

507.4

Auditing Private and Public Clouds, Containers, and Networks

2021

SANS

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

August 10, 2021

PLEASE READ THE TERMS AND CONDITIONS OF THIS COURSEWARE LICENSE AGREEMENT ("CLA") CAREFULLY BEFORE USING ANY OF THE COURSEWARE ASSOCIATED WITH THE SANS COURSE. THIS IS A LEGAL AND ENFORCEABLE CONTRACT BETWEEN YOU (THE "USER") AND SANS INSTITUTE FOR THE COURSEWARE. YOU AGREE THAT THIS AGREEMENT IS ENFORCEABLE LIKE ANY WRITTEN NEGOTIATED AGREEMENT SIGNED BY YOU.

With this CLA, SANS Institute hereby grants User a personal, non-exclusive license to use the Courseware subject to the terms of this agreement. Courseware includes all printed materials, including course books and lab workbooks, as well as any digital or other media, virtual machines, and/or data sets distributed by SANS Institute to User for use in the SANS class associated with the Courseware. User agrees that the CLA is the complete and exclusive statement of agreement between SANS Institute and you and that this CLA supersedes any oral or written proposal, agreement or other communication relating to the subject matter of this CLA.

BY ACCEPTING THIS COURSEWARE, USER AGREES TO BE BOUND BY THE TERMS OF THIS CLA. BY ACCEPTING THIS SOFTWARE, USER AGREES THAT ANY BREACH OF THE TERMS OF THIS CLA MAY CAUSE IRREPARABLE HARM AND SIGNIFICANT INJURY TO SANS INSTITUTE, AND THAT SANS INSTITUTE MAY ENFORCE THESE PROVISIONS BY INJUNCTION (WITHOUT THE NECESSITY OF POSTING BOND) SPECIFIC PERFORMANCE, OR OTHER EQUITABLE RELIEF.

If User does not agree, User may return the Courseware to SANS Institute for a full refund, if applicable.

User may not copy, reproduce, re-publish, distribute, display, modify or create derivative works based upon all or any portion of the Courseware, in any medium whether printed, electronic or otherwise, for any purpose, without the express prior written consent of SANS Institute. Additionally, User may not sell, rent, lease, trade, or otherwise transfer the Courseware in any way, shape, or form without the express written consent of SANS Institute.

If any provision of this CLA is declared unenforceable in any jurisdiction, then such provision shall be deemed to be severable from this CLA and shall not affect the remainder thereof. An amendment or addendum to this CLA may accompany this Courseware.

SANS acknowledges that any and all software and/or tools, graphics, images, tables, charts or graphs presented in this Courseware are the sole property of their respective trademark/registered/copyright owners, including:

AirDrop, AirPort, AirPort Time Capsule, Apple, Apple Remote Desktop, Apple TV, App Nap, Back to My Mac, Boot Camp, Cocoa, FaceTime, FileVault, Finder, FireWire, FireWire logo, iCal, iChat, iLife, iMac, iMessage, iPad, iPad Air, iPad Mini, iPhone, iPhoto, iPod, iPod classic, iPod shuffle, iPod nano, iPod touch, iTunes, iTunes logo, iWork, Keychain, Keynote, Mac, Mac Logo, MacBook, MacBook Air, MacBook Pro, Macintosh, Mac OS, Mac Pro, Numbers, OS X, Pages, Passbook, Retina, Safari, Siri, Spaces, Spotlight, There's an app for that, Time Capsule, Time Machine, Touch ID, Xcode, Xserve, App Store, and iCloud are registered trademarks of Apple Inc.

PMP® and PMBOK® are registered trademarks of PMI.

SOF-ELK® is a registered trademark of Lewes Technology Consulting, LLC. Used with permission.

SIFT® is a registered trademark of Harbingers, LLC. Used with permission.

Governing Law: This Agreement shall be governed by the laws of the State of Maryland, USA.

AUD507.4

Auditing & Monitoring Networks, Perimeters, & Systems

SANS

Auditing Private and Public Clouds, Containers, and Networks


© 2021 Risenhoover Consulting, Inc. | All Rights Reserved | Version G01_03

Welcome to Section Four of the SANS AUD507 course! This course is written, maintained, and frequently taught by Clay Risenhoover. I am always looking for ways to improve this courseware. If you have questions or suggestions for how to improve the course, or if you need any additional materials referenced during the class, please let me know. If you find errors or inaccuracies in the course books, I encourage you to pass those on to me. You can email me at clay@risenhooverconsulting.com. Please put either “SANS” or “AUD507” in the subject line, to ensure I see the email.

The entire content of this and every other volume in this course is © 2021 Risenhoover Consulting, Inc.

August 10, 2021

TABLE OF CONTENTS	PAGE
Introduction to Cloud Technologies	3
Private Clouds and Hypervisors	9
Exercise 4.1: Auditing Hypervisors	37
The Public Cloud	38
Containers	52
Exercise 4.2: Auditing Docker Security	76
Networks and Firewalls	77
Exercise 4.3: Wireshark, Switch Configuration Symptoms and Device Configuration Auditing	142
Wi-Fi and VPNs	143
Public Services	155
Exercise 4.4: Auditing Public Services	177

 AUD507 | Auditing & Monitoring Networks, Perimeters, & Systems 2

This page intentionally left blank.

August 10, 2021

Course Roadmap

- Enterprise Audit Fundamentals; Discovery and Scanning Tools
- PowerShell, Windows System, and Domain Auditing
- Advanced UNIX Auditing and Monitoring
- **Auditing Private and Public Clouds, Containers, and Networks**
- Auditing Web Applications
- Audit Wars!

Section Four

1. Private Clouds

- Introduction to Cloud Technologies
- Private Clouds and Hypervisors
- Exercise 4.1 - Auditing Hypervisors

2. The Public Cloud

3. Containers

4. Networks and Firewalls

5. WIFI and VPNs

6. Public Services

This page intentionally left blank.

August 10, 2021

What Is the Cloud?

- NIST SP 800-145:
 - Ubiquitous, convenient, on-demand:
 - Shared pool:
 - Networks, servers, storage, applications, and services
 - Rapid provisioning
- Another definition:
 - I have no idea where my stuff is



According to NIST Special Publication 800-145, the cloud is “...a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction” (<https://u.aud507.com/1-7>). NIST further explains that there are a number of characteristics that cause a service to fit into this definition. These include:

- On-demand self-service provisioning of resources by the consumer
- Broad network access (which allows for utilization and management from a large number of types of clients)
- Resource pooling to allow for reservations or simply dynamic reassignment of resources dependent on utilization
- Rapid elasticity (which refers specifically to the ability to rapidly allocate and deallocate server and resource instances without provider intervention)
- Typically, an implementation of a metered service similar to the “old days” where we paid for time sharing on systems

As nice as this definition is, for most people the cloud simply means, “I have no idea where my stuff is!”

Typical Cloud Strategies

- Private
 - Organization hosts its own infrastructure in its own data centers
- Public
 - Organization hosts computing and storage on someone else's systems
 - AWS/Microsoft Azure, etc.
- Hybrid
 - Mixture of private cloud with some public cloud services

It's important for auditors to understand the strategies commonly employed by enterprises which utilize cloud technologies. In a private cloud implementation, the enterprise owns and operates all the virtualization hardware and infrastructure. Simply put, the private cloud strategy means adding virtualization technologies to existing enterprise architecture.

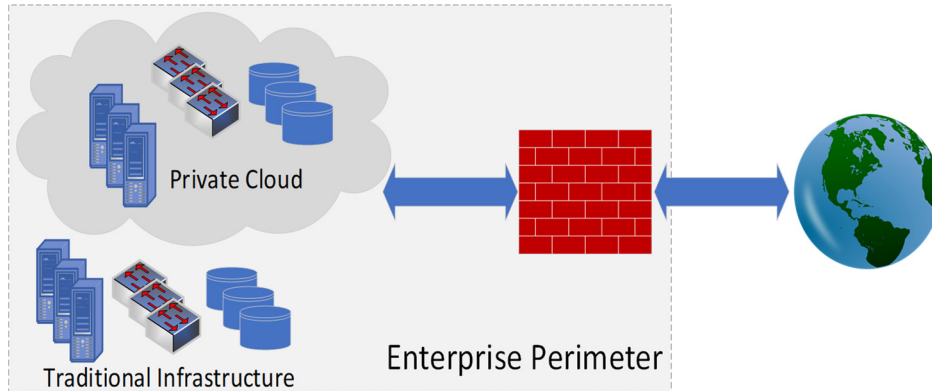
In a public cloud implementation, the enterprise eliminates the need for much of their own infrastructure by utilizing cloud providers for all of their critical functions, such as email, website hosting, productivity, and accounting software. Even directory services and authentication can be outsourced.

Many organizations "dip their toe in the water" of cloud technology by designing hybrid strategies. In a hybrid deployment, the enterprise maintains some private cloud resources while using public cloud offerings for other functions. In many ways, this is the most difficult strategy to secure, because it becomes very difficult to define "the perimeter" of the enterprise network.

August 10, 2021

Typical Strategies: Private Cloud

- Organization owns/controls everything
- Audit the hypervisor/network – traditional audit techniques



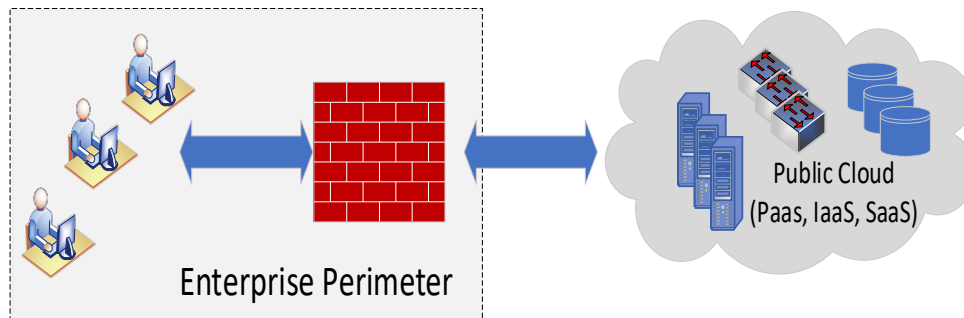
In a private cloud deployment, the enterprise owns and controls all of the information systems they use. There will often be some mix of traditional hardware-based infrastructures, such as physical servers, switches, and routers, combined with virtual machines and virtual networks. IT staff will manage the hardware either through hands-on techniques or from within the enterprise's own network. Management traffic and business data should not normally need to leave the organization's internal network.

This model has a well-defined perimeter because all of the important resources exist within the organization's premises. It is also relatively easy to audit because the job of identifying scope is quite manageable. The auditor will examine the configuration of the traditional, hardware-based devices, the virtualization hypervisors, the virtual network, and the virtual servers using the techniques taught in this course.

This model is fairly rare today since many enterprises have adopted at least *some* cloud-based services as part of their business systems.

Typical Strategies: Public Cloud

- Organization hosts computing, storage, and applications on someone else's systems
- AWS, Microsoft Azure, Google Apps, etc.
- Audit the APIs, access controls, logging, and security features



In a public cloud deployment scenario, the organization uses external cloud service providers for all of their critical infrastructure and applications. For example, a company may use Google Apps for its productivity software, Microsoft Azure Active Directory for authentication and directory services, Dropbox for file sharing, and have web servers running in Amazon Web Services.

In this model, the company keeps few, if any, information assets on the premises. Systems managers use application programming interfaces (APIs) and web applications to manage applications and servers. Because the assets largely exist outside the organization, auditing is performed differently. The auditor will check the security of keys used to access management interfaces on services, user authentication methods and the access control mechanisms used by applications. The auditor will also examine how the services are configured to do logging, what security audit trails exist, and whether the features of each cloud service have been properly configured.

Security testing of public cloud services will always require some sort of permission from the provider before testing begins. Failure to gain proper permission could result in the service provider denying the company access to the service for a time. Effectively, a poorly implemented audit could result in availability issues for the company.

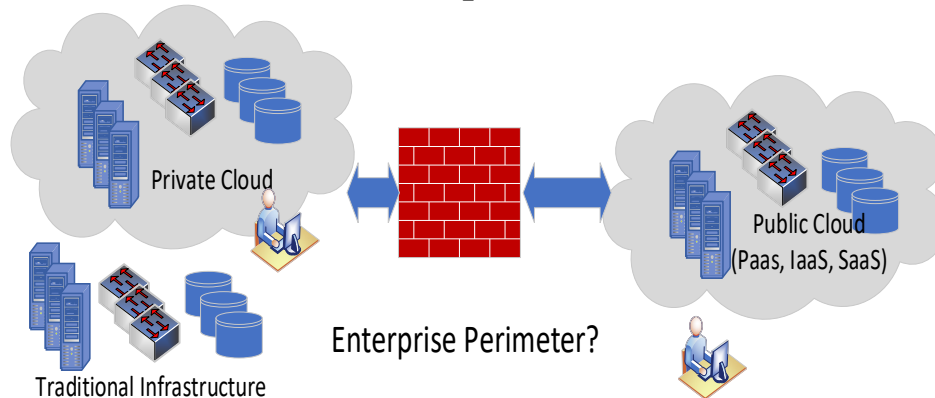
The Center for Internet Security offers benchmark documents for properly securing some of the popular cloud platforms and the virtual systems which can be run on them.

See the benchmarks at:

<https://u.aud507.com/1-3>

Typical Strategies: Hybrid Cloud

- Many organizations start here
- Most difficult to secure – multiple data flows to control



Hybrid deployments, which combine the use of private and public cloud services by the same organization, are a common first-step into the use of public cloud services. They, unfortunately, are probably the most difficult to secure and audit properly. The interaction between these private and public elements usually means that a lot of different traffic types will be allowed to pass the “perimeter” firewall. The fact that the organization’s proprietary data exists both on premises and in the cloud makes the idea of a perimeter much harder to define. The organization may be allowing authentication traffic, email, file storage, database access, and other sensitive traffic through the firewall with very little control.

Before auditing this type of deployment, the auditor must fully understand the expected flow of information between the “inside” and “outside” networks and then verify that ONLY those flows are being allowed. The auditor must also obtain good inventory and scoping information to ensure full coverage of all services used. Testing permission is still required before auditing public services.

Private Clouds: Virtualization Definitions

- Hypervisor: Software interface between the hardware and the virtualized operating system
- Type-I Hypervisor: Bare metal
 - Very thin virtualization layer to manage physical resources
- Type-II Hypervisor: Hosted
 - Software application running within a hosting OS

The software that manages the resources available and distributes them to the virtualized operating systems is known as a hypervisor. The hypervisor usually provides a management interface or application programming interface (API) to allow the administrator to configure the virtual systems. Administrators can define the amount of RAM available to the virtualized system, the size and number of virtual hard drives, the number of network interfaces, the configuration of network interfaces, and the connection to other hardware available to the system. Hypervisors are differentiated into two general categories: Type-I and Type-II.

Type-I hypervisors are also known as “bare metal” hypervisors. The name derives from the belief that the virtualization software runs directly on the hardware with no operating system beneath it. While this presents a good mental picture, in reality there is an operating system; most Type-I hypervisors run on top of a very lightweight Linux distribution. The use of a lightweight operating system has one primary, and very large, advantage over Type-II hypervisors. Type-I hypervisors can make almost all of the resources of the hardware available for use in virtualization since there is so little overhead from the host operating system.

Type-II hypervisors are also known as “hosted” hypervisors. This name derives from the fact that the virtualization software is installed into and runs under the control of a host operating system. Examples of Type-II hypervisors are VMware Workstation, QEMU, and VirtualBox. The advantage of this approach is that the computer on which the hypervisor is installed can be used for other purposes at the same time that the hypervisor is running. The disadvantage is that the hosting operating system will typically consume a substantial number of resources, in addition to ultimately being in control of how much CPU time the hypervisor may use to virtualize operating systems.

More Definitions

- **Virtualization Extensions: Hardware virtualization optimizations**
 - Enabled through UEFI or BIOS interface
 - Usually, customizations that allow for very rapid task switching in the hardware
- **SAN: Storage Area Network**
 - Mandatory for an enterprise virtualization solution
 - Provides a high-speed shared storage medium
 - Without this, failover is typically not possible

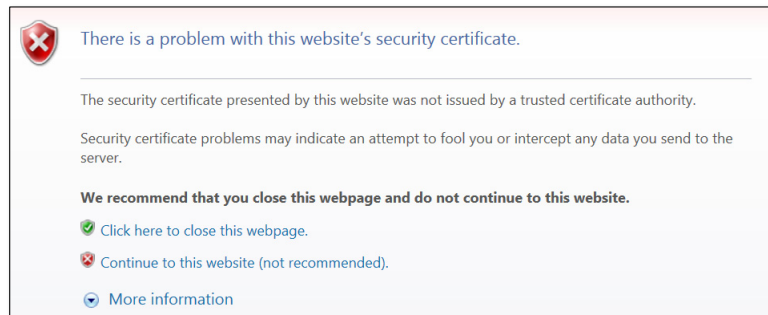
Virtualization extensions are specialized features that processor and motherboard vendors include in the hardware that can be optionally enabled. These features are specifically designed with virtualization tasks in mind. For example, in a virtualization application, we need to simulate multiple operating systems running on multiple pieces of hardware. The virtualization extensions add hardware-based support for this, effectively allowing the virtualization layer to partition the CPU and treat it as though there are a number of separate, physical CPUs, even though there may only be one physical CPU present. This turns out to have tremendous performance benefits because it moves the “partitioning” of the CPU out of the software and into the hardware itself. In practice, the hardware simulates that each core in a processor is a separate physical processor.

These extensions are typically enabled or disabled through the BIOS or UEFI management interface. The settings can be in a number of locations in the BIOS/UEFI settings. They are sometimes found under the “Security” settings, even though this is somewhat counterintuitive. They can bear a number of names, including “VT-x,” “Intel Virtualization Technology,” and “AMD V,” depending on the BIOS vendor and the chip manufacturer.

A SAN, or Storage Area Network, is pretty much a requirement for any enterprise virtualization installation. In order to provide failover between the hypervisors, the systems will need a shared medium for storing the image of the hard drive. Without this, it is not possible to fail over to another system. Unless the virtualization solution provides a mechanism to keep the hard drive images in lockstep for any changes, there would be no way for another system to take over the load without interruption of services.

Common Issues (I)

- Self-signed certificates
 - Directly impact confidentiality and integrity
 - Attacker can act as a man in the middle
 - Administrators train themselves to accept invalid certificates



Some issues are common to any kind of virtualization installation. The first is that of so-called “self-signed” certificates. While it is literally true that the server or service was configured to sign its own certificate, a better way to think about this is that the administrators have not bothered to install a certificate at all. In other words, while there is a certificate, it’s not because the administrators put it there. They simply accepted the default settings.

This is a rather important issue. If the certificate has not been signed by a trusted CA, any administrator connecting will be informed that the certificate is not trusted. What will he do? Tell the administration tool or web browser to ignore the invalid certificate and connect anyway. This creates a situation where an attacker who is able to inject himself as a man in the middle can now present his own certificate. This will most likely go undetected by the administrators since they have trained themselves to accept the error. Therefore, this is both a confidentiality issue, since the attacker can view the administration traffic unencrypted and potentially steal confidential data, and an integrity issue, since the attacker now has credentials that can be used to modify the data or the configuration. Availability comes into play as well, since the attacker with these credentials could shut down, delete, or otherwise make unavailable the virtualized environment.

Common Issues (2)

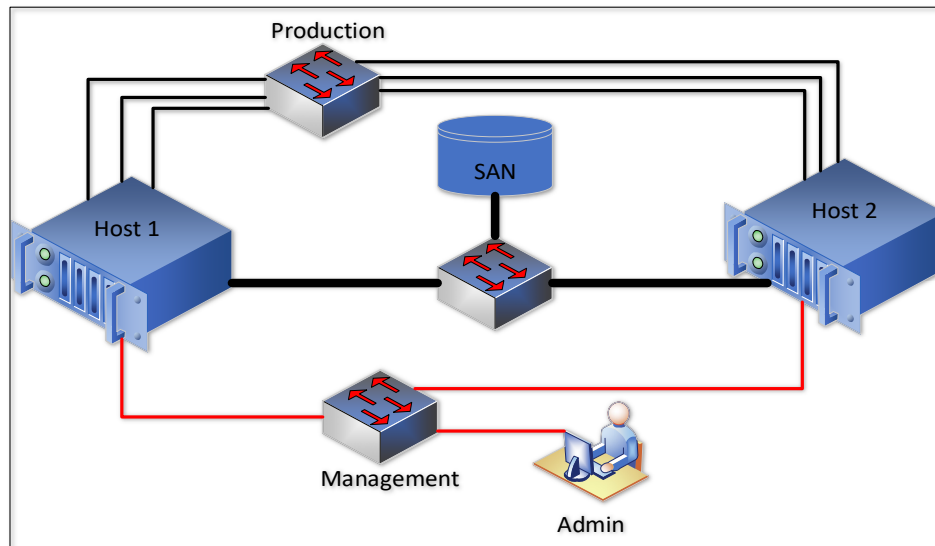
- Isolation of management interfaces
 - Principle of least privilege
 - Can impact confidentiality and integrity if credentials are guessed
 - Can impact availability if the administrative accounts are locked out
- SAN links on dedicated switches
 - Data on SAN links are *not* encrypted
 - VLANs can (and will) bleed data
- Network Adapters
 - Sufficient network adapters to satisfy operational requirements

Another issue is that of isolating the management interfaces. If someone can connect to the management interface, there is always the potential of an attack against the credentials, which impacts confidentiality, integrity, and availability. The impact is the same as that described in the discussion on self-signed certificates. Availability is also involved for another reason. If the system implements any kind of credential lockout, the attacker has the opportunity to impact our ability to administer our own systems until the administrative accounts are unlocked.

As you already know, the SAN acts as the shared data storage repository for a cluster of virtualization servers. This allows failover capabilities to successfully shift a running VM from one piece of hardware to another. For performance reasons, these SAN links are never encrypted. In fact, it is unusual for a SAN to provide network-level encryption options since this would impact performance. Since this data isn't encrypted, it is vital for confidentiality that the deployment guarantees that the data will not inadvertently be leaked. It is best that the network connections to and from the SAN are made through dedicated switching hardware. While it is possible to use VLANs to try to enforce isolation on the data, switches are known to leak or bleed data between VLANs. We will discuss this in more detail during later in this course section.

One more common issue that we will mention is that of network interfaces. There is no formula to calculate precisely how many interfaces a virtualization server ought to have, but at a bare minimum it must be two. Remember that the administration interfaces and SAN interfaces must be isolated from the virtual machines being hosted. Likely, two is insufficient, though. This all depends on the total utilization.

Virtualization Networks



The network diagram in this slide represents a VERY simple virtualization deployment. These are the features to note:

- Each host in the cluster has a dedicated high-speed network connection to the storage area network (SAN). Isolating the SAN traffic allows for maximum data throughput, and it protects the confidentiality of the sensitive (and unencrypted) data on that network.
- Each host has a dedicated management interface on a subnet dedicated to administrative traffic. Only administrator workstations and managed devices should exist on this network.
- Each host has a sufficient number of network interfaces dedicated for use by the virtual machines. The number of interfaces required will vary, but it should be enough to ensure that the virtual machines have sufficient bandwidth available.

More Common Issues (I)

- **Administrative Credentials**
 - Centralized authentication, likely to domain
 - Domain-level groups used for RBAC
 - Local credentials very strong and escrowed
- **Time sync and centralized logging into SIEM**

Regardless of the brand of virtualization solution, we would also like to verify that the administrative credentials are strong and well controlled. One of the best ways to satisfy this is to leverage a central authentication system, like Active Directory, as the authentication service for the virtualization credentials. Doing so means that we automatically benefit from all of the password controls, and possibly even two-factor authentication requirements, of the domain. Not only will this allow us to use domain-level groups to provide for role-based access control (RBAC), but it also means that when an administrator leaves the organization, we may simply disable his account rather than change all of the passwords for the management systems.

Virtualization servers will also have “local” credentials that can be used to administer the servers individually. We must also verify that these local credentials are very strong and that they are escrowed for emergencies. Sometimes you will find that these credentials are still set to the default or are simply weak. This is usually because of an oversight since these are only used to build the original virtualization cluster, after which different credentials are used for management.

Regardless of the server we use, we should also verify that the system has been configured to synchronize to a good time source. This facilitates the authentication services but also ensures that log timestamps make sense. We should also verify that the logs will be aggregated to a central repository, typically the enterprise’s security incident and event management (SIEM) system.

More Common Issues (2)

- Appropriately sized servers for failover
- Business continuity/disaster recovery
- Backups
 - Different from replication
 - “Immutable” backups to protect against ransomware

What if one node goes down?



Host 1
CPU: **83%**
RAM: 59%



Host 2
CPU: **91%**
RAM: 65%



Host 3
CPU: **88%**
RAM: 52%



Host 4
CPU: **90%**
RAM: 45%

When our organization virtualizes many servers down to a few, disaster recovery and continuity becomes even more important. Virtualization can save us a great deal of money, but it can also end up creating a great deal of risk if we have not properly accounted for things that can go wrong. For this reason, despite the advertisements that talk about replacing a room full of servers with a single machine, it is obviously a bad idea to have a single virtualization server running everything. If it fails, everything fails.

Therefore, we must always have at least two servers. If we have only two servers, then both servers must be operating at less than 50% total utilization, or we will be unable to fail over without a serious performance impact. Two servers, however, are unlikely to be able to handle all of our needs. As a result, we will have many servers operating within clusters. Our audit should investigate whether or not the individual systems within the cluster have enough resources left over after normal operations to handle additional load should a part of the cluster go down.

With this in mind, we are also interested in the total load across the cluster. If we have just enough resources across the cluster to handle the failure of any single system, we are at a breaking point. If even one or two more virtualized operating systems are added in, will we cross a threshold where we can no longer successfully fail over without a significant loss of performance? It is wise to ask for documentation that proves that the failure scenarios have actually been tested; when is the last time that a few of the servers in a cluster were powered off abruptly to simulate a failure? Was this test successful and were all business continuity and recovery requirements met?

The organization must also be performing backups of virtual machines. Many enterprises now use replication of VMs from one cluster to another as a disaster recovery control. This is a good thing to do, but it does not replace file-level backups for recovery from things like accidental file deletion or ransomware attacks. Many vendors now offer “immutable” backup options on their storage appliances, which can be used to prevent ransomware from encrypting the online backups.

VMware-Specific

- vSphere Hypervisor
 - 6.5 or 6.7 – End of life October 2022
 - (Covid extension)
 - 7.0 – Released April 2020 – End of life April 2025
- vCenter Server
 - Can be appliance (VCSA) or physical
- Best practices guide has some bad advice in it
 - Virtualizing everything creates potential continuity issues
 - Making Domain Administrators the VMware Administrators is clearly wrong
- Strong rights concept
 - Domain-level groups to grant privileges within VMware infrastructure



VMware is, by far, the market leader in enterprise data center virtualization. One of the very first concerns is whether we are running the most current version of the product. Today, we should be running version 7 or above, although support for the 6.5/6.7 versions was extended into 2022 due to the Covid pandemic. Because version 6.5 still supports some older CPUs and chipsets, the existence of that version in your environment could be a clue that you have hardware nearing obsolescence.

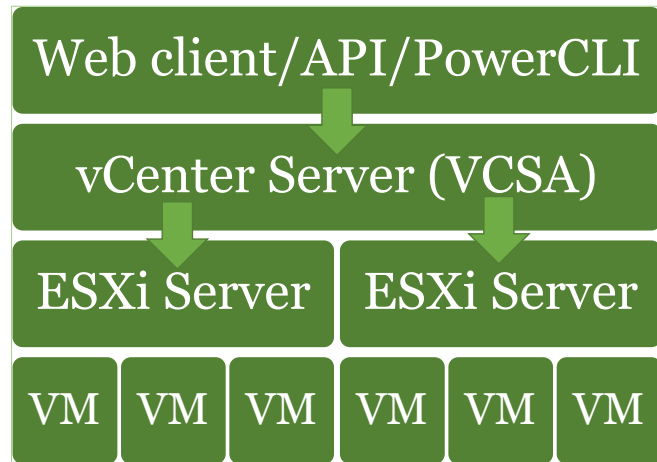
A basic requirement for an enterprise installation will include a vCenter Server. This server can be run as a standalone system on a Microsoft Server OS or as a Linux-based virtual appliance, known as the vCenter server appliance (VCSA). While the appliance is very attractive from a cost perspective, the standalone system allows us to manage our virtualization infrastructure even if the backing datastore (SAN) is offline. Of course, it does create a point of failure, too, so it is a matter of tradeoffs.

VMware publishes a best practices guide for their virtualization solution. Recognize that while many of these are good practices, some are best for VMware, but not necessarily your organization. For example, the best practices guide urges you to virtualize absolutely everything. This can create some very serious continuity issues unless it is done carefully and with full redundancy for all components—including the SAN.

The best practices guide also mentions making Domain Administrators the VMware Administrators. This actually is not a good security practice. Instead, while we should be using domain-level groups to manage VMware security, Domain Administrators are probably not the right people to administer our virtual environment. These domain-level groups that are created can be granted very granular rights within the VMware environment, and that provides for very good principle of least privilege controls.

VMWare Infrastructure Components

- vCenter server manages ESXi hypervisors
- ESXi hypervisors host the virtual machines
- In production, you'll query the VCSA for host information



A large majority of the measurements we take audits are related to ESXi hypervisor systems and the vCenter servers which are used to manage them.

You will normally make queries against the vCenter server, which is used as a central management point for all the ESXi hypervisors in the environment. The ESXi servers house the virtual machines (VMs).

Due to licensing constraints, in the exercises, you will query a single ESXi server directly. In production, you would likely run queries for a number of ESXi hosts against the vCenter server which manages them. We often use foreach loops to get the same settings for every host managed by the vCenter machine.

You will often see vCenter deployed as a virtual appliance called a vCenter Server Appliance (VCSA). We will use VCSA as a shorthand for your vCenter server, even though it could actually be deployed as a physical machine.

