SEC401 | SECURITY ESSENTIALS BOOTCAMP STYLE

401.1
Network Security Essentials

SANS

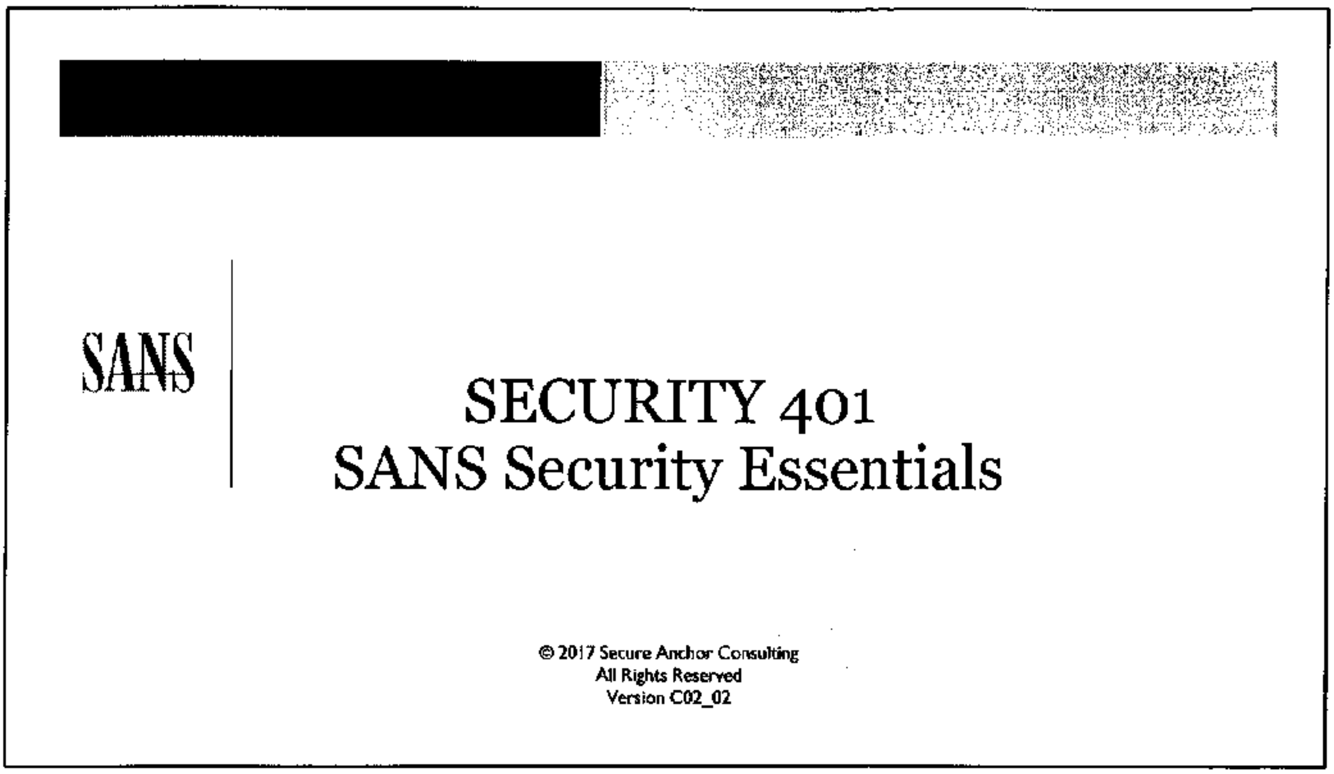
THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | sans.org

SEC401 | SECURITY ESSENTIALS BOOTCAMP STYLE

401.1
Network Security Essentials

SANS

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | sans.org

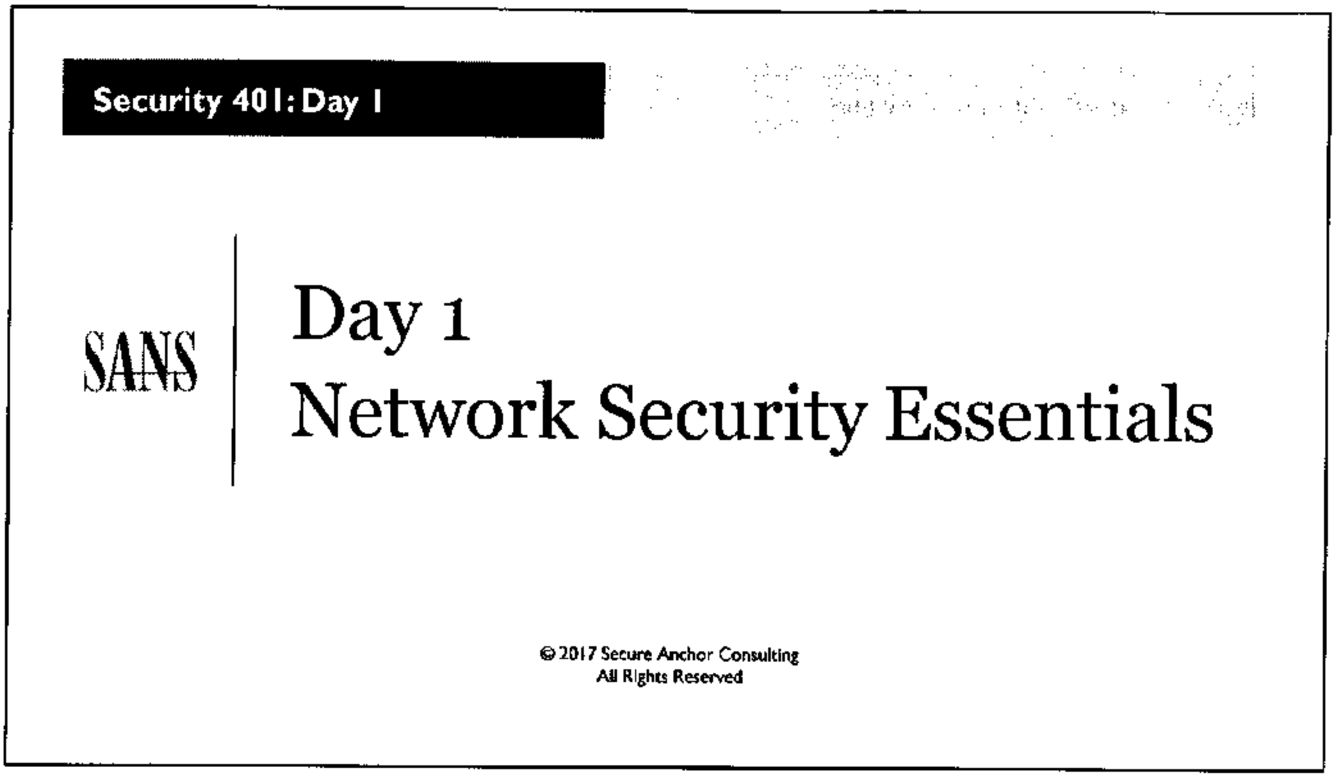


SANS

SECURITY 401
SANS Security Essentials

Version C02_02

This page intentionally left blank.



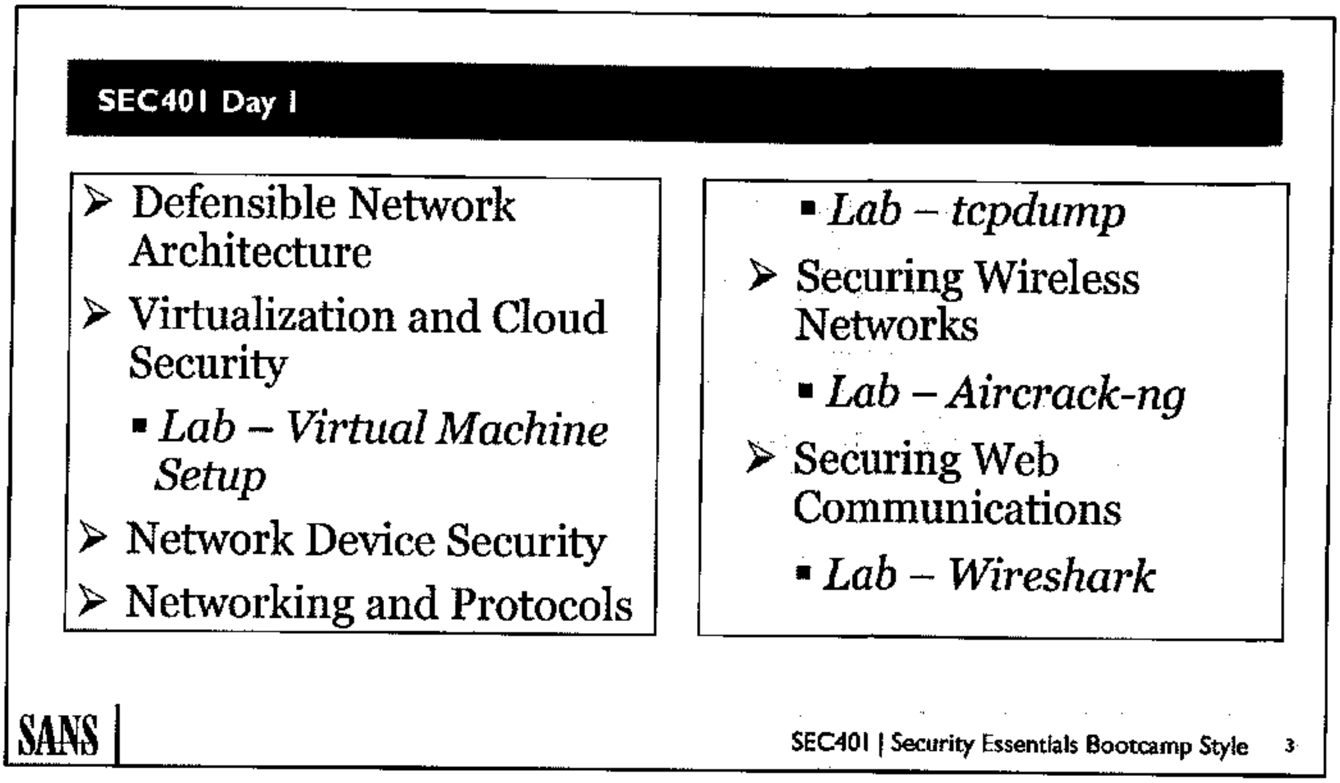
Security 401: Day 1

SANS

Day 1
Network Security Essentials

© 2017 Secure Anchor Consulting
All Rights Reserved

This page intentionally left blank.



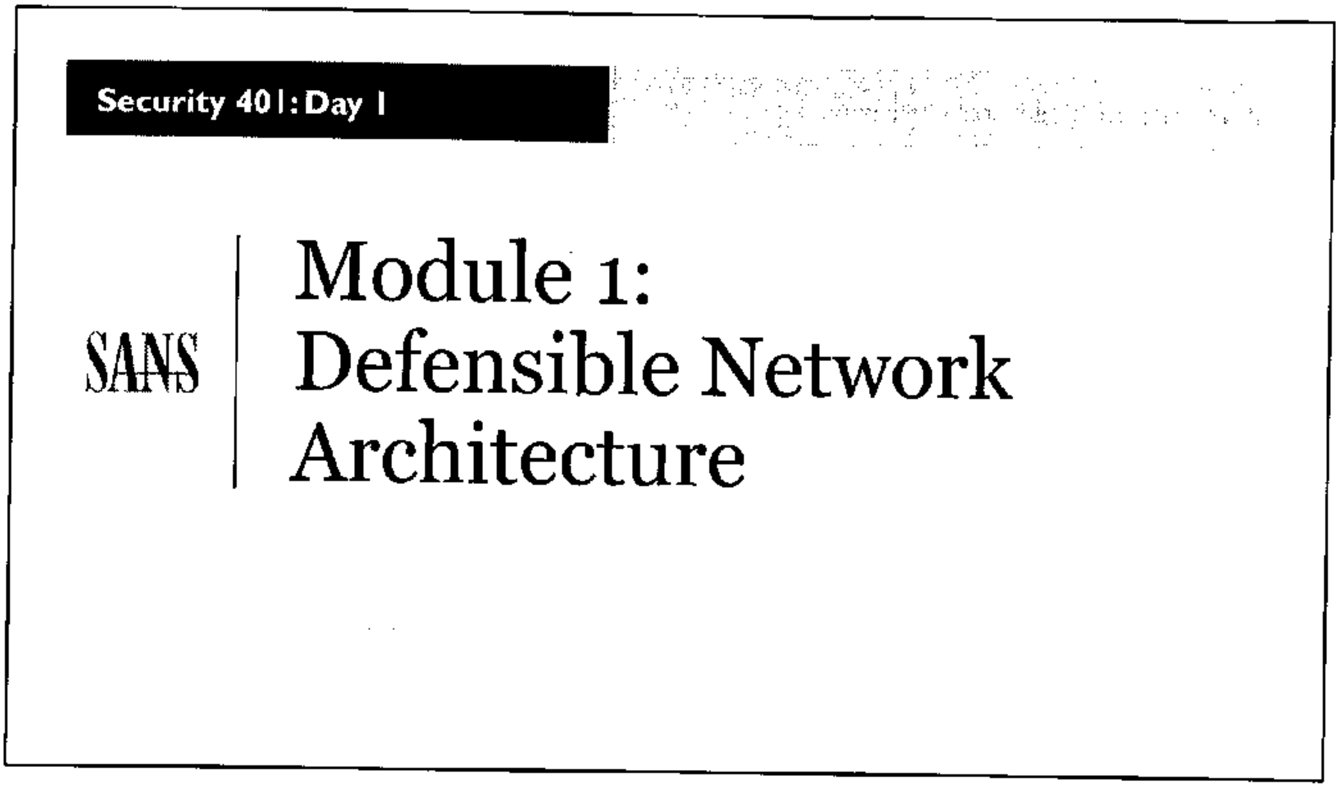
SEC401 Day 1

- ➢ Defensible Network Architecture
- ➢ Virtualization and Cloud Security
  - ▪ *Lab – Virtual Machine Setup*
- ➢ Network Device Security
- ➢ Networking and Protocols
  - ▪ *Lab – tcpdump*
- ➢ Securing Wireless Networks
  - ▪ *Lab – Aircrack-ng*
- ➢ Securing Web Communications
  - ▪ *Lab – Wireshark*

SANS

SEC401 | Security Essentials Bootcamp Style 3

This page intentionally left blank.

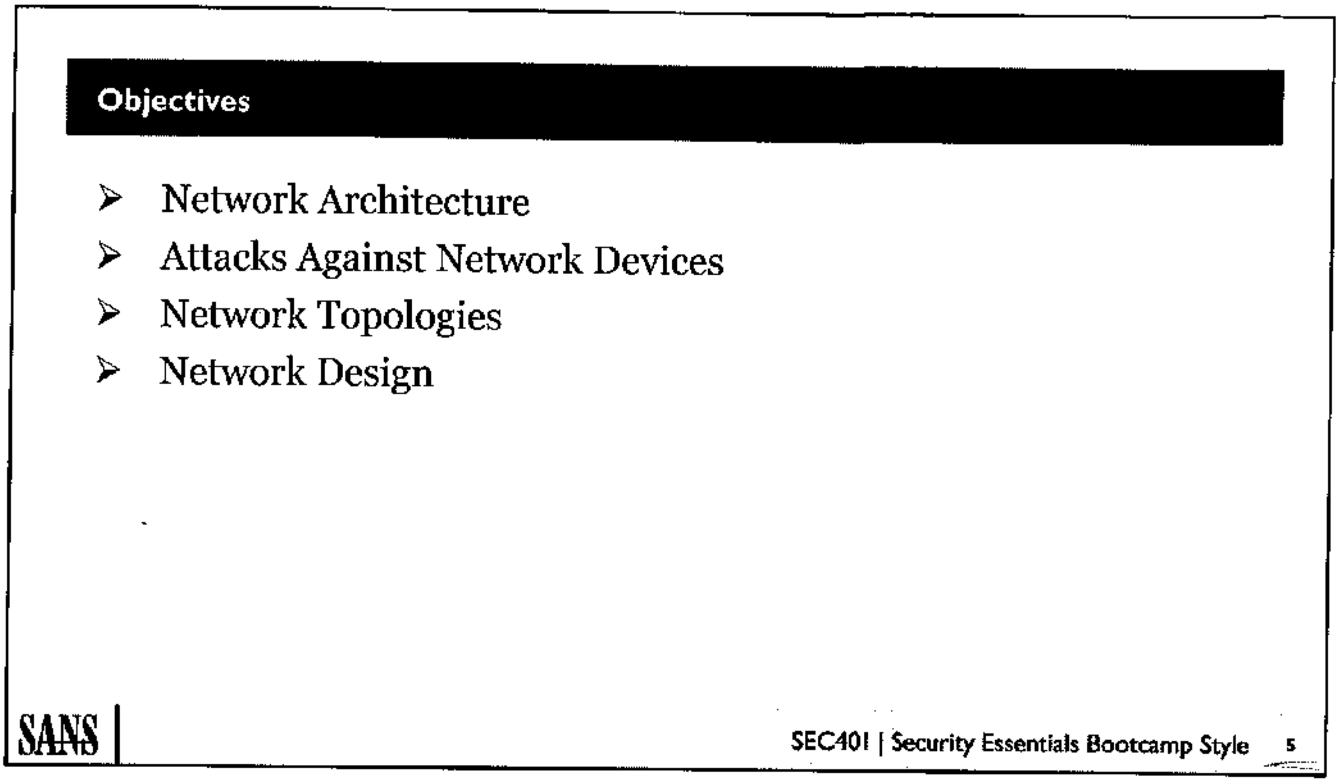


Security 401: Day 1

SANS

Module 1: Defensible Network Architecture

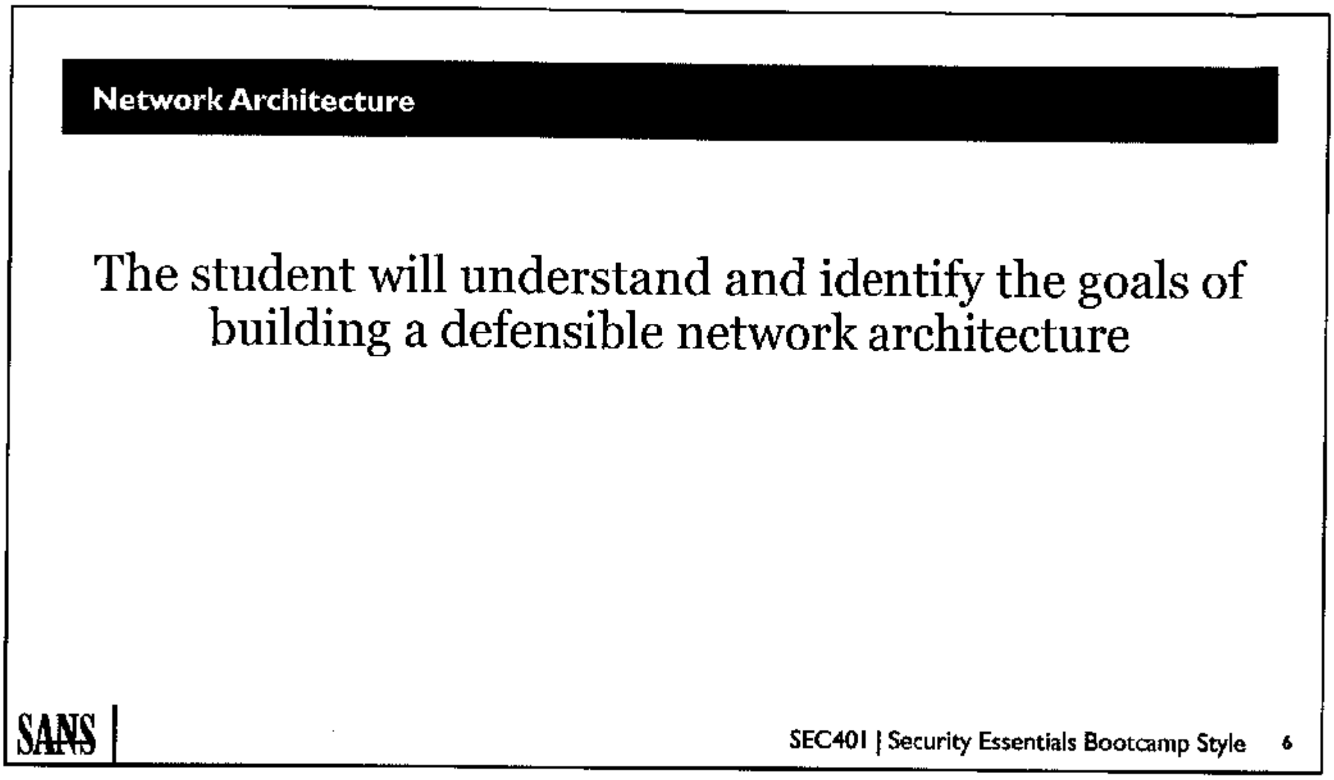
Module 1: Defensible Network Architecture
This page intentionally left blank.



Objectives

- Network Architecture
- Attacks Against Network Devices
- Network Topologies
- Network Design

In order to properly secure and defend a network, you must first have a clear and strong understanding of both the logical and physical components of an architecture. In security (and design), huge mistakes have been made because the security architect did not look at the system as a whole, but rather focused on a particular problem which weakened the overall analysis. It's always important to remember that analyzing the security of something as complex as a network is in itself a complex process.



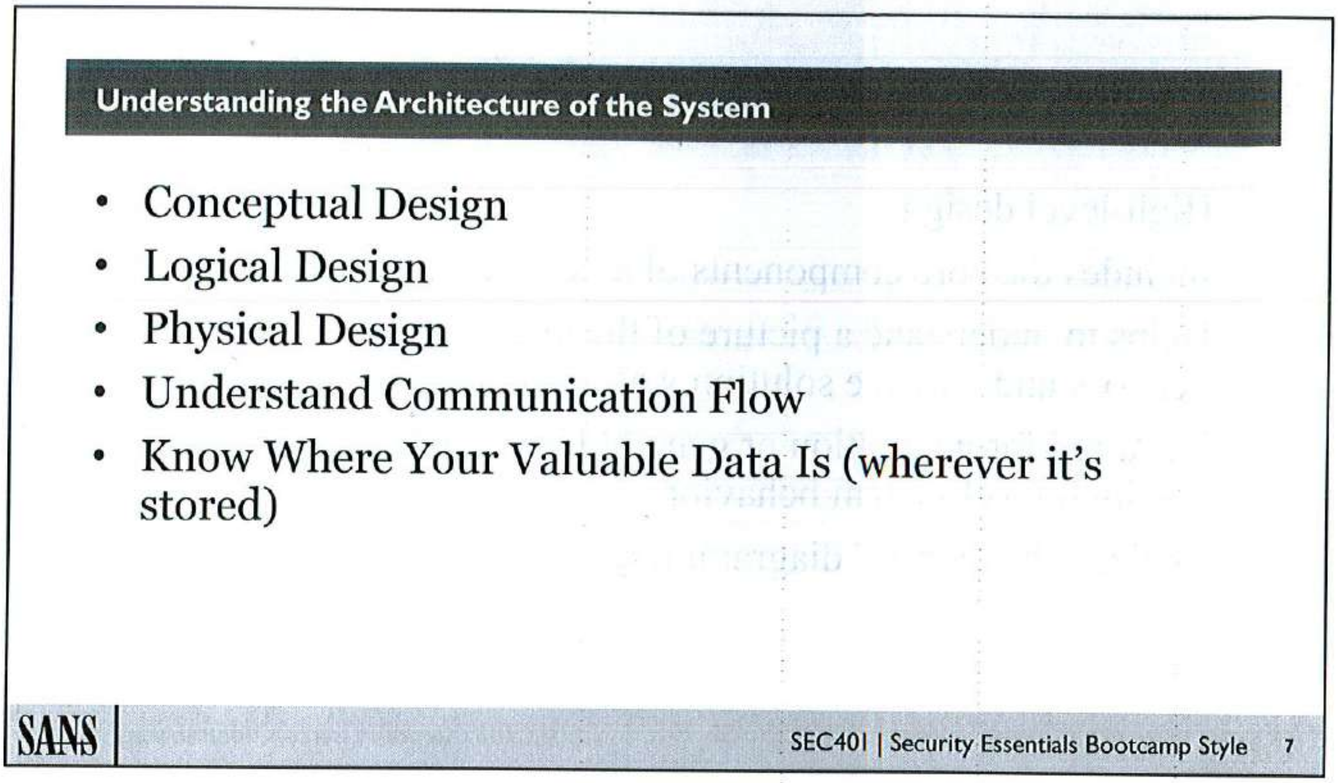
Network Architecture

The student will understand and identify the goals of building a defensible network architecture

SANS

SEC401 | Security Essentials Bootcamp Style 6

This page intentionally left blank.



Understanding the Architecture of the System

- Conceptual Design
- Logical Design
- Physical Design
- Understand Communication Flow
- Know Where Your Valuable Data Is (wherever it's stored)

SANS

SEC401 | Security Essentials Bootcamp Style 7

The Conceptual Design is a high-level design that includes the core components of a network architecture. This abstracted view provides a security professional with an understandable picture of the overall purpose of the network and why the solution was designed the way it was. The conceptual design will include the major technology systems, any external systems that are required for integration or general functionality, data flow, and high-level system behavior. This design method uses the "black box" diagramming approach to simplify complex systems into a single component with general role or purpose.

The Logical Design represents each logical function in the system. It is much more detailed and should include all the major components in the network plus their relationships. Detailed data flows and connections are also mapped out in the logical design. This design is created primarily for developers and security architects. Logical designs include business services, application names, and other relevant information for development purposes but typically do not include server names or addresses.

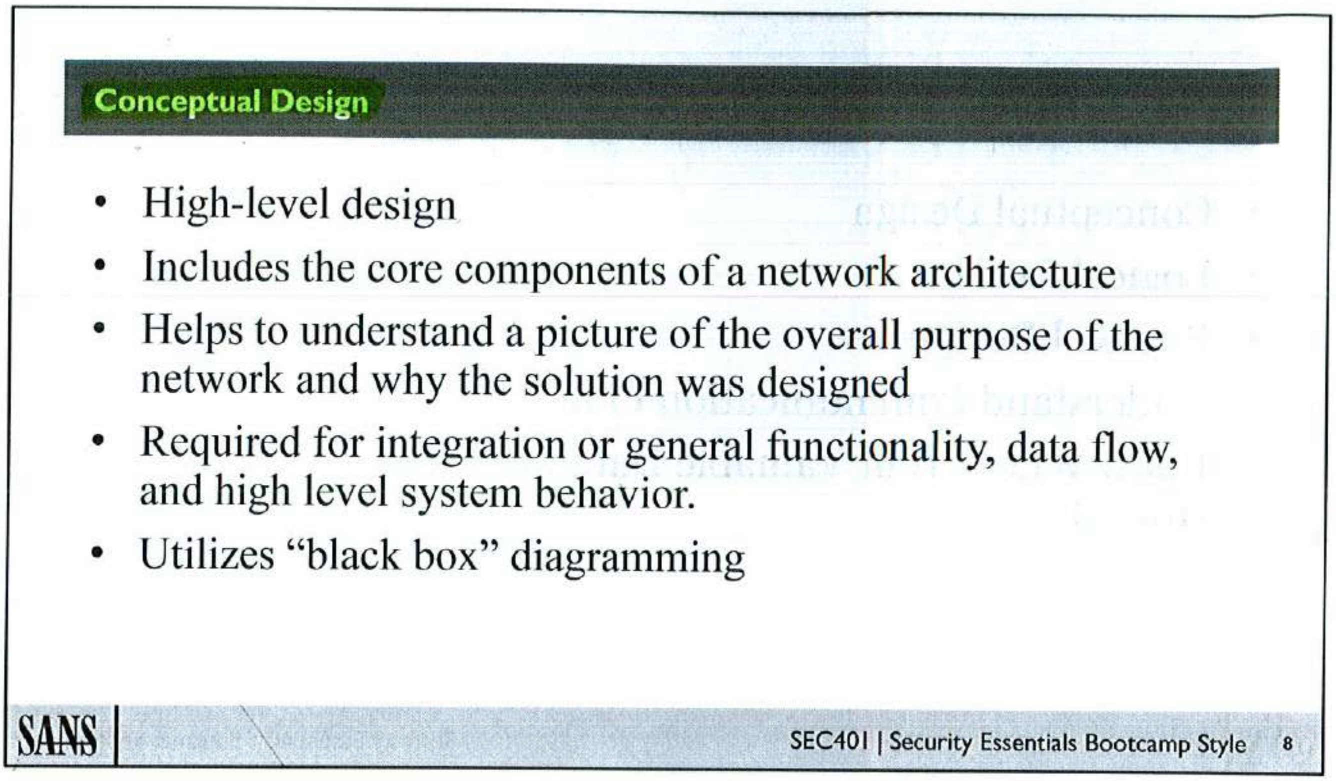
The Physical Design has all the major components and entities identified within specific physical servers and locations. The physical design is usually the last design created before final implementation and may be used as a resource by the final implementation team. This design level has all known details such as operating systems, version numbers, and even relevant patches. Any physical constraints or limitations are also identified within the server components, data flows, or connections.

Understanding the Communication Flow begins with the logical architecture. Every communication flow, whether for data exchange or control messages, should be diagrammed regardless of its purpose.

Knowing Where Your Valuable Data Is also begins with the logical architecture. And just as you need to know every communication flow, you also need to know where every last file of your valuable data resides.

Reference

1. Securing Systems: Applied Security Architecture and Threat Models. By: Brook S.E. Schoenfield. Auerbach Publications © 2015



Conceptual Design

- High-level design
- Includes the core components of a network architecture
- Helps to understand a picture of the overall purpose of the network and why the solution was designed
- Required for integration or general functionality, data flow, and high level system behavior.
- Utilizes “black box” diagramming

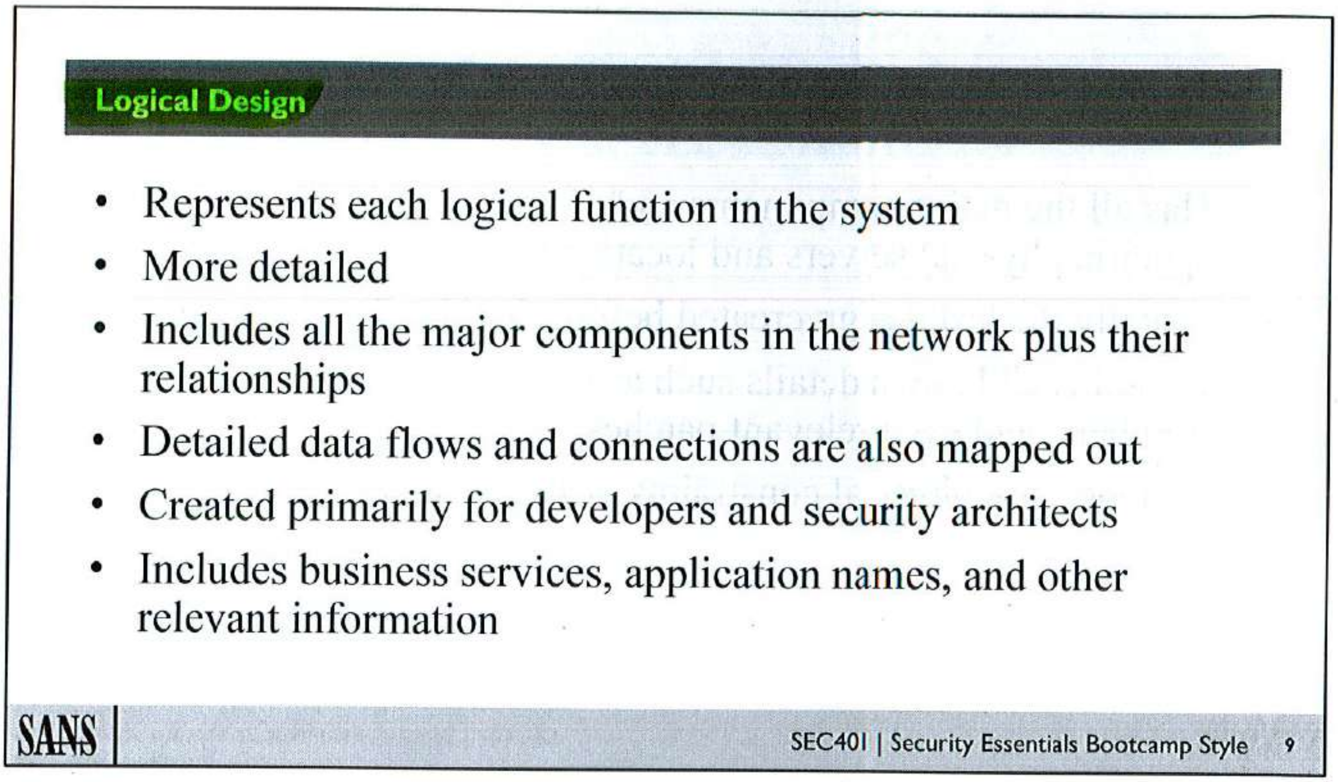
SANS

SEC401 | Security Essentials Bootcamp Style 8

The key purposes of a high-level design with a security architecture is to provide an overview of an entire system with the intent of identifying the main components that will be deployed. This type of overview is critically important, especially with something as complex as a modern network. Fielding the core components of a network requires support and contributions from many distinct professional disciplines, each with critical skill sets. This is where strong conceptual design will be compatible with any sub-designs and with the big picture.

The conceptual design should briefly describe all platforms, systems, products, services and processes that it depends on and include any important changes that may need to be made to them. A conceptual design can even incorporate considerations for all significant commercial, legal, environmental, security, safety and technical risks, issues, and assumptions. The conceptual design will also help in identifying every type of end-user which will allow the architects to give due consideration to the customer experience they are aiming for.

Utilizing black box diagramming is a method of depicting a device, system or object only in terms of its inputs and outputs without requiring any knowledge of its internal workings. This simplifies the characteristics of the network during the design phase.



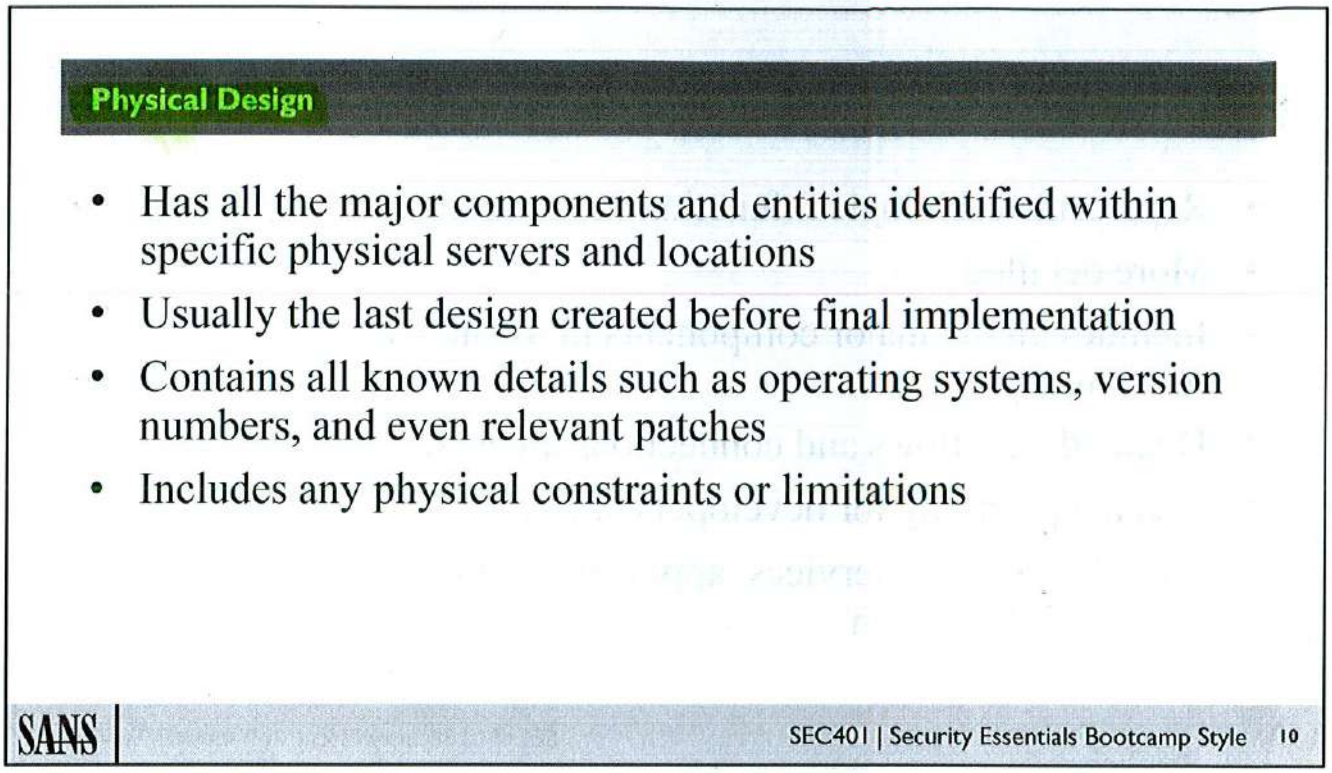
Logical Design

- Represents each logical function in the system
- More detailed
- Includes all the major components in the network plus their relationships
- Detailed data flows and connections are also mapped out
- Created primarily for developers and security architects
- Includes business services, application names, and other relevant information

A logical network design, from the architect and user point of view, depicts how data passes between devices on the network. It is far more detailed than the conceptual design and begins to break out how the network will actually operate in the real world. The logical design differs from a physical design in that it doesn't reflect the connections between the actual physical cables, the computers, and other network systems. An example of this point would be the physical design would specify the shape of the network's topology, while a logical design wouldn't.

Typically a logical network design is shown as a detailed network diagram that uses icons and other depictions to show workstations, servers, printers, scanners, routers, switches, hubs, firewalls and other network devices. This design may show how the cables connect to make the network and will usually specify the type of workstations based on operating system.

Including business services, application names and other relevant functional information is key in the logical design as well. In the end, the network is being designed and built to serve a greater function which is to enable a business or organization to function more efficiently and effectively. Looking at what is running on the network and why will provide unique insights on how to improve security that may be different from the same network architecture running completely different business services and applications.



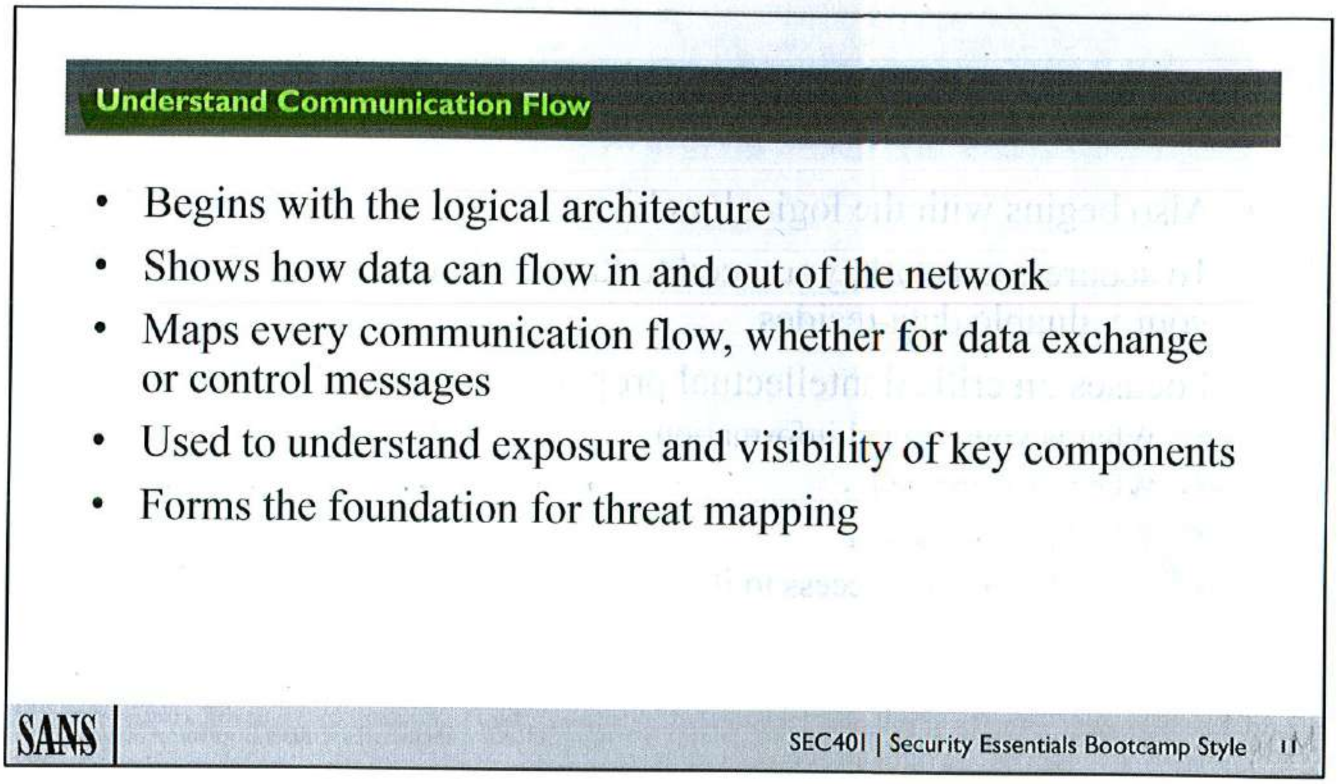
Physical Design

- Has all the major components and entities identified within specific physical servers and locations
- Usually the last design created before final implementation
- Contains all known details such as operating systems, version numbers, and even relevant patches
- Includes any physical constraints or limitations

A physical design has all major components and entities identified within specific physical servers and locations or specific software services, objects, or solutions. This is the design level that represents how the network and all its components will actually behave while still being "on paper". The physical design is the last one created before the network design is implemented. This design usually precludes or may be included and extended by the final implementation team into an implementation design.

The physical design shows all known details such as operating systems, version numbers, and even patches that are relevant. This is a particularly important to get ahead of since software updates perform a myriad of tasks. Updates are regularly available for operating systems, network devices, and individual software programs. Performing these updates will deliver a multitude of revisions to your computer. This often includes adding any new features that have been developed, removing outdated features which may have vulnerabilities, updating drivers which may also have exploits, delivering bug fixes, and most importantly, fixing any security holes that have been discovered.

Any physical constraints or limitations should also be identified within the server components, data flows, or connections. Physical constraints can affect the cost of deployment and maintenance as well as create (or eliminate) some security challenges depending on how the physical network is deployed.

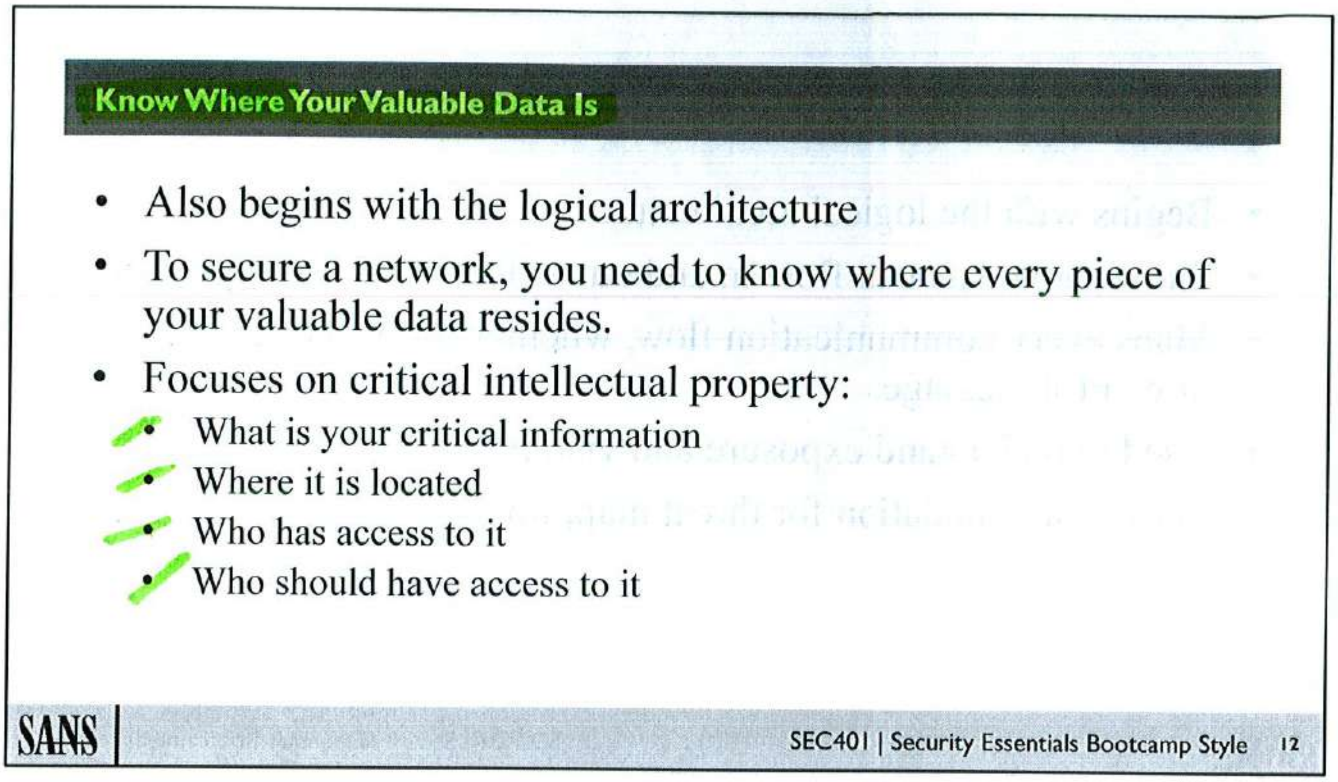


Understand Communication Flow

- Begins with the logical architecture
- Shows how data can flow in and out of the network
- Maps every communication flow, whether for data exchange or control messages
- Used to understand exposure and visibility of key components
- Forms the foundation for threat mapping

Just as you need to understand what components are on your network and how they all behave and interrelate, it is just as critically important to have a complete and comprehensive understanding of how communications flow over the network. Without this understanding, sensitive data could be transmitted and stored incorrectly or worse leave the network entirely. Using the logical architecture diagram you can get a clear understanding of how and where data flows in and out of your network and determine the appropriate security posture. This communication flow map helps you understand the exposure and visibility of key networks components to attackers or malicious insiders. This understanding forms the foundations for creating a threat map and implementing appropriate security controls and countermeasures.

By understanding what information can flow in and out of different segments of a network is important to not only understanding the point of compromise but the amount of damage that can be caused by an adversary. In many incidents, one of the reasons that the adversary was able to cause so much damage is because the organization did not understand how the data flowed within the network. Often there are connections that are setup that people are not aware. Always remember that if the offense knows more than the defense, the defense is going to lose. Communication flow helps the defense get a better understanding of exposures and proactively fix them.



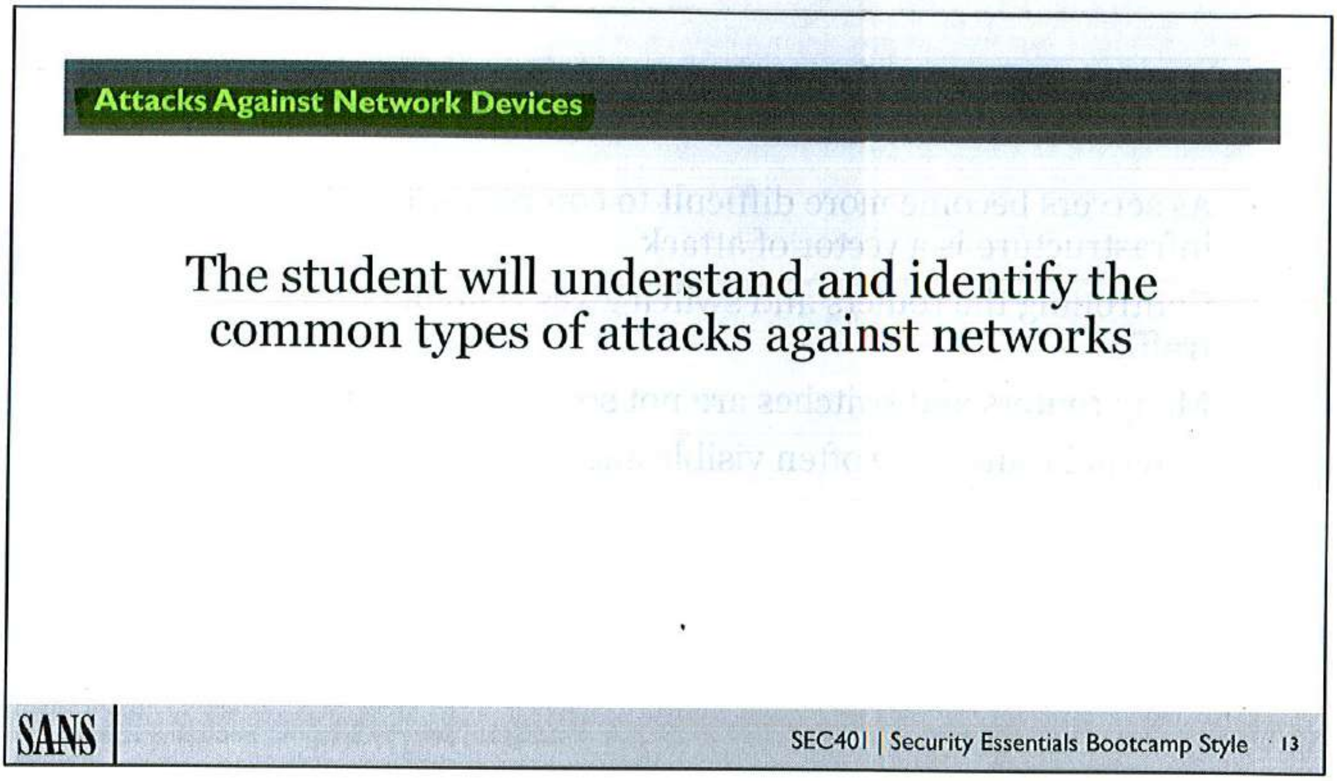
Know Where Your Valuable Data Is

- Also begins with the logical architecture
- To secure a network, you need to know where every piece of your valuable data resides.
- Focuses on critical intellectual property:
  - What is your critical information
  - Where it is located
  - Who has access to it
  - Who should have access to it

SANS | SEC401 | Security Essentials Bootcamp Style 12

In any organization whether large or small all data is not created equal. Some data is routine and incidental while other information can be very sensitive, the loss of which could cause irreparable harm to an organization. Another key part of security architecture is the understanding of where your data will reside once it's passed over your network to its destination. This understanding again begins with the logical architecture. The best approach to take is to look at intellectual property (IP). Identify what is the critical IP in your organization and how and where is it stored on your network? What security measures are in place to make sure confidentiality, integrity and availability are sound. Where are these storage drives physically located? Are they secure from both electronic and physical attack? Additionally, you will need to know who has access to this information and just as important who "should" have access to it?

In many organizations, critical data is unaccounted for and stored in many locations across the network with various levels of security. Remember, an adversary only has to find one location to compromise critical data. Remember that you are only as strong as your weakest link. Inadvertent data storage on vulnerable systems is a weak link in many environments. Therefore a key component of an effective architecture is to de-scope the problem. Reduce the number of areas in which critical data is stored and implement effective defensive measures against those data repositories.



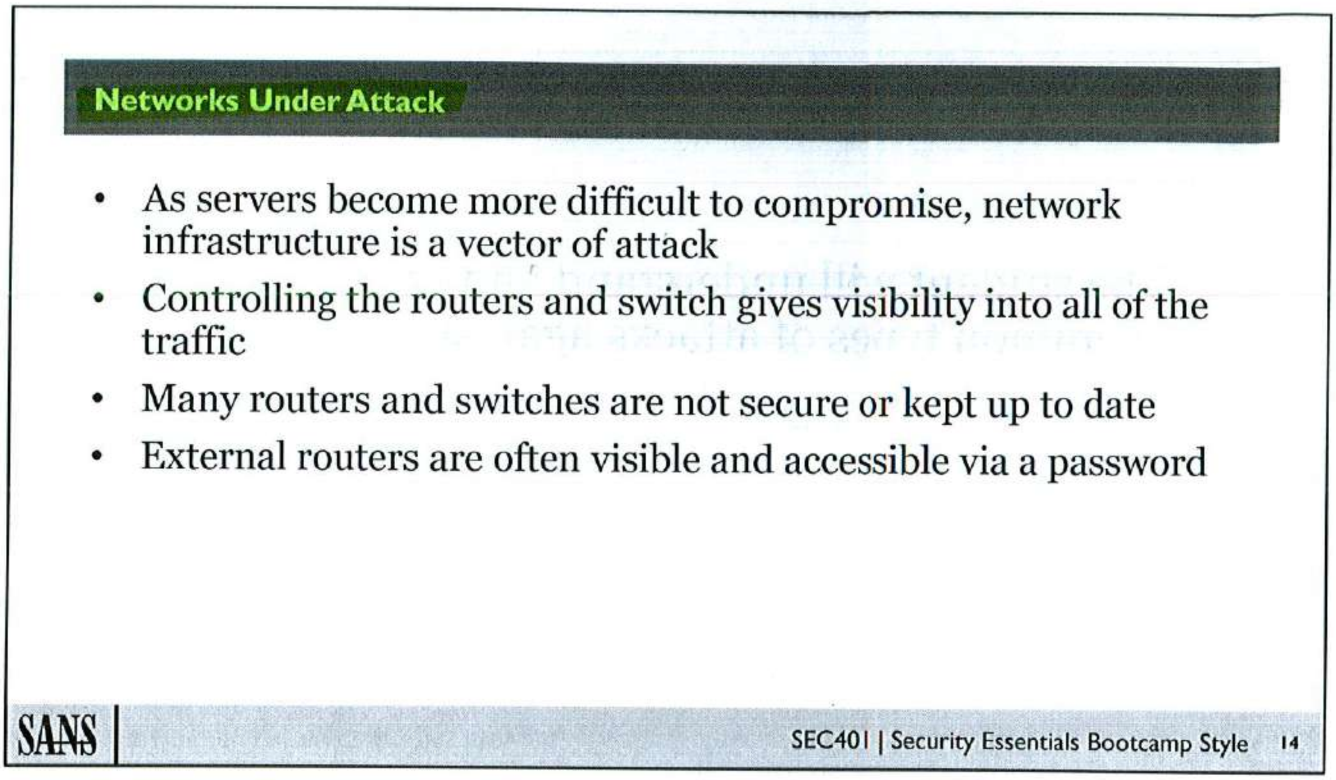
Attacks Against Network Devices

The student will understand and identify the common types of attacks against networks

SANS

SEC401 | Security Essentials Bootcamp Style 13

This page intentionally left blank.



Networks Under Attack

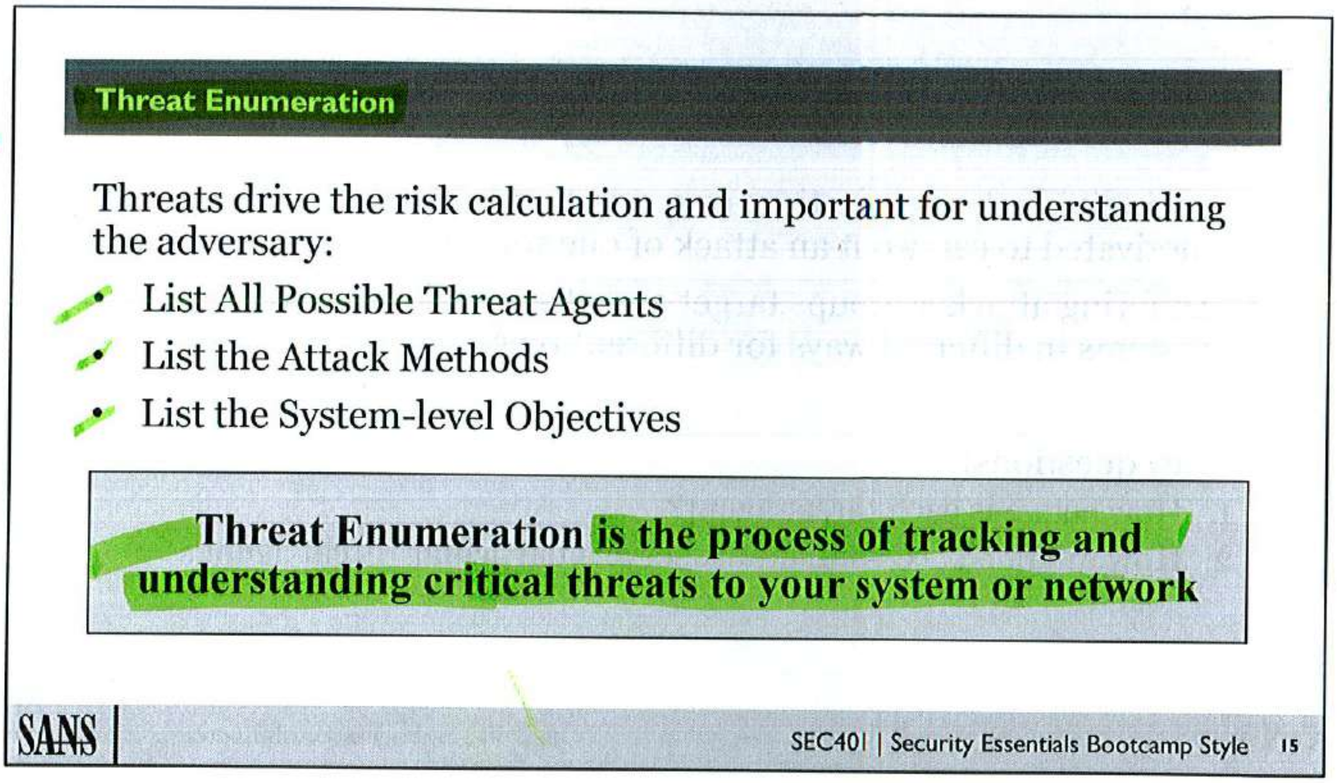
- As servers become more difficult to compromise, network infrastructure is a vector of attack
- Controlling the routers and switch gives visibility into all of the traffic
- Many routers and switches are not secure or kept up to date
- External routers are often visible and accessible via a password

When most organizations think about security, the focus is often on hardening of servers. Patching, updating, securing and proper configuration management are all key areas of focus for most security teams. While properly securing servers is important, many entities forget about the network infrastructure. If the network infrastructure is compromised, security measures can be bypassed, authentication can be sniffed and critical data can be captured. As important a network is to an organization, it is amazing how many times it is ignored from a security perspective.

It is not uncommon to find routers and switches that have not been patched and running old versions of the IOS. While the infrastructure of a network is important, the focus is often solely on availability. If the network is up and running, everything must be fine. The failed logic in this approach is that the functionality might be fine, but if the security is weak and network devices are compromised, the game is over very quickly.

Routers connect different networks together. Not only do they connect an organization to the Internet but they also connect internal networks together. If a router is compromised, it provides an adversary visibility into all of the traffic going over the network. Since external routers are often directly accessible from the Internet with a public IP, they become a prime target of attack. If the external router is compromised, an adversary has an ultimate sniffer where they can monitor all traffic going in and out of a network.

Switches connect computers together to form network or virtual networks known as VLANs. Since VLAN's are becoming a more common way of isolating and segmenting networks, if a sniffer is compromised, an adversary can perform VLAN hopping, potentially bypassing security controls that have been put in place.



Threat Enumeration

Threats drive the risk calculation and important for understanding the adversary:

- List All Possible Threat Agents
- List the Attack Methods
- List the System-level Objectives

Threat Enumeration is the process of tracking and understanding critical threats to your system or network

SANS | SEC401 | Security Essentials Bootcamp Style 15

Threat Enumeration, at its most basic, is the process of understanding threats to your system or network. Once an architecture is represented in a way that supports security analysis the next step is the enumeration and examination of all relevant threats.

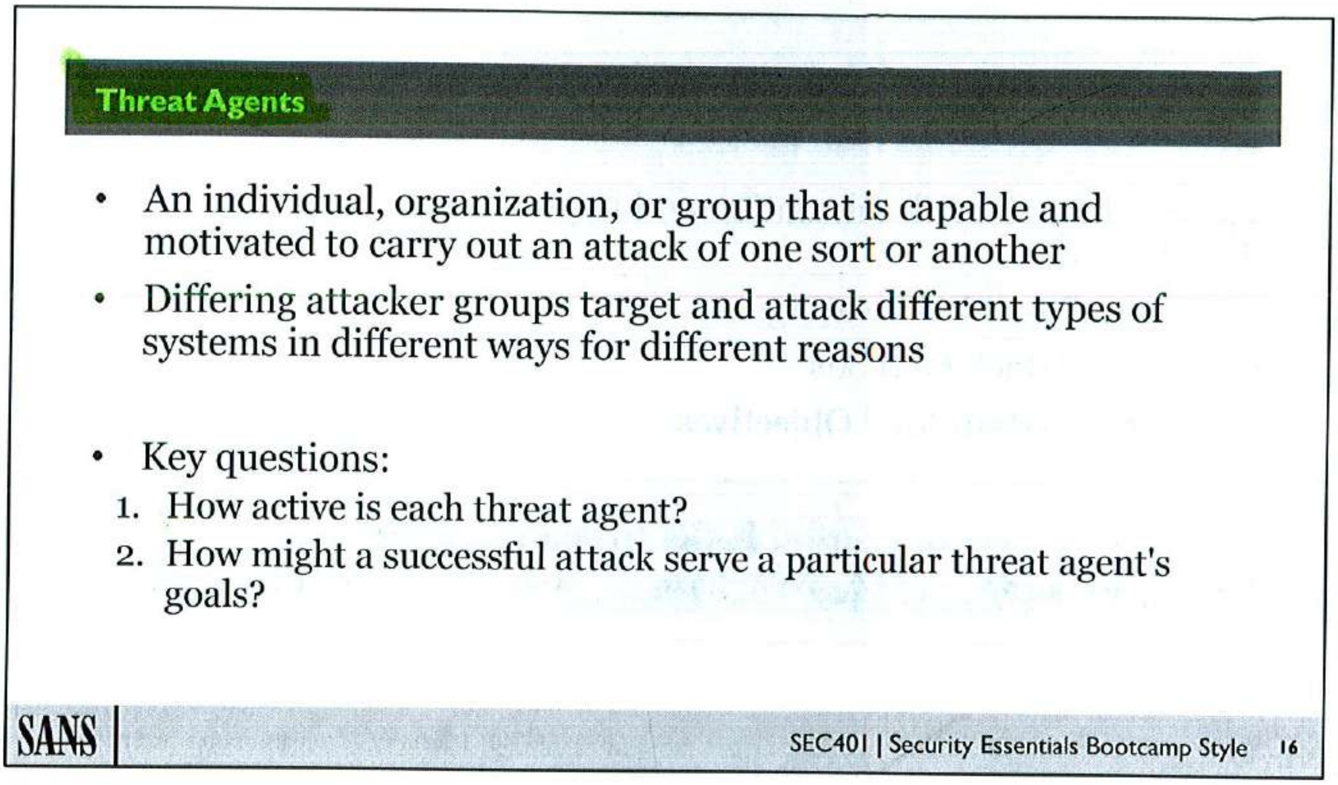
List All Possible Threat Agents - The overall question here is "Who are the threat agents and who will be most interested in attacking the network?" As we all know any system that is open to Internet traffic will be attacked continuously and the majority of these attacks are undirected or untargeted. Cyber criminals, in general, have a low-risk tolerance; so, many times these sweeps use unsophisticated, automated scripts attempting well-known attacks against possibly unpatched systems. Aside from basic threat agents, there are those interested in stealing financial information or in various forms of cyber fraud such as a denial of service attack (DoS) as a way to blackmail organizations.

Attack Methods - A cyber criminal's overall goal is to maximize profit with minimal effort; in other words, to make as much money as possible. Because of this fact a cyber criminal who intends on attacking an organization's system or network will attempt to use as much pre-existing and proven technology as possible. There is a huge black market for attack tools, tools similar to the attack and penetration tool, Metasploit.

System-level Objectives - Attackers have several system-level objectives. If their goal is to execute unintended commands or access data without proper authorization they will likely attempt a SQL or LDAP Injection. If they want to execute scripts in a browser, hijack a user session, alter or deface a website or redirect users to another site they will use Cross-Site Scripting (XSS). Cyber criminals will use Cross-Site Request Forgery (CSRF) to generate what appears to be legitimate requests from the victim machine or force a victim's browser to send a forged HTTP request. Other system level objectives involve redirects and forwards where the attacker will forward users to other websites, redirect victims to malware sites or use forwards to access unauthorized pages.

References

1. Designing Security Architecture Solutions. By: Jay Ramachandran. John Wiley & Sons © 2002
2. Securing Systems: Applied Security Architecture and Threat Models. By: Brook S.E. Schoenfield. Auerbach Publications © 2015



Threat Agents

- An individual, organization, or group that is capable and motivated to carry out an attack of one sort or another
- Differing attacker groups target and attack different types of systems in different ways for different reasons

- Key questions:

1. How active is each threat agent?
2. How might a successful attack serve a particular threat agent's goals?

SANS SEC401 | Security Essentials Bootcamp Style 16

Differing attacker groups target and attack different types of systems in different ways for different reasons. Each unique type of attacker is called a "threat agent" which is simply an individual, organization, or group that is capable and motivated to carry out an attack of one sort or another. Threat agents are not created equal. They have different goals, different methods, different capabilities and access, and they have different risk tolerances that dictate the different lengths they will go to be successful. With this insight into these differences, you can see why it is important to understand who your attackers are and why they might attack you. It is also important to remember that human threat agents always have these three attributes: they are intelligent, adaptive and creative.

It is important to note that a threat can originate from something other than a human being. Natural disasters, such as earthquakes and tornadoes, are most certainly threats to computer systems. However, for the purpose of this discussion, we are focusing on human attackers.

When evaluating the probability of attack, it is important to know if there are large numbers or very few of each sort of attackers. Some relevant questions to ask yourself are:

1. How active is each threat agent?
2. How might a successful attack serve a particular threat agent's goals?

There are 3 broad categories of threat agents. They are:

1. Cyber Criminals and Organized Crime: They are pulling in billions—sometimes tens of billions—of dollars each year. What do cyber criminals want? The simple answer is money. There is money to be made in cybercrime.

2. Cyber Espionage: These threat agents have a distinctly different goal, which has nothing to do with money and everything to do with information and disruption. Advanced persistent threats (APTs) are well-named because these attack efforts can be multi-year, multidimensional, and are often highly targeted. Many cyber espionage threat agents have significant numbers of people with which to

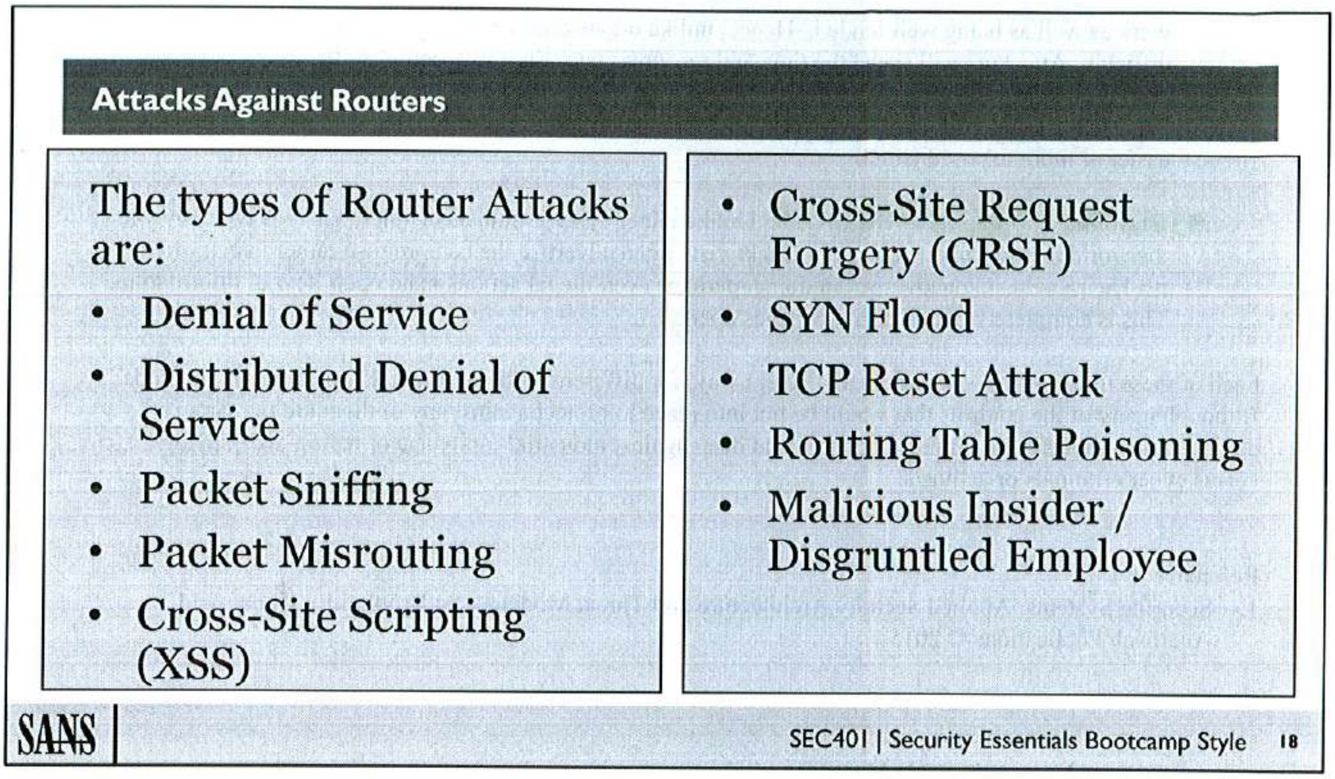
work as well as being well funded. Hence, unlike organized cyber criminals, no challenge is too difficult. Attackers will spend the time and resources necessary to accomplish the job. Like cyber criminals, APT is a risk-averse strategy, attempting to hide the intrusion and any compromise. Persistence is an attribute. This is very unlike the pattern of cyber criminals, who prefer to find an easier or more exposed target.

3. The Computer Activist or "Hacktivist": Unlike either cyber criminals or spies, activists typically want the world to know about a breach. Hacktivists often advertise the compromise rather than try to hide it. Their goal is to uncover wrongdoing, perhaps even illegal actions in an open flow of information. This is completely opposite to how spies operate.

Each of these threat agents operates in a different way, for different motivations, and with different methods. Although many of the controls that would be put into place to protect against any of them are the same, a defense-in-depth has to be far more rigorous and deep against industrial espionage or nation-state spying versus cyber criminals or activists.

Reference

1. Securing Systems: Applied Security Architecture and Threat Models. By: Brook S.E. Schoenfield. Auerbach Publications © 2015



Attacks Against Routers

The types of Router Attacks are:

- Denial of Service
- Distributed Denial of Service
- Packet Sniffing
- Packet Misrouting
- Cross-Site Scripting (XSS)
- Cross-Site Request Forgery (CRSF)
- SYN Flood
- TCP Reset Attack
- Routing Table Poisoning
- Malicious Insider / Disgruntled Employee

SANS | SEC401 | Security Essentials Bootcamp Style 18

The Denial of Service attack is achieved by flooding the router with requests thereby affecting its availability. Sending a large amount of ICMP packets from multiple sources makes the router unable to process traffic effectively and therefore makes it unable to provide services.

A Distributed Denial of Service attack uses a conscripted army of computers referred to as "zombie" hosts which are infected with a piece of software designed to send packets to routers all at the same time. DDoS attacks can use hundreds and potentially thousands of computers at once. When the attacker triggers the script from the infected computers, they target routers and overpower their resources.

Packet sniffing is a method of capturing data using a malicious program that monitors to network traffic. In the past, packet sniffing required physical access to the wired Ethernet environment, but with the prevalence of wireless routers today, packet sniffing has become a more common threat since unauthorized access doesn't require direct access the physical network.

Packet Misrouting is a type of where the router is injected with malicious code that causes the router to simply mistreat the packets. In short, the router operations become disrupted and the router becomes unable to carry out its own routing processes. This type of attack is very difficult to find and fix because the router disruption creates loops, denial of service conditions and other congestion on the network.

Cross-Site Scripting is a web-based attack used to reveal the physical location of a user, often without their knowledge. The attacker gets the user's MAC address through the router and then uses a location service, such as Google Location Service, to determine where the user is. This type of attack only works if the user logs into their router and then visits a website with an XSS exploit.

Cross-Site Request Forgery is another web-based attack for gaining control of web applications such as web-based e-mail. This attack is also effective on routers. The attack often comes in the form of an e-mail message with a link to an external website. Once the link is clicked a script takes over the router and provides unauthorized access to the attacker. Most victims don't realize they are the victim of this type of attack. Rebooting the router will remove the script and nullify the attack.

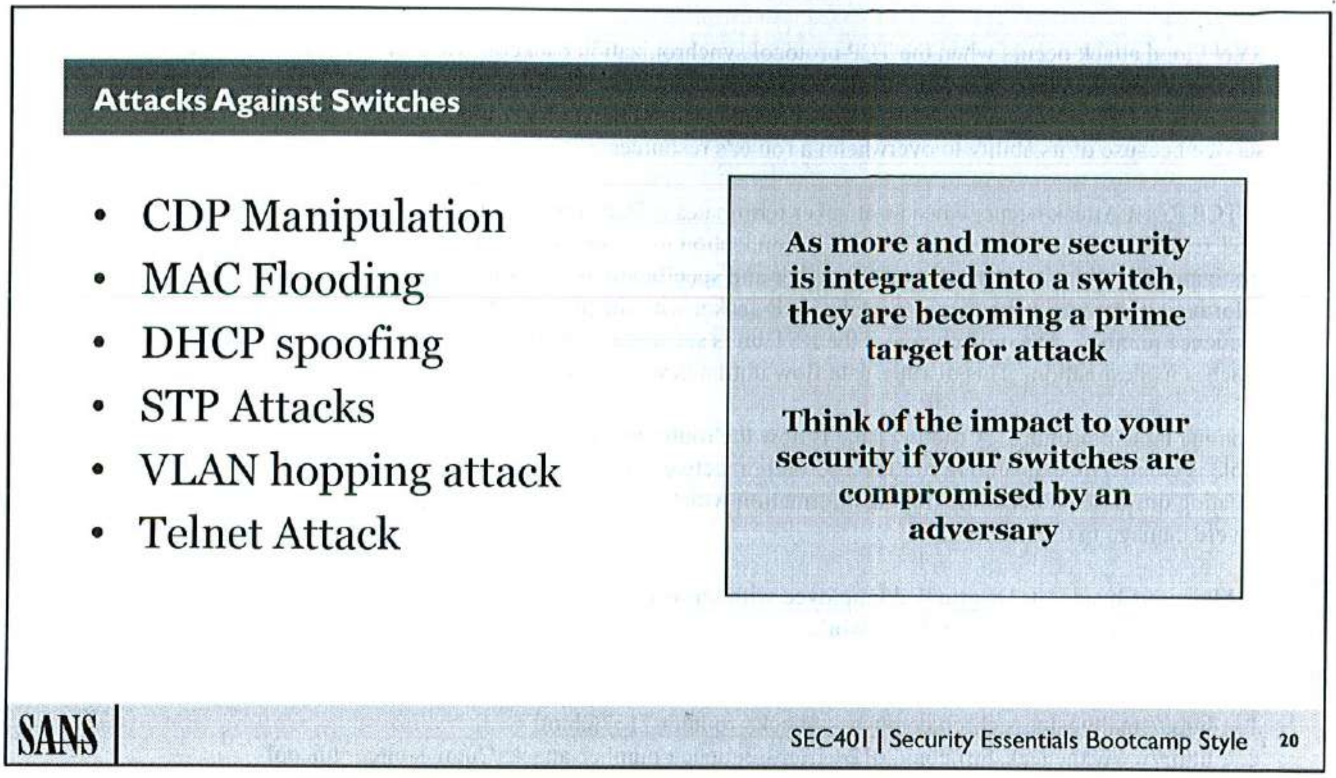
SYN Flood attack occurs when the TCP protocol synchronization packet is used for malicious reasons. The attacker will send a large number of TCP/SYN packets to the destination server. Since the server will be unable to establish a connection the address becomes unreachable. SYN Flooding is a form of denial of service because of its ability to overwhelm a router's resources.

A TCP Reset Attack occurs when an attacker terminates a TCP connection with a spoofed packet by using a RST (reset) bit. The attacker sniffs the TCP connection to get the source IP address, source port number, destination IP address, destination port number and specifically the ongoing sequence number. With this information, the attacker can create a fake TCP packet with the proper source IP, port, destination and sequence number. The only change is the RST bit is set which terminates the connection as soon as the packet reaches its destination. This disrupts data flow until a new session is established.

Routing table poisoning. A routing table is how the router moves the packets in the network. The routing table is created by exchanging routing information between routers. Routing table poisoning is done by creating unwanted edits to the routing information which are broadcasted by routers. This attack can cause severe damage in the network.

A Malicious Insider or Disgruntled Employee with knowledge of the network can access routers without authorization and compromise the network.

References
1. http://smallbusiness.chron.com/types-attacks-routers-71576.html
2. http://www.thegeekstuff.com/2012/01/tcp-sequence-number-attacks/?utm_source=tuicool



Attacks Against Switches

- CDP Manipulation
- MAC Flooding
- DHCP spoofing
- STP Attacks
- VLAN hopping attack
- Telnet Attack

As more and more security is integrated into a switch, they are becoming a prime target for attack

Think of the impact to your security if your switches are compromised by an adversary

SANS

SEC401 | Security Essentials Bootcamp Style 20

CDP Manipulation: The CDP packets are enabled by default on Cisco switches and they are transmitted in the clear. This allows an attacker to analyze the packets and gain information about the network device. The attacker can use this to exploit known vulnerabilities against the device. The solution is to disable CDP on non-management interfaces.

MAC Flooding: An attacker floods the content addressable memory table with MAC addresses. This flood overwhelms the switch storage capacity. The switch defaults down to operating as a hub which gives the attacker the ability to sniff the traffic along that segment of the network.

DHCP spoofing. This is another form of "man-in-the-middle" attack where the attacker listens for DHCP requests then answers them with the attackers IP address as the default gateway.

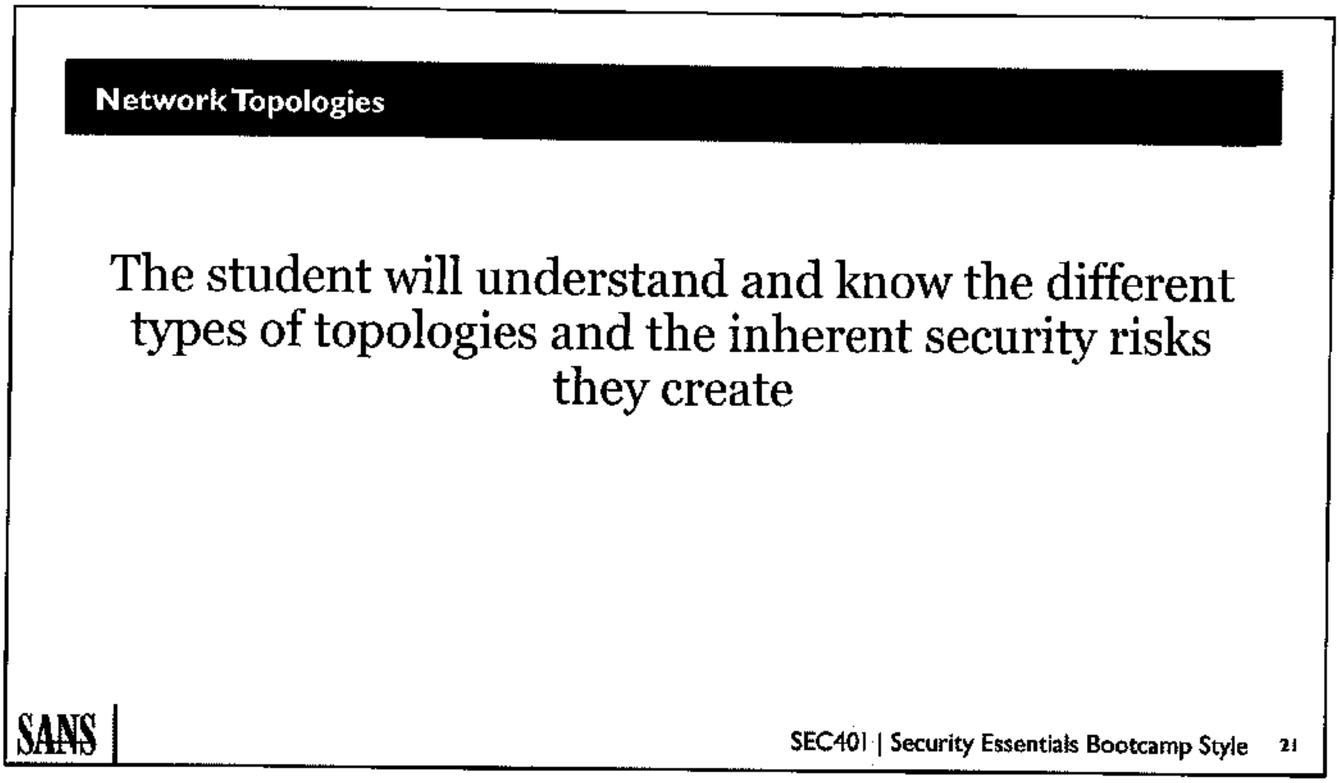
STP Attacks: The spanning tree protocol allows a switch to function by altering the ports so that it can block or forward various conditions in accordance with the kinds of segments they are linked to. There are different types of attacks that directly target the spanning tree protocol; one type of attack involves sending of RAW configuration bridge protocol data unit (BPDU) packets which is a data message transmitted across a local area network to detect loops in network topologies.

VLAN hopping attack. With this, an attacker can create and send positioned frames from one VLAN, using spoofed 802.1Q tags, in such a way that the packet ends up on a completely different VLAN without passing through the router.

Telnet Attack is actually misnamed, more specifically it is a distributed SYN attack. Windows operating systems have an accessible telnet executable which can be used to set up a TCP session. This technique can create the conditions of a denial of service attack and is popular with criminal botnet operators since the executable is already part of the operating system.

References

1. https://howdoesinternetwork.com/2011/switch-security-attacks
2. http://www.cisco.com/c/en/us/support/docs/ip/access-lists/13608-21.html

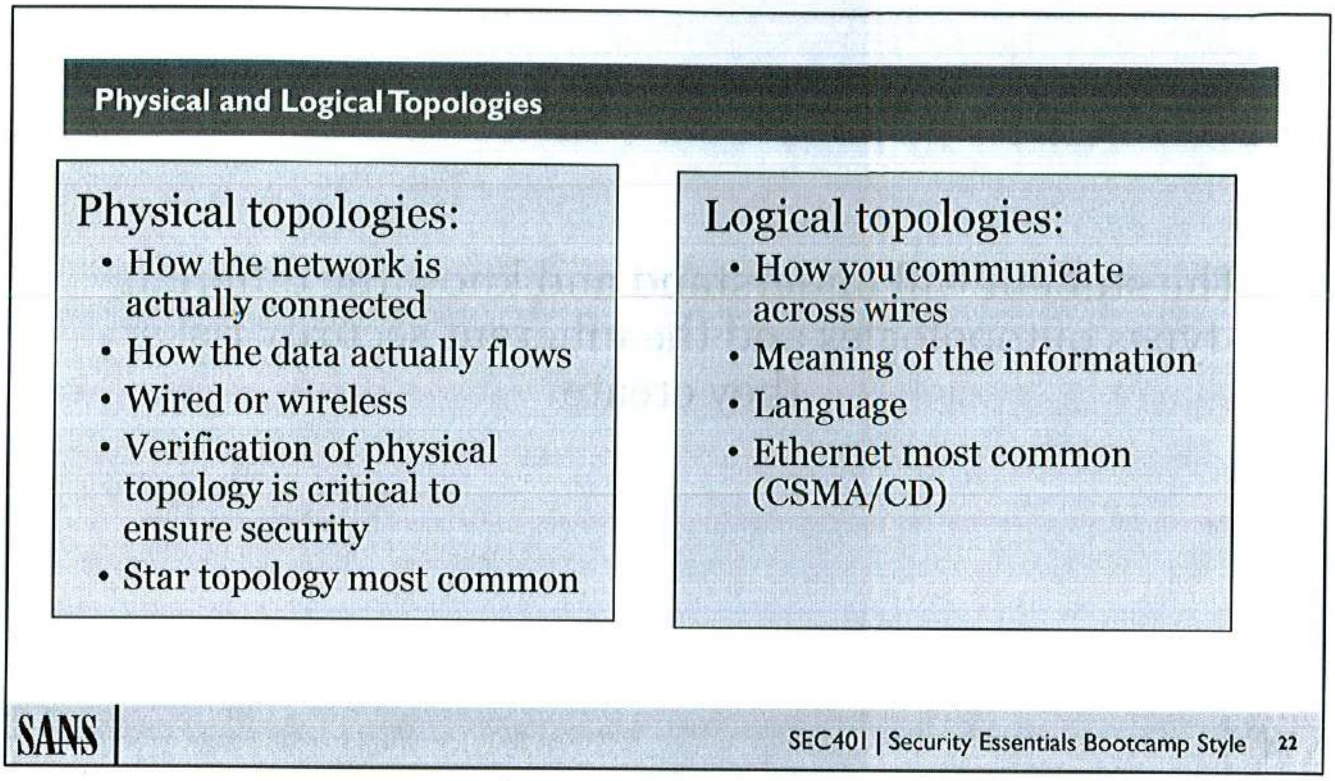


Network Topologies

The student will understand and know the different types of topologies and the inherent security risks they create

SANS | SEC401 | Security Essentials Bootcamp Style 21

This page intentionally left blank.



Physical and Logical Topologies

Physical topologies:

- How the network is actually connected
- How the data actually flows
- Wired or wireless
- Verification of physical topology is critical to ensure security
- Star topology most common

Logical topologies:

- How you communicate across wires
- Meaning of the information
- Language
- Ethernet most common (CSMA/CD)

SANS

SEC401 | Security Essentials Bootcamp Style 22

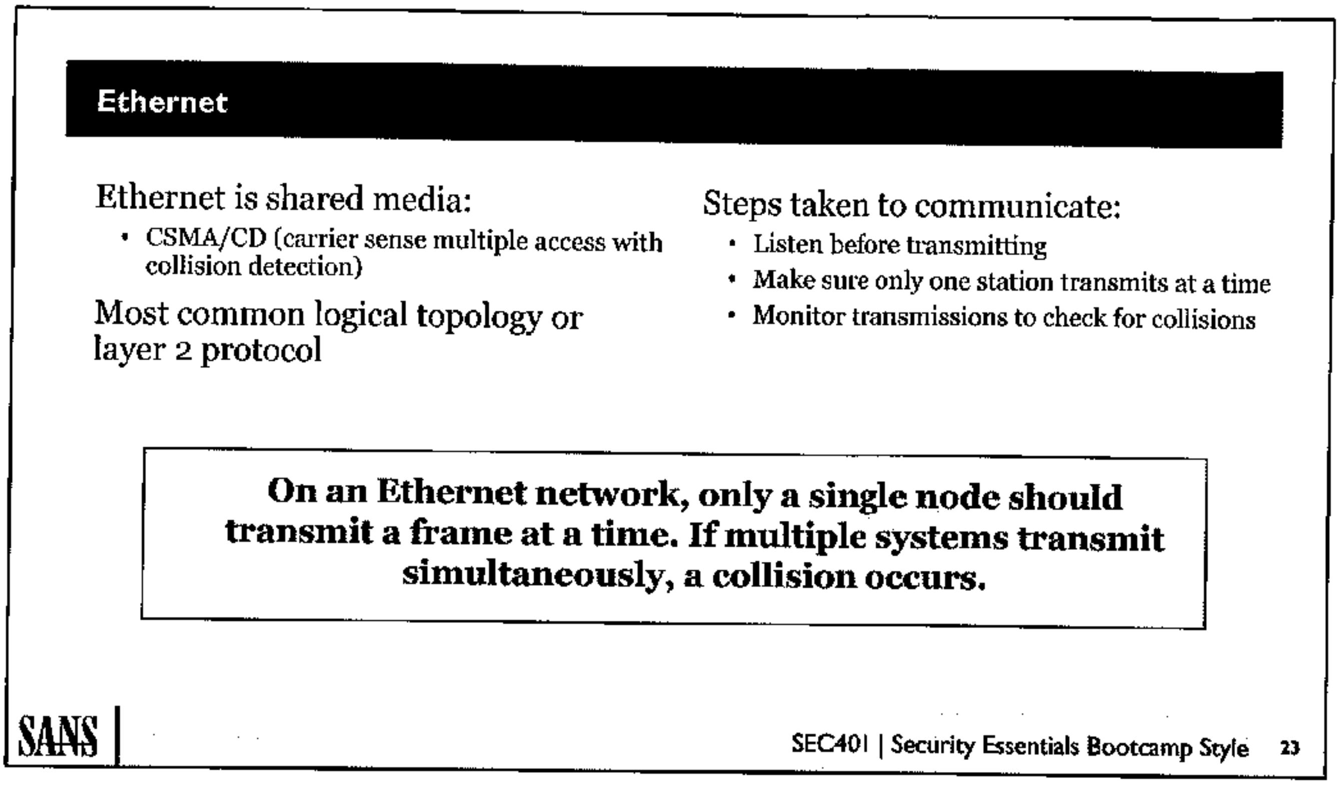
There are two types of topologies that together describe how systems on a network are able to communicate. To secure a network, you have to understand how they are physically connected and how they communicate.

A physical topology describes how the network is wired together. It is the layout of how systems are connected via cables or wireless devices. Wire-based physical topologies are easy to visualize because they are interconnected according to simple geometric patterns.

After the systems are interconnected, they must know the rules for sending signals to each other. These protocols are responsible for making sure that a signal sent by a system finds its way to its destination. The process that the protocol follows to send data over the cable, regardless of how it is physically wired, can be described using a logical topology.

To better understand the distinction between physical and logical topologies, consider how humans communicate. In most cases, our verbal interactions are guided by the grammar of a particular language, such as English. The English language has numerous rules that dictate how we should form words and sentences to help provide meaning to what we say. English grammar, then, is our logical topology, which describes our communication protocols. The physical topology of human interactions defines the systems that we use to communicate. For example, a telephone is one such physical topology; postal mail is another. A single logical topology (English) can be used with multiple physical topologies (telephone and mail). Similarly, each communication system can act as a carrier for different human languages.

A network security professional should understand the physical topology that is in place and make sure that it is implemented correctly. Changing one wire in a physical topology can greatly reduce or eliminate security, such as bypassing a firewall. Once the physical topology is understood, it is critical that you evaluate the logical topology that is in use and make sure it is properly secured.



Ethernet

Ethernet is shared media:

- CSMA/CD (carrier sense multiple access with collision detection)

Most common logical topology or layer 2 protocol

Steps taken to communicate:

- Listen before transmitting
- Make sure only one station transmits at a time
- Monitor transmissions to check for collisions

On an Ethernet network, only a single node should transmit a frame at a time. If multiple systems transmit simultaneously, a collision occurs.

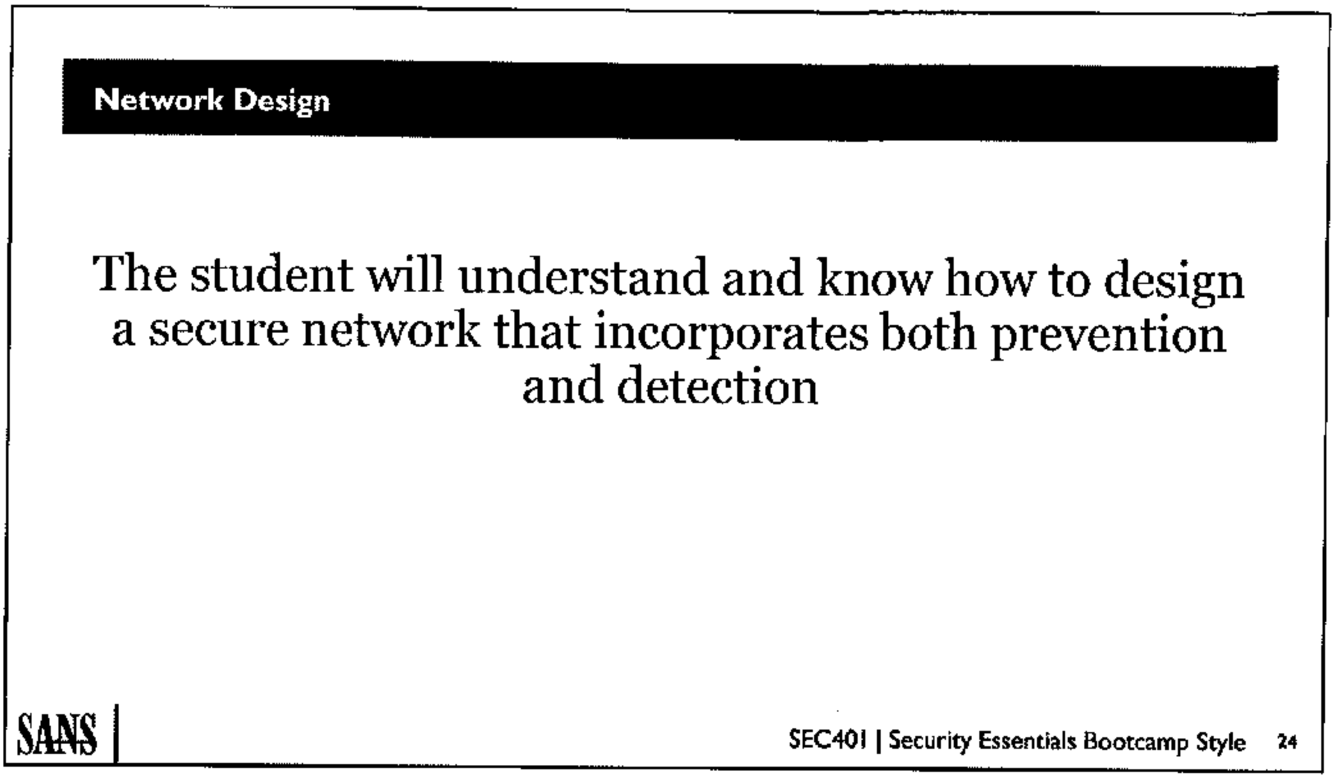
SANS | SEC401 | Security Essentials Bootcamp Style 23

Ethernet is the most popular media access protocol (Layer 2 protocol) currently used on LANs. In fact, it is nearly ubiquitous for networking, with the exception of the backbone itself. A chunk of data transmitted by Ethernet over the wire is called a frame. On an Ethernet network, only a single node should transmit a frame at a time. If multiple systems transmit simultaneously, a collision occurs. This collision can cause both signals to fail and require the systems to retransmit their frames.

To keep the number of collisions to a minimum, a system is required to check whether anyone else is already transmitting before placing a frame on the wire. If another system's signal is already on the wire, the system is expected to wait, according to the algorithm designed, to give each node a fair shot at using the network. If the line is clear, the system generates a signal and monitors the transmission to make sure that no collision occurred. These properties are summarized under Ethernet's designation as a carrier sense multiple access/collision detection (CSMA/CD) protocol.

Ethernet specifications actually define more than just protocols for sending signals over the wire. Other properties include cabling requirements for transferring data at desired rates and the maximum length of the wire segment. In addition, Ethernet standards specify which physical topology should be used for a particular type of Ethernet communication.

Because Ethernet is the most common type of Layer 2 protocol in use on networks, it is important to understand how it works. The shared segment aspect gives maximum flexibility, but it can also cause availability problems with collisions and confidentiality problems with sniffing the traffic.

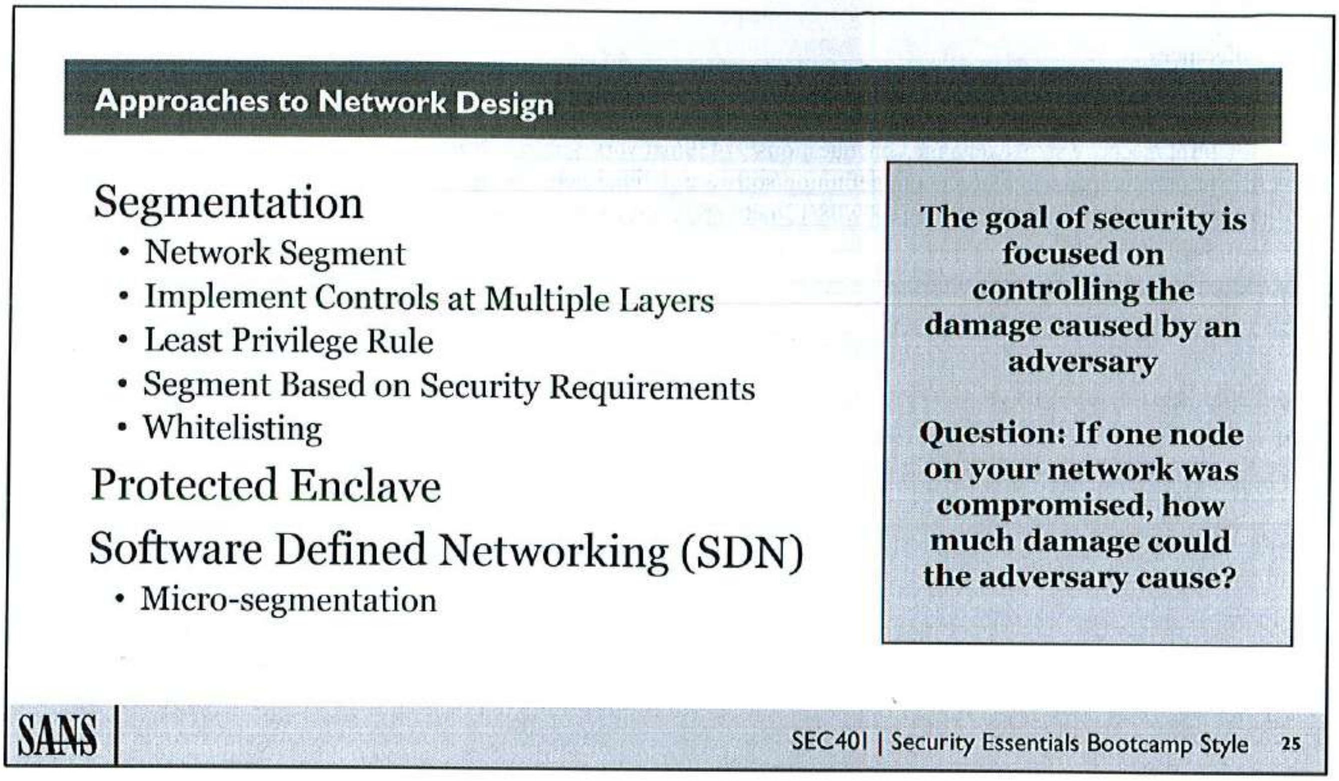


Network Design

The student will understand and know how to design a secure network that incorporates both prevention and detection

SANS | SEC401 | Security Essentials Bootcamp Style 24

This page intentionally left blank.



Approaches to Network Design

Segmentation

- Network Segment
- Implement Controls at Multiple Layers
- Least Privilege Rule
- Segment Based on Security Requirements
- Whitelisting

Protected Enclave

Software Defined Networking (SDN)

- Micro-segmentation

The goal of security is focused on controlling the damage caused by an adversary

Question: If one node on your network was compromised, how much damage could the adversary cause?

Network segmentation in computer networking is the act of splitting a computer network from the rest of the network by a device such as a repeater, hub, bridge, switch or router where each subnet is a network segment. A segment can contain one or multiple computers. Firewall's and VLANs provide a way to partition the network into smaller zones as well. Advantages of such splitting are primarily for boosting performance and improving security.

Implement Controls at Multiple Layers - the more layers of segmentation you can add the harder it will be for an attacker to gain unauthorized access. The number of layers requires a common sense approach and should be manageable from an operations standpoint.

Least Privilege Rule - access should only be provided to the user or system that is needed and nothing else.

Segment Based on Security Requirements - define zones based on where certain sensitive information resides.

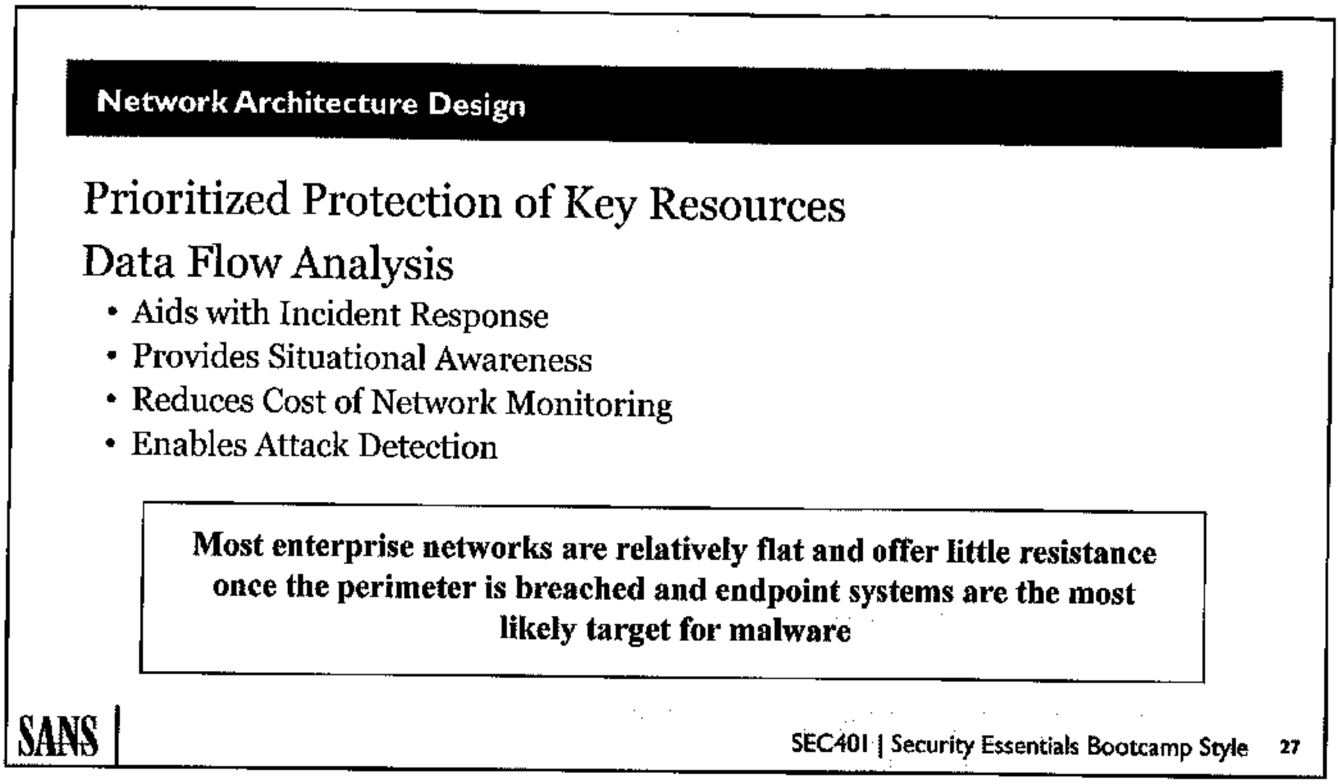
Whitelisting - rather than try to block all the "bad" things, it is simpler and more effective to define what you know to be good communication paths and block everything else.

A Protected Enclave is a segment of the internal network defined by a common set of security policies. It is necessary when the confidentiality, integrity, or availability of a set of resources is different from those on the rest of the network.

Software Defined Network (SDN) is a broad term covering several kinds of network technologies with the intent of making the network as flexible and agile as a virtual machine or virtualized storage. An SDN holds a lot of promise when it comes to network segmentation. As some of the more recent data breaches have shown, incorrect network segmentation can drastically increase your exposure to data theft or system attacks. With SDN the concept of "micro-segmentation" comes into play where traffic between any two endpoints can be analyzed and filtered based on a set policy.

References

1. http://www.linfo.org/network_segment.html
2. http://www.securityweek.com/improving-security-proper-network-segmentation
3. http://security.stackexchange.com/questions/97430/network-security-from-the-inside
4. http://searchsdn.techtarget.com/definition/software-defined-networking-SDN
5. https://trustcc.wordpress.com/2009/08/13/network-enclaves-%E2%80%93-enhanced-internal-network-segmentation/



Network Architecture Design

Prioritized Protection of Key Resources

Data Flow Analysis

- Aids with Incident Response
- Provides Situational Awareness
- Reduces Cost of Network Monitoring
- Enables Attack Detection

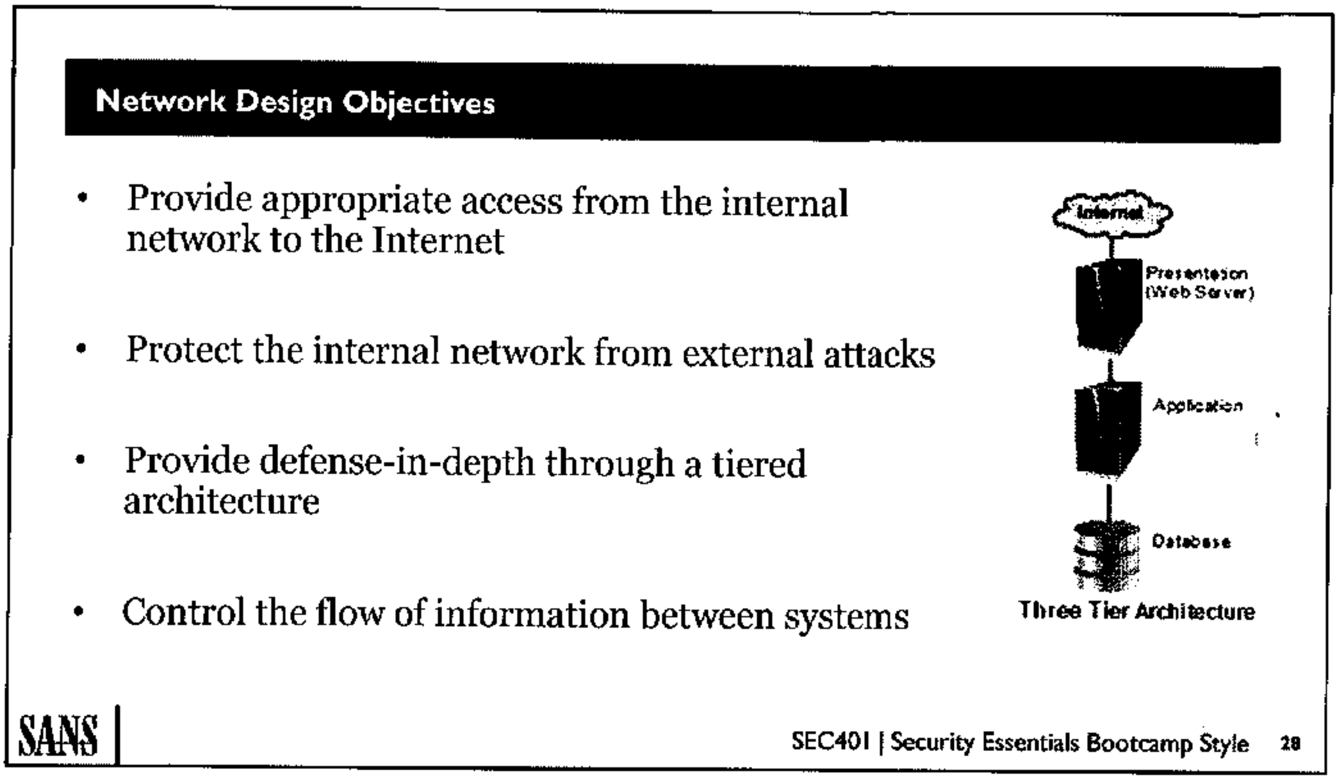
Most enterprise networks are relatively flat and offer little resistance once the perimeter is breached and endpoint systems are the most likely target for malware

In order to avoid being overwhelmed by security vulnerabilities, the best approach is proper prioritization. Most enterprise networks are relatively flat and offer little resistance once the perimeter is breached and desktop systems are the most likely target for malware. So creating some separation between desktop systems and the critical data stored on servers is a big step in protecting the network. Next would be to separate desktops from each other to limit desktop reconnaissance and worm type propagation between desktops. The LAN connected to desktops should be considered permanently hostile and therefore the desktops should only be allowed the minimum data required to operate. Servers should be separate from each other, and from the desktops, with firewalls.

Data Flow or Net Flow is a method of collecting IP traffic information and monitoring network traffic. Through the analysis of flow data, a picture of network traffic flow and volume can be built. Net Flow lets you can see where network traffic is coming from and going to and how much traffic is being generated. Net Flow provides a 24×7 account of all network activity and when an incident does occur, the information needed to identify the root cause and begin cleaning up is in the stored Net Flow.

Net Flow allows you to see what's happening on your network beyond the perimeter which also is the most difficult place to bring security analysis to bear. With Net Flow you can see smartphones, IP phones, laptops, servers, and virtualized infrastructure at this layer. The larger and more distributed your network the more value Net Flow traffic will provide. It only requires a few commands entered on the router to have network visibility at a specific location. Net Flow-based analysis relies on algorithms and behavior rather than signature matching which allows you to detect attacks that may not have a signature yet, sometimes referred to as zero-day attacks.

References
1. https://blog.qualys.com/news/2017/01/17/overwhelmed-by-security-vulnerabilities-heres-how-to-prioritize
2. https://www.plixer.com/blog/netflow/top-5-uses-of-netflow-for-network-security/
3. https://insights.sei.cmu.edu/sei_blog/2016/09/traffic-analysis-for-network-security-two-approaches-for-going-beyond-network-flow-data.html



Network Design Objectives

- Provide appropriate access from the internal network to the Internet
- Protect the internal network from external attacks
- Provide defense-in-depth through a tiered architecture
- Control the flow of information between systems

Three Tier Architecture

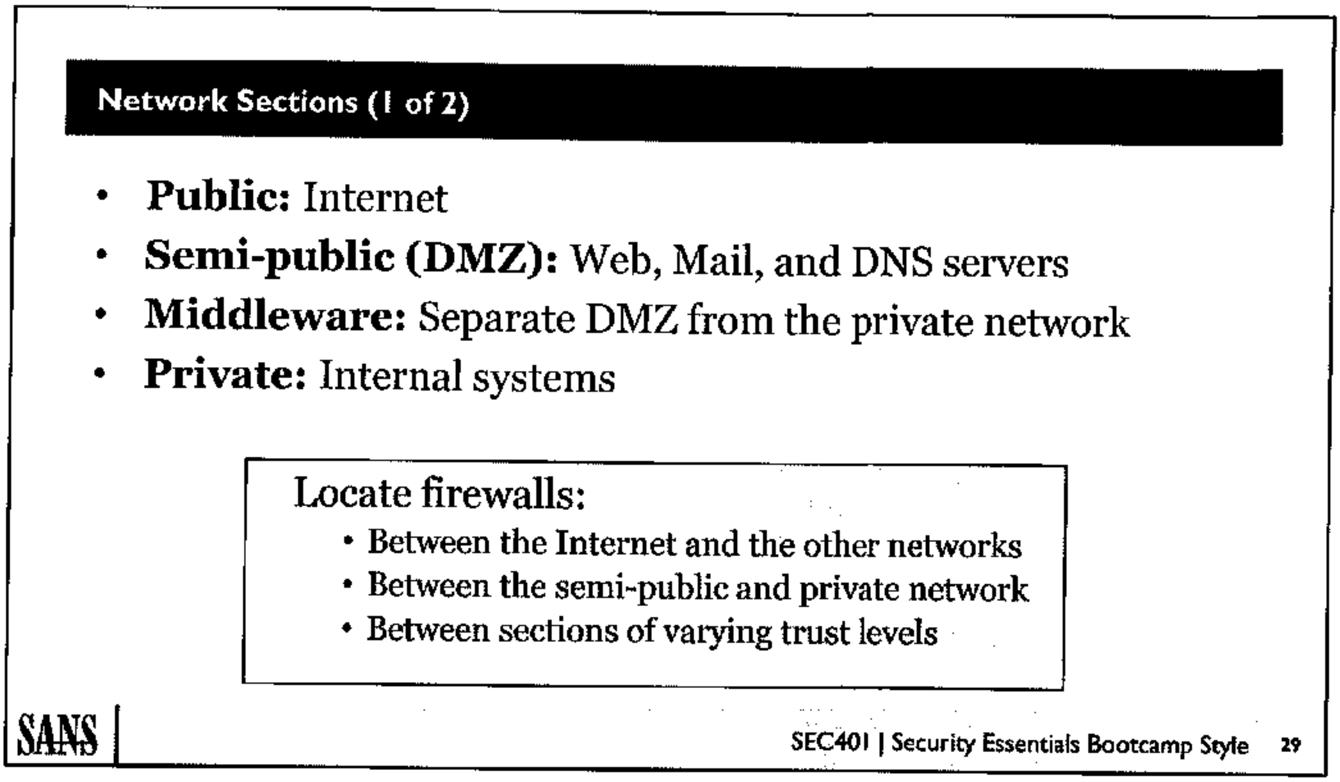
SANS | SEC401 | Security Essentials Bootcamp Style 28

We want to walk you through the fundamental steps of designing a basic network architecture, building upon what you have learned so far. In this example, one of the requirements for the network that we need to design is to allow internal users to access the Internet. Additionally, certain systems located on the company's network need to be reachable from the Internet, including

- Web server that displays information about the company and its products
- Mail server that allows the company's employees to send and receive e-mail
- DNS server that hosts records for the company's public domain (such as "example.com")

According to these requirements, we need to provide limited access from the Internet to our network. However, aside from the servers listed previously, we do not want any Internet user to access our internal systems. Because of the link to the Internet, our defenses need to be designed to protect the network from external attacks.

Most of our design decisions will be based on the approach of "defense-in-depth," which advocates the use of multiple layers of protection to guard against failure of a single security component. One of the elements of defense-in-depth is the principle of resource separation, which we use when dividing the internal network into several sections.



Network Sections (1 of 2)

- **Public:** Internet
- **Semi-public (DMZ):** Web, Mail, and DNS servers
- **Middleware:** Separate DMZ from the private network
- **Private:** Internal systems

Locate firewalls:

- Between the Internet and the other networks
- Between the semi-public and private network
- Between sections of varying trust levels

SANS

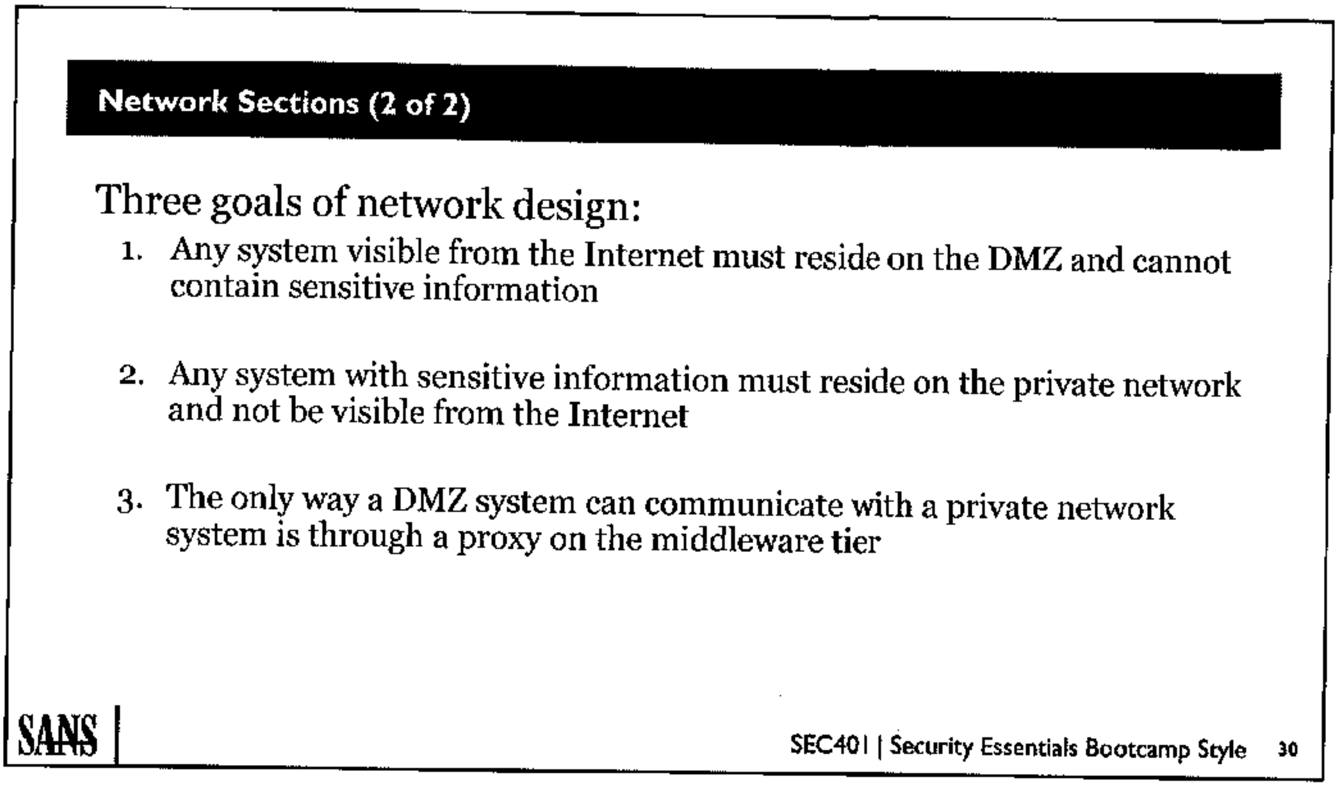
SEC401 | Security Essentials Bootcamp Style 29

Security is always a balance between functionality and security. The key rule we always follow is to give an entity the least access it needs, while still allowing them to perform their job. With network architecture, the key is to provide proper segmentation so that a person can access the appropriate data by reducing the risk of potential compromise.

