SEC301 | INTRODUCTION TO CYBER SECURITY
GIAC Information Security Fundamentals (GISF)

301.5

Cyber Security Technologies & Web Security

SANS | GIAC CERTIFICATIONS

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

SEC301.5 | Introduction to Cyber Security

SANS

Cyber Security Technologies & Web Security

This page intentionally left blank.

Module 14: Network Security Technologies

- Compartmentalization
- Firewalls and DMZs
- IDS/IPS
- Content Filter
- Unified Threat Management
- Sniffer
- Penetration Testing
- Active Defense
- Threat Hunting

COURSE ROADMAP

- **Module 14: Network Security Technologies**
  - Lab 5.1: Firewall Builder
- Module 15: Browser and Web Security
- Module 16: System Security

Module 14: Network Security Technologies

This module covers a great number of topics. They are all related in that they are types of security topologies: Ways of breaking up a network to give us the ability to implement controls. In some cases, they are the actual controls.

As we move through this section, keep the first topic, compartmentalization, in mind. Everything else in this module is part of how we accomplish compartmentalization.

Compartmentalization

- Separating a network into distinct areas
- Compartmentalization versus Segmentation
  - Segmentation = dividing a network for efficient management
    - Networks, subnetworks, VLANs, and so on separated by routers
  - Compartmentalization = dividing a network into security zones
    - Intranets, extranets, enclaves, and so on separated by security devices

Enclave: ˈen-ˌklāv, ˈän-ˌklāv
A distinctly bounded area enclosed within a larger area.

Compartmentalization

Network *segmentation* has been done for a long time by the network operations folks. They use routers, VLANs, subnetworks, and so on to break up a network. But segmentation is done for efficient management of IP addresses, routing traffic efficiently, and so on. It is purely a network management issue.

Compartmentalization does the same thing: Break up a network into different areas. But compartmentalization does this for an entirely different reason. We break up the network by functional area. For example, let's put everyone in Human Resources on one network and everyone in accounting on a different network. Doing this enables us to control who can see different kinds of traffic more easily. If you put a user from HR and a user from accounting on the same network, the odds that one of them may see some of the other's traffic goes up dramatically. This could lead to security issues.

Part of a compartmentalization plan could include creating enclaves for especially sensitive systems. Examples include any central authentication server, the Research and Development server, and so on. These tend to be servers with extremely sensitive information on them. By using firewalls, IDSs and IPSs, and some of the other technologies we talk about here, we can create *enclaves* (distinctly bounded areas enclosed within a larger area). We can then carefully control traffic to those systems and allow only the systems administrators to actually access them directly.

Firewalls (1)

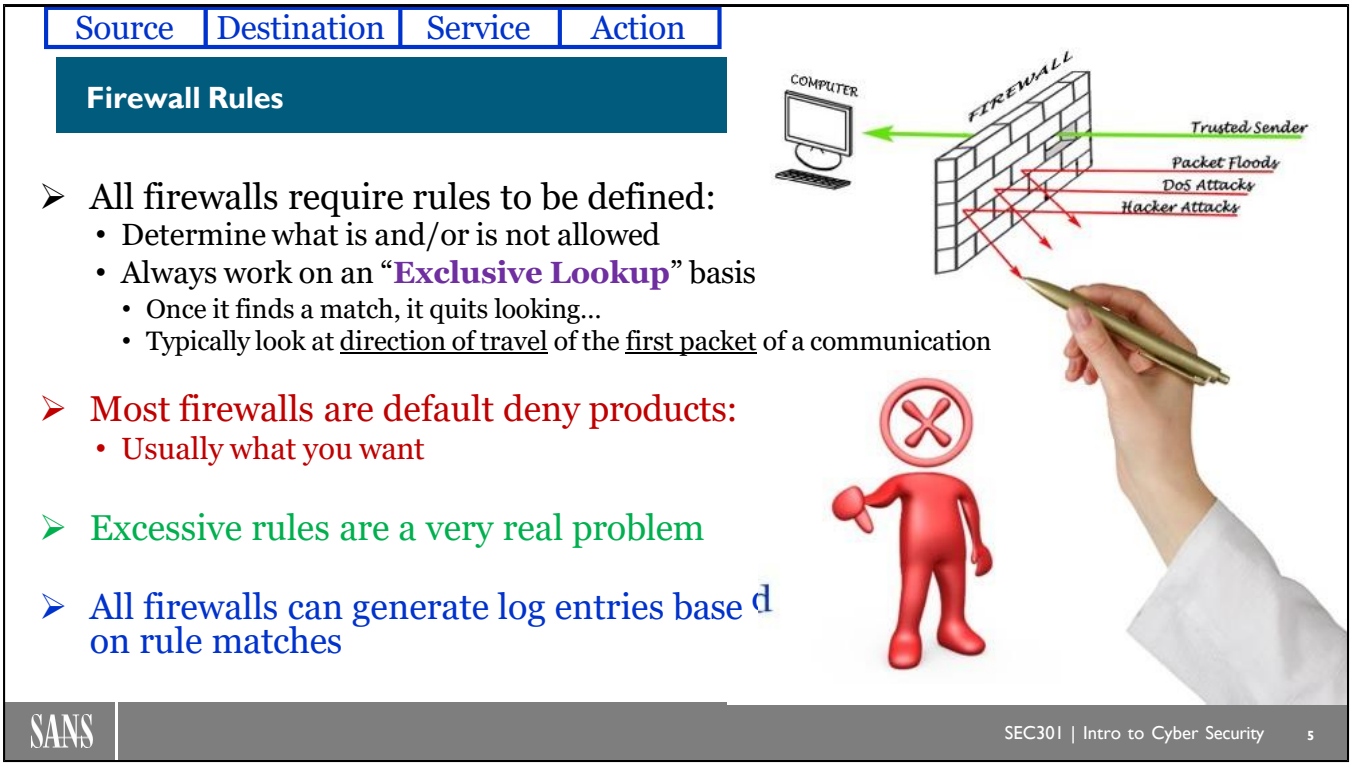
- A primary mechanism to provide security separation:
  - Fundamentally: To keep people off a network
  - Also used to prevent employees from going where they shouldn't
- Firewall types:
  - Packet filter
  - Proxy
  - Stateful Inspection
- Web App Firewall:
  - HTTP-aware firewall, it understands web traffic
  - Can filter out SQL injection, CGI invalid input, and more

Firewalls (1)

The *firewall* is the primary mechanism to provide security separation between two networks. Most people think of firewalls as going between the internal network and the internet, and they absolutely do belong there! But they can also be used (returning to our example from a few minutes ago) to keep people from HR out of the accounting network and vice versa. As already stated, the firewall is also a significant part of how we create private enclaves for sensitive systems.

Here, we discuss three types of firewalls. We use common terminology but do realize that some vendors may have different names for everything we talk about. Because understanding firewalls is so important to understanding fundamental network security, we devote time to each of the three types listed here in upcoming slides.

There is a fourth type of firewall that has become important and deserves mention. The Web App Firewall (or Web Proxy or half a dozen similar names) is a special purpose type of firewall. It sits in front of your web server, meaning between the web server and the web browser, and controls all traffic to and from that server. A good Web App Firewall is going to be fully knowledgeable about attacks against web servers and can block those attacks in real time. For example, they would know exactly how to recognize an SQL Injection attack and prevent it from working. They should also protect your website from defacement.



Firewalls (2)

Although every firewall type and product are different, some general statements apply across the board. Those include what you see here.

Every firewall product requires you to configure rules of some kind. Exactly what those rules look like as far as their syntax or the interface you use to create them can vary from product to product. But in one way or another, you need to put rules on your firewall. On a packet filtering router, the rules are called an Access Control List (ACL) that you load onto the router and apply to an interface. For most if not all stateful inspection and proxy firewalls, there is some kind of Graphical User Interface (GUI—pronounced Gooey) management interface you will use to configure the rules.

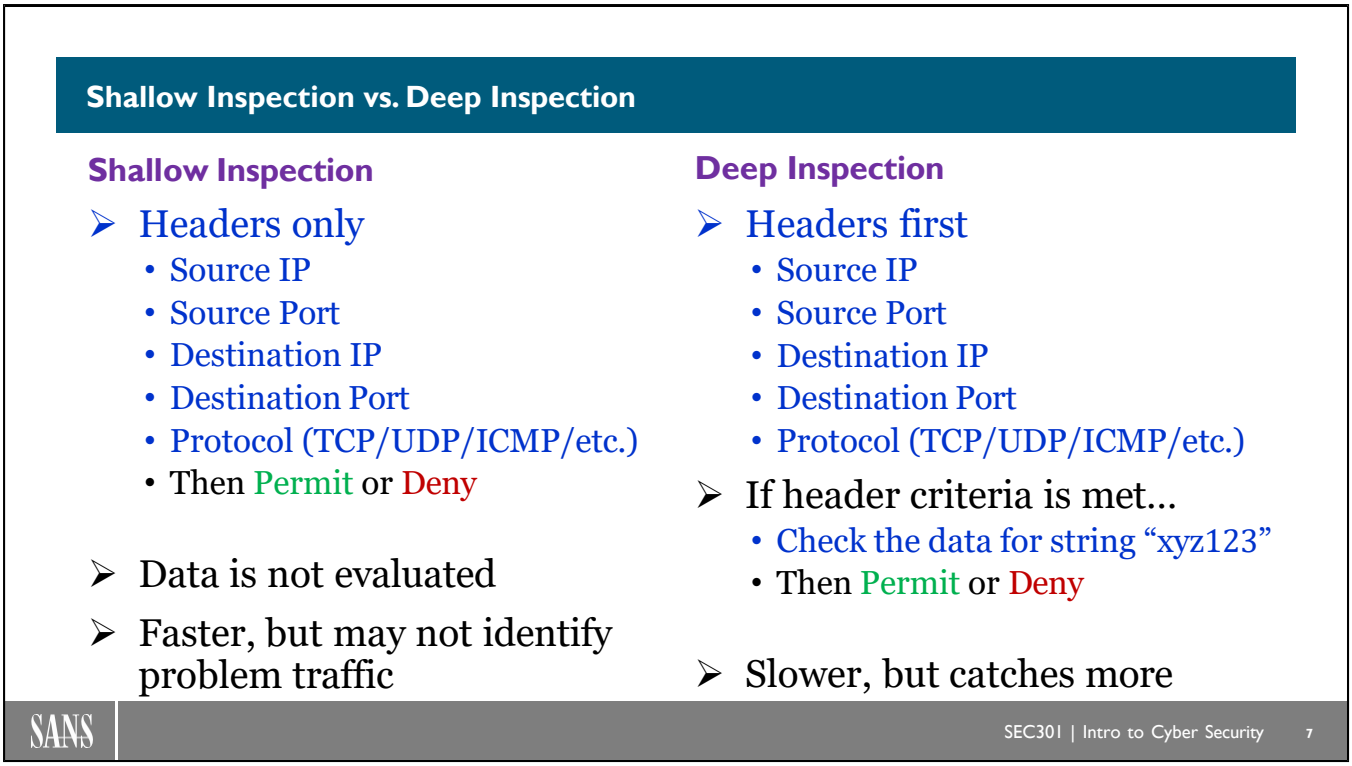
How do you know what rules you should put on the firewall? It's a simple answer or should be. Your organization's policy tells you what rules you need. If your policy states that internal users can surf the web, that is a firewall rule you need to create.

Most firewalls on the market today are *default deny* type products. In other words, you put in rules to allow necessary traffic and everything else is automatically denied. For the vast majority of organizations, that is the wanted default configuration. There are some firewalls that *default allow*. In certain situations, this can also be good. The important thing is, you *must* know which default your product uses. That default dictates everything else about how you configure your firewall. Configuring a default deny product is different from configuring a default allow.

Excessive rules—or "rule creep"—is a very real problem with firewalls. As new firewall admins come in, they add their own rules. They don't know what the existing rules are for, so they leave them there. This can result in a LOT of extra rules on a firewall that simply don't need to be there. ***Please*** always add a comment to your firewall rules so the next person in the job will have some idea why the rule was added.

All firewalls we have ever worked with can also generate logs of some kind. Some of the products have advanced logging and alerting capability, often enabling you to use filters to narrow down log entries when searching for specific types of entries. On firewalls with more advanced logging capability, they often have alerting that can tie into your Simple Network Management Protocol (SNMP) infrastructure, or other similar alerting capability.

Some firewalls have extremely rudimentary logging capability. Indeed, some have nothing more than sending logs to a standard syslog server. Although this is certainly better than no logs at all, it is not the level of logging the more advanced products give.



Shallow Inspection vs. Deep Inspection

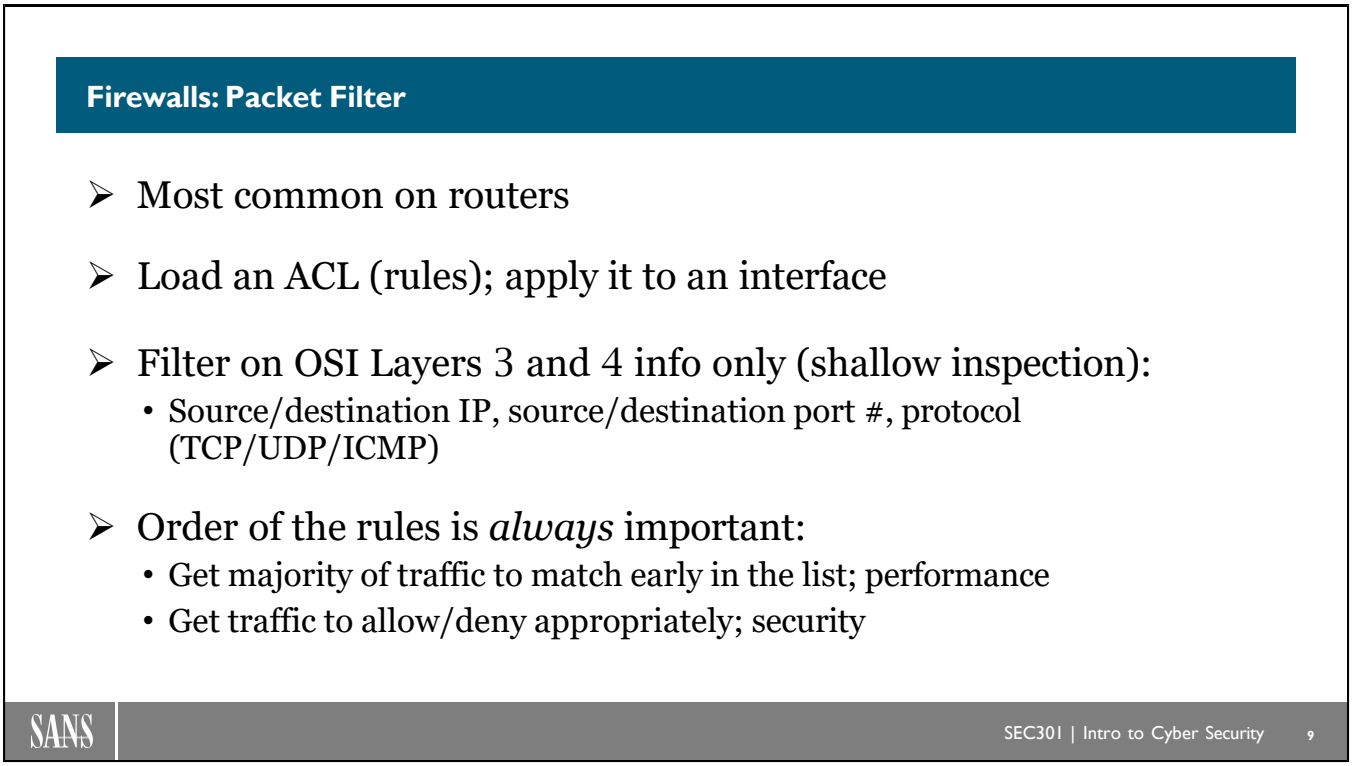
As firewalls analyze packets, they use one of two methods. They are most commonly called "Shallow Inspection" and "Deep Inspection".

Shallow inspection means that the firewall only examines the header information, not the data. Most of the time, this means that it will look at the source and destination IP addresses, the source and destination port numbers, and the protocol (filtering on the protocol field of the IP header). In other words, if the packet is

- coming from any computer with the network address 10.10.10.0,
- is going to the specific address 10.10.22.7,
- is from a port number equal to or greater than 49152,
- and is going to a port number 53 (DNS),
- and it is TCP traffic, then deny it.
- But if meets all of those criteria and is UDP, then allow it.

Deep inspection begins by doing the same thing as shallow inspection: It analyzes the IP addresses, ports, and protocol information. If that information matches the criteria you specify, it can then go into the data portion of the packet to look for information there. For example, it might look for a particular string. If it finds that string, it can allow or deny the packet.

For example, there are two versions of SSH: Version 1 and version 2; version 1 has several vulnerabilities so you only want to use version 2. Unfortunately, they both use TCP port 22. So a shallow inspection firewall can only allow or deny all SSH traffic, but cannot do so by version. A deep inspection firewall can allow TCP port 22 and then if it sees the code for version 1 in the data, deny the connection, if it sees the code for version 2 in the data, allow the connection.



Firewalls: Packet Filter

You most commonly find the basic packet filter firewall technology implemented on routers. Most routers on the market have the capability built in to them—you just have to configure it. In this way, they are considered inexpensive because you already had the router to begin with.

To set up a packet filtering router, you create a list of filtering rules called an Access Control List (ACL). You load that into the router's configuration. Finally, you apply that ACL to an interface. At that point, it begins filtering traffic.

When you create the rules for the ACL, you can filter on information only from the OSI Layers 3 and 4 headers. Specifically, you can filter on the source and destination IP addresses, the source and destination port number, and the protocol type such as TCP/UDP/ICMP/etc. (based on the protocol field of the IP header). In other words, you can create a rule that says:

If the packet is:

- From the 10.1.1.0/24 network and an ephemeral port number
- To the 10.2.2.2 server and a destination port number of 53 (DNS)
- Is *TCP* traffic

That indicates it is a DNS Zone Transfer, which our policy says we should deny.

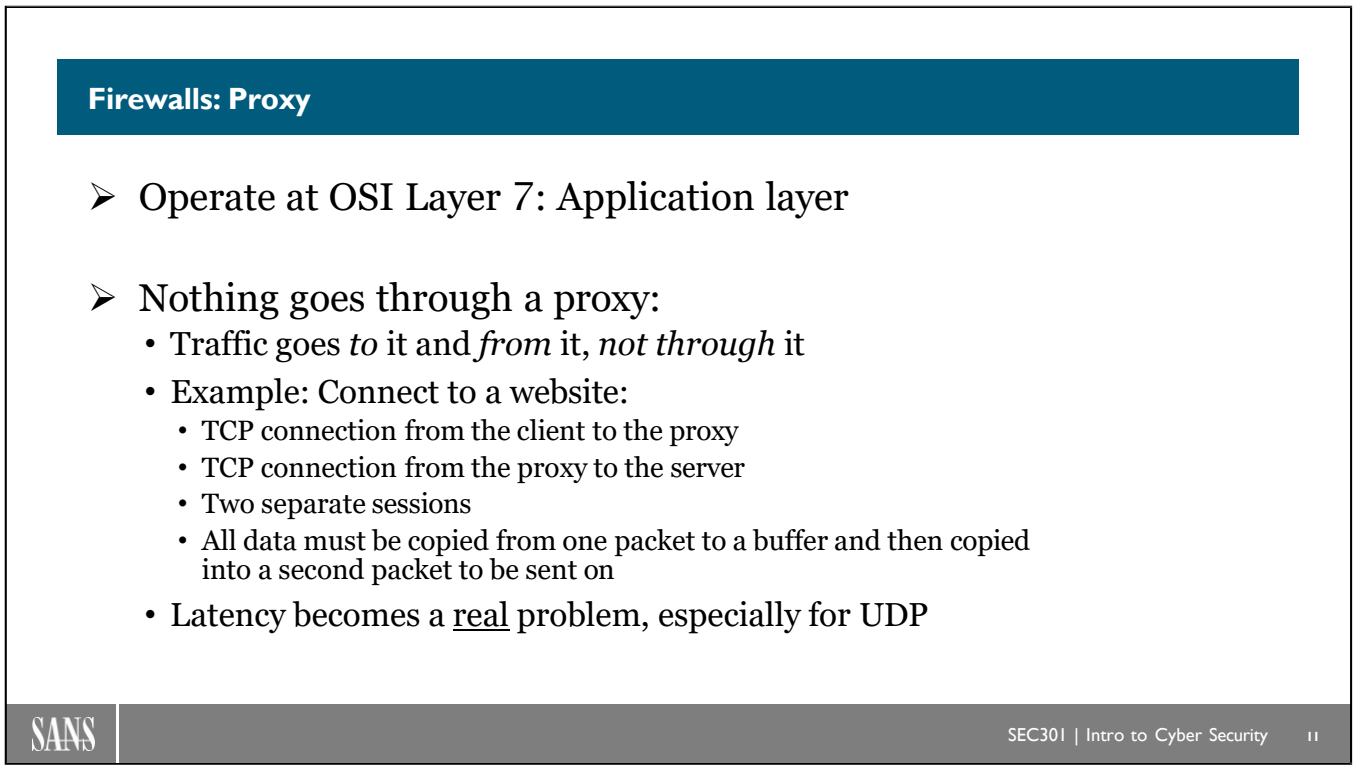
But,

If the packet is:

- From the 10.1.1.0/24 network and an ephemeral port number
- To the 10.2.2.2 server and a destination port number of 53 (DNS)
- Is *UDP* traffic

That indicates it is a DNS address lookup, which the policy says we should allow. (Notice that the only difference in the criteria is that one is TCP and the other is UDP.)

With a pure packet filter technology, that is as much granularity as you can get.



Firewalls: Proxy

The Proxy firewall has become rare today, at least for the firewall sitting between a corporate network and the internet. Specialized proxies sitting in front of web servers and similar devices are still quite common.

The proxy firewall operates at OSI Layer 7. This means that the logic that checks the validity of the packet is at the application layer of the firewall.

One of the most important things to understand about a proxy firewall is that *nothing* goes *through* a proxy. Traffic goes *to* the device. Traffic goes *from* the device. No traffic goes *through* the device.

This means that every connection to the internet results in two separate connections: One from the client to the proxy and one from the proxy to the internet server. From a security perspective, this is good. The internet server has no idea that the client system exists; all its communication is with the proxy. From a performance perspective, this is horrible. Each time a packet containing data arrives at a proxy, the data has to be copied out of the packet and into a buffer. A new packet has to be created to go out the other side, and the data is copied from the buffer into the new packet.

This can cause unacceptable latency and is the primary reason that proxy firewalls have fallen out of favor. This is especially true for UDP traffic because UDP has no state, and each packet has to be approved independently. VoIP communication (which uses UDP) is unacceptably slow through a proxy.

Firewalls: Stateful Inspection

- The most common firewall type today
- Underlying technology = Packet Filter
  - The "State Engine" and "state table" make it a much more powerful technology
  - Strengthens basic packet filter capability
- All can do shallow inspection
- Many can do deep inspection

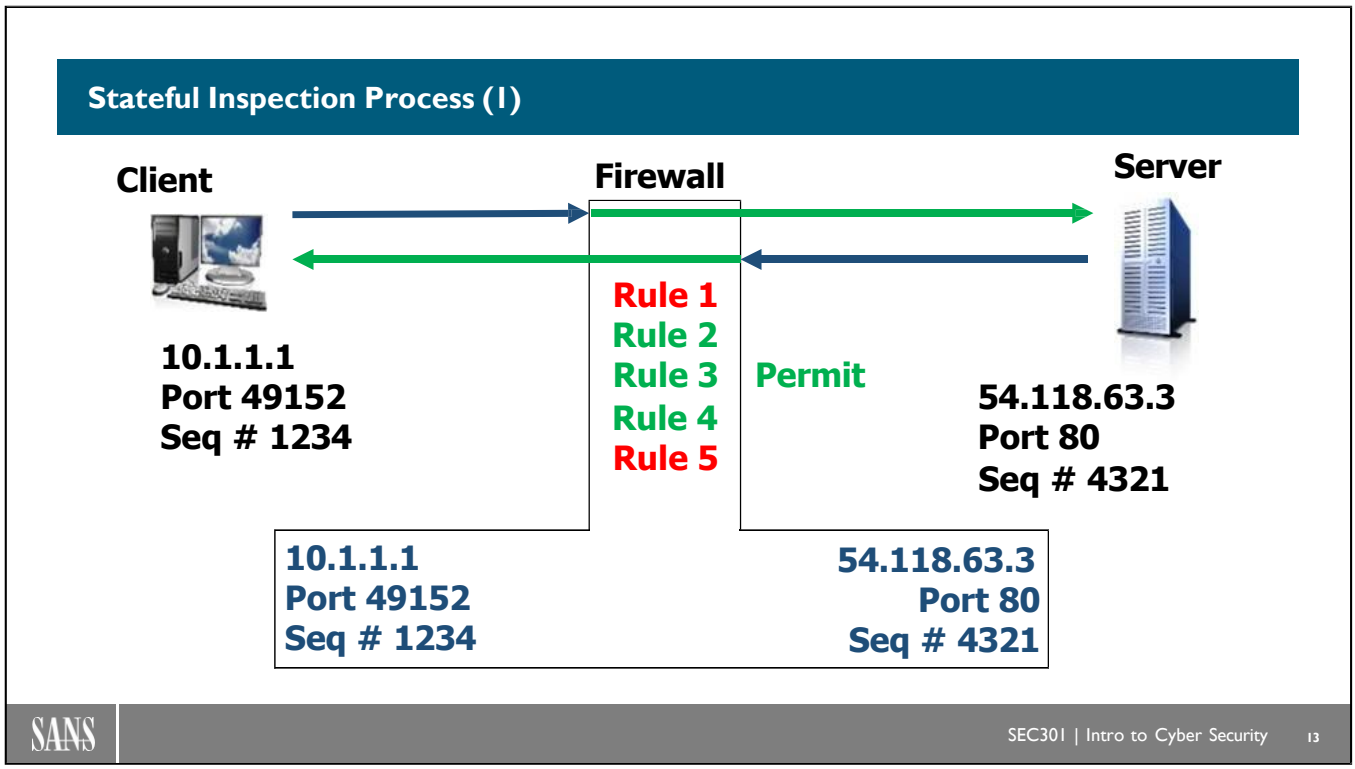
Firewalls: Stateful Inspection

A stateful inspection firewall is based on packet filtering. Therefore, everything we said about packet filtering remains true. However, there is something called the *state engine*, which uses a state table that takes basic packet filtering to a whole new level.

The initial packet of any communication must match a rule in the ruleset to be allowed through. After that happens, the initial packet is allowed through the firewall. As it traverses through, the firewall takes information from the packet (source and destination IP addresses, port numbers, sequence numbers, and more) and builds a state table entry.

When the distant system sends its response, the firewall does not compare it to the ruleset. Instead, it looks in the state table. If the packet matches the state table entry, meaning it sees the IP addressing is correct and the port numbers are correct, and all the rest matches, it allows the response packet back through.

