DSci526:
Secure Systems Administration

NIST Best Practice
(continued)
Host Administration
Protection Domains

Prof. Clifford Neuman

Lecture 4
10 February 2021
Online

Course Identification

- DSci 526
  - Secure Systems Administration (4 units)
- Class meeting schedule
  - Usually 2PM to 5:20PM Wednesday
  - Online
- Class communication
  - dsci526@csclass.info
  - Goes to instructor and any assistants and is archived.

Announcement

- Alternate Schedule for February 17th Lecture
  - Wednesday 17 February
  - 10:30AM to 1:50PM
  - Same Zoom Link

Student Presentation Topics

- I have only received 19 out of 24 proposals.
- I am organizing those received into the following sessions.
- For those who have not sent proposals, please send me your proposed topics as soon as possible, and please avoid picking topics that have already been assigned.

Configuration Management

- Configuration Management has been proposed as a topic by 1 student.
  - Marco Gomez

- This topic is on the class schedule for next week 2/17/21 but I am willing to have this presentation held later in the semester depending on student preferences.

Secure Cloud Administration

- One student has proposed a topic related to securely managing cloud deployments.
  - Sarahzin Chowdhury - Cloud Access Security Brokers

- This topic will be scheduled for March 3rd.

February 24th – Red Teaming

- Red Teaming has been proposed as a topic by 5 students and I encourage you to work together to prepare a joint presentation where each of you presents a different aspect of the topic.
  - Hanzhou Zhang
  - Yang Xue
  - Abhishek Tatti
  - Doug Platt
  - Shagun Bhatia

- I will give your teams 100 minutes to present.
- The date for this topic is 3 weeks from now. If your team requires more time to prepare, I can shift class topics around so that you can go a little bit later in the semester.

March 3rd – Incident Response Planning

- Incident Response has been proposed as a topic by 2 students and I encourage you to work together to prepare a joint presentation where each of you presents a different aspect of the topic.
  - Carol Varkey
  - Amarbir Singh

- I will give your team 40 minutes to present.

March 17th – Secure Networking

- Christopher Samayoa (Network Access Control)
- Shanice Williams – Network Monitoring - WireShark

- Two students have proposed in the area of network security. For those that have not yet proposed topics, you may propose on a different aspect of network security (i.e. other than NAC and WireShark), and we can fit your presentation in to this week.

March 31st – Security Incident Event Management

- SIEM topics have been proposed by 3 students. I encourage you to work together to prepare a joint presentation where each of you presents a different aspect of the topic.
  - Malavika Prabhakar
  - Anthony Cassar
  - Dwayne Robinson (Network Perimeter Detection)

- I will give your team One Hour to present.

Linux Related Topics

- We will cover some aspect of Linux administration today or next week, but that is too soon for students to prepare. Therefore, we will slot the presentations on these topics into a later week in the semester.
- Students presenting on this topic:
  - Azzam Alsaeed – SELinux
  - Alejandro Najera – Linux Administration
  - Tejas Pandey – Identity Management in Linux
  - Ayush Ambastha – Linux Kernel Security

- I will give your team one hour 20 minutes to present.

DSci526:
Secure Systems Administration

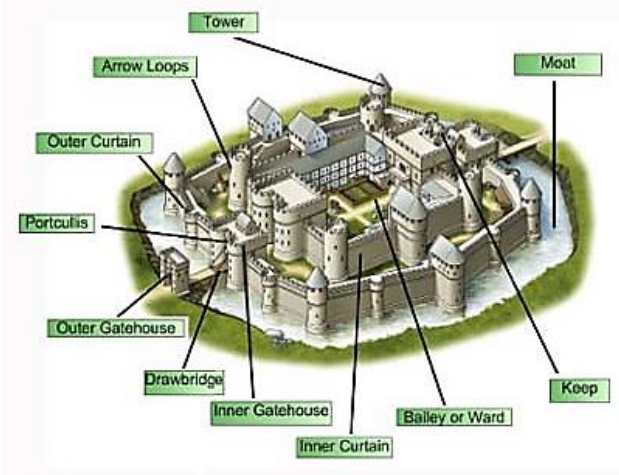
NIST Best Practice
(continued)

Prof. Clifford Neuman

Lecture 4
10 February 2021
Online

Set 3.13 System and Comm Protection

- 3.13.1. Monitor, control, and protect communications (i.e., information transmitted or received by organizational systems) at the external boundaries and key internal boundaries of organizational systems.
- 3.13.2. Employ architectural designs, software development techniques, and systems engineering principles that promote effective information security within organizational systems.
- 3.13.3. Separate user functionality from system management functionality.
- 3.13.4. Prevent unauthorized and unintended information transfer via shared system resources.
- 3.13.5. Implement subnetworks for publicly accessible system components that are physically or logically separated from internal networks.