If you look at the requirements for systems that reside on our network, you will probably notice that they can be grouped into several categories, according to the type of information that they contain:

- **Public:** These resources reside on the Internet and, from the perspective of our company's network, cannot be trusted.
- **Semi-public:** These resources are our contributions to the Internet, and they take the form of web publications, e-mail messages, and DNS records. Semi-public servers must be reachable from the Internet and might also have to access the Internet.
- **Middleware:** Used to separate the DMZ from the private network. Often contains proxy servers that can filter and block unauthorized access. Provides an extra layer of protection because connections between the DMZ and private network are high risk.
- **Private:** These are the company's internal systems. We have no desire to provide any services from this category to users on the Internet. Therefore, we want to actively protect information that resides here.

Systems in each category serve a similar purpose and have common security requirements. This allows us to group resources within a category by placing them into a common network section. In such a design, our view of the networked world will be split into three sections: public, semi-public, and private. You can further subdivide based on potential risk and system functionality.



Network Sections (2 of 2)

Three goals of network design:

1. Any system visible from the Internet must reside on the DMZ and cannot contain sensitive information
2. Any system with sensitive information must reside on the private network and not be visible from the Internet
3. The only way a DMZ system can communicate with a private network system is through a proxy on the middleware tier

SANS | SEC401 | Security Essentials Bootcamp Style 30

In order to implement proper security, there are 3 core rules that we want to follow:

1. Any system visible from the Internet must reside on the DMZ and cannot contain sensitive information
2. Any system with sensitive information must reside on the private network and not be visible from the Internet
3. The only way a DMZ system can communicate with a private network system is through a proxy on the middleware tier

These 3 rules are an example of resource separation at work; we placed systems with different security requirements into separate areas. In our example, grouping similar resources allow us to control how they interact with each other. Enforcement of such access restrictions is frequently the job of a firewall.

Installing a firewall is one of the most fundamental ways to protect our systems from an external attack. The decision to use a firewall is simple enough, but a more interesting question is, "Where should we place it?" The idea is to position the firewall in a location that would allow it to control access or restrict traffic that crosses boundaries of network sections.

First, the firewall needs to be located in a place that allows it to ensure that any outbound traffic is legitimate. By the same token, we also want it to control inbound connections. This means that the firewall must be in a position that allows it to grant connection requests to our web, mail, and DNS servers. The firewall should also be able to block inbound connections to systems on the private section of the network.

To help determine the optimal place for the firewall, consider the basic paths that traffic can traverse on our network:

From private systems to the Internet
From private systems to semi-public servers
From semi-public servers to the Internet
From the Internet to semi-public servers

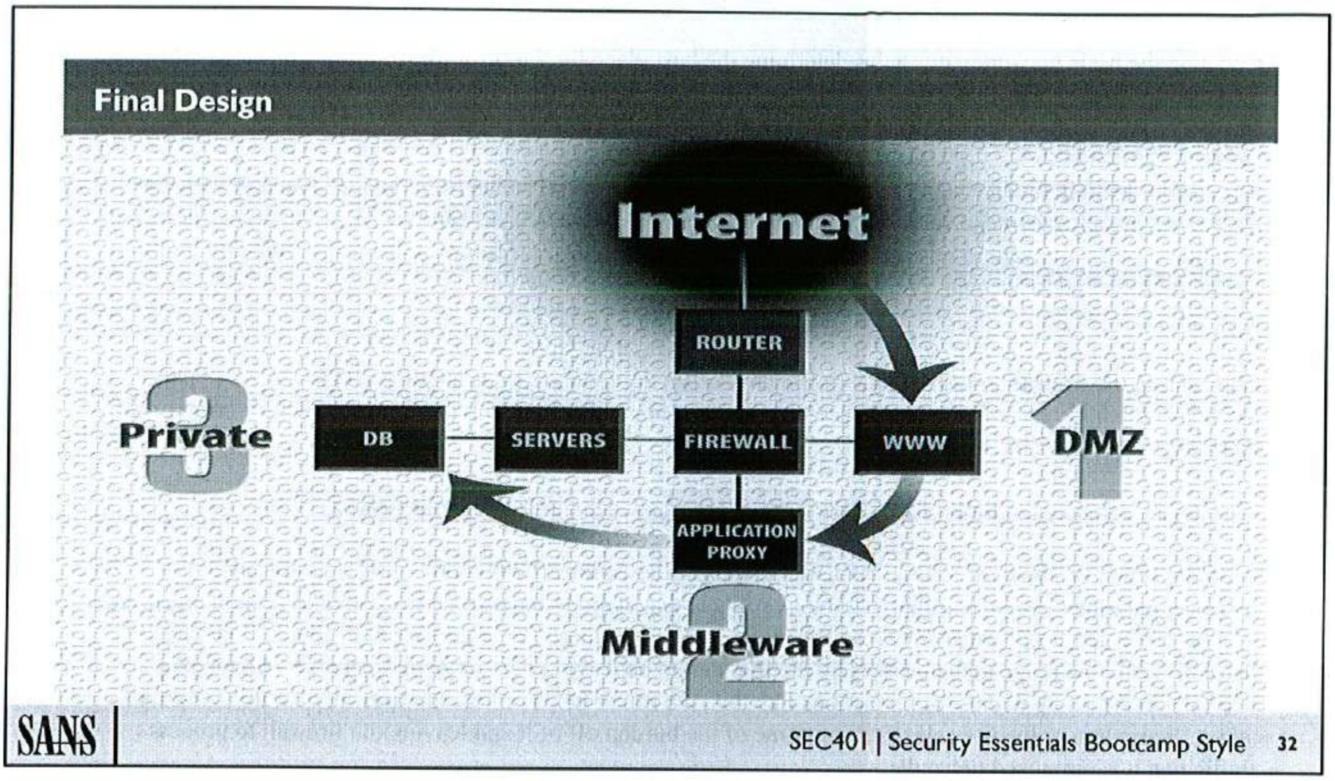
Knowing the basic network paths helps determine the best place for the firewall. It should be located at the intersections of the paths that we outlined previously. Now that the firewall's placement has been determined, let's look into adding additional defensive layers to further protect the network.

Defense-in-depth is fundamental to the design of a secure network. It stems from the idea that software can have flaws, people can make configuration mistakes, and hardware devices can fail. To compensate for events like these, we do not want to rely on a single mechanism to defend our resources. Instead, we deploy multiple layers of protection to account for the possibility that one of them may fail.

In the context of our example, separating systems into several network sections is one defense layer. Configuring the firewall to restrict how traffic crosses section boundaries is another. Yet another defense mechanism that we can employ is a device that operates in conjunction with the firewall when filtering traffic that leaves and enters the network.

This device can take the form of a border router, placed between our ISP and our firewall.

A border router can be used to filter out certain types of network traffic that obviously are unwanted. For instance, the router can be configured to block packets that claim to have come from invalid IP addresses, such as those allocated for private addresses used by RFC 1918. Similarly, only packets with source IP addresses that fall within the range assigned to us by the ISP should be leaving our company's network. Although a firewall can be set up to enforce restrictions like this, the router helps protect the firewall host itself from an attack. It also assists the firewall by taking some of the burden off of it and leaving the firewall to process traffic that it is optimized to handle.



Final Design

Internet

ROUTER

3 Private

DB

SERVERS

FIREWALL

WWW

1 DMZ

APPLICATION PROXY

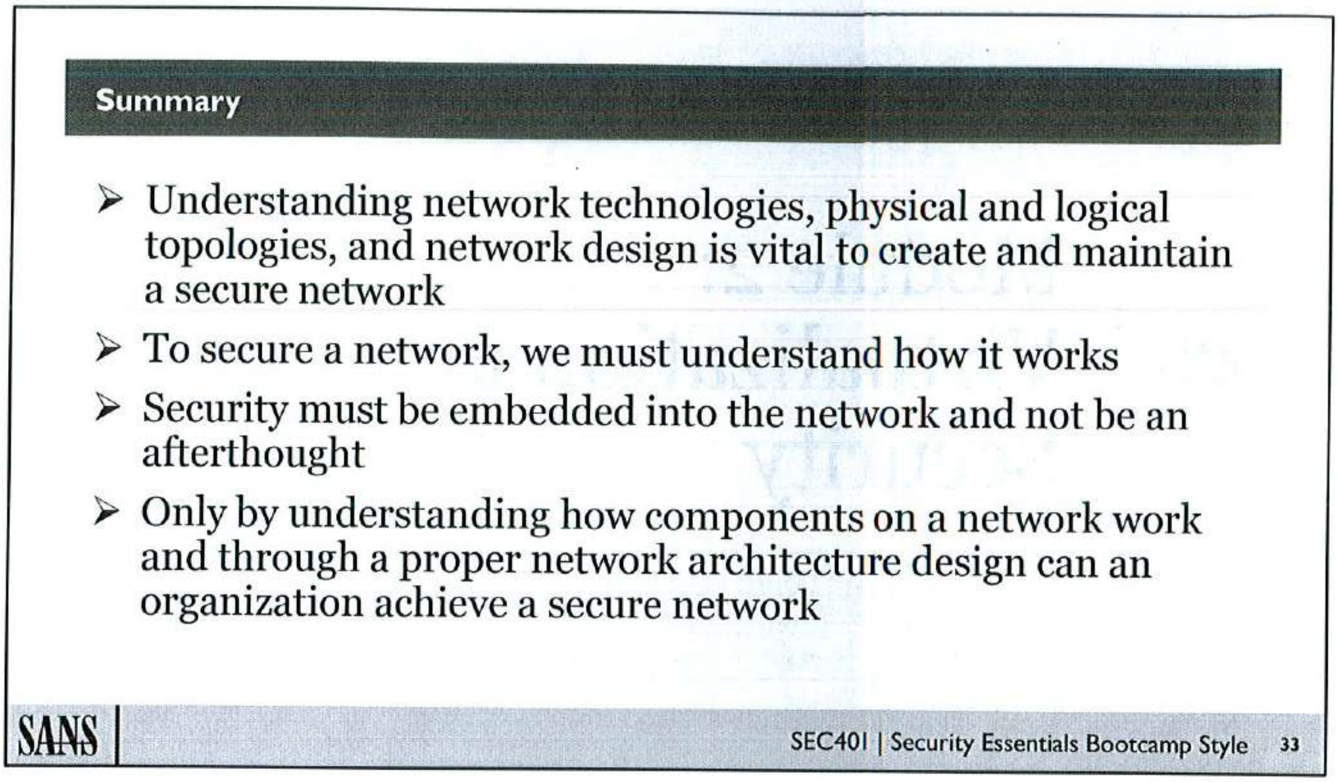
2 Middleware

SANS

SEC401 | Security Essentials Bootcamp Style 32

This slide represents our final design. Traffic passes to and from the Internet through the border router, which filters out traffic that is obviously not wanted using access control lists (ACLs). The firewall controls the flow of traffic to and from semi-public (DMZ) and private network sections. Switches within each section guard against sniffers that might be installed on compromised systems. To further enhance security through the principles of defense-in-depth, all hosts (servers and workstations) in our overall configuration would actively run antivirus software with the latest signature updates. Hosts might also have host-based firewalls applied or host-based intrusion detection systems.

The concept of defense-in-depth can be applied to different types of networks, at the office and at home. For example, if you were putting together a home network, you might use a basic firewall built into your cable modem/DSL router to guard against direct attacks from the Internet. However, the firewall cannot protect you against all threats. For instance, you may receive a virus via an infected document that was e-mailed to you or handed to you on a USB. It would be a good idea to install antivirus software on your workstations to account for such attack vectors. A malicious website that you visited might attempt to exploit a vulnerability in your web browser. A firewall and antivirus software might help combat this threat, but keeping up with application and OS patches provides another highly effective layer of protection. Similarly, a malicious worm might require access through some service port and be blocked from your system by the firewall. Although some exploit might penetrate a firewall, a securely configured authentication system or intrusion detection system (IDS) can thwart the attack. Not relying on a single security mechanism to protect you against attacks is the fundamental principle of defense-in-depth.



Summary

- Understanding network technologies, physical and logical topologies, and network design is vital to create and maintain a secure network
- To secure a network, we must understand how it works
- Security must be embedded into the network and not be an afterthought
- Only by understanding how components on a network work and through a proper network architecture design can an organization achieve a secure network

SANS

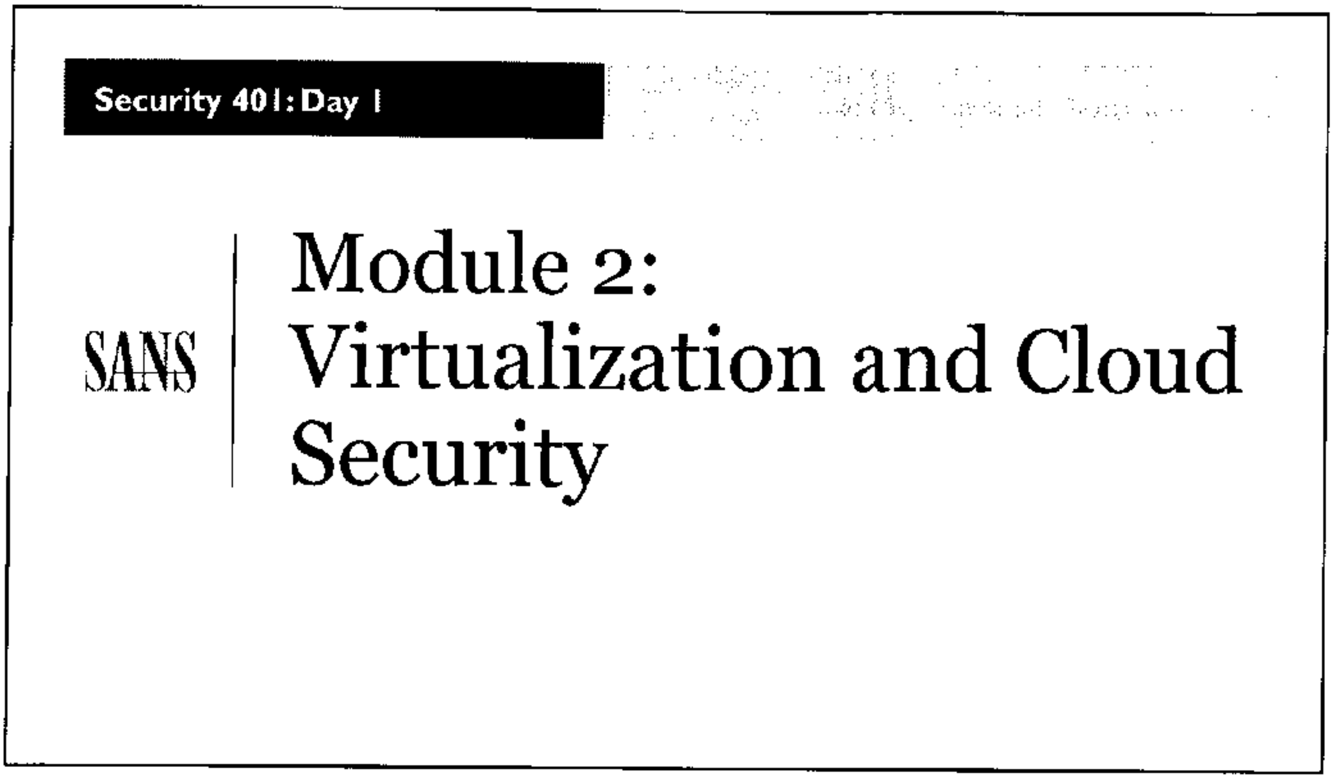
SEC401 | Security Essentials Bootcamp Style 33

Let's summarize the critical components of network architecture and design.

Only by understanding and knowing how an architecture is connected and configured can an organization implement proper security. From how the wires are connected to the data that flows over those wires, knowledge is power in making sure that a network is properly secured. Too often, organizations buy state-of-the-art components and are surprised when a security breach occurs. Components do not make a network secure. Only by understanding how components on a network work and through a proper network architecture design can an organization achieve a secure network.

A physical topology describes how cables and devices are wired together to form a functional network. Among physical topologies, star is most commonly used on LANs. It is the most versatile because it can support all the media access protocols. Network nodes connected according to a physical star topology are wired to a common device, such as a switch.

A logical topology is independent of the physical topology and describes low-level communication mechanisms that allow systems on the network to exchange signals with one another. Ethernet is the most popular LAN protocol for sending signals over the wire. On an Ethernet network, only a single frame can be transmitted at a time to prevent collisions. Ethernet nodes are required to monitor the status of the signals that they issue to detect collisions and resend the frames if necessary. Systems on Ethernet networks are identified by unique MAC addresses, which usually are embedded into their network cards.

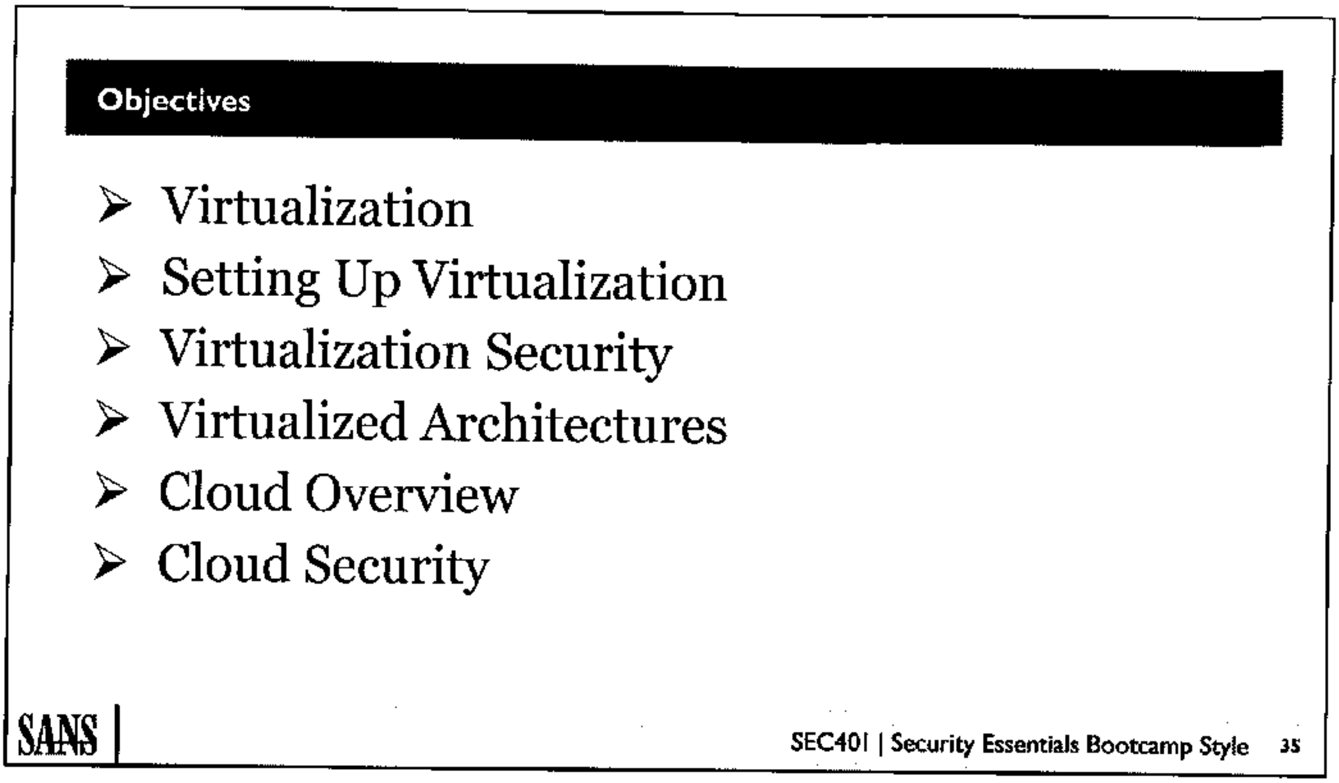


Security 401: Day 1

SANS

Module 2: Virtualization and Cloud Security

Module 2: Virtualization and Cloud Security
This page intentionally left blank.



Objectives

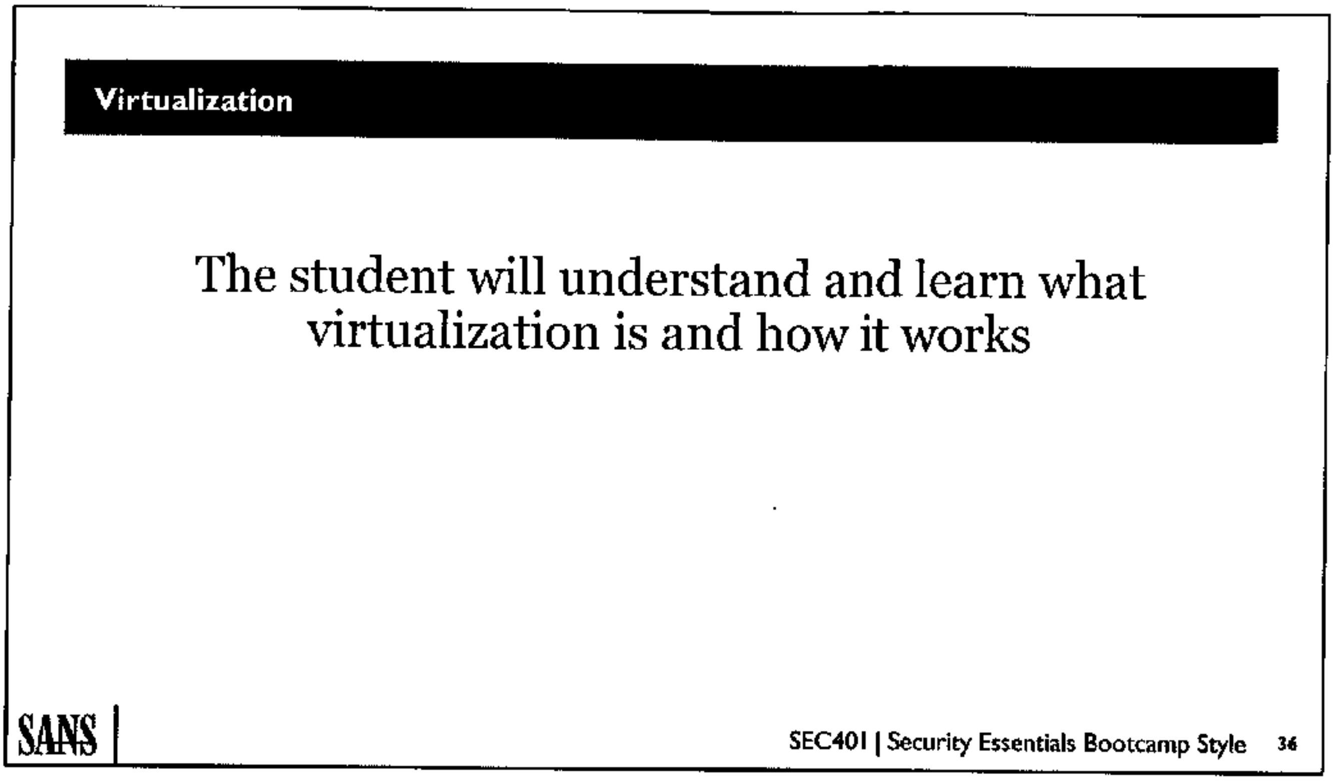
- ➢ Virtualization
- ➢ Setting Up Virtualization
- ➢ Virtualization Security
- ➢ Virtualized Architectures
- ➢ Cloud Overview
- ➢ Cloud Security

SANS | SEC401 | Security Essentials Bootcamp Style 35

Technology is always changing and it is critical to stay up to speed on the new advances. In many cases, new developments in IT provide enhanced functionality and cost-saving capabilities, but security is sometimes overlooked or not addressed until it is too late. Therefore it is always important to strike a balance between functionality and security. In this module, we will look at two related areas of technology: virtualization and cloud services.

Virtualization has changed the traditional computing platform. Instead of having one computer with the associated hardware and one operating system installed, now an organization can have one computer with multiple operating systems installed. While this allows for better utilization of resources, it could also create additional security concerns. In this module, we will examine what virtualization is, how to set it up, including configuring the lab environment you will use for the rest of the course, securing a virtualized environment and virtualization architectures.

The second part of this module will focus on cloud services and security. In the simplest sense, cloud allows for the IT infrastructure to be outsourced to a third party. For most organizations, running and maintaining a data center is not their primary business. It is infrastructure that is needed to support the business. With cloud, the IT infrastructure can be outsourced to a third party, so the organization can focus on their primary business. While this can create potential cost savings, it also introduces some security issues because an organization has less control over the day to day implementation of security. With cloud-based services we will look at an overview of the cloud and some of the ways to implement effective security.

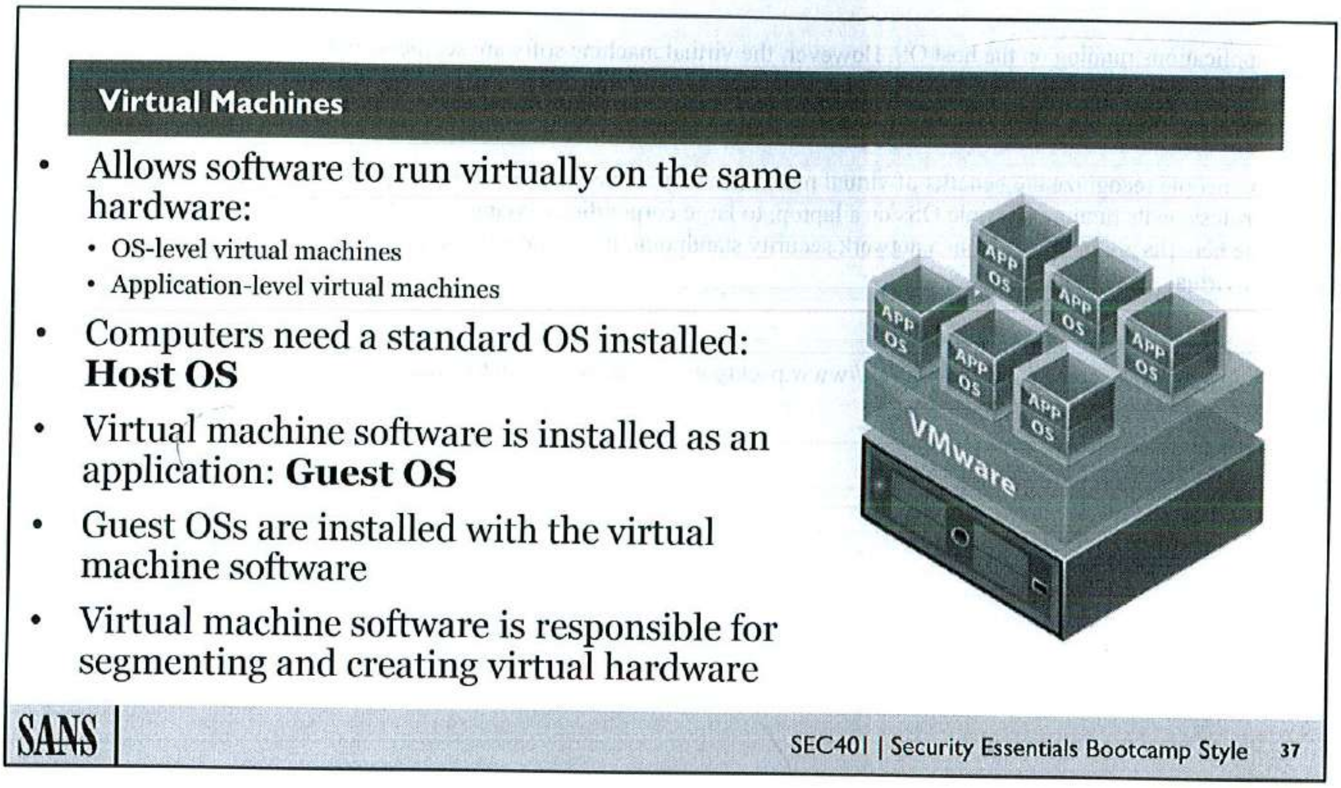


Virtualization

The student will understand and learn what virtualization is and how it works

SANS | SEC401 | Security Essentials Bootcamp Style 36

The following references were used for the sections on virtualization and security.
http://www.catbird.com/vsecurity/best-practices
http://www.datacenterknowledge.com/archives/2015/03/09/virtualization-security-overcoming-risks/
https://www.techopedia.com/definition/30243/virtualization-security
http://searchservervirtualization.techtarget.com/tip/Step-by-step-virtualization-Using-virtualization-to-improve-security-part-1
http://www.computerweekly.com/feature/Hypervisor-security-New-techniques-for-securing-virtual-machines
http://searchcloudsecurity.techtarget.com/answer/Can-virtual-machine-introspection-improve-cloud-security
https://cloudsecurityalliance.org
https://blog.cloudsecurityalliance.org/2011/03/21/three-cloud-computing-data-security-risks-that-can%E2%80%99t-be-overlooked/



Virtual Machines

- Allows software to run virtually on the same hardware:
  - OS-level virtual machines
  - Application-level virtual machines
- Computers need a standard OS installed: **Host OS**
- Virtual machine software is installed as an application: **Guest OS**
- Guest OSs are installed with the virtual machine software
- Virtual machine software is responsible for segmenting and creating virtual hardware

SANS

SEC401 | Security Essentials Bootcamp Style 37

Even though the price of hardware decreases every day, there are still many reasons why you would not want to purchase more of it. In addition to hardware's less expensive price tag, processors are becoming more powerful and memory is less expensive. Therefore, many companies have high-end servers and use only a small percentage of their processing capabilities. Running several pieces of software virtually on the same hardware has numerous benefits.

The most common form of virtual machine is an operating system (OS) virtual machine. The OS virtual machine enables multiple operating systems to run independently on the same hardware. You can use a virtual machine as a laptop that needs to run different operating systems or as high-end server farms that need to run many applications at a reduced cost in hardware. The first use is more of a novelty and does not have huge cost savings to an organization. However, it is still a powerful tool for a network security professional who can run UNIX tools on a Windows system without having to perform a reinstallation of any components. This is one of the main reasons we cover virtual machines in this section. The modern security professional must have access to both Windows and UNIX operating systems because there are some tools that will run on only one OS. Dual-boot machines require reinstallation of key components. Multiple laptops require additional hardware, and bootable CDs have limited read-only capabilities. Virtual machines are a perfect solution for solving this problem.

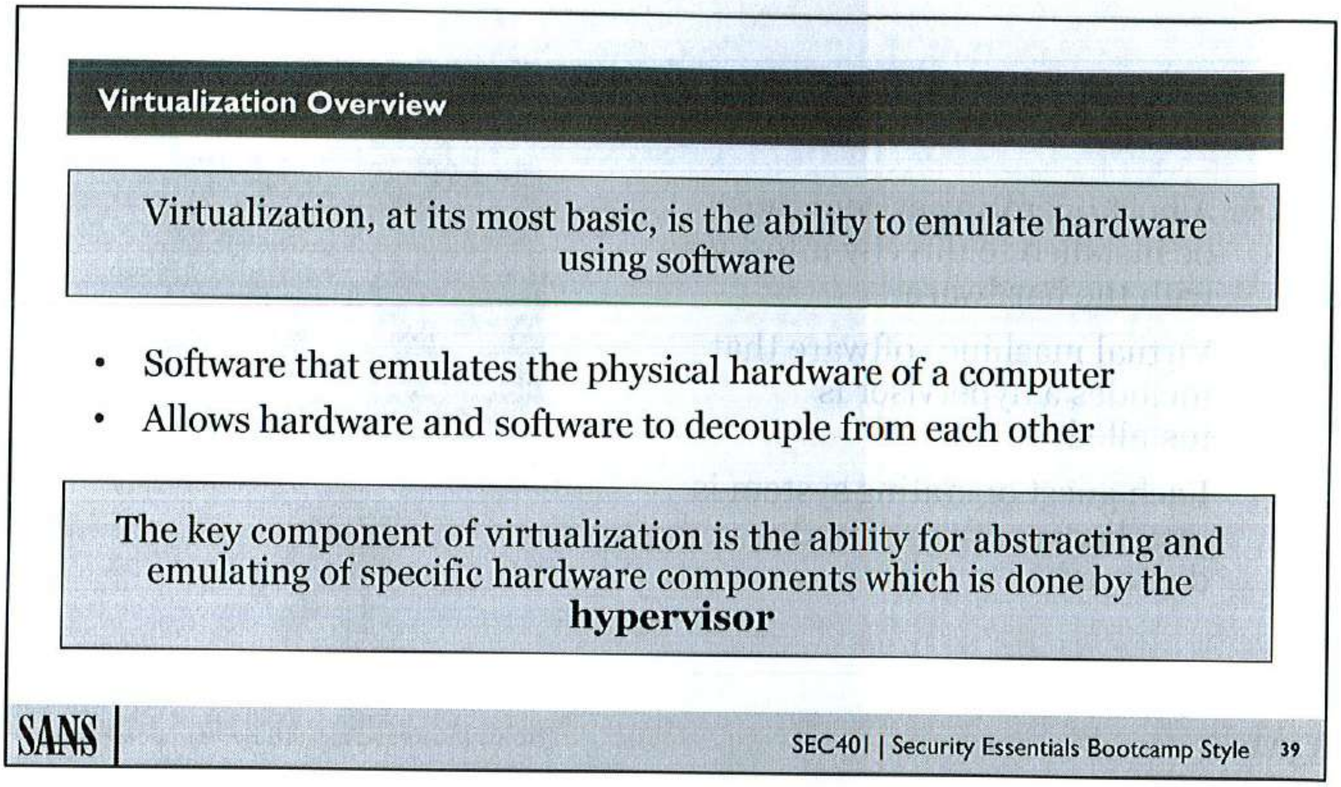
As applications become more powerful, attackers are taking advantage of the increased functionalities to find vulnerabilities in the software. Removing these applications is not always an option, yet they are a prime source of malicious code that can infect an entire system. An exploit in a web browser can be used to compromise the entire system. One solution is to run application-level virtual machines. Now an application runs in a separate virtual machine so if it is exploited or compromised, the scope of the attacker is limited to a separate system and cannot compromise the main operating system in which all the other applications run.

When talking about virtual machines, it is important to be able to distinguish between the main operating system and the virtual software. Because a computer needs an operating system to boot up the computer, it is referred to as the host operating system. On the host operating system, you would install a virtual machine application, such as VMware. This virtual machine software enables you to run multiple guest operating systems that are actually applications running on the host OS. However, the virtual machine software segments out memory and hardware so they look and act like independent OSs, even though there is always one host and one or more guest operating systems at any given time.

As people recognize the benefits of virtual machines, they will continue to grow in popularity. From security professionals running multiple OSs on a laptop, to large corporations saving millions of dollars on hardware, the benefits are obvious. From a network security standpoint, it is critical to understand the value and benefit of virtual machines.

Reference

1. Virtual Machine Concepts - https://www.packtpub.com/books/content/virtual-machine-concepts



Virtualization Overview

Virtualization, at its most basic, is the ability to emulate hardware using software

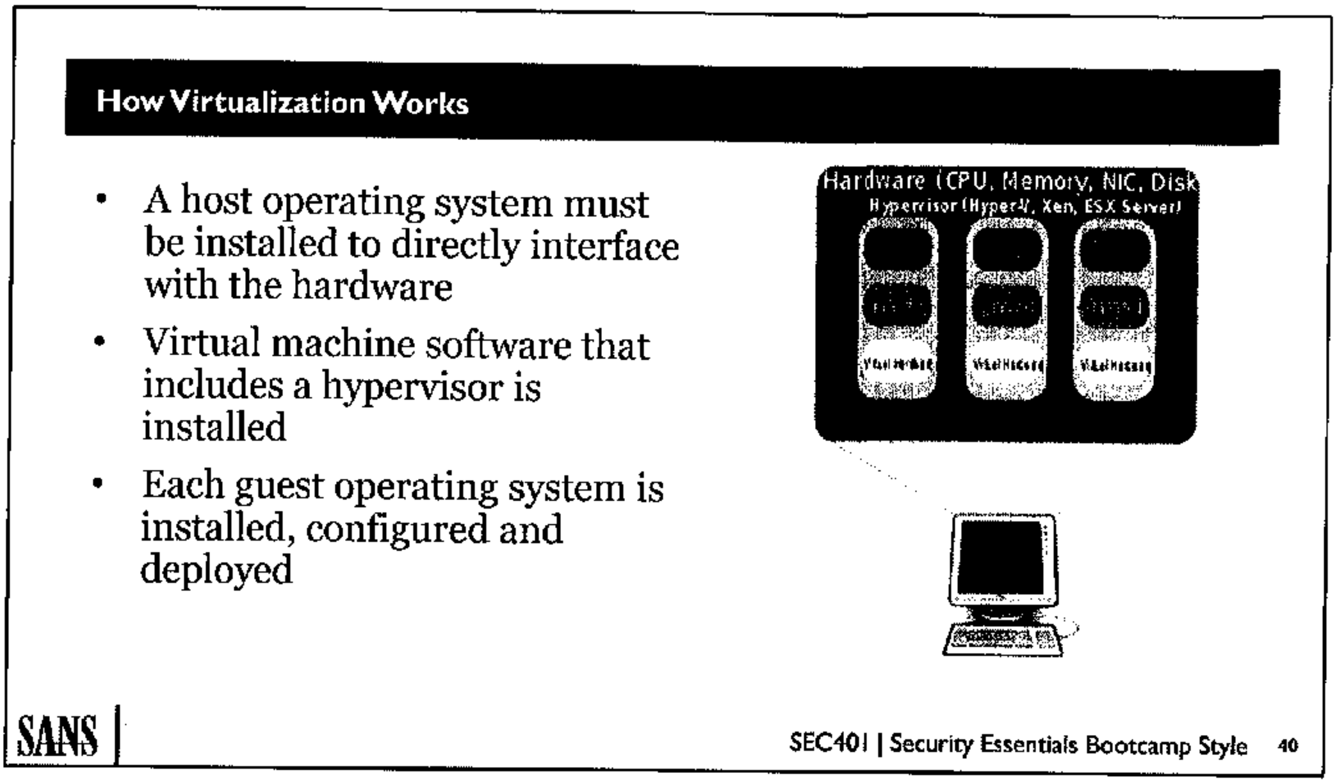
- Software that emulates the physical hardware of a computer
- Allows hardware and software to decouple from each other

The key component of virtualization is the ability for abstracting and emulating of specific hardware components which is done by the **hypervisor**

SANS | SEC401 | Security Essentials Bootcamp Style 39

Virtualization, at its most basic, is the ability to emulate hardware using software. This is valuable and useful as it allows for multiple types of operating systems, or multiple versions of one type, to be used without having to configure different hardware platforms. The way the software accomplishes this is through the user of a hypervisor. In any computer, at its most basic, an operating system of some type still needs to boot up on some type of physical hardware. This operating system can be a full OS such as Windows or Linux, or a stripped down OS designed to boot up and operate in a manner that provides virtualization. In both cases, an operating system is loaded, at which point the emulation software is loaded on top. This emulation software is the hypervisor.

Since the hypervisor is responsible for abstracting and emulating specific hardware components and configurations, this gives the software more flexibility to decouple the operating system from the specific underlying hardware. When a virtual machine or guest operating system boots up, the hardware that the operating system detects is actually the simulated hardware presented by the hypervisor, not the actual hardware itself. This decoupling adds tremendous versatility and allows the hypervisor to run multiple virtual environments each of which may have slightly different hardware requirements.



How Virtualization Works

- A host operating system must be installed to directly interface with the hardware
- Virtual machine software that includes a hypervisor is installed
- Each guest operating system is installed, configured and deployed

Hardware (CPU, Memory, NIC, Disk
Hypervisor (Hyper-V, Xen, ESX Server)

SANS | SEC401 | Security Essentials Bootcamp Style 40

Virtual machine software (VM software) is a simple emulator for a computer, all created in software. You install a VM program on top of another operating system, known as the host operating system, such as Windows or Linux. Then, you can boot up virtual computers in the VM software, each of which is referred to as a guest operating system or a virtual machine. Each guest OS has its own memory allocation, virtualized network adapters, hard drive(s), and other hardware components. The different guests and the host appear to be truly independent operating systems, all running on the same hardware.

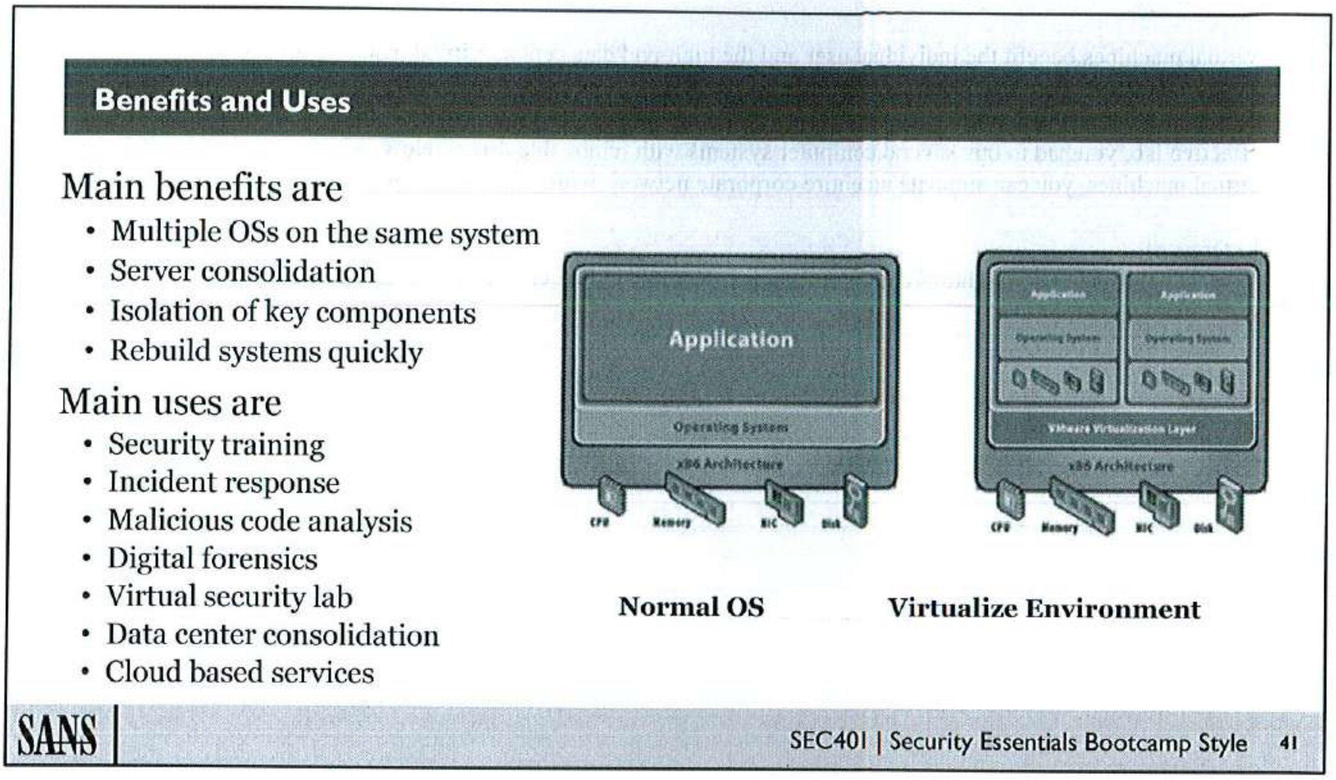
VM software is loaded on the host operating system, just like any other application. After it is installed, its job is to interface with all the hardware components and create virtualized hardware environments so other operating systems and applications can be installed. The operating system or application is not aware that it is running on a virtual machine. Therefore, you can install four different guest operating systems, each with a virtualized network interface card (NIC). This means each one can be assigned a unique IP address and connect and interface as if it was installed on four different computer systems and connected with a switch.

Virtualization is the workhorse of VM software. The VM software has to take the physical memory, process, NIC, and other hardware components and create virtualized components for the different guest operating systems to interface with. The VM software has to track the exchange between the physical and virtual components to make sure everything works properly.

When troubleshooting an operating system, it is important to remember whether it is a host or guest operating system. Although they seem to work the same way, there are subtle differences between the two. A host operating system directly accesses the hardware, and the guest operating system accesses virtual hardware through an emulator.

Reference

1. Virtual Machines - https://en.wikipedia.org/wiki/Virtual_machine



Benefits and Uses

Main benefits are

- Multiple OSs on the same system
- Server consolidation
- Isolation of key components
- Rebuild systems quickly

Main uses are

- Security training
- Incident response
- Malicious code analysis
- Digital forensics
- Virtual security lab
- Data center consolidation
- Cloud based services

Normal OS

Virtualize Environment

Virtual machines have had tremendous growth. In the early days, when hardware was slow and expensive, you could barely run one operating system on a computer and a few applications without having performance issues. Starting in 2000, processors became fast, and many systems had dual-core processors. In addition, it was not uncommon to have a minimum of 4GB of memory, even in laptops. This meant that on the average system, the processor and memory utilization were less than 30%, which means plenty of power was available. This extra capacity could easily be used to run one or several virtualized operating systems.

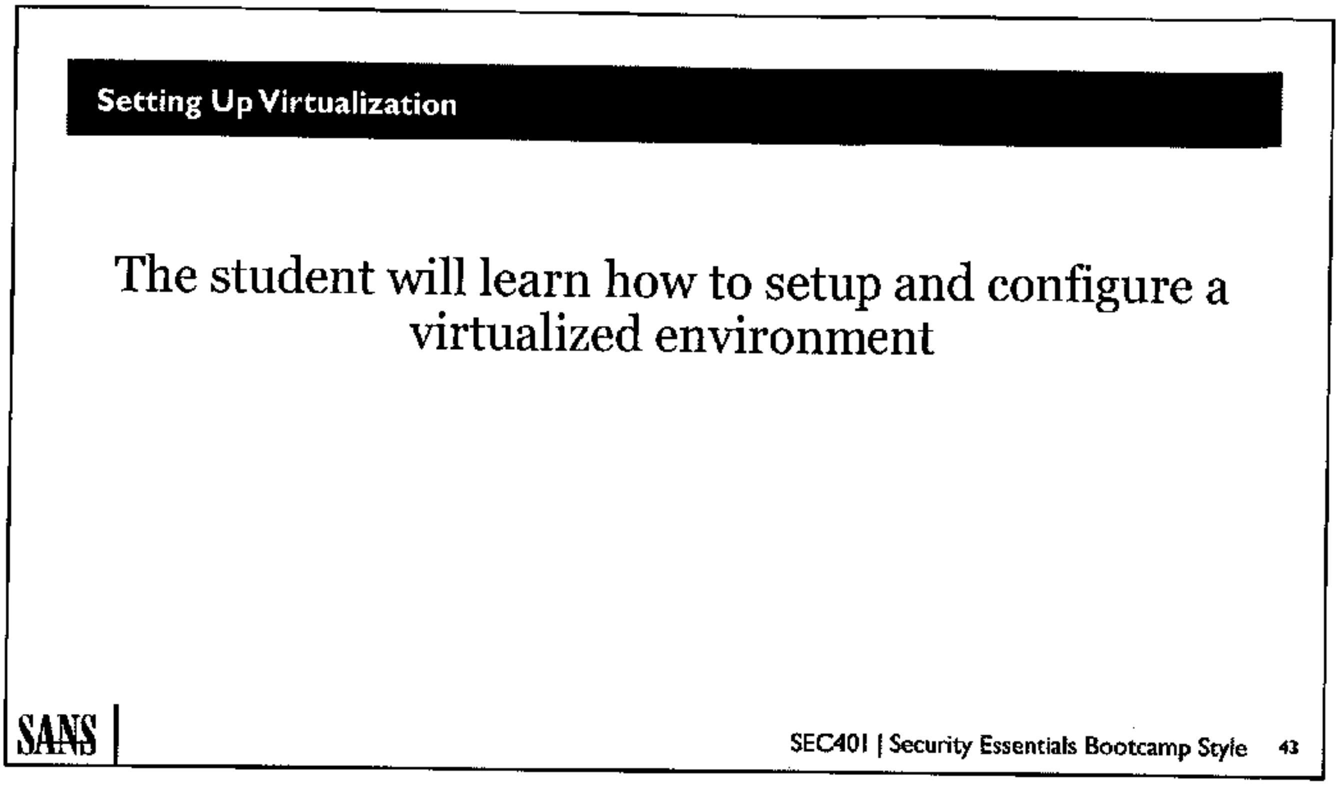
One of the main and most obvious benefits is being able to run multiple operating systems on the same computer. Whether you are bringing a laptop to a client and need to run tools on Windows and UNIX, or you are a Mac user who wants to be able to access applications that are built for Windows operating systems, virtual machines enable you to use one computer to run many operating systems at the same time. In the past, you would have to either bring multiple computers or set up a dual-boot system. Neither one of these is a great option. From a server perspective, instead of having to buy a large number of systems, a small number of higher-end systems can run the same number of operating systems with less hardware to maintain.

From a Disaster Recovery Plan (DRP) perspective, instead of having ten servers with one OS and no redundancy on each, you could have two systems with five operating systems and four failover systems each and still have less hardware than before. If a system goes down, you do not have to rebuild the system from CDs or restore from a backup. You would just load the guest image and get the system running in minutes. From a security perspective, key components can be isolated and contained to reduce or limit exposure.

With the main benefits being ease of use and portability, the uses of virtual machines are almost endless. Starting with the most obvious, virtual machines are great for training. For example, you can bring a single computer and run multiple operating systems that are pre-built by the instructor. Incident response, malicious code analysis, and digital forensics all allow a security professional to have a portable security lab on a single computer with all the tools and software they need at their fingertips.

Virtual machines benefit the individual user and the high-end data center. Virtual machines enable security professionals to access a wide range of operating systems without having to purchase expensive hardware. This enables us to create virtual security labs with minimal effort. Before virtualization, if you wanted an effective lab, you had to buy several computer systems with removable drives. Now, with a few systems and virtual machines, you can simulate an entire corporate network with minimal expense.

Reference
1. What is virtualization? - http://cdm-it.epfl.ch/dnn/Home/tabid/62/entryid/197/language/en-US/virtual-machines-whats-this.aspx

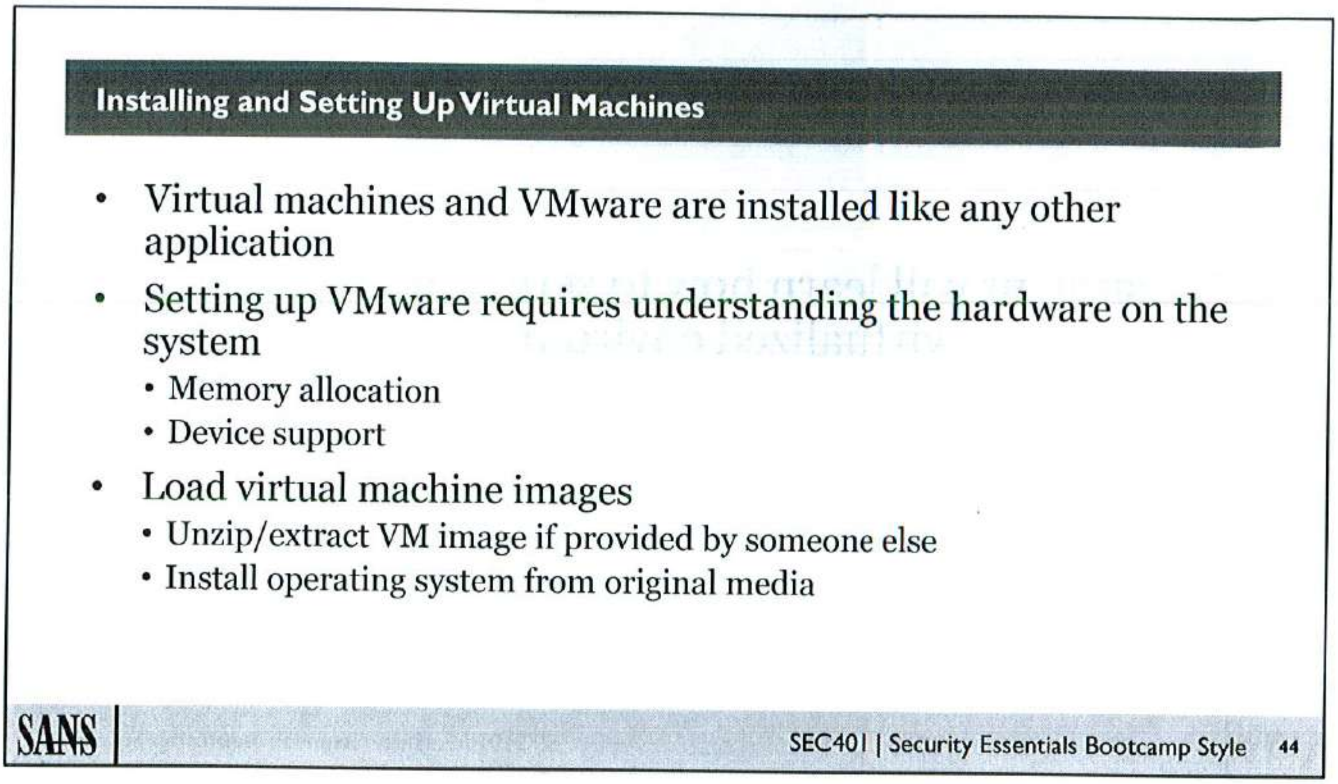


Setting Up Virtualization

The student will learn how to setup and configure a virtualized environment

SANS | SEC401 | Security Essentials Bootcamp Style 43

This page intentionally left blank.



Installing and Setting Up Virtual Machines

- Virtual machines and VMware are installed like any other application
- Setting up VMware requires understanding the hardware on the system
  - Memory allocation
  - Device support
- Load virtual machine images
  - Unzip/extract VM image if provided by someone else
  - Install operating system from original media

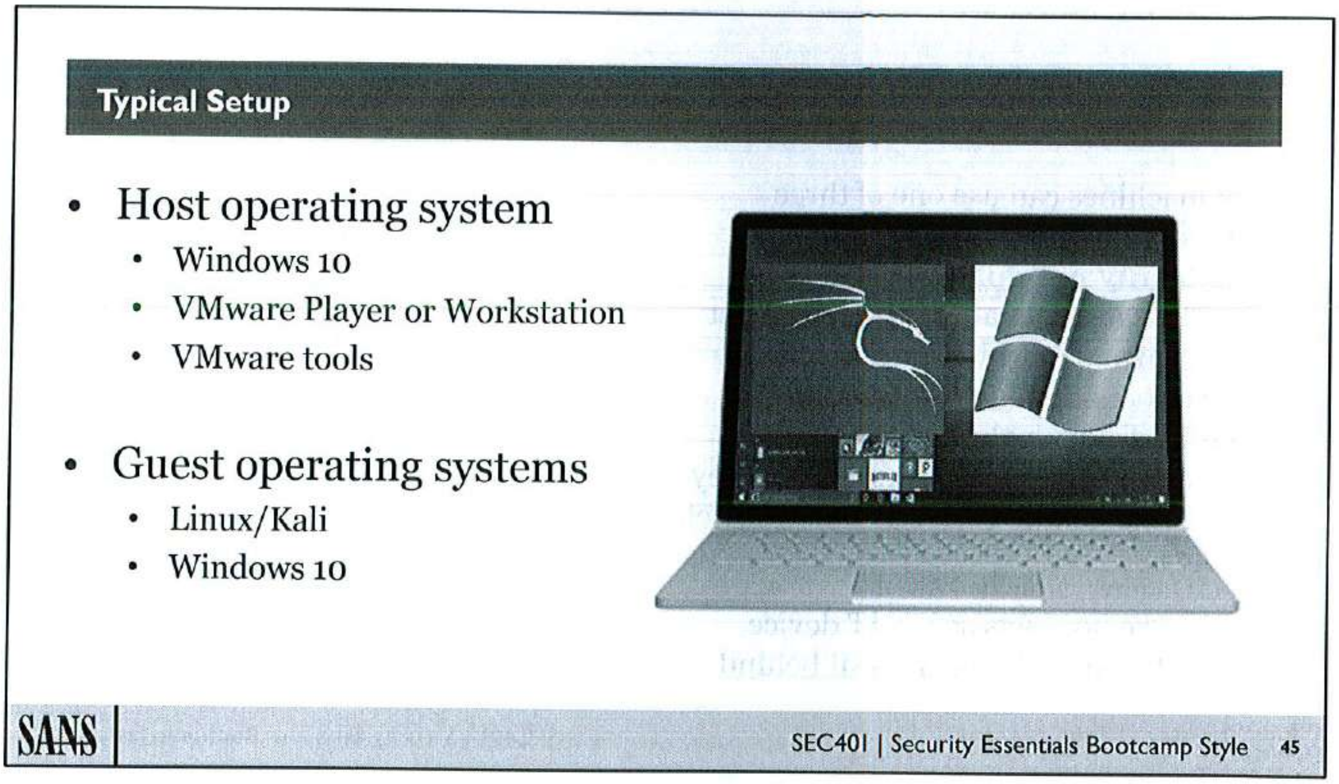
SANS | SEC401 | Security Essentials Bootcamp Style 44

In order to harness the power of virtual machines, two distinct components need to be installed: the virtual machine software and the OS/application images. The virtual machine software (such as VMware) is an application that is installed and responsible for interfacing with the host OS and the hardware to create a virtualized environment that other software can run in. Once this environment is set up, software or applications need to be installed to run in the virtual environment.

VMware Player and Workstation are just like any other application that gets installed on a host operating system. The software has an installation wizard that walks you through the process. Because Player is a scaled-back free version, it has fewer screens and options when installing. Once the VMware software is on your system, you then have to configure it prior to loading software images on the system.

After your VMware software is installed, you need to load images to get to the true power of the software. The images are the actual operating systems or applications that run in a virtual environment and can be created in several ways. First, you can install the software, just like you would a real operating system. For example, put the Windows or Linux CD in your system, and it boots up in the virtual window and installs the software. Or, obtain a pre-configured image either from a removable drive, or, because they are very large, in the form of a zip or ISO image. When this is extracted to a directory, point your VMware software to the directory and it will run.

Installing and setting up virtual machines is a straightforward process. Because it is so easy, it is one of the many reasons virtual machines are so popular. In a matter of minutes, you can have a new operating system running on your box or test out a new application. Instead of having multiple machines or constantly having to rebuild a machine, virtualization allows you to quickly create new development and production environments.



Typical Setup

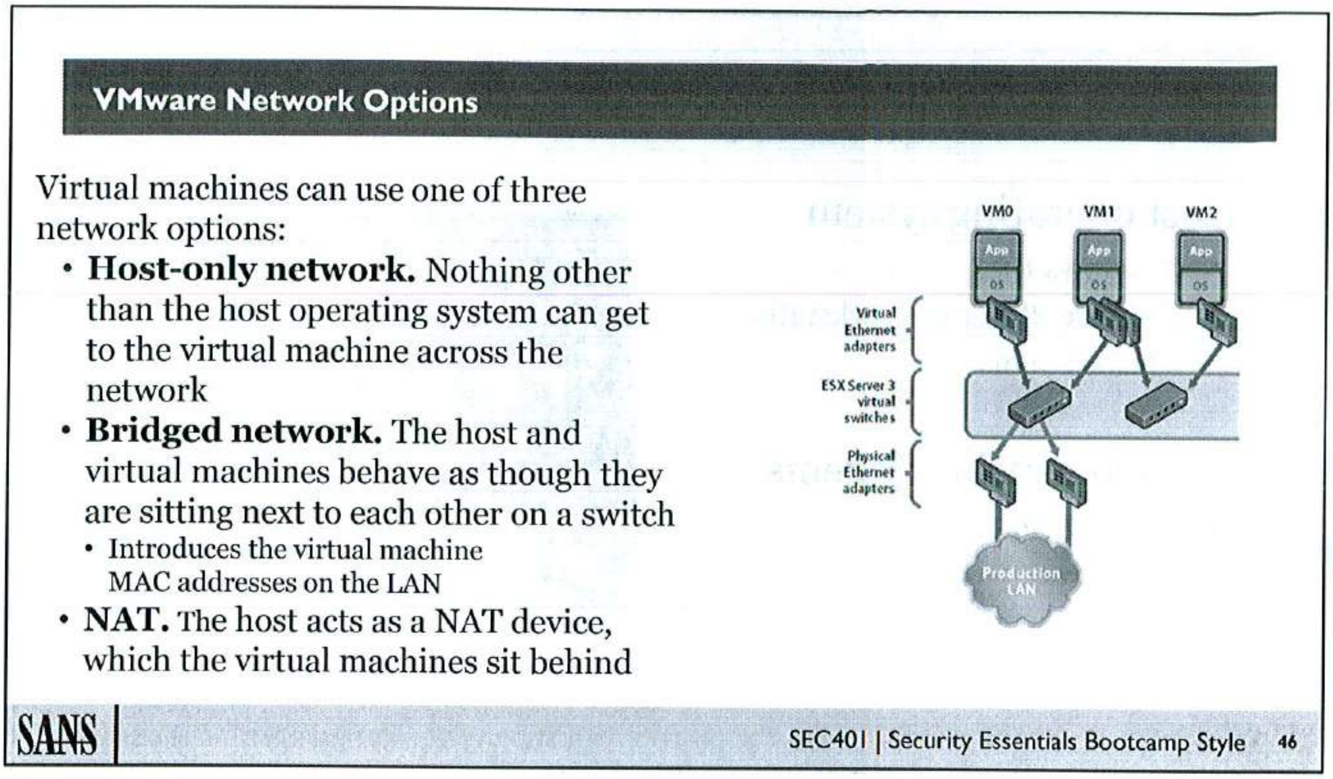
- Host operating system
  - Windows 10
  - VMware Player or Workstation
  - VMware tools

- Guest operating systems
  - Linux/Kali
  - Windows 10

You can utilize and harness the power of virtual machines in many ways, depending on personal preference, the job, or other requirements. However, to baseline the process for this class and other classes, we are going to define a standard VMware setup. We use this setup in this class. If you want to use a different setup, you can, but this one provides a common baseline for this module and other follow-on classes. As a starting point, we use VMware as our virtual machine software.

Most laptops or desktop devices in which you are going to run virtual machines have a Windows operating system installed by default. In addition, many corporations require the use of Windows for desktop systems. Therefore, it is a natural choice to use it as the host operating system. Because you'll often want to add virtual machines to an existing environment, this configuration enables you to simply install a new application on your system without requiring you to reinstall other applications or make changes to existing systems. However, remember that one of the benefits of virtual machines is the capability to easily reload the operating system. If you are running tools on the host operating system and you crash the system, it is much harder to recover than if it was running on a guest operating system.

If you work in security, you need to have access to both Windows and UNIX/Linux operating systems. Some tools run only on Windows, and some tools run only on Linux. Therefore, you need to know how to use both operating systems. The original solution to this was dual-boot systems or bootable CDs, but the main problem is that you can run only one operating system at any given time. Ideally, you would simultaneously run both operating systems. Because Windows is installed on most systems automatically, it makes sense to run a variant of Linux as a guest operating system. In addition, because Linux is free, you can freely distribute Linux images with all the tools installed. The tools can be distributed to people who can foot the guest operating system and access the tools without being required to install or compile programs.



VM0
VM1
VM2
App
OS
App
OS
App
OS
Virtual Ethernet adapters
ESX Server 3 virtual switches
Physical Ethernet adapters
Production LAN

VMware Network Options

Virtual machines can use one of three network options:

- **Host-only network.** Nothing other than the host operating system can get to the virtual machine across the network
- **Bridged network.** The host and virtual machines behave as though they are sitting next to each other on a switch
  - Introduces the virtual machine MAC addresses on the LAN
- **NAT.** The host acts as a NAT device, which the virtual machines sit behind

SANS | SEC401 | Security Essentials Bootcamp Style 46

One of the many benefits of running virtual machines is the ability to connect between different operating systems. For example, you might want to create a virtual lab where the host and guest operating systems can communicate only on the local system, but act as if they are connected through a switch. Or you might want your guest operating system to have its own separate IP address and access the network independently of the host operating system. The good news is that VMware gives you plenty of options for setting up and controlling networking on your system.

Network Options

Virtual machines can use one of three network options:

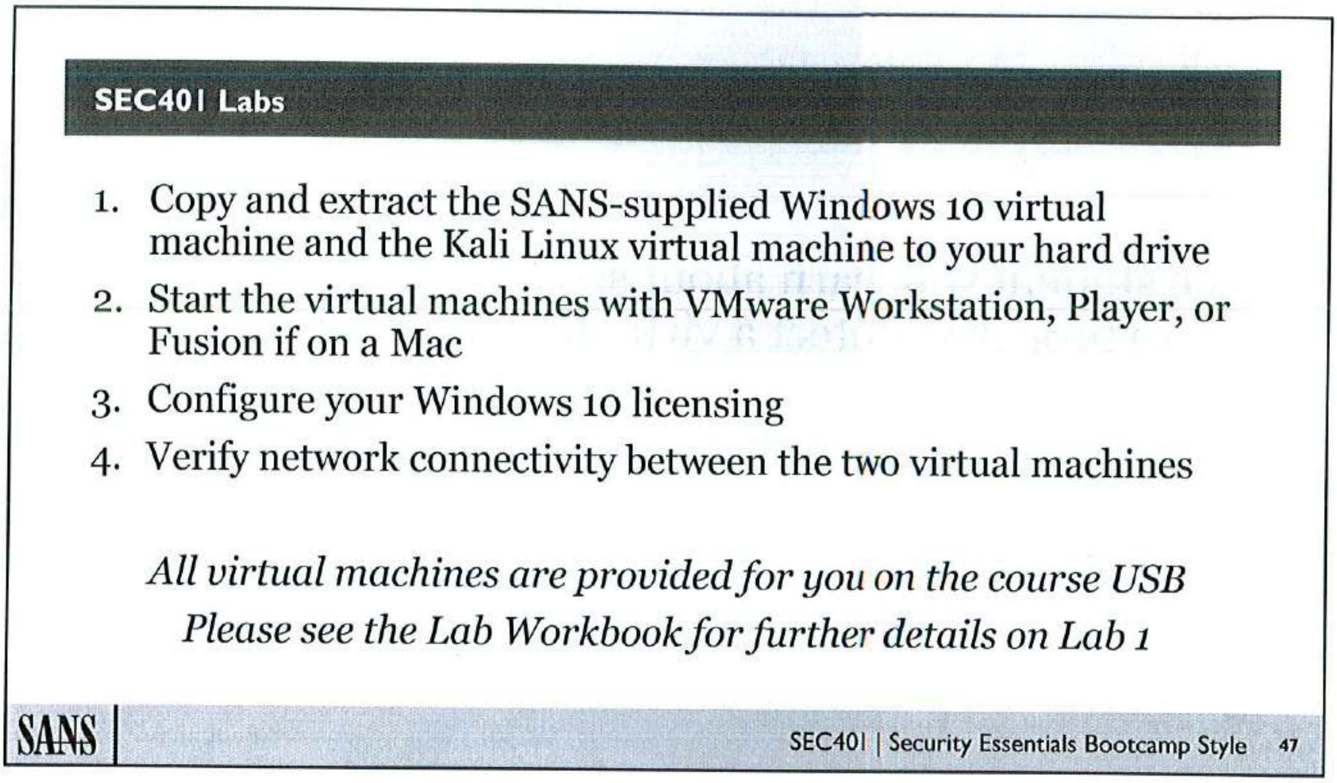
- **Host-only network:** With this option, nothing other than the host operating system can communicate with the virtual machine.
- **Bridged network:** With this configuration, the host and virtual machines behave as though they are sitting next to each other on a switch. This introduces the virtual machine MAC address on the LAN. Also, it puts the host network interface in promiscuous mode. (To capture traffic destined for the virtual machines, the host will have to grab packets destined for MAC addresses that don't match the hardware address.)
- **NAT:** In this mode, the host acts as a NAT device, which the virtual machines sit behind. All packets get their source IP translated so that they appear to have come from the host instead of the guest operating system.

The two most common methods you will use are host-only and bridged networks.

By harnessing the full power of virtual machines, you can run various security tools across different operating systems and test them both locally and across the network. VMware gives you full flexibility for running a range of network options on your local system.

Reference

1. VMWare Brings Networking into the Server - http://www.networkworld.com/article/2350468/cisco-subnet/vmware-brings-networking-into-the-server.html



SEC401 Labs

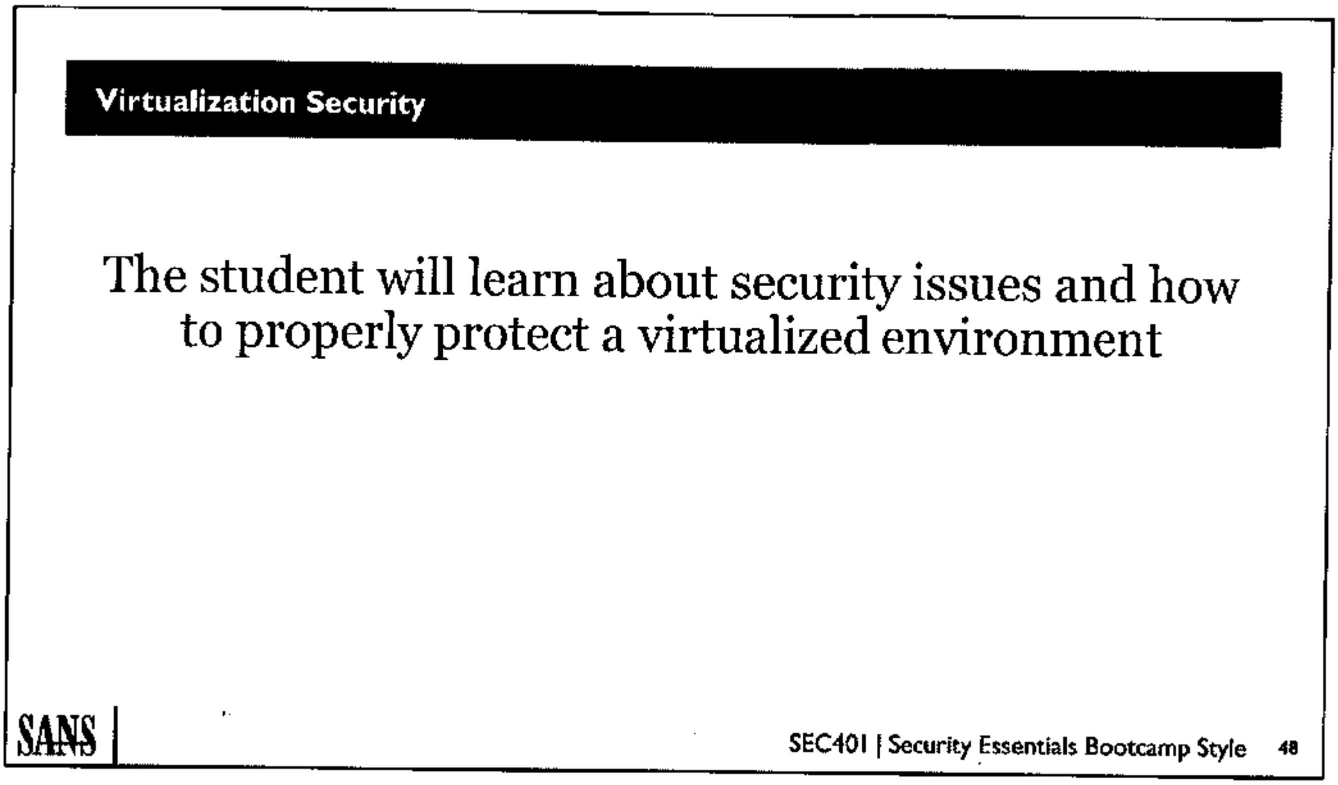
1. Copy and extract the SANS-supplied Windows 10 virtual machine and the Kali Linux virtual machine to your hard drive
2. Start the virtual machines with VMware Workstation, Player, or Fusion if on a Mac
3. Configure your Windows 10 licensing
4. Verify network connectivity between the two virtual machines

All virtual machines are provided for you on the course USB

Please see the Lab Workbook for further details on Lab 1

SANS | SEC401 | Security Essentials Bootcamp Style 47

The SEC401 Student USB contains two virtual machines that you use in this course and outside of the class. The two virtual machines are Windows 10 and the Kali Linux 2.0 distribution. The instructor will explain the process for setting this up in more detail when you start the first lab.

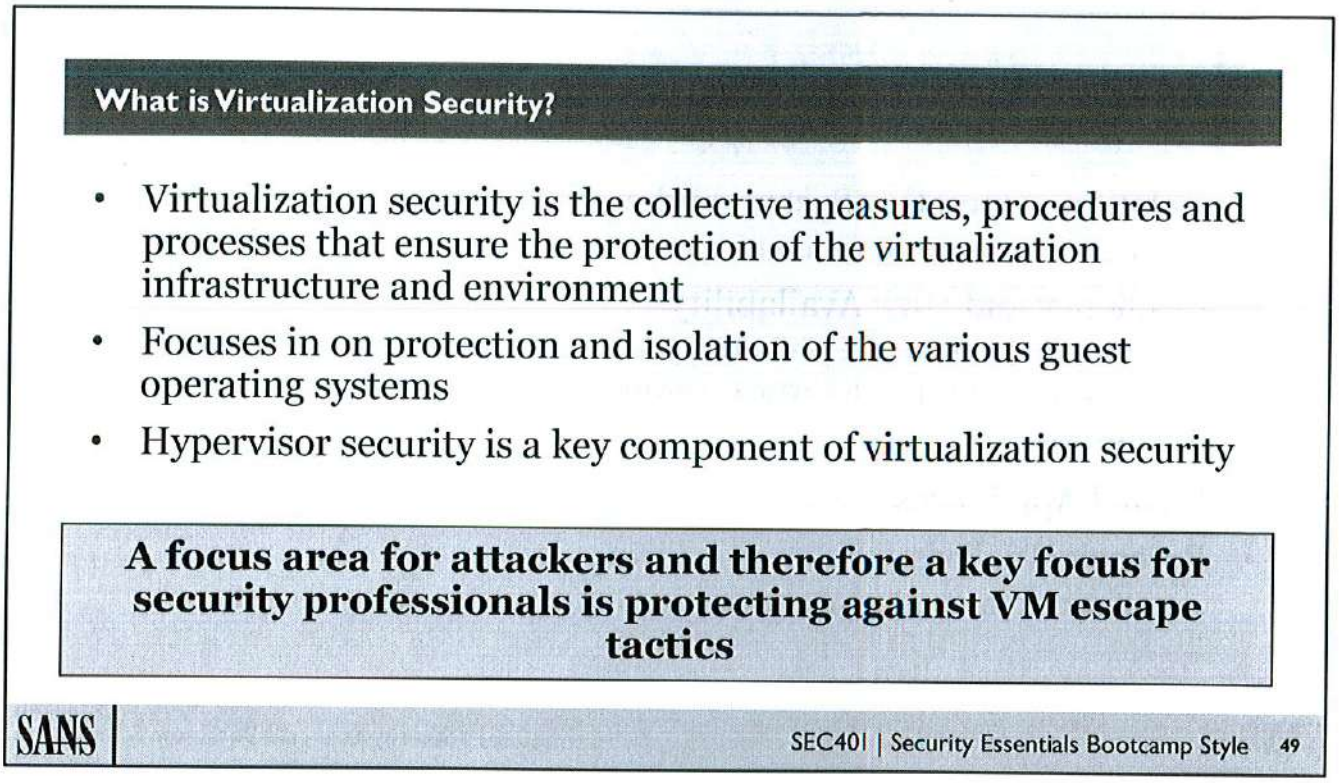


Virtualization Security

The student will learn about security issues and how to properly protect a virtualized environment

SANS | SEC401 | Security Essentials Bootcamp Style 48

This page intentionally left blank.



What is Virtualization Security?

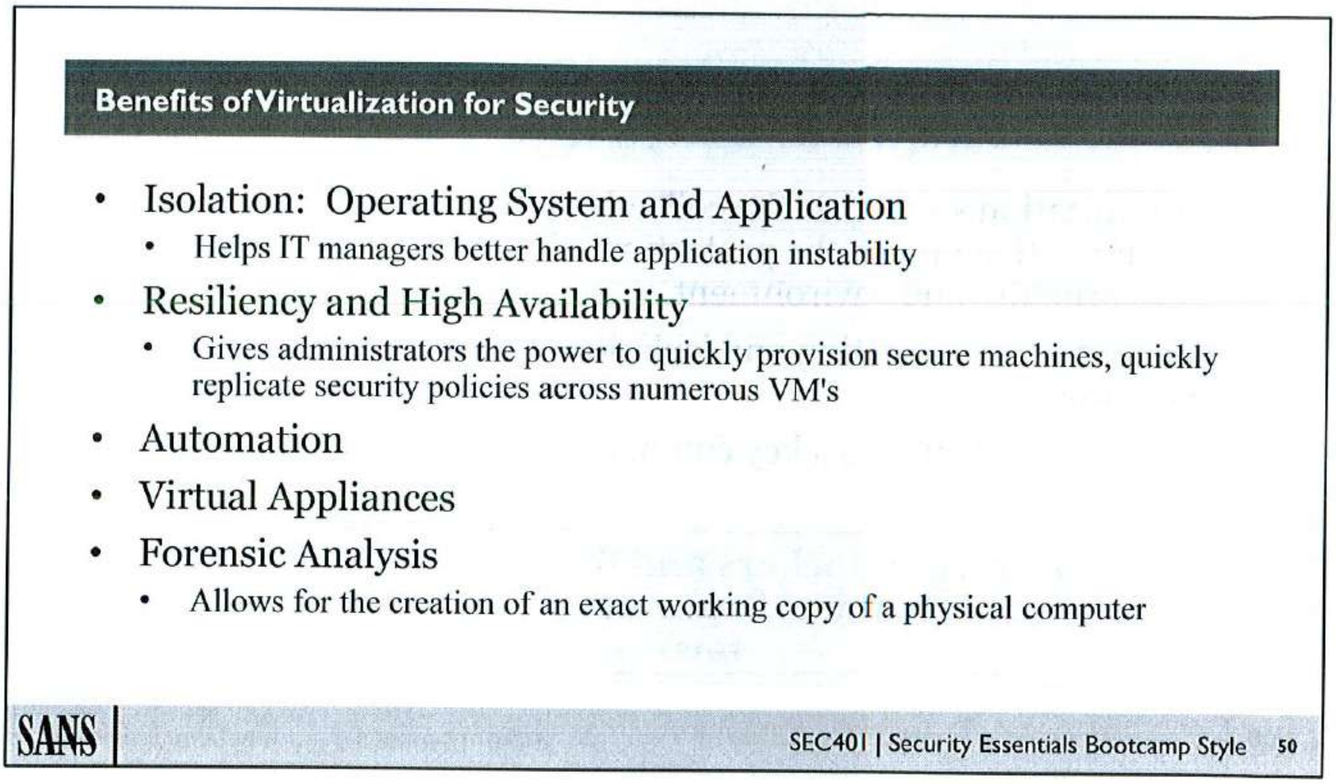
- Virtualization security is the collective measures, procedures and processes that ensure the protection of the virtualization infrastructure and environment
- Focuses in on protection and isolation of the various guest operating systems
- Hypervisor security is a key component of virtualization security

A focus area for attackers and therefore a key focus for security professionals is protecting against VM escape tactics

SANS SEC401 | Security Essentials Bootcamp Style 49

Virtualization security, much like network security or application security, addresses the security issues faced by the components that make up a virtualized environment and the methods by which these security issues can be prevented or minimized. Virtualization security includes processes such as the implementation of security controls and procedures on each virtual machine; securing the virtual machines, the virtual network and any other virtual appliances from the underlying physical device; and the creation and implementation of security policy across the whole virtual environment.

One of the main areas of focus for virtualization security is proper protection and isolation of the various guest operating systems that are running on a host computer. If a server is running a large number of guest operating systems, adversaries would really like to be able to compromise the system once and be able to access all of the other virtual machines. This method is often known as VM escape tactics and is a key area of focus for security professionals.



Benefits of Virtualization for Security

- Isolation: Operating System and Application
  - Helps IT managers better handle application instability
- Resiliency and High Availability
  - Gives administrators the power to quickly provision secure machines, quickly replicate security policies across numerous VM's
- Automation
- Virtual Appliances
- Forensic Analysis
  - Allows for the creation of an exact working copy of a physical computer

SANS | SEC401 | Security Essentials Bootcamp Style 50

Isolation from other operating systems and applications is one of the key benefits of virtualization. A virtual machine runs at a lower level of permissions than the hypervisor they are running on top of which keeps any compromised virtual machine from breaking out into other systems. Additionally, the code for Type 1 hypervisors, also referred to as bare-metal hypervisors, (hypervisors that run directly on the hardware) are kept as small as possible which greatly reduces the attack surface an intruder has to work with.

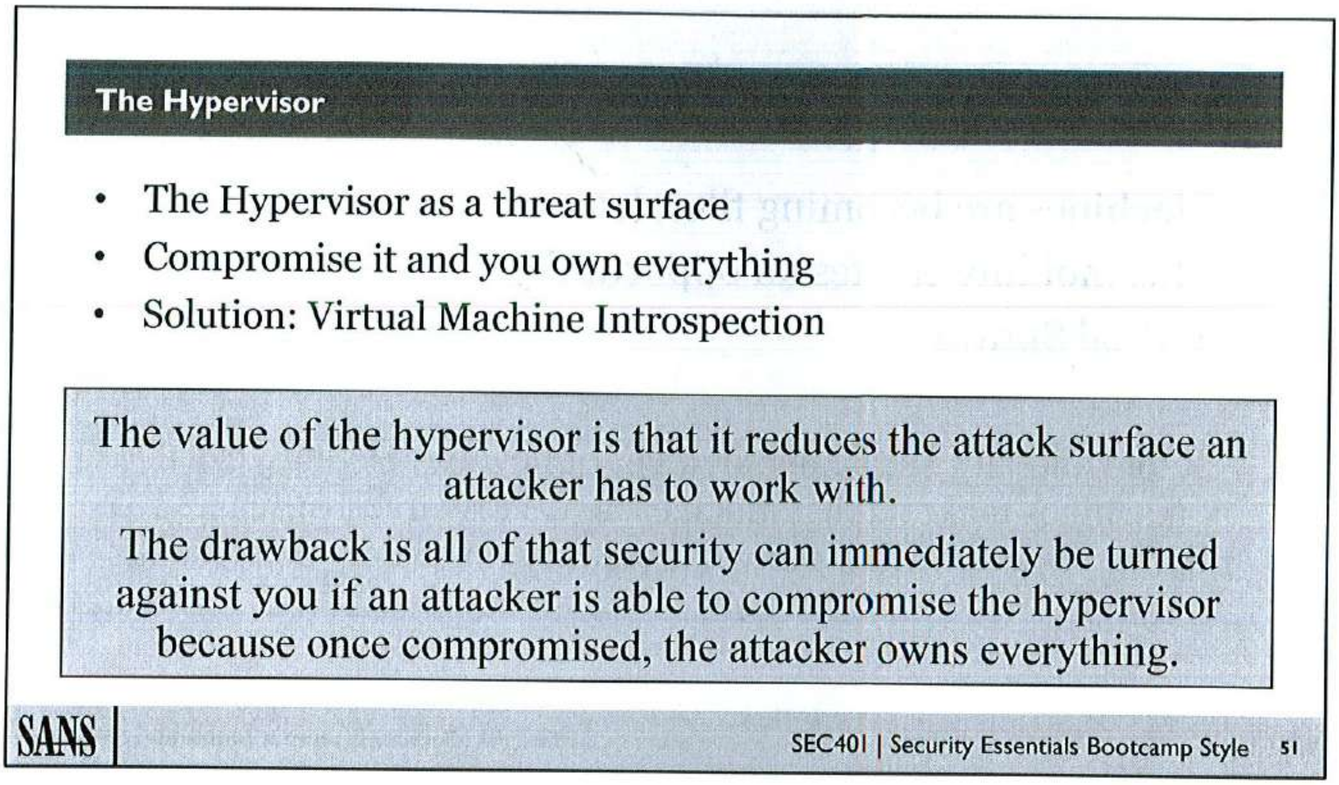
Virtual machines are also well suited for application isolation. This type of isolation helps IT managers better handle application instability, which could waste resources or worse lead to a full system crash; and application compromising, which could lead to local privilege escalation and unauthorized access by an intruder.

Resiliency and high availability is another pillar of virtualization. This quality of virtualization gives administrators the power to quickly provision secure machines, quickly replicate security policies across numerous VM's, automatically set up firewall rulesets for classes of servers and automatically quarantine compromised or out of compliance systems. The resiliency and high availability aspects also lend to sophisticated automation techniques which further reduce configuration and recovery times.

Virtualization greatly reduces the time and costs of disaster recovery operations. Instead of saving thousands of individual files and restoring them from a backup, the whole virtual machine can be copied as a backup even while it is currently running. This backup often appears as a unique large file, which is a tremendous time saver compared with reinstalling an operating system and restoring data files. Often a virtual machine can be restored from backups in a matter of minutes.

Virtual appliances, which are pre-configured virtual machine images, ready to run on a hypervisor, are also very useful for virtualization security. Virtual appliances can be custom built to perform one or two functions very well and nothing else. Depending on the need and the environment a virtual appliance can further reduce the attack surface in a computing environment.

Aside from all the security benefits and better use of resources, virtualization is much easier to manage, which creates flexible systems and lowers overall costs and even electricity bills.



The Hypervisor

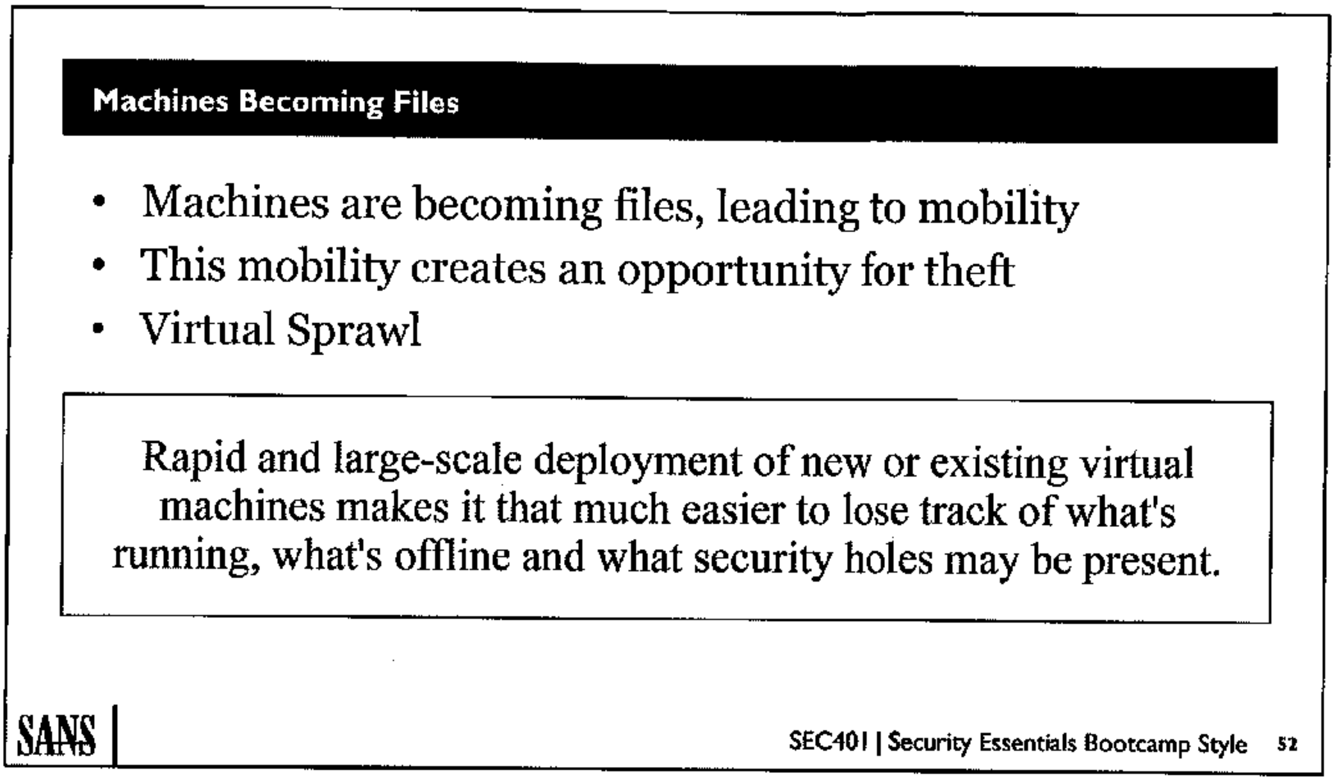
- The Hypervisor as a threat surface
- Compromise it and you own everything
- Solution: Virtual Machine Introspection

The value of the hypervisor is that it reduces the attack surface an attacker has to work with.

The drawback is all of that security can immediately be turned against you if an attacker is able to compromise the hypervisor because once compromised, the attacker owns everything.

Hypervisors are purposely written to be robust and secure. However, like all software, they will inevitably contain vulnerabilities which, if discovered, could be exploited by an attacker. The value of the hypervisor is that it reduces the attack surface an attacker has to work with and that in of itself creates a great deal of security. The drawback is all of that security can immediately be turned against you if an attacker is able to compromise the hypervisor because once compromised, the attacker owns everything.

Fortunately, there is a security solution for the hypervisor and it is called virtual machine introspection. As the name suggests, virtual machine introspection allows an administrator to monitor all the events within a virtual environment so any unusual behavior can be caught early. VM introspection is implemented in the hypervisor and it monitors the state of the hypervisor and all the virtual machines running on the physical server. VM introspections are specifically designed to have minimal effects on both the hypervisor and any virtual machines it is looking at. Introspection can occur in different ways. Some examples include doing a memory introspection or a system events introspection, looking at system calls, interrupts and I/O device events, as well as live processes. Virtual machine introspection can also detect malware actions that are designed to avoid antimalware detection since the virtual operating systems behavior is being observed from the outside. Since introspection functions execute in the hypervisor, it is possible to trace and monitor every interaction between a guest operating system or application and the underlying hardware making it a powerful tool for virtualization security.



Machines Becoming Files

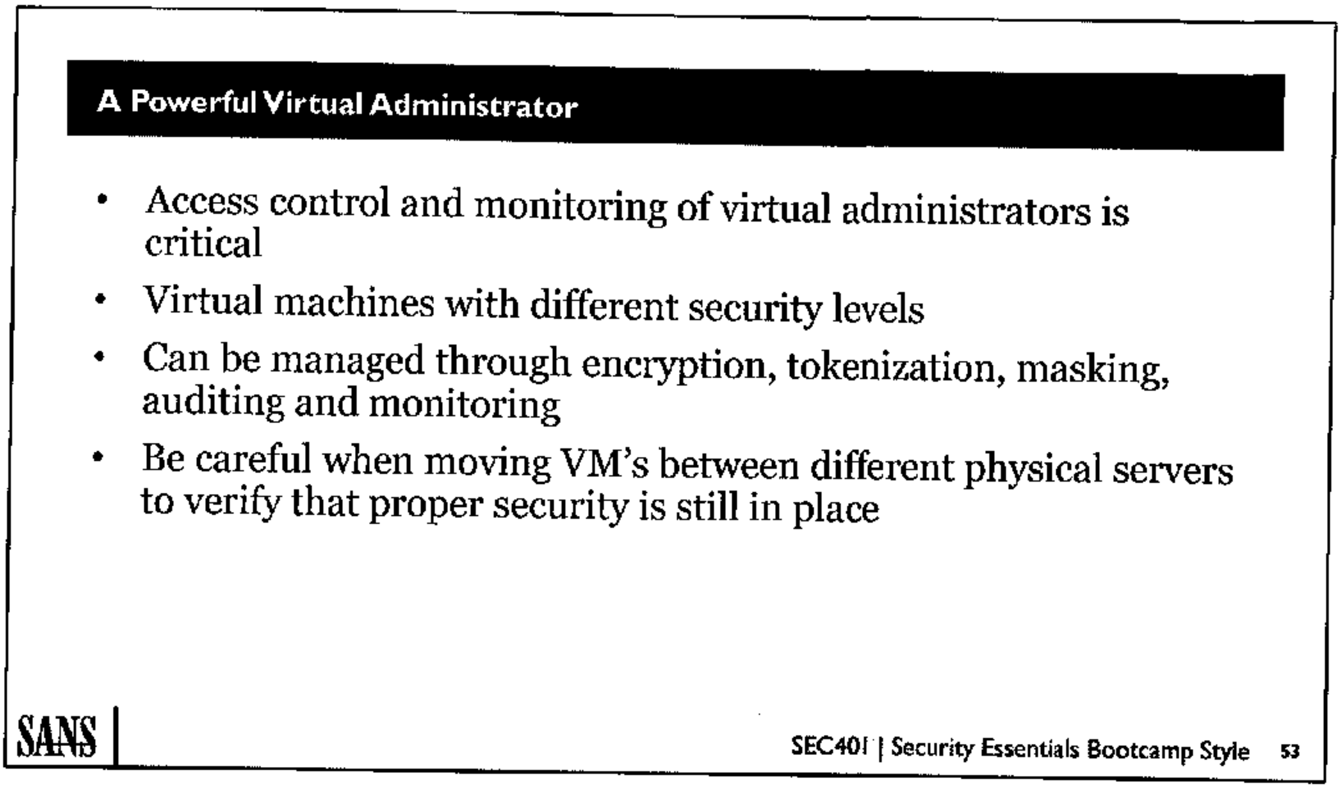
- Machines are becoming files, leading to mobility
- This mobility creates an opportunity for theft
- Virtual Sprawl

Rapid and large-scale deployment of new or existing virtual machines makes it that much easier to lose track of what's running, what's offline and what security holes may be present.

Virtualized environments are dynamic by design and often rapidly change on a regular basis. Unlike physical environments, dozens, even hundreds of virtual machines can be powered up in a matter of minutes. This type of rapid and large-scale deployment of new or existing virtual machines makes it that much easier to lose track of what's running, what's offline and what security holes may be present.

This is called virtual sprawl which refers to the condition in an operating environment where the number of virtual machines in existence reaches a point where they can no longer be effectively managed or secured. In this situation, the security of all the virtual machines can no longer be guaranteed. Attackers will use the virtual sprawl as a cover to locate a virtual machine that is offline, using it to gain access to an organization's systems.

Fortunately, virtual sprawl can be managed, to a degree, with the right policies and automation techniques in place but like any system, either physical or virtual, overwhelm can occur if it is allowed to happen through mismanagement or understaffing.



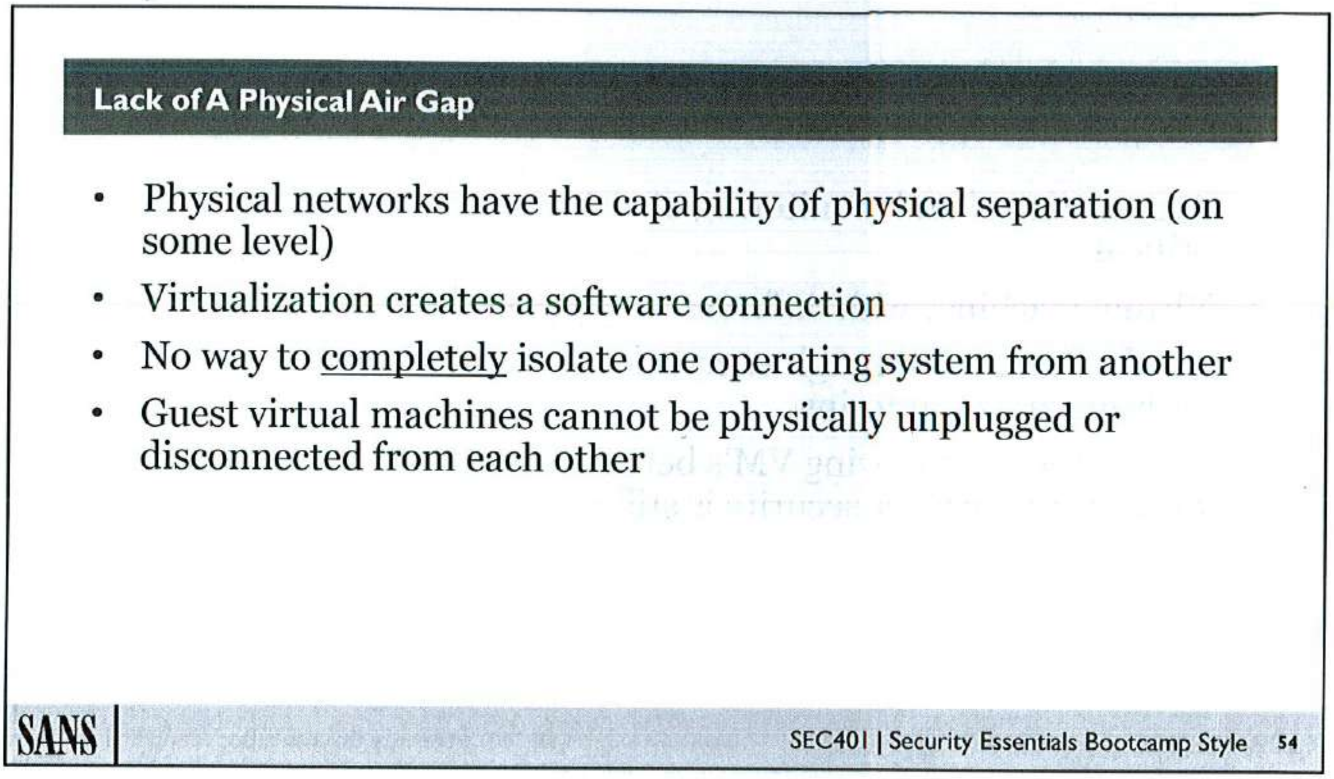
A Powerful Virtual Administrator

- Access control and monitoring of virtual administrators is critical
- Virtual machines with different security levels
- Can be managed through encryption, tokenization, masking, auditing and monitoring
- Be careful when moving VM's between different physical servers to verify that proper security is still in place

Access control and monitoring of virtual administrators are critical to ensuring data is secure. It is becoming more and more common to run background checks on virtual and cloud administrators as well having significant physical monitoring in place such as card keys for entry to the data center, cameras, and even direct monitoring by security personnel.

On the system itself, there are a number of ways to address administrator security such as through encryption, tokenization, masking, auditing, and monitoring. Regardless of the final solution and in all of these cases, it is critical to make sure the solution cannot be easily defeated, even by privileged users, and will work well in the distributed environment.

Along with the dynamic nature of virtual machines, workloads themselves can be moved just as quickly. This poses a security risk. To elaborate, a certain workload may need a high level of security, and the initial virtual machine the workload is assigned to may provide that security. But when faced with the need to make room for more mission-critical workloads, without proper checks and balances in place, it could easily be moved to a new virtual machine with lower level security, thereby creating a potential security risk.



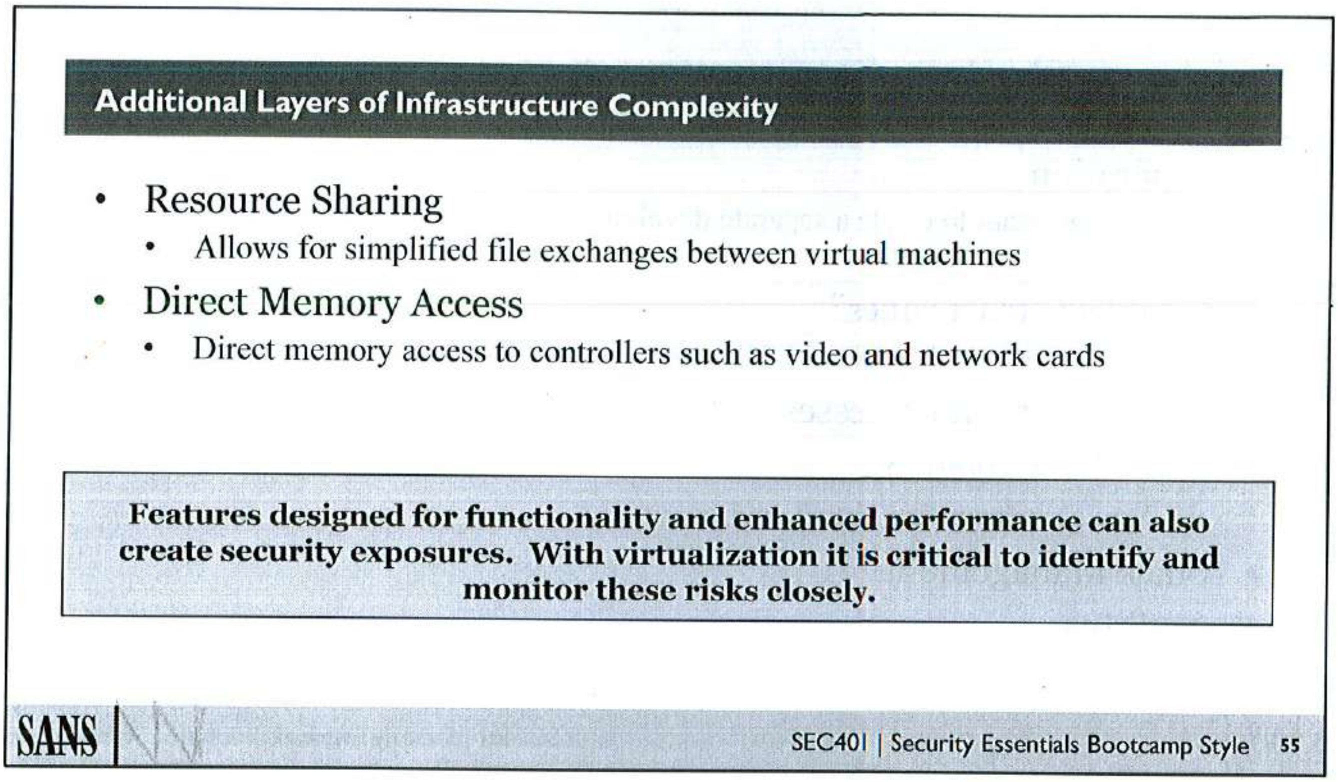
Lack of A Physical Air Gap

- Physical networks have the capability of physical separation (on some level)
- Virtualization creates a software connection
- No way to completely isolate one operating system from another
- Guest virtual machines cannot be physically unplugged or disconnected from each other

SANS | SEC401 | Security Essentials Bootcamp Style 54

In traditional, physical networks there was always some type of air gap that existed, or could be made to exist by unplugging a cable, between systems on a network. As an example, two computers that are connected on the same Ethernet LAN can only communicate with each other via that network. And if that network cable is disconnected or a firewall is put in between those two computers they won't be able to communicate with each other.

In a virtualized environment, the hypervisor always creates a software connection between systems, not a physical one. And because it is software based, there is no way to completely isolate one operating system from another. The only way to ensure physical separation in a virtual environment is to migrate one of the operating system virtual machines to a different hardware platform. It is this persistent software connection that leads network security administrators to feel that virtualization can never be configured as securely as a legacy network.



Additional Layers of Infrastructure Complexity

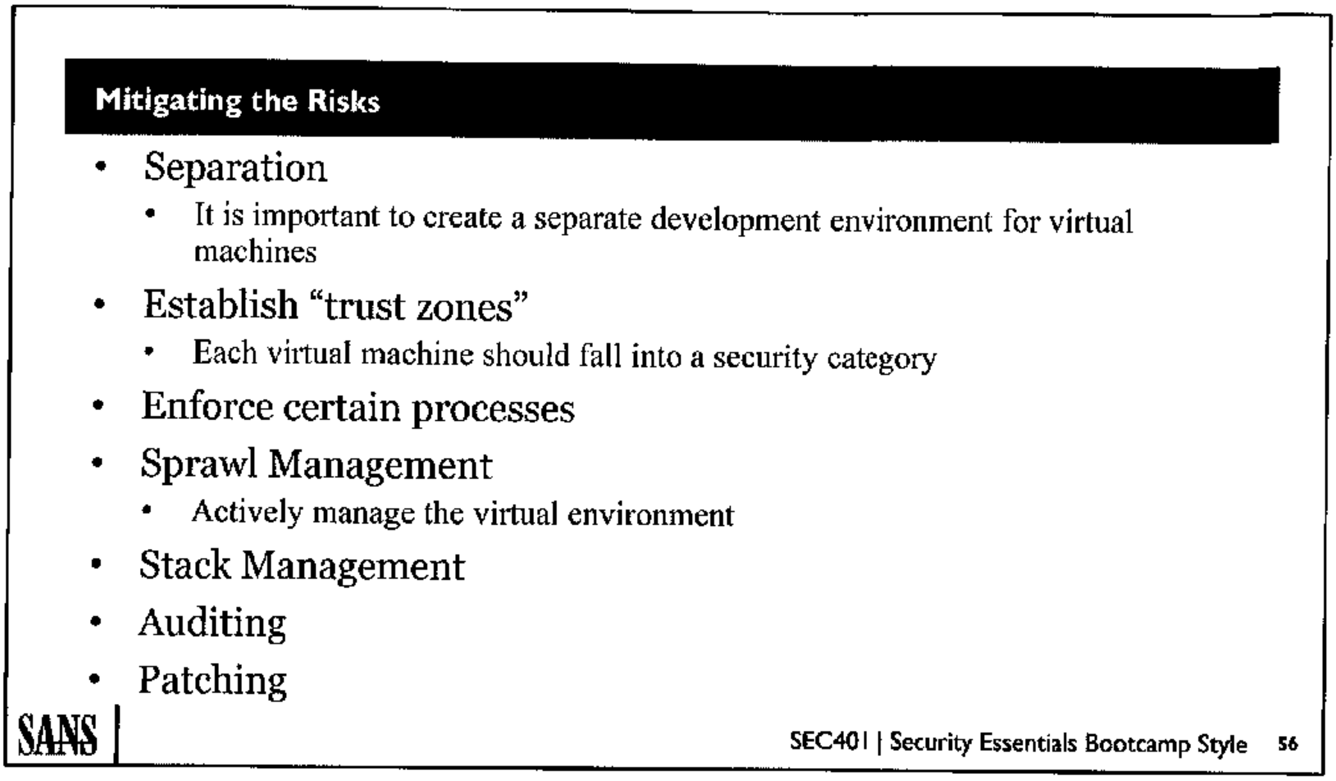
- Resource Sharing
  - Allows for simplified file exchanges between virtual machines
- Direct Memory Access
  - Direct memory access to controllers such as video and network cards

Features designed for functionality and enhanced performance can also create security exposures. With virtualization it is critical to identify and monitor these risks closely.

With the additional complexity that can arise in a virtualized environment the need for monitoring for events and anomalies also becomes more complex. This added complexity can make it more difficult to identify already challenging security issues such as advanced persistent threats. Some specific complexities to be aware of and manage are resource sharing and direct memory access.

Resource and storage sharing allows for simplified file exchanges between virtual machines and virtual machines and their host operating systems. This flexibility introduces a measure of increased risk as well. Features like clipboard sharing can increase functionality, but can also open up similar security holes.

Virtual machines also have direct memory access to controllers such as video and network cards. This access allows the virtual machine and the hypervisor to create better performance but also increase the risks. Direct Memory Access or DMA has the potential to allow an attacker to use the storage on a peripheral device in order to move code off of the virtual machine.



Mitigating the Risks

- Separation
  - It is important to create a separate development environment for virtual machines
- Establish "trust zones"
  - Each virtual machine should fall into a security category
- Enforce certain processes
- Sprawl Management
  - Actively manage the virtual environment
- Stack Management
- Auditing
- Patching

SANS | SEC401 | Security Essentials Bootcamp Style 56

Despite the benefits of virtualization, there are, and always should be, concerns about the security risks associated with virtualization. Fortunately, the risks are manageable. To conclude this section we'll cover a list of tactics that, if followed, will help mitigate potential threats to virtual environments without the need for burdensome, expensive processes and solutions.
Separation: It is important to create a separate development environment for virtual machines. This allows you test virtual machines safely before they are put into production on the live network.

Establish "Trust Zones" for different deployed environments. This can be likened to security classifications or levels of sensitivity. Each virtual machine should fall into a security category regardless of its function and this is where it should reside on the network with the appropriate security controls in place for the appropriate level of security.

Process enforcement: Enable IT-specific virtualization processes to increase efficiency and simplify management.

Sprawl management: Actively manage the virtual environment. Know what is being used, what's needed and what's not.

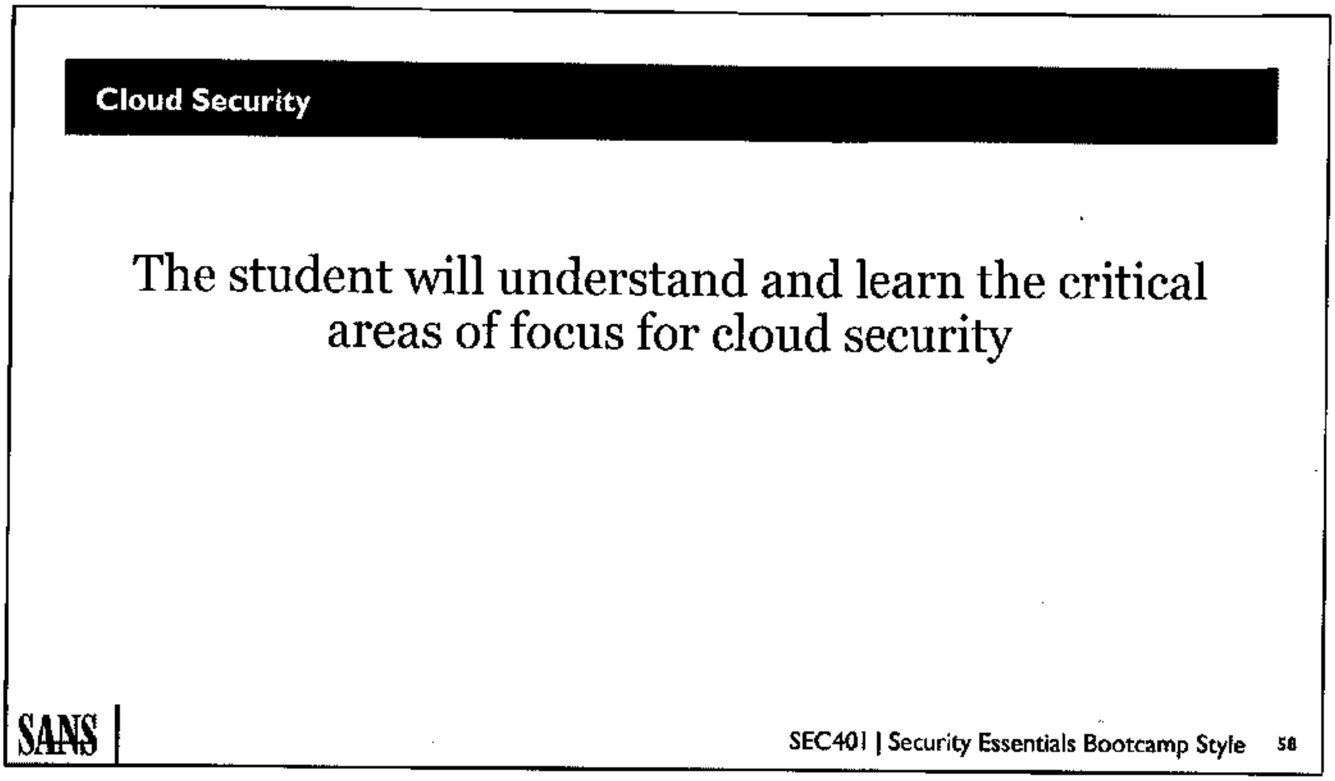
Complete stack management: Focus on end-to-end connections within the virtual environment to ensure there are no gaps.

Built-in auditing: Leverage tools to automate security checks. Making auditing a part of your overall infrastructure will simply the review process.

Patching: Implement a patch maintenance and management process and schedule to make sure patches are up-to-date for both online and offline virtual machines.

References

1. http://www.catbird.com/vsecurity/best-practices
2. http://www.datacenterknowledge.com/archives/2015/03/09/virtualization-security-overcoming-risks/
3. https://www.techopedia.com/definition/30243/virtualization-security
4. http://searchservervirtualization.techtarget.com/tip/Step-by-step-virtualization-Using-virtualization-to-improve-security-part-1
5. http://www.computerweekly.com/feature/Hypervisor-security-New-techniques-for-securing-virtual-machines
6. http://searchcloudsecurity.techtarget.com/answer/Can-virtual-machine-introspection-improve-cloud-security
7. https://cloudsecurityalliance.org
8. https://blog.cloudsecurityalliance.org/2011/03/21/three-cloud-computing-data-security-risks-that-can%E2%80%99t-be-overlooked/

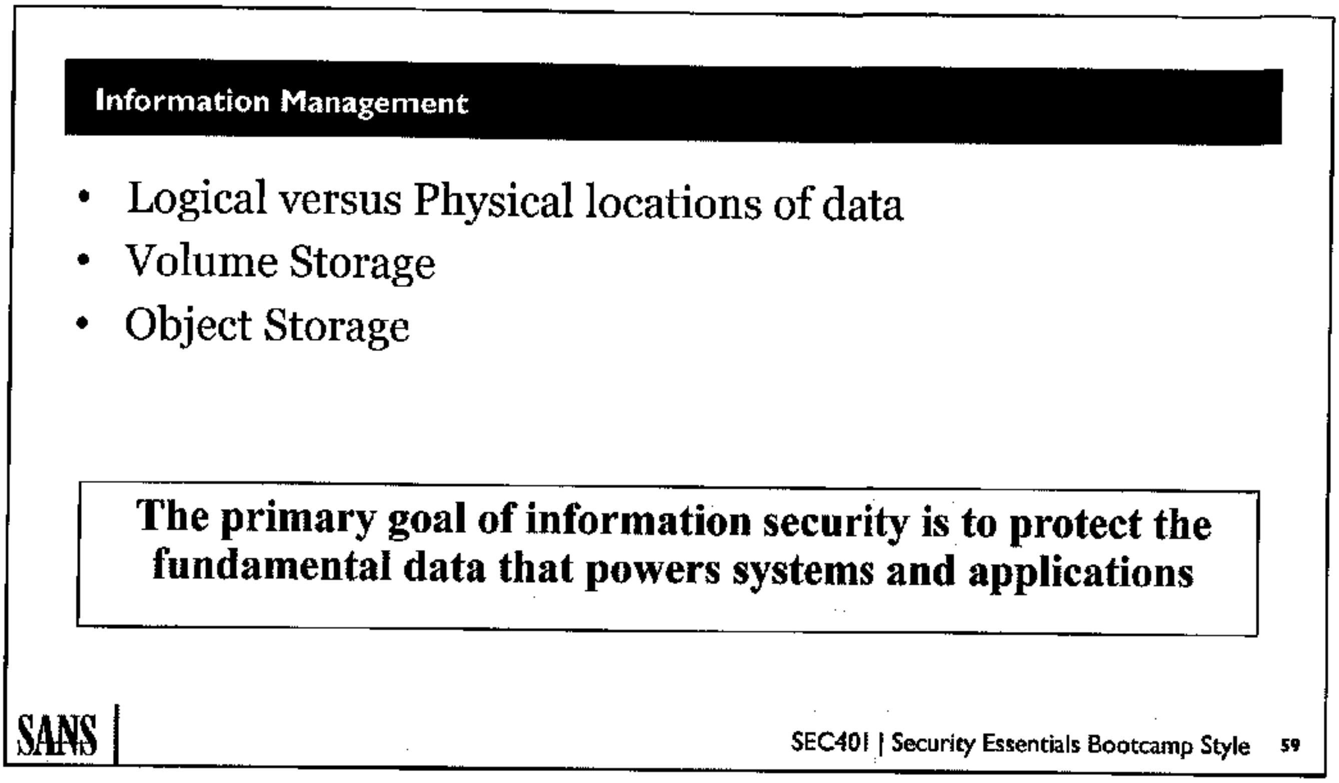


Cloud Security

The student will understand and learn the critical areas of focus for cloud security

SANS | SEC401 | Security Essentials Bootcamp Style 58

This page intentionally left blank.



Information Management

- Logical versus Physical locations of data
- Volume Storage
- Object Storage

The primary goal of information security is to protect the fundamental data that powers systems and applications

SANS

SEC401 | Security Essentials Bootcamp Style 59

The primary goal of information security is to protect the fundamental data that powers systems and applications.

