1.1 Security Controls

Monday, 29 July 2024 8:49 PM

Security controls help prevent security events to help minimise impact and limit damage done by threat actors. Assets to cons ider are data, physical property, computer systems and more.

Technical Controls: Controls implemented using systems
- Operating system control
- Firewalls, anti-virus, etc.

Managerial Controls: Administrative controls associated with security design and implementation
- Security policies, standard operating procedures.
- Governance, policies, procedures and strategies for managing risk within an organisation
- LEGAL !! !!

Operational Controls: Controls implemented by people instead of systems
- Security guards, awareness programs
- Guard Shack

Physical Controls: Limits physical access
- Guard shack can be physical or operational
- Fences, locks
- Badge readers

Categories	Preventive	Deterrent	Detective	Corrective	Compensating	Directive
Technical	Firewall	Splash Screen	System logs	Backup recovery	Block instead of patch	File Storage policies
Managerial	On-boarding policy	Threat of demotion	Review login reports	Policies for reporting issues	Separation of duties	Compliance policies
Operational	Guard shack	Front Reception Desk	Property patrols	Contacting law enforcement	Guard duties	Security Policy Training
Physical	Door lock	Posted Warning Sign	Motion detectors	Fire Extinguisher	Backup generator	Signs: Authorised personnel only

These are not inclusive lists: There are many categories of controls, some organisations will combine types.
There are multiple security controls for each category and type: Some security controls may exist in multiple types or categories, new security controls are created as systems and processes evolve. Your organisation may use very different controls.

Preventive: Blocks access to a resource
Prevent access:
- Firewall rules
- Follow security policy
- Guard shack checks all identification
- Enable door locks

Deterrent: Discourage an intrusion attempt, does not directly prevent access
Makes an attacker think twice:
- Application splash screens
- Threat of demotion
- Front reception desk
- Posted warning signs

Detective: identify and log an intrusion attempt, may not prevent access.
Find the issue:
- Collect and review system logs
- Review login reports
- Regularly patrol the property
- Enable motion detectors

Corrective: Apply a control after an event has been detected, reverse the impact of an event and continue operating with minimal downtime .
Correct the problem:
- Restoring from backups can mitigate a ransomware infection
- Create policies for reporting security issues
- Contact law enforcement to manage criminal activity
- Use a fire extinguisher

Compensating: Control using other means, existing controls aren't sufficient, may be temporary.
Prevent the exploitation of a weakness:
- Firewall blocks a specific application instead of patching the app
- Implement a separation of duties
- Require simultaneous guard duties
- Generator used after power outage

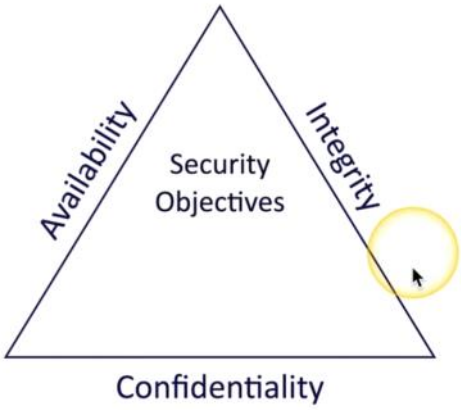
Directive: Direct a subject towards security compliance, a relatively weak security control.

Do this, please:

- Store all sensitive files in a protected folder
- Create compliance policies and procedures
- Train users on proper security policy
- Post a sign for "Authorised Personnel Only"

1.2 The CIA Triad

Tuesday, 30 July 2024 2:23 AM



The CIA Triad is an easy way to remember the **fundamentals** of I.T Security.

Confidentiality:
- Prevent disclosure of information to unauthorised individuals or systems.

Integrity:
- Prevent disclosure of information to unauthorised individuals or systems.

Availability:
- Systems and networks must be up and running.

Diving deeper into the above:

Confidentiality: Certain information should only be known to certain people.
- Prevent unauthorised information disclosure.
- **Encrypt** messages so only certain people can read it
- Selectively restrict access to a resource using access controls.
- Two factor authentication.

Integrity: Data is stored and transferred as intended.
- Any modification to the data would be identified.
- Hashing ensures integrity by producing a unique hash value for data, the value will change if the data is altered in any way.
- Digital signatures takes a hash an encrypts it with an asymmetric encryption algorithm to further verify the integrity of the data.
- Certificates combined with digital signatures will further ensure verification.
- Non-repudiation provides proof of integrity, can be asserted to be genuine.

Availability: Information is accessible to authorised users.
- Redundancy, build services that will always be available.
- Fault tolerance, to ensure that a system will continue to run even when a failure occurs.
- Patching, ensures the systems are always as stable as possible and be up to date with security holes to reduce the changes of an exploit.

