding more options to an administrative program.

- ***Feature*** - A feature patch increases the use of the system, an added feature.
- ***Security*** - A security patch fixes a vulnerability in the system due to unexpected conditions the system is in or a misstep in programming.

1.3b NETWORK SECURITY

- Securing a single system is challenging enough, but securing thousands of connected systems is even more difficult. Network security focuses heavily on access control since it can be outside the system's direct control. Messages sent between computers are contained in packets with logistical information in a header, similar to an envelope containing a letter with "to" and "from" addresses.
- Network devices, such as routers, forward packets from one system to another until the final destination is reached. Communication occurs with discrete messages, like sending letters with different envelopes for each page. Routers use routing protocols to learn the network layout and determine where to send packets. While many network types and protocols exist, this discussion focuses on communication security and associated vulnerabilities rather than specific technologies.
- As with computer security, there are various characteristics of network security. These are summarized in the following list:
  - ***Transmission Security*****:** The protection of data as it is transmitted from one location to another.
  - ***Protocol Security*****:** The construction of packets and how they are processed and used to transmit information.
  - ***Routing Protocol Security*****:** The information that is shared by network devices to work together to support communications.
  - ***Network Access Security*****:** Controlling connectivity from one network to another based on protocol specifics.

Transmission Security

- Network security involves protecting information in transit, such as ensuring that sensitive data is encrypted and authenticated to prevent unauthorized access or changes. Security protocols, such as IPSec, SSL, and SSH, offer encryption and authentication to protect information from unauthorized interactions.
- However, network sniffers can collect packets on a network segment and capture information that is in cleartext. E-mail, FTP, Telnet, SMTP, and POP are examples of protocols that often transmit data in plaintext, making them vulnerable to unauthorized access.
- Applying authentication and encryption to a data stream can help prevent unauthorized access, but it is not a complete solution. Secure communications are an effective and cost-efficient way to protect against common security vulnerabilities.

Protocol Security

- Network security is not only concerned with protecting information in transit but also the protocols used to support communication. TCP/IP is the most commonly used protocol today and is the foundation for many other protocols and services. However, it was developed without much consideration for security, resulting in many protocol weaknesses that can be exploited by attackers.
- One of the most notable attacks is Denial-of-Service attacks, which use basic features of the protocol to bring systems down. Another weakness is IP spoofing, where attackers replace the source IP address of a packet to make it appear as though it's coming from a trusted source.
- This technique can be used to circumvent firewalls, routers, switches, intrusion detection systems, and systems to support an attack. Attackers can also manipulate sequence numbers within the TCP/IP protocol to predict the communication and gain direct access to the server. It is important to note that

vulnerabilities exist in both foundation and higher-level protocols, and these vulnerabilities can be manipulated to circumvent security measures.

- Therefore, it is crucial to apply proper security protocols, such as SSL, FTP, IPSec, and POP, to protect against potential attacks. While secure communications are an effective solution to a common security exposure, they are not a complete solution and should be combined with other security measures to provide comprehensive protection.

Routing Protocol Security

- Routing protocols are critical for the communication between network systems. They enable sharing of network information and facilitate appropriate forwarding of data by a group of devices. These protocols determine the most efficient path for data to travel based on network availability, performance, and cost of connection.
- Figure 4.2 is an example of a large network supported by the OSPF (Open Shortest Path First) routing protocol. OSPF (Open Shortest Path First) is an example of a routing protocol that uses "areas" to define borders for summarizing network routes to different regions or departments.

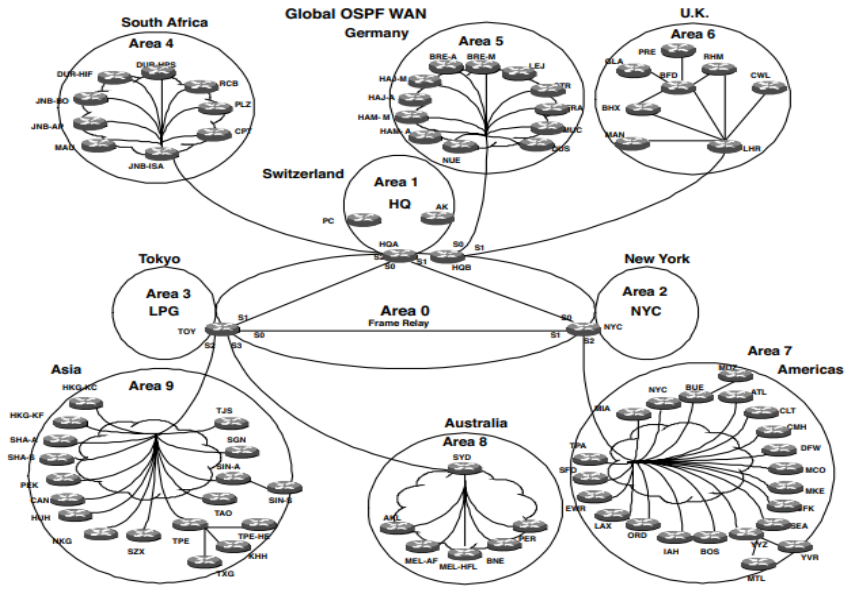

FIGURE 4.2 A Global OSPF Network Design

- Figure 4.3 shows an example of applying MD5 (Message Digest 5) authentication to OSPF communications. Based on Figure 4.3, the Listings 4.1 and 4.2 are sample configurations for Cisco routers using MD5 authentication for OSPF.

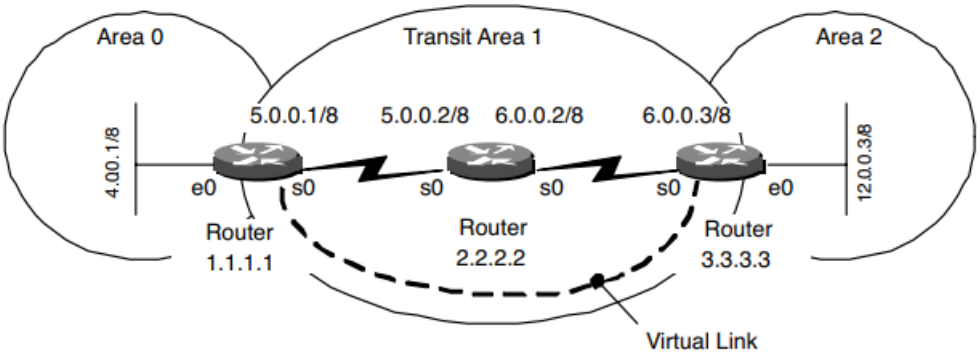

FIGURE 4.3 OSPF Network Design Example Configuration

- However, routing protocols are vulnerable to hackers who can manipulate the routing information in a router to reroute information, change their attack vector, or cause a denial of service. To secure routing protocols, authentication can be applied to the exchange of information to ensure integrity. Virtual private networking, which offers secured communication using encryption and authentication, can also be used to secure routing information between devices.