VMware Issues

- **Licensing**
 - Traditionally licensed per socket
 - Since March 2020, socket license allows UP TO 32 cores
- **Management interface**
 - HTML-5 web interface — how is it secured?
 - Is SSH enabled? How is it locked down? Are we using keys or passwords?
- **Training**
 - Are administrators trained before they are asked to administer systems?
- **VMKernel ports versus Virtual Machine ports**
 - The two should not be mixed without very good reason

Licensing within the VMware landscape can be very complex. I have seen a number of installations where the license installed on the servers does not fully utilize the hardware available, or it doesn't meet the DR/BCP requirements of the organization. Inquire about the requirements and hardware and then compare those answers to what's shown in the licensing status for the server.

For years, the management client for VMware was a separate application called vSphere. VMware now uses an HTML-5 interface. Administration can also happen using the CLI via SSH, or over the network using the VMware API or PowerCLI modules. If the host is in "lockdown" mode, SSH and possibly (in strict lockdown mode) even direct console UI (DCUI) logons can be disabled. If a host in strict lockdown mode loses connection to the VCenter server, it must be re-installed.

A surprising issue that seems to come up frequently when dealing with VMware installations is the training of the administrators. Anecdotally, it seems that many administrators handling these systems have never had formal training in the technology. If asked, they will normally express that they would love training, which could be a good recommendation in an audit report.

To emphasize the importance of restricting access to the administrative interfaces mentioned in the general section, within VMware, if someone can connect to the web interface, he can begin guessing credentials. If he can guess the credentials, he then can browse to the datastore where the virtual machines are stored. From there, he can download an image of the hard drive, completely compromising confidentiality.

The administration and SAN data all pass over "VMKernel" network ports. These are the ports that must be protected. Another specific protection is verifying that the network interface cards (NICs) used for virtual machines are separated from VMKernel ports without a very specific, needed exception.

VMWare Audit Tools

- VMware:
 - Web UI (auditors can be granted read-only access)
 - SSH (probably not appropriate for auditors)
 - VMware API
 - PowerCLI
 - As Built Report PowerShell modules
 - Robware RV Tools

Auditors have a number of options for gathering audit evidence from hypervisor environments. For VMware, data can be obtained from the Web UI and SSH, but these are sub-optimal for evidence gathering, since the auditor would need to use screenshots for most information.

The VMware API allows for programmatic access to vSphere information. It is used by many tools, like Robware RVTools.

PowerCLI is perfect for gathering information from hypervisors using scripts or one-off commands and saving the results in any format supported by PowerShell (which is pretty much any format).

August 10, 2021

Audit Tools: PowerCLI for VMware

- PowerShell module for administering/querying vCenter or ESXi servers
- Run against vCenter, it can query a whole datacenter
- Settings can be queried and saved to a file for later analysis

```
PS C:\Scripts> Connect-VIServer -Server 10.50.7.31 -Credential $cred
Name                               Port  User
----                               -
10.50.7.31                         443   root
PS C:\Scripts> Get-VMHost
Name                               ConnectionState PowerState NumCpu CpuUsageMhz
----                               -
10.50.7.31                         Connected      PoweredOn      2      46
```

PowerCLI is a downloadable set of modules for managing and querying a VMware environment. Information on pretty much every aspect of the datacenter, cluster, datastores, and ESXi hosts is easily queried with PowerCLI.

Audit Tools: AsBuiltReport for VMware ESXi/vSphere

- PowerShell module to build a report on the state of a virtualization environment
- Other datacenter products supported as well
- Pre-installed on your Windows 10 VM

```
New-AsBuiltReport -Report VMware.ESXi -Target 'esxi'  
-Format HTML -OutputPath 'C:\Users\auditor\Documents\  
-Timestamp -Credential $cred
```

AsBuiltReport is a set of PowerShell modules which can query datacenter devices and build inventory and setting reports into text or HTML files, or even to a Word document. It uses the VMware PowerShell modules to query either a vCenter server or ESXi host for a wealth of information which can be useful to an auditor.

The AsBuiltReport module is installed on your Windows 10 VM.

August 10, 2021

ESXi AsBuiltReport (I)

1.1 Hardware

The following section details the host hardware configuration for AUD507-ESXi-1.localdomain.

Host	AUD507-ESXi-1.localdomain
Connection State	Connected
ID	HostSystem-ha-host
Manufacturer	VMware, Inc.
Model	VMware7,1
Serial Number	
Asset Tag	Unknown
Processor Type	Intel(R) Core(TM) i7-6820HQ CPU @ 2.70GHz
HyperThreading	Disabled
Number of CPU Sockets	2
Number of CPU Cores	2
Number of CPU Threads	2
CPU Total / Used	5.42 GHz / 0.04 GHz
Memory Total / Used	4.00 GB / 1.36 GB

Here is a small part of a report generated by AsBuiltReport against an ESXi host. This section has some basic information about the hardware installed in the ESXi host.

August 10, 2021

ESXi AsBuiltReport (2)

1.4.3 Standard Virtual Switches

The following section details the standard virtual switch configuration for esxi.

Virtual Switch	MTU	Number of Ports	Number of Ports Available
vsSAN	1500	1536	1524
vsVM	1500	1536	1524
vSwitch0	1500	1536	1524

1.4.3.1 Virtual Switch Security

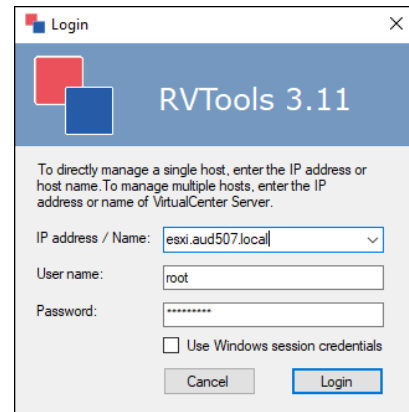
Virtual Switch	Promiscuous Mode	MAC Address Changes	Forged Transmits
vsSAN	Reject	Reject	Reject
vsVM	Reject	Reject	Reject
vSwitch0	Reject	Accept	Accept

In this report section, AsBuiltReport is detailing settings for the virtual switches on the ESXi host.

August 10, 2021

Robware RVTools (I)

- Windows GUI to gather inventory information from VMware infrastructure
- Multiple tabs of information about the host, cluster, or datacenter being reviewed
- Can export as CSV or Excel document for later review



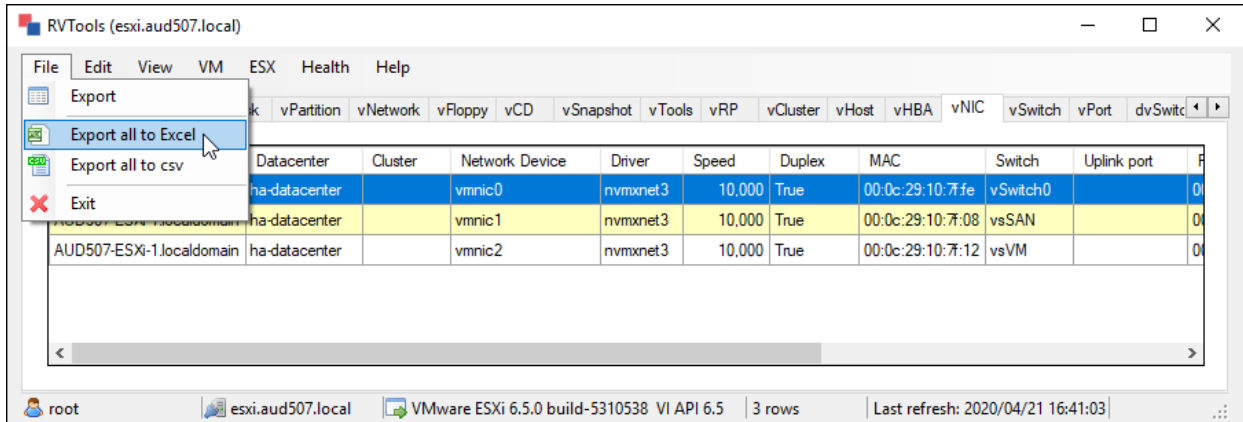
RVTools from Robware is a free Windows program that can gather and save inventory information about a VMware infrastructure. This tool can export the gathered information into CSV or Excel formats for later review by the auditor.

Information is available through RVTools about the server hardware, virtual machines, CPU and memory utilization, virtual networking, and distributed virtual networks, to name just a few categories.

RVTools is installed on your Windows 10 VM if you would like to experiment with it.

August 10, 2021

Robware RVTools (2)



This screenshot shows RVTools' tab for virtual NIC information and the export to CSV and Excel functions. ALL the data collected by RVTools from a host or vCenter server will be saved to the output file.

August 10, 2021

Data Available from VMWare Infrastructure

- Software versions/patches
- Lockdown mode
- User configuration
- Host configuration
- VM Settings

In this section, we will cover some of the more common measurement categories for VMWare infrastructure. We start with a discussion of VMWare patching.

August 10, 2021

VMWare - Software Versions

- ESXi software version is expressed as a "build number"
- Builds are rollups of all previous patches
- Build numbers are specific to the ESXi version
 - 6.5, 6.7 and 7.0 are currently supported

```
PS C:\> Get-VMHost | Select-Object
Name,ConnectionState,PowerState,Version, Build | FT *
```

Name	ConnectionState	PowerState	Version	Build
10.50.7.31	Connected	PoweredOn	6.5.0	16576891
10.50.7.32	Connected	PoweredOn	6.7.0	10302608

ESXi is really a lightweight Linux distribution with a hypervisor running on top of it. But the underlying OS and the hypervisor will need to be patched to keep them secure.

As of this writing, there are three supported versions of VMWare ESXi: 6.5, 6.7 and 7.0. You can determine if you are at the most recent version by examining the build number for the installation. You can check the build number against the official list on the VMWare website (next slide).

August 10, 2021

VMWare Build Numbers

- VMWare maintains a knowledge base article with all builds listed
- Ensure you are running most current (approved) build for your environment

Version	Release Name	Release Date	Build Number	Installer Build Number
ESXi 7.0 Update 1c	ESXi 7.0 Update 1c	12/17/2020	17325551	NA
ESXi 7.0 Update 1c (security Only)	ESXi 7.0 Update 1c (security Only)	12/17/2020	17325020	N/A
ESXi 7.0 Update 1b	ESXi 7.0 Update 1b	11/19/2020	17168206	N/A
ESXi 7.0 Update 1a	ESXi 7.0 Update 1a	11/04/2020	17119627	N/A
ESXi 7.0 Update 1	ESXi 7.0 Update 1	10/06/2020	16850804	N/A

The knowledge base article at <https://kb.vmware.com/s/article/2143832> lists all of the build numbers for all versions of ESXi. Part of your enterprise standards should include approved version and build numbers for your VMWare software.

In an exercise, you will download and process JSON data from a REST web service to build a PowerShell object which contains all the ESXi version information. You can compare your build number against the object to see if it is current.

August 10, 2021

VMWare – Patching

- Patches usually applied by Update Manager service on vCenter server
- Patches are installed as VIBs (vSphere installation bundle)
- List patches on a host with Get-ESXcli cmdlet
 - Emulates running local CLI commands on the host

```
PS C:\> (Get-ESXcli -Server esxi1).software.vib.list() |
Select -First 1

AcceptanceLevel : VMwareCertified
CreationDate     : 2016-10-27
ID              : VMW_bootbank_ata-libata-92_3.00.9.2-16vmw.650..
InstallDate     : 2019-11-08
Version         : 3.00.9.2-16vmw.650.0.0.4564106
```

Patching on ESXi servers is usually managed by the vCenter server, using compliance profiles. Patches are installed using VIB (vSphere installation bundle) packages which function as roll-ups of all previous patches.

Lists of patches are available using the Get-ESXcli cmdlet to query the software subsystem.

August 10, 2021

VMWare – ESXi User Information

- Local user accounts should normally be restricted in a vCenter environment
- Query local accounts with the Get-VMHostAccount cmdlet

```
PS C:\> Get-VMHostAccount -Server esxi1 | FL *
```

ServerId	: /VIServer=root@esxi1:443/
Server	: esxi1
Description	: Administrator
Domain	:
ShellAccessEnabled	: True
Name	: root

User information for an ESXi host can be queried directly, using the Get-VMHostAccount cmdlet. We would normally expect there to be very few local ESXi users in a vCenter-controlled environment.

We'll talk more about local accounts when we discuss lockdown modes later in the section.

August 10, 2021

VMWare – Host Configuration

- Get-VMHost ExtensionData object has lots of good configuration information
- ExtensionData.Config contains many configuration items

```
PS C:\> (Get-VMHost -name esx11 ).ExtensionData.Config

Host                : HostSystem-ha-host
Product             : VMware.Vim.AboutInfo
DeploymentInfo      : VMware.Vim.HostDeploymentInfo
Network             : VMware.Vim.HostNetworkInfo
Vmotion             : VMware.Vim.HostVMotionInfo
Service             : VMware.Vim.HostServiceInfo
Firewall            : VMware.Vim.HostFirewallInfo
...
```

You'll do most of your settings gathering using the Get-VMHost cmdlet. It has a property call "ExtensionData" which contains objects representing a lot of useful information about the host.

Pay particular attention to the "Config" object stored under extension data. If you need to query a configuration setting, you can probably to get it using this object.

August 10, 2021

VMWare Service Status

- PowerCLI provides cmdlets for querying services generically
 - GET-VMHostService
- NTP and Syslog have their own cmdlets
 - Get-VMHostNtpServer and Get-VMHostSysLogServer

```
PS C:\> Get-VMHost -Name esx1 | Get-VMHostNtpServer
pool.ntp.org
```

```
PS C:\> Get-VMHost -Name esx1 | Get-VMHostService |
  Where-Object {$_.key-eq "ntpd"}
```

Key	Label	Policy	Running	Required
---	-----	-----	-----	-----
ntpd	NTP Daemon	off	False	False

To check the service settings on a ESXi host, you can use the Get-VMHostService cmdlet. It takes in a VMHost object as input and returns information about the installed services on the host and their status.

In the screenshots, we are checking the NTP server setting for the host using an NTP-specific cmdlet (there is a similar one for Syslog settings). We also check the status of the service. Notice that even though an NTP server is configured, it really doesn't matter, because the service is disabled and not running. Be sure to check ALL the relevant settings during your compliance checks!

VMWare – Lockdown Mode

- Requires management tasks to happen through vCenter
- Two levels of lockdown
 - Normal: Administrative users on the exception list can still access the direct console user interface (DCUI)
 - Strict: No administration, except through vCenter (reinstall if vCenter access is lost). DCUI is disabled.

```
PS C:\> (Get-VMHost -name esx1 ).ExtensionData.Config.LockdownMode  
lockdownDisabled
```

Lockdown mode can enhance the security of an ESXi host by making it less likely that an attacker can gain control of the system through SSH or the physical console. Lockdown mode requires vCenter for managing the host.

There are two lockdown modes available:

- Normal: the DCUI is still enabled, but only administrative users whose account is in the exception list are allowed to access the DCUI.
- Strict: the DCUI is completely disabled. If SSH is available, it may still be accessed by users on the exception list. In strict mode, it's possible that if the vCenter settings are lost on the host, it will need to be reinstalled to regain access.

August 10, 2021

VMWare – ESX CLI

- Every command line interface command is exposed as a function

```
[root@ESXi-1:~] esxcli system account list
User ID  Description
-----  -
root     Administrator
dcui     DCUI User
vpxuser  VMware VirtualCenter administration account

PS C:\> (Get-ESXCLI -Server 10.50.7.32).system.account.list()
Description                               UserID
-----
Administrator                             root
DCUI User                                 dcui
VMware VirtualCenter administration account vpxuser
```

ESXi has a rich CLI available to administrators. All of the CLI commands are exposed in PowerCLI using a cmdlet called "Get-ESXCLI."

Each CLI command is exposed as a method call for the cmdlet.

In the screenshot, you see the native CLI and PowerCLI ways of listing the local users on the host.

Xen Specific

- Lock down access
 - Ensure SSH does not permit direct root logons
 - Check for information disclosures (default web pages)
 - Prevent physical access without a password
 - If local authentication is used, ensure there are speed bumps
 - Disable local root console
- Check for extra services
 - Can the administrator account for everything that is running
- Restrict connections to XAPI, requiring TLS
- Prevent promiscuous mode for VMs



In addition to all of the common configuration and security questions discussed previously, some other items are specific to Xen. The first is that we must ensure that the system is properly secured both remotely and locally. XenServer will have the SSH service running by default. Further, the SSH configuration will permit someone with the root password or key to log on directly. This makes it difficult to tie administrative actions to a specific administrator. To address this, the `/etc/ssh/sshd_config` file should have a line that states the following:

PermitRootLogin No.

There is also the possibility of information disclosure. The default web configuration will display the API information for accessing the XenServer API on the default web page. This page should be modified or otherwise made inaccessible to unauthorized users.

Local access also needs to be checked. If someone has physical access, it is possible to reboot the system into an administrative single-user mode without requiring a password. To modify this, the `/etc/inittab` file should have the existing configuration modified to include `~:S:wait:/sbin/sulogin`. Similarly, the root console is automatically available. This should be modified to require authentication as well.

It is important to verify that no extra services are running on the system, especially remotely accessible services. To discover which services are configured to run at startup, the administrator can use the command **chkconfig --list | grep 3:on**. One of the remotely accessible services is the Xen API. By default, this runs on both port 80 and port 443. Access to the port 80 service must be blocked to prevent unencrypted API credentials from being used.

Finally, virtual machines should not be permitted to run in promiscuous mode. If they can, they will be able to see all of the traffic, both in and out, that passes over the physical interface to which they are connected. This could be a serious confidentiality issue and can be controlled by the administrator with the command **xe pif-param-set uuid= other-config:promiscuous="off"**.

Microsoft Hyper-V Specific

- Should be installed as a Server Core
 - There should be no other roles installed
- Centralized VM storage
 - Default puts VM images onto local drives, not SAN
 - Consider using BitLocker for VM storage
- Check who is in Hyper-V Administrators group
- Hyper-V servers should be in their own OU



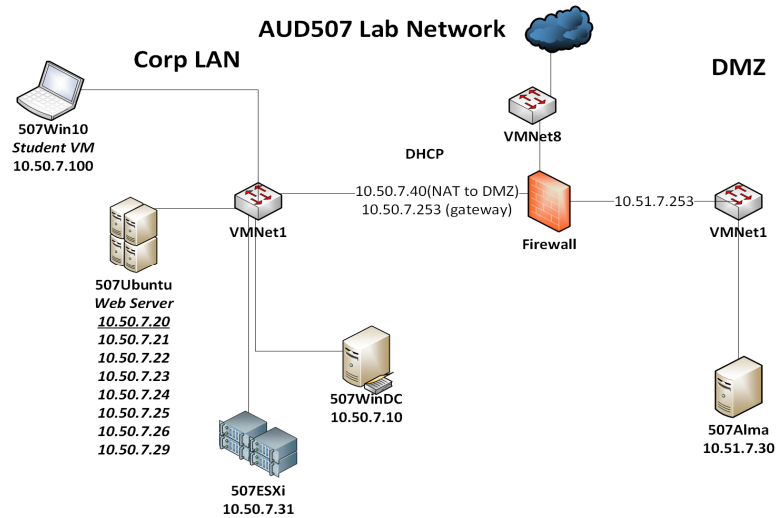
In addition to all of the general questions previously mentioned, there are a few Hyper-V specific items we would like to examine. The first is that if we are using Hyper-V for enterprise virtualization, it is best that Hyper-V is installed as a Type-I hypervisor as a Server Core install and with no other roles installed. No other software or applications should be installed either.

It is also good to verify that the files related to the virtual machines have been stored on a SAN or other shared storage medium rather than on the local drive of the Hyper-V server. The default is for files to be stored locally, which will prevent failover from working properly. While I am not a strong proponent of encrypting drives that live in data centers since BitLocker is built into the operating system, you might consider enabling BitLocker for the Hyper-V drives if this is important to your organization.

Hyper-V, being a Microsoft product, integrates tightly into the domain. Every Hyper-V server will have a local Hyper-V Administrators group. Like the other virtualization technology, we should verify that only a limited number of trained individuals are members of this group.

An additional point, which is sometimes overlooked, is where the Hyper-V servers are located within the Active Directory organizational structure. We discuss group policy in much greater detail in the Windows section; for now, we will simply say that the best practice is to put your Hyper-V servers within their own, separate OU. This will prevent them from inheriting other domain-level and server-level group policies, so the policies that are applied should be checked for completeness. It also prevents a seemingly innocuous group policy change to some other policy from having an unintended impact on the Hyper-V servers.

Exercise 4.1 - Auditing Hypervisors



This page intentionally left blank.

August 10, 2021

Course Roadmap

- Enterprise Audit Fundamentals; Discovery and Scanning Tools
- PowerShell, Windows System, and Domain Auditing
- Advanced UNIX Auditing and Monitoring
- **Auditing Private and Public Clouds, Containers, and Networks**
- Auditing Web Applications
- Audit Wars!

Section Four

1. Private Clouds
- 2. The Public Cloud**
 - Shared Responsibility
 - Security of the Cloud
 - Security in the Cloud
3. Containers
4. Networks and Firewalls
5. WIFI and VPNs
6. Public Services

This page intentionally left blank.

August 10, 2021

Understand What You Have

- **IaaS**
 - Infrastructure
- **PaaS**
 - Platform
- **SaaS**
 - Software
- **SECaaS**
 - Security
- **DaaS**
 - Data
- **TEaaS**
 - Test Environment
- **STaaS**
 - Storage
- **APIaaS**
 - API

Although NIST defines only three major categories for cloud services, many, many flavors are available. These days you can take any word(s) you like and stick “as a service” to the end, and you have a cloud service.

More than anything, these are now marketing buzzwords. When you think about it, the idea of having your stuff—be that data, systems, or infrastructure—residing in someone else’s data center is not a new one. The real innovation is the advent of high-reliability virtualization with the ability to use a simple interface to perform elastic allocations.

Want some proof? Consider this. The ability to create virtual systems existed in the 1980s on IBM mainframes. The administrator could use command-line tools to create virtual machines and allocate resources to them. The difference today is that Amazon’s interface is much easier to use than the IBM monitoring commands.

August 10, 2021

Cloud Security: Shared Responsibility Models

- Most providers separate their responsibility from the customer's
- Cloud provider: Security *of* the cloud
- Cloud customer: Security *in* the cloud

SERVICE TYPE	PHYSICAL SECURITY	OS PATCHING	APPLICATION PATCHING	DATA SECURITY
Software (SaaS)	Provider	Provider	Provider	Customer
Platform (PaaS)	Provider	Provider	Customer	Customer
Infrastructure (IaaS)	Provider	Customer	Customer	Customer

Many of the major cloud providers use the concept of a shared responsibility model to describe the security efforts required by the provider and the customer. A commonly used saying is that the provider is responsible for the security *OF* the cloud, while the customer is responsible for operating securely *IN* the cloud. Amazon has articulated this view very well in their documentation on the subject, available at:

<https://u.aud507.com/1-8>

Certain things will always be the responsibility of the provider, like physical security and power protection for the data centers. Others will always belong to the customer, like managing keys used for data encryption and adding and removing authorized users from systems.

The chart on this slide gives examples of how these responsibilities may vary among the common cloud service models.

Security of the Cloud: CSA CCM (I)

- Cloud Security Alliance
 - Cloud Controls Matrix
- Audit and assurance
- Application and interface security
- Business continuity management, operational resilience
- Change control, configuration management
- Cryptography, encryption/key management
- Data center security
- Data security, privacy lifecycle



The Cloud Security Alliance (CSA) is an industry organization which represents many of the largest cloud providers. The CSA's Cloud Controls Matrix (CCM) document details the types of controls which the CSA members have agreed should be implemented by all cloud providers. It can provide organizations with a framework for evaluating the security efforts of a provider, and a neutral basis for comparing providers. Many large providers produce self-assessment questionnaires detailing their compliance with the CCM as a part of the CSA's STAR certification program.

This page and the next contain a list of the control domains in the CCM.

Audit and assurance describes the types of audit programs to be implemented

Application and interface security involves the secure development of client-facing APIs and other interfaces

Business continuity/operational resilience covers the provider's continuity and disaster controls

Change control/configuration management describes the maintenance of applications and systems

Cryptography, encryption and key management ensures the use of proper encryption and describes the controls for management of encryption keys

Data center security describes the physical security controls for the data center

Data security and privacy lifecycle covers the creation and destruction of data and the management and stewardship of customer data

Security of the Cloud: CSA CCM (2)

- Governance, risk, compliance
- Human resources
- Identity and access management
- Interoperability and portability
- Infrastructure and virtualization security
- Logging, monitoring
- Security incident management, e-discovery, and cloud forensics



(Continued from the previous page)

Governance, risk and compliance covers how the provider oversees the management of risk

Human resources includes the administrative controls over hiring, firing, and managing employees

Identity and access management discusses authorization, access, and accounting for internal applications used by the provider

Interoperability and portability covers the ability of cloud service customers to easily create, retrieve and update their data which is in the custody of the provider and the way in which APIs and other software components should securely communicate

Infrastructure and virtualization security controls the secure operations of the cloud infrastructure including the use of secure management protocols, system classification and segregation of provider and customer data

Logging and monitoring covers the creation, management and monitoring of logs, including time synchronization for systems

Security incident management, e-discovery, and cloud forensics is concerned with the provider maintaining plans for incident handling, evidentiary chain of custody, and digital forensics

Security of the Cloud: CSA CCM (3)

- Supply chain management, transparency, accountability
- Threat and vulnerability management
- Universal endpoint management



(Continued from the previous page)

Supply chain management requires the provider to manage the risk introduced by using products and services provided by third parties

Threat and vulnerability management includes endpoint malware protections and vulnerability identification and mitigation efforts

Universal endpoint management involves the management of endpoints, mobile devices, including bring-your-own device (BYOD) plans, encryption, and user training

August 10, 2021

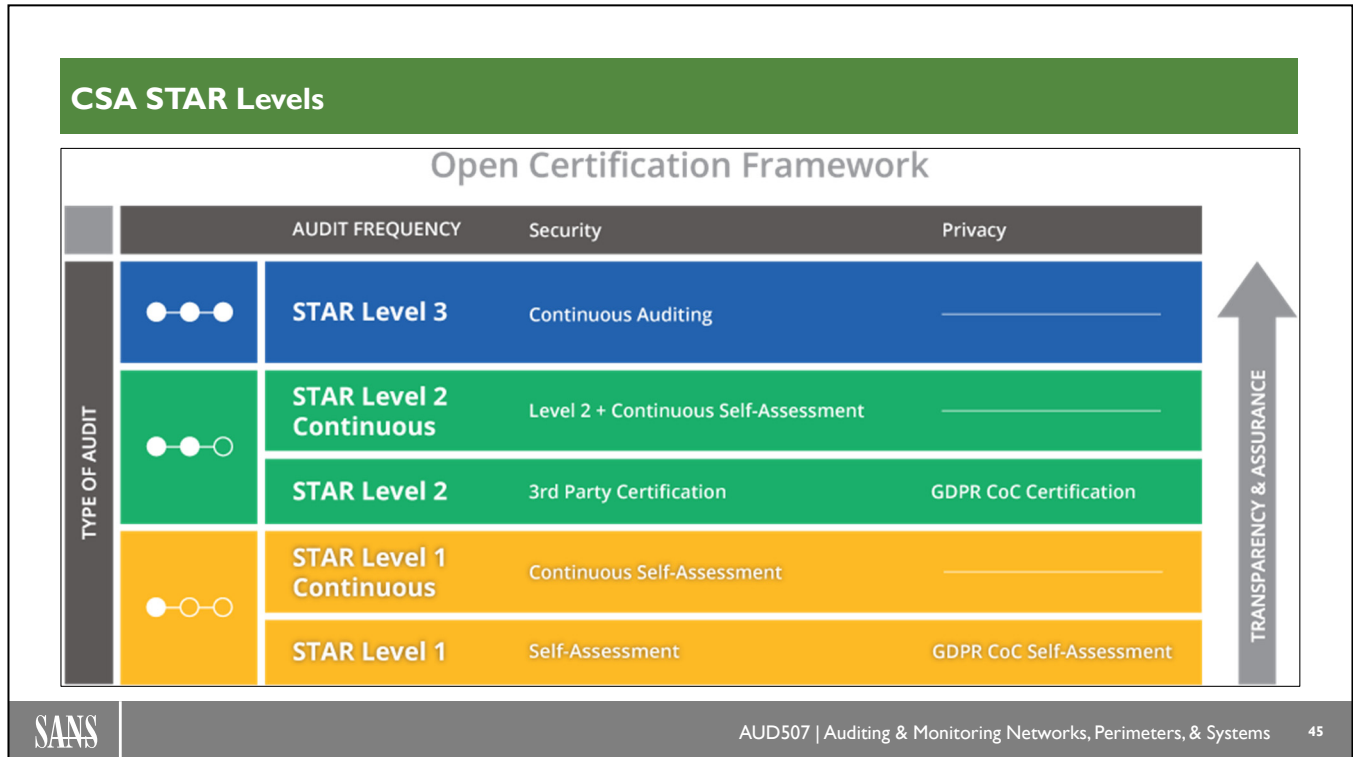
CSA Security Trust Assurance and Risk (STAR) Program

- Open framework for security certification for cloud service providers
- STAR Registry maintains list of controls provided by participating cloud service providers
- Multiples levels:
 - Level 1: Self-assessment (Security or GDPR)
 - Level 2: 3rd party certification
 - Level 3: Continuous auditing



The Cloud Security Alliance maintains the STAR certification and STAR registry to allow customers of cloud service providers (CSPs) to compare the controls implemented by various providers in order to choose the provider that best meets their needs. The STAR certification program consists of several levels, ranging from self-assessment to certified systems which are then continuously monitored.

August 10, 2021



STAR Level 1 is based on a self-assessment questionnaire done by the CSP.

STAR Level 2 is based on an assessment, attestation engagement, or certification performed by an independent third party. These reviews will be based on the Cloud Controls Matrix plus the standards of the other certification.

STAR Level 3 is not fully implemented yet but will apparently combine Level 2 certification with continuous monitoring and validation.

More on STAR Levels 2 and 3 on the next page...