1.2 Non-repudiation

Tuesday, 30 July 2024 4:49 AM

You can't deny what you've said. It can be compared to signing a contract, your signature adds non-repudiation, others can verify your signature.

Proof of integrity: Verify data does not change. The data remains accurate and consistent.

In Cryptography, we can use a hash.

- Represents data as a short string of text
- A message digest, a fingerprint.
- If the data changes, the hash changes.
- Doesn't necessarily associate data with an individual, it will only tell you if the data has changed.

Proof of origin:

Integrity

- Prove the message was not changed

Authentication

- Prove the source of the message

Non-repudiation

- Make sure the signature isn't fake

Asymmetric Encryption

- Sign with the private key and an associated public key.

Creating a digital signature:

1. The sender uses their private key to encrypt (sign) a hash of the message, this create a digital signature.
2. The recipient uses the sender's public key to decrypt the digital signature. If the decrypted hash matches the hash of the received message, it verifies that the message was sent by the holder of the private key and that it has not been altered.

This method is used to ensure authenticity and integrity of the message, but not confidentiality.

1.2 AAA Framework: Authentication, Authorisation and Accounting

Tuesday, 30 July 2024 2:23 AM

A security model used to control access to compute resources, ensure secure communication and manager user identities.

Identification: This is who you claim to be, usually your username

Authentication: To verify the identity of a user or system. It requires users to provide credentials to confirm their identity.
- Password login
- Biometric scanning
- Multi-factor authentication.

Authorisation: Based on your identification and authentication, what access do you have
- Access Control Lists (ACL),
- Role-based access controls (RBAC),
- Attribute-based access control (ABAC).

Accounting: To track user activities and resource usage. It involves logging and monitoring user actions to create an audit trail.
- Logging access attempts
- Tracking user actions
- Usage reports.

An organisation has a trusted Certificate Authority (CA). Most organisations maintain their own Cas.

The organisation will create a certificate for a device and it is further used as an authentication factor to verify that it really was digitally signed by the CA.

Authorisation Models:

The user or device has now authenticated, we need to apply an authorisation model now. Associating individual users access rights does not scale well, using an authorisation model to define roles, organisations, attributes and other types of characteristics is more effective.

The use of an authorisation model will help reduce complexity and create a clear relationship between the user and the resource. Instead of mapping every individual user to an **abstraction (group),** it is easier to allocate permissions to them all.

1.2 Gap Analysis

Tuesday, 30 July 2024 2:23 AM

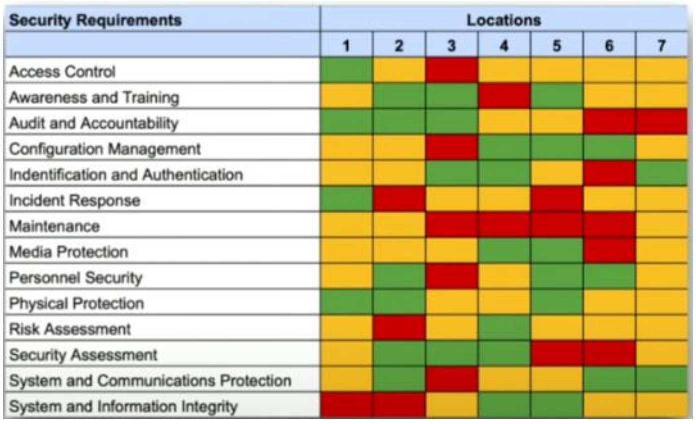
Where you are compared to where you want to be.
The "gap" between the two.

A baseline will give you something to work towards, and an idea on where the goal should be. Also categorised as an internal set of goals.

Compare and Contrast:

- Examine the current processes
- Research existing IT systems
- Evaluate existing security policies
- Identify weaknesses, along with the most effective processes

A detailed analysis can help examine broad security categories and break those into smaller, more manageable segments.



Security Requirements	Locations						
	1	2	3	4	5	6	7
Access Control	🟩	🟨	🟥	🟨	🟨	🟨	🟨
Awareness and Training	🟨	🟩	🟩	🟥	🟩	🟨	🟨
Audit and Accountability	🟩	🟩	🟩	🟨	🟨	🟥	🟥
Configuration Management	🟨	🟨	🟥	🟩	🟩	🟩	🟨
Indentification and Authentication	🟨	🟨	🟩	🟩	🟨	🟥	🟩
Incident Response	🟩	🟥	🟨	🟨	🟥	🟨	🟨
Maintenance	🟨	🟨	🟥	🟥	🟥	🟥	🟨
Media Protection	🟨	🟨	🟨	🟩	🟩	🟥	🟨
Personnel Security	🟨	🟩	🟥	🟨	🟩	🟩	🟨
Physical Protection	🟩	🟩	🟨	🟨	🟩	🟨	🟨
Risk Assessment	🟨	🟥	🟨	🟩	🟨	🟨	🟨
Security Assessment	🟨	🟩	🟩	🟩	🟥	🟥	🟨
System and Communications Protection	🟨	🟩	🟥	🟨	🟨	🟩	🟩
System and Information Integrity	🟥	🟥	🟨	🟩	🟩	🟨	🟨

The analysis report: The final comparison.