Set 3.13 System and Comm Protection (2 of 3)

- 3.13.6. Deny network communications traffic by default and allow network communications traffic by exception (i.e., deny all, permit by exception).
- 3.13.7. Prevent remote devices from simultaneously establishing non-remote connections with organizational systems and communicating via some other connection to resources in external networks.
- 3.13.8. Implement cryptographic mechanisms to prevent unauthorized disclosure of CUI during transmission unless otherwise protected by alternative physical safeguards.
- 3.13.9. Terminate network connections associated with communications sessions at the end of the sessions or after a defined period of inactivity.
- 3.13.10. Establish and manage cryptographic keys for cryptography employed in organizational systems.

Set 3.13 System and Comm Protection (3 of 3)

- 3.13.11. Employ FIPS-validated cryptography when used to protect the confidentiality of CUI. (for now this means AES and appropriate PK systems)
- 3.13.12. Prohibit remote activation of collaborative computing devices and provide indication of devices in use to users present at the device.
- 3.13.13. Control and monitor the use of mobile code.
- 3.13.14. Control and monitor the use of Voice over Internet Protocol (VoIP) technologies
- 3.13.15. Protect the authenticity of communications sessions
- 3.13.16. Protect the confidentiality of CUI at rest.

Set 3.14 System & Information Integrity

- 3.14.1. Identify, report, and correct information and system flaws in a timely manner.
- 3.14.2. Provide protection from malicious code at appropriate locations within organizational systems.
- 3.14.3. Monitor system security alerts and advisories and take appropriate actions in response.
- 3.14.4. Update malicious code protection mechanisms when new releases are available.
- 3.14.5. Perform periodic scans of organizational system and real-time scans of files from external sources as files are downloaded, opened, or executed.
- 3.14.6. Monitor organizational system including inbound and outbound communications traffic, to detect attacks and indicators of potential attacks.
- 3.14.7. Identify unauthorized use of organizational system

Risk Assessment and Security Plan

- Start with a list of all your assets, especially sensitive information. Where is it resident on your systems.
  - Who needs access, and who does not.
  - Can you segregate parts of the system needing CUI.

- Assess the controls in place on each of your system components against the 110 controls in NIST SP 800-171.
  - What's missing – i.e. what don't you meet.
  - What do you need to do to meet these requirements
  - Are there interim steps you can take (e.g. segregation – don't process CUI on some systems).
- Complete a System Security Plan
  - This will be a focus later in today's lecture

Set 3.11 Risk Assessment

- 3.11.1. Periodically assess the risk to organizational operations (including mission, functions, image, or reputation), organizational assets, and individuals, resulting from the operation of organizational information systems and the associated processing, storage, or transmission of CUI.
DEFINE YOUR PERIMETER

- 3.11.2 Scan for vulnerabilities in the information system and applications periodically and when new vulnerabilities affecting the system are identified.

- 3.11.3 Remediate vulnerabilities in accordance with assessments of risk.

Risk Management

The goal of security in all organization is to manage risk. The first step in managing risk is your risk assessment (3.11.1)

Once risks are assessed, characterized, and quantified, your organization takes steps to mitigate those risks, balancing the cost of the mitigation against the potential loss resulting from the risk.

Set 3.12 Security Assessment

- 3.12.1. Periodically assess the security controls in organizational systems to determine if the controls are effective in their application.
- 3.12.2. Develop and implement plans of action designed to correct deficiencies and reduce or eliminate vulnerabilities in organizational systems.
- 3.12.3. Monitor security controls on an ongoing basis to ensure the continued effectiveness of the controls.
- 3.12.4. Develop, document, and periodically update system security plans that describe system boundaries, system environments of operation, how security requirements are implemented, and the relationships with or connections to other systems.