August 10, 2021

CSA STAR Level 2/3 Certifications

- CSA STAR Attestation - SOC2 + STAR
- CSA STAR Certification - ISO27001 + STAR
- C-STAR – for greater China market
- Level 2 Continuous – Third-party certification + continuous self-assessment
- Level 3 is not implemented yet
 - Will include monitoring against Service Level Objectives (SLO) and Service Qualitative Objectives (SQO)

Here are some commonly-used STAR Level 2 certifications:

- In a CSA STAR Attestation, the CSP undergoes an audit for SOC2 certification against the Trust Service Principles from the AICPA AT-101 standard, with the addition of the Cloud Controls Matrix (CCM)
- In a STAR Certification, the CCM is combined with an ISO 27001 certification
- C-STAR certification is intended for the greater China market
- STAR Level 2 Continuous combines a third-party certification with a continuous self-assessment scheme

There are other STAR Level 2 options used for national and international jurisdictions.

STAR Level 3 is not yet implemented, but it will include continuous monitoring against an organization's Service Level Objectives (SLO) and Service Quality Objectives (SQO).

CSA STAR Registry

[Home](#) > [STAR](#) > Registry

Find a provider with the right level of security and data privacy for your organization.

[Submit to the Registry →](#)

[Ask a provider to submit to the registry →](#)

Filter Your Results ▲

[Reset all filters](#)

View Only

☐ CSA Trusted Cloud Providers



Amazon

Amazon Web Services provides a highly reliable, scalable, low-cost infrastructure platform in the cloud that powers hundreds of thousands of businesses in 190 c...

Listed Since: 08/04/2020



Submissions:

[CAIQ ⓘ](#)



Submissions:

[Certification ⓘ](#)




[View Listing](#)

The STAR Registry maintains a list of controls provided by participating cloud service providers in a searchable website. In this screenshot, we have searched for the AWS self-assessment questionnaire. When we click on the “Self-Assessment” button, we are taken to the download page for the document.

August 10, 2021

CSA STAR Registry Self-Assessment PDF Download




Level 1: Self-Assessment

At level one organizations can submit one or both of the security and privacy self-assessments. These are based off of the Cloud Controls Matrix and the CSA Code of Conduct for GDPR Compliance.

Security Self-Assessment

[Download Self-assessment](#)





Level 2: Third-Party Audit

Organizations looking for a third-party audit can choose from one or more of the security and privacy audits and certifications.

STAR Certification: ISO/IEC 27001:2013

A technology-neutral certification leveraging the requirements of the ISO/IEC 27001:2013 management system standard together with the CSA Cloud Controls Matrix (CCM).

[Download Certification Supporting Asset](#)

The information below is provided as a companion to the CSA STAR Certificate.

Client Name

Client URL

Client Description (200 words or less)

Amazon Web Services is a subsidiary of Amazon providing on-demand cloud computing platforms and APIs to individuals, companies, and governments, on a managed pay-as-you-go basis. In 2006, Amazon Web Services (AWS) began offering IT infrastructure services to businesses in the form of web services – most commonly known as cloud computing.

The company has its headquarters in Seattle, Washington, USA.

Scope

The scope of this CSA STAR CCM 3.0.1 certification is bounded by specified services of Amazon Web Services, Inc. and specified facilities. The Cloud Security Framework is centrally managed out of Amazon Web Services, Inc. headquarters in Seattle, Washington, United States of America.

The in-scope applications, systems, people, and processes are globally implemented and operated by teams out of an explicit set of facilities that comprise Amazon Web Services, Inc. and are specifically defined in the scope and bounds.

The Amazon Web Services, Inc. (AWS) scope includes the services as mentioned on <https://aws.amazon.com/compliance/iso-certified/>; the locations and AWS Service and Supporting Resources are noted in the following section of this certificate.

The Cloud Security Management System mentioned in the above scope is restricted as defined in "CSMS Manual" version 2020.04, dated October 1, 2020.

The Cloud Security Management System is centrally managed out of AWS headquarters in Seattle, Washington, United States of America.

CSA Version Used

☐ 1.4
 ☒ 3.0.1
 ☐ 3.0

Number of Sites

Certificate Country

Certificate Expiry Date

Term of Certificate

Certification Body

Certificate Number

The STAR Self-Assessment questionnaire is available for download as a PDF. In this case, we have downloaded the 1-page certificate for AWS's STAR certification for 106 cloud products.

Security in the Cloud (I)

- Authentication
 - Multifactor for administrators (if not everyone)
 - Key-based authentication
 - No shared (root) accounts
 - Cloud access security brokers
 - Login monitoring
- Logging and monitoring
 - Alert on security configuration changes
 - Monitor access to sensitive resources

Cloud customers are also responsible for maintaining security during their use of cloud services. The business should require multifactor authentication at least for their administrative users. Given the vulnerability of cloud services to internet-based brute force authentication attacks, the second factor can provide an additional layer of protection. Key-based authentication, in which the user authenticates using a public/private key pair, is better than password-based authentication, which can be more easily brute forced. Businesses should not use shared “root” or administrative accounts, because it is important to attribute all administrative actions to specific users.

Many businesses use cloud access security brokers (CASBs) as an intermediary between the users and their cloud applications. The CASB allows the business to maintain one set of authentication and access control data for all their applications, making it easier to add and remove users and to change their level of access to applications.

Most cloud applications provide audit logging for authentication and sometimes access control. Logging may need to be explicitly enabled by the administrators. Many providers offer mechanisms for tracking access to sensitive data, and some even offer to alert when security changes are made.

Security in the Cloud (2)

- Asset inventory
- Secure data storage
 - Permissions
 - Backups
 - Encryption
- OS/application patches
- Network security
 - Limit access to administrative interfaces
 - Use VPN technology

Maintaining an inventory of server and application assets is more difficult in the cloud. The business should track all virtual machines and applications in use. A good inventory makes auditing much easier and can often lead to cost savings as the organization can use the data to manage the number of running instances to reduce their service cost.

Security of the data stored in the cloud service is normally the customer's responsibility also. Common tasks include assigning appropriate permissions, performing and testing backups, and maintaining encryption of sensitive data.

In many cloud models, the customer is responsible for at least some of the patching duties on the systems. With infrastructure as a service (IaaS), the customer probably does all the patching. With software as a service, the customer might not do any of the patching tasks.

Properly securing the administrative interfaces used on the cloud provider may require using access control lists to limit the IP ranges from which administrators can connect or using virtual private networks to add a layer of authentication and encryption before administrative interfaces can be reached.

Course Roadmap

- Enterprise Audit Fundamentals; Discovery and Scanning Tools
- PowerShell, Windows System, and Domain Auditing
- Advanced UNIX Auditing and Monitoring
- **Auditing Private and Public Clouds, Containers, and Networks**
- Auditing Web Applications
- Audit Wars!

Section Four

1. Private Clouds
2. The Public Cloud
3. **Containers**
 - Intro
 - Docker
 - Kubernetes
 - Exercise 4.2 - Auditing Docker Security
4. Networks and Firewalls
5. WIFI and VPNs
6. Public Services

This page intentionally left blank.

August 10, 2021

Containers: Application Virtualization with Docker, Etc.

- Application + dependencies in one package
- Deploy on any operating system
- Deploy as many as you like
- Minimal resource use
- *Do I need a full virtual machine for my PHP app?*



A step beyond machine virtualization is containerization. For many enterprise needs, a full virtual machine is overkill. A container is a virtual environment which contains only what's needed to run a particular application or service. Container images tend to be much smaller than full virtual machines, which allows for much greater density on existing hardware.

If a development shop needs to host a new PHP application, for instance, it can deploy a container image with a minimal Debian Linux install which includes Apache and the PHP modules installed. The container environment, such as Docker, will manage starting and stopping the image and expose whatever network ports are served by the application. If the demand for that application grows, the container system can be told to spin up more instances to meet the requirements.

Very often, organizations will use orchestration systems like Kubernetes to manage container images and to coordinate the communication between them. Matt Butcher has written a great "Illustrated Children's Guide to Kubernetes," one of the best descriptions of containers and orchestration we have seen. It is available at:

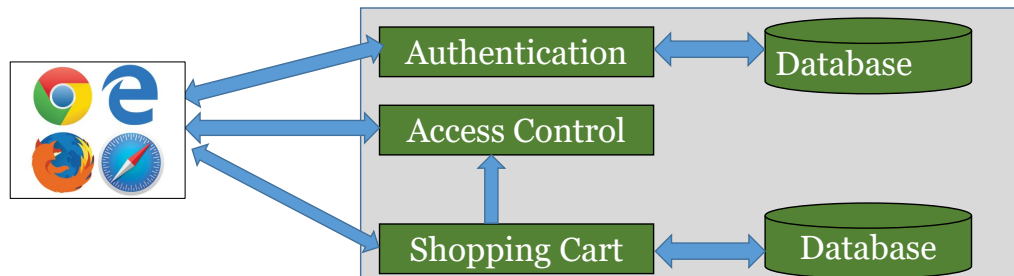
<https://u.aud507.com/1-16> (video)

and

<https://u.aud507.com/1-17>

Containers: Microservices

- Very small footprint – even more applications on my hardware
- Easy to add/move/remove to meet demand
- Allows for microservices
 - Decompose large applications into component services that interact with each other and the user interface:



One common use for containers is to implement microservices. These are small software components which provide part of the functionality of a larger application. Microservices allow developers to build reusable modules which implement a single function. These services can be used together to deliver larger enterprise applications which scale easily by running more instances of busy services.

In this example, the user interface runs as JavaScript in the user's web browser. Microservices running as containers provide authentication, access control, and shopping cart functions. Some of the services communicate with each other or with backend databases. All of these services together comprise the application. We'll explore how to audit this type of application in the web application section of this course.

August 10, 2021

Containers: Audit Concerns

- Host operating system security
- Docker daemon configuration
- Configuration file security
- Image file security
- Configuration of runtime environment

- We'll use Docker as our model
 - CIS Docker Benchmark
 - Many settings must be applied at the Docker command line or in the daemon.json file (which must be created by the admins)

When auditing containers, there are a number of considerations involving the host operating system and the security of the container environment. Because container images are stored on the host's filesystem, it's important to restrict access to images. The Docker daemon itself must be configured properly to ensure that images are handled securely. The confidentiality and integrity of image and configuration files need to be protected, as well.

There are some good resources available for auditors wanting to learn about auditing containers. ISACA has a container auditing guide, free for ISACA members (\$50 for everyone else), available at:

<https://u.aud507.com/1-18>

There's a good paper in the SANS reading room at:

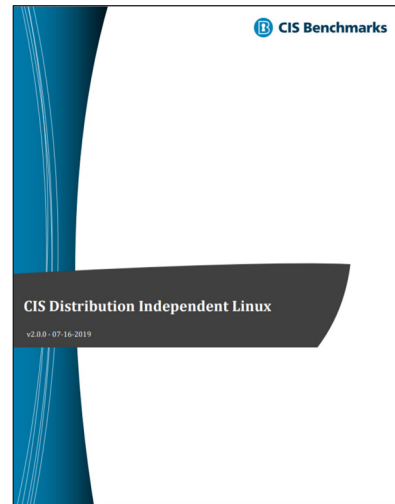
<https://u.aud507.com/1-19>

The Cloud Security Alliance started a container working group in mid-2018. They haven't produced much work as of this writing, but it may be worth a look at some point:

<https://u.aud507.com/1-20>

Host Security

- Harden the host OS
 - Usually Linux
 - Can use CIS Level 1 or 2 server benchmarks
- Keep Docker service patched
 - Vulnerability in Docker daemon could expose all hosted applications and services to attack



The CIS Docker benchmark recommends that Docker be run on a hardened host operating system. Since the underlying OS is usually Linux, there are a number of appropriate benchmarks which may be suitable to implement for hardening the server. For most organizations, this will involve implementing all (or at least most) of the Level One benchmark settings.

The docker service must be kept up-to-date as well. Docker's daemons usually run with root-level privileges, so a vulnerability or flaw in the daemon could have disastrous consequences for the entire host.

August 10, 2021

Linux-Specific Host Security (I)

- Place containers on a dedicated disk partition
 - Limit system problems caused by full filesystem
- Limit number of users with ability to control Docker daemon
 - Docker normally runs as root
 - Members of the “docker” group

```
# mountpoint -- "$(docker info -f '{{ .DockerRootDir }}')"  
  
/var/lib/docker is not a mountpoint
```



Docker containers should be placed on their own dedicated disk partition to avoid any denial-of-service to the host which would be caused by the container storage filesystem filling up. This is common practice on Linux/Unix systems to segregate system components.

On most hosts, members of the “docker” group have the ability to control the action of the Docker daemon. Because Docker runs with elevated privileges, a malicious user in that group could potentially do damage to not only the containers, but the host itself.

August 10, 2021

Linux-Specific Host Security (2)

- Use auditd rules to track access to important files and directories
 - Linux kernel-level audit daemon
 - Covered in Unix section
- Audit/log EVERYTHING:
 - /var/lib/docker – information about containers
 - /etc/docker – config files, TLS keys
 - /etc/default/docker – default docker settings
 - /etc/sysconfig/docker (Red Hat) – config settings
 - /etc/docker/daemon.json – config settings
 - /usr/bin/containerd – binary to manage container lifecycle

The Linux kernel has a built-in audit feature known as auditd, the audit daemon. We worked with the audit daemon (auditd) and its associated control, search, and reporting tools in the Unix section of this course. Auditd has the ability to monitor all types of access to files and directories.

CIS recommends using auditd to log access to a number of configuration directories, configuration files, and binaries to detect things like unauthorized modification. This logging is crucial for maintaining the integrity of the Docker system.

August 10, 2021

Docker Daemon Configuration (I)

- Restrict inter-container traffic on the default bridge – information leakage/sniffing
- Set log level to info (debug only for troubleshooting)
- Allow Docker to modify iptables rules
 - Consistent application
 - Admins can't forget
- Use only secure registries
 - TLS
 - Trusted certificate authorities
 - “Insecure” registries can be cached to allow indefinite use

By default, Docker allows all containers to share access to the default bridge. This inter-container networking is often not needed, however, and leaving it enabled can expose the system to risk. Containers may be able to sniff the traffic for other containers, violating the principle of least privilege and harming confidentiality.

CIS recommends that Docker be configured to log fairly verbosely, using the “info” level - only slightly less verbose than “debug.” I further recommend that these logs be sent to a SIEM for correlation and analysis in the event there is a security incident involving the host.

Docker should be able to create and modify iptables (the Linux host firewall) rules for its containers. This allows Docker to ensure that only current rules are running on the host and to ensure availability for new containers without administrator intervention.

Also, the host should be configured to import container images only from “secure” container registries—those which use TLS with valid server certificates. This protects the image management traffic against interception and interference.

Docker Daemon Configuration (2)

- Don't use aufs filesystem driver
 - Old and crash-prone
 - Could indicate old Linux kernel
 - Expect OverlayFS on newer systems
- If Docker daemon is bound to TCP (not ideal), also use TLS client authentication
 - Docker usually runs as root
 - Should limit access to authorized hosts

```
# docker info --format 'Storage Driver: {{ .Driver }}'  
Storage Driver: overlay2
```



The aufs (Another Union File System) filesystem driver was developed for Linux in 2006 and is still included in some (semi-modern) systems like Ubuntu Linux 16.04 (released in April 2016). It has been mostly replaced by the OverlayFS in more modern Linuxes. Seeing this driver running on a host may indicate that the Linux distribution (or at least the Linux kernel) on the host is quite out-of-date.

Docker is usually bound to a local Unix socket but may be bound to a TCP port to allow for remote control. If Docker IS exposed via TCP, the administrators should ensure that the traffic is protected by TLS and that access is limited to only authorized IP addresses.

August 10, 2021

Docker Daemon Configuration (3)

- Use ulimit settings to constrain resource use
- Enable live restore to keep containers running when the Docker daemon is unavailable
 - Protects container availability

```
# docker info --format '{{ .LiveRestoreEnabled }}'  
false
```



The ulimit setting for a process instructs the kernel to restrain the use of resources by that process. Ulimit settings can control things like memory utilization, CPU time, the number of files which can be locked, etc. This setting can protect the system against out-of-control forking of processes, and other resource-exhaustion issues. By default, Docker places no restrictions on processes' resource allocation.

The live restore option allows Docker containers to keep running even when the Docker daemon is not running. This allows for container availability even during maintenance operations against the daemon binaries.

These settings must be passed on the command line or set in the daemon.json file created by the administrators.

August 10, 2021

Docker Daemon Configuration (4)

- Disable userland proxy service
 - Redundant if using “hairpin” NAT
- No experimental features in production
- Disallow new privileges for containers
 - No_new_priv bit in kernel
 - Forbids privilege escalation via SUID/SGID

```
# docker version --format '{{ .Server.Experimental }}'  
false
```



Docker ships with two different proxy services for containers. The userland proxy is used to expose the ports served by containers to the production network. This same service is provided by Docker’s hairpin NAT service, which is more commonly used. The userland proxy can and should be shut down if it is not used.

The `--experimental` flag passed on the Docker daemon’s command line enables the use of “experimental” features which may not have been fully tested for stability and security yet. These features should not be used in production.

The Linux kernel provides a bit that can be set for a process that prohibits it from escalating privilege via the Linux SUID and SGID bits, which allow a binary to run with the privileges of the binary file’s owner. This setting can be added at the daemon command line or in the `daemon.json` file.

August 10, 2021

Docker Configuration File Security (I)

- Maintain appropriate ownership and permissions on binaries and configuration files
- Examples: docker.service and docker.socket files:
 - Owned by root:root
 - Permissions of 644
 - Ensure non-root users cannot alter
- More on next two slides...

```
# systemctl show -p FragmentPath docker.service
FragmentPath=/lib/systemd/system/docker.service
```

```
# ls -l /lib/systemd/system/docker.service
-rw-r--r-- 1 root root 1683 Mar 10 20:24 /lib/systemd/system/docker.service
```



There are a number of files and directories which are required for proper operation of the Docker environment. The next two slides list several of the files and directories as well as the CIS recommendations for their ownership and permissions.

The screenshot on this slide shows how the auditor can check the permissions for one of the files.

August 10, 2021

Docker Configuration File Security (2)

Filesystem Object	User	Group	Permissions
docker.service	root	root	644
docker.socket	root	root	644
/etc/docker directory	root	root	755
/etc/docker/certs.d/*	root	root	444
docker.sock	root	docker	660
daemon.json	root	root	644

As you can see, most Docker files should be owned by root, with others being prohibited from writing. This makes sense: because the files are owned by root, if a malicious user could alter them, they may be able to escalate their privileges by leveraging the Docker daemon and binaries.

August 10, 2021

Docker Configuration File Security (3)

Filesystem Object	User	Group	Permissions
/etc/default/docker	root	root	644
/etc/sysconfig/docker	root	root	644
TLS CA certificate	root	root	444
TLS server certificate	root	root	444
TLS certificate key	root	root	400

More file ownership and permission recommendations are given above.

August 10, 2021

Docker Image File Security (I)

- Applications hosted by containers should be run as a non-root user
- Use trusted base (source) images for building new containers
- Scan images for security flaws and rebuild with updates
- Remove unneeded software packages from containers; use minimalist base images instead of full Linux distributions if possible

Docker containers have local users configured. Often the creator of the container image will run the program hosted by the container as root when it's not necessary. Least privilege dictates that the application should be run with non-root privileges when possible. Additionally, unneeded local users and extraneous software packages should be removed from the container.

As with all software development projects, containers should be built on a trusted foundation. Letting strangers on the internet build the base system you're using to host your business-critical applications might not be the best idea! Similarly, all images should be scanned for known security issues. In immutable environments, a new container with the latest software should be built and deployed as part of the next release cycle.

August 10, 2021

Docker Image File Security (2)

- Turn on Docker content trust - set \$DOCKER_CONTENT_TRUST environment variable to 1
- Add HEALTHCHECK settings on images - availability
- Avoid stand-alone update commands (like yum update) in Dockerfiles

```
# echo $DOCKER_CONTENT_TRUST
#
# docker inspect --format='{{ .Config.Healthcheck }}'
buggybank
<nil>
```



Content trust allows the use of digital signatures to verify images downloaded from or sent to a Docker registry. This protects the integrity of the images being used.

The HEALTHCHECK option on a container is used to check that the application is still operating correctly. Docker can terminate or restart non-functioning containers proactively.

Running a package update in a single line in a Dockerfile can cause the package updates to be cached in an overlay layer for the image filesystem, effectively trapping those updates and preventing them from being replaced later with current copies.

August 10, 2021

Docker Image File Security (3)

- Remove SUID and SGID permissions in containers
- Use COPY instead of ADD to copy files into a container
- Don't save secrets, like passwords or encryption keys, in the image
- Verify software packages installed in the image

```
# docker history feltsecure/owasp-bwapp:latest | grep ADD
/bin/sh -c #(nop) ADD file:4eacfc1aa3adff00b... 1.11kB
/bin/sh -c #(nop) ADD file:846d81e0679354c2e... 1.12kB
/bin/sh -c #(nop) ADD file:1b12f1071297cfbb0... 84B
/bin/sh -c #(nop) ADD file:dadf942589f45f87e... 86B
```



Removing SUID and SGID permissions from files which don't need them is simply applying least principle to the container.

The ADD command can retrieve files from remote URLs, which may be unsafe. COPY gets the files from the local machine which is building the image, which is considered a better practice. Because the command history for an image can be queried by users of the container, it is not safe to use passwords or keys in any commands run in the container.

When software is installed in a Dockerfile, GPG keys or cryptographic hashes should be used to verify the integrity of the software.

August 10, 2021

Docker Runtime Security

- Run containers with AppArmor or SELinux security options
- Don't run containers with elevated privileges
- Constrain the system resources used by containers
- Limit sharing of filesystem and network resources between the container and host

The principle of least privilege is king when it comes to running containers:

- Use AppArmor and SELinux profiles to protect applications and enforce access controls
- Run containers without elevated privileges whenever possible
- Tightly constrain the resources a container is allowed to use
- Limit the containers access to the host's resources, including networks and filesystems

August 10, 2021

Container Orchestration

- Standalone Docker hosts can't provide enterprise-class availability and reliability
- Orchestration tools manage the container lifecycle and allow for scaling up and down as needed to meet demand
- Common orchestration tools:
 - Kubernetes, Kubernetes engine (Google Cloud Platform), Amazon Elastic Kubernetes Service (EKS)
 - Docker Swarm
 - Azure Service Fabric
 - Helios (for Docker, by Spotify)



Orchestration tools help to provide the reliability, scalability and availability required to make use of containers in the enterprise. Most major cloud providers provide at least one (sometimes many) container orchestration tool for their clients. These orchestration tools allocate resources on physical hosts (usually as part of a cluster), manage network connections and proxying, and provide high-availability using clustering technologies.

Kubernetes is the basis for many of these systems, so we will use it as the example for how to secure a container orchestration tool.

August 10, 2021

Container Orchestration: Kubernetes (I)

- Kubernetes - Greek word for a ship's captain
 - Sometimes abbreviated “k8s”
- Grew out of Google's Borg project, a container-oriented cluster-management system
- Pods used to group containers to be run on the same physical host
 - Pods can be replicated using templates for easy scaling and rolling or blue/green deployments
- Services used to publish container services to known IP:port sockets
 - Services can be located using Kubernetes service discovery

Have you ever wondered how Google achieves the high availability and reliability it does for its cloud services, like Gmail and YouTube? They were pioneers in the field of site reliability engineering (in fact their engineers wrote a good book about that topic). Google developed a system called “Borg” to manage containers running on physical hardware in their datacenters. Many of the Borg features were included (and improved upon) in Kubernetes.

Kubernetes uses “pods” to group one or more containers together to be run on the same physical host. Kubernetes provides and manages the network connections for the pods. With a pod template, part of the replication controller, Kubernetes can replicate the pod as many times as needed to meet availability or performance demands. Templates are also useful for “rolling” or “blue/green” deployments used with immutable architectures. In immutable architecture, containers are not ever patched. Instead, they are replaced with new versions of the container image with the updated software included. The organization can then slowly deploy copies of the new container image, while removing the old versions from production. This has the advantage of allowing for quick roll-back in the event the new image is not correct in some way.

Kubernetes uses “services” to map the services offered by a container to a specific IP:port combination. Kubernetes service discovery allows other applications to locate the service provided by a container.

Container Orchestration: Kubernetes (2)

- Volumes are used to provide storage to containers
- Persistent storage volumes use local or network/cluster filesystems
- Local ephemeral storage has no performance SLA support
 - /var/log directory in the container is often mounted this way
- Namespaces provide isolation for related pods, replication controllers, and volumes
 - Gives a degree of privacy

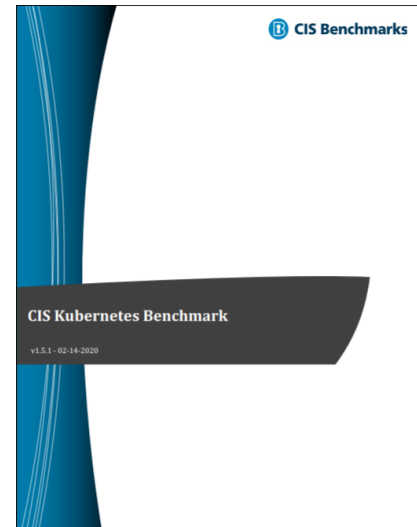
Volumes are used to allow containers to read and write information. They are mounted to a directory in the container's filesystem and can be either persistent or ephemeral. Ephemeral storage is used for transient data and, accordingly, is usually not as well protected.

Namespaces are used to group like objects together in Kubernetes. This provides some isolation from the other objects in the cluster.

August 10, 2021

Kubernetes Audit Considerations (I)

- Security of configuration files
- API server
- Controller manager
- Scheduler
- etcd security
- Authentication/authorization
- Logging



Of course, CIS has a benchmark for Kubernetes! We won't cover it in the same level of detail as we did for Docker, but here are the highlights:

- As with Docker, the security of the configuration files is paramount to protecting the security of the cluster and its nodes.
- The API server which exposes the management interface of Kubernetes should be secured like any API, with strong authentication and role-based access control.
- The control server monitors cluster state using the API server. Controller functions should be configured to use good authentication and reduce the attack surface of the controller itself.
- The scheduler is used to match nodes to cluster nodes to be run. The scheduler should be bound to the loopback address and not do performance profiling unless it is needed. These settings also help to reduce the attack surface.
- Etcd manages settings for the cluster and should be protected with proper access controls—usually TLS authentication for cluster nodes.
- Kubernetes currently has no way to revoke old user authentication certificates, so they should be avoided for now.
- The cluster should have an audit policy to require logging that is appropriate to the cluster's risk profile.

Kubernetes Audit Considerations (2)

- Worker node configuration
 - Configuration file security
 - Kubelet (agent which runs a node in a cluster) settings
- Security policies
 - Role-based access control for service accounts
 - Pod security
 - Network and CNI (container network interface)
 - Secrets management

Just like for master nodes, the worker nodes in a cluster should have proper security setting for their configuration files and binaries. Kubelet is the agent which does the work on a cluster node.

Access to administer a Kubernetes cluster should be limited using RBAC. Privilege escalation for containers should be forbidden wherever possible to further reduce the attack surface.

Network policies should be used to control network flow enforced by whatever Container Network Interface (CNI) is in use.

Finally, where possible, secrets should be maintained in secrets managers external to the Kubernetes environment, like HashiCorp's Vault product. When secrets are used locally, store them in data volumes rather than environment variables.

August 10, 2021

Beyond Containers: Serverless Architectures

- Microservices used in many modern apps
- Even a container may be overkill for a single function
- Functions can run in “serverless” model
- Provider hosts the code and runs it on-demand
- Examples:
 - AWS Lambda
 - Azure Functions
 - Google Cloud Functions
 - Knative (pronounced Kay-nay-tive)

If containers are the next wave of virtualization, then serverless architectures are the next iteration of microservices. With containers, we abstract at the scope of the application. With serverless architectures, we abstract at the function level. Do I really need a full container to run one programmed function? The answer may be no.

Services like Amazon Web Services (AWS) Lambda provide a way to host your code on someone else's server to be run on-demand. Developers can write code in a Lambda-compatible language like C# and host that code in an AWS data center. When the code is needed (for example, when a web browser makes a query to my web service), AWS spins up the code, runs it, returns a result and then shuts the code back down. The customer pays only for the actual computer resources used to host the queries. Serverless code can be scaled to meet demand, can access backend databases like AWS DynamoDB, and requires absolutely no server administration on the part of the customer.

Serverless Best Practices

- Maintain an inventory
 - Tags can help
 - Watch repositories for additions
- Limit functions to a single IAM role
- Integrate security checks into the deployment pipeline
- Access control: Limit access to authorized users, functions, services, VMs, subnets, etc.
- Audit/log all runs of the function

Your organization should maintain an inventory of all technology assets, including serverless functions. Most serverless providers give a way of attaching “tags” to objects, which can be very helpful in locating those assets in an inventory.

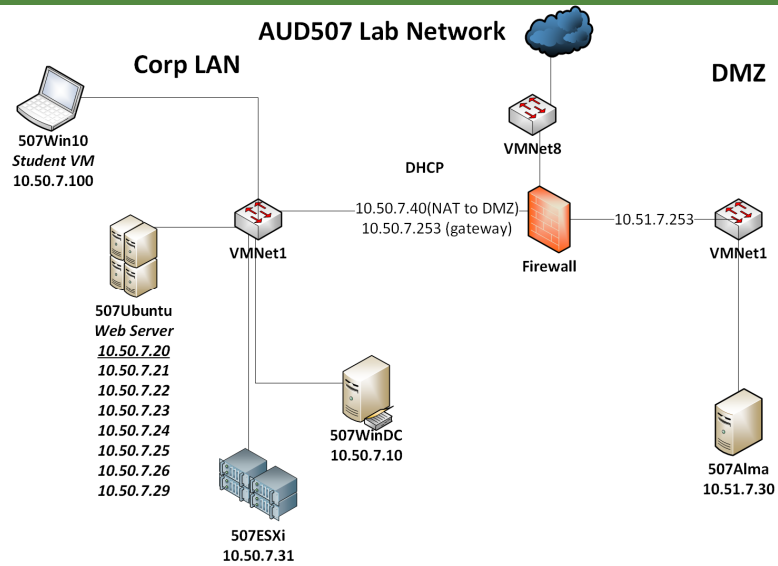
Serverless functions should be limited to a single role within the IAM strategy as an enforcement of least privilege.

