Security Architecture

How to provide secure infrastructure

Barbara Krasovec and Daniel Kouřil, EGI CSIRT

Prague, September 2022

What is security architecture?

Some definitions:

- Overall system required to protect your infrastructure (processes and procedures involved in preventing, mitigation and investigating different threats)
- Security principles, methods and models designed to keep your infrastructure safe.
- Security design that addresses potential risks involved in certain scenarios.
- Security control, security policies and security guidelines.
- Security policies and procedures to **prevent, protect, detect, respond and recover**

Security architecture objectives

Security architecture applies to systems, people and network infrastructure. It enables building security into systems:

- design,
- implementation,
- management,
- risk management.

Security architecture main aspects

Security investments

Where to invest if the budget for security is limited? To detection or prevention?

- Both.
- Security threats evolve, malware attacks and zero attacks are constant.
- Thinking about incident response when it already happens is poor strategy.
- Know your data, create backups, harden individual systems, update software regularly, segment network into multiple subnets, use firewall and monitor the activities.

Security architecture focus

- Identifying data stores and their value/sensitivity,
- understanding of critical services,
- restricting access to the data stores,
- threat analysis and risk assessment.

Some security principles

First we will have a look at some basic security concepts:

- defense in depth,
- zero trust,
- least privilege access.

Defense in depth

- The objective is to minimise the effect of the compromise,
- multiple layers and methods of protection: technical, organizational, personnel,
- prevent and mitigate the consequences of security breach,
- if one level of protection fails, the subsequent level is available,
- when a single technical, human or other failure occurs, system should not be compromised,
- in practice: e.g. use firewall on the network border and internally.

Zero trust architecture

Zero trust means that you don't automatically believe everything inside your firewall can be trusted. Zero trust architecture principles:

- Know your architecture, users, devices etc.,
- authenticate and authorize everywhere,
- use MFA,
- assess your user behavior, devices and services status,
- establish security policies,
- don't trust any network,
- monitor users, services, devices.

Zero trust

Zero trust concept evolved over the years. In the 90s this meant providing a firewall, later on, with additional networks in place, it involved hardening systems individually, then detection became the principal focus.
Major changes in security happened with moving the services in the cloud and with mobility, and remote work.

Least privilege access

Principle of least privilege (POLP) access means granting minimum level of access rights to users and services to perform their job.
Why is this important?

- reduces attack surface,
- decreases chances of an attack,
- facilitates service deployment in larger environment,
- improves system stability.

Security Design Principles

- The context: understand the components of your system, its objectives, address short-comings, separate responibilities, understand threat model.
- Design system: network segments, services, communication channels, authn and authz options.
- Harden system compromises.
- Include least privilege approach.
- Identify critical services and sensitive data.
- Provide mechanisms for compromise detection (collect logs and monitor events).
- Reduce attack surface, reduce impact of the compromise and failure.
- Provide incident response plan.

CIS controls

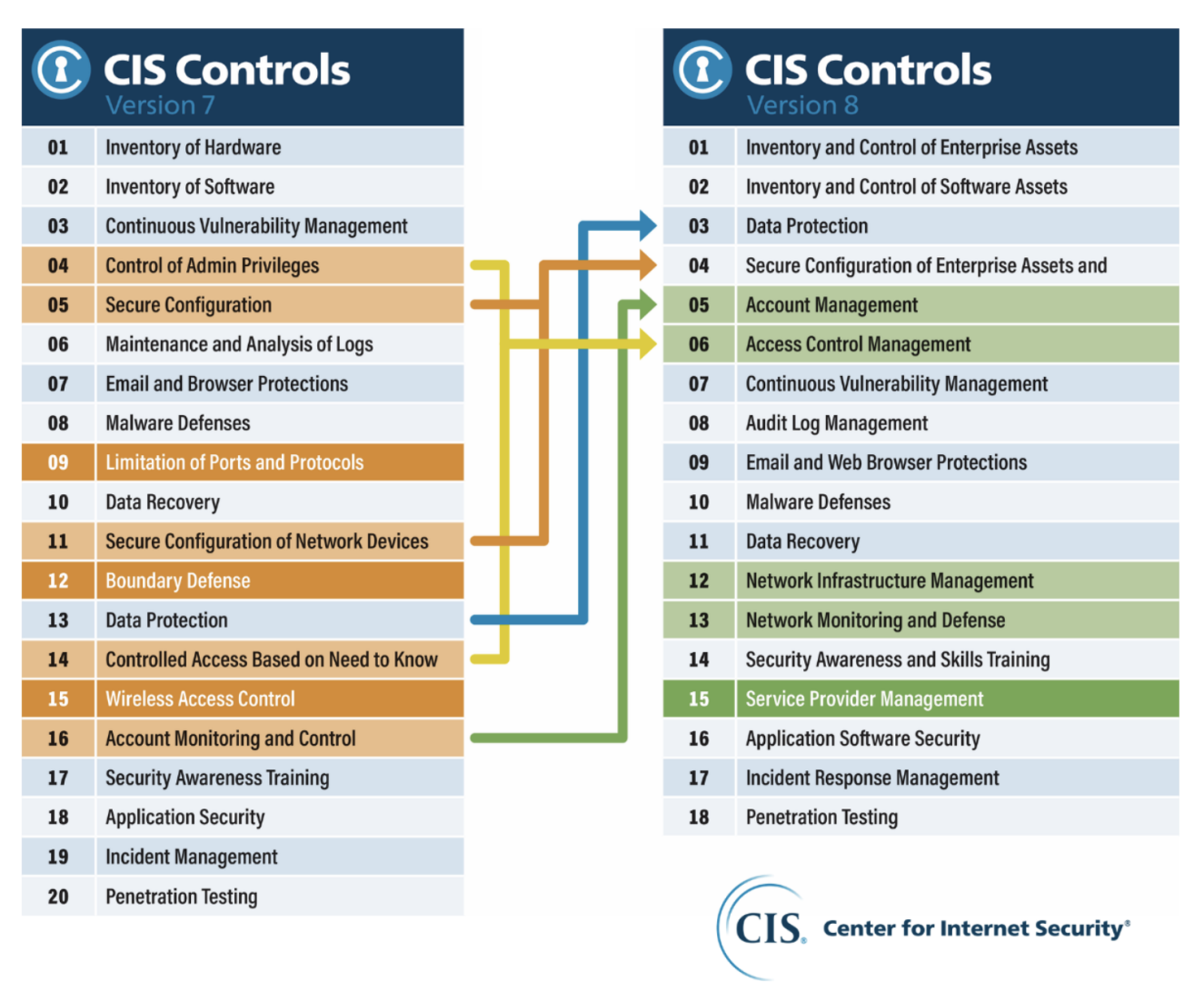
Also known as Critical Security controls, `https://www.cisecurity.org/controls`, developed by Center for Internet security, contain a set of actions for system cyber defense.

- **Basic**: should be implemented in every organization
- **Foundational**: best practices that would be recommended to implement
- **Organizational**: focus on people and processes involved in cybersecurity

CIS controls (2)

- CIS controls are used to identify common exploits,
- provide recommendations on how to defend (safeguards),
- are measurable,
- each safeguard has a description (for small office, for large organization with IT, for organization with security expert group).

CIS controls (3)



https://www.sans.org/blog/cis-controls-v8

CIS benchmarks

How to translate a CIS safeguard to action - configuration guidelines

- more than 100 benchmarks available,
- more than 25 vendor products included,
- many vendors implement CIS benchmarks (such as Nessus, OpenVAS etc.).