So stateful inspection gets its name because (after the initial packet matches a permit rule) all subsequent packets are allowed based on the "state of a current connection." After the firewall sees the session terminate (with a FIN or RST packet), it closes the connection back off and removes the state table entry, so the communication path is open only as long as the connection needs it.



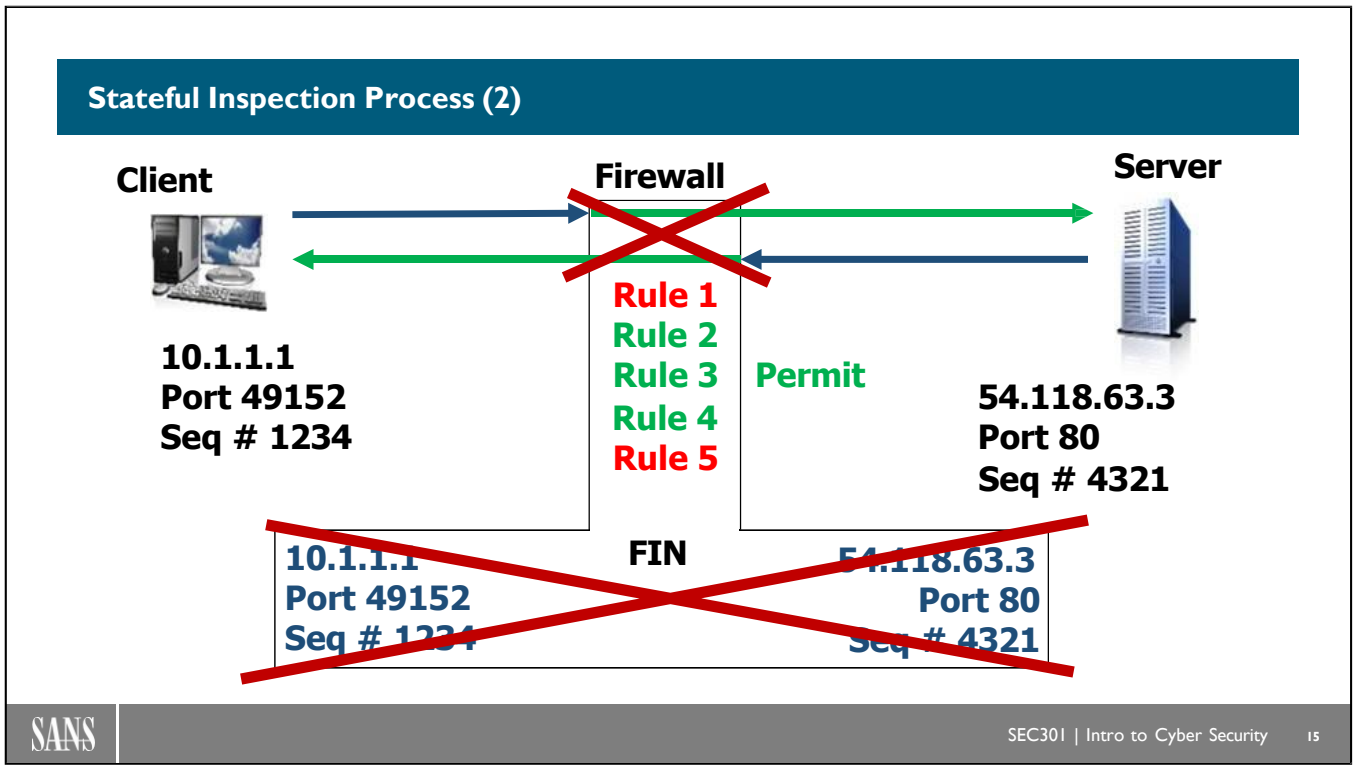
Stateful Inspection Process (1)

Here, we see a client and a server with a stateful inspection firewall in between. To start the process, the user at the client PC opens their web browser and types in the IP address of the web server on the right.

- The TCP SYN packet travels from the client toward the server but hits the firewall.
- The firewall stops that packet from going further until it can inspect it and make a decision.
- The firewall sees that the packet is from an internal IP address going to a public server. With a destination port of 80, the user asks to use the server's web service.
- The firewall looks in its state table but does not find a match. (We return to the state table shortly.)
- Because there is no match in the state table, the firewall parses through the rules to see if there is a match there. Sure enough, rule 3 says that internal IP-addressed computers can go to external websites: The action is to be permitted.
- The firewall dynamically reconfigures the packet filter to allow the packet to go through. (Another name for this type of firewall is *Dynamic Packet Filter*.)
- As the packet traverses through the firewall, the firewall takes information from the packet and creates a state table entry (denoted at the bottom part of the firewall in the diagram).
- Among other information, the state table entry includes the source and destination IP addresses,
  the source and destination port numbers, the source sequence number, and more.

(The process continues on the next page.)

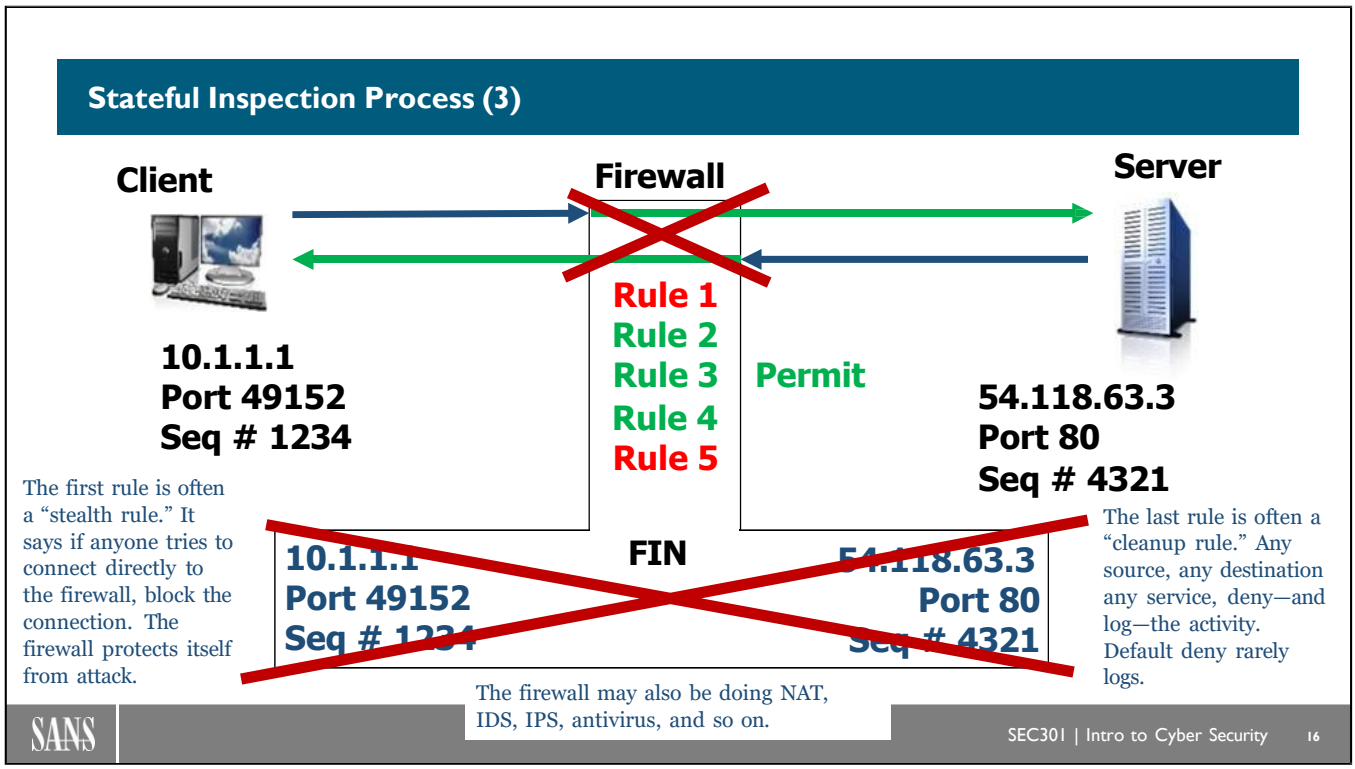
- When the SYN packet arrives at the server, it responds with the second step of the TCP handshake, a SYN/ACK (assuming the server has port 80 open for connections).
- When the SYN/ACK hits the firewall, it is stopped for inspection.
- The firewall looks in the state table entry and sees that the SYN/ACK is coming from the correct IP and port number and that it is going to the correct IP and port number, and so on.
- BASED ON THE STATE OF A CURRENT CONNECTION, it makes sense to dynamically reconfigure the packet filter and allow the SYN/ACK through.
- The firewall updates the state table with any new information (such as the server's sequence number).
- As the communication continues to flow back and forth, the packets are allowed through the firewall based on the state table entry, not based on the rules.
- Only the first packet of any communication has to parse the rules.
- This is true regardless of the type of traffic that passes through the firewall, whether it be TCP, UDP, ICMP, or anything else. A stateful firewall handles all traffic with equal efficiency.



Stateful Inspection Process (2)

Eventually, either the client or the server determines that it is time to end the connection. When this happens, the system sends a TCP FIN packet to the other end, signaling that it wants to begin the four-step TCP teardown.

As the FIN packet traverses through the firewall, the firewall makes note of the FIN in the state table. The firewall allows the four-step teardown to complete. As soon as the teardown is done, the firewall removes the state table entry and dynamically reconfigures the packet filter to close the ports off again. Those ports are open only as long as the state of the communication continues to dictate that they should remain open.

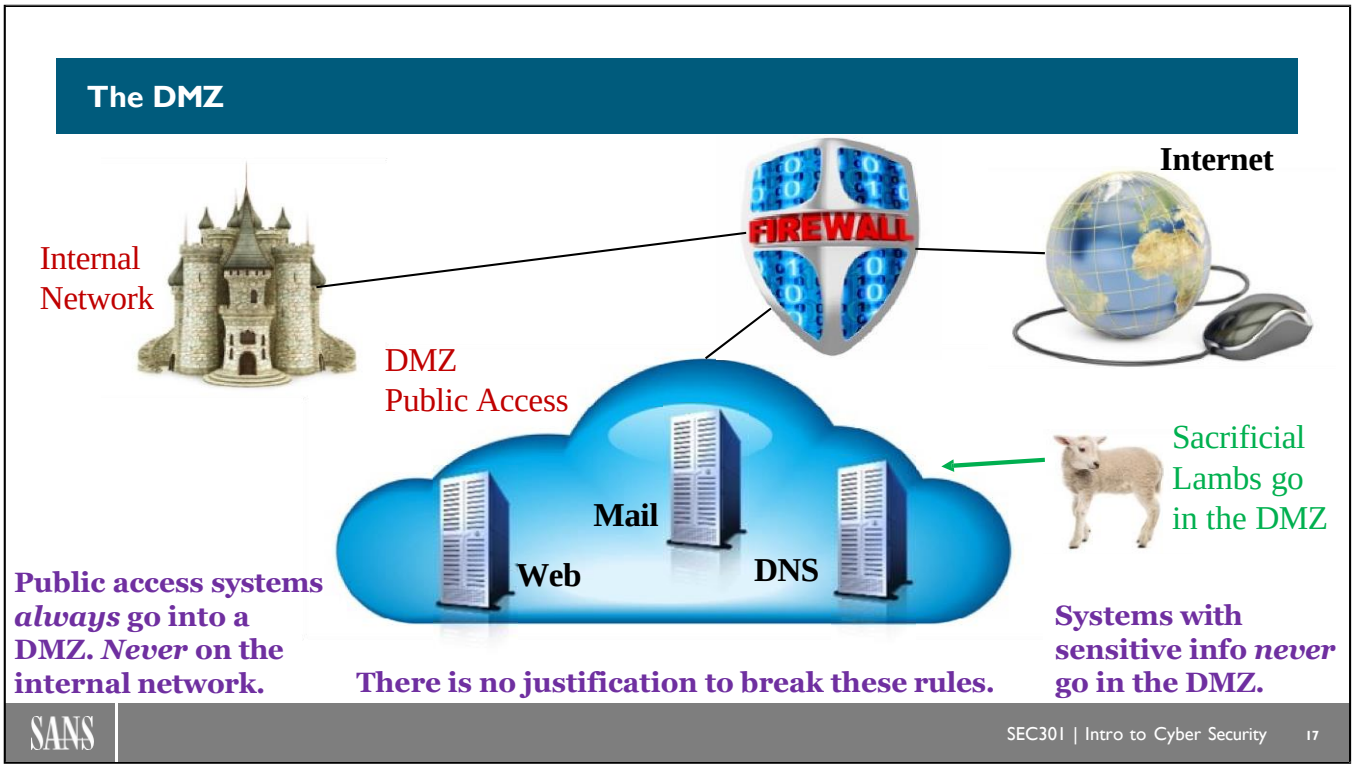


Stateful Inspection Process (3)

Do note that, depending heavily on both the brand and configuration of the stateful inspection firewall, it may do many other chores for us at the same time that it does stateful inspection. Examples of what it may or may not be doing include:

- Network Address Translation (NAT)
- Intrusion Detection System (IDS)
- Intrusion Prevention System (IPS)
- Gateway antivirus
- Content filtering
- Virtual private network (VPN) encryption and decryption

The list of possibilities is quite long here. As a general rule of thumb, the more money you spend on a firewall, the more of these features the product will have. Indeed, in some cases, each of these features is an additional cost add-on to the base firewall license.



The DMZ

The Demilitarized Zone (DMZ) of your network is the public access area of your network. In other words, this is where you put all of the servers that you will let people you don't know connect to.

On your internal network, you only allow your own users that you can authenticate, and those who have had security skills/awareness training. This is a fairly high degree of control.

The DMZ is where you put the web server that you have to allow anonymous access to. People you do not know, who you cannot authenticate, and who have never trained are allowed to connect to that server. It is not just one server—you also have your mail server, your DNS server, and perhaps others. Sooner or later, someone is going to find something they can exploit on one of these servers. They will be exploited at some point—it is simply a question of how long it will take and how bad it will be.

To make the "how long will it take" be as long as possible:

- Harden the operating system of any DMZ server. Remove anything and everything that is not absolutely required for that system to perform its mission in life.
- Keeping all systems patched is important, but especially for DMZ servers. Patches should be applied in the DMZ as quickly as possible after release. Yes, you still need to test the patches first, but any patch that applies to DMZ systems should have expedited testing.

- Use the firewall in front of the DMZ to limit access to DMZ servers only on the required ports. You have to allow some access, of course. A World Wide Web (WWW) server must, by definition, be accessible by everyone in the world on TCP Port 80 and possibly TCP Port 443. Therefore, those ports must be allowed through the firewall. But only those ports and only to that one IP address. Be as restrictive in your firewall rules as you possibly can and still get the job done.

To make the "how bad will it be" as minimal as possible:

- Again, employ your firewall to limit access by the DMZ server to internal systems. Preferably, DMZ servers have zero access to internal systems. In some cases, that is not possible. For example, your web server may require access to an internal database server. If that is the case, the firewall between the DMZ and the internal system should *only* allow that one DMZ server to go to that one internal database server, and only on required ports and so on. Again, be as restrictive as possible in your firewall rules.
- Have good backups of any data that resides on DMZ servers, especially data that you cannot reconstruct on your own (customer orders, for example).
- Have hot-swap servers ready to put online at a moment's notice. This way, when a server is compromised, you can take it offline for analysis and put a hot-swap server in place to continue providing service with minimal interruption.
    - If the hot-swap server capability is not possible, have the ability to reimage and redeploy DMZ servers quickly. This can cause you to lose the ability to analyze how the compromise occurred (increasing likelihood of recompromise) but will allow you to put an uncompromised server back in service quickly.

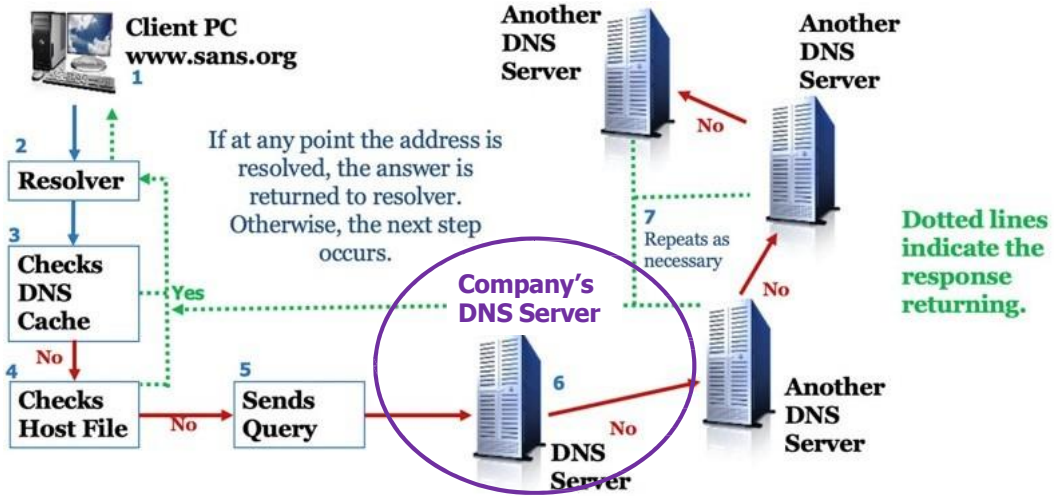
A critical point: Never—ever—for any reason—under any circumstance put a public access server on your internal network! There is absolutely no possible justification for doing so. This is perhaps the single worst network design mistake that you can possibly make.

You could block sans.org via a firewall rule, but this method is zero latency!

Sinkholes

Remember the DNS process...

If at any point the address resolves, the process stops. In other words, if step 6 knows the address, step 7 does not happen.



What if step 6 resolved sans.org to the IP address 0.0.0.0? The address is "sinkholed".

Sinkholes

Firewall rules are great for blocking traffic. Many companies use them not only to filter incoming traffic to only what is allowed, but to stop employees from going to certain sites as well. The potential problem with doing so is the more firewall rules you have, the more latency you will introduce on the network. Too much latency and you will be told to remove the security mechanism.

A second method of stopping internal employees from going to unauthorized locations on the internet is called a sinkhole. Assuming the DNS server in step 6 of the diagram is the company's DNS server, you can easily configure any number of sinkholes.

In the example on this slide, we configured the company's DNS server to resolve sans.org to the IP address 0.0.0.0. That address is not going to route properly, so the traffic cannot reach the destination.

Part of understanding sinkholes is to remember the DNS process. Once an address resolves, the process ends. So if there is a resolution in step number 6, step number 7 does not occur.

Earlier in the course, we mentioned some public DNS providers that offer "family friendly" DNS service that will not properly resolve adult sites. Many of those actually work by setting those addresses to resolve to all zeros.

Content Filter

- Blocks access based on content:
  - List of blocked sites
  - Offensive language / drug paraphernalia / adult sites / social media / etc.
  - Pictures with too much "skin tone" color
- Most think of blocking web traffic:
  - And that is certainly common
  - But can also block email:
    - Containing proprietary information
    - Flames – "hotly worded emails"

Content Filter

A web content filter is software and/or hardware that blocks access by users to internet sites based on the content. Generally speaking, you can configure a device of this nature by selecting categories of sites you don't want your users going to. Common categories include pornography, hate speech, illegal drug-related, and similar such sites.

Less common is the email content filter. These can be useful in stopping inappropriate email. Examples of such emails include *flames* (emails with abusive, vulgar, or otherwise inflammatory language). Typically, software of this nature either blocks the message entirely, gives senders a 24-hour cooling off period before asking if they still want to send it, or forwards it to their boss for evaluation.

You can also usually configure a list of keywords into email content filters. Say you have a super-secret project going on within the company. By configuring a list of keywords that would indicate a message deals with that project, you can stop such email from accidentally (or intentionally) being sent outside your corporate network.

IDS/IPS (1)

- Intrusion Detection System (IDS):
  - Automated system watching for signs of an attack:
    - Performs continuous security monitoring
  - Uses signature, anomaly, or behavior detection
  - Problems:
    - False positive / false negative / true negative / true positive
      - See chart on next slide
- Intrusion Prevention System (IPS):
  - An IDS that can shut down attacks
  - IF they are correctly identified first

IDS/IPS (1)
An **Intrusion Detection System (IDS)** is an automated system watching for signs of an attack. Some IDSs are devices connected to the network and watch network traffic for attacks, and some run on the host (PC) and watch for signs of attack there. They are respectively called Network IDS and Host IDS.

IDS systems use a combination of signature and anomaly or behavior detection. Both are similar to what we discussed in anti-malware. Signature analysis has a database of packet header settings and/or strings in the data portion of the packet. The IDS monitors the packets to determine if it sees those patterns.

Anomaly detection is very much like heuristics in anti-malware. The IDS is watching for attack-like activity on the network.

An **Intrusion Prevention System (IPS)** is simply an IDS that can shut down the attack. Because it is not possible to shut the attack down until it first identifies the attack, an IPS must first be an IDS.

IDS/IPS (2)

Events of Interest (EOI)	ATTACK	ALERT/BLOCK
TRUE NEGATIVE	N	N
TRUE POSITIVE	Y	Y
FALSE NEGATIVE	Y	N
FALSE POSITIVE	N	Y

Question:
Which is worse?
- False Negative?
- False Positive?

IDS/IPS (2)

This chart shows one of the most important points to understand about IDS/IPS. When an IDS/IPS recognizes an Event of Interest, what kind of event is it? How do we categorize it?

- True Negative (It is legitimate traffic, and the IDS/IPS got it correct.)
- True Positive (It is attack traffic, and the IDS/IPS got it correct.)
- False Negative (It is attack traffic, but the IDS/IPS thinks it is legitimate.)
- False Positive(It is legitimate traffic, but the IDS/IPS thinks it is an attack.)

From the chart, you can see that the first two are the two we want to see the most of. They essentially mean that the IDS/IPS identified and classified the traffic correctly. The second two, you hope not to see. Reality, however, is that you see them a great deal, especially if the settings on your IDS/IPS are not done very well.

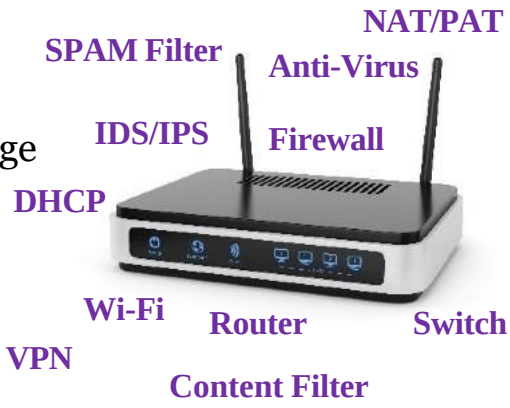
The question becomes, which is worse? The false negative or the false positive? Because the false negative means that the IDS/IPS is missing attacks, many say it is the worst by far. Certainly, it is not a good thing. However, if you have too many false positives, they become white noise and people stop paying attention to what the IDS/IPS is reporting. Certainly, this is just as bad. There are numerous documented cases of IDS systems reporting attacks and exfiltration of data, but nobody notices because there are many gigabytes per hour of false positives being generated.

With IPS, you cannot allow false positives at all, since that would mean the security device is blocking legitimate traffic! This cannot be allowed to happen.

A.K.A. = UTM (Unified Threat Management)

All-In-One Security Appliance

- Several network devices inside a single box
- Connected via internal switch ports:
  - Usually has external switch ports as well
- Advantage: One device to buy and manage
- Disadvantage: May not be able to view traffic between devices
  - Can make troubleshooting difficult



All-In-One Security Appliance

The other name for an *all-in-one security appliance* is Unified Threat Management device. If you have a wireless router at home that you paid at least $30 for, then you most likely have an all-in-one device right in your house.

Exactly which functionality the device you purchased happens to have will vary. But in the little 6-inch-square and 2-inch-tall device, you might find a Layer 2 switch, a Layer 3 router, a content filter, an IDS/IPS, a gateway antimalware scanner, a spam filter, a NAT device (doing port address translation specifically), a VPN concentrator, your Wi-Fi access point, and your stateful inspection firewall.

The clear advantage is that you have all 10 of those network devices inside one box that you paid very little for. If you had to purchase them all separately, it could easily cost you $1,000 or more.

The drawback is that, unless the device has the functionality built into it, you can't see the traffic between those pieces of network gear; it is all internal to that system. This makes troubleshooting virtually impossible.

Now, take that home network device and beef it up to handle enterprise-level traffic. You have a corporate-level UTM device with the same advantages and possible disadvantages. (It is more likely for an enterprise-level system to have an internal traffic monitoring capability.)

Sniffer (1)

- Captures and displays a copy of every packet going across the part of the network it is connected to
  - Can only show you packets it can "see"
  - Connect to a switch you can see: Traffic to/from your computer, broadcast traffic, multicast traffic if you are a recipient, traffic generated by the switch
- Absolute must-have for network diagnostics:
  - If you can't see what's happening, you can't fix it
- Price range: Free to several thousand dollars
- Obvious security issue:
  - Sniffing traffic such as data, username, and password

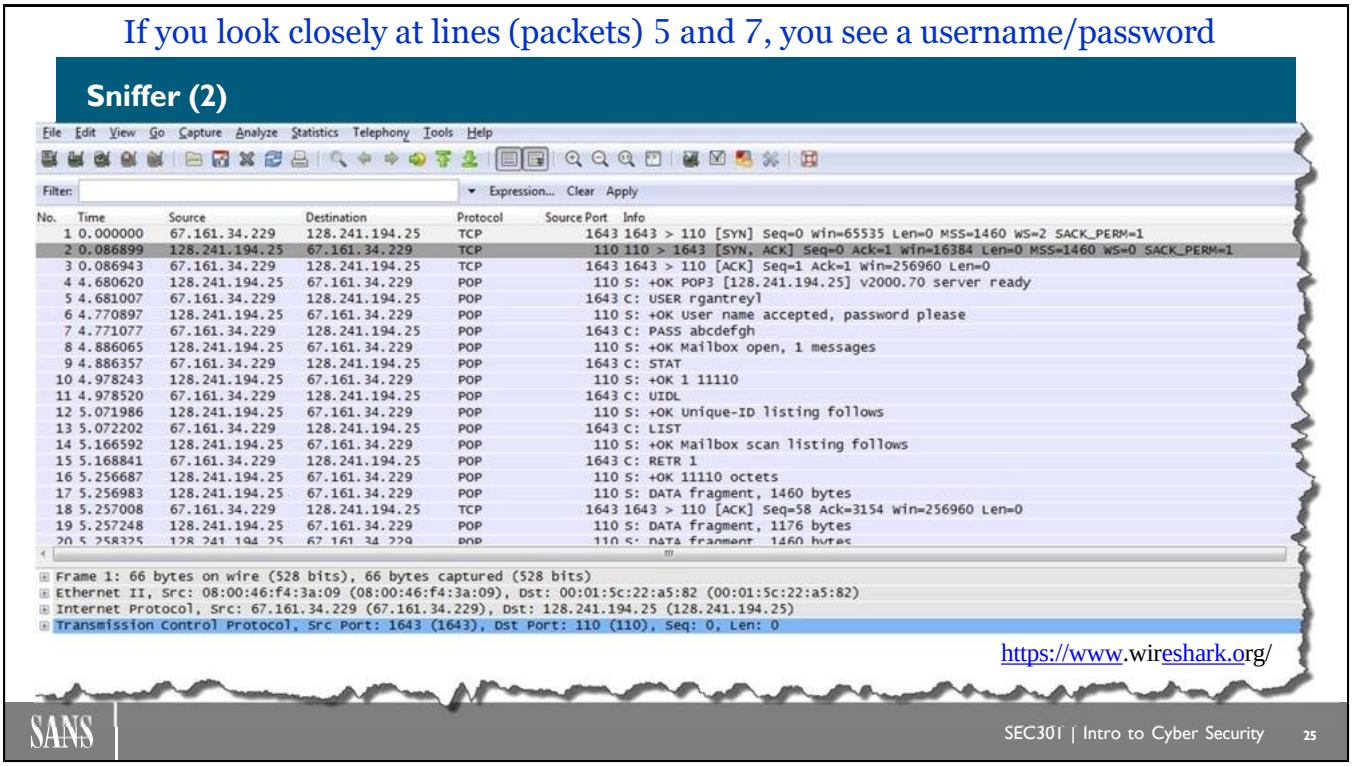
Sniffer (1)

Next, we look at a device most commonly called a *sniffer* but properly named a *protocol analyzer*. (We use the terms interchangeably because that is what the industry does; however, of the two terms, sniffer is the more common.) What the tool does is show you the packets that go across your network. After you learn to "read" the output of a sniffer, you can much more easily troubleshoot a network.