Security checks should be included in the development pipeline to ensure that only well-tested code is deployed to production.

Access to functions should be as limited as possible, particularly if the function has direct access to sensitive data. Restrictions can be placed by user, subnet, virtual machine, etc.

Finally, make sure that all access to the function is logged to persistent storage to aid in troubleshooting, incident handling, and attribution.

Exercise 4.2 - Auditing Docker Security



This page intentionally left blank.

August 10, 2021

Course Roadmap

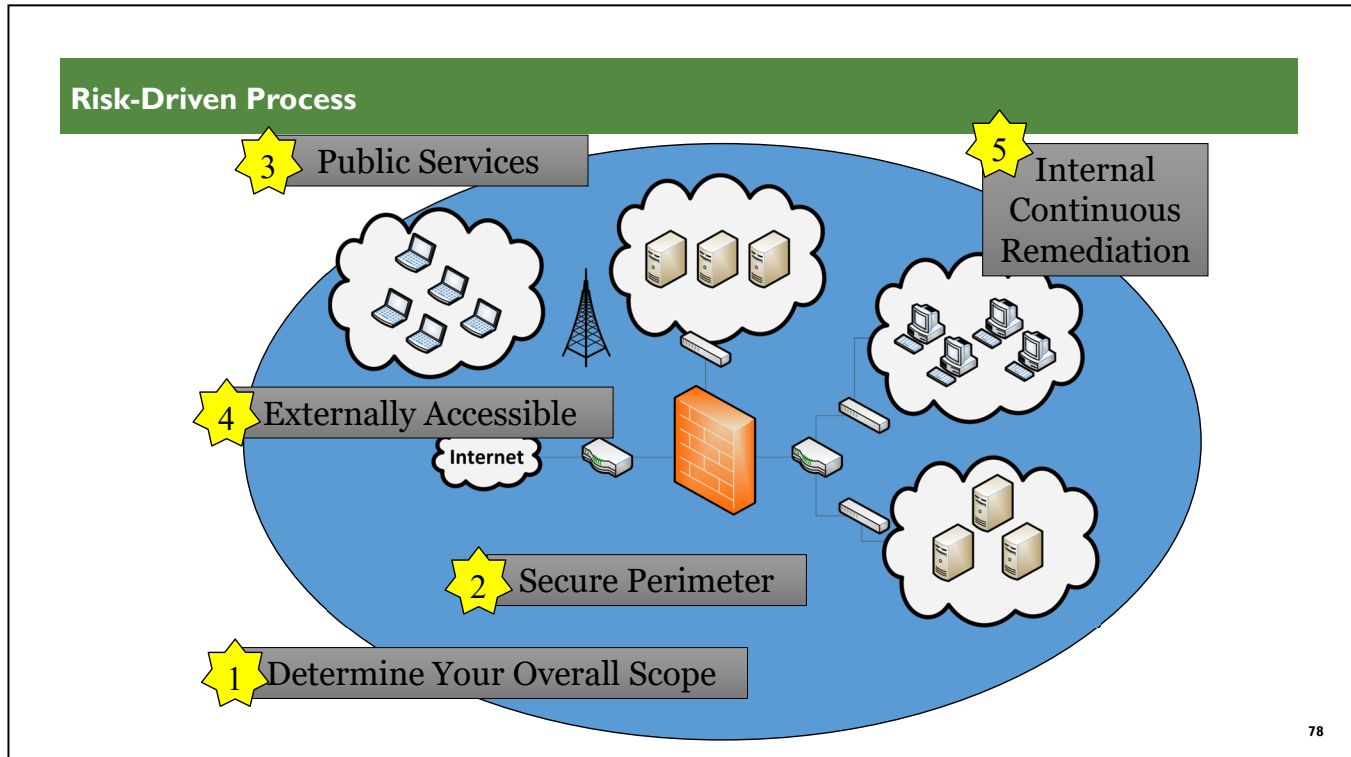
- Enterprise Audit Fundamentals; Discovery and Scanning Tools
- PowerShell, Windows System, and Domain Auditing
- Advanced UNIX Auditing and Monitoring
- **Auditing Private and Public Clouds, Containers, and Networks**
- Auditing Web Applications
- Audit Wars!

Section Four

1. Private Clouds
2. The Public Cloud
3. Containers
4. **Networks and Firewalls**
 - Risk Driven Process
 - Networks
 - Cisco Device Auditing
 - Exercise 4.3 - Wireshark, Switch Configuration Symptoms and Device Configuration Auditing
5. WIFI and VPNs
6. Public Services

This page intentionally left blank.

August 10, 2021



Another valuable thing to realize is that the overall design of the course is an example of the risk-based approach we discussed during the first course section. If we are charged with securing, auditing, or assessing the enterprise, how do we know where to start? Remember our discussions of risk and alignment with the business objectives. We need to understand, in the context of the business, which business systems are most critical to the actual mission-critical objectives. With this understanding, we can then create a program that seeks to evaluate technical systems based on their overall relationship with those business systems and objectives.

Because every organization will be somewhat unique, we take a more generalized approach in the course. After determining exactly which systems in the enterprise we are responsible for (our scope), we begin by analyzing the perimeter. That is actually what we are doing in this section.

We want to start with the primary threat-mitigation system in our enterprise: The firewall. After the firewall is thoroughly audited and validated, the next highest risk for penetration from the outside would be publicly exposed systems such as DNS servers, mail servers, and web servers. Included in this category would be wireless access, out-of-band management, and other similar entryways. After these are secured, we want to analyze all the systems and services that are externally accessible to third parties such as suppliers and distributors. Not only do third parties have access to these, but often those third parties might even have an interest in, shall we say, extending their access. With these issues addressed, we can then turn our attention to internal system security and continuous monitoring and remediation processes.

Although we target perimeters and networks here, the picture shown in the slide is a macro view of the entire course. We drill primarily into perimeters and network infrastructure, which are absolutely critical from a security perspective. Elsewhere in the course, we dig deeply into web application security, the public service in our enterprise that is likely the source of greatest risk.

The Key: Have a Plan!

- You may start somewhere else:
 - Have a thought-out plan
 - Take into account sources of risk
 - Account for organizational priorities

What this should illustrate for you is a critical aspect of the planning process discussed in Section 1. Having a plan is absolutely key to your success as an auditor, as a security officer, as a security engineer, as a system administrator... It does not matter what your job role is.

Take the time to think things out. Strategize. Take into account sources of risk for *your* organization or systems. Identify organizational risks and risks within your business sector. The plan that we outline in this course is such an approach, but you can customize it based on your experience and the specific issues within your enterprise.

As a basic example, I would typically advise organizations that the last thing they should be looking for on their networks is malware. Hunting for malware often involves running vulnerability scanners in an attempt to identify backdoors and other suspicious network services running on endpoints. Although this may initially sound like an important exercise, in the overall context of enterprise security, validating that the firewall is actually protecting us, and that the overall perimeter is secure is far more important. Why? Because if we hunt for malware, we will absolutely find stuff! However, what we find will most often not be malware! Instead, we will be spending an enormous amount of time chasing down false positives as a result of one-off services running throughout our enterprise on ports that seem suspicious.

This can actually serve to hinder the overall security process because all the false positives desensitize the organization to the reality of the threats!

However, if we are engaged by an organization that has been recently compromised, the need to hunt for malware rises significantly in importance! It is still critical that the perimeter is secured first, but the next step would likely be hunting down malware rather than dealing with public services.

Networks and Switches

- We start with networks and switches:
 - Typically start auditing firewalls
 - Much of what we cover now applies to routers and firewalls
 - Working up the network stack
 - Cisco as the “research” example:
 - Who has something that’s Cisco



Realize that in terms of risk, Layer 2, Ethernet, and VLANs are not where we’d like to start. We’d like to begin with routers and firewalls because those are the perimeter security devices. However, because we’re eventually going to have to discuss Layer 2, we may as well just start there because it is the underlying transport for our firewalls and routers.

Although portions of what we discuss in this section are specific to Layer 2 (things such as VLANs, Spanning Tree, TRILL, and other topics), there are also many things that these systems have in common with all the rest of our network infrastructure components, including our routers and firewalls—specifically, how they are administered, what logging is enabled, how they are secured, and many other items.

As we cover the material, we try to avoid repeating ourselves; there is far too much material here to cover any topic more than one time. This means that after we discuss the proper way to manage a generic network device (discussed in the context of switches in this section), we do not revisit that particular configuration requirement in the subsequent technologies. In other words, all the general items that we cover earlier in the section apply to all the other technologies that we discuss later in the material.

It is useful to have a specific kind of system to use as an exemplar of the concepts under discussion. For this purpose, we have selected Cisco as the example. Why Cisco? Well, ask yourself this: “Do we have any Cisco hardware anywhere in our enterprise?” Chances are the answer is either “We are a Cisco shop” or at least “Yes, we have Cisco stuff somewhere.” Cisco is truly ubiquitous.

Another great reason to pick Cisco devices is that it has an extremely wide range of features. You will hear more about this as we discuss routers and firewalls. They support some of the worst features in addition to having support for some of the best features in the same category. This enables us to discuss all the types of things you will encounter while researching just one type of device!

Ethernet

- You should already know:
 - LAN protocol
 - CSMA/CD
 - IEEE 802.3 standard
 - OSI Layer 2 protocol
- What you may not know:
 - Almost no security
 - You can't write Ethernet firewall rules



Let's start with Ethernet and a quick reminder of what you should already know. Ethernet is the most commonly used protocol for interconnecting computer systems. It was originally invented in around 1973 by a gentleman named Robert Metcalfe at Xerox. Ethernet was revolutionary because it allowed for shared network connectivity by multiple machines, which was not a new idea, without an artificial means of brokering communication. Specifically, it allowed for more than one system on the network to have the capability to speak at the same time.

Ethernet, in fact, is defined by this capability. For something to qualify as Ethernet, it must be Carrier Sense Multiple Access with Collision Detection (CSMA/CD). The standard that fully defines Ethernet is IEEE 802.3, which is also the most common encapsulation type used for Ethernet today. There actually are other Ethernet encapsulations that could be used (802.2, SNAP, and more), and you may actually find these on networks that have been around for a while, but from our perspective, they all work about the same way.

The biggest thing that you need to know about Ethernet and Layer 2 in general is that absolutely no security controls can be implemented at this layer. Now, it is true that I can configure things like Media Access Control (MAC) address filtering on switches and other network devices to prevent a system from speaking, but that's about it.

Here's a great way to illustrate what we mean by "no security." Imagine that we configure the firewall on a host to say, "If anyone asks you for anything, don't answer. In fact, you're not even allowed to initiate connections to anyone!" Effectively, we are putting in a DENY ALL policy, both inbound and outbound. When this is configured, a remote host attempts to ping the IP address of our locked-down host. To do so, the host that wants to send the ping begins with a Layer 2 ARP who-has packet, attempting to determine which MAC has that IP address. Despite all the firewall rules, the locked-down host instantly responds, sending an ARP is-at packet back! The firewall is like a castle gate that comes all the way down... but stops short of the ground, leaving a space for things to sneak underneath.

Switches

- Primary reason for creation:
 - Limits the collision domain
 - Introduction of VLAN technology
 - Reduces cost per port
- Not a security system:
 - Has security side effects
 - Frequently deployed insecurely

The way that Ethernet is designed, it is intended to have shared media with a shared collision and broadcast domain. Unfortunately, this is not terribly efficient when we start to have a hundred or more hosts on a LAN. While everything will still work well, especially when there is high utilization on the network, things will get slow, dropping the overall bandwidth that any particular node experiences dramatically.

To improve this, switch technology was created. Switches operate at Layer 2 and segregate the collision domain. Much like an old-time telephone operator who connects one phone line to another with a “switchboard,” creating a circuit, the switch creates a packet-switched connection between two points. As you know, this means that packets should be sent only to ports on which the destination MAC address for the packet appears.

Switches were a wonderful advance in networking technologies in the early 1990s, but they were significantly more expensive than the hubs they were replacing. To alleviate this, VLAN technology was created. This is an important point: VLANs were not created to make networks secure or to provide security features. VLAN technology was created to reduce the overall cost per port.

Cost per port is calculated for switches based on the overall cost of the switch divided by the number of ports that are actually populated or in use. If I have a 48-port switch but populate only 10 ports, the cost per port is the total cost of the device divided by 10. VLANs enable us to fully populate a switch while simultaneously segregating the collision domains *and*, between VLANs, the broadcast domain.

This certainly has positive security side effects. Because this feature was not designed as a security feature, however, VLAN technology is easy to defeat from a security point of view. Added to this, an improper deployment makes some type of compromise trivial.

VLAN

- Virtual LAN:
 - Can serve multiple networks from a single switch:
 - Limits the broadcast domain
 - Can trunk data between switches:
 - Makes systems that are on different physical switches seem as though they are all local

So, what is a VLAN? VLAN simply stands for Virtual LAN. In other words, through this technology, we can virtually make it appear that a group of physically separated systems, possibly even connected to different switches in different geographic regions, are all on the same local LAN. This is accomplished by assigning ports on a switch into a specific VLAN. As a point of interest, although we might assign names to VLANs, these names are only available at the management interface level; the technology itself supports only numbered VLANs.

Because the switch can have different ports assigned to different VLANs, at the switch level it is theoretically not possible to have data pass from a port in one VLAN to a port in another VLAN without passing out of the switch and through some type of Layer 3 routing device, like a router. This is accomplished primarily by limiting the broadcast domain, which prevents hosts from discovering one another at Layer 2.

To support the capability of a geographically separated VLAN, or simply supporting more than one switch with ports on the same VLAN, it is necessary to provide a mechanism for the switches to share data. Proprietary solutions to this problem are achieved through the use of backplane connections in a switch stack, effectively turning a stack of switches into a single large switch. All the data for VLANs that must pass to another switch in the stack is trunked over this proprietary backplane connection.

What if the switches are not physically in the same stack? No problem. All our enterprise switches also support the capability to trunk this data “in band.” This does not mean that the trunked data is visible to all hosts, but that we are actually using switch ports to trunk the data to another switch.

This trunking concept is actually important for us because an incorrect configuration can lead to a malicious user tricking a switch into trunking, or bundling, all the data for multiple VLANs to him, providing a wonderful means for data collection and attack.

How It Works

- Inbound packets are tagged using 802.1Q protocol:
 - Up to 4,096 VLANs possible:
 - Two, however, are reserved
- Trunk links between switches:
 - All packets for member VLANs passed between switches



The way that the switches handle VLANs is by tagging packets as they enter the switch port. The entering packet is tagged with the VLAN ID of the current VLAN for that port. This tagging will be done by adding a tag header to the packet, not by modifying the existing packet.

In the Cisco world, the protocol that is used for this is 802.1Q. Although not all vendors use 802.1Q, they all use some protocol that essentially accomplishes the same task. One of the things that's so interesting about this mechanism is that the switch actually does not keep track of which port the packet entered the switching fiber through. In fact, it creates a potential avenue for an attacker to "hop" VLANs.

Hopping VLANs is usually accomplished by taking advantage of trunking. Trunking describes a set of ports that are interconnected between two or more switches over which all the VLAN traffic is bundled between the switches. If a user workstation manages to convince a switch that it is also a switch, it is possible for the user system to select which VLAN it is on because it can pre-tag the data before handing it to the switch. Because there is no validation or tracking for this tagged data, the user can essentially choose which VLAN to join.

Of course, if the user can convince the switch that he is a switch, he can also subscribe to multiple VLANs, convincing the switch that he has ports on those VLANs. This will convince the switch to begin trunking all the VLAN data to the attacker, which allows for easy sniffing.

Management Layer Protocols

- Domain management:
 - Who's who in the VLAN domain:
 - **VTP:** Cisco
 - **GVRP:** HP
 - **MVRP:** IEEE Standard
- Bridging/Switching:
 - Getting packets from here to there:
 - STP and MSTP
 - SPB or TRILL



When we have two or more switches configured into a VLAN domain or VLAN trunking domain, there needs to be a single switch that acts as the master or manager of the domain. The job of this system is to keep track of who's who and to define the proper path for packets to take through the switching fiber. Why? Because using these management protocols, it is possible to completely interconnect all the switches, which allows for full redundancy even if one or more switches goes down.

The primary management of the VLAN domain is done using a protocol such as Cisco's VLAN Trunking Protocol (VTP), HP's GARP VLAN Registration Protocol (GVRP), or the IEEE Standard's Multiple VLAN Registration Protocol (MVRP). This protocol is used to track which VLANs exist and can be secured to require authentication or other requirements to join the trunking domain.

After the switches are connected, a secondary Layer 2 management protocol such as Spanning Tree Protocol (STP) or Multiple Spanning Tree Protocol (MSTP) is used to control the path through the switching fiber. Alternative protocols that accomplish essentially the same task are Shortest Path Bridging (SPB) and TRansparent Interconnection of Lots of Links (TRILL). Of these, TRILL and SPB are the newcomers on the scene. We are waiting to see which of these two turns out to be the new "right way" to do things. Currently, Cisco fully supports TRILL, with no support for SPB. However, the rest of the industry seems to be leaning toward SPB. We currently seem to be in the Blue-ray versus HD-DVD limbo, so we'd advise using hardware that supports both.

Why Do We Care?

- Many rely on these to provide secure network segregation
- There are many common misconfiguration issues
- Even the actual security features can be defeated (at times)

The reason that we care about all the details, though, is that many organizations are relying on VLAN features to provide secure network segregation. The risk is that if there are any misconfigurations, compromising VLANs is exceptionally easy. Add to this some of the natural behaviors of switches, and the problem is exacerbated.

The three most common configuration issues are a failure to configure a password to secure the trunking domain protocol, failure to remove Layer 2 management protocols (VTP, STP, MSTP, TRILL, and so on) from user-facing ports, and failure to secure the configuration of the switch. The first one is obvious. The trunking domain has to have a password configured. Frequently it is not configured because the administrators don't see a need to do so; their perspective is that the users can't see it or get to it, not realizing how switches and VLANs can be attacked.

The second issue is fairly easy to resolve; though sometimes other network configuration issues prevent us from fixing it properly. For example, if we use a "Converged Network" configuration with a VOIP handset, we must have the Layer 2 management protocols enabled on user-facing ports. We'll give you a specific way to test for this shortly. We will also give you details of specific checks for the third issue.

The last bullet on the slide, however, makes an interesting assertion. Even features within switches that have been specifically designed as security features are easy to defeat. We'll show you an example in a few minutes when we discuss a Private VLAN Bypass attack, but this should raise your awareness that there are serious security concerns at Layer 2!

Audit Items

- How is it managed
- Logging appropriate
- Is it up to date
- What services does it offer
- Only required ports are active
- VLANs properly secured



Now that we have some foundation, let's start dealing with auditing the device. Although some of the questions that we ask here are specific to switches (for example, are the VLANs properly secured?), others are applicable to all systems that we cover in the course (for example, is the logging appropriate?).

Before we jump into answering these questions and doing the research to see what the answers might look like, I'd like to remind you of the value of conducting interviews. Even the questions above form an excellent basis for such an interview. Let's just take one of these and examine what that interview might sound like.

We would provide the administrators who will be interviewed with a set of questions much like what is listed in the slide. We might indicate in an introductory paragraph or cover letter that we would like the administrator to come to the meeting prepared to discuss the questions posed. An example of how to word a question might be, "If VLANs are used in the network, what steps have been taken to safeguard the network from attack or manipulation?"

The interview is not an intense grilling. It's a friendly discussion. It allows us to get a feel for the approach that the administrator and his team take toward the issues presented, and it can give us insight into items that we need to examine more closely. It also allows us to identify issues that we don't need to spend time on at the console with the administrator.

What would a proper configuration look like for issues like these? How can these controls be circumvented? Let's examine these questions.

Management

- Managed over a secure channel:
 - TLS
 - IPSec
 - SSH
 - SNMPv3: Proper config discussed later
 - Dark network



For a network device, firewall, router, or switch to be managed properly, we would expect the management to occur over a secure channel. Our preference would be to find that a strongly encrypted protocol or connection is used to accomplish this. SSH and IPSec are common solutions to facilitate this.

At times we may find that HTTPS is used. This may raise a red flag for you. If the device in question is something like a wireless access point, this may not be a problem. If the device is a switch or a router, it likely is a problem.

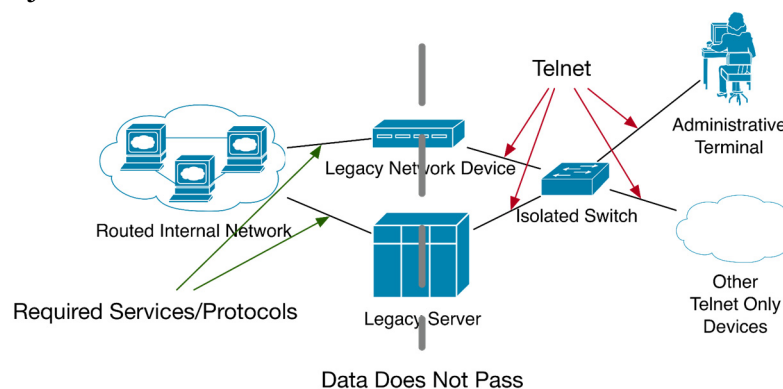
Even though these devices generally provide a web-based interface, almost all seasoned administrators will tell you that if you are using the web interface to manage a router or switch, you are doing it incorrectly. The reason is that the web interface represents a potential vulnerability in addition to the fact that many of the most important security capabilities of the device are simply not configurable through the web interface.

Another possible problem is that the device is an older piece of equipment and does not support a strongly encrypted link. This issue has become rarer to see with network devices, but it remains a fairly common occurrence with legacy systems. In this case, a good approach is to create a private LAN, a dark network, that is not routed or reachable from the production network. All the management interfaces for these legacy systems are connected to this dark network, along with a management workstation or gateway. The administrators can now use a secure protocol to connect to this gateway and then use an insecure protocol to perform the actual administrative tasks in a safe way.

SNMP could potentially be in use for managing devices. We will expand on how this should be configured in a little while. For right now, we'll just say that there are more questions to ask if SNMP (Simple Network Management Protocol) is used for management.

Compensating Controls

- What if your switches/routers/other things cannot support TLS/SSH/Encrypted management?
 - Secure Admin Objective: Protected from **interference and interception**



If you run into a situation where the organization has some legacy equipment and encrypted methods are not available for the administration channels, this doesn't automatically mean that the network fails the inspection. Instead, look to see if appropriate compensating controls have been implemented. Alternatively, you may even recommend that the organization implement compensating controls.

Ideally, every organization would prefer it if its systems all met all of the security and operational requirements. Unfortunately, this is rarely the case. Over time, systems become outdated, manufacturers go out of business, and it takes time and money to retool the business operation to accommodate for this. Simultaneously, legacy systems with security vulnerabilities, including a lack of encrypted administration features, persist in our network.

Compensating controls are sufficient, provided that they meet the original objective of the actual control. When it comes to the requirement of encrypted administrative channels, what is the actual objective? It's not to encrypt the data. Encrypting data is the *control*. The *objective* is to protect the administrative activities from interference, interception, and unauthorized access. Is it possible to do this with Telnet?

At first it may seem that the answer is "no." Take a moment to look at the network diagram in the slide, however. Notice that the legacy systems all have network interfaces that are facing the internal network. They also have additional interfaces that are connected to an isolated, private, "dark" LAN. Other than other legacy systems, the only thing on this LAN is an administrative terminal. Administrators seeking to work with these systems must first either connect to or physically sit down in front of the administrative terminal. From there, the administrators can use Telnet to connect to the legacy systems. Does this protect the administrative session from interference, interception, and unauthorized access? Absolutely.

Centralized Authentication?

- Enterprise-class systems all support centralized authentication:
 - TACACS and RADIUS are most common:
 - Solves password change issues, user departure issues, and prevents brute forcing
- Local passwords:
 - Escrowed for emergencies
 - Must be changed on use

For managing the device, it is not sufficient for it simply to be done over a secure channel. We would additionally like to find that strong password requirements are enforced and that users are distinguished from one another in the logs. The trouble is that almost all network devices on the market do not support these types of features. Even if they support the creation of separate user accounts and passwords, none of them have password-strength or length-of-expiration enforcement.

For this reason, centralized authentication must be configured. This allows us to reuse a trusted component in our infrastructure. In this case, that component will be a trusted credential store of some kind, most likely an Active Directory domain.

Using Remote Authentication Dial-In User Service (RADIUS) from a domain controller, we can create a domain-level group that is authorized to log in to routers. With this configured, and with the network devices configured to use RADIUS, each administrator must authenticate using his domain credentials when managing a switch or router.

This allows us to enforce strong password requirements and password change requirements, and it vastly simplifies the process when an administrator leaves the organization. Instead of having to change all the infrastructure passwords (which is rarely done in practice), we simply need to disable his domain account.

This does not mean that there is no place for local passwords on these systems. There should be an emergency account configured with a long and strong password that is escrowed for emergencies. If this password is ever used, it must be reset and re-escrowed.

TACACS (Terminal Access Controller Access Control System) is simply an alternative authentication protocol that typically runs over port 49. Cisco has created proprietary extensions (TACACS+), so you may find this solution in use in some environments.

Research Example

- Cisco supports all of these:
 - Requires AAA to be enabled:
 - “aaa new-model” with “default” configured
 - Requires SSH key to be generated:
 - “crypto key generate rsa”
 - Requires RADIUS to be configured:
 - “radius-server” commands
 - Should have a recovery mechanism:
 - Local user configured but restricted

Although Cisco supports a Terminal Access Controller Access Control System (TACACS), they also support RADIUS. Because RADIUS is more commonly used for this type of management, we'll look at how a RADIUS configuration might look.

To use RADIUS (or TACACS), the Cisco device requires the Authentication, Authorization, and Accounting system (AAA) to be enabled and properly configured. For the centralized authentication to function, we need to include RADIUS configuration options as well. We'll see an example on the next slide.

Turning on the authentication is one thing. We still need to ensure that a secure channel is in use. Because the most common approach to this across vendors is to use Secure Shell (SSH), we'll include that piece of research in our Cisco example. Using SSH on a Cisco device requires that the device have an SSH key. Either this can be statically configured, or we can tell the device to generate and store an RSA key for use in SSH sessions.

Finally, we would like to see that an emergency password for recovery has been configured and the password escrowed. Let's see what this all looks like.

Example Requiring SSH

```
! First configure an emergency recovery password
username emergency privilege 0 password 5 $1$Z3fs00.p$7alNA92A

! Turn on Authentication, Authorization & Accounting
aaa new-model

!Configure Radius authentication
aaa authentication login default group radius local
radius-server windows_server.ourdomain.com
radius-server key R@d1usP@55w0rdF0rUs3rAuthR3qu3sts

!Now configure SSH
ip domain-name router.ourdomain.com
crypto key generate rsa
line vty 0 4
transport input ssh
```



Let's examine and explain this configuration one piece at a time. As we do so, do not view this as a memorization exercise! Instead, consider whether you can figure out the gist of what the individual elements mean without an explanation. Take a moment now and just read through the lines in the slide and see if you can figure out, generally, what is happening. Can you identify all the pieces from the last slide?

Starting at the top, the username line creates a local user on the network device named "emergency" and sets the password. This emergency account is configured at privilege level 0 (no privileges) and has an MD5 password hash (specified by password 5). In other words, the password is not readable. This is good! We should not read plaintext passwords in the configuration here.

Next, we see the AAA being enabled, one of our requirements from the last slide. Following this, the "aaa authentication" line configures AAA to use RADIUS as the primary authentication mechanism for logins. If, for some reason, the RADIUS server is unreachable, the fallback method of "local," the emergency username and password, may be used. Another item checked off.

Following this we see the device name configured and an SSH host key generated, allowing SSH to be used. This is not all that is required, however. Note the configuration that begins "line vty." This configures the virtual terminal (or remote login) service. Immediately after that, we find the line "transport input ssh." This is the line that forces SSH to be used and is what disables the use of Telnet.

Hopefully, you deciphered the general outline of what is happening even without the description. However, even if you couldn't, you can likely see that we have met all the requirements from the last slide! The things that help us out here are the comments. Not every line requires a comment, but we should expect administrators to document configurations. This is a primary way that this is done!

Credentials Secure

- Verify that any local credentials on the host are securely stored:
- Cisco supports multiple forms:
 - Type 4 – SHA256 hash – deprecated
 - Type 5 – MD5 hash
 - Password on last slide
 - Type 7
 - Trivial to “Crack”
 - Type 8 – Password-based key derivation function (PBKDF2)
 - Type 9 – Scrypt
- There are some we can't fix, like the RADIUS server password on the last slide

Type 7 Password:	12090404011C03162E
Crack Password	
Plain text:	password

Even though we use centralized authentication, local credentials should be configured for emergencies. It is also common to configure the device so that logging in to it requires centralized credentials but then moving into the administrative mode with the enable commands requires that we enter a local password that rarely changes. Whatever the case, those local passwords must never be stored in plaintext. There are encrypted and hashed options available.

Cisco Type 7 passwords have been a part of the Cisco system for a long time. These passwords are not strongly encrypted, though they aren't in plaintext either. In a Type 7 password, the first bytes are used to specify an index into a known string of characters. Using this value, the remaining values in the password are then encrypted. All sorts of websites out there will decrypt these passwords instantly for you for free!

Type 5 passwords are much stronger. Rather than using a home-grown encoding algorithm, these passwords are hashed using MD5 with a salt. The configuration on the last slide uses an MD5 hash. Can you see how we know this? Notice that the configuration of the emergency account includes “password 5.” That 5 is what indicates that an MD5 hash is in use. If a 7 appears, it's a type 7 password. If a 0 appears, then the password is in plaintext.

Type 4 passwords used SHA-256, but that algorithm is now deprecated and should not be used. Types 8 and 9 use much more modern salted hashing algorithms.

Be conscious of the fact that there are some things that just can't be fixed. For example, in the last slide, you may have noticed that we can read the password that this device uses to authenticate to the RADIUS server. Unfortunately, there is just nothing that can be done about this. The router/firewall/switch needs to speak to the server (Active Directory in this case) and to do so it must have credentials. The only real mitigating factor here is that the password that we can read isn't the password to authentication to this network device.

What If You Use SNMP?