CIS controls - Network infrastructure mgmt

Safeguards

NUMBER	TITLE/DESCRIPTION	ASSET TYPE	SECURITY FUNCTION	IG1	IG2	IG3
12.1	**Ensure Network Infrastructure is Up-to-Date** Ensure network infrastructure is kept up-to-date. Example implementations include running the latest stable release of software and/or using currently supported network-as-a-service (NaaS) offerings. Review software versions monthly, or more frequently, to verify software support.	Network	Protect	●	●	●
12.2	**Establish and Maintain a Secure Network Architecture** Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.	Network	Protect		●	●
12.3	**Securely Manage Network Infrastructure** Securely manage network infrastructure. Example implementations include version-controlled-infrastructure-as-code, and the use of secure network protocols, such as SSH and HTTPS.	Network	Protect		●	●
12.4	**Establish and Maintain Architecture Diagram(s)** Establish and maintain architecture diagram(s) and/or other network system documentation. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	Network	Identify		●	●
12.5	**Centralize Network Authentication, Authorization, and Auditing (AAA)** Centralize network AAA.	Network	Protect		●	●
12.6	**Use of Secure Network Management and Communication Protocols** Use secure network management and communication protocols (e.g., 802.1X, Wi-Fi Protected Access 2 (WPA2) Enterprise or greater).	Network	Protect		●	●

CIS controls - Network monitoring

Safeguards

NUMBER	TITLE/DESCRIPTION	ASSET TYPE	SECURITY FUNCTION	IG1	IG2	IG3
13.1	**Centralize Security Event Alerting** Centralize security event alerting across enterprise assets for log correlation and analysis. Best practice implementation requires the use of a SIEM, which includes vendor-defined event correlation alerts. A log analytics platform configured with security-relevant correlation alerts also satisfies this Safeguard.	Network	Detect		●	●
13.2	**Deploy a Host-Based Intrusion Detection Solution** Deploy a host-based intrusion detection solution on enterprise assets, where appropriate and/or supported.	Devices	Detect		●	●
13.3	**Deploy a Network Intrusion Detection Solution** Deploy a network intrusion detection solution on enterprise assets, where appropriate. Example implementations include the use of a Network Intrusion Detection System (NIDS) or equivalent cloud service provider (CSP) service.	Network	Detect		●	●
13.4	**Perform Traffic Filtering Between Network Segments** Perform traffic filtering between network segments, where appropriate.	Network	Protect		●	●

CIS benchmark - Network configuration Linux

3 Network Configuration

This section provides guidance on for securing the network configuration of the system through kernel parameters, access list control, and firewall settings.

Note:

- sysctl settings are defined through files in `/usr/lib/sysctl.d/`, `/run/sysctl.d/`, and `/etc/sysctl.d/`.
- Files must have the the "`.conf`" extension.
- Vendors settings live in `/usr/lib/sysctl.d/`
- To override a whole file, create a new file with the same name in `/etc/sysctl.d/` and put new settings there.
- To override only specific settings, add a file with a lexically later name in `/etc/sysctl.d/` and put new settings there.
- The paths where sysctl preload files usually exist
    - `/run/sysctl.d/*.conf`
    - `/etc/sysctl.d/*.conf`
    - `/usr/local/lib/sysctl.d/*.conf`
    - `/usr/lib/sysctl.d/*.conf`
    - `/lib/sysctl.d/*.conf`
    - `/etc/sysctl.conf`

CIS benchmark - Remove services

Remediation:

Run the following command to remove the package containing the service:

```
# dnf remove <package_name>
```

OR If required packages have a dependency:
Run the following command to stop and mask the service:

```
# systemctl --now mask <service_name>
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.		●	●
v7	9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.		●	●

NIST standards

National Institute of Standards and Technology, non regulatory government agency prepares guidelines and standards for recommended security controls for information systems.

- How to categorise and protect your data?
- How to conduct risk assessments?
- How to prepare a security plan?