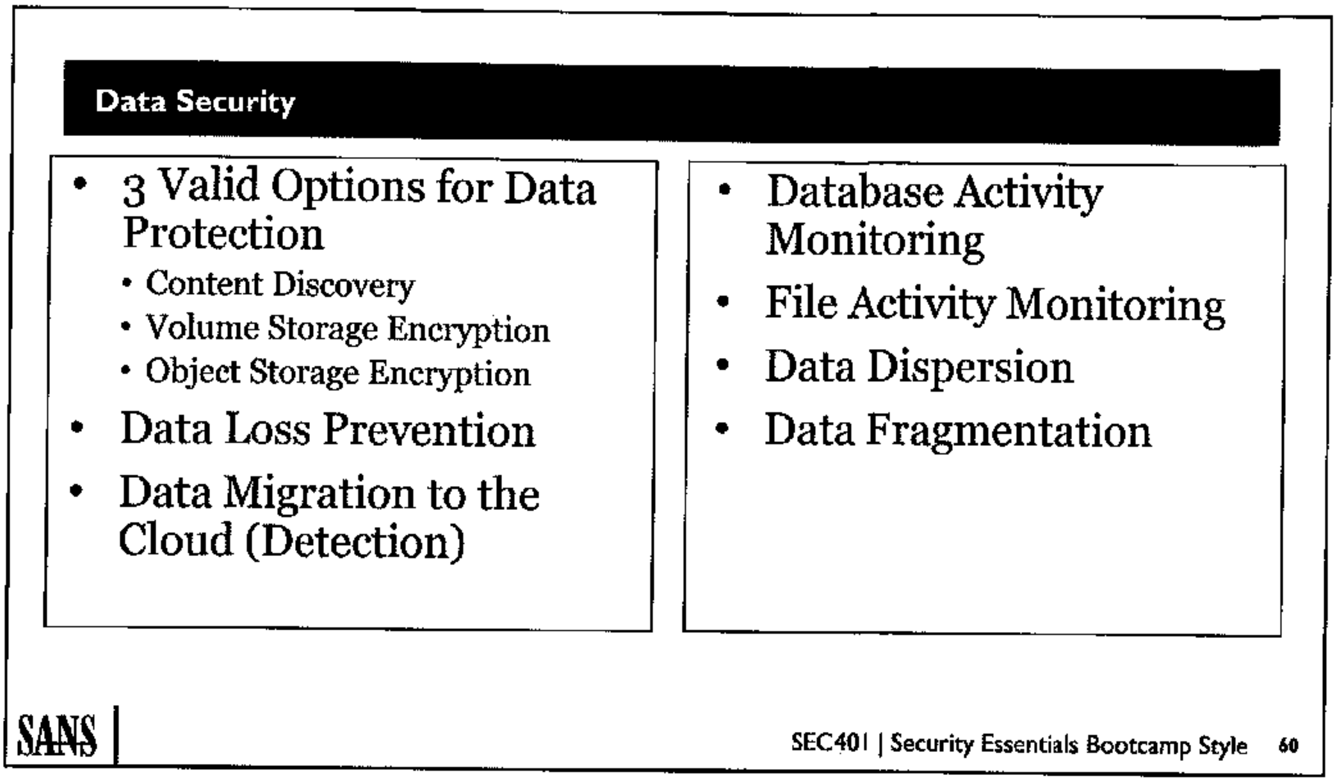
As companies transition to cloud computing, the traditional methods of securing data are challenged by cloud-based architectures. Managing information in the era of cloud computing begins with managing internal data and cloud migrations. This extends to securing information in cross-organization applications and services.

In short, Information management and data security in the cloud demands both new strategies and technical architectures.

Logical vs. physical locations of data
This can be illustrated by thinking of the lifecycle not as a single, linear operation, but as a series of smaller lifecycles running in different operating environments—sometimes physical, sometimes logical. At nearly any phase data can move into, out of, and between these environments.
Where data may be geographically located has important legal and regulatory ramifications. Due to all the potential regulatory, contractual, and other jurisdictional issues, it is extremely important to understand both the logical and physical locations of data.

Volume storage – This includes volumes attached to IaaS instances, typically as a virtual hard drive. Volumes often use data dispersion to support resiliency and security.

Object storage – Object storage is sometimes referred to as file storage. Rather than a virtual hard drive, object storage is more like a file share accessed via API's or web interface. DropBox is a good example of this.



Data Security

- 3 Valid Options for Data Protection
  - Content Discovery
  - Volume Storage Encryption
  - Object Storage Encryption
- Data Loss Prevention
- Data Migration to the Cloud (Detection)

- Database Activity Monitoring
- File Activity Monitoring
- Data Dispersion
- Data Fragmentation

SANS | SEC401 | Security Essentials Bootcamp Style 60

Three valid options for protecting data:

Content Discovery - Content discovery includes the tools and processes to identify sensitive information in storage. It allows the organization to define policies based on information type, structure, or classification and then scans stored data using advanced content analysis techniques to identify locations and policy violations. Content discovery may also be available as a managed service.
Volume Storage Encryption - it protects volumes from snapshot cloning and exposure – it protects volumes from being explored by the cloud provider – and it protects volumes from being exposed by physical loss of drives.
Object Storage Encryption - Object storage encryption protects from many of the same risks as volume storage. Since object storage is more often exposed to public networks, it also allows the user to implement Virtual Private Storage.

Like a VPN, a VPS allows use of a public shared infrastructure while still protecting data, since only those with the encryption keys can read the data even if it is otherwise exposed. Some types of object storage encryption are:

- File/Folder encryption - Use standard file/folder encryption to encrypt the data before placing in object storage.
- Client/Application encryption - When object storage is used as the back-end for an application (including mobile applications), encrypt the data using an encryption engine embedded in the application or client.
- Proxy encryption - Data passes through an encryption proxy before being sent to object storage.

Data Loss Prevention
Is defined as Products that, based on central policies, identify, monitor, and protect data at rest, in motion, and in use, through deep content analysis.
DLP can provide options for how data found to be in violation of policy is to be handled. Some ways in which data loss prevention can be handled is 1) Data can be blocked, basically stopping a workflow or 2) it can be allowed to proceed after the data has been appropriately encrypted.

Data Migration to the Cloud (Detection)

A common challenge organizations face with the cloud is managing data. Aside from traditional data security controls such as access controls or encryption, there are two other steps to help manage unapproved data moving to cloud services:

The first is to monitor for large internal data migrations with Database Activity Monitoring and File Activity Monitoring, seeing as before data can move to the cloud it needs to be pulled from its existing repository. Database Activity Monitoring can detect when an administrator or other user pulls a large data set or replicates a database, which could indicate a migration. File Activity Monitoring provides similar protection for file repositories, such as file shares.

The second is to monitor for data moving to the cloud with URL filters and Data Loss Prevention tools. URL filtering allows you to monitor (and prevent) users connecting to cloud services. DLP tools look at the actual data/content being transmitted, not just the destination. Thus the user can generate alerts (or block) based on the classification of the data. For example, the user can allow corporate private data to go to an approved cloud service but block the same content from migrating to an unapproved service.

Database Activity Monitoring is defined as: Database Activity Monitors capture and record, at a minimum, all Structured Query Language (SQL) activity in real-time or near real-time, including database administrator activity, across multiple database platforms; and can generate alerts on policy violations. Database Activity Monitoring supports near real-time monitoring of database activity and alerts based on policy violations, such as SQL injection attacks or an administrator replicating the database without approval. DAM tools for cloud environments are typically agent-based connecting to a central collection server

File Activity Monitoring is defined as: Products that monitor and record all activity within designated file repositories at the user level, and generate alerts on policy violations. File Activity Monitoring for cloud requires use of an endpoint agent or placing a physical appliance between the cloud storage and the cloud consumers.

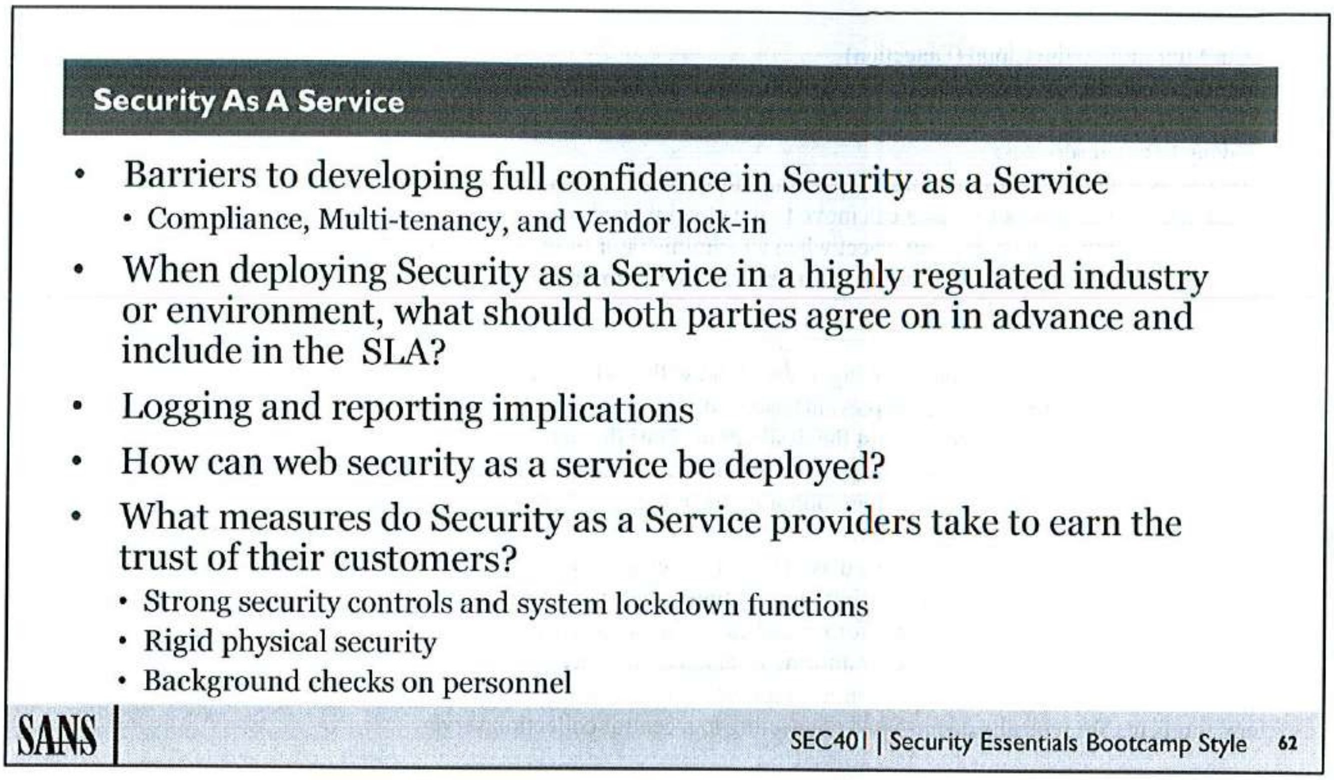
Data Dispersion

Data (Information) Dispersion is a technique that is commonly used to improve data security but without the use of encryption mechanisms. These sorts of algorithms are capable of providing high availability and assurance for data stored in the cloud, by means of data fragmentation, and are common in many cloud platforms.

Data Fragmentation

In a fragmentation scheme, a file is split into a specific number fragments; all of these are signed and then distributed to number remote servers. The user then can reconstruct the file by accessing a certain number of arbitrarily chosen fragments. The fragmentation mechanism can also be used for storing long-lived data in the cloud with high assurance.

When fragmentation is used along with encryption, data security is enhanced since an adversary has to compromise a number of cloud nodes in order to retrieve enough of the fragments of the file and then has to break the encryption mechanism being used.



Security As A Service

- Barriers to developing full confidence in Security as a Service
  - Compliance, Multi-tenancy, and Vendor lock-in
- When deploying Security as a Service in a highly regulated industry or environment, what should both parties agree on in advance and include in the SLA?
- Logging and reporting implications
- How can web security as a service be deployed?
- What measures do Security as a Service providers take to earn the trust of their customers?
  - Strong security controls and system lockdown functions
  - Rigid physical security
  - Background checks on personnel

Some security concerns are around compliance, multi-tenancy, and vendor lock-in. While these are being cited as inhibitors to the migration of security into the cloud, these same concerns exist with traditional data centers. Compliance has been raised as a concern given the global regulatory environment. Security as Service providers should be cognizant of the geographical and regional regulations that affect the services and their consumers, and this can be built into the offerings and service implementations.

The most prudent Security as a Service providers often enlist mediation and legal services to preemptively resolve the regulatory needs of the consumer with the regional regulatory requirements of a jurisdiction. As with any cloud service, multi-tenancy presents concerns of data leakage between virtual instances. Security as a Service providers should take significant precautions to ensure data is highly compartmentalized and any data that is shared is anonymized to protect the identity and source.

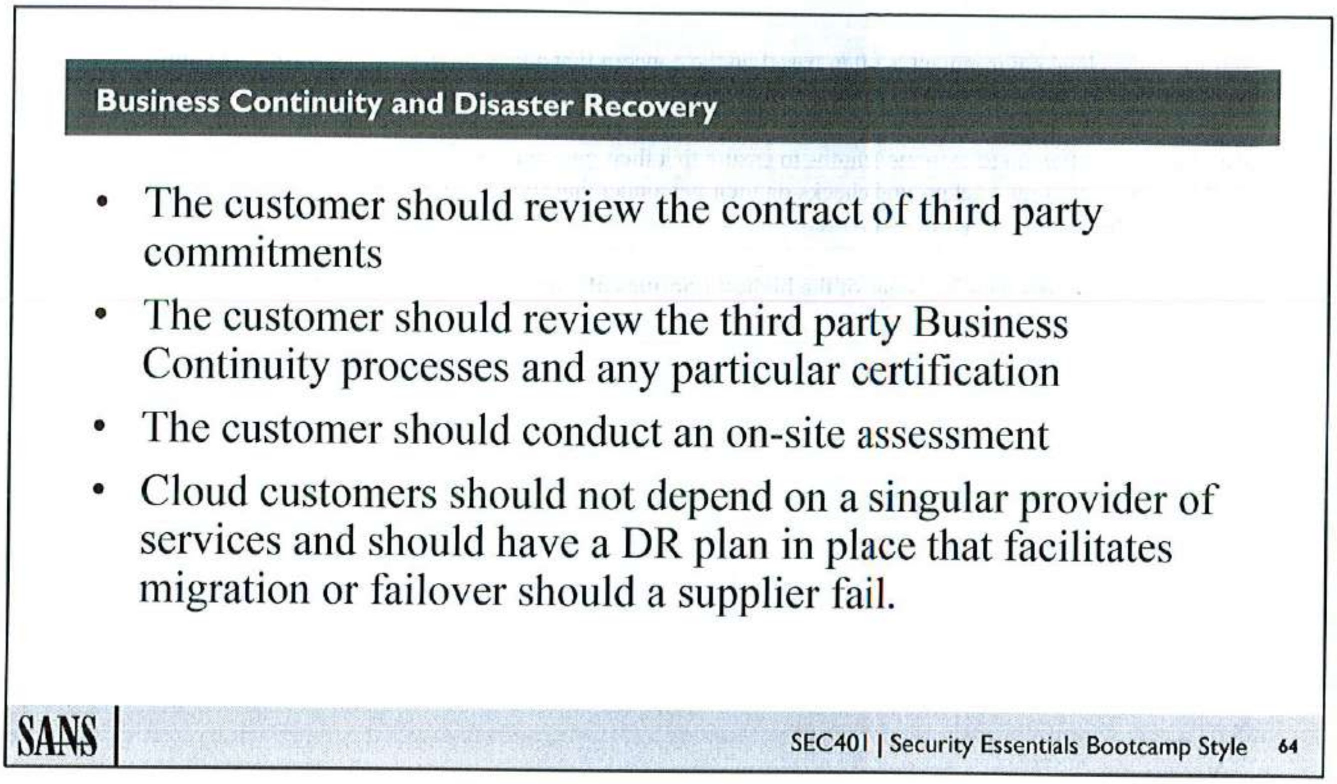
When utilizing a Security as a Service the vendor might have proprietary standards. In the event the customer seeks a new provider, they must concern themselves with an orderly transition and somehow find a way for the existing data and log files to be translated correctly and in a forensically sound manner. It is important to note that other than multi-tenancy, each of these concerns are not "cloud unique" but are problems faced by both in-house models and outsourcing models. When deploying Security as a Service in a highly regulated industry or environment, agreement on the metrics defining the service level required to achieve regulatory objectives should be negotiated in parallel with the SLA documents defining service.

When utilizing a Security as a Service vendor many or all security logging, compliance, and reporting into the custody of a provider might sometimes have proprietary standards and customers must concern themselves with an orderly and forensically sound transition if necessary.

Web Security as a service is a protective, detective, and reactive technical control. When deployed it offers real-time protection either on premise through software / appliance installation or via the Cloud by proxying or redirecting web traffic to the cloud provider.

Security in the cloud environment is often based on the concern that a lack of visibility into security controls means systems are not locked down as well as that they are in traditional data centers and the personnel lacks the proper credentials and background checks. Security as a Service providers recognize the fragility of the relationship and often go to extreme lengths to ensure that their environment is locked down as much as possible. They often run background checks on their personnel that rival even the toughest government background checks, and they run them often.

Physical and personnel security is one of the highest priorities of a Security as a Service provider.



Business Continuity and Disaster Recovery

- The customer should review the contract of third party commitments
- The customer should review the third party Business Continuity processes and any particular certification
- The customer should conduct an on-site assessment
- Cloud customers should not depend on a singular provider of services and should have a DR plan in place that facilitates migration or failover should a supplier fail.

SANS

SEC401 | Security Essentials Bootcamp Style 64

The evolution of cloud services has enabled business entities to do more with less: fewer resources and better operating efficiency. This has many tangible benefits for business, yet there are inherent security risks that must be evaluated, addressed, and resolved before businesses will have confidence in securely outsourcing their IT requirements to cloud service providers. Some of the security risks associated with cloud computing are unique, and it is in this context the business continuity, disaster recovery, and traditional security environments of a cloud service provider need to be assessed thoroughly.

BC recommendations:

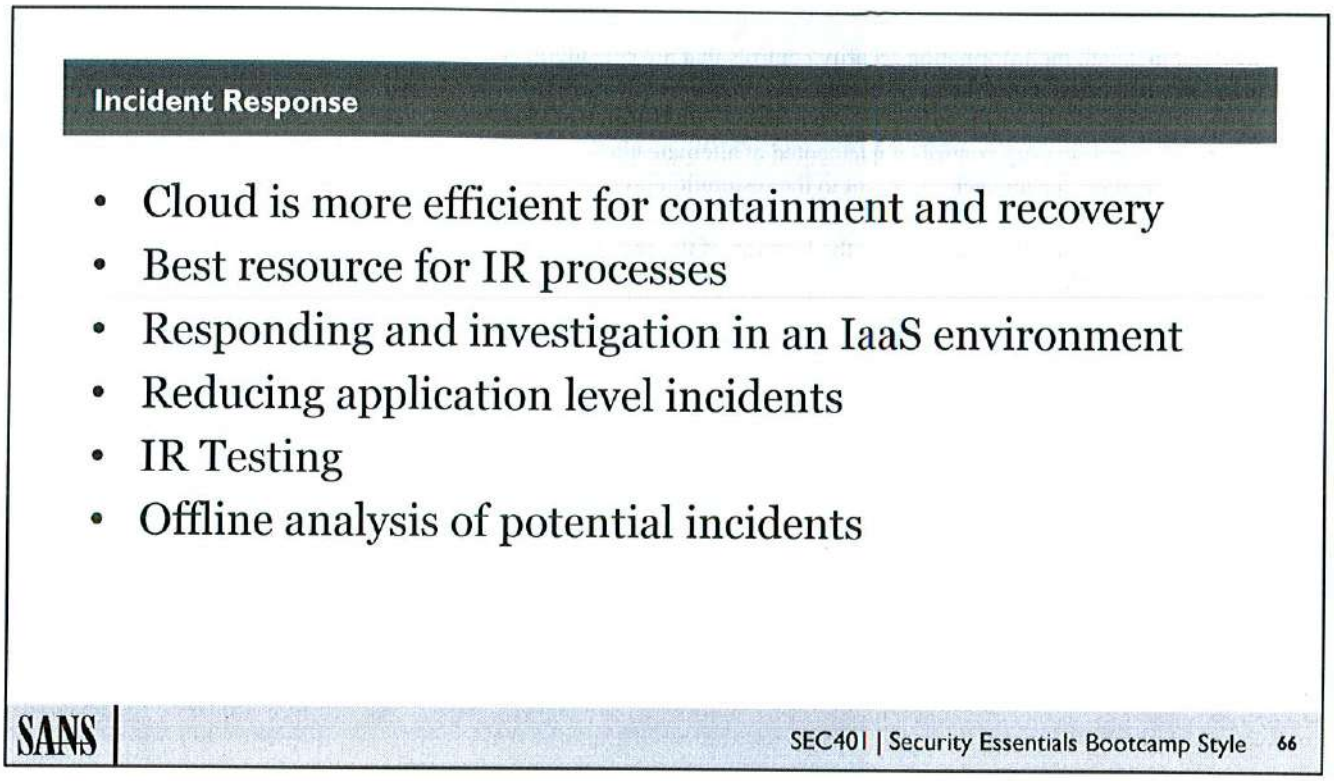
- The customer should review the contract of third party commitments to maintain continuity of the provisioned service.
- The customer should review the third party Business Continuity processes and any particular certification.
- The customer should conduct an on-site assessment of the CSP (Cloud Service Provider) facility to confirm and verify the asserted controls used to maintain the continuity of the service.
- The customer should ensure that he/she receives confirmation of any BCP/DR tests undertaken by the CSP.

DR Recommendations

- Cloud customers should not depend on a singular provider of services and should have a DR plan in place that facilitates migration or failover should a supplier fail.
- IaaS providers should have contractual agreements with multiple platform providers and have the tools in place to rapidly restore systems in the event of loss.
- Additional DR recommendations are listed in the CSA Guidance document.

It is important to review the service providers documented restoration plan. This plan should include details on the priorities regarding restoration sequencing. This should correlate directly with the SLA, as contractually committed, with regards to the services acquired by the customer and the criticality of the service. Be sure to spell out in detail, the Information security controls that are considered and implemented during the restoration process, which should include as an example:

- Clearances of staff involved during the restoration process
- Physical security controls implemented at alternate site
- Specified dependencies relevant to the restoration process – meaning other suppliers and outsource partners.
- And a minimum separation for the location of the secondary site if the primary site is made unavailable



Incident Response

- Cloud is more efficient for containment and recovery
- Best resource for IR processes
- Responding and investigation in an IaaS environment
- Reducing application level incidents
- IR Testing
- Offline analysis of potential incidents

SANS

SEC401 | Security Essentials Bootcamp Style 66

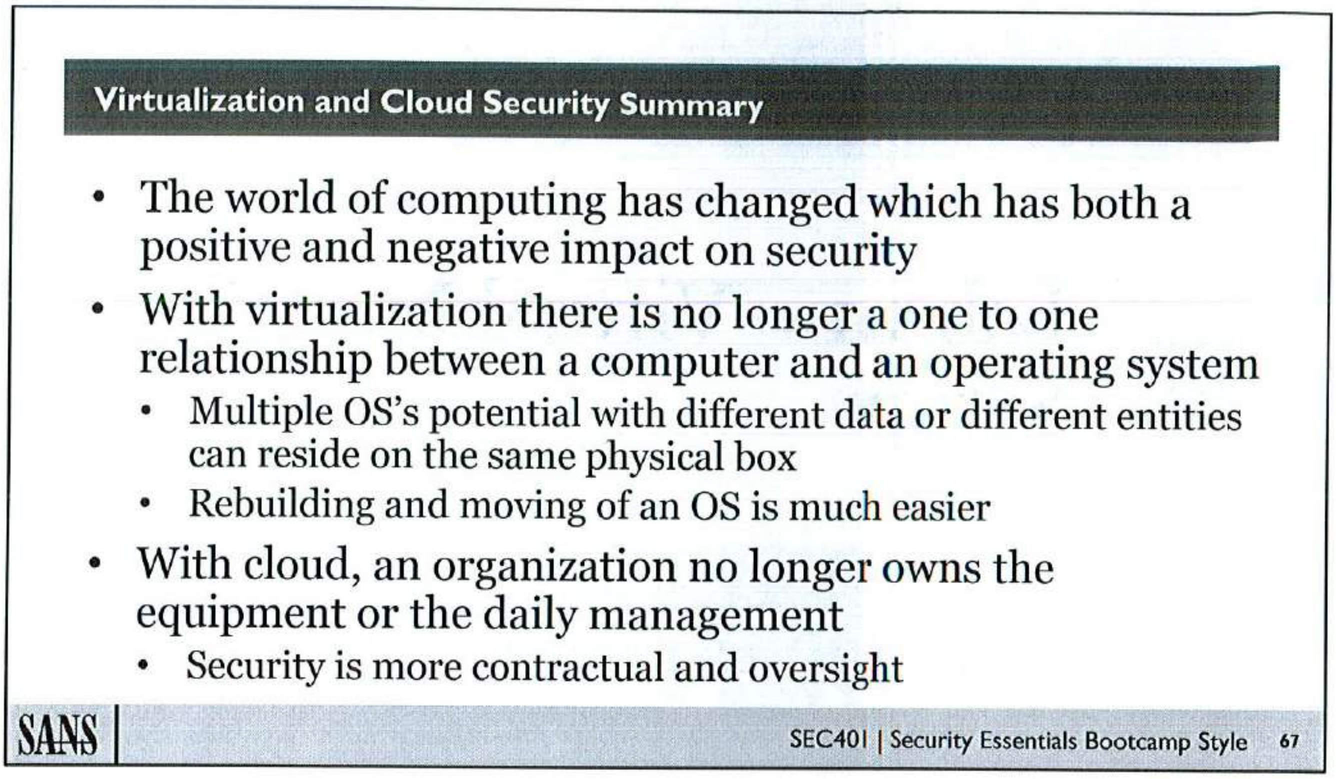
Incident Response (IR) is one of the cornerstones of information security management. Even the most diligent planning, implementation, and execution of preventive security controls cannot completely eliminate the possibility of an attack on the information assets. One of the central questions for organizations moving into the cloud must, therefore, be: what must be done to enable efficient and effective handling of security incidents that involve resources in the cloud?

Cloud computing presents good opportunities for incident responders. Cloud continuous monitoring systems can reduce the time it takes to undertake an incident handling exercise or deliver an enhanced response to an incident. Virtualization technologies, and the elasticity inherent in cloud computing platforms, may allow for more efficient and effective containment and recovery and often with less service interruption than might typically be experienced with more traditional data center technologies.

In an Infrastructure as a Service environment, a greater degree of responsibility and capability for detecting and responding to security incidents usually resides with the customer. However, even in IaaS there are significant dependencies on the Cloud provider. Data from physical hosts, network devices, shared services, and security devices like firewalls must be delivered by the Cloud provider. Some providers are already provisioning the capability to deliver this information to their customers and managed security service providers are advertising cloud-based solutions to receive and process this type data.

With Infrastructure as a Service configurations that have software level incidents, a customer can conduct forensic investigations of their own virtual instances but will not be able to investigate network components controlled by the Cloud provider.

Additionally, standard forensic activities such as the investigation of network traffic in general, access to snapshots of memory, or the creation of a hard disk image require investigative support to be provided by the Cloud provider as well. With platform service and software service security incidents that have their root cause in the underlying infrastructure, the cloud customer is almost completely reliant on analysis support from the Cloud provider. And as mentioned previously, roles and responsibilities in IR must be agreed upon in the SLAs.



Virtualization and Cloud Security Summary

- The world of computing has changed which has both a positive and negative impact on security
- With virtualization there is no longer a one to one relationship between a computer and an operating system
  - Multiple OS's potential with different data or different entities can reside on the same physical box
  - Rebuilding and moving of an OS is much easier
- With cloud, an organization no longer owns the equipment or the daily management
  - Security is more contractual and oversight

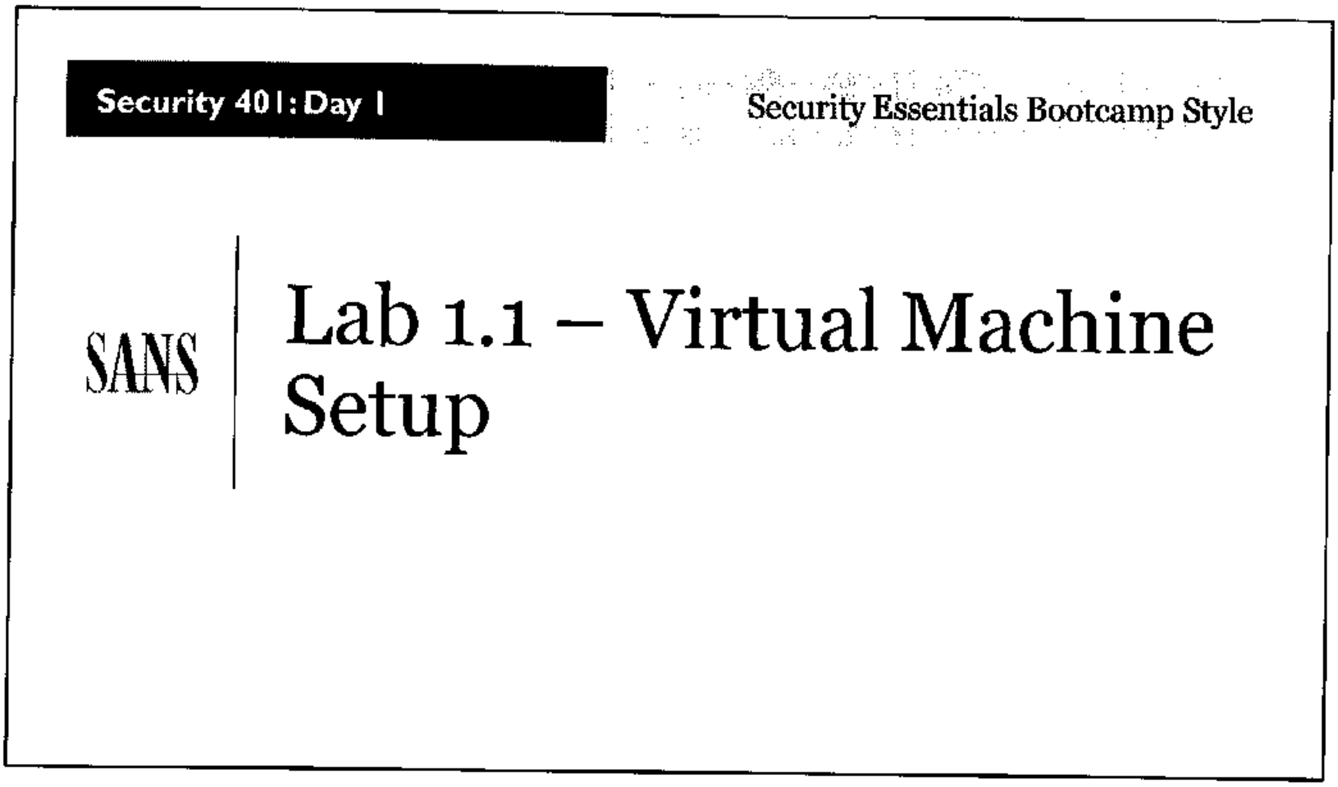
SANS

SEC401 | Security Essentials Bootcamp Style 67

The world of computing has changed which creates both unique opportunities and unique challenges, especially from a security standpoint. If you worked in security in the 90's the focus was primarily on securing a computer, which contained a single operating system and locking down the applications. This would include identifying open ports and vulnerable services and working closely with the administrator that built the system. Rebuilding systems, including patching or modifying the kernel of the OS was time-consuming and not very scalable. Each administrator would have to go to their data center, access the physical system and perform the appropriate maintenance and monitoring of the system. With virtualization and the cloud, things have changed.

With virtualization, there is no longer a one to one relationship between a computer and an operating system. While a computer still needs a host operating system for initial boot up and management of the physical resources, one physical computer can contain hundreds of guest operating systems all performing different functions and for different purposes. While this provides great flexibility and maximum usage of the hardware investment that organizations make, it could also greatly increase the exposure. If the virtualization is not configured correctly, an adversary could break into one system and potentially have access to many operating systems, applications and critical data.

The cloud is also a game-changer in that it allows for outsourcing of data center operations. It allows organizations to focus on their primary business and let a cloud provider focus on maintaining and supporting the IT infrastructure and/or applications that are needed to run the business. While cloud providers sole focus is on functionality and security, an organization has lost a level of control of verifying and validating the security. Depending on the cloud deployment, in some cases security is now focused more on contractual SLA's (service level agreements), reviewing reports and oversight. The bottom line to remember with the cloud is that you can outsource IT, but you cannot outsource liability. Depending on the contracts, an organization can still be exposed to liability from their customers, if a cloud provider does not implement effective security and the organization's information is compromised.



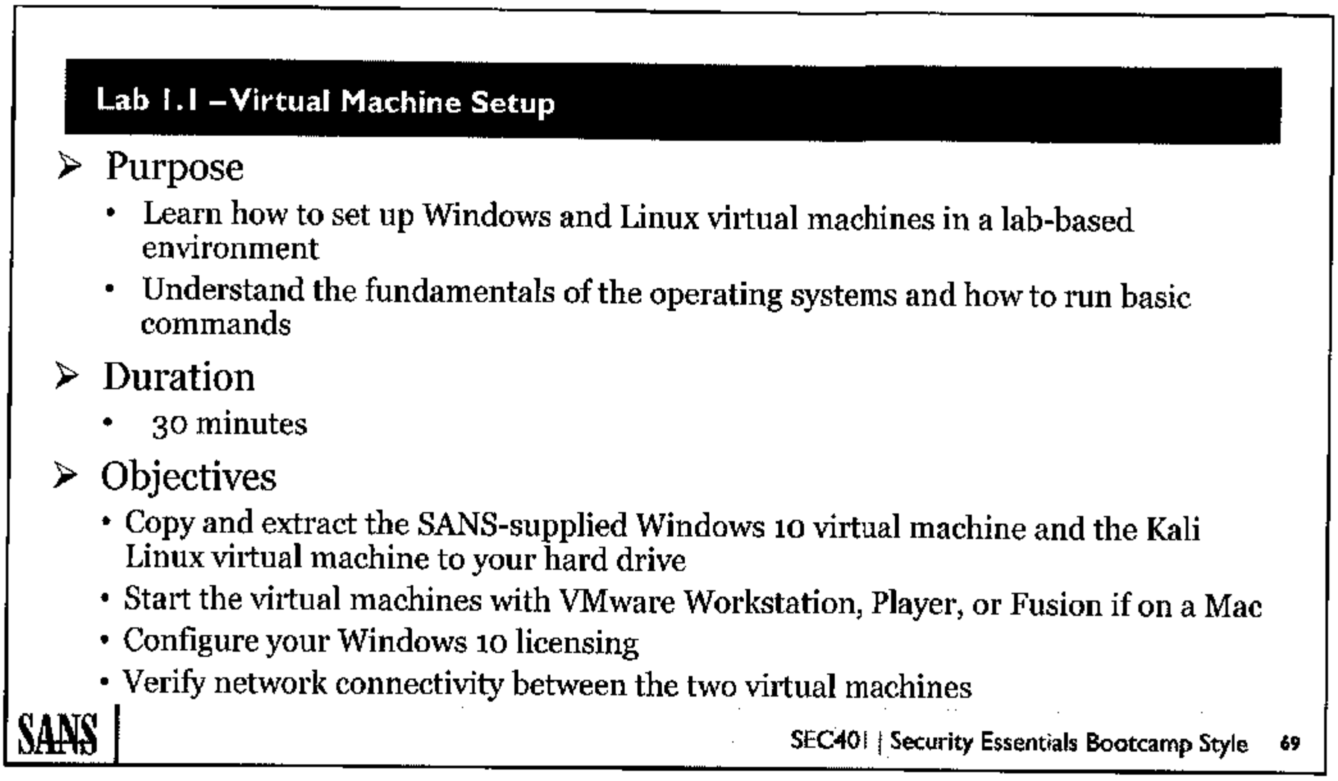
Security 401: Day 1

Security Essentials Bootcamp Style

SANS

Lab 1.1 – Virtual Machine Setup

In the module, you learned about the many ways virtual machines can be utilized in both lab and production environments. In this lab, you use a virtualization application to run multiple virtual machines that will be used during the rest of the course for many interactive labs.



Lab 1.1 –Virtual Machine Setup

- Purpose
  - Learn how to set up Windows and Linux virtual machines in a lab-based environment
  - Understand the fundamentals of the operating systems and how to run basic commands
- Duration
  - 30 minutes
- Objectives
  - Copy and extract the SANS-supplied Windows 10 virtual machine and the Kali Linux virtual machine to your hard drive
  - Start the virtual machines with VMware Workstation, Player, or Fusion if on a Mac
  - Configure your Windows 10 licensing
  - Verify network connectivity between the two virtual machines

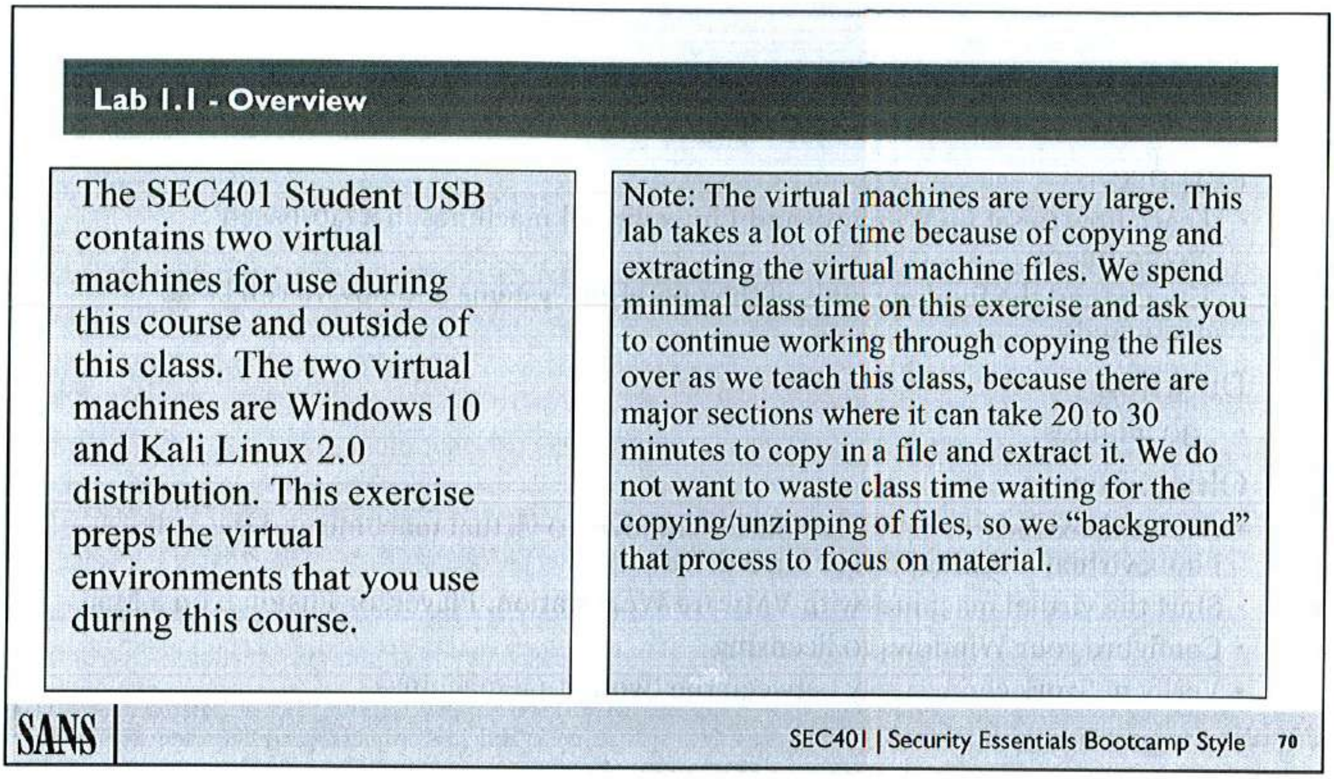
SANS | SEC401 | Security Essentials Bootcamp Style 69

Purpose

- Learn how to set up Windows and Linux virtual machines in a lab based environment.
- Understand the fundamentals of the operating systems and how to run basic commands.

Objectives

- Copy and extract the SANS-supplied Windows 10 virtual machine and the Kali Linux virtual machine onto your hard drive.
- Start the virtual machines with VMware Workstation, Player, or Fusion if on a Mac.
- Configure your Windows 10 licensing.
- Verify network connectivity between the two virtual machines.



Lab 1.1 - Overview

The SEC401 Student USB contains two virtual machines for use during this course and outside of this class. The two virtual machines are Windows 10 and Kali Linux 2.0 distribution. This exercise preps the virtual environments that you use during this course.

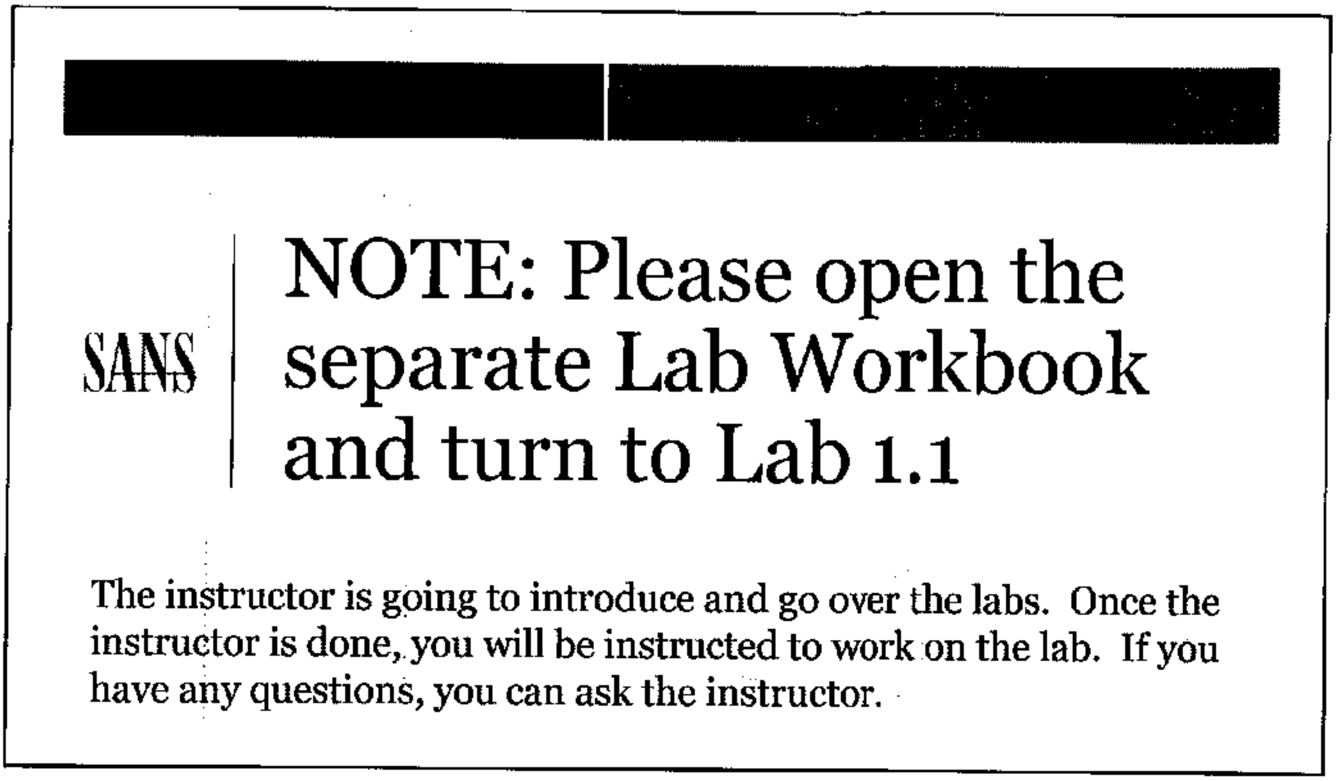
Note: The virtual machines are very large. This lab takes a lot of time because of copying and extracting the virtual machine files. We spend minimal class time on this exercise and ask you to continue working through copying the files over as we teach this class, because there are major sections where it can take 20 to 30 minutes to copy in a file and extract it. We do not want to waste class time waiting for the copying/unzipping of files, so we "background" that process to focus on material.

SANS

SEC401 | Security Essentials Bootcamp Style 70

The SEC401 Student USB contains two virtual machines for use during this course and outside of this class. The two virtual machines are Windows 10 and Kali Linux 2.0 distribution. This exercise preps the virtual environments that you use during this course.

Note: The virtual machines are very large. This lab takes some time because of copying and extracting the virtual machine files. We spend minimal class time on this exercise and ask you to continue working through copying the files over as we teach the class because there are major sections where it can take 20 to 30 minutes to copy a file and extract it. We do not want to waste precious class time waiting for the copying/unzipping of files, so we "background" that process to focus on the material.

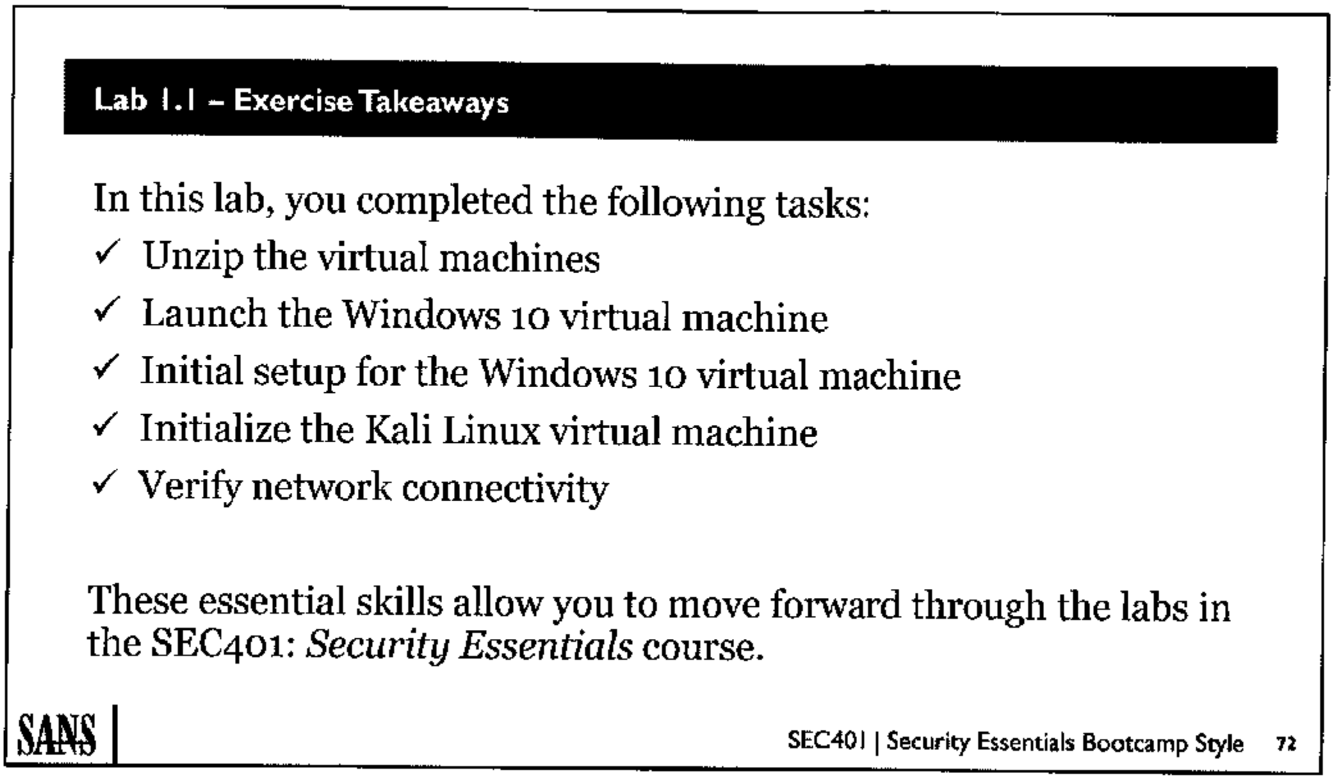


SANS

NOTE: Please open the separate Lab Workbook and turn to Lab 1.1

The instructor is going to introduce and go over the labs. Once the instructor is done, you will be instructed to work on the lab. If you have any questions, you can ask the instructor.

This page intentionally left blank.



Lab 1.1 – Exercise Takeaways

In this lab, you completed the following tasks:

- ✓ Unzip the virtual machines
- ✓ Launch the Windows 10 virtual machine
- ✓ Initial setup for the Windows 10 virtual machine
- ✓ Initialize the Kali Linux virtual machine
- ✓ Verify network connectivity

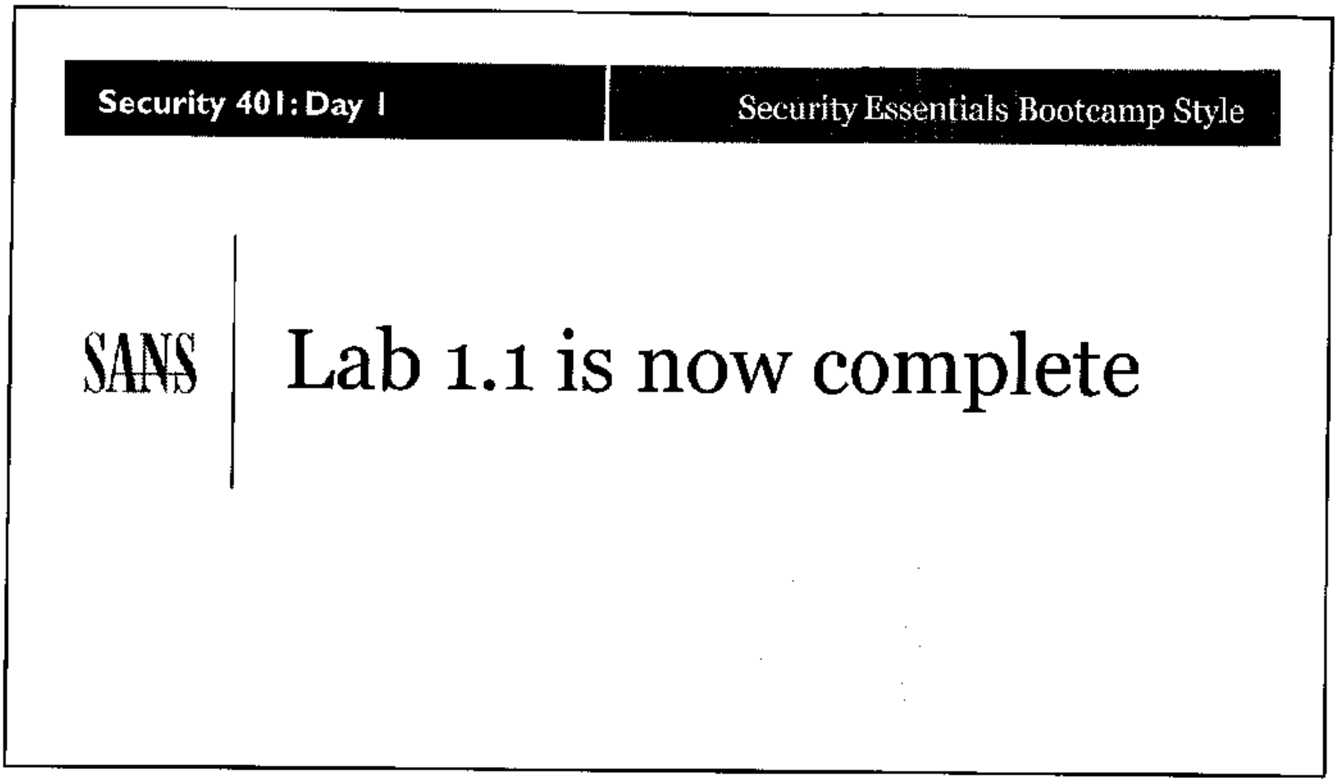
These essential skills allow you to move forward through the labs in the SEC401: *Security Essentials* course.

SANS | SEC401 | Security Essentials Bootcamp Style 72

In this lab, you completed the following tasks:

- ✓ Unzip the virtual machines.
- ✓ Launch the Windows 10 virtual machine.
- ✓ Initial setup for the Windows 10 virtual machine.
- ✓ Initialize the Kali Linux virtual machine.
- ✓ Verify network connectivity.

These essential skills allow you to move forward through the labs in the SEC401: *Security Essentials* course.



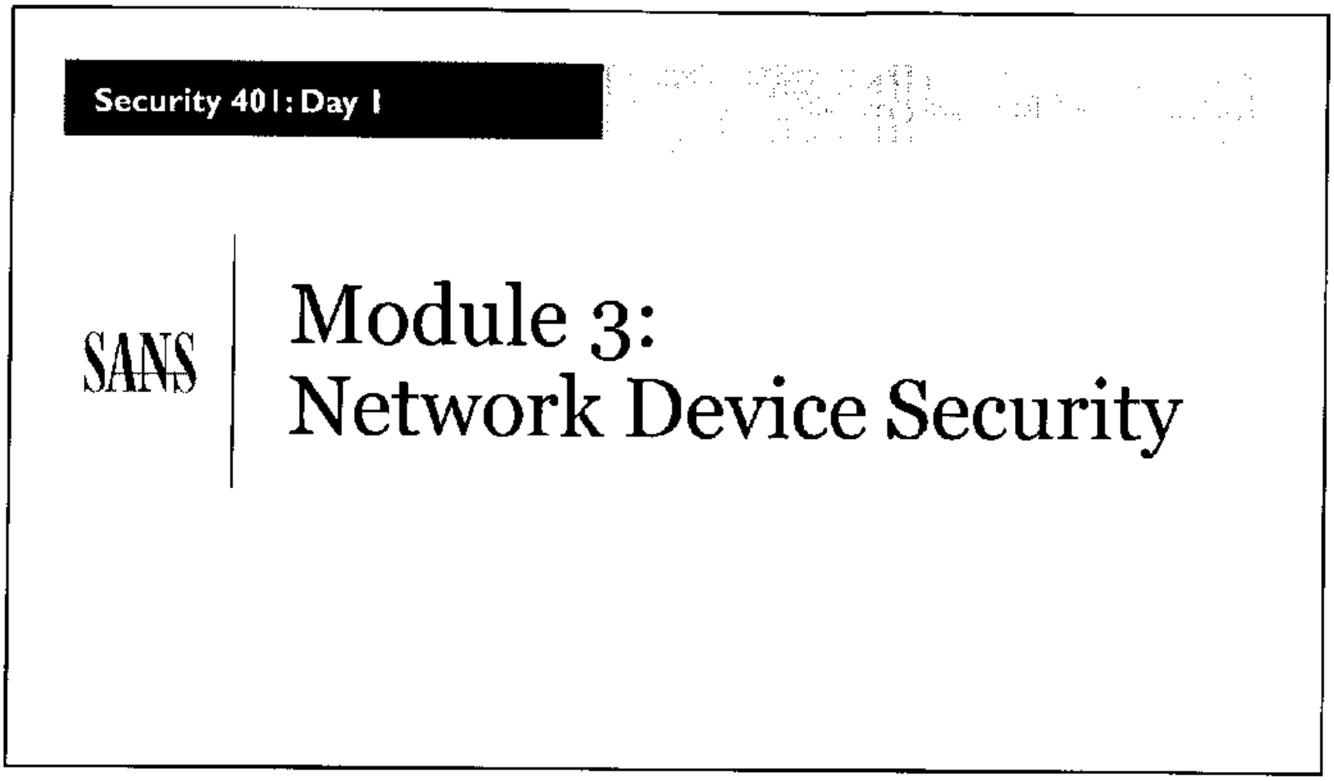
Security 401: Day 1

Security Essentials Bootcamp Style

SANS

Lab 1.1 is now complete

This page intentionally left blank.



Security 401: Day 1

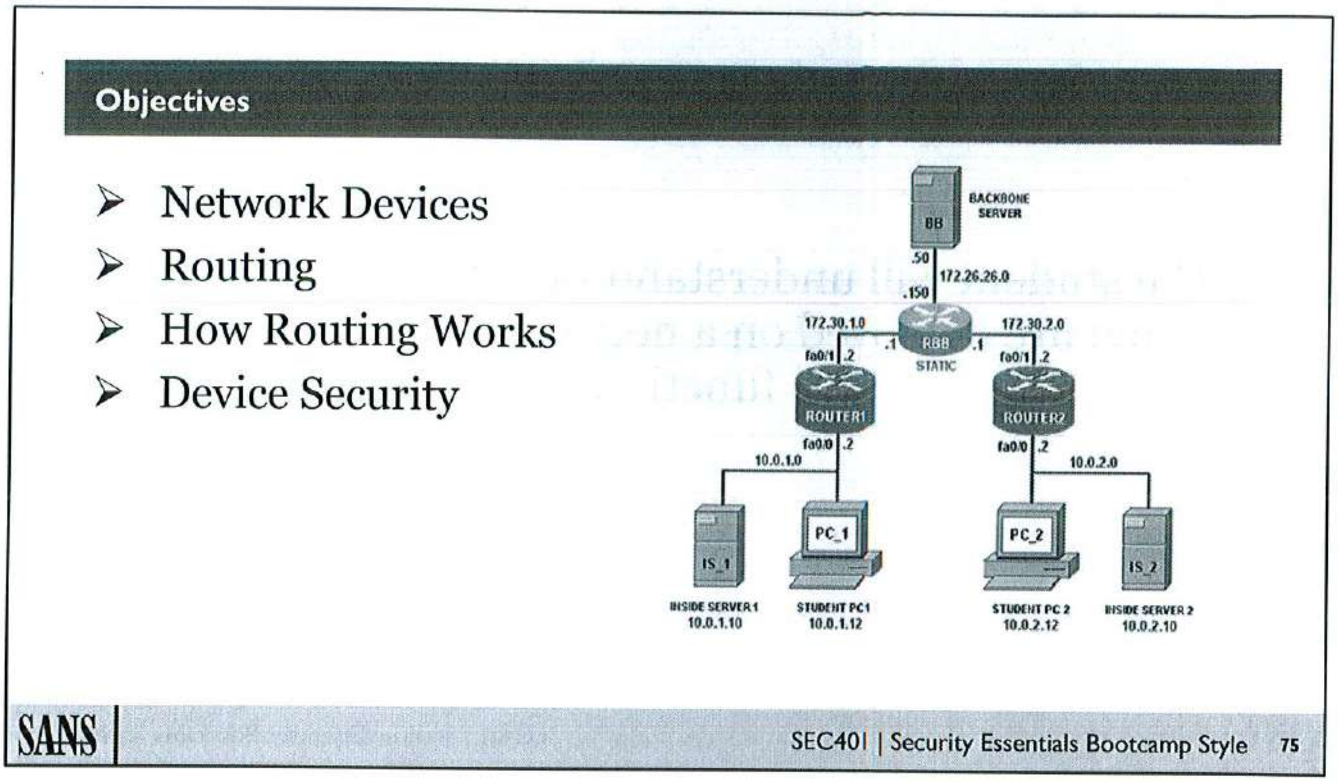
SANS

Module 3: Network Device Security

Module 3: Network Device Security

The following references are used for routers:
http://searchnetworking.techtarget.com/tip/Hardening-your-router-in-9-easy-steps
http://packetpushers.net/cisco-ios-device-hardening/
http://stackoverflow.com/questions/23935095/how-are-mac-addresses-used-in-routing-packets
http://www.informit.com/articles/article.aspx?p=131034&seqNum=5
https://askleo.com/whats_the_difference_between_a_mac_address_and_an_ip_address/
http://www.erg.abdn.ac.uk/users/gorry/course/inet-pages/arp.html

The following references are used for switches:
https://isc.sans.edu/forums/diary/Switch+hardening+on+your+network/6910/
http://www.larrytalkstech.com/port-forwarding-small-network-security/
http://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Security/Baseline_Security/securebasebook/sec_chap7.html
http://searchsecurity.techtarget.com/WLAN-security-Best-practices-for-wireless-network-security



Objectives

- ➢ Network Devices
- ➢ Routing
- ➢ How Routing Works
- ➢ Device Security

SANS | SEC401 | Security Essentials Bootcamp Style 75

A network device is a component used to connect a computer or other electronic devices together so they can access files or resources such as printers.
In order to implement proper security, you have to understand what the various components on a network are. In this module, we will look at how the various components work and methods to properly secure them.

A Router is a network component designed to take information that arrives through your network connection and deliver it to your computer. The router will also choose the best route for the data packet so that you receive the information quickly.

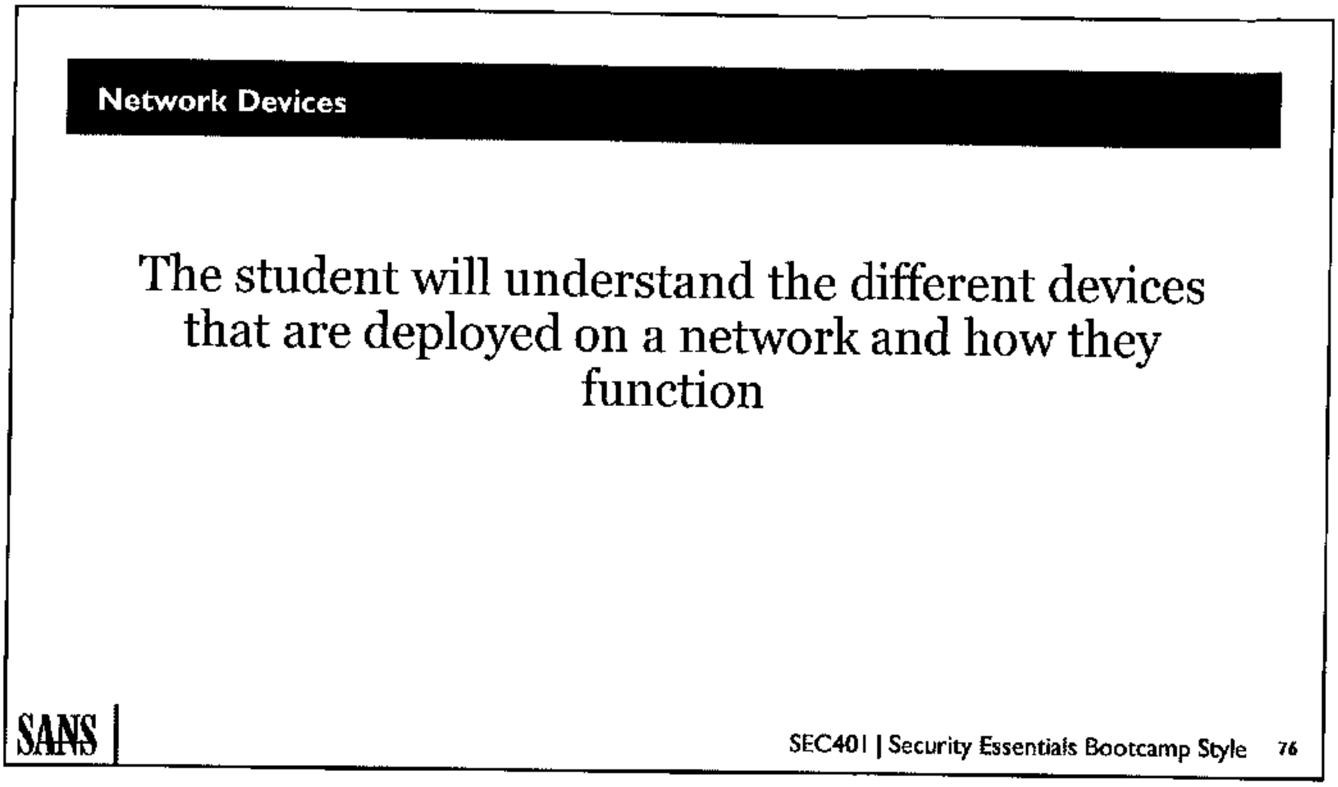
A Switch serves as a controller, enabling networked devices to talk to each other efficiently. Most business networks today use switches to connect computers, printers, and servers within a building which saves money through information sharing and resource allocation.

A Hub is the most basic networking device that connects multiple computers or other network devices together. Hubs have no routing tables or intelligence regarding where to send information and broadcasts all network data across each connection.

A Bridge creates a single aggregate network from multiple communication networks or network segments. Bridging is distinct from routing, which allows multiple different networks to communicate independently while remaining separate.

References

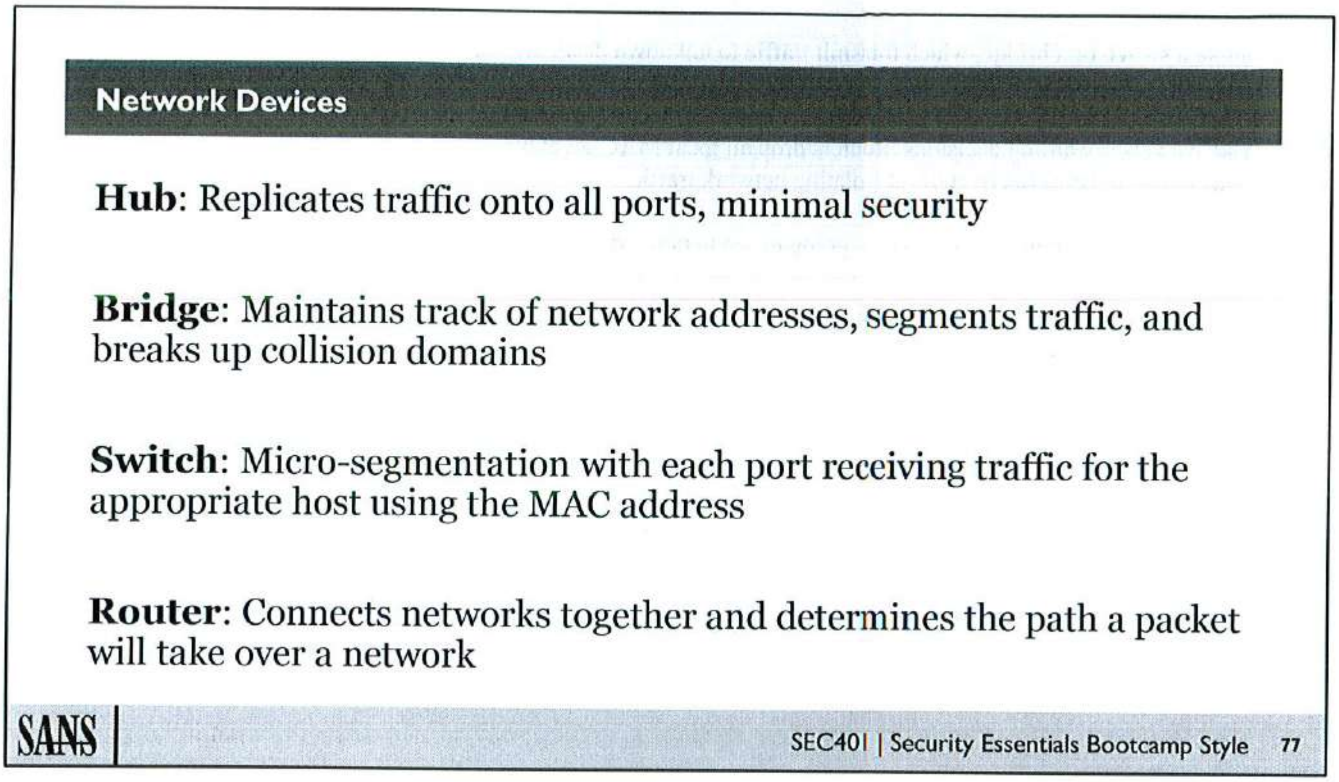
1. http://www.cisco.com/c/en/us/solutions/small-business/resource-center/connect-employees-offices/network-switch-how.html
2. http://whatismyipaddress.com/router
3. http://www.computerhope.com/jargon/h/hub.htm
4. https://www.netdevgroup.com/content/cnap/topologies/security_router_pod.html



Network Devices

The student will understand the different devices that are deployed on a network and how they function

This page intentionally left blank.



Network Devices

Hub: Replicates traffic onto all ports, minimal security

Bridge: Maintains track of network addresses, segments traffic, and breaks up collision domains

Switch: Micro-segmentation with each port receiving traffic for the appropriate host using the MAC address

Router: Connects networks together and determines the path a packet will take over a network

SANS SEC401 | Security Essentials Bootcamp Style 77

Several types of devices are commonly used at the core of the network to provide a reliable and flexible communication medium. It is important to understand how they function because each has inherent security strengths and weaknesses.

A hub operates by "repeating" data that it receives on one port to its other ports. As a result, a data frame transmitted by one system is retransmitted to all other systems connected to the hub. A classic hub does not have traffic-monitoring capabilities and cannot control which ports should or should not receive the frame, forming a large collision domain. This property of a hub has significant security implications because a system connected to the hub may be able to intercept a data frame destined for someone else.

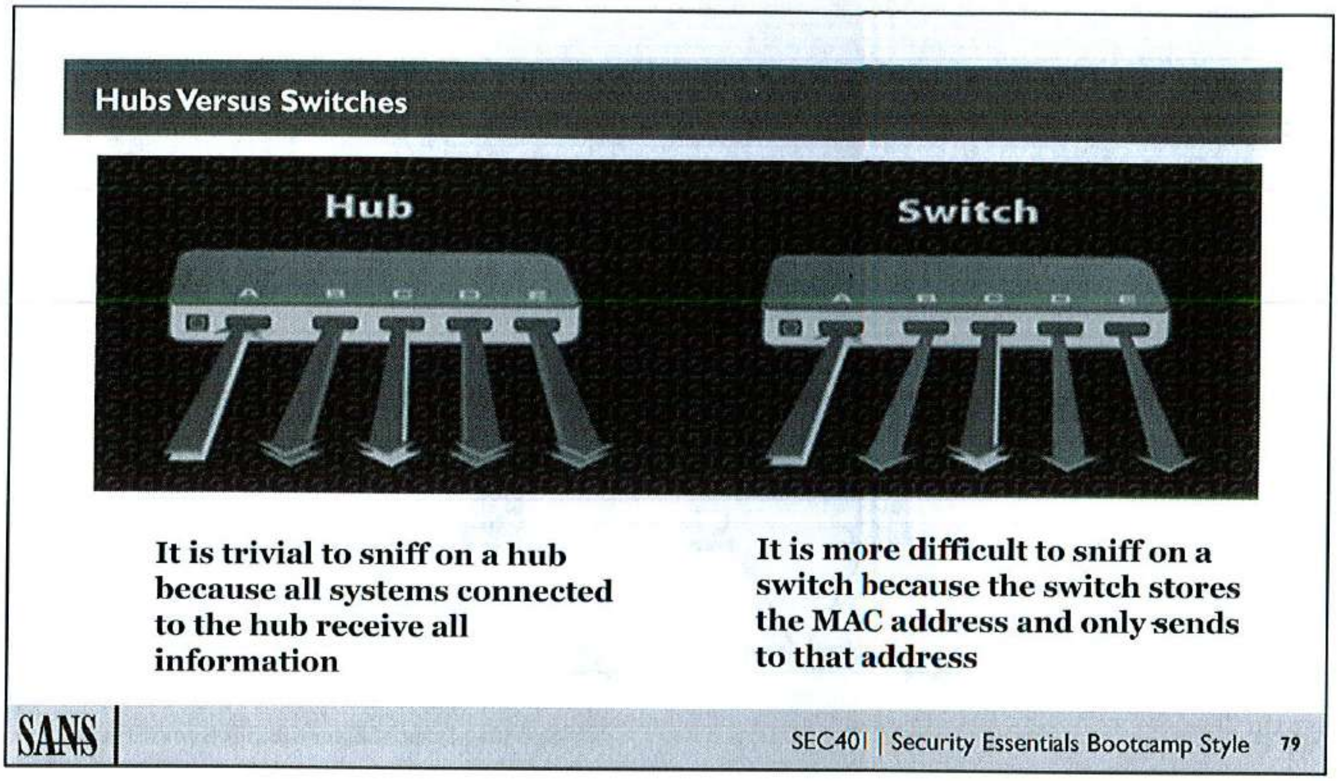
A bridge connects two physical segments of a network in much the same way as an over-the-water bridge connects two sections of a road. When a bridge receives a data frame on one of its ports, it makes a decision about whether the data should be sent to the other port. This functionality allows a bridge to automatically control the flow of data between network segments that it connects.

A network switch combines the functionality of a hub and a bridge into a single device. If you think of a switch as a bridge with more than two ports, you will get the idea. Like a hub, a switch can retransmit data to multiple ports. Additionally, an Ethernet switch keeps track of MAC addresses attached to each of its ports, which grants it the traffic control capabilities of a bridge. By monitoring and controlling traffic between its ports, a switch directs a data frame only to the system or network segment for which it is destined, narrowing each port to its own collision domain. Sniffing becomes ineffective with switches.

Routers are often considered to be perimeter devices because they interconnect logical networks. A switch or a bridge, on the other hand, connects physical segments that reside on the same logical network. Much of the Internet relies on routers for determining which paths packets should take to get from one network to another. Similar to a switch or a bridge, a router makes decisions about where to direct data that passes through it. A switch makes its decisions by tracking MAC addresses, whereas a router operates on a higher layer by looking at IP addresses when forwarding packets.

Unlike a switch or a bridge, which transmit traffic to unknown destinations, a router drops traffic if it does not know where to send it. Routers need to be explicitly configured with information that defines paths for directing traffic to all reachable networks. The router stores this information in a routing table, which it uses to make packet-forwarding decisions. Routers drop all local MAC-level broadcast traffic by default, thereby contributing to their effectiveness at isolating network traffic.

The two most common devices you are going to see in networks today are routers and switches.



Hubs Versus Switches

Hub
A B C D E
Switch
A B C D E

It is trivial to sniff on a hub because all systems connected to the hub receive all information

It is more difficult to sniff on a switch because the switch stores the MAC address and only sends to that address

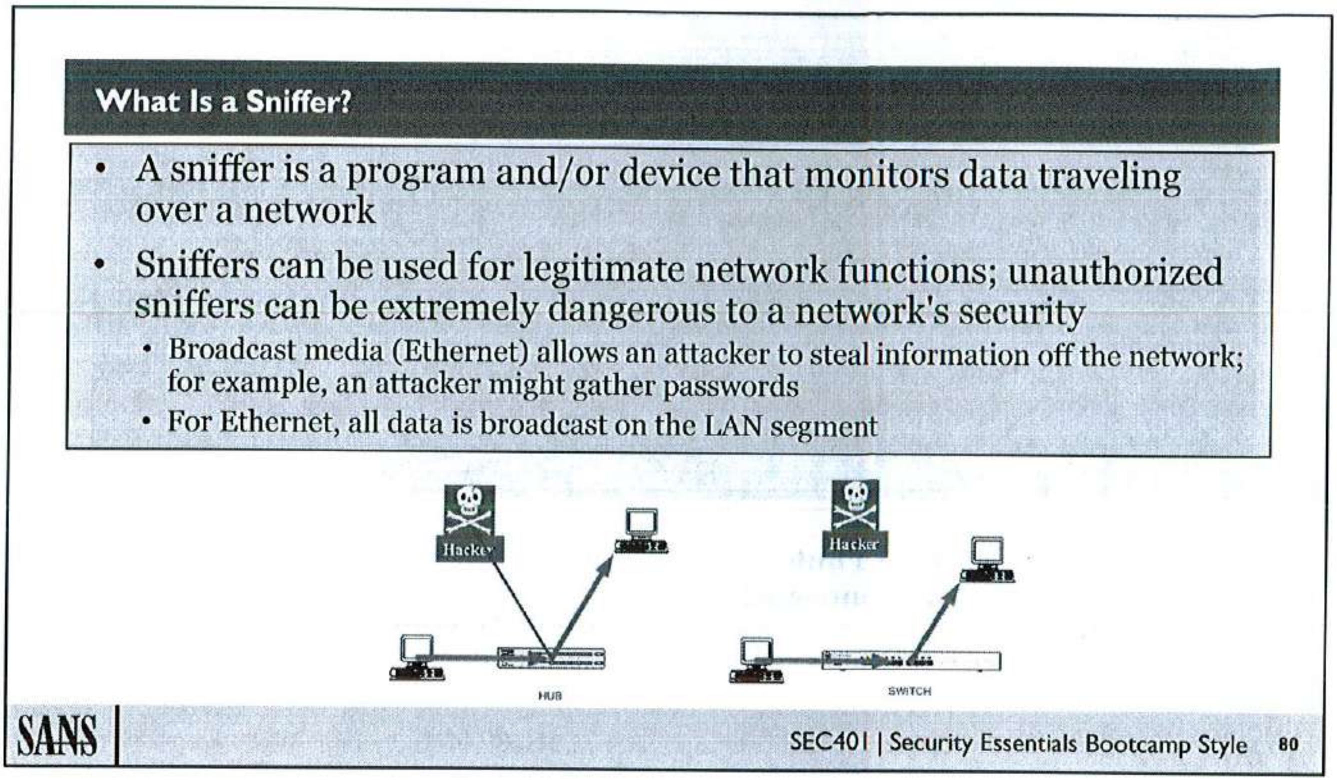
SANS

SEC401 | Security Essentials Bootcamp Style 79

Both hubs and switches are used to connect devices together to form a network. The difference between the two devices is based on whether the analysis is done at the host or at the device level. When two devices are going to directly communicate, the sending computer has to identify the MAC address of the receiving computers network interface card (NIC). This is typically burnt into the NIC and is a 6-byte value. When the receiving computer receives a frame, it will check to make sure that the destination MAC address matches the MAC address of the NIC. If they match it will process the frame and if they do not match, it will discard the frame.

With a hub, no additional analysis is performed. When computer A wants to send a frame to computer C, it will specify computer C's MAC address as the destination MAC and send the frame to the hub. The hub will just relay the frame to every device connected to it and perform no additional analysis. It is up to each receiving computer to check the MAC address to see if the frame is for their system. While hubs are fairly basic they do represent some security issues. Since every computer receives all of the frames being sent over the network, if a computer installs a sniffer, it would be able to access all of the information being sent to every other computer. A sniffer is a piece of software that puts a NIC into promiscuous mode. This, in essence, tells the NIC to accept every frame regardless of whether the MAC address matches or not.

A switch takes the logic that is performed by the computers to check the make address and the switch performs that analysis. When a computer connects to a switch, the switch will probe the system to identify the MAC address of the computer connected to a specific port. The switch will store the MAC address for each computer in its memory. Now when computer A wants to communicate with computer C, it will send the frame to the switch. The switch will examine the frame, look at the destination MAC address and send it only to the port that contains the MAC address. Switches increase security by reducing visibility. Now if computer E is running a sniffer, it will not see all of the frames going across the network.



What Is a Sniffer?

- A sniffer is a program and/or device that monitors data traveling over a network
- Sniffers can be used for legitimate network functions; unauthorized sniffers can be extremely dangerous to a network's security
  - Broadcast media (Ethernet) allows an attacker to steal information off the network; for example, an attacker might gather passwords
  - For Ethernet, all data is broadcast on the LAN segment

SANS | SEC401 | Security Essentials Bootcamp Style 80

Data is sent across a network in discrete bundles called packets. For a sniffer to see packets, they have to arrive on the network interface of the host running the sniffer. Out of all the packets traversing the network, which ones reach the sniffer host depends on the network device connecting the hosts on the network.

One of the most popular networking technologies is Ethernet. Each device on an Ethernet, be it a computer, router, or hardware sniffer, has a 6-byte physical address or Ethernet address. This physical or Ethernet address is the same as the MAC address. Devices with multiple Ethernet interfaces often have different physical addresses for each interface, but not always. Usually, the physical address is written as a series of hexadecimal numbers, each representing a byte, separated by colons (00:30:65:01:b8:c5, for example). The physical address of a device has nothing to do with its IP address. Devices on an Ethernet use the Address Resolution Protocol (ARP) to determine what physical address an IP is associated with, so they know where to send IP datagrams.

An Ethernet packet, also called a frame, contains in its header the physical addresses of the source and destination devices. Usually, a device automatically accepts a frame whose destination address is the same as its own.

Hubs connect traditional Ethernet networks. When a hub sees an incoming frame, it simply broadcasts it out all of its ports. Such shared Ethernets are inherently sniffable; all the data transmitted by any host is visible to all other hosts on the hub.

Because of the broadcast nature of shared Ethernet, hosts have to be selective about which frames they accept. A host normally rejects frames that are not destined for it. But, to get as large a sample as possible, sniffers use the interface's promiscuous mode to accept all frames, even those intended for other hosts.

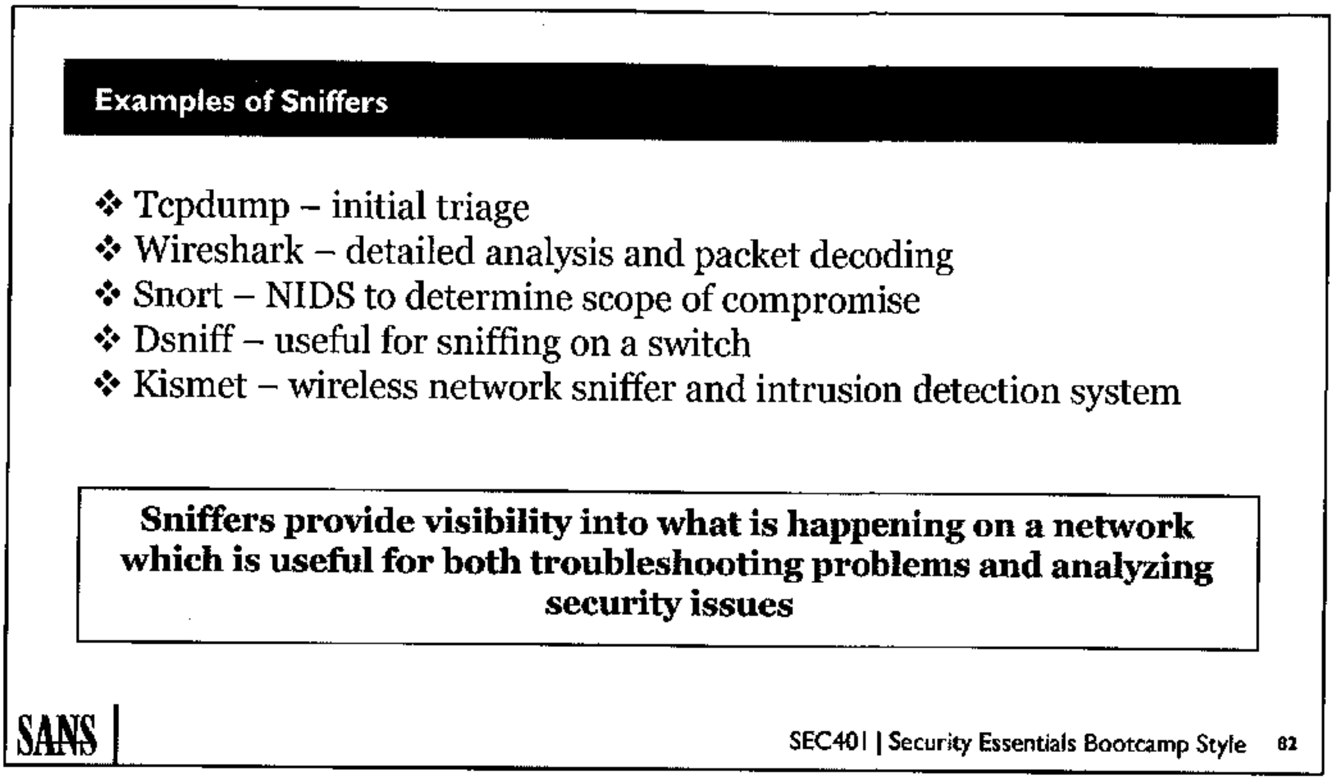
It's not a bad idea to periodically check network interfaces to see if they are in promiscuous mode without your knowledge. If they are, an attacker might have compromised your machine and installed a sniffer.

The Ethernet switch is more intelligent than the hub. By inspecting incoming frames, the switch determines the physical address of the destination host and then finds the port on the switch to which the system is connected. Then, it sends the packet out on the correct port.

Because a sniffer on a switched network sees only traffic to and from the sniffer's host, promiscuous mode has no effect. For this reason, the use of switched networking is considered a security feature. After all, a compromised machine on a switched network can sniff only passwords and sensitive data going to and from that machine, something that could be done by other means, anyway (keyboard sniffing or just perusing the file system).

But, it is still possible to sniff traffic from other hosts on a switched Ethernet network by impersonating a local router, for example. Tools, such as dsniff and Ettercap, facilitate sniffing in a switched environment. Even though switched networking improves security by preventing casual sniffing, strong cryptography should be used to protect passwords and sensitive data.

Reference
1. Winsock Packet Editor - http://www.qweas.com/downloads/system/other/overview-winsock-packet-editor-wpe-pro.html



Examples of Sniffers

- ❖ Tcpdump – initial triage
- ❖ Wireshark – detailed analysis and packet decoding
- ❖ Snort – NIDS to determine scope of compromise
- ❖ Dsniff – useful for sniffing on a switch
- ❖ Kismet – wireless network sniffer and intrusion detection system

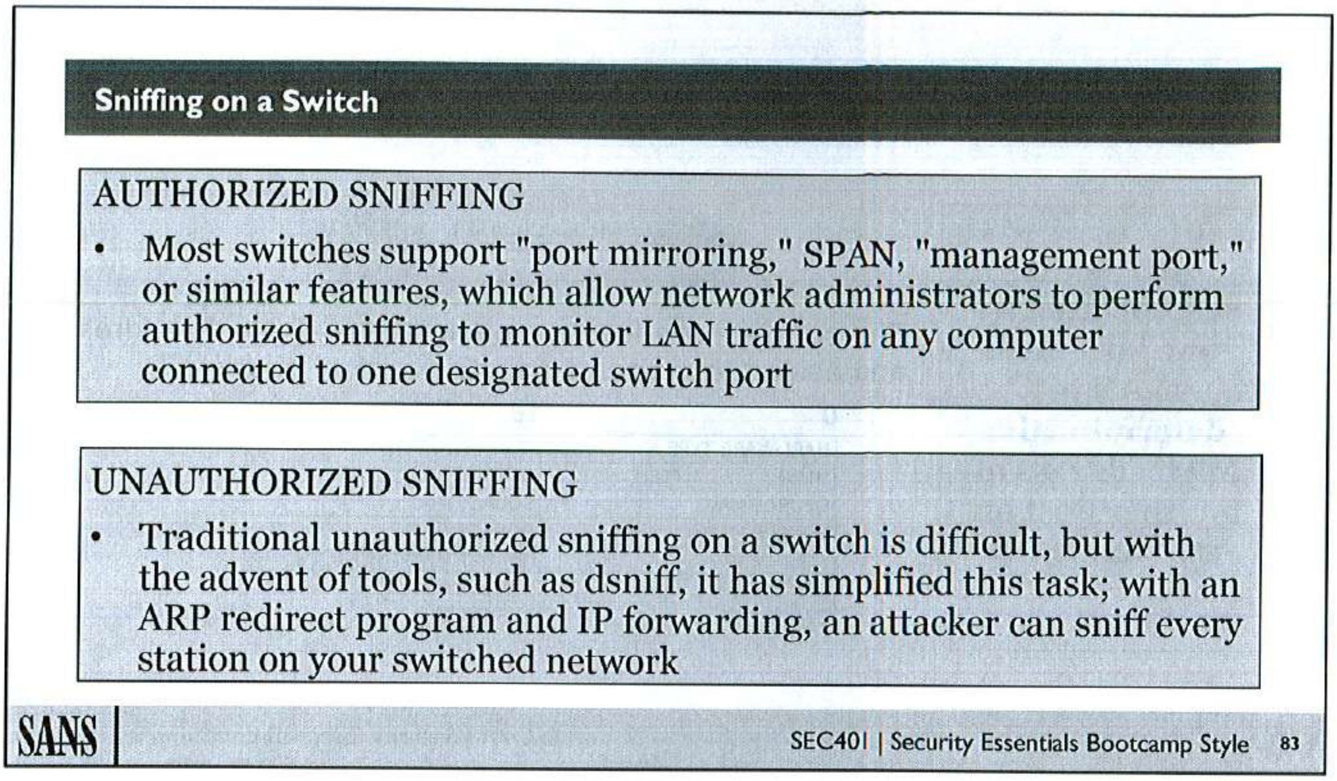
Sniffers provide visibility into what is happening on a network which is useful for both troubleshooting problems and analyzing security issues

SANS | SEC401 | Security Essentials Bootcamp Style 82

The first step in analyzing probes and other malicious traffic is to see it with your own eyes. It is common sense to want to see the guts of a system when it is not working properly. That is why the sniffer is a favorite tool of system and network administrators. Because sniffers can gather all information transmitted on a network at a given time, including passwords and other sensitive data, they are also a favorite among attackers.

Sniffers can be hardware devices that physically attach to the network, but more commonly, they are software programs that run on networked computers. The sniffers that come bundled with your operating system are designed as tools for the system administrator. No matter what your needs, your interest, or your budget, there probably is a sniffer out there that does what you want. Some sniffers are designed for more specialized, nefarious purposes. Sniffers bundled with rootkits often are designed to seek out usernames and passwords in the network data and extract them into files. To use this type of sniffer, the attackers do not need any technical knowledge. They just run a program, and after a while, they have a file full of usernames and passwords they can use for further intrusion on the network.

Attackers can also use your own sniffers against you. Is this an argument against using sniffers as network-analysis tools? Of course not; sniffers are too valuable for you to do without entirely, and attackers can always bring their own. But, it's worth the effort to keep sniffers out of easy reach of a potential attacker. It's bad enough if one of your production servers gets compromised, but worse still if the attacker is able to use a locally installed copy of tcpdump to capture passwords or other sensitive information.



Sniffing on a Switch

AUTHORIZED SNIFFING

- Most switches support "port mirroring," SPAN, "management port," or similar features, which allow network administrators to perform authorized sniffing to monitor LAN traffic on any computer connected to one designated switch port

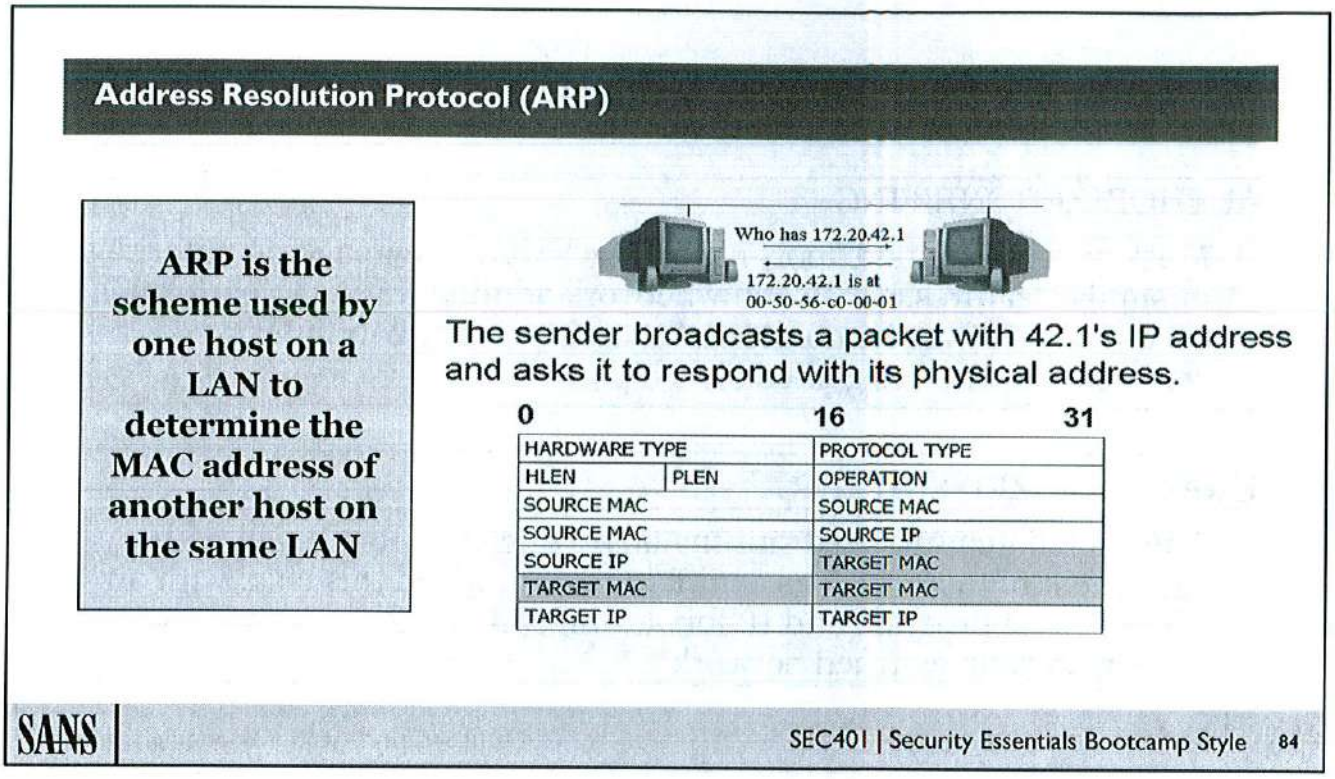
UNAUTHORIZED SNIFFING

- Traditional unauthorized sniffing on a switch is difficult, but with the advent of tools, such as dsniff, it has simplified this task; with an ARP redirect program and IP forwarding, an attacker can sniff every station on your switched network

SANS SEC401 | Security Essentials Bootcamp Style 83

To sniff traffic on a switched segment is not an impossible operation. dsniff is software that accomplishes this task in a simple way: The attacker's system sends out a forged ARP packet to the target system, telling it that its default gateway has changed to the attacker's system. When the target system sends traffic on the network, it sends it to the attacker's system first, which then forwards the packet on to its original destination as if nothing ever happened.

Another utility to perform this task is Ettercap. Ettercap is a second-generation sniffer that allows you to sniff all the traffic, even in a switched network. Ettercap relies heavily on ARP cache poisoning (where the attacker sends false ARP replies to associate his MAC address to the IP addresses of both the source and the target hosts).



Address Resolution Protocol (ARP)

ARP is the scheme used by one host on a LAN to determine the MAC address of another host on the same LAN

The sender broadcasts a packet with 42.1's IP address and asks it to respond with its physical address.

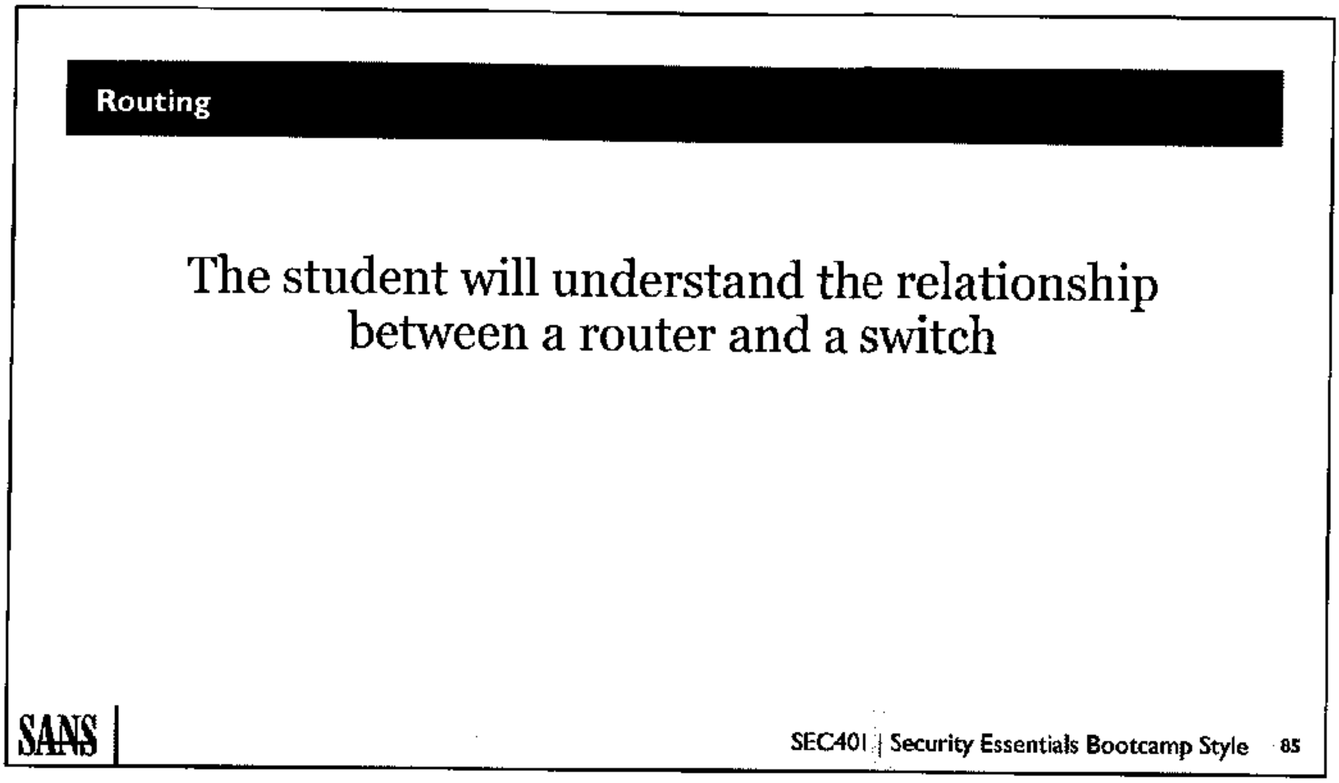
0		16	31
HARDWARE TYPE		PROTOCOL TYPE	
HLEN	PLEN	OPERATION	
SOURCE MAC		SOURCE MAC	
SOURCE MAC		SOURCE IP	
SOURCE IP		TARGET MAC	
TARGET MAC		TARGET MAC	
TARGET IP		TARGET IP	

ARP, described in RFC 826, is the scheme used by one host on a LAN to determine the MAC address of another host on the same LAN. There are three scenarios in which ARP will be used: a host looking for the MAC address of another host on the LAN; a host looking for the MAC address of the default gateway; or a router looking for the MAC address of a host or another router on a LAN. The process is straightforward.

The ARP request is a Layer 2 broadcast and looks something like this: "Who has 172.20.42.1?; tell 172.20.42.2." The 172.20.42.1 machine should reply back to 172.20.42.2 with its MAC address. Now the original machine will place this entry in its ARP cache in case it needs it in the near future.

Reference

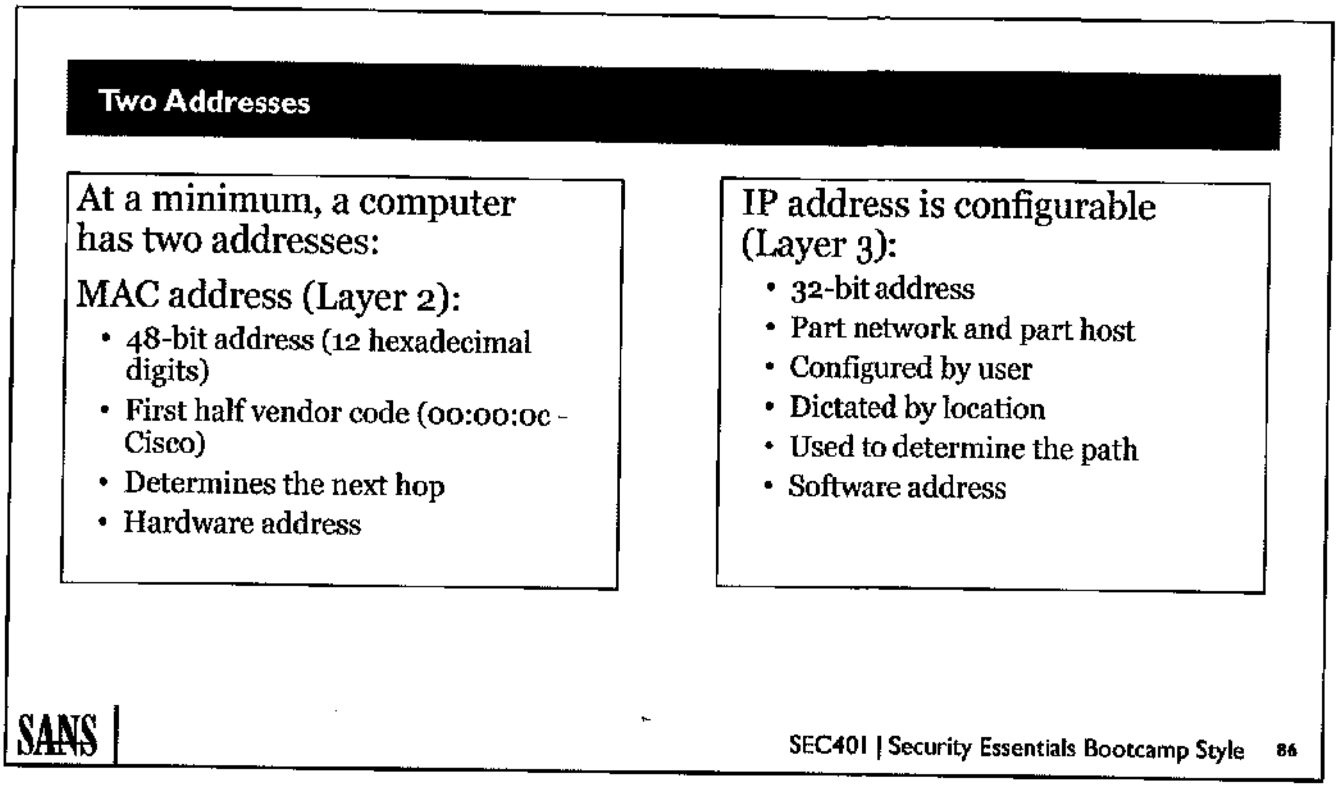
1. https://en.wikipedia.org/wiki/ARP_spoofing#/media/File:ARP_Spoofing.svg.



Routing

The student will understand the relationship between a router and a switch

This page intentionally left blank.



Two Addresses

At a minimum, a computer has two addresses:

MAC address (Layer 2):

- 48-bit address (12 hexadecimal digits)
- First half vendor code (00:00:0c - Cisco)
- Determines the next hop
- Hardware address

IP address is configurable (Layer 3):

- 32-bit address
- Part network and part host
- Configured by user
- Dictated by location
- Used to determine the path
- Software address

SANS | SEC401 | Security Essentials Bootcamp Style 86

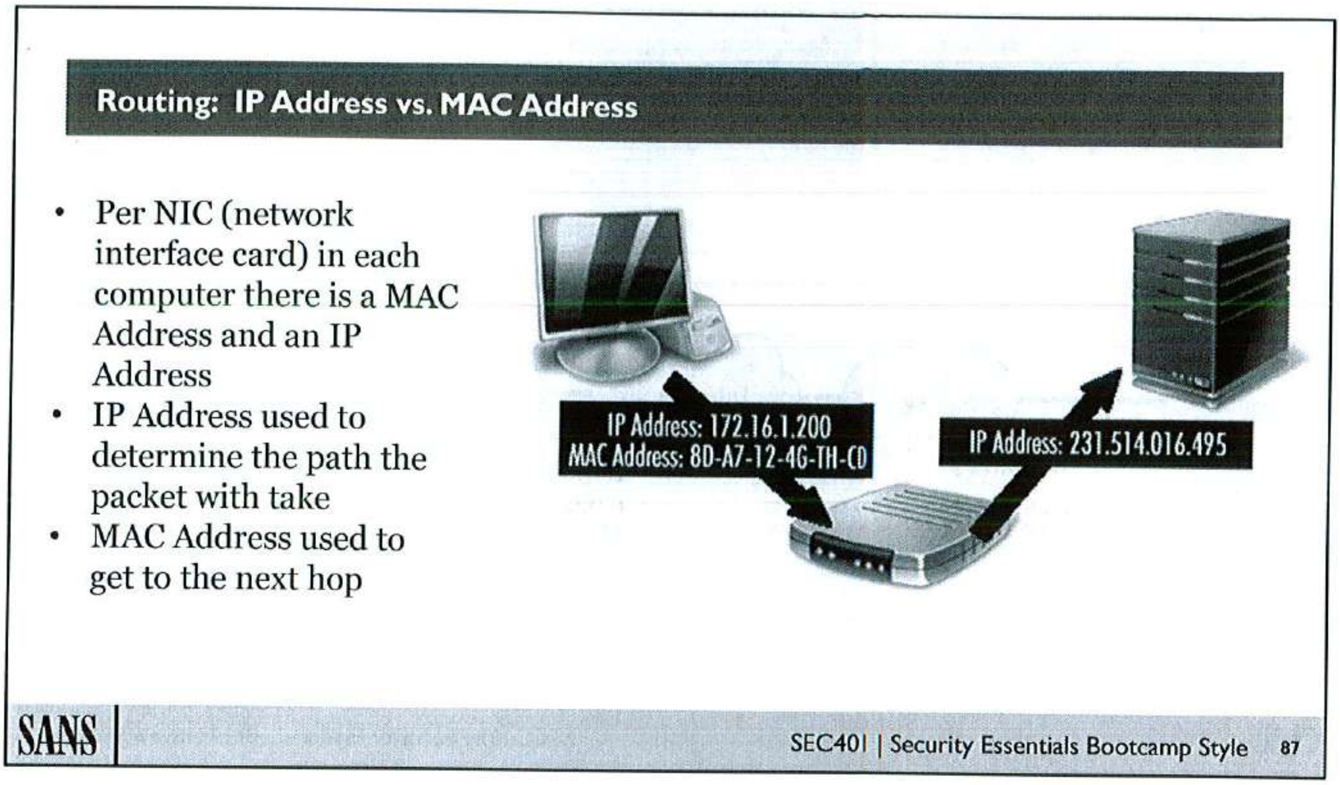
It is important to remember that network interfaces have two addresses associated with them, namely a hardware address and a software address.

A hardware address is the data link layer (Layer 2) address associated with the network interface. If the network is a Frame Relay network, for example, the hardware address is a 10-bit data link connection identifier (DLCI). If the host is attached to an IEEE LAN (such as 802.3/Ethernet), the hardware address is a 48-bit media access control (MAC) address. MAC addresses are uniquely allocated to each network interface card (NIC) and are not meant to change.

A MAC address typically is written as 12 hexadecimal digits grouped in pairs (bytes): 00-00-0c-34-17-a3 is an example of a typical MAC address. The first 24 bits of the address contain a vendor code, and the second half of the address is a unique number assigned by the vendor. MAC addresses generally are burned into NICs during the manufacturing process. Cisco Systems' vendor code, for example, is 00-00-0c, and Sun Microsystems' vendor code is 08-00-20. Readers can look up the MAC vendor codes at http://standards.ieee.org/regauth/oui/index.shtml.

A software address, such as an IP address, is the network layer protocol address. This address is specific to the network layer protocol and actual network to which the host is attached. If the computer supports multiple network layer protocols, the NIC has multiple software addresses.

An IP address is 32 bits or 4 bytes in length and is usually written in dotted-decimal format, such as 10.5.10.37. An IP address is hierarchical for routing purposes; the first part is the network identifier (NET_ID), and the second part is the host identifier (HOST_ID).



Routing: IP Address vs. MAC Address

- Per NIC (network interface card) in each computer there is a MAC Address and an IP Address
- IP Address used to determine the path the packet with take
- MAC Address used to get to the next hop

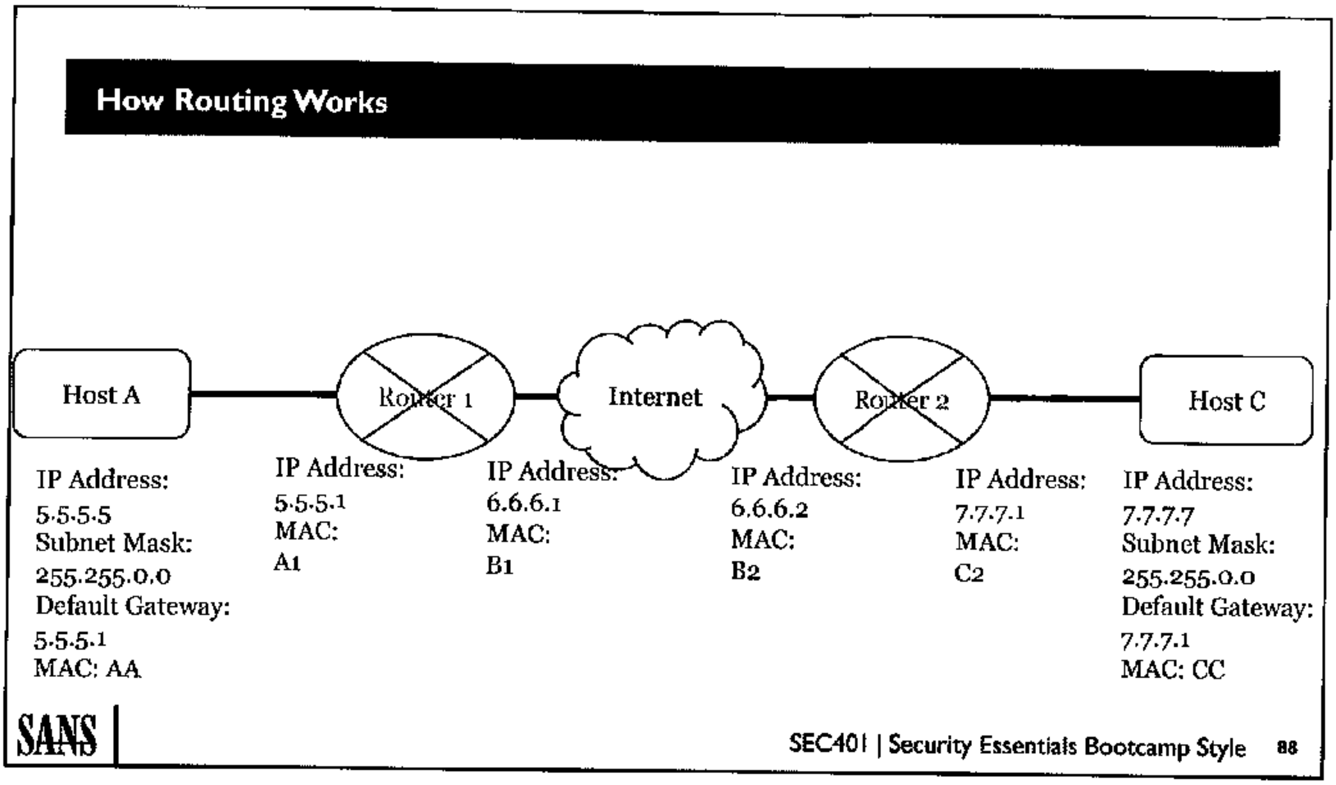
SANS SEC401 | Security Essentials Bootcamp Style 87

An IP address is a unique string of numbers separated by periods that is assigned to every device on a network so that device can be located on that network. The internet is just a very large network and every device connected to it has an IP address. "Address" is the operative word when thinking about IP addresses and in order for them to work correctly, there is no room for error—just as with a physical address for a house. Each part of the IP address is significant and necessary for a packet to reach its destination in much the same way that a zip code, city, street name and house number are all critical components for the delivery of a physical package.

A MAC (or Machine Access Control) address is best thought of as a type of serial number that is assigned to every network adapter. No two NIC's should have the same MAC address because if you had two network cards in the same network with the same MAC address it would cause severe problems with data arriving at the correct location. A good analogy for this would similar to being an employee in a company and having an employee's clone begin working there as well; it would create a lot of confusion very quickly. This fictitious clone could work in another company without any problems, but while the clone is in the original employee's company (or network) there would be constant confusion. While a MAC address is generally considered permanent, it is possible to change, spoof, or mask them.

References

1. http://searchnetworking.techtarget.com/tip/Hardening-your-router-in-9-easy-steps
2. http://packetpushers.net/cisco-ios-device-hardening/
3. http://stackoverflow.com/questions/23935095/how-are-mac-addresses-used-in-routing-packets
4. http://www.informit.com/articles/article.aspx?p=131034&seqNum=5
5. https://askleo.com/whats_the_difference_between_a_mac_address_and_an_ip_address/
6. http://www.erg.abdn.ac.uk/users/gorry/course/inet-pages/arp.html
7. http://whatismyipaddress.com/ip-basics



How Routing Works

Host A

Router 1

Internet

Router 2

Host C

IP Address:
5.5.5.5
Subnet Mask:
255.255.0.0
Default Gateway:
5.5.5.1
MAC: AA

IP Address:
5.5.5.1
MAC:
A1

IP Address:
6.6.6.1
MAC:
B1

IP Address:
6.6.6.2
MAC:
B2

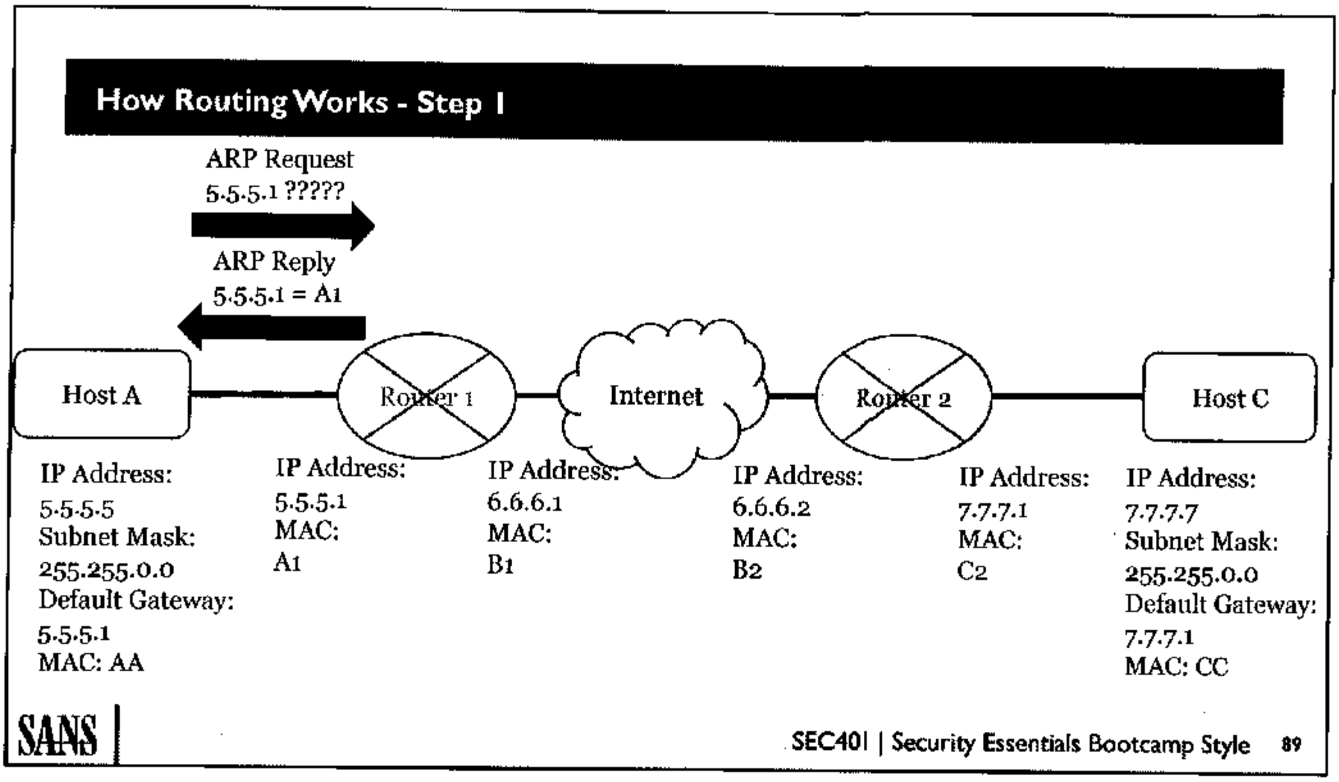
IP Address:
7.7.7.1
MAC:
C2

IP Address:
7.7.7.7
Subnet Mask:
255.255.0.0
Default Gateway:
7.7.7.1
MAC: CC

SANS

SEC401 | Security Essentials Bootcamp Style 88

In order to understand how routing works and the relationship between IP and MAC addresses, let's look at a simple diagram. In this network, Host A wants to communicate with Host C. Each computer is connected to a router which is connected via the Internet. In reality, there would also be a switch between each host and the router and multiple routers on the Internet, but we are intentionally keeping it simple to show how the process works. It is also important to note that the MAC addresses have been shortened for brevity.