3.12.2 Plan of Action and Milestones

Most Federal sites have Cyber Security Resources and some additional materials are noted below and in the References section at the end of this training. The documents below are Plans of Actions and Milestones from some other problem domains, and they may be useful to you as examples. They are not intended to be copied for use as your own plan.

- FedRAMP Plan of Action and Milestones (POA&M) Template
- Centers for Medicare & Medicaid Services - Plan of Action and Milestones Process Guide

DSci526:
Secure Systems Administration

Protecting the Local System
Composition of Systems
Protection Domains

Prof. Clifford Neuman

Lecture 4
10 February 2021
Online

Host Administration

Many security issues today are the result of poor system administration.

- Failure to implement least privilege
- Poor management of user accounts
- Mismanagement of remote access
- Managing permissions incorrectly
- Allowing vulnerable programs to run
- Not keeping required programs up to date
- Misconfiguration of applications

- Not just Linux, but many server machines are implemented on Linux, so that is our focus

Linux Security Administration

Read a guide on the topic:

http://www.linuxtopia.org/online_books/linux_administrators_security_guide/

- **Physical Console Security**
- **Administration**
- Backup
- File system and file security
- User authentication
- System and user logging
- SELinux (or capabilities in CentOS)
- Attach Detection
- Testing Security
- Linux firewall capabilities
- Linux network security
- Lunux software management
- Linux Encrytion
- Limiting and monitoring users
- Malicious code
- Linux VPNs
- Linux Kernel Security
- Linux security resources

Console Security

Basic concern with physical security of resources

- Access to console and hardware allows installation of new OS
- But keep in mind that many virtualized systems provide access to the console remotely.

Administration

Many tools supporting administration

- Tools like Sudo should be used but:

Critical Vulnerability Patched in 'sudo' Utility for Unix-Like OSes

Flaw exists in versions of sudo going back nearly 10 years; USCYBERCOM recommends organizations patch immediately.

The maintainer of sudo, a utility in nearly all Unix and Linux-based operating systems, this week patched a critical buffer overflow vulnerability in the program that gives unauthenticated local users a way to gain root privileges on a host system.

Qualys security researchers who discovered the nearly 10-year-old bug (CVE-2021-3156) say it was first introduced in July 2011 and impacts all versions of sudo from 1.8.2 to 1.8.31p2 and 1.9.0 through 1.9.5p1.

The researchers were able to independently verify the vulnerability and exploit it in multiple ways to gain root privileges on Debian 10 with sudo 1.8.27; Ubuntu 20.04 and sudo 1.8.31; and Fedora 33 with sudo 1.9.2, according to Qualys. Other operating systems and distributions are likely vulnerable to the same issue.

Administration

Many tools supporting administration

- Tools like Sudo should be used but:

- Other administrative tools create an increased attack surface in that an adversary compromising remote administration tools can affect the security of the administered system.
  - So, be sure to lock down these tools
- Centralized administration also has benefits for security, as a common policy (such as disabling accounts of past employees, etc) can be managed centrally, keeping you from overlooking some forgotten system which provides a path back into your organization.

Administrator Remote Access

Today, such access probably is needed over the network to manage systems.

SSH

- With encrypted channel and PK authentication.
- Consider multi-hop attacks and implications.
  - What are they?

VNC

- Widely used for GUI access to virtual machines as well as remote console access.
- There have been many vulnerabilities - https://support.realvnc.com/Knowledgebase/Article/View/266/
- Good idea to manage access to port at firewall (inc host based)

Remote Execution

- Using SSH and other tools

Linux Security Administration

Read a guide on the topic:

http://www.linuxtopia.org/online_books/linux_administrators_security_guide/

- Physical Console Security
- Administration
- **Backup**
- File system and file security
- User authentication
- System and user logging
- SELinux (or capabilities in CentOS)
- Attach Detection
- Testing Security
- Linux firewall capabilities
- Linux network security
- Lunux software management
- Linux Encrytion
- Limiting and monitoring users
- Malicious code
- Linux VPNs
- Linux Kernel Security
- Linux security resources

Linux Security Administration

Read a guide on the topic:

http://www.linuxtopia.org/online_books/linux_administrators_security_guide/

- Physical Console Security
- Administration
- Backup
- **File system and file security**
- User authentication
- System and user logging
- SELinux (or capabilities in CentOS)
- Attach Detection
- Testing Security
- Linux firewall capabilities
- Linux network security
- Lunux software management
- Linux Encrytion
- **Limiting and monitoring users**
- Malicious code
- Linux VPNs
- Linux Kernel Security
- Linux security resources