- How to implement security controls?
- How to measure performance and efficiency?
- How to process data?

`https://www.nist.gov/cybersecurity`

NIST cybersecurity focus

NIST contributes to the following cybersecurity topics:

Cybersecurity Topics

- → Cryptography
- → Cybersecurity measurement
- → Privacy engineering
- → Securing emerging technologies
- → Trustworthy platforms
- → Cybersecurity education and workforce development
- → Identity & access management
- → Risk Management
- → Trustworthy networks

NIST standards for Measurements for Information Security:

Standards/Guidelines

These are standard publications and guidelines that provide perspectives and frameworks to inform, measure, and manage cybersecurity vulnerabilities and exposures.

SP 800-55 Rev. 1 Performance Measurement Guide for Information Security

This document provides guidance on how an organization, using metrics, identifies the adequacy of in-place security controls, policies, and procedures. NIST is planning to update this Special Publication.

SP 800-30 Rev.1 Guide for Conducting Risk Assessment

This guide provides a foundation for the development of an effective risk management program, containing both the definitions and the practical guidance necessary for assessing and mitigating risks identified within IT systems throughout their system development life cycle.

SP 800-39 Managing Information Security Risk: Organization, Mission, and Information System View

This document provides guidance for an integrated, organization-wide program for managing information security risk to organizational operations (i.e., mission, functions, image, and reputation), organizational assets, individuals, other organizations, and the Nation resulting from the operation and use of federal information systems.

ISO/IEC Standard 19249:2017

Catalogue of architectural and design principles for secure products, systems and applications (last review 2021)

- architectural security principles (virtualisation, redundancy, domain separation etc)
- design principles (how to minimize attack surface, privileges, access control)
- system evaluations
- probably not widely used, as it needs to be purchased and a lot of other free material is available.
- some critics that it doesn't cover advanced material in the field

CIS controls eliminate risks?

Yes, but they are hard to implement, especially for a newbie, advanced system knowledge is required.

Establish security policies

Each organisation should have security policies in place. Use the documentation that is already available:

- AARC Project: `https://aarc-project.eu/policies/policy-development-kit/`
- WISE: `https://wise-community.org/published_documents`

Hardware security

Hardware security considerations

Protecting on-premise systems from natural or human tampering (network devices, IoT devices). It includes:

- procurement process,
- supply chain,
- device security - physical security, software security,
- encryption.

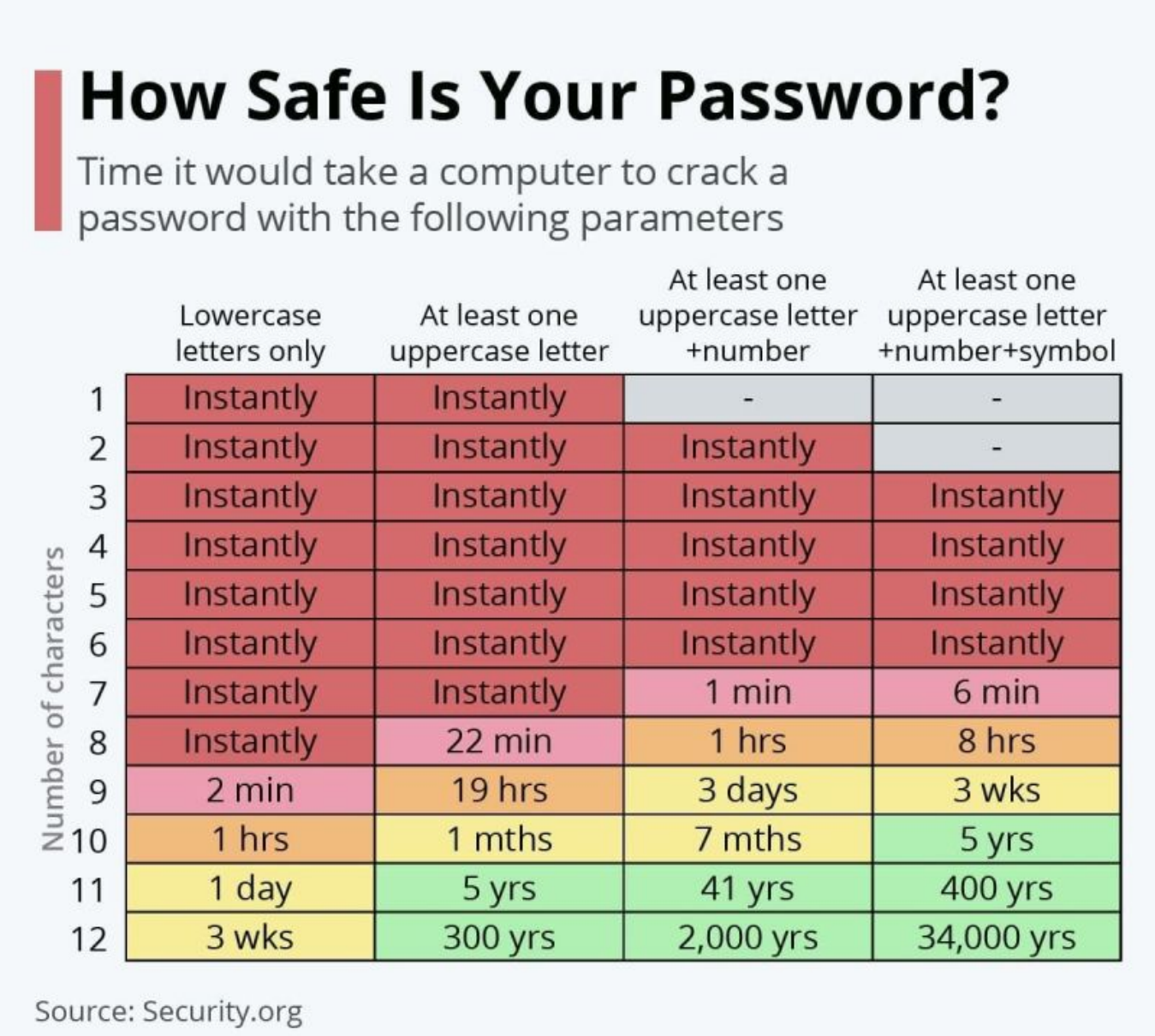
Disable unused interfaces (physically, in BIOS, from OS) or configure them in restrictive manner, e.g. USB device whitelisting.

OS Security

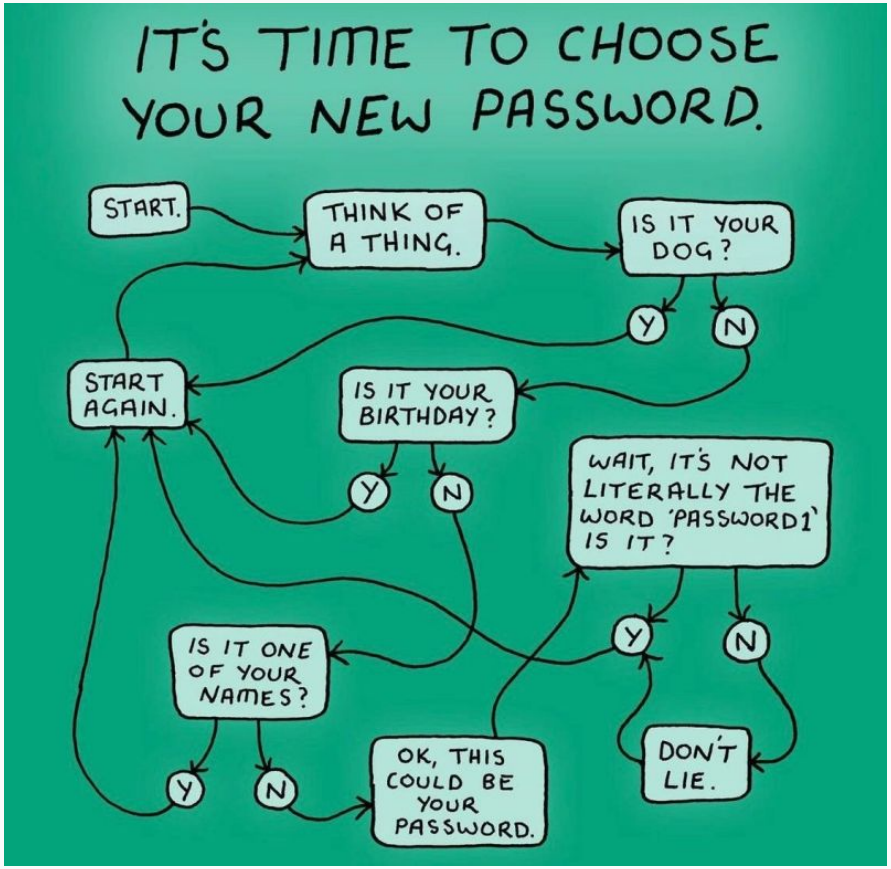
Essentials of operating system security

- modify kernel settings at runtime (sysctl), blacklist uneeded kernel modules,
- network: close unneeded ports, limit access and services (firewall)
- protect files - minimise access rights, FIM,
- software installed: minimize the number of installed packages,
- automate OS deployment, use configuration management tools,
- access: use SSH-keys to login, use auditing, MFA, password change policy,
- security software: enable SElinux, use AppArmor to limit capabilities of programs,
- logging and monitoring: use central logging.

Password reality



Time to choose a strong password



Seriously: use passphrases, they are secure and easy to remember, using a space in the password is good practice.

Password policy Linux

[1] Set number of days for password Expiration.
Users must change their password within the days.
This setting impacts only when creating a user, not impacts to exisiting users.
If set to exisiting users, run the command [chage -M (days) (user)].