LISTING 4.1
Router 1.1.1.1

```
hostname r1.1.1.1
interface Loopback0
ip address 1.1.1.1 255.0.0.0
interface Ethernet0
ip address 4.0.0.1 255.0.0.0
ip ospf message-digest-key 1 md5 cisco
!-- The MD5 authentication key is
!-- configured on the interface as "cisco."
interface Serial0
ip address 5.0.0.1 255.0.0.0
clockrate 64000
!
router ospf 2
network 4.0.0.0 0.255.255.255 area 0
network 5.0.0.0 0.255.255.255 area 1
area 0 authentication message-digest
!-- This command enables MD5 authentication for area 0
!-- on the router.
area 1 virtual-link 3.3.3.3 message-digest-key 1 md5 cisco
!-- This command creates the virtual link between Router
!-- 1.1.1.1 and Router 3.3.3.3 after successful authentication.
```

- Figure 4.4 shows an example of using a combination of Generic Routing Encapsulation (GRE) and IPSec to provide isolation and security for OSPF while traversing the Internet. As with typical Internet connections there is a firewall connecting the network to the Internet and an internal router supporting the intranet, in this example, networks 11.11.11.11 and 22.22.22.22. A Generic Routing Encapsulation (GRE) tunnel is created between the two intranet routers, Rodney and House.
- The GRE tunnel provides a conduit for OSPF to interact with other systems without affecting the protocol's ability to "map" the network. IPSec encrypts a simple IP protocol to support routing services.

LISTING 4.2
Router 3.3.3.3

```
hostname r3.3.3.3
interface Loopback0
ip address 3.3.3.3 255.0.0.0
interface Ethernet0
ip address 12.0.0.3 255.0.0.0
interface Serial0
ip address 6.0.0.3 255.0.0.0
!
router ospf 2
network 12.0.0.0 0.255.255.255 area 2
network 6.0.0.0 0.255.255.255 area 1
area 0 authentication message-digest
!-- This command enables MD5 authentication for area 0
!-- on the router.
area 1 virtual-link 1.1.1.1 message-digest-key 1 md5 cisco
!-- This command creates the virtual link to area 0 via
!-- the transit area 1.
```

- When performing a penetration test, routing protocols can be used to learn about the network layout, which is helpful in creating an attack plan. This is especially valuable when network mapping techniques and tools fail, but an insecure router is accessible by a tester who has all the routing tables representing internal systems.

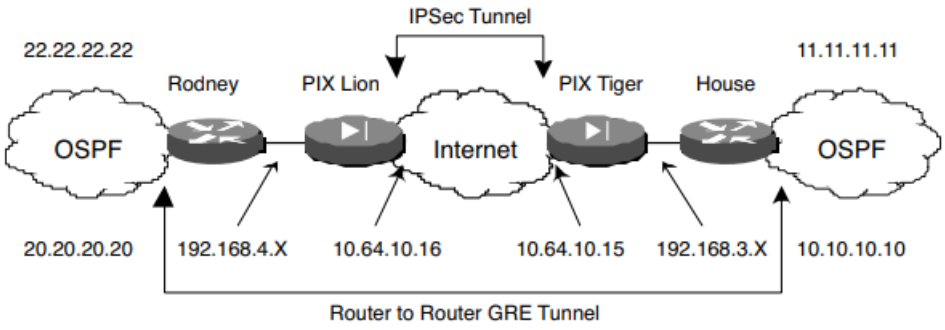

FIGURE 4.4 VPN and GRE Used to Protect OSPF Protocols over the Internet

- In conclusion, routing protocols are the foundation of how data is routed from one location to another in a network. They are vulnerable to attacks and require authentication and virtual private networking for securing routing information between devices.
- A combination of GRE and IPSec can be used to provide isolation and security for OSPF while traversing the Internet. When performing a penetration test, routing protocols can be used to learn about the network layout, which is helpful in creating an attack plan.

Network Access Controls

- Network security is achieved through access controls, which limit communication between systems and applications. Application ports in TCP/IP headers are used to identify specific services associated with a communication. Standard applications are assigned numbers 1 to 1023, while high ports are used for bidirectional communications.

- Firewalls use rules to allow or block packets based on destination port numbers. By limiting access to certain ports, network security is enhanced. Hackers can exploit any aspect of network security controls to launch an attack.

1.3c SERVICE SECURITY

- Services are processes that run on a computer to provide common functions for applications, users, or other services. Services fall into two very similar categories:

    1. *Operational*: A process that provides a service to applications or users for functionality.
    2. *Network*: A process that supports the exchange of information for network services.

- The following are examples of operational services used in Microsoft Windows:
    - ✓ *Security Accounts Manager*: Stores security information for local user accounts.
    - ✓ *Plug and Play*: Enables a computer to recognize and adapt to hardware changes with little or no user input.
    - ✓ *Net Logon*: Supports pass-through authentication of account logon events for computers in a domain.
    - ✓ *Event Log*: Enables event log messages issued by Windows-based programs and components to be viewed in Event Viewer. This service cannot be stopped.
    - ✓ *Logical Disk Manager*: Configures hard disk drives and volumes. The service only runs for configuration processes and then stops.
    - ✓ *Indexing Service*: Indexes contents and properties of files on local and remote computers; provides rapid access to files through flexible querying language.
    - ✓ *DNS*: Resolves and caches Domain Name System (DNS) names. If this service is stopped, DNS names will not be resolved and Internet services not located.
    - ✓ *telnet*: Enables a remote user to log on to a computer and run programs. If this service is stopped, remote user access to programs might be unavailable.
    - ✓ *FTP*: Allows the exchange of files over the network.
- Services are a popular target for hackers due to their potential vulnerabilities and impact on a system. Services have privileged access to other resources, and with numerous services running on a typical computer, there are countless opportunities for attackers. Service insecurity is a predominant avenue of attack affecting millions of computers, as seen in the SQL Slammer worm and the RPC service exploited by the Blaster worm.
- In both cases, security patches were available before the worms were released, highlighting the importance of proper management. Services can be used as a gateway into an operating system or application, making them a substantial threat to company security.
- It is essential to configure security levels for services and limit their access to other parts of the system. Network security can also be improved by applying access controls and limiting the availability of communications between systems or applications.
- Application ports can be used to limit access to a system, and a firewall can look up these ports in the header to determine whether packets should be dropped or passed through for communications. Network security is realized through different controls placed on the interaction of systems and the movement of data, and hackers have the opportunity to interact with any one of these attributes to perform an attack.

1.3d APPLICATION SECURITY

- Applications are the last layer of security in the hierarchy of security measures. They are a collection of utilities, libraries, and executables used to perform a wide range of tasks. Applications can have vulnerabilities and weaknesses that hackers can exploit to gain unauthorized access to a system.

- Some vulnerabilities require complete access to the system, while others are targeted by hackers because they provide an opportunity to obtain greater access. Bugs or errors in application development and design can lead to significant disruptions and create security holes that hackers can exploit.
- Bugs in applications can impact several systems, including routers, servers, workstations, databases, e-mail programs, web browsers, and back-office applications. Hackers can use bugs in programs to gain greater access or support other attacks.
- For instance, hackers can inject code or scripts into an application to obtain data about a system through the error. Buffer overruns are another example of vulnerabilities that hackers can use to implant code and execute it. Application vulnerabilities are the most time-consuming for security professionals.
- Administrators have to review vulnerabilities, determine if the application is vulnerable to an attack, analyze the cost of fixing the vulnerability, and implement the necessary changes to rectify the problem. Overall, bugs in applications are a significant security concern that needs to be addressed to protect systems from attacks.