Actually, it is just about impossible to imagine trying to manage a network without a protocol analyzer. There are many times when a network administrator simply cannot diagnose and fix problems on a network without a sniffer in their list of tools.

There are also obvious security issues, which we examine on the next few slides.

In reality, when you use a tool of this nature, you are running two pieces of software. The first is the sniffer that captures a copy of the packet. The sniffer then sends it to the second piece of software, which is a protocol analyzer. That is the software that interprets the contents of the packet and displays that information to you in a more human-readable form. Wireshark, which we will talk about next, is technically the Protocol Analyzer—but *everyone* calls it a sniffer.



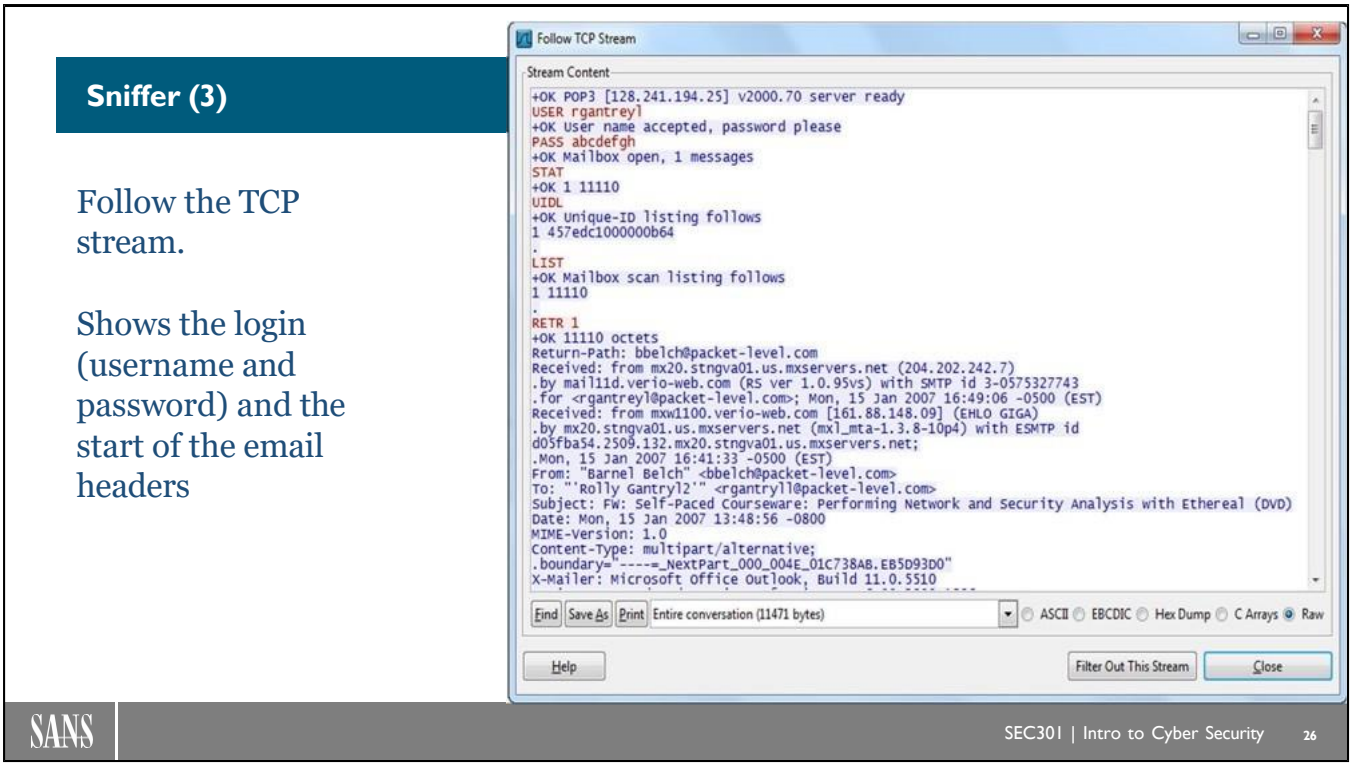
Sniffer (2)

This screenshot shows part of the screen of the Wireshark protocol analyzer. In the top portion, you see some of the information on each packet displayed on each line. (Note the lines are numbered—we refer to some of those line numbers.) Each of those lines show a summary of information about the packet, but the complete details are available farther down the screen.

This particular packet capture shows someone checking their email with the common POP3 protocol.

POP3 authenticates the user to the mail server by sending their username and password cleartext across the network. If you look above at packets 5 and 7, you see that username and password. If we were to continue to scroll down in the list of packets, we would see all the retrieved email and read this user's email quite easily. We could even reconstruct any email attachments and open them in the appropriate software.

But there is an even easier way to read this user's email. In the Wireshark software, if we right-click any of the packets in the list and from that menu select **Follow Stream**, we can see what appears on the next slide.



Sniffer (3)

This is what we see when we follow the TCP stream. The first, third, fifth, and so on lines are in blue text. The second, fourth, sixth and so on are in red text. The blue text indicates that information was sent from the server to the client. The red text indicates the information was sent from the client to the server.

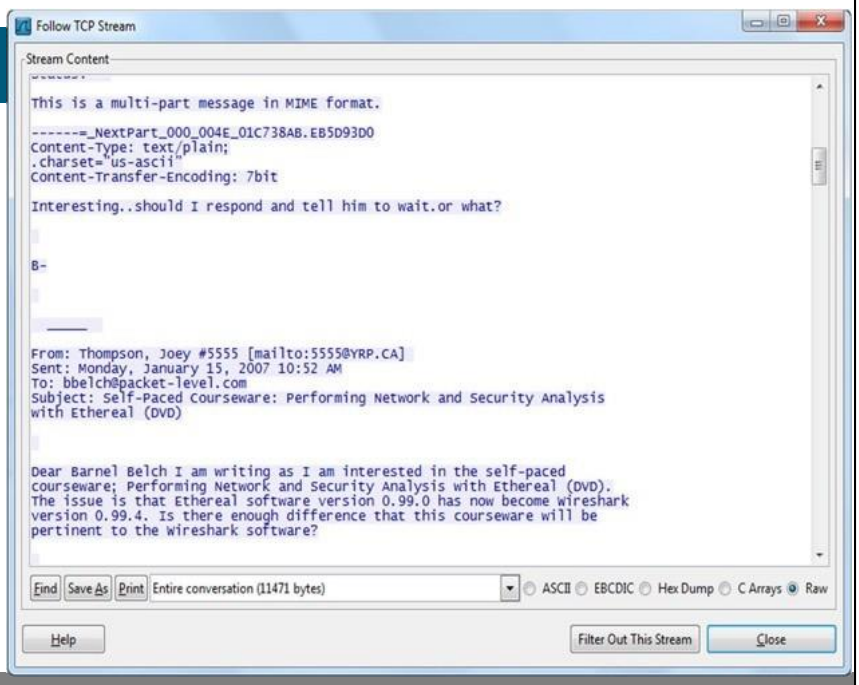
If you look at the second line, you see the username. The fourth line shows you the password. The blue +OK on the fifth line tells us that is a correct username and password combination.

As we move lower on this screenshot, the line RETR 1 is the last red text we see here. That is the command from the client to the mail server asking to retrieve the first mail message. From there, we see the blue text indicating the server is sending information back to the client. If you look closely at the last several lines of the screenshot, those are the start of the email headers.

NOTE: In this screenshot, the scroll bar is at the top of the window.

Sniffer (4)

Same window as the last slide, scrolled down to show the actual email text



Sniffer (4)

In this screenshot, notice that the scroll bar is farther down the window. We have moved farther down the information being sent by the mail server to the client.

What we see here is the actual email addresses and the contents of the email message.

Now you see why we mentioned that there was a potential security issue with sniffers. You certainly have network administrators who need to use them. Corporate policy should spell out exactly who is authorized to use them, when, and how.

You should also have a strict policy on the handling and control of packet capture files. If someone captures traffic on your network for an hour, they now have a file containing all the data that traversed that part of your network for an hour. There is a distinct possibility that there is sensitive information contained in that file. It needs to be protected accordingly.

As stated earlier in the course, users have to be notified of any monitoring. Using Wireshark or other sniffers is a form of monitoring requiring that type of notification.

Port Scanners

- Category of vulnerability scanner:
  - They simply look for open ports:
    - The fact that a port is open is a vulnerability
    - It just may be a vulnerability you can't avoid
  - ACK scan, Xmas scan, and more
- Nmap—Free and award winning—Very Goo d:
  - Good GUI: Zenmap
  - Also does OS enumeration
  - Application enumeration

Port Scanners

To attack a system remotely, there are two prerequisites to success. First, the attacker must know the IP address of the system. Second, they must know what port number they are going to connect to when they get there. Other information can be helpful to the attacker (such as OS and version, applications running, and more), but without those first two, nothing else does them any good.

So, one of the things we want to make certain of is that we have systems listening only on ports that are absolutely necessary. Where the attacker uses port scanning to find ports to attack, security staff can use a port scanner to determine what ports we can close (so they can't be attacked).

As you have probably figured out by now, a port scanner is a tool that tells us if a port on a remote system is open or closed. There are a variety of methods of accomplishing this, but they all have the same basics in common. The port scanning software sends traffic to a system and looks at the responses it receives back. For TCP ports, if the port scanner gets a SYN/ACK packet (remember the flags from the TCP header?), the port is open. If the port scanner gets an RST (reset) packet back, the port is closed.

Without question, the best free port scanner is called Nmap and its GUI is called Zenmap. This software is free and has won some well-deserved awards. It is a case of something that does one thing and does it extraordinarily well. It can do every type of port scan ever invented, then once it has discovered the system and open ports, it is able to enumerate the exact operating system and applications running on open ports.

Not only does Nmap know how to do every kind of port scan ever invented, it can also do a good job of OS enumeration. It can tell you with a high degree of accuracy exactly what operating system is running on the other end. It does this in part through passive OS enumeration techniques. Every operating system creates packets in slightly different ways, which creates a sort of fingerprint of an OS. Nmap has a database of those fingerprints. It compares packets coming back from a target to that database and tells us what operating system is running on the other end.

Nmap can also enumerate the application running on the distant system. For example, it can tell us exactly which web server software a web server is using.

Reference

http://nmap.org/

Vulnerability Scanners

- Network scanners:
  - Look at multiple systems checking for vulnerabilities
  - Looking for vulnerabilities exploitable via the network
- Host scanners:
  - Scan a single host
  - Will find things a network scanner might miss (such as missing patch)
- Both tools are *very* easy to run
  - The problem is the huge reports generated:
    - Scan a few select systems to establish a baseline
    - Fix problems on all systems
    - Then scan the rest of the systems for differences
- Both report false positives

Vulnerability Scanners

Network vulnerability scanners take port scanning to the next level. They start out by doing a port scan. After they know the open ports, they typically start trying to determine the operating system and applications on the other side of those ports. When they determine the OS and/or application, they try to determine the version and patch level. After they have that information, they compare it to their database of known vulnerabilities. So, in other words, the vulnerability scanner gives you a report that basically says, "That system is running Windows 2008 Service Pack 1, so it is probably vulnerable to the XYZ vulnerability." (Hopefully, it also tells you some information about that vulnerability and how to fix it.)

Host vulnerability scanners do some of the same things. They don't normally need to include a port scanning function, however. They do determine the operating system and application versions and patch levels. Then, using their database of vulnerabilities, they report on what they think the host is vulnerable to.

False positives are the problem with both of these tools. If the tool misidentifies the version or patch level, clearly the report will be flawed.

Huge reports are another potential problem. The author of this course once ran a network vulnerability scanner against a single class C network (254 computers). The report it generated was more than 57,000 pages. That is a completely unmanageable report size. (You can't even open the document; it is too large.)

Exploit Software

- Vulnerability scanner says:
  - That is Win2K3 at patch level 12
  - It should be vulnerable to the xyz vulnerability
  - Includes that in the report, but it could be a false positive
- Exploit software does determine the OS and patch level:
  - But launches the actual exploit to see if it succeeds before reporting
  - Little chance of a false positive
- Two well-known tools:
  - Metasploit (freeware): https://www.metasploit.com/
  - Core Impact (commercial): https://www.coresecurity.com/

Exploit Software

A newer category of vulnerability scanner is *exploit software*. Where the traditional vulnerability scanner is susceptible to false positives, exploit software is much less so.

The traditional scanner says, "That's Windows 2003 at patch level 12, so it is probably vulnerable to the XYZ vulnerability, BUT I'm not actually sure of that …"

Exploit software, by contrast, says, "That's Windows 2003 at patch level 12, so it is probably vulnerable to the XYZ vulnerability, but I'm not actually sure of that, so let's exploit it to find out for certain. If the exploit works, then we *know* it is vulnerable."

Penetration Testing

- Very common, easy sell for consultants:
  - Our experts break in before the hackers do!
  - The end of the security process, not the beginning:
    - Perform a full assessment
    - Fix problems found
    - Do Penetration Test (PenTest) to see what was missed
- The point of penetration tests:
  - You want them to succeed
  - If they don't get in, you don't learn anything
- Red teams, Tiger teams, and any other cool term marketing can dream up

Penetration Testing

Penetration testing is common; in part, it tends to be a fairly easy sell for consulting companies. When the salesman says, "Let's see if our experts can break in before the hackers do," the potential client tends to understand the point of the consulting engagement.

However, contrary to what some will tell you, penetration testing is *not* the first thing you do to secure your network. It is one of the last. Before you even consider a penetration test, you should be following much of the guidance in this course and other SANS courses to secure your network as well as possible. After you have the network as locked down as you can, then you bring in a penetration testing team to see if you missed anything. That is the role of penetration testing.

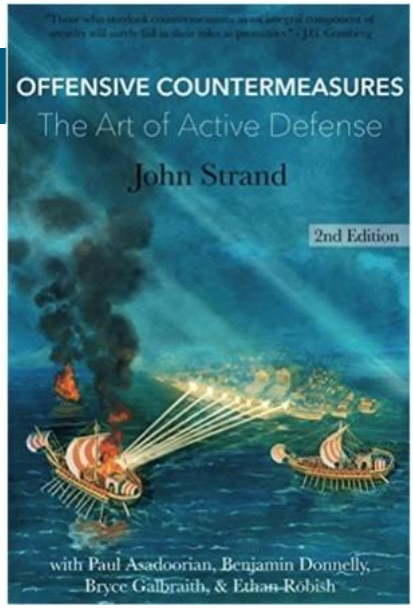
Unless you set expectations with management, hiring a penetration testing team can be a dangerous thing to do. If you hire a penetration testing team and it doesn't get into your network, you hired the wrong team. You want it to succeed. That is the only way you learn anything about the security of your network. Make sure management understands: The fact that the penetration testing team is successful does *not* mean you did a poor job of securing your network.

You will hear penetration testing teams being called a variety of terms, including red teams, tiger teams, and a bunch of other cool names. Don't let the fancy name fool you. Look at the resumes of the penetration testers to decide which team you want to hire.

NOTE: This does NOT include "hacking back"!

Offensive Countermeasures & Active Defense

- Defense with an offensive twist
- Send attackers misinformation to misdirect
  - Fake IP's, Honey Creds, Honey Systems, Honey Ports, firewall faking responses, etc.
- Also geolocating attackers
- ADHD – Active Defense Harbinger Distribution
  - A Linux distro with many active defense tools built in
- Get Legal Advice before implementing!



https://amzn.to/2UYAuzS
ISBN-13: 978-1974671694

Offensive Countermeasures & Active Defense

A newer category of Cyber Defense is to add a bit of offense to our defense. This is often referred to as either "Offensive Countermeasures" or "Active Defense". Always seek legal advice before implementing active defense measures. Depending on jurisdiction, some of these activities may not be legal.

With active defense, you typically send attackers misinformation to misdirect their activities. Some of the typical methods include:

- Fake IP addresses reported in a sweep scan. In other words, you may only have one live IP address, but the attacker just found one hundred. Which one(s) should they attack?
- Honey Credentials: You rename the Administrator account to Bob, then create a fake account named "Administrator". Any attempt to log into Administrator generates an alarm.
- Honey Systems: Often tied into the fake IP's above, there are fake systems on those IP's. In other words, it looks like the attacker has found a whole network of computers, but none of them exist.
- Honey Ports: When an attacker does a port scan, they find one hundred open ports, but only one of them is actually active.
- Firewalls faking responses: You have a firewall rule that blocks a packet, but sends the exact SYN/ACK packet the attacker would expect to see if their traffic was not blocked.

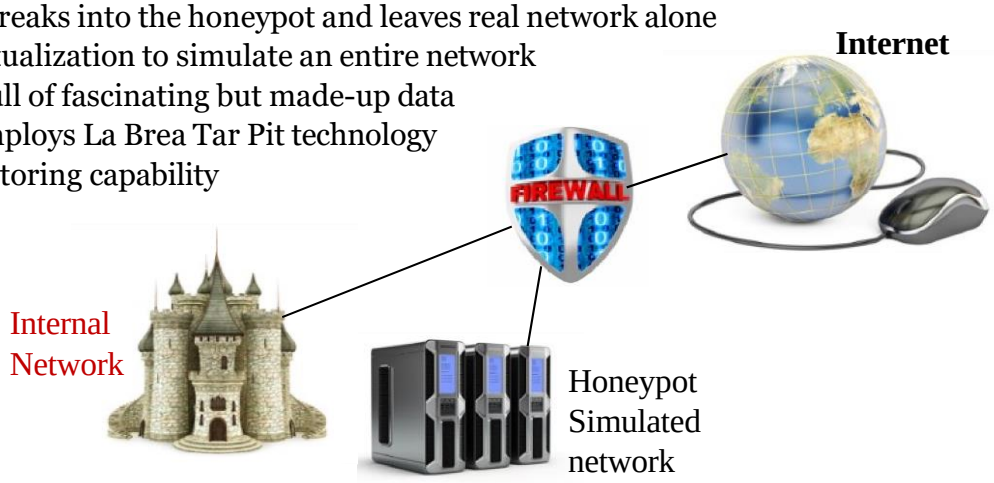
As you read through the list above, you may think to yourself, “Ok, that would be a really cool thing to do, but how do I do it?” The answer is that John Strand and his team at Black Hills Security have created a Linux distribution called ADHD or Active Defense Harbinger Distribution. ADHD has tools built into it that will do everything on the list above and much more.

NOTE: Active Defense does NOT include “hacking back”. That is both illegal and unethical.

Honeypots

- A system or systems designed to misdirect the attacker:
  - The attacker breaks into the honeypot and leaves real network alone
  - Often uses virtualization to simulate an entire network
  - Honeypot is full of fascinating but made-up data
  - Sometimes employs La Brea Tar Pit technology
  - Includes monitoring capability

This is just one type of honeypot. There are "honey systems," "honey tokens," and so on.



Honeypots

The idea behind a Honeypot is straightforward. It is an IT resource of some kind that has no valid reason for existing. Therefore, there is no legitimate reason anyone would attempt to access that resource. Therefore, if anyone does try to access the resource, it is clearly malicious. It is a form of intrusion detection.

When we say, "IT resource," we mean pretty much anything. A network, a single server, a single port on a server, a file, a database table, or anything else you decide to set up.

The basic idea is that if attackers spend all their time trying to gain and increase access to the Honeypot, they leave the real IT resources alone. Because you combine the Honeypot with an alerting mechanism, you know the attacker is there. This gives you additional time to react.

They can also provide insight into the methods the attacker is employing. For example, are they attempting to use a zero-day exploit that you were not yet aware of and therefore not protected against? If you can identify this, you can modify your defenses to protect you from the methods attackers use.

isc.sans.org – the Internet Storm Center – Continually updated threat blog & daily podcast.

Threat Hunting

- Traditional Security = "Keep the bad guys out." Still Good!
- Threat Hunting = "Assume the bad guys are in—find them"
- Example:
  - You capture network traffic and see a Command and Control (C2) channel
    - Someone has control of one of your internal systems – Indicator of Compromise (IoC)
  - Trace the traffic to the internal system—analyze to find the compromise
- Requires very good baselines of network traffic
  - Until you can define normal, it is not possible to define abnormal
- Usually relies on "threat feeds" with info on recent attacks, etc.
- A primary goal is to reduce "dwell time"
  - Time between breach and discovery

Threat Hunting

Traditional Cyber Security focuses primarily on keeping bad people off our networks. This is a good goal and one you should certainly continue to pursue. However, as stated earlier in the course, any defensive mechanism created by a human being can be defeated by a human being if they want to badly enough, have the time, have the knowledge, and have the resources. In a perfect world, our Intrusion Detection Systems would immediately alert us to the breach and we would be able to take action. In reality, IDS has proven to be far from perfect. In fact, the current dwell time (the time between the breach occurring and the time when we detect the breach) is currently around 89 days.

Threat Hunting takes the approach that we assume the bad people are already on our networks and we need to go find them. A common example of how you might find the bad actor is that you continuously monitor network traffic. When the monitoring detects an indicator of a Command and Control (C2) channel, this is a very strong "Indicator of Compromise" (IoC). In other words, someone has probably taken control of one of your internal systems. You track the network traffic to that internal system. Analyzing that system should reveal the actual breach.

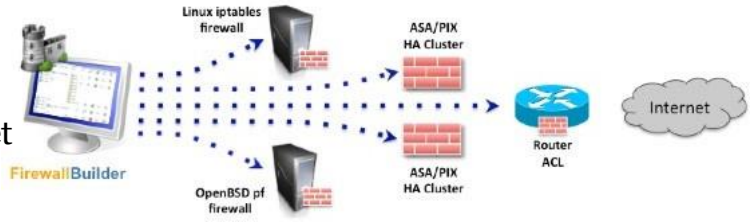
Doing Threat Hunting requires very good baselines of your network. A baseline tells you what is normal. Until you can accurately define normal, it is not possible to define abnormal. Threat Hunting also typically requires that you subscribe to one or more "threat feeds". This is a feed of information on new attacks, attack methods, and so on. There are many commercial threat feed products on the market. There is also The SANS Internet Storm Center (https://isc.sans.org). It is manned 24 hours per day and has handlers monitoring input from

over 6,000 sensors around the globe. The handlers monitor the data coming in from these sensors, analyze it, and write blog posts about what they see. There is also a daily podcast you can listen to. The SANS Internet Storm Center is a free service with a wealth of information.

Lab Time

➢ LAB 5.1: Firewall Builder:

➢ Objectives:
- Understand Firewall Objects
- Understand Firewall Rules
- Create a simple firewall ruleset
  - Focus on the Why
  - Not on the Mechanics



- Estimated completion time: 45 minutes

See the policy statements used in this lab on the next slide.

Lab Time

Lab 5.1: Firewall Builder

Policy Statements for the Firewall Builder Lab

- We must protect the firewall from attack
- DNS must function—the world can use our DNS server and we can use others
- The world can visit our web server with HTTP and HTTPS
- Our mail server can both send and receive email (SMTP)
- Our users can both send and retrieve emails from our mail server—requires secure protocols and authentication (SMTPS / POP3S / IMAPS)
- Our users can surf the web using both HTTP and HTTPS

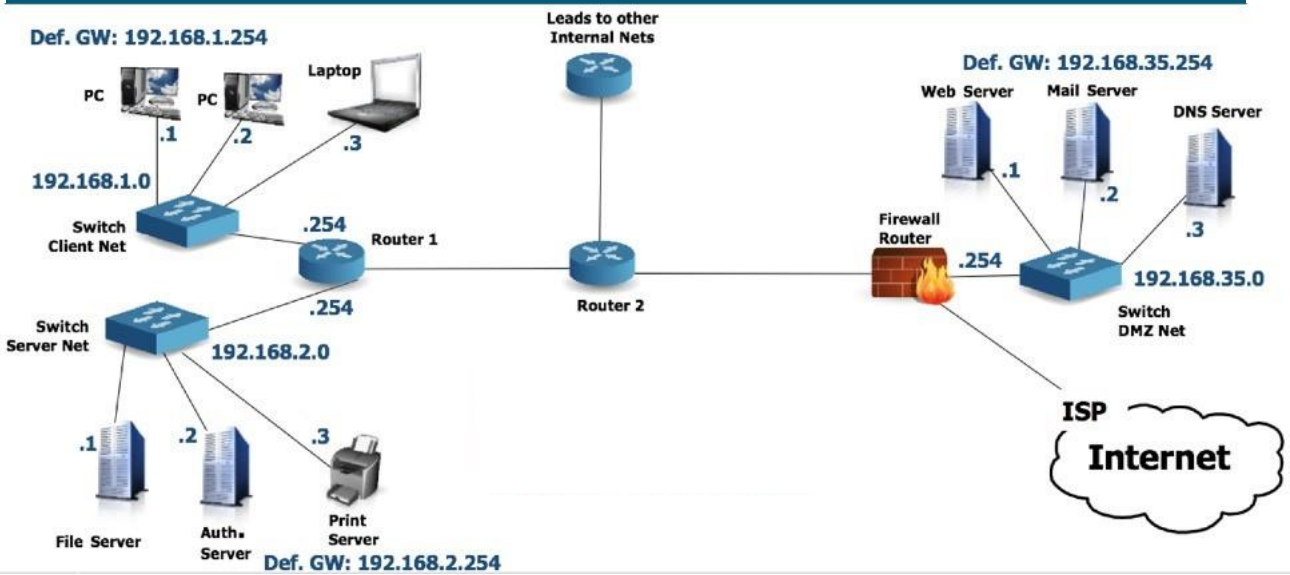
- All other activity is explicitly denied, and any such activity is logged

The network diagram for this lab is on the next page …

This page is part of the Firewall Builder lab. It is intentionally left blank.