```
[root@dlp ~]# vi /etc/login.defs

# line 39 : set password Expiration days (example below means 60 days)
PASS_MAX_DAYS 60
```

[2] Set minimum number of days available of password.
Users must use their password at least this days after changing it.
This setting impacts only when creating a user, not impacts to exisiting users.
If set to exisiting users, run the command [chage -m (days) (user)].

```
[root@dlp ~]# vi /etc/login.defs

# line 40 : minimum number of days available (example below means 1 day)
PASS_MIN_DAYS 1
```

[3] Set number of days for warnings before expiration.
This setting impacts only when creating a user, not impacts to exisiting users.
If set to exisiting users, run the command [chage -W (days) (user)].

```
[root@dlp ~]# vi /etc/login.defs

# line 42 : set number of days for warnings (example below means 7 day)
PASS_WARN_AGE 7
```

[4] Limit using a password that was used in past.
Users can not set the same password within the generation.

```
[root@dlp ~]# vi /etc/pam.d/system-auth
```

Source: https://www.server-world.info

Login nodes and user interfaces

- Apply password change policy or MFA,
- configure to use strong passwords,
- monitor user activity,
- lock accounts after multiple failed attempts (with PAM system-auth),
- blacklist IP after multiple failed logins (fail2ban),
- disable password login if possible,
- configure remote logging,
- keep process accounting (psacct).
- keep track on what users execute (`https://github.com/CERN-CERT/activity_klog`)

Too complex services

If services are too complex to access, noone will use them.

Configuration Management

Process of monitoring/deploying the hardware and software configuration in line with IT policies.

- Enables consistency and automation,
- enables traceability of configuration changes,
- reduced security breaches,
- reduced time to restore service,
- efficient change management,
- easier upgrade automation,
- higher quality of service,
- control over running processes and permissions over the files,
- configuration backup and documentation.

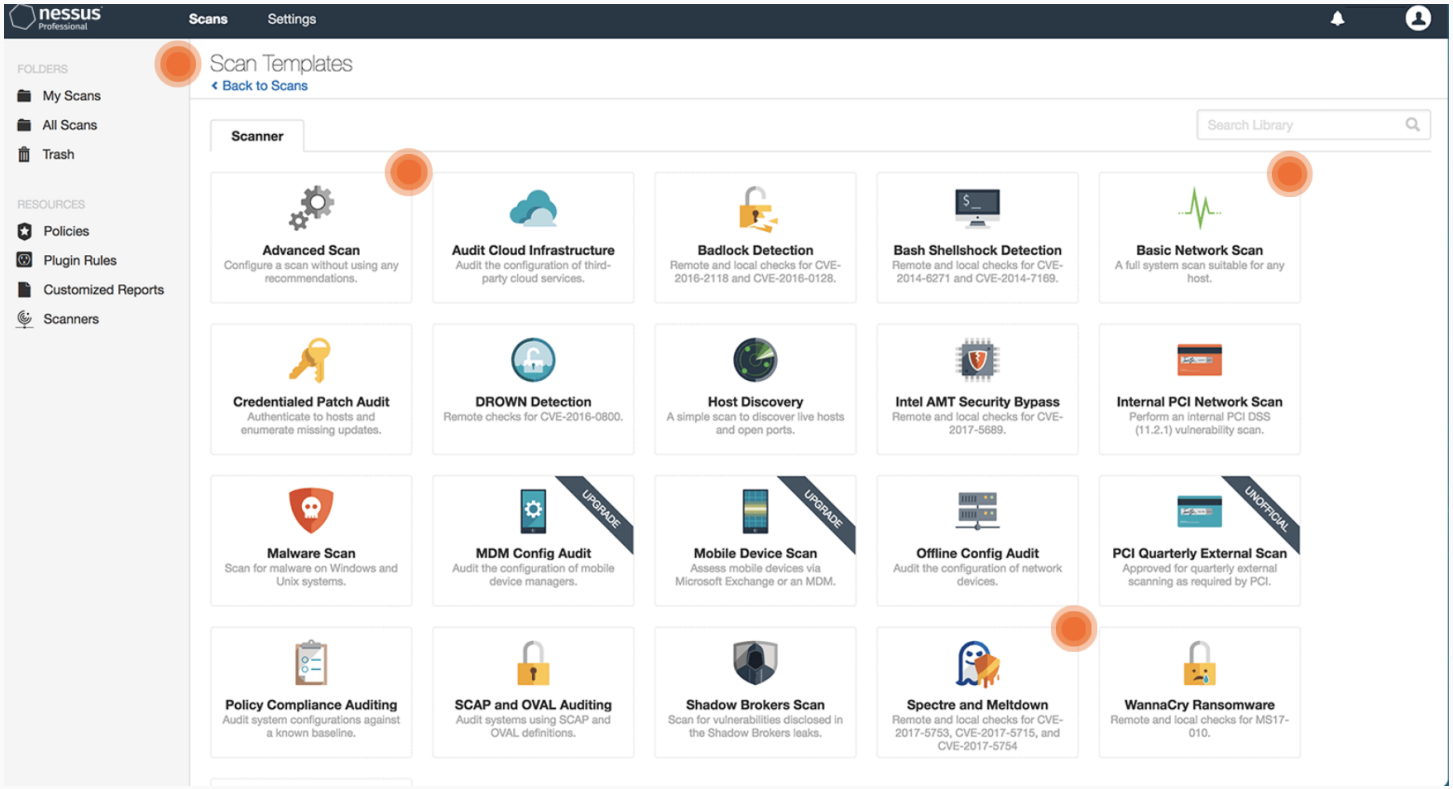
Configuration Management Tools

Linux hardening tools

Advanced task for a sysadmin. Checklists available, but demand knowledge. Some Linux hardening tools available:

- Nessus: security vulnerability scanning tool (checks services and alerts about misconfigurations)
- Zeus: configuration audit, security assessment, self-assessment, system hardening for AWS
- OpenSCAP: vulnerability scanning and security audit tool
- Lynis: scan system for expired SSLs, outdated software, no password user accounts, files etc.
- many others..

Nessus



Source: `https://www.tenable.com/products/nessus/demo`

OpenSCAP security standards

Security Content Automation Protocol (SCAP) is a framework for security standards,it provides tools for assessment, measurement and enforcement of security baselines - how to harden your system and detect misconfigurations.

- Guidelines for Linux,
- validated by NIST (National Institute of Standards and Technology),
- CIS control included,
- command-line tool oscap, GUI is scap-workbench,
- note that the tool has a limited span of checks and guidelines.

OpenSCAP report for CentOS 8

Title	Severity	Result
Guide to the Secure Configuration of Red Hat Enterprise Linux 8 1x fail 1x notchecked		
System Settings 1x fail 1x notchecked		
Installing and Maintaining Software 1x notchecked		
System and Software Integrity		
GNOME Desktop Environment		
Updating Software 1x notchecked		
Ensure gpgcheck Enabled In Main yum Configuration	high	notapplicable
Ensure gpgcheck Enabled for All yum Package Repositories	high	pass
Ensure Red Hat GPG Key Installed	high	pass
Ensure Software Patches Installed	high	notchecked
Account and Access Control		
Protect Accounts by Configuring PAM		
Set Lockouts for Failed Password Attempts		
Set Password Quality Requirements		
Set Password Hashing Algorithm		
Ensure PAM Displays Last Logon/Access Notification	low	notapplicable
Protect Physical Console Access		
Protect Accounts by Restricting Password-Based Login		

Lynis

Security auditing tool for systems running Linux or Unix-based operating system

- Security scan,
- file permissions checks,
- tips for additional OS hardening: kernel parameters (sysctl), SSH configuration, PAM configuration etc.,
- vendor guides included,
- supports multiple standards, such as NIST and also CIS benchmarks.

Lynis report

```
[+] Kernel
------------------------------------
  - Checking default runlevel                               [ runlevel 3 ]
  - Checking CPU support (NX/PAE)
    CPU support: PAE and/or NoeXecute supported             [ FOUND ]
  - Checking kernel version and release                     [ DONE ]
  - Checking kernel type                                    [ DONE ]
  - Checking loaded kernel modules                          [ DONE ]
      Found 42 active modules
  - Checking Linux kernel configuration file                [ FOUND ]
  - Checking default I/O kernel scheduler                   [ NOT FOUND ]
  - Checking core dumps configuration
    - configuration in systemd conf files                   [ DEFAULT ]
    - configuration in etc/profile                          [ DEFAULT ]
    - 'hard' configuration in security/limits.conf          [ DEFAULT ]
    - 'soft' configuration in security/limits.conf          [ DEFAULT ]
    - Checking setuid core dumps configuration              [ DISABLED ]
  - Check if reboot is needed                               [ YES ]

[+] Memory and Processes
------------------------------------
  - Checking /proc/meminfo                                  [ FOUND ]
  - Searching for dead/zombie processes                     [ NOT FOUND ]
  - Searching for IO waiting processes                      [ NOT FOUND ]
  - Search prelink tooling                                  [ NOT FOUND ]