- Detailed baseline objectives
- A clear view of the current state
- "How to get from where you are, to where you want to be"

1.2 Zero Trust Security

Tuesday, 30 July 2024 2:23 AM

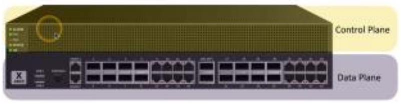
Many networks are relatively open on the inside.

Zero Trust is a holistic approach to network security, covers every device, every process and every user on the network.
- **Never trust, always verify - Micro segmentation and least privilege access principles**
- Everything must be verified
- Nothing is inherently trusted
- Multi-factor authenticatino, encryption, system permissions, additional firewalls, monitoring and analytics, etc.

Planes of operation: Split the network into functional planes
- Applies to physical, virtual and cloud components.

Data plane: The part of the device performing the security process, a switch, router or firewall. Tools that help move data from one part of the network to another.
- Process the frames, packets and network data
- Processing, forwarding, trunking, encrypting, NAT.

Control plane: This is where we manage all of the actions occurring in the data plane.
- Define policies and rules
- Determine how packets should be forwarded
- Routing tables, session tables, NAT tables.



Adaptive Identity: A dynamic approach to managing user identities and access privileges within an IT environment.
- **Behavioural Analysis**
  Monitoring and analysing user behaviour to detect anomalies that may indicate potential security risk.
- **Risk-Based Authentication:**
  Adjusting authentication requirements based on the assessed risk level, such as requiring multi -factor authentication for high-risk activities.
- **Contextual Awareness:**
  Considering various factors such as the users location, device, and time of access to make real -time access decisions
- **Continuous Monitoring:**
  Continuously assessing user activities to update risk profiles and dynamically adjust access controls.

Threat Scope Reduction:
- Decrease the number of possible entry points

Policy-Driven access control
- Combine the adaptive identity with a predefined set of rules.

Security Zones: Security is more than a one-to-one relationship.

Where are you coming from and where are you going.
- Trusted, untrusted
- Internal network, external network
- VPN 1, VPN 5, VPN 11
- Marketing, IT, accounting, human resources.

Using Zones may be enough by itself to deny access.
- For example, **Untrusted** to **Trusted** zone traffic

Some zones are implicitly trusted:
- For example **Trusted** to **Internal** zone traffic.

Policy Enforcement Point: Subjects and systems, end-users, applications and non-human entities.
- A gatekeeper, all traffic must pass through this point to decide if the traffic is allowed through the PEP or not.

Allow, monitor, and terminate connections.

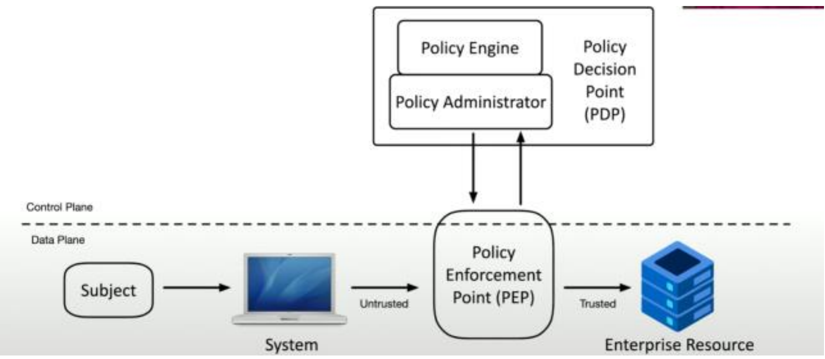
Applying trust in the planes:

Policy decision point:
- There's a process for making an authentication decision.

Policy Engine:
- Evaluates each access decision based on policy and other information sources
- Grant, deny or revoke.

Policy Administrator:
- Communicates with the policy enforcement point
- Generates access tokens or credentials
- Tells the PEP to allow or disallow access.

Zero Trust Across Planes



1.2 Physical Security

Tuesday, 30 July 2024 2:23 AM

Barricades / bollards: Prevent access

- **Channel people through a specific access point**
- Keeps other things out
- Allow people, prevent cars and trucks
- Identify safety concerns and prevent injury
- Can be used to an extreme: concrete barriers/bollards, moats.

Access Control Vestibules: A room you must pass through to gain access to the rest of the building.

- All doors are normally unlocked, opening one door causes others to lock
- Or all doors are normally locked, unlocking one door prevents others from being unlocked
- **One door open/other locked**. **When one is open, the other cannot be unlocked.**
- One at a time, controlled groups, managed through an area.
- **They are designed to allow or control access through a particular area.**

Fencing: Builds a perimeter, usually very obvious.