The Solution to this lab is on the next page …

Firewall Builder Lab: Network Diagram



This page is part of the Firewall Builder lab. It is intentionally left blank.

An enlarged version of this is in the notes…

Firewall Builder Lab: The Final Ruleset

	Source	Destination	Service	Interface	Direction	Action	Time	Options	Comment
0	Any	Firewall	Any	Any	Both	Deny	Any	log	Stealth rule. Deny any attempt to attack the firewall Note: Setting allow 192.168.1.1 (admin workstation) to SSH to the firewall.
1	Any	DNS_Server	DNS	Any	Both	Accept	Any		Allow everyone (internal & external) to use our DNS server.
2	DNS_Server	Any	DNS	Any	Both	Accept	Any		Allow our DNS server to send Querries.
3	Any	Web Server	TCP http TCP https	Any	Both	Accept	Any		Allow everyone (internal and external) to visit the web page.
4	Any	Mail Server	TCP smtp	Any	Both	Accept	Any		Allow receiving email from external servers.
5	Mail Server	Any	TCP smtp	Any	Both	Accept	Any		Allow our mail server to send email out to the world.
6	Net-1	Mail Server	TCP pop3s TCP imaps TCP smtps	Any	Both	Accept	Any		Allow users to retrieve email via POP3S or IMAPS (both TLS Encrypted to the Mail Server) Allow users to send email via SMTPS (TLS Encrypted to the Mail Server)
7	Net-1	Any	TCP http TCP https	Any	Both	Accept	Any		Allow the users on Net 1 (client network) to surf the Internet.
8	Any	Any	Any	Any	Both	Deny	Any	log	Cleanup Rule - Anything not allowed above is denied and logged.

This page is part of the Firewall Builder lab. It is intentionally left blank.

Firewall Builder Lab: The Final Ruleset

	Source	Destination	Service	Interface	Direction	Action	Time	Options	Comment
0	Any	Firewall	Any	Any	Both	Deny	Any	log	Stealth rule. Deny any attempt to attack the firewall Note: Setting allow 192.168.1.1 (admin workstation) to SSH to the firewall.
1	Any	DNS_Server	DNS	Any	Both	Accept	Any		Allow everyone (internal & external) to use our DNS server.
2	DNS_Server	Any	DNS	Any	Both	Accept	Any		Allow our DNS server to send Querries.
3	Any	Web Server	http https	Any	Both	Accept	Any		Allow everyone (internal and external) to visit the web page.
4	Any	Mail Server	smtp	Any	Both	Accept	Any		Allow receiving email from external servers.
5	Mail Server	Any	smtp	Any	Both	Accept	Any		Allow our mail server to send email out to the world.
6	Net-1	Mail Server	pop3s imaps smtps	Any	Both	Accept	Any		Allow users to retrieve email via POP3S or IMAPS (both TLS Encrypted to the Mail Server) Allow users to send email via SMTPS (TLS Encrypted to the Mail Server)
7	Net-1	Any	http https	Any	Both	Accept	Any		Allow the users on Net 1 (client network) to surf the Internet.
8	Any	Any	Any	Any	Both	Deny	Any	log	Cleanup Rule - Anything not allowed above is denied and logged.

Module 15: Browser and Web Security

- Browser Security
- Executable Content
- Secure Coding

COURSE ROADMAP

- ➢ Module 14: Network Security Technologies
- Lab 5.1: Firewall Builder
- ➢ **Module 15: Browser and Web Security**
- ➢ Module 16: System Security

Module 15: Browser and Web Security

This page intentionally left blank.

Browser Security (1)

- One of the most used applications
- And now one of the most dangerous:
  - A browser downloads files to your computer; that's its primary job
  - Most of those files are benign
  - Some are anything but!
- Firewall security improved significantly:
  - Attacking network servers through a firewall is no longer easy
  - The easiest attack option became the client-side attack:
    - Trick the client into downloading the malware and compromising itself
    - Then utilize the client as a pivot to the rest of the internal network

Browser Security (1)

The browser is one of the most used applications, one of the most useful applications, and one of the most dangerous applications. At the simplest level, a browser downloads files from a server and then processes them. The idea of the processing is that it turns them into attractive web pages and displays them. But as we have asked for and been given more functionality out of our browsers, the capability for both good and bad have increased.

Understand that most of what you download and view on the web is completely benign. But some of it is anything but.

Part of the story here is that firewalls have improved a great deal from their early days. Assuming a proper configuration on a firewall, it is difficult to break through one and attack internal servers.

But you cannot prevent what you allow. When you allow your user to surf the web, you are automatically allowing them to download web content. If some of that content is malicious, it can effectively trick the client (the browser) into asking to be compromised. The client downloads a malware trojan. At that point, the attacker can take control of the internal client and use it as a jumping off point (commonly called a pivot) to attack the rest of your network.

Browser Security (2)

- Historically, default settings were more for functionality than security:
  - Some browsers are more secure out-of-the-box than others:
    - Internet Explorer used to support more unsafe protocols:
      - No longer true with IE9, but that is only for Win7 and later
    - All now incorporate configurable pop-up blockers and cookie managers
- Malicious add-ons (aka browser helper objects, extensions, or plugins):
  - Add-ons extend the browser's capabilities: Many are good to use
  - Now a target of attackers: Many are Trojans / Malware

Browser Security (2)

This has improved a great deal in the last few years, but it is still true that most browser default settings have more to do with functionality than security.

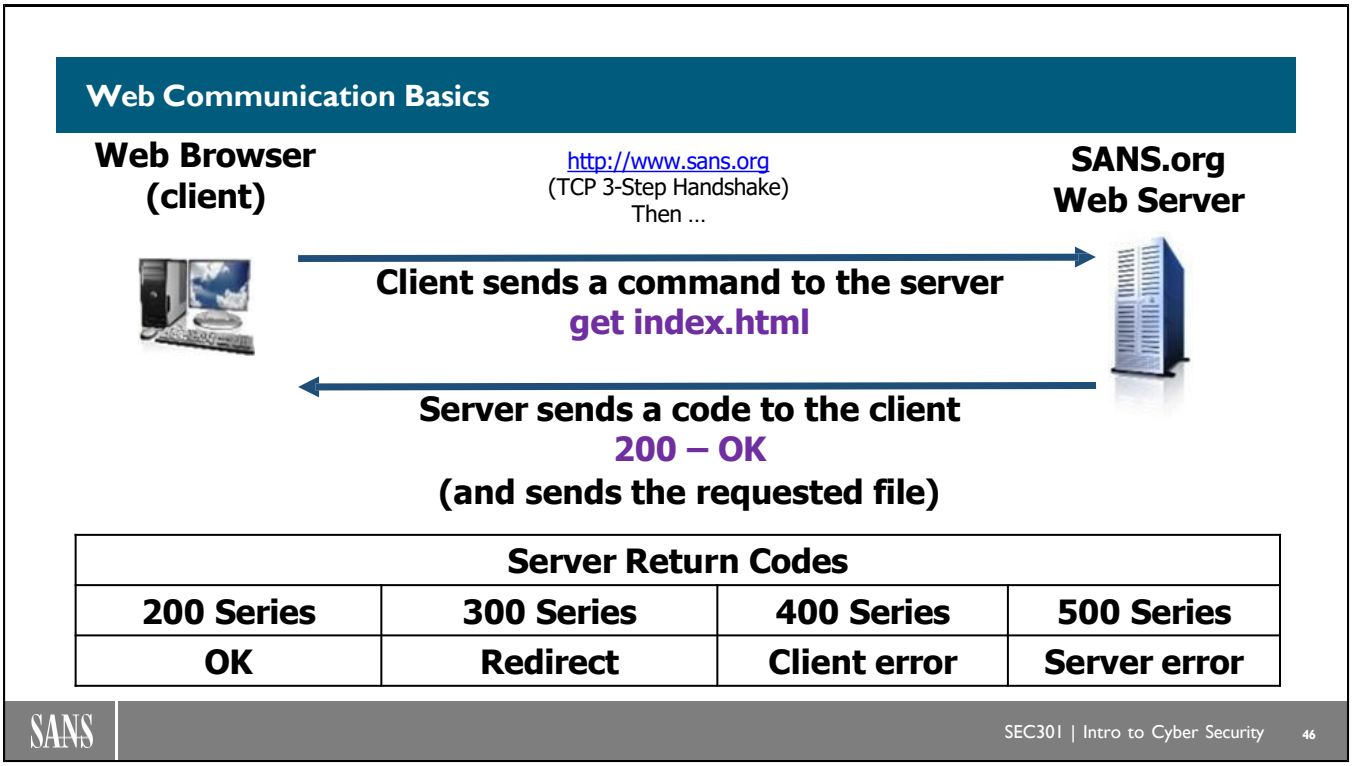
We are about to examine the cookie in detail. These small text files can greatly enhance our browsing experience. They can also be used to track our surfing activity, our purchases, etc.

We also look at the security features of recent releases of some of the more common browsers. However, no matter how good the security features of your browser, if you choose to install the wrong browser add-on, you can potentially defeat every security feature of the browser. Many of these add-ons now fall squarely into the category of malware. (Note: The terms *browser add-on, add-in, plugin,* and *browser helper object* are all synonymous.)

You can scan all your browsers for free by going to the following link. When you do, Qualys asks to install a browser plugin. Let it do so. (This one is safe.) Thereafter, every time you visit that link, it checks to see if you have malicious or outdated plugins. You can then click a link to either disable or update the plugin.

Reference

https://browsercheck.qualys.com/



Web Communication Basics

When a computer user opens a web browser and enters a URL, the user sees a web page appear in that browser. What most computer users do not realize is that there is A LOT of activity that happens to get that web page into that browser. And yes, we do mean A LOT! On the next few pages (and in the upcoming demo), you will get some idea of exactly what we mean by "A LOT" in this particular context.

First, HyperText Transfer Protocol (HTTP) is the protocol used to transfer a web page and all of its elements from the server to the web browser (or client). HTTP requires the use of Transmission Control Protocol or TCP that we learned about a few days ago.

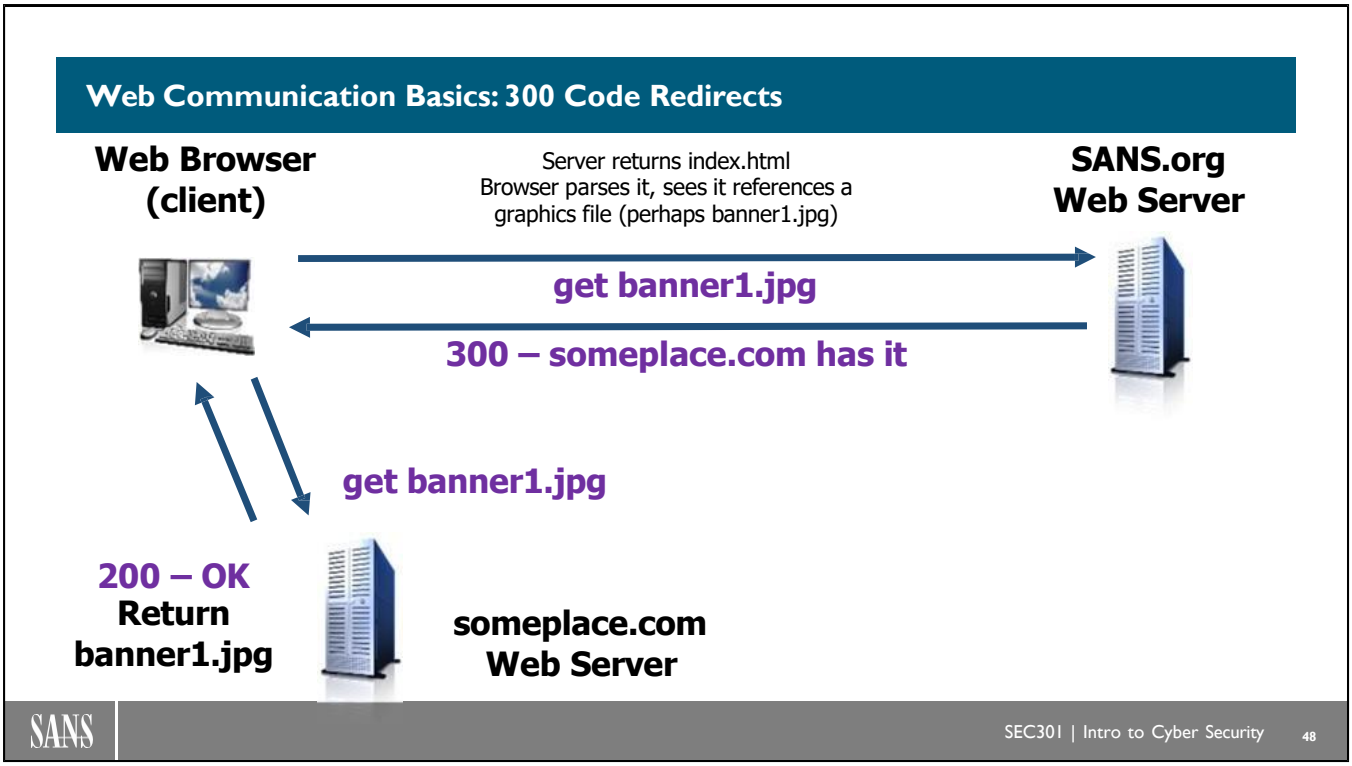
So, when a computer user opens a web browser and types a URL such as http://www.sans.org, the first thing that happens is the TCP three-step handshake. Once that is complete, the client and server have a fully established TCP session and can begin communicating.

Next, the browser sends a command to the server. If the URL line does not specify another filename, the default command is "get index.html." When the server receives that command, it returns a code. For example, the server hopefully returns the code 200, which technically means "OK," or nontechnically, everything is working as expected and the server is going to return the file requested by the browser.

Before we look at the different codes, it is essential to know two facts. The client always sends the command to the server. The server always sends a code back to the client. The server never sends commands, and the client never sends codes.

The codes and their definitions are:

- 200—Command OK: Everything is working fine.
- 300—Redirect: We will look at this one on the next slide.
- 400—Client error, e.g., 404 File Not Found: You asked the server for a file, and it does not have it or know where to find it. Another critical 400 series code for security practitioners is 403 Access Denied, meaning someone attempted to access a directory they do not have access to. Excessive 403 codes may indicate someone attempting unauthorized activity on a web server.
- 500—Server Error: You asked the server to perform some action, it attempted to do so but failed.



Web Communication Basics: 300 Code Redirects

For our discussions in this class and for the upcoming demo, the 300 series codes require additional explanation. In addition to knowing this for the class and demo, this is vital information to understand when attempting to secure and control web traffic.

When the client browser receives the index.html file, it begins parsing (or analyzing) the code contained in that file. At some point in the code, it references a graphics file (this might simply be a graphic on the web page, but it is also often a banner ad of some sort).

When the browser sees the reference to the graphic, it knows it has to retrieve that graphic in order to display the web page correctly. It therefore connects back to the web server and requests the graphics file (in the slide above, we are calling it "banner1.jpg"). If the server possesses the file, it will return it to the browser as you would expect. However, if the server does not have the file, but knows that it is located on a different server, it will return a 300 Redirect code along with the address (in the slide, someplace.com) of the server that has the file. When the browser receives the 300 code and the new address, it establishes a connection to that server and retrieves the graphics file from there.

Numbered paragraphs in the notes match the numbers on the slide. Direction Key: Both: ↔ From Client: → From Server: ←

Web Communication Basics: The Full Story

Web Browser (client)

http://www.sans.org

There are often more steps not shown …
A four-step teardown after step 4.
A three-step handshake before step 5.
A four-step teardown after step 6.

Step #	Direction	Server	Function / Command / Code
1	↔	DNS	DNS Lookup & reply for sans.org
2	↔	sans.org	Three-step handshake with sans.org
3	→	sans.org	get index.html
4	←	sans.org	200 – OK Send index.html to the browser
5	→	sans.org	get banner1.jpg
6	←	sans.org	300 redirect someplace.com
7	↔	DNS	DNS Lookup & reply for someplace.com
8	↔	someplace.com	Three-step handshake with someplace.com
9	→	someplace.com	get banner1.jpg
10	←	someplace.com	200 – OK Send banner1.jpg to the browser
11	→	sans.org	get banner2.jpg
12	←	sans.org	300 redirect to anotherplace.com
13	↔	DNS	DNS Lookup & reply for anotherplace.com
…	This process may continue 100 times or more for each web page you load.		

SANS SEC301 | Intro to Cyber Security 49

Web Communication Basics: The Full Story

With an understanding of the basic process of an HTTP redirect, let's take a moment and look at the full, detailed process. Note that something of this sort happens each time you load a single web page. You see the page appear in the browser. These are all the steps happening behind the scenes.

You open a browser and type http://www.sans.org into the URL line and press the enter key:

1. Your computer sends a DNS query to the DNS server asking for the IP address of sans.org, and the DNS server returns that IP address.
2. The web client completes the TCP three-step handshake with the sans.org web server.
3. The browser sends the "get index.html" command to the sans.org web server.
4. The sans.org server responds with a 200 code and the index.html file to the web browser.
5. The browser parses the index.html file and sees a reference to a graphics file it will require in order to display the web page. The client does another three-way handshake with the server and sends a "get banner1.jpg" command.
6. The sans.org web server sends a 300 code to the client. Along with the 300 code, the server sends the address where banner1.jpg can be obtained. In this case, that address is someplace.com.
7. The web browser sends a DNS query for someplace.com and receives that server's IP address back.
8. The web browser initiates a TCP three-step handshake with the IP address of the someplace.com web server.

9. The web browser sends a "get banner1.jpg" to the someplace.com server.
10. The someplace.com web server sends a 200 OK code back to the web browser along with the banner1.jpg file.
11. Meanwhile, the browser has continued parsing the index.html file and sees a reference to a second graphics file (banner2.jpg). The browser will require that file in order to properly render the web page. So, the browser sends a "get banner2.jpg" to the sans.org web server.
12. The sans.org web server replies with a 300 code and an address where banner2.jpg can be found: anotherplace.com in this case.
13. The web browser sends a DNS query to the DNS server and receives the IP address of anotherplace.com's web server.

Every time you load a web page, this process repeats anywhere from a handful of times to 100 or more. It depends on the graphics depicted on the page and where they are located. Sites that have a lot of banner ads is where you will see this the most.

Also, note that we still have not shown every possible step (we ran out of room on the slide). In most cases, there would be a four-step TCP teardown after step 4, and a TCP three-step setup before step 5, and another four-step teardown after step 6, and so on.

For easier reading, the table containing the steps is on the following page:

Step #	Direction	Server	Function / Command / Code
1	⟷	DNS	DNS Lookup & reply for sans.org
2	⟷	sans.org	Three-step handshake with sans.org
3	→	sans.org	get index.html
4	←	sans.org	200 – OK Send index.html to the browser
5	→	sans.org	get banner1.jpg
6	←	sans.org	300 redirect someplace.com
7	⟷	DNS	DNS Lookup & reply for someplace.com
8	⟷	someplace.com	Three-step handshake with someplace.com
9	→	someplace.com	get banner1.jpg
10	←	someplace.com	200 – OK Send banner1.jpg to the browser
11	→	sans.org	get banner2.jpg
12	←	sans.org	300 redirect to anotherplace.com
13	⟷	DNS	DNS Lookup & reply for anotherplace.com
...	This process may continue 100 times or more for each web page you load.		

What Are Cookies?

- A small text file downloaded to your computer from a web server:
  - It can only contain a small amount of text, usually a "unique identifier"
  - They CANNOT contain software, images, etc.
  - Only the site that places a cookie can read that cookie
- Two basic types:
  - Non-persistent: Stored in RAM only (not on the hard drive)
  - Persistent: Stored on the hard drive
- Cookie Repositories:
  - If you have 5 browsers, you have 5 repositories: One per browser
  - In other words, 5 directories containing cookies: One per browser

What Are Cookies?

Many people misunderstand cookies. Wild rumors have circulated for years that cookies could install software on your computer, compromise your system, and contain images. None of this is true or accurate.

In reality, cookies are just small text files placed on your computer by a web server. They can only contain a small amount of text, which in most cases is a "unique identifier." This is how some websites can greet you by name when you visit the site, even when you have not authenticated (more on that in a moment).

There are two basic types of cookies, persistent and non-persistent. Each performs a different role.

Non-persistent cookies only reside in the Random Access Memory (RAM) of your system (they are not on the hard drive). Commonly, these cookies do temporary tracking of activity, such as your progress through the checkout process at an E-commerce site.

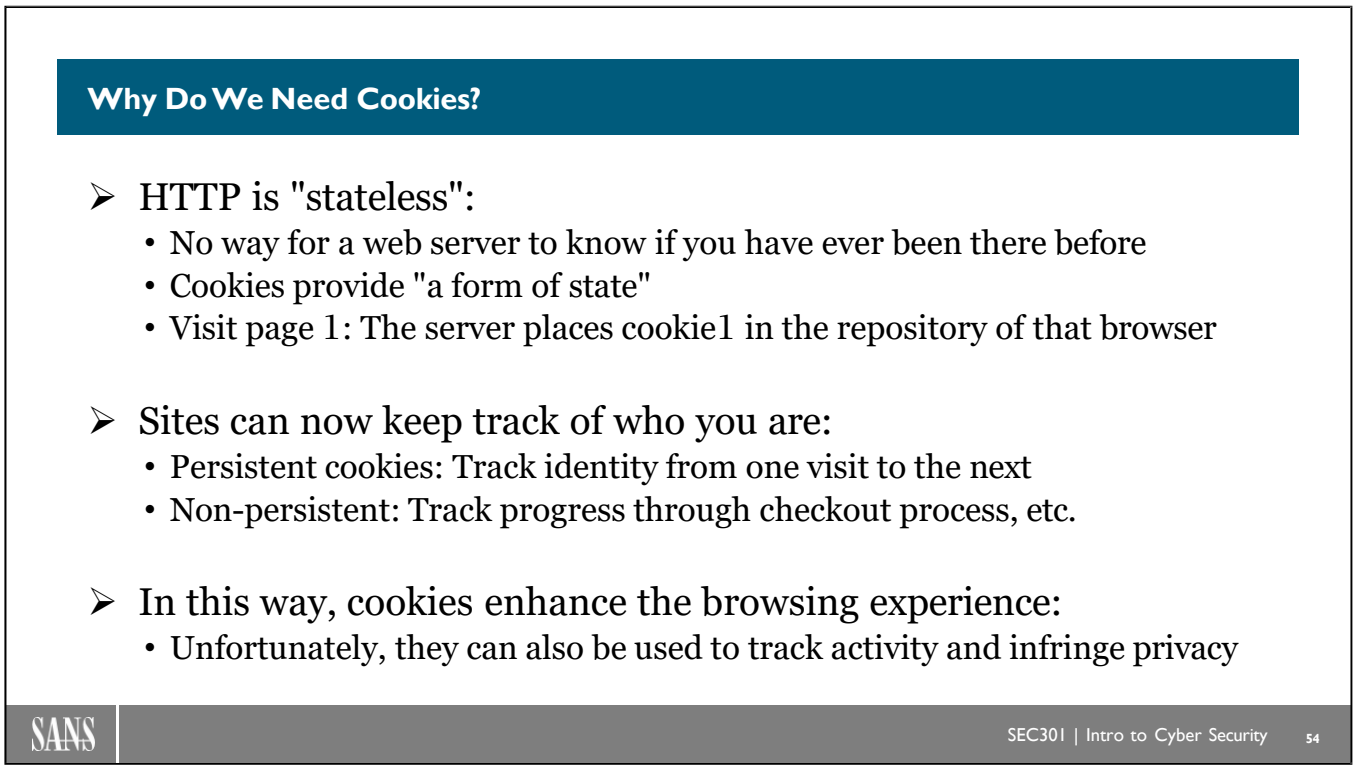
Persistent cookies go onto the hard disk drive of your computer. Each browser you have installed on the computer has its own, independent cookie repository. For example, if you have five browsers installed on the computer, there are five directories on the computer as well. Each directory is a cookie repository for one of those browsers (e.g., one for Internet Explorer, one for Google Chrome, one for Firefox) These persistent cookies most commonly provide long-term identity tracking.

You then click to access your shopping cart and begin the checkout process. Amazon.com reads their persistent cookie to know who you are (so they show your shopping cart info and not mine). They then read the two non-persistent cookies, get the unique codes from each, and show you Product A and Product B in your shopping cart. As you move through the rest of the checkout process, Amazon continues to place non-persistent cookies indicating where you are in the process.

Once you complete the purchase, all of the non-persistent cookies are (usually) removed from RAM. The persistent cookie from Amazon.com remains in your Chrome repository. (Note: if the non-persistent cookie is not removed from RAM by the server, it will typically remain in RAM until you close the browser.)

Reference

https://us.norton.com/internetsecurity-privacy-what-are-cookies.html



Why Do We Need Cookies?

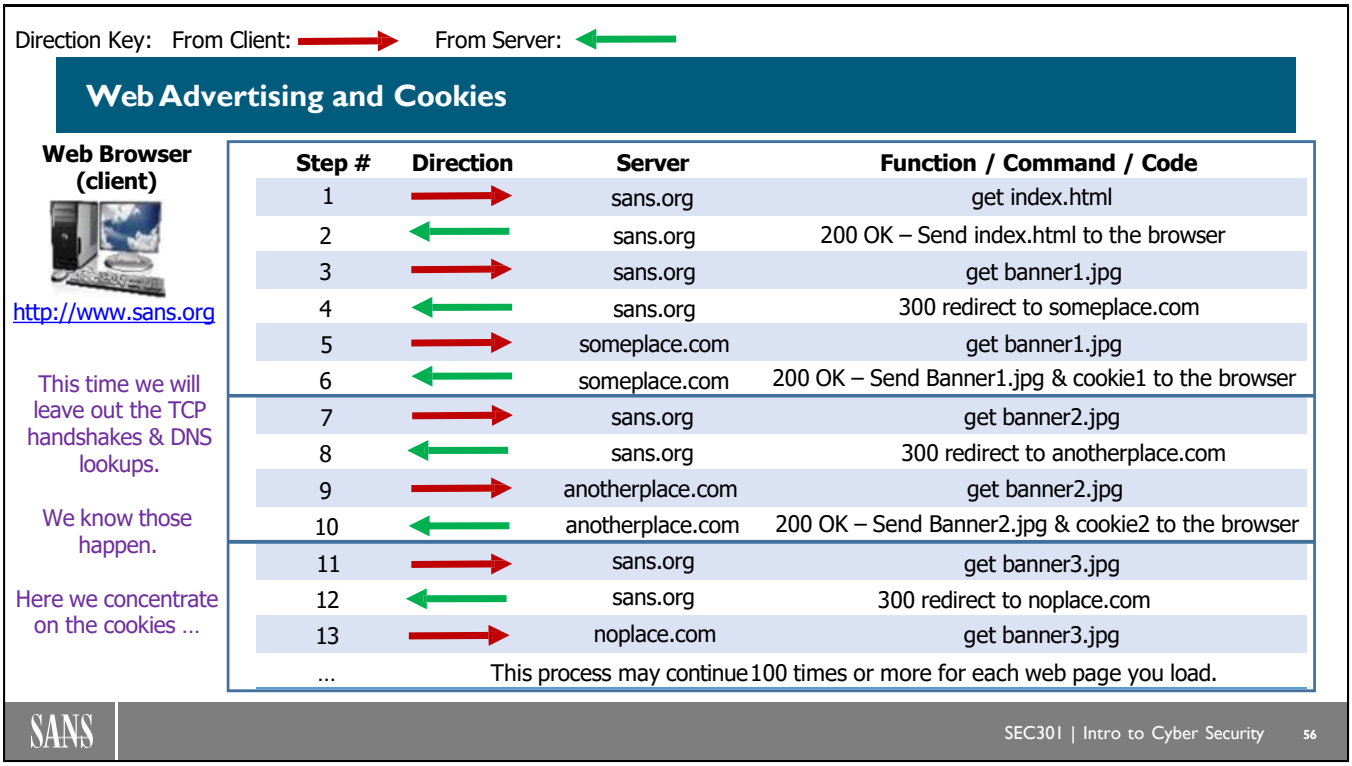
The HTTP protocol (the protocol used in web transmissions) is said to be stateless. By *stateless*, we mean that each transaction is an independent unit with no relation to any transactions that come before or after it. When you request a web page, your computer connects to the server, gets the page, and then closes the connection. The next time you request a page, your computer makes a new connection to the server. That server does not know or realize that you are the previous visitor.