[+] Users, Groups and Authentication
------------------------------------
  - Administrator accounts                                  [ OK ]
  - Unique UIDs                                             [ OK ]
  - Consistency of group files (grpck)                      [ OK ]
  - Unique group IDs                                        [ OK ]
  - Unique group names                                      [ OK ]
  - Password file consistency                               [ OK ]
  - Password hashing methods                                [ SUGGESTION ]
  - Checking password hashing rounds                        [ DISABLED ]
  - Query system users (non daemons)                        [ DONE ]
  - NIS+ authentication support                             [ NOT ENABLED ]
  - NIS authentication support                              [ NOT ENABLED ]
  - Sudoers file(s)                                         [ FOUND ]
```

Devops and security

DevSecOps is a set of practices,policies, approaches and tools, used by IT, Dev and Ops to increase delivering applications and services at high velocity, securely.

- Interesting project to follow: `https://dev-sec.io/`
- Github materials: `https://github.com/dev-sec/`
- OS hardening using automation tools on different OS

Logging

- what to log?
- problem are different formats, timestamps, timezones
- use centralised log management, then analyse
- normalise logs (same format for all)
- provide log rotation
- specify log rotation policy (diskspace, regulatory requirements)
- visualise vital logs
- software: NXlogs, ELK, Graylog, Loki, rsyslog, syslog-ng

Logging checklist

SANS **CRITICAL LOG REVIEW CHECKLIST FOR SECURITY INCIDENTS**

This cheat sheet presents a checklist for reviewing critical logs when responding to a security incident. It can also be used for routine log review.

GENERAL APPROACH

1. Identify which log sources and automated tools you can use during the analysis.
2. Copy log records to a single location where you will be able to review them.
3. Minimize "noise" by removing routine, repetitive log entries from view after confirming that they are benign.
4. Determine whether you can rely on logs' time stamps; consider time zone differences.
5. Focus on recent changes, failures, errors, status changes, access and administration events, and other events unusual for your environment.
6. Go backwards in time from now to reconstruct actions after and before the incident.
7. Correlate activities across different logs to get a comprehensive picture.
8. Develop theories about what occurred; explore logs to confirm or disprove them.

POTENTIAL SECURITY LOG SOURCES

- Server and workstation operating system logs
- Application logs (e.g., web server, database server)
- Security tool logs (e.g., anti-virus, change detection, intrusion detection/prevention system)
- Outbound proxy logs and end-user application logs
- Remember to consider other, non-log sources for security events.

TYPICAL LOG LOCATIONS

- Linux OS and core applications: /var/log
- Windows OS and core applications: Windows Event Log (Security, System, Application)
- Network devices: usually logged via Syslog; some use proprietary locations and formats

WHAT TO LOOK FOR ON LINUX

Successful user login	"Accepted password", "Accepted publickey", "session opened"
Failed user login	"authentication failure", "failed password"
User log-off	"session closed"
User account change or deletion	"password changed", "new user", "delete user"
Sudo actions	"sudo: … COMMAND=…" "FAILED su"
Service failure	"failed" or "failure"

WHAT TO LOOK FOR ON WINDOWS

- Event IDs are listed below for Windows 2000/XP. For Vista/7 security event ID, add 4096 to the event ID.
- Most of the events below are in the Security log; many are only logged on the domain controller.

User logon/logoff events	Successful logon 528, 540; failed logon 529-537, 539; logoff 538, 551, etc
User account changes	Created 624; enabled 626; changed 642; disabled 629; deleted 630
Password changes	To self: 628; to others: 627
Service started or stopped	7035, 7036, etc.
Object access denied (if auditing enabled)	560, 567, etc

WHAT TO LOOK FOR ON NETWORK DEVICES

- Look at both inbound and outbound activities.
- Examples below show log excerpts from Cisco ASA logs; other devices have similar functionality.

Traffic allowed on firewall	"Built … connection", "access-list … permitted"
Traffic blocked on firewall	"access-list … denied", "deny inbound"; "Deny … by"
Bytes transferred (large files?)	"Teardown TCP connection … duration … bytes …"
Bandwidth and protocol usage	"limit … exceeded", "CPU utilization"
Detected attack activity	"attack from"
User account changes	"user added", "user deleted", "User priv level changed"
Administrator access	"AAA user …", "User … locked out", "login failed"

WHAT TO LOOK FOR ON WEB SERVERS

- Excessive access attempts to non-existent files
- Code (SQL, HTML) seen as part of the URL
- Access to extensions you have not implemented
- Web service stopped/started/failed messages
- Access to "risky" pages that accept user input
- Look at logs on all servers in the load balancer pool
- Error code 200 on files that are not yours

Failed user authentication	Error code 401, 403
Invalid request	Error code 400
Internal server error	Error code 500

OTHER RESOURCES

- **Windows event ID lookup:** www.eventid.net
- **A listing of many Windows Security Log events:** ultimatewindowssecurity.com/.../Default.aspx
- **Log analysis references:** www.loganalysis.org
- **A list of open-source log analysis tools:** securitywarriorconsulting.com/logtools
- **Anton Chuvakin's log management blog:** securitywarriorconsulting.com/logmanagementblog
- **Other security incident response-related cheat sheets:** zeltser.com/cheat-sheets

Authored by Anton Chuvakin (chuvakin.org) and Lenny Zeltser (zeltser.com).

Reviewed by Anand Sastry.

Distributed according to the Creative Commons v3 "Attribution" License.

Cheat sheet version 1.0.

https://www.sans.org/brochure/course/log-management-in-depth/6

FIS and HIDS

- **FIM** is a software that monitors and detects file changes that could be indicative of a cyberattack and reports them.
- **HIDS** stands for host-based intrusion detection system and represents an application that is monitoring a computer or network for suspicious activities. (also **NIDS** = network intrusion detection system).
- HIDS tools: OSSEC, Wazuh, AIDE
- poorly configured FIM and HIDS systems can lead to excessive alerts causing Alert Fatigue

Integrity monitoring

- some FIM software: Tripwire, Samhain, OSSEC
- you can set audit.rules on Linux, but only check sensible/critical folders
- check trusted computing base

```
#kernel modules
lib/modules
#binaries:
/bin, /sbin, /usr/bin, /usr/local/bin, /usr/local/sbin,
/usr/sbin
#system configurations:
/etc
#critical files in
/boot, /var/spool, /home
```

Auditd - Search Logs with ausearch

```
~# ausearch --message USER_LOGIN --interpret | less
----
type=USER_LOGIN msg=audit(06/16/2022 07:44:45.104:1919347) : pid=3905900 uid=root auid=unset ses=unset subj=system_u:system_r:sshd_t:s
1023 msg='op=login acct=(unknown) exe=/usr/sbin/sshd hostname=? addr=157.245.245.11 terminal=ssh res=failed'
----
type=USER_LOGIN msg=audit(06/16/2022 07:44:55.132:1919370) : pid=3905953 uid=root auid=unset ses=unset subj=system_u:system_r:sshd_t:s
1023 msg='op=login acct=root exe=/usr/sbin/sshd hostname=? addr=157.230.98.148 terminal=ssh res=failed'
----
type=USER_LOGIN msg=audit(06/16/2022 07:44:55.924:1919376) : pid=3905950 uid=root auid=unset ses=unset subj=system_u:system_r:sshd_t:s
1023 msg='op=login acct=root exe=/usr/sbin/sshd hostname=? addr=159.65.171.230 terminal=ssh res=failed'
----
type=USER_LOGIN msg=audit(06/16/2022 07:45:01.267:1919383) : pid=3905957 uid=root auid=unset ses=unset subj=system_u:system_r:sshd_t:s
1023 msg='op=login acct=(unknown) exe=/usr/sbin/sshd hostname=? addr=103.246.240.28 terminal=ssh res=failed'
----
type=USER_LOGIN msg=audit(06/16/2022 07:45:11.482:1919395) : pid=3906040 uid=root auid=unset ses=unset subj=system_u:system_r:sshd_t:s
1023 msg='op=login acct=(unknown) exe=/usr/sbin/sshd hostname=? addr=198.12.227.59 terminal=ssh res=failed'
----
type=USER_LOGIN msg=audit(06/16/2022 07:45:33.156:1919407) : pid=3906069 uid=root auid=unset ses=unset subj=system_u:system_r:sshd_t:s
1023 msg='op=login acct=(unknown) exe=/usr/sbin/sshd hostname=? addr=180.76.106.84 terminal=ssh res=failed'
----
type=USER_LOGIN msg=audit(06/16/2022 07:45:44.932:1919429) : pid=3906145 uid=root auid=unset ses=unset subj=system_u:system_r:sshd_t:s
1023 msg='op=login acct=(unknown) exe=/usr/sbin/sshd hostname=? addr=188.128.39.113 terminal=ssh res=failed'
----
type=USER_LOGIN msg=audit(06/16/2022 07:45:45.698:1919435) : pid=3906146 uid=root auid=unset ses=unset subj=system_u:system_r:sshd_t:s
1023 msg='op=login acct=(unknown) exe=/usr/sbin/sshd hostname=? addr=159.65.171.230 terminal=ssh res=failed'
----
type=USER_LOGIN msg=audit(06/16/2022 07:45:58.330:1919453) : pid=3906255 uid=root auid=unset ses=unset subj=system_u:system_r:sshd_t:s
1023 msg='op=login acct=root exe=/usr/sbin/sshd hostname=? addr=82.118.225.196 terminal=ssh res=failed'
```

Firewall

A physical firewall device is usually the number one security measure.

- Physical appliance: placed between the uplink and systems, filters traffic before it reaches the system (Palo Alto, Cisco, Fortinet and others)
- Software firewall: iptables, firewalld, nftables; filters traffic on the host
- Best option: use both hardware (outer perimeter) and software firewall (inner layers)

Rootkit detectors

- rkhunter
- chrootkit

Discovering and deleting a rootkit on your server is just the beginning of the problem solving: how did the rootkit get to the server, how was it installed, what has been changed on the system?

Rkhunter scan report

```
Checking for rootkits...