- Transparent or opaque, see through the fence (or not)
- **Robust**
- **Prevent climbing**, razor wire or built high.

Video Surveillance: CCTV, can be used to replace physical guards.

- Camera features are important, motion recognition to alarm and alert when something moves
- **Object detection** can identify a license plate or persons face
- **Networked together and recorded over time**

Guards and access badges:

- **Physical protection** at the reception area of a facility
- Validates identification of existing employee
- **Two-person integrity**/control to minimise exposure to an attack, no single person has access to a physical asset.
- **Access badge** to provide a picture, name or other details, it must be worn at all times and is electronically logged

Lighting: More light means more security.

- Attackers avoid the light, Non IR cameras can see better.
- Specialised design for better overall light, lighting angles may be important for facial recognition.

Sensors: infrared, detects infrared radiation in both light and dark areas, common in motion detectors.

- **Pressure**, detects a change in force, floor and window sensors
- **Microwave** detects movement across a large area.
- **Ultrasonic** signals receive reflected sound waves, it will detect motion and collision detection, etc.

1.2 Deception and Disruption

Tuesday, 30 July 2024 2:23 AM

Honeypots: Attracts the bad guys and trap them there.

- Helps you see what type of attacks are being used and what type of systems are they trying to attack.
- Virtual world that attracts automated systems/attackers.

Honeynets: A real network including more than a single device.

- Servers, workstations, routers, switches, firewalls, etc.
- A larger deception network with one or more honeypots in hopes to keep attackers interested.

Honeyfiles: Attract attackers with fake files/information.

- Something bright and shiny
- Bait for the honeynet (passwords.txt)
- Add many honeyfiles to file shares.
- An alert is sent if the file is accessed, a virtual bear trap.

Honeytokens: Traceable data to track the malicious actors.

- Adds traceable data to the honeynet
- If the data is stolen youll know where it came from.
- API credentials that do not actually provide access but a notification is sent to you when they're used.
- Fake email addresses, constantly monitor the internet to see who posts it.
- Honeytokens can be any type of data, database records, browser cookies, web page pixels, etc.

1.3 Change Management

Wednesday, 31 July 2024 12:11 AM

How to make a change: You need to go through a formal process to make sure the change will work properly.

- Upgrade software,
- Patch an application,
- Change firewall configuration,
- Modify switch ports

One of the most common risks in the enterprise:

- Occurs very frequently
- A system that is less updated, is probably less secure
- Often overlooked or ignored.

Have clear policies:

- Frequency, duration, installation process, rollback procedures.

Sometimes extremely difficult to implement:

- It's hard to change corporate culture

Change Approval Process: A formal process to managing change to maintain the uptime and availability of our systems and that everyone is informed.

- Avoid downtime, confusion and mistakes

A typical approval process:

1. Complete the request forms
2. Determine the purpose of the change
3. Identify the scope of the change
4. Schedule a date and time of the change
5. Determine affected systems and the impact
6. Analyse the risk associated with the change
7. Get approval from the change control board
8. Get end-user acceptance after the change is complete.

Ownership: Involves assigning responsibility for managing and overseeing the change process to specific individuals or groups.

- Clear accountability
- Decision-making authority
- Leadership and Guidance

Stakeholders:

- Individuals or groups affected by or having an interest in the change
- Support and buy in
- Providing input and feedback
- Managing resistance, addressing concerns early
- Maintaining communication

Impact Analysis:

- Assess how the change will affect various aspects of the organisation
- Identify risk
- Evaluating effects on business processes
- Planning resource allocation
- Setting expectations

Determine a risk value:

- High, medium or low

The risks can be minor or far-reaching:

- The "fix" doesn't really fix anything
- The fix breaks something else
- Operating system failures
- Data corruption

What's the risk with not making the change?

- Security vulnerability
- Application unavailability
- Unexpected downtime to other services

Test results: Sandbox Testing Environment

- You can perform as many tests as you'd like and you'd have no effect on production systems.
- A technological safe space where you can make mistakes, try different techniques and perform extensive testing after the fact to see if the update worked correctly.
- Try the upgrade and apply the patch
- Test and confirm before deployment
- Confirm the backout plan, mov everything back to the original, a sandbox can't consider every possibility.

Backout Plan: You should always have a way to revert your changes, prepare for the worst, hope for the best.

- Always create backups before making any changes to a system.

Maintenance window: Finding time to implement a change

- When is the change happening?
- During the workday may not be the best option. Potential downtime would affect a large part of production.
- Overnights are often a better choice. (Especially challenging for 24-hour production schedules)
- The time of the year may be a consideration, retail networks are frozen during the holiday season.