How Routing Works - Step 1

IP Address:
5.5.5.5
Subnet Mask:
255.255.0.0
Default Gateway:
5.5.5.1
MAC: AA

IP Address:
5.5.5.1
MAC:
A1

IP Address:
6.6.6.1
MAC:
B1

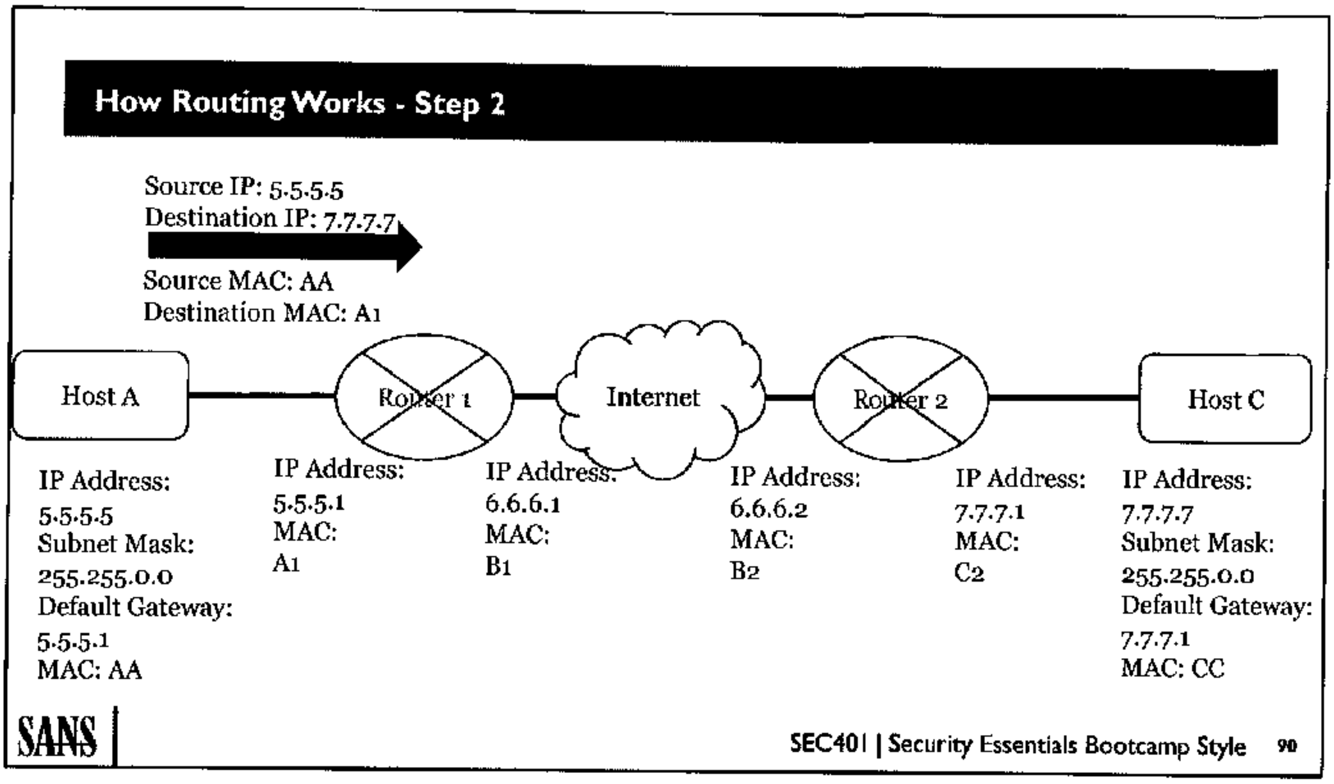
IP Address:
6.6.6.2
MAC:
B2

IP Address:
7.7.7.1
MAC:
C2

IP Address:
7.7.7.7
Subnet Mask:
255.255.0.0
Default Gateway:
7.7.7.1
MAC: CC

SANS

STEP 1: Using the subnet mask Host A determines that it is on the 5.5 network and Host C is on the 7.7 network. Since they are on different networks, Host A needs to send the packet to Router 1, its default gateway which is 5.5.5.1. At this point, Host A only knows the IP address of its default gateway. In order to connect to Router 1 at layer 2, it needs to know its MAC address. It looks in its ARP cache and there is not an entry. Therefore it has to send out an ARP request in order to get the MAC address of Router 1. Router 1 replies with the MAC address.



How Routing Works - Step 2

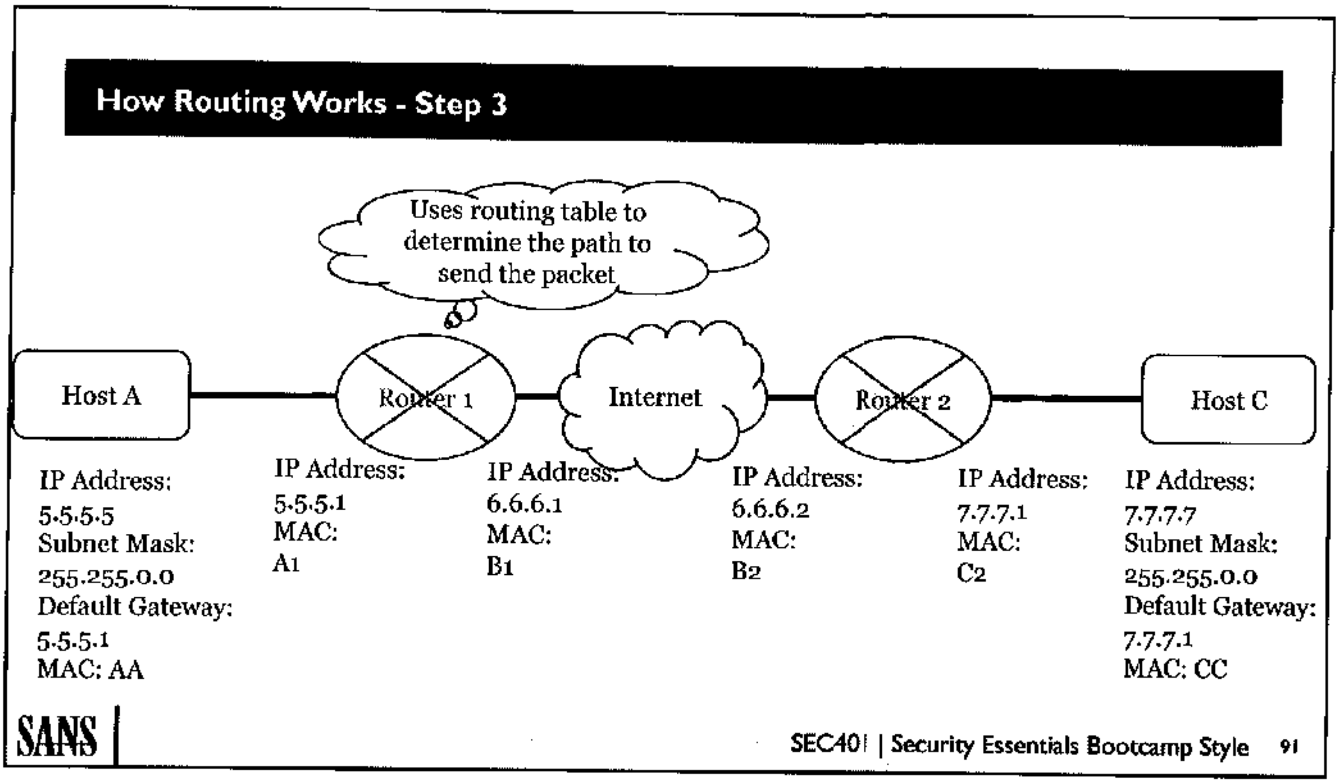
Source IP: 5.5.5.5
Destination IP: 7.7.7.7

Source MAC: AA
Destination MAC: A1

Host A	Router 1	Router 1	Router 2	Router 2	Host C
IP Address: 5.5.5.5 Subnet Mask: 255.255.0.0 Default Gateway: 5.5.5.1 MAC: AA	IP Address: 5.5.5.1 MAC: A1	IP Address: 6.6.6.1 MAC: B1	IP Address: 6.6.6.2 MAC: B2	IP Address: 7.7.7.1 MAC: C2	IP Address: 7.7.7.7 Subnet Mask: 255.255.0.0 Default Gateway: 7.7.7.1 MAC: CC

SANS | SEC401 | Security Essentials Bootcamp Style 90

STEP 2: Now that Host A knows the MAC Address of Router 1, it is ready to send out the information. The frame at layer 2 has the source MAC of Host A and the destination MAC of Router 1. The MAC address is used to get to the next hop. The packet at layer 3 has the source IP of Host A and the Destination IP of Host C. The IP Address is used to determine the path the packet is going to take.



How Routing Works - Step 3

Uses routing table to determine the path to send the packet

Host A

Router 1

Internet

Router 2

Host C

IP Address:
5.5.5.5
Subnet Mask:
255.255.0.0
Default Gateway:
5.5.5.1
MAC: AA

IP Address:
5.5.5.1
MAC:
A1

IP Address:
6.6.6.1
MAC:
B1

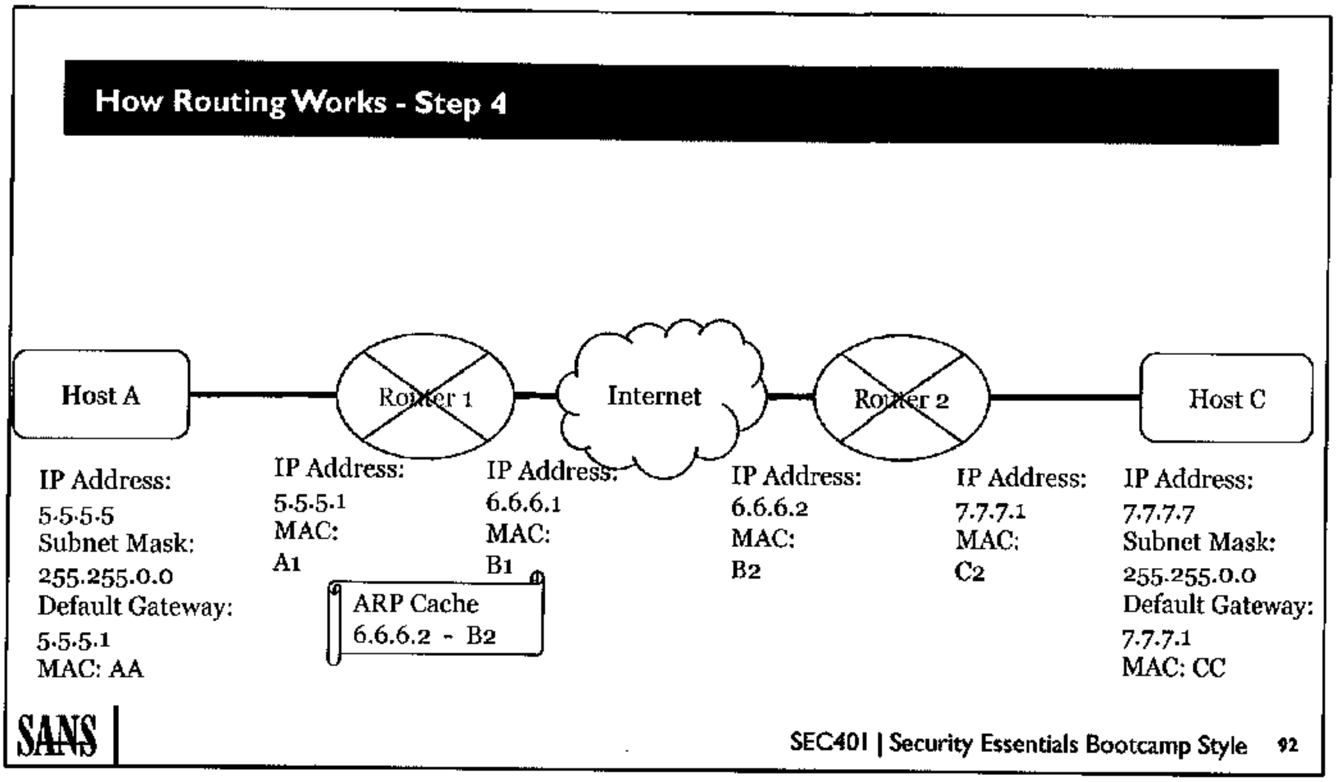
IP Address:
6.6.6.2
MAC:
B2

IP Address:
7.7.7.1
MAC:
C2

IP Address:
7.7.7.7
Subnet Mask:
255.255.0.0
Default Gateway:
7.7.7.1
MAC: CC

SANS

STEP 3: Once Router 1 receives the information it looks in its routing table to determine the path the packet needs to take. The routing table specifies that the way you get to the 7 network is by going through Router 2. Now Router 1 knows that it needs to send the packet to Router 2's IP Address which is 6.6.6.2



How Routing Works - Step 4

Host A — Router 1 — Internet — Router 2 — Host C

Host A:
IP Address:
5.5.5.5
Subnet Mask:
255.255.0.0
Default Gateway:
5.5.5.1
MAC: AA

Router 1:
IP Address:
5.5.5.1
MAC:
A1

IP Address:
6.6.6.1
MAC:
B1

ARP Cache
6.6.6.2 - B2

Router 2:
IP Address:
6.6.6.2
MAC:
B2

IP Address:
7.7.7.1
MAC:
C2

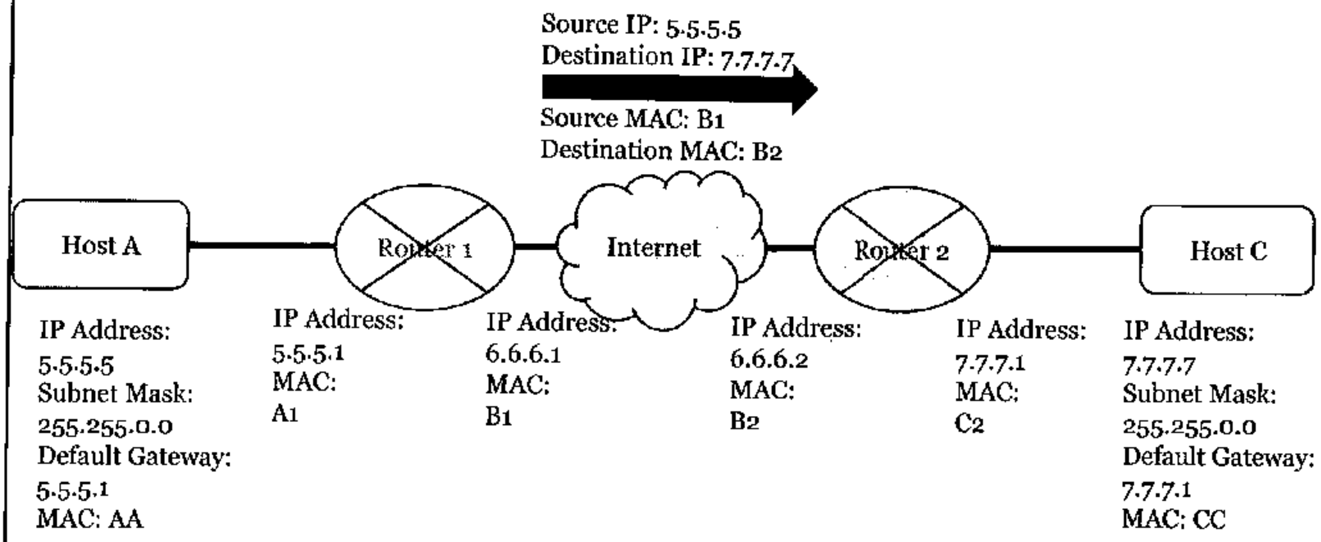
Host C:
IP Address:
7.7.7.7
Subnet Mask:
255.255.0.0
Default Gateway:
7.7.7.1
MAC: CC

SANS

SEC401 | Security Essentials Bootcamp Style 92

STEP 4: Once Router 1 knows that it needs to send the packet to Router 2's IP Address which is 6.6.6.2, it needs to know its MAC Address. Since these routers communicate often there is an entry in Router 1's ARP cache. It uses the ARP cache to determine that the MAC Address for Router 2 is B2.

How Routing Works - Step 5



Source IP: 5.5.5.5
Destination IP: 7.7.7.7

Source MAC: B1
Destination MAC: B2

Host A

Router 1

Internet

Router 2

Host C

IP Address:
5.5.5.5
Subnet Mask:
255.255.0.0
Default Gateway:
5.5.5.1
MAC: AA

IP Address:
5.5.5.1
MAC:
A1

IP Address:
6.6.6.1
MAC:
B1

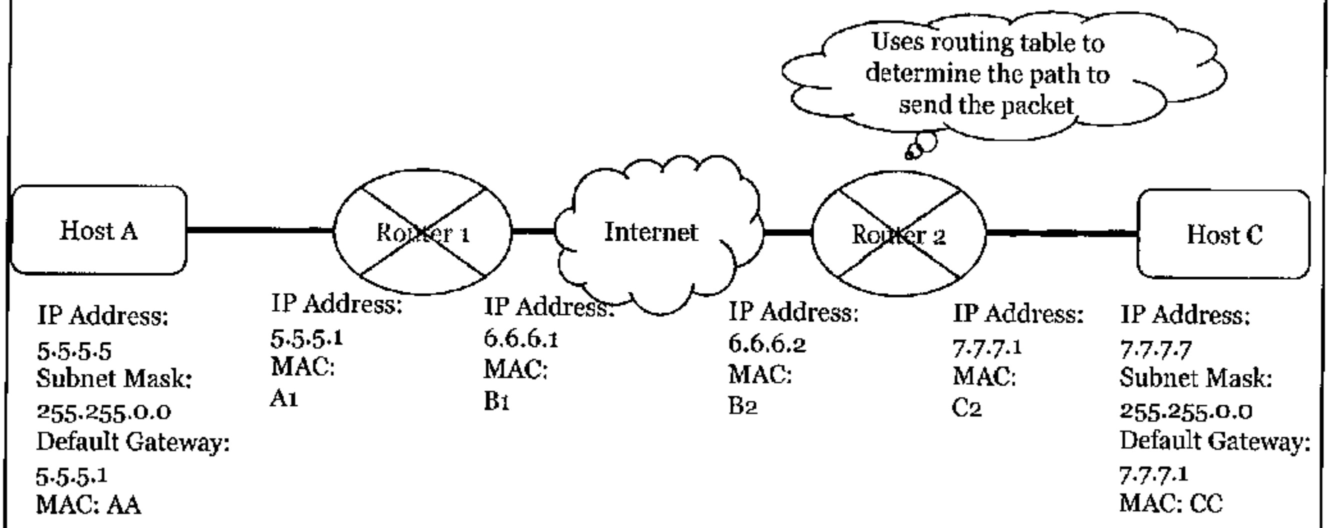
IP Address:
6.6.6.2
MAC:
B2

IP Address:
7.7.7.1
MAC:
C2

IP Address:
7.7.7.7
Subnet Mask:
255.255.0.0
Default Gateway:
7.7.7.1
MAC: CC

STEP 5: Now that Router 1 knows the MAC Address of Router 2, it is ready to send out the information. The frame at layer 2 has the source MAC of Router 1's external interface and the destination MAC of Router 2's external interface. The MAC address is used to get to the next hop. The packet at layer 3 has the source IP of Host A and the Destination IP of Host C. The IP Address is used to determine the path the packet is going to take.

How Routing Works - Step 6



IP Address:
5.5.5.5
Subnet Mask:
255.255.0.0
Default Gateway:
5.5.5.1
MAC: AA

IP Address:
5.5.5.1
MAC:
A1

IP Address:
6.6.6.1
MAC:
B1

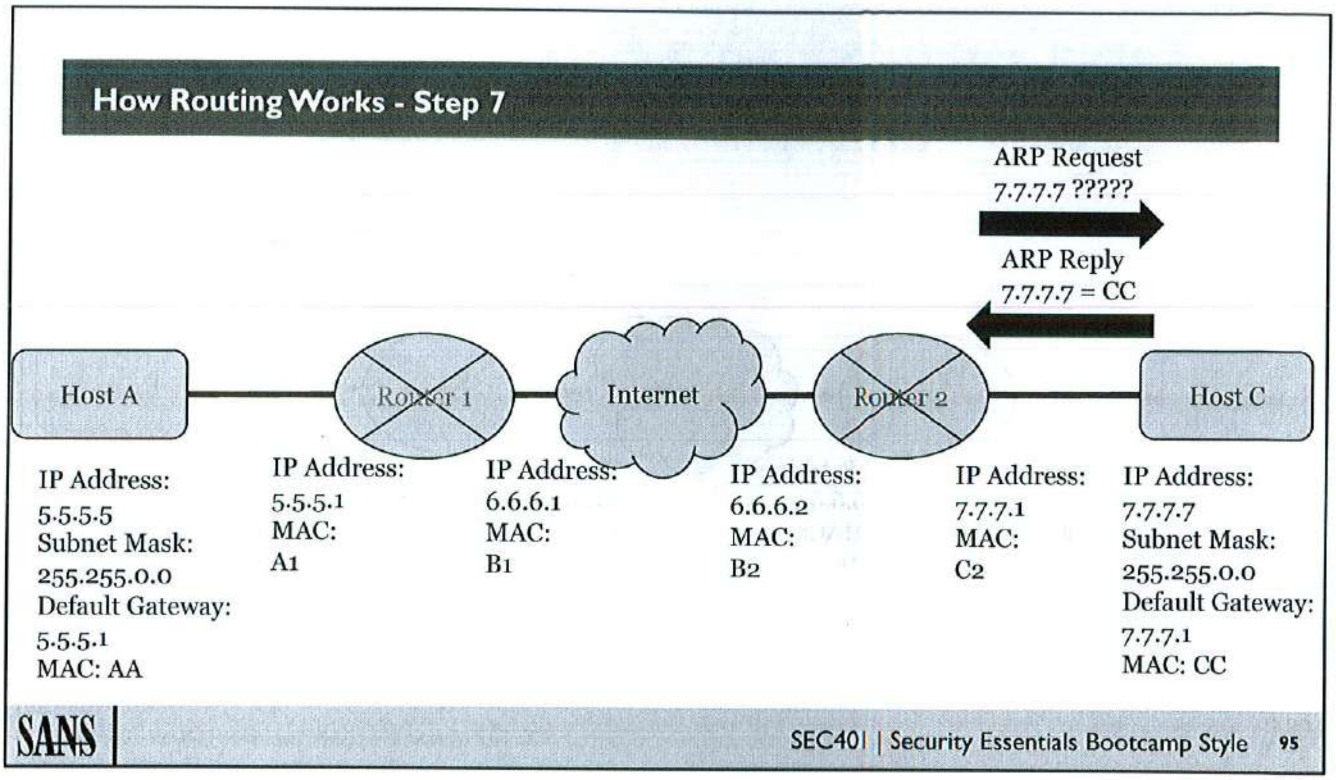
IP Address:
6.6.6.2
MAC:
B2

IP Address:
7.7.7.1
MAC:
C2

IP Address:
7.7.7.7
Subnet Mask:
255.255.0.0
Default Gateway:
7.7.7.1
MAC: CC

SANS

STEP 6: Once Router 2 receives the information it looks in its routing table to determine the path the packet needs to take. The routing table specifies that the way you get to the 7 network is by directly sending it to Host C. Now Router 2 knows that it needs to send the packet to Host C's IP Address which is 7.7.7.7



How Routing Works - Step 7

ARP Request
7.7.7.7 ?????

ARP Reply
7.7.7.7 = CC

Host A — Router 1 — Internet — Router 2 — Host C

Host A:
IP Address:
5.5.5.5
Subnet Mask:
255.255.0.0
Default Gateway:
5.5.5.1
MAC: AA

Router 1:
IP Address:
5.5.5.1
MAC:
A1

IP Address:
6.6.6.1
MAC:
B1

Router 2:
IP Address:
6.6.6.2
MAC:
B2

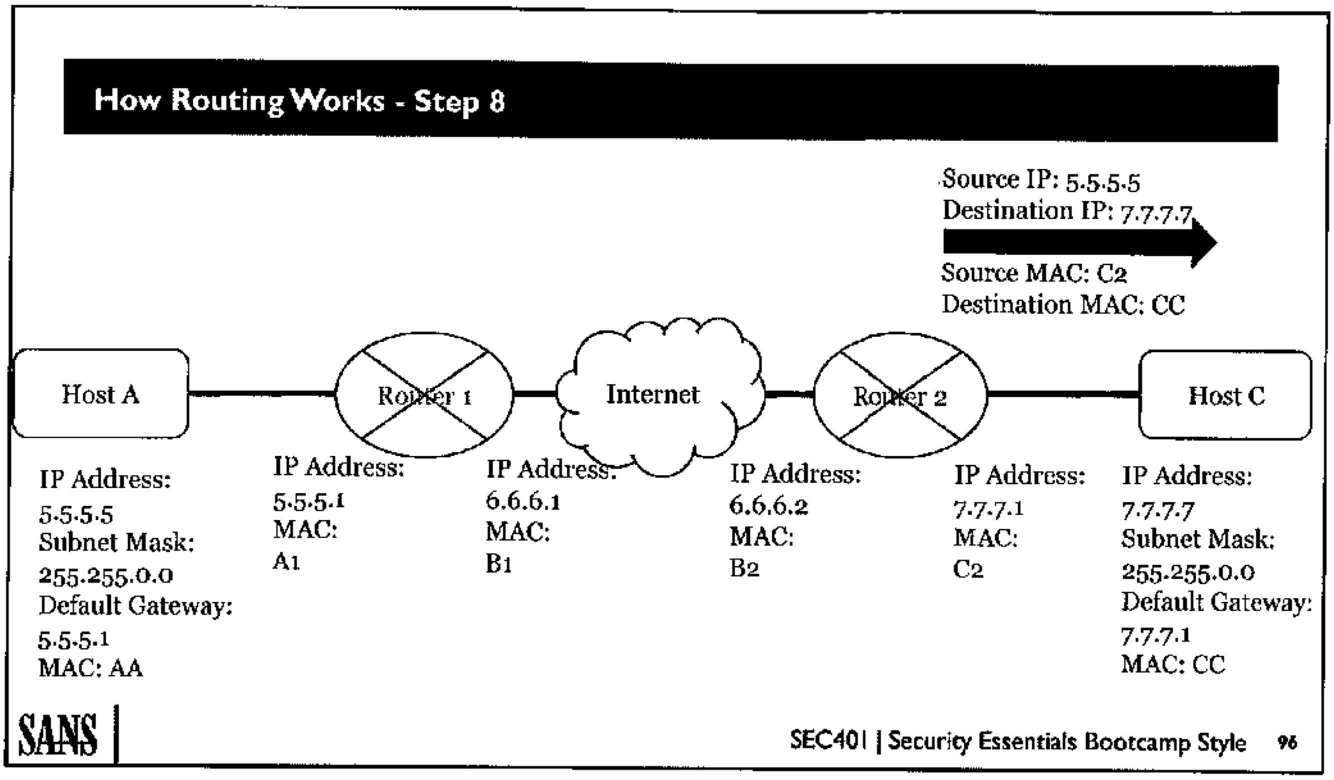
IP Address:
7.7.7.1
MAC:
C2

Host C:
IP Address:
7.7.7.7
Subnet Mask:
255.255.0.0
Default Gateway:
7.7.7.1
MAC: CC

SANS

SEC401 | Security Essentials Bootcamp Style 95

STEP 7: At this point Router 2 only knows the IP address of Host C. In order to connect to Host C at layer 2, it needs to know it MAC address. It looks in its ARP cache and there is not an entry. Therefore it has to send out an ARP request in order to get the MAC address of Host C. Host C replies with the MAC address.



How Routing Works - Step 8

Source IP: 5.5.5.5
Destination IP: 7.7.7.7

Source MAC: C2
Destination MAC: CC

Host A

Router 1

Internet

Router 2

Host C

IP Address:
5.5.5.5
Subnet Mask:
255.255.0.0
Default Gateway:
5.5.5.1
MAC: AA

IP Address:
5.5.5.1
MAC:
A1

IP Address:
6.6.6.1
MAC:
B1

IP Address:
6.6.6.2
MAC:
B2

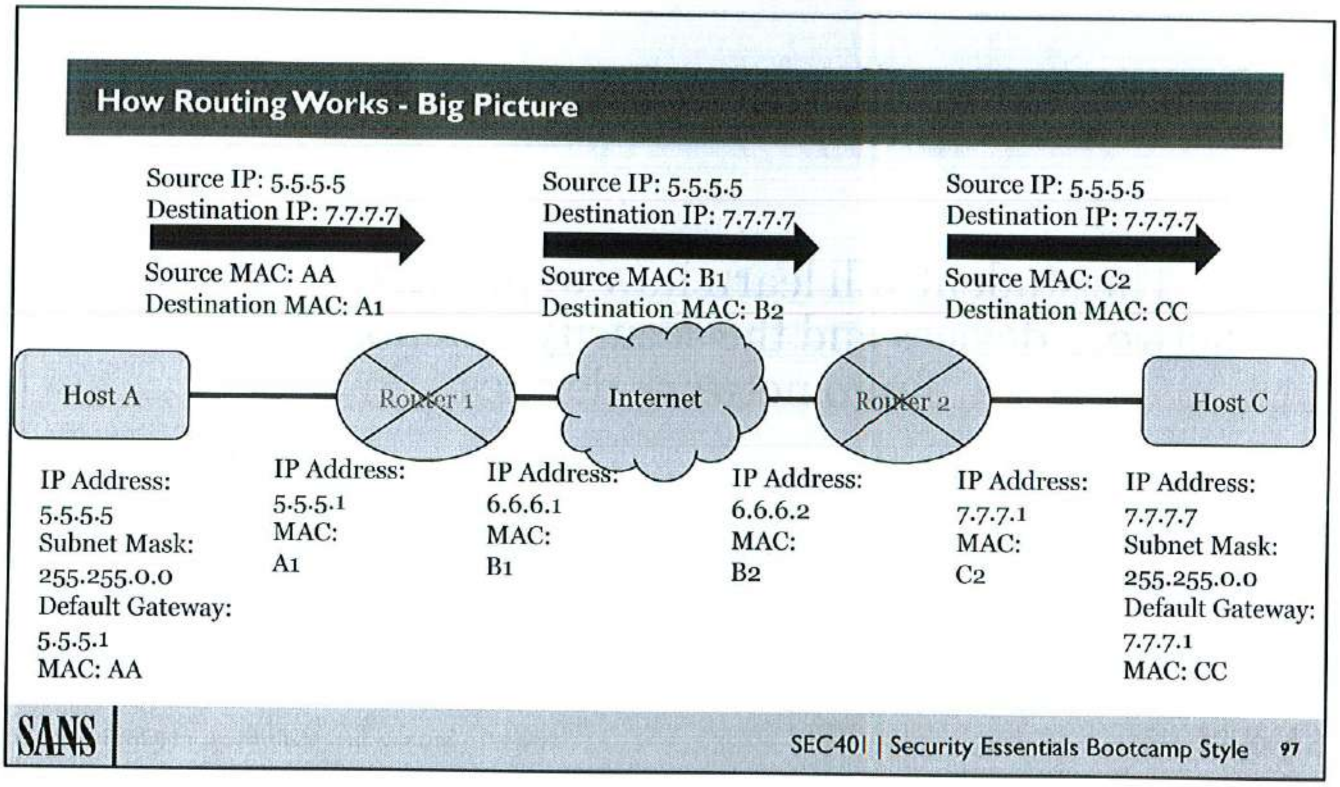
IP Address:
7.7.7.1
MAC:
C2

IP Address:
7.7.7.7
Subnet Mask:
255.255.0.0
Default Gateway:
7.7.7.1
MAC: CC

SANS

SEC401 | Security Essentials Bootcamp Style 96

STEP 8: Now that Router 2 knows the MAC Address of Host C, it is ready to send out the information. The frame at layer 2 has the source MAC of Router 2's internal interface and the destination MAC of Host C. The MAC address is used to get to the next hop. The packet at layer 3 has the source IP of Host A and the Destination IP of Host C. The IP Address is used to determine the path the packet is going to take.



How Routing Works - Big Picture

Source IP: 5.5.5.5
Destination IP: 7.7.7.7
Source MAC: AA
Destination MAC: A1

Source IP: 5.5.5.5
Destination IP: 7.7.7.7
Source MAC: B1
Destination MAC: B2

Source IP: 5.5.5.5
Destination IP: 7.7.7.7
Source MAC: C2
Destination MAC: CC

IP Address: 5.5.5.5
Subnet Mask: 255.255.0.0
Default Gateway: 5.5.5.1
MAC: AA

IP Address: 5.5.5.1
MAC: A1

IP Address: 6.6.6.1
MAC: B1

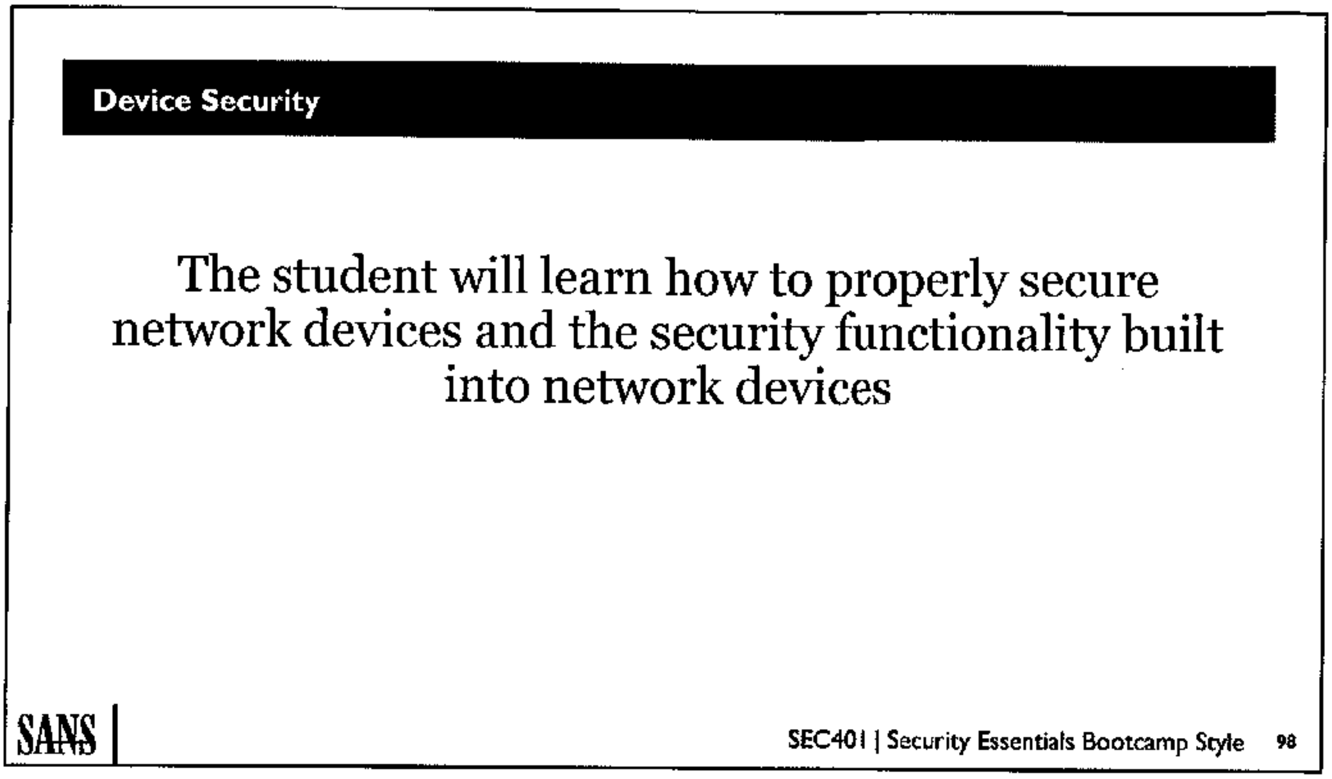
IP Address: 6.6.6.2
MAC: B2

IP Address: 7.7.7.1
MAC: C2

IP Address: 7.7.7.7
Subnet Mask: 255.255.0.0
Default Gateway: 7.7.7.1
MAC: CC

SANS | SEC401 | Security Essentials Bootcamp Style 97

This slide shows the big picture of how routing works. It is important to note that the IP Address stays the same to determine the path but the MAC Address is constantly changing to get to the next hop. This is how the process works on the Internet; the only difference is there are typically more than two routers between two computers. Otherwise, the process would work in the manner illustrated in this section.

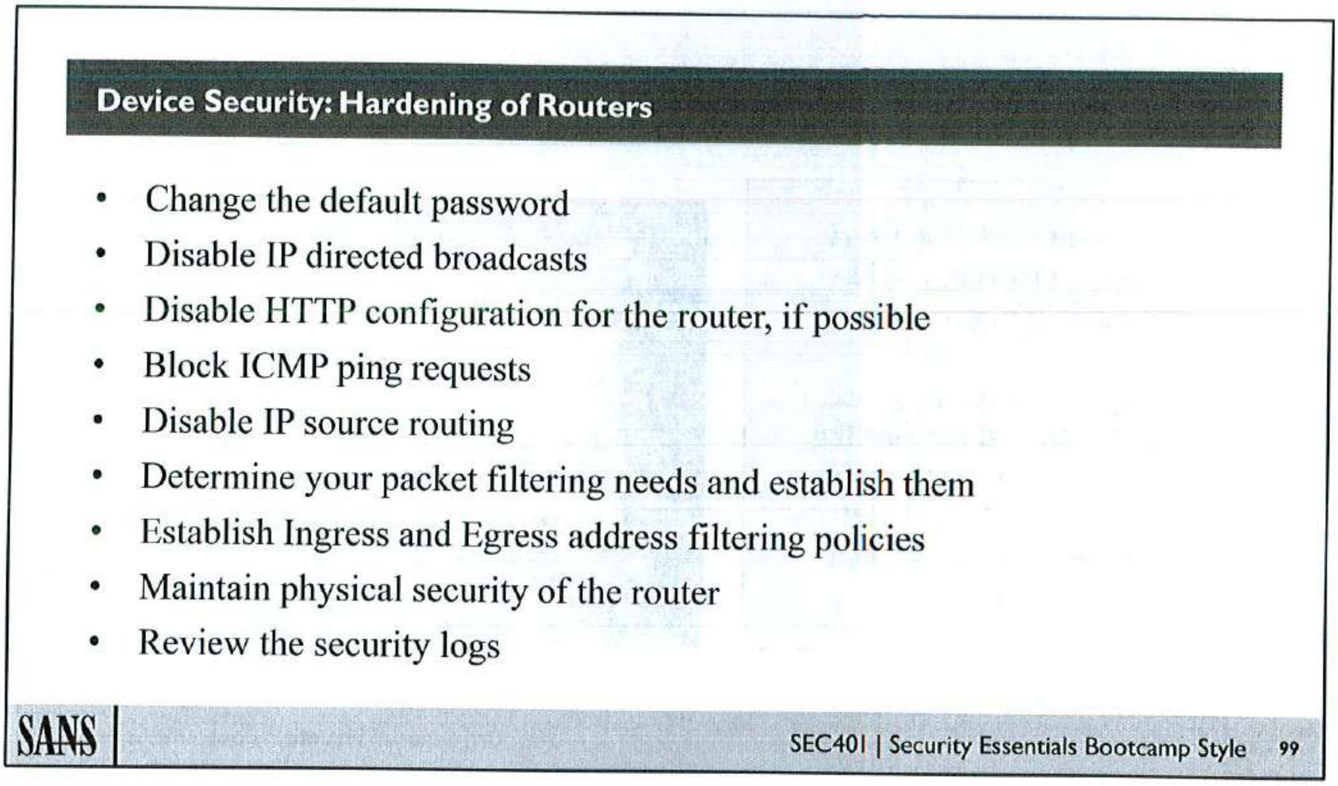


Device Security

The student will learn how to properly secure network devices and the security functionality built into network devices

SANS

This page intentionally left blank.



Device Security: Hardening of Routers

- Change the default password
- Disable IP directed broadcasts
- Disable HTTP configuration for the router, if possible
- Block ICMP ping requests
- Disable IP source routing
- Determine your packet filtering needs and establish them
- Establish Ingress and Egress address filtering policies
- Maintain physical security of the router
- Review the security logs

SANS SEC401 | Security Essentials Bootcamp Style 99

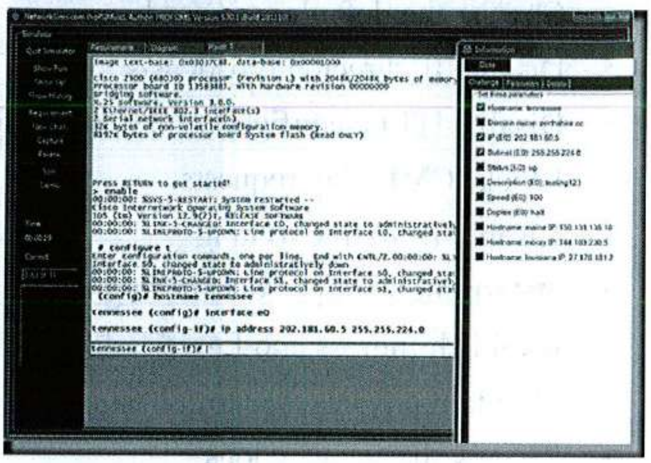
Hardening is usually the process of securing a system by reducing its surface of vulnerability, which is larger when a system performs more functions. In theory, a single-function system is more secure than a multipurpose one.

Routers have few vulnerabilities for would-be attackers to exploit because the devices tend to be less complicated than traditional operating systems. However, they do have some potential vulnerabilities. In order to protect against these vulnerabilities you will want to:
Change the default password
Disable IP directed broadcasts
Disable HTTP configuration for the router, if possible
Block ICMP ping requests
Disable IP source routing
Determine your packet filtering needs and establish them
Establish Ingress and Egress address filtering policies
Maintain physical security of the router
Review the security logs

Hardening of Routers: Latest IOS

An important part of network hardening is ensuring you have the latest IOS version.

Hardening is an ongoing process of ensuring that all networking software and router firmware are updated with the latest vendor supplied patches and fixes.



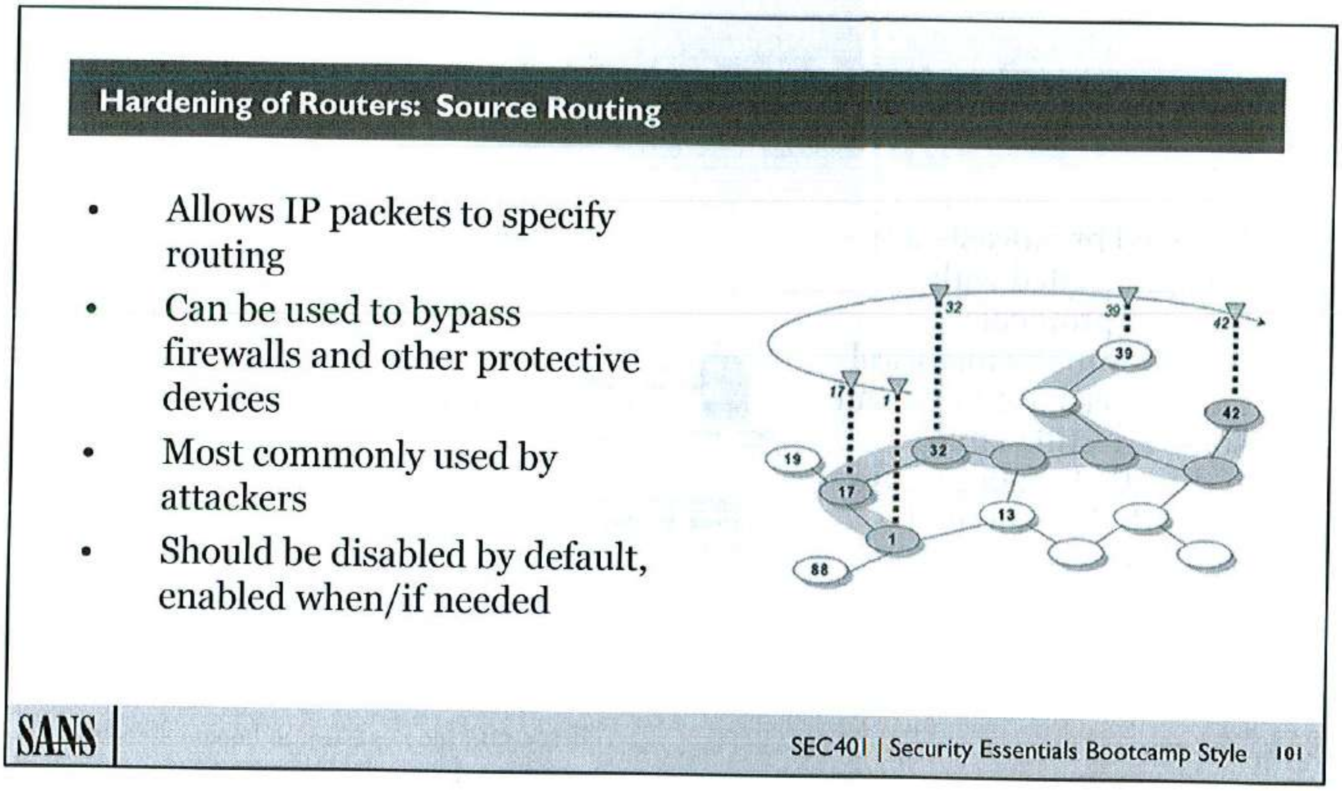
```
Image text-base: 0x0303?C48, data-base: 0x00001000

cisco 2500 (68030) processor (revision L) with 2048K/2048K bytes of memor
Processor board ID 13583887, with hardware revision 00000000
Bridging software.
X.25 software, Version 3.0.0.
2 Ethernet/IEEE 802.3 interface(s)
2 Serial network interface(s)
32K bytes of non-volatile configuration memory.
8192K bytes of processor board System flash (Read ONLY)



Press RETURN to get started!
> enable
00:00:00: %SYS-5-RESTART: System restarted --
Cisco Internetwork Operating System Software
IOS (tm) Version 12.9(7)T, RELEASE SOFTWARE
00:00:00: %LINK-5-CHANGED: Interface E0, changed state to administratively
00:00:00: %LINEPROTO-5-UPDOWN: Line protocol on Interface E0, changed sta

 # configure t
Enter configuration commands, one per line.  End with CNTL/Z.00:00:00: %L
Interface S0, changed state to administratively down
00:00:00: %LINEPROTO-5-UPDOWN: Line protocol on Interface S0, changed sta
00:00:00: %LINK-5-CHANGED: Interface S1, changed state to administratively
00:00:00: %LINEPROTO-5-UPDOWN: Line protocol on Interface S1, changed sta
 (config)# hostname tennessee

tennessee (config)# interface e0

tennessee (config-if)# ip address 202.181.60.5 255.255.224.0

tennessee (config-if)# |
```

An important part of network hardening is ensuring you have the latest IOS version. Hardening is an ongoing process of ensuring that all networking software and router firmware are updated with the latest vendor supplied patches and fixes. An IOS, or any operating system for that matter, should be viewed as a perishable commodity. Once it's out in the world, there will immediately be people using it and potentially finding flaws and bugs or people actively looking to exploit it. The more time that passes, the more will be learned about the strengths and weaknesses of the system. Fortunately, knowledge of these strengths and weaknesses also reaches the manufacturer. They, then, repair deficiencies (hopefully) and continue to make improvements. Cisco, as an example, ships an IOS on every device. That IOS is never current due to the time spent being shipped and then distributed. If the router has been warehoused for a while, the IOS can be very old.

It is important to note that while Cisco was used in this example, this is true for most router vendors.

References

1. https://supportforums.cisco.com/discussion/12257556/what-ip-source-route
2. http://www.juniper.net/documentation/en_US/junos/topics/concept/ip-directed-broadcast-ex-series.html
3. http://www.dslreports.com/forum/r10416483-closing-down-unnecessary-ports
4. http://www.wikihow.com/Close-Port-21



Hardening of Routers: Source Routing

- Allows IP packets to specify routing
- Can be used to bypass firewalls and other protective devices
- Most commonly used by attackers
- Should be disabled by default, enabled when/if needed

32
39
42
17
1
39
42
32
19
17
13
1
88

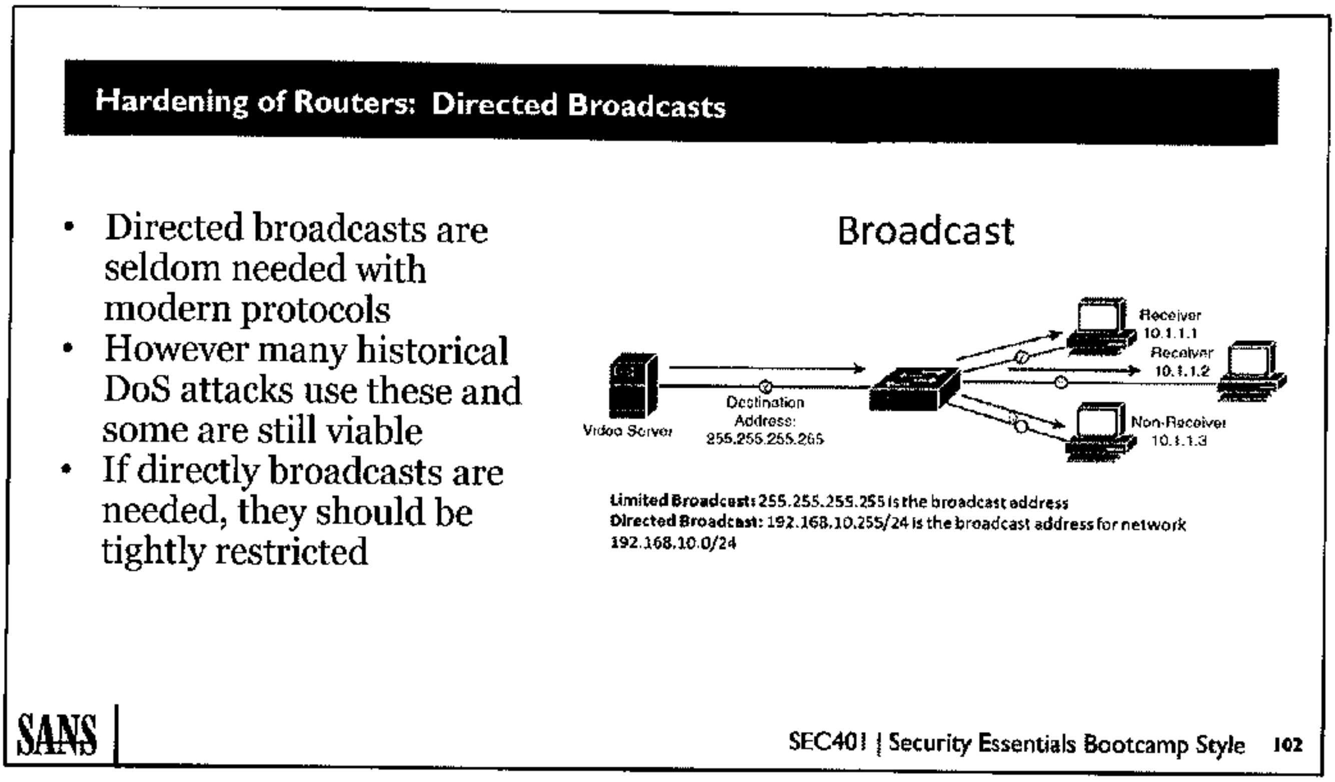
SANS

Source routing is information in an IP header that lets the source host dictate the path the packet uses to get to its destination. If the path were determined by intermediate gateways it could allow a source to go around security devices that are typically in the path between source and destination.

Source routing in the real world would be much like driving cross country in such a manner that is most efficient to the driver, such as through the use of interstates and highways. However, if certain interstate areas are known to have a high concentration of police and security checkpoints, a nefarious person wishing to not get caught could create a route that avoids these major checkpoints while still arriving at their desired destination.

References

1. https://supportforums.cisco.com/discussion/12257556/what-ip-source-route
2. http://www.juniper.net/documentation/en_US/junos/topics/concept/ip-directed-broadcast-ex-series.html
3. http://www.dslreports.com/forum/r10416483-closing-down-unnecessary-ports
4. http://www.wikihow.com/Close-Port-21



Hardening of Routers: Directed Broadcasts

- Directed broadcasts are seldom needed with modern protocols
- However many historical DoS attacks use these and some are still viable
- If directly broadcasts are needed, they should be tightly restricted

Broadcast

Video Server
Destination Address: 255.255.255.255
Receiver 10.1.1.1
Receiver 10.1.1.2
Non-Receiver 10.1.1.3

Limited Broadcast: 255.255.255.255 is the broadcast address
Directed Broadcast: 192.168.10.255/24 is the broadcast address for network 192.168.10.0/24

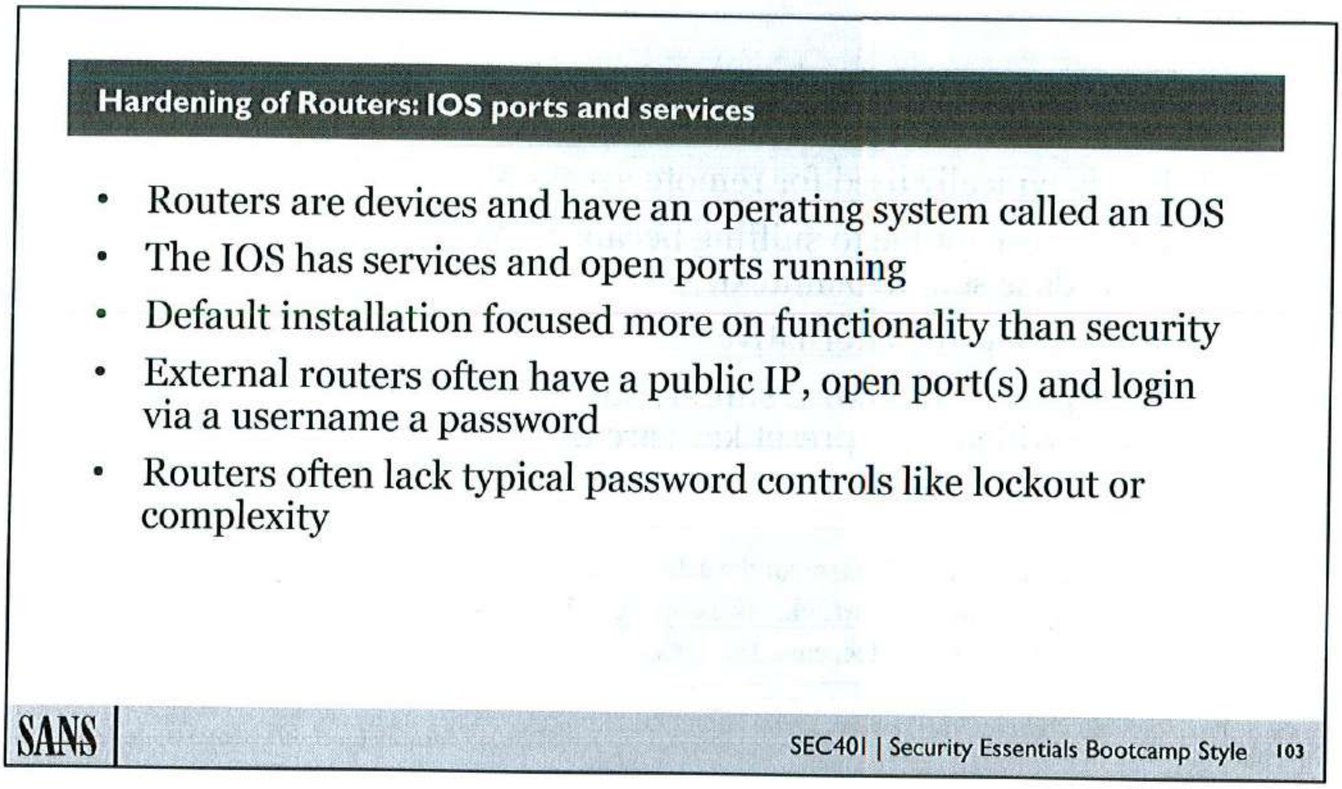
SANS

SEC401 | Security Essentials Bootcamp Style 102

IP directed broadcast helps you implement remote administration tasks. This is an IP address that speaks to "all hosts" on a particular network similar to a public address system inside a high school. If an announcement is made to "all faculty and students" everyone in the school knows to listen to the message regardless of the fact that each individual in the school is unique. The method of how IP directed broadcast works is by flooding the target subnet with broadcast packets but without broadcasting to the entire network. To continue with our public address analogy, a subnet could be considered a particular school within a larger school district which would be the entire network. The public address announcement only goes to that particular school and not to every school in the school district.

References

1. https://supportforums.cisco.com/discussion/12257556/what-ip-source-route
2. http://www.juniper.net/documentation/en_US/junos/topics/concept/ip-directed-broadcast-ex-series.html
3. http://www.dslreports.com/forum/r10416483-closing-down-unnecessary-ports
4. http://www.wikihow.com/Close-Port-21



Hardening of Routers: IOS ports and services

- Routers are devices and have an operating system called an IOS
- The IOS has services and open ports running
- Default installation focused more on functionality than security
- External routers often have a public IP, open port(s) and login via a username a password
- Routers often lack typical password controls like lockout or complexity

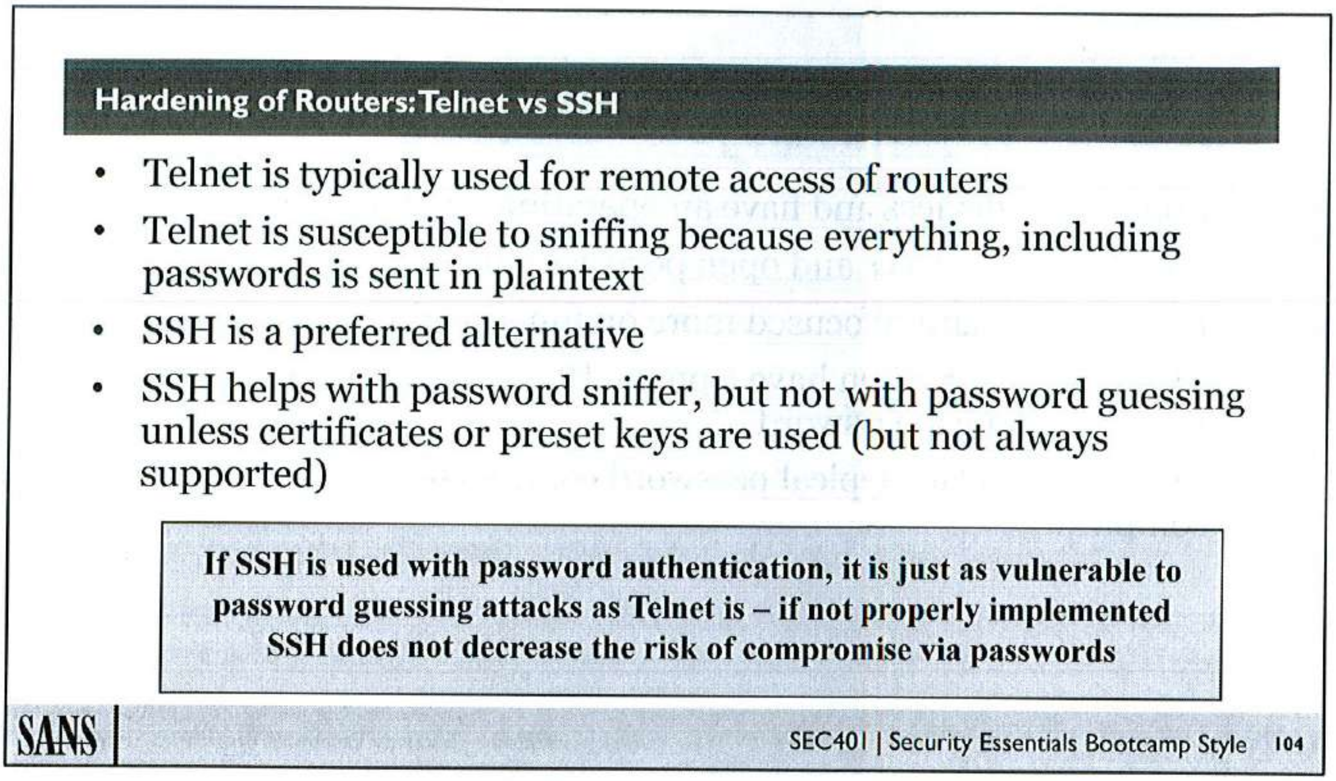
SANS

SEC401 | Security Essentials Bootcamp Style 103

Any device that is connected to a network typically has an operating system that controls how it functions and operates. In order for devices to communicate over a network, they need to have an IP address and open ports. Open ports with a visible IP address become an avenue of attack and potential compromise. Since most vendors focus on functionality, a default installation of an operating system is often not very secure because it has extraneous ports open and services running.

Routers and switches are no different than a computer and have an operating system called an IOS. The IOS not only allows for the device to perform its functionality, but often allows remote access. Since routers connect different networks together, most organization have a router that connects their organization's network with the Internet. For remote access for administrative purposes, routers often have public IP addresses and open ports for connectivity. This provides an avenue of attack for an adversary.

While this access is often based off of a password, many routers do not have robust password controls in place. For example, many routers do not enforce password complexity, password expiration or password lockout. Plus, it is common for an organization to use the same password for multiple routers and multiple sites. In addition, since many routers do not have robust logging, organizations typically have minimal to no visibility into whether someone is trying to compromise a router.



Hardening of Routers: Telnet vs SSH

- Telnet is typically used for remote access of routers
- Telnet is susceptible to sniffing because everything, including passwords is sent in plaintext
- SSH is a preferred alternative
- SSH helps with password sniffer, but not with password guessing unless certificates or preset keys are used (but not always supported)

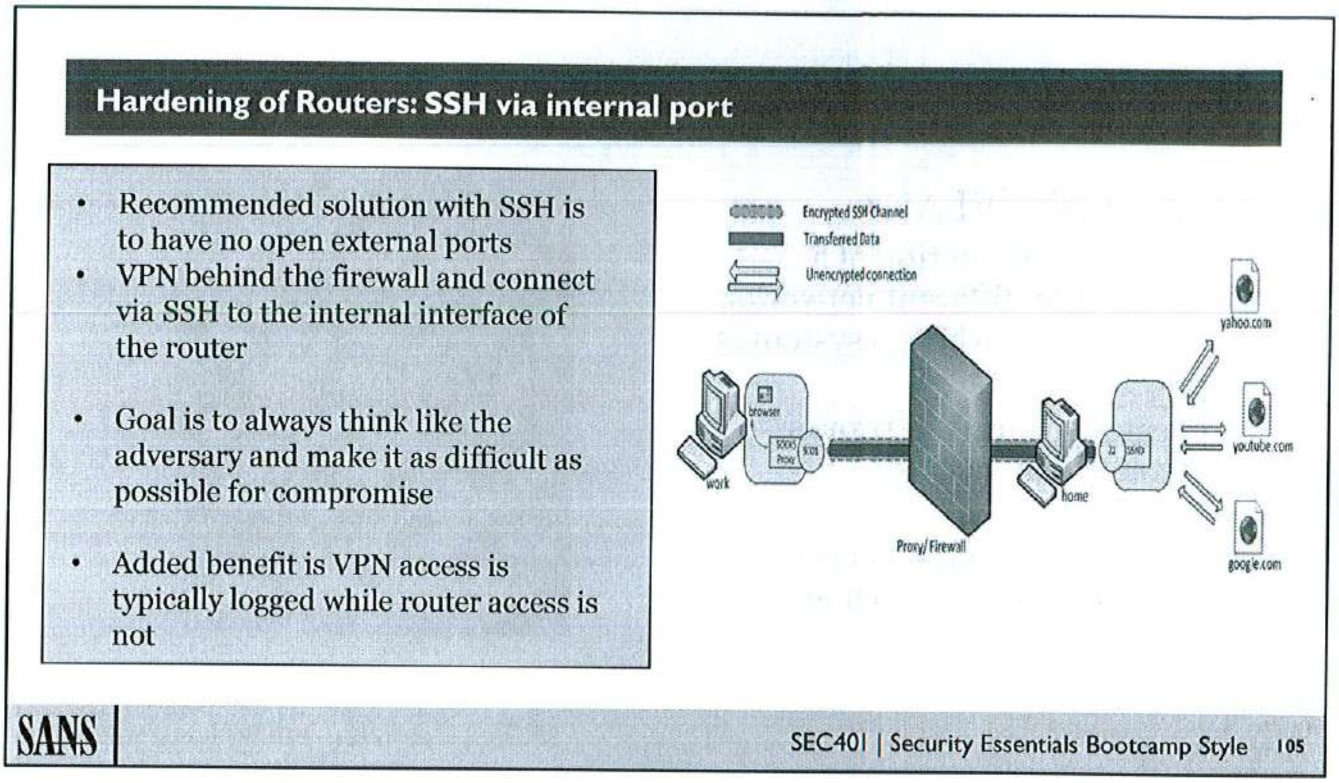
If SSH is used with password authentication, it is just as vulnerable to password guessing attacks as Telnet is – if not properly implemented SSH does not decrease the risk of compromise via passwords

SANS | SEC401 | Security Essentials Bootcamp Style 104

Most routers were typically configured to allow remote access via telnet. Telnet has often gotten criticism because everything is sent plaintext. Therefore if someone is running a sniffer and capturing a valid authentication request, an adversary can capture the password and access the router remotely. Based on this risk it is often recommended to use SSH to remotely access routers because all of the information is now encrypted. This presents two potential issues and challenges. First, in order to run SSH, cryptographic libraries have to be installed and configured. Since some smaller routers have limited hardware and limited computational power, this presents a challenge in some cases. Running SSH is not always an option on all routers.

The second issue is that just switching to SSH often does not solve the fundamental problem. The fundamental problem with accessing routers remotely is not that an adversary can sniff the password, but that an adversary can brute force the password because passwords are often not very complex, not changed very often, do not have lockout in place and are the same across an organization. Therefore merely switching from Telnet to SSH still using password authentication gives organizations a false sense of security, but does not fix the problem.

In order to fix the password problem with SSH either digital certificates or pre-shared keys need to be utilized. The problem is that this adds additional resources to the router. However, in most cases, if a router can support and implement SSH, it can usually support certificates or pre-shared keys.



Hardening of Routers: SSH via internal port

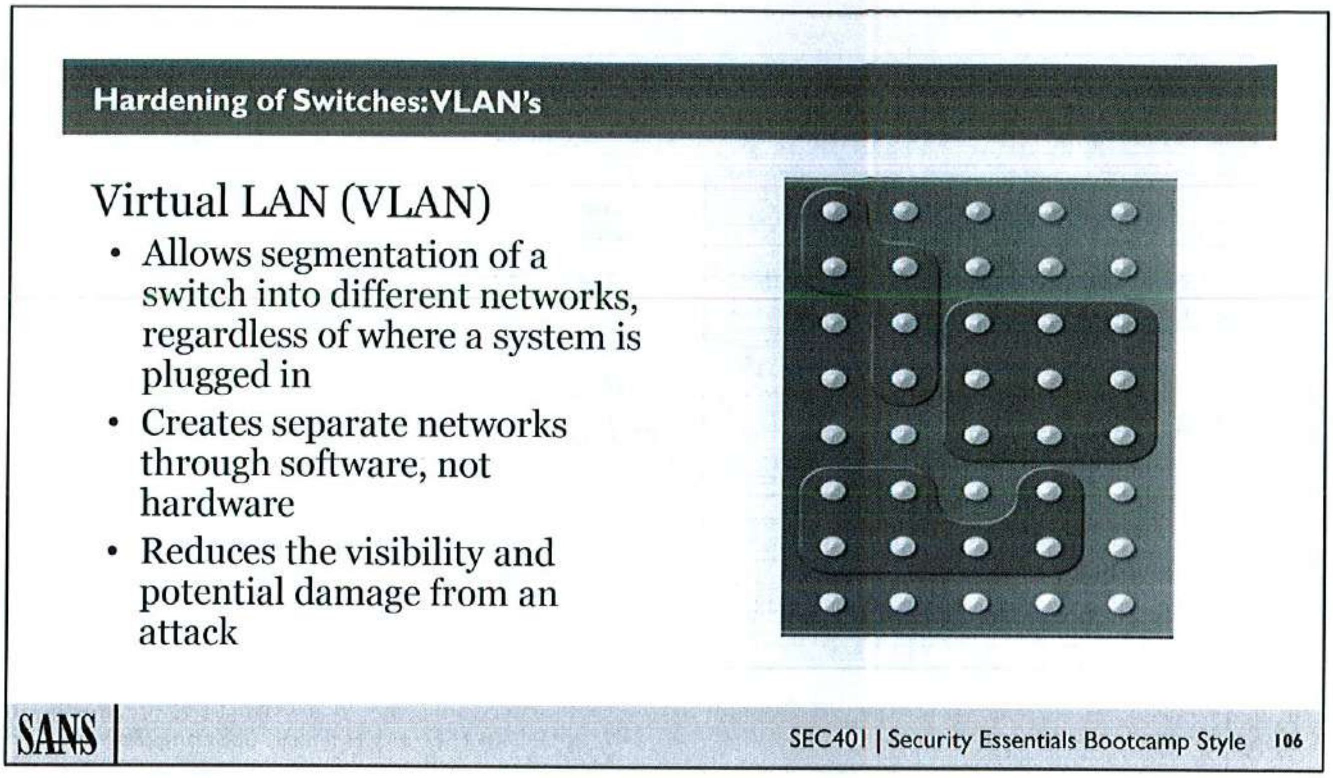
- Recommended solution with SSH is to have no open external ports
- VPN behind the firewall and connect via SSH to the internal interface of the router
- Goal is to always think like the adversary and make it as difficult as possible for compromise
- Added benefit is VPN access is typically logged while router access is not

While using pre-shared keys and/or certificates with SSH helps, there is still an open port from the Internet. An open port is still a security risk, even if additional protections are put in place. The best option is to have no open ports from the Internet. The initial reaction is but the router needs to be remotely accessible and you cannot access it remotely if there is no open port. That is actually not true. The trick is that most organizations have a VPN server on a DMZ which is behind the router. Therefore, the solution is to VPN into the organization and then, from the VPN, connect to an open port on the internal interface of the router. This provides an easy solution for administrators and makes life more difficult for the adversary.

The added benefit is that this allows for early detection because most VPN servers are logged, while most routers are not. Now if all access comes through a controlled VPN server with full logging and monitoring, even if an adversary tries to break in, there is a more likely chance of detection.

References:

1. https://supportforums.cisco.com/discussion/12257556/what-ip-source-route
2. http://www.juniper.net/documentation/en_US/junos/topics/concept/ip-directed-broadcast-ex-series.html
3. http://www.dslreports.com/forum/r10416483-closing-down-unnecessary-ports
4. http://www.wikihow.com/Close-Port-21



Hardening of Switches: VLAN's

Virtual LAN (VLAN)

- Allows segmentation of a switch into different networks, regardless of where a system is plugged in
- Creates separate networks through software, not hardware
- Reduces the visibility and potential damage from an attack

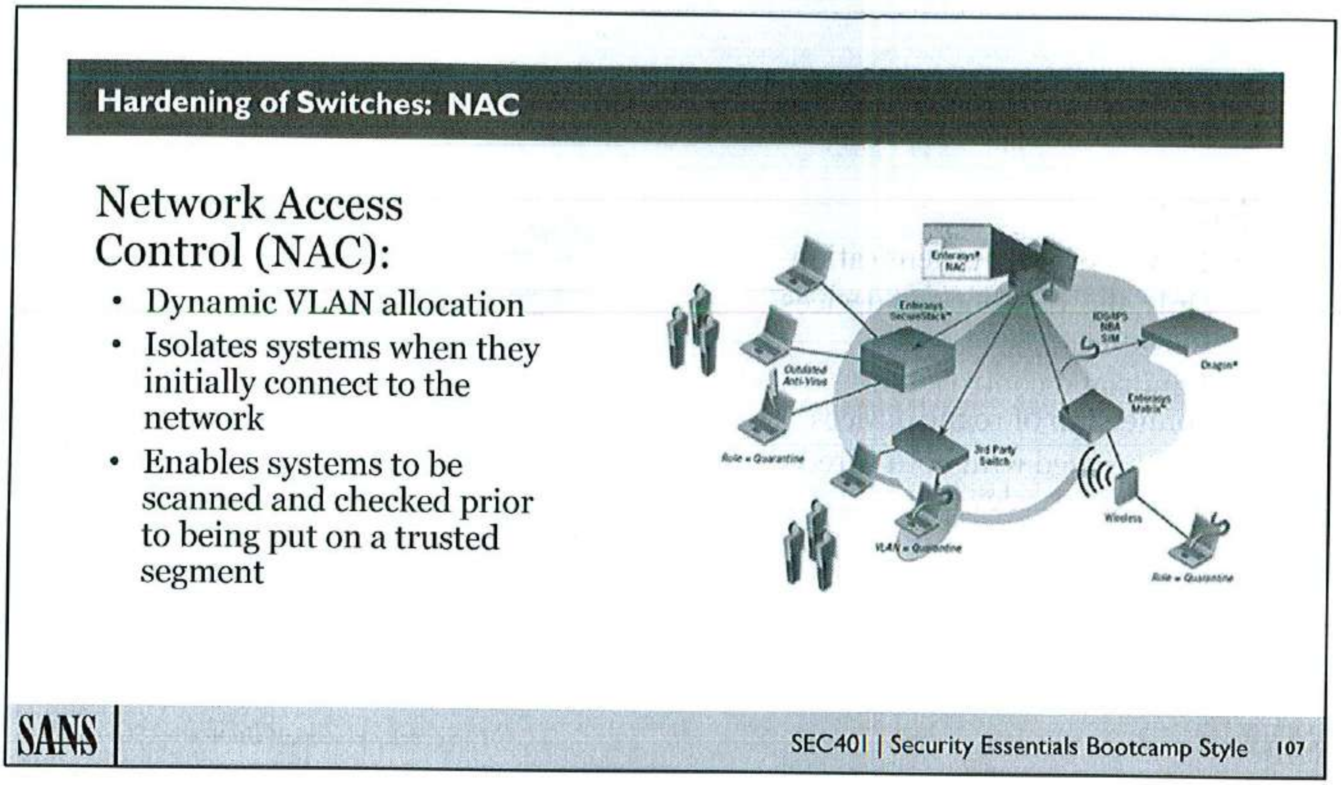
SANS

SEC401 | Security Essentials Bootcamp Style 106

Networks continue to be an avenue that malicious code uses to not only attack systems but spread across a network. Although many pieces of software can be installed on a host to protect them, hardware solutions are still one of the best ways to prevent an attack. Two of the most common methods of doing this are to limit the scope of a system through virtual LANs (VLANs) and prevent systems from connecting to trusted networks through Network Access Control (NAC).

A virtual LAN (VLAN) allows you to take a large physical switch and regardless of where systems are plugged in, segment them into different networks based on function or access required. For example, if 100 employees and servers are all plugged into the same switch, you can limit the access or visibility by placing them on separate segments. Least privilege states that you give someone the least access he needs to do his job. For example, you can take all the accounting employees and their respective servers and put them on a separate VLAN.

If 1,000 systems are plugged into a switch and the systems are on a flat network, when one system gets compromised, all **999** of other systems can also be compromised. However, if you create VLANs and put 100 systems on 10 separate VLANs and control traffic between the VLANs, when 1 system gets compromised, it compromises only 100 systems, not 1,000. VLANs help control the visibility of systems on a network.



Hardening of Switches: NAC

Network Access Control (NAC):

- Dynamic VLAN allocation
- Isolates systems when they initially connect to the network
- Enables systems to be scanned and checked prior to being put on a trusted segment

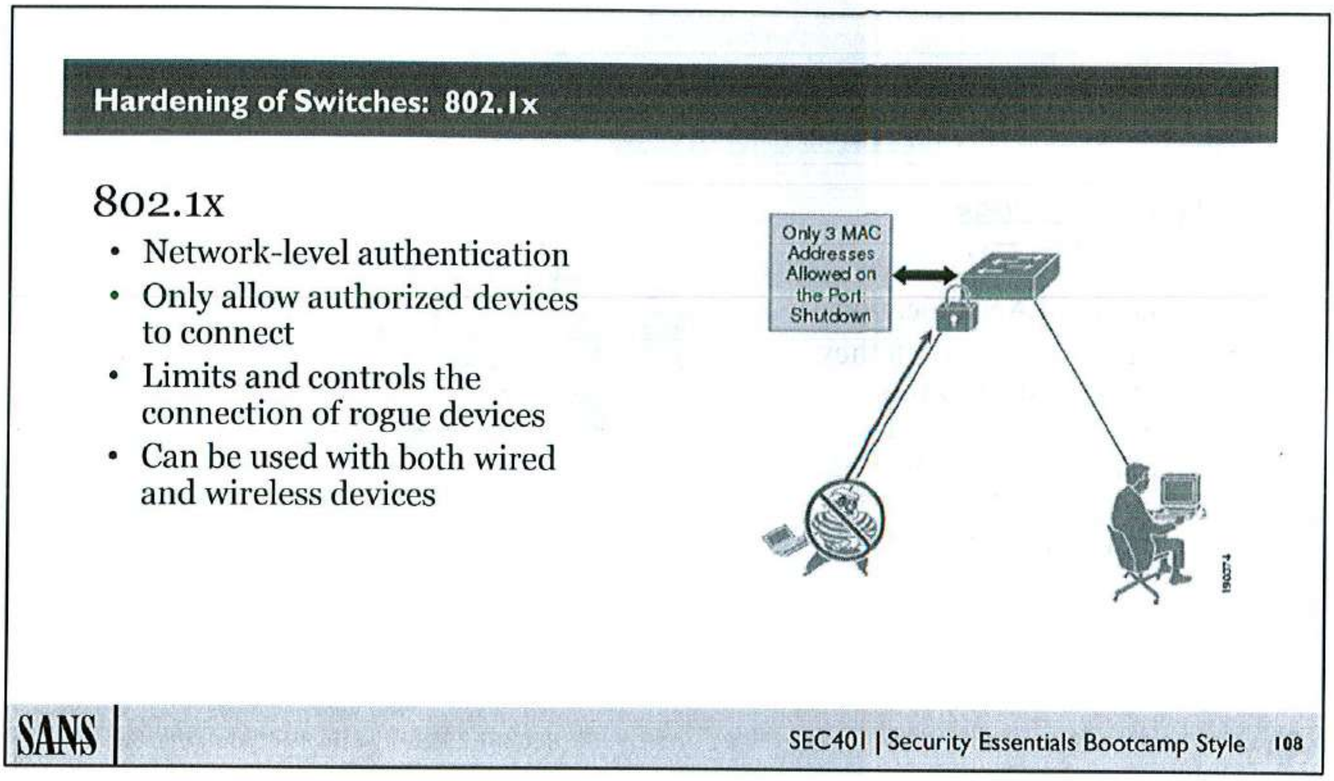
SANS | SEC401 | Security Essentials Bootcamp Style 107

One of the other problems today is that a laptop is out of the office for two weeks while an employee is traveling. This means the laptop is plugged into many untrusted networks and has a high chance of getting infected with malicious code. The infected laptop gets plugged back into the trusted network and infects several hosts. Network Access Control (NAC) allows systems to be placed on isolated VLANs until they have been scanned and properly patched, limiting their exposure to infecting other systems.

A key motto of security is prevention is ideal, but detection is a must. There is no way to completely prevent an attack from occurring. However, if a single system becomes infected, you can prevent it from connecting to critical networks through NAC and stop it from spreading to a large number of hosts through VLANs. NAC and VLANs both have value by themselves, but together, they provide a robust measure of protection for networks.

Reference

1. http://www.networkworld.com/article/2350482/cisco-subnet/is-cisco-still-guilty-of-providing-no-input-into-support-for-non-cisco-environments-.html



Hardening of Switches: 802.1x

802.1x

- Network-level authentication
- Only allow authorized devices to connect
- Limits and controls the connection of rogue devices
- Can be used with both wired and wireless devices

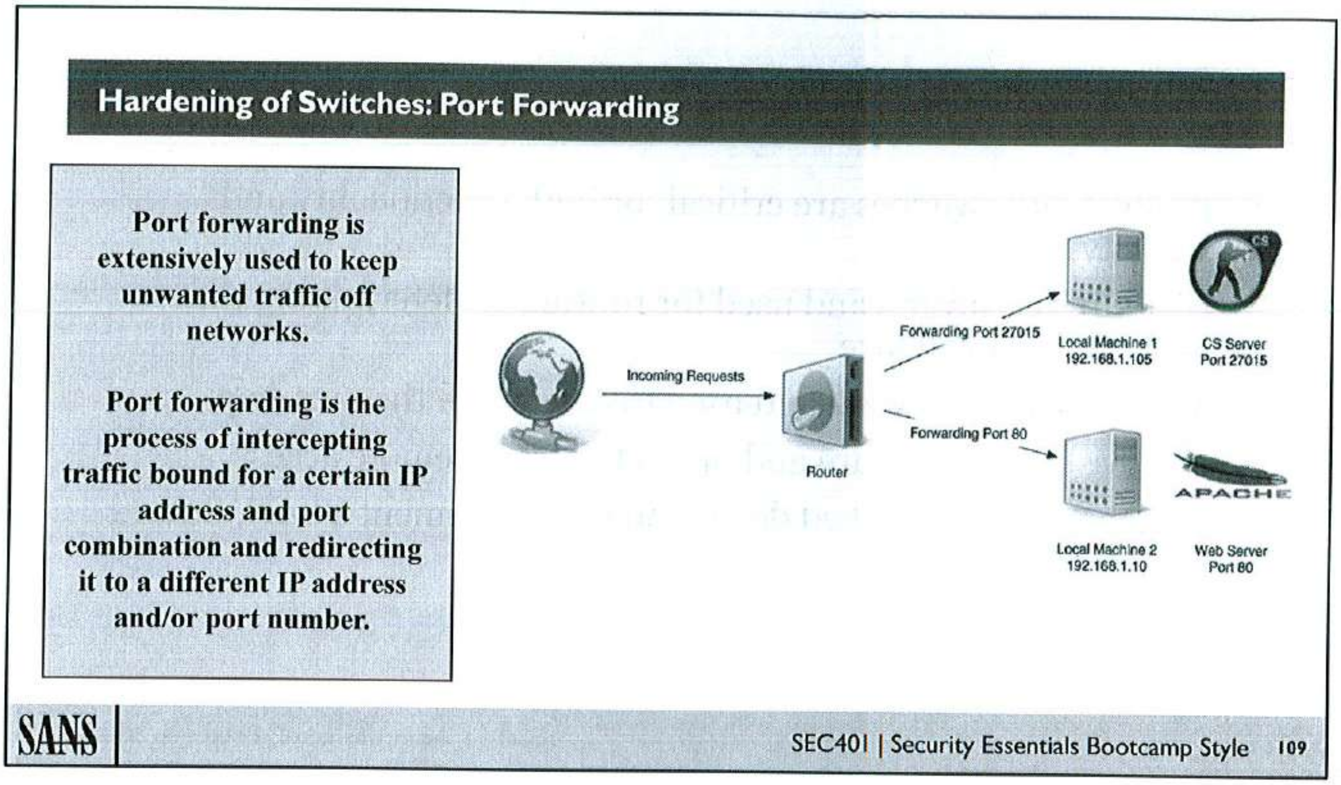
Only 3 MAC Addresses Allowed on the Port: Shutdown
190274

SANS | SEC401 | Security Essentials Bootcamp Style 108

Most switches have 802.1x port security features such as network level authentication which requires the connecting user to authenticate themselves before the session is established with a server. This is similar to a house guest ringing the doorbell and waiting for the owner to answer and invite them in as opposed to just walking in. Switches have other security feature such as preventing ports from learning more than 1 MAC address and assigning MAC addresses to ports. Many 802.1x wireless APs (access points) can be configured with a list of MAC addresses to allow or block. Since MAC addresses can be forged the 802.1X standard has a framework for combining port-level access control with some type of authentication.

Switch References

1. https://isc.sans.edu/forums/diary/Switch+hardening+on+your+network/6910/
2. http://www.larrytalkstech.com/port-forwarding-small-network-security/
3. http://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Security/Baseline_Security/securebasebook/sec_chap7.html
4. http://searchsecurity.techtarget.com/WLAN-security-Best-practices-for-wireless-network-security



Hardening of Switches: Port Forwarding

Port forwarding is extensively used to keep unwanted traffic off networks.

Port forwarding is the process of intercepting traffic bound for a certain IP address and port combination and redirecting it to a different IP address and/or port number.

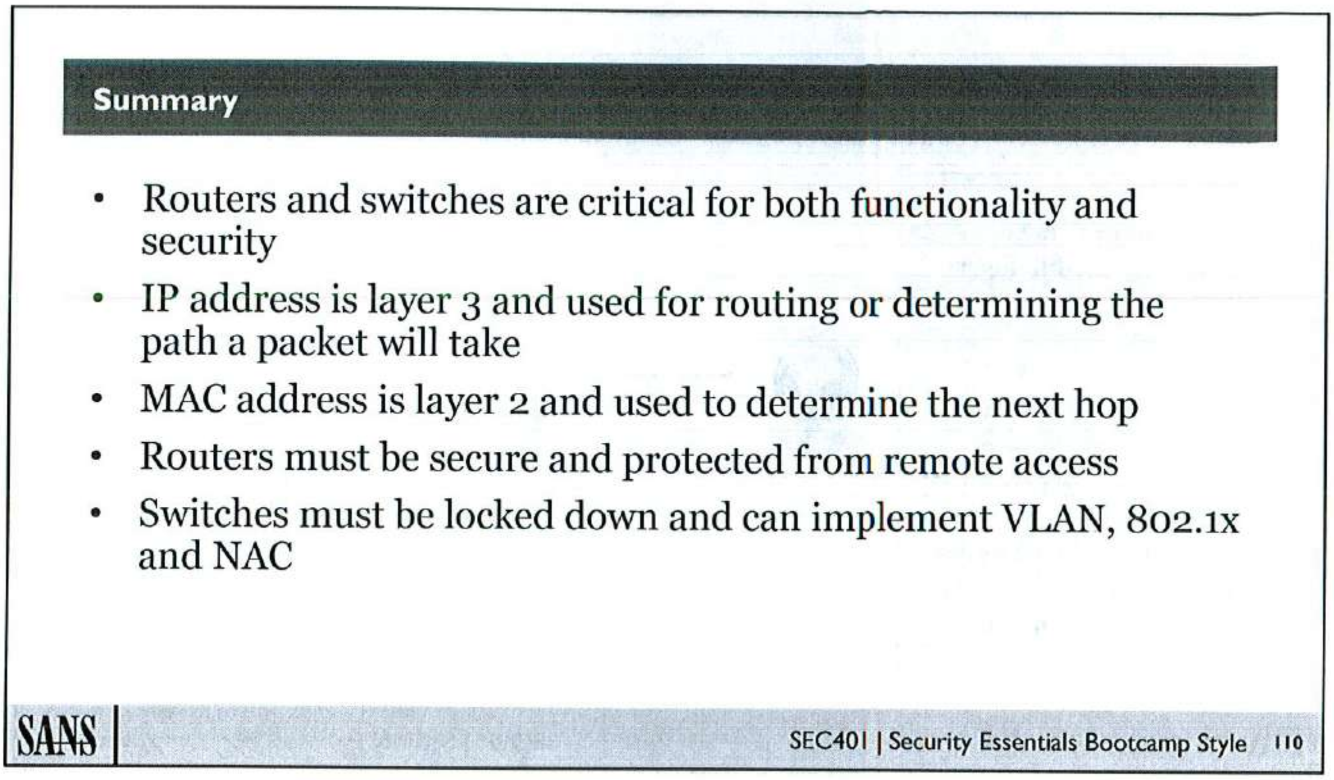
SANS | SEC401 | Security Essentials Bootcamp Style 109

Port forwarding is extensively used to keep unwanted traffic off networks. Port forwarding is the process of intercepting traffic bound for a certain IP address and port combination and redirecting it to a different IP address and/or port number. This hides exactly what services are running on the network, using only IP address to carry out multiple tasks, and dropping all unrelated traffic at the firewall. Port forwarding can be thought of like a parent in a large household that intercepts all the mail delivered to the house and filters it before the rest of the family sees it. This allow the parent to organize the mail and packages based on the person it is supposed to go to, even going as far as delivering the mail to their respective rooms while at the same time filtering out the fliers and junk mail that is addressed to "resident" rather than a specific member of the household.

From the outside, the sender of the junk mail addressing items to "resident" has no insight as to who is in the house and who did or did not receive the message. If the sender of the junk mail was malicious there would be no feedback given as to whether the attempt was successful or not and they would move on.

References

1. https://isc.sans.edu/forums/diary/Switch+hardening+on+your+network/6910/
2. http://www.larrytalkstech.com/port-forwarding-small-network-security/
3. http://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Security/Baseline_Security/securebasebook/sec_chap7.html
4. http://searchsecurity.techtarget.com/WLAN-security-Best-practices-for-wireless-network-security

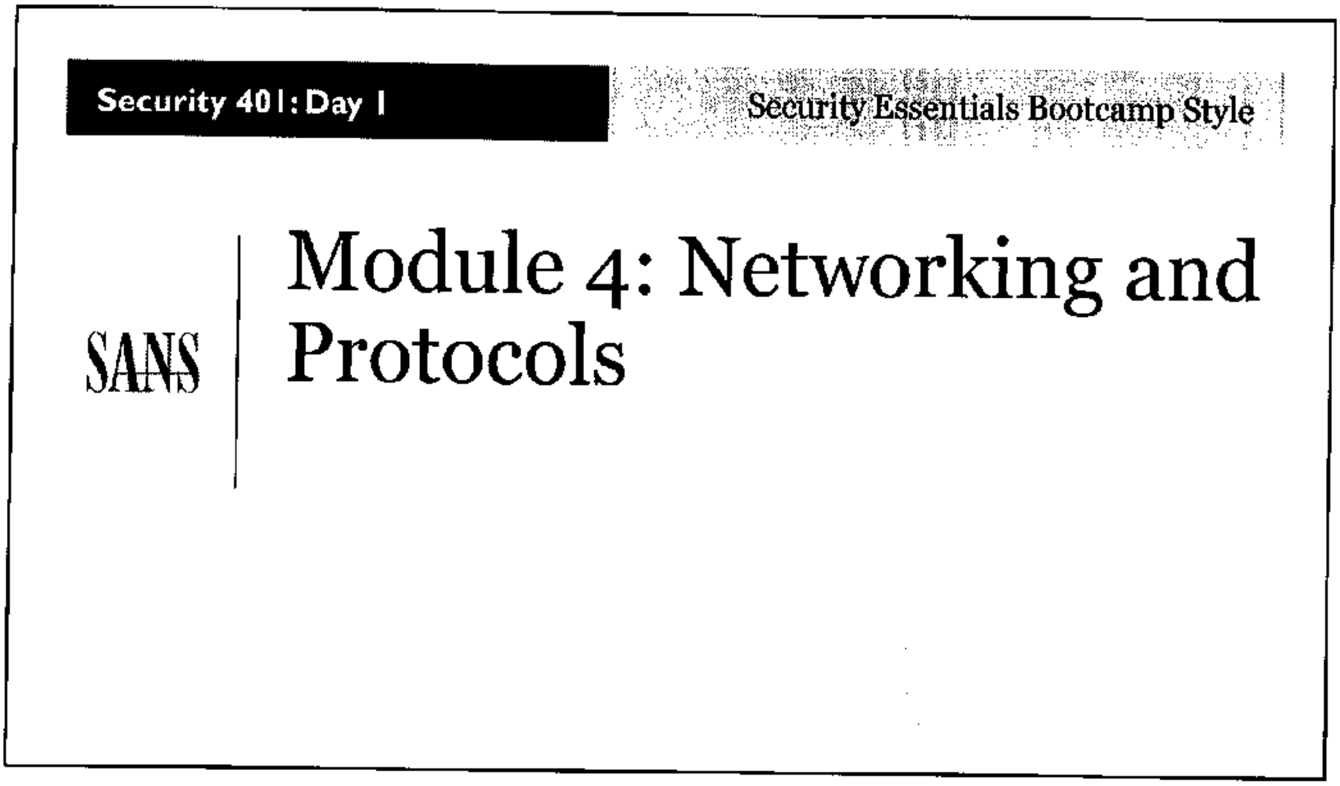


Summary

- Routers and switches are critical for both functionality and security
- IP address is layer 3 and used for routing or determining the path a packet will take
- MAC address is layer 2 and used to determine the next hop
- Routers must be secure and protected from remote access
- Switches must be locked down and can implement VLAN, 802.1x and NAC

An area that is often very critical to the security of an organization but often overlooked is network device security, mainly routers and switches. The network infrastructure if often viewed primarily from a functionality and stability perspective and often it is forgotten that it has components that can be compromised. One of the reasons for this is that many security professionals do not understand how routers and switches work. Therefore this module focused on gaining a better understanding of how routers and switches work and the relationship between layer 2 MAC addresses and layer 3 IP addresses.

The module finished by looking at some key areas of security when focused on routers and switches. The most important is to make sure the IOS is properly secured and hardened. With traditional computers we know the rule to never use a default install of an operating system because it has extraneous services running and open ports. The same principles hold true for routers and switches. While both devices need to be secure, since routers often have a public IP address and are visible from the Internet, additional care needs to be put in place to lock them down.



Security 401: Day 1

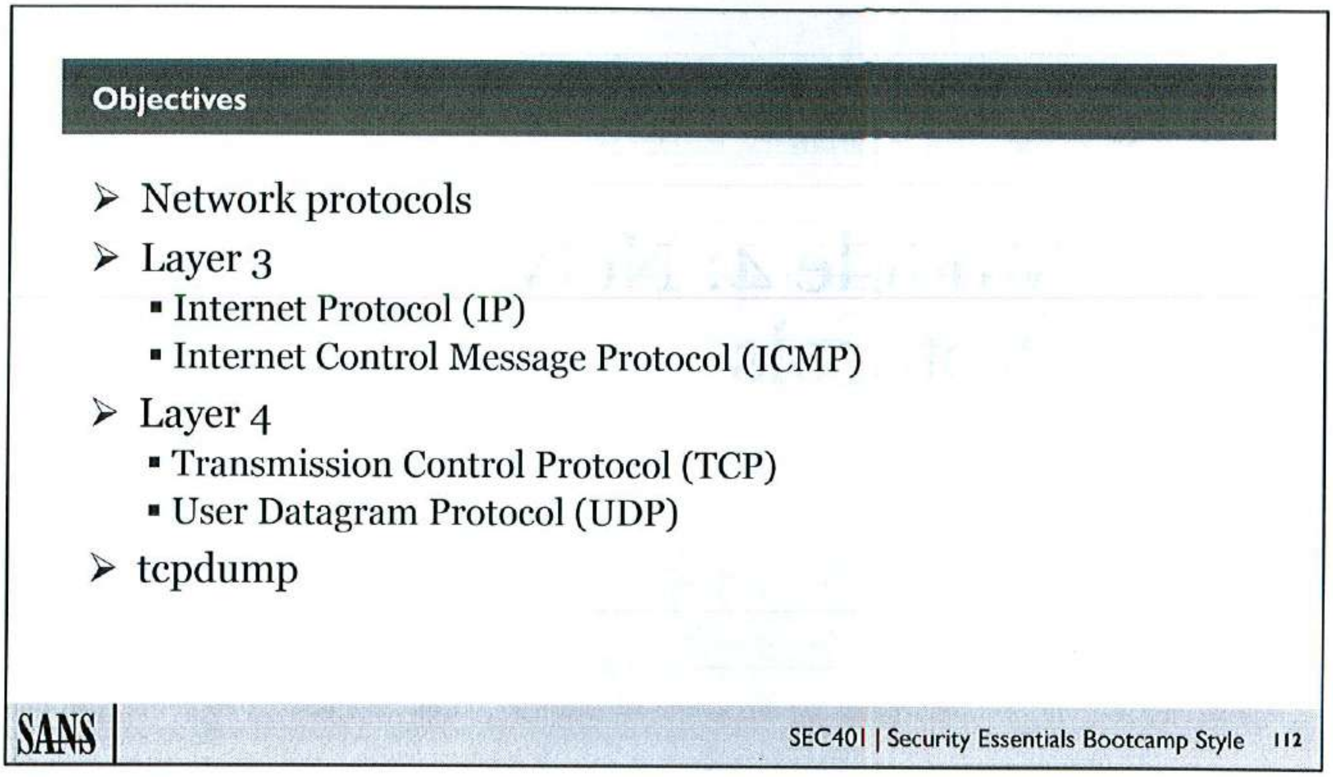
Security Essentials Bootcamp Style

SANS

Module 4: Networking and Protocols

Module 4: Networking and Protocols

You cannot get very far in any subject without a good basic understanding of that topic's fundamentals. You are studying information security, and because much information that needs to be secure is transmitted across networks, a good understanding of how these networks actually work is critical. A solid understanding of the interworkings of networks allows you to be more effective in recognizing, analyzing, and responding to the latest (perhaps unpublished) attacks. This module introduces the core areas of computer networks and protocols.



Objectives

- ➢ Network protocols
- ➢ Layer 3
  - ▪ Internet Protocol (IP)
  - ▪ Internet Control Message Protocol (ICMP)
- ➢ Layer 4
  - ▪ Transmission Control Protocol (TCP)
  - ▪ User Datagram Protocol (UDP)
- ➢ tcpdump

SANS | SEC401 | Security Essentials Bootcamp Style 112

We begin with a quick refresher on protocols and protocol stacks; the basic building blocks of network communications. We examine and compare two of the most common examples: the OSI reference model and TCP/IP.

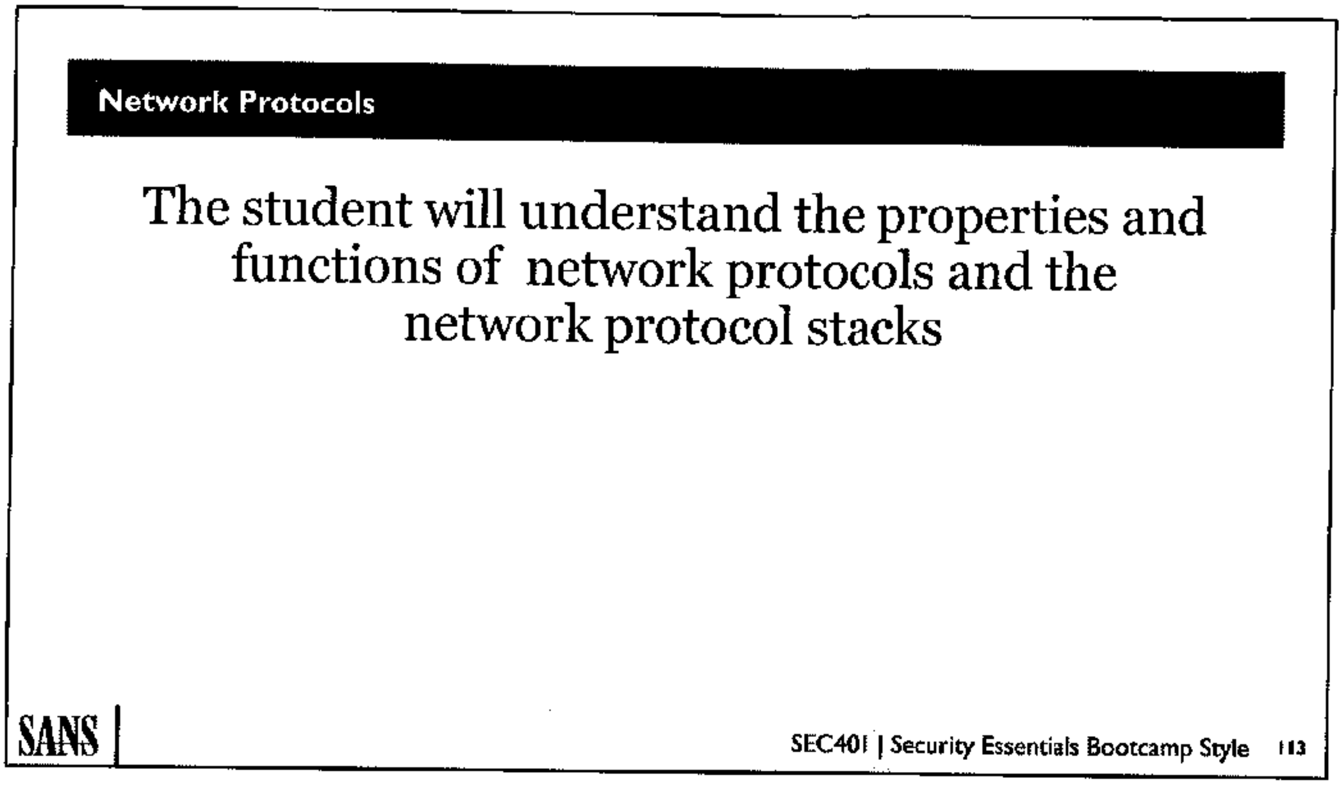
After you master the basics of protocols, this module shows you how they work together to structure network transmissions into frames and packets as they are sent across the wire. Additionally, we examine the Internet Protocol (IP) packet header to see what you can learn from it.

Once you have an understanding of how IPv4 networks work, we look at IP version 6 (IPv6), which was designed to address some of the limitations present in IPv4 networks. We discuss how IPv6 varies from IPv4, how IPv6 is addressed, and some of the features of IPv6.

We also examine the network layer's Internet Control Message Protocol (ICMP). IP relies on ICMP for network status messages and error reporting. ICMP messages are common on any IP network, so ICMP is just as important as TCP and UDP from a security standpoint.

IP never was intended to stand by itself. Although it is possible to write programs that talk directly to the network layer (IP), it is rarely done, except in certain limited cases, such as network-testing tools or hacking utilities. Most programs do not want to have to deal with the level of complexity that speaking directly to the network layer brings and instead are written to make use of higher-level protocols residing in the transport layer. For IP, these are the User Datagram Protocol (UDP) and the Transmission Control Protocol (TCP).

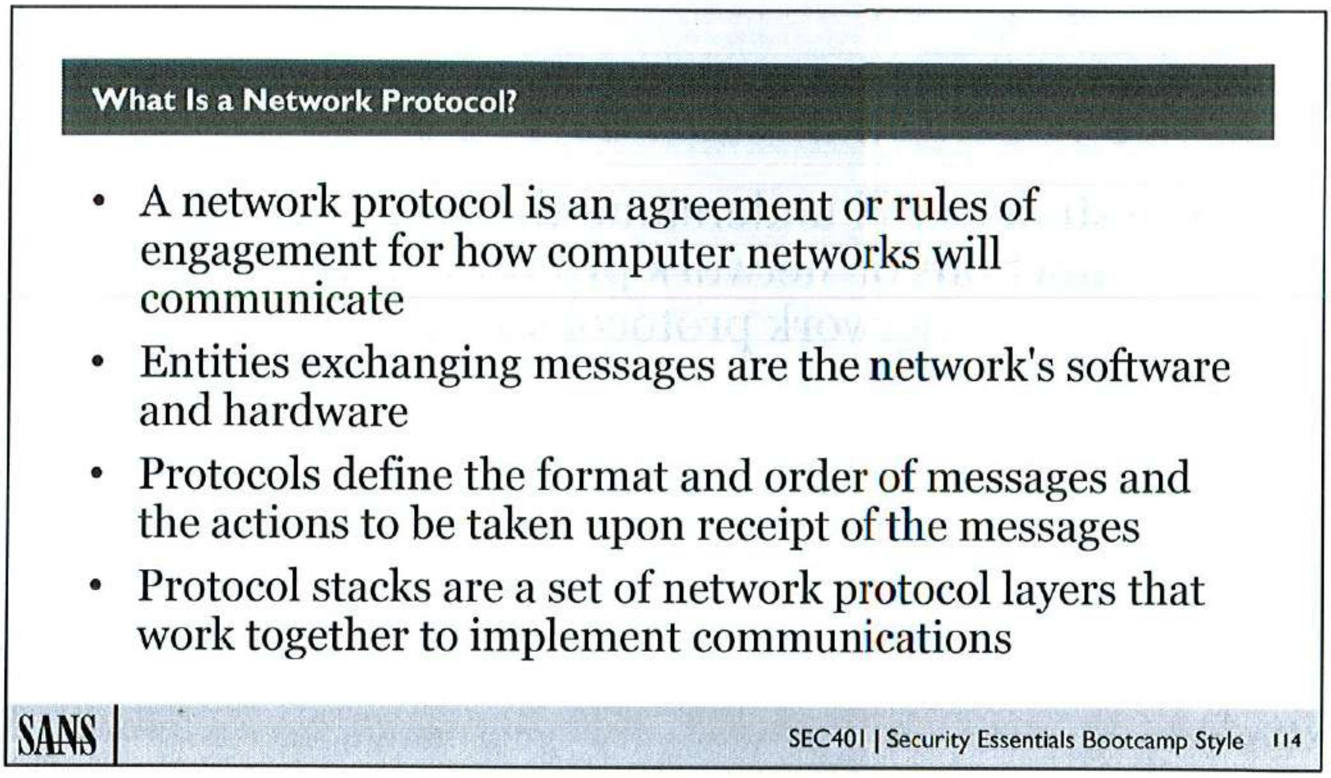
We end this module by looking at how we can analyze packets by using a tool called tcpdump.



Network Protocols

The student will understand the properties and functions of network protocols and the network protocol stacks

This page intentionally left blank.



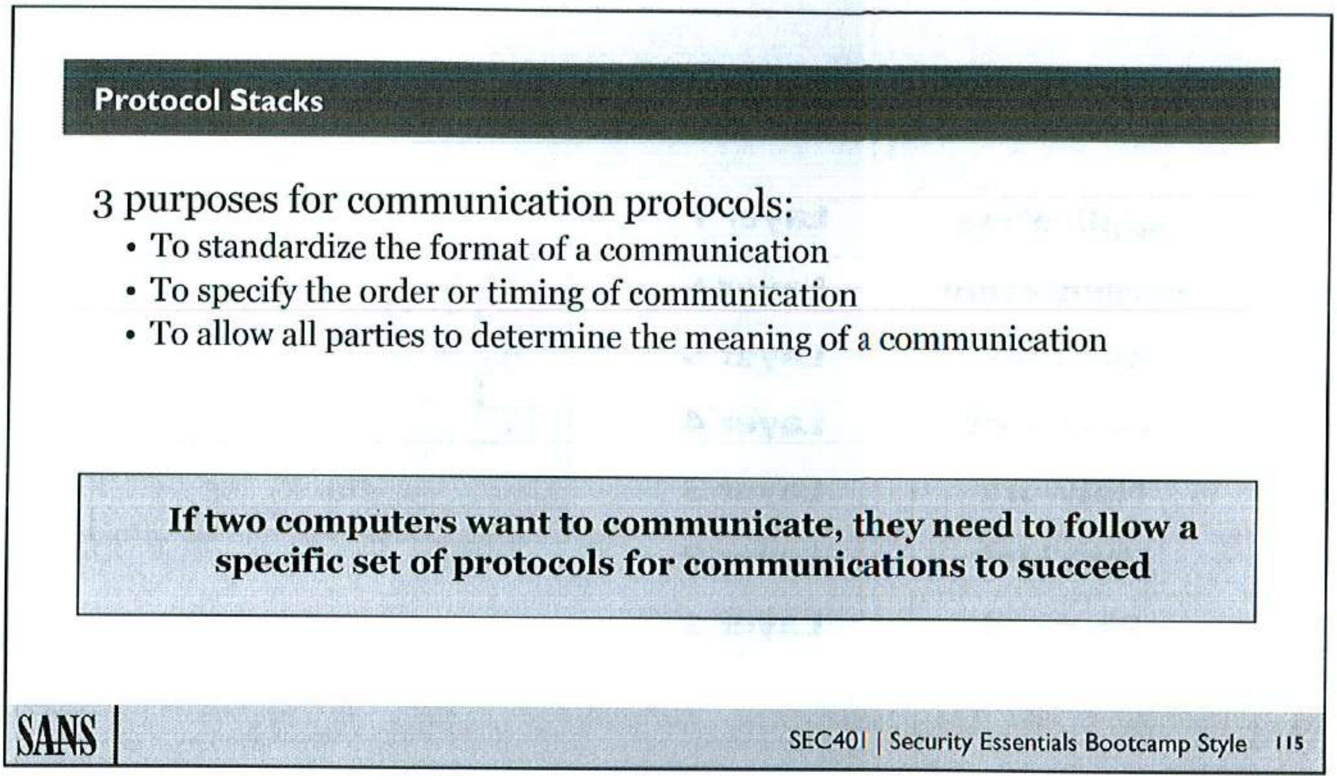
What Is a Network Protocol?

- A network protocol is an agreement or rules of engagement for how computer networks will communicate
- Entities exchanging messages are the network's software and hardware
- Protocols define the format and order of messages and the actions to be taken upon receipt of the messages
- Protocol stacks are a set of network protocol layers that work together to implement communications

SANS | SEC401 | Security Essentials Bootcamp Style 114

In the broadest sense, a protocol is nothing more than an agreement of how different entities will act and react in certain circumstances. A medical protocol prescribes a course of treatment for a certain disease. A diplomatic protocol is the basis for a formal treaty that, for example, might specify how two nations will allow free trade along a common border.

A communications/network protocol establishes an agreement between network entities, such as hosts and servers, for how they will communicate. When protocols are worked out in advance, they are effective and efficient. If any participant breaks the protocol, the communication gets confused or can break down altogether. You have probably been in a situation in which you had "interoperable" hardware or software products that were based on the same standards but were not actually compatible. Odds are, one or both of those products deviated from the standard and implemented the protocol differently. This is why we require strict conformance to standard protocols.



Protocol Stacks

3 purposes for communication protocols:

- To standardize the format of a communication
- To specify the order or timing of communication
- To allow all parties to determine the meaning of a communication

If two computers want to communicate, they need to follow a specific set of protocols for communications to succeed

SANS

SEC401 | Security Essentials Bootcamp Style 115

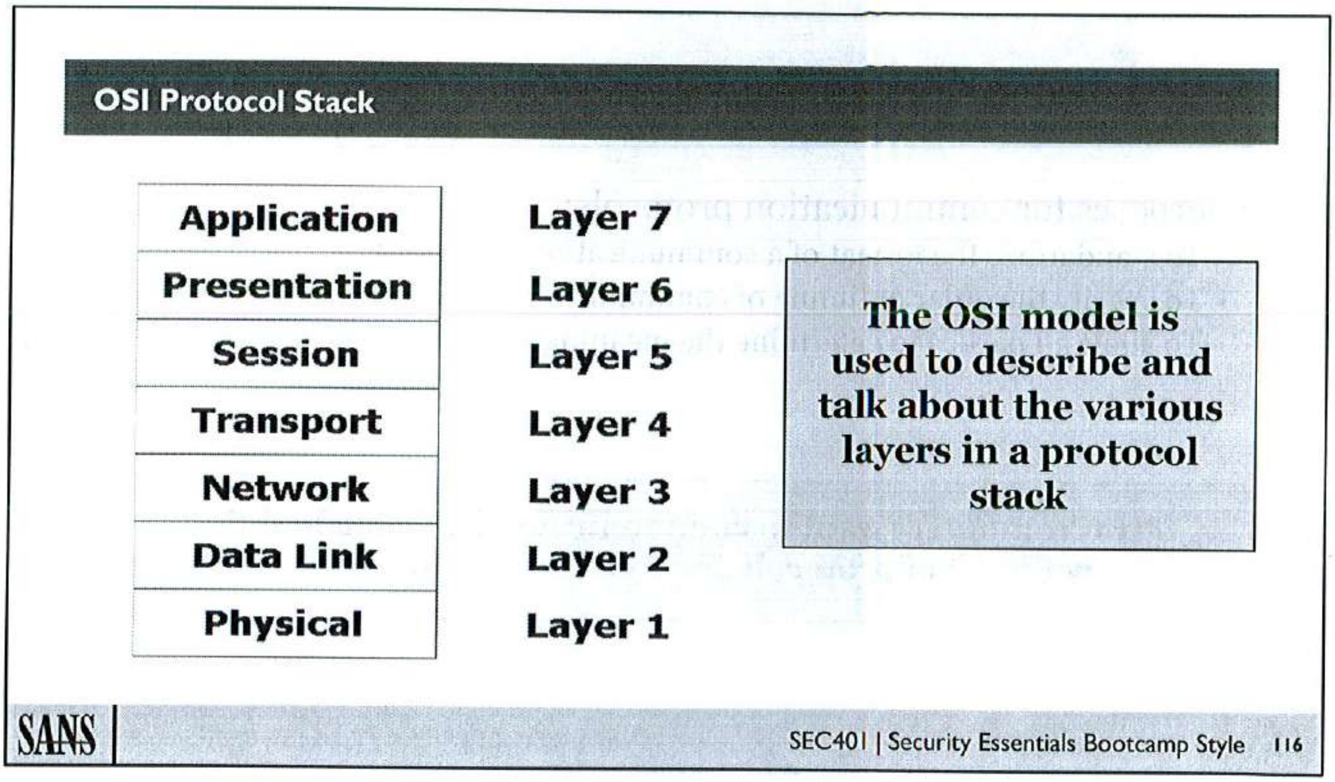
There are three basic purposes for communications protocols:

- To standardize the format of a communication
- To specify the order or timing of communication
- To allow all parties to determine the meaning of a communication

As long as both sides of the communication are using the same protocol and implement it properly, communication is successful.

If two computers want to communicate, they need to follow a specific set of protocols for communications to succeed. Numerous protocols are involved. Some protocols concern themselves with breaking up a transmission into smaller bunches of data called packets. Some make sure that each packet has the proper information in the proper locations. Others describe how information is copied from your computer to the network cable. Still, others ensure that packets all get to the right place in the proper order. Even with a transaction as seemingly simple as fetching a web page, a number of protocols are required to allow the communication to succeed. In computer communications, these layered protocols are referred to as a protocol stack.

In order for two or more entities to be able to communicate, they need to have standard rules of engagement. A protocol provides those rules so computers anywhere in the world can communicate. To make the protocol easier to manage, it is broken down into a protocol stack, where each layer receives a service from the layer below it and provides a service to the layer above it.



OSI Protocol Stack

Application	Layer 7
Presentation	Layer 6
Session	Layer 5
Transport	Layer 4
Network	Layer 3
Data Link	Layer 2
Physical	Layer 1

The OSI model is used to describe and talk about the various layers in a protocol stack

SANS

SEC401 | Security Essentials Bootcamp Style 116

In order to be able to allow computers to communicate across a network, a protocol stack is required. The OSI model is one of two protocol stacks that we are going to cover. The OSI model is used to describe and talk about the various layers in a protocol stack.

The standard reference model for protocol stacks is the International Standards Organization's (ISO) Open Systems Interconnect (OSI) model. The OSI model divides network communications into seven layers.

The physical layer handles transmission across the physical media. This includes such things as electrical pulses on wires, light pulses on fiber, connection specifications between the interface hardware and the network cable, and voltage regulation.

The data link layer connects the physical part of the network (cables and electrical signals) with the abstract part (packets and data streams).

The network layer handles the network address scheme and connectivity of multiple network segments. It describes how systems on different network segments find and communicate with each other.

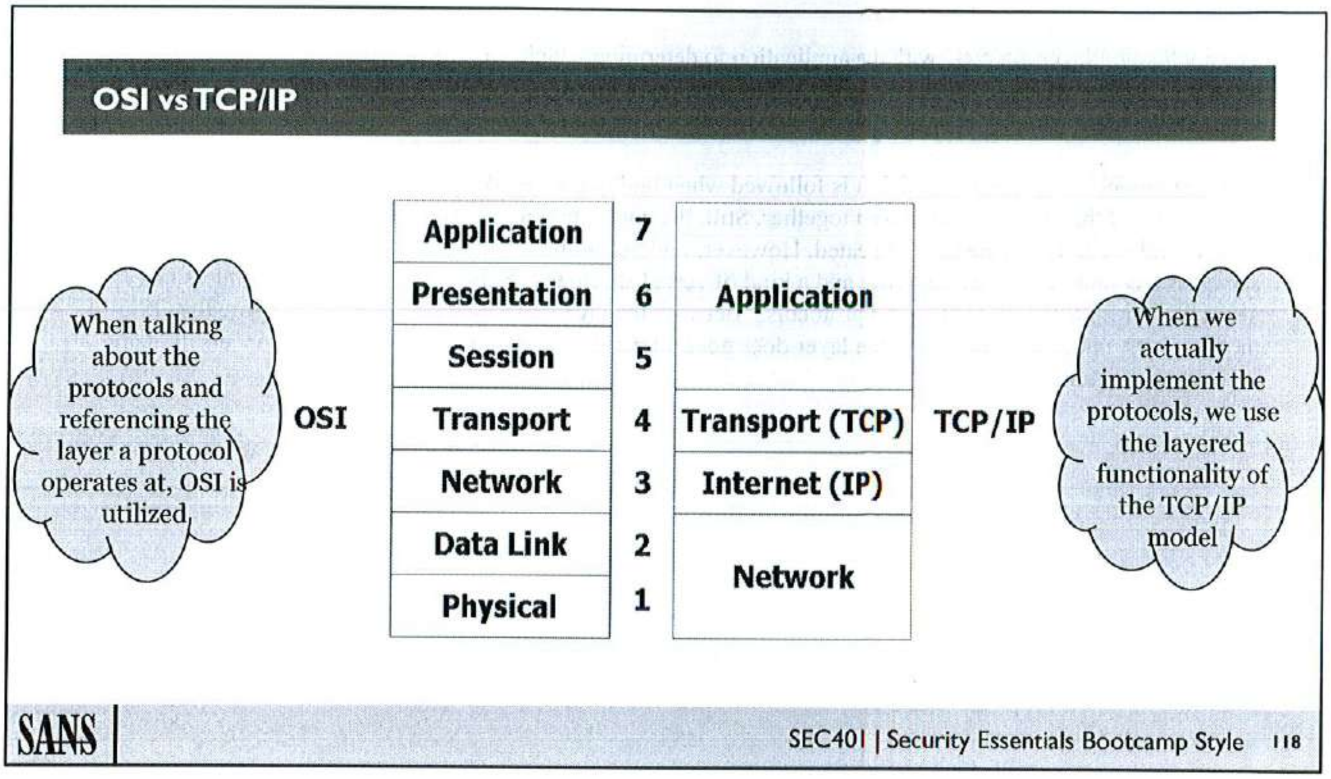
The transport layer interacts with your data and prepares it to be transmitted across the network. It is this layer that ensures reliable connectivity from end-to-end. The transport layer also handles the sequencing of packets in a transmission.