  Performing check of known rootkit files and directories
    55808 Trojan - Variant A                                 [ Not found ]
    ADM Worm                                                 [ Not found ]
    AjaKit Rootkit                                           [ Not found ]
    Adore Rootkit                                            [ Not found ]
    aPa Kit                                                  [ Not found ]
    Apache Worm                                              [ Not found ]
    Ambient (ark) Rootkit                                    [ Not found ]
    Balaur Rootkit                                           [ Not found ]
    BeastKit Rootkit                                         [ Not found ]
    beX2 Rootkit                                             [ Not found ]
    BOBKit Rootkit                                           [ Not found ]
    cb Rootkit                                               [ Not found ]
    CiNIK Worm (Slapper.B variant)                           [ Not found ]
    Danny-Boy's Abuse Kit                                    [ Not found ]
    Devil RootKit                                            [ Not found ]
    Diamorphine LKM                                          [ Not found ]
    Dica-Kit Rootkit                                         [ Not found ]
    Dreams Rootkit                                           [ Not found ]
    Duarawkz Rootkit                                         [ Not found ]
    Ebury backdoor                                           [ Not found ]
    Enye LKM                                                 [ Not found ]
    Flea Linux Rootkit                                       [ Not found ]
    Fu Rootkit                                               [ Not found ]
    Fuck`it Rootkit                                          [ Not found ]
    GasKit Rootkit                                           [ Not found ]
    Heroin LKM                                               [ Not found ]
    HjC Kit                                                  [ Not found ]
    ignoKit Rootkit                                          [ Not found ]
    IntoXonia-NG Rootkit                                     [ Not found ]
    Irix Rootkit                                             [ Not found ]
    Jynx Rootkit                                             [ Not found ]
```

OS security summary

- secure configuration is key, not checking logs and using different security scanning tools
- fail2ban is fine, but keep your SSH configuration secure
- minimise trusted computing base (the smaller the better)
- follow vulnerabilities and patch asap
- least privilege rule (give programs and users only privileges that are required for them to work) - zero trust rule

Physical security

Physical security

- Prevent unauthorised access of personnel, equipment, installations, information,
- protect resources against damage, espionage, sabotage and criminal activity,
- use locked and alarmed doors, fences, guards, CCTV cameras,
- use electronic detection and assessment systems,
- illuminated detection zones,
- armed security for vital area,
- design physical security plan (PSP) + SOP (standard operating procedures).

Network security

Essentials of secure network design

Where is the valuable data? Who has access to it?

- Physical topology: how is the network connected?
- Logical topology: how do services communicate? What is the meaning of the information?

System and network hardening

Fundamental security principle: **reduce attack surface**

- Disable default services that are not needed,
- restrict default permissions,
- close unneeded ports,
- use strong passwords and enforce password change policy,
- start by denying all access/ports, then allow only that which has been explicitly permitted,
- detect if you can't prevent.

Network segmentation

It refers to segregation of the network to multiple sub-networks (segments) by a device (switch, router, hub, bridge..) with the aim to improve security and performance (reduced attack surface), by using:

- access control/firewalls,
- VLANs (virtual local area network),
- SDN (software defined network).

How to segregate network?

- Least privilege rule: only provide access to system that is necessary, nothing else.
- Define zones based on the location of the sensitive data and functionality.
- Do not make system too complex.

Enterprise network

Most enterprise networks are flat, which is very problematic in case of breach, especially if desktop computers are included, which are an easy target for malware.

- 1st step: put servers and desktops into 2 separate subnets, put firewall between them
- 2nd step: monitor network traffic (eg Netflow)
- 3rd step: create another segment for the applications that need to be accessed from the internet, DMZ zone

Eg. Problem with DHCP and flat networks: each device can send DHCP reply

Common network segments

Plan - Analyse - Design - Build - Test - Deploy - Improve The basic network segments are:

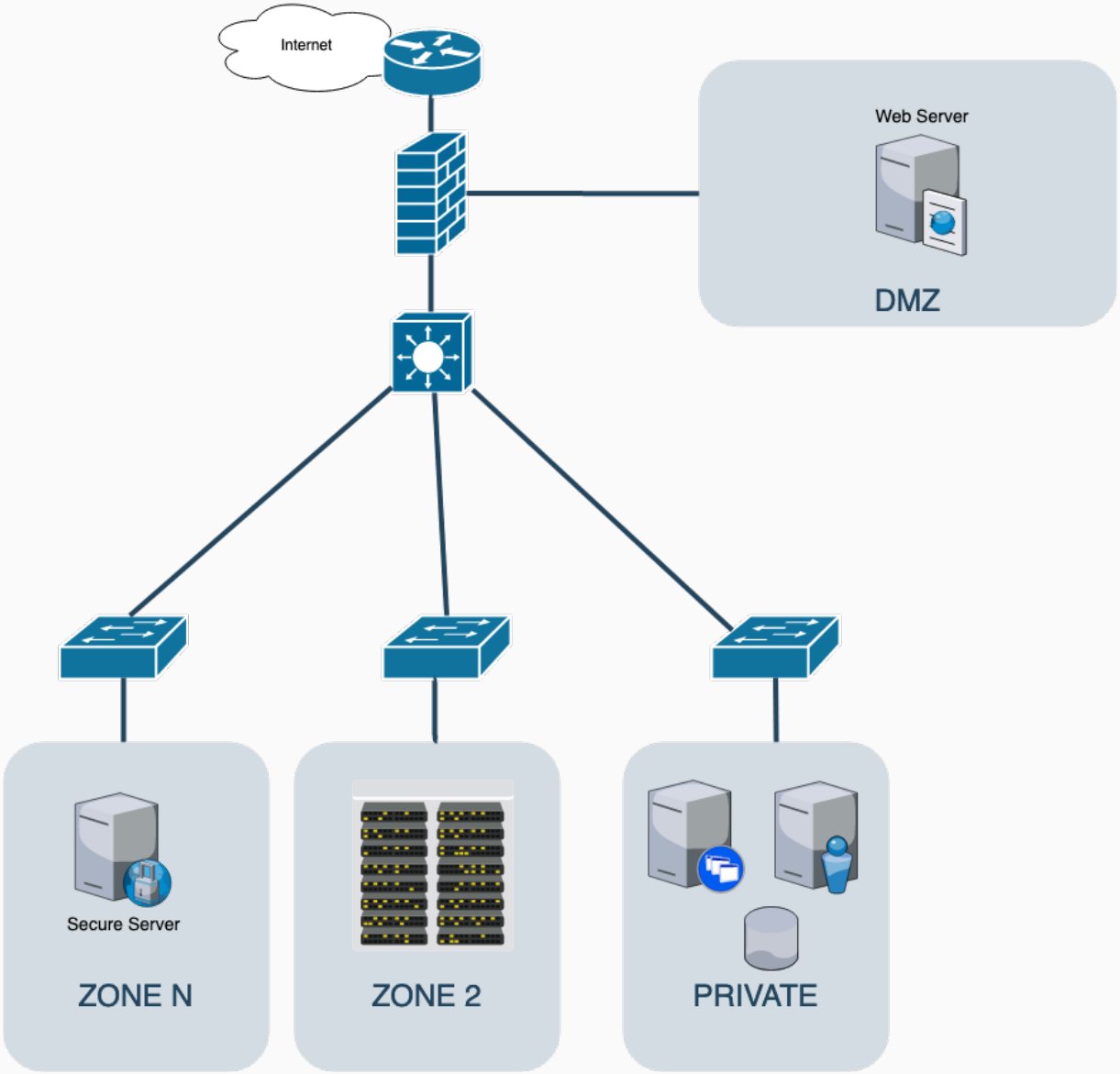
- Public network: Internet (contains no sensitive data, is not under control of the organisation),
- DMZ network (semi-public),services need access to the Internet: web, mail, DNS etc.,
- middleware network is used to separate DMZ from private network (filtered access, proxy servers),