  - A vulnerability in Microsoft's Internet Explorer (IE) versions 5.5 and 6.0 allows hackers to execute arbitrary scripts remotely using cookies. The error is in how IE handles security zones, which makes it possible for a hacker to run programs embedded in a cookie. As cookies are considered part of the "Local Zone," they are trusted and accepted, allowing hackers to gain access.

  - Snort, an open-source intrusion detection system, is vulnerable to denial of service (DoS) attacks. Snort 1.8.3 does not define the minimum ICMP header size, which allows remote attackers to crash and dump the system by sending malformed ICMP packets. This flaw can be exploited by a hacker to shut down the Snort-based intrusion detection system, allowing them to continue the attack unnoticed or recorded.

  - RealPlayer version 8.0 and earlier is vulnerable to remote code execution. Hackers can execute code that exceeds the length of the header by containing it in the length value of the header.

  - The Microsoft Exchange Server 2000 System Attendant gives the “Everyone” group privileges to the WinReg key, which could allow remote attackers to read or modify registry keys.

  - Microsoft Outlook 8.5's seemingly harmless feature, "Automatically put people I reply to in my address book," does not check to see if the "reply to" address matches the "from" address. As a result, a remote attacker can spoof a legitimate address and intercept messages intended for others, with this option enabled.

  - Internet Explorer versions 5.01, 5.5, and 6.0 allow remote attackers to read files on a remote system by exploiting malformed requests to the GetObject function, bypassing some of GetObject's security checks.

  - In Microsoft Windows NT and Windows 2000, a trusting domain that receives authorization information from a trusted domain does not verify if the trusted domain is authoritative for all listed Security Identifiers (SIDs). This vulnerability allows a remote attacker to inject SIDs from other domains into the authorization data of the trusting domain, giving them Domain Administrator privileges.

- A good application development policy should define requirements and coding standards. During a code review of an application, the standards and practices can be compared directly to the application architecture in an effort to reduce vulnerabilities at the time of development. When executed correctly, code reviews will uncover many straightforward but dangerous security violations, such as:

- ✓ Buffer overflows
- ✓ Race conditions
- ✓ Tainted input
- ✓ Format string issues
- ✓ Trust management
- ✓ Third-party package connectivity
- ✓ Input validation
- ✓ Temporary file or memory usage
- ✓ Poor cryptography
- ✓ Appropriate logging and auditing

- Writing secure code is a challenging task for application developers, as they need to consider all potential points of attack that may be exploited by malicious actors. However, the task becomes even more difficult when developers introduce their own vulnerabilities.
- Some of these may be simple oversights, but they can represent significant risks to customers. For example, Microsoft SQL Server and Microsoft Data Engine shipped with a null default password on the administrative account in November 2001, making it vulnerable to attacks if the password was not changed.
- Similarly, Oracle Database Server version 9iAS had several default log-in accounts with known usernames and passwords that were made publicly available on the Internet, making it easy for attackers to gain access to an Oracle server. In both cases, these security issues could have been prevented if the software vendors had included security as part of their development process.
- While software vendors often provide patches, updates, or workarounds to fix these oversights, it is crucial that developers prioritize application security to prevent such vulnerabilities from occurring in the first place. Overall, application security requires a significant investment of time and effort to ensure that software is not vulnerable to human error.

1.3e SECURITY ARCHITECTURE

- Security architecture for companies in today's internet-enabled economy. With the integration of systems and applications with the internet becoming increasingly common for businesses, security has grown more complex both in the business environment and philosophically.
- As a result, a security architecture is necessary to provide a point of reference for decision-making when changes in demands and environment occur. This architecture must not only exist but also interact with the business objectives and provide a fundamental guide for new technology and requirements. The security architecture should allow flexibility in operations and provide a variety of access management and layered security to accommodate the dynamics of business, technology, and environments.
- Examples of security architectures are provided by various organizations, such as the Department of Defense, National Security Agency, Federal Bureau of Investigation, National Institute of Standards and Technology, Internet Engineering Task Force, and CERT. While they range in complexity and cost, there is a consistent theme among them that can be applied to today's internet-enabled economy.

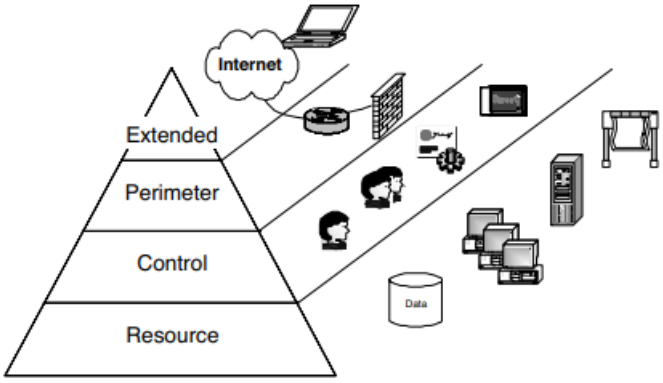

FIGURE 4.5 Example of a Typical Security Architecture

- Commonalities among many of the architectures that are available are four layers that can be identified to promote sound security integration and management of technology, information, and policy (see Figure 4.5).
  - ✓ The resource layer is where services and data reside. It is the home of servers, applications, databases, workstations, and storage.
  - ✓ One of the more critical and complex is the control layer, which provides identity and access management services. Moreover, the control layer is the point where policy becomes reality in the technical space. It provides management with the policy and is the point where policy is bound to data to promote greater authorization across the other characteristics of the entire security architecture.
  - ✓ There is the perimeter layer, which enforces a logical boundary between the Internet and the intranet, departments, applications, and even users.
  - ✓ Finally, the extended layer is a growing entity in its own right. This represents the externally facing envelope of influence and security, such as remote access risks, application access, and E-commerce.
- In today's fast-paced economy, businesses face various challenges that require them to work more closely with external entities to maintain an agile value chain. Managing internal and external relationships and information flow is crucial to achieving this. Security is often seen as rigid, but it cannot remain static due to mergers, acquisitions, or environmental changes.
- A security architecture is a policy-supporting overlay that can interact with users, resources, and external influences. To provide the desired flexibility, the architecture must be well-conceived and built for general purposes. Each layer should have its own characteristics that can be interchanged with other layers to optimize functionality. Each layer is loosely coupled with the next, reducing redundancy and allowing for flexibility.
- However, security infrastructures rarely follow a comprehensive architecture due to legacy systems, complex business requirements, and mergers and acquisitions. As a result, organizations often focus on a point solution or strengthening within a layer, which weakens the interaction between layers and can represent a vulnerability.

a) Resource Layer

- Resources in an organization can include systems, applications, users, databases, services, printers, networks, operating systems, and data. These resources are critical to business operations and must be protected and accessed securely. However, not all resources require the same level of security, and identifying them can be challenging for large organizations with multiple business units.
- It is crucial for security managers to understand the resources and their value to the company's continued success. Without this knowledge, a successful penetration test cannot be conducted, and

vulnerabilities cannot be translated into actionable remediation plans. It is unlikely that businesses will invest in fixing security holes that cannot be financially justified.