Least Privilege - Users

Users

- Restrict host access to users with need to access
- Don’t share root password
  - Separate account and limited use of sudo
- Account Management
  - /etc/passwd and shadow password file
    - Strong passwords
  - Network based password mechanisms
    - YP/NIS (bad)
    - Encryption Based (good)
    - SSH pk based login
  - Group management

Least Privilege - Processes

- Programs / Processes
  - UserID
    - Create unique for process or application
    - Have system startup run as user
      /etc/init.d and update-rc.d
  - Data Access
    - Set groups on files / devices
    - Chmod, chown, and chgrp
  - Linux Containers
    - Better than the old standby “chroot”, it provides a lightweight virtual environment, not quite as isolated as a separate VM.

Linux Security Administration

Read a guide on the topic:

http://www.linuxtopia.org/online_books/linux_administrators_security_guide/

- Physical Console Security
- Administration
- Backup
- File system and file security
- **User authentication**
- System and user logging
- SELinux (or capabilities in CentOS)
- Attack Detection
- Testing Security
- Linux firewall capabilities
- Linux network security
- Lunux software management
- Linux Encryption
- **Limiting and monitoring users**
- Malicious code
- Linux VPNs
- Linux Kernel Security
- Linux security resources

Managing User Accounts

Managing Accounts

- Need organizational process to remove account when user leaves organization or changes roles.
  - This is an argument for central management
- This is especially important if user had admin access
  - What kinds of accounts might be left around
- Normal linux tools don't cut it for more than 3 or 4 systems.
  - Why?

Authentication

- Many Tools
  - Password based
  - NIS / Yellow Pages
  - Kerberos and other enterprise systems
  - Pluggable Authentication Modules
  - Federated Identity

Linux Security Administration

Read a guide on the topic:

http://www.linuxtopia.org/online_books/linux_administrators_security_guide/

- Physical Console Security
- Administration
- Backup
- File system and file security
- User authentication
- System and user logging
- SELinux (or capabilities in CentOS)
- Attack Detection
- Testing Security
- **Linux firewall capabilities**
- Linux network security
- Linux software management
- Linux Encryption
- Limiting and monitoring users
- Malicious code
- **Linux VPNs**
- Linux Kernel Security
- Linux security resources

Linux Firewall Capabilities

IPChains
- Supports sequences of rules similar to what is found in appliance firewalls

IPTables (NetFilter)
- Stateful and extensible.
- Can support NATing

IPSec Security Policy Database

These are important in administering servers, and especially gateways.

Linux Security Administration

Read a guide on the topic:

http://www.linuxtopia.org/online_books/linux_administrators_security_guide/

- Physical Console Security
- Administration
- Backup
- File system and file security
- User authentication
- System and user logging
- SELinux (or capabilities in CentOS)
- Attack Detection
- Testing Security
- Linux firewall capabilities
- Linux network security
- Lunux software management
- Linux Encryption
- Limiting and monitoring users
- **Malicious code**
- Linux VPNs
- Linux Kernel Security
- Linux security resources

Chroot

Unix (and hence Linux) call that “changes the root directory of the calling process to that specified in path. This directory will be used for pathnames beginning with /. The root directory is inherited by all children of the calling process.”

“it is not intended to be used for any kind of security purpose, neither to fully sandbox a process nor to restrict filesystem system calls. In the past, chroot() has been used by daemons to restrict themselves prior to passing paths supplied by untrusted users to system calls such as open(2).