The session layer handles the establishment and maintenance of connections between systems. It negotiates the connection, sets it up, maintains it, and makes sure that information exchanged across the connection is in sync on both sides.

The presentation layer makes sure that the data sent from one side of the connection is received in a format that is useful to the other side. For example, if the sender compresses the data prior to transmission, the presentation layer on the receiving end would have to decompress it before the receiver could use it.

The application layer interacts with the application to determine which network services are required. When a program requires access to the network, the application layer manages requests from the program to the other layers down the stack.

The OSI model is a reference model. It is followed when building network applications; however, in many cases, some of the layers are combined together. Still, the core functionality is present. Most protocol stacks do not have all seven layers clearly delineated. However, understanding the OSI model is important because it serves as a common point of reference and a kind of verbal shorthand. Network engineers and vendors talk about "Layer 2 switches" or "Layer 3 protocols." Because the layers to which they are referring are the OSI model layers, understanding what each layer does goes a long way toward understanding the conversation and securing your network services.



OSI vs TCP/IP

When talking about the protocols and referencing the layer a protocol operates at, OSI is utilized

OSI		TCP/IP
Application	7	Application
Presentation	6	
Session	5	
Transport	4	Transport (TCP)
Network	3	Internet (IP)
Data Link	2	Network
Physical	1	

When we actually implement the protocols, we use the layered functionality of the TCP/IP model

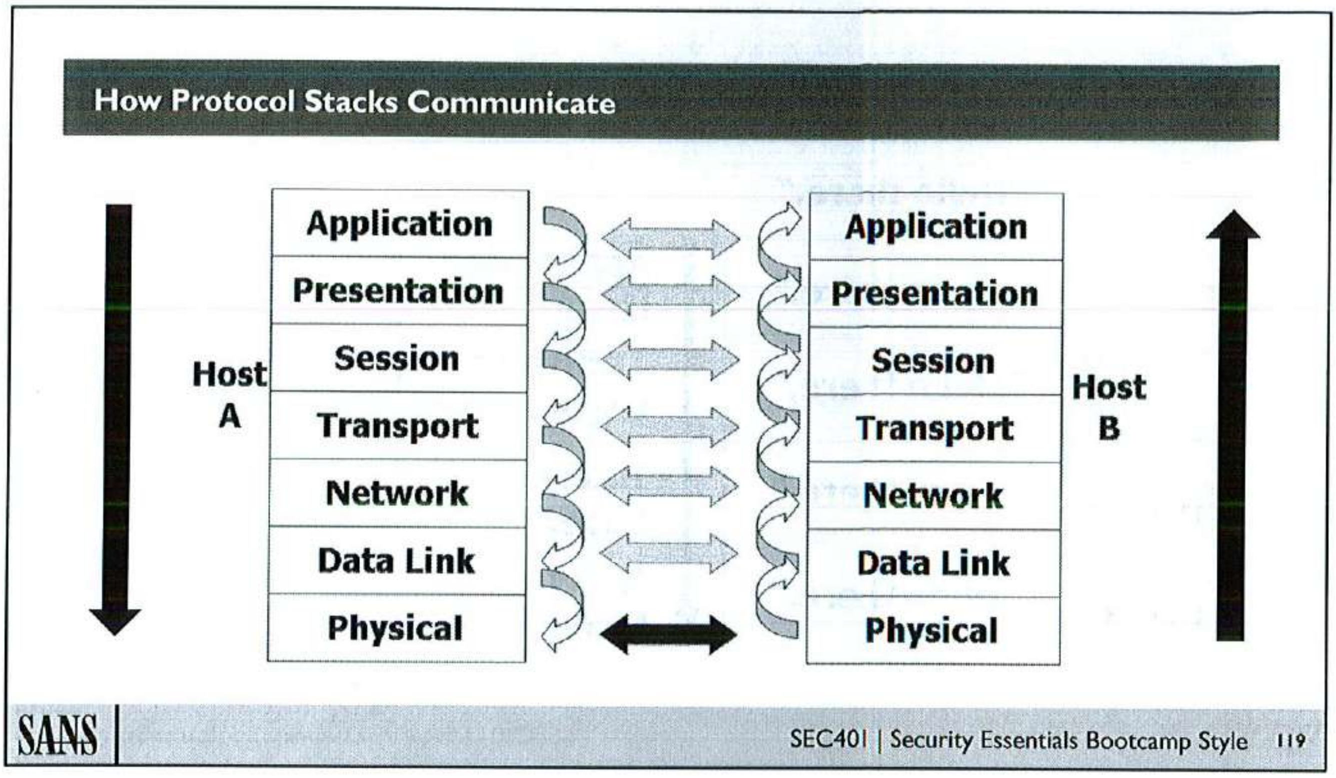
SANS | SEC401 | Security Essentials Bootcamp Style 118

Many people ask which protocol stack is better: OSI or TCP/IP. The answer is both. In practice, both are utilized and need to be understood by students. When talking about the protocols and referencing the layer a protocol operates at, OSI is utilized. For example, we always say that routing is a Layer 3 function. However, when we actually implement the protocols, we use the layered functionality of the TCP/IP model.

The Transmission Control Protocol/Internet Protocol (TCP/IP) stack has only four layers: the network layer, the internet layer, the transport layer, and the application layer. Even though the stack has only four layers as compared to the seven-layer OSI model, it still performs the same functions. It just means that because there are fewer layers, each layer has to do a little more work.

As you can see, the OSI model is more granular. The OSI model splits apart some functionality that was combined in the TCP/IP model. The network layer in the TCP/IP model comprises both the physical and the data link layers in the OSI model, while the application layer in TCP/IP encompasses the application, presentation, and session layers of OSI. The OSI model is more detailed because it was designed to support protocols other than just TCP/IP. By creating more layers, the designers made it easier to break down the functionality of each protocol and build more specific interfaces and linkages between the layers.

Even though each model breaks down the functionality a bit differently, no matter which model you use, it must perform all the functions required to take a piece of application data, place it into a packet, put that packet on the wire, and deliver it safely and efficiently to its destination.



How Protocol Stacks Communicate

Host A

- Application
- Presentation
- Session
- Transport
- Network
- Data Link
- Physical

Host B

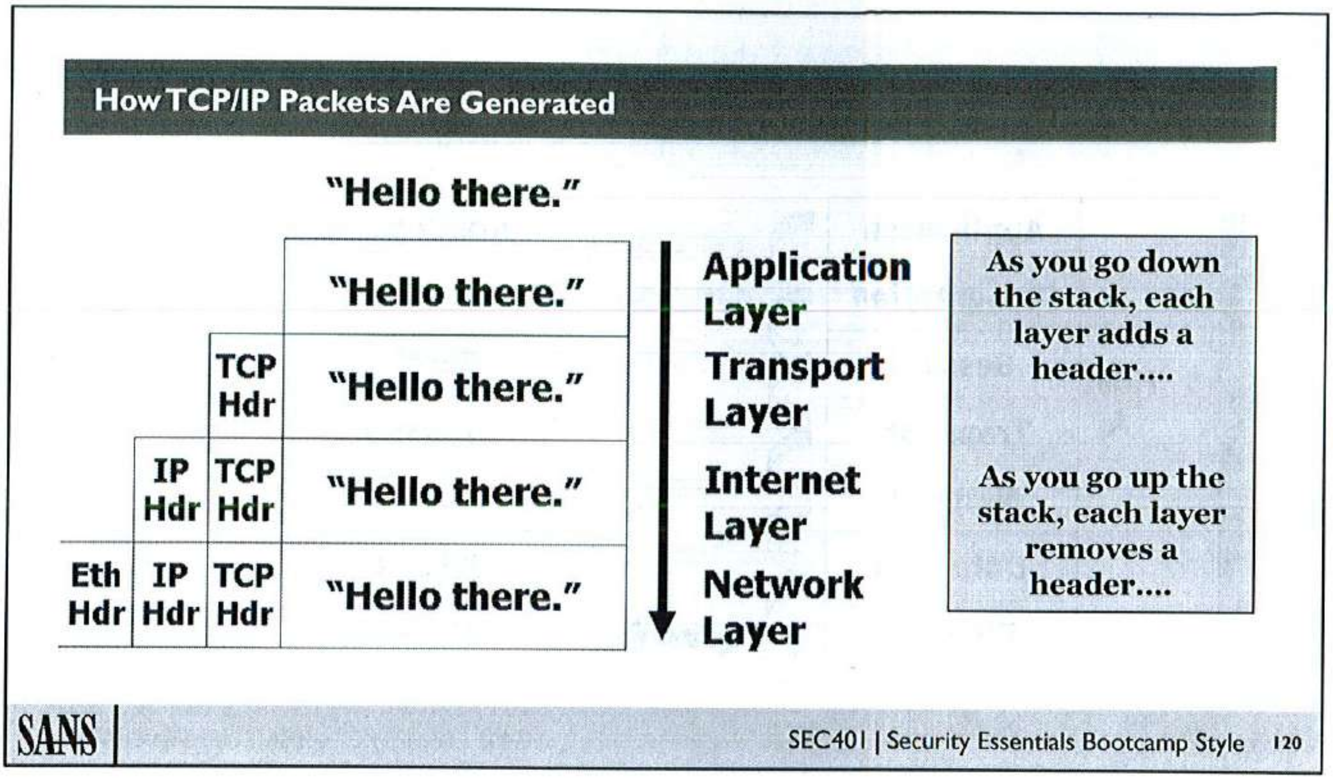
- Application
- Presentation
- Session
- Transport
- Network
- Data Link
- Physical

SANS

SEC401 | Security Essentials Bootcamp Style 119

The basic principal of stack-based communication is that data from one layer of the stack can be understood only by the corresponding layer on the remote computer. In other words, the application layer on Host A exchanges information with the application layer on Host B, and the transport layer on Host A exchanges information with the transport layer on Host B, and so on. However, each layer is only aware of the requirements that must be fulfilled to communicate with the corresponding layer on the remote host. Each layer is not aware of the contents of the data from other layers and, in fact, does not need to be aware of specifics of the other layers for communications to succeed. This layer independence does have security implications.

For example, assume you want to pretend to be another computer and spoof its IP address, perhaps to give an incorrect DNS answer. You would have to be in the communication path so you could sniff the packet and keep it from reaching the real DNS server. Then, you could craft the answer. But, what if you did not carefully spoof the IP header? Then, the TTL would be wrong. If the IP layer and UDP layer work together, they could probably notice a change. But, they don't. This is an important security principle. Layer independence makes it easier and faster to write and maintain networking software, but your security systems need to consider all the information in all of the layers. What this means is that each layer needs to deal only with its own communication requirements and then pass the data down the stack so each subsequent layer can satisfy its own requirements. Each layer takes the data from the layer above it, satisfies its own requirements by adding its own data, and then passes it to the next layer down the stack. On the receiving end, the data is passed up the stack with each subsequent layer removing the data added by its peer layer and passing the data up the stack until, finally, the application layer receives the data.



How TCP/IP Packets Are Generated

"Hello there."

"Hello there." — Application Layer

TCP Hdr | "Hello there." — Transport Layer

IP Hdr | TCP Hdr | "Hello there." — Internet Layer

Eth Hdr | IP Hdr | TCP Hdr | "Hello there." — Network Layer

As you go down the stack, each layer adds a header....

As you go up the stack, each layer removes a header....

It is important to understand how a packet is generated as it moves through the TCP/IP stack. Ultimately, each layer on the sender needs to communicate with the same layer on the receiving computer. However, they cannot directly talk because you must go down the stack, across the network, and back up the stack on the receiving system. The way this is accomplished is by having each layer add a header as you go down the stack on the sender and each system remove a header on the receiving system as it goes up the stack. By performing this function, the header that is created by a given layer on the sender is received by the corresponding layer on the receiving system.

To start with, the application layer takes information from the application itself. In this case, we send the phrase "Hello there." to another computer. A program gives the "Hello there." to the stack's application layer, which creates an empty packet and places the "Hello there." inside. The application layer then sends the packet to the transport layer.

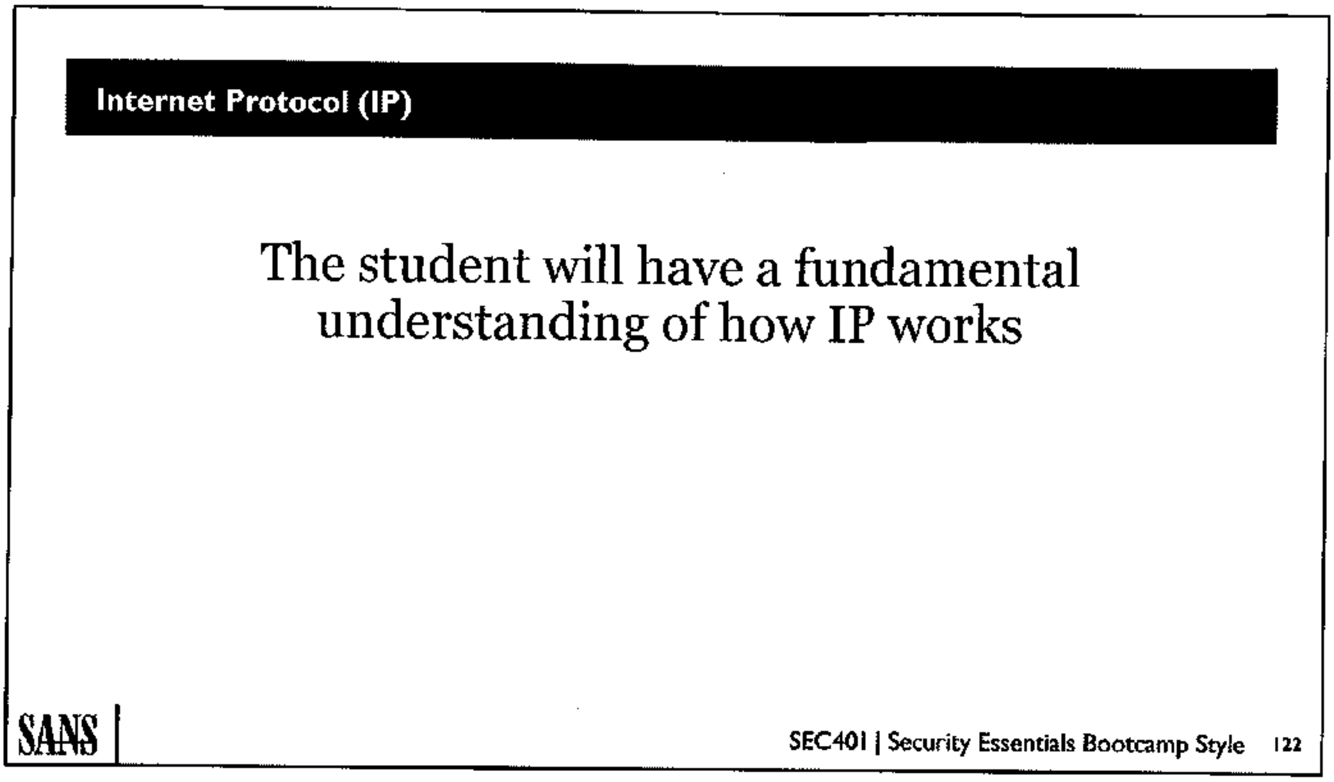
The transport layer takes the packet and adds a header to it. The header has all the information that the transport layer on the other side of the connection needs to determine what to do with the packet. After the transport header is put on the packet, it is given to the internet layer.

The internet layer puts another header in front of the packet. Like the transport layer before it, this header gives information for the internet layer on the other end. After this header is attached, the packet is sent to the network layer.

As you probably have guessed by now, the network layer puts its own header on the packet. This header assists the routers and gateways between the two machines in sending the packet along its way. After this final header is placed on the packet, it is put on the wire and sent to its final destination.

What happens when the remote computer receives the data? The operation starts again, but this time in reverse. The network layer strips off the header its counterpart put on the packet in the first place and then passes the rest of the packet up to the internet layer. The process continues up through the rest of the stack, each layer removing only the information placed in the packet by its counterpart in the sending host's stack until the original "Hello there." string reaches the remote application. This process of stripping off one layer's headers and passing the rest of the packet up to the next higher layer is known as decapsulation.

Each layer in the stack adds its own header to the packet and passes it along to the next lower layer. It encapsulates the packet it was given in protocol headers of its own before passing it on to the next lower layer, which performs its own encapsulation. The process is performed in reverse on the receiving system.



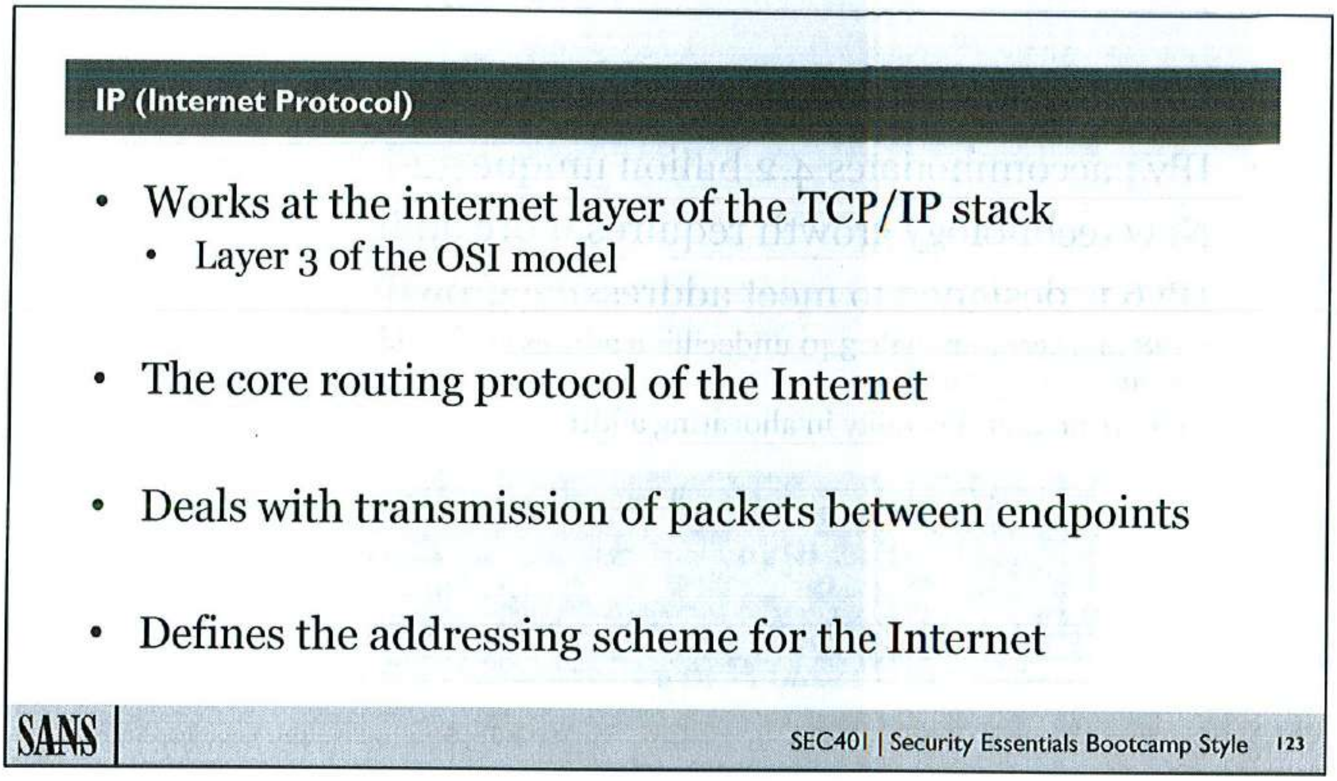
Internet Protocol (IP)

The student will have a fundamental understanding of how IP works

SANS

SEC401 | Security Essentials Bootcamp Style 122

This page intentionally left blank.



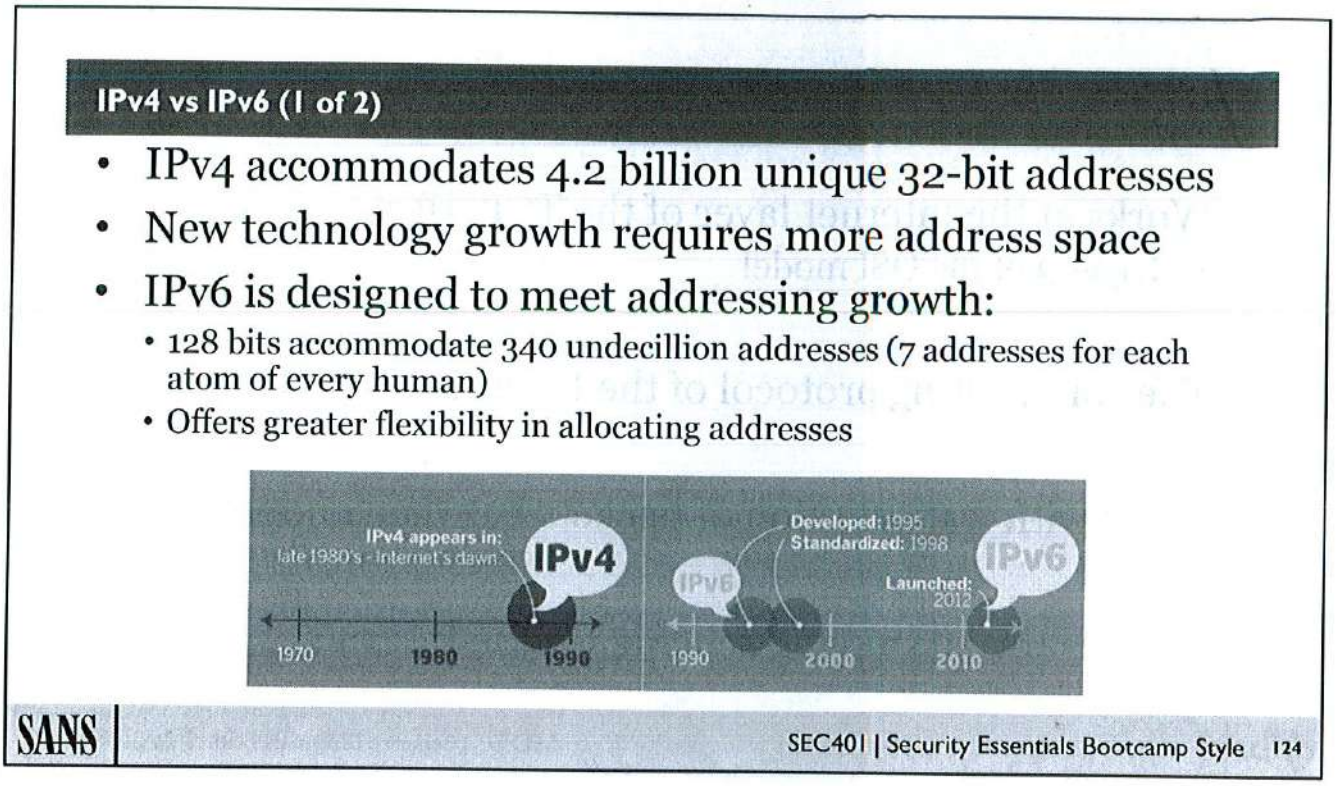
IP (Internet Protocol)

- Works at the internet layer of the TCP/IP stack
  - Layer 3 of the OSI model
- The core routing protocol of the Internet
- Deals with transmission of packets between endpoints
- Defines the addressing scheme for the Internet

IP is the basis for all communication on the Internet. It's so important that it even gets its own layer in the TCP/IP stack! The primary purpose of IP is to handle the transmission of packets between network endpoints, usually single hosts identified with a unique address. IP includes some features that provide basic measures of fault-tolerance (Time to Live, checksum), traffic prioritization (type of service), and support for the fragmentation of large packets into multiple smaller packets (ID field, fragment offset).

IP is singularly focused on routing packets from point A to point B on the network as quickly and efficiently as possible. IP does not provide any mechanisms for guaranteed delivery or delivery in sequence. Instead, it relies on upper-layer protocols and applications to provide those mechanisms, as appropriate, for the application.

IP defines the IP addressing scheme that allows each host to be uniquely identified. It also defines the rules used to route packets between hosts, whether close or separated by large distances.



IPv4 vs IPv6 (1 of 2)

- IPv4 accommodates 4.2 billion unique 32-bit addresses
- New technology growth requires more address space
- IPv6 is designed to meet addressing growth:
  - 128 bits accommodate 340 undecillion addresses (7 addresses for each atom of every human)
  - Offers greater flexibility in allocating addresses

IPv4 appears in:
late 1980's - Internet's dawn
IPv4
1970
1980
1990
IPv6
Developed: 1995
Standardized: 1998
Launched: 2012
IPv6
1990
2000
2010

SANS

SEC401 | Security Essentials Bootcamp Style 124

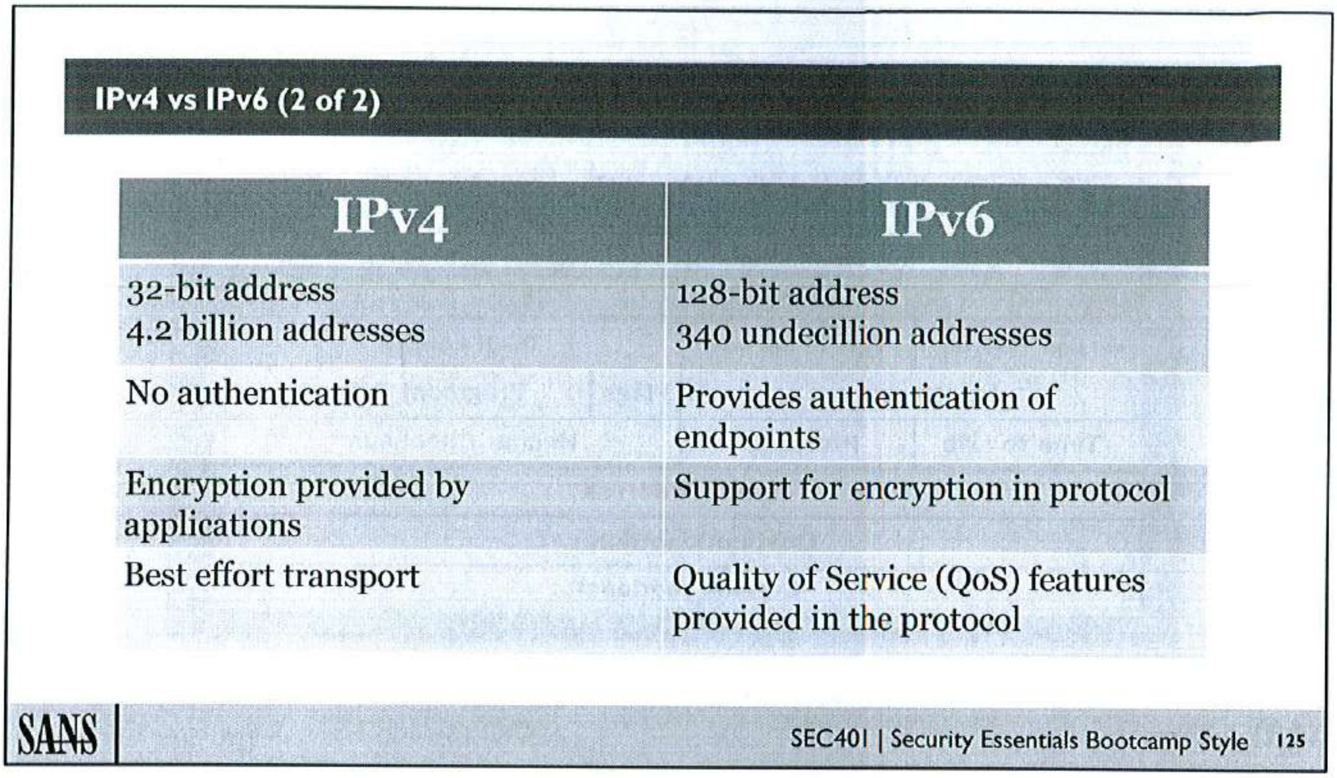
IPv6 is a protocol designed to supersede IPv4 addressing while supporting the growth of the Internet. Although the IPv4 protocol accommodates 4.2 billion unique IP addresses with a 32-bit address, the allocation of IP addresses on the Internet was not completed in the most efficient manner, leaving a shortage of available IP addresses. With deployment of technologies, such as NAT and CIDR, the Internet continued its growth, but was still somewhat limited without the widespread availability of globally unique IP addresses. New technologies, such as mobile phones, connecting to the Internet have increased demand for addresses, and so has the spread of Internet technology to populous countries, such as China and India. As a result, a new mechanism was needed to accommodate continued growth and adoption of Internet-connected technology.

The IPv6 protocol was designed to meet these growth demands, expanding the address size from 32 bits to 128 bits. A 128-bit address is approximately 340 undecillion addresses or 340,282,366,920,938,463,463,374,607,431,768,211,456. With this many unique addresses, the IPv6 protocol can accommodate 7 unique IP addresses for each atom in every human on earth.

Of course, all of our atoms don't need that many IP addresses. Instead, the sheer volume of available IP addresses provides for more flexible deployment of address space on the Internet. For example, ISPs will be able to geographically assign IPv6 prefixes to different parts of the world, allowing for the simplified routing of traffic on the Internet. Organizations can obtain an IPv6 prefix with sufficient available addressing to accommodate all present and future addressing needs.

Reference

1. What is an IP address? IPv4 vs IPv6 explained - http://www.tracemyip.org/what-is-ipv4-ipv6-address.htm



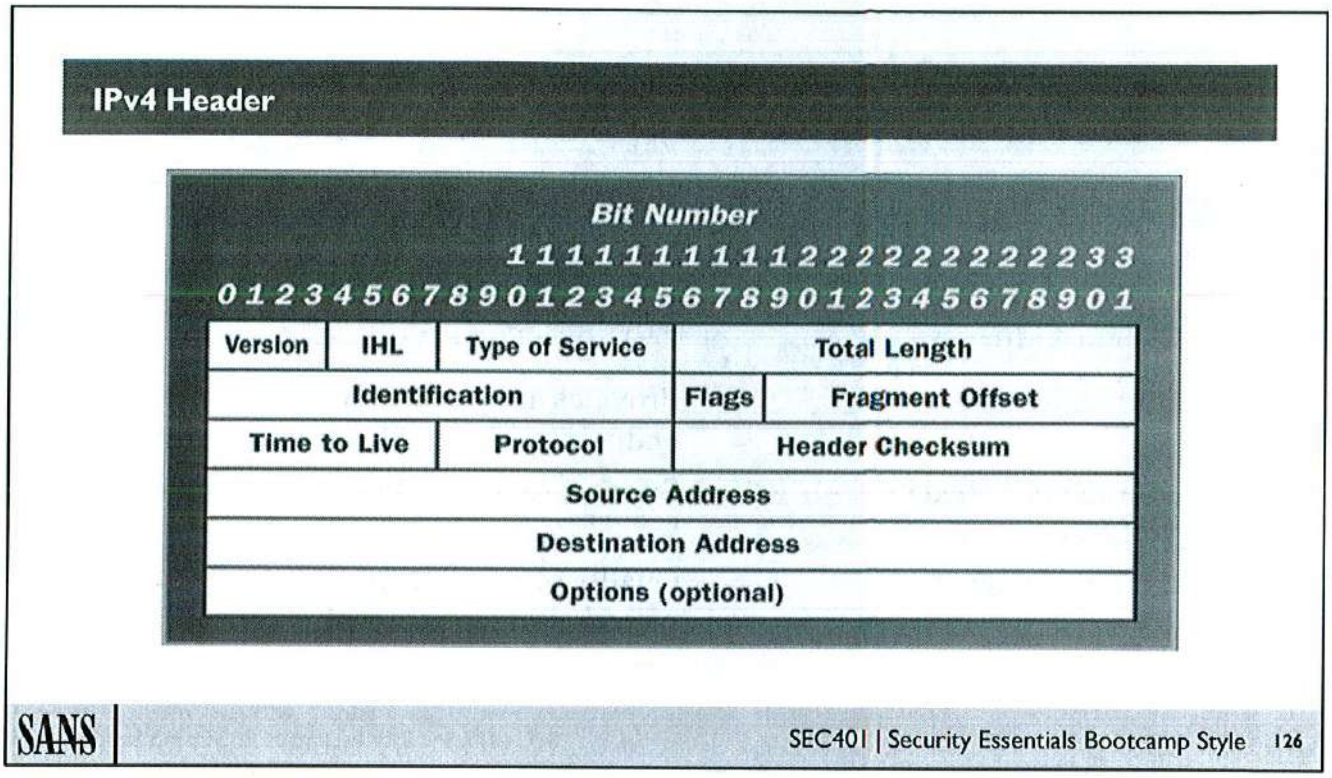
IPv4 vs IPv6 (2 of 2)

IPv4	IPv6
32-bit address 4.2 billion addresses	128-bit address 340 undecillion addresses
No authentication	Provides authentication of endpoints
Encryption provided by applications	Support for encryption in protocol
Best effort transport	Quality of Service (QoS) features provided in the protocol

IPv4 Versus IPv6

When IPv4 was conceived, it was designed to support academia and scholarly research. Nobody conceived that the Internet would be supporting e-commerce and mission-critical and real-time sensitive applications. With IPv6, consideration was made for authentication, security of the communication, and quality of service (QoS) features.

IPv6 incorporates aspects of IPsec to provide authentication of endpoints and encryption of the packets in transport. Also included in IPv6 are QoS features that permit real-time sensitive applications, such as VoIP and interactive media, to take priority over less critical packet streams.



IPv4 Header

<table>
<tr><th colspan="32">Bit Number</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>3</td><td>3</td></tr>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td></tr>
<tr><td colspan="4">Version</td><td colspan="4">IHL</td><td colspan="8">Type of Service</td><td colspan="16">Total Length</td></tr>
<tr><td colspan="16">Identification</td><td colspan="3">Flags</td><td colspan="13">Fragment Offset</td></tr>
<tr><td colspan="8">Time to Live</td><td colspan="8">Protocol</td><td colspan="16">Header Checksum</td></tr>
<tr><td colspan="32">Source Address</td></tr>
<tr><td colspan="32">Destination Address</td></tr>
<tr><td colspan="32">Options (optional)</td></tr>
</table>

SANS

Here, you see a diagram of how the bits inside an IPv4 packet header are laid out. Pay particular attention to the way the diagram is labeled, as this is the standard way of looking at a packet header. Across the top, the bits are numbered from 0 on the left to 31 on the far right, for a total of 32 bits. 32 bits equal 4 bytes. When counting within packet headers, always start counting bits and bytes starting from 0.

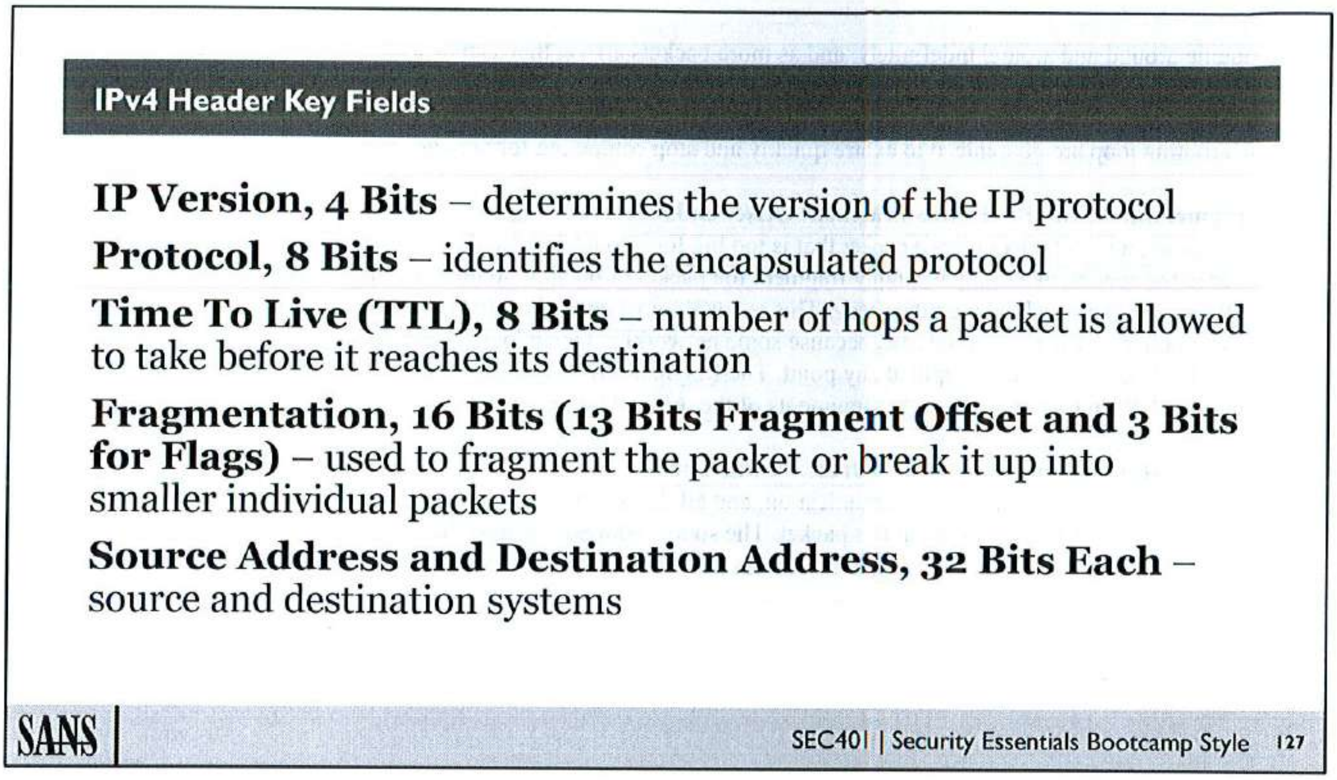
Most IP headers have no options set and will have a length of 20 bytes. The first byte is byte 0, and if no options are set, the last is byte 19.

If options are used, the header will be longer than 20 bytes. The options field can be of variable length, but must end on a 4-byte boundary.

Some IP options are

- **Record Route**: Tells a router to add its IP address to the options field.
- **IP Timestamp**: Tells a router to write a timestamp into the options field.
- **Strict Source Routing**: Allows the sender to specify the exact route a packet should take to the destination.
- **Loose Source Routing**: Allows the sender to specify a list of routers a packet must pass through. It may also traverse other routers if required.

Although options have valid uses, especially for network troubleshooting, they are rarely used by legitimate traffic.



IPv4 Header Key Fields

IP Version, 4 Bits – determines the version of the IP protocol

Protocol, 8 Bits – identifies the encapsulated protocol

Time To Live (TTL), 8 Bits – number of hops a packet is allowed to take before it reaches its destination

Fragmentation, 16 Bits (13 Bits Fragment Offset and 3 Bits for Flags) – used to fragment the packet or break it up into smaller individual packets

Source Address and Destination Address, 32 Bits Each – source and destination systems

SANS | SEC401 | Security Essentials Bootcamp Style 127

All the IP header fields are important. If a packet is missing even one, it probably will not reach its destination; or if it did, it would most likely be unusable. For our purposes, however, some fields are more important than others. Let's take a look at some of them.

As we go through these fields, keep in mind that they are part of the IP header, which resides at the internet layer. These fields are common to all IP packets, but the individual protocols that depend on IP—TCP, UDP, and ICMP—each has its own headers, which we examine later.

IP Version, 4 Bits

This contains a short integer corresponding to the Internet Protocol version used to create this packet. The most common value is 4 for IPv4. IPv6 is becoming increasingly popular and will use a value of 6.

Protocol, 8 Bits

This is another integer that denotes the exact type of IP message encapsulated in this packet. Although the meanings of the possible values are standardized, the values themselves are fairly arbitrary. For a TCP packet, expect this value to be 6 (decimal). For a UDP packet, the value is 17 (decimal). ICMP packets carry a value of 1 (decimal) in this field.

Time To Live (TTL), 8 Bits

A packet's TTL specifies how many hops a packet is allowed to take before it reaches its destination. For example, a typical TTL value of 32 says that the packet can go through a maximum of 32 routers on its way to the destination. Each time the packet passes through a router, the router subtracts one from the TTL and places this new TTL in the packet when it sends it on its way. If the new value is zero, however, instead of forwarding the packet, the router drops the packet. If the router's administrator is friendly, it sends an ICMP "Destination Unreachable" packet back to the original sender to let it know that its packet was never delivered.

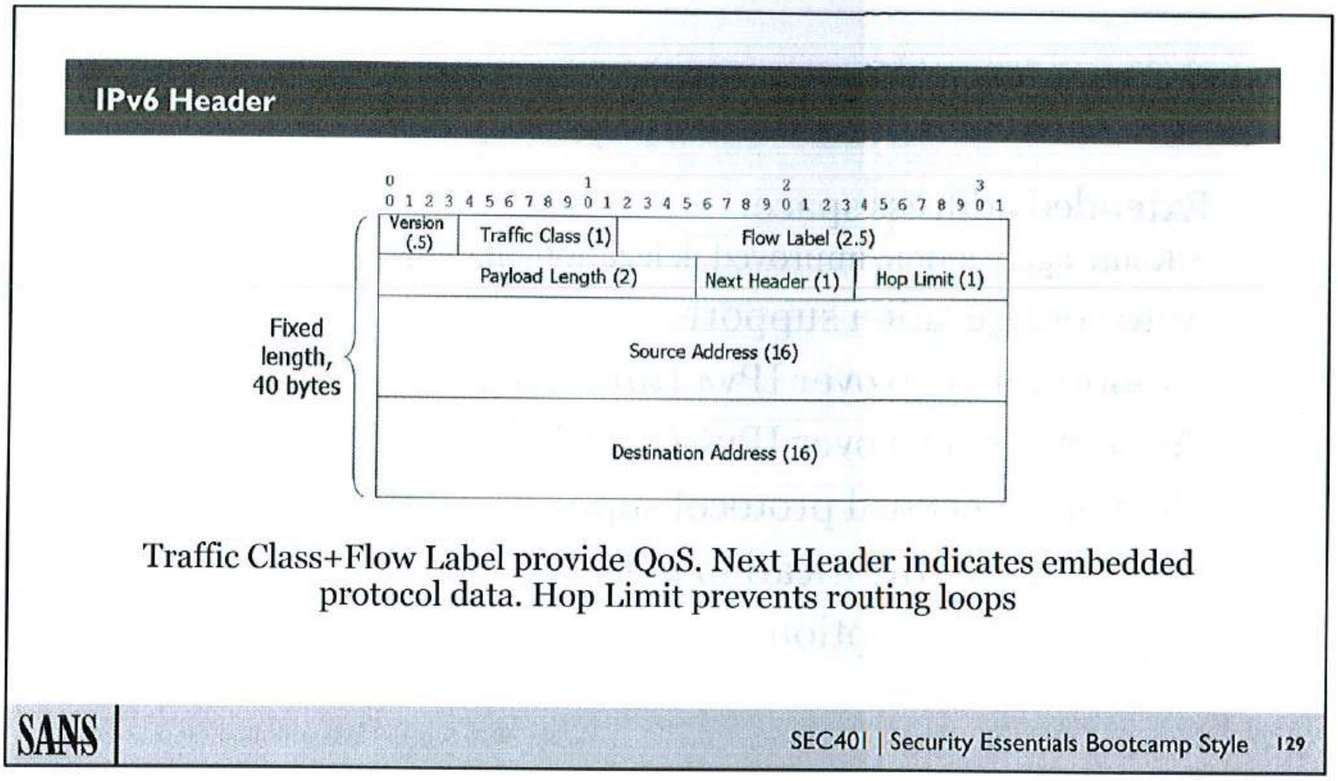
TTLs guard against routing loops, where two or more poorly configured routers repeatedly exchange the same packet over and over again in the mistaken hope of getting it to its final destination. The packets can keep looping around and around indefinitely, and as more packets arrive, they can be added to the loop. Pretty soon, the routers involved are able to do nothing more than continually try to deliver the same undeliverable packets and they are unable to accept any new traffic, bringing down that part of the network. By using TTLs, packets in a routing loop are guaranteed to expire quickly and stop competing for network resources.

Fragmentation, 16 Bits (13 Bits Fragment Offset and 3 Bits for Flags)

Sometimes, a router encounters a packet that is too big for it to retransmit all at once. Rather than signaling an error to the sender, most routers simply fragment the packet or break it up into two or more smaller individual packets, and then send them on their way. This is common when packets traverse different types of network links on their way across the Internet because some networking technologies have different maximum packet sizes. Packets can usually be split at any point. The fragment offset field tells the sender where this particular fragment falls in relation to the other fragments of the original larger packet.

Source Address and Destination Address, 32 Bits Each

Both computers involved in the communication, and all the routers and network devices between them, need to know who is talking to whom in this packet. The source address contains the IP address of the packet's sender. The destination address lists the IP address of its intended recipient.



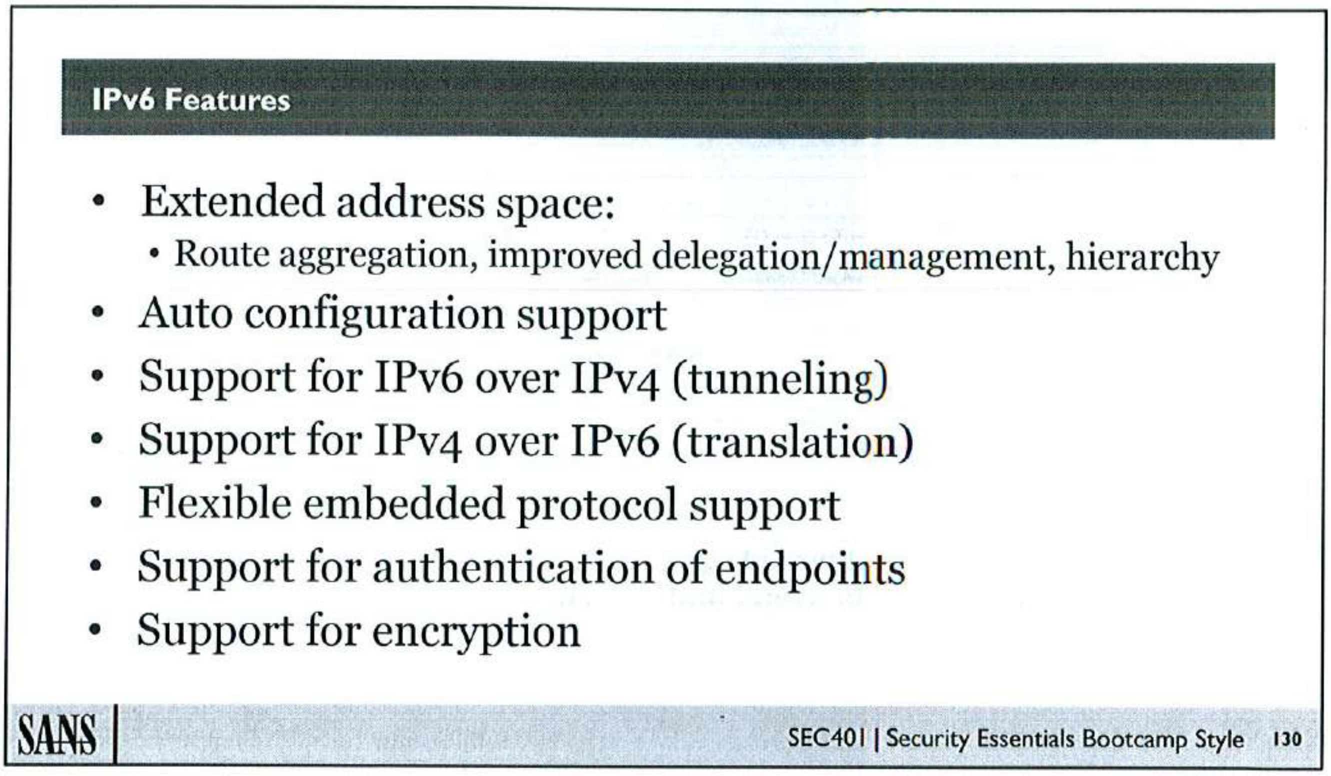
IPv6 Header

Traffic Class+Flow Label provide QoS. Next Header indicates embedded protocol data. Hop Limit prevents routing loops

IPv6 Header

To accommodate the changes in the IPv6 protocol, the header information has changed by removing superseded functionality from the IPv4 header and introducing some new fields:

- **Version**: 4 bits. The version field indicates the packet is IPv6 and is always a 6.
- **Traffic Class**: 1 byte/8 bits. The traffic class field is used to specify the priority of the packet for QoS.
- **Flow Label**: 20 bits. The flow label field is used for QoS management to convey special handling functions for the packet.
- **Payload Length**: 2 bytes/16 bits. The payload length field specifies the length of the packet in a quantity of bytes.
- **Next Header**: 1 byte/8 bits. The next header field specifies the next encapsulated protocol in the payload of the packet. The values that are assigned to IPv4 embedded protocols (such as TCP, UDP, and ICMP) are forward-compatible with the IPv6 next header field.
- **Hop Limit**: 1 byte/8 bits. The hop limit field is used to prevent routing loops by decrementing the hop limit value at each router. This is similar to the TTL field used in the IPv4 header.
- **Source Address**: 16 bytes/128 bits. This field is the source address of the IPv6 station transmitting the packet.
- **Destination Address**: 16 bytes/128 bits. This field represents the destination or recipient of the IPv6 packet.



IPv6 Features

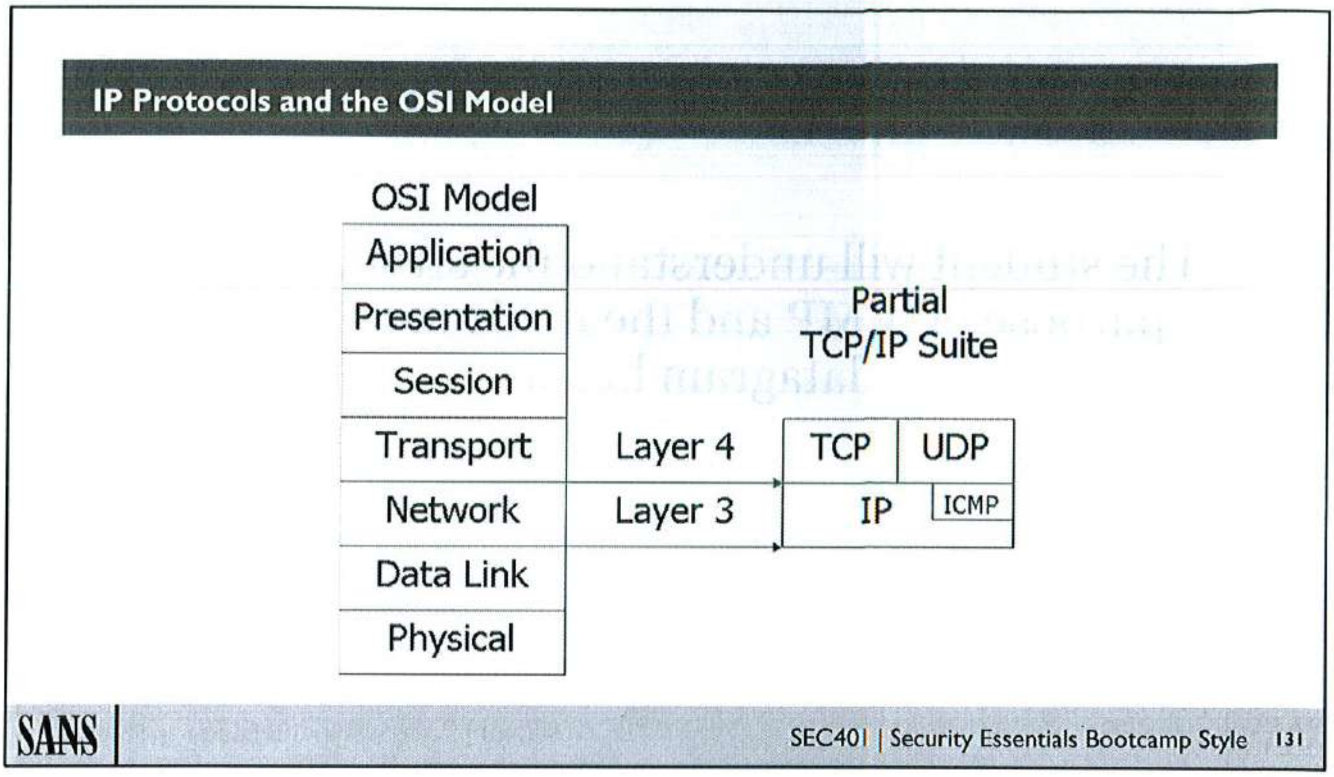
- Extended address space:
  - Route aggregation, improved delegation/management, hierarchy
- Auto configuration support
- Support for IPv6 over IPv4 (tunneling)
- Support for IPv4 over IPv6 (translation)
- Flexible embedded protocol support
- Support for authentication of endpoints
- Support for encryption

SANS | SEC401 | Security Essentials Bootcamp Style 130

A key feature of IPv6 is the expansion of address space, permitting route aggregation on core Internet routers through geographic address space allocation, improving delegation and management of addresses to organizations and ISPs alike, as well as providing hierarchical distribution of address space that makes troubleshooting and Internet routing simpler.

Another valuable feature of IPv6 is support for addressing auto configuration. Anyone who has been responsible for manually assigning IP addresses to hosts understands that this is a problematic and cumbersome process. With 128 bits of address space, it becomes possible to use the globally unique MAC addresses on all network cards as IP addresses. In this way, administrators can simply introduce a new node to an IPv6 network without manually specifying an IP address; the IP address is configured automatically based on the local MAC address and advertisement information from the default gateway on the network.

Another significant change in the IPv6 protocol is the use of a fixed IP header. Although the IPv4 header can expand to include additional information, such as strict or loose source routing, the IPv6 protocol has a fixed header length of 40 bytes. In order to accommodate additional flexibility in the protocol, IPv6 introduces a next header field that indicates what the embedded protocol contained in the packet payload is. This is similar to IPv4's embedded protocol field, but unlike this field, the next protocol can include multiple embedded protocol fields, one right after another.



IP Protocols and the OSI Model

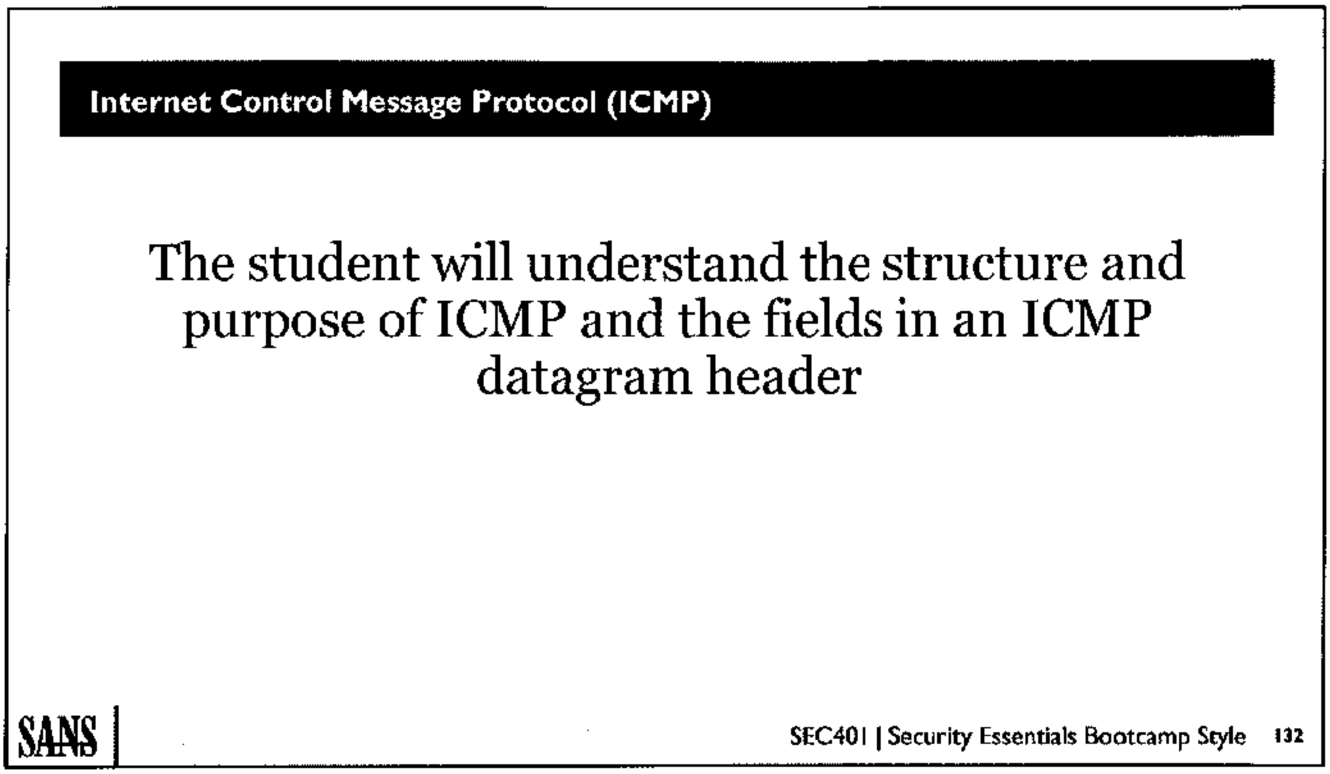
OSI Model

OSI Model		Partial TCP/IP Suite
Application		
Presentation		
Session		
Transport	Layer 4	TCP, UDP
Network	Layer 3	IP, ICMP
Data Link		
Physical		

Before we begin our discussions of TCP, UDP, and ICMP, we must review their relationship to the Open Systems Interconnection Reference Model (OSI model).

The network layer, commonly referred to as Layer 3 of the OSI model, is responsible for determining routes to be taken between two network devices and for handling flow control, segmentation/desegmentation, and error-control functions. ICMP performs a subset of these functions and works with the IP protocol, along with some other sub-protocols of the TCP/IP suite, to complete this functional portion of the OSI model.

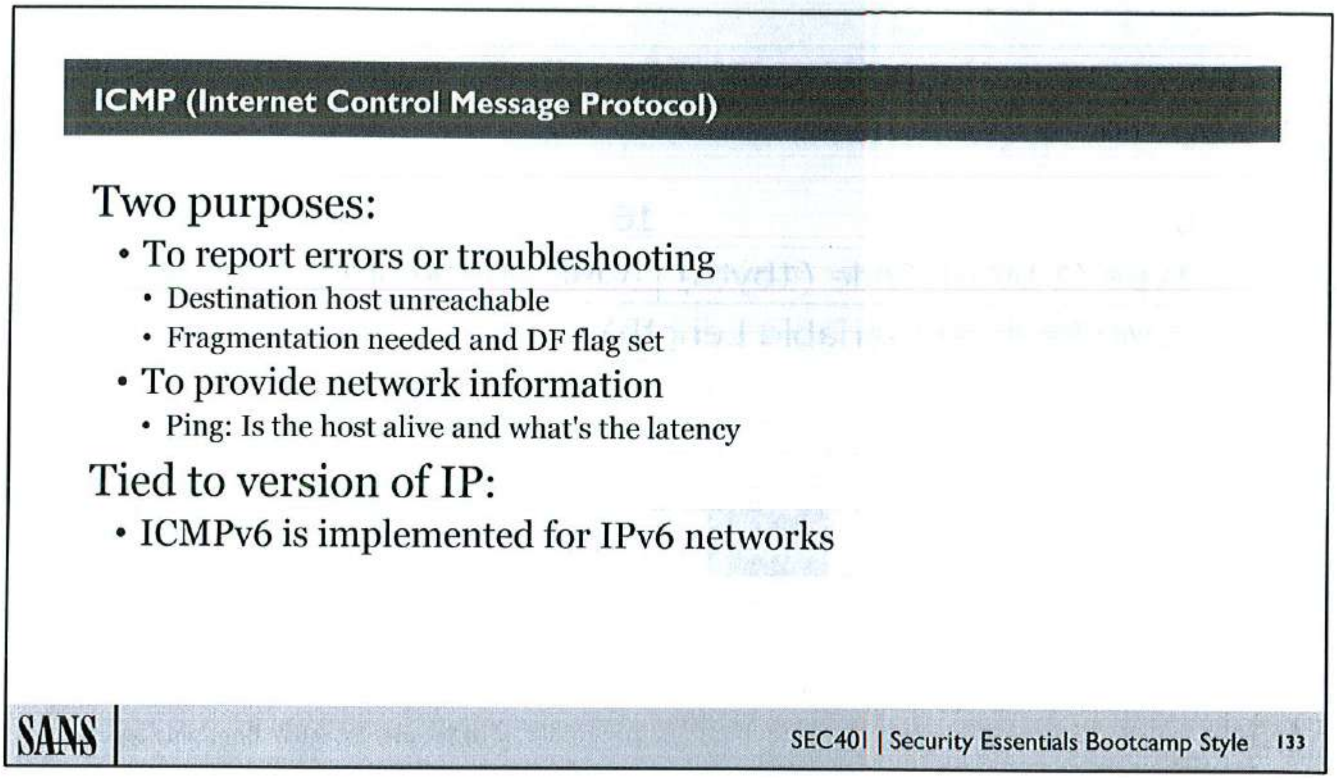
The transport layer, commonly referred to as Layer 4 of the OSI model, is responsible for the transmission of data between the two endpoint systems involved in the communication. Issues related to reliability and cost-effective data transfer belong to this layer. In the TCP/IP suite, TCP and UDP are the most used and well-known protocols that function at this layer.



Internet Control Message Protocol (ICMP)

The student will understand the structure and purpose of ICMP and the fields in an ICMP datagram header

This page intentionally left blank.



ICMP (Internet Control Message Protocol)

Two purposes:

- To report errors or troubleshooting
  - Destination host unreachable
  - Fragmentation needed and DF flag set
- To provide network information
  - Ping: Is the host alive and what's the latency

Tied to version of IP:

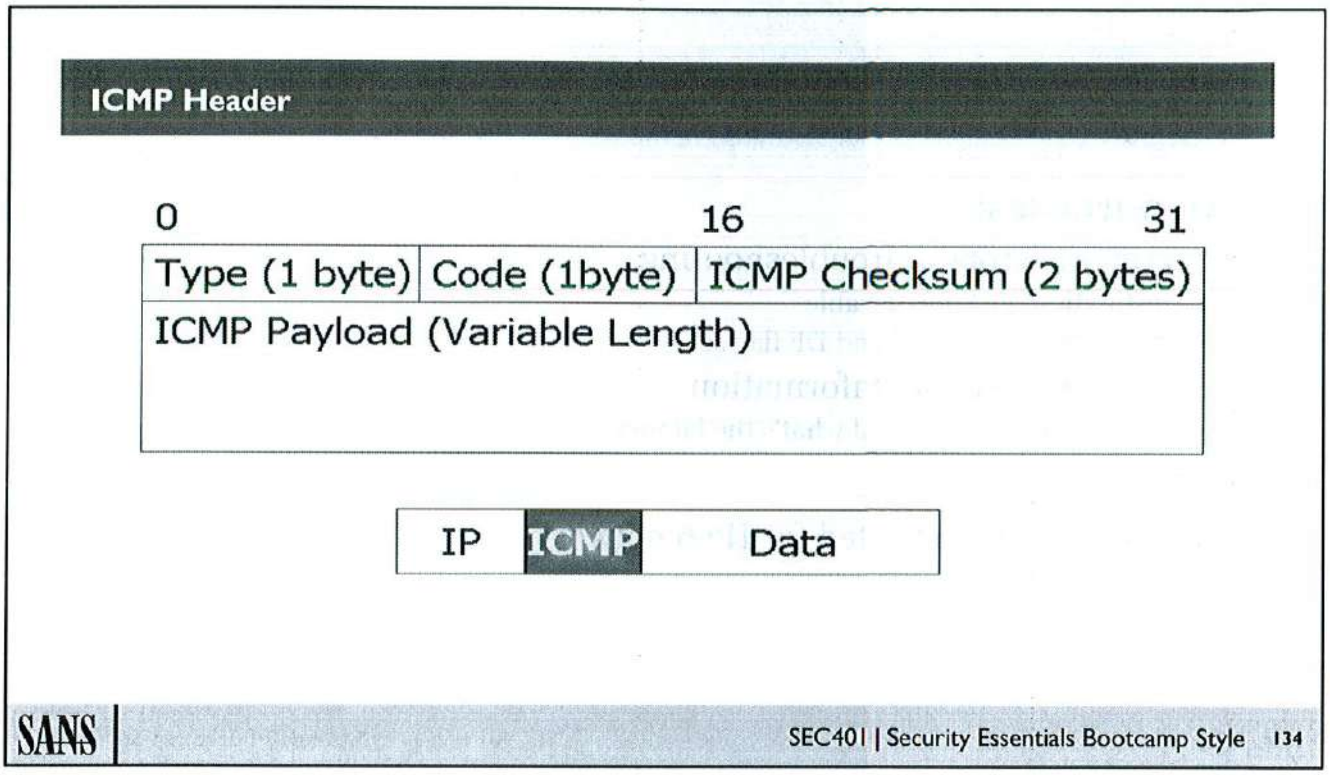
- ICMPv6 is implemented for IPv6 networks

SANS SEC401 | Security Essentials Bootcamp Style 133

ICMP is a network layer protocol, unlike TCP and UDP, which are part of the transport layer. As such, ICMP actually is a peer of IP, even though it is still encapsulated in an IP packet. In fact, IP, TCP, and UDP all rely on ICMP to provide information about network conditions, as well as for status and error messages pertaining to their transmissions.

ICMP is a simple protocol. It is datagram based, like IP and UDP. Most ICMP transactions require only one or two packets. ICMP packets have only three header fields, fewer fields even than UDP, and one of them is just a checksum!

ICMP is an important protocol because it carries critical and fundamental information about the state of a network and error conditions that occur. It is not meant as a protocol to be used for the transmission of data; rather, it is designed for error reporting and network-based troubleshooting techniques. Many of the other protocols rely upon ICMP to perform functions and communicate error conditions.



ICMP Header

0 16 31

Type (1 byte)	Code (1byte)	ICMP Checksum (2 bytes)
ICMP Payload (Variable Length)		

IP	ICMP	Data

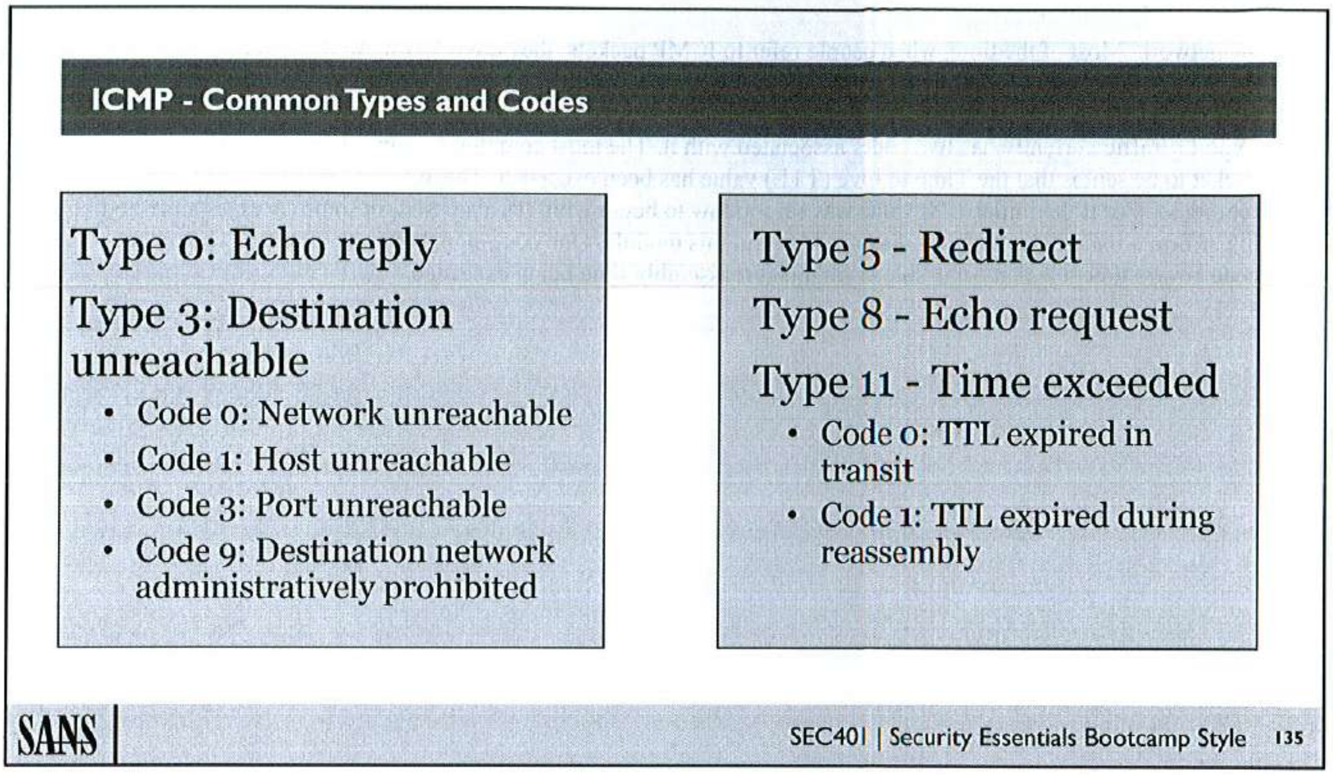
The ICMP header includes various types of fields. Each field is described here.

ICMP Type
The type field contains an integer that identifies which type of ICMP packet is being sent. There are 8 bits allocated to hold the type. Check out the IANA page (at http://www.iana.org/assignments/icmp-parameters), which lists the many defined options for the ICMP type field.

ICMP Code
The code also has a bearing on the type. For many messages, ICMP code acts as a sort of subtype. When the type field is 3, the packet is an ICMP Destination Unreachable packet. The code can give the receiver much more detailed information. A code of 3 indicates that the host is available, but the specific port requested is not listening. A code of 9 might indicate that a router or firewall rule blocked your communication to the remote host.

ICMP Checksum
The ICMP checksum is computed as a 16-bit one's complement of the header and data portion of an ICMP packet, assuming the checksum field itself is set to all zeroes.

The ICMP Payload
The content of the packet's payload might also be important to the receiver. When a host generates an ICMP error message, it always includes the entire IP header of the packet that caused the error condition. It also includes the first 8 bytes of the IP payload, which is the beginning of the TCP or UDP header containing the source and destination ports. This lets the original sender know exactly which packet caused the error and, consequently, to which application it should deliver the error message.



ICMP - Common Types and Codes

Type 0: Echo reply

Type 3: Destination unreachable

- Code 0: Network unreachable
- Code 1: Host unreachable
- Code 3: Port unreachable
- Code 9: Destination network administratively prohibited

Type 5 - Redirect

Type 8 - Echo request

Type 11 - Time exceeded

- Code 0: TTL expired in transit
- Code 1: TTL expired during reassembly

SANS

SEC401 | Security Essentials Bootcamp Style 135

The types and codes depicted on the slides cover only a small portion of all the types and codes that exist. However, these are a good representative sample of the most common types of ICMP packets that we need to familiarize ourselves with.

A Type 0 packet currently has no codes that carry any special meaning. This packet is typically referred to as a ping response to an ICMP Type 8 packet. The Type 0 packet is used to tell us that the remote host is reachable on the network and the amount of latency (measured by the sending host) to that device.

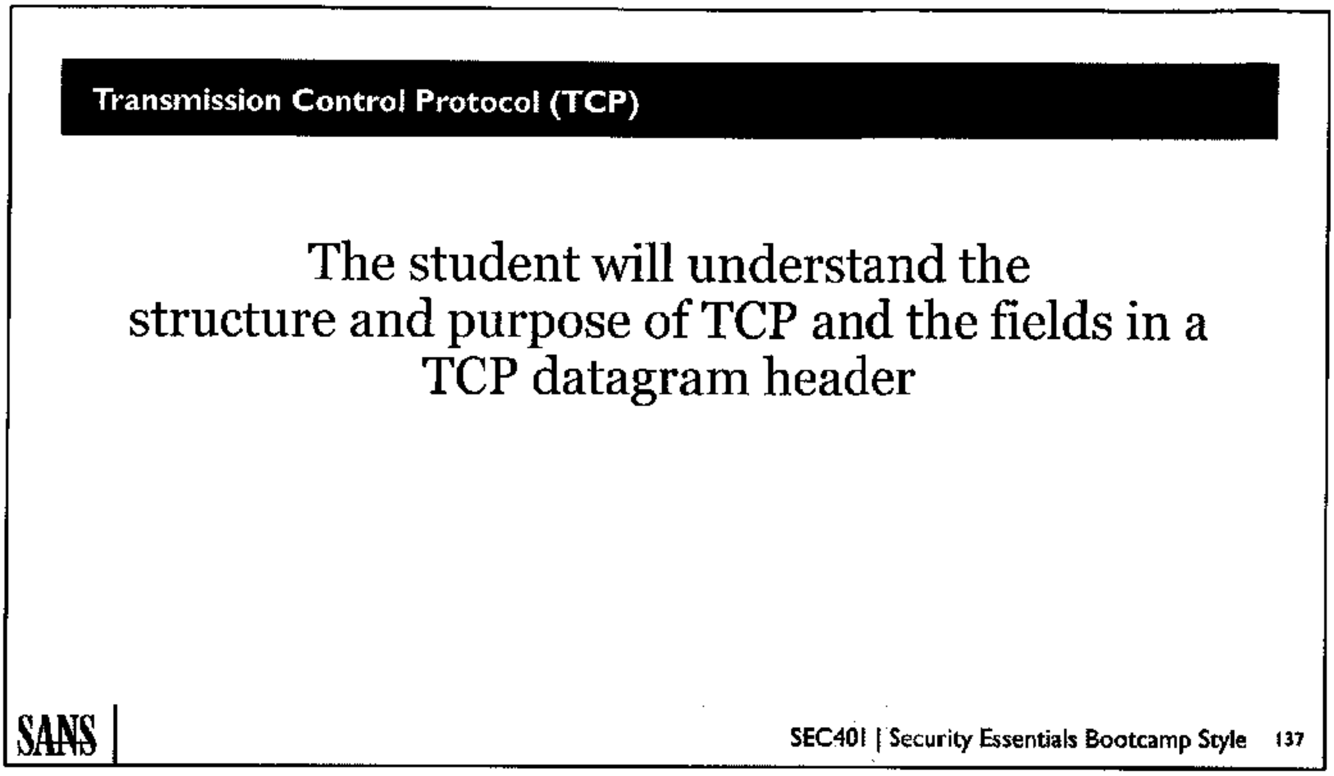
At this time, there is a total of 15 codes associated with the Type 3 ICMP type. There can be many reasons why a destination may be unreachable. The code value is meant to provide more information as to the reason that the remote host cannot be reached. If a router is unable to pass a packet to the next hop for a destination network (such as if an ISP connection goes down), it is probable that a "network unreachable" code will be returned to the sending host. However, if the ISP connection is up but the destination host simply doesn't exist on the network, then a "host unreachable" code will be returned. Furthermore, if the host does exist on the network, but the requested UDP port (usually only seen used for UDP, as TCP has its own mechanism for handling this) is not an open port, then a "port unreachable" code may be returned.

Finally, if some type of administrative access control list prevents access to the destination network, even if the network path fundamentally exists, a "destination network administratively prohibited" code may be returned. These are just a few of the codes associated with this type.

There are currently four codes associated with a Type 5 ICMP packet. This type is used to affect the routing table on the receiving device to change where packets are sent. The Type 5 packet must be used cautiously because an attacker can potentially abuse these types of ICMP packets to redirect traffic to a location where he or she can easily view and manipulate them. In many cases, routers are configured to ignore ICMP redirect packets for just this reason.

A Type 8 packet is used to elicit a response. Preferably, the response will be an ICMP Type 0 packet, but due to any number of issues, the response could be a variety of other ICMP type/codes depending on the status of the network. Most of the time, when people refer to ICMP packets, they are referring to them in the context of the ping protocol that works mostly with Type 8 and Type 0 traffic.

Type 11 traffic currently has two codes associated with it. The most common reason for an ICMP Type 11 packet to be sent is that the Time to Live (TTL) value has been exceeded. This may occur in the event of a routing loop or if the initial TTL value was set too low to begin with. It's also used, in some cases, as a method to perform a traceroute, which is discussed later in this module. One code signals this occurrence. The other code covers timeouts that occur due to fragment reassembly time being exceeded.

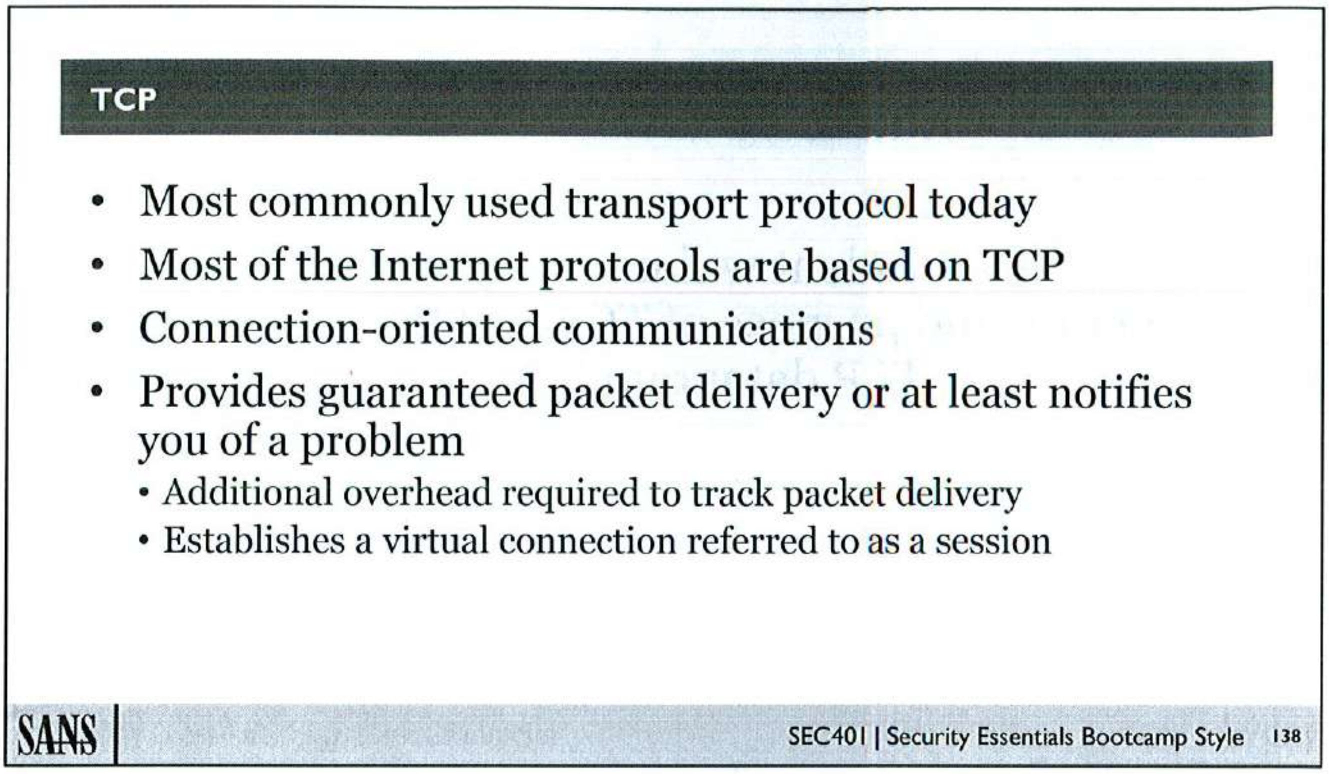


Transmission Control Protocol (TCP)

The student will understand the structure and purpose of TCP and the fields in a TCP datagram header

SANS | SEC401 | Security Essentials Bootcamp Style 137

This page intentionally left blank.



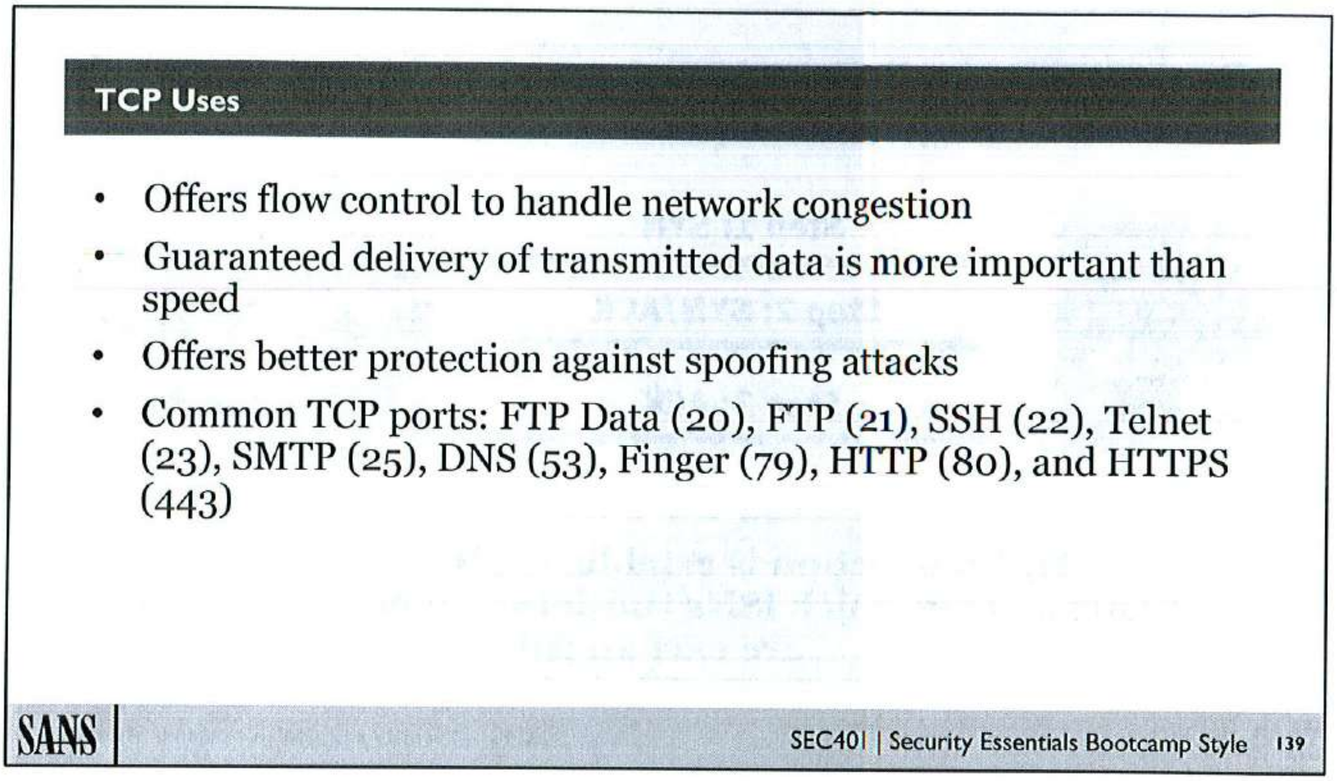
TCP

- Most commonly used transport protocol today
- Most of the Internet protocols are based on TCP
- Connection-oriented communications
- Provides guaranteed packet delivery or at least notifies you of a problem
  - Additional overhead required to track packet delivery
  - Establishes a virtual connection referred to as a session

SANS

TCP is the most commonly used transport layer protocol today. It establishes a virtual connection, often referred to as a session, between the hosts. Unlike UDP, which blindly sends datagrams and hopes they arrive, TCP can guarantee that the packet arrives or at least that it notifies you of a problem. Because of this guarantee, TCP often is a network programmer's protocol of choice. It is probably the easier of the two protocols to program for because most of the error handling is down inside the transport layer and out of sight from the application code. TCP is especially useful for any application in which there are more than one or two network hops between two computers because more hops equal more chances for errors to be introduced into the communication.

Most of the Internet protocols you use every day are based on TCP. Some examples include HTTP (HyperText Transfer Protocol), used by web servers and browsers; FTP (File Transfer Protocol), used to transfer files to and SSH (secure shell) used for making a secure connection to a system.



TCP Uses

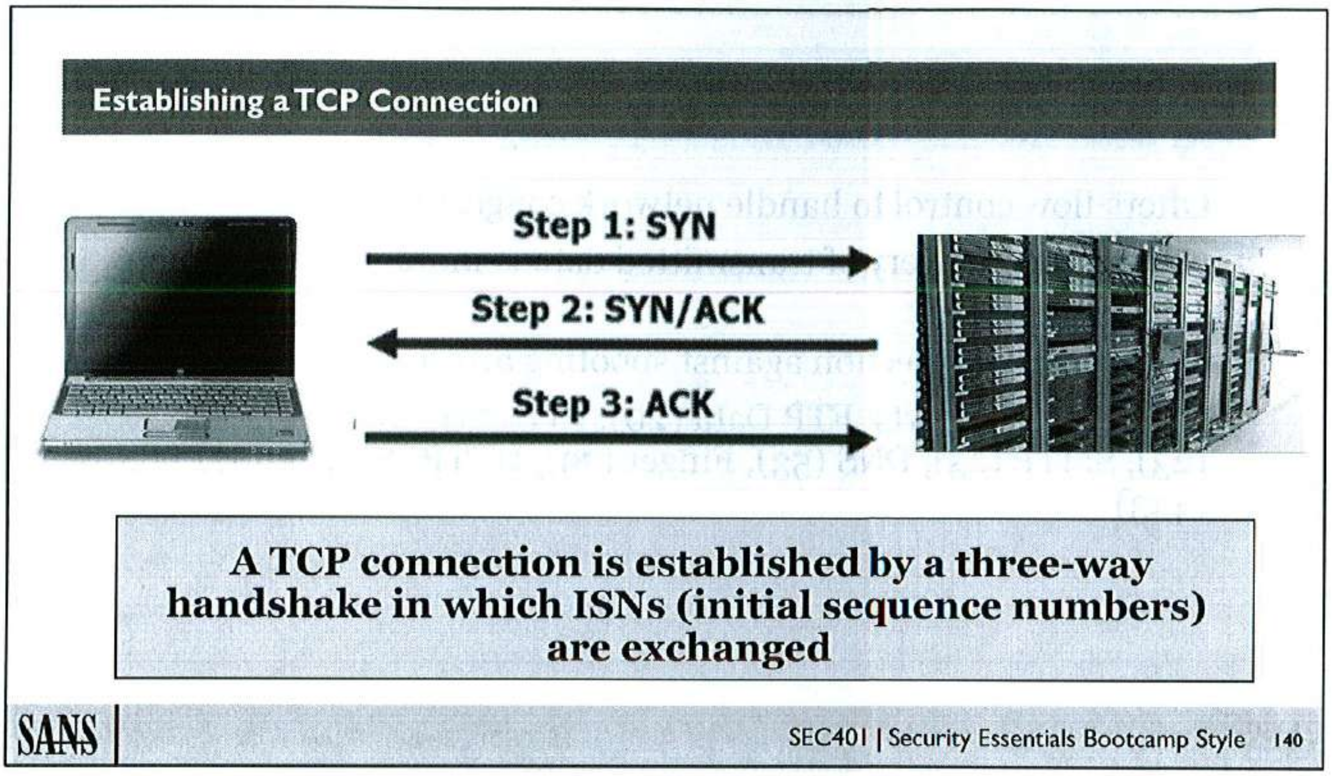
- Offers flow control to handle network congestion
- Guaranteed delivery of transmitted data is more important than speed
- Offers better protection against spoofing attacks
- Common TCP ports: FTP Data (20), FTP (21), SSH (22), Telnet (23), SMTP (25), DNS (53), Finger (79), HTTP (80), and HTTPS (443)

There are many reasons that TCP is more popular than UDP. In fact, we mainly have TCP to thank for the fact that the Internet is as reliable as it is. Without the congestion control capabilities of TCP, it is probable that systems attached to the Internet would send so much data that the accumulation of all hosts on the Internet would cause a fundamental failure due to inadequate bandwidth capabilities on the Internet backbone. Because of this congestion control, systems are able to self-regulate their bandwidth usage in order to adapt to changing conditions of bandwidth capacity.

Another important feature is that TCP allows for more data to be sent in a single packet than UDP. This is especially helpful at reducing the overall processor overhead when transmitting large amounts of data between two hosts.

Likely, the biggest reason that TCP is the most popular transport protocol is the built-in capabilities that work to ensure all the data for a particular session is received. This is achieved by error detection and efficient methods to resend missing data and reconstruct the data in the order in which it is intended with confidence that the entire data stream has been received.

Application developers generally prefer TCP to UDP due to a reduced requirement that higher-level applications validate that information was received in the same order and without error in transmission when compared with the way it was sent. Because these details are handled as part of the TCP mechanisms, developers can put this concern aside and spend more time working on the core functionality of the application.



Establishing a TCP Connection

Step 1: SYN

Step 2: SYN/ACK

Step 3: ACK

A TCP connection is established by a three-way handshake in which ISNs (initial sequence numbers) are exchanged

SANS

SEC401 | Security Essentials Bootcamp Style 140

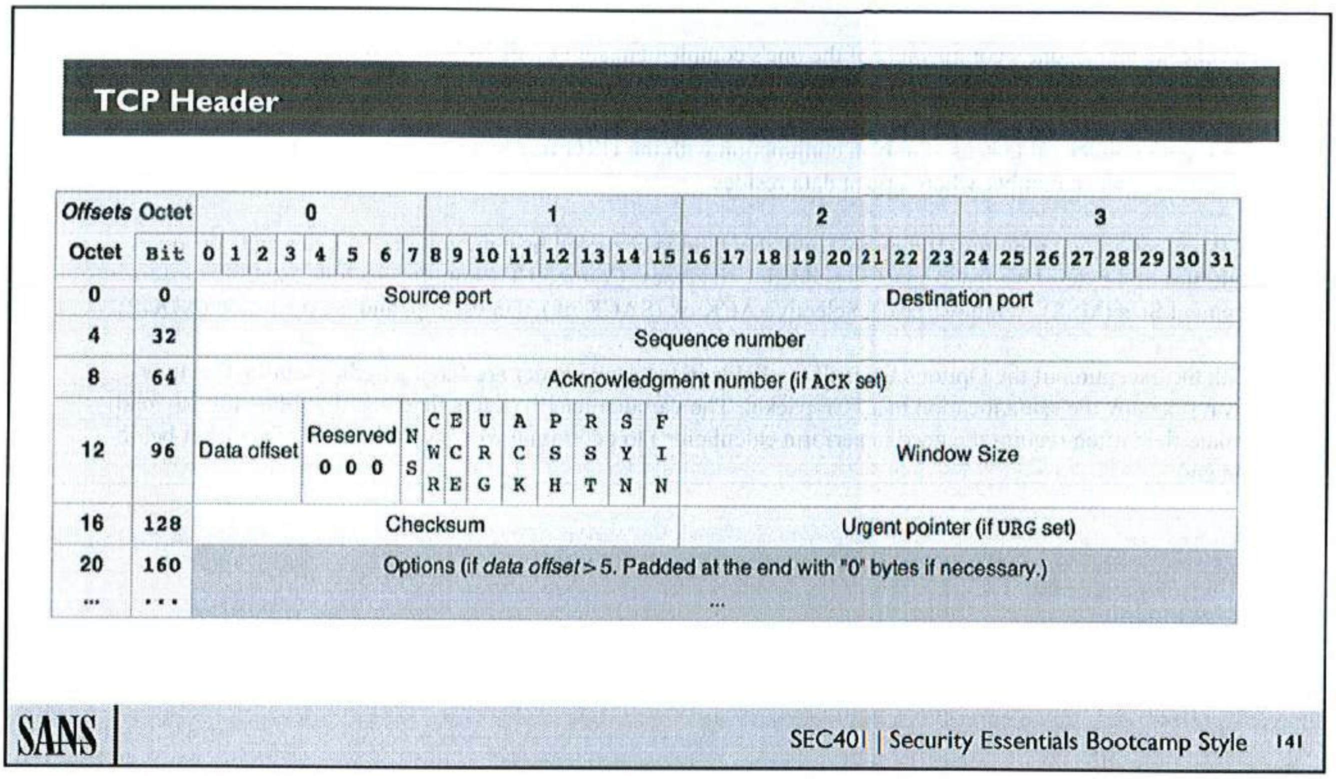
TCP connections are established using the three-way handshake. This procedure is required before the two hosts can exchange any data. In the three-way handshake, segments are often named after the flags they have set. Therefore, a segment containing a lone SYN segment is also called a SYN, and a lone ACK segment is called an ACK. A segment with both SYN and ACK is called a SYN-ACK.

The client initiates the three-way handshake by sending a SYN to signal a request for a TCP connection to the server. Then, if the server is up and offering the desired service, it can accept the incoming connection and respond to the SYN. The response consists of both an acknowledgment of the client's initial connection request (the ACK flag is set) and a connection request of its own (the SYN flag is set), together in a single packet (a SYN-ACK).

Finally, after the client receives the SYN-ACK, it sends a final ACK to the server. After the server receives the ACK, the three-way handshake is complete, and the connection has been established. The two servers can then exchange data.

After a connection is established, the ACK flag is set for every packet. As a result, the presence of the ACK can indicate whether a connection has been established or not. In fact, simple packet filters allow all packets with ACK set and assume that they are part of an established connection. It is trivial to circumvent such a filter by crafting a packet with the ACK bit set. This technique is often used to probe a network behind a filtering device and called an ACK scan.

To minimize traffic, ACKs are "piggy-backed" (as frequently as possible) onto packets containing data, as opposed to sending a packet with just an ACK. The ACKs confirm to the client and server that both ends are still using the connection.



TCP Header

<table>
<tr><td colspan="2"><i>Offsets</i> Octet</td><td colspan="8">0</td><td colspan="8">1</td><td colspan="8">2</td><td colspan="8">3</td></tr>
<tr><td>Octet</td><td>Bit</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td></tr>
<tr><td>0</td><td>0</td><td colspan="16">Source port</td><td colspan="16">Destination port</td></tr>
<tr><td>4</td><td>32</td><td colspan="32">Sequence number</td></tr>
<tr><td>8</td><td>64</td><td colspan="32">Acknowledgment number (if ACK set)</td></tr>
<tr><td>12</td><td>96</td><td colspan="4">Data offset</td><td colspan="3">Reserved
0 0 0</td><td>NS</td><td>CWR</td><td>ECE</td><td>URG</td><td>ACK</td><td>PSH</td><td>RST</td><td>SYN</td><td>FIN</td><td colspan="16">Window Size</td></tr>
<tr><td>16</td><td>128</td><td colspan="16">Checksum</td><td colspan="16">Urgent pointer (if URG set)</td></tr>
<tr><td>20
...</td><td>160
...</td><td colspan="32">Options (if <i>data offset</i> &gt; 5. Padded at the end with "0" bytes if necessary.)
...</td></tr>
</table>

SANS | SEC401 | Security Essentials Bootcamp Style 141

Because TCP is a much more heavyweight protocol than UDP, it requires a much larger header. The normal TCP header is 20 bytes! Because most TCP implementations also specify options, it can grow even larger. From a security standpoint, some of these fields are more important than others. Let's look at some of the key elements of the TCP header.

The source port indicates the port on the sender that the receiver should reply to, while the destination port indicates the service on the receiver to which the packet should be delivered. Valid port numbers are 1 through 65,535.

The sequence number is a value used to indicate the first data octet for the segment being sent.

The acknowledgment number indicates the value of the next sequence number the sender is expecting to receive from the other party to the communication.

Data Offset (or header length) indicates the number of 32-bit words (4 bytes) that are contained in the TCP header for this packet. The minimum decimal value for this field is 5 to indicate 20 bytes of information (4 bytes * 5 = 20 bytes).

When the TCP protocol was initially created, there were 6 bits reserved for future use. Since that time, the last 3 bits in this segment have been identified for explicit congestion notification (ECN) usage.

These 6 bits signal whether specific TCP flags are on or off. The flags in order are Urgent (URG), Acknowledgment (ACK), Push (PSH), Reset, (RST), Synchronize (SYN), and Finish (FIN). There are three additional bits located at the most significant position of the flag's byte that are used for ECN (explicit congestion notification).

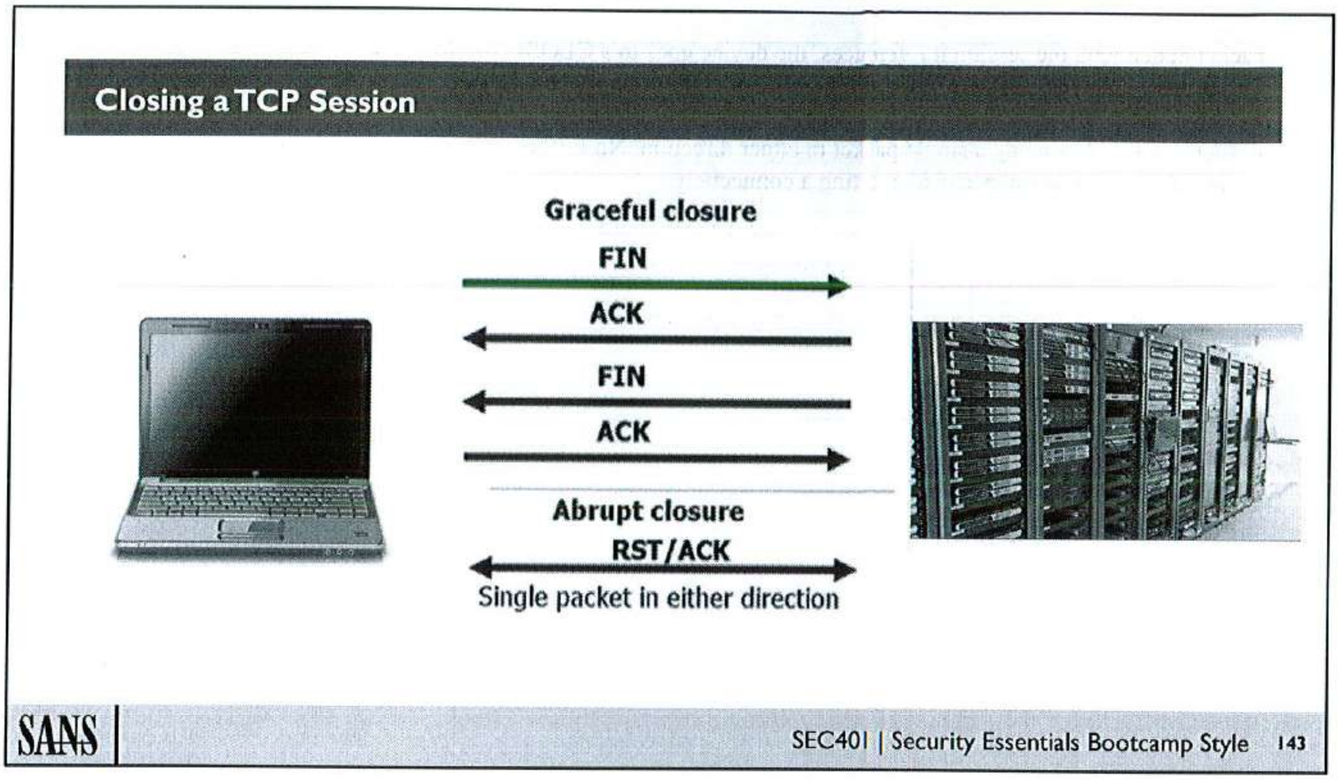
The window value indicates the number of data octets that the sender of this segment is willing to accept.

The checksum is a one's complement of the one's complement sum of all 16-bit words in the header and text for purposes of identifying data that may have been corrupted in transmission.

The urgent pointer value is used only in conjunction with the URG flag being set. This field points to an offset from the sequence number where urgent data resides.

TCP options are not required. If they are used, this field is occupied by a minimum of 4 bytes and consumes multiples of 4 bytes. Unneeded space is padded with zeroes. Common TCP options include Maximum Segment Size (MSS), Windows scale, Selective ACK ok (SACK ok), Timestamp, and No operation (NOOP).

With the exception of the Options field, all the fields in the TCP header are fixed length, meaning that they always occupy the same location in a TCP packet. The variable length of this field and the following payload or data field often require the need to perform calculations to determine where certain parts of a packet begin and end.



Closing a TCP Session

Graceful closure

FIN

ACK

FIN

ACK

Abrupt closure

RST/ACK

Single packet in either direction

SANS

SEC401 | Security Essentials Bootcamp Style 143

This slide shows a sample TCP session, illustrating the two ways TCP closes connections on the network. The example assumes that a PC is connecting to a server over the network, but this same process holds true for any TCP session established between any two devices.

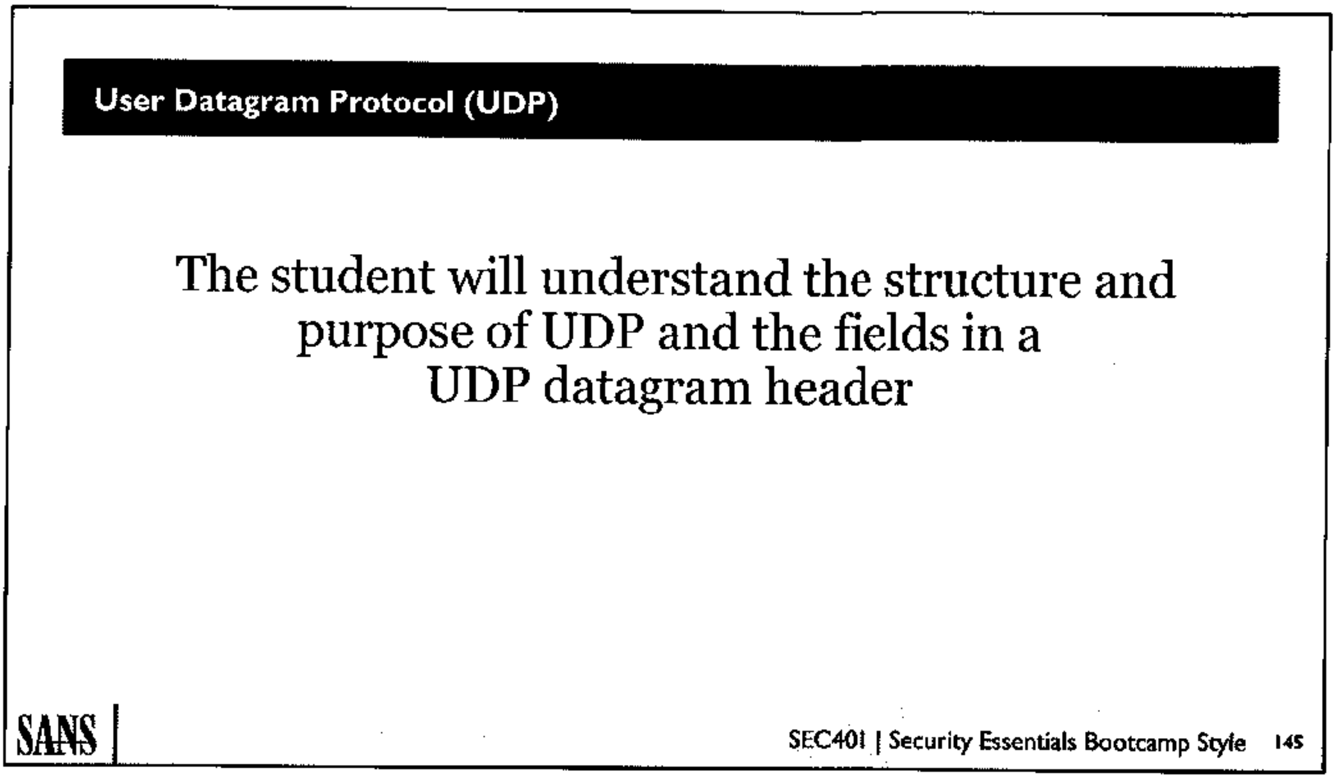
The arrows in the figure represent the direction of the communications. An arrow pointing from the PC to the server means that the PC is sending a message to the server; an arrow pointing from the server to the PC means that the server is sending a message to the PC. The RST, ACK, and FIN labels represent the different types of packets that are used during session setup and close. The RST packet is used to reset, or abruptly close, the communications. The ACK packet sends an acknowledgment of the message back to the originator. The FIN packet starts the process of finishing the connection.

The top of the slide shows how a connection is gracefully torn down. When the time comes to close the connection, each end of the connection must be closed separately. Assuming that the PC wants to close the connection first, the process starts when the PC sends a FIN packet to the server. The FIN portion indicates to the server that the PC wants to close the connection (continuing with the sequence count it has been using with the server). The server responds by sending an ACK to the PC that is acknowledging the FIN that the PC sent. Next, the server sends a FIN packet to the PC to close its side of the connection. Finally, the PC sends an ACK to the server to acknowledge the FIN.

The bottom portion of the figure illustrates a single packet abrupt closure of a TCP session. When abruptly closing a connection between two machines, either side can send a single packet to do so.

The process starts when one system or another sends a RST packet to the other. If the receiving device is in a LISTEN state for the destination port, the RST packet is ignored. If the receiving device is in the SYN-RECEIVED state, but was previously in the LISTEN state, it returns to the LISTEN state.

LISTEN state means that the system is waiting for a connection and there are no active connections to the port or no systems trying to establish a connection. This is typically how a server operates with all open ports in a LISTEN state, waiting for a client to connect. If the receiving device has an open session and the details of the packet match with the session it references, the device goes to a CLOSED state and sends no further packets for the session and advises higher level processes that the connection has been closed. Assuming a session had already been established, the sending device will set the session to a CLOSED state. Note that sessions closed in such a way are done by a single packet in either direction. No further packets are required. Closing a session in such a way is sometimes called aborting a connection.

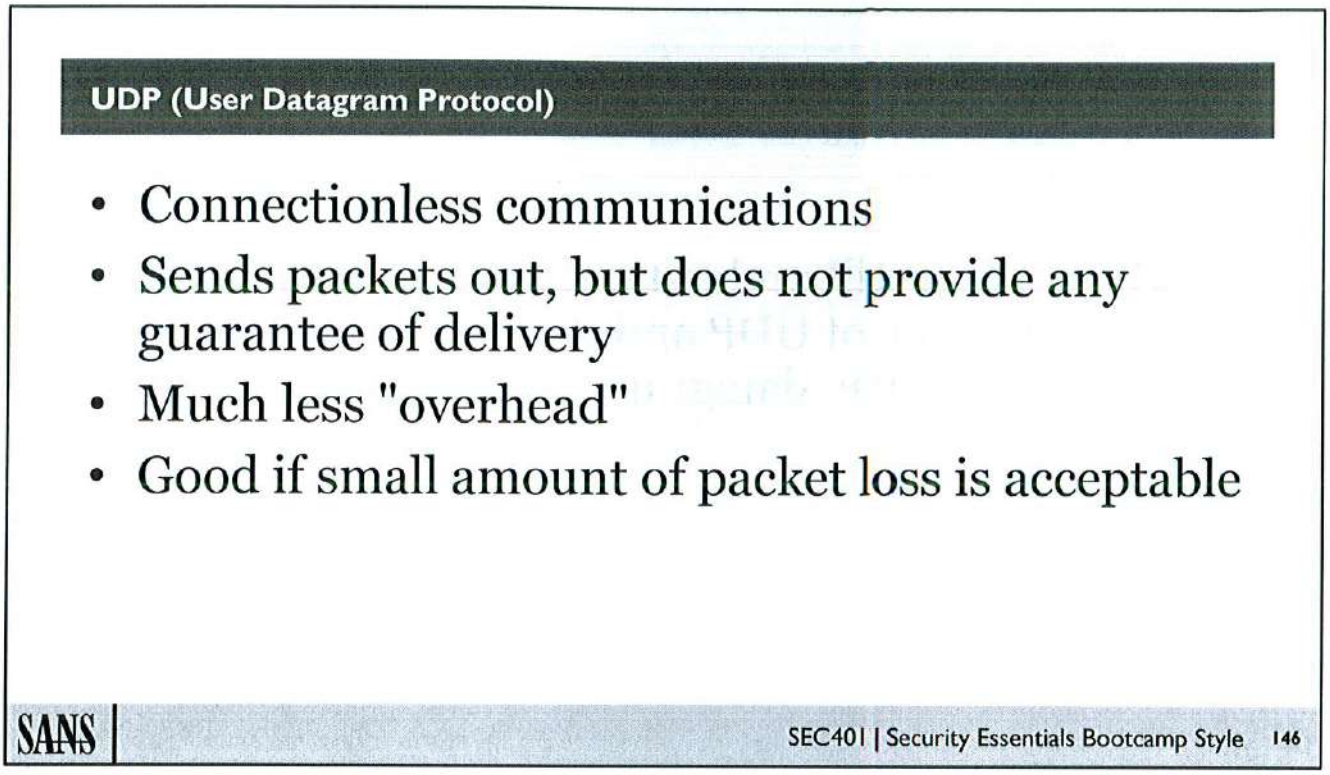


User Datagram Protocol (UDP)

The student will understand the structure and purpose of UDP and the fields in a UDP datagram header

SEC401 | Security Essentials Bootcamp Style 145

This page intentionally left blank.



UDP (User Datagram Protocol)

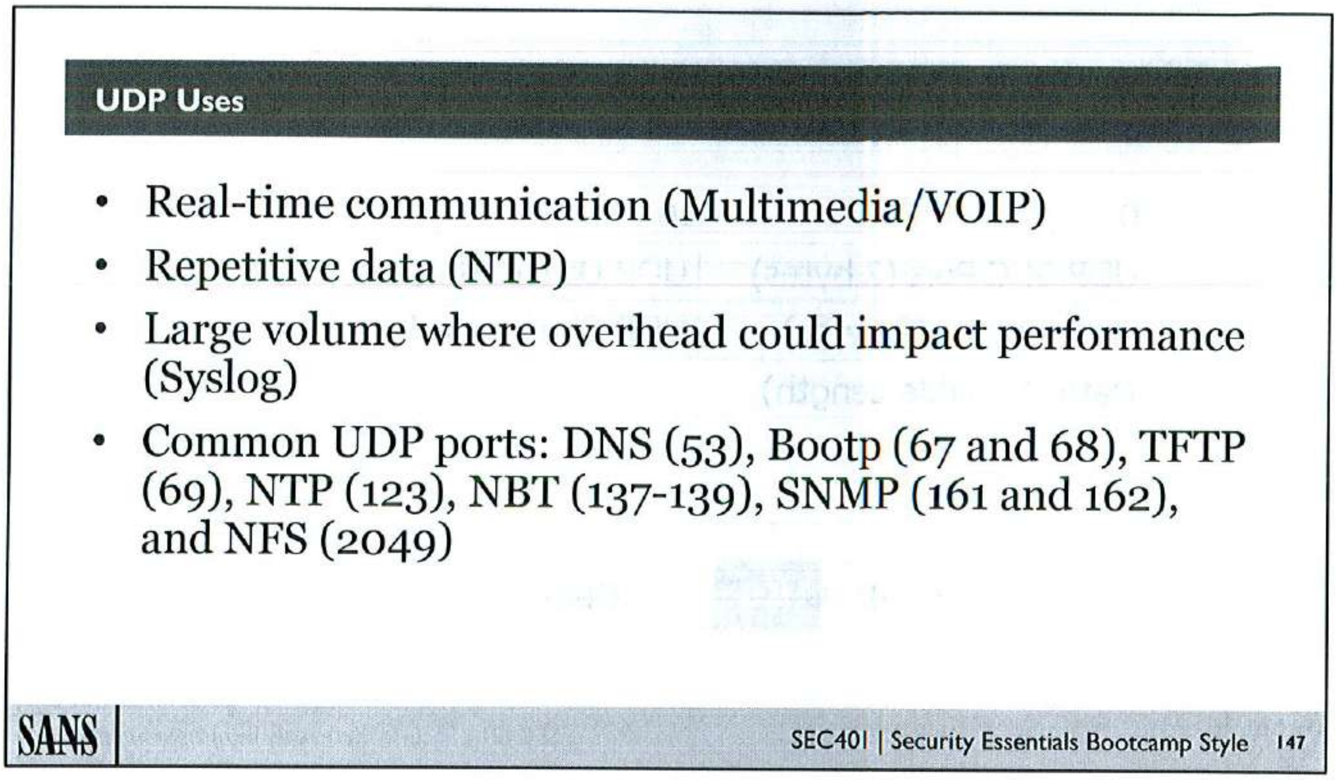
- Connectionless communications
- Sends packets out, but does not provide any guarantee of delivery
- Much less "overhead"
- Good if small amount of packet loss is acceptable

SANS | SEC401 | Security Essentials Bootcamp Style 146

UDP is the simpler of the two transport layer protocols typically used with IP. Even though it is simpler, UDP is a useful, important protocol in common use by many applications today.

UDP's goal is to be a fast, efficient protocol for reliable networks. In other words, it tries to achieve greater overall throughput by sacrificing a lot of computationally expensive error checking. Unlike some other protocols, UDP does not include the concept of a connection. The sender simply places a UDP packet on the wire without even checking to see if the receiving machine is up, let alone warning it that data is about to arrive. Furthermore, after the sender transmits a UDP packet, the sender essentially forgets about it. The sender never even confirms that the packet made it to its destination. There is also no guarantee that if the packets do arrive; they will be in the same order as they were sent. Because each packet finds its own way through the network, they often take different routes. For longer journeys, some packets inevitably arrive out of sequence; but that is the receiving application's problem, not UDP's. The UDP header does include a simple checksum that can determine if the packet was accidentally modified en route; but technically, even this is considered an optional part of the protocol (although, in practice, it should always be enabled). In short, UDP does very little error checking or exception handling of any kind.

It may sound as if UDP should be avoided because it does not perform error checking or have re-transmission capability. Statistically speaking, error checking is hardly ever needed on a fast, relatively error-free network, because almost all packets arrive in the proper order. By doing away with all the checking, UDP can transmit data at a much higher rate. Of course, the application then has to assume the extra burden of planning for exceptional conditions when they occur. The error-handling code is only invoked if an actual error occurs, rather than every time a protocol operation happens. This approach saves a lot of CPU time. Although the approach puts more of a burden on the application's programmer, it also provides them with a great deal of flexibility and power with which to work; so the tradeoff often makes sense.



UDP Uses

- Real-time communication (Multimedia/VOIP)
- Repetitive data (NTP)
- Large volume where overhead could impact performance (Syslog)
- Common UDP ports: DNS (53), Bootp (67 and 68), TFTP (69), NTP (123), NBT (137-139), SNMP (161 and 162), and NFS (2049)

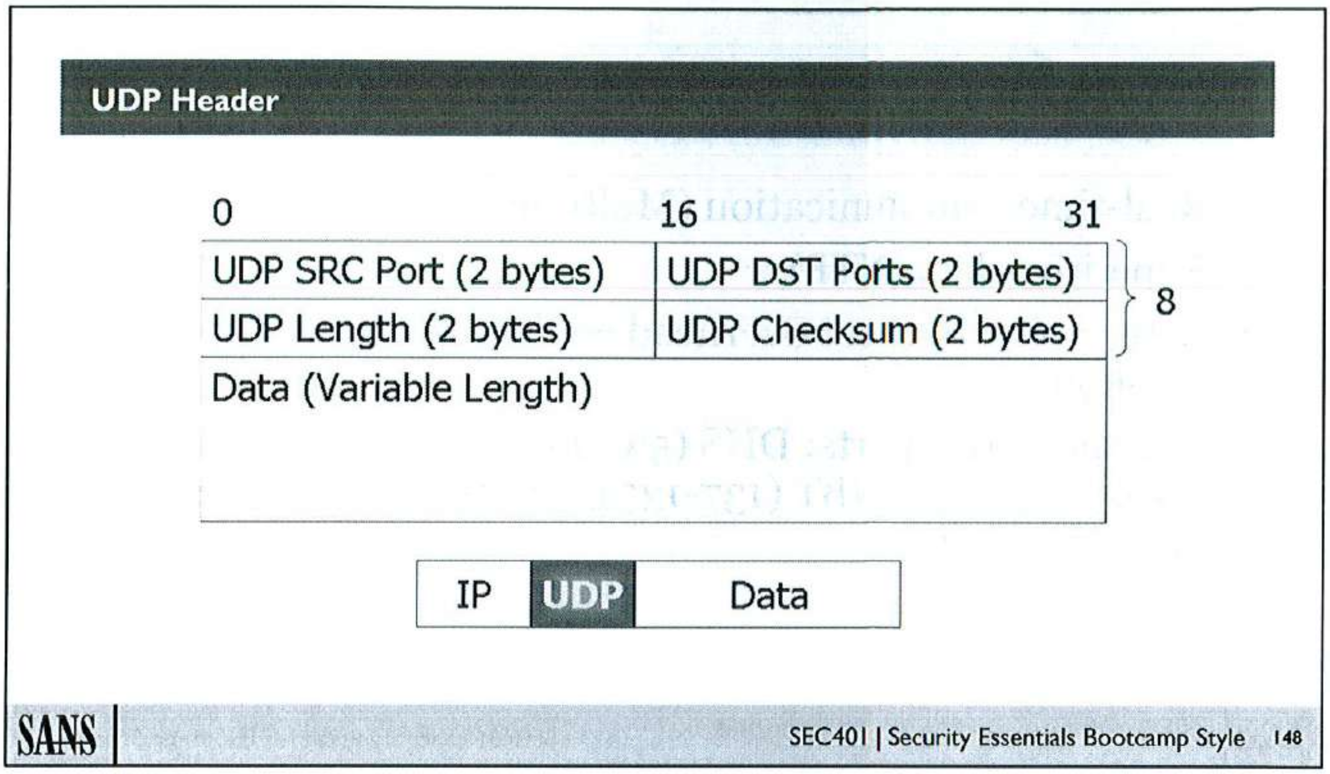
SANS SEC401 | Security Essentials Bootcamp Style 147

UDP is typically used in situations in which it is okay if some packets are lost or reordered. In a streaming-audio application, for example, each packet contains such a minuscule amount of audio data that the client probably can afford to lose one or two packets in succession without suffering a noticeable lack of quality. In addition, because it is real-time communication, re-transmitting the packets does not make sense.