b) Control Layer

- The control layer manages access to resources and groups the systems that perform this task. Ideally, all identification, authentication, and authorization would be controlled by a single system, but legacy systems and different security control approaches have resulted in a fragmented security architecture. Many organizations have multiple types of authentication systems that lack centralized management or provisioning.
- However, the trend is shifting towards integrating identity management solutions to provide a common authentication system and access controls in heterogeneous environments. The control layer poses a challenge for penetration testing, and testers may find ways around it by seeking vulnerabilities or exploiting holes in the system. Despite being complex and challenging to identify and categorize, understanding the control layer is crucial for establishing test goals and interpreting results.

c) Perimeter

- Perimeter security is the initial layer of protection in a security model that separates an organization's network from external networks. While firewalls are necessary for defining the perimeter, additional technologies like intrusion detection and prevention systems facilitate secure communications between trusted and untrusted networks.
- However, relying solely on perimeter security is flawed, and companies must implement additional security measures like access controls and monitoring. Hackers have historically tried to bypass firewalls through "fire walking," leading to new solutions from vendors. Companies must regularly tune their systems to thwart sophisticated attacks, but they often become overwhelmed with alarms and reduce sensitivity.
- Penetration testing can establish an effective baseline and fine-tune perimeter security technology, but companies must make logical decisions and understand benefits and losses while withholding information about IDSs during testing.

d) Extended

- The extended layer of security refers to the protection of information assets beyond the perimeter of the network. This includes various access points, such as email, PDAs, wireless messaging, and VPNs. However, the exposure of digital assets at the termination point outside the perimeter is a concern. Extranets, remote users, and customers also fall within the extended layer of security.
- Organizations must address legal issues, regulatory requirements, and SLAs to ensure security measures are enacted correctly. Identifying users and systems of partners on both sides of the network is critical for security.
- Extended networks pose significant security challenges for organizations. Penetration testers face a virtual line between the customer's network and the partner's network that hackers can exploit. If partner networks and communications are included in a penetration test without proper agreements, the result could be disastrous.
- The risks associated with extended networks include technology security, access management, legal agreements, and support issues. Detailed documentation of the architecture and environment is critical to planning a sound security model and making decisions on the scope and scale of a penetration test.

1.4 INFORMATION SECURITY PROGRAM

- A security program is essential for managing the complexities of information security in organizations. It provides guidance and a foundation for implementing security throughout a company, and without

one, security-related activities are typically only tactical. An information security program manages overall business risk by identifying and quantifying risk in a meaningful way.

- Risk tolerance varies based on the organization, and a well-constructed security program enables the identification of assets, their value, and impact on the business in the event of loss or damage. Security programs promote best practices for securing information systems and managing risk. Having a defined security program is crucial for integrating the results of an ethical hack into the organization's business needs.

1.4.1 SCOPE OF INFORMATION SECURITY PROGRAMS

- An information security program aims to protect an organization's information assets by preserving confidentiality, integrity, and availability. However, it is common for organizations to limit security programs to network- and host-based security, overlooking physical security and personnel.
- A **comprehensive security** program requires a multidisciplinary approach to risk analysis that considers all aspects of an organization's information assets.
- **Physical security** includes physical access controls and handling procedures for physical media. Even with robust network security controls, confidential printouts and unsanitized magnetic media can be obtained through activities such as dumpster diving. Personnel education and awareness are also essential elements of a comprehensive security program. Role identification allows for the specific assignment of information security responsibilities and role-based training to improve awareness of potential threats.
- An **effective security** program is not only a technical problem but also requires a multifaceted solution. Ethical hacking, or the act of exploiting assumed layers of control, can be used to test security and identify weaknesses. Security must be applied in layers to ensure that there are no gaps between the controls that protect information as it is created, transmitted, and stored.

1.4.2 THE PROCESS OF INFORMATION SECURITY

- Effective information security is an iterative process (see Figure 5.1) that must identify and mitigate present risk, as well as allow feedback to mitigate future risk. When creating an information security program, starting at a higher level of risk ensures alignment with business elements rather than technical nuances.
- This nontechnical, business-level approach to security allows for efficient and applicable security planning and ensures alignment with the company's economics. Technical specifics can be taken into consideration during other phases of the framework.

a) IDENTIFY RISK

- The identification of risk involves assessing assets, threats, and vulnerabilities. Assets can be tangible or intangible, while threats are potential risks to assets and vulnerabilities are weaknesses that can be exploited.
- The first step in information security is to identify these risks. Ethical hacking can be useful in identifying vulnerabilities, but proper planning and assignment of scope is necessary for effective testing.
- The scope of vulnerability depends on its impact on business success, and traditional vulnerabilities are not always associated with technology. Risk analysis can provide empirical data to create an effective security program.