- Are there other options
 - Vulnerable to brute forcing
- Verify that it is SNMPv3
- Verify that it is used in a secure manner:
 - Usernames, not community strings
 - Hashed key-based authentication
 - Not DES - AES

While we're on the topic of remote management of a network device, let's revisit SNMP. As was mentioned earlier, if there is another option for managing the device, we'd probably prefer to use that. SNMP is not an especially secure mechanism, and the fact that it operates over UDP (User Datagram Protocol) makes it much more susceptible to replays and impersonations.

If, however, we use SNMP to manage devices, we need to verify that the systems are properly using the SNMP version 3 security features. When version 3 was formalized as a standard in 2004 (RFC 3411), it added some important security capabilities. Specifically, confidentiality was addressed by providing encryption capabilities, integrity was taken into account by adding message hashing for validation, and authentication capabilities were added through message signing. Even so, just because these features are available does not mean they are configured. And even if they are configured, it does not mean that secure options have been selected!

For example, under SNMPv3, you could choose DES as the encryption method. As you are likely aware, however, DES is not a useful algorithm for encrypting these days unless we are going to have multiple rounds with multiple keys. Another option is AES. If AES is used, we can have some confidence that our messages are secure. Similarly, while authentication can be enabled, whether or not that will be a plaintext username, or a hash-based authentication is configurable!

The key to knowing if it has been done correctly is to remember what we set out as the objective of this control. We said that remote management of the device must be protected against interception, manipulation, and impersonation. This would mean that we need all these security features enabled, or the device is not well secured.

Logging

- Centralized (typically syslog):
 - Timestamps synchronized
 - Informational recommended
 - Analysis and reporting
- Console:
 - At least warnings:
 - Frequently disabled for admin convenience



Logging must also be enabled. Without proper logging, we not only lose the ability to detect and react to faults in the logs, but we also lose the ability to account for administrator activity within the system. With logs enabled it is much easier to compare the administrator activity to authorized change control. Without it, everything must be inspected every time.

Proper log configuration would mean that the logs are being centralized. Network devices of the sort that we are discussing all support syslog style logging. Some have additional support for more secure forms of logging. If we have these additional features and choose to use them, that's wonderful. Even without them, however, simply having the logging enabled and configured to centralize the messages is sufficient in almost all cases. Frankly, securing the logs at the network level isn't our biggest problem. Much bigger problems are getting the logging centralized and then getting someone to look at the logs! Therefore, we will want to not only verify that the logging is enabled and centralized but that there is also a system or process for regular review of those logs.

When configuring logging, we can use a range of log settings to control which messages are tracked. These range all the way from emergency events down to debugging events. For remote logging, Informational is a good selection. This will include important information about configuration changes in addition to warning, critical, and emergency events.

We should also verify that the console of network devices is configured to generate logs. It is not unusual for an administrator to disable console logging to prevent log messages from cluttering the screen while she is troubleshooting or reconfiguring the device. Unfortunately, this means that if she inadvertently breaks something while she is there, she is unlikely to realize it immediately unless it affects the issue she is working on at the moment. If this is set to Warnings, she should see events like routes dropping, adding, interfaces going down, and more.

Logging Configuration Example

```
!Configure timestamps for milliseconds and display timezone
service timestamps log datetime msec show-timezone
```

```
!Configure multiple syslog servers for redundancy
logging syslog1.domain.com
logging syslog2.domain.com
logging trap informational
```

```
!Ensure console logging is enabled at warning
logging console warning
```

```
!Configure the clock to use NTP, UTC and no DST
ntp server time.apple.com
clock timezone utc
no clock summer-time
```

So, what would all this look like within a device? This slide shows an example. Just as we did previously, start by looking over the configuration and seeing how many of the items from the last slide you can identify or check off.

First, the log timestamps are configured to include millisecond resolution and the time zone of the system they were collected on. When it comes to the time zone and the actual time, all the clocks for our entire infrastructure must be synchronized to a common source. NTP is the most common way to do this. In addition, because these are simply text-based logs, we strongly recommend that you select a single time zone and have all systems configured to this. This will have no impact on users! The time that they see is localized on their desktops.

With that configured, the next several lines configure the device to forward all the log messages to two different remote syslog servers. This provides redundancy should one server be unavailable. In addition, we can see that the “logging trap” sets the system to send informational messages to these services. “Trapping” means that we are sending events. Compare that to the next configuration line. “Logging console warning” configures the console so that all warning messages appear there, exactly as we specified.

The last few lines are configuring the time synchronization and the time zone. Note the final line. For Cisco systems, this line tells them that daylight savings time is not observed. This is an extremely important setting. Without this setting, not only will our timestamps adjust automatically, but we also end up with a troublesome problem of not knowing when they will change unless we actively tell the system when to start and stop daylight savings time. Best practice is to simply disable daylight savings time to keep the logs all synchronized.

Patched/Up to Date

- Interview:
 - Select three recent alerts
 - Was it patched
 - Waiver in place
 - Review Change Control:
 - Patch tested
 - Back-out procedure documented
 - Device validated after patching
 - If it fails, dig for more

Returning to our interview process, we want to inquire about how patch management is actually done. Of course, we are already familiar with what the policy and procedure documents say; what we want to see is how well the administrators know this process and to hear whether they have followed it.

To facilitate this discussion, it is good to select three recent vendor notifications that are relevant to the infrastructure and should have been addressed. Ask the administrator to bring with him to the meeting any documentation related to those specific notifications.

During the meeting, we are looking to answer a few simple questions:

1. Was it patched?
2. Was it patched on time?
3. Was there a testing process before it moved to production?
4. Was there a back-out or recovery procedure in place?
5. Was it properly authorized and documented?
6. If it wasn't patched, is there an appropriate waiver in place?

What if the administrator cannot produce certain pieces of documentation? If this is the case, then we would want to let the administrator know that we will follow up with questions about additional issues that should have been addressed when we work with him to validate the device. In other words, if there appear to be issues, we look for more issues. If everything looks fine, we move on to other areas.

During the technical validation, we are looking for the administrator to demonstrate that the patches were actually applied. At times, we have found all the documentation in place but the patches missing.

Caution

- Reasons sometimes given:
 - Doesn't apply
 - We're not using that feature
 - We can't apply it
- Analyze the situation:
 - Actually doesn't apply
 - Feature actively disabled
 - Are you saying it's obsolete

As you work with the administrator on this process, we want to caution you about some things you might hear. To be clear, we're not saying that these are invalid claims, but we strongly suggest that you analyze the situation in context before accepting or rejecting any of these (or similar) situations.

You may hear the administrator say, "That doesn't apply to us." If he makes this claim, he should clearly explain why that is the case. For example, imagine that there is a vendor notification that the device, a switch, is vulnerable to a VLAN tagging attack in which a user could attach VLAN tags to the packet before handing it to the switch and thereby hop from one VLAN to another. You inquire about the device and the administrator says, "No, we haven't applied that patch because it doesn't apply to us." "But we have that switch," you think. What if the administrator explains that as a result of that alert the decision was made to not rely on VLANs with that particular brand of switch. Instead, the switches have all VLAN capabilities disabled and are used to physically segregate LANs. Obviously, that specific patch would not be required in this case.

A similar response might indicate that the feature isn't in use. If this is the response, you should seek evidence that demonstrates that the feature has been actively disabled. If it has not, that feature may end up being activated by some future patch.

Perhaps the most concerning response, however, is, "We can't apply that patch." In fact, the administrator may even have a waiver for this patch. The waiver means that the administrator is off the hook, but you are there to advise on risk. If you perceive that this is a critical risk, dig into why it can't be patched. Is the system obsolete? Is there some other configuration or design error? Is there a way to address this or otherwise mitigate the risk? Especially if we're talking about obsolete systems, make sure that management understands that waivers don't eliminate risk! The cost of upgrading that hardware will often be far less than the overall impact of the risk.

Services

- Applicable Principles:
 - Principle of Least Privilege
 - Economy of Mechanism
- Services
 - There are a lot
 - Necessity depends on deployment: CDP, for example

All network devices support ancillary services. If you have a web server, no doubt there are a number of services also installed that have absolutely nothing to do with serving web pages. Needless to say, any service that is not required for a system to perform its primary mission must be disabled or it creates a vulnerability vector on that system.

Switches, routers, and firewalls are no different. They all support many, many services, the vast majority of which are not necessary for the system to achieve its goals. Therefore, according to the principle of economy of mechanism, all of them should be actively disabled. What if there is a service that is required? No problem. However, we would want to verify that the principle of least privilege has been applied in this case. In other words, verify that the service is running only with the rights required for it to do its job. Of course, it will work when it runs with the highest privilege, and that is the easiest way to run it, but this means that any vulnerability in that service will lead to a compromise of the highest credentials available.

This is another situation in which there is no “one size fits all,” even within a single organization. We have to look at exactly how the device is used and what requirements exist. For example, the router interface that faces the server segment likely does not need the BOOTP agent running. The same would be true of the core routers. The routers facing the user segments, however, likely do have the BOOTP agent operating as a forwarder for DHCP requests.

Similarly, CDP in the Cisco world should be disabled because it broadcasts a lot of configuration and VLAN information in cleartext on the network. However, if you use VOIP with a converged network configuration, you must leave CDP enabled to allow the phones to auto-configure. Again, everything unnecessary should be disabled, but make sure you actually understand what is necessary and what is not!

Actively Disabling Services

```
!Disable unnecessary services
no service tcp-small-services
no service udp-small-services
no ip finger
no ip bootp server
no service dhcp
no mop enabled
no ip domain-lookup
no service pad
no http server
no http secure-server
no service config
no cdp run
no lldp run global
```

What might this look like in the configuration file of the exemplar device we use for our research? The slide above illustrates this. We can see that all the small services (echo, chargen, and so on) have been disabled on both TCP and UDP. The Finger service, which allows a remote user to request user information and see who's logged in, is disabled. BOOTP and DHCP are also turned off, as is MOP.

If you're not familiar with Maintenance Operations Protocol (MOP), don't worry too much about it. It is rarely found today, but in a legacy environment with perhaps VAX systems, you may find that satellite VMS systems use MOP to perform their initial boot, a bit like a BOOTP-based system might.

For Cisco devices, we also see DNS services turned off, the web service disabled, the remote configuration service turned off, CDP disabled, and LLDP disabled. Everything that we need, nothing that we don't!

Administrators can be tempted to rely on the configuration defaults of the embedded operating system rather than explicitly disabling unnecessary services. This is very risky! Vendors are constantly changing their minds when it comes to which services should be enabled or disabled by default. To prevent a service that was previously disabled by default from becoming a vulnerability when it is automatically enabled, best practice is to explicitly disable everything that is not required.

VLAN Management

- Verify:
 - VLAN domain is password protected
 - Layer 2 management protocols limited to switch-switch and management ports
 - VOIP can present real challenges here
 - Use BPDU guard type controls
 - Only switches and admins on VLAN 1

With those general device configuration issues addressed, we can now turn our attention to switch-specific issues. Essentially, we want to find that all efforts have been taken to defend the device against the types of attacks and issues previously mentioned.

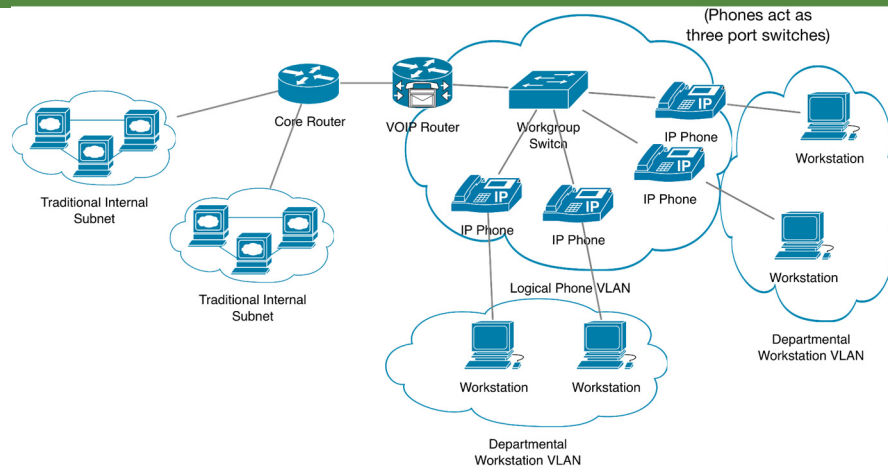
Recall that we said that the biggest problem is that no one has bothered to secure the VLANs. This is actually one of our first questions: Has a strong password been configured to protect both the configuration and the VLAN domain?

The other enormous issue is when we have Layer 2 management protocols accessible or enabled on user-facing ports. As we said, these protocols should only be available on ports that require them. This typically means only ports that are facing or connected to other switches. This is one of the most important configuration settings to secure a VLAN environment. Consider this: What if there were no VTP domain password configured, but user-facing ports had all the VLAN management protocols disabled? This would make it vastly more difficult to attack. Unfortunately, how our network is used can prevent us from doing this. Again, VOIP with network convergence will *require* that these management protocols are enabled on user-facing switch ports. ☹

Especially in these cases (even if this is not an issue you have), it is useful to have BPDU protections enabled. Bridge Protocol Data Unit (BPDU) controls enable us to limit which switches can actually be members of the VTP domain. In addition, it can enable us to lock down the root of the STP or MSTP so that an unauthorized device cannot take control of the switching fiber. Without a control of this sort, an election can be triggered and the device with the lowest MAC address will become the new root.

An additional protection is to keep users off of VLAN 1. Only switches and administrators should be on this VLAN. Switches more or less “reserve” this VLAN for themselves.

VLANs and VOIP



We often protect the phones from the computers, but are we protecting the computers from the phones, too?

VOIP (Voice over IP) can introduce some very specific vulnerabilities into your network. One of the biggest issues is related, again, to network controls. In our experience, your VOIP network may or may not be managed by IT personnel; frequently, facilities staff will be responsible for the VOIP portion. Either way, individuals tasked with deploying and managing the VOIP network and phones will rarely think a great deal about security. After all, the VOIP network and phones are all behind the firewall.

The reality is, however, that VOIP phones tend to be deployed with very few security protections. All of the phones we have worked with have embedded UNIX-style operating systems with command-line interfaces, frequently with default passwords configured. This makes it clear that protections are needed to defend the phones from the rest of the network, in addition to security standards and standard configurations for phones that are deployed on the network.

Even if the organization has implemented controls to defend the phones from the network, it is very rare to add controls to defend the network from the phones. Remember that every single phone has a 10- or 100-megabit network cable plugged into it. It is trivial for someone with physical access (like an employee) to impersonate a phone, even bypassing network authentication controls like 802.1X (which we will discuss briefly later in this book).

Therefore, in light of the foregoing points, network-level ACLs both in the core of the network and in the switches at the fringes of the network become important protective measures. Coupled with logging, these can also act as detective controls, provided the logs are being reviewed and correlated.

Securing VLANs

```
!Set up our VTP domain name to logically associate vlans
vtp domain CorpVLANs

!Set switch to operate as a VLAN server using vlan database
vtp mode server vlan

!Configure a VTP domain password - all switches need this
vtp password Sup3rStr0ngP@55w0rd

!If a port unexpectedly tries to participate in spanning
!tree, disable the port
spanning-tree portfast bpduguard

!Enable VTP on required ports
interface gigabitethernet 0/1
vtp
```

Let's see what this would look like in the real world. Again, look over the configuration and see if you can find the elements that we are looking for being configured. After that, read on for an explanation of what you see.

First, a VLAN Trunking Protocol (VTP) domain is configured and given the logical name CorpVLANs. Next, the switch is configured to act as a server, housing the VLAN database, within the VTP domain. In addition, a password is configured. All switches within the VTP domain need this password to exchange data. This password is used to authenticate VTP packets and serves to prevent a user from injecting any kind of configuration packet to interfere with the proper operation of the domain.

The next line configures one of the BPDU guard options: Spanning Tree—a protocol for deciding the path a packet takes through the aggregate switching fiber—needs to be defended. This line says that if Spanning Tree packets appear on any port that is not configured to be connected to a neighboring switch, that port should be immediately shut down. This is an effective way for not only preventing corruption but also for stopping an attack in its tracks!

Lastly, we're looking at the configuration of just one of the switch ports. You can see "vtp" listed, which enables the VTP on that specific port. This means that the Layer 2 management protocols we are talking about would be enabled on this port. This would also imply that this port is connected to another switch within the VTP domain. Easy!

Securing Ports (I)

- New switch: All ports VLAN 1
 - Process should be:
 - Secure configuration
 - Create “parking lot”
 - Move all ports to parking lot VLAN
 - Allocate ports to specific VLANs as needed
 - Disable all unused ports
 - Enable port security functions



Let's revisit VLAN 1 for a moment. This was mentioned on a previous slide and received only a sentence or two of explanation. VLAN 1 is the default VLAN for all the ports on the switch unless you actively configure them to be on some other VLAN.

As a result of this default behavior, it is not unusual to find a switch where either all or at least many of the user ports are on VLAN 1. Why? Because when the switch was first deployed, it was serving users who were all on the same subnet in our network. If this is the case, VLANs are not necessary. Because they weren't necessary, the administrator did not bother to change the default configuration.

Later, additional users were added who were on different subnets, and VLANs began to be configured on this switch. Rather than moving all the users off of VLAN 1, the administrator simply started creating new VLANs and left the original users where they were. Why? Because moving them is work, and because many network administrators do not appreciate the vulnerabilities inherent in our switching environments!

The correct way to configure a new switch is to immediately move all the unused ports into a parking lot and disable them. VLAN 1, by policy, is reserved for the switch and inter-switch ports. Now as ports are allocated, they are moved out of the parking lot and configured for the VLAN that they should be on.

Why move the ports to a parking lot? Isn't it enough to just disable them? Disabling is good, but the parking lot is even better long term because if a port is later enabled accidentally, it should not be connected to a network that goes anywhere.

It's also excellent to enable any port security functions if these are available. Let's see what those might be as we examine an actual configuration.

Securing Ports (2)

```
!Disable unused port
interface ethernet 2/1
shutdown

!Standard ethernet layer 2 switch port
interface ethernet 2/2
switchport

!Remember mac addresses, with a maximum of 2 in any 30 minute
!window. If a third address appears, shut down the port.
switchport port-security mac-address sticky
switchport port-security maximum 2
switchport port-security aging type inactivity
switchport port-security aging time 30
switchport port-security violation shutdown
```

If you go back two slides to the last configuration you saw and then come back here, you will realize that this is simply a continuation of the same configuration. In fact, all the configuration slides so far come from the same file.

In this part of the configuration, we can see two ports being configured. Port 2/1 is being disabled because it is not in use. Port 2/2, however, is enabled and is marked as a switchport. This enables the port for use.

The elements below set up a variety of security controls for all enabled switchports. Let's explain each one in turn. The first, "mac-address sticky," means that the switch will remember which MAC last appeared at each port. This enables us to prevent computers from moving around, or at least to detect it.

The next option, "maximum 2," configures the switch so that no more than a maximum of two MAC addresses may appear on any port. This is a great setting; though, if we are cascading switches, we would need to disable this setting on any ports that are cascaded.

As wonderful as this setting is, the following three lines govern how it works and are important. Let's start with the last line first. This line means that if a third MAC address is detected, the switch port is disabled. An alternative would be to just prevent the new MAC from functioning, but this is easily bypassed. Disabling the port requires help desk intervention and provides us with detection capability.

What if we need to replace a computer? No problem. We have also configured the switch so that we may have a maximum of two MAC addresses within any 30-minute window. How is that window measured? The aging does not start until the port is inactive. This gives us great port security, allows for the help desk to replace computers, and provides us with a detection mechanism if someone is "messing around."

Securing Layer 2

- Requirements:
 - Remove unneeded protocols
 - Lock down ports
 - Protect VLAN topology
- A lot of attacks
 - MAC flooding, ARP poisoning, leveraging private VLANs, and more

We said earlier that one of the best protections at Layer 2 is verifying that the management protocols are not available on the client ports and locking down unused ports. We also talked about defending the VTP domain as another protection. But what are the risks?

If we don't know of the risks, it can be difficult to justify these requirements to the organization. We will describe just one or two of these attacks. Remember our context, though. VLANs, although useful and although they have security side effects, are not designed as a security technology. Their development was driven by economic benefits.

First, if we connect a sniffer to any user port on a moderately busy switch, we occasionally see packets that are neither from nor to us come out of the port. We're not just talking about broadcast traffic, here. We mean actual TCP connections with full data for two hosts, neither of which are us. We could see packets for hosts that aren't even on our VLAN! Why does this happen?

The switch has a limited amount of memory and capacity for switching data. This is often referred to as the *switching fiber*. Imagine a 48-port switch with gigabit ports. Imagine that it has 20 gigabits of switching fiber. What if computers on 30 or more ports are all streaming large data files simultaneously? That switch will do its best to keep up, but when the switching queue fills, it will flush all the pending packets out of all the ports as quickly as possible, no longer switching! Now that it has recovered, it will begin switching again. This could have serious ramifications if we have VLANs of different sensitivity levels on the same switch!

It is possible for an attacker to try to create this type of situation. Typically, this is accomplished not by sending a lot of data but by injecting thousands of unsolicited ARP replies. The switch caches these addresses in its CAM (Content Addressable Memory) table (a list of MAC addresses), and when that fills, it may begin to broadcast packets to all ports!

Layer 2 Management Protocols

- Things to look for:
 - VTP
 - GVRP
 - STP
 - MSTP
 - HSRP
 - VLMP
 - MVRP



When it gets right down to it, the network engineers should have a clear understanding of exactly what's on the network and why it's there. For us, though, the slide provides a good starter list of protocols that we should not see if we were to run a sniffer connected to a user-facing port. All the protocols listed are related to managing the switching fiber, managing VLANs, registering hosts into VLANs, providing failover capabilities for routers and switches, and so on.

To be clear, there's nothing wrong with using these protocols. In fact, if we use VLANs, we must use at least one or two of them! However, these protocols should be enabled only on ports that connect directly to other switches (trunking ports) or ports used for administrative purposes. A user workstation should not see Spanning Tree packets being broadcast at it. If these protocols are user-facing, then we are saying that it is almost always an indication of an insecure and inappropriate configuration.

August 10, 2021

Is There an Easier Way?

- Yes, absolutely
 - Right now, we're in our "deep dive"
 - We don't need to configure but we do need to "interpret"
- We'll use a wonderful configuration analysis tool later in this section
 - Works for switches, routers, firewalls, and so on

You might be wondering whether you need to know all this stuff. The answer is, "sort of." You don't need to be an "expert" in networking protocols and Layer 2 functionality, but you should certainly have enough familiarity that you can look at a configuration and puzzle out, at least in general terms, what's happening. Even better, you can go to a configuration with a specific question like "Is the device configured to require an encrypted management channel?" and be able to answer that question.

There is good news, however. You may have noticed (and will notice more so in a few minutes) that we haven't looked at how to test these items beyond asking questions about the configuration. The good news is that we will look at a useful tool that can be used for routers, firewalls, and switches to perform an analysis of the configuration easily!

For now, just hold on and keep in mind that we're going to step back to discuss continuous management processes that should be in place. Although we will want to look at the configurations on occasion, the processes that maintain these configurations are the important things.

Super Simple Test

- What's wrong with this picture
 - User-facing ports should *not* have Layer 2 management protocols bound

```
09:15:49.751485 ARP, Request who-has 192.168.2.146 tell 192.168.2.251, length 46
09:15:50.148090 ARP, Request who-has 192.168.2.1 (00:03:e3:00:4f:0b) tell 192.168.2.5, length 28
09:15:50.149238 ARP, Reply 192.168.2.1 is-at 00:03:e3:00:4f:0b, length 46
09:15:50.751525 ARP, Request who-has 192.168.2.146 tell 192.168.2.251, length 46
09:15:50.932991 STP 802.1d, Config, Flags [none], bridge-id 8000.00:04:c1:c1:a2:c0.800d, length 43
09:15:51.756654 ARP, Request who-has 192.168.2.146 tell 192.168.2.251, length 46
09:15:52.755846 ARP, Request who-has 192.168.2.146 tell 192.168.2.251, length 46
09:15:52.933497 STP 802.1d, Config, Flags [none], bridge-id 8000.00:04:c1:c1:a2:c0.800d, length 43
09:15:53.756966 ARP, Request who-has 192.168.2.146 tell 192.168.2.251, length 46
09:15:54.761032 ARP, Request who-has 192.168.2.146 tell 192.168.2.251, length 46
09:15:54.933796 STP 802.1d, Config, Flags [none], bridge-id 8000.00:04:c1:c1:a2:c0.800d, length 43
09:15:55.761127 ARP, Request who-has 192.168.2.146 tell 192.168.2.251, length 46
```

We've said repeatedly that the Layer 2 management protocols, such as STP, MSTP, TRILL, SPD, VTP, and HSRP, should all be pruned from the user-facing ports. We've also taken a look at what some of that might look like in the switch configuration. Is there an easy way to validate this without having to learn everything there is to know about a specific switch? Yes.

An extremely simple test is to simply ask the administrator to fire up a sniffer on a machine connected to a typical user port. Let the sniffer run for a few minutes. You will see ARP broadcasts, Windows broadcasts, and other sorts of typical, normal network "stuff." However, if you see Layer 2 management packets, you know the switch has at least that protocol configured and enabled on that port! This would fail the validation test.

In this slide, we've given you an example of this. Although an auditor does not have to be a network engineer or a packet ninja, he should at least be able to identify the protocols that are appearing. Doing so simply requires a little bit of time and a search engine. In this case, we can see 802.1d present, which is Spanning Tree Protocol (STP). Clearly, this switch port is not properly configured.

Already Covered

- Remember, we are building throughout this section
- We will not repeat what was covered:
 - Patching
 - Potential legacy status
 - Unneeded services
 - Device management

Please remember what was said earlier in the book. We will strive to cover any particular issue only one time, even though it may apply to a later technology. For example, it is critical that we verify that our routers and firewalls are also securely managed, well patched, have clocks synchronized, have logging enabled, have centralized logging, and so on.

In this section, therefore, we assume that these issues are all addressed in the audit as well. In fact, you may see things related to these matters come up in the lab exercises. Of course, if you have any questions about how those previously covered issues apply to the technology at hand, you can always ask. If you aren't at a live conference, remember that I'm always happy to hear from you and will strive to help you out with any questions if you simply email me.

August 10, 2021

Audit Items

- Securely managed
- Patched and up to date
- Deployment provides DiD
- ACLs are correct according to CC
- Standards applied correctly
- Periodic validation completed



So then, our audit program and interview would include questions that dig into how the device is managed, regardless of whether it is patched, and other items are already covered.

Moving on to items that have not yet been dealt with, we are interested in examining the topology of the network—in particular, the placement of the firewall or router with filtering controls—in the context of the business information flows. What we're looking to do is identify that the principle of Defense-in-Depth (DiD) has, in fact, been applied. Remember from our discussions in Section 1 that this does not just mean that there should be a lot of layers. Instead, it means that the layers are architected in such a way as to provide for protection, the ability to detect threats, and the potential ability to react to those threats.

For this to be effective, though, we must also verify that the access control lists (ACLs) that are in place make sense for this business and the security requirements of the organization. To know this, we have to understand how those high-level standards would be expressed in the particular technology, like a firewall. Each ACL would also have to be examined to verify that there is authorization for that rule or group of rules according to the Change Control (CC) system in the organization.

An important aspect of managing routers and firewall rules well is good documentation. Not only are we verifying that the change control for an ACL exists, but we would also like to verify that there is adequate documentation right in the ruleset. Regardless of the firewall or router we use, all of them allow the administrator to attach comments or notes to rules and to group rules. This is true even if the configuration is a simple text file like we find in a Cisco PIX or ASA device!

Although ensuring that the proper rules are in the device, it is equally important to actually validate that the device performs as expected. Everything defined should pass according to the security requirements. We should prove that these things do pass and that nothing else can.

Before We Start

- To provide Defense-in-Depth, we look for multiple controls:
 - Prevent, Detect, and React
 - Organizations tend to be “Protect”-focused
- We also want to verify proper placement in relation to information

So how do we know if Defense-in-Depth is accounted for? How do we know we have protect, detect, and react capabilities? How do we know if these systems are deployed in the correct places?

To answer these questions, we have to look at the network topology. Before we do so, however, we have to know what the topology *ought* to look like.

Here is an important point: Network administrators are great at building networks. When they build networks, however, they are usually concerned only with building the roads. They ask about traffic volume and where exits should be so that they know how big to make the pipes and where to put the switches and routers. They almost never, however, ask what the vehicles, or packets, will carry! Without this knowledge, it is easy to create a network that works wonderfully but fails to account for the security requirements that the organization truly needs to secure itself.

Information Flow

- How information gets from here to there, and why:
 - Enables us to identify control points
 - Naturally reveals security zones
- Compare information flow to what the physical diagram looks like

What we're talking about is information flow. If you work in the Payment Card Industry (PCI) world, you may be somewhat familiar with this idea. As a result of attending our classes, the PCI folks ended up adding a requirement for an Information Flow Diagram as a part of a PCI assessment audit. What is this, and how do you draw one? We're about to do one together.

First, an Information Flow Diagram simply seeks to illustrate where information resides, where it moves to, and how. We are going to do this with an Information Technology and Security lens, but you can create exactly the same kind of diagram for information that is maintained only in physical form. This type of approach forces us to examine data movement. Data movement allows us to identify security zones and tends to indicate where control points ought to be.

After this diagram has been created, we can compare the requirements that we find with the actual physical topology that exists. It is our hope that we find that the physical topology has all the things it needs to meet the requirements that the Information Flow Diagram indicates are necessary.