Standard Operating Procedure: Change management is critical, it affects everyone in the organisation,

- The process must be well documented
- Should be available on the intranet with all standard processes and procedures
- Changes to the process are reflected in the standard.

1.3 Technical Change Management

Wednesday, 31 July 2024 12:11 AM

Put the change management process into action: Execute the plan
- There's no such thing as a simple upgrade, can have many moving parts
- Separate evens may be required
- Change management is often concerned with "what" needs to change.
- The technical team is concerned with "how" to change it

Allow/Deny list: Any application can be dangerous.
- Vulnerabilities
- Trojan Horses
- Malware
- Security Policy can control app execution, allow/deny list.

Allow List:
- Nothing runs unless it's approved
- Very restrictive

Deny List:
- Nothing on the "bad list" can be executed
- Antivirus, anti-malware

Restricted Activities:
- The scope of a change is important, defines exactly which components are covered
- A change approval isn't permission to make any change, the change control approval is very specific
- The scope may need to be expanded during the change window, it's impossible to prepare for all possible outcomes
- The change management process determines the next steps, there are processes in place to make sure the change is successful.

Downtime: Services will eventually be unavailable.
- The change process can be disruptive
- Usually scheduled during non-productive hours
- For systems working **24/7**, try to prevent any downtime by switching to a **secondary system**, upgrade the primary, then switch back.
- Minimise any downtime events, the process should be as automated as possible, switch back to secondary system if issues appear, should be part of the backout plan.
- Send emails and calendar updates.

Restarts: It's common to require a restart
- Implement the new configuration
- Reboot the OS, power cycle the switch, bounce the service
- Can the system recover from a power outage?

Legacy Applications: Some applications were here before you arrived.
- They'll be here when you leave
- Often no longer supported by the developer, you're not the support team.
- Fear of the unknown, face your fears and document the system. It may not be as bad as you think!
- May be quirky, create specific processes and procedures. You'll become the expert!!

Dependencies: To complete A, you must complete B.
- A service will not start without other active services
- An application requires a specific library version
- Modifying one component may require changing or restarting other components, this can be challenging to manage
- Dependencies may occur across systems, upgrade the firewall code first, then upgrade the firewall management software.

Documentation: It can be challenging to keep up with changes

- Documentation can become outdated very quickly
- Require with the change management process
- Updating diagrams, modifications to network configurations, address updates
- Updating policies/procedures, adding new systems may require new procedures

Version Control: Track changes to a file or configuration data over time

- Easily revert to a previous setting
- Many opportunities to manage versions, router configurations, Windows OS patches, application registry entries.

1.4 Public Key Infrastructure (PKI)

Wednesday, 31 July 2024 12:11 AM

Policies, procedures, hardware, software, people:

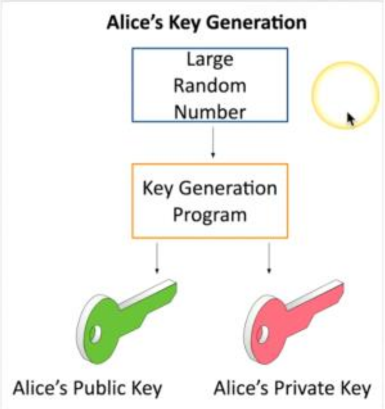
- Responsible for creating, distributing, managing, storing and revoking digital certificates.
- This is a big endeavour and requires a lot of planning.
- PKI is also referred to the binding of public keys to people or devices, the certificate authority. It's all about trust

Symmetric Encryption: A single, shared key

- Encrypt with the key
- Decrypt with the same key
- If it gets out, you'll need a new key
- Secret key algorithm, a shared secret
- Doesn't scale very well, challenging to distribute.
- Difficult above 10 or more members.
- Very fast to use, less overhead than asymmetric encryption, often combined with asymmetric encryption

Asymmetric Encryption: Two or more mathematically related keys

- Private key, kept private
- Public key, anyone can see this key
- The private key is the only key that can decrypt data encrypted with the public key. You can't derive the private key from the public key.



The Key Pair: Asymmetric encryption, Public Key Cryptography

- Everyone can have the public key, only Alice has access to the private key.

Key Generation:

- Build both the public and private key at the same time
- Lots of randomisation
- Large prime numbers
- Lots and lots of maths

Key Escrow: Someone else holds your decryption keys

- Your private keys are in the hands of a 3rd-party
- This may be within your own organisation
- This can be a legitimate business arrangement, a business might need access to employee information.
- Government agencies may need to decrypt partner data.

Handing your private key to someone else may seem controversial, however it is sometimes necessary.

1.4 Encrypting Data

Wednesday, 31 July 2024 3:29 AM

Encrypting stored data, protecting data on storage devices.

- SSH, hard drive, USB drive, cloud storage, etc.
- This is data at rest

Full-disk and partition/volume encryption

- BitLocker, FileVault, etc.