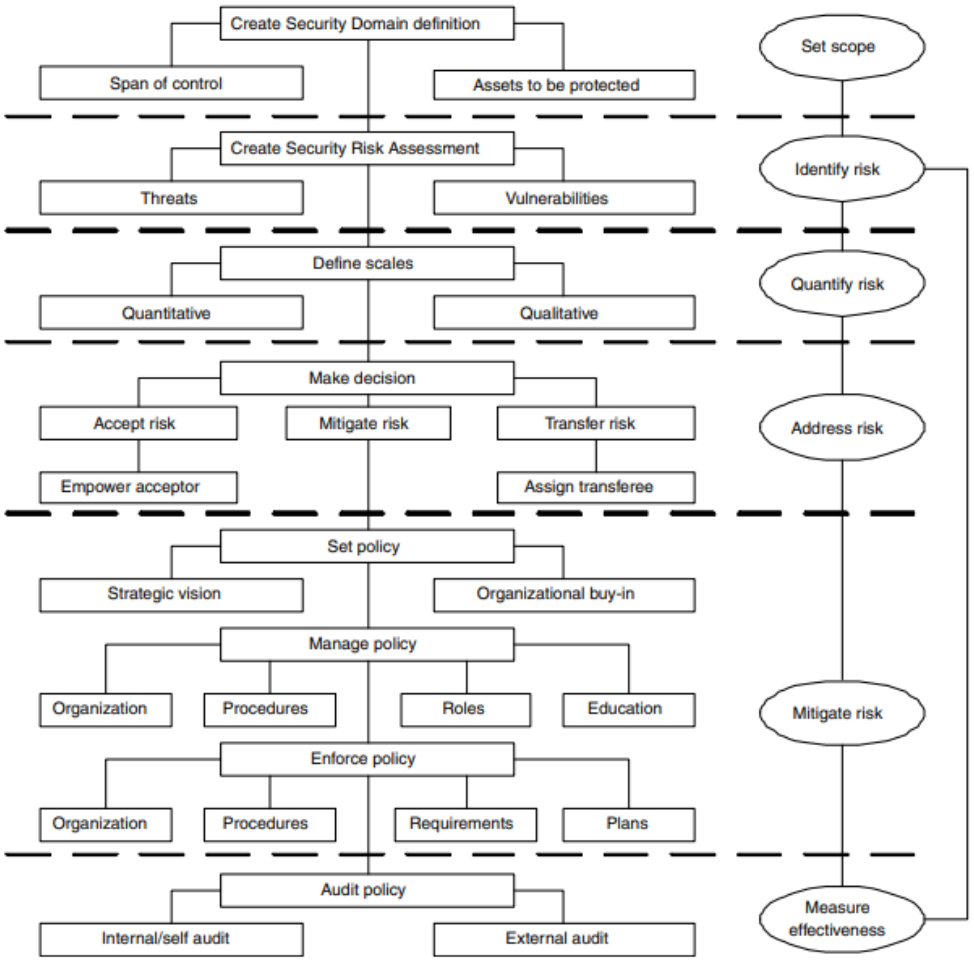

FIGURE 5.1 The Process of Risk Management within a Program

Risk Analysis Process

- The process of risk analysis, which involves identifying potential threats to a system or information, evaluating the probability of a threat agent exploiting a vulnerability and causing damage, disruption, or loss, and determining the resources required to protect against those risks in a cost-effective and proportionate manner. The main purpose of performing a risk analysis is to align an organization's security program objectives with its business objectives and requirements, and to quantify or qualify the impact of potential threats on the loss of business functionality. As demonstrated in Figure 5.2, A regular risk analysis can help a company focus its security resources where they are needed most, and a penetration test can expose vulnerabilities that would be considered high risk for a common infrastructure without a firm understanding of the value of the exposed assets.
- The two main results of a risk analysis are the identification of risks and the cost versus benefit justification of the countermeasures. It emphasizes the need for businesses to determine how much risk exposure they can afford and implement proportionate security measures to protect valuable systems and assets.
- The benefits of conducting a risk analysis include providing investment guidance for security controls, influencing hardware and software system design decisions, and focusing security resources on areas most at risk. Regular risk analysis and penetration testing help organizations continually improve their security program and control their risk exposure.

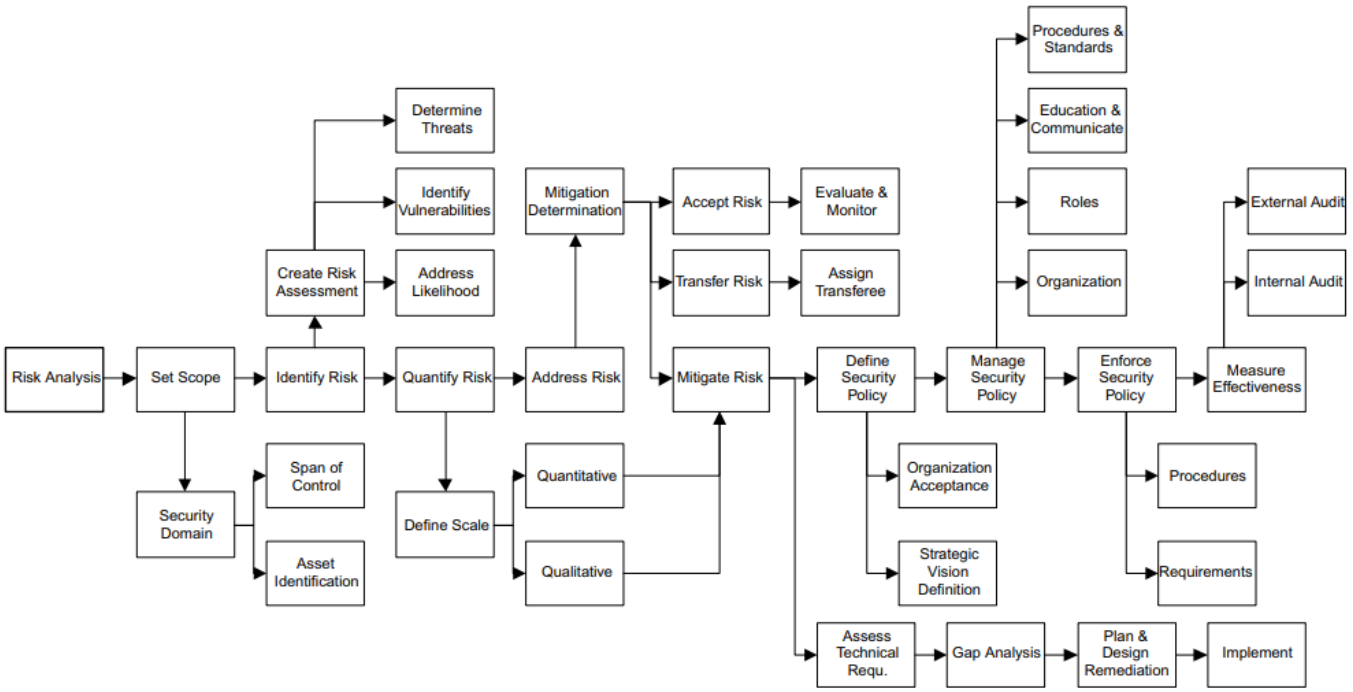

FIGURE 5.2 Detailed Risk Analysis Process

- The steps involved in performing a risk analysis. Initially, the core business functions and requirements , understanding the value of networking and application assets, and establishing relationships among business units, partners, and customers. The organization's structure and fundamental business processes are then broken down into logical elements in Figure 5.3 to determine the impact of loss in case of any threat.
- The analysis then assesses the technology related to providing critical business functions and determines their exposure to threats and likelihood of exploitation. The implemented security measures are identified and assessed, and their risks are mitigated based on their importance in realizing security. Finally, the other technical elements of the organization are assessed to determine their exposures and the data they maintain.
- Risk analysis is a complex and time-consuming process that involves identifying and measuring the value of digital assets, determining input and output requirements, and assessing threats based on vulnerabilities and their likelihood of exploitation.
- Ethical hacking is popular among companies to gauge the severity of vulnerabilities, but it is essential to have a firm understanding of asset valuation to obtain meaningful results. Companies approach risk differently, and some use ethical hacking as an initiating factor for investment or a measuring device to validate assumptions made within the risk analysis project. It is crucial to integrate regular tests into an ongoing security strategy to stimulate risk management continually.

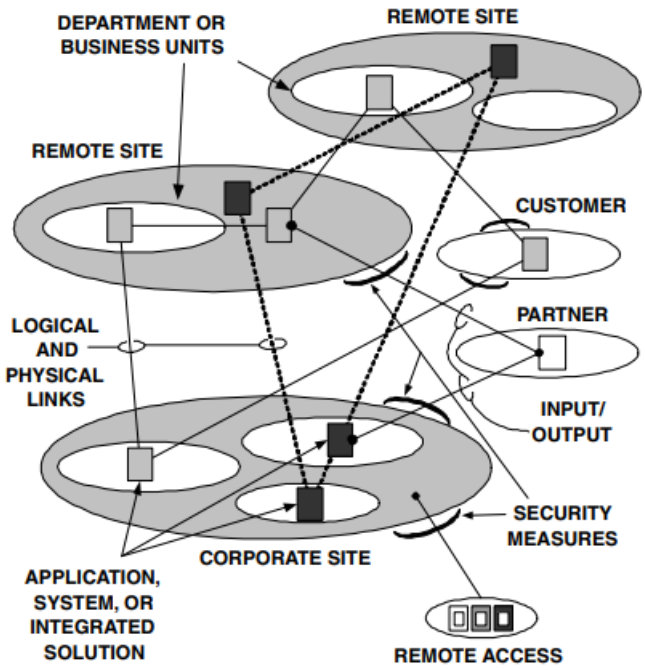

FIGURE 5.3 Breaking the Organization into Logical and Some Physical Components to Initially Simplify the Process

b. QUANTIFY RISK

- The importance of quantifying identified risks to allow for prioritization and the development of a risk mitigation strategy. The quantification scheme should take into account the nature of the organization and be of benefit to decision makers. Prioritized risks can be addressed first, and a foundation can be set for logically applied security, which may require revenue.
- The quantification scheme can be based on dollars per event for E-commerce organizations or a relative ranking of high to low for non-commercial organizations. Ultimately, risks can be ranked based on predetermined and consensual criteria.
- The two main methods for assessing risk when creating a business case: quantitative and qualitative.