Requirements

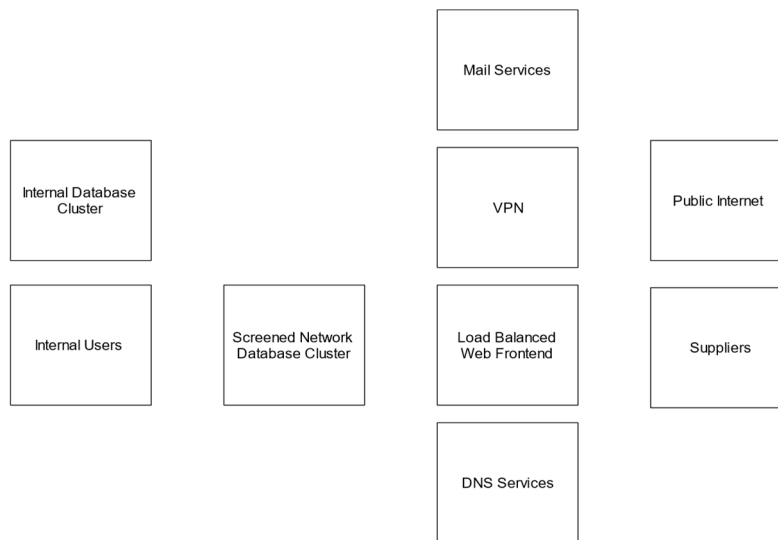
- Example information flow requirements:
 - VPN access to allow suppliers to access private website/database
 - Public web farm
 - Clustered database supporting web farm
 - Synchronization with internal database cluster
 - Mail and DNS services
 - Internal users able to browse websites and send email

Let's work through an example. In our example organization, we have a fairly simple network. The network provides the following access/services:

- Suppliers can connect into a private web cluster via a VPN
- The public can access our web cluster
- There is a database cluster supporting the web farm for both suppliers and the public
- The database supporting the web cluster synchronizes periodically with an internal database cluster
- We offer public email and DNS services
- Our internal users should be able to access our own website, email, and the internet for web browsing

With these requirements documented, we next turn our attention to drawing some simple diagrams to show where data is, how it moves, and which paths communication must take.

Lay Out the Functional Units

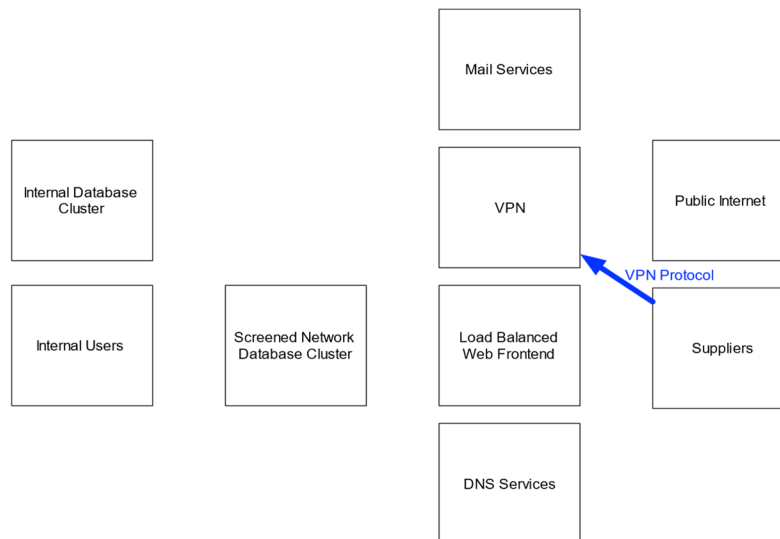


To begin with, we simply draw some boxes to represent logical "objects" of the different components or entities in our description. For example, we described a web cluster and a database cluster, but we have used individual boxes to represent these. Frankly, we don't care if there is one web server or a thousand inside of that cluster; the information flows are the same for the purposes of documentation.

When laying out the diagram, you may find it useful to use some sort of system. For example, in this case, we have chosen to lay things out from the point of view of likely sensitivity or confidentiality. Likely, the internal users and database are more sensitive than the database cluster that supports the website. Similarly, that database cluster is likely more sensitive than the web server it supports, and that web server and the other public services are more sensitive than the public internet.

At this point, don't worry about getting things "just right." Laying things out in this way is only done to limit the amount of erasing and moving that we might have to do later. In the end, it may turn out that we don't quite have things in the correct locations, but we can always move them if necessary.

VPN Connections from Suppliers

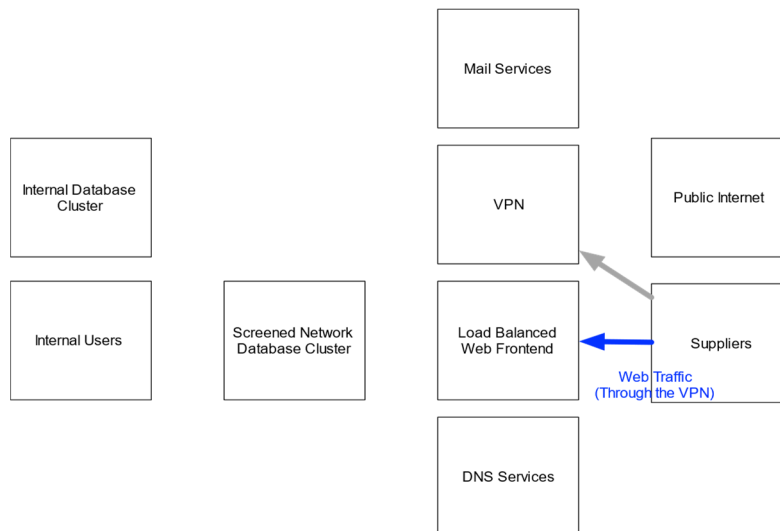


With the basic functional units identified, we now set out to begin documenting the information flows. We begin with the VPN, in this case. There's no specific reason we've chosen it other than the fact that it happens to be first on the list of required information flows.

We simply draw an arrow showing how information moves. We've also attached a label to it, indicating how the communication occurs. Notice that we are not defining the ports, the vendor, the encryption mechanism, etc. None of that matters at this high a level. We are simply documenting that this is a required information flow.

August 10, 2021

Suppliers' Access to the Web Frontend (Over the VPN)

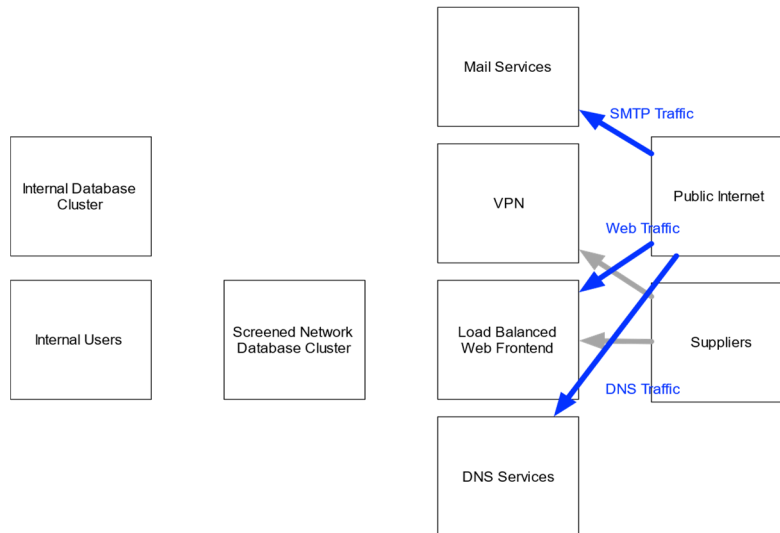


Next, we add in the next information flow. In this diagram, it would probably be best to draw the arrow in such a way as to make it apparent that the connection to the web server is occurring through the VPN, but we know that the books will ultimately be printed in black and white, so trying to embed one arrow inside of another will likely not print well!

In any event, simply document the information flows as they are required. There's nothing complex here at all!

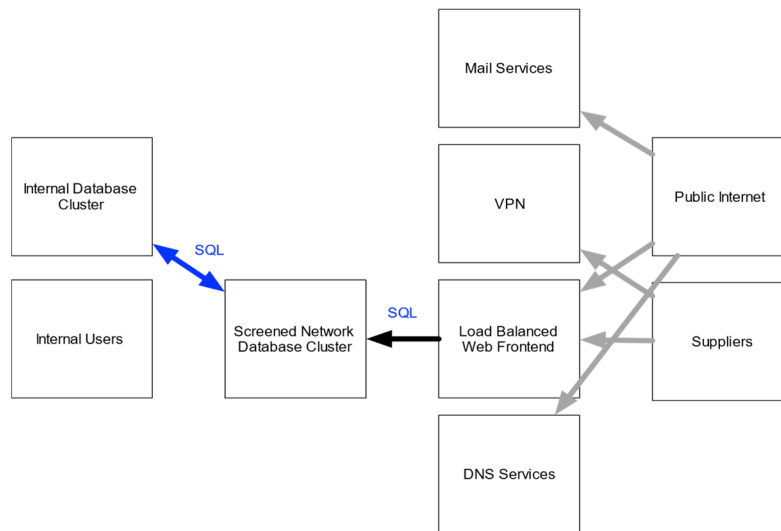
August 10, 2021

Internet Access to Public Services



Let's move things along more briskly. In this slide you can see that we have added information flows describing public access to our public services. Again, the ports are not important, only the kind of information that passes over the connections (in this case, web traffic, mail traffic, and DNS traffic). For the purposes of an Information Flow Diagram, there is no need to distinguish between port 80 web traffic and port 443 traffic. Those details are technical implementation details that will become important later when we are reviewing our network controls.

Web Cluster to Database / Database Synchronization

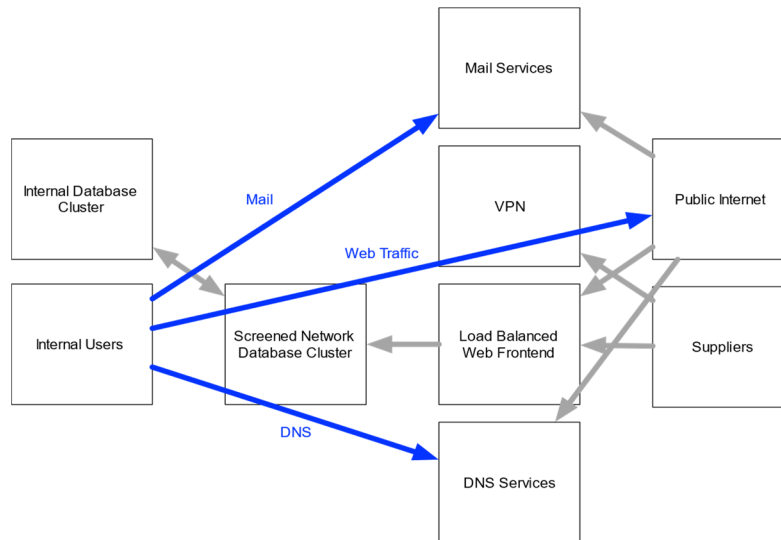


With all of our forward-facing information flows diagrammed, we now find ourselves examining the communication between the web server and the database cluster. According to the business process owners, the suppliers never actually communicate directly with the database cluster. Instead, all of their communication occurs through the web application frontend.

That database cluster, however, does replicate data to and from the internal database cluster. Similarly, the internal database cluster will occasionally initiate communication with the screened database cluster. To indicate this, notice that we have used a double-headed arrow. In the context of our diagram, we are using the directionality of the arrow to indicate who initiates communications.

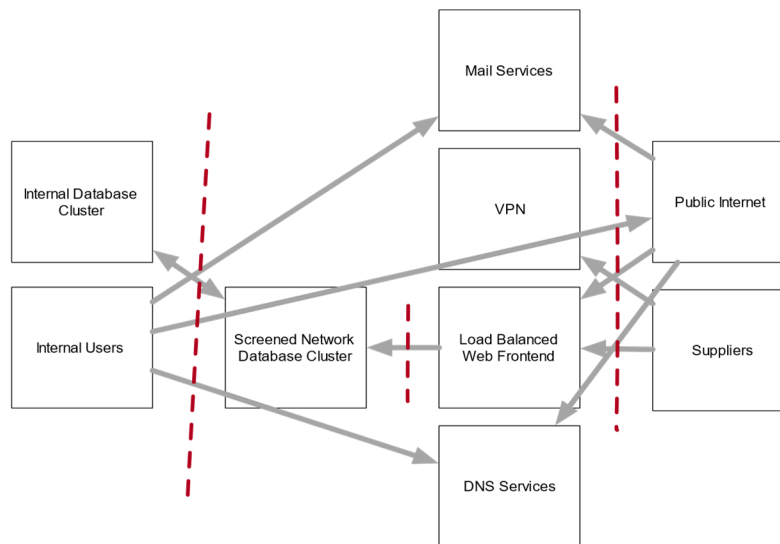
August 10, 2021

Internal Users Accessing the Internet / DNS / Mail (I)



The last set of information flows that we require are those related to our internal users. These users must be able to browse the internet using web services. They also require the ability to send and receive email through our publicly accessible mail server. Also, in order to accommodate their web activities, they use our organization's DNS server to perform DNS lookups for public sites.

Internal Users Accessing the Internet / DNS / Mail (2)

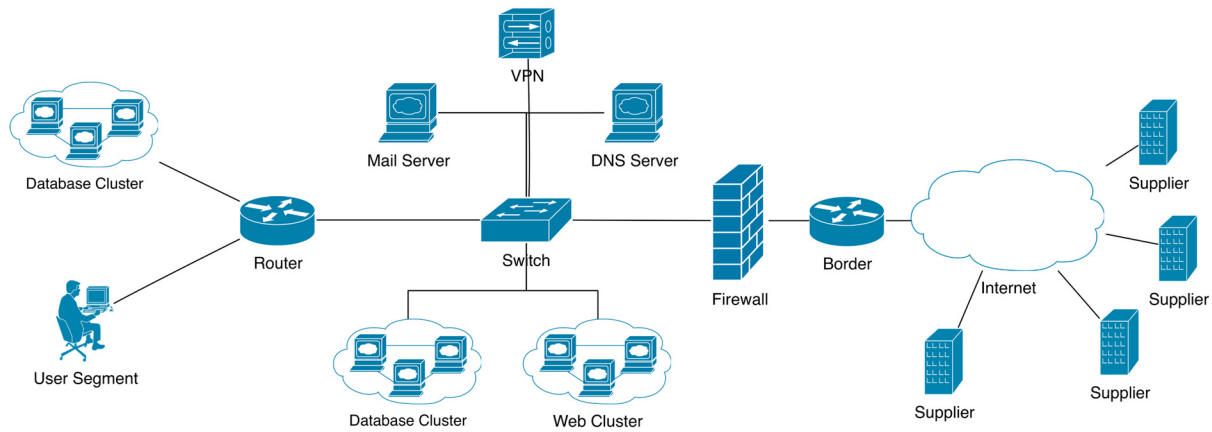


Now that we have diagrammed all of our required information flows, the next step is to identify where controls should be located. To do this, simply draw lines between systems based on the sensitivity of the information that they contain or the trust that your organization has for them. For example, you might feel that both the public web frontend and the screened network database cluster are at the same trust level; they are both in a hazardous location network-wise. However, most would agree that the database cluster likely has data that is more sensitive than the web cluster. To indicate this difference in trust or sensitivity, we have drawn a line across that information flow. The same is true of the publicly exposed services and the public internet and our suppliers, as well as the internal users and database and the screened network.

Every place where one of these lines crosses an information flow is an indication that there should be a network control in place that restricts the flow to only what is required. With this information in hand, we are ready to look at the network diagram!

August 10, 2021

How about This Topology?

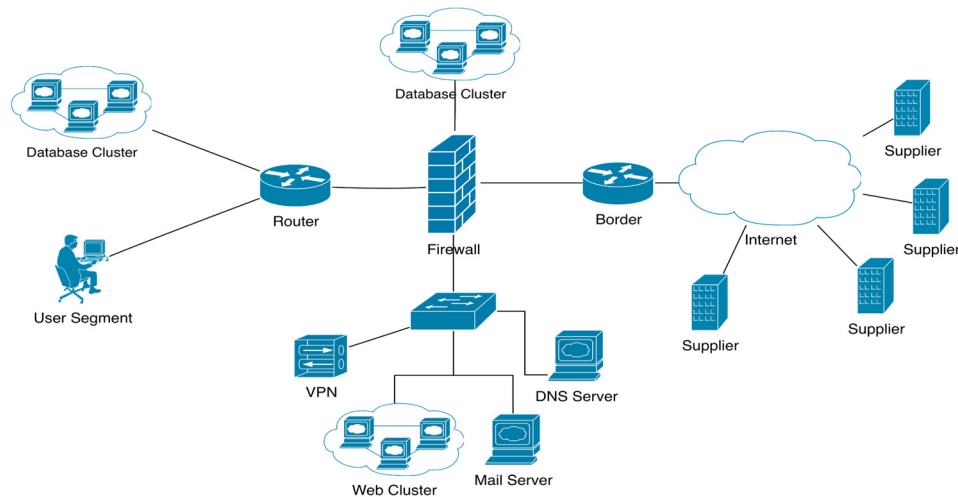


Let's compare the requirements that we have identified to the actual physical topology. In the context of the Information Flow Diagram we just created, consider the logical topology shown in the slide. The question that we have is, "Is this topology a reasonable approach to meeting all the Information Security requirements for the information flows defined?"

In this case, the answer is, "Not really." Although it is a nice diagram and there is a firewall, it does not appear that the firewall can adequately control the information flows. Why not? Notice that the VPN is sitting behind the firewall. Although this offers the VPN protection, it means that the firewall has no opportunity to limit information flows and data access after the data reaches the VPN endpoint. Because the data is encrypted, the firewall will simply allow all the VPN data over IPsec to pass. Additionally, there are no controls between the various systems in the screened network and the database cluster. Finally, unless there are ACLs on the internal router, there are also no controls between these publicly exposed systems and the internal networks.

The next question is, "Could we make a minor adjustment or two that would fix this?"

Can We Fix It?



The answer is, of course we can fix it! However, we'd like to suggest extreme caution during this process.

When you first introduce the purpose of the process we have just worked through, network administrators and engineers may feel as though you are getting ready to re-engineer their network. Please assure them that this is not the case! In fact, it would be good to introduce the process something like this:

"What we'd like to do now is validate that the network topology that you have in place fulfills all the operational security requirements. Now, I'm sure that it does because your network seems to be working. Still, we want to use this formal process so that we can document that."

What we said here we mean sincerely. Even if it turns out that the topology is not ideal, it certainly must be close if we can conduct business. Looking at the diagram in the slide, you can see that we've just made one small adjustment to the overall design, and it now enables us to meet all the information flow requirements because we now have a control at all trust borders. This enables us to enforce the information flow requirements properly.

Looking at our adjustment, you might think that this requires the purchase of a brand-new firewall. Is that actually true, though? Isn't it possible that we have simply added interfaces or networks to the existing firewall? Although that does make this firewall more complicated, it is absolutely possible to meet the information flow requirements in this way.

Firewalls

- Primary mission:
 - Keep bad people out
- Three primary types:
 - Packet filters
 - Stateful filters
 - Proxies
- NextGen firewalls are completely different

What, exactly, is a network control? Well, if we're talking about a protective or preventative control, then this is most likely a firewall or possibly a router with ACLs. Obviously, we would prefer to use firewalls as controls, but often the cost involved when considered with the placement makes a router a reasonable choice, especially when dealing with interior networks.

Firewalls, however, are the types of systems that should be in the perimeter. Not all firewalls are created equal. In fact, there are three primary types of firewalls that are in wide use today. There is another type of firewall that has been emerging over the last few years, but we will deal with these next-generation firewalls a little bit later.

Generally speaking, we have organized the list of firewall types in the slide according to cost. Packet filters tend to be the cheapest; proxies tend to be the most expensive. They are also organized in terms of required horsepower. Because packet filtering is very, very fast, the hardware requirements are much lower. Because proxying requires far more processing, for a slower process, much faster hardware is required. These are also organized from what are considered to be the easiest to bypass (or least secure) up to the most difficult to bypass (most secure).

Don't misunderstand. This doesn't mean that packet filters can't ever be secure. If I asked you if you'd prefer to be in a maximum-security prison or a minimum-security prison, you would pick the minimum-security prison. This doesn't mean that it's not a prison; it just means that it's less "secure" overall—easier to escape from, easier to smuggle stuff into. Let's dig into the differences between these types of devices.

Static Packet Filtering

- Looks only at *this* packet
- There is no context... it's derived
- Easy to fool:
 - Looks for keys like ACK packets
 - How hard is it to craft an ACK

Packet filters are not sophisticated devices. They make decisions on whether a particular packet may pass by comparing that packet, and only that packet, to the rules or ACLs that are in place. They have no ability to apply “context” to the packet by considering what has already happened between the hosts involved in the communication. This behavior is what makes bypassing them, in many cases, trivial.

Because a packet filtering firewall looks at only a single packet, this makes it simple to come up with strategies to bypass the configured rules. To combat this, most vendors provide extra “features” that try to simulate a more stateful approach. For example, Cisco devices that are operating as packet filters can be configured to permit only “established” packets.

Although many devices identify an established packet by looking for the ACK bit to be enabled, Cisco approaches this differently. Cisco looks at the problem and says, “The only time that the SYN bit is set alone is during the initial three-way handshake.” Because of this view, Cisco products using the “established” keyword will pass any packet as long as the SYN bit is not set alone.

Regardless of whether we are checking for SYN by itself or to see if ACK is enabled, how difficult is this to bypass? Are there tools that can be used to generate packets that can be used to trick this device?

Crafting an ACK

- Turns out it's simple

```
sh-3.2# /usr/local/sbin/hping2 -A -s 13229 -p 80 10.128.128.128
HPING 10.128.128.128 (en0 10.128.128.128): A set, 40 headers + 0 data bytes
len=46 ip=10.128.128.128 ttl=64 DF id=0 sport=80 flags=R seq=0 win=0 rtt=2.8 ms
DUPI len=46 ip=10.128.128.128 ttl=64 DF id=0 sport=80 flags=SA seq=0 win=5840 rt
len=46 ip=10.128.128.128 ttl=64 DF id=0 sport=80 flags=R seq=1 win=0 rtt=2.8 ms
DUPI len=46 ip=10.128.128.128 ttl=64 DF id=0 sport=80 flags=SA seq=1 win=5840 rt
len=46 ip=10.128.128.128 ttl=64 DF id=0 sport=80 flags=R seq=2 win=0 rtt=1.1 ms
```

- Easily bypasses packet filters:
 - Cisco “Established” keyword, for example

Of course, there are tools that can generate packets!

In this slide, we have an example command line with the output of a tool named HPing. This tool is designed not only as a packet-crafting tool but also as a network exploration and experimentation tool. From the command line, we can specify exactly what we want any particular packet to look like. How is this used for exploration and experimentation? Consider these questions:

- If you send a SYN packet to a closed port, what is the correct response from the server?
- If you send an ACK packet to an open port without having a session, what is the correct response from the server?
- If you send a SYN/FIN packet to an open port, what is the correct response from the server?
- If you send a packet with no code bits enabled to an open port, what is the expected response?
- If you send a packet with no code bits enabled to a closed port, what is the expected response?

Take a moment and write down what you think the answers to these would be. Then, we'll check to see how close you came!

Stateful Filtering

- Stateful filtering considers the context of the overall conversation
- Track initial connection attempts and match them up with late responses
- Requires more resources/processing time than static filtering

Stateful filtering is much more secure than static packet filtering. With stateful filtering, we are essentially telling the firewall what kinds of things are permitted as a baseline. For example, we tell the firewall to allow outbound connections to port 80 and 443 so that our users can browse to web pages. With this configured, the firewall will gladly allow these packets out, but it will also track these connections.

What this means is that when a packet comes into the firewall from the internet from port 80 and it claims to be a response (a SYN-ACK or an ACK), the firewall will not just allow it to pass. Instead, it will consult the information that it is tracking for known connections. If this packet does not fit into any of the known conversations, it will drop the packet.

Stateful firewalls are not all equal when it comes to their capability to track sessions. Of course, they will also have limitations on the total number of connections they can track. In fact, a denial of service against a stateful device is easy to accomplish. If we find a port that is permitted through and that is tracked for state, we can simply open millions and millions of connections relatively quickly. At some point, the device will become memory starved in its state tables. When this happens, the connections that have been quiescent for the longest period of time will be dropped out, effectively terminating those connections.

These firewalls are also not equal in terms of how smart they are about state. Some are simply matching the IP port pair while others are looking more deeply, checking to see if the TCP sequence numbers make sense for the specific question being filtered.

Researching ACLs: Cisco Example

- Requires ACLs:
 - Two main types:
 - **Standard:** Filter based on source only
 - **Extended:** Filter based on port, protocol, and so on
 - Type specified by number:
 - **Standard:** 1–99 or 1300–1999
 - **Extended:** 100–199 or 2000–2699
 - Or by name!

Let's look at some examples of configuring a router and a firewall with some rules. Let's do some research to see what sorts of features Cisco has for these things.

Our initial research reveals that Cisco routers and firewalls all support the creation of two types of ACLs: Standard ACLs and extended ACLs.

Standard ACLs are rudimentary. They permit you to only write a rule that permits or denies traffic based on the source address of the packet. Even though this is simple, it is still useful. Because the rules are so simple, they are processed very, very quickly. If we are trying to block or permit something that can be defined at an address level, this is a great way to do it.

Standard ACL rules are traditionally configured into numbered access lists. Although the administrator can arbitrarily select any number that would be in range for the correct kind of rule, the number ranges are important. Access lists numbered 1–99 or 1300–1999 define standard access lists *only*.

However, if an access control list has a number in the range 100–199 or 2000–2699, it would define an extended access list. Extended access control lists enable you to filter not only on the source address but also on the destination, the protocol, the port number, or other options, depending on the protocol in use. It also allows for the use of an “established” keyword for TCP, implementing the “Is SYN set alone?” test that we described previously. Extended ACLs are also the basis for the more advanced features like reflexive access lists, which we will discuss shortly.

Although numbered ACLs are widely used in the Cisco world, there is a far better way to go. We can name access control lists. Not only does this make auditing the device easier but it makes administration much, much easier!

ACL Reading Example

```
access-list 1 permit 128.226.0.0 0.0.255.255
access-list 1 deny any log

access-list 101 permit tcp 128.226.0.0/16 any eq 80
access-list 101 permit tcp 128.226.0.0/16 any eq 443
access-list 101 permit udp 128.226.0.0/16 any eq 53
access-list 101 deny ip any any log

ip access-list extended InboundRules
  permit tcp any 128.226.1.50 eq 80
  permit tcp any 128.226.0.0 0.0.255.255 established
  deny ip any any

interface Ethernet0
  nameif inside
  ip address 128.226.1.1 255.255.255.0
  ip access-group 101 in

interface Ethernet1
  nameif outside
  ip address 67.32.12.4
  ip access-group InboundRules in
```

Let's take a look at some actual ACLs and see if we can understand them. Begin by looking over the ACLs in this slide. How much can you figure out on your own? After that, start reading the following notes or participate in the class discussion.

First, we find a standard access control list being created. How do we know that it's a standard list? Because it's access list number 1. This also means that we are limited to filtering on the source address. In this case, we are permitting all the packets from a particular network to pass while everything else is denied and logged.

August 10, 2021

Research Aside: Wildcards

- Most devices use network masks:
 - Cisco can, too
 - Wildcards are the default
- Enables you to create a single rule to cover many cases:
 - Requires some network engineering knowledge

While examining the ACLs on the last slide, you likely noticed how addresses were written out. Within the Cisco rules language, there are some built-in shortcuts for commonly used network or host addresses. For instance, we can use “all” to define an address of 0.0.0.0 255.255.255.255. We can use host 192.168.1.1 to define 192.168.1.1 0.0.0.0. But what are those numbers that come after the address? In most networked systems, we would find a network mask in that location.

Cisco systems can certainly be configured to use network masks, but the default within access control lists is to use Cisco wildcards. Wildcards are simply the inverse of the network mask. In other words, if the network mask is 255.255.255.0, the wildcard that is equivalent would be 0.0.0.255.

Wildcards (and network masks for that matter) within firewall or router ACLs are used to allow us to create a single rule that covers a range of hosts. This is likely not a new idea for you. What might be new, and confusing, is how a competent network engineer might use these. He can create single, simple-looking rules that cover a wide number of hosts precisely.

Cisco Wildcards

- Rule of thumb:
 - Zeroes to the left, ones to the right
 - Anything else requires explanation
- Consider this:
 - permit 192.168.2.0 0.0.0.251
 - Which addresses will be permitted?
 - Why?

When looking at access control lists on a firewall or router, Cisco or not, we are interested in seeing what is happening with the wildcards or network masks. Truly understanding them is not actually that difficult, but it requires that we can decode them into their binary equivalents. As you likely know, when you look at an IPv4 address, it is called a “dotted quad,” and when we refer to the individual values, we call them “octets.” Why octets? Because each value represents 1 byte in the 4-byte IP address.

The same is true of the network mask or wildcard. With Cisco wildcards, the IP address listed to the left is used as a template to compare inbound or outbound packets to. The wildcard is used to mask off pieces of that address that can be, essentially, ignored. If a binary value of zero is found in the corresponding bit field, that bit in the address of the packet being considered must match the “template” bit exactly. If the bit is a 1, however, then it is masked off, allowing the bit to be on or off and to still match.

That may sound complicated. Let’s boil it down to a rule of thumb. We would like to find that all the bits that are turned off are to the left of the wildcard and all the bits that are turned on are to the right. If the system uses network masks instead of this unusual Cisco notation, we would simply invert our rule: Ones to the left, zeroes to the right.

Using some other arrangement of bits is absolutely legal but requires much greater expertise on the part of the administrator. If he has done something else, he should easily and competently explain it to us, perhaps even standing up in front of a whiteboard and showing how the bits all line up.

Consider the example in the slide. With a wildcard of 0.0.0.251, the binary values would be 00000000.00000000.00000000.11111011. What would be the effect of this wildcard? It would allow us to write a single rule that would cover a large number of hosts. Awesomely powerful, but full of potential for error!

Wildcard Walkthrough

- 1s are wild, 0s are “requirements”

192.168.2.0	0.0.0.251
0	0
0	0
0	251
00000000	00000000
00000000	11111011

Let's just take this apart. If we were to find this kind of wildcard in use during an audit, one of the things that we would expect is that the administrator could explain it. We would listen to him for a few seconds and then, likely, interrupt him and ask him to use a whiteboard to walk us through it. What would that look like?

We would expect to hear facts, such as “Everyplace in the wildcard that there's a one, it will match anything, but anywhere that there's a zero, it must match the rule exactly.” What does this mean? What ones? What zeroes?

Look at the slide. Notice that we've taken the wildcard and split it into its constituent parts. Underneath that we've further analyzed it, representing each value in the wildcard as the bits in a byte. Here are the zeroes and ones.