(source, chroot man page)

Jails (FreeBSD)

https://www.freebsd.org/doc/handbook/jails.html

"Jails improve on the concept of the traditional chroot environment in several ways. In a traditional chroot environment, processes are only limited in the part of the file system they can access. The rest of the system resources, system users, running processes, and the networking subsystem are shared by the chrooted processes and the processes of the host system. Jails expand this model by virtualizing access to the file system, the set of users, and the networking subsystem. More fine-grained controls are available for tuning the access of a jailed environment. Jails can be considered as a type of operating system-level virtualization."
(Matteo Riondato)

Jails vs Trusted Computing

Trusted computing provides an inside-out jail

- Jails protect what is on the outside from that within.
- Trusted computing protects the inside from the rest of the system

From Previous Discussion Network Containment Tools

- Creation of network protection domains
  - Firewalls
  - VLANs
  - VPNs for access
  - Ipsec
- Define required characteristics
  - Where is encryption required

Containment Technologies

- Network Containment
  - Firewalls
  - Virtual Lans (VLANS)
  - Virtual Private Networks (VPNs)
  - Encryption
    - SSL, TLS, IPSec, and IPv6 Security
    - End to End
      - Application encapsulation
      - Trusted Computing Key Management
      - Guards
- Network Administration

Containment Technologies

- Containment Within a Computer
  - OS Enforced Access Control
    - MAC or DAC
  - Application Enforced Access Control
    - Database access policies
    - Web access policies (e.g. .htaccess)
  - Specific Technologies
    - Virtual Memory or Segment Architectures
    - Reference Monitory / Access Control
    - User mode vs System Mode
    - Trusted Computing
- System Administration

Containment Technologies

- System Containment
  - Encryption Based
  - Guards
  - Object Encryption

Protection Domain

- An abstract concept for the set of objects and operations on those objects that may be performed by a process.
  - If the access by two entities is the same, then they operate in the same protection domain.
- A process usually operates in its own protection domain.
  - Despite overlap with other processed in the objects that may be accessed, the other processes usually do not have the same access to the first processes memory resources.
- But, one can define protection domains around objects at different types of access, for example, network access.

Controlling Access to Data by Protection Domains

- General Containment
  - System Boundaries
    - Data exists in memory (V or NonV, Primary or Secondary) of a system.
    - It can only be accessed from outside that system with:
      - Physical Access to the peripheral
      - Assistance by a process running on that system
    - Does this apply to NAS?
    - Does this apply to cloud storage?

Processes and Concentric Protection Domains

- Process Boundaries
  - Managed by OS
  - Limits access by processes to their own memory
  - Limits access to storage according to permissions (DAC,MAC)
  - May assign labels to data based on processes protection domain (labels)
- System has full access, Administrator might have full access
  - MAC and Trusted computing can control admin access

Network Containment

- When data is sent across a network
  - It should be considered accessible by all computer on the network segments traversed
  - Unless that data is encrypted
- When a process on a system can communicate with a process in the network.
  - It should be considered subvertable by any process with which it communicates.
  - A subverted process can not control access to information within its protection domain.
- Network Containment
  - Controls the segments of which data can traverse (outbound)
  - Controls communication (inbound) that is capable of subverting a process or accessing data.

Host Administration Guidance

- Create multiple protection domains
  - Don’t run anything as root (or as little as possible)
- Configure access to resources carefully
- Use Host Based Firewalls as added barrier to communications
  - Reduce the attack surface
  - Consider iptables (packet filters)

Network Administration Guidance

- Use firewalls to contain access
  - Distributed Host Based may be okay and more effective for some environments – embedded even better.
- Disallow by default
  - Open a flow only when defined by application and system architecture.
- VLAn's good, but unless enforced by network hardware or encryption, subverted hosts can circumvent.

Administering Encryption

- Encryption can provide containment independent of the integrity of the systems connected physically to the stored or transmitted data.
  - Reduces protection of data to protection of the key
  - Still circumventable when access to plaintext exists.
- Key Management issues
  - Can leverage trusted hardware
    - Smartcards, Secure Elements, TPM's, Intel's Trusted Execution Technology (TXT)
  - Often too complex to manage at level of authorized users

DSci526:
Secure Systems Administration

First Group Project
(individual part)

Prof. Clifford Neuman

Lecture 5
17 February 2021
Online

Banking Scenario

- Your organization must:
  - Maintain a database of account holders
  - A database of account balances
  - Enable web access by customers who:
    - Can update their personal information
    - Check their account balance
    - Transfer funds to another account (by number)
    - View transactions on their account
    - Submit an image of a check for deposit
      - (check should be viewable, but you do not need to scan it or process it)
- Access is needed
  - Via web from the open internet
  - Outbound email confirming transactions
  - All other interactions may be limited by information flow policies to internal machines.