- private network: internal services (contains sensitive information) - only access from DMZ is possible.

Firewall is usually placed between public and other networks.

Basics for network topology design

- Allow internal users to access the internet,
- services that require Internet access should be limited,
- access to the internal services should be prohibited from public network, it should be restricted to DMZ,
- resources in public network cannot be trusted,
- system that is visible from the Internet cannot contain sensitive data (should be in DMZ),
- DMZ communicates with private network via proxy.

Network topology example



Network attacks

Network and switches are some sort of network nodes, they are target of malicious attacks and should be secured as any other node and kept updated.

- DoS,
- packet sniffing,
- packet misrouting,
- SYN Flood,
- brute force attacks,
- MITM attack,
- ARP cache poisoning,
- etc.

How to prevent such network attacks?

- Account lock out,
- rate limiting (policying),
- enable IP source verify (customer cannot spoof its IP address),
- LPTS = local packet transport service - configure allowed settings (e.g number of allowed ICMP packets, number of TCP sessions etc.),
- provide continuous monitoring.

beginframe[Attack mitigation software]

Attack mitigation software

Usually appliances, deployed between router and network firewall, commercial solutions. Prevent from DDoS attacks (blackholes, scrubbing), brute force attacks, syn flood attacks etc.

- Arbor Edge Defense (AED) is an inline security appliance deployed at the network perimeter (i.e. between the internet router and network firewall).

- F5 Silverline DDoS prevention

- Radware Defense pro

Software defined networking

The objective is to make network as flexible and as agile as a VM. SDN enables microsegmentation and decreases the exposure to system attacks.

Device Security

Similar security prevention as for other servers.

- Keep the software updated,
- change default password,
- disable HTTP configuration for routers,
- disable IP directed broadcasts,
- block ICMP ping,
- disable IP source routing,
- establish ACLs,
- establish ingress/egress address filtering policy,
- provide physical security of the devices,
- monitor logs,
- restrict SNMP, route advertising.

IPv6 vs IPv4 security

Is IPv6 networking more secure?

- autoconfiguration support
- IPv6 over IPv4 tunneling support, IPv4 over IPv6 support
- flexible protocol support: NDP (network discover protocol), SLAAC (stateless address autoconfigration)
- support for encryption
- support for IPsec - authentication, integrity and protection against replay attacks
- better QOS support (better availability)
- packet fragmentation is done by hosts only

Although it enables multiple enhancements, it isn't more secure.

Traffic sniffers

Sniffer is a program that monitors data traveling over network.

- Snort
- tcpdump
- Wireshark
- dsniff (for switches)
- Kismet (for wireless)
- nmap

Network security tools

- Wireshark + tshark - network sniffer
- Metasploit - scanners for more than 1500 operations
- Nessus - identifies and corrects faulty updates
- OpenVAS - checks configuration and basic web flaws
- Argus - open-source network analysis tool
- tcpdump - network sniffer
- Kali linux - bootable Linux with multiple security and forensics tools
- Snort - network intrusion detection and prevention system (traffic analysis)
- Suricata - IPS
- Netcat - utility that reads/writes data accross TCP/UDP network connections
- nmap

Complexity vs usability

In network design it is important to find a compromise between the complexity (security) of the network and its usability. If you make your network too complex it will be difficult to manage

Network design recap

- start with good planning (identify components, access, critical data etc)
- plan growth
- design multitier network (network segments) - by functionality and data flows
- provide security (firewall, ACLs etc)
- provide monitoring and IDS, IPS
- provide redundancy for critical services
- implement IPv6
- use secure protocols for transfers
- maintain network documentation

Data security

Privacy vs Security

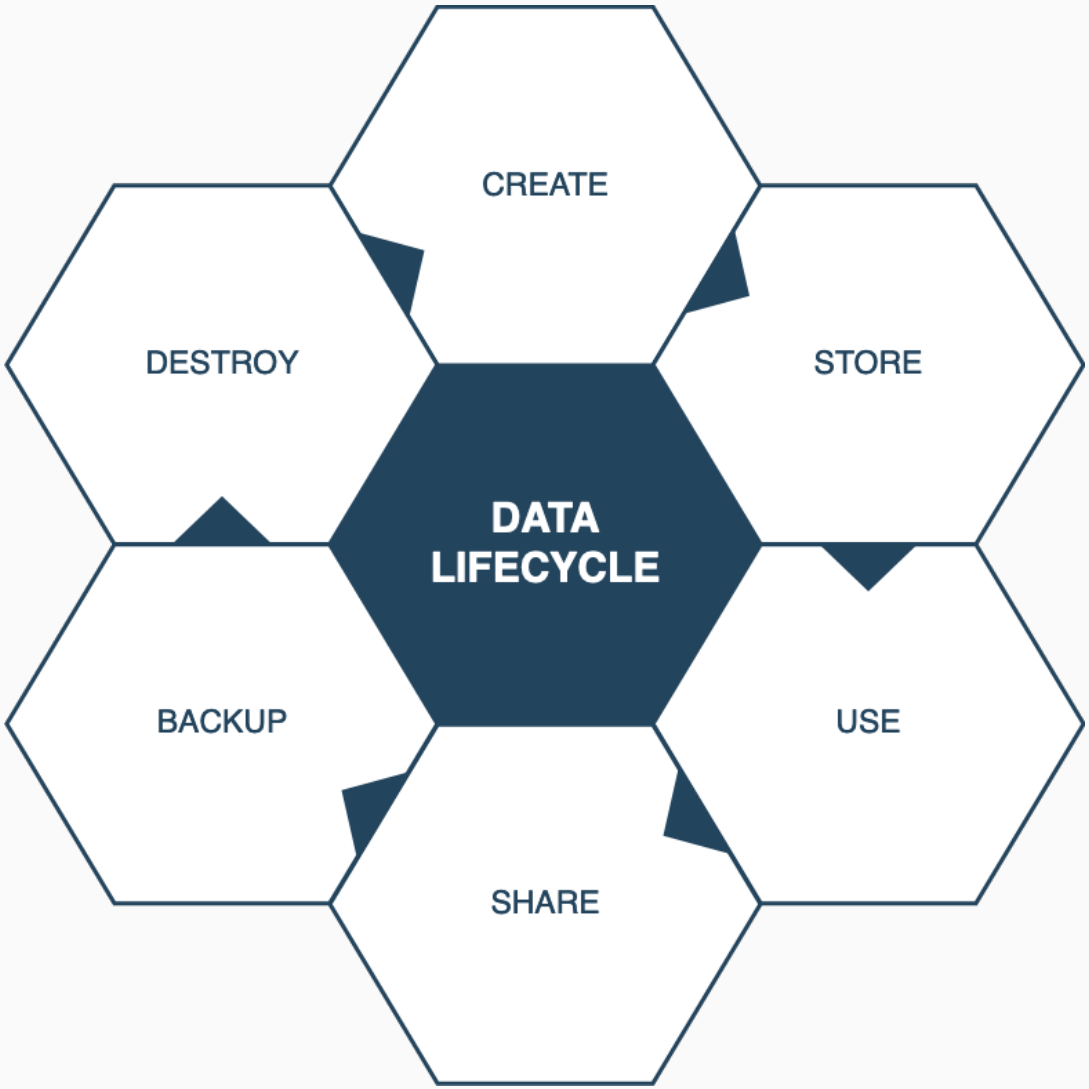
- Data security protects data from malicious threats: activity monitoring, network security, access control, encryption, authentication.
- Data privacy addresses proper handling, processing, storage of data: security policies and permissions.

In order to ensure privacy, we need security.

Data privacy and security considerations

- Provide lifecycle management,
- data transfers restricted and allowed over secure channels,
- restrict access to data (ACLs, firewall, authN, authZ)
- provide backup and replication,
- encryption and key management (on AWS, newly addedd resource will be terminated if encryption is not enabled),
- least privilege concept enforced,
- obscure raw data and only display selected portions during operations,
- apply SIEM, FIM.

Data lifecycle



Virtualisation security

Virtualisation and cloud

- **virtualisation is a technology**: it allows creating multiple environments from a single, physical hardware system
- **cloud is an environment**: it can include bare-metal, virtualisation, or container software

Why does cloud security matter?

- hypervisors are prime targets of attacks (single point of failure),
- if hypervisor host is vulnerable, everything else on it is vulnerable,