  Quantitative analysis involves the use of data to quantify amounts of potential loss, such as the amount of money or data that could be lost due to an attack. Annualized Loss Expectancy (ALE) is one algorithm that calculates the expected loss by multiplying the amount of loss with the Annualized Rate of Occurrence (ARO). Quantitative analysis is useful for determining the financial impact of an attack on resources.

  Qualitative assessments are based on a forecast of potential loss, taking into account various calculated factors such as the ARO and Exposure Factor (EF). These types of assessments are commonly used by organizations with high market-value-to-asset ratios. One major source of operating risk in the IT environment is the incompatibility of technical systems with business strategy. To mitigate this risk, enterprises should establish a framework for aligning technology decisions with business demands and risk.

- It is important for enterprises to document the risks and underlying assumptions involved in their business arrangements to adjust the relationship as assumptions evolve or prove to be inaccurate due to the rapid pace of business and technology change.

Inherent risk

- Inherent risk refers to the potential risks or threats that are inherent in the networking, applications, services, or systems of an organization. These risks can arise due to various unrelated faults that may combine to create a significant vulnerability.
- To mitigate inherent risks, organizations typically use multiple layers of security measures, each with different levels of granularity and focus. However, it's possible that the interaction between these security systems may not support end-to-end security, leaving vulnerabilities in the system.
- There are two levels of control within inherent risk: pervasive and detailed. Pervasive controls are those that are spread throughout the entire organization or have the potential to be spread throughout the organization. The level of pervasive control should be considered at the appropriate level for the organization. Detailed controls, on the other hand, are specific to the systems within the organization and the resources responsible for them. Both pervasive and detailed controls are essential to manage inherent risk effectively.

Control Risk

- Control risk refers to the potential level of harm or loss that can occur within an enterprise due to weaknesses that are not prevented, detected, or corrected in a timely manner by the internal control system.
- The level of control risk is typically considered high during the risk analysis process unless there are identified and effective internal controls that have been tested and proven to operate appropriately. For example, a manual review of system logs would increase the level of control risk, while an automated process for processing logs would reduce the level of control risk.

Detection risk

- Detection risk, on the other hand, refers to the risk associated with the ability to detect an attack or event. In an enterprise, the detection risk associated with identifying breaches of security in an application system is usually high due to poor monitoring practices or poorly tuned technology.

HANDLING RISK

- There are four methods to eliminate or reduce the level of risk associated with the vulnerabilities and asset valuation. Four methods are identified: transference, denial, reduction, and acceptance.
  - **Transference** involves transferring risk to an insurance company or a third-party provider.
  - **Denial** involves ignoring the risk and can be dangerous.
  - **Reduction** involves implementing countermeasures to reduce risk, such as modifying technology or making business-related changes.
  - **Acceptance** involves accepting the risk and understanding the potential losses associated with it. Enterprises may accept a specific level of risk when the cost-to-benefit ratio indicates that the cost of mitigating the risk exceeds the risk itself.

Address Risk

- Address risk by prioritizing it and making informed decisions. Options include accepting, transferring, or mitigating risk based on factors such as probability, harm, and cost. Risk acceptance may be justified if the cost to mitigate is higher than the value of the asset being protected.
- Risk transfer is common in the insurance industry, while mitigation involves deploying control mechanisms to lower the risk's probability, harm, or cost. Examples of control mechanisms include access control strategies, encryption, and backup strategies.

Mitigate risk

- To mitigate risk, an organization must select and deploy controls, starting with a high-level risk-based mitigation strategy. This strategy should outline the organization's goals and enable the definition of subordinate controls such as standards, processes, procedures, configurations, and devices. Multiple layers of supporting controls are common, and a management infrastructure is necessary to maintain their effectiveness and relevance.

Measure Effectiveness

- The need for organizations to measure the effectiveness of their information security processes and controls. The process should be iterative and adaptable, and there are regulatory and organizational reasons for doing so.
- Security services management provides guidance on developing metrics to identify the adequacy of existing security controls, policies, and procedures, and to justify security control investments.
- Metrics should be based on defined security performance goals and objectives, and they monitor the level of implementation, effectiveness, and efficiency of security controls, and identify opportunities for improvement.

1.4.3 COMPONENT PARTS OF INFORMATION SECURITY PROGRAMS

- Any information security program will consist of component parts, as shown in Figure 5.4, that implement the process of information security.

a) Risk Assessment

- A risk assessment in information security is a process of identifying and quantifying potential risks, which is essential for addressing them. The first step in the risk assessment process is to define the initial security domain.
- This involves acknowledging the span of control and relevant assets and defining physical and logical boundaries in accordance with the security architecture model, which includes extended, perimeter, control, and resource layers. This definition is used to modularize the security program and establish the scope of other program components, such as the incident response plan.
- The modularization of the security program offers cohesiveness and flexibility, as well as a vehicle to document due diligence. The risk assessment is a living document with established ownership and review, and any incident response feedback can seamlessly feed back into the risk analysis process to close the loop.
- The accuracy and thoroughness of the risk assessment is essential for its effectiveness, and the advantages of modularity in a security program warrant extra effort in the initial definition of risk assessment security domains.

b) Management System

- The information security management system is designed to address risk, whether it is accepted, transferred, or mitigated. It is increasingly being viewed as analogous to Total Quality Management systems and is adopting internationally recognized standards. One such standard is ISO17799, which focuses on ten functional control areas, including:
  - ✓ **Information Security** Policy addressing management support, ongoing commitment, and direction in accomplishing information security goals;
  - ✓ **Organizational Security** addressing the need for a management framework to create, sustain, and manage the security infrastructure;