Banking Example Discuss Assignment

- Enumerate the Data
  - Security requirements
- What has access to the data
  - Software components
  - Users – through software components running with identity of particular users or groups
  - Software components run on systems
    - Ideally in their own protection domain
    - But systems have administrator/root access
  - What does this mean for your containment architecture?

Individual Group Assignment

(due by 11:59PM Monday 2/15/21)

- Submit through Drop-box on D2L
- System Structure for Banking Example.
  - Enumerate the classes of data
  - Enumerate the classes of users
  - Suggest the structure of the protection domains
  - Enumerate the systems (software components)
    - And where they should be placed in the domains
    - Any special configurations, choice of O/S, etc.
- Say a little bit about your background and abilities that you will contribute to your group.
  - Which parts of deployment of the system you support (discuss)
- I will assign to groups based on capabilities
  - You will then send your initial assignment to your group members and begin to organize activities to deploy the project using virtual machines among your team.

Preparation for Lab Activities

- You should have already installed free version of vmplayer or virtualbox on your own machine
- Configured some version / dist of Linux as a guest OS.
- Run two instances simultaneously

In your groups you will

- Install servers on some of the VM's
- Utilize VPN's and dynamic DNS to allow interconnection between the VMs.

Connecting to VMs

- VNC – Virtual Network Computing
  - Install TightVNC or other Client on machine from which access is attempted.
  - Install and configure VNC server on Virtual Machine
  - A VNC Server can be run inside your VM, or in the hypervisor
    - Inside the VM is likely easier
- Portmapping a must
  - Find the IP using dynamic DNS
  - But multiple VM's on a shared NAT need to be mapped manually to different ports.

Group Exercise One

- Decide on the software components to be deployed to implement software requirements on next slide.
  - Custom development should be simple scripts.
  - Use packages for database and other components.
- Decide on the VM's to be created to run those software components.
  - You can run more than one software component within a VM if you choose.
  - Decide on the methods you will use to contain access to those software components, and to the information managed by those components.
- Configure communication between VM's and to the outside
- Install packages
- Write scripts and demonstrate basic flow through system.
- Report on progress as group before class on February 24th.

DSci526:
Secure Systems Administration

Advance Slides
Configuration Management

Prof. Clifford Neuman

Lecture 5
17 February 2021
Online

Configuration Management

A process for consistently establishing and maintaining the characteristics of the components of a system relevant for the proper functioning of a system.

- Proper functioning includes:
  - Security
  - Updates and security patches.
  - Detection and prevention of unauthorized changes.
- Components includes all system assets:
  - Hardware
  - Software
  - Credentials
  - Licenses
- Characteristics includes:
  - Accounts
  - Settings
  - Polices.

Purpose of CM

- To Maintain Consistency of a system and its attributes with a technical baseline over the systems life.
- CM is part of system's security assurance cycle.
- Reduce the management workload for a collection of systems.
- Reduce the attack surface of a collection of systems by reducing the differences between individual systems within the collection.

Configuration Management

Example Attack Scenarios for Misconfigured Systems

- An app server admin console was automatically installed and was not removed. An attacker discovered the standard admin pages and succeeded to log in with default passwords.
- Directory listing is not disabled on a server. An attacker discovered she can simply list directories to find any file.
- An app server with sample applications that have well known security flaws attackers can use to compromise the server.

Any More?

Out of Date?

Vulnerable Apps?

It Starts with an Inventory

- Catalog of systems
  - What is approved for connection
    - Prevent access by uncatalogued systems
  - For each system:
    - Serial Number, Tag, MACs, IPs
    - Location, Owner, Admin
    - Make/Model, Hardware Features
    - Include routers, hubs, printers, other network attached items.
    - Purpose
    - Software (OS, patches, applications, etc)

It Starts with an Inventory

- Catalog of software
  - What is approved for use
    - Detect unauthorized installs
  - For each system:
    - Name, Version, Patch Level
    - Checksum
    - License information
    - System requirements
    - Security considerations/implications
    - Anything else

Aspects of Configuration Management

- Organizational and Process
  - Change Management
  - Admission Policy
- Technical
  - Dependency Management
  - Patch Management (also organizational)
- Live evaluation and detection
  - Admission Control
  - Whitelists
  - Change detection

Organization aspects of CM

- Identify configuration items to be controlled
  - document user requirements, system design and development, software version, interface control, data flows and network diagrams, test plans and procedures, etc.
  - Use schema or comply with organizational policy to provide unique identifiers for each item.
- Identify the level of CM to be controlled, and determine the hierarchy of each level
  - nature of the system, configuration items, components, etc.
- Identify all baselines to be managed
  - user requirements, system requirements, design, development, experimentation, sustainment, etc.