File Encryption:

- EFS (Encrypting File System)
- Many third party utilities perform the same result.

Database Encryption: Protecting stored data, and the transmission of that data.

- **Transparent encryption**, encrypts all database information with a symmetric key
- **Record-level encryption**, encrypt individual columns, use separate symmetric keys for each column
- **Transport encryption,** protect data traversing the network. Browsers communicate using HTTPS
- **VPN (Virtual Private Network),** Encrypts all data transmitted over the network, regardless of the application. Client-based VPN using SSL/TLS, site-to-site VPN using IPsec.

Encryption Algorithms: There are many different ways to encrypt data, the same formula must be used during encryption and decryption.

- Both sides decide on the algorithm before encrypting data, the details are often hidden from the end-user.
- There are advantages and disadvantages between algorithms: Security level, speed, complexity of implementation.

Cryptographic Keys: There is very little that isn't known about the cryptographic process.

- The algorithm is usually a known entity
- The only thing you don't know is the key
- The key determines the output, encrypted data, hash value, digital signature
- Similar to a door lock

Key Lengths: Larger keys tend to be more secure

- Prevent brute-forc attacks
- Attackers can try every possible key combination

Symmetric Encryption:

- 128-bit or larger symmetric keys are common
- These numbers get larger and larger as time goes on.

Asymmetric Encryption:

- Complex calculations of prime numbers
- Larger keys than symmetric encryption
- Common to see key lengths of 30,72 bits or larger.

Key Stretching: Make a weak key stronger by performing multiple processes

- Hash a password, hash the hash of the password, and continue.
- Brute-force attacks would require reversing each of those hashes, they'd have to spend much more time to decrypt.

1.4 Key Exchange

Wednesday, 31 July 2024 3:29 AM

A logistical challenge, how do you share an encryption key across an insecure medium without physically transferring the key?

Out-of-band key exchange: Don't send the symmetric key over the net.

- Telephone
- Courier
- In-person, etc.

In-band key exchange: On the network

- Protect the key with additional encryption.
- Use asymmetric encryption to deliver a symmetric key

Real-time encryption/decryption: Share a symmetric session key using asymmetric encryption

- Client encrypts a random (symmetric) key with a server's public key
- The server decrypts this shared key and uses it to encrypt data
- This is the session key

Symmetric key from asymmetric keys:

- Use public and private key cryptography to create a symmetric key

1.4 Encryption Technologies

Wednesday, 31 July 2024 3:29 AM

Trusted Platform Module (TPM), a specification for cryptographic functions.

- Cryptography hardware on a device
- Cryptographic processer, random number generator, key generators
- Persistent memory, unique keys burned in during manufacturing
- Versatile memory, storage keys, hardware configuration information, securely store BitLocker keys
- Password protected, no dictionary attacks.

Hardware Security Module (HSM): Securely store thousands of cryptographic keys

- Used in large environments
- Clusters, redundant power
- High-end cryptographic hardware, plug-in card or separate hardware device
- Key backup, secure storage in hardware.
- Cryptographic accelerators, offload that CPU overhead from other devices

Key Management System: On premises, cloud-based, many different keys for many different services

- Manage all keys from a centralised manager, separate the encryption keys from the data.
- Create keys for specific service or cloud provider, (SSL/TLS, SSH, etc.)
- Associate keys with specific users, rotate keys on regular intervals
- Log key use and important events.

Keeping data private: our data is located in many different places

- Mobile phones, cloud, laptops, etc.
- The most private data is often physically closest to us
- Attackers are always finding new techniques, it's a race to stay one step ahead
- Data is changing constantly, we need to keep the data protected.

Secure Enclave: A protected area for our secrets

- Often implemented as a hardware processor
- Isolated from the main processor
- Many different technologies and names
- Provides extensive security features
- Has its own boot ROM
- Monitors the system boot process
- True random number generator
- Real-time memory encryption
- Root Cryptographic keys
- Performs AES encryption in the hardware device AND MORE

1.4 Obfuscation

Wednesday, 31 July 2024 3:29 AM

The process of making something easy to understand to making it unclear.

- It's now much more difficult to understand
- If you know how the obfuscation is done, you'll be able to gain access to the data.
- The idea is you're hiding information in plain site, only if you know how it's hidden, you'll recognise there's data hidden within that object.
- Hiding information inside an image is known as **Steganography.**

Steganography: Greek for "Concealed writing."