Also, UDP is often used for applications that do not send much data, perhaps just a handful of bytes; so, the applications do not mind retransmitting the data if it happens to get lost. For DNS, the queries and responses usually can fit inside a single packet, so UDP is a quick and easy choice for a transport protocol. In most cases, the packets go through fine, but the loss of one, two, or even several packets poses no great problem. The time it takes to recover from the occasional dropped packet is more than made up for by the time saved by not checking for errors that rarely happen anyway. It is easy to retransmit a query if the client does not receive a response in a reasonable amount of time.

Other important UDP-based protocols include

- **Network Time Protocol** (NTP): Synchronizes time.
- **BOOTP/DHCP protocols**: Automatically configures network interfaces and load operating systems via the network when they start up.
- **Network File System** (NFS): Supports file sharing for UNIX-based networks.
- **Simple Network Management Protocol** (SNMP): Used as a management tool to query network- and server-based devices for monitoring or troubleshooting purposes.
- **Trivial File Transfer Protocol** (TFTP): Used as a method to transfer files from one device to another without requiring authentication. TFTP's most common use is in updating code on network-based devices.



UDP Header

0	16	31
UDP SRC Port (2 bytes)	UDP DST Ports (2 bytes)	8
UDP Length (2 bytes)	UDP Checksum (2 bytes)	
Data (Variable Length)		

IP	UDP	Data

As packet headers go, UDP is pretty simple. There are only four fields: source port, destination port, datagram length, and checksum. Each field is exactly 2 bytes long. A mere 8 bytes of overhead per packet is pretty good! Let's examine these fields in detail.

UDP uses the concept of ports to help get datagrams to and from the proper applications. Ports are simply ID numbers associated with certain applications running on a host. When one host wants to send datagrams to a server process running on another host, it needs to know what port that process is listening to. If we consider a computer to be similar to an apartment building, the applications running on it are its residents and the port numbers are the apartment numbers in which the residents live.

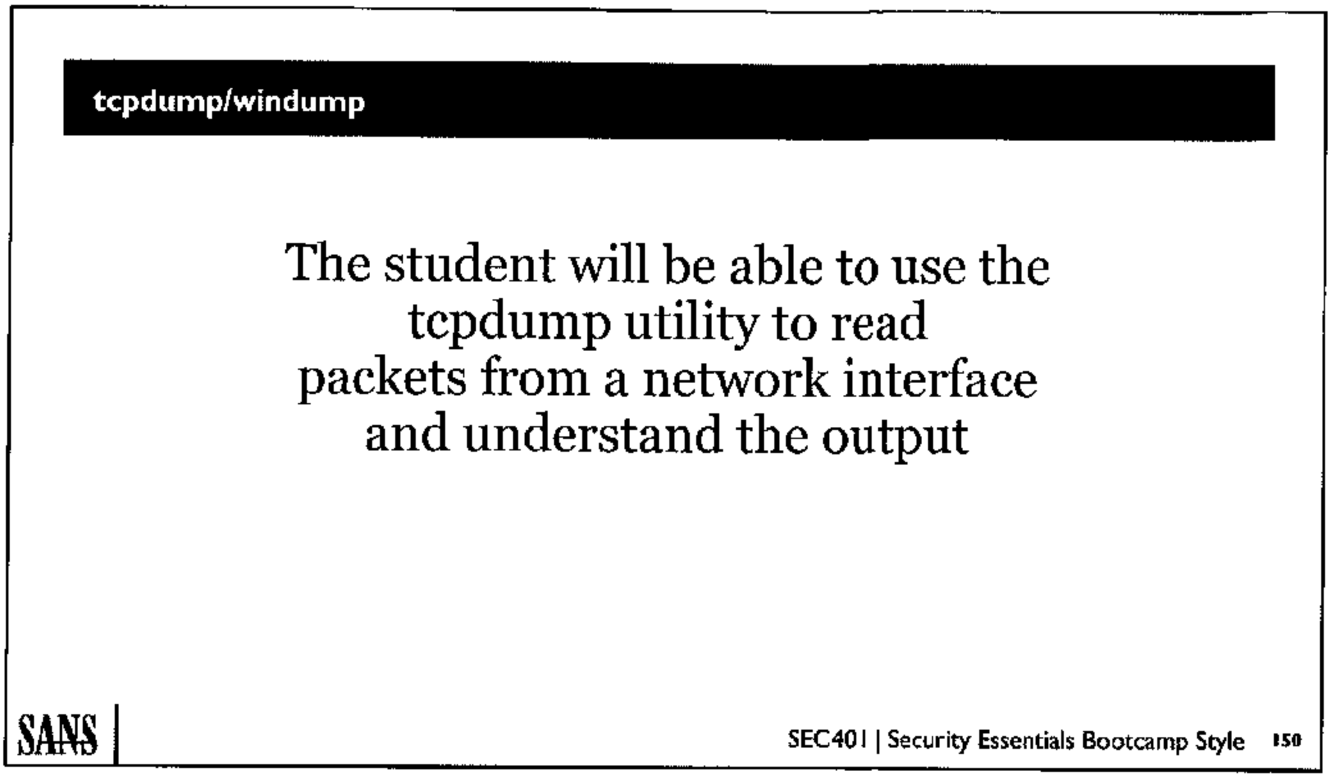
Most server ports are well-known, like DNS servers that listen to port 53 no matter how many times they are restarted or the machine is rebooted. Well-known ports are usually considered those from 1 to 1023. Clients usually use ephemeral ports; ports that change each time the client application runs. Ephemeral ports are numbered above the well-known ports (greater than 1023).

For TCP and UDP, the source port indicates the port to which the sender is bound, while the destination port indicates the service on the receiver to which the packet should be delivered. Valid port numbers are 0 through 65,535.

Datagram length is simply the length of the UDP portion of the packet, which includes the UDP header and the payload. Theoretically, a datagram could carry no data, setting the minimum value here at 8 (just the size of the header). The theoretical maximum is 65,535, although many implementations do not allow datagrams to be that long.

The datagram's checksum is technically an optional component, though almost every UDP implementation uses it. If specified, it allows the transport layer to detect when the UDP headers or the payload data (but not the IP headers, which have their own checksum) have been modified in transit. This is trivial to recompute, so an attacker interested in modifying a UDP packet has no problem doing so and then generating a new checksum. This isn't a security feature so much as a way to detect accidental transmission problems.

UDP is a great choice for a transport protocol if you have a fast, reliable network and need either high throughput or quick response times (or both). By avoiding expensive error checking, applications can take advantage of UDP's quick and responsive nature. Still, it is not a perfect protocol for all uses. Its greatest strength is also one of its greatest weaknesses. Because it does no error checking of its own, the application programmer must take up this burden.

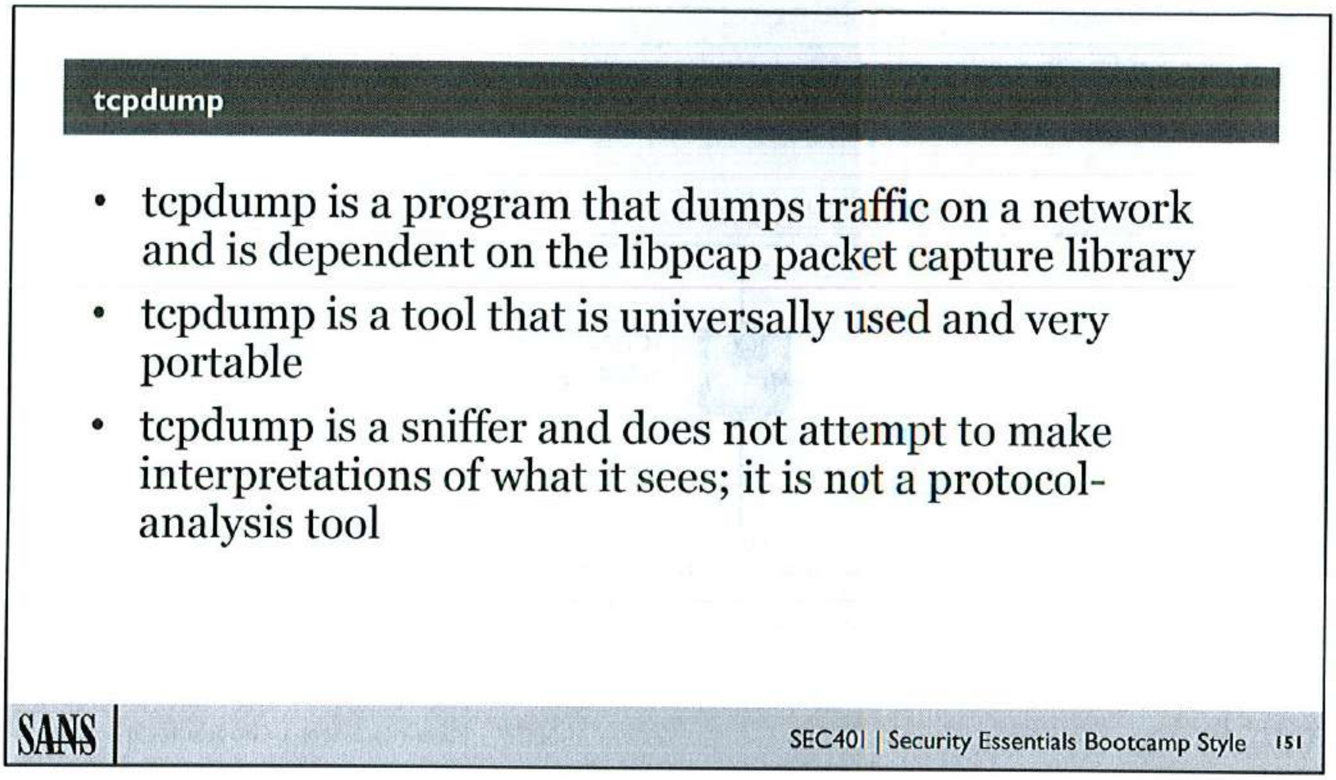


tcpdump/windump

The student will be able to use the tcpdump utility to read packets from a network interface and understand the output

SANS | SEC401 | Security Essentials Bootcamp Style 150

This page intentionally left blank.



tcpdump

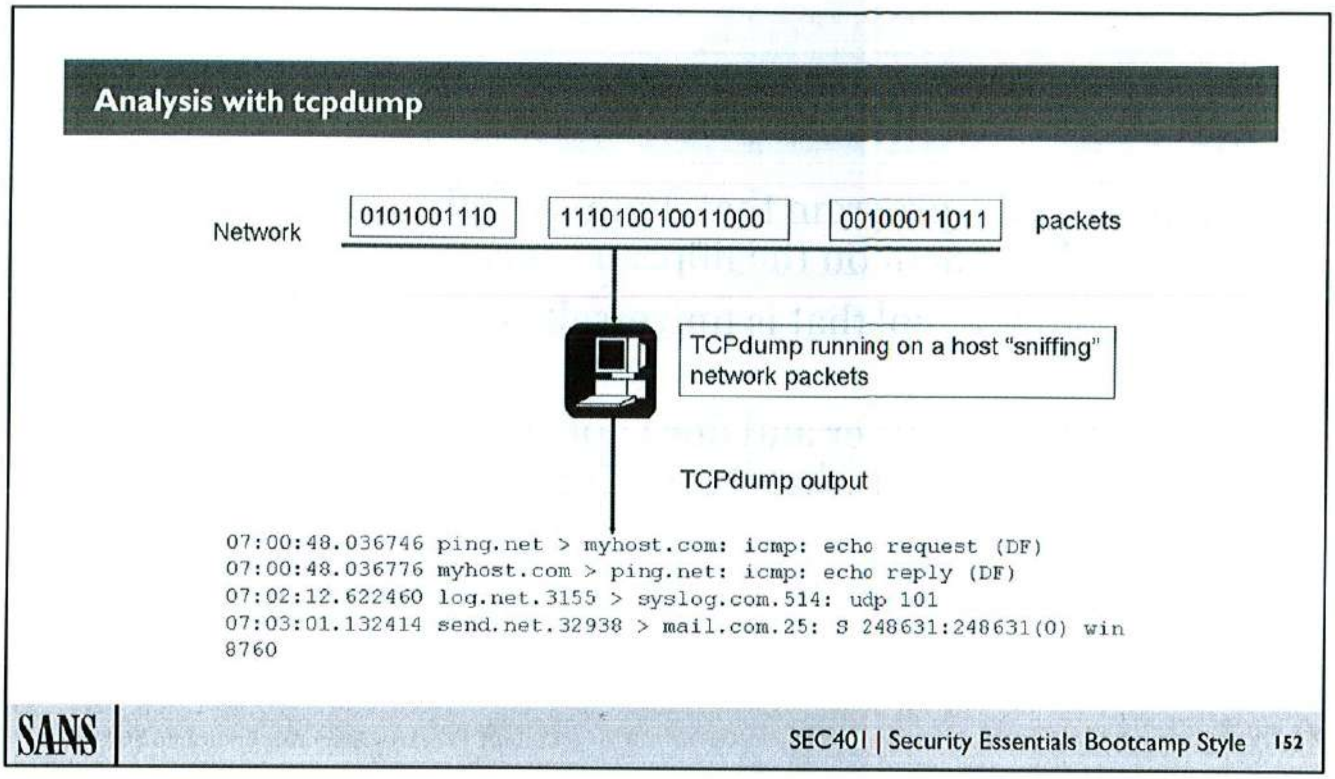
- tcpdump is a program that dumps traffic on a network and is dependent on the libpcap packet capture library
- tcpdump is a tool that is universally used and very portable
- tcpdump is a sniffer and does not attempt to make interpretations of what it sees; it is not a protocol-analysis tool

SANS | SEC401 | Security Essentials Bootcamp Style 151

One of the most popular sniffers available is tcpdump, which runs primarily on UNIX and even ships with a few UNIX distributions. A Windows port, windump, is also available. Both are free.

tcpdump's command-line interface is not fancy but does provide a versatile filtering language. Command-line options can be used to control tcpdump's behavior. After an overview of filters and command-line options, we move on to actual tcpdump analysis of UDP, TCP, and ICMP traffic. Because each protocol behaves differently on the network, tcpdump uses a different format for displaying them.

tcpdump is a tool that is universally used and very portable. It can be used on just about any platform to assist you in the analysis of network traffic. It does not attempt to make interpretations of what it sees; therefore, you, the analyst, have to have the training and knowledge to interpret the data.



Analysis with tcpdump

Network

0101001110 | 111010010011000 | 00100011011

packets

TCPdump running on a host "sniffing" network packets

TCPdump output

```
07:00:48.036746 ping.net > myhost.com: icmp: echo request (DF)
07:00:48.036776 myhost.com > ping.net: icmp: echo reply (DF)
07:02:12.622460 log.net.3155 > syslog.com.514: udp 101
07:03:01.132414 send.net.32938 > mail.com.25: S 248631:248631(0) win
8760
```

SANS

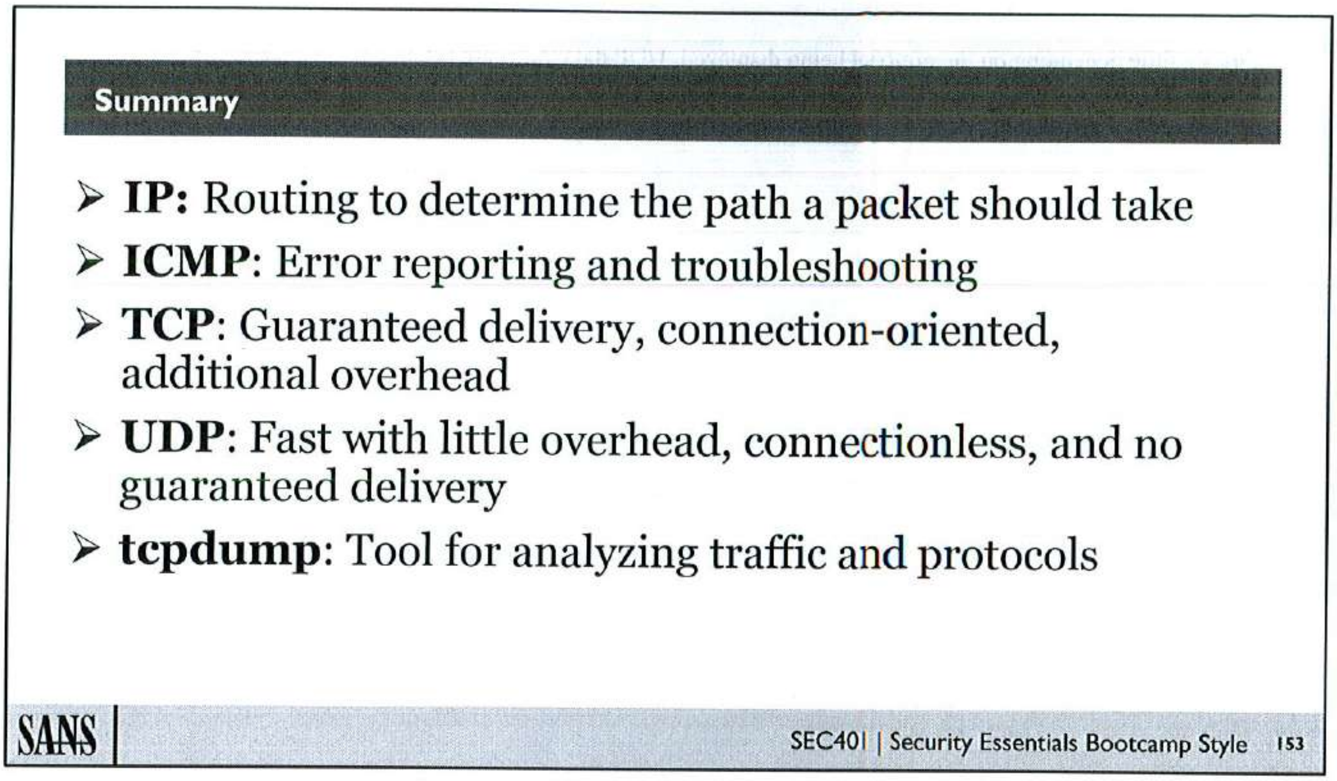
SEC401 | Security Essentials Bootcamp Style 152

By default, tcpdump collects all the packets visible to the host it runs on and produces formatted output containing information on those packets. On a busy network, tcpdump with no arguments generates output faster than you can read it, let alone analyze it. You probably would not be interested in most of the output anyway.

You can cut the output down to a more manageable size through the use of filters: Boolean expressions evaluated for each packet tcpdump sees. Filters can be specified right on the command line or in a file specified by the -F option.

If you are only looking for Telnet traffic, it would be tedious to wade through all the TCP traffic on the network and pick out just the packets destined for port 23. So, you let the filter do the work for you. The word dst port specifies that you will only see packets headed for that port, so you won't see any return traffic from the Telnet server back to the client. Note that just because a packet is heading for port 23 does not mean that it is Telnet traffic; it is simply a good guess. You would have to dump the raw packet, which you learn how to do later, and look for evidence of the Telnet protocol in the TCP payload.

tcpdump tcp	only dump TCP packets
tcpdump tcp and dst port 23	only dump TCP packets destined for port 23 (usually used for Telnet)
tcpdump host nmap.edu	only dump packets to or from nmap.edu



Summary

- **IP:** Routing to determine the path a packet should take
- **ICMP**: Error reporting and troubleshooting
- **TCP**: Guaranteed delivery, connection-oriented, additional overhead
- **UDP**: Fast with little overhead, connectionless, and no guaranteed delivery
- **tcpdump**: Tool for analyzing traffic and protocols

SANS SEC401 | Security Essentials Bootcamp Style 153

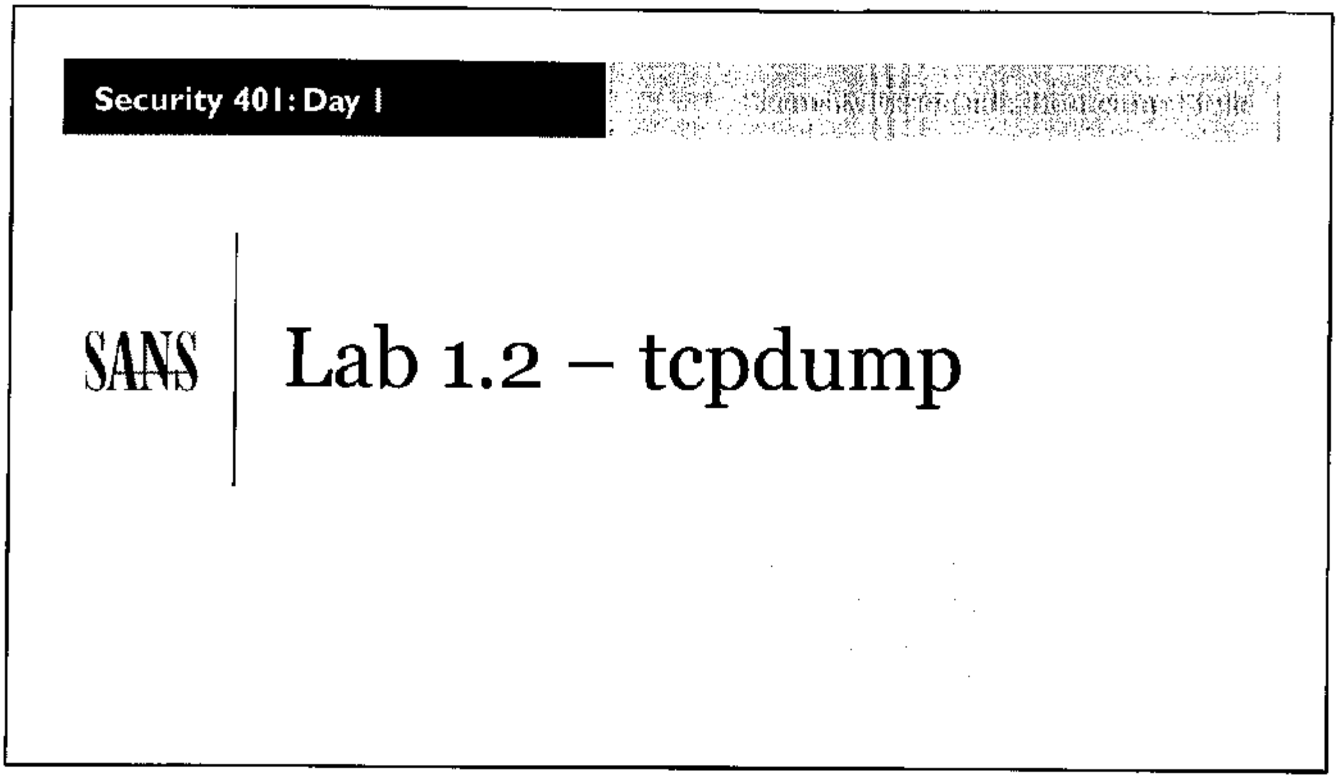
First, you learned theory about protocols and their organization into stacks. The OSI reference model is a widely used standard. Despite that, the number of people who really understand it is probably lower than it should be. If you remember the OSI protocol stack, you'll do well. The same goes for TCP/IP. Most people tend to view TCP/IP as a single, monolithic protocol; but those of us in the know realize that this isn't so. In fact, you could say that its strength comes from its layered approach to making different protocols work together smoothly.

After IP we looked at ICMP network. ICMP is a network layer protocol just like IP, but it is not typically used to send application data. Instead, IP uses ICMP to convey information about hosts and network conditions. Two of the most common uses for ICMP include the ping program, which tests to see whether a host is responding on the network, and the traceroute command, which maps the route packets take as they travel over the network. ICMP also signals certain error conditions to an IP stack. For example, if a router is unable to deliver a packet, it usually returns some sort of "Destination Unreachable" message to the sender. You can think of ICMP as the signaling protocol used to let IP stacks communicate conditions with each other.

One of the most obvious differences between UDP and TCP is the size of their packet headers. TCP, being a more detailed protocol, requires a lot more information about a given packet. UDP gets by with a mere 8 bytes of header information, while TCP requires at least 20. The two protocols have some fields in common, such as the source and destination ports, but TCP adds several other fields designed to help facilitate efficient and reliable transmission.

We also talked a lot about TCP connections: what they are, how to establish them, and how to close them down. A single TCP connection actually involves two channels: one for sending data and another for receiving it. Closing a connection requires a similar procedure or a reset. Knowledge of both of these processes undoubtedly will be useful to you when you're trying to analyze suspicious behavior on your network.

Finally, we looked at tcpdump. tcpdump is a free sniffer with a powerful filtering language that makes it easy to analyze just about any type of traffic. Although tcpdump has a fairly standard default output format, it varies a little depending on the protocol being displayed. UDP datagrams are often, but not always explicitly labeled. TCP segments are easy to spot because TCP flags, sequence numbers, acknowledgment numbers, and options are always displayed. ICMP datagrams are always labeled as such, so they're always easy to recognize.



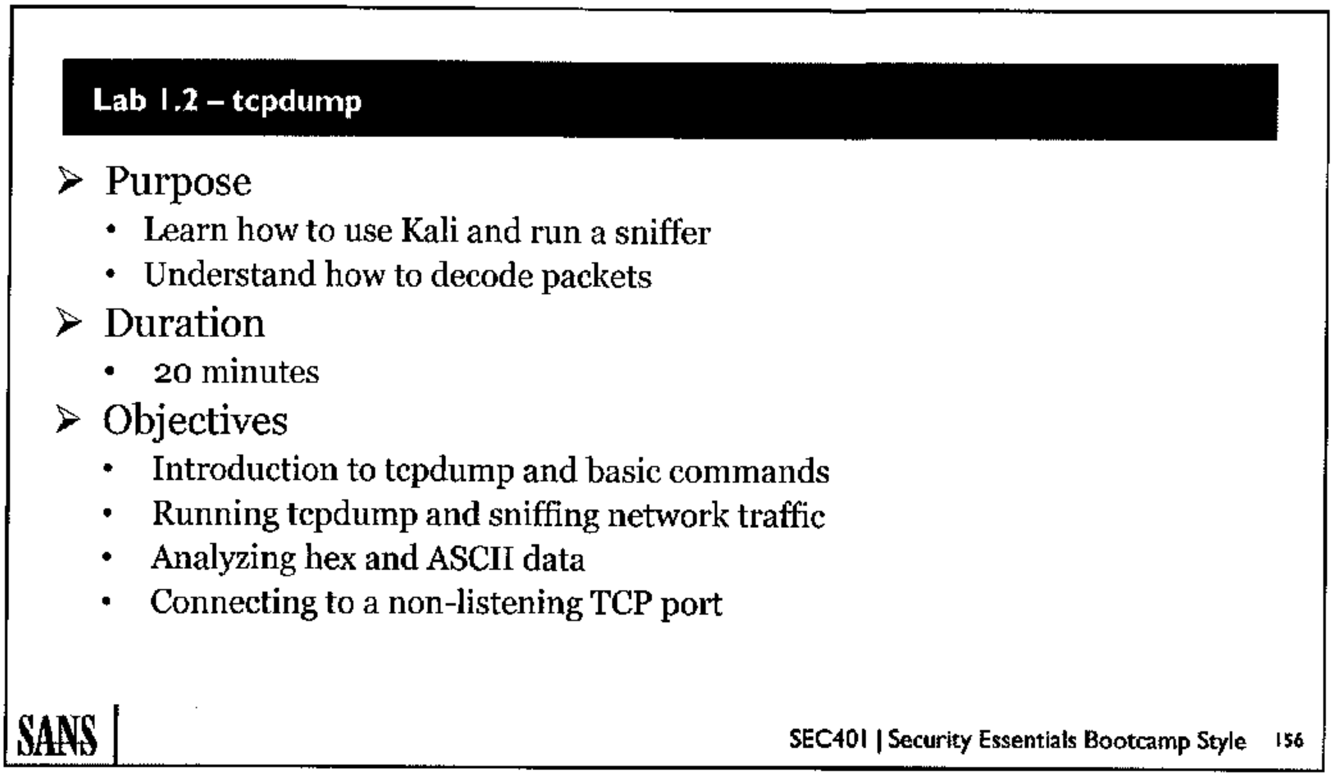
Security 401: Day 1

Security Essentials Bootcamp Style

SANS

Lab 1.2 – tcpdump

In the previous module, you learned about the tcpdump tool. tcpdump is a powerful yet simple network sniffer that displays traffic from your Ethernet adapter. Basic filtering can be applied to limit the traffic displayed or recorded to a file to only specific IP addresses, networks, TCP/UDP ports, and/or ICMP packets. tcpdump does not make any interpretation of the data and leaves that up to the analyst. The tool relies on the libpcap driver interface for the actual capturing of Ethernet frames from the wire. Detailed information about the tcpdump and libpcap project can be found at http://www.tcpdump.org/.



Lab 1.2 – tcpdump

- Purpose
  - Learn how to use Kali and run a sniffer
  - Understand how to decode packets
- Duration
  - 20 minutes
- Objectives
  - Introduction to tcpdump and basic commands
  - Running tcpdump and sniffing network traffic
  - Analyzing hex and ASCII data
  - Connecting to a non-listening TCP port

SANS

Purpose

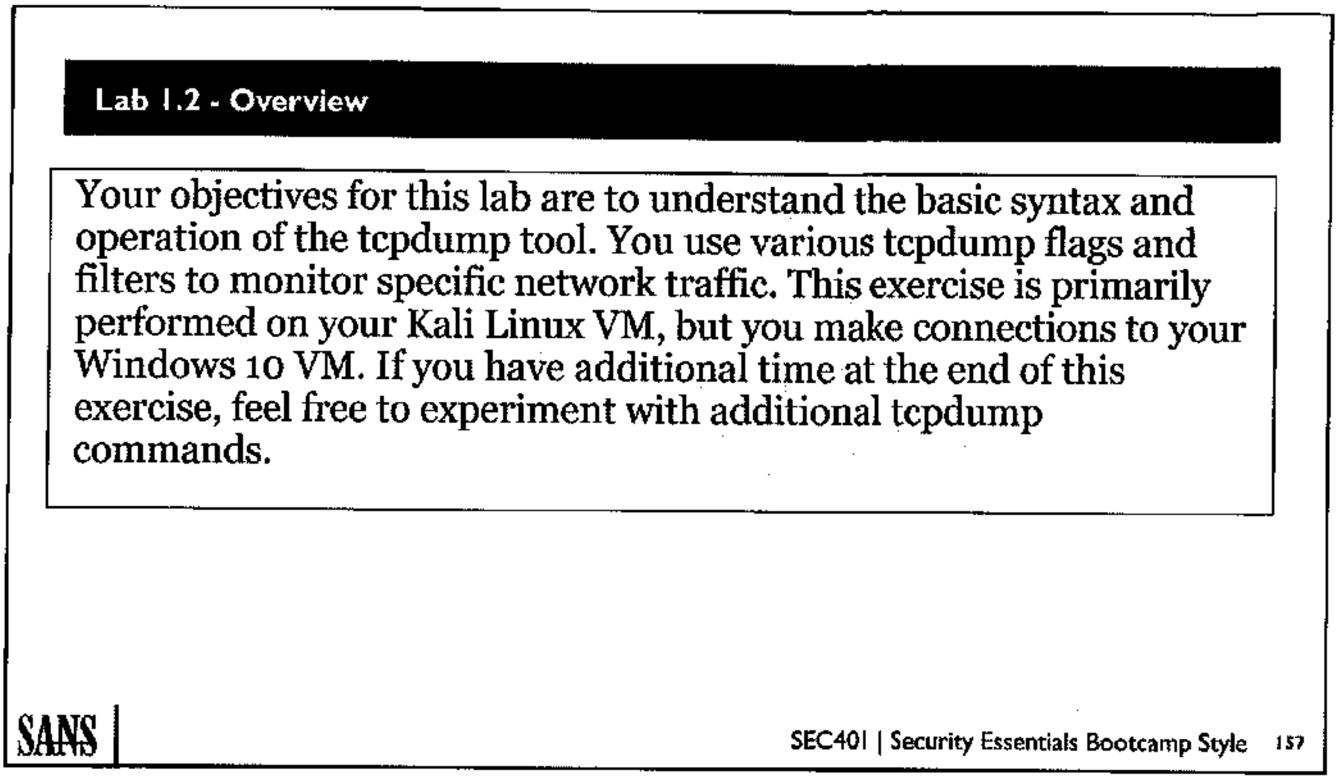
- Learn how to use Kali and run a sniffer
- Understand how to decode packets

Duration

- 20 minutes
- The estimated duration of this lab is based on the average amount of time required to make it through to the end. All labs are repeatable both inside and outside of the classroom, and it is strongly recommended that you take the time to repeat the labs both for further learning and practice toward the GIAC Security Essentials Certification (GSEC).

Objectives

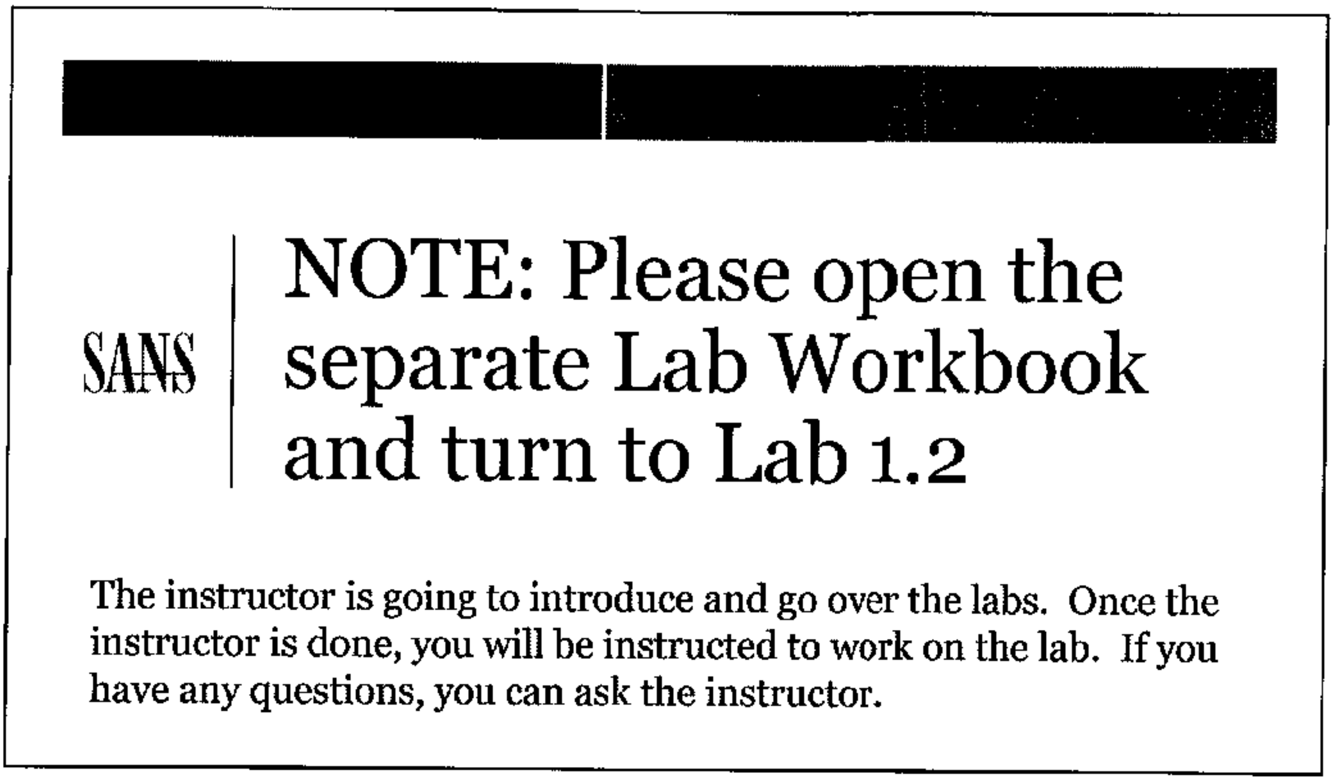
- Introduction to tcpdump and basic commands
- Running tcpdump and sniffing network traffic
- Analyzing hex and ASCII data
- Connecting to a non-listening TCP port



Lab 1.2 - Overview

Your objectives for this lab are to understand the basic syntax and operation of the tcpdump tool. You use various tcpdump flags and filters to monitor specific network traffic. This exercise is primarily performed on your Kali Linux VM, but you make connections to your Windows 10 VM. If you have additional time at the end of this exercise, feel free to experiment with additional tcpdump commands.

Your objectives for this lab are to understand the basic syntax and operation of the tcpdump tool. You use various tcpdump flags and filters to monitor specific network traffic. This exercise primarily is performed on your Kali Linux VM, but you make connections to your Windows 10 VM. If you have additional time at the end of this exercise, experiment with additional tcpdump commands.

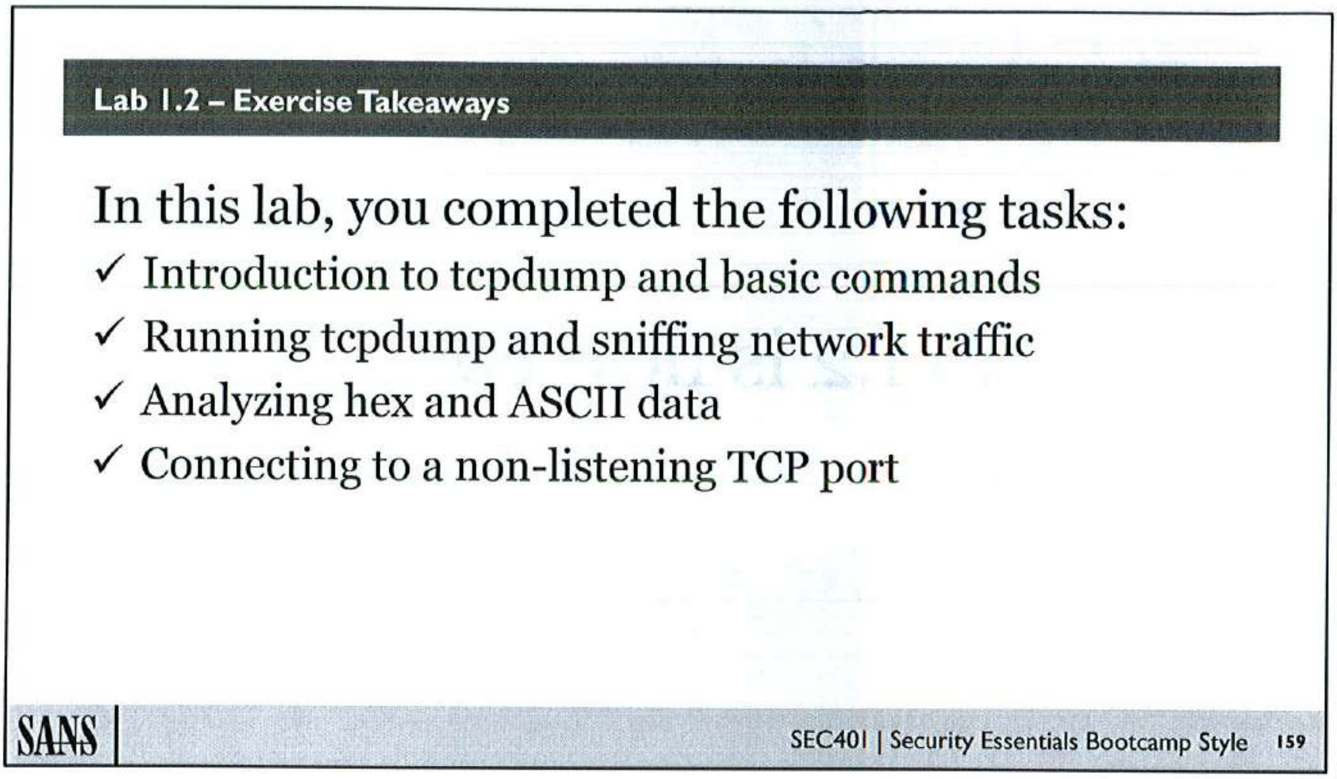


SANS

NOTE: Please open the separate Lab Workbook and turn to Lab 1.2

The instructor is going to introduce and go over the labs. Once the instructor is done, you will be instructed to work on the lab. If you have any questions, you can ask the instructor.

This page intentionally left blank.



Lab 1.2 – Exercise Takeaways

In this lab, you completed the following tasks:

- ✓ Introduction to tcpdump and basic commands
- ✓ Running tcpdump and sniffing network traffic
- ✓ Analyzing hex and ASCII data
- ✓ Connecting to a non-listening TCP port

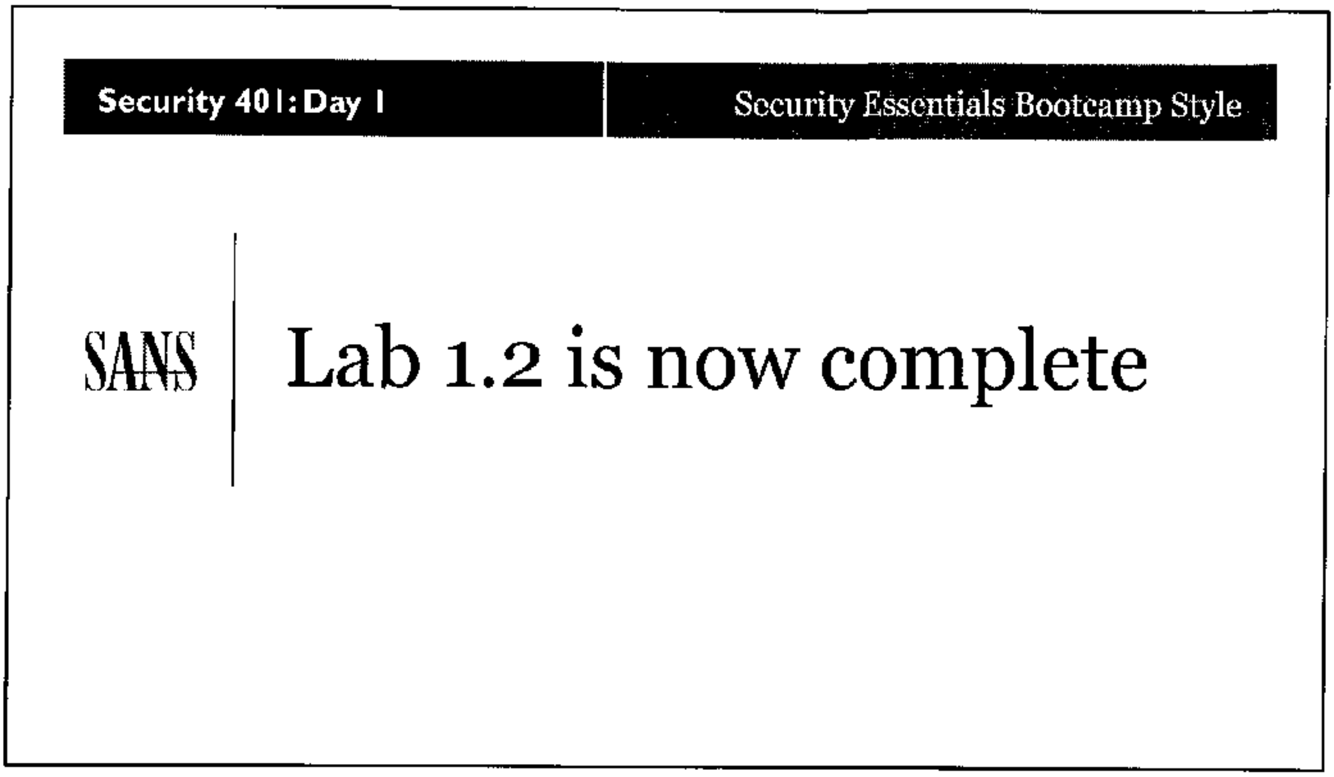
SANS

SEC401 | Security Essentials Bootcamp Style 159

In this lab, you completed the following tasks:

- ✓ Introduction to tcpdump and basic commands
- ✓ Running tcpdump and sniffing network traffic
- ✓ Analyzing hex and ASCII data
- ✓ Connecting to a non-listening TCP port

We have barely scratched the surface with the features offered by tcpdump. You can continue looking at tasks, such as analyzing UDP traffic, specifying more complex filters, dealing with encrypted IPSEC data, saving results to the file system, and more. One of the biggest advantages to learning tcpdump is that it is often installed on Linux and *NIX systems by default. As a security professional, being able to use tools built into a system without the need to install additional components is highly desirable. There is also the Windows-ported tool called windump, which can be installed on most Windows systems.



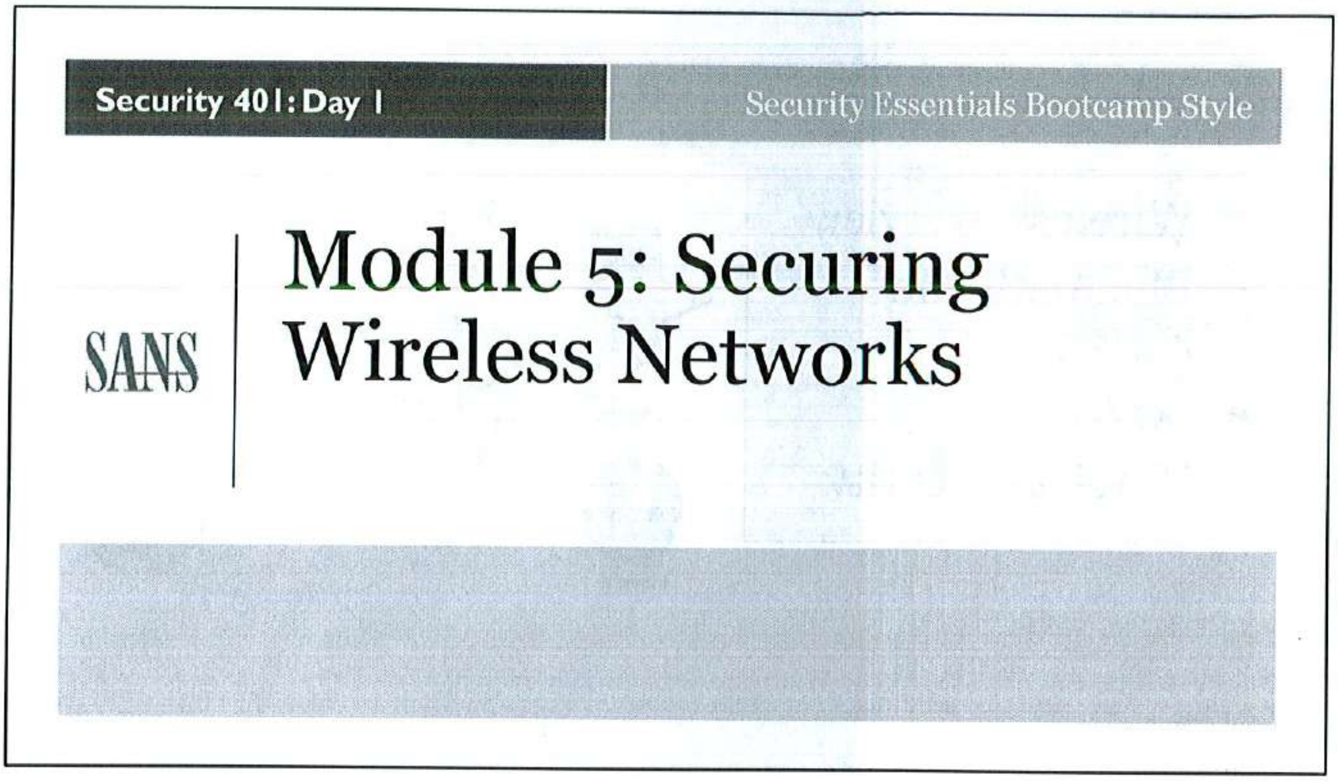
Security 401: Day 1

Security Essentials Bootcamp Style

SANS

Lab 1.2 is now complete

This page intentionally left blank.



Security 401: Day 1

Security Essentials Bootcamp Style

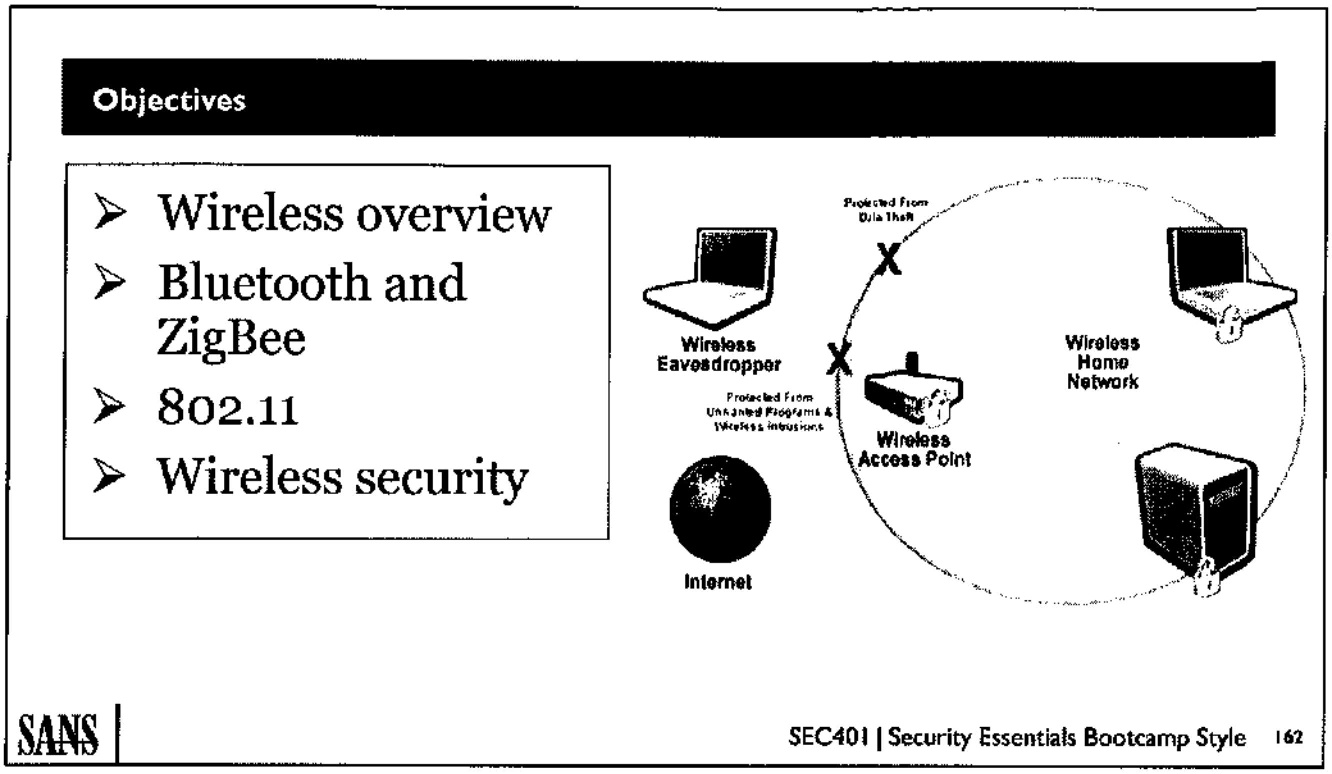
SANS

Module 5: Securing Wireless Networks

Module 5: Securing Wireless Networks

What does wireless networking mean to you? To some, it means complete and total mobility, the freedom to make calls, send e-mail, or surf the Internet from anywhere in the world without being chained to a wire. Perhaps you see it as a cost-effective way to extend your corporate network or maybe as a way to increase productivity on the factory floor. Others see wireless as an opportunity to take advantage of weak security measures for anonymous Internet access or to avoid the restrictions of your corporate firewall when attempting to exploit servers and hosts.

It is important to remember that while wireless can greatly enhance our lives and increase productivity, anything that can be used for good, can be used for evil. Adversaries also like wireless because it provides an easy way to be able to access potentially sensitivity information without having to worry about physical boundaries or wires.



Objectives

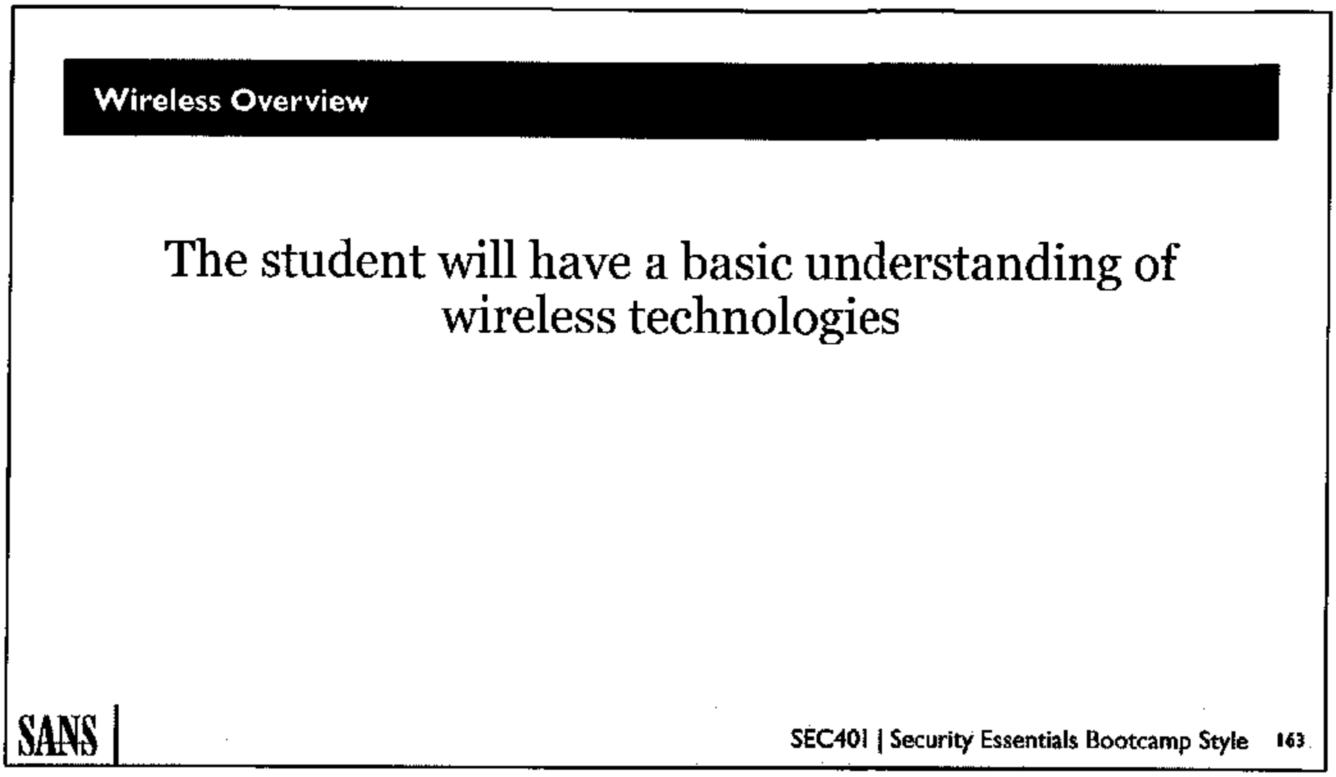
- Wireless overview
- Bluetooth and ZigBee
- 802.11
- Wireless security

Protected From Data Theft
X
Wireless Eavesdropper
X
Protected From Unwanted Programs & Wireless Intrusions
Wireless Access Point
Wireless Home Network
Internet

This module covers an often misunderstood, if not overlooked, aspect of deploying and utilizing wireless networks: security. As an individual tasked with securing your organization's resources, we present the key information elements that allow you to understand the risks and limitations of wireless networking. We discuss how wireless networks are being used in enterprises today and some of the common architectures and protocol implementations of wireless networking. We also review common misconceptions about wireless security, the top five risks of wireless networks, and look at some recommendations for planning a secure wireless LAN.

Reference

1. Vandy's 123 Blog: Wireless Security - https://vandy123.wordpress.com/

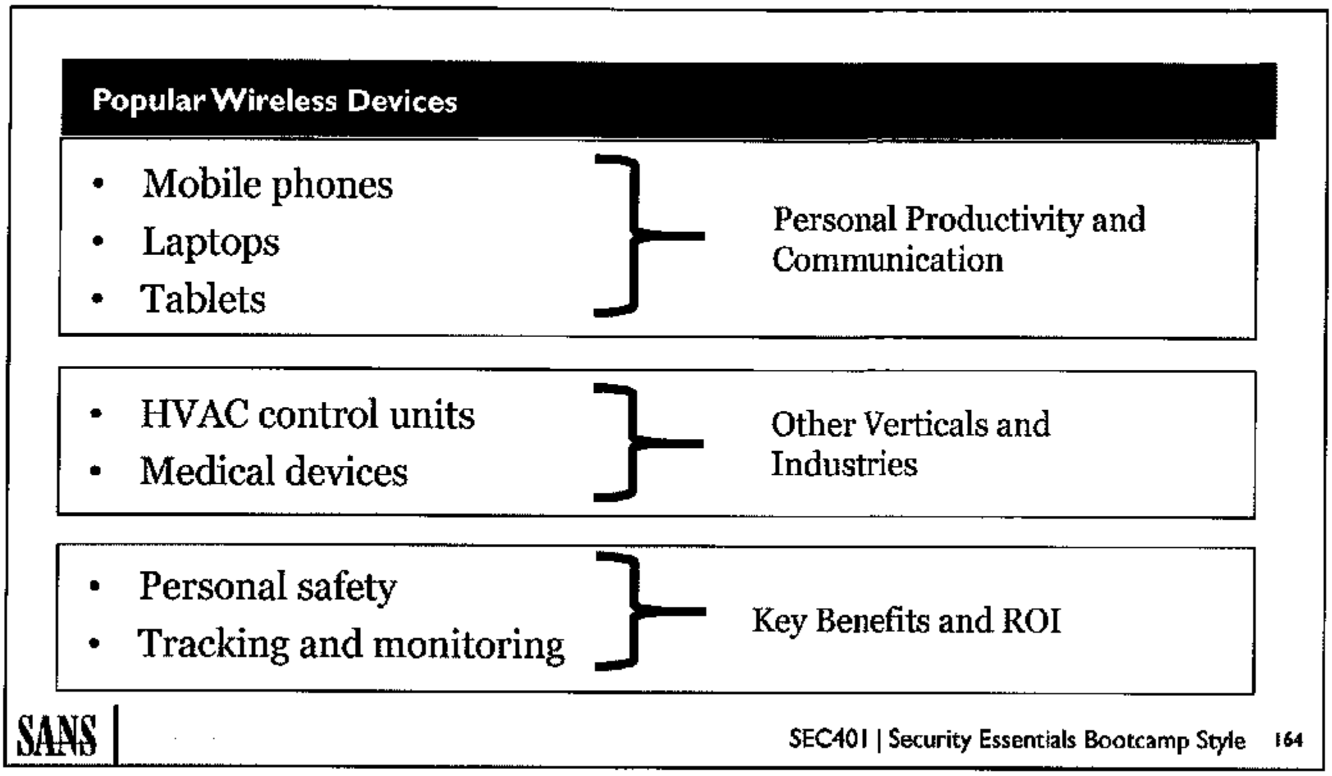


Wireless Overview

The student will have a basic understanding of wireless technologies

SANS | SEC401 | Security Essentials Bootcamp Style 163

This page intentionally left blank.



Popular Wireless Devices

- Mobile phones
- Laptops
- Tablets

Personal Productivity and Communication

- HVAC control units
- Medical devices

Other Verticals and Industries

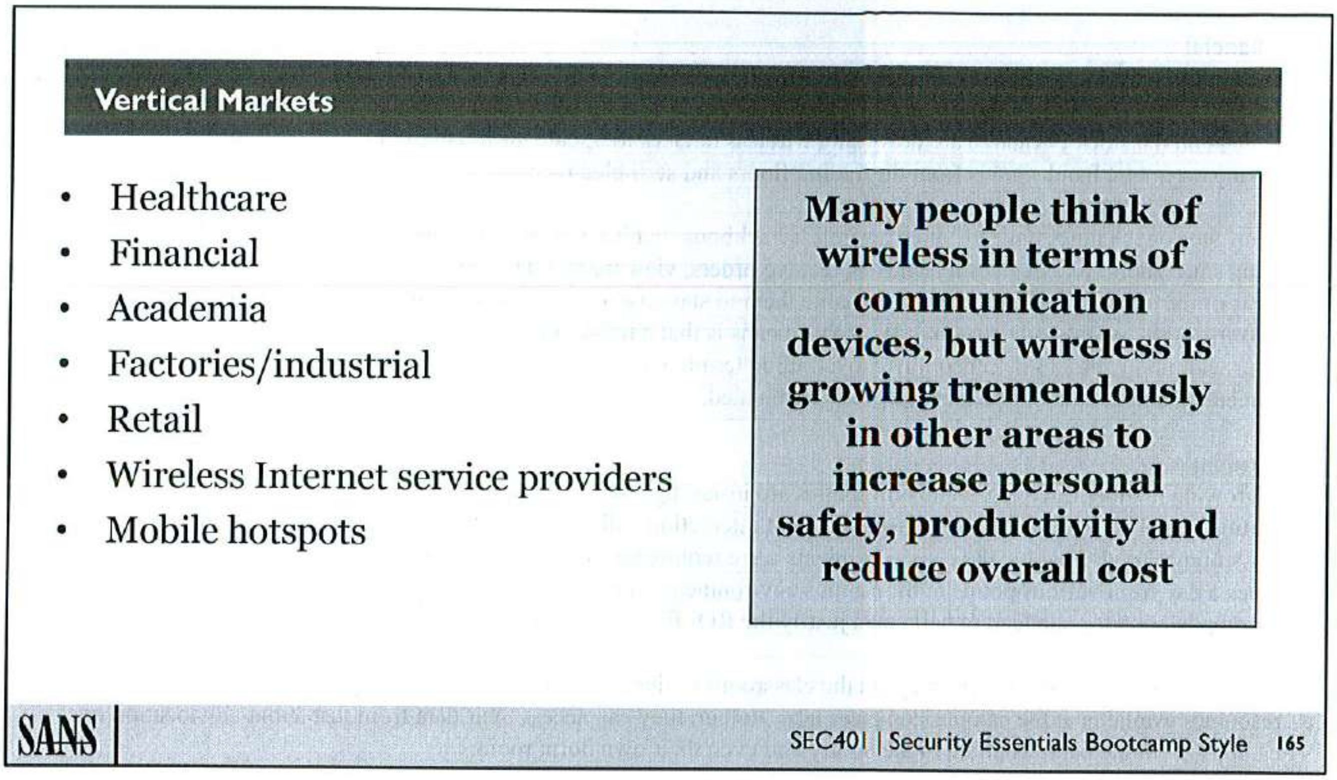
- Personal safety
- Tracking and monitoring

Key Benefits and ROI

SANS | SEC401 | Security Essentials Bootcamp Style 164

In recent years, the adoption of wireless technology has grown significantly. With the cost of wireless adapters and access points consistently falling, enterprise organizations and consumers alike have readily accepted wireless technology for all of their mobile devices, including mobile phones, handheld computers, laptops, and more.

Interestingly, wireless technology is spreading from the traditional computing environment (laptops, desktops, IP networks) into different vertical markets with technologies such as Bluetooth and ZigBee. It's not uncommon for previously unlikely devices, such as HVAC controllers, to utilize wireless technology for management functions.



Vertical Markets

- Healthcare
- Financial
- Academia
- Factories/industrial
- Retail
- Wireless Internet service providers
- Mobile hotspots

Many people think of wireless in terms of communication devices, but wireless is growing tremendously in other areas to increase personal safety, productivity and reduce overall cost

SANS

SEC401 | Security Essentials Bootcamp Style 165

As the wireless market continues to grow and evolve, we see targeted solutions for a multitude of industries. Some of the biggest vertical markets include healthcare, financial, academia, industrial, and retail.

Healthcare

Hospitals are constantly faced with the daunting task of lowering operating costs while still giving their patients superior quality and personalized care. For many hospitals, fees for services are pre-determined and are often less than the actual monetary amount needed to recoup the cost of performing the service. It does not take a brain surgeon to figure out that operating at a financial loss does not lend itself well to longevity in the marketplace. How can hospitals lower their costs, attract more patients, and get those patients through the system quickly while still maintaining a personal relationship? The answer to that question, as evidenced by increased deployment in the healthcare industry, is wireless networks.

Hospitals are quickly realizing there is a huge ROI (Return on Investment) in being able to leverage their backend databases, applications, and so on, in a mobile fashion. Doctors are using wireless networks to access patient records, submit prescriptions to the pharmacy, query databases, or even consult with other physicians. Traditionally, doctors and nurses would jot down patient information on a chart and enter it manually into the system at a later time. By enabling the nurse or doctor to enter that same information into a mobile device, the administrative burden is significantly lessened.

This process also decreases the amount of time a patient spends in the hospital. For example, a doctor might wish to discharge one of her patients at the start of her rounds. Without wireless capabilities, she might not enter the discharge information into the system until she completes her rounds two hours later; it could take another two hours to be accessed by the patient's nurse, who is away from her floor terminal. Had the doctor and nurse been equipped with wireless devices, the information would have been communicated immediately, the patient discharged, and that bed filled by another patient. Not only did that four-hour timeframe inconvenience the patient, but it also tied up a bed that could have been used for another patient. In short, more patients equal more revenue.

Financial

The ability to access real-time information is critical to the success of a financial institution. Stock markets are volatile, prices fluctuate and the success of a particular transaction is based on the speed with which it occurred. Traders on the stock exchange are now using wireless devices to update information in real-time, rather than relying on frantic hand signals from the trading floors and scribbled receipts.

Many stock exchanges have installed a wireless backbone that has helped define the future of trading. Brokers are using smartphone-like devices to send and receive orders, view market data, and send instant messages to other areas of the exchange. Wireless phones enable them to stay in constant contact with their partner firms, other brokers, or the wired trade terminal. What this means is that a transaction can now happen from anywhere on the floor without the broker being confined to a trading terminal or wired phone line. Furthermore, the transactions are immediate and the need for paper receipts is eliminated.

Academia

A growing number of colleges and universities are installing wireless networks in classrooms, libraries, and dormitories. This approach fosters greater student interaction, collaboration, and research opportunities within the classroom setting. Imagine the chaos if students were required to plug their laptops into a wired network several times a day from various points in the campus environment. The amount of cabling it would take just to accomplish this for one lecture hall could justify the ROI for installing wireless access points across the campus.

The wireless technology extends beyond the classroom setting. Students no longer need to compete for the limited resources available at the campus computer labs. Rather, they can access their data from just about any location on campus, from the campus union to the library and even their own dorm room.

Industrial/Factory

Inventory tracking, quality assurance, and material management are just a few examples of how wireless technologies have evolved in the industrial environment. Laptops can be mounted on forklifts in a distribution center to communicate to the workers that products need to be loaded onto trailers. Inventory can be tracked via wireless scanners as it moves through an assembly line. Giving engineers the ability to track changes, production issues, or faulty processes with handheld devices rather than pen and paper enhances quality assurance.

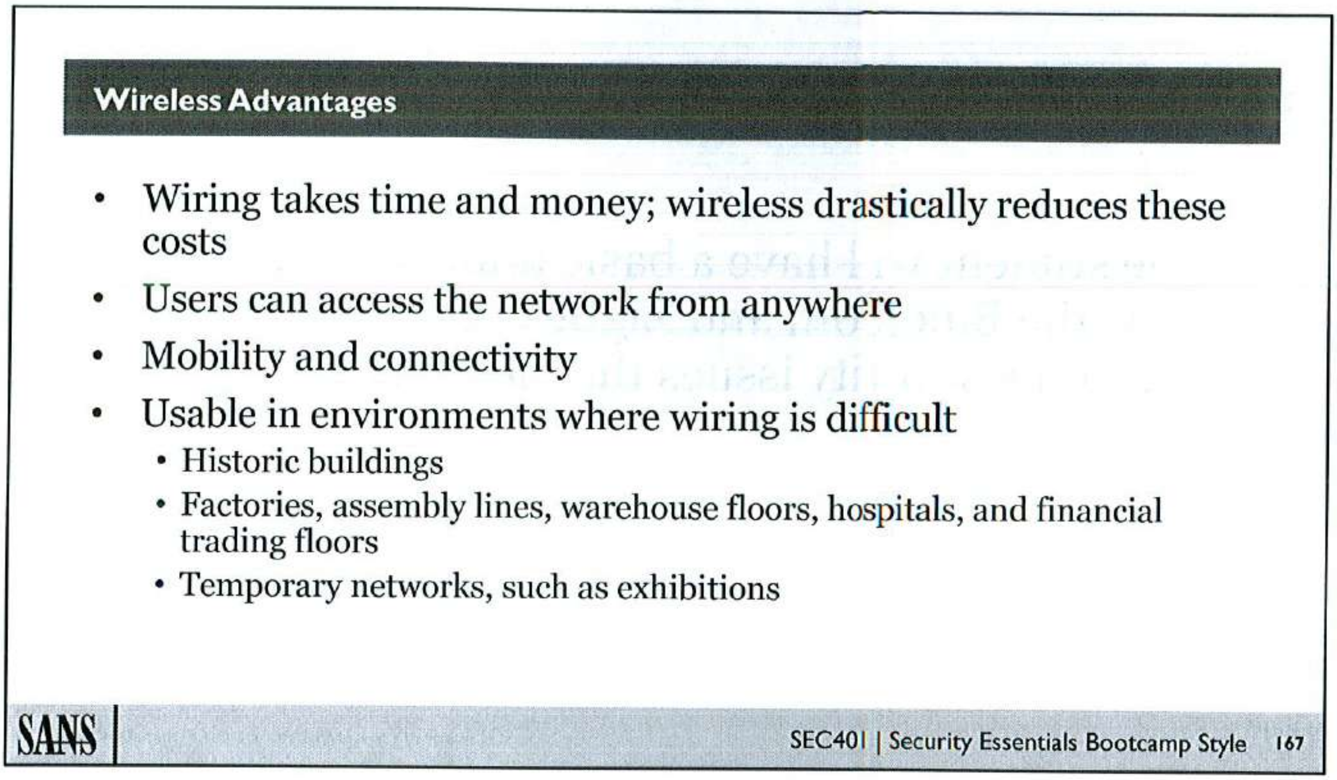
Retail and Restaurants

One of the primary applications used in the restaurant industry is for food orders being transmitted to the kitchen via a handheld device. Customer service is also enhanced by the usage of pen-based tablets by the hostess to track open tables and how many people are currently on the waiting list. The goal is to move people through the restaurant quickly while providing the best possible customer service.

Retail chains are migrating toward wireless networks to extend their point-of-sale (POS) solutions. Companies can add more cash registers and communicate the POS data to a central location without having to run cable to several locations throughout the store.

Wireless Internet Service Providers

In addition to benefiting traditional vertical markets, wireless has encouraged the growth of new vertical markets as well, including wireless Internet service providers and wireless hotspot providers. Both provide wireless Internet access to enterprises and consumers, directed at consumer or business locations, or supplying access to locations where wireless is desirable, such as airports or coffee houses. Although these particular markets have suffered from intermittent growth and downsizing, it's clear that wireless is an innovative market that will continue to accommodate growth in the future.



Wireless Advantages

- Wiring takes time and money; wireless drastically reduces these costs
- Users can access the network from anywhere
- Mobility and connectivity
- Usable in environments where wiring is difficult
  - Historic buildings
  - Factories, assembly lines, warehouse floors, hospitals, and financial trading floors
  - Temporary networks, such as exhibitions

SANS

SEC401 | Security Essentials Bootcamp Style 167

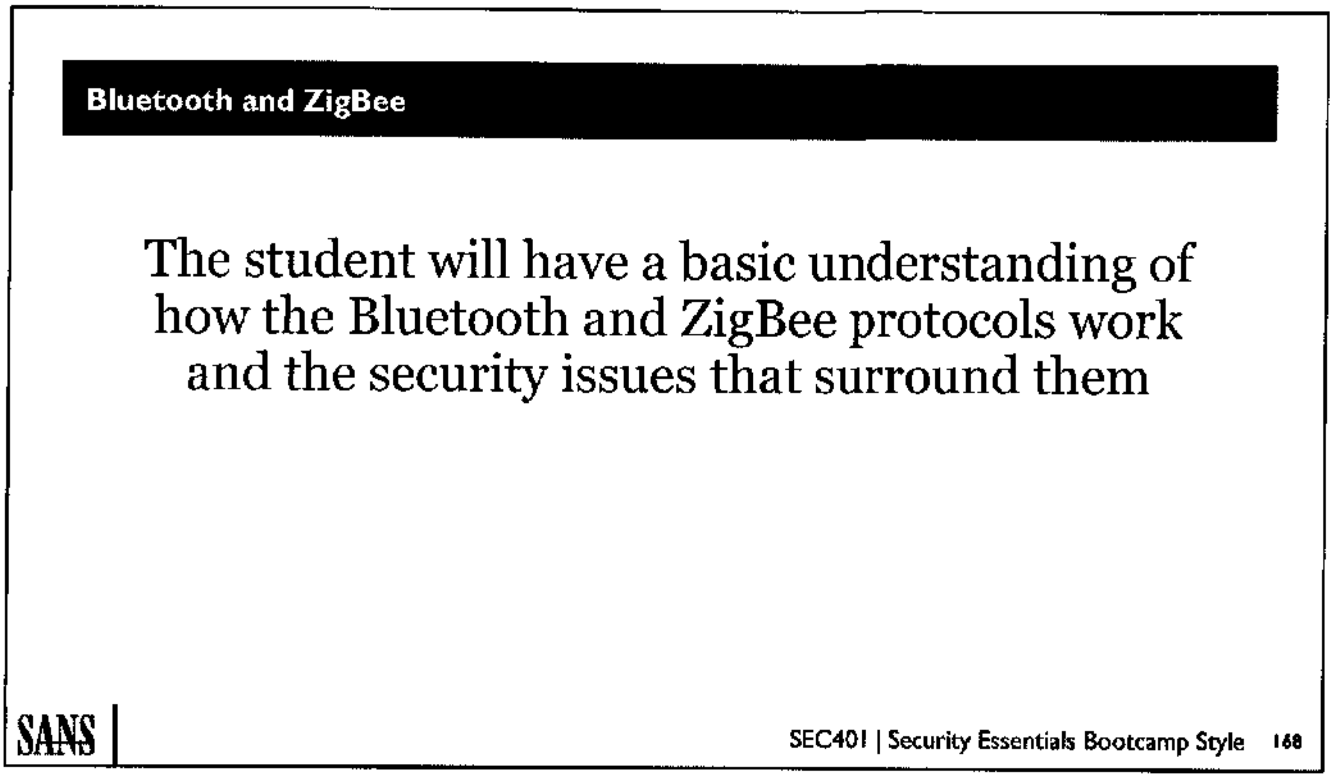
Why wireless? Perhaps the better question would be, why not wireless? As we have shown, freedom is the siren song of wireless networks. By itself, the ability to be completely mobile while staying connected is a great reason to deploy wireless technologies, but there are other compelling examples that also warrant attention.

Enabling connectivity where it simply was not possible before is an attractive reason to deploy wireless networks. Factories and warehouses are a prime example of wireless technologies enabling extended connectivity. Before the viability of mobile computing, it would have been a labor-intensive, time-consuming, and expensive endeavor to wire a factory floor for network connectivity. Another example would be to extend your network over obstacles that make wire connections difficult, such as a temporary need to electronically traverse an airport runway. For most organizations, wiring for connectivity within budget simply might not be feasible. Using wireless technology, it is now simply a matter of installing a few pieces of equipment. Networks can be extended, literally, in a matter of minutes.

Increased productivity and ease of use are obvious choices but should not be underestimated. Users can be connected and mobile, meaning they can roam the building with their laptop with little or no disruption in service. For example, if a user is scheduled to give a presentation in the west conference room. Normally, they would have to log off the network, shut down their machine, and disconnect the network cable.

It would be nice to simply pick up the laptop and move to the conference room without having to log off and shut down. It also increases productivity in the process!

Some companies are offering wireless access as a value-added service for their customers. Want to check your e-mail while sipping your mocha latte at the local coffee shop? Need to dump some stock before getting on that plane? Not a problem! Airports, coffee shops, shopping malls, and other areas where large groups of people gather are quickly realizing they can add value or service to their customers at relatively little cost by installing wireless access points to the Internet.



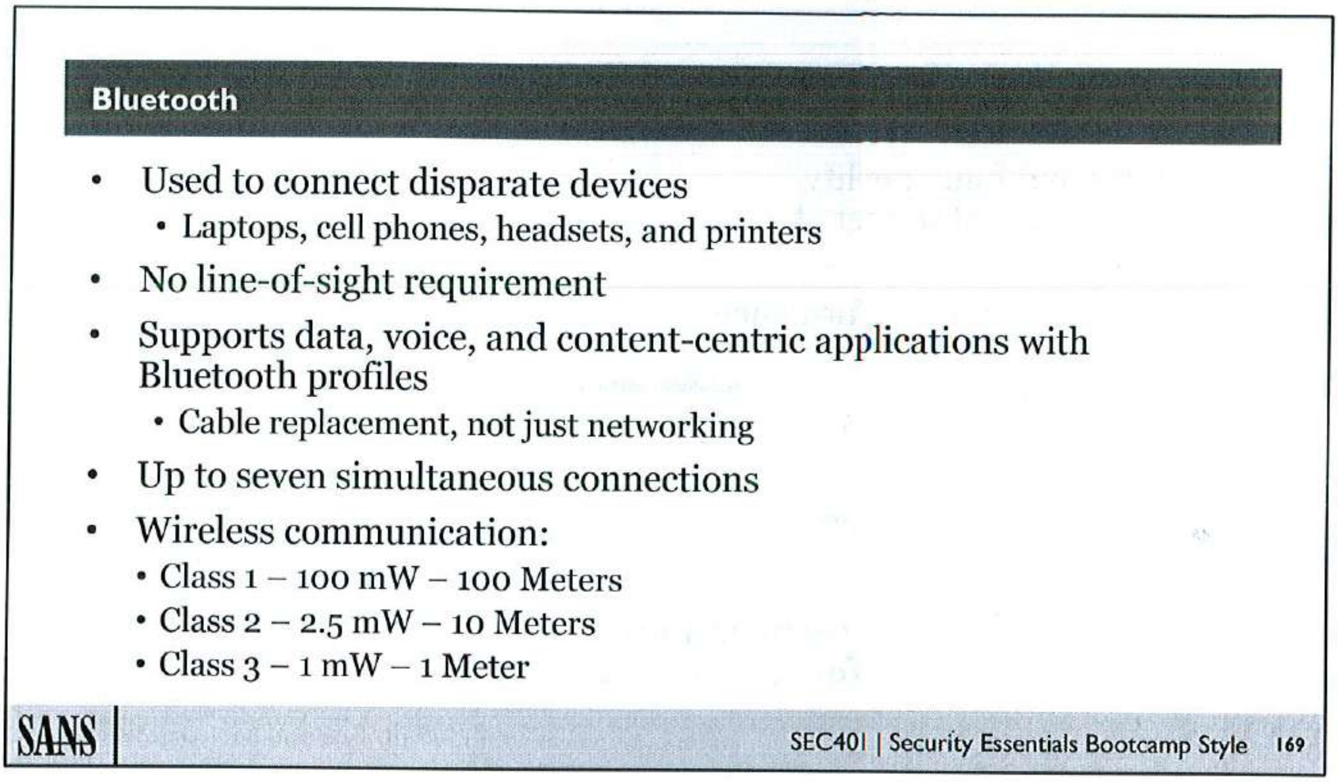
Bluetooth and ZigBee

The student will have a basic understanding of how the Bluetooth and ZigBee protocols work and the security issues that surround them

SANS

SEC401 | Security Essentials Bootcamp Style 168

This page intentionally left blank.



Bluetooth

- Used to connect disparate devices
  - Laptops, cell phones, headsets, and printers
- No line-of-sight requirement
- Supports data, voice, and content-centric applications with Bluetooth profiles
  - Cable replacement, not just networking
- Up to seven simultaneous connections
- Wireless communication:
  - Class 1 – 100 mW – 100 Meters
  - Class 2 – 2.5 mW – 10 Meters
  - Class 3 – 1 mW – 1 Meter

SANS | SEC401 | Security Essentials Bootcamp Style 169

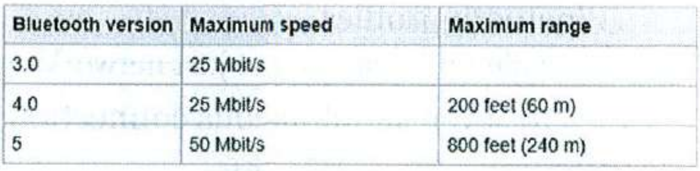
Bluetooth was first announced in 1998. It promised to be an affordable, low-power wireless solution that would be attractive as a cable-replacement technology. Using Bluetooth, vendors would no longer need to ship unique networking cables with their products, relying on an inexpensive, integrated wireless card for all connectivity needs.

Using Bluetooth, consumers can connect any type of supported wireless device, from cell phones to headsets, to talk with each other using a common technology. For example, you could use your cell phone to pull phone numbers from your laptop, or you could send documents from your laptop to a remote printer without physically being connected.

Unlike infrared networks that had limited success as a cable-replacement technology, Bluetooth has no line-of-sight requirements, making it much more flexible and user-friendly. Supporting data, voice, and content-centric applications such as streaming video or other data sources, Bluetooth networks meet a wide range of functionality requirements making it attractive to different connectivity needs. Capable of supporting up to seven simultaneous connections, a single Bluetooth adapter can communicate with several nearby devices simultaneously without being forced to connect and disconnect from different networks.

Bluetooth 5

- Features and functionality focused on IoT (Internet of Things)
- Increase in overall performance
  - Double the speed
  - Quadruple the range
  - Increased bandwidth over low ene transmission
- Supports 2 Mbit transfers
- Higher output power



Bluetooth version	Maximum speed	Maximum range
3.0	25 Mbit/s	
4.0	25 Mbit/s	200 feet (60 m)
5	50 Mbit/s	800 feet (240 m)

Bluetooth 5 was focused more on functionality and performance, not security

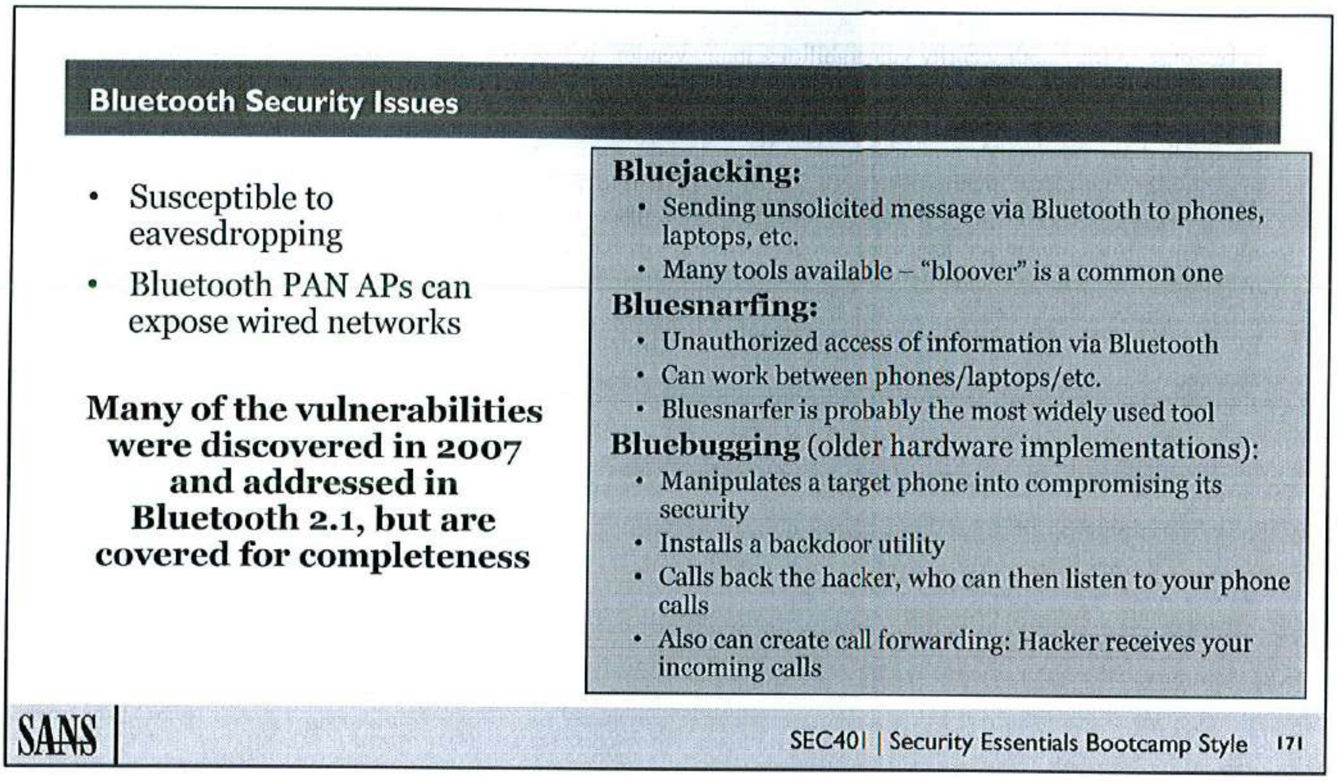
As with any standard, new technology drives new developments and enhancements. Bluetooth is no different. In June 2016, the new version of Bluetooth was released v5. An important side note is that starting with v5, it was determined to simplify the overall versioning and utilize a single number instead of a subversion such as version 4.1 and 4.2. Instead of being version 5.0, the version is simple 5.

Most of the features introduced in version 5 are focused on functionality and performance with a heavy emphasis to address Internet of Things (IoT). As every device and component that we utilized in our lives is being computer enabled, such as thermostats and ovens, the need to be able to communicate in a wireless manner is important. In order to help address this need, Bluetooth version 5 is heavily focused on addressing this problem. For example, with the latest version of Bluetooth the speed is doubled, the overall range is quadrupled and there is an increase of bandwidth to support low energy transmissions.

While security is always a concern in Bluetooth, many of the security issues were identified and fixed in version 2.1. These specific issues such as pairing will be addressed later in this module.

Reference

1. Bluetooth - https://en.wikipedia.org/wiki/Bluetooth



Bluetooth Security Issues

- Susceptible to eavesdropping
- Bluetooth PAN APs can expose wired networks

Many of the vulnerabilities were discovered in 2007 and addressed in Bluetooth 2.1, but are covered for completeness

Bluejacking:

- Sending unsolicited message via Bluetooth to phones, laptops, etc.
- Many tools available – "bloover" is a common one

Bluesnarfing:

- Unauthorized access of information via Bluetooth
- Can work between phones/laptops/etc.
- Bluesnarfer is probably the most widely used tool

Bluebugging (older hardware implementations):

- Manipulates a target phone into compromising its security
- Installs a backdoor utility
- Calls back the hacker, who can then listen to your phone calls
- Also can create call forwarding: Hacker receives your incoming calls

As mentioned in the slide, many of the security issues with Bluetooth were discovered in 2007 and addressed in Bluetooth 2.1 (available in 2009), but are covered in this section for completeness.

Like all wireless communication mechanisms, Bluetooth networks are susceptible to eavesdropping attacks where an attacker can passively or actively monitor communication between devices on the Bluetooth piconet. The use of FHSS networking makes passive monitoring on Bluetooth networks a more complex attack for an adversary, but it is only through obscurity that an attacker with the right equipment (a Bluetooth protocol analyzer) can passively capture and record all Bluetooth network activity within range.

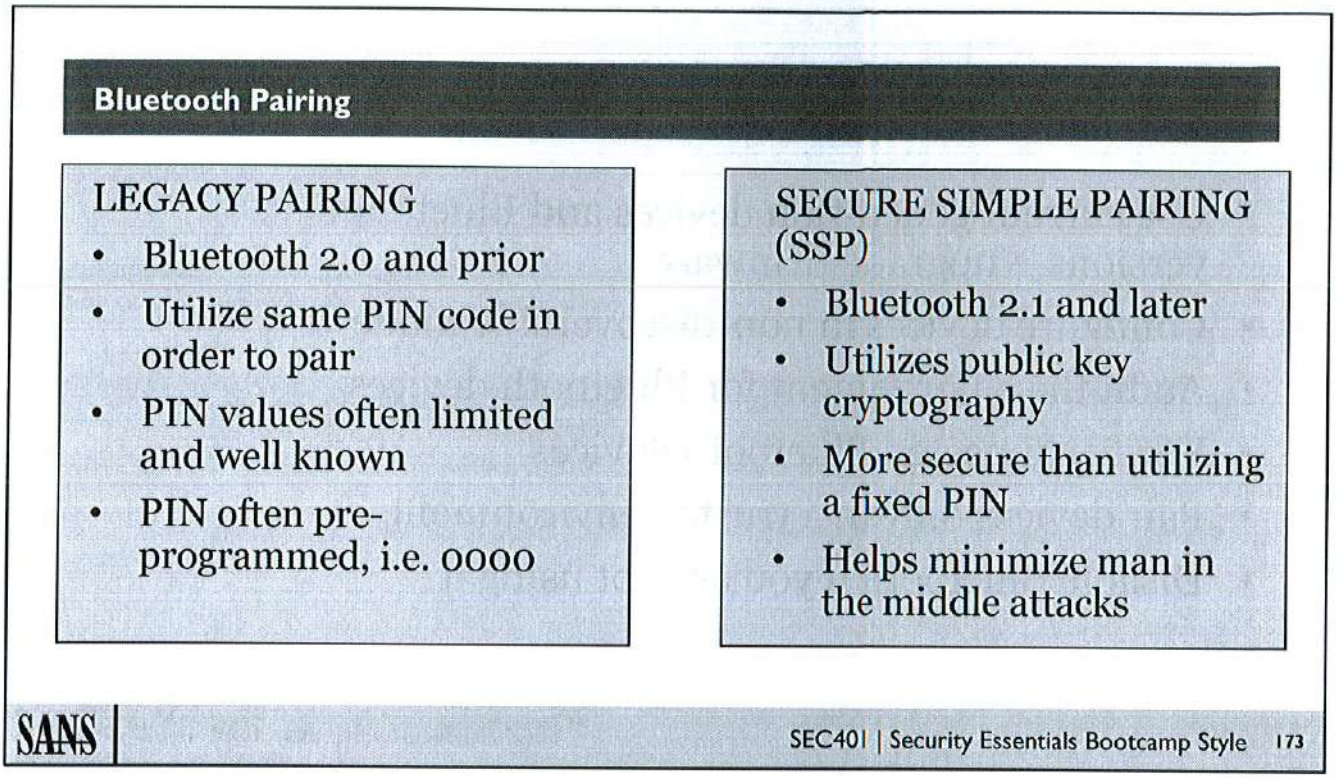
The Bluetooth algorithm used for encryption and authentication is based on the SAFER+ cipher by Cylink. SAFER+ was one of the submission candidates for the AES algorithm, but was rejected due to performance limitations with the reference implementation, and due to a weakness that reduced the effective key length when used with 256-bit keys. In the Bluetooth implementation of SAFER+, 128-bit keys are used for authentication and encryption, but are based on the relatively weak input of a PIN. If an attacker is able to capture the pairing process with a Bluetooth sniffer, it is possible to mount a brute-force attack against the PIN, recover a 4-digit PIN fairly quickly. What's more, many Bluetooth users will select weak PINs such as "0000" or "1234," further simplifying the attack.

In the past several years, security researchers are picking up an increased level of interest in Bluetooth PANs.

Tools that can be used by an attacker to locate and circumvent the security of Bluetooth networks include "RedFang" and "BlueSniff". Armed with these tools, it is possible for an attacker to locate and attack Bluetooth networks. Combined with the ability to correctly guess or attack the selection of a PIN, it's also possible for an attacker to monitor telephone conversations using a Bluetooth headset, decrypt data exchanged between a laptop and mobile device, or intercept keyboard and mouse commands from Bluetooth-enabled devices. What's more, some Bluetooth devices can be configured to extend access to the wired network over a wireless connection, similar to an 802.11 wireless network, with the Bluetooth Network Encapsulation Protocol (BNEP).

In response to Bluetooth security vulnerabilities, many vendors will recommend to customers that they configure their Bluetooth devices in "non-discoverable" mode after initially pairing with other Bluetooth devices. This will prevent a Bluetooth device from being casually discovered, but does not ultimately protect Bluetooth devices from a determined attacker. Because a device in non-discoverable mode must still respond to a PAGE request from another Bluetooth device, it is possible to scan all possible Bluetooth MAC addresses (BD_ADDR) for a given range to identify a device in non-discoverable mode. Further, if an attacker has access to a Bluetooth sniffer device, he can get access to raw RF signals and identify the presence of Bluetooth networks by examining packet traces.

Even if non-discoverable mode doesn't ultimately protect Bluetooth devices, it will often deter a casual attacker.



Bluetooth Pairing

LEGACY PAIRING

- Bluetooth 2.0 and prior
- Utilize same PIN code in order to pair
- PIN values often limited and well known
- PIN often pre-programmed, i.e. 0000

SECURE SIMPLE PAIRING (SSP)

- Bluetooth 2.1 and later
- Utilizes public key cryptography
- More secure than utilizing a fixed PIN
- Helps minimize man in the middle attacks

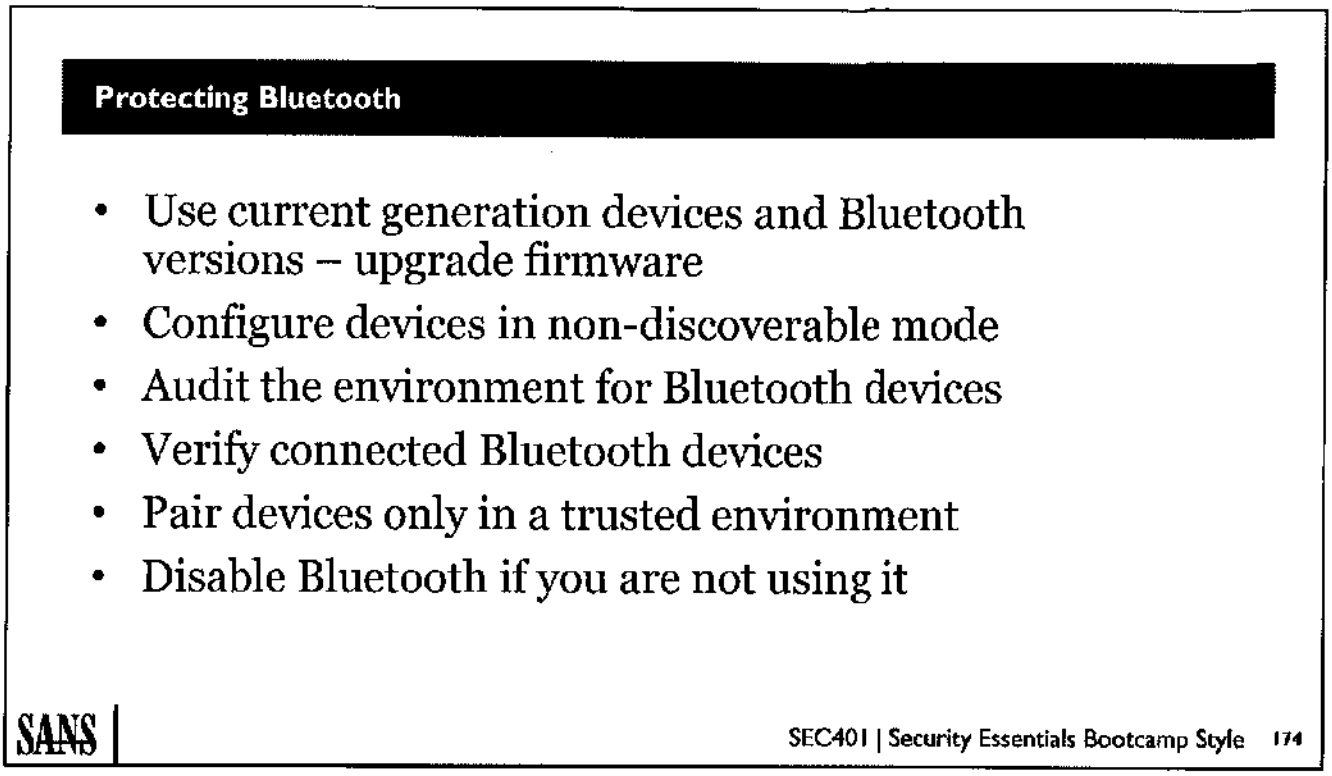
A key component of the security of Bluetooth is the pairing process. Pairing occurs when two devices decide to connect so that one device can access another device. This involves exchanging a secret and when both devices have the same secret they have created a bond which is also known as pairing. In order for 2 devices to connect via Bluetooth, there must be a bond that is created through the pairing process. Depending on the version of Bluetooth that is being used, there are different types of pairing.

Prior to Bluetooth 2.1, devices used a process known as legacy pairing. Anyone who utilized Bluetooth many years ago is probably very familiar with this process. This involves utilizing a PIN code that is used to establish a bond between the 2 devices that are pairing. The problem is the security is only as a good as the PIN and in many cases, the PIN was easy to guess and/or pre-programmed into the devices. Many of us probably remember pairing a device and using super-secure codes (read with excessive sarcasm) such as 0000, 1234, 4321, or several other simple known variants.

To overcome the issue of using a PIN that could be easily guessable, secure simple pairing (SSP) was created and as the name implies it was meant to overcome the security issues of using a simple PIN and to be simple. SSP started to be available with Bluetooth 2.1. To overcome the limitations of a PIN, SSP utilized public key cryptography in order for 2 devices to pair. While PIN's are no longer required, in order to improve security some implementations will prompt the user that pairing is occurring and have the user verify the device before the bond is created.

Reference

1. Bluetooth - https://en.wikipedia.org/wiki/Bluetooth



Protecting Bluetooth

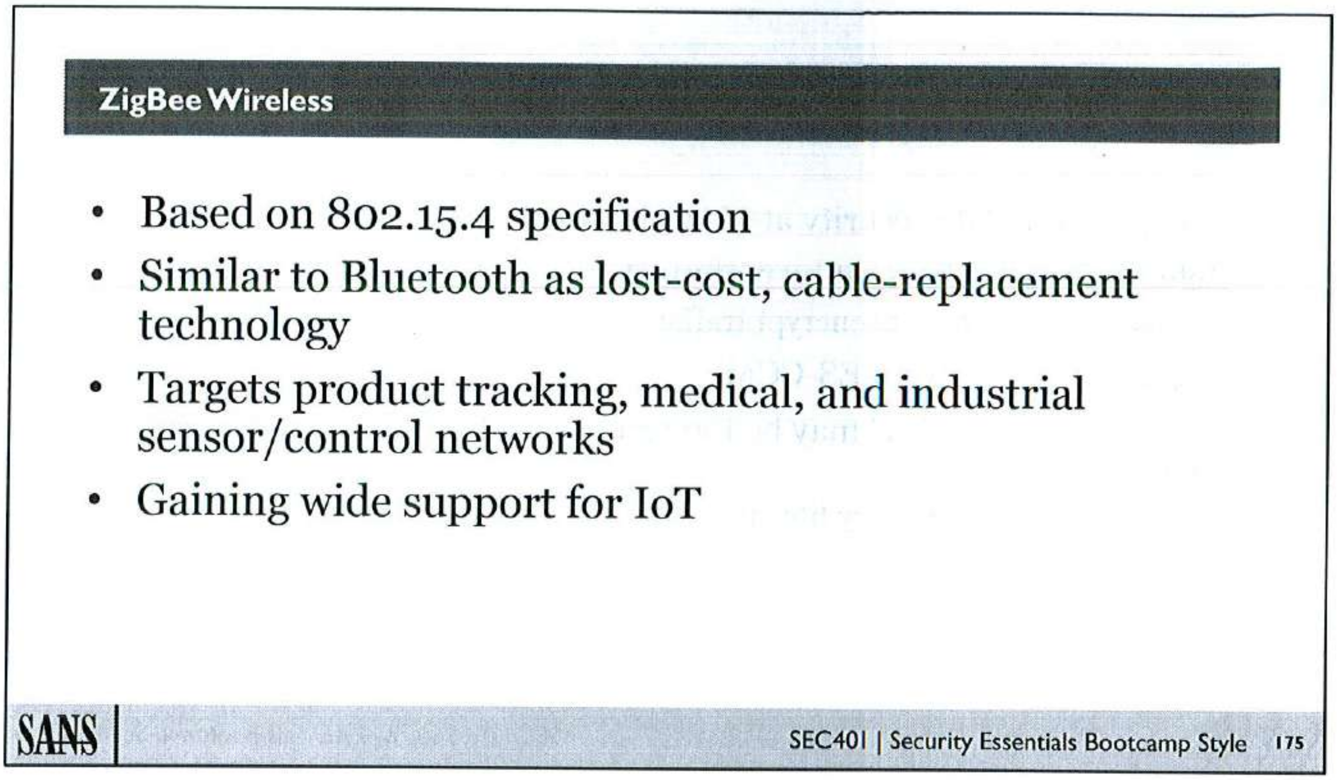
- Use current generation devices and Bluetooth versions – upgrade firmware
- Configure devices in non-discoverable mode
- Audit the environment for Bluetooth devices
- Verify connected Bluetooth devices
- Pair devices only in a trusted environment
- Disable Bluetooth if you are not using it

In order to protect the privacy and confidentiality of Bluetooth networks, it is recommended users configure their devices in non-discoverable mode after completing the initial pairing process. Ultimately, this will not defeat a determined attacker, but will likely mitigate the majority of casual attacks that exploit vulnerable Bluetooth devices.

Using scanning tools, administrators can manually identify and assess Bluetooth devices in use throughout their organizations by walking through their facilities while scanning for devices.

When initially pairing devices, it is recommended that pairing happens in a secure environment that is at least likely to be free from an attacker passively capturing network traffic. For example, if devices need to be paired to exchange data over Bluetooth, pair the two devices in a secure office environment before connecting the two devices in a public location such as a coffee house or conference venue. This limits the risk of an attacker being able to capture enough information to mount a brute-force attack.

Finally, the Bluetooth specification accommodates a stronger level of encryption support, utilizing public key cryptography that defeats many of the attacks focusing on PIN selection.



ZigBee Wireless

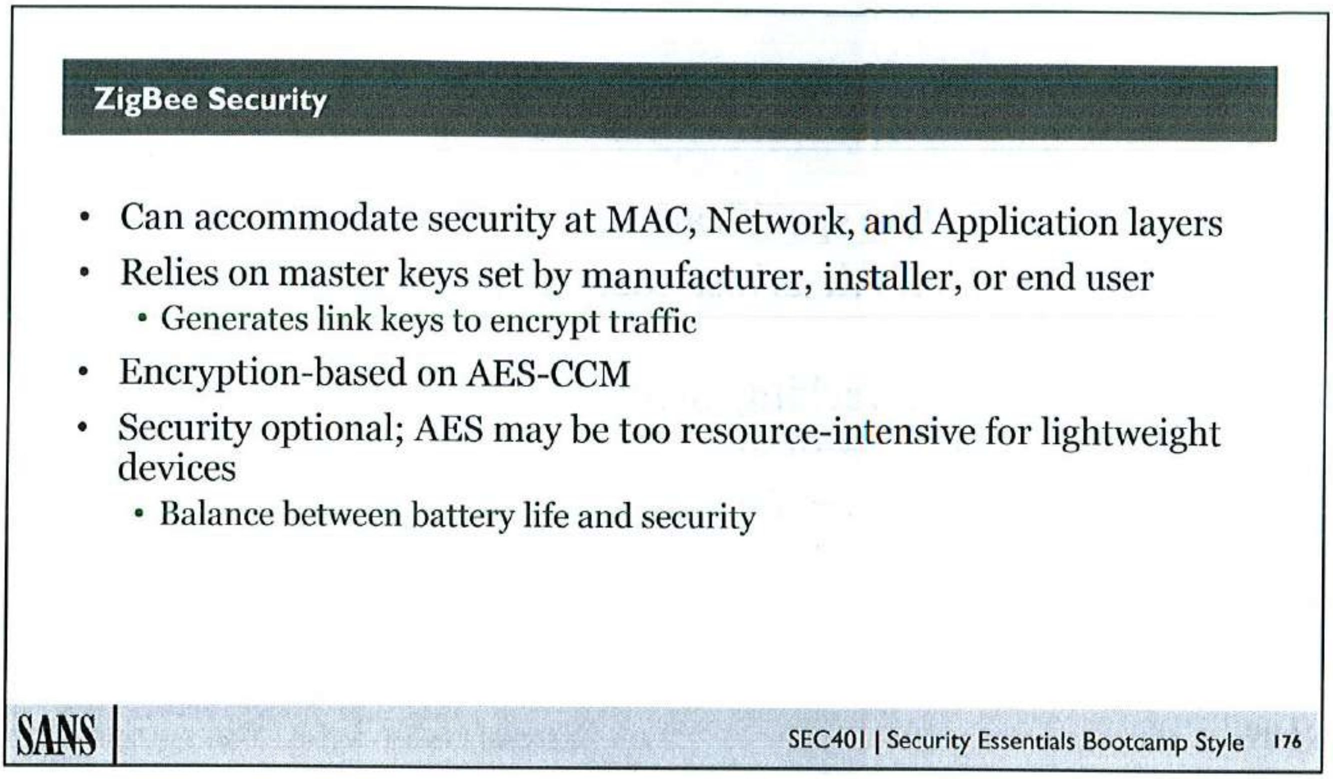
- Based on 802.15.4 specification
- Similar to Bluetooth as lost-cost, cable-replacement technology
- Targets product tracking, medical, and industrial sensor/control networks
- Gaining wide support for IoT

SANS

SEC401 | Security Essentials Bootcamp Style 175

ZigBee is another emerging wireless technology, based on the IEEE 802.15.4 specification. Similar to Bluetooth technology, ZigBee is a competing wireless specification designed at replacing cables, but targets specific vertical markets instead of general consumers. While Bluetooth is attractive as a general-purpose cable replacement technology, ZigBee is designed for use in product tracking, medical device monitoring, industrial sensor monitoring, control networks, and home automation systems. Unlike Bluetooth, ZigBee is designed to be a simple protocol implementation, requiring fewer memory and processor resources to deploy ZigBee technology.

One example of a ZigBee deployment has been publicized by Honeywell International, embedding ZigBee radios in HVAC systems. With ZigBee wireless support, an administrator can connect to the HVAC device with a ZigBee client card to manage and monitor maintenance history, utilization, and control information.



ZigBee Security

- Can accommodate security at MAC, Network, and Application layers
- Relies on master keys set by manufacturer, installer, or end user
  - Generates link keys to encrypt traffic
- Encryption-based on AES-CCM
- Security optional; AES may be too resource-intensive for lightweight devices
  - Balance between battery life and security

SANS | SEC401 | Security Essentials Bootcamp Style 176

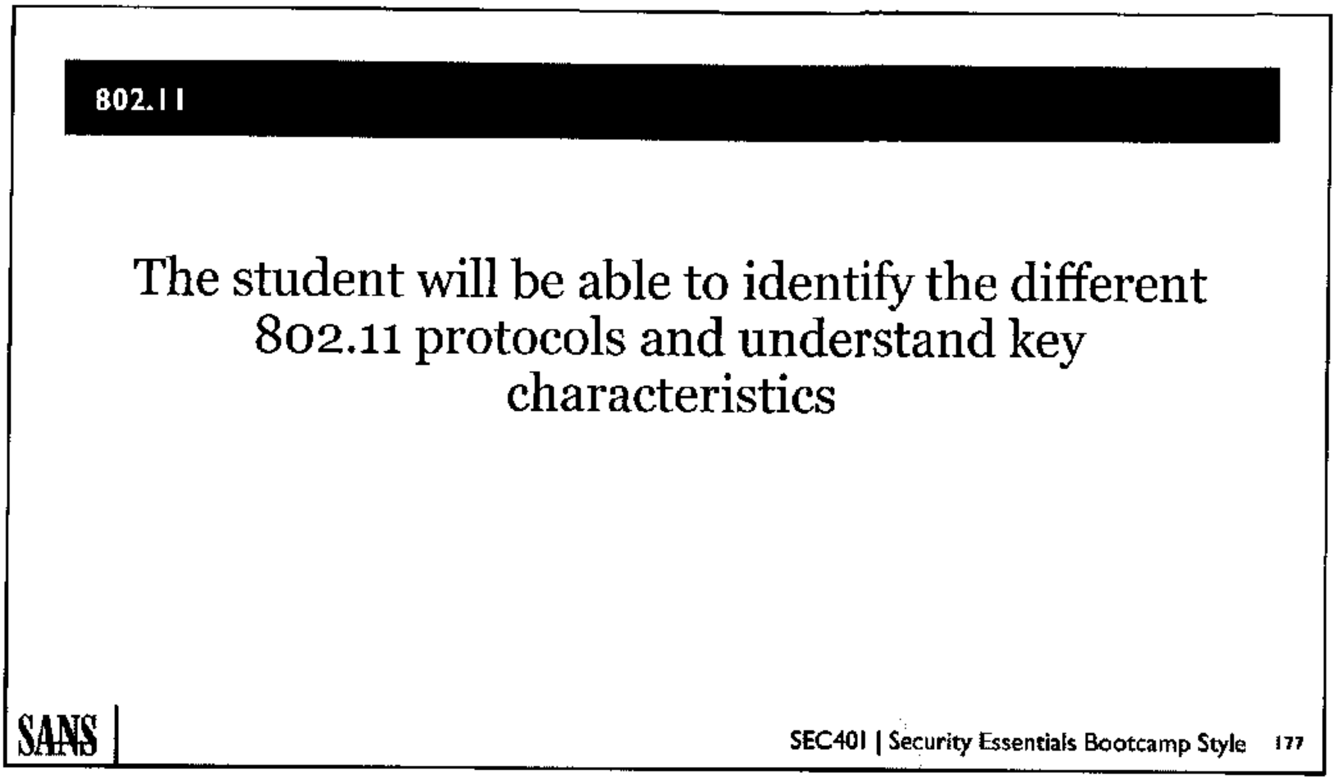
The ZigBee specification has a section dedicated to the security of ZigBee networks, accommodating security at the MAC, Network, and Application Layers. This allows ZigBee application developers to have extra flexibility in where they implement security functions, relying on security at the MAC Layer or at the upper-layer network and application functions. For simplicity, however, the ZigBee specification requires that the same key is used for all three layers of security, implying that if an attacker has compromised the MAC Layer link key, they will also be able to decrypt traffic at the Network Layer and Application Layer using the same key.

The selection of a key is done at installation time and is set by the manufacturer, installer, or end user. After specifying an initial "master" key, two ZigBee devices will establish a mutual link key that is used to authorize other ZigBee nodes and to encrypt and decrypt traffic. This is beneficial to the end user, as a compromised link key will reveal only the data between two nodes; it does not also reveal the secrecy of the master key and link keys used between other ZigBee devices.

The encryption algorithm for ZigBee networks is based on the Advanced Encryption Standard (AES) Rijndael cipher in CCM mode. CCM stands for Counter with CBC-MAC; CBC-MAC stands for Cipher-Block-Chaining, Message Authentication Code.

Rijndael is a respected cipher due to its selection as the replacement for the former Digital Encryption Standard encryption algorithm. Combined with CCM, Rijndael also provides integrity protection that prevents encrypted traffic from being modified during transmit.

While AES is a strong cipher that is suitable for many different encryption needs, it is also a resource-intensive protocol, which may be too much of a burden for lightweight ZigBee devices to accommodate. For this reason, the ZigBee specification has made the use of AES and encryption optional. This allows organizations to sell ZigBee products with the ZigBee seal of approval, without necessarily disclosing whether or not the product supports the ZigBee security mechanisms. Organizations should work closely with vendors to identify what security options are available with their ZigBee products to ensure they are taking advantage of the confidentiality and integrity protection mechanisms available.



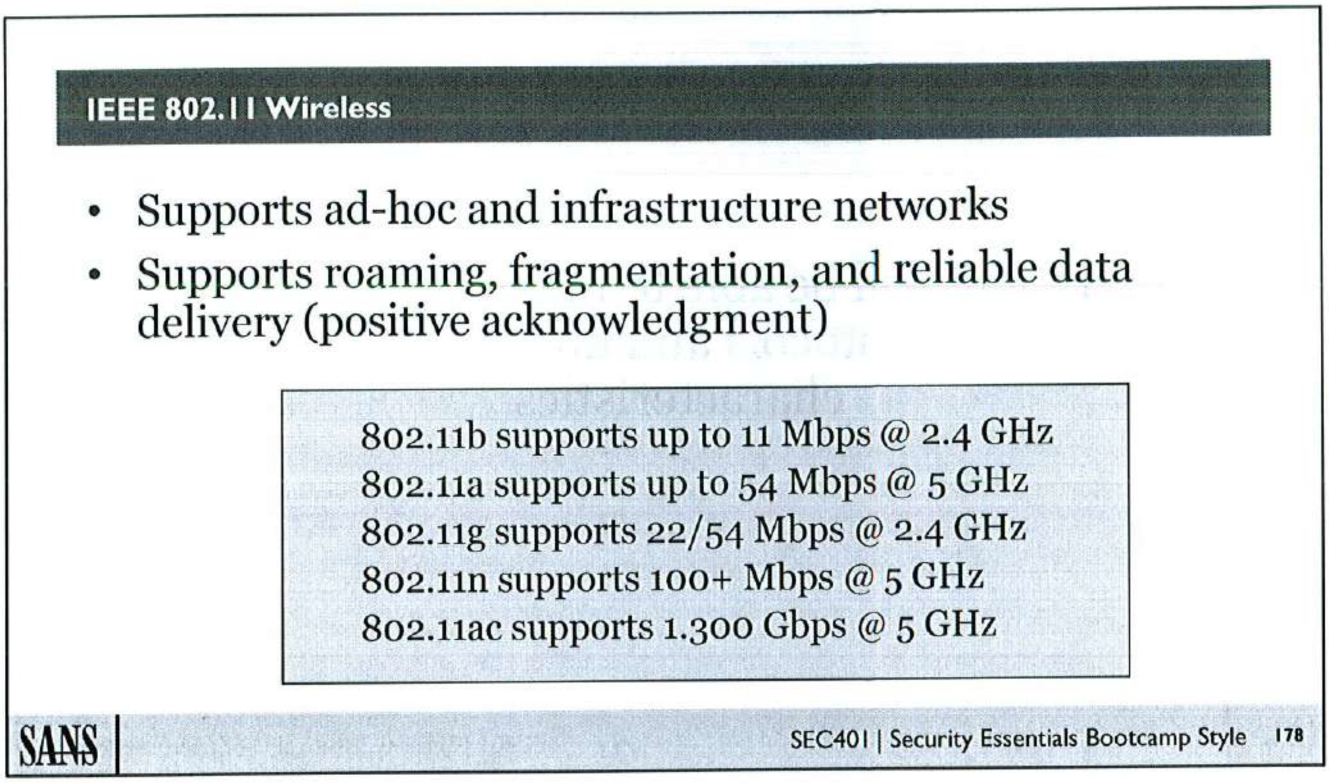
802.11

The student will be able to identify the different 802.11 protocols and understand key characteristics

SANS

SEC401 | Security Essentials Bootcamp Style 177

This page intentionally left blank.