The lack of state in web communications presents a real problem for an application like E-commerce. Sites such as Amazon want to recognize their customers from one visit to the next. Those sites also require a method of tracking user activity during the same visit, such as, while a customer is going through the checkout process.

For example, you use Chrome to visit Amazon.com and log in to your account. When you do, Amazon places a cookie in the Chrome cookie repository on your local hard drive. That cookie contains a unique identifier generated by Amazon. In Amazon's database, that unique identifier equates to your account. Once this is in place, you can close the browser, turn off your PC, and so on. But the next time you use Chrome on that same computer to visit Amazon, they read the unique identifier in the persistent cookie, look that identifier up in their database, and say hello to you on their webpage.

While you are shopping at Amazon, you click to purchase Product A. When you do, Amazon's web server places a non-persistent cookie in the RAM of your computer. That non-persistent

cookie contains a unique identifier that Amazon's database equates to Product A. You then click to purchase Product B. Amazon places a non-persistent cookie with a code indicating Product B. Note that each time you move from one page to another on their site, Amazon is also reading the persistent cookie in Chrome's repository to keep track of who you are.



Web Advertising and Cookies

To understand how the cookie works with web advertising, you have to first understand the redirects that we covered a few pages back. Many (almost all) of the banner ad providers place a cookie in the browser's cookie repository. That cookie often contains a numeric value that indicates to the banner ad provider that you had visited a particular site (sans.org in the example above).

If you look at the slide, three different banner ad providers are sending both banner ads and cookies to the browsing system. Each of the cookies indicate to that banner ad provider that this individual visited sans.org. This sounds pretty innocent, until you extrapolate out over time what is happening.

Let's say that in the course of an hour, the person sitting on the computer above visits 100 different websites of interest to them. At each of those websites, those three banner ad providers provided the banner graphics, placed cookies indicating that individual was at that site, AND read all the other cookies they had placed indicating all the other sites the person visited. At the end of the hour, those three banner ad providers would know every web page this individual has visited in that time.

What would the banner ad providers do with that information? Sell it to advertisers! So, if those 100 sites all dealt with Cyber Security, then the individual doing the browsing would start seeing banner ads on other websites dealing with Cyber Security, would start receiving junk mail dealing

with Cyber Security, and even start getting calls from telemarketers dealing with Cyber Security-related things. Yes, in case you have not noticed yet, since you signed up for this class at https://www.sans.org, you will now start seeing banner ads about a company called SANS. We use this method of advertising as well.

Reference

https://www.theguardian.com/technology/2012/apr/23/cookies-and-web-tracking-intro

Leaking Browser Information

- Referrer site
- Browser type
- OS and version
- Cookies available?
- Screen dimension and depth
- Active code runnable?
- Countermeasures:
  - Proxy servers
  - Browser security settings

Leaking Browser Information

When you click a link on a web page (or enter a URL in the browser window), your browser transmits information to the site you are contacting. To begin, your IP address (the IP address of your proxy or Network Address Translation [NAT] device) gets sent with every packet. This exchange is a part of the normal TCP/IP communication, and it can be used as part of a larger information-gathering effort. To get more insight into what can take place, browse to the http://mybrowserinfo.com/ and click "See Detailed Location and Browser Information"[1] to see the large amount of information that a web server can get about you, your system, and your environment. Among the information gathered (and why it is important) is the following:

- Whether your system accepts cookies. This information tells the servers whether they can store information that can trace your movements around the site or around the web.
- The website you were on just before you got to mybrowserinfo.com.
- Your browser type and operating system. Someone can use these facts to launch specific attacks against your system.
- Whether you have JavaScript, VBScript, or Java enabled. Perhaps attackers can get malicious code to execute on your system.
- Your screen dimensions and color depth. Websites can use this information for displaying
  pop-up ads.

Reference

[1] http://mybrowserinfo.com/

Currently, only IE 11 is officially supported. Internet Explorer is at "End of Life".

IE 9+ Security Enhancements

- SmartScreen Filter:
  - Watches for suspicious activity by website
  - Checks sites against a dynamic list of reported phishing/malware sites
  - Application reputation
- URL filtering improvements
- ActiveX filtering
- Cross-site scripting (XSS) attack protection
- Tab isolation
- In-private browsing
- Stores password encrypted in the registry

IE 9+ Security Enhancements

Internet Explorer has a long, sad history of exploits. This is especially true with older versions. The good news is that newer versions (anything since Internet Explorer v9) have robust security features. Unfortunately, those features rely on things that are only available in Windows Vista and later. Therefore, Internet Explorer 9 and later will not run on Windows XP.

In Protected Mode, Internet Explorer runs with low permissions. So even if you are logged in as an administrator, IE does not have that level of permissions. The first feature to be aware of is the SmartScreen Filter. This sends every URL you enter to the Microsoft cloud to be checked against a list of known malicious sites. If the URL matches one of those bad sites, it blocks you from going there until you approve it. This feature also watches for suspicious activity by websites. For example, it actively blocks attacks that fall into the Cross-Site Scripting category.

There is Tab isolation, meaning that it is difficult for something malicious in one open tab to interfere with something in another open tab. It also has improved InPrivate Browsing, which after you enter that mode, any browsing history, cookies, and so on are deleted when you close each InPrivate tab. That information can still be recovered forensically but is not readily available.

Finally, if you ask it to remember your password for a website, it now stores passwords in a private encrypted area of the registry. Older versions of Internet Explorer stored those in world readable ASCII text files.

Microsoft Edge Browser

- Introduced in Windows 10 (will replace IE)
  - Originally called "Spartan"
  - Does not support many dangerous functions
  - Only allows a limited # of plugins
    - (237 as of 04/2021: Many are security plugins)
- Like Internet Explorer:
  - Runs with limited permissions
  - SmartScreen filter to block XSS
  - Tab isolation
  - "InPrivate" browsing mode
  - Secure password storage

Note: In April 2021, Microsoft released a beta of Edge now based on the open source Chromium browser engine—The same browser engine used by Chrome and Opera. It is currently too early to tell just what that will mean—except we know everything about Edge will probably change. (No update as of August 2021)

Microsoft Edge Browser

Microsoft created a browser called "Spartan," which was a perfect name for this browser because it was indeed very spartan in its design. It does not support many of the advanced features found in other browsers. Since these are features that very few people know about or use, and since these features are a primary attack vector against browsers, leaving those elements out created a more secure browser. The Spartan browser was part of Windows 10, but two days before the release of Windows 10, Microsoft changed the name of the browser from Spartan to Edge.

The Edge browser has all of the features of Internet Explorer (when IE is running in the default "Protected Mode"). As stated above, it also does not support many of the advanced but dangerous features that other browsers support. Finally, it only allows a limited number of browser extensions (101 as of September 2018). Microsoft has vetted each of those extensions.

When you add all of these security features (and lack of dangerous features) together, you can make a strong case that Edge is the most secure browser available today. Many will disagree with that statement, but the argument is solid nonetheless.

Firefox Security

- Anti-phishing and malware protection
- Secure software installation
- Antivirus integration:
  - Depends on your AV package
- Private browsing
- Do not track settings per site
- Encrypts password storage (and can be password-protected)
- Many security add-ons available

The spring 2021 version (v67) enhanced privacy protection GREATLY!

Firefox Security

For some time, Firefox took the approach that if you wanted a secure browser, you needed to do your own research and install plugins that would do security tasks for you. It builds few security features directly into its browser.

This is changing rapidly. Every version that comes out has more robust security features included. Its browser now includes anti-phishing and some malware protection. If you have the right anti-malware package, it integrates closely with it. Like Internet Explorer, Firefox has robust Private Browsing to remove your tracks while browsing.

It also defaults to sending Do Not Track requests to websites. This is a polite request to a website not to use cookies or other means of tracking surfing activity. The website does not have to honor the request. (Internet Explorer and Google Chrome have followed suit and now send Do Not Track requests as well.)

Firefox encrypts any password storage, and you can password-protect access to that encrypted database.

One advantage of Firefox is that there are many security plugins available. Just know that some of them make actually surfing the web difficult.

Google Chrome Security

- Anti-phishing and malware protection
- Browser runs with extremely limited permissions:
  - Each page is provided limited permissions as well (see sandboxing)
- Tab sandboxing:
  - Each page is given its own rendering process
- Stores password in encrypted SQL database
- Many security add-ons available
- Over 50% market share

Google is in the process of stopping extensions from blocking adds.

Google Chrome Security

Like Internet Explorer and Firefox, Google's Chrome browser has robust anti-phishing and some malware protections. The browser runs with limited permissions regardless of how you log in to the system.

Chrome includes strong sandboxing of tabs, meaning that it is difficult for maliciousness in one tab to interfere with another tab.

Any stored passwords are stored in an encrypted SQL database, which you can also password-protect. Like Firefox, there are many security plugins for Chrome.

Safari Security

- Security improving rapidly
- Private browsing
- Third-party cookie control
- Tab sandboxing
- V11 and later:
  - Intelligent Tracking Prevention
  - Reader mode (shows text content only)
  - Per site settings
- Now updated more frequently

Safari Security

The built-in Mac browser (no longer available on Windows) is called Safari. Historically, it lagged behind other browsers in the security department. That is changing rapidly as Apple adds more security features, particularly in version 11 released in November 2017.

Safari now has all the security features you would expect to find, including tab sandboxing, private browsing mode, etc. It also includes third-party cookie control, which most other browsers do not have.

Also, with version 11, Apple added "Intelligent Tracking Prevention" to make it more difficult for websites to track your surfing activities. They also included "Reader Mode," which, for some websites, will only load the text of the site without all of the graphics, and so forth. This will make browsing both faster and safer.

Finally, in version 11, Apple included the ability to do per-site configuration settings. In other words, you can set Safari to Always load site XYZ.COM in reader mode. Of course, you can override these settings at any time if you need to.

Historically, Safari was not updated as often as other browsers. Apple has now started updating the Safari browser on a much more frequent basis.

Active Content: Making the Web More Interactive

- Traditional method: All processing done on the server
  - Highly inefficient for busy sites
- Newer method: Most processing done on the client
  - Code downloads from the server to the client and executes there
  - Huge efficiency savings
  - Potential danger to the client
- Common forms:
  - JavaScript
  - Java applications and applets
  - ActiveX controls

Active Content: Making the Web More Interactive

In the old days, web servers basically delivered static (or slowly changing) information to users. The basic operation was an endless round of:

1. The browser requests a page of text.
2. The server sends the page of text.
3. Go to step 1.

That process worked well for a while, but soon users started wanting more interaction from their servers, and web developers wanted to give it to them.

Enter active content. *Active content* is a term for program code that is embedded in the contents of a web page. When a web browser accesses the page, the embedded code automatically downloads and executes on the user's workstation. Other terms sometimes used to describe active content include *executable content*, *active code*, or *mobile code*.

Active Content Risks

- Runs on the user's system and is the foundation for an interactive experience:
  - Software can do anything you can do
- The balancing act:
  - Risks versus gain
  - Security versus convenience
- Countermeasures:
  - Use selectively if possible
  - Firewall and proxy filtering

Active Content Risks

What do Java applets and ActiveX controls do? Well, almost anything their designer can dream of. Active content can be extremely useful for unleashing the power of network communications, but these technologies have a lot of security risks. Because they run locally on your computer, they can often do almost anything you can do as an ordinary user.

Of course, these technologies claim to have security mechanisms to prevent evil things from happening, but almost all of them have had flaws or bugs that allowed bad things to happen anyway. If running these types of programs is risky, why do it at all? Well, because they are so cool! That's basically what it comes down to. Many of these programs do incredibly useful, valuable, or entertaining things. The value they might add to an application or a website is extremely high to users. This fact points to one of the classic trade-offs in security: Balancing the benefits of an action against the risks of that action. You've probably heard of the experiment where the mouse got an electric shock every time it tried to get some cheese. Despite the risk, the cheese benefit was so important that the mouse was willing to risk getting shocked.

It would be easy to tell people, "Turn off all active content in your browser," and many security people do just that. But that demand might be unrealistic in today's world. Many websites inside and outside your organization use Java and ActiveX to perform useful or necessary services.

To block access to them would mean a serious productivity loss to your users. What might be a more realistic approach is to turn off active content for most sites. This can be accomplished

through your organization's firewall or proxy server, or it can be programmed into the personal firewall or proxy service run on your end user systems. When you have a site that you need or want access to, turn it on temporarily for that site. Then, turn it off again when you leave the site. It's a bit cumbersome, but this process does offer a nice balance of security and convenience.

JavaScript – 1997 --- HTML5 – 2014

JavaScript & HTML5

- JavaScript = Lightweight but powerful scripting language
  - The script instructions are embedded in the web page
  - Executed by the browser while displaying the page
- JavaScript is heavily used throughout the web
  - 95.2% of all websites use JavaScript (includes all those below)
- HTML5 includes JavaScript as part of its functionality
  - HTML5 is used on 78.5% of all websites and is increasing in popularity rapidly
  - Eventually, all sites using JavaScript will transition to HTML5
  - Gmail, YouTube, Facebook, Amazon, etc. all use HTML5 (and therefore JavaScript)

HTML: Hypertext Markup Language – the code used to create web pages

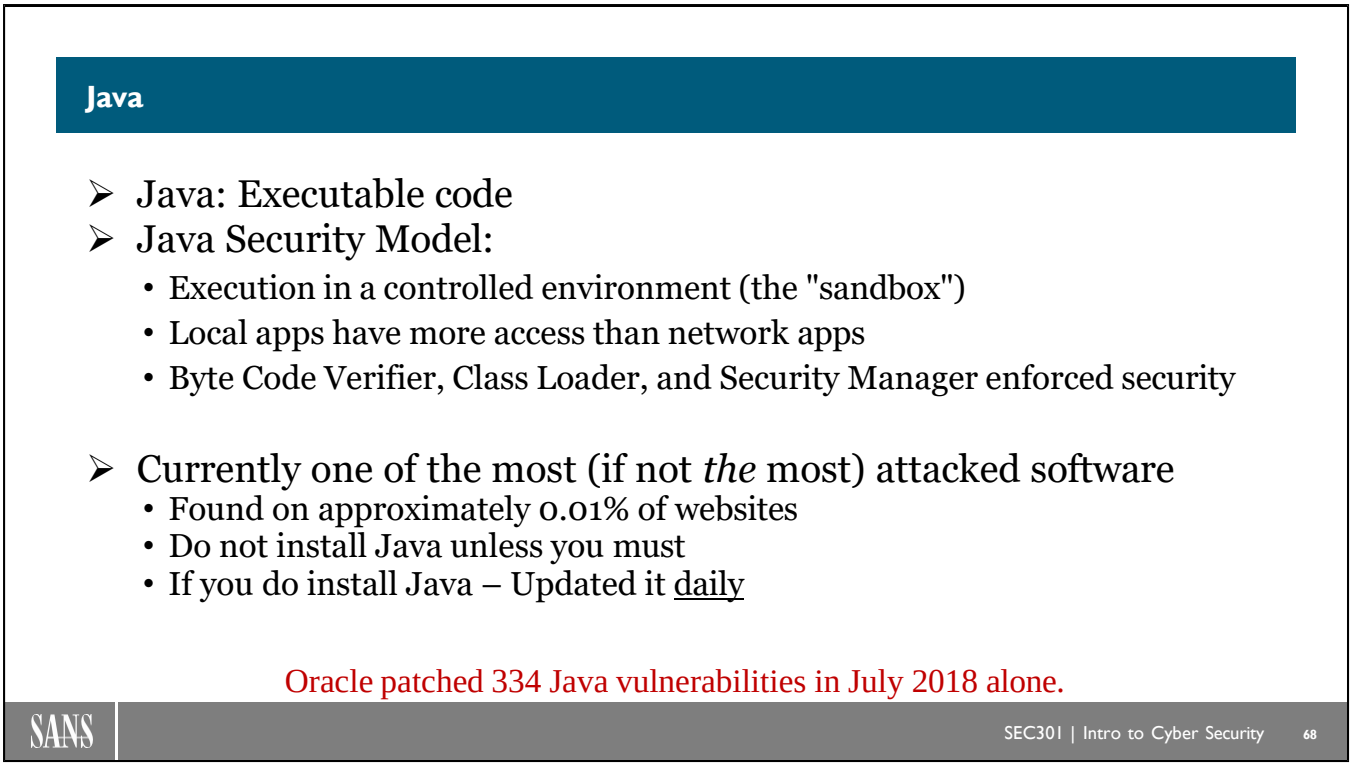
JavaScript & HTML5

The most popular web scripting language by far is JavaScript. It is a very lightweight scripting language, meaning that it does not require a lot of computing power to provide a ton of functionality. Despite this fact, it is extremely powerful. Released in 1997, it has been around for a long time and has powered a significant majority of websites. In fact, as of July 2021, it powered 95.2% of all websites. Highly interactive sites such as Gmail, YouTube, Facebook, Amazon, and a long list of others all use JavaScript.

The newer technology in this space is HTML5, which came out in 2014. It is a much more powerful version of the Hypertext Markup Language (HTML) code used to create web pages. Part of the reason it is so much more powerful is that it incorporates JavaScript into its functionality. So in other words, instead of JavaScript being an add-on to the web page, it is part of the core code used to create the web page. As of July 2021, 78.5% of all websites utilize HTML. All of the websites mentioned in the last paragraph utilize HTML5.

Currently, only 16.7% of websites that use JavaScript use an older version of HTML. Most sites use HTML5 and get their JavaScript functionality that way.

Reference:
https://w3techs.com/technologies/details/cp-javascript/all/all
https://w3techs.com/technologies/details/ml-html5/all/all



Java

Java and JavaScript might sound similar, but actually, they are two different technologies. Java is a programming language (meaning you must compile Java applications) whereas JavaScript is a scripting language. While these are two different technologies, they both perform a lot of similar tasks, and both are extremely powerful.

JavaScript is incredibly common. In fact, it is difficult to find a web page that does not require it (though HTML5 has all of the same capabilities and is becoming more common). JavaScript can create highly interactive websites. For example, the Gmail web interface is in JavaScript (with two add-ons call JSON and AJAX).

Java is a programming environment that relies on the Java Virtual Machine (JVM) for its applications to run. Because Java applications run in a JVM, they are cross-platform, meaning the same application can run on Windows, Mac, Linux, etc. without modification. This makes Java very useful to both the IT community and to the Hacker community. In fact, since early 2013, Java is the most attacked software in the world.

Author Note:

Java became such a target for attacks that in the spring of 2013, security practitioners encouraged the removal of it from computers where possible. The author does not use Java (and will not purchase a product that requires it). If the author *had* to use Java for something, he would put it in a special virtual machine that is only used for that purpose.

Secure Coding

- Ideally, security and Code Reviews are an integral part of software development:
  - Unfortunately, this is often not the case
- Error and exception handling:
  - When errors occur, software should remain in a secure state
- Input validation:
  - Handles all types of input, even input never dreamed of by the coders
  - Done correctly, invalid input types do not process
- Trapdoor control:
  - When a programmer implements a testing trapdoor, it must be removed prior to releasing the software; otherwise, it becomes a backdoor

Secure Coding

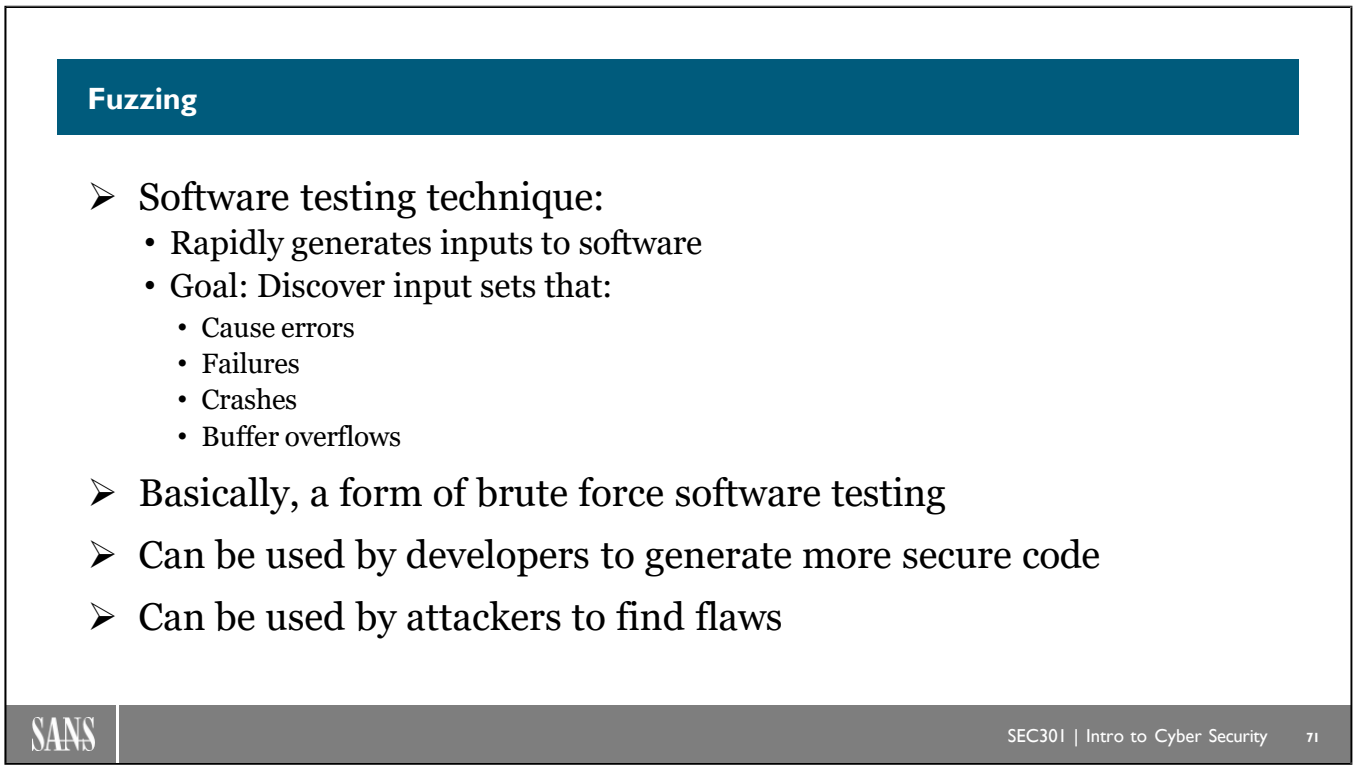
One of the biggest problems in security today is insecure coding practices. When companies develop software for their web applications in particular, insecure coding is a huge problem. In fact, if you look at the most common non-malware based attacks today, four of the five most common can be traced back to this issue. (The most common non-malware attacks are spear phishing, drive-by download, cross-site scripting, buffer overflow, and input validation attacks. Only spear phishing does not find its roots in improper coding practices.)

Fixing this problem requires first and foremost that we start training programmers to write more secure code. Unfortunately, this is not happening on any kind of large scale. A primary place we need to improve our coding practices is in proper error handling. A simple example of where this is not done well would be a website's shopping cart allowing a visitor to put a negative value in the quantity field when ordering a product. The result of allowing that can be a credit on the credit card instead of a charge. Proper input validation is the fix for this. If the input does not make sense (a negative number for the quantity?), then reject the input.

Next, we need to improve our code review practices. Unfortunately, in many organizations, getting the code in place quickly takes precedence. Code reviews take time and companies don't want to spend that time.

Lastly, we need to ensure that all trapdoor functions are removed before the software is released. Trapdoors are a common testing technique. They are essentially backdoors into the functionality of the code that programmers use to test functionality. Doing this is a common and accepted practice
in programming.

The problem arises when the trapdoor is not removed, and the code is released for use. If the trapdoor is discovered, it becomes a backdoor into the software. There have been many examples of this through the years.



Fuzzing

Fuzzers are tools that can help programmers develop more secure code. They rapidly send invalid inputs to the software and see what happens. Does the software handle invalid input correctly, or does it crash, or does it cause the software to behave incorrectly?

Hackers are using fuzzers every day to discover zero-day exploits. In fact, this has become incredibly common.

A "zero-day exploit" is a newly discovered exploit that has not been communicated to the public or to vendors. In other words, only the attackers know it exists. Therefore, we are defenseless against a zero-day exploit.

If developers used the fuzzer to test the software, perhaps they could find that zero-day exploit and fix the problem before anyone else has a chance to discover it. Some organizations are now doing this on a regular basis. But many still are not.

References

https://www.owasp.org/index.php/Fuzzing

http://www.qasec.com/2007/02/using-fuzzers-in-software-testing.html

Module 16: System Security

- OS Hardening
- Patch Management
- Virtual Machines
- Cloud Computing
- Backups

COURSE ROADMAP

- Module 14: Network Security Technologies
  - Lab 5.1: Firewall Builder
- Module 15: Browser and Web Security
- **Module 16: System Security**

Module 16: System Security

Securing our networks and data means securing the systems that make up that network and contain the data. In this section, we look at several issues dealing with securing the PC and server.

OS Hardening (1)

- The process of removing unwanted/unneeded applications and services from a computer:
  - You disable a service and then delete it
  - You remove everything that is not completely required for that system to perform its mission in life
  - Even "innocent" client applications like telnet

- Center for Internet Security:
  - http://www.cisecurity.org/

OS Hardening (1)

We start with the discussion of OS hardening. Most operating systems have far more services and ports enabled by default than are needed. *OS hardening* is the process of getting rid of as much of that as you can.