In all the bit positions in which there is a zero, a packet being checked must precisely match the bits in the left portion of the rule (the 192.168.2.0 piece). Anywhere that a one appears, the bit can have any value. Let's look at a few examples to better understand what is meant.

Example (I)

- Template: 192.168.2.0
 - Wildcard mask: 0.0.0.251
 - Packet from: 192.168.2.18
- | | | |
|------------|---|-------------------|
| - Mask:251 | = | 11111011 |
| - Template | = | 00000 <u>0</u> 00 |
| - 18 | = | 00010 <u>0</u> 10 |
- Packet permitted

Let's take the case of a packet coming from host 192.168.2.18. We can see that the 192.168.2 part of that precisely matches the left side of the original rule, so we're going to assume that this piece matches. Let's turn our attention to the final octet.

When we convert the decimal value 18 to binary, we arrive at 00010010. Notice that the "template" (or left side of the original rule) is a zero, which is eight 0 bits in binary. In addition, we have the 251 from the wildcard shown in binary. We can see that the position in which the zero appears in the wildcard has been marked in bold. Recall that this means that the packet being considered and the template must match in this position *exactly*.

When we look at this position in the other 2 bytes, we find that the bytes *do* match. As a result, this packet will match the rule and be permitted.

It is important to understand that what allows this packet through is *not* the fact that the wildcard has a zero and the packet has a zero. What allows the match is that the template and the packet *have the same bit value in the same position where the zero appears*.

Example (2)

- Packet from: 192.168.2.24
 - Mask:251 = 11111011
 - Template = 00000000
 - 24 = 00011000
- Packet permitted

Consider another example. Another packet has arrived, this time from 192.168.2.24.

The address is again dissected. This time the byte comes out as 00011000 in binary. Notice that, again, the third bit from the right is zero, matching the zero from the template. This packet is also permitted.

August 10, 2021

Last Example

- Packet from: 192.168.2.28
 - Mask:255 = 11111011
 - Template = 00000000
 - 28 = 00011100
- Packet *not* permitted
 - Why? Because template was:
 - 192.168.2.0
 - What if it had been:
 - 192.168.2.4

As a final example, consider the address 192.168.2.28. When we represent 28 in binary, we get 00011100. In contrast to the other packets seen so far, this packet will *not* be permitted. Notice that the third bit from the right is now a 1. Because it is a 1 and the template has a 0 in this position *and* the wildcard marks this position as an exact match, the packet will be rejected.

But what if the *template* had been 192.168.2.4? In that case, the template would have been 00000100. In this case, with *exactly* the same wildcard, the packet would be *permitted*! Can you see why?

If you said, “Because the template has that bit on and the wildcard requires that the packet have *exactly the same bit* turned on,” then you are correct! Congratulations!

If you don’t immediately see that this is the case, see if you can work it out on paper. Ultimately, however, bit masking and wildcards could be viewed as somewhat out of the realm of an auditor; it is not critical that you can specify or analyze these personally. What about the administrator at the whiteboard? It should be abundantly clear to us that he absolutely understands how they work and what they do and do not permit. If he can’t explain this, then perhaps he should not be using them!

Cisco Example: Moving from Packet Filter to Stateful Firewall

- Standard and extended ACLs are typically static packet filtering
 - Lowest form of protection, easily bypassed
- Cisco also supports "reflexive" rules
 - Allows complementary rules to work in concert for stateful filtering

```
ip access-list extended OutboundRules
  permit tcp 128.226.0.0/16 any eq 443 reflect sslTable
  permit udp 128.226.0.0/16 any eq 53
  deny ip any any log

ip access-list extended InboundStateful
  evaluate httpTable
  evaluate sslTable
```

Reflexive rules are Cisco's most basic answer to creating stateful rules in a basic firewall or router device. Creating them requires that you have two rules that complement each other. If you look at the two access lists defined in this slide, you will see that they are both extended access lists. In this particular case, you can see that we have used named access lists (OutboundRules and InboundStateful). Functionally, there is absolutely no difference between numbered ACLs and named ACLs, though we would strongly recommend that, if your organization uses Cisco devices, your internal standards should require the use of named access lists. They vastly simplify the work of future administrators and the auditors, allowing them to quickly identify the general function of an access list, rather than having to puzzle it out by reading every rule and comparing that to which interface the access list is applied to. Additionally, if you wish to use reflexive ACLs for stateful filtering, you *must* use named extended lists.

Note that the first rule shown in access list OutboundRules is a permit rule for outbound packets going to destination port 443. This will act as a packet filtering rule, permitting these packets. On the end of that line, however, we have added the "reflect" option along with a name (in this case, "sslTable"). This tells the network device that, whenever an outbound packet matches, the device should add information about that connection (source, destination, ports, etc.) to an internal "state table" that resides in memory. Now, when inbound packets arrive in the device, they will be evaluated against that state table. If and only if a packet attempting to enter the network is in that state table (or the httpTable), that packet will be permitted. Otherwise, it will be denied.

You can likely see why this is much greater protection. We are no longer relying simply on the state of the TCP flags and the port numbers; the device is actually keeping track, in a sense, of previous packets that have been seen and checking to see if this packet fits into that context. Cisco provides other features that go even further, as do other manufacturers.

Which Way?

- Firewall rules are applied inbound and outbound:
 - Network perspective
- Router ACLs are applied In and Out:
 - Router perspective
 - Better to apply them all as “In” rules:
 - Do not insist

Many firewall systems, especially with graphical user interfaces (GUIs), automatically apply any rules that you create to an interface. This is also generally true on routers where a graphical interface is used to create ACLs attached to interfaces.

Firewalls and routers that make use of text-based configurations, however, enable you to create context-free rules. What we mean by this is that the rule can be created without having to define the interface to which it is applied. In fact, this is how rules are intended to be created! The process that an administrator would follow would be to write some rules as a part of an access control list and then, after the rules are written, to apply them to an interface.

As crazy as it may sound, you will on occasion find that an administrator has written perfectly valid rules that completely account for all the organizational requirements... but he has forgotten to apply them to any of the interfaces on the device. Sometimes, this is an oversight; sometimes it is a training or knowledge issue. This is actually one of the reasons we take the time to learn how to perform a firewall validation.

Most firewalls have a well-developed notion of “inside” and “outside.” This allows us to write and apply rules from the network perspective.

Routers, however, do not. They do not actually know which interface the enterprise is connected to. Instead, all the rules are applied from the point of view of the router. When a packet enters the router, whether it has entered from the internet or from the enterprise, that packet is “inbound.” Therefore, when rules are applied, we usually (and certainly on Cisco routers) have to specify whether the access list is applied “in” or “out.” When examining ACLs of this type, we would prefer to find that they are all applied as “in” rules. This is the most efficient way to process the rules and packets, yet we would not insist on this as a requirement. There can be good reasons for mixing “ins” and “outs.”

Rules

- For firewalls and routers:
 - Access to the device itself
 - Passing private addresses
 - Passing internal addresses
 - Allow only protocols/ports required for business needs:
 - Even on internal devices

So, what sorts of ACLs would we expect to find? Certainly, based on our information flow requirements discussion, we would expect to find ACLs that achieve all the information flow control requirements that have been identified. Even internal routers can benefit from this approach. Remember our discussion of Defense-in-Depth. Routers are in a wonderful position to act as network controls to limit who can talk to what. For example, if we configured the router that feeds the server segment to allow only workstations to connect to the specific services that the servers offer, even if an extra service were running, it would be impossible to connect to it from a workstation segment.

The approach that we describe here implies that our organization is taking a “deny all by default” stance, not only for inbound traffic but also for outbound traffic. When we do so, especially if this has not been the approach of our organization in the past, there can be significant resistance. It is critical for the organization to understand that absolutely anything that is needed for business operations can be turned on. As long as someone can present some business case for Torrent downloads in the enterprise, for example, the security team will open up the firewall to permit that behavior. This is a far better approach than letting everything out and then trying to lock down the stuff that’s causing trouble.

In addition to these rules, we would like to find rules that limit who can talk to the router, switch, or firewall. We would expect to find that only administrators or the addresses on the network segment that administrators reside on can connect to the devices.

We would also want to see that the device is configured to prevent our internal addressing from leaking to the outside, whether those are private addresses or actual addresses. It is absolutely best practice to push all our outbound network traffic through a NAT (Network Address Translation) device, concealing the actual internal addresses and addressing scheme.

Routing Protections

- How are we handling routing
 - OSPF, EIGRP, RIP
 - How is it secured
- Additional routing protections
 - Do we accept or forward redirects
 - What are we doing with IPv6 router advertisements
 - What about IPv4 Source Routing
- Internal devices should be as securely configured as our edge devices!
 - Zero trust

Other issues that need to be examined are not ACL-driven. We want to understand both how dynamic routing is being accomplished within the enterprise (and through the perimeter) and how that dynamic routing configuration has been secured. For example, if we use Routing Information Protocol (RIP), it will be difficult to secure the routes. Why? Because this is simply a datagram-based broadcast protocol. Each router (and possibly firewall) in the environment with RIP configured will periodically (approximately every 30 seconds) broadcast a complete list of all the aggregate networks it knows how to reach with a cost measured in network hops. Neighboring routers or systems configured to listen for RIP broadcasts will use this information to populate routing tables.

Open Shortest Path First (OSPF) and Extended Interior Gateway Routing Protocol (EIGRP) achieve the same goals as RIP, but both of these support security features to prevent an untrusted host from injecting arbitrary routes or modifying existing routes. Just because these features exist, however, does not mean that they have been configured. We want to inquire to determine how our systems are actually configured.

We're also interested in both how these network devices react to packets that could be used to alter routing, and whether these devices allow these types of packets to be passed to endpoints. Let's consider a few of these packets.

ICMP Redirect messages are supposed to be generated by routing devices to inform a source host that there is a better or preferred path to reach a destination host. Regardless of static routing configurations, unless it has been configured not to, that source host will accept that redirect and actually update its routing table.

IPv6 handles network discovery and routing completely differently from IPv4. Neighbor discovery is used over ICMP to find neighbors, and IPv6 router advertisements are used for routes. If someone begins advertising an IPv6 route and we have IPv6 enabled, our system will prefer this route over any IPv4 route.

Review the Rules

- You, the administrator, and a change control person:
 - Start with rule #1
 - Administrator reads and explains
 - Ask questions based on requirements
 - Identify missing, duplicated, or otherwise incorrect configurations

Let's continue on to how we actually audit the rules on the router or firewall. We absolutely want to have a meeting with the administrator. We would also like to have someone who has change control authority with us at the meeting, especially if it is the first time that the device is being audited and validated. In almost every case, my experience has been that the first time looking at a device with any kind of substantial configuration there will be something overlooked, usually something not documented, that needs to be adjusted. Rather than waiting longer to fix things or manufacturing findings by writing the administrator up for repairing the configuration, it's easier to have someone with us who can approve any necessary changes.

Of course, we would have asked the administrator to provide a copy of the configuration to us well in advance of the meeting. When we ask for this configuration, we always remind the administrator to redact the password hashes out of the configuration file. We can verify that these appear to be configured correctly when we meet together. We don't want a copy of the hashes, though.

Having the configuration ahead of time gives us the opportunity to do research into any configuration options we don't recognize. We also have the time to map the rules out with the information flow requirements that we have already determined. Before we go to the meeting, we already know which rules we have questions about.

When the meeting starts, we will ask the administrator to begin at the top of the configuration and begin reading and explaining the configuration to us. This gives us an opportunity to assess his overall competence with the technology, but it also provides him an opportunity to find any errors. This is always more pleasant than us pointing them out!

As he reads, we're looking to verify that all requirements have been met and possibly identify any duplication, missing, or otherwise incorrect entries within the configuration.

Next-Generation Firewalls

- Traditional firewalls can perform “deep packet inspection”:
 - Slows them dramatically
- NG firewalls are application-aware:
 - Policies applied based on business
 - Rules applied to users/groups rather than IP addresses
 - Essentially, information flow controls

The new kids on the block in the world of firewalls are next-generation (NG) firewalls. These devices represent a substantial paradigm shift for network security. Interestingly, because of this adjusted approach, it has been our experience that management of these devices can be a steep learning curve for a seasoned firewall administrator. However, it is often easy to teach someone who is *not* a firewall administrator how to configure and manage these devices!

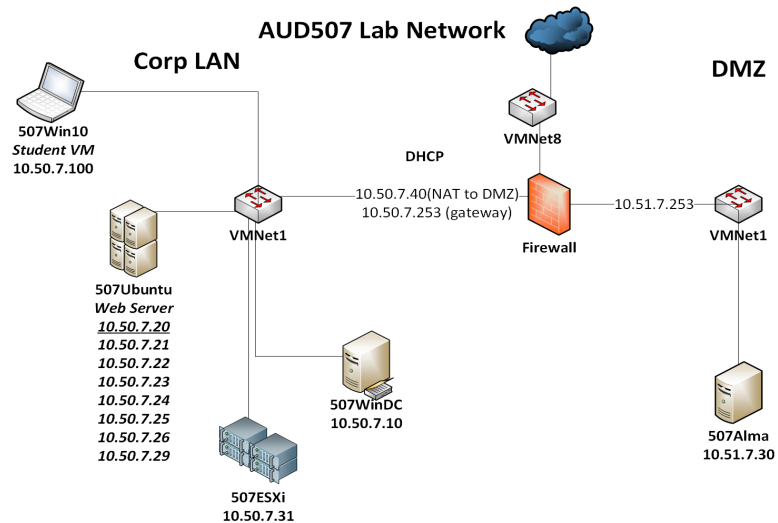
So, what changes? Many traditional stateful firewalls enable you to perform some degree of deep packet inspection. Deep packet inspection means that the device may have a rule that permits traffic on port 80, but recognizing that anything could potentially be on that port, the device can look into the packets to determine if it is actually web traffic. For instance, in a Cisco firewall or router, you can add the “inspect http” action to a permit rule to ask it to verify that only HTTP traffic is found on that port. If something else is there, it will block the traffic.

Next-generation firewalls are primarily deep packet inspectors. They are optimized to perform this deep analysis only one time, allowing them to operate at or very close to wire speed. Although you can create rules based on addresses and ports, the primary way of specifying rules is based on users, groups, applications, and data.

In a real sense, creating rules in a next-generation firewall is defining rules that match our information flow. It’s true that we are doing the same thing in traditional firewalls, but because these devices “understand” the content, it allows us to model these information flows much more closely. There is a downside, of course. If we use some of custom protocol or something that the device does not understand natively, we are essentially reverting to a classical packet filter or stateful device.

Probably the best-known example of a next-generation firewall on the market today is the Palo Alto series of firewall appliances.

Exercise 4.3 - Wireshark, Switch Configuration Symptoms and Device Configuration



This page intentionally left blank.

August 10, 2021

Course Roadmap

- Enterprise Audit Fundamentals; Discovery and Scanning Tools
- PowerShell, Windows System, and Domain Auditing
- Advanced UNIX Auditing and Monitoring
- **Auditing Private and Public Clouds, Containers, and Networks**
- Auditing Web Applications
- Audit Wars!

Section Four

1. Private Clouds
2. The Public Cloud
3. Containers
4. Networks and Firewalls
- 5. WIFI and VPNs**
6. Public Services

This page intentionally left blank.

August 10, 2021

Wireless “Ethernet”

- Ethernet is defined as CSMA/CD:
 - Wireless isn’t that. It’s CSMA/CA.
- Provides data encryption:
 - History of issues
 - Only good way to do it today:



WPA3 in Enterprise Mode

The first of these externally, or even publicly, accessible systems that we discuss is wireless technology. Wireless actually comes in a lot of shapes and sizes these days. You can take what we discuss now and extend it to other sorts of wireless systems that we have (Zigbee, for example, which is used in many building control systems).

Wireless, 802.11, is where we focus. Interestingly, what we call “Wireless Ethernet” does not actually conform to the standard that defines Ethernet. For something to be “Ethernet,” it must be Carrier Sense Multiple Access with Collision Detection (CSMA/CD). Wireless actually uses a different strategy. Because the shared medium is the air and because these different systems that are communicating may have interference from systems on different networks or even from entirely different communications systems, they cannot perform collision detection. Instead, they use collision avoidance strategies. These include Direct Sequence Spread Spectrum (DSSS), which is a technique for not only duplicating and spreading the bits in each message out over multiple frequencies but also switching from one frequency to another. This is the way some wireless analysis tools like InSSIDer show your wireless activity as covering multiple channels simultaneously.

802.11 wireless solutions have had a history of security issues. Recall the discussion that we had regarding time-based analysis. To properly defend a system, we have to understand what the protection is and then assume that the protection fails. In wireless, probably more so than with other systems that we have, this is an easy thing to do because there have been so many issues.

Before we dig into the history of wireless, let’s just come out and tell you the right way to deploy it today. The only acceptable deployment, from a security perspective, within an enterprise today would be WPA3 running in Enterprise Mode. We’ll define what this is shortly.

Wireless Encryption: WEP

- Started with 40-bit WEP:
 - RC-4: Digital implementation of a one-time pad
 - 64-bit key, 24 bits are a “random” IV:
 - 40 bits effective key length
 - Later changed to a 128-bit key:
 - 104 bits effective key length (still a 24-bit IV)

When 802.11 wireless technology was first created in the '90s, the working group developing the standard recognized that broadcasting data wirelessly could lead to a lot of confidentiality issues. For this reason, they created Wired Equivalent Privacy (WEP). Their stated goal was to make your wireless communication just as secure as your wired communication. Because there have been so many security issues, some security people say that these folks were 100% successful because your wired communication actually isn't that secure.

In any event, to provide this protection they needed to select an encryption algorithm. They wanted something that would be very, very fast and very secure. They also wanted something that would allow for variable payload sizes. This led them to choose RC-4.

RC-4 is a digital implementation of a one-time pad. A one-time pad describes a code system where two people communicating have code pads. Each sheet on the pad tells the individuals how to encode a message, essentially acting as a key. When one of the people sends a message, he uses the top sheet on the pad to encrypt his message and then he destroys the top sheet on the pad. His partner with the matching pad uses that same top sheet to decode the message and then destroys his sheet as well.

One-time pads provide Perfect Forward Secrecy (PFS). You may have seen this term attached to the description of your VPN. What it means is that even if an attacker could convince you to encrypt and send a known plaintext, allowing him to break that key, this would not give him insight into any previous or any future message. This is an important feature of a crypto-system.

Unfortunately, a one-time pad becomes easier to break the more often you use a key. WEP used a 64-bit key (later a 128-bit key), 24 bits of which were a supposedly random value. The other 40 bits were a preconfigured key. Can you see the problem? The more packets we send, the more likely we will repeat the key. In fact, if we send 2^{24} packets, we *must* repeat the key, allowing the key to be broken!

Wireless Encryption: WPA

- WPA:
 - Wi-Fi Protected Access
 - Created by Wi-Fi Alliance:
 - Had to fix it fast
 - Couldn't wait for IETF working group
 - RC-4
 - Adds TKIP
 - NOT A STANDARD

The discovery of the problem with WEP was made early. In fact, there are other weaknesses called key scheduling problems, where certain keys and certain initialization vectors allow the key to be recovered almost instantly. Because the technology was still new from the consumer point of view, this represented a tremendous risk for the technology vendors. If this was not fixed quickly, billions of dollars invested in bringing these products to market would be lost.

Because the standards committee would take years to solve the problem, the industry formed a group known as the Wi-Fi Alliance. This group came up with an approach to solving the problem that came to be called WPA: Wi-Fi Protected Access.

Since WPA was created by the industry and never ratified as a standard, you can sometimes run into issues in which some WPA systems cannot talk to others. Because this is not a standard, we strongly recommend that it not be used.

How did WPA “fix” things? In reality, WPA is essentially WEP with a few refinements. The most important change is that it introduces Temporary Key Integrity Protocol (TKIP). The primary problem with RC-4 in this context is that the key is being repeated and that, as soon as the key is repeated, the strength of the encryption is immediately compromised. TKIP then forces the key to change periodically.

The second adjustment is that the initial key is not used as-is. Instead, it is hashed with the SSID. This is intended to prevent a precomputation attack, because an attacker would have to know the SSID ahead of time to precompute his attack dictionary.

Wireless Encryption: WPA2

- WPA2:
 - RC-4 removed
 - AES is used
 - Like WPA, SSID is hashed against key to further protect security
 - PSK mode is vulnerable to attack: Less than 10 minutes for most networks
 - Enterprise Mode is required

WPA2 was ratified as a standard in 2004. It took a holistic approach to solving the security problems in Wi-Fi. Rather than just patching WEP, it completely removed RC-4 and replaced it with AES, which, it turns out, is not subject to the same sort of problems as RC-4 in this context because it is a block cipher rather than a stream cipher. In addition, like in WPA, the SSID is hashed against the key to provide some protection against precomputation attacks.

WPA2 on its own is not sufficient, however! It turns out that WPA2 configured with a pre-shared key (PSK) is easily compromised. As a specific example, if your SSID is any of the top 1,000 and your passphrase is any of the top 1,000,000, your WPA2 can typically be broken in under 10 minutes. For proof and a discussion of this, have a look at <https://u.aud507.com/2-5>

Of course, we could choose some other SSID and a more complex passphrase. This is a great idea, but it has limitations. One of the primary limiting factors is smart devices. Imagine trying to get your employees to key a 27-character random passphrase with special characters into their smartphones!

WPA2 Enterprise Mode

- Enterprise Mode leverages 802.1X:
 - 802.1X is NOT a networking protocol
 - Authentication protocol
 - Does not deal with encryption
 - Typically used to securely prove identity and issue keys
 - Certificates are the most common way to use this in wireless

Therefore, the best way to protect these networks today is Enterprise Mode. Enterprise Mode, typically implemented with certificates, leverages 802.1X authentication to mutually authenticate the device to the access point and the access point to the device. This also eliminates the need for the pre-shared key, which can be attacked. Instead, the certificate-based authentication provides the mechanism for obtaining the current group key that is securing the wireless network.

It is important to realize that 802.1X has nothing to do with wireless directly. The 802 standards all define technical standards, but the numbering to the right of the dot tells you what it's for. 802.1q, for example, is a VLAN tagging protocol. 802.2 is an Ethernet encapsulation standard. 802.3 is another, and more commonly used, Ethernet encapsulation standard. 802.11 defines wireless Ethernet standards. 802.1X is an authentication protocol.

802.1X does not deal with encryption. It is only for authentication and key exchange.

Wireless Encryption: WPA3

- Pre-shared key mode replaced with simultaneous authentication of equals (SAE)
 - Resistant to offline cracking
 - Adds forward secrecy protections
- Enterprise Mode
 - Requires 192-bit or higher keys during authentication - NSA Suite B
 - AES-192 with 256-bit Galois/Counter Mode Protocol (GCMP-256)
 - New key management protocols
 - Protection for management frames

At the time of this writing, WPA3 has just been released as a new standard and has not been widely deployed. This new standard (the first upgrade to WPA in about 15 years), introduces SAE (simultaneous authentication of equals) as a replacement for pre-shared key mode. SAE is supposed to bring better encryption and key management to personal Wi-Fi networks and allow for better security even when users select weak network keys. The SAE protocols were originally developed for use in Wi-Fi mesh networks.

Enterprise Mode uses a whole new suite of cryptographic protections, based on the NSA's Commercial National Security Algorithm Suite B. AES is now implemented with a 192-bit key and GCMP-256. The key creation and management protocols have been overhauled also. WPA3 enterprise also requires protections for Wi-Fi management frames.

More information about this new standard is available at:

<https://www.wi-fi.org/discover-wi-fi/security>

Wireless Intrusion Prevention

- Can be expensive
- Sensors deployed strategically
- Educate system about floor plan
- Configure locations of sensors
- New host:
 - Where it is (physically)
 - Shoot it down?



Wireless intrusion prevention systems (WIPS) can be very effective at maintain security on your WiFi networks.

These systems tend to be somewhat costly. Examples of these kinds of solutions are the Cisco Adaptive Wireless IPS, Aruba Wireless Intrusion Protection, and others. These solutions will cost you at least several thousand dollars, in addition to the sensors that need to be deployed.

Typically, the way that these systems work is that you license the WIPS software and deploy a number of sensors and antennas throughout your physical premises. After these are deployed with proper overlap, a survey-type tool enables you to identify where they are physically and enables you to import a diagram of your floor plan.

After this is all configured properly, the system can alert you to any number of things. For example, perhaps you have it configured to alert you when a new access point appears. Not only does it tell you about that device, but if you have taken the time to configure everything correctly, it pinpoints exactly where that device is within your physical building to within a few feet. This is actually incredibly valuable, especially in a densely populated area. This enables you to quickly distinguish signals that are originating from within your perimeter as opposed to signals that are outside of your perimeter.

In addition to identifying rogue hosts and access points, these systems typically have active response capabilities, enabling you to begin sending deauthentication and other types of wireless attacks at the unauthorized system, preventing it from connecting to anything.

VPNs (I)

- Ideally, separate from firewall:
 - Firewall: Keep people out
 - VPN: Let people in:
 - Separation of duties issue
 - Economy of mechanism issue
- Password resets on VPN
 - Typically the help desk



This brings us to the topic of VPNs (Virtual Private Networks). Many enterprises use a VPN that is included as a feature or option in the firewall platform. The primary reason that this configuration is used is cost. If the VPN is already included in the firewall, why should we buy another device?

Although this is a valid argument, let's take a look at this problem from a security and risk perspective rather than a cost perspective. In the end, your enterprise may choose to use an integrated firewall/VPN solution. Still, we should understand why this might not be an ideal configuration.

First, consider the mission of the firewall. The primary task that it has is to keep bad people out. However, the mission of the VPN is to let people in. This certainly seems like a conflict of interest. The principle of separation of duties would seem to indicate that we should not join these services.

Another principle that comes into play is economy of mechanism. Firewalls are complex systems. VPNs are also complex systems in their own right. Gluing VPN technology to a firewall seems to make a *more* complex system. Because simple systems tend to be more secure, it would be preferable to keep these systems separated.

A more convincing risk-based consideration of this exists, too. Our firewall is typically managed by firewall administrators. Our first reaction would be that the same is true of the VPN. But is it actually? Unless we integrate our VPN authentication into our domain (which is a good thing!), who do VPN users call to reset passwords? I can guarantee you that it's not your firewall administrator. Those users are calling the help desk.

Here's the thing: Firewalls do not have a well-developed sense of user rights. You either are an administrator or you are not. This means that if the help desk is resetting passwords on the firewall/VPN device, they are likely administrators on the firewall! This feels like a very risky thing to do!

VPNs (2)

- Firewall should be final authority:
 - Provides an additional layer
 - Guards against error
- Best to own remote hardware
 - How else can I require settings
- If not, a remote desktop solution is best:
 - Window into network, not a tunnel

Instead, we would say that the firewall should be the final arbiter of what passes through our perimeter. This means that, even though the firewall would be protecting the VPN, the decrypted VPN data that is actually trying to enter the organization should be forced to pass through the firewall. This allows for separation of duties and keeps a misconfigured VPN permit rule from allowing access to resources that should not be accessed from outside of the physical enterprise.

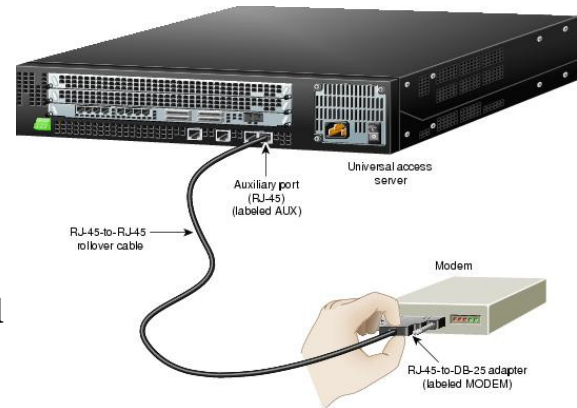
Aside from our users, VPNs also come into play when establishing business partner connections. In past years, we may have paid for a private frame relay link between us and our partner. These days, it is far cheaper to simply set up a branch office VPN tunnel. If we establish such a link, it is always best to own both ends of the connection from a hardware point of view. If we don't, how can we be sure that the settings are correct?

Exactly the same principle applies to our enterprise users! If we don't own the hardware that they connect to the enterprise with, how can we possibly require or enforce settings! Even though our VPN technologies can provide some capabilities for performing health checking of the connecting host, unless that host is locked down, it's easy to trick the VPN health check without even trying! Most of these solutions are simply using the Microsoft API to ask the system if it is patched. If the system doesn't know any better it will answer that question with a "yes"!

If we are not going to own the systems that our users access our network from remotely, some sort of remoted desktop or terminal server access is likely the best way to go. When you "connect," you actually aren't taking the data down to your computer. Instead, you are viewing the data on the remote systems through your web browser or a dedicated client program using a remote-desktop style system. Although the user could still take screenshots, the actual data passing into the enterprise can be limited to the remote desktop connection, dramatically increasing our security.

Out-of-Band Management

- How do I manage infrastructure
 - Firewalls, VPNs, switches, routers...
 - What if the VPN is down
- It's okay to have another way in:
 - It must be documented
 - It must adhere to security standards
 - These are frequently poorly managed



One of the other big issues to be aware of and to ask about is out-of-band management. This can be handled in several ways. Some of these have big security issues.

One out-of-band management mechanism applies most frequently to switches, routers, storage arrays, mini-computers, and mainframes. If we lease this equipment or have a service contract with the equipment, quite often there will be a plain old RJ-11 telephone connection plugged into a modem hooked directly up to the serial interface of the system. This is potentially an enormous security risk because these serial connections are rarely monitored and typically have no lockout capabilities. Although it is true that an attacker entering the enterprise this way would have a low-speed connection, he can typically convert that into a high-speed connection easily by simply opening an outbound connection over a permitted protocol through the firewall.

The other out-of-band mechanism is the “backdoor.” We define a backdoor to be an undocumented mechanism allowing access to systems over the network. Administrators often create these systems specifically to bypass our firewall and/or VPN. The positive reason for doing this is to allow the administrator to access the system even if the firewall or VPN is down.

In reality, we don't have a problem with these types of interfaces. The problem isn't that they exist; the problem is that they aren't documented! If they aren't documented, they are rarely well maintained. Look for these things, ask about them, and make sure that they are documented and approved (or removed if they are not approved).