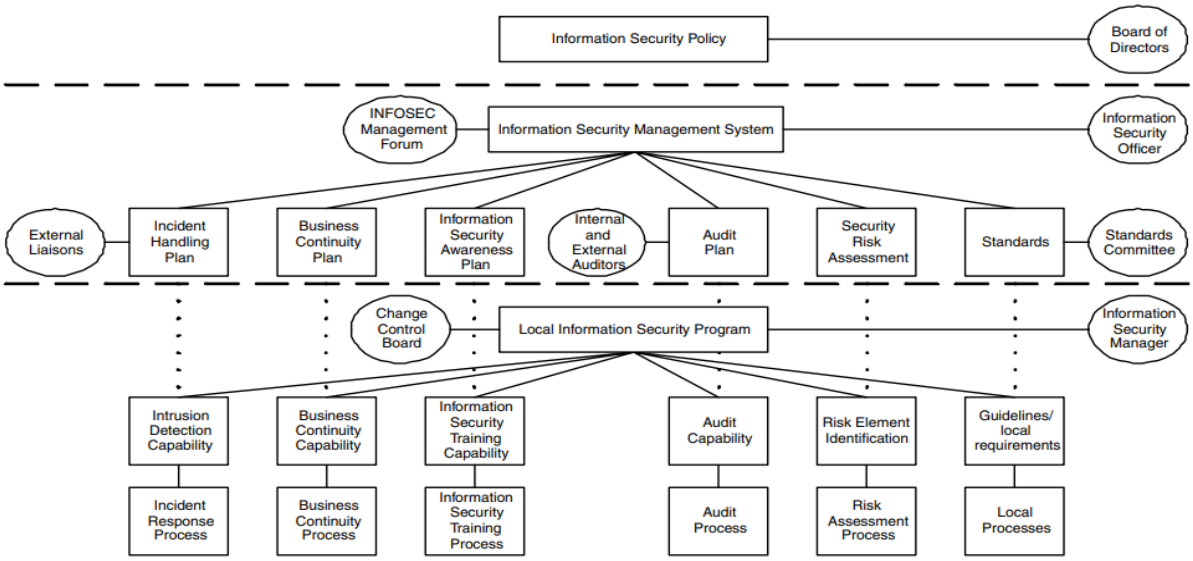

FIGURE 5.4 Example of an Information Security Program Structure

- ✓ **Asset Classification** and Control addressing the ability of the security infrastructure to protect organizational assets;
- ✓ **Personnel Security** addressing an organization's ability to mitigate risk inherent in human interaction;
- ✓ **Physical and Environmental Security** addressing risk inherent to the organization's premise;
- ✓ **Communications and Operations Management** addressing an organization's ability to ensure correct, secure, and repeatable operation of its assets;
- ✓ **Access Control** addressing an organization's ability to control access to assets based upon business and security requirements;
- ✓ **System Development and Maintenance** addressing an organization's ability to ensure that information system security controls are both incorporated and maintained;
- ✓ **Business Continuity Management** addressing an organization's ability to counteract interruptions to normal operations; and
- ✓ **Compliance addressing** an organization's ability to remain compliant with regulatory, statutory, contractual, and security requirements.

- Security management based upon ISO17799 takes a holistic approach to information security, evaluating all aspects of an organization's ability to manage risk. The ten functional control areas provide a checklist of items to be assessed when creating a security program and selecting controls.
- Security management systems define the functional requirements of the security architecture model control layer, and their scope and requirements are determined by the results of the risk assessment, which is fed by a penetration test. Components of the system include security organizations, codified practices, and ancillary support programs.
- Security organizations address the individual's role in the security program.
  - ✓ **Functional Roles** allow assignment of specific security responsibilities such as Information Security Officers.
  - ✓ **Information Security Management Committees** are chartered with specific tasks such as Configuration Control Boards.
  - ✓ **Multidisciplinary Management** Forums are tasked with promoting information security awareness throughout the organization with codified practices that refine an organization's risk mitigation strategy to a level of granularity that can be implemented.
  - ✓ **Policies** express conceptual goals of upper management defining the risk mitigation strategy.
  - ✓ **Standards** define measurable requirements in support of policy goals.

- ✓ **Guidelines** offer best practice advice on how to meet standard requirements.
- ✓ **Procedures** furnish step-by-step instructions to create a consistent and repeatable process.

- Ancillary programs are designed to address risks that are not covered by security organizations or codified practices. They may be managed externally and may liaise with the security program in some organizations. For instance, business continuity may operate independently, and security awareness may be managed by HR or training.
  - ✓ **Business Continuity** programs ensure the sustainability of the organization.
  - ✓ **Incident Management** programs respond to anomalies.
  - ✓ **Security Awareness** programs educate an organization's personnel on information security issues.
- Creating a security management system is unique to each organization and there is no one-size-fits-all approach. Any implementation must be supported by identified risks and gain the full support of upper management. Organizational culture and politics must also be considered. Buy-in from stakeholders at all levels is crucial to initial success and ongoing effectiveness.

c) Controls

- Controls come in many forms, including physical devices, configurations, roles, and processes, affecting networks, platforms, roles, and operations. Many controls require subordinate or supporting controls. For example:
- A firewall is a network control device used to enforce network access and service requirements. The firewall requires:
  - ✓ A supporting procedure for authorized users and services
  - ✓ A supporting role to administer the device
  - ✓ A supporting organization for configuration control
- A sniffer is a network control device used to monitor traffic for both network management and anomaly detection.
  - ✓ A supporting monitoring policy may be required to mitigate an additional risk of illegal eavesdropping or invasion of privacy.
- Hardening scripts are platform controls used to modify system configurations to minimize effectiveness of common system exploits.
  - ✓ role must track and update the scripts.
- System logging is a control that includes:
  - ✓ A device such as a log server
  - ✓ A configuration to enable logging on each device
  - ✓ A role to analyze the log files.
- Functional role definitions assign and evaluate infosec responsibilities and training. An effective system ensures everyone knows their duties and is trained to respond accordingly. Procedural controls guarantee a consistent and repeatable infosec process.
- Controls, such as standard operational procedures, standardize outcomes. They implement management's risk mitigation strategy validated by risk assessment.

d) Maintenance Plan

- An effective security program needs regular maintenance due to the evolving threat environment. The maintenance plan includes program review and audit to ensure the protection of the security architecture model. The review should start at the top with management reaffirming program goals yearly.
- Security risk assessments, procedures, and standards should be analyzed for continued relevance. Program audit should measure effectiveness and introduce findings for program enhancement.

- Regularly scheduled internal or self-audits can detect unauthorized changes or vulnerabilities. External audits, which may be legally required, provide unbiased third-party evaluation based on standards or legislative requirements.

1.4.4 RISK ANALYSIS AND ETHICAL HACKING

- The role of risk analysis and ethical hacking in the world of information security is to assess the security posture of an organization and identify vulnerabilities that could be exploited by malicious actors. The decision to use one over the other is often based on factors such as interpretation, scale, goals, and cost.
- While risk analysis is collaborative and involves working with the company to learn about the environment and identify potential risks, ethical hacking is typically autonomous and involves direct interaction with the system to identify vulnerabilities.
- A risk analysis can be very focused and specific, evaluating a particular solution, department, or application in a short period of time. On the other hand, an ethical hack can be a large-scale endeavour if the entire target company is to be evaluated.
- Ultimately, the choice between a risk analysis and an ethical hack depends on the specific goals and needs of the organization. Both techniques can be effective in identifying vulnerabilities and improving security, and it's up to the company to determine which approach is best for them.
- Using both risk analysis and ethical hacking in conjunction can be effective in assessing an organization's security posture (see Table 5.1). A risk analysis determines the value of assets and evaluates their exposure to threats, including evaluating security controls, while ethical hacking is most valuable in identifying vulnerabilities and determining the level of effort required to exploit them. Ethical hacking also helps evaluate security controls and finds weaknesses in their implementation and use.

TABLE 5.1
Role of Ethical Hacking and Risk Analysis in Evaluating Security

Evaluating Threats and Vulnerabilities	Determining Effectiveness of Security Controls	Establishing Value of Assets
Ethical hacking	Ethical hacking and risk analysis	Risk analysis

- There are situations when using ethical hacking is clearly more effective than performing a risk analysis, and vice versa. However, there are pros and cons for each, and the choice between the two depends on the specific needs and goals of the organization.
- This provides examples of scenarios and the typical assessment type employed, along with the pros and cons of each. The scope and scale are considered interchangeable and cannot be used exclusively to express one type over another.
- Table 5.2 provides a general perspective of the differences between ethical hacking and risk analysis given some basic scenarios, highlighting the diversity in approach and results.
- The ethical hacking and risk analysis are both valid approaches to security assessments and provide valuable information for improving security practices.