You need to evaluate each system individually because each system has different requirements. You need to leave different things enabled on a web server than on a mail server, for example. Also, be aware that operating systems have dependencies. If you remove service X because you don't think you need it, you may break service Y that you do need.

When you know the minimum level of services required for that system to do what it needs to do, you disable everything else. After you disable something, you then delete it from the system completely. By deleting the service, you don't have to worry about keeping it patched or that some piece of malware might re-enable it.

You can get good consensus hardening guidelines for multiple operating systems, as well as for routers and switches, from the Center for Internet Security:

http://www.cisecurity.org

OS Hardening (2)

- While hardening systems:
  - Protect management interfaces and applications:
    - Remove the management applications if they are not needed
    - If they are needed, ensure only sysadmins can attempt to run them
  - Remove any unnecessary accounts:
    - I.e., if you can remove the "guest" account, do
    - If you can't remove it, set a massively complex password and disable it
- Application hardening:
  - For the OS to be secure, the applications must be as secure as possible
  - The same principles apply to applications:
    - If the feature is not needed, disable it when you can; secure it when you can't

OS Hardening (2)

Some systems have various management interfaces and/or applications. Two examples include the Remote Desktop Protocol and the Remote Registry Editing services in Windows.

If these and similar services are not going to be used to administer the system, they should be disabled and removed from the system. Each one that is required should be secured as well as possible. In particular, ensure that only authorized system administrators can use them.

The system should have accounts only for authorized users and administrators. In particular, any "guest" accounts should be removed if at all possible. Some versions of the Windows operating system do not allow you to remove the guest account. In this case, set a massively complex 127-character password (the longest Windows allows) and then disable the account. This way, if malware or an attacker succeeds in re-enabling the guest account, they still have to defeat that password to use it.

For the operating system to be considered secure, you have to secure the applications running on that OS. The exact same principles apply here as with the OS. If you don't need it, uninstall it. If you do need it, ensure it runs with the minimum possible permissions. Ensure only personnel who should be using that application can use it. If there is an available security feature, enable and configure it, even if you are not sure how it can help you.

Patches (1)

- Keeping patches up-to-date is one of the most important things for security and stability

- For Windows, use the Windows update feature:
  - Since WinXP SP2, automatic update is included in security center
  - Only does important updates, not optional updates or hardware updates
  - You can choose Windows Server Update Service (WSUS) servers instead of accessing via the web
- Macs always do automatic updates on the OS and all software
  - Unless you disable it, of course

Patches (1)
You cannot hear this too many times: Keeping computers (both the OS and applications) up-to-date with patches is one of the most important things you can do today. Doing so is important for both security and usability/stability.

On Windows, use the built-in Windows Update (choose Microsoft Update on Windows XP). This does have an automatic update feature, but that applies only to "important" updates. There are also optional updates and possibly hardware updates that do not apply automatically. You have to use the Windows update application to check for those.

Automatic updates are a double-edged sword. On the one hand, it is good that patches are applied quickly. On the other hand, it can also be a bad thing. Sometimes, patches can cause systems to become unstable and crash. Testing updates is necessary. For the home environment, automatic updates are probably fine. For production environments, you need to test patches before applying them, especially on servers.

Some organizations choose to use the Windows Server Update Service (WSUS). This is a replacement update server. That server connects to Microsoft to get any updates. Your internal clients and servers then connect to the WSUS server to get updates. Until the administrator goes into WSUS and approves the updates, none of them will be pushed to your systems.

Linux and Mac operating systems also have automatic update capability.

Patches (2)

- You really need to be on at least one mailing list:
  - http://www.sans.org/newsletters/
    - SANS NewsBites, SANS OUCH newsletter, & SANS @Risk newsletters
  - https://docs.microsoft.com
  - http://www.securityfocus.com/archive
    (a collection of newsletters to select from)

Patches (2)

There are several mailing lists that you can subscribe to for finding out about available updates. You need to subscribe to at least one of these.

http://www.sans.org/newsletters

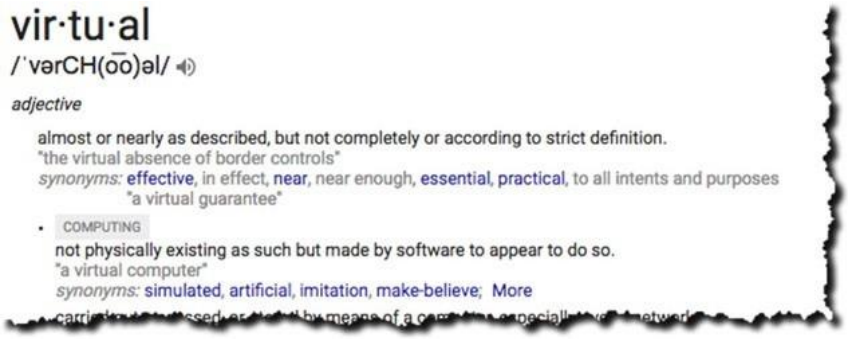
https://technet.microsoft.com/security/

http://www.securityfocus.com/archive

In addition to those you see here, if you rely on a particular vendor's product, you should be on whatever update announcement newsletters that vendor offers as well.

Virtualization

Question: What is a virtual computer?



- A computer that is "nearly a computer but does not physically exist; instead, it is made by software to appear to exist"
- Or: A simulated computer

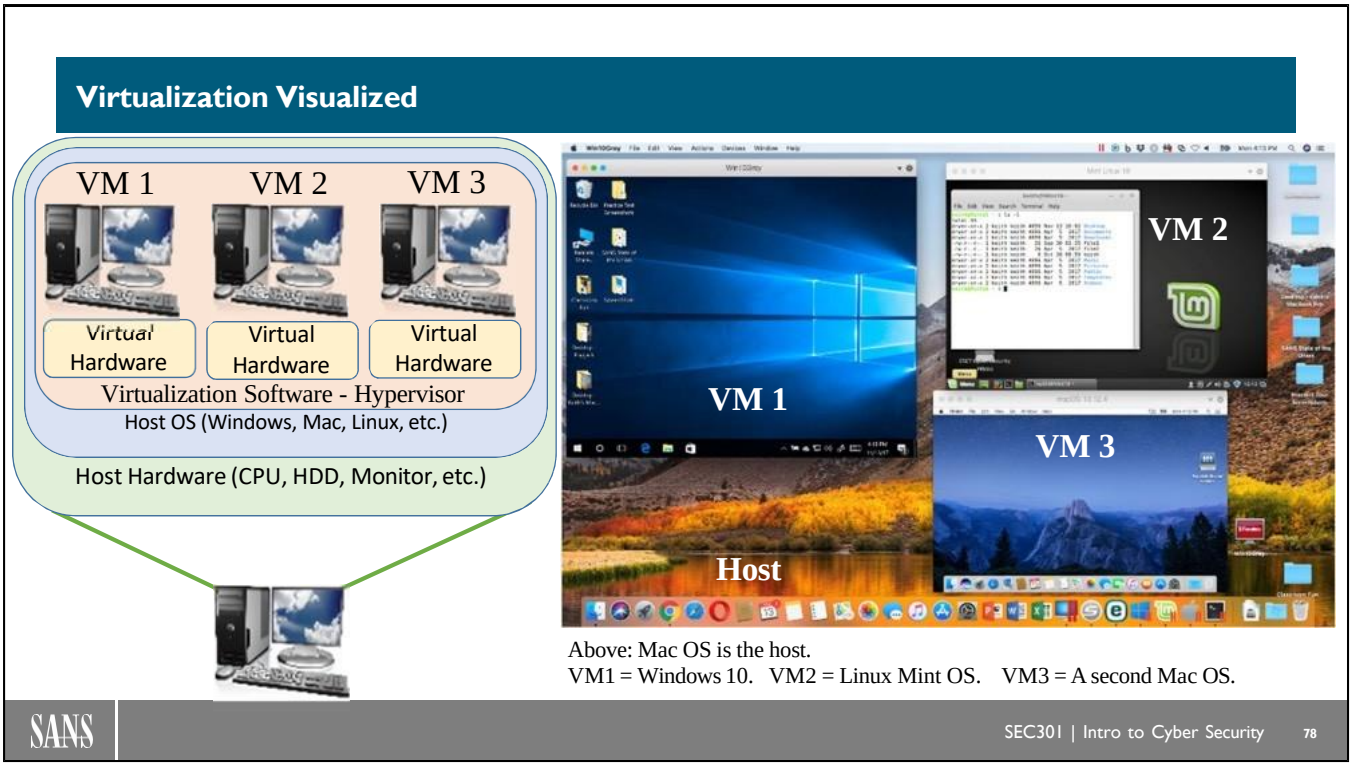
Virtualization

To understand virtualization in computers, one must first answer the question, "What is a virtual machine?" Looking at the definition of the word "virtual" is a good place to start in answering that question.

"Almost or nearly as described ...." So, a virtual computer is almost or nearly a computer. The computing-specific definition states, "not physically existing as such but made by software to appear to do so." If we combine these definitions, we will get a computer that is "nearly a computer but does not physically exist; instead, it is made by software to appear to exist."

Some understand it better by noticing the synonym "simulated." So, one could say that a virtual computer is a simulated computer.

Screenshot source: Google search for "define virtual."



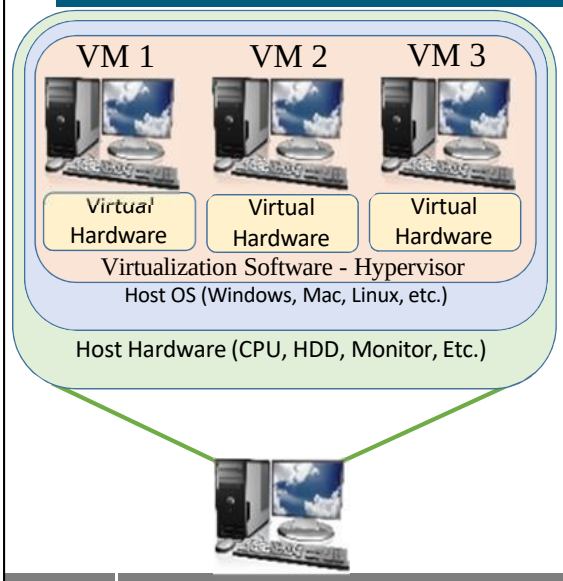
Virtualization Visualized

On the right-hand side of the slide above, you see a screenshot taken on one of the author's Mac laptops. Across the bottom of the screenshot, you see the "Dock" found on the bottom of the screen in Mac OS. Just above that, toward the right, you see the word "Host." This is to help us understand that the host operating system in this case happens to be Mac. Above that, you see a darker window labeled "VM 1," which in this case happens to be Windows 10. To the right of that toward the top, you see a window labeled "VM 2." In this window, you find the Linux operating system. Just below that, you see another window labeled "VM 3" running a second instance of the Mac OS.

So, to say it another way, in that screenshot, you see a total of four computers running. What this looks like from a slightly more technical perspective is depicted in the diagram on the left-hand side of the slide. There you see the computer at the bottom, with its hardware (CPU, HDD, Monitor, etc.). That is running the host operating system (Windows, Mac, Linux, etc.). The Operating System is running the virtualization software, which includes the all-important hypervisor. The virtualization software creates fake (or virtualized) hardware for each virtual computer.

If you look at each of those virtual computers' configuration, each would show its own, dedicated hardware. For each virtual machine, the settings will show a NIC, a video card, a hard disk drive, and so on. Again, all of that hardware is fake and being virtualized by the virtualization software, but the guest operating system does not know that.

Virtualization Explained



- Each guest OS believes it is running on its own dedicated hardware
- Hypervisor intercepts all system calls:
  - Guest OS sends info to memory, hard drive, screen, and so on just as always
  - Hypervisor redirects that output to the real hardware
    - Hypervisor is the secret to good virtualization
- You can run one to thousands of VMs on a single host
  - You just need the hardware resources to handle it
- The ROI makes this popular:
  - Less power, less cooling, less floor space, etc.
  - Big $$$ savings in the data center

Virtualization Explained

A bit of trickery has to be going on in the background to make virtualization work. Previously, we stated that the guest OS wholeheartedly believes it is running on dedicated hardware. As far as that OS is concerned, there is a physical hard drive, video card, NIC card, CPU, and all the other components that a physical computer has.

In reality, those components are shared with that guest OS, any other running guest OSs and the host OS. The magic that makes this work is called the hypervisor. When the guest OS needs to write something into RAM, it uses the same system call that it would typically use to do so. When it sends the RAM write request to its "hardware," the hypervisor intercepts the write request and redirects it to the real physical RAM to be written. The same holds true when the guest OS needs to display something on the screen or any other function: The hypervisor intercepts the request and redirects it to the real, physical hardware. On the actual physical hard drive of the computer is a large file, usually with a (dot)VMDK file extension. When the guest OS needs to read from or write to its hard drive, it does so in the standard way. The hypervisor intercepts that access to the hard drive and redirects it, reading or writing with the (dot)VMDK file.

Cloud Computing

What is a cloud?

cloud com·put·ing

noun

the practice of using a network of remote servers hosted on the Internet to store, manage, and process data, rather than a local server or a personal computer.

- The "remote servers" are virtual computers on data center hardware
  - You can do virtualization without doing cloud
  - You cannot do cloud without using virtual computers

Cloud Computing

With an understanding of virtual computing, we can move into a discussion of cloud computing. The first question to ask here is, "What is a cloud?" According to Google's online dictionary, cloud computing is, "the practice of using a network of remote servers hosted on the internet to store, manage, and process data, rather than a local server or a personal computer."

Almost without exception, those remote servers are virtual computers in the data center of a cloud provider (or possibly the company's own data center). On this slide and the next several that follow, you will see pictures of equipment racks in data centers. Physically, this is what "the cloud" looks like. Large cloud providers such as Amazon, Microsoft, Google, Rackspace, and so forth have data centers containing thousands of these equipment racks. Each of those equipment racks could be running 100+ physical computers. Each of those physical machines could be running over 1,000 virtual computers. If you do the math, each of those thousands of equipment racks is running over 100,000 virtual computers. That is the real cloud.

Screenshot source: Google search for "define cloud computing."

Software as a Service (SaaS)

- Cloud provider supplies and manages:
  - Hardware, OS, software, and data storage
  - Gmail, Google Apps, Salesforce, GoToMeeting
- Customer accesses the software app via the internet:
  - Most commonly via a browser (can be a provider supplied application)
  - An example of a thin-client model
- The customer has little control over any part of the experience:
  - But then, the customer does not have to worry about any of that either
  - The only IT expense for the customer is desktop (or laptop or tablet) that can run a browser

Software as a Service (SaaS)

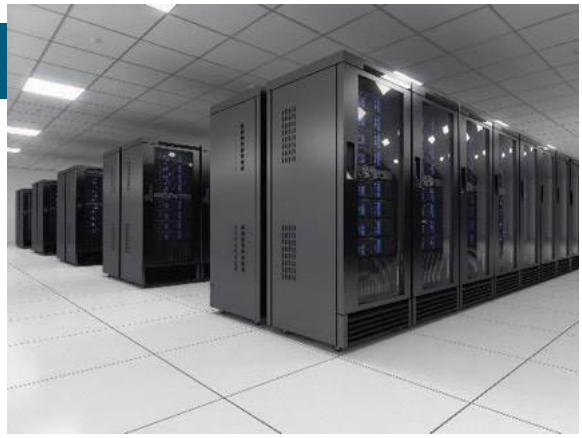
Software as a Service (SaaS) is probably the most common form of cloud computing. You have probably used it on many occasions, even if you did not realize you were doing so. For example, if you have a Gmail, Hotmail, Yahoo! Mail, or any other online mail service, you have used SaaS.

Here, the cloud provider (or provider of service) makes its software available to you, most commonly through a browser interface. You interact with the software in a *thin-client model*. In other words, little or nothing is stored on your local hard drive. For example, if you log in to your Gmail account and see there are 1,000 email messages in your inbox, you can click those messages to read them, reply to them, and so on. But those email messages are not stored on your local computer. In the case of Gmail, they are stored on servers in Google data centers. Only when you download an email attachment is anything stored locally.

The bad news: In SaaS situations, the customer has little control over any part of the experience. If Gmail (continuing with our earlier example) decides to make a change to its interface, you have to use that change. In other words, you use the service today and it looks one way; when you use it tomorrow, it might look completely different.

The good news: Your IT staff does not have to devote any resources to maintaining an email system (or whatever service is delivered via SaaS).

Platform as a Service (PaaS)



- Cloud provider supplies and manages:
  - Hardware, OS, core system applications
  - Also provides the data storage
- Customer provides and manages:
  - Production applications:
    - Word processing, spreadsheet, email, web, etc.
- Customer has some control over the production application:
  - But no control (or worries) about hardware, OS, etc.
  - No expense for maintaining a data center

Platform as a Service (PaaS)

The Platform as a Service (PaaS) solution is typically a service purchased by a corporate IT department. Instead of having their own computers in their own data center, they purchase computers from a cloud provider.

In reality, they are almost always actually purchasing one or more virtual servers. They tell the cloud provider what operating system they would like (Windows or Linux, typically). The cloud provider sets up the virtual server and grants the customer IT staff access to it.

The IT staff then manages the system remotely just as it would if it were located in its own data center. It installs software, configures services, and so on. The cloud provider is responsible for keeping the system running and usually for doing things such as keeping OS patches in place. The corporate IT staff is responsible for maintaining whatever applications it uses for the system to run.

Here, the corporate IT staff has more control than with SaaS solutions, but not as much control as it would have over its own server. For example, it may not dictate how quickly or even if certain OS patches will be applied.

Infrastructure as a Service (IaaS)

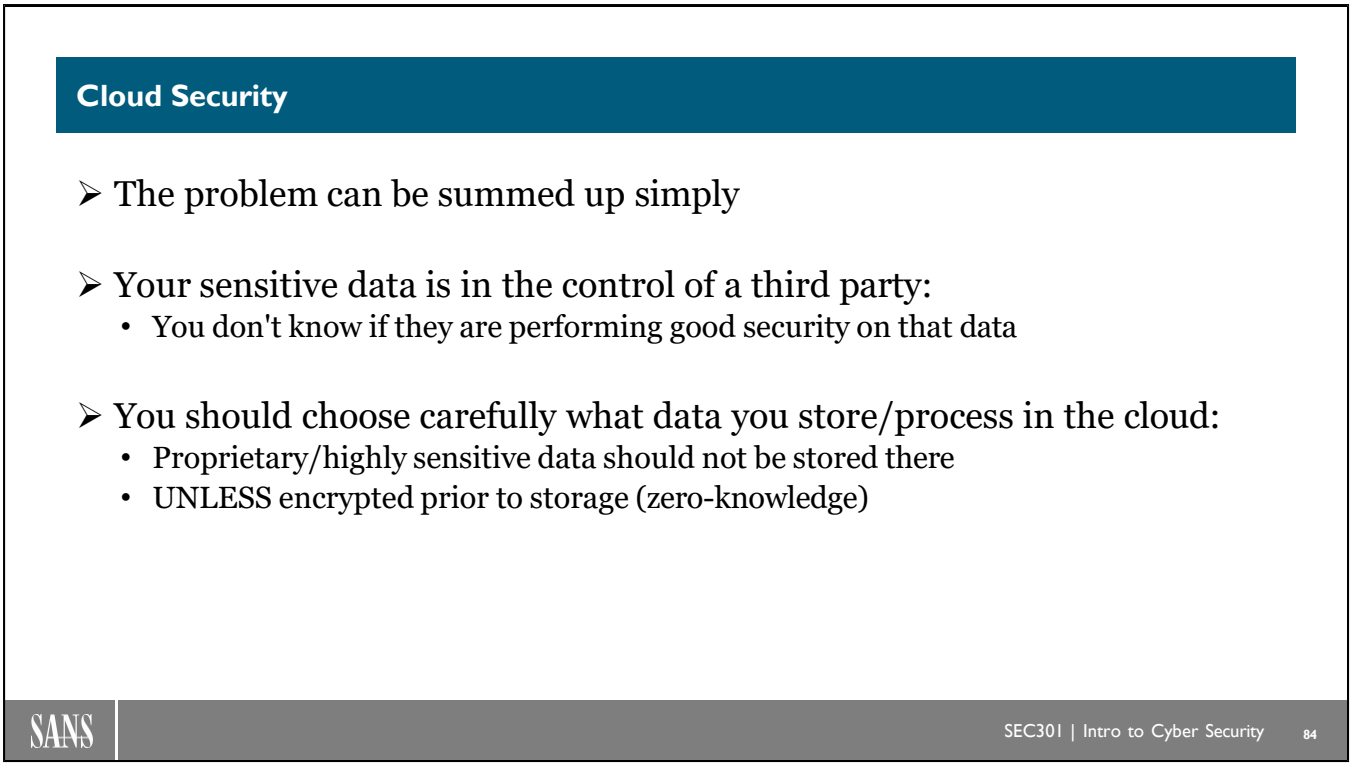
- Cloud provider supplies and manages:
  - Hardware and maybe the core OS
  - Provides data storage capability
- Customer manages the system as though it were in their own data center:
  - Replicates how Admins administer systems they own—these are just much farther away
- Customer has a fairly high degree of control:
  - And greater responsibility in managing the system
  - It is still much cheaper than maintaining an entire data center

Infrastructure as a Service (IaaS)

Infrastructure as a Service (IaaS) is sometimes referred to as a "server in the cloud." The cloud provider sets up the hardware (typically actually a virtual machine). The cloud provider in most cases also installs the base operating system of the customer's choice (Windows or Linux). From there, the cloud provider is responsible for providing the power and cooling to keep the system running and the internet connectivity so that it can be accessed. Nothing more.

The corporate IT staff that has purchased (or more accurately, rented) these servers perform all maintenance tasks. This is truly the same thing as an IT staff maintaining its own servers in its own remote data center (common). The only difference is that it doesn't happen to own that data center and it doesn't actually own the servers.

The IT staff has the greatest degree of control but also the greatest degree of responsibility for maintenance. That is the biggest difference between Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). How much control does the IT staff want to retain for themselves? More control usually means greater flexibility, but it also means greater responsibility for keeping things up-to-date and running properly.



Cloud Security

The cloud security problem is a difficult problem to solve. We are making progress in this area, but frankly, we don't have it all worked out yet.

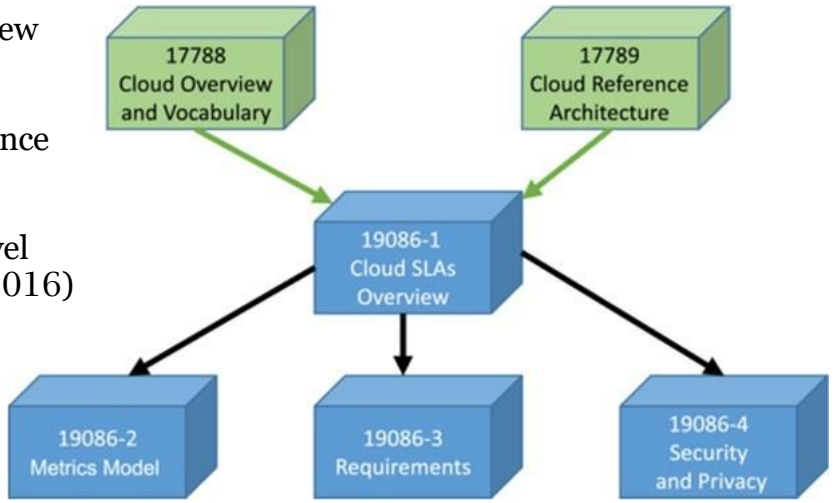
The crux of the problem is that your sensitive data is going to leave your protected network and be placed on another company's servers. You have minimal control over those servers (and no physical access control at all).

Who is typically responsible for the security of your data? You are. Read the agreement with the cloud provider carefully. It clearly (or perhaps not so clearly) states that you, the customer, retain responsibility for the security of your data. They are simply providing a service by making a platform for processing and storing that data available to you.

Choose the data you store in the cloud carefully. Highly sensitive data should not be stored in the cloud at all unless it is done in a *zero-knowledge implementation*. That term means that the data is encrypted before it ever leaves your local computer and is stored in the cloud *only* in an encrypted form.

Cloud Security: ISO Standards

- ISO 17788: Cloud Overview and Vocabulary (2009)
- ISO 17789: Cloud Reference and Architecture (2009)
- ISO 19086-1: Service Level Agreement Framework (2016)
  - ISO 19086-2, 3, and 4 are still in draft

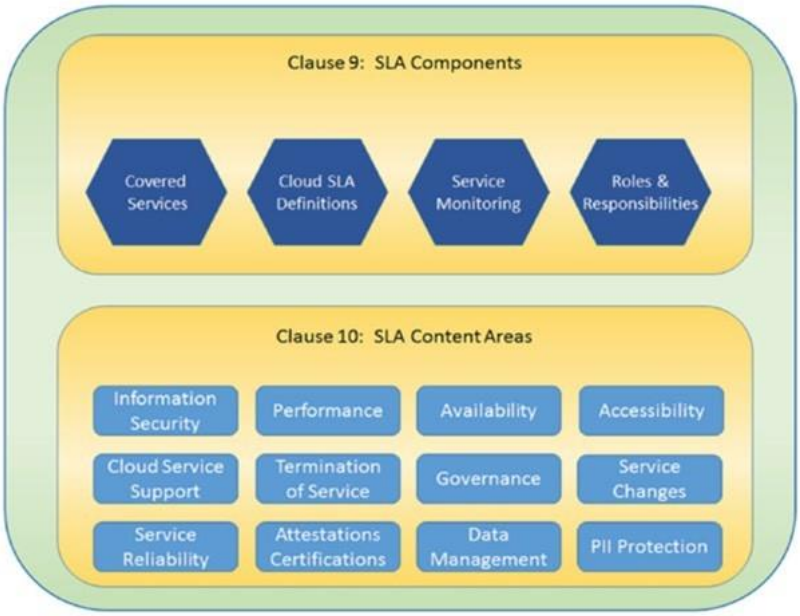


Cloud Security: ISO Standards

There are some ISO (International Standards Organization) standards already complete (and three more in the works) for cloud security. Two were published in 2009. Specifically, ISO 17788, "Cloud Overview and Vocabulary," which includes common cloud computing terms such as those we just discussed (SaaS, PaaS, and IaaS). ISO 17789, "Cloud Computing Reference Architecture," is populated with diagrams and descriptions of how cloud computing components relate to one another.

In late 2016, ISO 19086-1 was published, which describes Service Level Agreements (SLAs) that should be in place when you place important information in the cloud (see next page for more). The ISO body is working on ISO 19086-2 (Metrics Model), ISO 19086-3 (Requirements), and ISO 19086-4 (Security and Privacy), but as of this writing (July 2018), there is not an estimated completion date for these standards.

ISO 19086-2: SLA Concerns



ISO 19086-2 gives very thorough guidance on questions to ask and answers to obtain before putting your data in the cloud.