Configuration Control

- Develop a process to track all configuration item changes and intrusion in appropriate baseline.
- Build or provide specifications to build work products from the software configuration management system, or physical products from the hardware configuration management system.
- Purchase or develop tools for version control of source code, providing version control tracking to the line of code level. Implement an engineering release system to provide hardware version control.

Configuration Status Accounting

- Publish periodic reports describing the current configuration of each configuration item.
- Applies to all installed hardware, software, and other controlled assets during the entire life of the system.
- The reports should include software version and details of hardware in the system and testing.

Configuration Audits

- Performing periodic verifications of operational baselines for completeness.
- Assure that both functional and physical configuration meets the requirements.
- Tools, scripts or logs can be used when a change has occurred in a configuration.

Technical Aspects of CM

- Dependency Managers
  - Linux package managers
- Patch Management
  - Software update options
  - Software update center (linux)
  - Windows updates
  - App Stores
- Special Tools
  - Secuinia, others (later)
- New attack vectors
  - When to update

Tools: CM, evaluation and detection

- Live evaluation and detection
  - Admission Control
  - Whitelists
  - Change detection

Automated Tools: Ansible

- Free software developed by Red Hat
- Written in Python, can run on multiple platforms
- Use Playbook written in YAML to perform tasks
- Use SSH to communicate from the management machine.
- Target servers do not need to install agent files

Automated Tools: Ansible

- Example of tasks and handlers
  - Install WordPress configuration file, and restart httpd

```
tasks:
 - name: install WordPress configuration file
   template: src=wordpress.conf dest=/etc/httpd/conf.d/wordpress.conf
notify:
    - restart httpd
handlers:
    - name: restart httpd
      service: name=httpd state=restarted
```

Some other Tools

- Chef
  - Written in Ruby, can be easily customized.
  - Can run on multiple platforms.
- Puppet
  - Written in Ruby.
  - Can run on multiple platforms.
  - for large scale systems
- CFEngine
  - Can run on multiple platforms.
  - Describe the final state in which one wants to end up. The agent then ensures that the necessary steps are taken to end up.
- Microsoft PowerShell DSC
  - Standard since PowerShell 5.0 (Windows 8.1), desired scale configuration