- VMs can interfere with each other,
- resources and services are difficult to track,
- lack of knowledge of technical staff,
- data is sparsed on multiple servers and locations,
- all security risks present in traditional infrastructure are also present here.

Virtualisation security essentials

- don't use default credentials,
- don't mix production and development VMs on the same hypervisor, use different network or at least, different security group for production and development,
- use different credentials for production and development VMs,
- monitor all VMs (production, testing, development),
- shut down VMs that you don't need,
- always update offline VMs before putting them back online,
- maintain inventory of VMs,
- check for open ports, default passwords, unpatched software (nmap, Metasploit, OpenVAS, Nessus) - check also `https://github.com/dev-sec/puppet-os-hardening`

Cloud services

Consider the benefits of running services in the cloud.

- What are your risks?
- What are your responsibilities?
- Which domains are under your control and which in the hands of the cloud provider?
- Where will you store your data and how will you transfer it, use it?
- Are there any regulations about storing the data in the cloud?

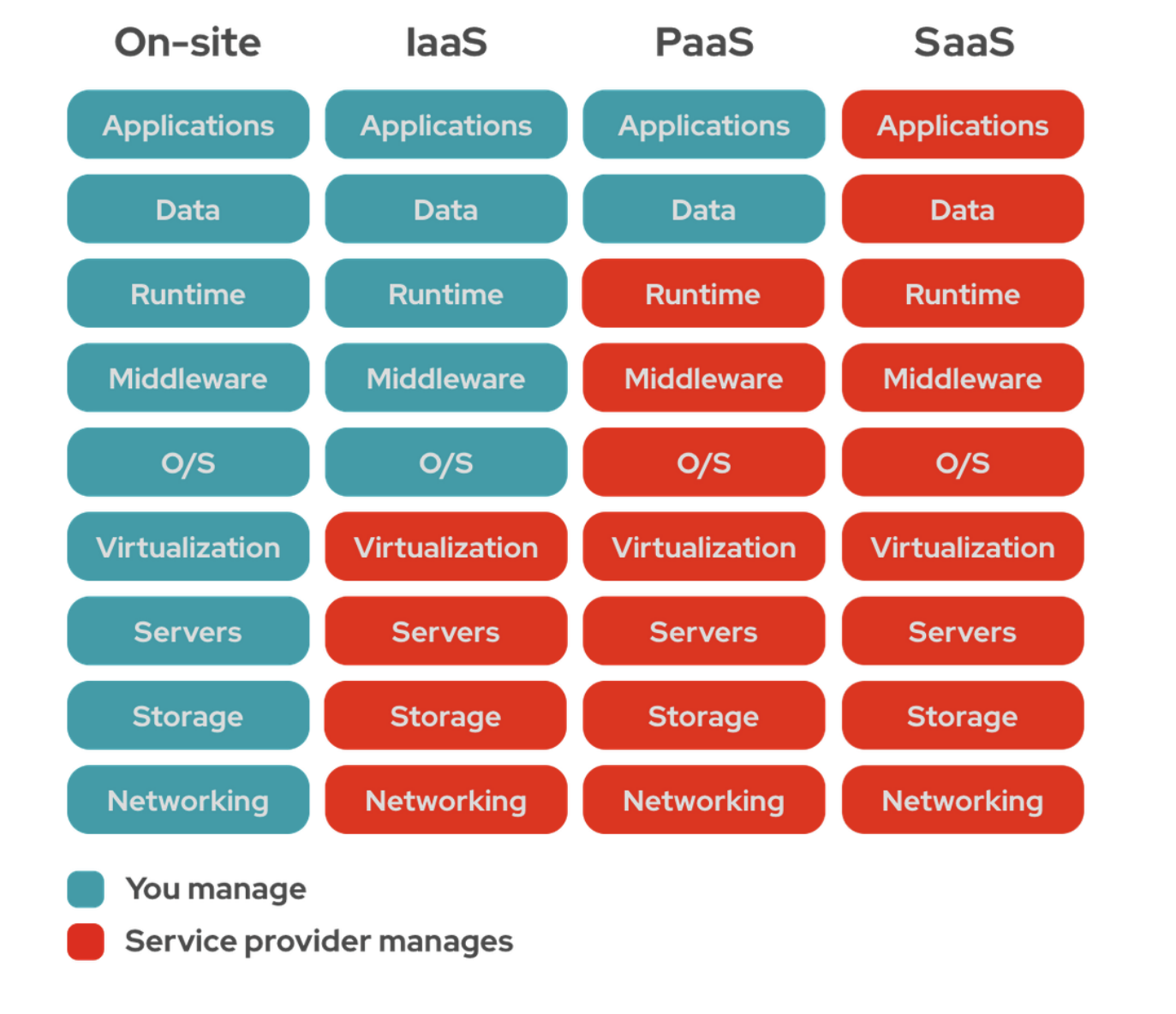
Cloud security challenges

- **for customer**: no longer access to the hypervisor or hardware (physical, host security), cannot control which customers host on the same host and how well they protect their VMs
- **for cloud provider**: complex network designs and no control over the state of VMs

Private vs private cloud

- **Private cloud**:
  - security is a responsibility of the organisation,
  - number of VMs is pretty stable,
  - scalability is limited,
  - bandwidth is limited,
  - data storage and access under control of the organisation,
  - potential of providing perfectly safe environment (behind a firewall).
- **Public cloud**:
  - shared responsibility between customer and cloud provider,
  - seemingly infinite resources,
  - main target for security attacks (security is big investment),
  - no control over data for customer,
  - customer needs to trust cloud provider.

Cloud models



Common threats in the cloud

- cyber attacks: DoS, spoofing, man-in-the-middle,
- escalation of privileges, unauthorized access,
- hijacking accounts,
- misconfigurations,
- internal/external threats,
- malware,
- data breaches,
- insecure interfaces/APIs,
- external data sharing and data transfers,
- insufficient tehnical skills,
- VM escape,
- leaked credentials (commited to git).

How to prevent common attacks?

- **Spoofing**: use SSH keys for authentication, TLS for communication, strong pasword policy, link Keystone with LDAP directory
- **Tampering**: use digital signatures for data integrity (Glance supports image signing), mandatory access control (MAC) and role based access control (RBAC) to protect services
- **Repudation**: central logging and auditing in place, SIEM, monitor networks of anomalies (IDS/IPS)
- **Data disclosure**: use encryption, MAC/RBAC
- **DoS**: redundant services (HA), use quotas per domain/project/user, isolate services from direct access, use proxy to access services from DMZ, good network design
- **Escalation of privileges**: MFA, restrict API, monitor

Questions?

References

- Aditya K. Sood: Empirical Cloud security, Mercury Learning
- Joseph Migga Kizza: Guide to Computer Network Security, Springer
- Silvano Gai: Building a future-proof Cloud Infrastructure
- Vickler Andy: Linux Security and Administration
- Chris Anley and other:The Shellcoder's Handbook Discovering and Exploiting Security Holes, Wiley Publishing
- Shuangbao Paul Wang: Computer Architecture and Organization, Springer
- Sean-Philip Oryano: CEH v9 certified ethical hacker study guide, Sybex

References (2)

- Kevin Mitnick: The art of deception - Controlling the Human Element of Security, Wiley
- Bruce Schneier: Secrets and Lies, Digital Security in a Networked World, Wiley
- Heather Adkins and other: Building Secure and Reliable Systems, O'Reilley
- Musaab Hasan, Zayed Balbahaith: Mastering Linux Security, Lambert Academic Publishing
- Thomas Limoncelli: The practice of System and Network administration
- Daniel Regalado and all: Gray Hat Hacking, McGraw Hill Education
- Donald A. Tevault : Mastering Linux Security and Hardening, Packt Publishing

References (3)

- James Turnbull: Hardening Linux, APress
- NIST NCP: `https://ncp.nist.gov/repository`
- CIS benchmarks: `https://www.cisecurity.org/cis-benchmarks/`
- CIS controls: `https://www.cisecurity.org/controls`
- How to secure anything, `https://github.com/veeral-patel/how-to-secure-anything`
- JISC cyber report 2022, `https://repository.jisc.ac.uk/8732/1/cyber-impact-report-2022.pdf`

References (4)

- Aditya K. Sood: Empirical Cloud security, Mercury Learning
- Chris Dotson: Practical Cloud security, O'Reilly Media
- Fabio Alessandro Locati: Openstack cloud security, Packt Publishing
- Ben Malisow: CCSP Certified Cloud Security Professional Official Study Guide, Sybex
- Silvano Gai: Building a future-proof Cloud Infrastructure
- Chris Binnie, Rory McCune: Cloud Native Security, Wiley Publishing
- Ben Silverman and Michael Solberg: OpenStack for Architects, Packt Publishing
- Donald A. Tevault : Mastering Linux Security and Hardening, Packt Publishing
- James Turnbull: Hardening Linux, APress