Or, if your data is already there, you should still ask these questions now!

86

ISO 19086-2: SLA Concerns

As the chart above reveals, the ISO 19086-2 standard provides very thorough guidance on the questions to ask and answers to get prior to putting sensitive data in the cloud. If your data is already in the cloud, it certainly cannot hurt to ask the cloud provider these questions. Even though you may not like the answers you get, at least you will have those answers.

Backup Criteria

- Copy on separate media
- Needed when (not if) the original is lost
- Must be easy to use:
  - Including automatic verification of data integrity
- Should have periodic test recovery

A *backup* is a copy of information that is stored
on media separate from the original.
Generally: Greater physical separation = greater security.

SANS | SEC301 | Intro to Cyber Security 87

Backup Criteria

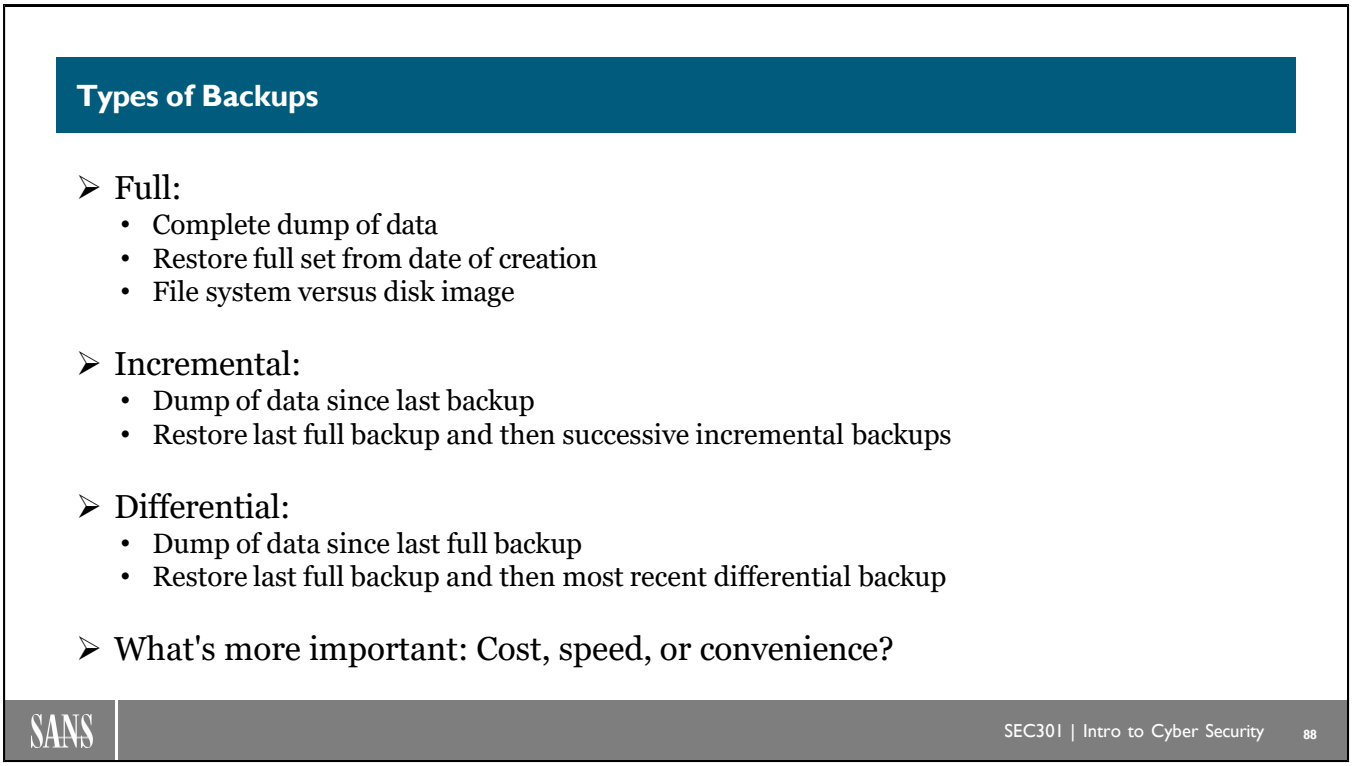
A backup is a copy of information that is stored on media separate from the original.[1] If the media on which the original is provided fails, the backup can be used to restore the information. Note that a copy is not necessarily the same as a backup. If the copy is on the same media as the original, it is a copy (but not a backup). If the copy is on different media, it is a backup.

Original information can be lost in a variety of ways. Users can accidentally delete files, data on media can become corrupted, and disk drives can become inoperable. Prudent practice takes this into consideration and provides for recovery in the event of failure.

Although it can be easy to say, "back up your data," it's important to implement a mechanism that is easy for users to execute. Telling users to back up their information from their workstation (or laptop) to the server and not providing a mostly transparent mechanism for doing so is an invitation to disaster.

Reference

[1] http://www.thefreedictionary.com/backup



Types of Backups

There are three basic types of backups: Full, incremental, and differential. Each involves different trade-offs in terms of speed and recoverability.

A full backup is just what it sounds like: A complete dump of all the data on a device or system. Full backups are the easiest to understand conceptually. If you want to back up your C: drive, dump it all out to a network tape drive or a few DVDs.

We should distinguish the difference between a full file system backup and a disk image backup. When you back up a system using a file system backup, the backup software traverses all the directories and files in the file system and systematically copies them to the backup location. What you get in the end is a copy of the file structure of the original device and all the information contained in those files. In contrast, a disk image backup examines each sector of the physical disk and copies them intact to the backup media. What you get in the end of a disk image backup is a sector-by-sector copy of the original device. This may include all unused sectors on the disk, as well as any slack space data that the drive may contain.

When you restore your data, the type of backup you used can affect what you have at the end of the restore. When you restore a file system backup to a device, the files from the backup media are copied to the existing file structure of the target device. Any files on the target device that have the same name and location as files on the backup device will get overwritten by the files from the backup. However, any files that were newly created on the target device will continue to exist on the target after the restoration has been completed.

In contrast, when you restore a disk image backup to a target device, each sector on the target is overwritten with the data from the backup. If you created new files on the target prior to the restoration, those files will be gone after the restoration completes.

Most production data center environments use the file system backup process, as they typically need to restore files and directories to recover lost data, and the device where the restoration takes place may have different physical properties than the device from which the backups were taken. If that is the case, the number of sectors and their location on the target will be different than that of the original device, making a sector-based restore difficult, if not impossible. Disk image backups are typically used when it is important to understand the precise physical state of the device, including unused sectors, such as in the case of a forensics investigation. It can also be a quick method of duplicating one device to another if the two devices share the same physical characteristics.

The advantage of full backups is that creating and tracking them is straightforward, as each set of backup media has a complete system image and gets labeled with the date the backup was taken. Taking a full backup might take quite a while, depending on the size of the system being backed up. When you need to restore your system, you determine the date you want to restore from, grab the tapes or disks from that day's full backup, and restore the data, in full, onto the system. The disadvantage that full backups have is that they can potentially use up huge amounts of backup media. The average CD-R stores only 700 megabytes of information. Even DVD-Rs store only a little under 5 gigabytes of data. If you have a full 160 GB disk drive, that's a lot of disks for each backup. For a large data center with hundreds of systems to back up, the physical storage requirements alone become cost-prohibitive.

Enter the *incremental backup*. Incremental backups take advantage of the relatively small number of files that actually change on a typical system during normal usage. The typical incremental backup method starts by taking a full backup of the system on a Sunday evening (for example). Then, for each of the following days (Monday through Saturday), an incremental backup is taken. On the following Sunday, another full backup is taken, and the cycle starts over again.

To restore data from an incremental system, you first start with restoring the most recent full backup prior to the date you want to restore from. So, for example, if you need to restore data as it appeared this past Wednesday morning, you must start by restoring the full backup from last Sunday evening's tapes. Then you restore the incremental backups from last Monday evening and last Tuesday evening, in that order. After you finish, the system will have all the data it had when you walked into work on Wednesday morning. The biggest advantage to the incremental system is that it saves a great deal of backup storage space. The full backups take a lot of room, but the incremental backups take only a small amount of space to keep up with the changes. The disadvantage of the incremental system is that it can take a long time to restore the data you need. If you take your full backup on Sunday evenings but you want to restore the system as it appeared on Sunday morning, you need to restore a full week's worth of tapes before you are through. You start by restoring the previous Sunday evening's backup, and then successively restore the backups from Monday evening, Tuesday evening, and so on through Saturday evening. That's seven rounds of restores! But that's the trade-off you make for using up as little backup storage space as possible.

A good compromise that combines the speed of full backups and the minimal storage space usage of incremental backups is the *differential backup*. With differential backups, you back up only files that have changed *since the last full backup*. This uses more backup storage space than the incremental method, but when it comes time to restore a system, you make up for it in speed.

Going back to the example we just used, if you want to restore the system to the way it looked on Sunday morning using a differential system, you start by restoring the full backup from the previous Sunday evening. Then, you restore only the differential tape from the following Saturday evening—only two rounds of restoring needed!

As with most security issues, which method you use is a trade-off between cost, speed, and convenience. If you need complete images of a system each time you back up, and you don't mind the extra system downtime that will take, use full backups. However, you pay for it in storage media costs. If saving money on storage media and minimum system downtime are the most important, and you don't mind spending some extra time restoring data, go with the incremental system. If speed of restoration is a priority, and you are willing to sacrifice some downtime and spend a little extra on storage media, the differential system is a good compromise.

Backing Up Systems, Applications, and Data

- Full system dump:
  - Quick and easy
  - Restores operating system, applications, settings, data…
    - Returns the system to a prior state
    - Like a time-machine for a computer
  - Very often the fastest & easiest way to recover from problems

For example, Mac "Time Machine"
- Hourly backup for 24 hours
- Daily backup for the last month
- Weekly backup for prior months as long as space allows
- Provides restore back to bootup with configuration and data as of the last backup

- Windows 10 Backup and Restore with File History has a similar capability

Backing Up Systems, Applications, and Data
Both Windows and Mac include a feature that performs a full system dump. In other words, it is not just a backup of your data – It is a backup of your data, your operating system, your applications, system settings, and so on. It is a full replication of your computer and everything it contains written to a directly connected external hard drive.

On Windows 10, this is called “Backup and Restore”. On Mac, it is called ”Mac Time Machine”. In the case of Mac for example, it performs full system images to the external drive hourly for 24 hours. It then keeps one of those as daily backup for a month, and one of those as a weekly backup for as long as the drive space allows. Once you run out of space, it deletes the oldest backup on the drive. In IT, we call this first-in-first-out or FIFO.

Why this is important: A while back, the course author was updating his Mac computer to a new version. For some reason, the update failed and his computer would no longer reboot. He used the built in Mac Time Machine utility to restore his system to a point two-hours in the past. The utility pulled the system image from two hours prior and rewrote everything on his hard drive. Essentially, he rolled the computer back two hours in time. It was like the attempted update never happened. He tried the update again and it worked perfectly.

The point is that if the author didn’t have a Mac Time Machine image to revert to, he would have had to reload the system from scratch, re-install applications, re-do settings, recover data, etc. It would have taken him several hours at least. Instead, the process took about 20 minutes. He is a big fan of full system images!!!

Backup Locations

- On-site:
  - Fast retrieval
  - Space requirements may outgrow avail ability
  - Disaster can wipe out backups
- Off-site (yours or someone else's):
  - Data is safe from disaster (at your site!)
  - Slower retrieval
  - Cost and location considerations
- Hybrid:
  - Recent backups on-site
  - Long-term storage off-site

Backup Locations

Let's assume you have a backup strategy, and perhaps you've even started doing a few backups. Now, where do you put all the tapes you are generating? Here, you have several options, again each with its own benefits and trade-offs.

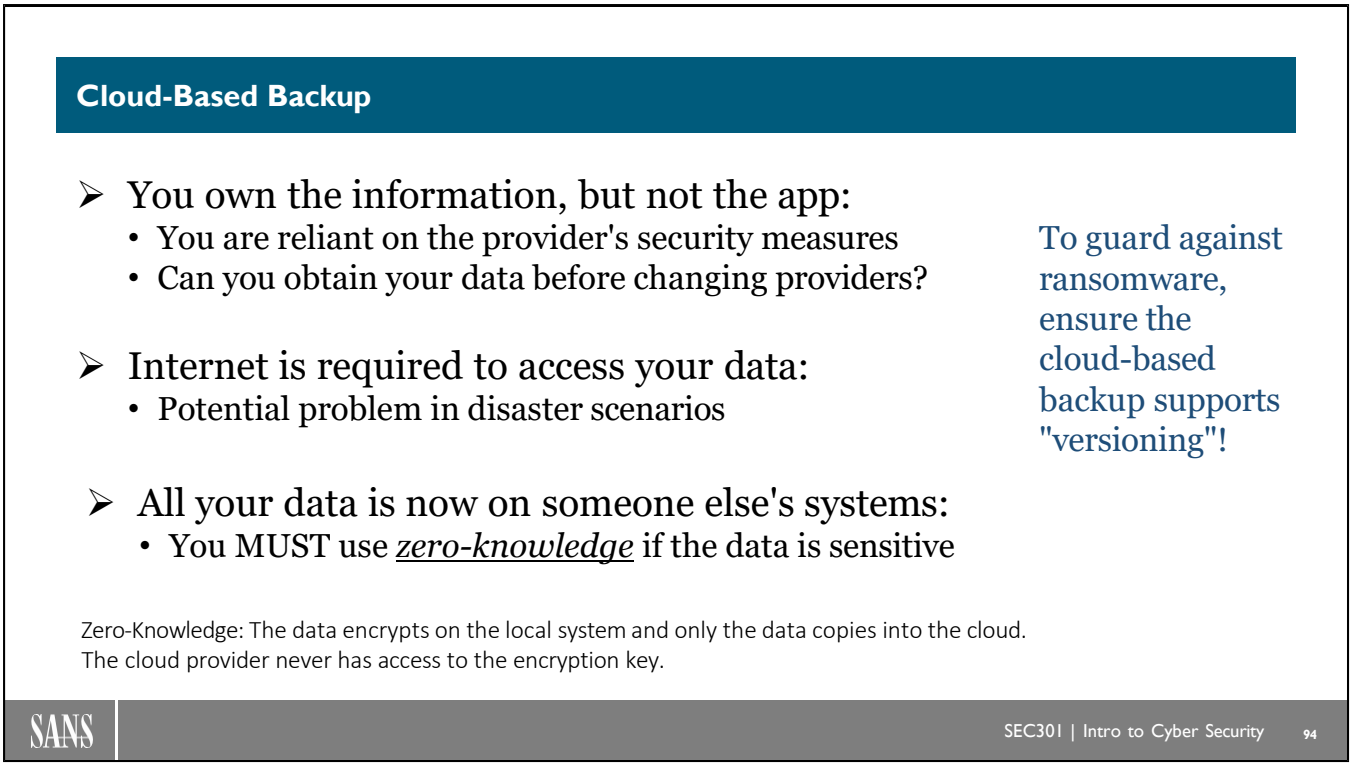
The initial inclination will be to store all the tapes at the location where the systems are so that storing and retrieving the tapes can be done quickly, facilitating quick recovery in a disaster. Unfortunately, if you take backups long enough, you may run out of enough available physical storage capacity to store all your data. In addition, if you have a disaster at your site (such as a fire or flood), your backup tapes may be lost along with the rest of the site. As backups are a primary disaster recovery mechanism, their loss in such a disaster would be ironic, to say the least.

To counter the threat of loss due to a disaster, many organizations choose to store backup tapes off-site, either at another location owned by the organization or at a commercial storage facility. Storage at another organization-owned location has its advantages, including cost and availability of space. Commercial sites offer such features as climate control, advanced fire detection and suppression, and enhanced security. Of course, all these benefits come at a price. There are a couple of considerations when you start to consider off-site storage locations. The first is the distance between the primary site and the backup location. A natural thought would be to select a backup site close to the primary site, as that facilitates quick retrieval of the tapes in the case of a disaster.

However, if the disaster is regional in nature (for example, a hurricane or regional power outage) a backup site in the same city or state may be facing the same disaster as the primary site.

Conversely, storing the backup tapes in another state or region lowers the risk of a common disaster but raises the cost of transporting tapes to and from the backup site. Then you must also consider the transportation method for getting the backups from the storage site to the backup site. For example, if you store your backup tapes at a commercial storage facility several states away and rely on an overnight courier to get them to the backup site, what do you do in a 9/11-like event where air traffic has been stopped? This may seem like an extreme example, but it highlights some of the logistical issues off-site storage may bring. The use of high-speed data networks for over-the-wire backup alleviates this concern somewhat, but at the cost of a slower backup and restore process due to line speed limitations.

One good compromise is a hybrid approach in which backups are stored at different locations depending on their age. The most recent backups (for example, tapes from the past two weeks or month) are stored in a safe facility on-site (perhaps a specially constructed room with tight security and enhanced fire suppression). This gives the organization quick access to the data that is most likely to be needed in a hurry. ("Oops! I just accidentally deleted the spreadsheet I've been working on all week!") Tapes older than that are shipped to an off-site facility for long-term storage because the company is less likely to need that on a regular basis.



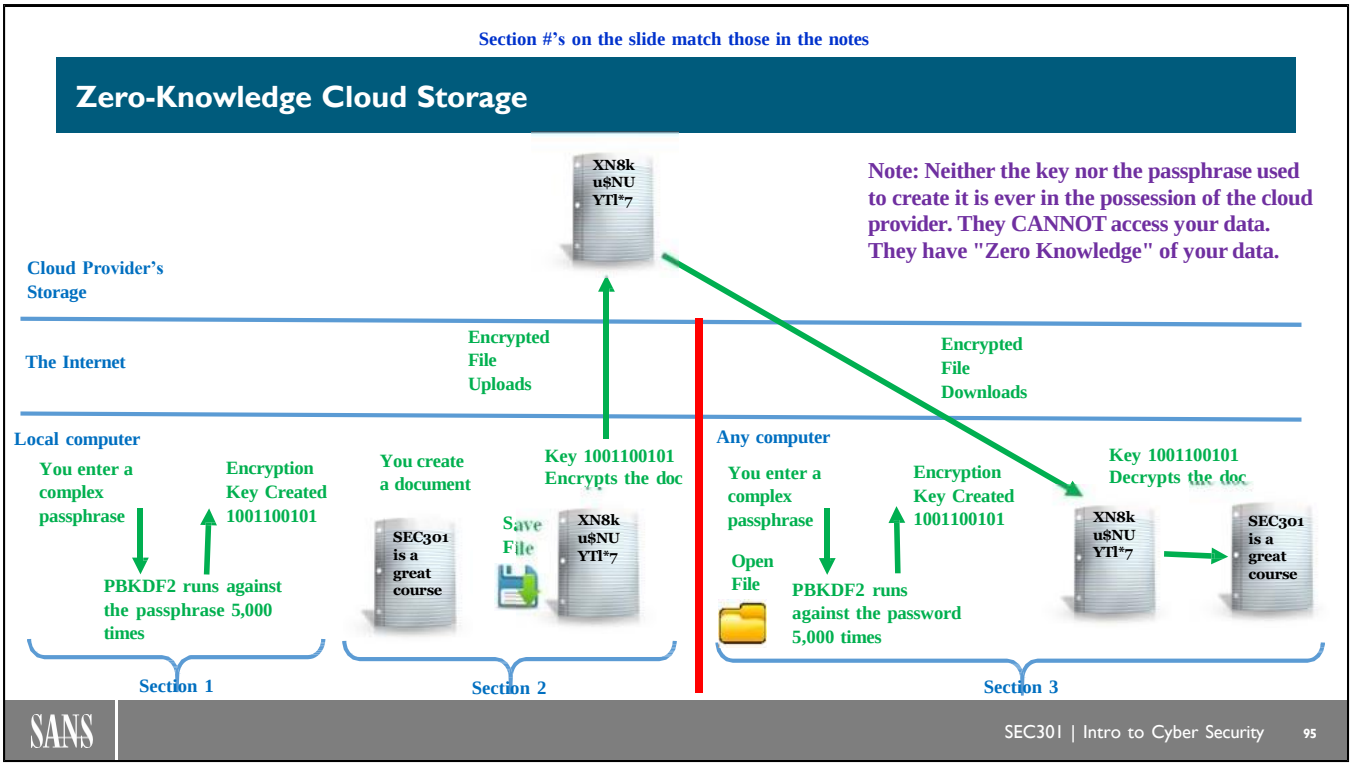
Cloud-Based Backup

Cloud-based backup is becoming extremely common. There are several high-quality services available for purchase. This type of backup is even built into the most recent versions of Microsoft's server platforms.

The idea is very simple: You send your data across the internet and store it on the cloud company's hard drives. This requires you to use the cloud provider's application for making backups. So, although you own your information, you do not own the application. This is especially important because it means you also have to use the provider's software to retrieve your data. If you decide to change backup companies, the provider could potentially block you from accessing your own data. Also, if the provider goes out of business, your data could become completely inaccessible. Choosing providers wisely is vitally important.

You also need to think disaster recovery scenarios through carefully. Although a cloud-based backup provider can be a terrific solution for day-to-day operations, in a disaster it may not be. If the disaster shuts the internet in your area down, your data is inaccessible until internet service is restored.

Last but not least, you are putting all your data in someone else's hands. Assuming the data in question is even a little bit sensitive, you *really* need to use zero-knowledge systems.



Zero-Knowledge Cloud Storage

When you put sensitive information into cloud storage, it is obviously important to protect that information. Several providers now provide "Zero-Knowledge" implementations. If you set this up correctly (and use a good, strong passphrase), it can be highly secure. The section numbers below match those on the slide above.

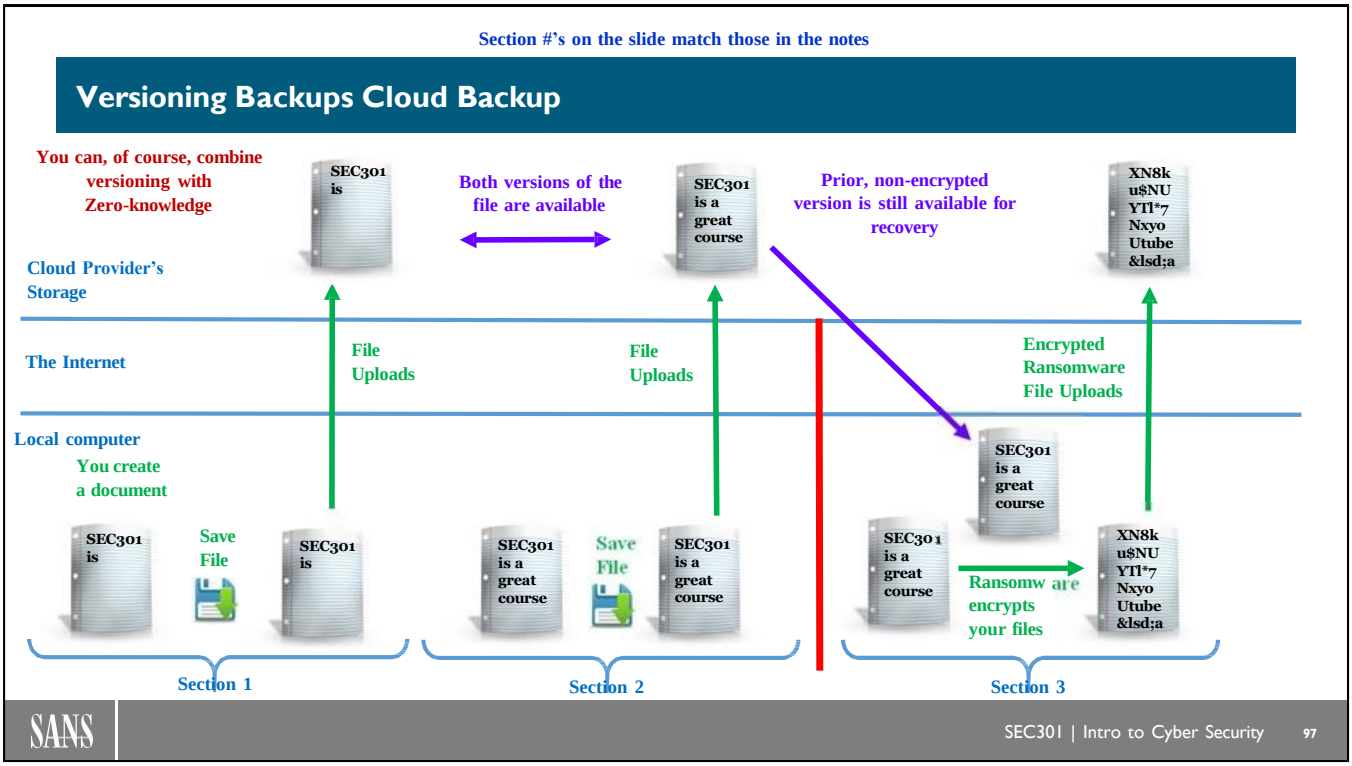
SECTION 1: On your local computer, you enter a good, strong passphrase. Software on your computer takes that passphrase and runs it through an extremely complex process called Password-Based Key Derivation Function 2 (PBKDF2). (The detail of PBKDF2 is far beyond the scope of this class, but just understand that it takes a passphrase and derives an encryption key from it.) In most implementations, the passphrase runs through the PBKDF2 process 5,000 times (each iteration makes the resulting key more random). The end result is an encryption key.

SECTION2: You then create a document of some type. When you save that document, the passphrase-derived key is used to encrypt the document. The encrypted form of the document uploads to the cloud storage provider.

SECTION3: On some computers with the zero-knowledge application installed (perhaps the same one, perhaps another one), you click to open the file stored in the cloud. The software prompts you for your passphrase. You enter the same passphrase you entered in section 1. That

passphrase runs through PBKDF2 the same number of times as before and generates the exact same encryption key. The encrypted file downloads, and the key decrypts the file.

NOTE: Neither the key nor the passphrase used to create it is ever in the possession of the cloud provider. They CANNOT access your data. They have "Zero Knowledge" of your data.



Versioning Backups Cloud Backup

It has always been a good idea to use versioning backup services. With the advent of Ransomware, the use of versioning backup is now critical! Note that the section numbers below match those in the slide.

SECTION 1: You create a document on your local computer. When you save that document, it is saved to your local hard drive. In addition, backup software you have running on the computer also places a copy of that file in the cloud provider's storage (vital—this needs to happen automatically).