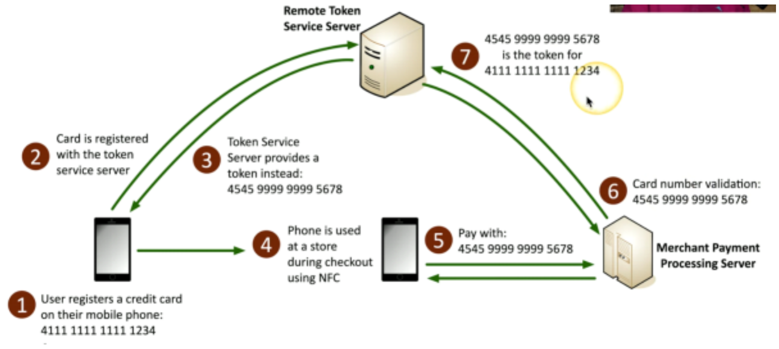
- Security through obscurity.
- Message is invisible but it's stored within the data contained with the image.
- The **covertext,** the container document or file.

Common steganography techniques:

- **Network based,** embed message in TCP packets.
- **Use an image,** embed the message in the image itself
- **Invisible watermarks,** yellow dots on printers (machine identification code).
- **Audio steganography,** modify the digital video file, interlace a secret message within the audio. Similar technique to image steganography.
- **Video Steganography,** a sequence of images, use image steganography on a larger scale, manage the signal to noise ratio, potentially transfer much more information.

Tokenisation: Replace sensitive data with a non-sensitive placeholder.

- SSH 266-12-1112 is now 691-61-8539. We are able to match the two together behind the scenes and transfer it across the network and on the other side it will make that switch to what the actual number might be.
- An attacker capturing the card numbers can't use them later
- This isn't encryption or hashing, the original data and token aren't mathematically related.


Remote Token Service Server
7 4545 9999 9999 5678 is the token for 4111 1111 1111 1234
2 Card is registered with the token service server
3 Token Service Server provides a token instead: 4545 9999 9999 5678
6 Card number validation: 4545 9999 9999 5678
4 Phone is used at a store during checkout using NFC
5 Pay with: 4545 9999 9999 5678
Merchant Payment Processing Server
1 User registers a credit card on their mobile phone: 4111 1111 1111 1234

Data masking: Data obfuscation, hide some of the original data.

- Protects PII and other sensitive data
- May only be hidden from view, the data may still be intact in storage. Control the view based on permissions.
- Many different techniques, substituting, shuffling, encrypting, masking out, etc.

1.4 Hashing and Digital Signatures

Wednesday, 31 July 2024 3:47 AM

Hashes: The process that transforms input data of any size into a fixed-length string of characters.

Represent data as a short string of text
- A message digest
- Fingerprint

One-way Trip:
- Impossible to rceover the original message from the digest
- Used to store passwords, confidentiality

Verify a downloaded document is the same as the original:
- Integrity

Can be a digital signature:
- Authentication
- Non-repudiation
- Integrity

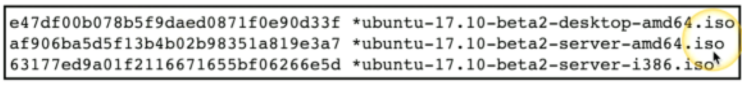
Collision:

Hash Functions:
- Take an input of any size
- Create a fixed size string
- Message digest, checksum

The hash should be unique:
- Different inputs should never create the same hash
- If they do, it's a collision

MD5 has a collision problem:
- Found in 1996
- Don't use MD5 for anything important

Practical Hashing: Verify a downloaded file
- Hashes may be provided on the download site
- Compare the downloaded file hash with the posted hash value



```
e47df00b078b5f9daed0871f0e90d33f *ubuntu-17.10-beta2-desktop-amd64.iso
af906ba5d5f13b4b02b98351a819e3a7 *ubuntu-17.10-beta2-server-amd64.iso
63177ed9a01f2116671655bf06266e5d *ubuntu-17.10-beta2-server-i386.iso
```

Password Storage:
- Instead of storing the password, store a salted hash
- Compare hashes during the authentication process
- Nobody ever knows your actual password

Adding some **Salt:** Salt, random data added to a password when hashing
- Every user gets their own random salt, the salt is commonly stored with the password.
- Rainbow table won't work with salted hashes, additional random value added to the original password.
- This slows things down for the brute force process. It doesn't completely stop the reverse engineering.

Digital Signatures:

Proves the message was not changed:
- Integrity

Proves the source of the message:
- Authentication

Make sure the signature isn't fake:
- Non-repudiation

Sign with the private key:
- The message doesn't need to be encrypted
- Nobody else can sign this

Verify with the public key:
- Any change in the message will invalidate the signature

1.4 Blockchain Technology

Wednesday, 31 July 2024 3:47 AM

A distributed ledger, keep track of transactions.

- Everyone on the blockchain network maintains the ledge
- Records and replicates to anyone and everyone
- There are many practical applications, payment processing, digital identification, supply chain monitoring or digital voting.