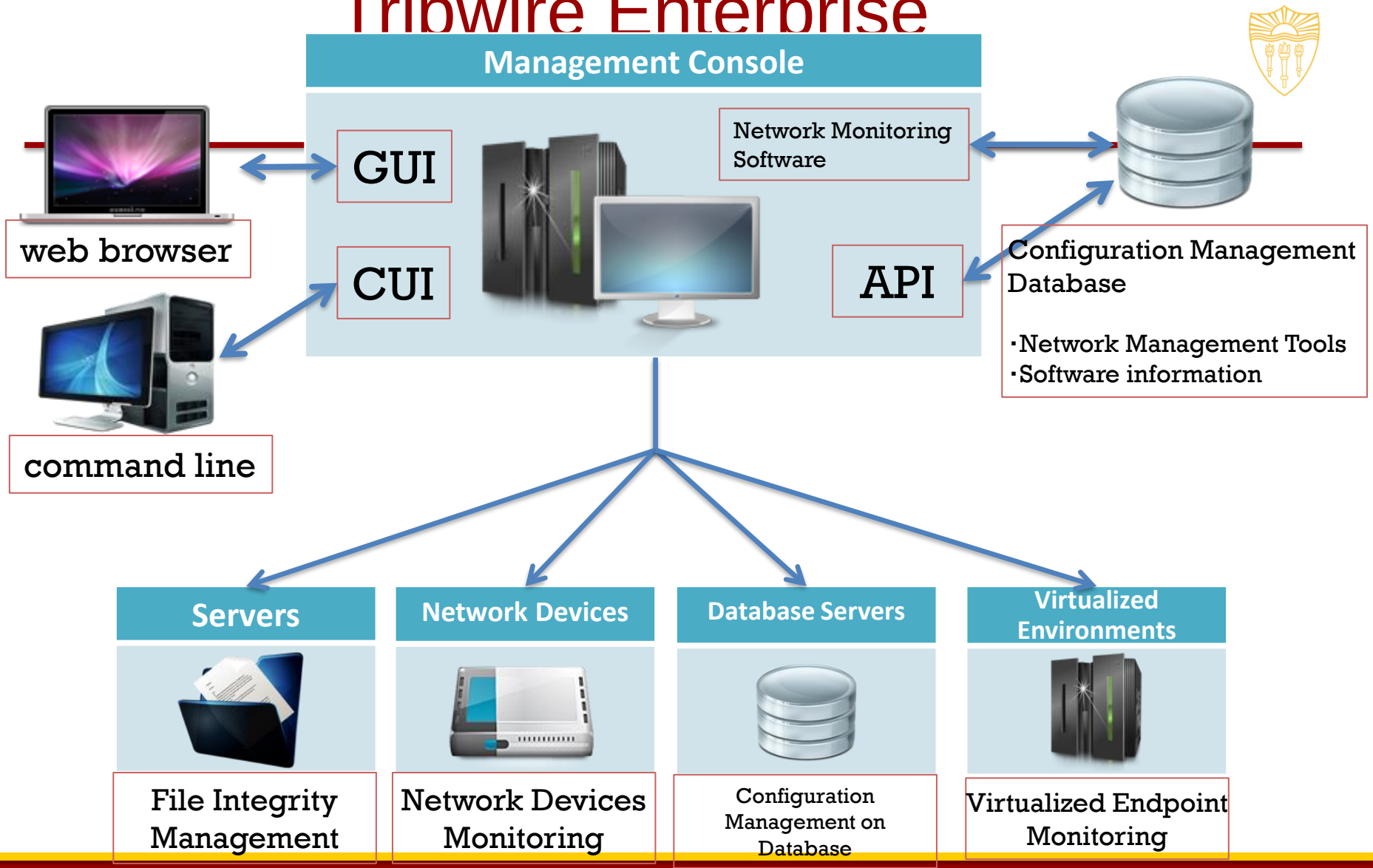
Automated Tools Detection: Tripwire

- Open source and enterprise developed by Tripwire, Inc.
  - Two versions, Open Source Version is not mainained or upgraded
- Detect changes to file system objects
- When first initialized, scans the file system and stores information in a database. Later, the same files are scanned and the results compared against the stored values in the database. Changes are reported to the user by emails.

File Integrity Monitoring

- Act of validating the integrity of operating system and application software files
- Calculate file signatures (Hash values) and compare it to baseline
- Should be performed periodically

Tripwire Enterprise



Open Source Integrity Tools

- Tripwire (some versions)
  - runs on Linux
- AFICK
  - Another File Integrity CheckKer
  - Perl based, deployment on Windows, Linux, Unix, Solaris.
- AIDE
  - Advanced Intrusion Detection Environment
  - runs on Linux

Hash Algorithms for File Signatures

	md5	sha1	sha256	sha512	md160	tiger	whirlpool	gost	crc32	haval
Tripwire	✔	✔	✔	✔						
AFICK	✔	✔							✔	✔
AIDE	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔

File Signature Bypass Issues

- Default MD5 signature can be bypassed by tools.
- Having a wide variety of simultaneous cryptographic generation algorithms can help to detect evasion through signature weaknesses.

How was detected change caused

- Tripwire, AFICK and AIDE note that something has happened
  - Modified? Executed?
- Need correlation with other logs

※Enterprise version does

Protecting the Tools

- Tripwire
  - Require two passphrases longer than eight characters
- AFICK
  - Calculate MD5 signature of itself right after the first installation
  - Can boot from CDROM
- AIDE
  - Signed and stored in the Ubuntu repository, automatically verified during the download and installation

Placement of CM Functions

- Central to the Enterprise
- On each system
- In the Network
- On storage or other servers
- Relationship to SIEM

Minimization in CM

- Create a few standard configurations
  - Fewer different systems to configure and possibly get wrong.
- Automate the configuration of machines within groups.
  - So that you don't leave one out of the update cycle

Push vs Pull

- Most end user systems pull updates
  - Windows updates
  - Linux software updater
- Much server infrastructure changes based on a push model.
  - Ansible (red hat)