TABLE 5.2
Pros and Cons of Ethical Hacking and Risk Analysis

Scenario:	Assessing Security of Internet-Facing Infrastructure	
Ethical Hacking (Typically Employed)	**Pros:** Identifies technical vulnerabilities Determines exposure to threats Establishes the level of effort required to exploit a vulnerability Provides a perspective of the infrastructure from an unknown entity (i.e., Internet public, competitor, etc.) Technically comprehensive (scan entire networks and groups) Provides information on necessary tools and tactics required to attack firewalls, services (e.g., DNS, FTP, etc.), and other infrastructure elements	**Cons:** Does not consider management practices and security policy Potentially affected by firewall or other chokepoint capabilities Exposure to detection by IDS/IPS or other monitoring Potential for adverse events (e.g., downtime, damage, etc.) Does not provide information or recommendations regarding elements outside of immediate observation Does not take asset value into consideration (Note: this is performed only through the tester's perception, not documented asset classification)
Risk Analysis	**Pros:** Considers all aspects of information security: technical configurations, management, operations, and policy (among others) Does not present a risk to the operations of Internet applications and systems Comprehensive configuration analysis of routers, firewalls, and systems Provides a detailed analysis of risk to Internet-facing systems, networks, and applications based on traditional Internet threats	**Cons:** Vulnerabilities are determined through investigation, not empirical evidence from system interaction Assumes level of effort to exploit a vulnerability Assumes potential vectors of attack (i.e., does not test for alternate routes to assets, but assumes them based on infrastructure Performed based on sampling or light vulnerability scanning (potentially not comprehensive)
Results	**Ethical Hacking:** Itemized list of vulnerabilities found on the Internet-facing systems An understanding of depth attained from the Internet Detailed analysis of exploitation, tools, and tactics used against the identifiable systems Raw data from the test Recommendations for remediation	**Risk Analysis:** Detailed analysis of security policies and practices used to manage the security controls Analysis of security architecture and recommendations for modification Asset valuation and exposure to common Internet threats Recommendations for remediation

Scenario:	Assessing Security of Specific Custom Web Application	
Ethical Hacking (Typically Employed)	**Pros:** Directly tests user data input and potential for processing errors Evaluates any client-side scripting, applications, or plug-ins Tests potential performance issues Can expose technical weaknesses permitting access to private information Manipulates cookies or other programming attributes to exploit the application	**Cons:** Does not include (typically) access to code or application elements not published or provided Does not address the planned applications developments Is not aware and cannot clearly evaluate the infrastructure attributes Does not address the management, operations, or processes supporting the application
Risk Analysis	**Pros:** Evaluates the supporting infrastructure and can make security recommendations on information flow controls Access to supporting data, systems, and business data to specifically determine level of impact Evaluates authentication procedures and interaction with supporting elements Can clearly determine the impact to the organization in the event of an outage or breach of security Identifies errors and opportunities for improving application development processes	**Cons:** Vulnerabilities in the application are based on code, process, and previous development phases and not on technical observation May not address client-side technical elements and make assumptions on remote system vulnerabilities Does not look for other, unrelated technical avenues for attack Cannot clearly evaluate the options to threats given various forms of attack
Results	**Ethical Hacking:** Detailed list of vulnerabilities and the level of access attained from exploitation Comprehensive understanding of software flaws and the resulting immediate impact	**Risk Analysis:** Detailed analysis of the potential impact in the event of attack Evaluation of software development practices Security review of the code

Scenario:	Assess Level of Risk from Internal Employees	
Ethical Hacking	**Pros:** Perform social engineering from outside or as an employee to evaluate the level of access and impact of an internal resource Can use vulnerability scanning tools to seek opportunities for greater access Directly exploit vulnerabilities (i.e., access secured areas, collect materials from other employee's desks, system access, etc.)	**Cons:** Potentially time consuming Limited to approved social engineering testing options Limited to the experience and capability of the tester Not exposed to defined policies, roles and responsibilities, and management processes Exposed to discovery
Risk Analysis (Typically Employed)	**Pros:** Evaluates the entire infrastructure for potential physical, network, and system (application) access Can evaluate the level of security controls based on business requirements Evaluates the existence of various level of controls and implementation Exposed to the interdependencies related to systems, departments, geography, and partnerships	**Cons:** Does not clearly evaluate the access of a given employee Must address all elements of the internal environment, even if a focused effort Does not test specific applications or technical solutions to determine discrete access
Results	**Ethical Hacking:** A detailed analysis of potential problems from one or a small group of employees Provide technical insights to internal network and application vulnerabilities Can provide specific materials and access available to internal employees and communicate the results	**Risk Analysis:** Detailed analysis of potential threats based on internal controls and configuration Analysis of employee management practices Evaluation of internal controls, policies and procedures, and recommendations

Scenario:	Assess Security of Internal Network or Segment	
Ethical Hacking (Typically Employed)	**Pros:** Provides greater insight to the scope of opportunities to internal employees to interact with systems and other networks Identifies discrete vulnerabilities at all layers in the network (i.e., physical, IP, services, systems, and applications)	**Cons:** Due to the openness of the infrastructure, it significantly increases the potential for affecting business operations Can result in an inordinate amount of vulnerabilities to sift through to determine next steps Assumes internal threats are sophisticated
Risk Analysis	**Pros:** Evaluates the infrastructure through controlled observations rather than explicit testing Not limited to the immediate technical environment and conclusion can be determined based on business-level information Information about vulnerabilities is typically associated with architecture and process (i.e., configuration management, access controls) as opposed to specific vulnerabilities	**Cons:** Does not clearly represent the perspective from an internal system on the network, or someone with specific credentials Does not typically provide specific vulnerabilities about systems or applications based on direct interaction
Results	**Ethical Hacking:** A list of vulnerabilities and how they were identified and potentially exploited Assists in fixing technical issues	**Risk Analysis:** Detailed analysis of the internal architecture and the potential exposures based on observations Assists in addressing the high-level technical concerns in addition to process changes

Scenario:	Assess Physical Security	
Ethical Hacking	**Pros:** Evaluates the security controls inherently designed to thwart human threats (See Note, Ch. 9: "The Physicality of Social Engineering") Has the potential to accurately reflect various threats Provides the option of comprehensive control and granularity	**Cons:** Requires substantial planning to ensure the potential threat is replicated Increases the liability associated with exploitation of physical controls
Risk Analysis (Typically Employed)	**Pros:** Determines the level of threat and vulnerabilities through evaluation of security controls Assesses the policies and procedures related to physical controls	**Cons:** Does not assess security based on tested weaknesses Level of threats and vulnerabilities based on interpretation of the controls as opposed to testing
Results	**Ethical Hacking:** Provides a list of vulnerabilities that contributed to the failure of controls Offers a detailed understanding of what is obtainable to a person at various points or stages in the test A detailed explanation of what was performed to thwart the security controls	**Risk Analysis:** Detailed analysis of physical controls, potential vulnerabilities, a collection of threats, and likelihood of exploitation Provides a collection of broad recommendations, including policy and process, to accommodate potential weakness