IEEE 802.11 Wireless

- Supports ad-hoc and infrastructure networks
- Supports roaming, fragmentation, and reliable data delivery (positive acknowledgment)

802.11b supports up to 11 Mbps @ 2.4 GHz
802.11a supports up to 54 Mbps @ 5 GHz
802.11g supports 22/54 Mbps @ 2.4 GHz
802.11n supports 100+ Mbps @ 5 GHz
802.11ac supports 1.300 Gbps @ 5 GHz

SANS

SEC401 | Security Essentials Bootcamp Style 178

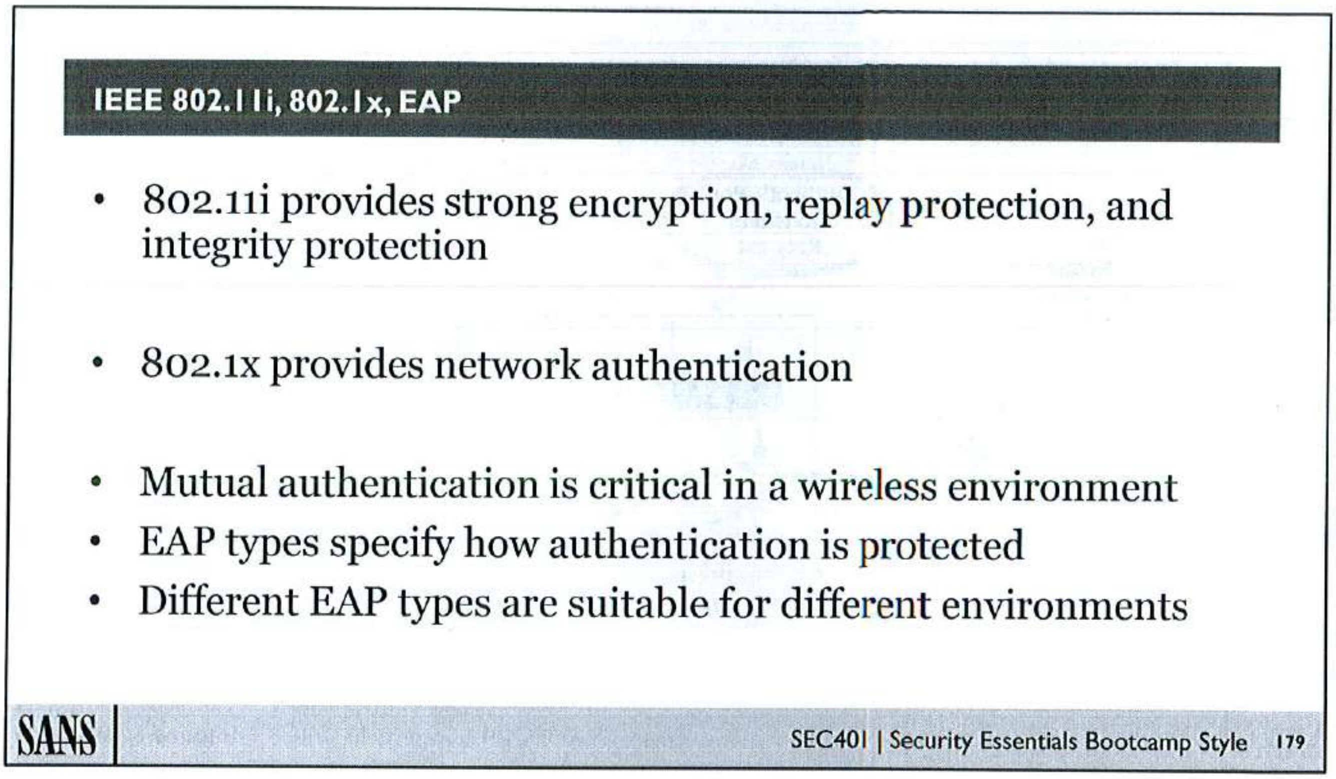
The 802.11 standard was approved in 1997 by the IEEE 802 Committee. This standard has several key elements that make it the most widely adopted wireless LAN standard in use today:

- Supports ad-hoc and infrastructure networks.
- Accommodates roaming between multiple access points without losing connectivity.
- Supports large data packets through the use of fragmentation at Layer 2.
- Provides reliable data delivery when experiencing interference due to a requirement to positively acknowledge all data traffic received from an access point or wireless station.
- Builds on existing standards for data encapsulation (802.2 LLC).
- Power conservation techniques extend the battery life of wireless devices.
- Albeit weak, the IEEE 802.11 specification included a method for encrypting data using a shared secret and the RC4 encryption algorithm. Known as wired equivalent privacy (WEP), this algorithm was the first IEEE standard to perform encryption and authentication at the data link layer.

The initial 802.11 specification branched into multiple specifications that utilize varying technology for frequency modulation that would support varying data rates:

- **IEEE 802.11b 1999**: Supports a theoretical throughput of 11 Mbps in the 2.4-GHz spectrum. Actual throughput for 802.11b networks is commonly closer to 6 Mbps.
- **IEEE 802.11a 1999**: Supports a theoretical rate of 54 Mbps. Using the 5-GHz spectrum, the 802.11a specification was not widely adopted until 2003.
- **IEEE 802.11g 2003**: Supports a theoretical rate of 56Mbps using the same frequency as 802.11b. Designed to boost the speed of networks running in the 2.4-GHz band, 802.11g suffers from many drawbacks in speed and performance when used in conjunction with 802.11b devices.
- **IEEE 802.11n (~2009)**: Supports actual throughput of 108 Mbps to 320 Mbps and drastically improves performance in both the 2.4-GHz and 5-GHz bands. The "n" standard is focused not only on increasing raw data speed, but on significantly reducing the amount of management overhead that robs the other standards of realized throughput.
- **IEEE I802.11ac (2014)**: Supports 5GHz – 1.3Gbs throughput (7Gbs possible in the future) via MU MIMO (Multi-User MIMO) and more channels.

The IEEE wireless LANs specifications are available at http://standards.ieee.org/about/get/802/802.11.html



IEEE 802.11i, 802.1x, EAP

- 802.11i provides strong encryption, replay protection, and integrity protection
- 802.1x provides network authentication
- Mutual authentication is critical in a wireless environment
- EAP types specify how authentication is protected
- Different EAP types are suitable for different environments

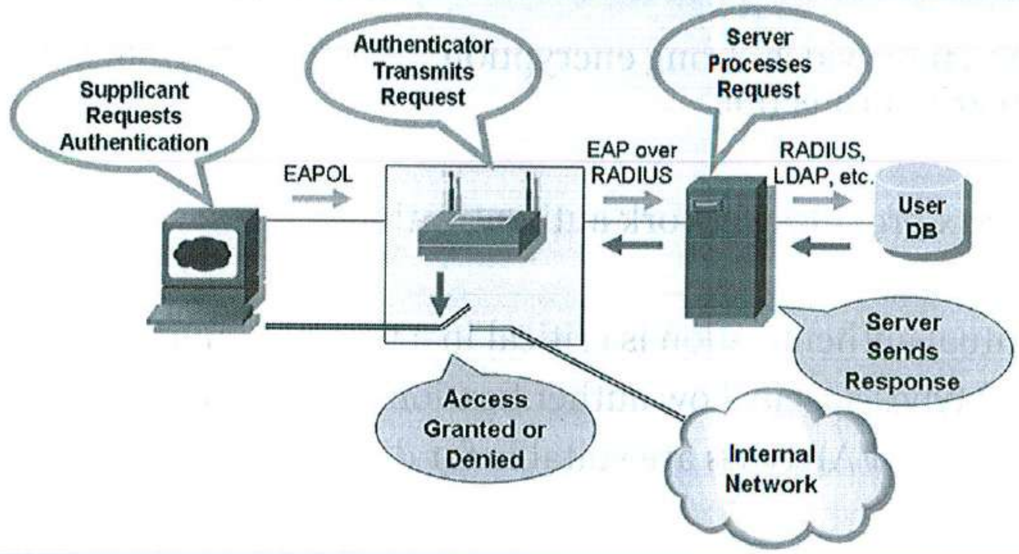
SANS | SEC401 | Security Essentials Bootcamp Style 179

To address the failures with the WEP encryption protocol, the IEEE started working on a new encryption standard for 802.11 wireless networks in 1998. Ratified on June 24, 2004, the 802.11i specification accommodated two replacement encryption mechanisms for WEP: one that could retrofit into existing hardware and a second design that would be a "completely secure" solution, requiring new hardware for implementation. Known as the Temporal Key Integrity Protocol (TKIP) and the Counter-Mode/CBC-MAC Protocol (CCMP), respectively, these algorithms represent a significantly more secure option for organizations to deploy wireless LANs. Both protocols protect information on the wireless network through strong encryption, replay protection, and integrity protection.

Although 802.11i accommodates privacy and encryption for network traffic, it does not address the issue of authentication. This was the task of the IEEE 802.1x working group, which designed a framework for network authentication. This authentication framework accommodates several different authentication protocols known as Extensible Authentication Protocol (EAP) types.

Several different options are available for organizations when selecting an EAP type. The correct EAP type for your organization is related to the client operating systems that are in use, the backend authentication systems (Windows Active Directory, LDAP, RADIUS, and so on), and the wireless access points and wireless cards in use.

802.1x Authentication

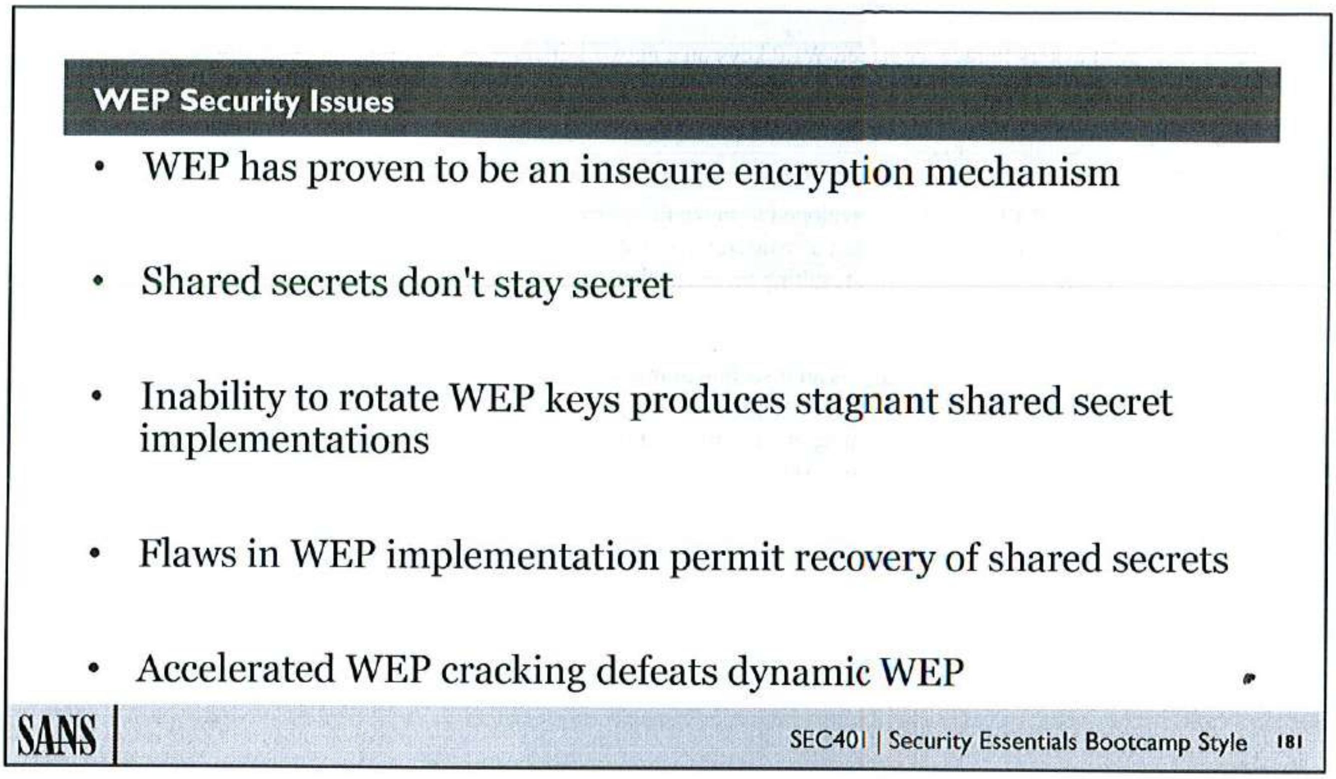


In order to deploy 802.1x network authentication, three components are necessary:

- **Supplicant**
  The supplicant is typically client software that understands the 802.1x protocol and one or more EAP types. The supplicant is responsible for forwarding authentication credentials supplied by a user or a digital certificate to an authenticating entity.
- **Authenticator**
  The authenticator is often a piece of networking hardware such as a wireless access point of a network switch that disables access to a given physical or logical port by default. The authenticator opens access on the port if the supplicant can successfully supply the necessary authentication credentials to verify their authorization to access network resources. Note that the authenticator is not responsible for authenticating the supplicant, it only passes information to the back-end authentication server and enables or disables access to the physical or logical port as directed by the authentication server.
- **Authentication Server**
  The authentication server is usually based on the Remote Dial-In User Service (RADIUS) protocol. The authentication server and the supplicant communicate through the authenticator to exchange authentication credentials before the authentication server instructs the authenticator to grant or deny access to the network.

In this illustration, the supplicant is on the left side of the diagram, attempting to access the internal network. The connection is not enabled by default since the authenticator requires the supplicant to authenticate first. In order to communicate with the authenticator, the supplicant uses a protocol known as EAP Over LAN (EAPOL) to initiate the 802.1x exchange for the specified EAP type. In turn, the authenticator passed the request along to the authentication server using a protocol known as EAP over RADIUS. The authentication server processes the request and may optionally refer to an external authentication database to verify the identity and authorization of the supplicant user.

Once the authentication server has successfully verified the identity and authorization of the supplicant, a message is returned to the authenticator to grant access to the supplicant. The authenticator forwards the response to the supplicant, and grants access to the internal network.



WEP Security Issues

- WEP has proven to be an insecure encryption mechanism
- Shared secrets don't stay secret
- Inability to rotate WEP keys produces stagnant shared secret implementations
- Flaws in WEP implementation permit recovery of shared secrets
- Accelerated WEP cracking defeats dynamic WEP

SANS | SEC401 | Security Essentials Bootcamp Style 181

Several weaknesses in attempts to secure 802.11 networks have made it difficult for administrators to keep their wireless networks safe. Attackers armed with the knowledge of common vulnerabilities in 802.11 networks are readily attacking 802.11 networks. These attacks are being launched to grant unauthorized access to wireless networks, perform DoS attacks, and collect sensitive information from private networks.

The wired equivalent privacy (WEP) algorithm was included in the IEEE 802.11 and later specifications to provide data confidentiality on wireless LANs. The 802.11-1997 specification introduced WEP as an implementation of the RC4 encryption protocol, the same encryption protocol used by the SSL and TLS protocols. WEP is based on a pre-shared secret that is common to all stations that participate in the same wireless network. In this implementation, an administrator would visit workstations that wanted to communicate on the wireless network and manually configure them with the same shared secret. The access points and clients would use the shared secrets to encrypt and decrypt data that was sent on the wireless network.

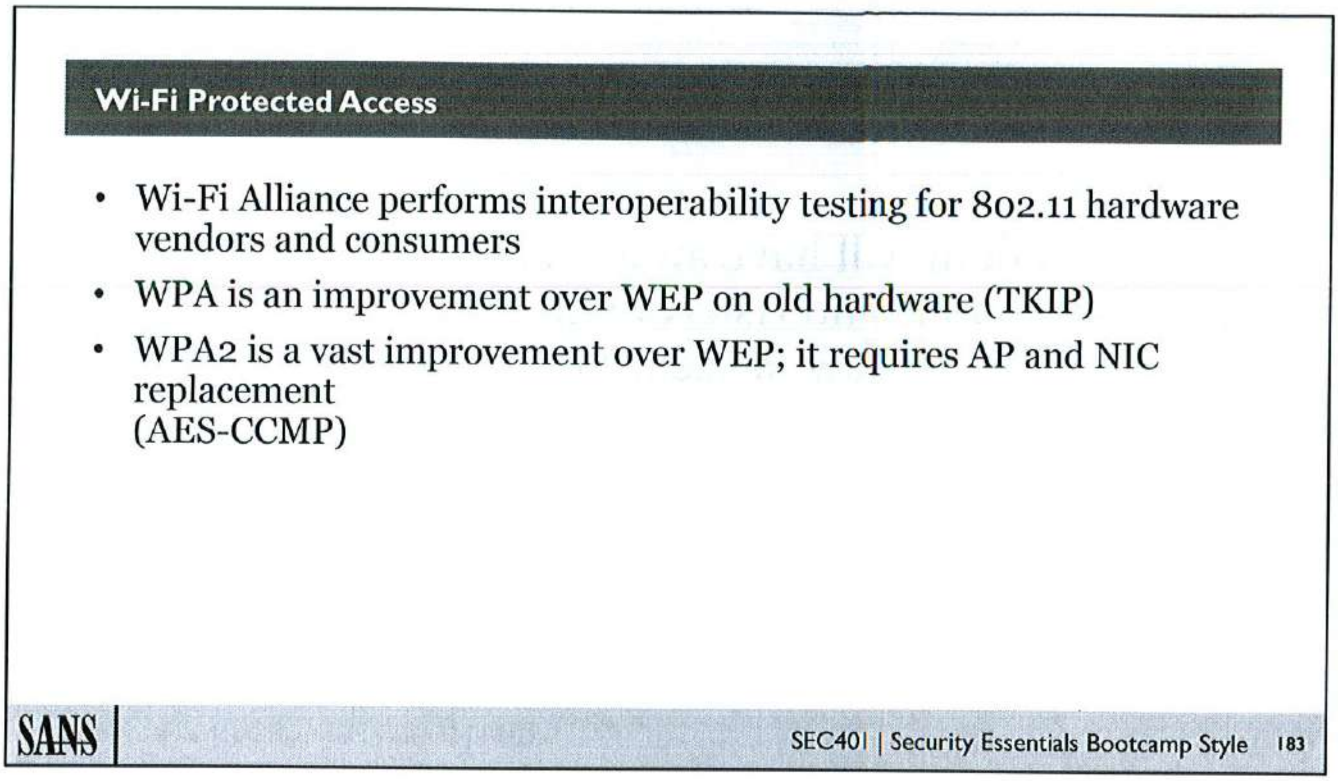
The 802.11 specification never included rotating the shared secrets on a wireless network, so many networks continued to use the same shared secret for all of their wireless clients. The inability to easily change the WEP keys on all workstations became an even more daunting problem as the utilization of wireless networks continued to grow and more people depended on the shared secret configured on their workstations.

Unfortunately, WEP proved to have another insufferable flaw in its implementation; Fluhrer, Mantin, and Shamir first reported this flaw in their paper, Weaknesses in the Key Scheduling Algorithm of RC4. Their paper identified a method by which an attacker could recover the shared secret from nothing more than the encrypted data collected from a wireless network. Shortly after the paper was published, tools were released that would recover the secret WEP key used on a network after collecting millions of packets from a wireless network. The process of recovering a WEP key in this fashion commonly took days on a network that had little or moderate traffic levels.

Tools used by attackers to recover shared WEP keys on a network include WEPCrack, AirSnort, and dwepcrack. These tools are written for Linux or BSD systems. Although effective for attacking wireless LANs utilizing the WEP algorithm, they required a dedicated attacker with patience to recover the shared secret WEP key from a wireless network.

Attack tools against WEP have been developed to make the process of recovering WEP keys and attacking wireless networks even faster. Tools such as wnet/reinj and WEPWedgie accelerate the process of collecting packets from a wireless network, often resulting in an attacker's ability to recover a shared secret from a network using WEP very quickly.

Recognizing that the WEP algorithm was an insufficient method of protecting wireless networks, the IEEE and IETF have developed alternate solutions to protect wireless LANs; however, many organizations still use WEP technology due to limitations with legacy hardware and because of the cost of replacing all hardware to accommodate stronger encryption mechanisms.



Wi-Fi Protected Access

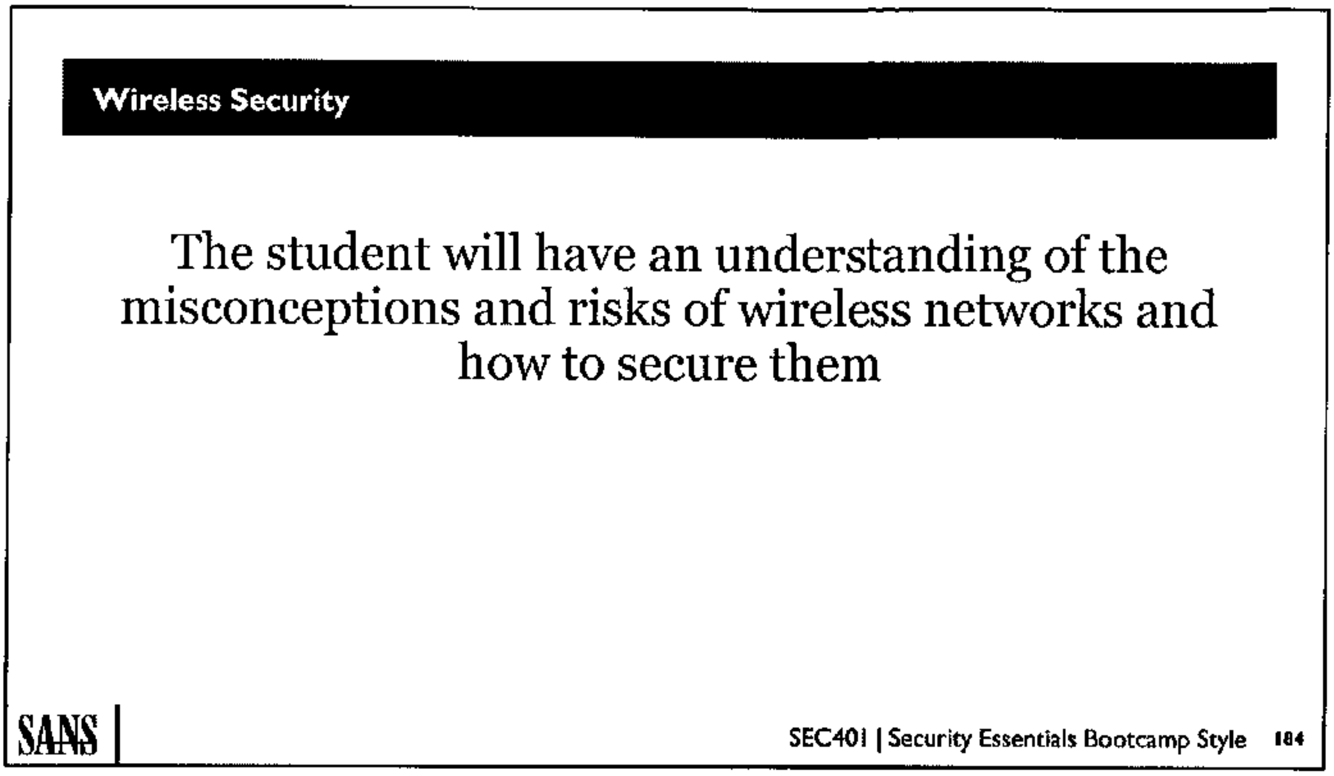
- Wi-Fi Alliance performs interoperability testing for 802.11 hardware vendors and consumers
- WPA is an improvement over WEP on old hardware (TKIP)
- WPA2 is a vast improvement over WEP; it requires AP and NIC replacement
(AES-CCMP)

SANS | SEC401 | Security Essentials Bootcamp Style 183

The Wi-Fi Protected Access (WPA) specification was adopted by the Wi-Fi Alliance before the IEEE 802.11i specification was completed to give organizations an opportunity to improve the security of wireless networks. In 2003, many organizations were becoming increasingly concerned about the security of wireless networks, without a clear solution from the IEEE to replace WEP. Although the IEEE 802.11i committee had formalized a replacement for WEP, the 802.11i specification was otherwise incomplete.

The Wi-Fi Alliance adopted the 2003 draft of the 802.11i specification and started performing interoperability testing for vendors using the Temporal Key Integrity Protocol (TKIP). This testing process certified a vendor product as WPA-compliant, focusing on the implementation of TKIP as a mechanism to replace WEP on existing hardware. After the 802.11i specification was ratified in June 2004, the Wi-Fi Alliance also adopted the AES-CCMP cipher mechanism designed for new hardware. AES-CCMP became synonymous with WPA2.

Organizations are encouraged to adopt the TKIP and AES-CCMP encryption mechanisms to improve the security of their 802.11 networks. Organizations can adopt TKIP with most existing hardware.

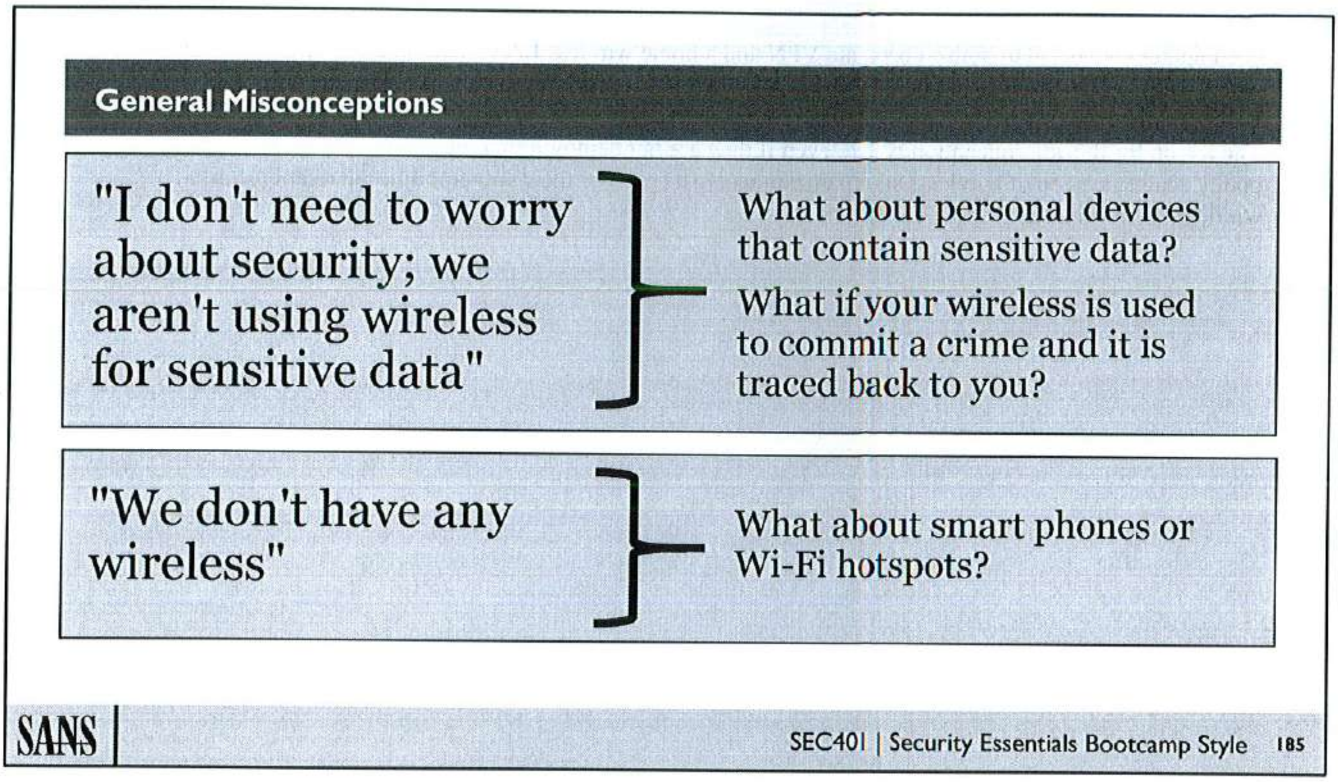


Wireless Security

The student will have an understanding of the misconceptions and risks of wireless networks and how to secure them

SEC401 | Security Essentials Bootcamp Style 184

This page intentionally left blank.



General Misconceptions

"I don't need to worry about security; we aren't using wireless for sensitive data"

- What about personal devices that contain sensitive data?
- What if your wireless is used to commit a crime and it is traced back to you?

"We don't have any wireless"

- What about smart phones or Wi-Fi hotspots?

The following are some general misconceptions that people often have about wireless.

"I don't need to worry about security; we aren't using wireless for sensitive data."
This is probably the most common misconception about wireless LAN security. Many organizations have installed wireless LANs for collaboration in small workgroups, for limited deployment to meet specific job functions, or simply to experiment with the technology. Typically deployed with their factory default configuration settings, these networks are often the most vulnerable. It is also common for an organization to simply forget about the "temporary" wireless LANs they install or to forego security measures because they believe the wireless LANs are not being used for business-critical functions.

The critical vulnerability with these wireless LAN installations is that they are completely open for an attacker to utilize for their own nefarious purposes. Attackers leverage these vulnerable networks to their advantage, using them to deliver SPAM e-mail messages, share copyright-infringed software, music, and movies, or gain anonymity when launching attacks against other organizations' networks. The lesson here is that no wireless networks should be deployed without first completely understanding the risks to the business in the event that they are exploited.

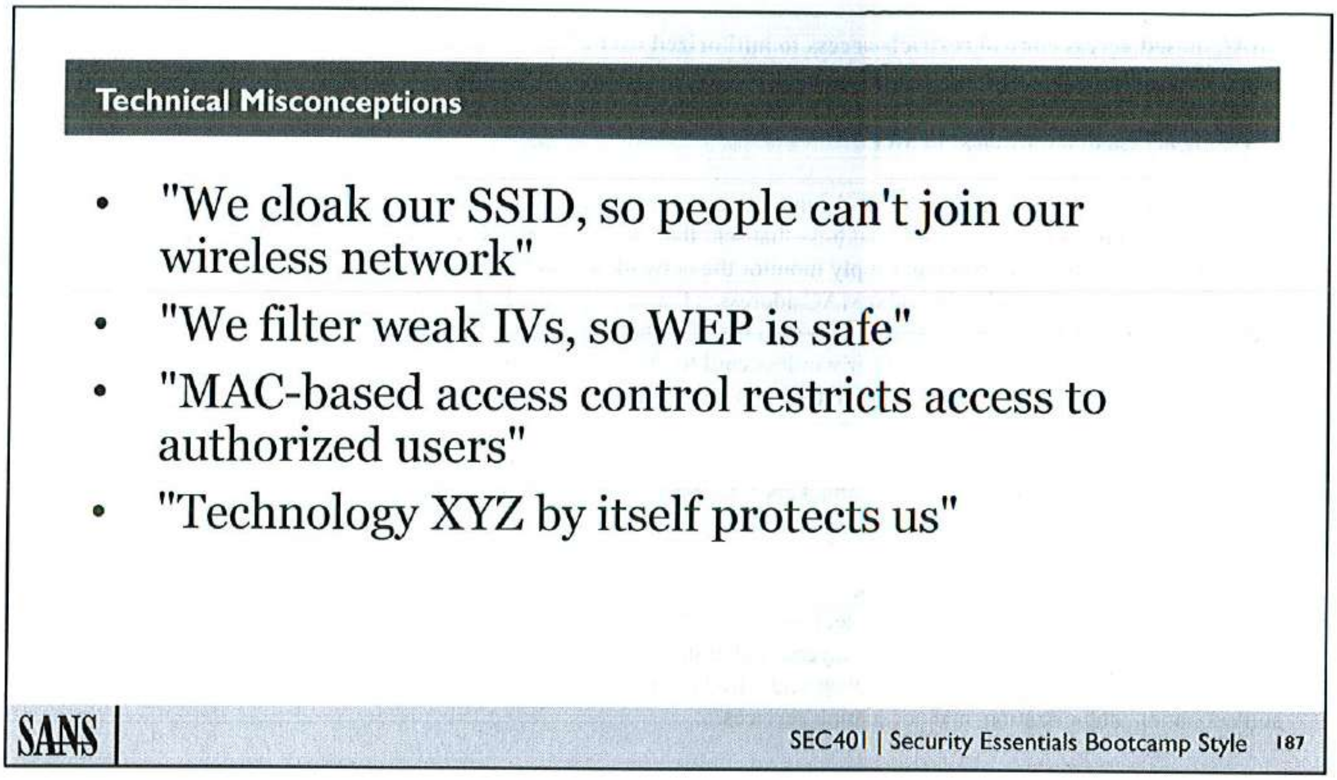
"We don't have any wireless."
Some organizations have adopted policies stating that wireless networks will not be used for their organization. Although this might be a good decision for some organizations, it is not a useful policy unless it can be enforced. Enforcing such a policy can be a time-consuming and difficult task for many organizations.

Organizations that believe they do not have any wireless networks connected to their corporate network often discover that they have unauthorized wireless access points, which are typically deployed with little or no security. Often deployed with innocent intentions, unauthorized wireless networks make their way into organizations, opening up access to the internal network and bypassing perimeter defense systems.

A final vulnerability to consider is the use of wireless LANs by users who connect to corporate networks from home over VPN. A home user is likely to have minimal security measures (if any) on their wireless network.

A computer connected to both a corporate VPN and a home wireless LAN gives an attacker the opportunity to compromise a vulnerable host and utilize the existing VPN connection to gain access to the corporate network.

The lesson for this misconception is that, even if they are not deploying wireless networks or they have a policy against the use of wireless, all organizations must consider their vulnerabilities in relationship to wireless networking.



Technical Misconceptions

- "We cloak our SSID, so people can't join our wireless network"
- "We filter weak IVs, so WEP is safe"
- "MAC-based access control restricts access to authorized users"
- "Technology XYZ by itself protects us"

The following are some technical misconceptions that people often have about wireless.

"We cloak our SSID, so people can't join our wireless network."
Although most access points offer the ability to hide the service set identifier (SSID) or network name of the wireless LAN, this feature should not be treated as a method of preventing unauthorized access. Although an attacker must have the SSID to join the wireless network, they can harvest this information even if an access point is configured not to advertise its network name. By passively listening to the network, an attacker can wait for a valid client to associate to the wireless network and capture the network name information. The lesson for this misconception is that the SSID should not be treated as if it were a password or a means of access control to the wireless network.

"We filter weak IVs, so WEP is safe."
To mitigate the vulnerabilities described in the Fluher, Mantin, and Shamir paper on weaknesses in the RC4 protocol, some vendors have made updated software available to customers to mitigate the ability of tools such as AirSnort and WEPCrack from being able to recover WEP keys. Although it effectively prevents these tools from recovering the shared secret used for WEP, weak IV filtering does not help administrators manage the management problems associated with using WEP to protect wireless networks. Specifically, administrators must still manage the distribution of shared secrets on all computers that participate in the wireless network without being able to easily update and rotate the WEP keys.

Weak IV filtering also does not prevent other attacks against WEP, including an attacker's ability to inject arbitrary packets into a WEP network without knowing the WEP keys. With filtering used to accelerate the WEP-cracking process, an attacker can utilize this weakness to port-scan, discover, and exploit hosts behind a wireless network—all without knowing the WEP key in use.

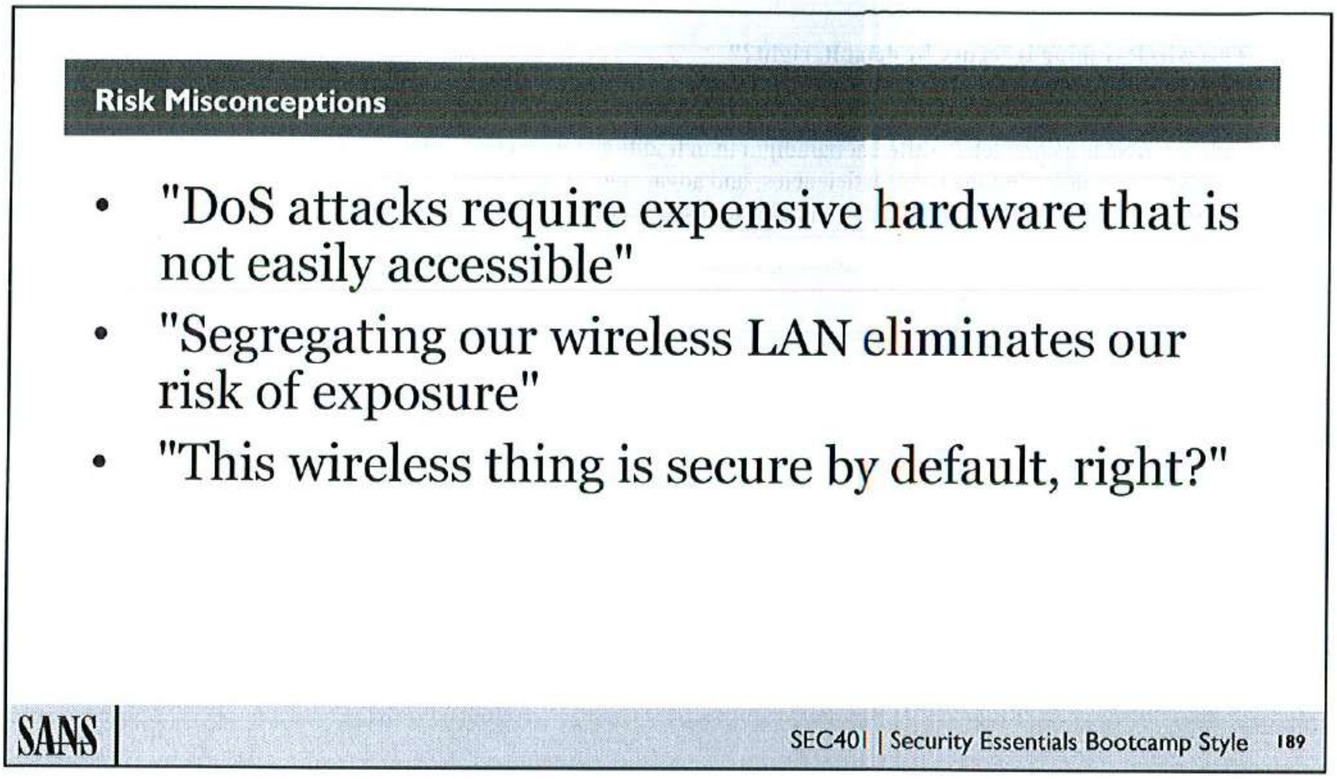
The lesson for this misconception is that, despite efforts to improve weaknesses in the protocol, the WEP protocol cannot be used as a mechanism to secure wireless networks.

"MAC-based access control restricts access to authorized users."
Nearly all access points offer the ability to maintain a list of MAC addresses that are allowed to connect to the wireless LAN. The MAC addresses on wireless cards are associated with specific users and used as a means of restricting access to the wireless network.

Unfortunately, this means of access control can be circumvented by an attacker. Each packet sent from a legitimate workstation identifies the source MAC address that sent the packet. An attacker who discovers a network restricting access based on MAC addresses can simply monitor the network to identify people who are sending traffic on the network. The attacker can identify valid MAC addresses that are permitted to use the wireless network, regardless of encryption protocols in use on the network. With a list of authorized clients, the attacker can simply select an authorized MAC address and change their wireless card to utilize the same MAC. After the attacker updates his computer with an authorized MAC address, they can communicate on the network as if they were an authorized user.

The lesson for this misconception is that attackers can easily circumvent MAC-based access control, which is not a viable means of access control for wireless networks.

"Technology XYZ by itself protects us."
No single technology can sufficiently protect wireless networks. Administrators wishing to protect their networks should consider deploying many security layers, including strong encryption methods, distributed firewalls, and intrusion detection systems between wireless and wired networks, personal firewalls on wireless clients, and authentication, authorization, and accounting services.



Risk Misconceptions

- "DoS attacks require expensive hardware that is not easily accessible"
- "Segregating our wireless LAN eliminates our risk of exposure"
- "This wireless thing is secure by default, right?"

The following are some risk misconceptions that people often have about wireless.

"DoS attacks require expensive hardware that is not easily accessible."
Unfortunately, this statement could not be more incorrect. DoS attacks against wireless networks are easy to implement because of weaknesses in the 802.11 specification and are impossible to prevent with current technology. With an inexpensive wireless card and readily available software, an attacker can launch DoS attacks against victim networks that completely disable all access to the wireless LAN. He can launch these attacks with breadth and target them against an entire building or office park, or against a single wireless LAN—potentially from a distance of several miles.

The lesson for this misconception is that DoS attacks are a real threat to organizations, and administrators have little opportunity to limit their exposure to attack.

"Segregating our wireless LAN eliminates our risk of exposure."
A common method for securing the implementation of wireless LANs is to segregate the wireless LAN away from internal networks with a firewall. Although it is a strong component of an overall wireless LAN security plan, this implementation still poses significant risks to an enterprise network.

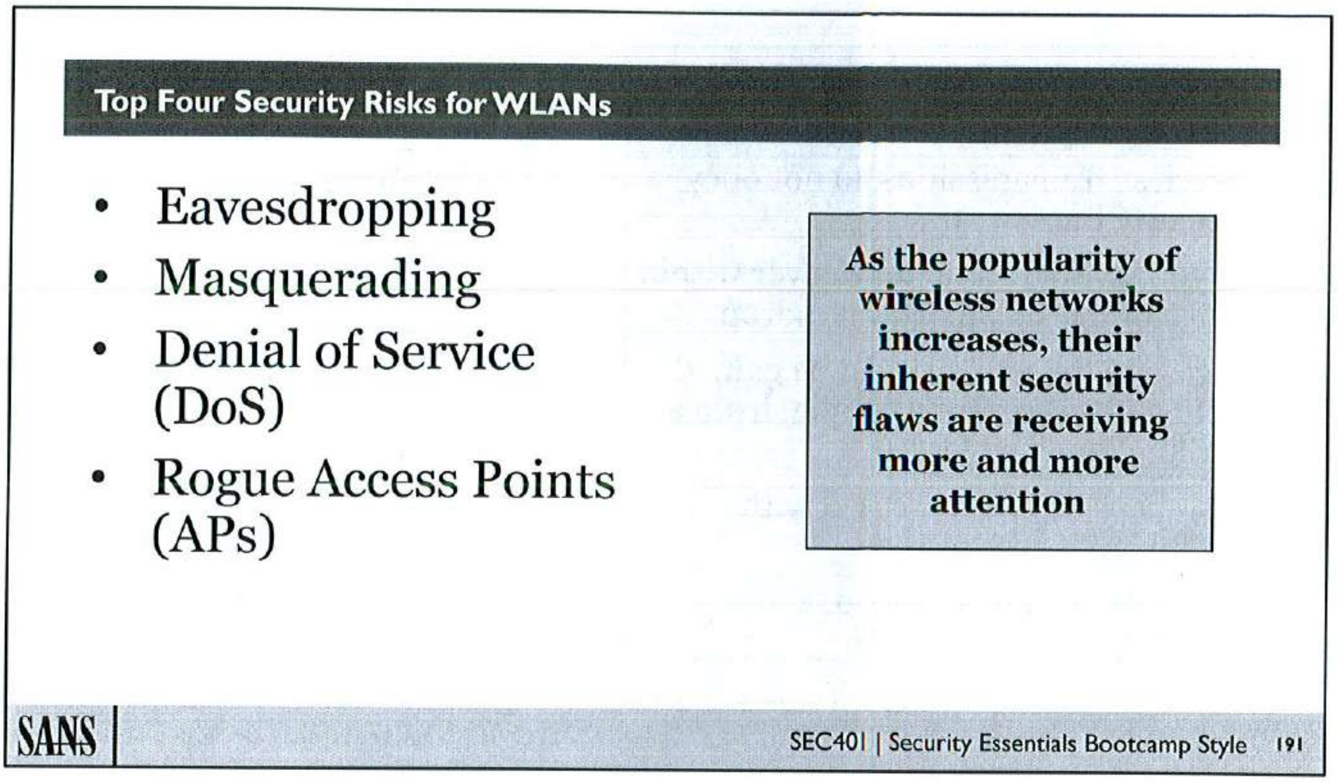
Segregated networks typically open access only to those applications that are required for wireless clients or VPN systems that require authentication and strong encryption to access internal systems.

An attacker wishing to circumvent the security of these installations has several attack options. Application-level flaws can be vulnerable to attack on exposed servers, or an attacker can attempt to exploit vulnerable clients to access established VPN tunnels and thus, internal networks.

The lesson for this misconception is that, although segregating wireless LANs away from production networks is a critical component of securing the enterprise, a defense-in-depth approach must be applied to adequately defend against attack.

"This wireless thing is secure by default, right?"

It is important for all levels of an organization—from the wireless LAN technicians up to the chief security officer—to understand the risks associated with deploying wireless networks. Obviously, the deployment of wireless LANs is a completely different paradigm than traditional wired networks. When securing enterprise networks, communicating the risks, deficiencies, and advantages of wireless LAN deployment to all levels of an organization is clearly a critical part of a defense-in-depth strategy.



Top Four Security Risks for WLANs

- Eavesdropping
- Masquerading
- Denial of Service (DoS)
- Rogue Access Points (APs)

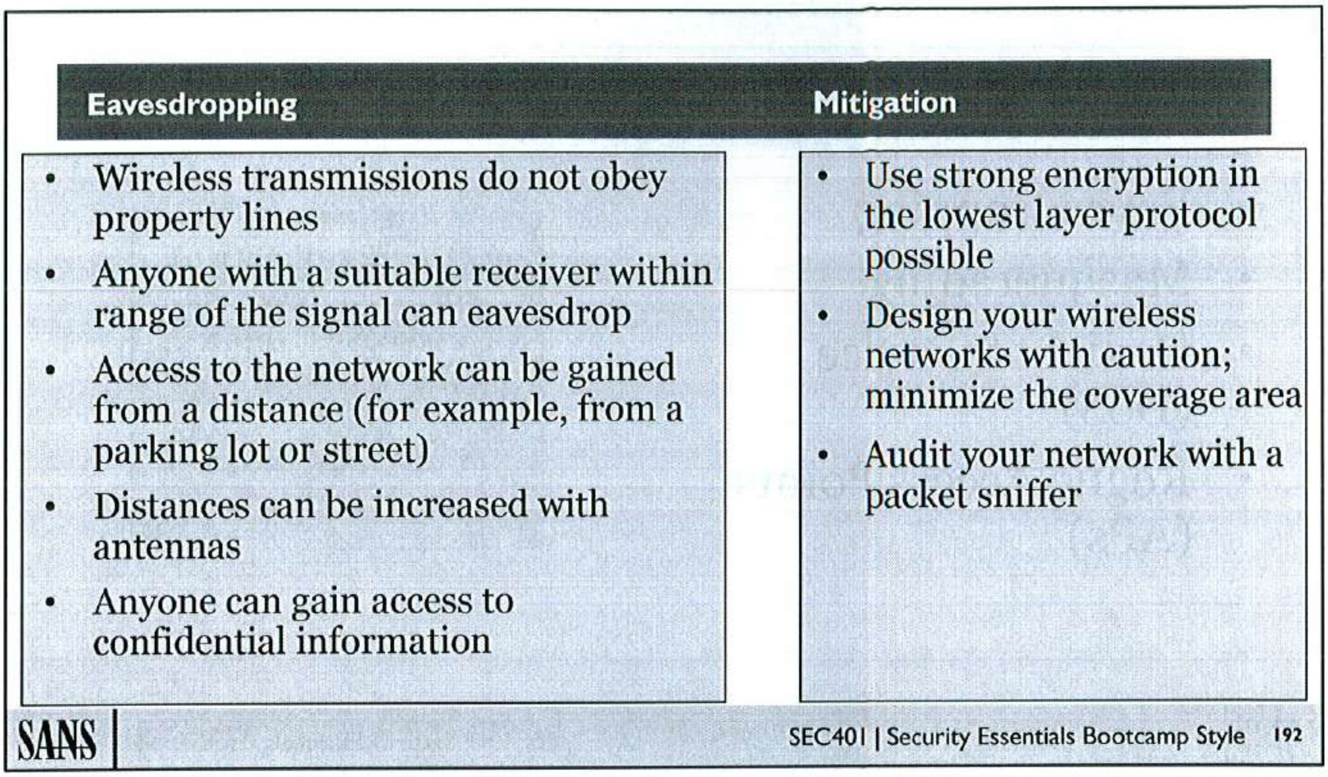
As the popularity of wireless networks increases, their inherent security flaws are receiving more and more attention

SANS

SEC401 | Security Essentials Bootcamp Style 191

As the popularity of wireless networks increases, their inherent security flaws are receiving more and more attention. In recent years, wireless security (or lack thereof) has become the press' media darling. Unfortunately, these reports are often incomplete or incorrect, and they often leave organizations with a false sense of security. This section focuses on the most critical security issues related to wireless networking:

- Eavesdropping
- Masquerading
- Denial of Service (DoS)
- Rogue APs



Eavesdropping

- Wireless transmissions do not obey property lines
- Anyone with a suitable receiver within range of the signal can eavesdrop
- Access to the network can be gained from a distance (for example, from a parking lot or street)
- Distances can be increased with antennas
- Anyone can gain access to confidential information

Mitigation

- Use strong encryption in the lowest layer protocol possible
- Design your wireless networks with caution; minimize the coverage area
- Audit your network with a packet sniffer

SANS | SEC401 | Security Essentials Bootcamp Style 192

Eavesdropping is a trivial matter in a wireless RF environment. Data sent over the radio path can be intercepted by anyone equipped with a suitable device that happens to be listening on the same frequency. As if that was not bad enough, the devices needed to perform eavesdropping are inexpensive and easy to use. But wait, it gets even worse! It is virtually impossible to detect a hacker listening in on your wireless communication.

As we learned earlier, wireless RF has the capability to travel beyond the confines of a building, and the signal can often be picked up from a distance. Would-be attackers can eavesdrop on wireless networks from remote locations, such as a company parking lot or building lobby, and potentially gain access to confidential information.

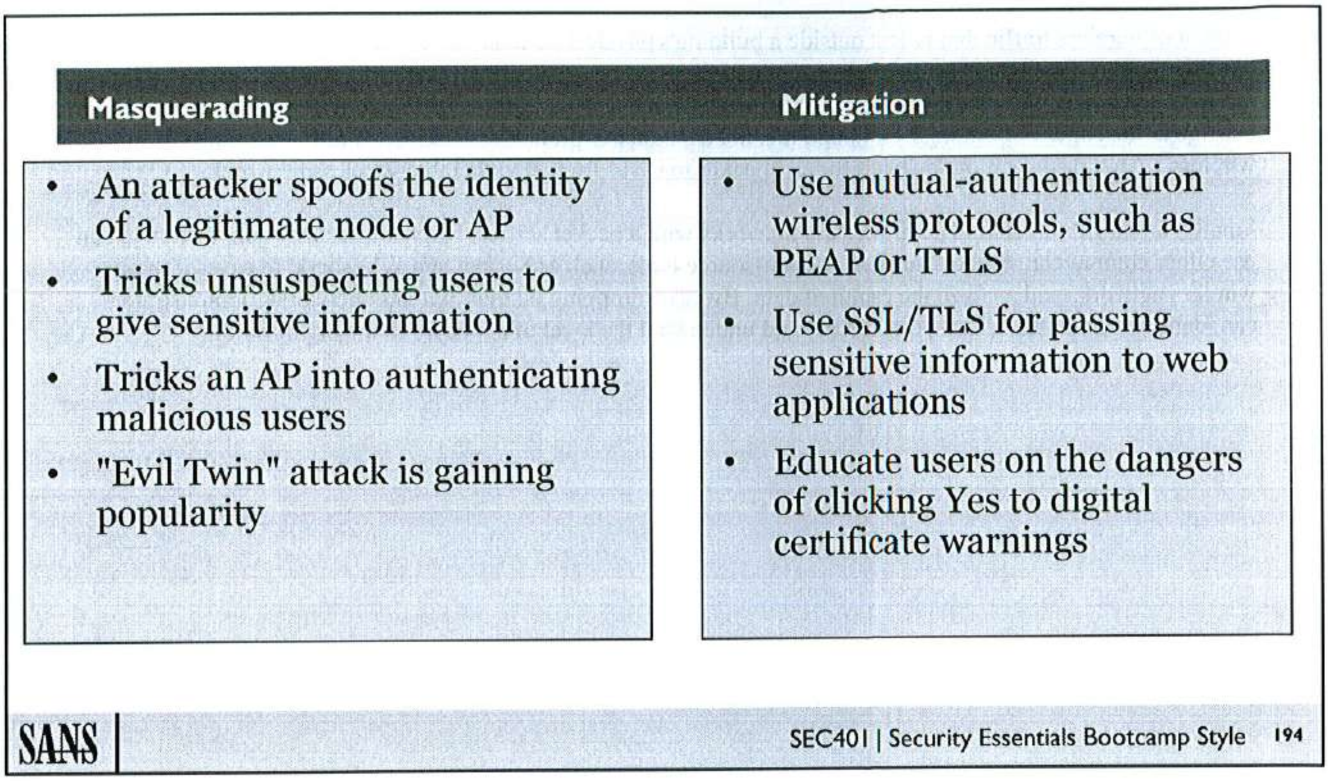
An attacker wishing to eavesdrop on a wireless network likely wants to place as much distance as possible between the victim network and the attacker's location to avoid detection. To aid them in this venture, attackers employ range-extending antennas connected to their wireless cards. These antennas are sometimes commercial tools purchased over the Internet, in ham radio shops, or are home-brewed using nothing more than basic components.

Even if an attacker can hear a transmission, they cannot make sense of the information if the data is protected by encryption. It is important to use strong encryption methods that operate at the lowest possible layer of the OSI model. Consider the amount of information an attacker can glean from the Cisco Discovery Protocol (CDP), a protocol that is not encrypted when using IPsec or other encryption protocols that work at the network layer of the OSI model.

We have seen that the WEP protocol is inadequate for protecting wireless networks. Organizations should deploy stronger encryption protocols, such as TKIP (WPA), if they do not adopt WPA2, which uses AES encryption.

When designing wireless networks, it is possible to select range-limiting antennas to limit the exposure of information outside of an organization's walls. Using alternate antennas, limiting signal output strength on radio cards, and placing wireless access points away from the exterior reaches of buildings will reduce the amount of wireless traffic that is sent outside a building's physical boundaries. Organizations should also consider working with their facilities' management departments to employ RF-limiting materials, such as wire-mesh installed in walls, ground-connected metal studs and beams, and even metal-additive in paint to limit RF leakage at the edges of buildings. Although it is not a singularly protective measure, limiting the range of wireless LANs makes it more difficult for an attacker to eavesdrop on vulnerable wireless networks.

Finally, organizations should audit wireless networks with a packet sniffer. LAN and security administrators can use either commercial wireless sniffers or open-source tools, such as Kismet and Wireshark, to eavesdrop on wireless networks and analyze the captured data. By eavesdropping on their wireless networks, administrators can identify vulnerable access points (APs) and understand the level of exposure to the organization.



Masquerading

- An attacker spoofs the identity of a legitimate node or AP
- Tricks unsuspecting users to give sensitive information
- Tricks an AP into authenticating malicious users
- "Evil Twin" attack is gaining popularity

Mitigation

- Use mutual-authentication wireless protocols, such as PEAP or TTLS
- Use SSL/TLS for passing sensitive information to web applications
- Educate users on the dangers of clicking Yes to digital certificate warnings

SANS

SEC401 | Security Essentials Bootcamp Style 194

Masquerading describes the activities of an attacker who impersonates the identity of legitimate nodes or access points in a wireless network. The attacker accomplishes this by spoofing identity information to impersonate an otherwise authorized client or access point. By making clients think that they are communicating with a legitimate access point, attackers can trick unsuspecting users into giving sensitive information, trick access points into believing that they are authorized clients, or launch DoS attacks.

By impersonating a legitimate access point, an attacker can offer network services to unsuspecting wireless users and try to trick them into giving up sensitive information, such as usernames, passwords, or even credit card information.

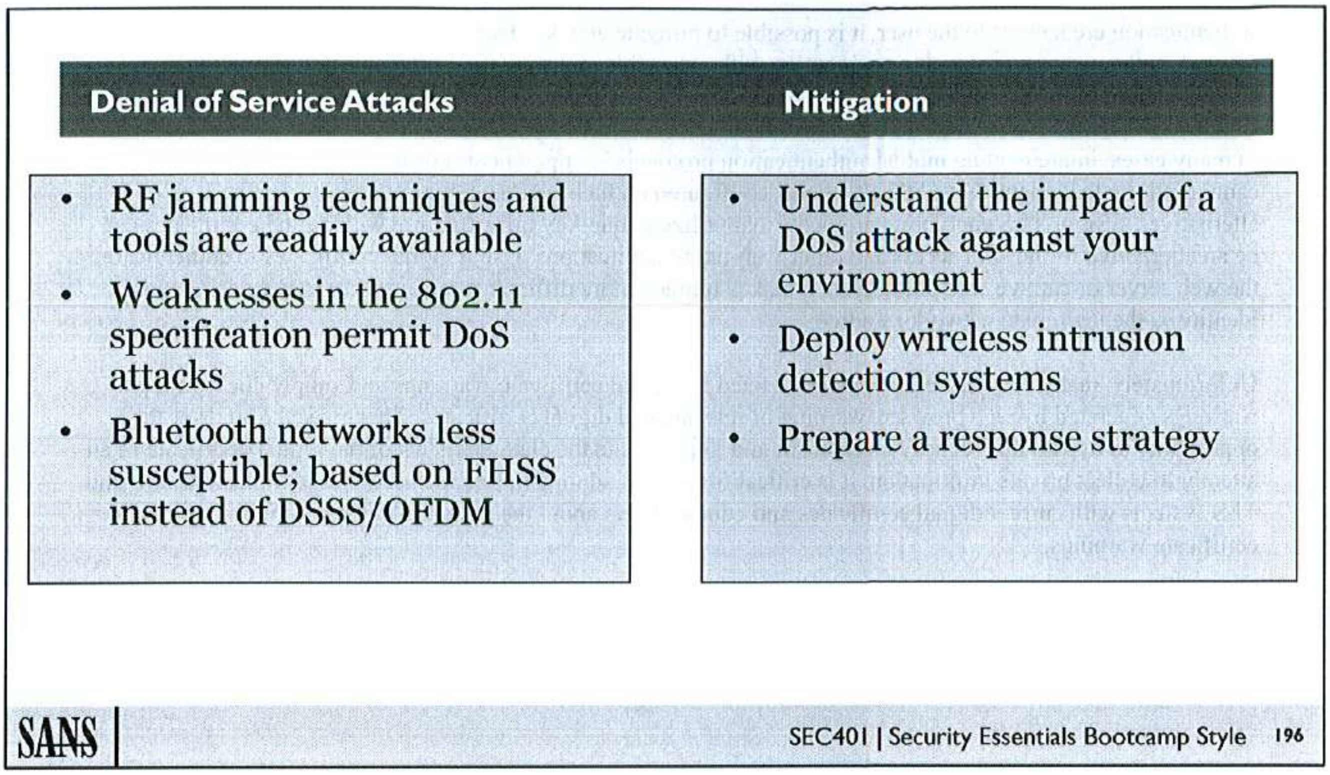
Captive web portals are a common means of authenticating wireless users without requiring an authentication client on workstations that need to access the wireless network. Commonly used for authenticating users at hotspots, for guest access to wireless networks, or at colleges that offer wireless access to students, a captive web portal intercepts requests for a web page and substitutes the page with a form that requests authentication. If a user enters authorized authentication credentials, they are granted access to resources beyond the web portal system. To grant access to only legitimate systems, the captive web portal system must keep track of authenticated and unauthenticated users. The web portal system tracks the MAC addresses of authenticated clients to permit access to network resources. Systems that use MAC addresses that are not in the explicit permit list are denied access until authentication.

To bypass this method of access control, an attacker can simply masquerade as an authenticated client by changing his MAC address. Sometimes, when combined with a DoS attack against the impersonated system, the attacker changes his MAC address to a system that was actively communicating on the network. The captive web portal system checks the traffic's MAC address and, unable to differentiate the attacker from the legitimate user, grants the attacker unrestricted access to the victim network.

To protect against masquerading attacks, you must have some mechanism in place to authenticate users to an access point and authenticate the access point to the user. By requiring the access point to present authentication credentials to the user, it is possible to mitigate attacks. This is possible using the IEEE 802.1X network authentication protocol in conjunction with extensible authentication protocols that support mutual authentication, such as EAP/TLS, PEAP, and TTLS.

In many cases, implementing mutual authentication protocols is impractical. For instance, hotspot locations cannot require that clients have 802.1X clients configured on their workstations and therefore must seek alternatives. SSL or TLS encryption protocols that utilize public-key infrastructure with digital certificates can be an alternative for hotspot access and other web-based applications. Using digital certificates to authenticate the web server or captive web portal system makes it much more difficult for an attacker to masquerade his identity as the legitimate network resource.

Unfortunately, many users have grown anesthetized by digital certificate warnings and simply click-through warnings generated by web browsers warning of mismatched digital certificates. This gives the attacker the opportunity to bypass this security mechanism and impersonate the characteristics of the digital certificate in an attempt to collect private information. It is critical for system administrators to successfully implement SSL and TLS systems with current digital certificates and educate users about the potential dangers of ignoring invalid certificate warnings.



Denial of Service Attacks

- RF jamming techniques and tools are readily available
- Weaknesses in the 802.11 specification permit DoS attacks
- Bluetooth networks less susceptible; based on FHSS instead of DSSS/OFDM

Mitigation

- Understand the impact of a DoS attack against your environment
- Deploy wireless intrusion detection systems
- Prepare a response strategy

SANS

SEC401 | Security Essentials Bootcamp Style 196

Wireless networks are an easy target for Denial of Service (DoS) attacks. Vulnerabilities in the 802.11 specification, flaws in the firmware of popular Wi-Fi cards, and weaknesses in the nature of radio communications offer attackers opportunities to shut down wireless networks at their leisure.

If an attacker has a powerful enough transceiver, they can generate so much radio interference that the targeted WLAN is unable to communicate effectively. Like eavesdropping, this kind of attack can be initiated from a distance. Although the attack is a bit more sophisticated than simple eavesdropping, the equipment needed is readily available, as well as instructions on how to carry out such an attack. Attackers can purchase RF-jamming equipment, which is designed to stress-test wireless frequencies to attack wireless networks. Alternatively, attackers can build their own RF-jamming tools using inexpensive hardware from popular electronics stores and plans that are available on the Internet. These tools are effective at stopping all wireless activity, often covering all available channels in the 2.4-GHz or higher frequencies. These RF-jamming attacks are specification-agnostic; they are equally effective against 802.11 and Bluetooth networks, as well as any other communication that uses the same frequency as the attacker.

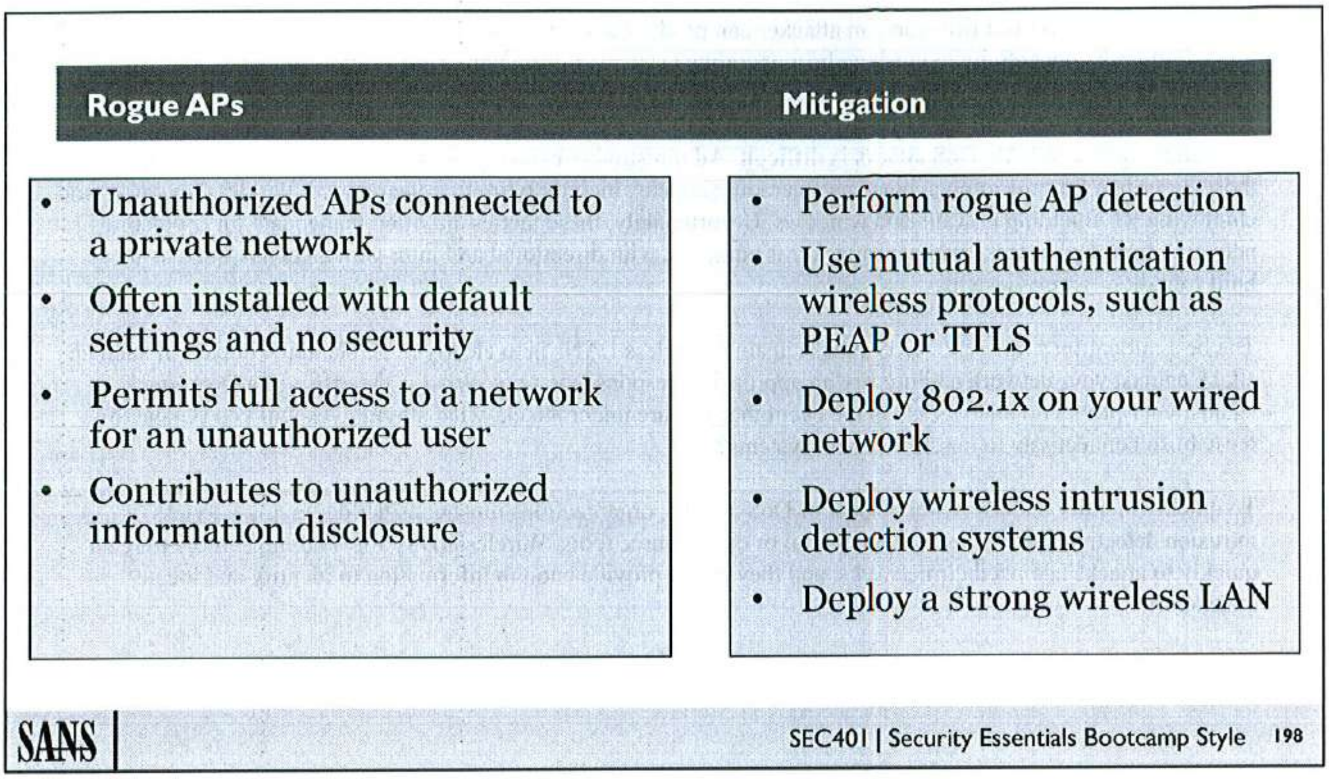
Using commodity wireless cards, attackers can masquerade their identity as legitimate stations or access points to launch DoS attacks. Because the 802.11 specification does not include any per-packet authentication mechanism, access points and stations do not have a way of verifying that each packet is indeed sent from its reported source address. Attackers use this weakness to send spoofed packets to victim clients on behalf of the access point, telling the victim to disconnect from the network. The victim station processes this packet as if the traffic was sent from the access point and disconnects from the wireless network. An attacker can send a sustained flood of these disconnect packets to the LAN broadcast address, thereby causing all stations to disconnect from the network, resulting in a sustained DoS attack.

All wireless cards rely on firmware that is bundled in non-volatile RAM to handle time-sensitive transmission functions. Administrators often overlook this firmware because it rarely requires upgrades and offers no configuration options. Recently uncovered vulnerabilities and flaws in Wi-Fi card firmware have led to a more effective means of launching DoS attacks against 802.11 networks. By sending specifically malformed frames to stations that run flawed firmware, an attacker can produce several undesirable results, ranging from complete loss of network connectivity to crashing host operating systems. Fortunately, card vendors have begun recognizing these flaws and are offering patched firmware to resolve these vulnerabilities.

Protecting against WLAN DoS attacks is difficult. Administrators can employ the same mitigation strategies as those described for protecting against masquerading attacks, including limiting the range of wireless networks and employing RF shielding in walls and windows. Unfortunately, these tactics are often inadequate for protecting networks from DoS attacks when an attacker is equipped with directional and high-gain antennas, such as those built from home-brew designs.

The best mitigation strategy for DoS attacks against wireless LANs is to clearly understand the impact of such an attack against your network and prepare an appropriate response strategy. What is the effect of a DoS attack against your production networks? In the event that you are under attack, what alternatives will you pursue to reestablish connectivity to mission-critical systems?

To quickly identify and assess the impact of DoS attacks, organizations must consider deploying wireless intrusion detection systems using commercial or open-source tools. Wireless IDSs allow administrators to react quickly to attacks against their networks, and they might provide enough information to identify and locate attackers.



Rogue APs

- Unauthorized APs connected to a private network
- Often installed with default settings and no security
- Permits full access to a network for an unauthorized user
- Contributes to unauthorized information disclosure

Mitigation

- Perform rogue AP detection
- Use mutual authentication wireless protocols, such as PEAP or TTLS
- Deploy 802.1x on your wired network
- Deploy wireless intrusion detection systems
- Deploy a strong wireless LAN

SANS | SEC401 | Security Essentials Bootcamp Style 198

A rogue access point (AP) is connected to a wired network without the authorization to provide wireless service to end users. Users who want wireless access or are unhappy with existing wireless services expose an organization's network by connecting an access point to the wired networks. These access points are commonly meant for home use and rarely offer anything beyond the most basic security settings. Attackers who identify these rogue access points can exploit the basic security settings and gain access to internal network resources. Administrators are often unaware of rogue access points on their networks until they are discovered as part of a vulnerability assessment or system compromise analysis.

Rogue access points can also contribute to unauthorized information disclosure when attackers eavesdrop on these connections. An employee might decide to deploy a rogue access point in a conference room with the intention of enabling a workgroup to easily communicate and share documents. Although well intentioned, the user often does not realize the risk and exposure of such an installation and cannot detect an attacker who harvests all the shared documents from a parking lot or other off-site location.

Rogue access points with little or no security pose an obvious threat to any organization. Often, organizations that have adopted "no wireless" policies are plagued with rogue access points because of the lack of any wireless access available to users.

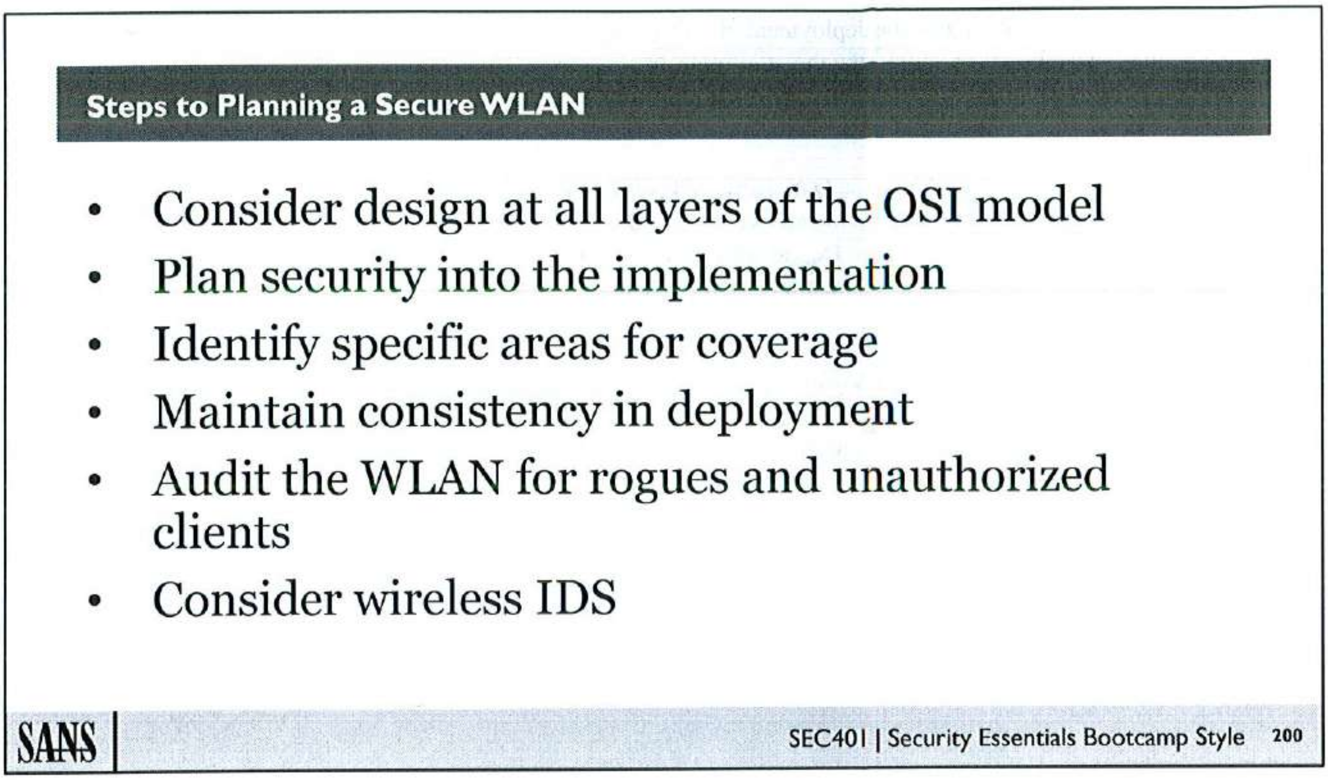
To mitigate rogue access points, administrators should perform rogue AP detection using commercial or open-source tools, such as Kismet and Wireshark. This method requires an administrator equipped with a laptop or handheld device to walk through the hallways and offices of all the buildings that might be risks for rogue access points to detect any unauthorized wireless activity. Unfortunately, this can be a difficult venture for large campuses that require alternate detection methods.

Scanning wired networks for characteristics that resemble wireless access points is an alternate, yet less reliable method of detecting rogue access points. Using vulnerability assessment tools, an administrator can scan all the nodes on the wired network to identify web pages, login banners, and other characteristics to identify potential rogue access points. This method is useful in detecting users who are not trying to hide their activity with otherwise stealthy tactics, such as shutting off ICMP echo responses and disabling administrative interfaces that might be used to identify them.

Some organizations are planning on the deployment of 802.1X network authentication on their wired networks to mitigate (among several security issues) the threat of rogue access points. By requiring nodes to authenticate to the network before being granted access, administrators can prevent users from connecting access points to their production networks.

In lieu of manual rogue AP detection, organizations should consider deploying WLAN intrusion detection systems to constantly monitor their facilities for rogue access points. Some WLAN IDS systems even implement "rogue AP countermeasures," which use attacker-like DoS attacks against discovered rogue access points to prevent anyone from connecting to the rogue until an administrator visits the site to remove the offending hardware.

Finally, organizations plagued with rogue APs should consider deploying wireless networks for their users. By taking control of wireless equipment deployment, administrators are in a much better position to set a policy that dictates how wireless LANs are used and to design and implement a preferred security solution for end users. Users are less likely to deploy their own rogue access points if a stable and reliable wireless LAN is available.



Steps to Planning a Secure WLAN

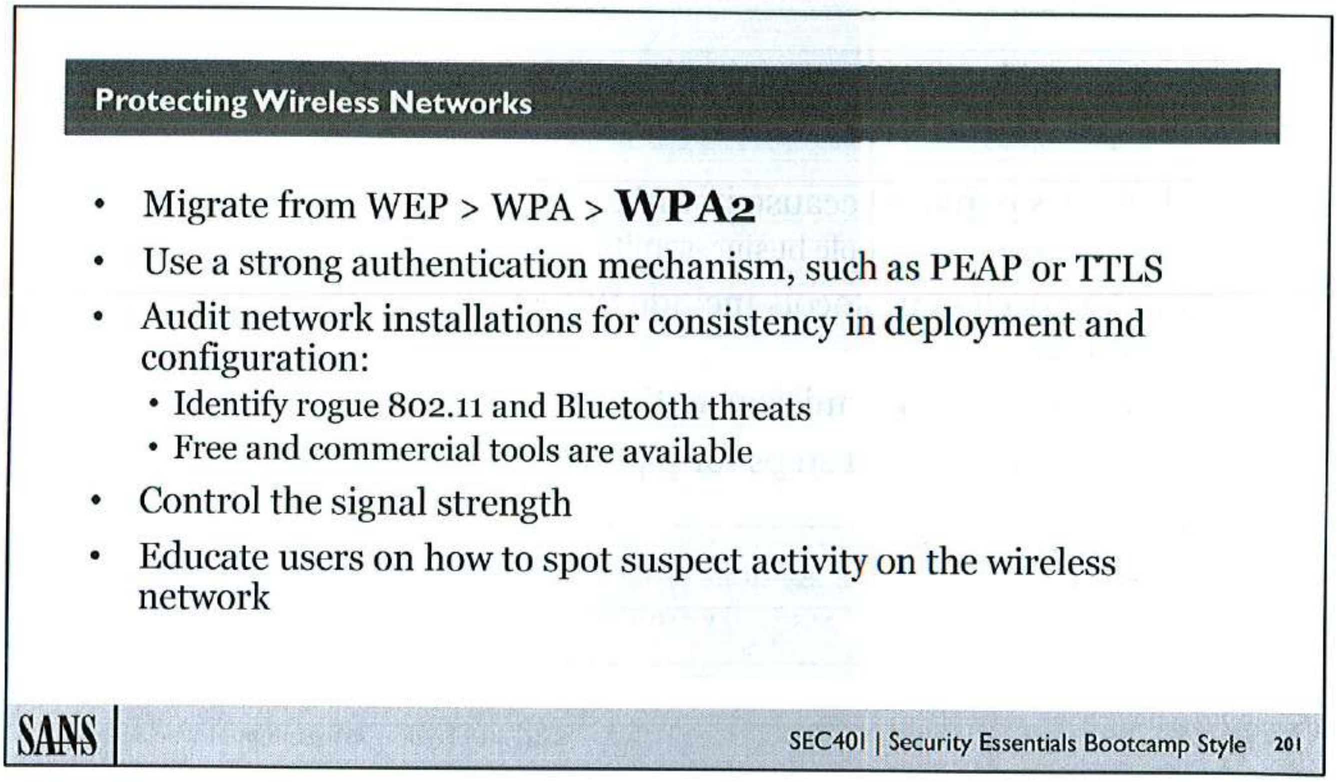
- Consider design at all layers of the OSI model
- Plan security into the implementation
- Identify specific areas for coverage
- Maintain consistency in deployment
- Audit the WLAN for rogues and unauthorized clients
- Consider wireless IDS

The key to protecting any wireless network is to employ a layered defense. There isn't any one technology that stops all attackers, but multiple layers of protection are more likely to defend wireless systems.

When designing wireless networks, consider security mechanisms at all layers of the OSI model. We have suggested using careful consideration when deploying wireless access points to limit the RF coverage at Layer 1 to mitigate eavesdropping and Denial of Service attacks and using strong encryption algorithms to protect data confidentiality at Layer 2. We can continue this process to carefully evaluate and deploy appropriate security mechanisms, such as firewalls at Layers 3 and 4, strong upper-layer protocols at Layers 5 and 6 that support non-repudiation of data, and application-layer defenses such as hardened systems and applications at Layer 7.

Deploying wireless LAN equipment in a consistent manner under a formal change-control process reduces the probability of attack due to misconfigured equipment. It is not uncommon to discover an access point that has default SNMP community strings, or no administrative password set. Using a consistent configuration and employing auditing of deployed systems helps reduce this threat. Administrators should regularly audit their facilities to detect rogue access points and other unauthorized equipment connected to corporate networks.

Wireless IDS systems are starting to become a more common tool for protecting wireless networks. Capable of alerting administrators to attacks and rogue access points, a wireless IDS is a valuable addition to securing networks even if you are not currently deploying wireless LANs.



Protecting Wireless Networks

- Migrate from WEP > WPA > **WPA2**
- Use a strong authentication mechanism, such as PEAP or TTLS
- Audit network installations for consistency in deployment and configuration:
  - Identify rogue 802.11 and Bluetooth threats
  - Free and commercial tools are available
- Control the signal strength
- Educate users on how to spot suspect activity on the wireless network

SANS SEC401 | Security Essentials Bootcamp Style 201

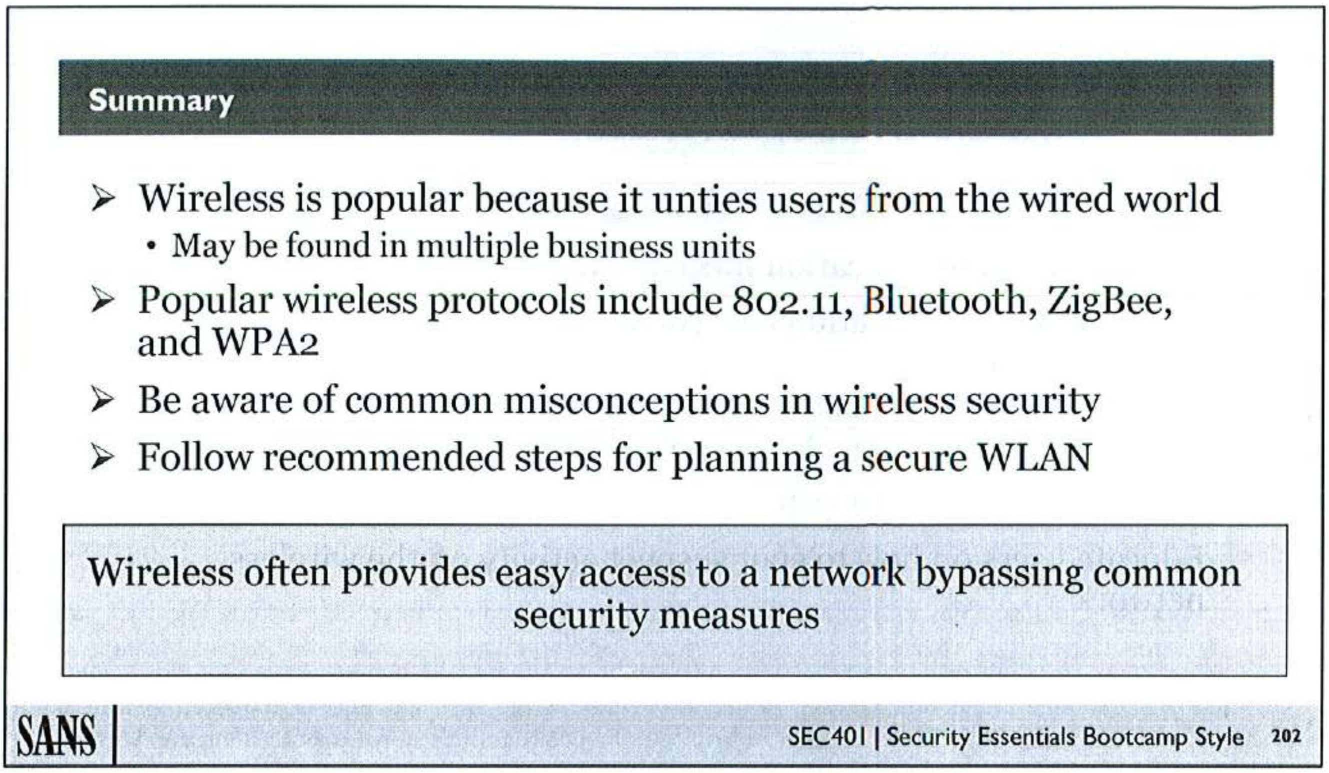
A few final recommendations on securing wireless networks follow.

For 802.11 networks, use authentication methods, such as PEAP or TTLS, that are capable of performing mutual authentication to mitigate man-in-the-middle attacks and masquerading attacks.

Always require mutual authentication between clients and infrastructure equipment. Trusting that the access point, Bluetooth device, or another wireless gateway are legitimate endpoints without verifying their authenticity likely results in compromised security for an organization.

Audit network installations to ensure that access points are deployed in a consistent manner. Eliminate the possibility of misconfigured access points with default administrator passwords, community strings, or HTTP-enabled configuration pages with consistent configurations for all equipment.

Finally, it is important to educate users on how to spot suspect activity on wireless networks. Consider the case of an attacker masquerading a hotspot in a coffee shop. If an end user is trained to never provide authentication credentials over an unencrypted HTTP connection, he can avoid having his username and password information falling into the wrong hands.



Summary

- ➢ Wireless is popular because it unties users from the wired world
  - May be found in multiple business units
- ➢ Popular wireless protocols include 802.11, Bluetooth, ZigBee, and WPA2
- ➢ Be aware of common misconceptions in wireless security
- ➢ Follow recommended steps for planning a secure WLAN

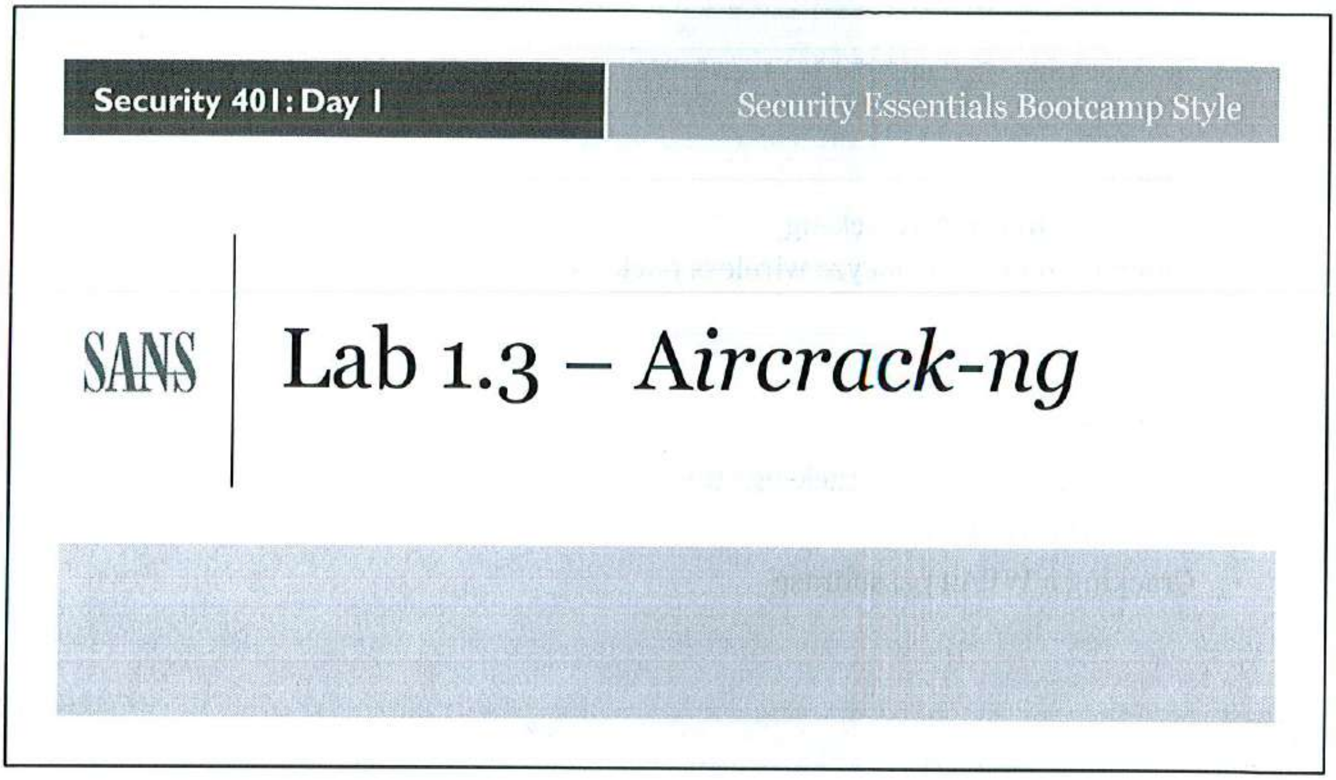
Wireless often provides easy access to a network bypassing common security measures

Users enjoy the freedom of roaming with their laptops, handheld computers, and mobile phones. As such, wireless networks have found their way into multiple areas of businesses and corporate networks.

Be aware of common misconceptions in wireless security. Relying on faulty information to protect the security of wireless networks can be a costly venture.

Be aware of the top four security risks for wireless networks: eavesdropping, masquerading, Denial of Service, and rogue access points.

Finally, use caution and follow recommended steps when designing wireless networks. Careful planning and execution is a critical component of protecting an enterprise from the risks associated with wireless networks.



Security 401: Day 1

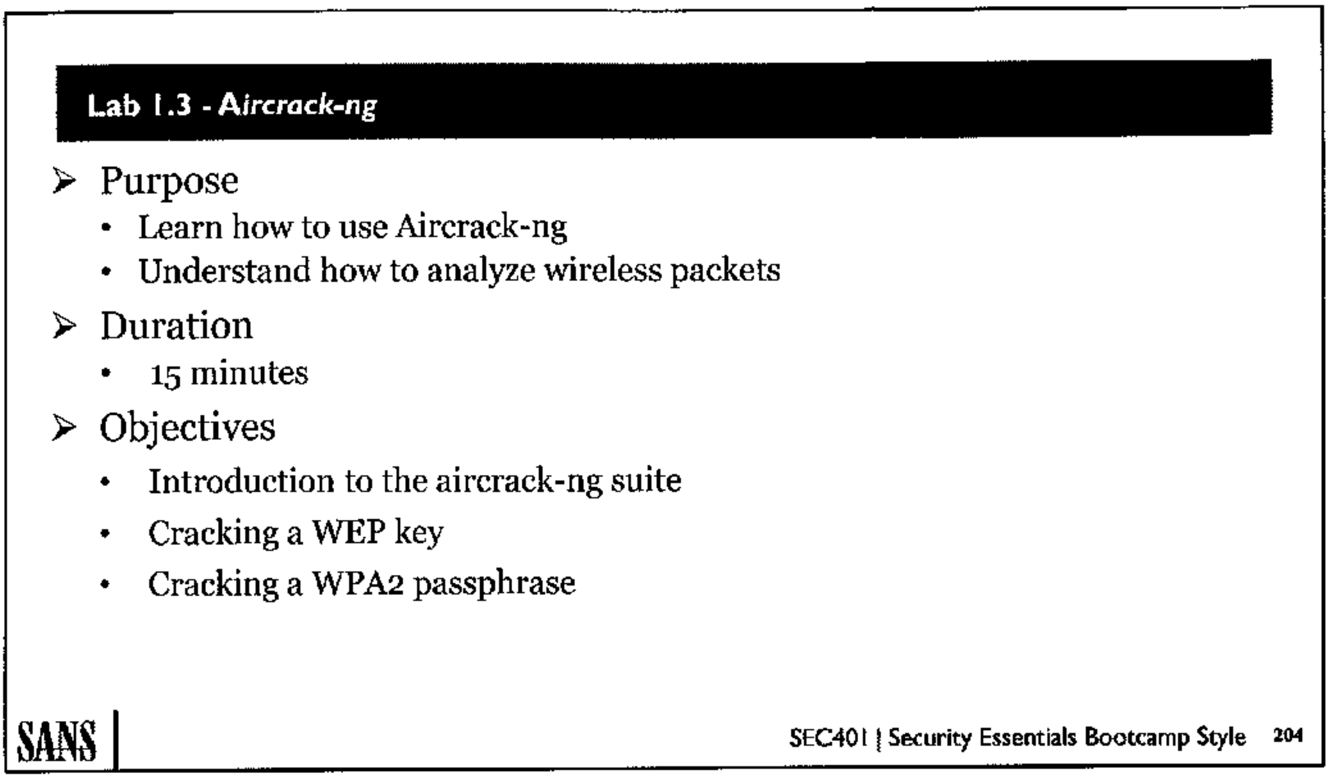
Security Essentials Bootcamp Style

SANS

Lab 1.3 – *Aircrack-ng*

In the previous module, you learned about the benefits and shortcomings of various widely used wireless protocols and technology. You use the aircrack-ng tool suite to assess the security of both the Wired Equivalent Privacy (WEP) security algorithm and Wi-Fi Protected Access (WPA) protocol associated with 802.11 wireless network security. Aircrack-ng is a freely available tool that is installed on your Kali Linux VM. The official distribution site is at http://www.aircrack-ng.org/. It was mostly written by Thomas d'Otreppe.

The use and role of each of these tools is covered accordingly when appropriate. Requiring each student to possess a wireless card that works with the drivers in Kali Linux and with these specific tools can be problematic, and therefore, wireless captures have been provided so that you experience the anticipated outcome for this lab.



Lab 1.3 - *Aircrack-ng*

- Purpose
  - Learn how to use Aircrack-ng
  - Understand how to analyze wireless packets
- Duration
  - 15 minutes
- Objectives
  - Introduction to the aircrack-ng suite
  - Cracking a WEP key
  - Cracking a WPA2 passphrase

SANS

SEC401 | Security Essentials Bootcamp Style 204

Purpose

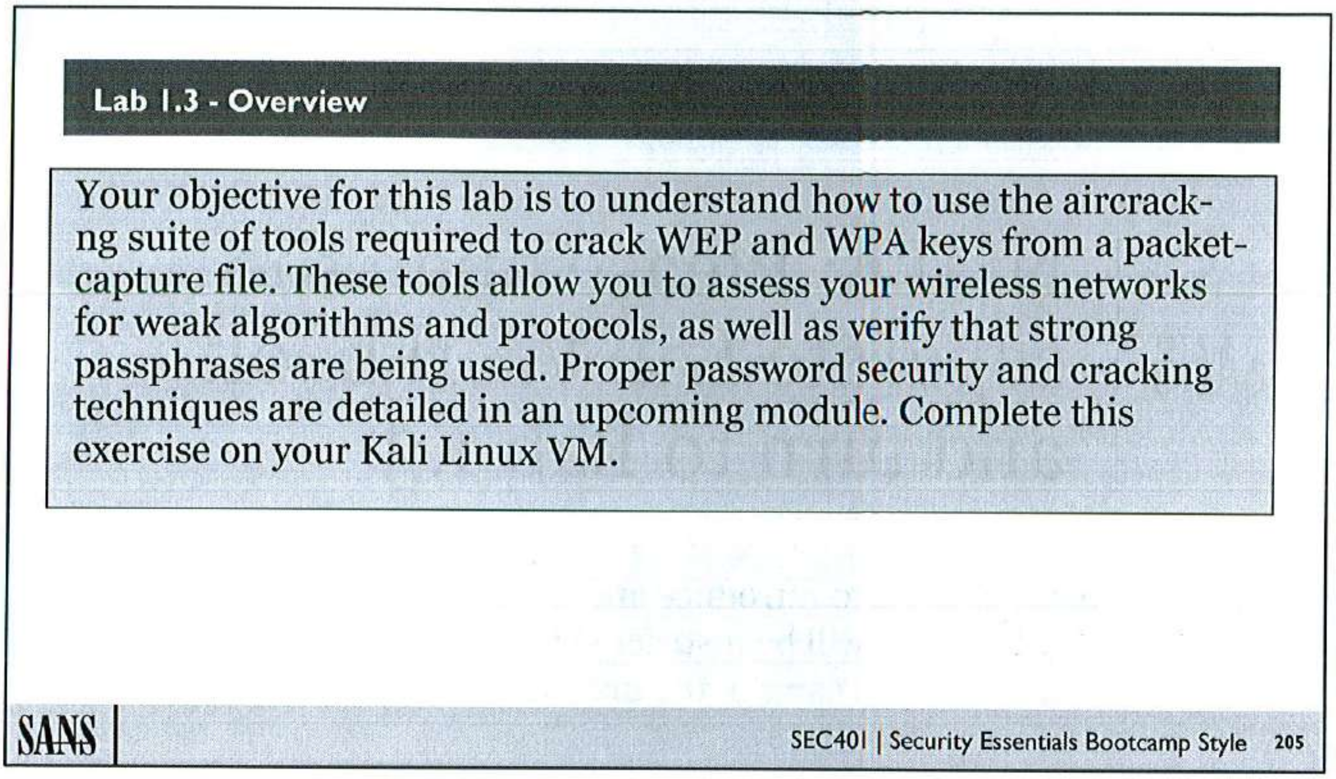
- Learn how to use Aircrack-ng
- Understand how to analyze wireless packets

Duration

- 15 minutes
- The estimated duration of this lab is based on the average amount of time required to make it through to the end. All labs are repeatable both inside and outside of the classroom, and it is strongly recommended that you take the time to repeat the labs both for further learning and practice toward the GIAC Security Essentials Certification (GSEC).

Objectives

- Introduction to the aircrack-ng suite
- Cracking a WEP key
- Cracking a WPA2 passphrase

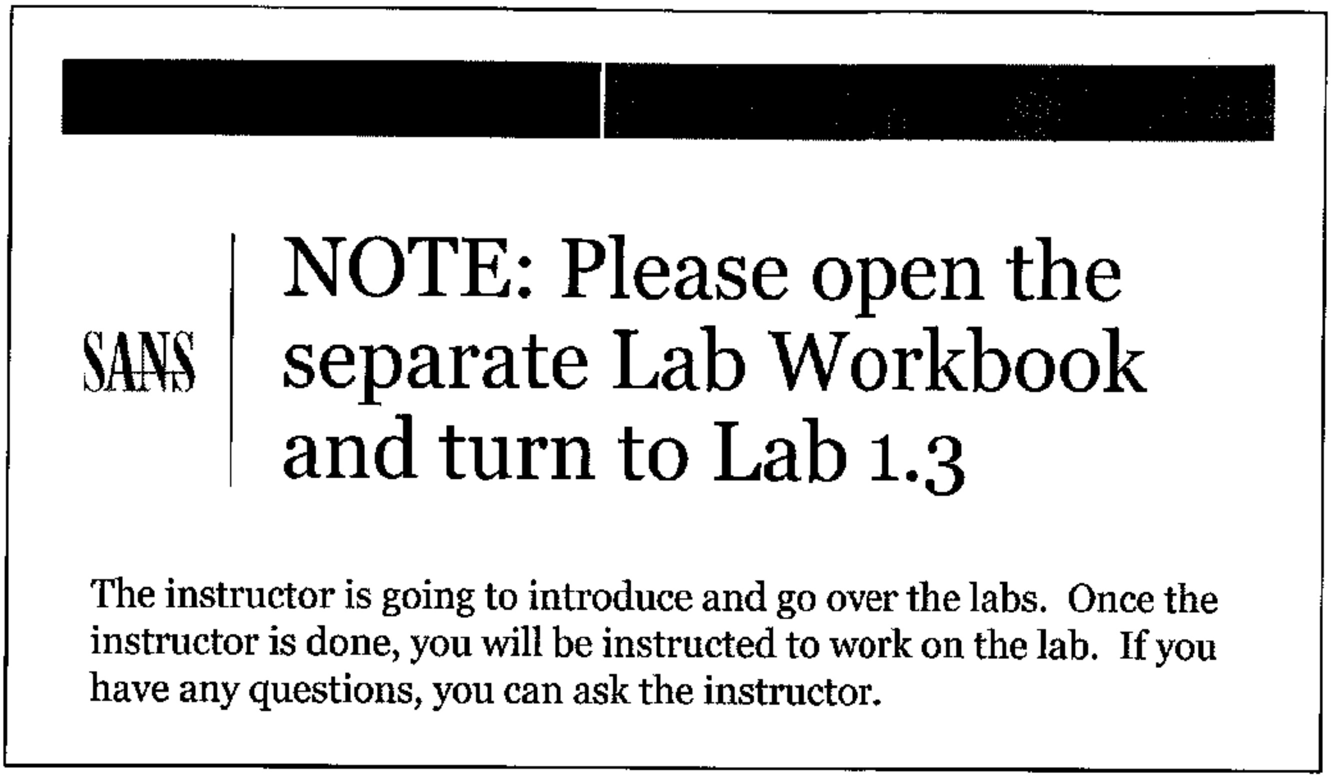


Lab 1.3 - Overview

Your objective for this lab is to understand how to use the aircrack-ng suite of tools required to crack WEP and WPA keys from a packet-capture file. These tools allow you to assess your wireless networks for weak algorithms and protocols, as well as verify that strong passphrases are being used. Proper password security and cracking techniques are detailed in an upcoming module. Complete this exercise on your Kali Linux VM.

SANS SEC401 | Security Essentials Bootcamp Style 205

Your objective for this lab is to understand how to use the aircrack-ng suite of tools required to crack WEP and WPA keys from a packet capture file. These tools allow you to assess your wireless networks for weak algorithms and protocols, as well as verify that strong passphrases are being used. Proper password security and cracking techniques are detailed in an upcoming module. Complete this exercise on your Kali Linux VM.

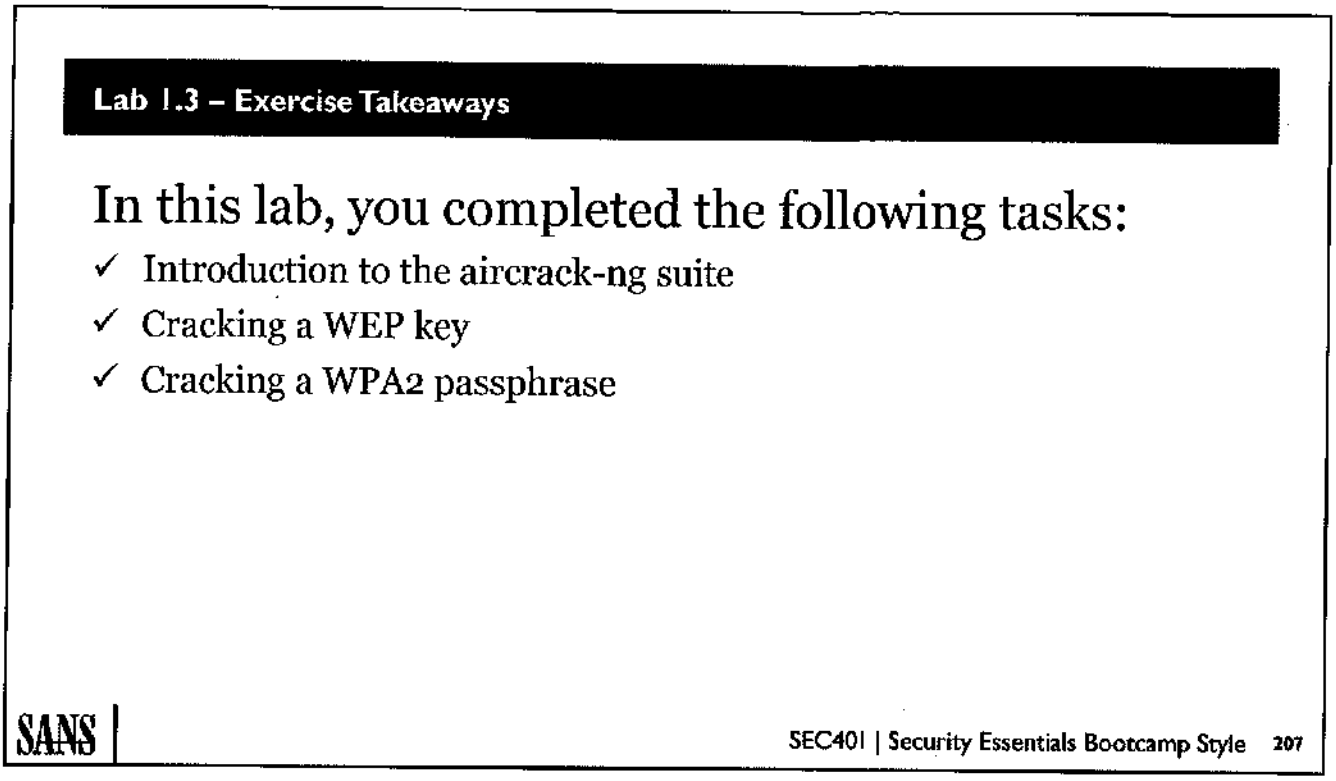


SANS

NOTE: Please open the separate Lab Workbook and turn to Lab 1.3

The instructor is going to introduce and go over the labs. Once the instructor is done, you will be instructed to work on the lab. If you have any questions, you can ask the instructor.

This page intentionally left blank.



Lab 1.3 – Exercise Takeaways

In this lab, you completed the following tasks:

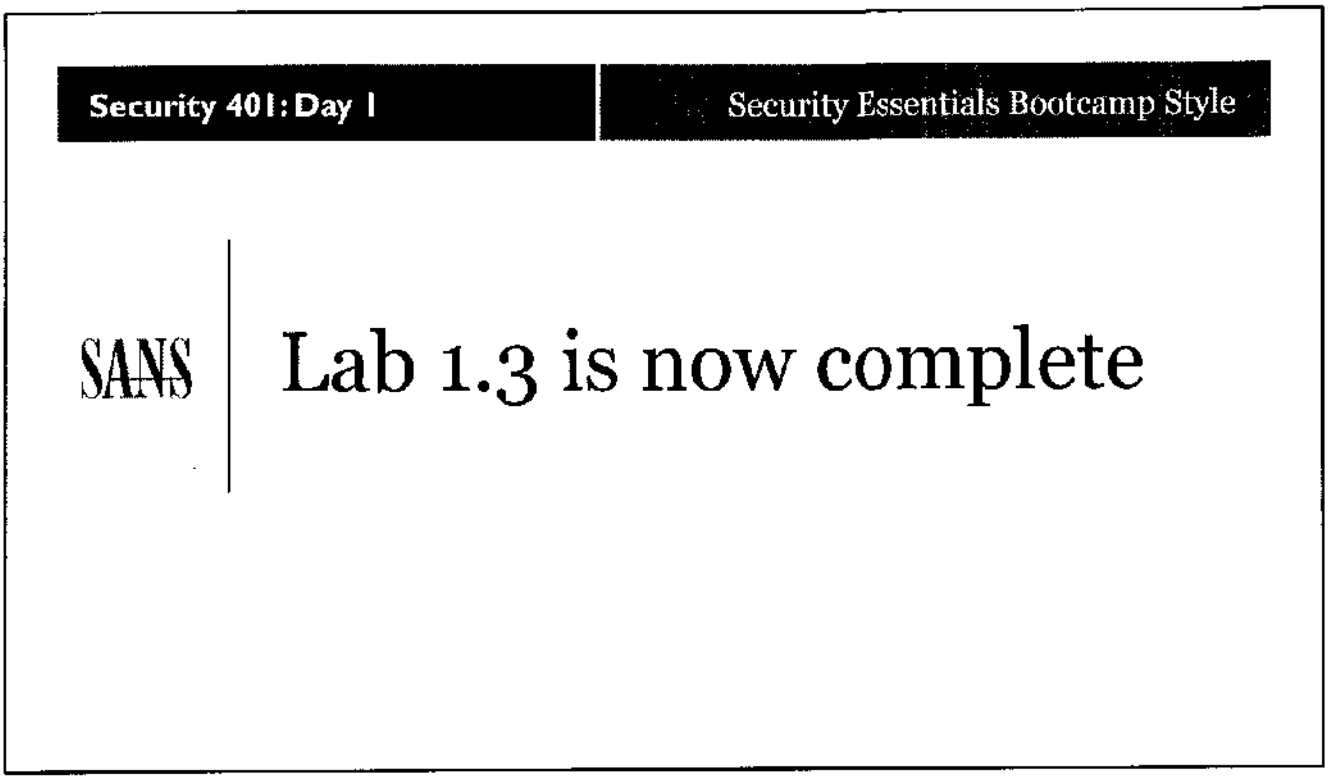
- ✓ Introduction to the aircrack-ng suite
- ✓ Cracking a WEP key
- ✓ Cracking a WPA2 passphrase

SANS | SEC401 | Security Essentials Bootcamp Style 207

In this lab, you completed the following tasks:

- ✓ Introduction to the aircrack-ng suite
- ✓ Cracking a WEP key
- ✓ Cracking a WPA2 passphrase

Many organizations use some form of wireless networking, whether it be 802.11, Bluetooth, ZigBee, cellular, RFIDs, or others. It is important to audit for the use of these technologies and maintain an inventory of approved implementations. Wireless networking can be incredibly convenient, but it must be appropriately balanced with security. An understanding of each technology and its limitations in regards to security can help you ensure that your organization is using the best options.

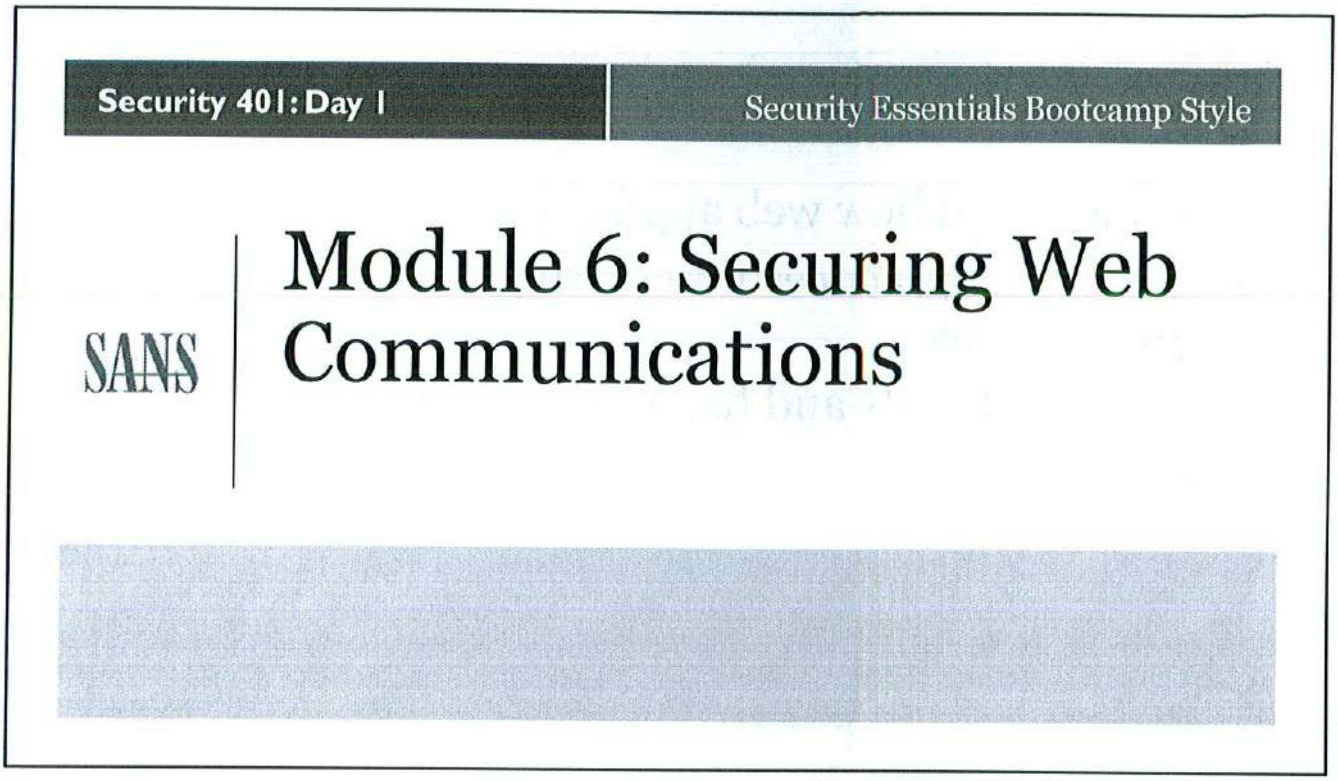


Security 401: Day 1

Security Essentials Bootcamp Style

Lab 1.3 is now complete

This page intentionally left blank.



Security 401: Day 1

Security Essentials Bootcamp Style

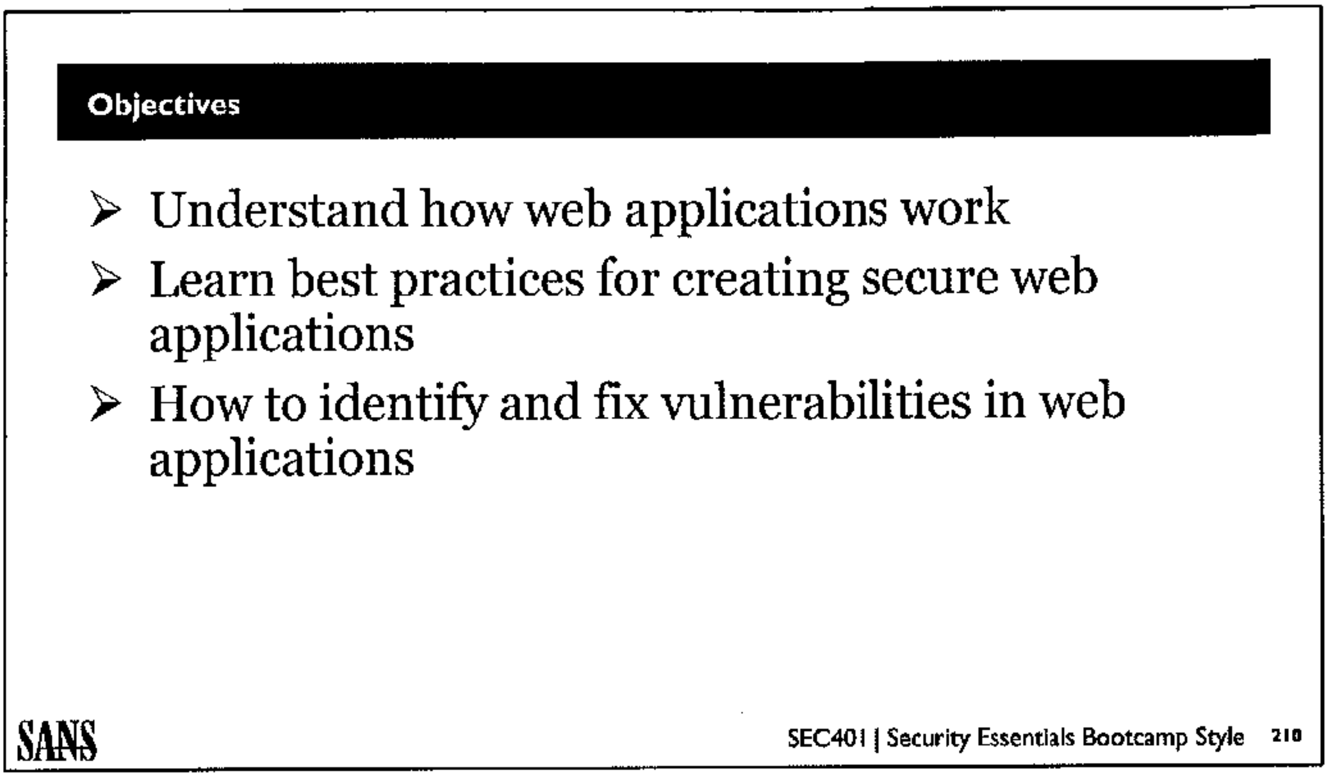
SANS

Module 6: Securing Web Communications

Module 6: Securing Web Communications

It seems as though half the world is on the web these days. From groceries to houses, from movie reviews to e-books, you can get almost any kind of goods or services online today. Things were much simpler in the early days of the web. At first, it was just a way of making static pages of information available to distant researchers. It wasn't nearly as useful as it is today, but at least the security model was well understood. In terms of protecting your server from compromise, standard host security practices were the order of the day.

Things are different now. Although a sizable portion of the web's content is still static, there are a lot of interactive web-based applications out there. Security is no longer simply a matter of hardening your web server; you must also design security into your applications and browsers. That's what this module is all about.



Objectives

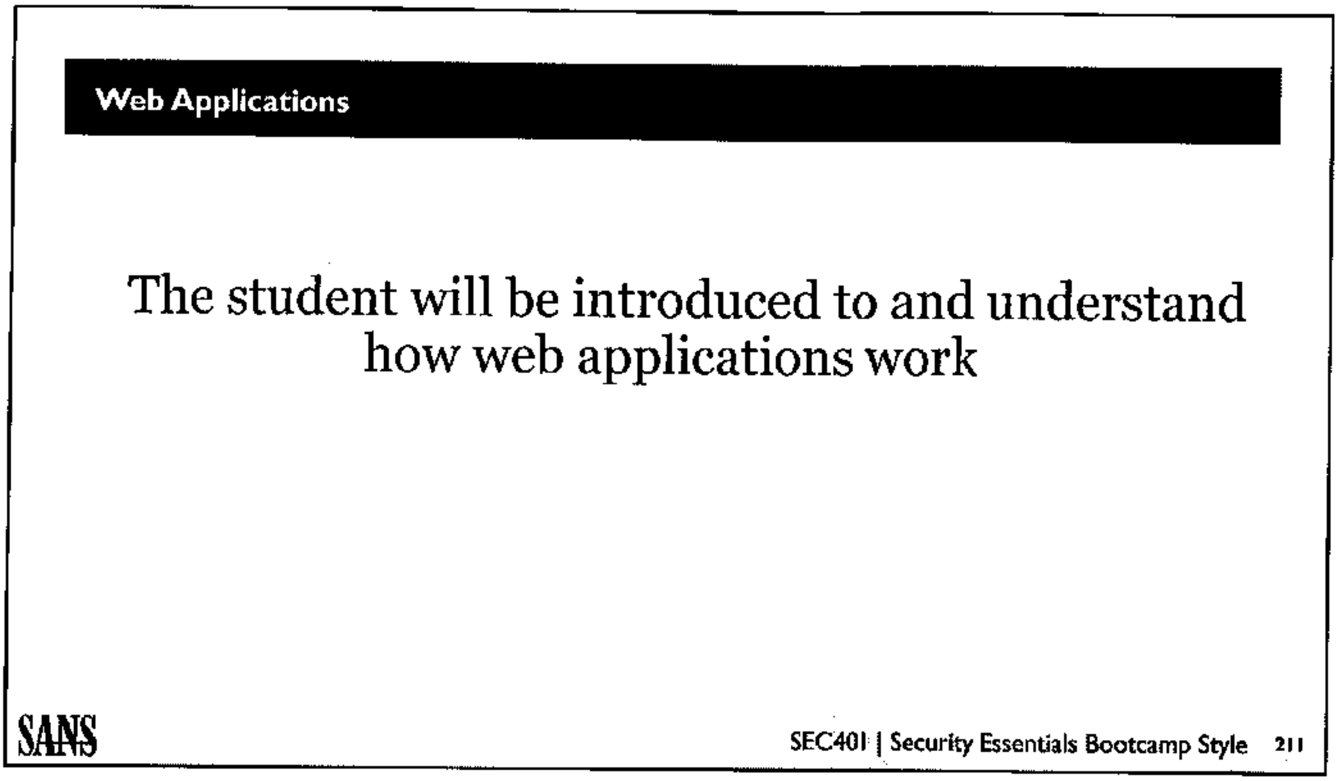
- Understand how web applications work
- Learn best practices for creating secure web applications
- How to identify and fix vulnerabilities in web applications

SANS

SEC401 | Security Essentials Bootcamp Style 210

In this module, we look at some of the most important things you need to know in order to design and deploy secure web applications. Our first stop on this journey is to explain the basics of how the web works. We are amazed at the number of technical professionals (even security practitioners) who deal with these technologies every day without really knowing how they work. It is very difficult to secure something if you do not understand how it works. A basic understanding of the underlying technology is an absolute prerequisite to secure design. We cover HTTP, HTML, forms, server, and client-side programming, cookies, authentication, and maintaining state.

After mastering the basics, we look at how to identify and fix vulnerabilities in web applications.