The blockchain process:

1. The transaction is requested. (The transaction could be any digital transaction from transferring bitcoins, medical records, data backups or transferring house title information.)
2. The transaction is sent to every computer (or node) in a decentralised network to be verified.
3. The verified transaction is added to a new block of data containing other recently verified transactions.
4. A secure code (hash) is calculated from the previous blocks of transaction data in the blockchain. The hash is added to the new block of verified transactions.
5. The block is added to the end of the blockchain, which I then updated to all nodes in the network for security.
6. If any blocks are altered, its hash and all following hashes in the chain are automatically recalculated, the altered chain will no longer match the chains stored by the rest of the network and will be rejected.

1.4 Certificates

Wednesday, 31 July 2024 3:47 AM

A public key certificate: Binds a public key with a digital signature and other details about the key holder.

A digital signature adds trust, PKI uses certificate authorities for additional trust. **Web of Trust** adds other users for additional trust.

Certificate creation can be built into the OS
- Part of windows domain services
- Many 3rd party options

What's in a digital certificate?
X.509
- Standard format

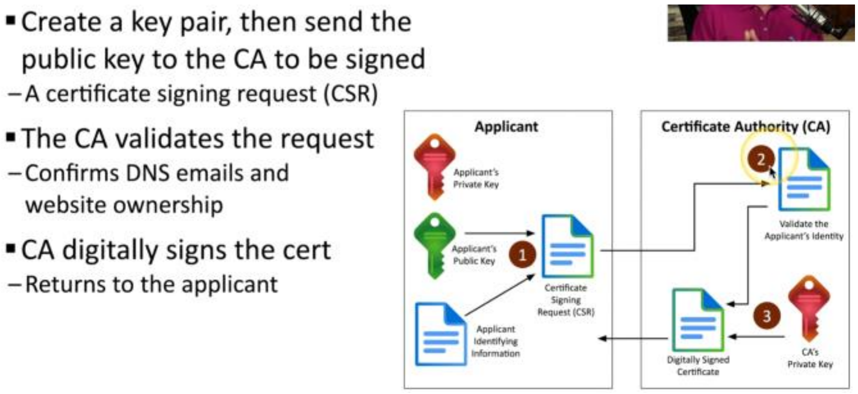
Certificate details:
- Serial number
- Version
- Signature algorithm
- Issuer
- Name of cert holder
- Public key
- Extensions
- And more.

Root of trust: Everything associated with IT security requires trust. A foundational characteristic
- How to build trust from something unknown? Someone/something trustworthy provides their approval
- Refer to the root of trust, an inherently trusted component, hardware, software, firmware, or other component.
- Hardware security module (HSM), secure enclave, certificate authority, etc.

How can you trust an unknown entity?
- Use of a trusted third-party or authority
- Certificate Authority (CA) has digitally signed the website certificate, you trust CA, therefore you trust the website.
- Real-time verification.

Third-party certificate authorities:
- Built-in to your browser.
- Purchase your web site certificate, it will be trusted by everyone's browser.
- CA is responsible for vetting the request, they will confirm the certificate owner, additional verification may be required by the CA.

Certificate Signing Requests:



- Create a key pair, then send the public key to the CA to be signed
  - A certificate signing request (CSR)
- The CA validates the request
  - Confirms DNS emails and website ownership
- CA digitally signs the cert
  - Returns to the applicant

Applicant
Applicant's Private Key
Applicant's Public Key
1
Certificate Signing Request (CSR)
Applicant Identifying Information
Certificate Authority (CA)
2
Validate the Applicant's Identity
3
Digitally Signed Certificate
CA's Private Key

Private Certificate Authorities: You are your own CA.
- Build it in-house
- Your devices must trust the internal CA

- Needed for medium-to-large organisations, many web servers and privacy requirements.
- Implement as part of your overall computing strategy, Windows Certificate Services, OpenCA are options to create your own CA.

Self-signed Certificates: Internal certificates don't need to be signed by a public CA

- Your company is the only one going to use it, no need to purchase trust for devices that already trust you.
- Build your own CA, issue your own certificates signed by your own CA
- Install CA certificate/trusted chain on all devices, they'll now trust any certificates signed by your internal CA. Works exactly like a certificate you purchased.

Wildcard Certificates: Subject Alternative Name (SAN)

- Extension to an X.509 certificate
- Lists additional identification information
- Allows a certificate to support many different domains
- Wildcard domain certificates are based on the name of the server, it will apply to all server names in a domain.

Key Revocation: Certificate Revocation List (CRL)

- Maintained by the Certificate Authority (CA)
- Can contain many revocations in a large file
- Changes all the time. **Heartbleed - Vulnerability in the OpenSSL application library.**

OSCP Stapling: Online Certificate Status Protocol

- Provides scalability for OCSP checks
- The CA is responsible for responding to all client OCSP requests, this may not scale well.
- Instead we have the certificate holder verify their own status. Status information is stored on the certificate holder's server.
- OCSP status is "stapled" into the SSL/TLS handshake, digitally signed by the CA.