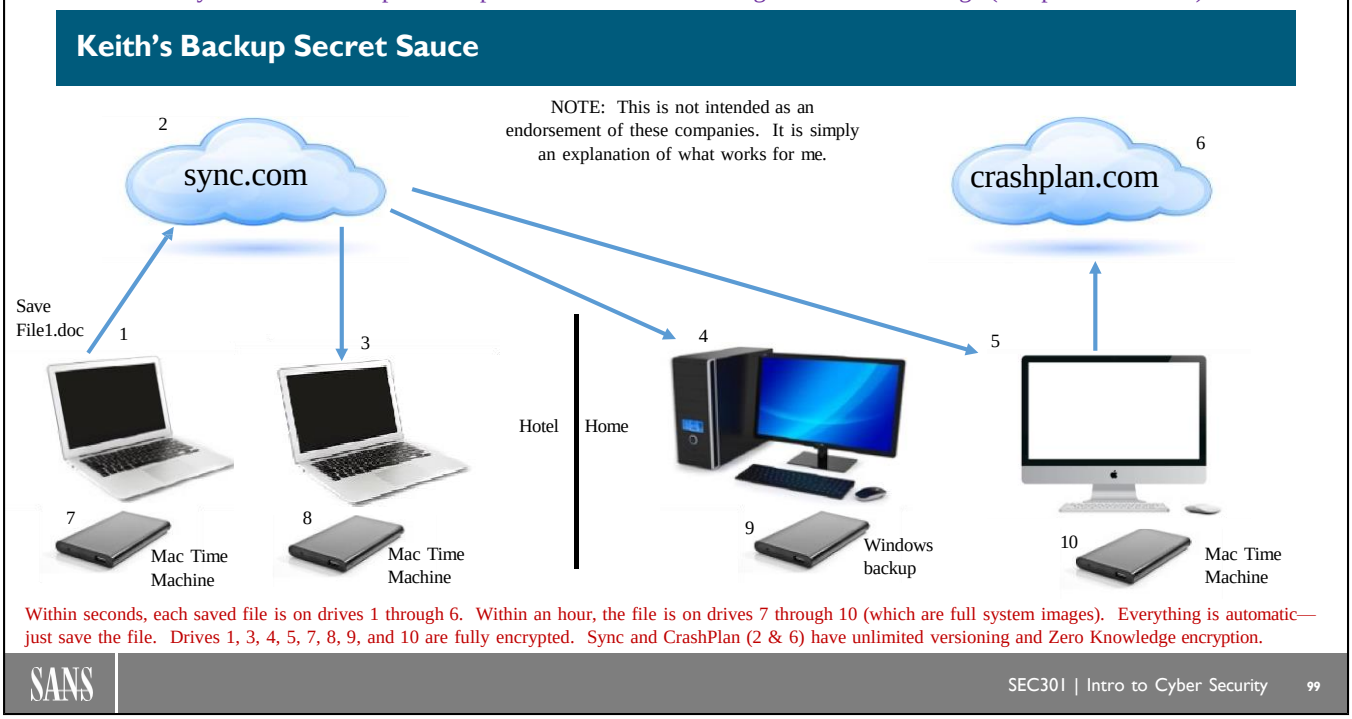
SECTION 2: At some point, you edit that document. When you save the changes, the new version is written onto the local hard drive and a copy goes into cloud storage. Notice that the cloud provider's storage system now contains two copies of your document: The one you originally saved and the new version. (Some cloud providers offer unlimited versions. Others keep a subset, for example, the last five versions, or the last 90 days' worth, etc.)

SECTION 3: Let's say that at some point, you get hit with Ransomware. The Ransomware encrypts all of your files, including the one in our example here. The encrypted version of this file will back up to the cloud provider's storage. However, because of versioning, your prior, non-encrypted versions of the document remain in the cloud provider's storage as well. Once you clean

the Ransomware off your system, you can simply download the most recent prior version, and you have your data back in a non-encrypted form.

Note: You can, of course, combine versioning and zero-knowledge for highly secure and durable backups.

Both sync.com and crashplan.com provide unlimited versioning and Zero Knowledge ($10 per month each)



Keith's Backup Secret Sauce

We have now talked about hybrid backup approaches where data is kept both locally and in the cloud. We have also talked about both versioning backup and Zero Knowledge backup. What might this all look like in a real-world implementation? Perhaps the easiest way to explain is to show the author's backup solution. Please note that while the author mentions two companies he uses in this solution, he is not providing an endorsement of those companies. He is simply saying that their services met his needs and budget.

First, the setup: The author carries two identical Mac laptops. One silver and one grey (so he can tell them apart). In the diagram above, those are in the Hotel side. At his home, he has a Windows desktop and and Mac desktop.

1. When he saves File1.doc on the silver laptop, it automatically backs up to Sync.com.
2. Sync is a service that offers unlimited versioning with Zero Knowledge—meaning that every version of File1.doc is kept there, and all of them are fully encrypted in such a way that Sync cannot see the contents.
3. Sync automatically replicates the file to the grey laptop, also in the hotel.
4. Sync automatically replicates the file to the Windows desktop (in the home office).
5. Sync automatically replicates the file to the Mac desktop (in the home office).
6. As soon as the file is on the Mac desktop, it is backed up to CrashPlan which is another company that provides unlimited versions with Zero Knowledge.

7. The author also has external USB drives connected. Within an hour, Mac Time Machine will create a full system image of the silver laptop. The image contains the OS, software, settings, and all data on the laptop including File1.doc.
8. There is an external USB drive connected to the grey laptop, so within an hour, there is a full system image on that drive as well.
9. The Windows desktop has an external USB attached, and within an hour, Windows Backup and Restore with File History creates a full system image of that computer including the OS, software, settings, and all data.
10. The Mac desktop has an external USB attached and Mac's Time Machine creates a full system image of that computer every hour as well.

So within seconds of saving File1.doc, it resides in six locations. Two copies are with the author in his hotel room, two are on computers in his home, and two are in the cloud. Within an hour, four more copies are made of the file as part of full system images.

Should a catastrophic data loss occur such as Ransomware, the quickest recovery will be to restore the system image of that system taken before the infection occurred. Remember that both Mac Time machine keep system images every hour for a day, every day for a month, and every month for as long as there is drive space. There should certainly be a clean image to roll back to. Of course, the files are also on other computers that hopefully did not get infected. And, if all else fails, all the data is kept securely by two cloud providers so that even if the data cannot be recovered from one for some reason, there is a second available.

It should be noted that a backup is a copy of your data and must also be protected. That is why Drives 1, 3, 4, 5, 7, 8, 9, and 10 are fully encrypted. Even if an external USB is lost or stolen, without the very complex passphrase needed to decrypt it, the data cannot be recovered. Of course, with both cloud providers providing Zero Knowledge, no one without the different complex passphrase can recover the data from there either.

I will admit, this is more backup than most individuals need. I'm protecting my company and my livelihood which justifies the cost.

Speaking of cost, as of this writing (April 17, 2021), both Sync.com and CrashPlan.com cost $10.00 per month. The external USB drives only need to cost about $50.00, assuming 1TB size. This really is not terribly expensive.

One note about a setup of this nature. Only one person should edit a file on one of the computers at a time. For example, if I were editing File1.doc in my hotel room at the same time my wife was editing that same file in the home office, it causes a conflict… literally meaning it would save one of our files with the name "File1-CONFLICT.doc". We would not loose our work, but it would mean combining the edits into a single file by hand. If I edit File1.doc while my wife edits File2.doc, that is not problem at all.

SANS

Cyber Security Technologies & Web Security

End Day 5

Congratulations!

This page intentionally left blank.

Course Resources and Contact Information

AUTHOR CONTACT
Keith Palmgren
keith@netip.com

SANS INSTITUTE
11200 Rockville Pike
Suite 200
North Bethesda, MD 20852
301.654.SANS(7267)

Cyber Defense RESOURCES
https://cyber-defense.sans.org/
Twitter: @SANSDefense

SANS EMAIL
GENERAL INQUIRIES: info@sans.org
REGISTRATION: registration@sans.org
TUITION: tuition@sans.org
PRESS/PR: press@sans.org

This page intentionally left blank.

This page intentionally left blank.

Appendix

SANS

Review Questions

This page intentionally left blank.

Questions

SANS

Network Security Technologies Review Question

This page intentionally left blank.

- What is breaking up a network for better management known as?
  A. Network Division
  B. Segmentation
  C. Departmental Organization
  D. Compartmentalization

- What is breaking a network into security zones called?
  A. Network Division
  B. Segmentation
  C. Departmental Organization
  D. Compartmentalization

This page intentionally left blank.

- What is the most fundamental purpose of a firewall?
  A. Controlling where internal users are allowed to go on the internet
  B. To keep the person off your network that does not belong there
  C. Performing Network Address Translation (NAT)
  D. Compartmentalization

- What mechanism do you use to dictate what traffic is allowed and what traffic is denied?
  A. Every firewall is different
  B. Written policy
  C. Firewall rules
  D. Firewalls do that automatically

This page intentionally left blank.

- Which traffic type goes through a proxy firewall?
  A. Any traffic allowed by the firewall rules
  B. Traffic from inside the network to the internet
  C. Network Address Translated traffic
  D. No traffic goes through a proxy

- Which traffic type goes through a stateful inspection firewall?
  A. Any traffic allowed by the firewall rules
  B. Traffic from inside the network to the internet
  C. Network Address Translated traffic
  D. No traffic goes through a stateful inspection firewall

This page intentionally left blank.

- What is the underlying, core technology behind stateful inspection?
  A. Proxying
  B. Packet filtering
  C. Network Address Translation
  D. Stateful inspection is its own technology

- What type of systems do you put in the DMZ?
  A. A DMZ has no servers, just client PCs
  B. Servers with sensitive data only
  C. Firewalls
  D. Public access systems

This page intentionally left blank.

- When is it justified to put public access servers on an internal network?
  A. Never
  B. As long as they are fully patched
  C. When that is the most convenient way to access them
  D. We always do that

- What is the acronym for an automated system that watches for signs of an attack called?
  A. DNS
  B. IPS
  C. ISP
  D. IDS

This page intentionally left blank.

- When an IDS watches for patterns of an attack in packets, what is it doing?
  A. Signature analysis
  B. Intrusion Detection and Prevention (IDP)
  C. You can't do that
  D. Anomaly analysis

- To stop attacks, an Intrusion Protection System must also be what?
  A. DNS
  B. IPS
  C. ISP
  D. IDS

This page intentionally left blank.

- When there is no valid reason for anyone to access an IT resource, what do you call that resource?
  A. A Honeypot
  B. A sweetspot
  C. Why would you do that?
  D. A nonresource

- What is a common name for a solution that prevents inappropriate web surfing?
  A. Access filter
  B. Content filter
  C. WebSense
  D. Surfing Ruleset

This page intentionally left blank.

- What is another name for an all-in-one security appliance?
  A. A Honeypot
  B. This is nothing but a marketing term—they don't really exist
  C. Layered Management Application Firewall
  D. Unified Threat Management (UTM)

- What is the less common but more accurate name for a sniffer?
  A. Protocol Dissector
  B. Wireshark
  C. Protocol Analyzer
  D. Packet Capture

This page intentionally left blank.

- What are two prerequisites for attacking a system remotely? (choose two)
  A. You have to know the IP address
  B. You have to know the operating system and version
  C. You have to know the server software and version
  D. You have to know the port number you will connect to

- What is the name of the Nmap Graphical User Interface (GUI)?
  A. Nmap
  B. WinMap
  C. Zenmap
  D. NmapFE

This page intentionally left blank.

- What is a common problem of both network and host vulnerability scanners?
  A. Neither can do port scanning
  B. They both give a lot of false positives
  C. They both give a lot of false negatives
  D. Neither can determine patch level

- How does exploit software differ from a vulnerability scanner?
  A. They are the same thing
  B. Vulnerability scanners produce a lot of false negatives
  C. Exploit software exploits the target system
  D. Vulnerability scanner reports are much more reliable

This page intentionally left blank.

- At what point in your security process do you employ penetration testing?
  A. Toward the end
  B. Toward the beginning
  C. It does not really matter, as long as you do it
  D. In the middle to check your progress

- What is the difference between a red team and a tiger team?
  A. The tiger team has had more extensive training
  B. The red team has had more extensive training
  C. These are two names for the same thing
  D. Red teams are defense; tiger teams are offense

This page intentionally left blank.

Questions

SANS

Browser and Web Security Review Question

This page intentionally left blank.

Browser and Web Security

Review Questions (1)

- What are three common names for small applications that extend a browser's capability? (choose three)
  - A. Browser extensions
  - B. Browser cookies
  - C. Browser plugins
  - D. Browser add-ons
- What is a small text file placed on your PC by a server called?
  - A. Malicious
  - B. Browser plugins
  - C. Cookies
  - D. Java

This page intentionally left blank.

- How many cookie repositories are on your PC?
  A. One shared by all browsers
  B. Each browser has at least two
  C. One per browser installed
  D. One for each type of cookie

- Why was the cookie initially invented?
  A. To give web communications a "form of statefulness"
  B. To allow tracking of web surfing activity
  C. To make shopping carts work
  D. To make web communications have no statefulness

This page intentionally left blank.

- Which of the following is one of the things your browser sends to the web server each time you connect?
  A. Username and password
  B. OS and version
  C. Amount of RAM installed
  D. Browsers only receive information, they don't send it

- What can a cookie contain?
  A. A small amount of text
  B. Graphics
  C. Java, JavaScript, and ActiveX
  D. They don't contain anything

This page intentionally left blank.

- Which of the following are examples of Active Content? (choose three)
  A. ActiveX
  B. Browser plugins
  C. Java
  D. JavaScript

- What is one of the most important things to do to ensure secure coding?
  A. Always write in Java
  B. Never remove Trapdoor functions
  C. Input Validation
  D. Place username / password combinations in all code

This page intentionally left blank.

Questions

SANS

System Security Review Question

This page intentionally left blank.

- When hardening an OS, you disable the XYZ service. What is the next step you should perform?
  A. Find the next service you want to disable
  B. Delete the service from the system entirely
  C. Configure your patching software to keep the service up-to-date
  D. There is nothing more you need to do

- If you cannot disable or remove a management utility, what must you do when hardening the OS?
  A. Find a way to disable or remove the utility
  B. Accept the risk
  C. Ensure only system administrators can use the utility

This page intentionally left blank.

- At what point did Microsoft include automatic patch updates?
  A. Starting with Windows Vista
  B. Starting with Windows 2000
  C. Starting with Windows 7
  D. Starting with Windows XP service pack 2

- What does a WSUS server do?
  A. Replace the Microsoft online update server
  B. Find unneeded OS services that you can disable
  C. It is a form of vulnerability scanner
  D. That is the name of the Microsoft Web Server

This page intentionally left blank.

- Which is an example of software that performs a full system dump?
  A. There is no such software
  B. Windows System Dump
  C. Mac Time Machine
  D. WSUS

- What must happen before a WSUS server pushes patches?
  A. The administrator must download the patches
  B. An administrator must approve the patches
  C. The administrator must write the patch installer
  D. Nothing—it is automatic—that's the idea!

This page intentionally left blank.

- What is a driving factor for deploying virtualization in a data center?
  A. Less power
  B. Return On Investment (ROI)
  C. Less cooling
  D. Less floor space

- What is the name of the function that intercepts and redirects functions of a virtual OS?
  A. Hypervisor
  B. Hyperdirector
  C. Virtual Machine Director
  D. Virtual Manager

This page intentionally left blank.

- What is a common example of Software as a Service?
  A. There are no common examples—this is rare
  B. Public operating systems
  C. Public word processing
  D. Public email systems (Gmail, Hotmail, etc.)

- What is the biggest difference between Platform as a Service and Infrastructure as a Service?
  A. The number of options available to the IT staff
  B. The amount of control and responsibility the IT staff has
  C. They are the same thing
  D. The amount of responsibility the IT staff has

This page intentionally left blank.

- What is data that is on a single drive? (choose two)
  A. Uhm … it's data on a single drive
  B. Data that has not been copied to another folder
  C. Data you do not care about losing
  D. Data that has been canonicalized

- Generally speaking, to make backups more secure, what do you increase?
  A. The quantum segregation of the data
  B. The number of integrity checks on the data
  C. The amount of distance separating data backups
  D. The number of people with access to the backups

This page intentionally left blank.

Review Questions (7)

- A full backup
  A. Is the only backup worth doing
  B. Is too expensive to consider doing
  C. Takes too long to do
  D. Backs up everything

- An incremental backup
  A. Backs up everything
  B. Backs up everything changed since the last full backup
  C. Backs up everything changed since the last backup of any kind
  D. Backs up the system state, but not data

This page intentionally left blank.

- A differential backup
  A. Backs up everything
  B. Backs up everything changed since the last full backup
  C. Backs up everything changed since the last backup of any kind
  D. Backs up the system state, but not data

- Assuming a goal of full recovery, which backup type is quickest to recover?
  A. Full backup
  B. Incremental backup
  C. Differential backup
  D. Scheduled backup

This page intentionally left blank.

- Which backup type is considered "medium time to make, medium time to recover"?
  A. Full backup
  B. Incremental backup
  C. Differential backup
  D. Scheduled backup

- Assuming a goal of full recovery, which backup is normally slowest to recover?
  A. Full backup
  B. Incremental backup
  C. Differential backup
  D. Scheduled backup

This page intentionally left blank.

Appendix

SANS

Answers to Review Questions

This page intentionally left blank.

Answers

SANS

Network Security Technologies Review Question Answers

This page intentionally left blank.

- What is breaking up a network for better management known as?
  A. Network Division
  B. Segmentation
  C. Departmental Organization
  D. Compartmentalization

- What is breaking a network into security zones called?
  A. Network Division
  B. Segmentation
  C. Departmental Organization
  D. Compartmentalization

This page intentionally left blank.

- What is the most fundamental purpose of a firewall?
  A. Controlling where internal users are allowed to go on the internet
  B. To keep the person off your network that does not belong there
  C. Performing Network Address Translation (NAT)
  D. Compartmentalization

- What mechanism do you use to dictate what traffic is allowed and what traffic is denied?
  A. Every firewall is different
  B. Written policy
  C. Firewall rules
  D. Firewalls do this automatically

This page intentionally left blank.

Security Topologies

Review Questions (3)

- Which traffic type goes through a proxy firewall?
  A. Any traffic allowed by the firewall rules
  B. Traffic from inside the network to the internet
  C. Network Address Translated traffic
  D. No traffic goes through a proxy

- Which traffic type goes through a stateful inspection firewall?
  A. Any traffic allowed by the firewall rules
  B. Traffic from inside the network to the internet
  C. Network Address Translated traffic
  D. No traffic goes through a stateful inspection firewall

This page intentionally left blank.

- What is the underlying, core technology behind stateful inspection?
  A. Proxying
  B. Packet filtering
  C. Network Address Translation
  D. Stateful inspection is its own technology

- What type of systems do you put in the DMZ?
  A. A DMZ has no servers, just client PCs
  B. Servers with sensitive data only
  C. Firewalls
  D. Public access systems

This page intentionally left blank.

- When is it justified to put public access servers on an internal network?
  - **A. Never**
  - B. As long as they are fully patched
  - C. When that is the most convenient way to access them
  - D. We always do that

- What is the acronym for an automated system that watches for signs of an attack called?
  - A. DNS
  - B. IPS
  - C. ISP
  - **D. IDS**

This page intentionally left blank.

- When an IDS watches for patterns of an attack in packets, what is it doing?
  - **A. Signature analysis**
  - B. Intrusion Detection and Prevention (IDP)
  - C. You can't do that
  - D. Anomaly analysis

- To stop attacks, an Intrusion Protection System must also be what?
  - A. DNS
  - B. IPS
  - C. ISP
  - **D. IDS**

This page intentionally left blank.

Security Topologies

Review Questions (7)

- When there is no valid reason for anyone to access an IT resource, what do you call that resource?
  - **A. A Honeypot**
  - B. A sweetspot
  - C. Why would you do that?
  - D. A nonresource
- What is a common name for a solution that prevents inappropriate web surfing?
  - A. Unnecessary
  - **B. Content filter**
  - C. WebSense
  - D. Surfing Ruleset

This page intentionally left blank.

- What is another name for an all-in-one security appliance?
  A. A Honeypot
  B. This is nothing but a marketing term—they don't really exist
  C. Layered Management Application Firewall
  D. Unified Threat Management (UTM)

- What is the less common but more accurate name for a sniffer?
  A. Protocol Dissector
  B. Wireshark
  C. Protocol Analyzer
  D. Packet Capture

This page intentionally left blank.

- What are two prerequisites for attacking a system remotely? (choose two)
  - **A. You have to know the IP address**
  - B. You have to know the operating system and version
  - C. You have to know the server software and version
  - **D. You have to know the port number you will connect to**

- What is the name of the Nmap Graphical User Interface (GUI)?
  - A. Nmap
  - B. WinMap
  - **C. Zenmap**
  - D. NmapFE

This page intentionally left blank.

- What is a common problem of both network and host vulnerability scanners?
  A. Neither can do port scanning
  B. They both give a lot of false positives
  C. They both give a lot of false negatives
  D. Neither can determine patch level

- How does exploit software differ from a vulnerability scanner?
  A. They are the same thing
  B. Vulnerability scanners produce a lot of false negatives
  C. Exploit software exploits the target system
  D. Vulnerability scanner reports are much more reliable

This page intentionally left blank.

- At what point in your security process do you employ penetration testing?
  - **A. Toward the end**
  - B. Toward the beginning
  - C. It does not really matter, as long as you do it
  - D. In the middle to check your progress

- What is the difference between a red team and a tiger team?
  - A. The tiger team has had more extensive training
  - B. The red team has had more extensive training
  - **C. These are two names for the same thing**
  - D. Red teams are defense; tiger teams are offense

This page intentionally left blank.

Answers

Browser and Web Security Review Question Answers

This page intentionally left blank.

Browser and Web Security

Review Questions (1)

- What are three common names for small applications that extend a browser's capability? (choose three)
  - **A. Browser extensions**
  - B. Browser cookies
  - **C. Browser plugins**
  - **D. Browser add-ons**

- What is a small text file placed on your PC by a server called?
  - A. Malicious
  - B. Browser plugins
  - **C. Cookies**
  - D. Java

This page intentionally left blank.

Review Questions (2)

- How many cookie repositories are on your PC?
  A. One shared by all browsers
  B. Each browser has at least two
  C. <u>One per browser installed</u>
  D. One for each type of cookie

- Why was the cookie initially invented?
  A. <u>To give web communications a "form of statefulness"</u>
  B. To allow tracking of web surfing activity
  C. To make shopping carts work
  D. To make web communications have no statefulness

This page intentionally left blank.

Browser and Web Security

Review Questions (3)

- Which of the following is one of the things your browser sends to the web server each time you connect?
  - A. Username and password
  - **B. OS and version**
  - C. Amount of RAM installed
  - D. Browsers only receive information, they don't send it
- What can a cookie contain?
  - **A. A small amount of text**
  - B. Graphics
  - C. Java, JavaScript, and ActiveX
  - D. They don't contain anything

This page intentionally left blank.

- Which of the following are examples of Active Content? (choose three)
  - **A. ActiveX**
  - B. Browser plugins
  - **C. Java**
  - **D. JavaScript**

- What is one of the most important things to do to ensure secure coding?
  - A. Always write in Java
  - B. Never remove Trapdoor functions
  - **C. Input Validation**
  - D. Place username / password combinations in all code

This page intentionally left blank.

Answers

SANS

System Security Review Question Answers

This page intentionally left blank.

- When hardening an OS, you disable the XYZ service. What is the next step you should perform?
  A. Find the next service you want to disable
  B. Delete the service from the system entirely
  C. Configure your patching software to keep the service up-to-date
  D. There is nothing more you need to do

- If you cannot disable or remove a management utility, what must you do when hardening the OS?
  A. Find a way to disable or remove the utility
  B. Accept the risk
  C. Ensure only system administrators can use the utility

This page intentionally left blank.

- At what point did Microsoft include automatic patch updates?
  A. Starting with Windows Vista
  B. Starting with Windows 2000
  C. Starting with Windows 7
  D. Starting with Windows XP service pack 2

- What does a WSUS server do?
  A. Replace the Microsoft online update server
  B. Find unneeded OS services that you can disable
  C. It is a form of vulnerability scanner
  D. That is the name of the Microsoft Web Server

This page intentionally left blank.

- Which is an example of software that performs a full system dump?
  A. There is no such software
  B. Windows System Dump
  C. Mac Time Machine
  D. WSUS

- What must happen before a WSUS server pushes patches?
  A. The administrator must download the patches
  B. An administrator must approve the patches
  C. The administrator must write the patch installer
  D. Nothing—it is automatic—that's the idea!

This page intentionally left blank.

- What is a driving factor for deploying virtualization in a data center?
  A. Less power
  B. Return On Investment (ROI)
  C. Less cooling
  D. Less floor space

- What is the name of the function that intercepts and redirects functions of a virtual OS?
  A. Hypervisor
  B. Hyperdirector
  C. Virtual Machine Director
  D. Virtual Manager

This page intentionally left blank.

- What is a common example of Software as a Service?
  A. There are no common examples—this is rare
  B. Public operating systems
  C. Public word processing
  D. Public email systems (Gmail, Hotmail, etc.)

- What is the biggest difference between Platform as a Service and Infrastructure as a Service?
  A. The number of options available to the IT staff
  B. The amount of control and responsibility the IT staff has
  C. They are the same thing
  D. The amount of responsibility the IT staff has

This page intentionally left blank.

- What is data that is on a single drive? (choose two)
  - **A. Uhm ... it's data on a single drive**
  - B. Data that has not been copied to another folder
  - **C. Data you do not care about losing**
  - D. Data that has been canonicalized

- Generally speaking, to make backups more secure, what do you increase?
  - A. The quantum segregation of the data
  - B. The number of integrity checks on the data
  - **C. The amount of distance separating data backups**
  - D. The number of people with access to the backups

This page intentionally left blank.

- A full backup
  A. Is the only backup worth doing
  B. Is too expensive to consider doing
  C. Takes too long to do
  D. Backs up everything

- An incremental backup
  A. Backs up everything
  B. Backs up everything changed since the last full backup
  C. Backs up everything changed since the last backup of any kind
  D. Backs up the system state, but not data

This page intentionally left blank.

- A differential backup
  A. Backs up everything
  B. Backs up everything changed since the last full backup
  C. Backs up everything changed since the last backup of any kind
  D. Backs up the system state, but not data

- Assuming a goal of full recovery, which backup type is quickest to recover?
  A. Full backup
  B. Incremental backup
  C. Differential backup
  D. Scheduled backup

This page intentionally left blank.

- Which backup type is considered "medium time to make, medium time to recover"?
  A. Full backup
  B. Incremental backup
  C. Differential backup
  D. Scheduled backup

- Assuming a goal of full recovery, which backup is normally slowest to recover?
  A. Full backup
  B. Incremental backup
  C. Differential backup
  D. Scheduled backup

This page intentionally left blank.

Course Resources and Contact Information

AUTHOR CONTACT
Keith Palmgren
keith@netip.com

SANS INSTITUTE
11200 Rockville Pike
Suite 200
North Bethesda, MD 20852
301.654.SANS(7267)

Cyber Defense RESOURCES
https://cyber-defense.sans.org/
Twitter: @SANSDefense

SANS EMAIL
GENERAL INQUIRIES: info@sans.org
REGISTRATION: registration@sans.org
TUITION: tuition@sans.org
PRESS/PR: press@sans.org

Many of the slide graphics in this course are provided through a royalty-free license with PresentationPro. http://www.presentationpro.com/