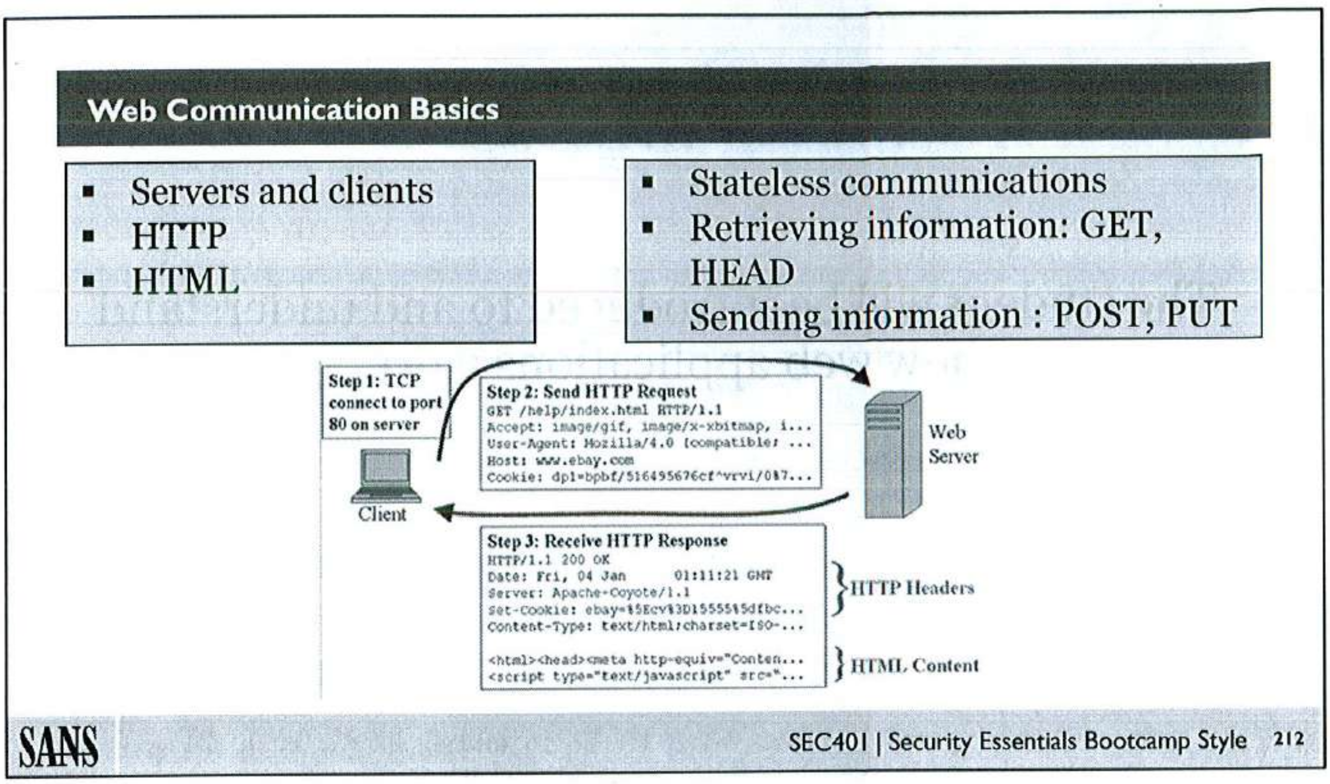
Web Applications

The student will be introduced to and understand how web applications work

SANS

SEC401 | Security Essentials Bootcamp Style 211

This page intentionally left blank.



Web Communication Basics

- Servers and clients
- HTTP
- HTML

- Stateless communications
- Retrieving information: GET, HEAD
- Sending information : POST, PUT

Step 1: TCP connect to port 80 on server

Step 2: Send HTTP Request

```
GET /help/index.html HTTP/1.1
Accept: image/gif, image/x-xbitmap, i...
User-Agent: Mozilla/4.0 (compatible; ...
Host: www.ebay.com
Cookie: dpl=bpbf/516495676cf^vrvi/0%7...
```

Step 3: Receive HTTP Response

```
HTTP/1.1 200 OK
Date: Fri, 04 Jan      01:11:21 GMT
Server: Apache-Coyote/1.1
Set-Cookie: ebay=%5Ecv%3D15555%5dfbc...
Content-Type: text/html;charset=ISO-...

<html><head><meta http-equiv="Conten...
<script type="text/javascript" src="...
```

SANS

SEC401 | Security Essentials Bootcamp Style 212

Before you can learn about the security of web-based applications, you must understand how they work. After all, you can't fix a car without knowing what the parts are and what they do. Similarly, you need to be familiar with a number of basic concepts in order for the rest of this module to make any sense. Let's look at them now.

If someone approached you on the street and asked you to define the web, how would you do it? From a user's point of view, you would probably talk about the vast amount of information available. You might wax poetic about the plethora of interactive applications, multimedia experiences, and virtual libraries, all accessible through a common interface. You might say all these things, and we wouldn't necessarily disagree with you. In a real sense, the web is made up of information, pure and simple.

From a more technical point of view, however, you can make a strong case for a much more concise definition of what makes up the web. Simply put, the web can be considered a transport mechanism for the information it contains. The same text can be printed in a book or it can be made available through your browser, but you probably wouldn't consider the paper copy to be part of the web. From this point of view, the definition of the web really boils down to the protocol browsers and servers use to communicate: the Hypertext Transfer Protocol (HTTP).

HTTP made its debut in 1990, inside the first web servers and browsers. Since then, the protocol has undergone several major revisions, but it is still recognizably similar to the original.

The HTTP protocol is transaction-oriented. Clients make requests and servers send responses. The protocol is stateless, in that once a server responds to a request, it generally forgets all about it. If the client makes another request some time later, there's no automatic way for the server to associate the client with any particular session.

The format of a transaction is simple. There are two parts: the client's request and the server's response.

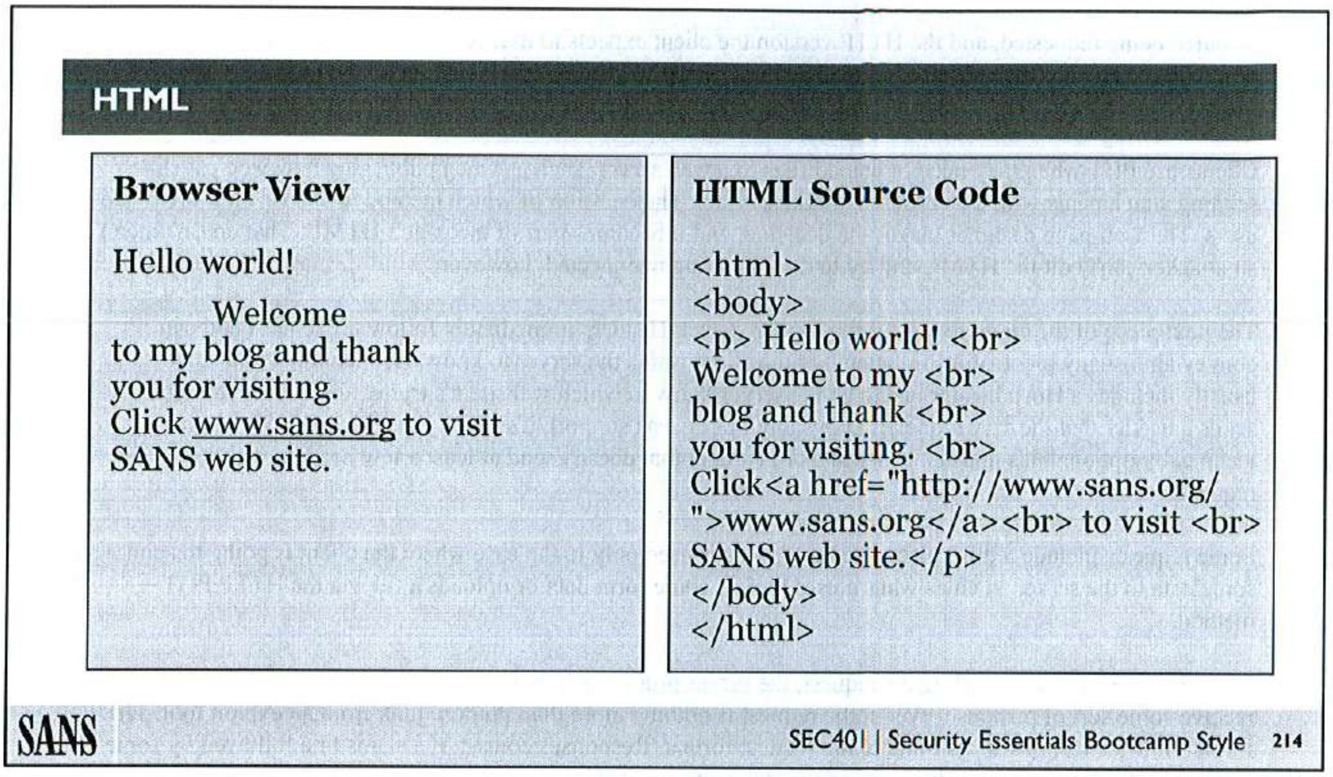
The client starts the conversation with the request. You can see an example of this in the first line of the previous slide. The request is a one-line string that includes the method (for example, the type of request), the resource being requested, and the HTTP version the client expects to use. In the current version of the protocol, the client is also required to send a Host: header to specify at which domain the request is aimed. This allows a single web server on a single IP address to process requests for multiple domains.

Clients use PUT when they need to upload files to a web server, such as when publishing new web pages or sending attachments with a web-based e-mail service. The decision of which method to use isn't left up to end users. The web-page designer makes the decision, and it becomes part of the page's HTML. That doesn't mean an attacker can't edit the HTML and try to do something unexpected, however.

The next piece of the client request is the header stanza. Headers immediately follow the request and can convey almost any piece of information that the client wants the server to know. As mentioned, the client usually includes a Host: header just to let the server know to which website it's trying to connect. If the client sends a header that the server doesn't know about or doesn't support, it's usually just ignored. Although technically optional, it's unusual to see an HTTP client that doesn't send at least a few headers with its requests.

Some requests include a third piece: the body. This is used only in the case where the client is going to send some data to the server, such as when a user POSTs some form data or uploads a file via the HTTP PUT method.

After the client finishes sending the request, the server processes it and sends back a response. All requests receive some sort of response, even if the request is nothing more than random junk from an exploit tool. The format of these responses is similar to the request format. Responses consist of a status line followed by some header lines and, usually, the body, which contains the requested resource.



HTML

Browser View

Hello world!
Welcome
to my blog and thank
you for visiting.
Click www.sans.org to visit
SANS web site.

HTML Source Code

```
<html>
<body>
<p> Hello world! <br>
Welcome to my <br>
blog and thank <br>
you for visiting. <br>
Click<a href="http://www.sans.org/
">www.sans.org</a><br> to visit <br>
SANS web site.</p>
</body>
</html>
```

SANS SEC401 | Security Essentials Bootcamp Style 214

The left side of this slide is a basic example of what text in a browser looks like. The right side shows the same content as source code. Markup tags are used to delineate and format text as well as hypertext links to other web-based resources. Although most code is more complex than this example, it still helps you understand how the page is generated.

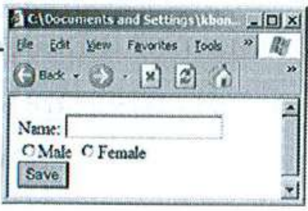
HTML was created by Tim Berners-Lee, whose parents met while working on the commercialization of the first electronic computer (with RAM) at the University of Manchester in 1948. Tim's main purpose for HTML was to allow for standard formatting of documents and to facilitate easy editing and uploading of web-based documents for the purposes of collaboration.

It is important to note that with HTML, what the user sees and what really happens can be two different things. For example, even though a link in a browser is showing a specific domain name, when a user clicks on the link, the actual site they are directed to could be a different, potentially malicious site. Therefore it is always very important to be careful before you click on any link.

HTML Forms

- Forms allow user input to be entered into a web page
- Hidden fields obscure data but typically do not provide security
- Form elements and data can be manipulated by code
- POST action sends form data in HTTP headers
- GET action posts form data appended with URL query string
  - GET might disclose data in "referrer" calls to other sites
  - GET data is easy to manipulate in address bar

```
<FORM action="http://example.org/bin/adduser" method="post">
Name: <INPUT type="text" id="name" MAXLENGTH="8"><BR>
<INPUT type="radio" name="gender" value="Male">Male
<INPUT type="radio" name="gender" value="Female">Female <BR>
<INPUT type="hidden" name="AddDate"
<INPUT type="submit" value="Save">
  </FORM>
```



Name:

○ Male ○ Female

Save

The most basic and popular method for a web application to take user input is through HTML forms. An HTML form is a section of a document containing special HTML elements called "form controls" (checkboxes, radio buttons, menus, and so on). Users generally complete a form by entering text, selecting menu items, or modifying other controls.

Hidden form elements allow the developer to include information in the form without having it displayed on the web page. This is useful in carrying information from one form to the next in applications that span multiple pages. Hidden form data is relatively easy for a user to view and manipulate. Although it can make the web application less cluttered and more user-friendly, hidden form elements do not typically increase the security of the web application.

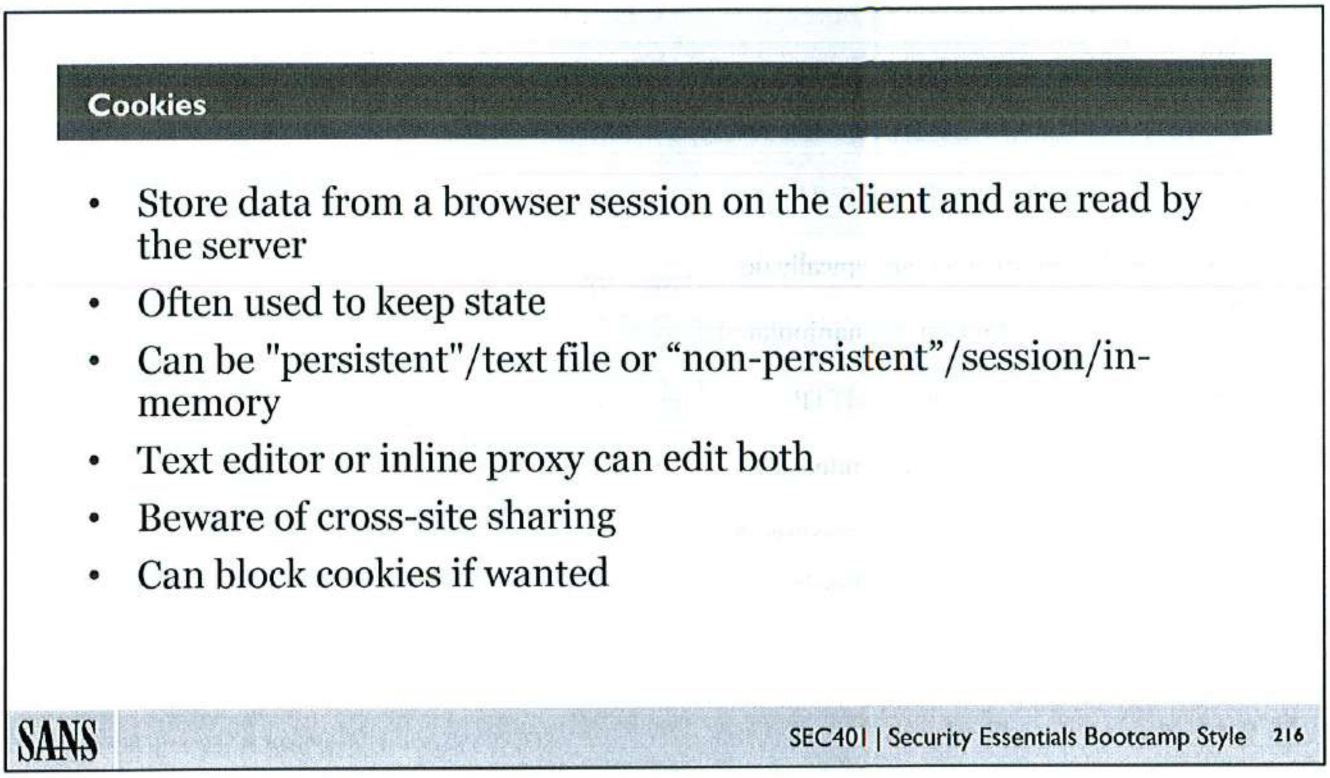
Dynamic HTML, JavaScript, and other client-side scripting tools allow form data and the form elements themselves to be manipulated by code being executed within the user's browser. This gives developers the flexibility to make the web page user-friendly. For example, the developer can have the application auto-fill the shipping address if the user clicks same as billing address, or even create a new text box form element for a gift message if the user indicates that the order is a gift. If an attacker can make changes to the code that handles form input, the attacker then can manipulate the form data before it is submitted or even change the destination server that the form sends its data to.

Depending on the browser configuration and the context of the web page, form variables and other web-page data might also be accessible by other browser windows, frames, or web pages.

When an HTML form is submitted, the form data is sent to the web server using one of two actions: GET or POST. With the GET action, the form data is appended to the URL query, whereas with the POST action, the form data is sent within the HTTP headers.

References

1. *RFC1738: URL Specification*, CGI Specification: http://www.w3.org/CGI/
2. http://www.w3.org/TR/html4/interact/forms.html



Cookies

- Store data from a browser session on the client and are read by the server
- Often used to keep state
- Can be "persistent"/text file or “non-persistent”/session/in-memory
- Text editor or inline proxy can edit both
- Beware of cross-site sharing
- Can block cookies if wanted

SANS SEC401 | Security Essentials Bootcamp Style 216

HTTP is a stateless protocol. That is, servers process requests and typically forget about them when they are done. If a client requests a series of web pages, there's no built-in association on the server side. This causes problems for session-oriented applications, where it's important to keep track of what the user is doing now and what they have done in the past. It would be impossible to build even a simple shopping cart system without some way of associating a user's past actions (adding items to the cart) with her current transaction (checking out).

Fortunately, web designers ran into this problem quite some time ago. They started by adding state information to the URLs themselves. In other words, a typical shopping cart system might have had ugly, complex URLs, such as the following:

http://www.sample.org/shopping/cart.cgi?acctno=182727338363&itemno1=12877&itemno2=92762&itemno3=89272.

They decided there must be a better way to save state information than to embed it in the URL, so they came up with the concept of cookies.

In web terms, a cookie is a named piece of data created by a web server and stored at the web browser. Both the name and the contents are chosen by the application and can be almost anything the programmer wants.

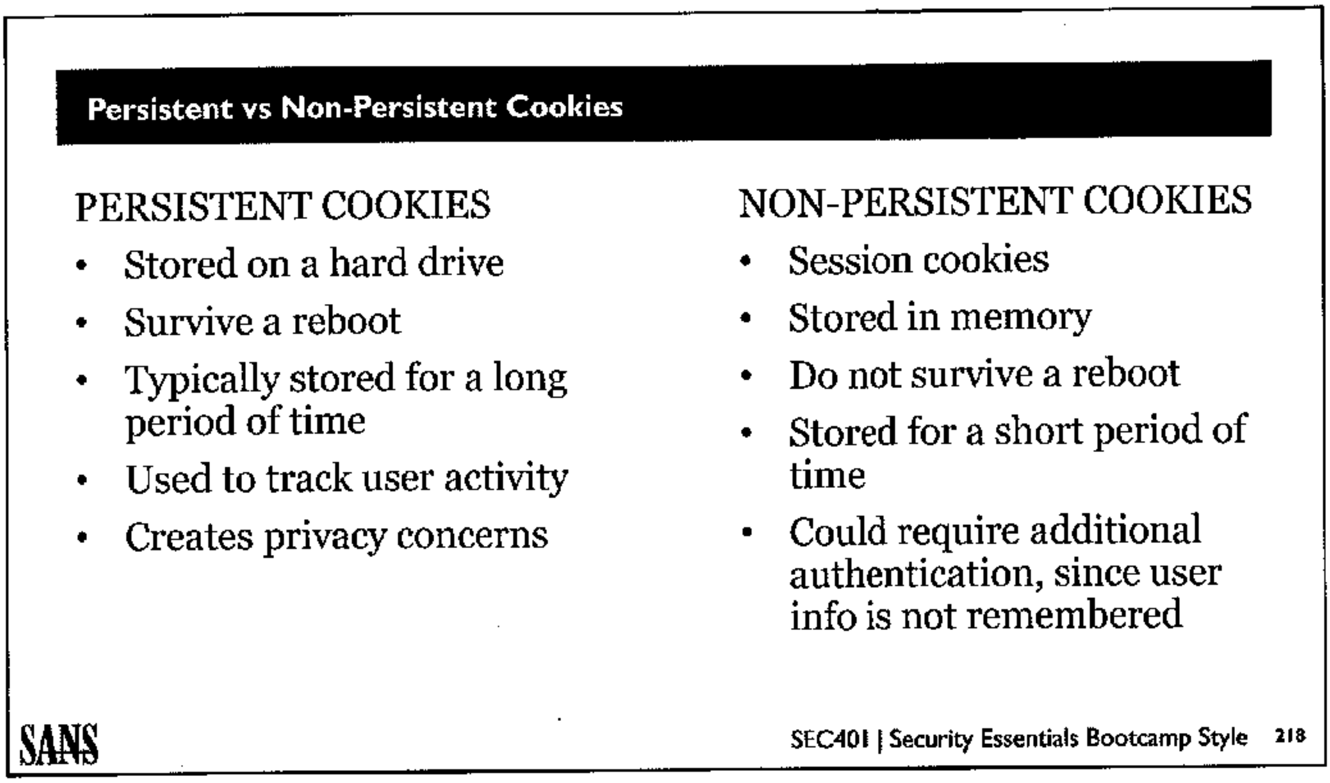
To set a cookie, a server adds the Set-Cookie header to one of its responses. After receiving the cookie, the web browser places it in a cookie header and sends it back with all subsequent requests to that server. The application processes the cookie just like it processes the other user-supplied data. Cookies can contain virtually anything, but they're most commonly used to keep track of user authentication and application session state.

Cookies do a great job of solving the state problem posed by HTTP, but as useful as they are, not everyone is a fan. Some see significant privacy risks associated with cookies, but at least a few of these concerns are attributable more to lack of understanding than the actual possibility for abuse.

Some people object to cookies because they mistakenly believe that they somehow magically take information from your computer and spread it around the Internet. Allow us to assure you that this is simply not so. To exist, a cookie must have been set by a web server and can be sent back only to that same web server (or at most, to other web servers within the same domain).

Furthermore, the web server must specify the contents of the cookie at the time it is created. It can't go snooping around your hard drive to find information it wants. If it has the information to place in the cookie, it had to get it from you in the first place. In other words, you had to have already provided this information to the web server (or to the company operating the web server). There's no way for the site to know your telephone number, for example, unless you've already given it to them. Thus, cookies can't violate your privacy because they simply contain information already known to the site.

Many people are also wary of cookies simply on the basis that you never know what information is stored in them. End users typically never see the cookies their browser accepts, and have no idea what they contain. If a site places sensitive information in a cookie, such as a credit card number or PIN, it could be vulnerable to eavesdropping as it is sent to the server with each request. Of course, leaving this information in the clear would be extremely irresponsible. Most websites encrypt the contents of these sorts of cookies, so recovering the sensitive data is much more difficult. Further, a server can set the optional secure flag on any cookie, which notifies the browser that it is only to send that cookie along with requests protected by SSL. That makes eavesdropping much more difficult.



Persistent vs Non-Persistent Cookies

PERSISTENT COOKIES

- Stored on a hard drive
- Survive a reboot
- Typically stored for a long period of time
- Used to track user activity
- Creates privacy concerns

NON-PERSISTENT COOKIES

- Session cookies
- Stored in memory
- Do not survive a reboot
- Stored for a short period of time
- Could require additional authentication, since user info is not remembered

SANS

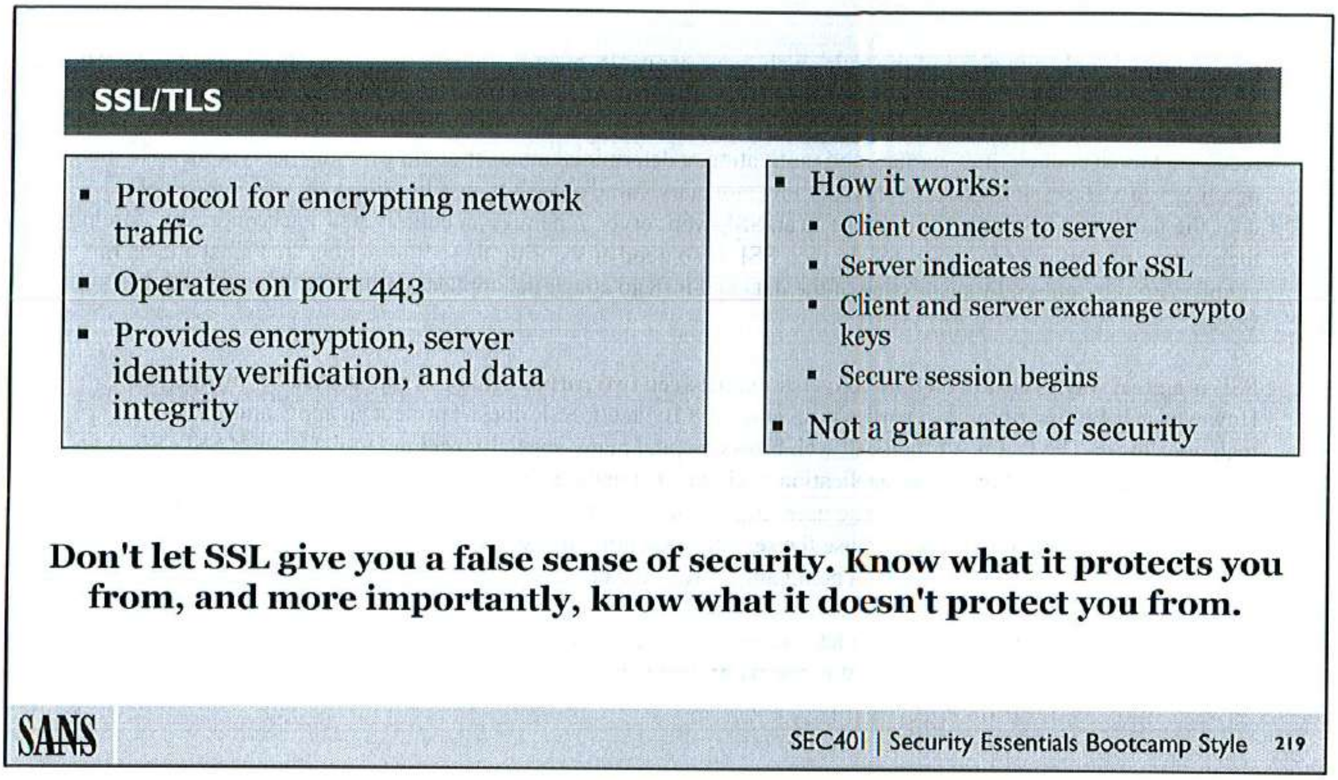
SEC401 | Security Essentials Bootcamp Style 218

So far, we've been talking about cookies as though they were all the same. There are actually two types of cookies you should know about: persistent cookies and session cookies. They both do the same things and act the same way; the difference is in how they're stored.

Most cookies are *persistent cookies*. That is, when a browser receives them, it stores them in a text file on the disk. When the browser exits, the cookies are still in the file, so the next time the browser starts up again, it can load them back into memory and they'll still be active. Persistent cookies have expiration dates, after which time the browser will delete them. Some sites get around this restriction by setting the expiration date to be years (sometimes decades) in the future, effectively causing the cookies to hang around indefinitely. These days, most browsers offer some way to view the cookies. A few minutes spent browsing through them can be quite an eye-opener. Of course, users can edit their own cookies, which might help them to assume the identity of another user or falsify other information, such as the price of a product.

The other type of cookie is the *session cookie*, sometimes called a "non-persistent cookie." As the name implies, session cookies are good only during the current browser session. They are usually stored only in memory and, when the browser exits, these cookies are lost forever. As you might guess, session cookies are good for applications that track their own state, especially if users might be accessing them from a shared computer (in a public library, for example). Although it takes extra effort on a user's part to ensure that their persistent cookies are deleted when they are finished with an application, discarding session cookies is much more convenient. After ending a session, simply closing the browser destroys them and renders them inaccessible to the next user of that computer.

In memory, cookies can still be edited by either a user or man in the middle by using an application that sits between the client computer and the web server, commonly called a *proxy*. An example of a web proxy is Paros (http://www.parosproxy.org/index.shtml).



SSL/TLS

- Protocol for encrypting network traffic
- Operates on port 443
- Provides encryption, server identity verification, and data integrity

- How it works:
  - Client connects to server
  - Server indicates need for SSL
  - Client and server exchange crypto keys
  - Secure session begins
- Not a guarantee of security

Don't let SSL give you a false sense of security. Know what it protects you from, and more importantly, know what it doesn't protect you from.

SANS

SEC401 | Security Essentials Bootcamp Style 219

Probably the most basic requirement for security on the web is for confidential communications. That is, third parties should not be able to eavesdrop on the conversation between browsers and servers. That's where the Secure Sockets Layer (SSL, also called TLS, or Transport Layer Security) comes in. SSL is a protocol that provides an encrypted tunnel between two SSL-aware applications. It's the de facto standard for secured communication and virtually all web browsers and HTTP servers support SSL, at least as an option. HTTP traffic over SSL uses port 443 by default, although this is subject to change by the server administrator.

Most people consider only SSL's encryption function; however, SSL performs three important functions for web applications:

- **Encryption**: Protect the confidentiality of data as it passes over the network.
- **Server identity verification**: Basically the name on the web server SSL certificate needs to exactly match the domain name in the browser's address bar. This confirms to the users that they are talking to the server to which they think they are talking.
- **Data integrity**: SSL ensures that the data sent between the two endpoints arrives whole and intact.

SSL connections start with a handshake phase to negotiate the type and strength of encryption to use (which could technically include no encryption, although uncommon in practice, if configured that way by the server administrator). The SSL specification describes several acceptable encryption algorithms. However, not every SSL client or server is required to support them all. The negotiation phase ensures that the best algorithm available to both sides is chosen.

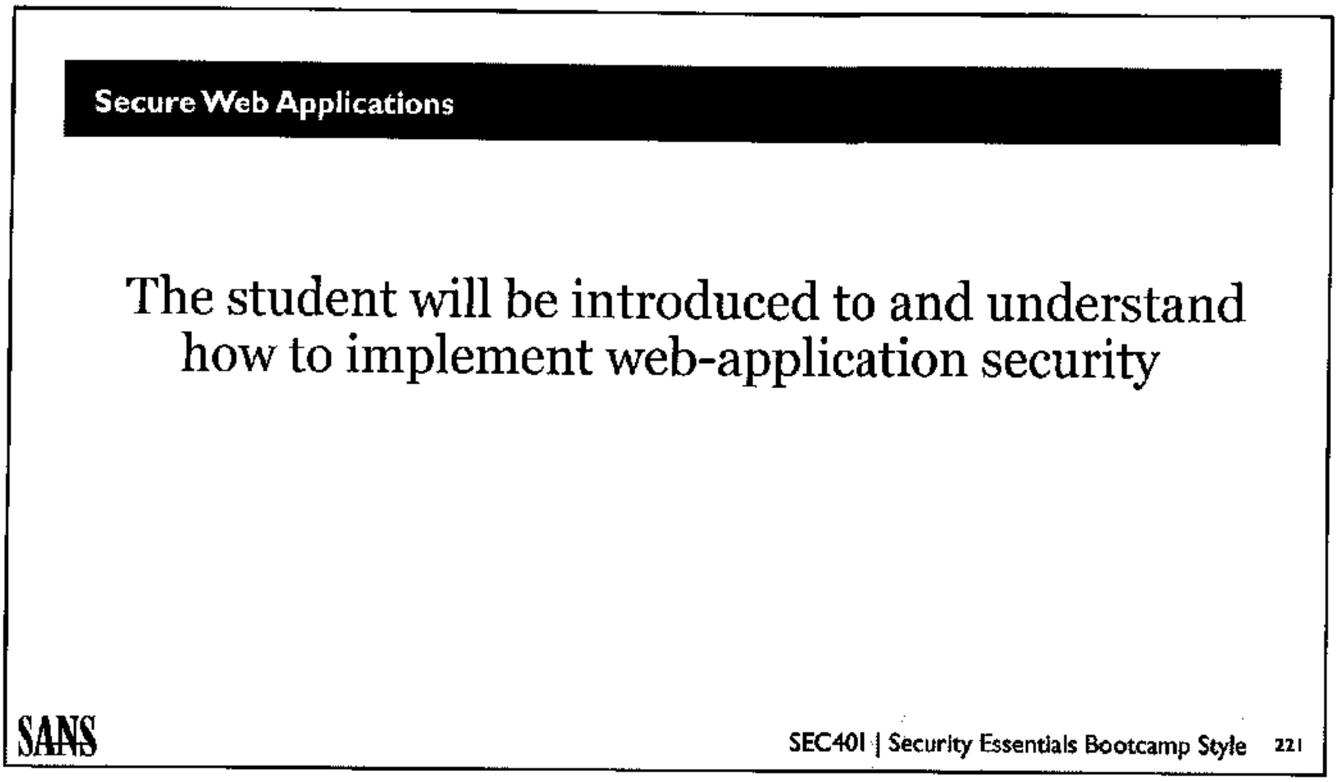
SSL encryption keys are symmetric. Each side shares a randomly generated secret key used to encrypt and decrypt the transmission. These secret keys are negotiated during the TLS/SSL setup, using algorithms such as RSA or Diffie-Hellman key exchange. The key exchanges allow two parties to create a shared secret key over a public communication channel.

During the SSL initialization, the server presents a public key certificate to the client, allowing the user's software to verify the server's identity. Clients can present their own certificate as well, which allows the server to verify the user's identity.

SSL security is only one piece of the entire web security puzzle, even though it's the one users interact with most often. The closed lock icon on most web browsers might give you a warm fuzzy feeling that your data is being encrypted as it's sent over the network, but the real question is what the receiving site does with it after it's received. Given enough time, money, and motivation, a determined attacker could certainly decrypt the contents of any given SSL session; but the relatively low monetary value of web transactions makes this impractical, as does the fact that every request you make to an SSL web server generates an entirely new encryption key, forcing the attacker to start over from scratch. In fact, SSL is so good at securing these transactions that most attackers simply skip the front-end transmission of the data and instead attack the application itself or the backend data store.

SSL is a great way to ensure that the conversation between two parties cannot be understood by anyone else. However, what if one of the two parties is an attacker? By itself, SSL doesn't protect an application from malicious users. The fact is, virtually all web browsers (and many specialty hacking tools) support SSL. So, if an attacker wants to try to break your application, SSL won't stop them. They can generate an encrypted, secure session themselves, just like a legitimate user, and still use it to do bad things to your application. In fact, they might actually prefer this method because the session encryption makes their actions essentially invisible to intrusion detection systems that rely on being able to read the contents of network sessions.

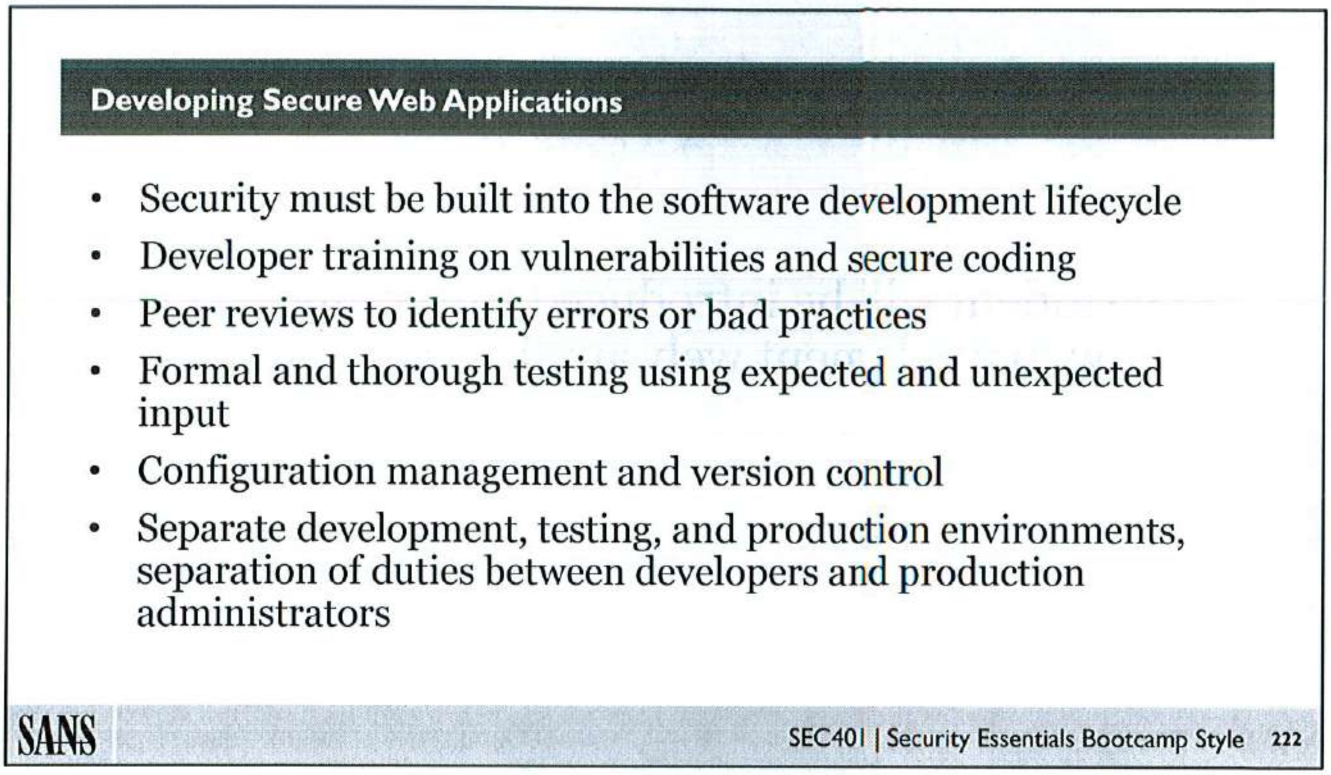
The lesson of this module is this: Don't let SSL give you a false sense of security. Know what it protects you from, and more importantly, know what it doesn't protect you from.



Secure Web Applications

The student will be introduced to and understand how to implement web-application security

This page intentionally left blank.



Developing Secure Web Applications

- Security must be built into the software development lifecycle
- Developer training on vulnerabilities and secure coding
- Peer reviews to identify errors or bad practices
- Formal and thorough testing using expected and unexpected input
- Configuration management and version control
- Separate development, testing, and production environments, separation of duties between developers and production administrators

One important way to prevent the introduction of vulnerabilities into a web application code base is to ensure that the organization's development, testing, and deployment process includes the following components.

A key way to reduce the number of vulnerabilities in web applications is to teach the developers about web-application attacks, common vulnerabilities and errors, and best practices to avoid those errors. Many concepts, such as input validation and session tracking, are common to most web applications. Each language or development platform also has attributes that can make developers more susceptible to certain types of mistakes, so it can be beneficial for developers to get training specifically on the common errors and best practices related to the chosen development platform.

A peer review is the manual examination of source code by a group of the author's peers. Peer reviews are effective for detecting defects or other errors and can be especially effective for finding security issues if reviewers are trained in common mistakes that lead to vulnerabilities. Reviews are an important supplement to testing because they can find errors earlier and can find errors that might not show up during debugging/testing.

Software testing is the process of executing a program or system with the intent of finding errors. A formal and complete test plan is necessary to ensure that testing is thorough and covers all possible events or scenarios that might occur when the program is placed in production. Many test plans generally deal with how the application will respond to expected input. With web-facing applications, it is important to also test how the application will respond to unexpected or invalid input. The test plan/test record should include each test performed, the expected result, and the actual result for each iteration of testing that is completed. Good test procedures often include a combination of manual and automated testing activities.

In addition to functionality testing, the development process should include application performance or load testing. This helps to demonstrate that the architecture and resources provided are sufficient for the web application's needs. This also helps to determine what thresholds exist and what risks might be present for Denial of Service attacks. In performance testing, careful attention should be paid to the error messages and other abnormal behaviors of the system under an excessive load to ensure they don't disclose sensitive information about the system or indicate the creation of other vulnerabilities.

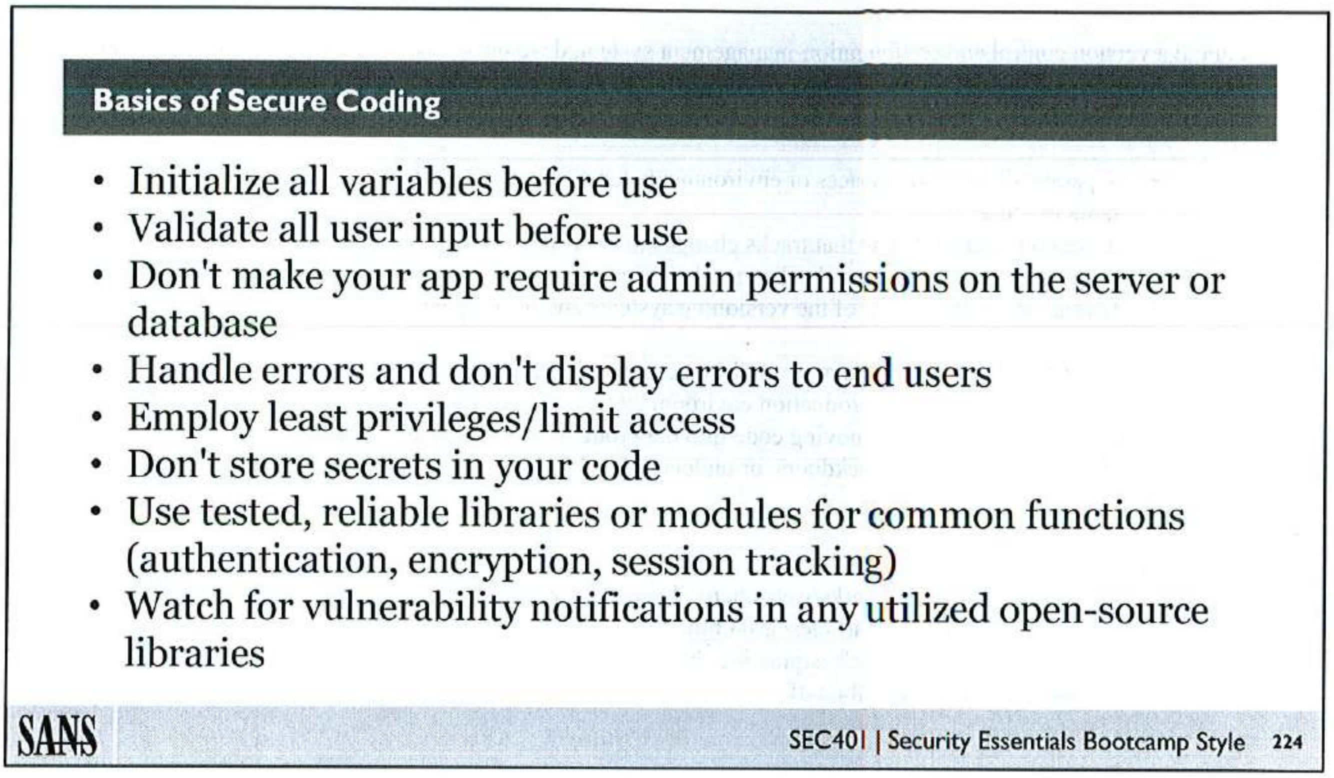
Without a version control and configuration-management system, developers can be working on older versions of code or can be making conflicting changes to code or systems. Some important components of configuration management include

- Separate, distinct workspaces or environments for different developers and different releases of the same product
- A version-control system that tracks changes to the code, allows developers to check in/check out components, and ensures code changes do not overlap
- Formal processes for use of the versioning systems and development environments

It is generally a best practice to have separate environments for development, testing, and production. Changes should never be made directly to the production environment, and where possible, a team separate from the developers should be responsible for moving code into the production environment. This helps ensure that processes are followed and tweaks, backdoors, or undocumented fixes aren't placed into the production environment and overlooked.

References

1. http://www.ibm.com/developerworks/websphere/library/techarticles/0306_perks/perks2.html
2. http://www.stevemcconnell.com/articles/art04.htm
3. http://www.perforce.com/perforce/bestpractices.html
4. http://www.auditnet.org/docs/CMbp.pdf



Basics of Secure Coding

- Initialize all variables before use
- Validate all user input before use
- Don't make your app require admin permissions on the server or database
- Handle errors and don't display errors to end users
- Employ least privileges/limit access
- Don't store secrets in your code
- Use tested, reliable libraries or modules for common functions (authentication, encryption, session tracking)
- Watch for vulnerability notifications in any utilized open-source libraries

Here are some tips that should help you create more secure code.

Different languages treat variable creation and initialization in different ways; however, in general, you should purposefully create and initialize each variable you use. Some web languages initialize variables from form or other input if the initialization is left up to the program. As an example, if a PHP programmer uses the variable $counter without initializing it, he would assume $counter would start counting at zero. However, under certain configurations, if a user added ?counter=12 to the URL query string in the browser, $counter would start at 12.

If the language supports public and private functions and variables, make functions and variables private. This means they can't be accessed or manipulated from areas outside of their intended purpose.

Before acting on any input provided from users, ensure that the input is valid and filter out any potentially harmful characters or strings.

Don't design your application such that it would need to do things that require administrative privileges, such as create queries or procedures on the database on the fly. Sometimes this might be an easier, faster, or more elegant solution, but you should develop a design with security in mind.

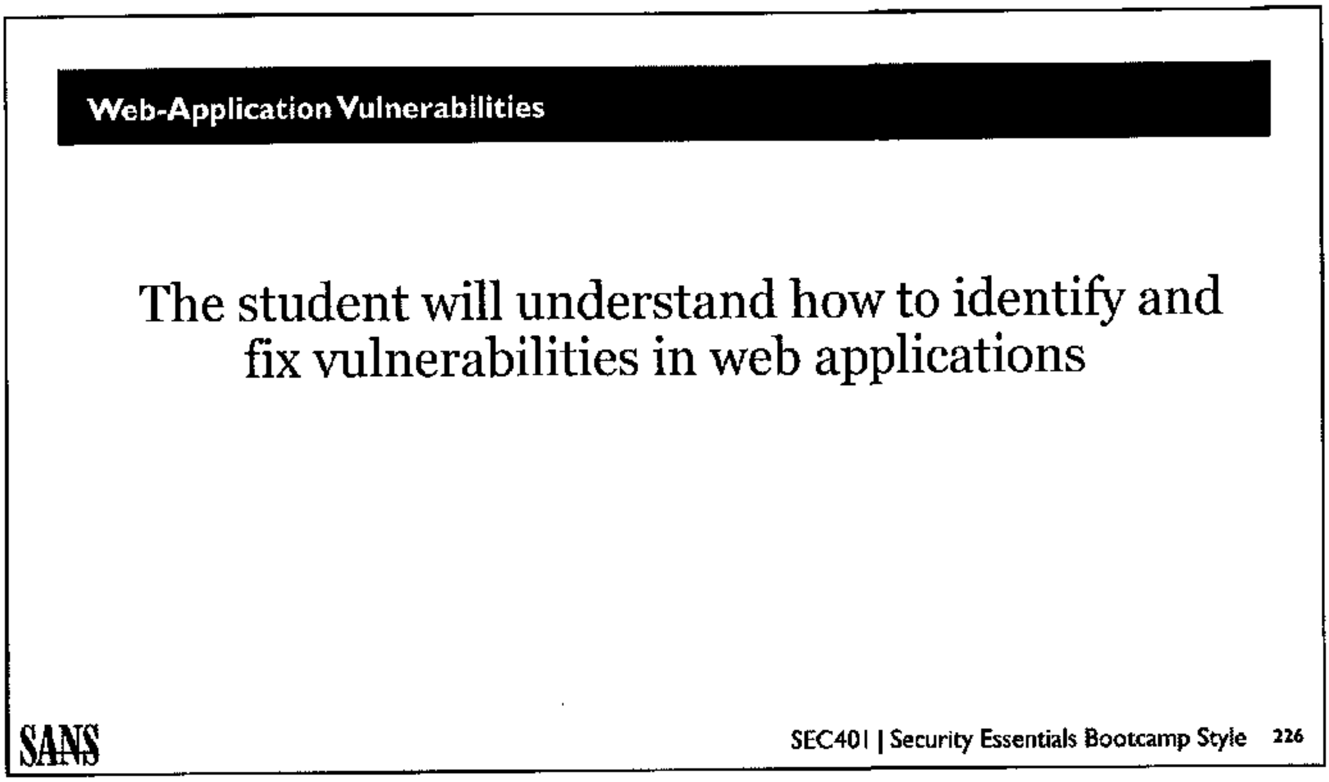
Check for error conditions when returning from all functions, and handle the errors gracefully. Do this even for situations in your code that should never happen. Handling errors gracefully involves stopping processing in a way that doesn't impact other users or corrupt data, logging a detailed error event to a log file, and displaying a vague, generic error message to the user. Ensure that any debugging that was enabled for your development and test environments is disabled in production.

Employ the principle of least privileges—only grant an account access to the resources it needs. This should be done within the context of your website—if appropriate, your website should have an authentication and access control mechanism that limits where users can go. This should also be done within the context of your web-development framework and the supporting systems—your web server account and related accounts should have access only to the resources they need to make the web application function.

Don't build secrets into your code thinking that users won't find them. Secrets can include things like backdoor passwords or alternative access methods, credentials for database or application server authentication, or encryption keys. As a corollary, don't rely on obfuscation (making codes more difficult to follow or understand) for security. Attackers who have the time, talents, and tools find the secrets.

Don't try to re-invent the wheel when it comes to components that have a security impact. This is especially true for encryption algorithms and session-tracking mechanisms. It is difficult to build these technologies without any vulnerabilities; if you use an off-the-shelf library that is well maintained and has been tested over time, you can have more confidence in the security of the code. Never create your own encryption code; homegrown encryption code is typically fragile and easy to break.

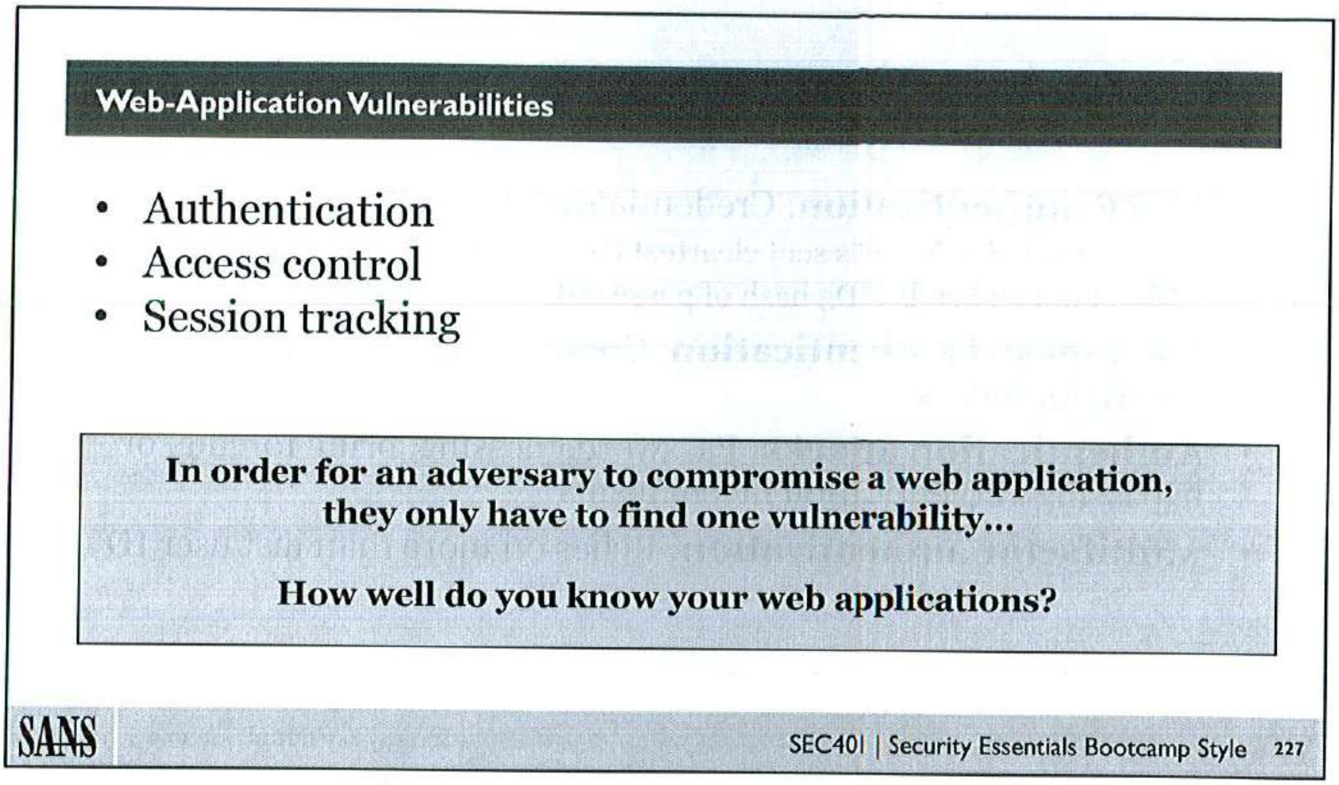
Off-the-shelf web parts, such as bulletin boards, e-mail interfaces, and shopping carts, commonly have vulnerabilities disclosed. When using these public libraries or web parts, make sure you maintain and patch those components in the same way you'd patch the web server. Watch for notifications of new vulnerabilities and new patches, and apply them when they become available.



Web-Application Vulnerabilities

The student will understand how to identify and fix vulnerabilities in web applications

This page intentionally left blank.



Web-Application Vulnerabilities

- Authentication
- Access control
- Session tracking

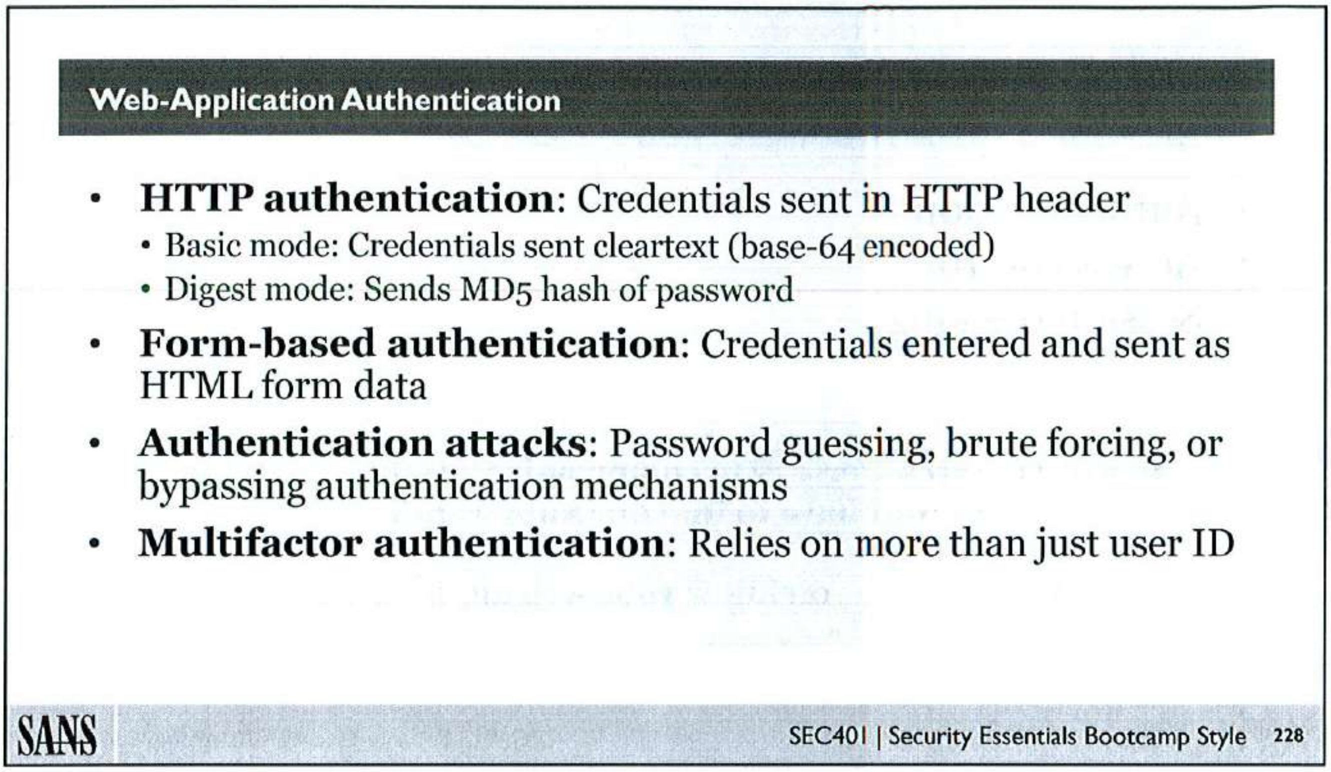
In order for an adversary to compromise a web application, they only have to find one vulnerability...

How well do you know your web applications?

SANS

SEC401 | Security Essentials Bootcamp Style 227

Now that we've covered the basics of web communications and application development, we are going to go into some of the common vulnerabilities in web applications and how to protect against them. The areas we cover include authentication, access control, and session tracking.



Web-Application Authentication

- **HTTP authentication**: Credentials sent in HTTP header
  - Basic mode: Credentials sent cleartext (base-64 encoded)
  - Digest mode: Sends MD5 hash of password
- **Form-based authentication**: Credentials entered and sent as HTML form data
- **Authentication attacks**: Password guessing, brute forcing, or bypassing authentication mechanisms
- **Multifactor authentication**: Relies on more than just user ID

Most web applications require authentication. To authenticate a user is to determine that user's identity with an appropriate level of confidence. The two most commonly seen web-authentication methods are HTTP authentication and HTML form-based authentication.

With HTTP authentication, the user's authentication credentials are sent within the HTTP headers. The two native HTTP authentication schemes available are basic authentication and digest authentication. The process for basic authentication is as follows:

1. The client sends a standard HTTP request to load a page:
   GET /documents/JulyReport.html HTTP/1.1
2. The server responds with HTTP 401 Authorization Required:
   HTTP/1.1 401 Authorization RequiredDate: Sun, 09 Dec 2016 19:35:01 GMTWWW-Authenticate: Basic realm="Users" ...
3. At this point, the browser displays a pop-up dialog prompting for a user ID and password.
4. After the user enters the password, the client sends the same request, but this time, the entered user ID and password are included in the headers. The user ID and password are base-64 encoded. This encoding is easily reversible; base-64 encoding is not encryption and provides no protection:
   GET /documents/JulyReport.html HTTP/1.1
   Authorization: Basic n3786sd9maGY5OWm8dcT=
5. After the web server receives this request, it decodes the base-64 encoded user ID and password and tests whether it is correct for a valid user on the system.

The process for digest authentication is similar; however, instead of simple base-64 encoding, it uses a one-way cryptographic MD5 hash to create a hashed password that is sent within the HTTP headers.

Form-based authentication is using HTML form fields to request the user's authentication credentials. It is common to use the <INPUT TYPE="PASSWORD"> tag for the password input field. This field type obscures the typed characters with asterisks as they are displayed on the screen, and the contents of password fields are not cached or auto-filled when you reload or navigate between screens. The user ID and password are sent clear-text along with any other form data, so a separate mechanism to create a secure channel, such as SSL, is required for secure form-based authentication.

One common attack against web-application authentication is guessing or brute forcing of user accounts and passwords. Many systems and web applications have default admin, test, or demo user accounts that many administrators forget to disable or change. Administrators should take care to remove or change any default accounts or passwords in their applications. Apart from default accounts, attackers often try to guess valid user IDs or passwords for applications. Web applications should give exactly the same response for all authentication errors (invalid user ID, invalid password, account locked, and so on). This prevents attackers from determining valid user IDs by guessing or brute forcing. It is also important to implement an account lockout policy for repeated incorrect password attempts. This makes it difficult for an attacker to guess a password through brute force or dictionary attacks.

Another attack against an authentication mechanism is bypassing the authentication mechanism. A typical user follows a known path through the web application, which usually starts with the login page. An attacker might know or be able to guess the names of other files or folders related to the web application. In this case, the attacker will try to type these addresses into his browser without first authenticating through the login page. To stop this attack, it is important to have all components of a web application test that the user is logged in before allowing access, and it is also important to block users from directly accessing resources, such as function libraries or include files, that support the web application but should never be loaded by users directly.

There are many weaknesses to authentication schemes that rely solely on user IDs and passwords. An attacker might be able to guess or brute-force the user's password, intercept the user's password if they can compromise the network with a sniffer or the user's computer with a keystroke logger, gather the password from the user's account on a different system, or get the user to disclose the password through phishing or other social engineering attacks.

In response to these weaknesses in password-based web authentication schemes, multifactor authentication is gaining popularity for stronger web-based authentication. Multifactor authentication is the use of more than one "factor" to verify a user's identity.

A password is something you know, which serves as a single factor for authentication. Certificate-based web authentication provides a second factor by relying on something you have, namely a client certificate. A client certificate is a digital file with a cryptographic signature that is provided to the user either by the website owner or a trusted third party. Client certificates work very well to validate a user's identity, but they have gained minimal popularity because the distribution, setup, and management of digital client certificates are difficult for both website operators and end users.

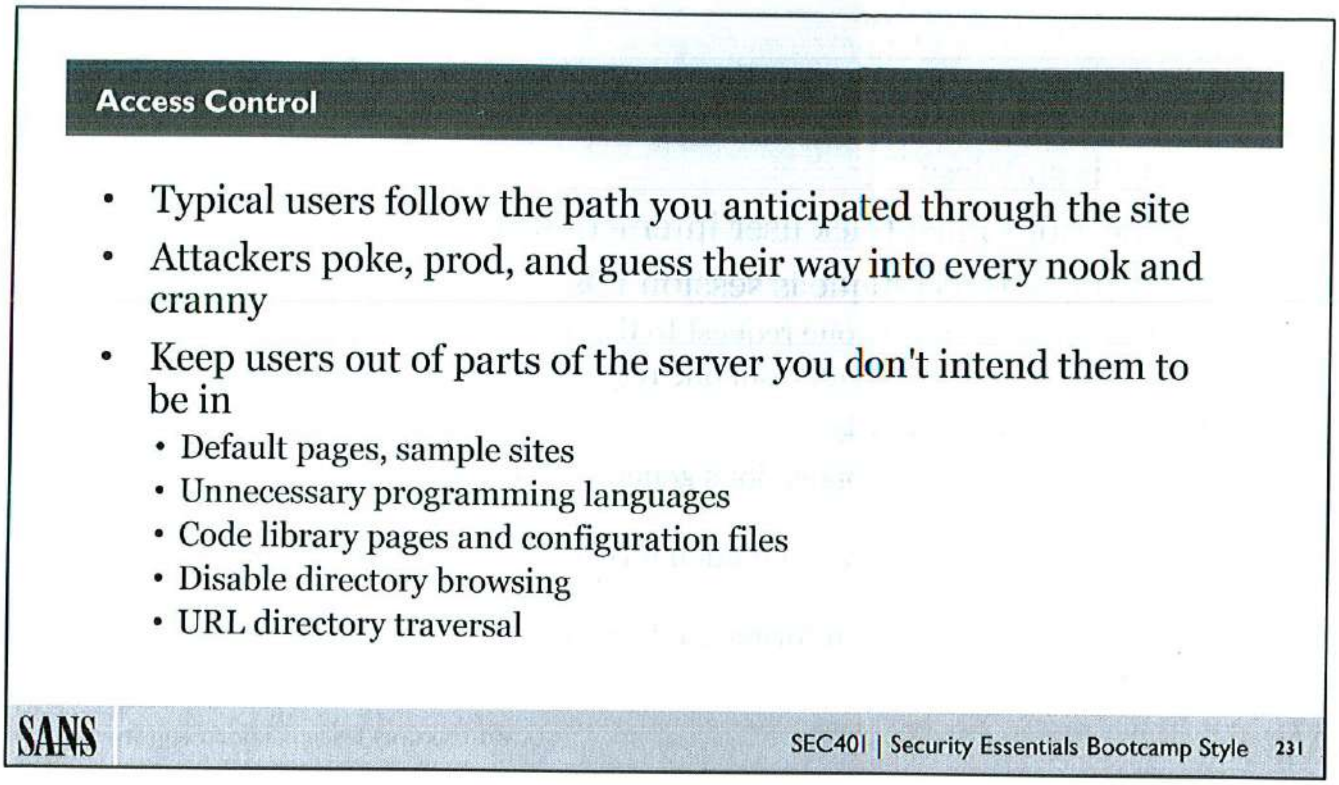
Token-based authentication schemes also rely on something you have: a token. The token is a small device that produces a one-time password for each authentication attempt. Generally, this works by initializing and running the same algorithm on both the web server system and the token, so both systems generate the same one-time password based either on time or sequence of prior passwords. Token authenticators work well because they are easy to use, but they are expensive relative to other mechanisms.

Another mechanism for providing one-time passwords is to provide the user the one-time password out-of-band during each authentication attempt. For example, after the user enters their user ID and password, the system looks up their phone number and sends the one-time password for this login attempt via an SMS text message.

Because of the expense and complexity of these multifactor authentication schemes, many web applications are beginning to rely on the "footprint" of the user's device as a second factor for authentication. Attributes the application can look for to confirm the footprint of the device can include cookies left by the web application during a prior visit, software or other signatures installed to the hard drive, the client IP address, and system and browser configuration. Footprint schemes are easier to implement than other methods, but they are also much easier to fool or break.

Finally, many web applications use challenge questions to confirm a user's identity. These are questions that relate to "favorite pet's name" or "high school mascot" that the user answers when she sets up the account and must provide the same answer later to confirm her identity. Challenge questions in conjunction with passwords are not a strong authentication mechanism because they rely only on things you know instead of other factors.

Few web applications rely on biometrics, due to the cost and compatibility issues with deploying biometric readers (thumbprint or retinal scanners, and so on) to the user base of a web application.



Access Control

- Typical users follow the path you anticipated through the site
- Attackers poke, prod, and guess their way into every nook and cranny
- Keep users out of parts of the server you don't intend them to be in
  - Default pages, sample sites
  - Unnecessary programming languages
  - Code library pages and configuration files
  - Disable directory browsing
  - URL directory traversal

Whether or not users are authenticated, make sure that your web users can't go where they are not supposed to go within the web server. Many web developers consider only the path that a typical user will take through the site. This could be the home page, the login page, and then any page that has a hyperlink from within the application. As a security professional, you need to consider the path the attacker will try to take.

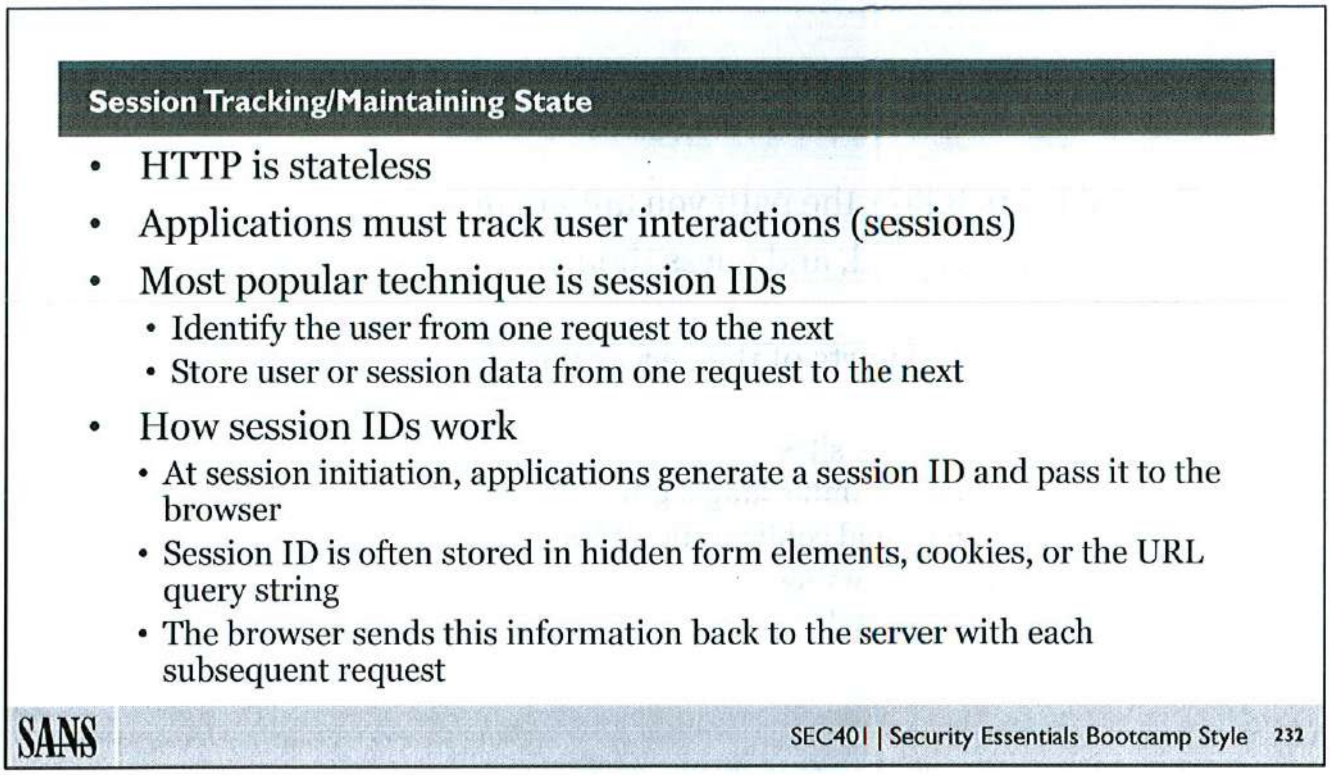
Many web-server applications provide default pages and sample sites.

Another potential weakness in your web server is programming languages that are installed and tied into the web server but are unnecessary.

Aside from the default installed languages and sample code, you also need to prevent web users from accessing custom code libraries and configuration files built along with your website. It is common for web developers to put shared functions and subroutines into separate files that are included or accessed by each page on the website. If these files exist within the folders published by the web server, an attacker might guess at the name and location of the files. You can block users from accessing code libraries and configuration files directly through naming, location, access rights, and other controls on the web server.

Many web servers permit users to browse directories. This means that the user can get a list of the files and folders within the directory of the website. This is generally not something you want to do, because it might allow the user to find code library pages, configuration files, older versions or backups of published pages, and other sensitive information that you did not intend to publish to the Internet. For most web servers, directory browsing can be disabled by implementing a default or index file in each directory, or it can be turned off site-wide within the web server configuration.

URL directory traversal attacks are kind of a combination of flawed access controls and an input attack. With directory traversal, the user exploits vulnerabilities on the web server to access restricted directories, execute commands, and view data outside of the directories meant to be published by the web server.



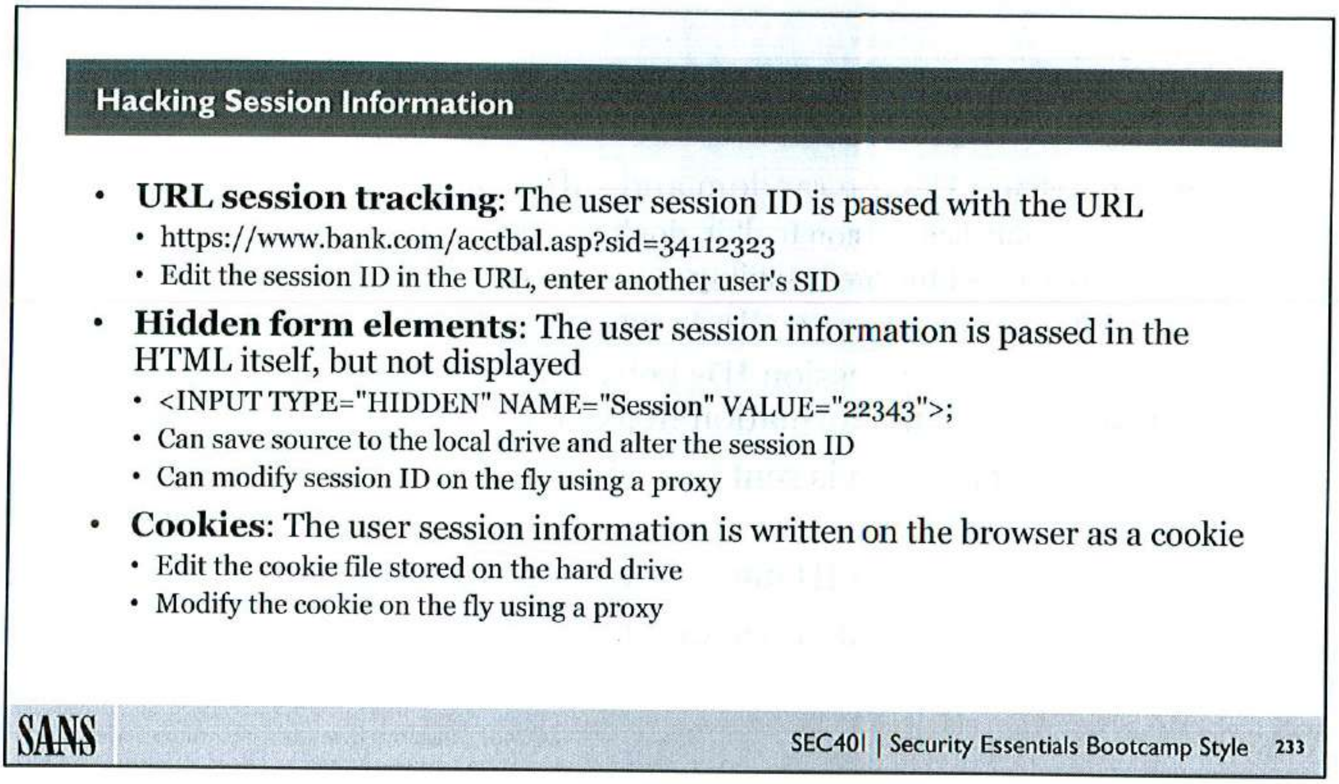
Session Tracking/Maintaining State

- HTTP is stateless
- Applications must track user interactions (sessions)
- Most popular technique is session IDs
  - Identify the user from one request to the next
  - Store user or session data from one request to the next
- How session IDs work
  - At session initiation, applications generate a session ID and pass it to the browser
  - Session ID is often stored in hidden form elements, cookies, or the URL query string
  - The browser sends this information back to the server with each subsequent request

SANS SEC401 | Security Essentials Bootcamp Style 232

HTTP is stateless. This means that, from the web server's perspective, each individual HTTP request and response pair is independent of all the previous and future requests and responses exchanged between the web server and the client's browser. If a web application needs to track a user over a series of web requests (creating a "session"), it needs to handle the tracking of that interaction itself.

The most popular technique for tracking a user through multiple web requests is the use of session IDs. When the user requests the first web page in their session, the server creates a unique identifier, usually a random number or string, and sends it back to the client along with his web request. This session ID is often stored as a hidden form element, part of the URL query string, or in a cookie. These methods cause the browser to send that same session ID back to the web server on all subsequent web requests.



Hacking Session Information

- **URL session tracking**: The user session ID is passed with the URL
  - https://www.bank.com/acctbal.asp?sid=34112323
  - Edit the session ID in the URL, enter another user's SID
- **Hidden form elements**: The user session information is passed in the HTML itself, but not displayed
  - <INPUT TYPE="HIDDEN" NAME="Session" VALUE="22343">;
  - Can save source to the local drive and alter the session ID
  - Can modify session ID on the fly using a proxy
- **Cookies**: The user session information is written on the browser as a cookie
  - Edit the cookie file stored on the hard drive
  - Modify the cookie on the fly using a proxy

SANS SEC401 | Security Essentials Bootcamp Style 233

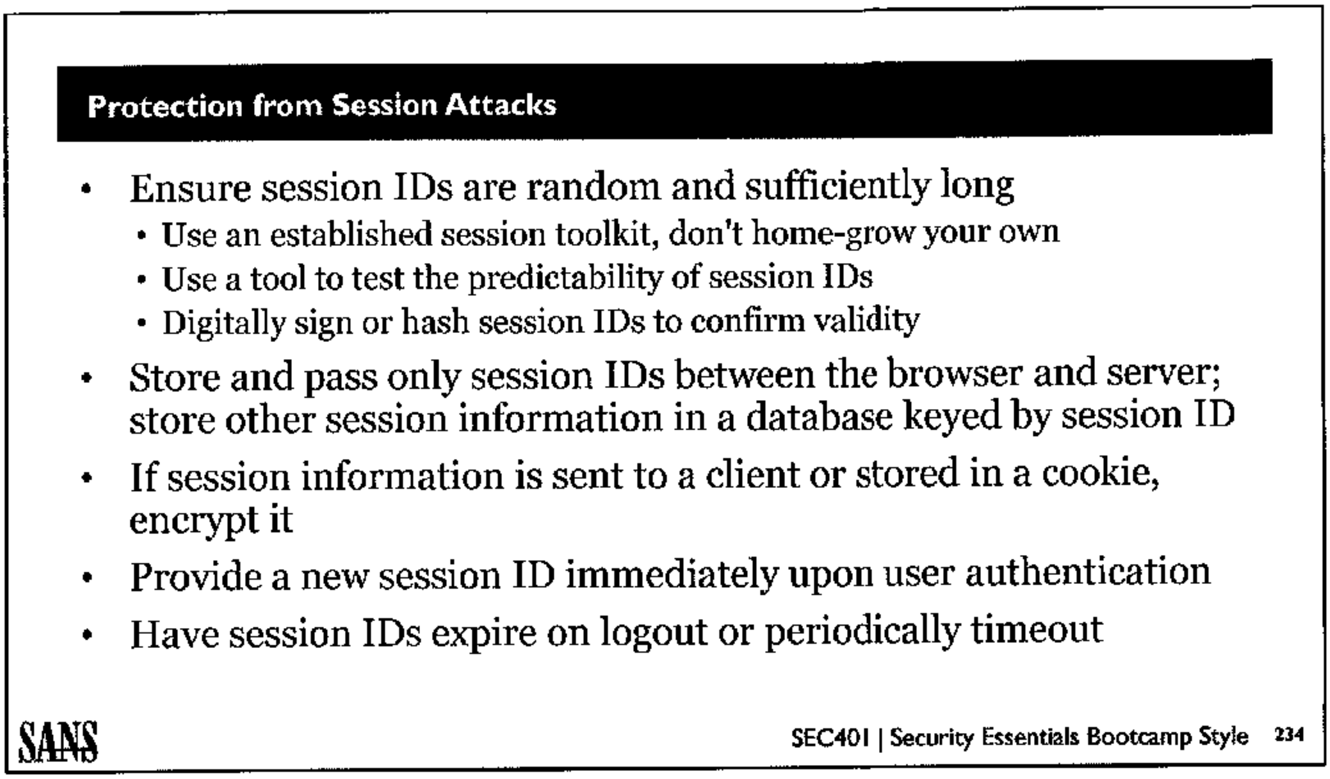
No matter what the delivery mechanism (the URL, hidden fields, or cookies), session state is a prime target for attack. Session attacks can be as simple as convincing the application you logged in as another user or as complex as tricking it into shipping a basketful of items that were never paid for.

The simplest mechanism a web application can use to save session information is to add state information to the URL. You might have seen this many times and not known it. Take a typical shopping cart system URL, such as http://www.sample.org/shopping.cgi?SessionId=2281702037272160283.
This method is the easiest to implement, but it's also the ugliest. The long, unidentifiable number in the URL would make most site designers cringe, and it's also trivial for the user to edit. All they need to do to try to attack this mechanism is to edit the number in their browser's URL bar.

Another common method, only slightly more difficult to implement, is to embed state information in hidden fields in an HTML form. Hidden fields are never displayed to the user, but are otherwise exactly like regular fields. If your application is already reading several field values to get its user input, why not read an extra value or two to get the session information?

Hidden fields are convenient, but aren't hard to alter. Attackers like to see hidden fields and will often use their browser's View Source command to look for them. If they happen to find one they want to change, they can simply save the entire HTML page to their hard drive, edit the field value, and load it back into their browser. If they fill out the other fields and click the 'Log on' button, they send the modified data instead of the original!

The third method of setting state is to use a cookie. Cookies are usually the preferred method of saving state because you have a little more control over them. You choose whether they are session or persistent cookies, and you can set the secure parameter to indicate whether or not they're allowed to be sent over non-SSL encrypted channels. Cookies, however, remain relatively easy for an end user to manipulate using tools such as the Paros or Burp web proxies.



Protection from Session Attacks

- Ensure session IDs are random and sufficiently long
  - Use an established session toolkit, don't home-grow your own
  - Use a tool to test the predictability of session IDs
  - Digitally sign or hash session IDs to confirm validity
- Store and pass only session IDs between the browser and server; store other session information in a database keyed by session ID
- If session information is sent to a client or stored in a cookie, encrypt it
- Provide a new session ID immediately upon user authentication
- Have session IDs expire on logout or periodically timeout

Most session attacks require the attacker to guess another user's session ID. You need to ensure that the session IDs are random, so an attacker cannot identify a pattern or algorithm that would allow him to guess other user's session IDs. You also need to ensure the session IDs are long enough to prevent an attacker from determining other valid session IDs through brute force.

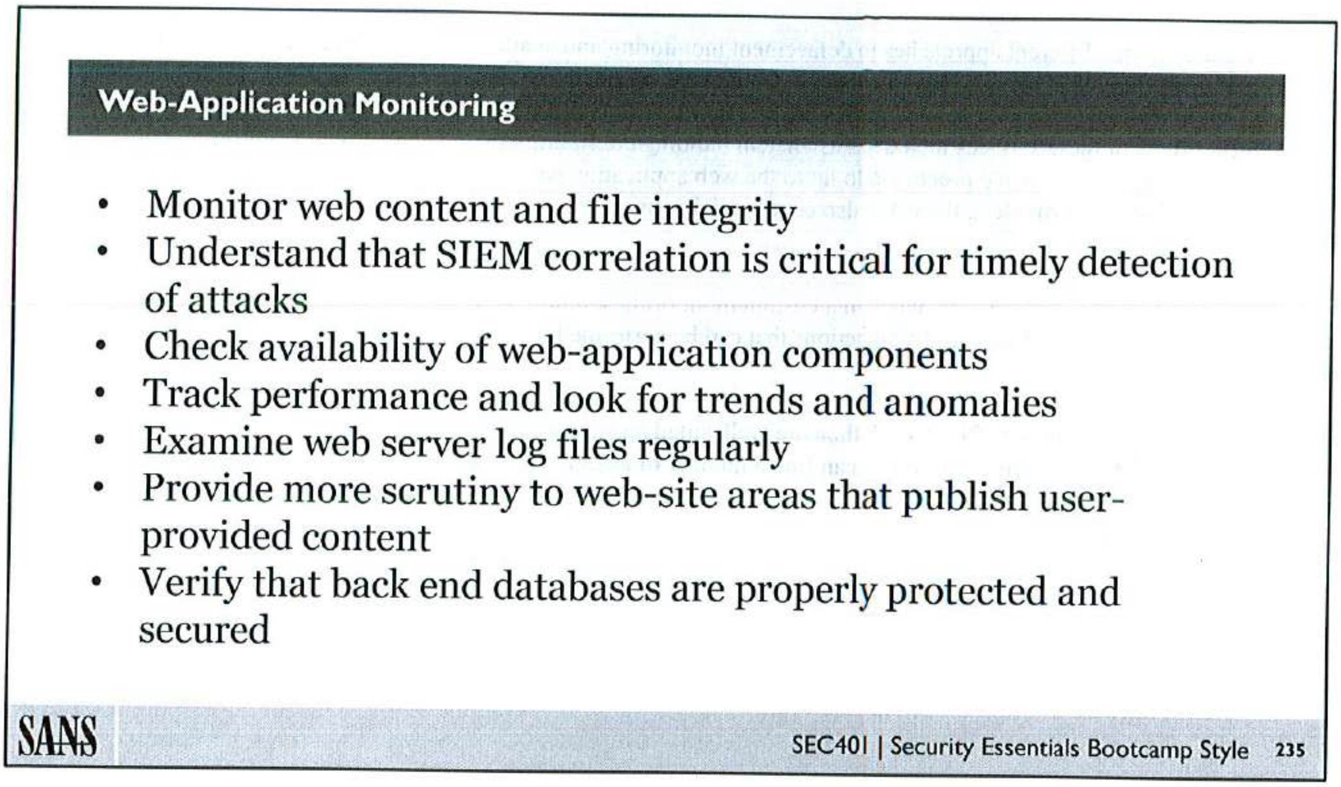
The best way to implement a session-tracking mechanism is to use one already available with your web-development framework. Session tracking has been around for a while, and the session tracking toolkits available with the common and mature web development frameworks today have been scrutinized and strengthened over time. There's no need to reinvent the wheel.

Tools are available that can help you test the predictability of session IDs. Burp, for example, can collect a sample of session identifiers and graph the distribution of session IDs over time. In addition to making session IDs random, you might consider hashing or digitally signing session IDs. Doing this, you can test each session ID received to ensure it is valid before acting upon it.

In a web transaction, developers often find it convenient to carry other session data from one page to the next by storing it in hidden fields or cookies.

For example, the user's account number or shopping-cart contents can be stored in hidden fields or cookie values. Any information sent to or received back from the client could potentially be intercepted or manipulated. It is best to pass only the session ID itself from one request to the next; other session data should be stored on a backend database for the web server. If session information other than the session ID needs to be stored on the client (for example, a user's buying preferences so you can recommend similar products the next time she visits your site), ensure the data is encrypted.

When a user first visits a site anonymously, they generally provide a session ID as the anonymous user. Once that user logs in, do not continue to use the same session ID. The authenticated user should be provided a new session ID. In addition, ensure your session IDs expire in a timely fashion. When a user logs out, their session ID should become invalid both on the client and on the server. If a user leaves the site without logging out, the session ID should expire within a short period of time.



Web-Application Monitoring

- Monitor web content and file integrity
- Understand that SIEM correlation is critical for timely detection of attacks
- Check availability of web-application components
- Track performance and look for trends and anomalies
- Examine web server log files regularly
- Provide more scrutiny to web-site areas that publish user-provided content
- Verify that back end databases are properly protected and secured

SANS SEC401 | Security Essentials Bootcamp Style 235

Web applications that face the Internet come under constant attack. A system administrator often does not know whether these attacks are successful or even if they are occurring without diligent monitoring. Here are some ways to help identify whether an attacker has successfully compromised your web application.

The most basic monitoring to be performed is some mechanism to know whether someone has defaced or made other unauthorized changes to parts of the web application. Basic web-site defacement generally involves vandalizing your website to spread a social or political message or for the hacker to show off and gain credibility among their peers. In a similar attack, the changes to the website could be more subtle. The attacker might change important links or forms to steal information or redirect the user to a malicious site. In another similar attack, the perpetrator might hide new web pages on your server. This is a common practice for phishing attacks, where the phisher uses your web server to host the web pages that are the link destinations for phishing e-mails.

You can detect defacement by monitoring the content produced by the web server and by monitoring the files and configurations on the web server itself. To monitor the content produced by the web server, you can implement a system that loads the web pages periodically (such as once an hour) and compares the newest result to a known valid prior page load. To monitor for changes on the web server itself, you use a file-integrity checker.

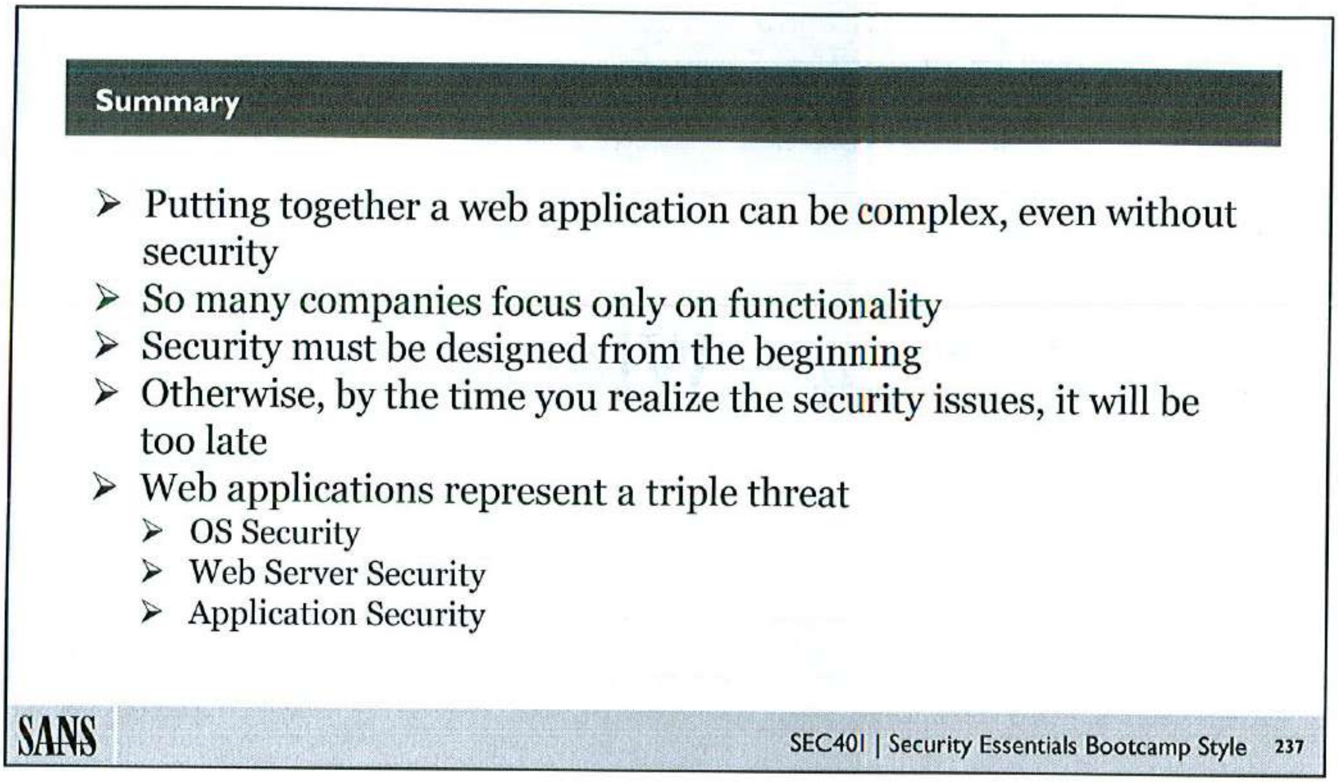
A file-integrity checker monitors the file system based on a number of preset rules and generates alerts when files are added, modified, or deleted out of compliance with those rules.

Implement a system to record and alert when the web application becomes unavailable. This provides the application owner protection against hardware, software, and network failures in addition to Denial of Service attacks. The most basic systems load a web page and generate an error condition if the web page does not load successfully. This helps to verify the functionality of the network and the web server, but does not test the web application or the supporting tiers, such as the database. A more complete and rigorous test would involve a recurring process that logs in as a valid user, performs a transaction or query within the application, and checks to see whether the expected result is returned.

There are several different approaches to defacement monitoring and availability monitoring. Because this is such a common and universal need, a large number of commercial service providers are now available that provide this service. In general, you pay a fee and get access to a configuration utility to set up monitoring processes on their servers. Many of these services include assistance in building/configuring more sophisticated monitors (for example, to build a multistep process to login to the web application and test a transaction). In addition to commercial service providers, there are also commercial software packages that you can purchase to install and run on your own servers.

Another approach is to build and maintain a custom monitoring solution. A home-grown solution provides more flexibility in the kinds of tests and alert actions that can be performed, but requires the skills and time to build and maintain the solution.

Scripting languages such as PERL or Python are well suited and seem to be frequently used for this task. Again, this is a common and universal need, so you can find a number of examples and code libraries available to assist you.



Summary

- ➢ Putting together a web application can be complex, even without security
- ➢ So many companies focus only on functionality
- ➢ Security must be designed from the beginning
- ➢ Otherwise, by the time you realize the security issues, it will be too late
- ➢ Web applications represent a triple threat
  - ➢ OS Security
  - ➢ Web Server Security
  - ➢ Application Security

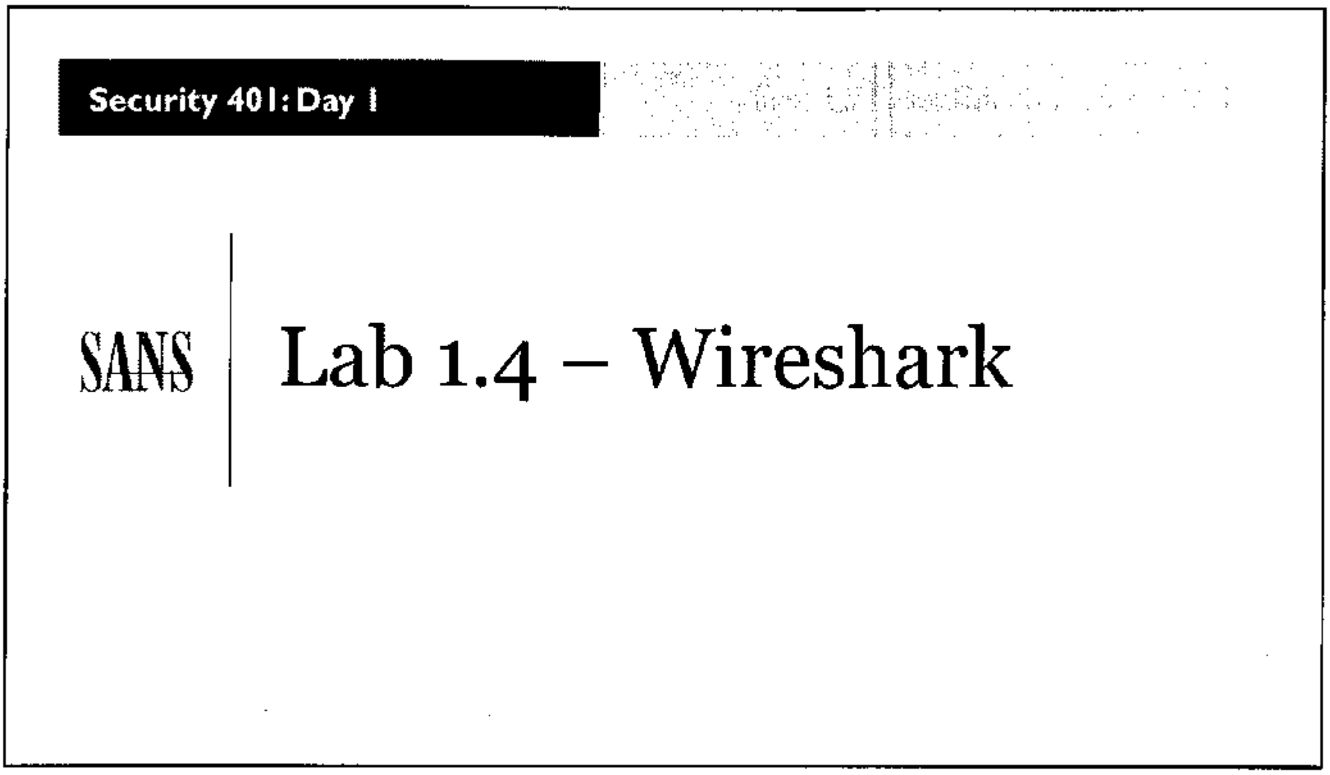
SANS SEC401 | Security Essentials Bootcamp Style 237

We started out learning about some basic web technologies. HTTP was first, where we showed you all about requests, methods, and headers. You also learned about HTML and forms. You learned about how applications can use SSL to establish secure communications and several ways to authenticate users: basic and digest authentication, form-based login, and certificate-base authentication.

With the basics out of the way, we learned about server and client-side web programming, along with some best practices for creating secure web applications. Your web-facing systems require extra diligence in hardening and monitoring. A formal web development and deployment process goes a long way in preventing the creation of vulnerabilities. If code development or server hosting is outsourced, thorough vendor management is also essential.

Next, we looked at common attacks against web applications and ways to defend against these attacks. We learned how web applications track sessions and how attackers can fool session tracking mechanisms to compromise your application. We also looked at vulnerabilities in access control, which allow web users to access data and run programs that you didn't intend to be available through the web-server application.

The web is a complex set of interlocking standards and protocols. Security in this environment is tough and requires constant vigilance. Secure your hosts, keep their patches up-to-date, follow safe programming practices, use appropriate technologies, and always, always, validate your input.

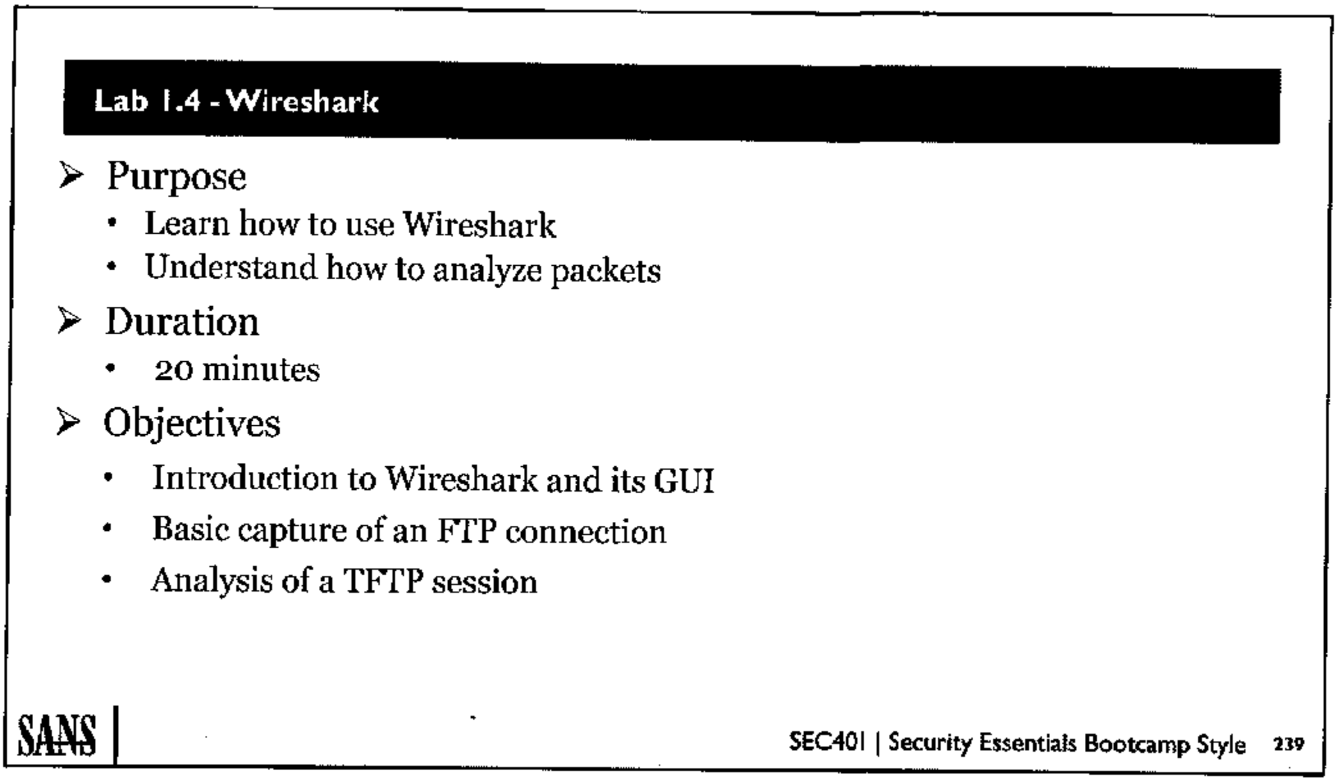


Security 401: Day 1

SANS

Lab 1.4 – Wireshark

Wireshark is the sniffer and protocol analyzer of choice by many information technology and security professionals, businesses, and academic institutions. It is freely available at https://www.wireshark.org/ and runs on Linux, Windows, and Mac OS X. Wireshark has a relatively easy to use graphical user interface (GUI) and can sniff using a myriad of Ethernet adapters, including wireless. You have the power to write your own protocol dissector for extensibility. It even has extended features, such as the ability to sniff and replay Voice over IP (VoIP) streams and handle Bluetooth interfaces.



Lab 1.4 - Wireshark

- Purpose
  - Learn how to use Wireshark
  - Understand how to analyze packets
- Duration
  - 20 minutes
- Objectives
  - Introduction to Wireshark and its GUI
  - Basic capture of an FTP connection
  - Analysis of a TFTP session

SANS

SEC401 | Security Essentials Bootcamp Style 239

Purpose

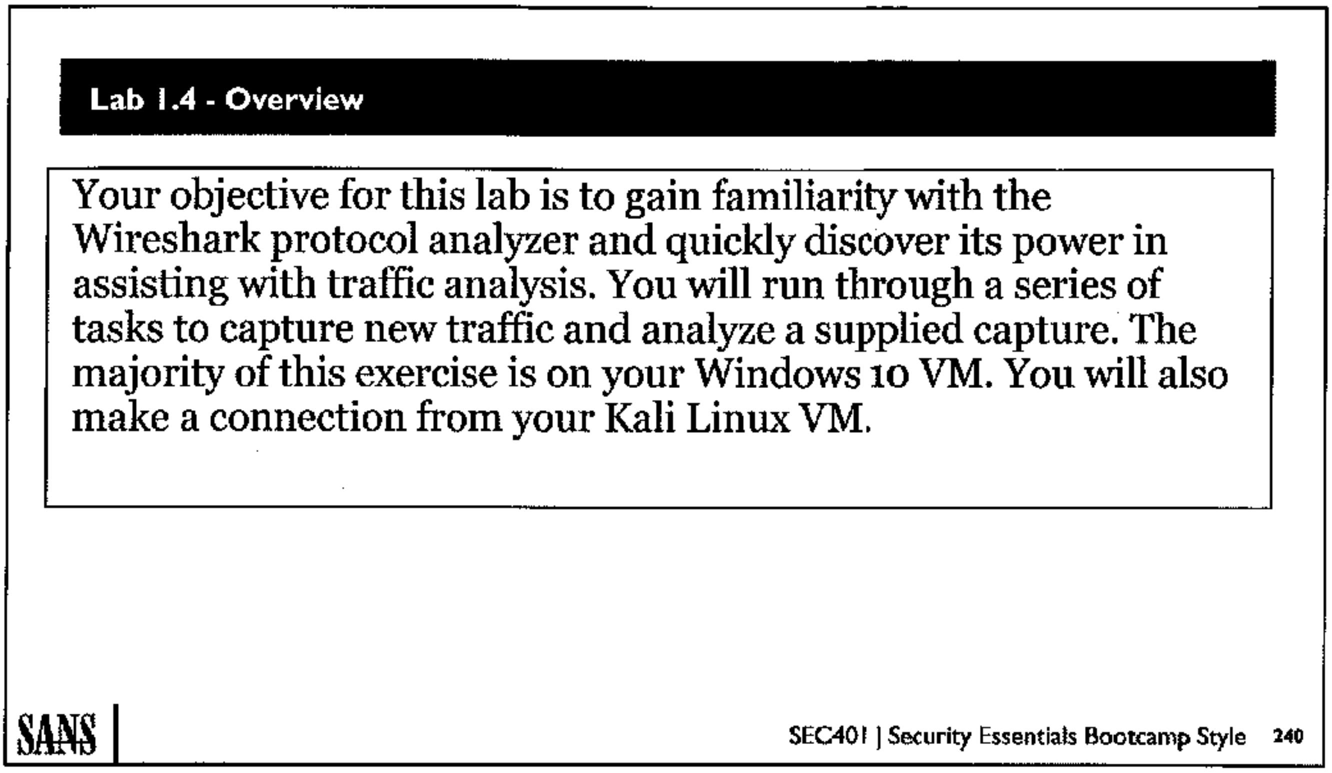
- Learn how to use Wireshark
- Understand how to analyze packets

Duration

- 20 minutes
- The estimated duration of this lab is based on the average amount of time required to make it through to the end. All labs are repeatable both inside and outside of the classroom, and it is strongly recommended that you take the time to repeat the labs both for further learning and practice toward the GIAC Security Essentials Certification (GSEC).

Objectives

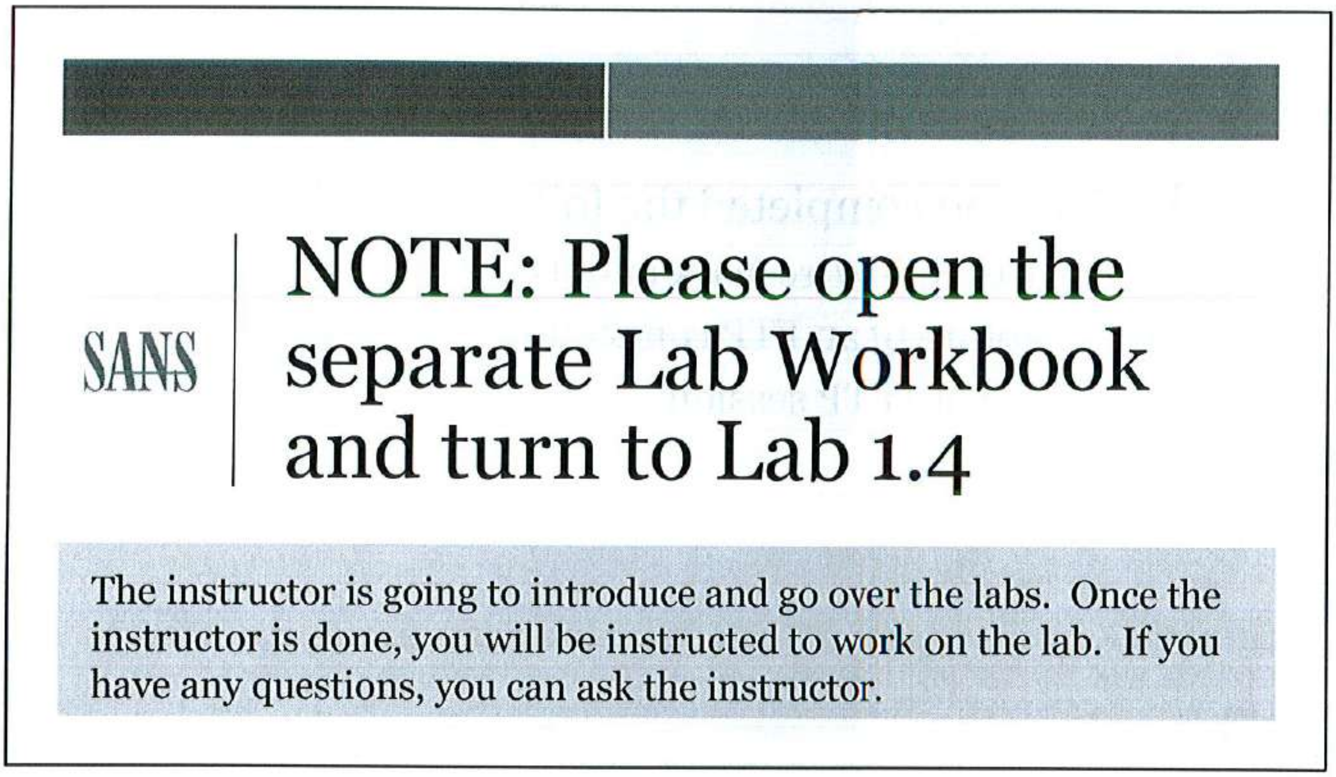
- Introduction to Wireshark and its GUI
- Basic capture of an FTP connection
- Analysis of a TFTP session



Lab 1.4 - Overview

Your objective for this lab is to gain familiarity with the Wireshark protocol analyzer and quickly discover its power in assisting with traffic analysis. You will run through a series of tasks to capture new traffic and analyze a supplied capture. The majority of this exercise is on your Windows 10 VM. You will also make a connection from your Kali Linux VM.

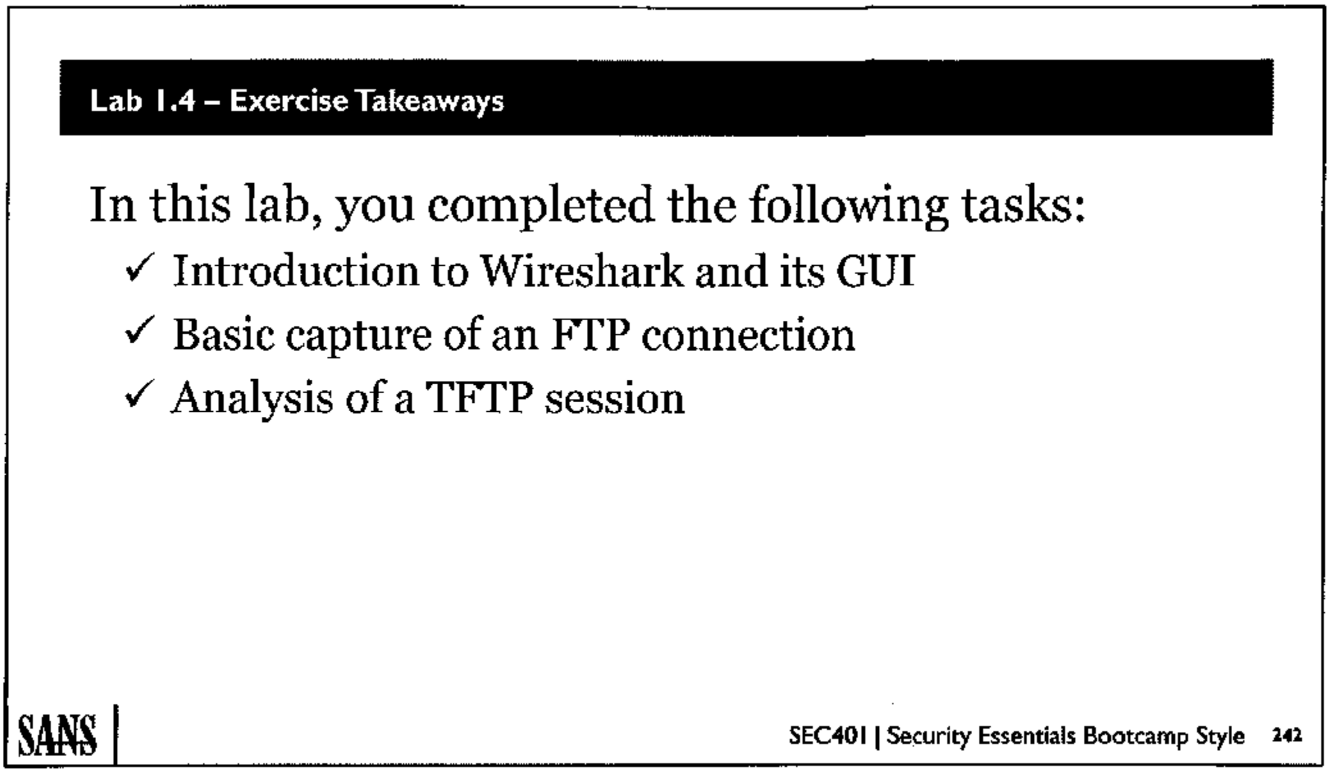
Your objective for this lab is to gain familiarity with the Wireshark protocol analyzer and quickly discover its power in assisting with traffic analysis. You will run through a series of tasks to capture new traffic and analyze a supplied capture. The majority of this exercise is on your Windows 10 VM. You will also make a connection from your Kali Linux VM.



NOTE: Please open the separate Lab Workbook and turn to Lab 1.4

The instructor is going to introduce and go over the labs. Once the instructor is done, you will be instructed to work on the lab. If you have any questions, you can ask the instructor.

This page intentionally left blank.



Lab 1.4 – Exercise Takeaways

In this lab, you completed the following tasks:

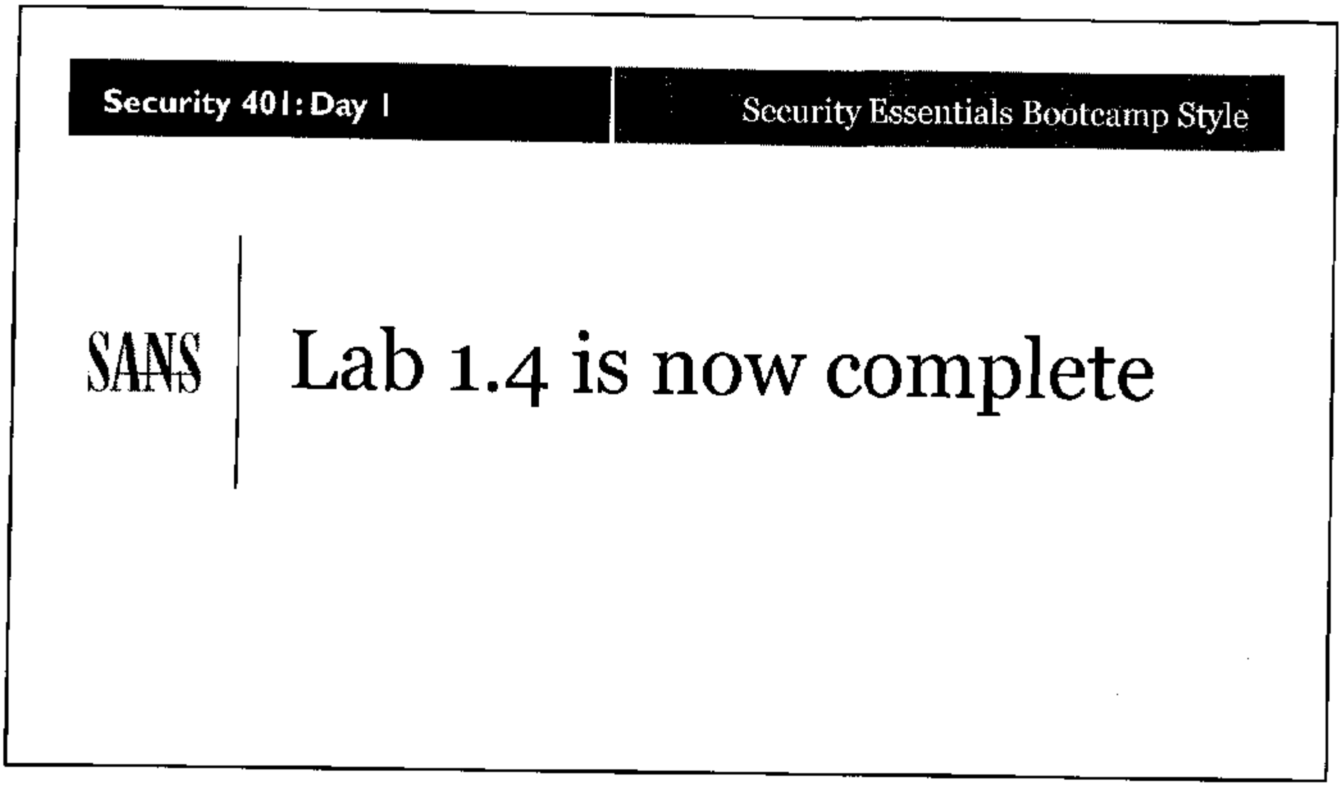
- ✓ Introduction to Wireshark and its GUI
- ✓ Basic capture of an FTP connection
- ✓ Analysis of a TFTP session

SANS | SEC401 | Security Essentials Bootcamp Style 242

In this lab, you completed the following tasks:

- ✓ Introduction to Wireshark and its GUI
- ✓ Basic capture of an FTP connection
- ✓ Analysis of a TFTP session

As you can quickly see, Wireshark is an amazing, user-friendly tool that has countless features. The ability to use Wireshark for packet and protocol analysis can save you countless hours and aid in troubleshooting network problems, security concerns, and many other uses. As with all sniffers, they must only be used with permission by trained technical staff. Sniffers may inadvertently expose sensitive data to the viewer. Most sniffers must also run under the context of an Administrator or Root in order to get the most value.



Security 401: Day 1

Security Essentials Bootcamp Style

SANS

Lab 1.4 is now complete

This page intentionally left blank.

"As usual, SANS courses pay for themselves by Day 2. By Day 3, you are itching to get back to the office to use what you've learned."
Ken Evans, Hewlett Packard Enterprise - Digital Investigation Services

SANS Programs
sans.org/programs

GIAC Certifications
Graduate Degree Programs
NetWars & CyberCity Ranges
Cyber Guardian
Security Awareness Training
CyberTalent Management
Group/Enterprise Purchase Arrangements
DoDD 8140
Community of Interest for NetSec
Cybersecurity Innovation Awards

Search SANSInstitute

SANS Free Resources
sans.org/security-resources

- E-Newsletters
  NewsBites: Bi-weekly digest of top news
  OUCH!: Monthly security awareness newsletter
  @RISK: Weekly summary of threats & mitigations
- Internet Storm Center
- CIS Critical Security Controls
- Blogs
- Security Posters
- Webcasts
- InfoSec Reading Room
- Top 25 Software Errors
- Security Policies
- Intrusion Detection FAQ
- Tip of the Day
- 20 Coolest Careers
- Security Glossary

SANS Institute
8120 Woodmont Avenue | Suite 310
Bethesda, MD 20814
301.654.SANS(7267)
info@sans.org