Third Party VPNs: Review the Contract

- Right to audit
- Who owns the systems
- Under what circumstances can we turn it off
 - What notification must we provide
- List of authorized users provided:
 - Right to refuse
- What about incidents
 - How quickly must you tell me
 - Can I have my people observe

As we started to state on the last page, VPNs these days are not just used for our users. They are the primary way we provide third parties and partners access to internal resources. Looking into these connections always requires that we examine the contracts to see who is responsible for what and how things are supposed to be done.

Looking at contracts is not especially fun or interesting. What we recommend is that you work to inform management, perhaps by identifying some weaknesses in existing contracts, to create a set of requirements for all connectivity-related contracts. After these are identified at a high level and communicated by management to legal, our contracts overall will do a better job of accounting for security issues.

A few of these requirements are listed in the slide. For example, by whom and how may the configuration of the systems and overall security be audited? Can we do it ourselves? Do we have to hire a third party? If it's not in the contract, you can't do it.

Another issue is circumstances that would allow us to terminate service. Has our partner been hacked? Are they experiencing a security event? How quickly do they have to inform us? Can we send or hire people to observe how they are dealing with it and evaluate the extent of the compromise? If we terminate service and the contract does not provide for it, we will likely become the subject of a lawsuit.

Another important item is a list, by name, of the individuals at the remote end who will have access to our data or systems. In addition, we reserve the right to refuse access to any user for any reason, such as former employees and known hackers.

You can find an excellent resource for these types of things in the ISO-27000 standard. The sections on third-party agreements and legal/contractual responsibilities provide a checklist of items to require in your contracts!

Course Roadmap

- Enterprise Audit Fundamentals; Discovery and Scanning Tools
- PowerShell, Windows System, and Domain Auditing
- Advanced UNIX Auditing and Monitoring
- **Auditing Private and Public Clouds, Containers, and Networks**
- Auditing Web Applications
- Audit Wars!

Section Four

1. Private Clouds
2. The Public Cloud
3. Containers
4. Networks and Firewalls
5. WIFI and VPNs
6. **Public Services**
 - DNS
 - SMTP
 - Exercise 4.4 - Auditing Public Services

This page intentionally left blank.

August 10, 2021

Patching?

- How well patched must publicly offered services be
 - Completely
 - No exceptions
- “But we’re not using that”
 - Then why is it turned on
 - Why is it installed

It goes without saying that although all systems must be patched, a publicly accessible or externally accessible system must be patched even more so, or even more quickly. Patching is the primary way that we eliminate vulnerabilities in our systems.

When dealing with patch management, you might face some challenges. At times, you will identify a service or system that is not fully patched; yet when you point it out to the administrator, he will say something like, “Yes, but we’re not actually using that.” Another response might be, “It doesn’t matter that we haven’t patched that because you can’t get to that from the internet.”

Think about these two responses. Right now, our context is publicly accessible systems. If an administrator is claiming that the service isn’t actually used, what is he telling you? He’s saying, “I have failed to remove the unnecessary services from my system.” Economy of mechanism and least privilege both insist that these unneeded services be disabled or, better, removed! When discussing this with the administrator or with the organization in general, you can absolutely point to organizational standards and requirements. However, if these do not speak to the issues you are finding, always try to relate these issues in terms of security principles. It’s not about the specific thing you’re finding, it’s about the principles involved.

Public Systems and Firewalls

- “But the firewall is protecting it”
 - The firewall protects it only from people on the outside
 - We know the firewall already has a permit rule for something:
 - Otherwise, this wouldn’t be a public service.
 - What if some other public service is compromised? $P = 0$

If they claim that because it can’t be reached over the internet it doesn’t matter, there’s something else that they are missing. Recall our discussion of time-based analysis. We pointed out that we have to assume that our security fails. In fact, if we have a publicly exposed service, we don’t have to imagine this. If we have a mail server listening on port 25, what is the value of P, or protection, on port 25 through the firewall? Zero! The firewall has a permit rule!

If someone can compromise any one of these systems through a service that is permitted, what prevents that attacker from going after the rest of the ports that no one bothered to patch now that he is behind the firewall?

Again, a principled-based approach to this problem enables us to see what can go wrong. In addition, this would be an excellent place to perform a formal risk assessment if the organization fails to respond to your findings. Clearly, the risk here is high!

August 10, 2021

DNS Extremely Important

- If I own your DNS, I own you:
 - Public DNS is determined by Whois:
 - Domain registrar specifies authoritative servers

```
Tintadgel:~ dhoelzer$ whois sans.org
Domain Name:SANS.ORG
Domain ID: D4201868-LROR
Creation Date: 1995-08-04T04:00:00Z
...
Name Server:DNS31A.SANS.ORG
Name Server:DNS31B.SANS.ORG
Name Server:DNS21A.SANS.ORG
Name Server:DNS21B.SANS.ORG
```

One publicly accessible system that everyone has is a Domain Naming System (DNS), which is essentially the directory service for the internet.

When a user wants to browse to Google, he does not open his browser and type `http://74.125.226.7/`. The user tells his browser, “I want to go to google.com.” Behind the scenes, the computer asks whichever DNS server has been configured where it can find google.com. If that DNS server doesn’t know the answer, it will “recurse.” What this means is that it goes and asks other DNS servers until it determines that the domain doesn’t exist, determines that the host doesn’t exist, or discovers the actual address. In many ways, the DNS service is like a phone book. We know names; the computers know where the numbers are.

The authoritative DNS servers for any domain are maintained in the Whois system. Whois, which operates over port 43, can be queried to determine where the authoritative servers are for any domain. In the slide, for example, we use Whois from a UNIX command line, and it tells us that SANS has four name servers configured to provide authoritative data.

For a moment, consider how critical the DNS system is to your organization. If someone can redirect name resolution for your domain to his own server, he essentially “owns” your domain. He doesn’t have to get your passwords, he doesn’t need to deface your website, and so on. Anytime anyone tries to find you, he completely controls where the data goes. Whenever someone sends you an email, he controls where that email will be delivered.

Clearly, we need to make sure our DNS servers are locked down, the records are correct, and we have good procedures for keeping things safe.

Social Engineering!

Domain Name: BINGHAMTON.EDU

Administrative Contact:

Michael Hizny
Binghamton University
4400 Vestal Parkway East
Computer Center 102H
Binghamton, NY 13902-2000
UNITED STATES
(607) 777-6420
abuse@binghamton.edu

← No names! List job
roles only!!



SANS

AUD507 | Auditing & Monitoring Networks, Perimeters, & Systems

159

There are some important items to look for when examining DNS configurations. First, use Whois to pull information on your domain. Can you see actual names of individuals listed in the registration information? As a rule of thumb, this is bad.

Looking at the example in the slide, would you agree that Michael Hizny is likely an individual with some significant level of authority? Could an attacker use this information to make some phone calls and possibly socially engineer someone into doing something that he shouldn't? We might even use this information to attack Michael, claiming to be the registrar calling to validate information with him or informing him of a new administration interface that he needs to use from now on!

There is another significant risk. If we have an individual's name here, what happens when that individual leaves the organization? Although that person no longer has access to his internal email account, all that he needs to do is contact the registrar. When he does, he will be asked to provide a photocopy of a government-issued ID and a request on organizational letterhead asking for the address to be changed. Of course, it would be trivial for Michael to provide the ID. What about the letterhead? Do you think the registrar has a database to tell them whether the letterhead that this former employee creates is real?

Yet another thing to check on is the expiration date for the domain. We don't have this pictured in the slide, but this data is also displayed when you perform a Whois. If the expiration date is anywhere in the next year, it is always good to ask who is responsible for renewing it. Often organizations pay registration fees for multiple years. When the registration comes due, it is easy to forget about it.

That leads to a final issue related to this slide. We'd like to find that there are general role-based email addresses listed and that these addresses are *actually reachable*! In other words, the email will actually show up in someone's inbox. Otherwise, any notifications about our domain will be overlooked or lost.

DNS Questions

- Standard questions:
 - Admin credentials, minimal services, time sync, patched, changes, and so on
- Do we own our DNS servers
- Will they allow public recursion
- Are zone transfers restricted
- Are we employing split DNS

The standard questions about how the system is administered, patched, how logging is configured, and so on all apply. For DNS, though, there are some additional issues to dig into.

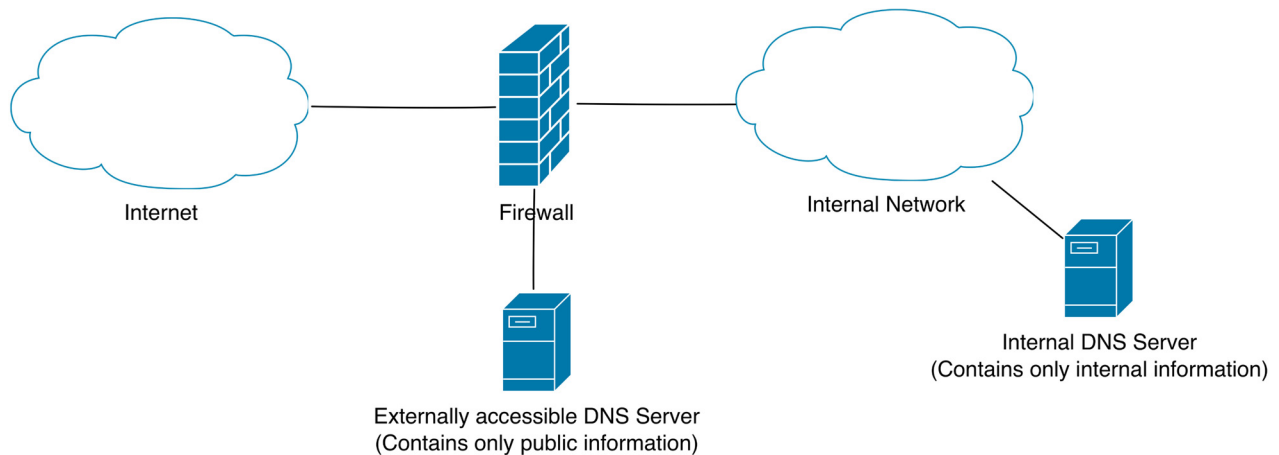
One of these is that we would prefer to find that we own and manage all our DNS servers. If we allow an ISP or someone else to manage our DNS entries, we are at their mercy for making emergency changes. In addition, we have no control over how well patched or otherwise well configured these servers are. Because DNS is so absolutely vital to the overall security of our domain, we should not be outsourcing the management of these services.

As for the configuration of the services, there are other important questions. For example, our servers should not be configured to allow recursion for addresses outside of our network. Remember that recursion means that the DNS server will try to find a DNS server that is authoritative to answer a particular request. If we permit recursion for outside queries, we create a configuration that is vulnerable to DNS cache poisoning. This would allow an attacker to inject invalid records for third parties into our DNS service.

We must also ensure that zone transfers are limited only to authorized hosts. Within DNS, a set of related hosts is known as a “zone.” For example, `enclaveforensics.com` is a zone. Another zone might be the subdomain `windows.enclaveforensics.com`. These related records are all stored in the same configuration file. DNS peers exchange this information through a zone transfer. If we do not secure this, anyone can ask the DNS server for a complete list of all hosts and addresses, instantly obtaining a network map without doing any scanning at all.

We would also like to find that we are employing a split-DNS arrangement.

Split DNS



Split-DNS means that we are splitting the knowledge of the information within the zone. The internal DNS server has all the information about internal hosts. It may also have information about external hosts in the zone, or it can be configured to forward requests to the external DNS server, which would most closely align with the definition of splitting the knowledge of the DNS zone.

The external DNS server, however, only has information about public-facing systems. The wonderful advantage is that even if these servers are compromised, it is not possible for them to reveal internal addressing details because they just don't know how the internal network is configured. As we said at the outset, we have split the knowledge of the zone.

Please note that the external DNS server can be convinced to give away secret information without being compromised! In the lab, you will examine how an attacker might perform targeted queries against the external DNS server, convincing it to reveal internal information. Let's talk about how that type of attack works.

August 10, 2021

Why Important?

- What if I...
 - Send queries directly to your server
 - Forward lookups:
 - Ask it for common names
 - Develop network map
 - Reverse lookups
 - Ask it about your registered address space
 - Ask it about private address space

What if I configured my system to send DNS queries directly to your public-facing DNS service rather than using the DNS server local to me. What if I created a list of commonly used hostnames and asked your DNS server about every one of those names? It's true that the majority of those hosts wouldn't exist; if your DNS server has information for one of these names, it will let me know! I can now create a network map of important hosts on your network without tripping any alarms!

We can do the same sort of thing with reverse queries. A reverse query is like using a reverse phone book. I have your phone number and I want to know what your name is. A reverse directory allows me to look up your number and find your name.

In a similar way, a reverse query says to the DNS server, "Hey, I think there's a host at X.X.X.X. Can you tell me its name?" If the reverse lookup zone is configured, the server happily returns the data. What's even better about this is that we tend to create reverse lookups only for servers that are important. This network map is even more useful! It might even be useful for discovering internal address space!

Example Query Results

```
Tintadgel:DNS dhoelzer$ ./mapper.rb 128.226.0.1 128.226.16.255 128.226.1.18
Starting at 128.226.0.1, counting up to 128.226.16.255
128.226.1.21:      ccsun1.cc.binghamton.edu
128.226.1.32:      bingaixa.cc.binghamton.edu
128.226.1.37:      kerbradius.cc.binghamton.edu
128.226.1.60:      podrouter1.cc.binghamton.edu
128.226.6.5:       bingnfs2.cc.binghamton.edu
128.226.6.20:      blackboard.cc.binghamton.edu
128.226.6.59:      selinux.cc.binghamton.edu
128.226.7.131:     iamdev.cc.binghamton.edu
128.226.7.132:     iamdevdb.cc.binghamton.edu
128.226.9.70:      passwordtest.binghamton.edu
128.226.9.71:      password.binghamton.edu
```

Here you can see an example of mapping out address space using reverse queries. Because we are performing only normal DNS queries, we aren't doing anything improper, unethical, illegal, or otherwise "bad." You can see how valuable this data is, though. The actual test returned many hundreds of results. I have extracted a few that tell us about interesting hosts. For example, would it surprise you if ccsun1 is some sort of Oracle system running Solaris? How about bingaixa? An AIX host? podrouter1 is likely a router somewhere.

We also see an NFS server, a blackboard (which is an online collaboration tool that has had vulnerabilities over the years), SELinux, a couple of Dev machines, and something named "password" and "passwordtest." As an attacker, all these would attract my attention.

You will use a Bash script with similar functionality in a lab exercise.

August 10, 2021

DNSSEC

- Actually starting to be used
 - Certificate used to sign responses
 - Can verify that an answer is authentic
- Unless we all do it, it's not a great protection
 - I don't have to forge your signature... I just don't sign it
- Issue: If the certificate expires, the domain falls over

DNSSEC has been talked about since 1999 in RFC-2538, which proposed a method of storing digital certificates in DNS for use in validating records. Even though everyone seems to recognize that securing DNS and DNS records is critically important, more than 20 years later we are still struggling to get DNSSEC deployed.

As of this writing, we are actually starting to see DNSSEC used. The US government, supposedly, has DNSSEC fully deployed. Google has signed DNS records available. Several other “early adopters” are supporting DNSSEC as well. Support for it is available in all the major DNS platforms that are used as DNS services today.

The idea is that a DNS server can know whether data presented is actually valid by examining the digital signature included with the answer. If the signature is not correct, the server knows to reject the data.

If you do choose to implement DNSSEC (and it is a good idea to implement it!), there is a potential process issue. Some who have implemented DNSSEC have experienced an issue in which the administrators forget to renew the certificate. If we create a certificate that is good for, say, five years, someone has to remember in five years to renew it. If we don't, we will not immediately realize that something is wrong. We can still get to the internet. We can still send email. However, no one who uses DNSSEC can email us, get to our website, or anything else! This can make it difficult for them to let us know that something is wrong!

The right way to fix this is to use automation to maintain the certificates. Protocols like ACME and services like LetsEncrypt have made it much easier to automate the maintenance of certificates for our servers.

DNSSEC – DS Record

- Delegation of signing (DS) resource record gives information about the DNSSEC protected zone file
- Provides digest of DNSKEY signing key
- Good way to validate that DNSSEC is in use
- Includes key tag (45586), algorithm (5), digest type (see notes) and key digest (20 or 32 octets)

```
$ dig DS ietf.org +short
45586 5 1 D0FDF996D1AF2CCDBDC942B02CB02D379629E20B
45586 5 2 67FCD7E0B9E0366309F3B6F7476DFF931D5226EDC5348CD80FD82A08
1DFCF6EE
```

Verifying DNSSEC for a domain can be accomplished using standard DNS lookup tools like dig.

A good starting point is to query for the delegation of signing (DS) resource record, which provides a digest of the DNS signing key installed on the server. This helps to confirm that DNSSEC is in use for the domain.

The response will contain:

- A key tag, used to identify the key for administrative purposes
- The public key encryption algorithm used by the key. In this case 5 = RSA/SHA1
- The digest type being returned in the record. 1 = SHA-1, 2 = SHA-256
- The Base64 encoded key digest

August 10, 2021

DNSSEC – DNSKEY Record

- DNSKEY record gives the public key for validating signatures
- TTL (for expiring responses from cache)
- Algorithm (list in notes)
- Key use:
 - 256 is a zone signing key (ZSK)
 - 257 is a key signing key (KSK)
- Base64-encoded public key

```
$ dig DNSKEY ietf.org +short

257 3 5 ETLgDoQ7 (truncated) rhsiD=
256 3 5 AwEAAdDE (truncated) qaYclBbhk=
```

Querying for the DNSKEY record will get you the public key needed to verify the signature on a DNS response. This resource record includes:

- A time-to-live (TTL) value which tells the resolver how many seconds to keep the query in the cache before expiring it
- The algorithm used to create the public key (possible values are given below)
- The key use. This will be either zone signing (ZSK = 256) or key signing (KSK = 257)

Possible protocols for the DNSKEY record include:

- 1 = RSA/MD5
- 2 = Diffie-Hellman
- 3 = DSA
- 4 = Reserved
- 5 = RSA/SHA1
- 6 = DSA/SHA1/NSEC3
- 7 = RSA/SHA1/NSEC3
- 8 = RSA/SHA-256
- 10 = RSA/SHA-512

DNSSEC – Validated Query (I)

- Add "+dnssec" to dig query
- Authentic data (ad) flag = response verified by resolver
- DNSSEC OK (do) flag = resolver is DNSSEC-aware
- Successful query will return a signature resource record RRSIG as an additional response

Dig can be used to test that appropriately signed results are being returned by a DNS server. Adding the "+dnssec" flag to the dig command line will cause the tool to request the signature resource record (RRSIG) along with the DNS resolution request.

The AD (authentic data) flag will be set by an intermediate resolving nameserver if it verifies the records being returned.

The DO (DNSSEC OK) flag is set by dig to let the server know that it is capable of receiving a signed DNS response.

If the query is successful, dig will receive not only the DNS record requested, but also a RRSIG response with the signature for that record. An example query is on the next slide.

August 10, 2021

DNSSEC – Validated Query (2)

```
$ dig ietf.org +dnssec +multi

; <<>> DiG 9.10.6 <<>> ietf.org +dnssec +multi
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 13514
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 1232

;; ANSWER SECTION:
ietf.org.                1234 IN A 4.31.198.44
ietf.org.                1234 IN RRSIG A 5 2 1800 (
    20220608220538 20210608210855 40452 ietf.org.
    oESug0b5NxoX2jKFDS/6c/4TxgRA== )
N: Tue Jun 08 21:14:21 CDT 2021
```

In this example, we have used dig to request a signed DNS response for the hostname ietf.org. The server returns the A record with the address of the host, and it add an additional RRSIG record with the signature for the response.

Remember that the "do" flag was used by dig to let the server know that it's okay to sign the results.

SMTP

- Typically, our internal mail infrastructure is insulated from the internet:
 - Exim, Postfix, Sendmail, and more
 - Acts as a “smarthost”
 - Relays all inbound mail in
 - Sends outbound mail out

Another absolutely vital public-facing service that we have is Simple Mail Transport Protocol (SMTP), which is a plaintext protocol used to relay email messages from one system to another.

Internally, you might use Exchange or something similar, but most organizations prefer not to connect domain-connected Windows systems directly to the internet. For this reason, it is extremely common to find that we are running some form of UNIX mail gateway that relays email in and out of our enterprise. This same gateway is used as a “smarthost” to relay all outbound email. A smarthost is a host that can figure out how to deliver any message anywhere.

Insulating our internal mail server from the internet is actually a good idea. However, both the public-facing and internal mail servers still need to be configured securely.

August 10, 2021

Open Relays/Addresses

- **Subscribe to a block list**
 - Pros and cons
- **Enforce RFC compliance**
 - This address is not RFC compliant: d_hoelzer@sans.org
- **Prevent relaying**
- **Permit VRFY or EXPN**
 - Username/address harvesting

Here are some things that should be verified in the configuration of our mail servers:

Are we using a block list? If so, it can help us to dramatically reduce the amount of junk mail that we receive. A block list is a publicly maintained list of hosts known to be open relays or known to be a source of spam or Unsolicited Commercial Email (UCE). Subscribing to a block list can have ramifications, though. If we have a customer or partner who ends up on one of these lists through no fault of his own, we will reject his email with an unfriendly message, accusing him of being a spammer!

Another item is whether we will enforce RFC compliance. In other words, will we require that servers which speak to us do so using the formalized standard for SMTP and ESMTP? Although this is a great feature to turn on and it can decrease the amount of junk mail, we may find that some of our partner email servers are not as well configured as ours and we may begin rejecting email.

A key configuration that must be verified is that our mail server will only relay mail to us or from us. It must never permit people on the internet to relay mail to other people on the internet. If we do let this happen, we will be popular with the phishing and spamming crowd.

We also want to ensure that any extra options that are not actually in use or required are disabled. Two examples are the VRFY and EXPN commands. These are a part of the SMTP standard, but they are never actually used between mail gateways. The VRFY command allows someone to verify whether an email address is valid. The EXPN command enables you to expand an address to the full name of the user. Both of these are used by individuals who are building “validated email lists” that are then sold to spammers and phishers. Again, neither of these is required for our email system to function properly.

SPF, DKIM and DMARC

- Sender policy framework (SPF) is used for detecting forged email senders
 - Uses DNS record to list authorized originating servers
- Domain keys identified mail (DKIM) is used to authenticate senders of emails to prevent spoofing
 - Uses digital signature to prove that email originated from a domain
 - Public key available through DNS query
- Domain-based message authentication, reporting and compliance (DMARC) is used to inform other mail servers of your SPF and DKIM policies and how to handle failures

The combination of SPF, DMARC and DKIM is used to reduce the risk that a malicious actor could spoof emails from your organization's domain.

SPF uses a text-type DNS record to return a list of all servers which are allowed to originate email for the associated domain. This can be used to tell other SMTP servers to reject the email if it does not come from one of the approved hosts (called a hard fail). It can also be used to tell server to accept mail from unauthorized servers but to mark it as having failed validation (called a soft fail).

DKIM adds a digital signature to messages to prove that they originated from an approved server. The public key which can be used to validate the signature is available through a DNS query to the domain.

DMARC uses a text-type DNS record to inform other servers of how they should handle failures in SPF and DKIM verification

August 10, 2021

Validating SPF

- Use dig to query for text records on the domain
- "dig txt domainname.com +short"
- Returns list of authorized servers
- Can use modifiers:
 - + = Pass this origin (can be omitted – listing an origin means to pass it)
 - ? = Neutral. Treat the origin like there is no policy
 - ~ = Soft fail. Accept the message but flag it
 - = Fail. Reject from this origin
- "Include" directive used to add in results from another domain (usually your email provider)

Dig can be used to query for the text record which contains the SPF. The result will describe the version and the rules associated with the policy. The bulk of the policy is dedicated to listing origins which should be accepted or rejected for the domain. The +, ?, ~ and - operators are used to modify or clarify the action that should be taken for a particular origin.

An example query with explanations is on the next slide.

August 10, 2021

SPF Validation Example

- V = SPF version 1
- ip4: and ip6: list of origin hosts/networks
- Include: add in the records from _spf.google.com (can be recursive)
- -all: reject everything else

```
$ dig txt ietf.org +short
"v=spf1 ip4:4.31.198.32/27 ip6:2001:1900:3001:0011::0/64
ip4:66.70.182.38 ip4:158.69.166.16/29 ip6:2607:5300:203:1b26::0/64
ip4:158.69.229.207 ip4:192.95.54.40/29 ip6:2607:5300:0060:9ccf::0/64 "
"include:_spf.google.com -all"
```

In this screenshot, we have used dig to retrieve the SPF policy for the ietf.org domain. In the results, we can see the SPF version number is SPF1. Following that we have a list of IPv4 subnets (like 4.31.198.32/27) IPv4 hosts (like 158.69.229.207) and IPv6 subnets (like 2607:5300:0060:9ccf::0/64) which should be allowed as originators (remember that the "+" modifier is implied if no other modifier is given. Next we are told that the list should include whatever is returned as the txt record for _spf.gogole.com. Finally, we are told to reject all other origins for this domain (-all).

August 10, 2021

Validating DKIM

- Query the domain with the appropriate "DKIM selector" – usually related to the email provider
 - Ask the administrators or analyze a message from the domain
- Text-type DNS record will contain the DKIM version, key algorithm and Base64 encoded key
- Email signatures can be validated using the key

```
$ dig txt google._domainkey.ondmarc.com +short  
"v=DKIM1; k=rsa; p=MIIBIjAN...B8QIDAQAB"
```

To validate DKIM for a domain, we again perform a lookup for a TXT record. In this case, however, we need to know the DKIM selector used with the domain. This selector should be available from the administrators, or you can get it by analyzing an email sent from the domain. In the DKIM-signature header for a signed email, there will be a field named "s" for selector. That field will contain the selector which should be used at the beginning of the query.

The result will include the DKIM version (DKIM1), the key type (RSA) and the base64-encoded key (truncated in this screenshot).

Validating DMARC

- Query DNS for the "_dmarc" text record
- Result will include:
 - DMARC version
 - Policy (none, quarantine or reject)
 - RUA (reporting URI for aggregate data) and/or
 - RUF (reporting URI for failure data)

```
$ dig txt _dmarc.sans.org +short  
"v=DMARC1; p=quarantine; rua=mailto:dmarc_agg@vali.email"
```

To validate DMARC, we query for the "_dmarc" TXT record in the domain. The result will include the DMARC version (DMARC1), the policy (quarantine), and the reporting URIs for aggregate data and failure reports, in this case an email address (dmarc_agg@vali.email).

August 10, 2021

Message Encryption

- Inquire about interdomain encryption solutions:
 - Not hard
 - S/MIME and PGP are most common.
 - Biggest questions:
 - How are we publishing and obtaining revocations?
 - How do we recover email for departed users?

Before we move on from mail servers, there's another issue to point out. We would like to inquire about inter-organizational encryption. In other words, when we need to exchange secure messages with a third party, how are we doing that?

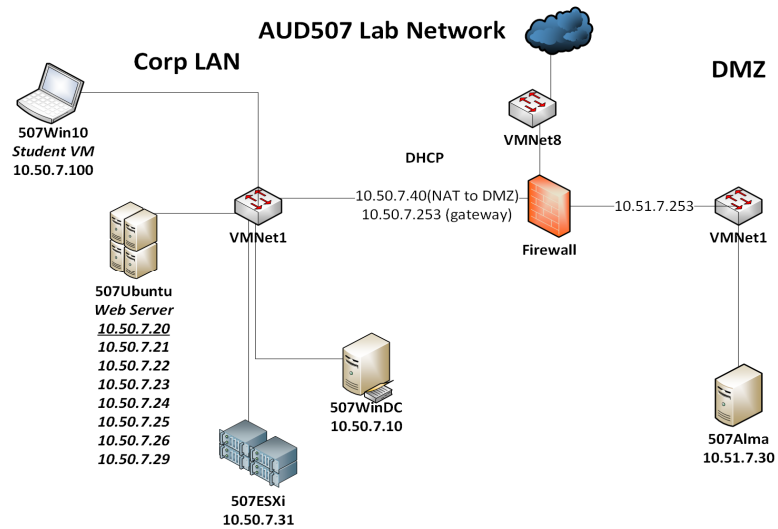
Encryption between organizations is not hard. The two most common solutions for this are S/MIME and PGP, though there are other providers with radically different approaches involving drop boxes and possibly even two-factor authentication. The advantage of S/MIME is that it's pretty much built into everything these days. All I have to do is import a certificate for my email address and I'm ready to go. PGP is a bit more work. We definitely have to install either a free or commercial client and get it integrated with our mail client.

For both of these, though, we need a way of obtaining the keys or certificates of the people with whom we will communicate. We also need our data published. Do we have a key server for this purpose? Are we using free internet solutions to publish this data? How do we validate that a certificate sent to us by a third party is actually its certificate and not a forged certificate?

Another issue is, what happens when a user is terminated? How do we publish terminations? Is the CRL (Certificate Revocation List) accessible? Is it reasonable that a remote client will even consult our revocation list? This is actually one of the fundamental problems with PKI when we do not share a common infrastructure.

We would also like to inquire about access to the email of departed users when encryption has been used. Are we issuing the certificates or keys? Do we maintain an escrowed copy of that key so that we can read the email in the mailbox of a departed user? The time to find out the answer is before you need access.

Exercise 4.4 - Auditing Public Services



This page intentionally left blank.

August 10, 2021

Daily Status Update Agenda

- Fieldwork completed today:
 - Audited stand-alone hypervisors
 - Audited Docker setup
 - Audited switch and firewall configs
 - Audited DNS and SMTP servers
- Any findings
- Recommendations
- Questions for auditee

Think back on the fieldwork you completed today.

Were there any problems with the hypervisors, either ESXi or Xen?

Was the Docker configuration appropriate? What risks exist around Docker?

Were the switch and firewall configurations appropriate?

Were the public services configured properly?

August 10, 2021

Thank You!



This brings us to the end of Day 2. If you are taking the class at a conference, please take a moment to complete an evaluation form. You will be given a different evaluation every day of the class.

August 10, 2021