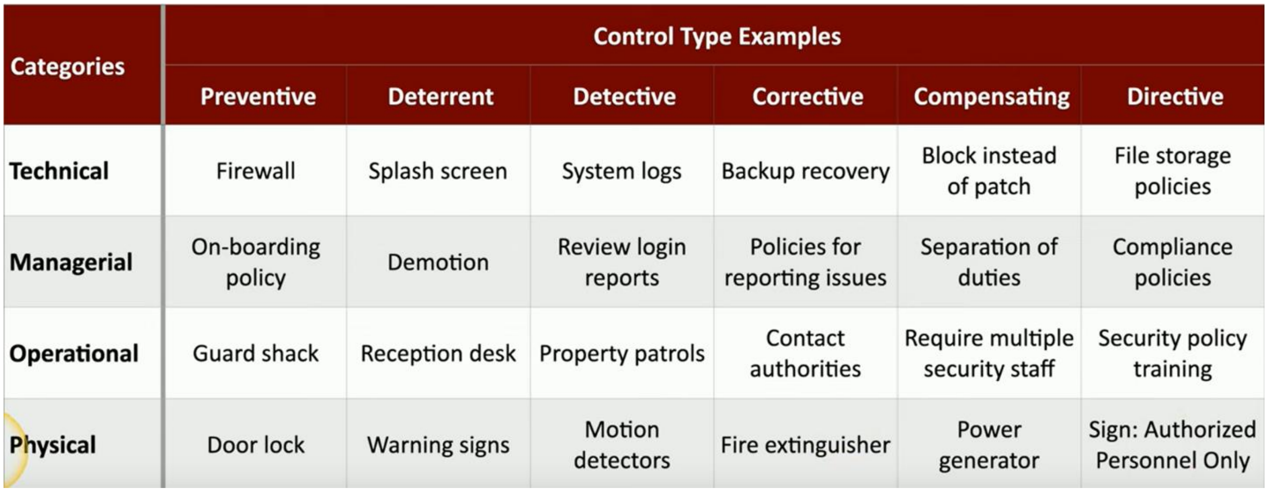
1.1

Full objective: "Compare and contrast various types of security controls."

Security controls

- Various security risks exist for varied assets (data, physical property, computer systems)
- If we recognize security controls, we can prevent events, minimize impact, and limit damage
- Categories:
  - Technical controls
    - Implemented using systems
    - Operating system controls
    - Firewalls, anti-virus
  - Managerial controls
    - Administrative controls
      - design and implementation
    - Policies and standard procedures
  - Operational controls
    - Implemented by people instead of technology
    - Security guards, awareness programs
  - Physical controls
    - Limit physical access (they are also physical things like generators)
    - Guard shack, fences, locks, badges
- Control types:
  - Preventive: block access to a resource [like security policies, firewall rule security checks door locks,etc]
  - Deterrent: discourage an intrusion attempt but does not directly prevent it [like warning signs, reception desk,etc]

- Detective: identify and log an intrusion attempt but may not prevent access[like login reports, system logs,motion detectors,etc]
- Corrective: applied after an event is detected, reversing impact, continue with minimal down time[Restore backups, policies to report issues, contact law enforcement, fire retardants]
- Compensating: existing controls insufficient, temporarily control **using other means** [firewall blocking a certain app, implement separation od duties, require security guards, generator for power]
- Directive: direct a subject towards a compliance (weak control, based on suggestion/policy)[do this pls typa shi]
    - Relatively weak security



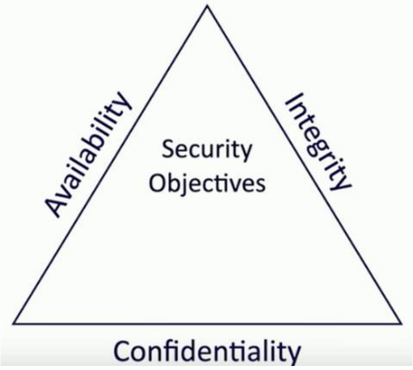
Categories	Control Type Examples					
	Preventive	Deterrent	Detective	Corrective	Compensating	Directive
Technical	Firewall	Splash screen	System logs	Backup recovery	Block instead of patch	File storage policies
Managerial	On-boarding policy	Demotion	Review login reports	Policies for reporting issues	Separation of duties	Compliance policies
Operational	Guard shack	Reception desk	Property patrols	Contact authorities	Require multiple security staff	Security policy training
Physical	Door lock	Warning signs	Motion detectors	Fire extinguisher	Power generator	Sign: Authorized Personnel Only

1.2

Full objective: "Summarize fundamental security concepts."

The CIA Triad(also called AIC Triad)

- Combination of principles, the fundamentals of security (not to be confused with the Central Intelligence Agency)
- **C - Confidentiality**: prevent disclosure of info to those unauthorized
- **I - Integrity**: information cannot be modified without detection
- **A - Availability**: systems and networks must be up and running

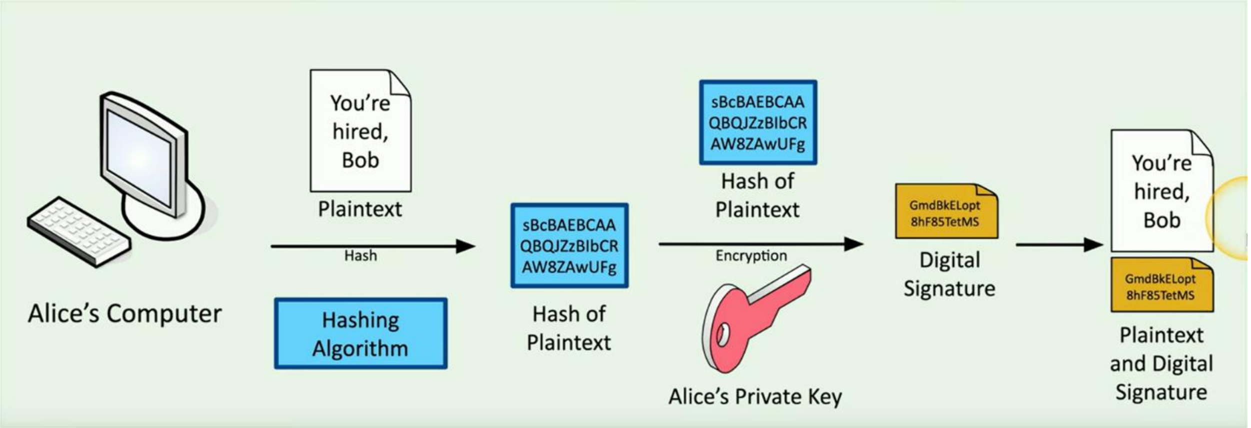


- Confidentiality
  - Certain info should only be known to certain people
  - Encryption, access controls, auth
- Integrity
  - Data is stored and transferred as intended
    - Data is not changed/modified
  - Hashing, digital signatures, certificates, non-repudiation
- Availability
  - Info accessible to authorized user
  - Redundancy, fault tolerance, patching [patching ensures stability and security]

Fault Tolerance → Multiple Components ensures system runs even if some components fail

Non-repudiation

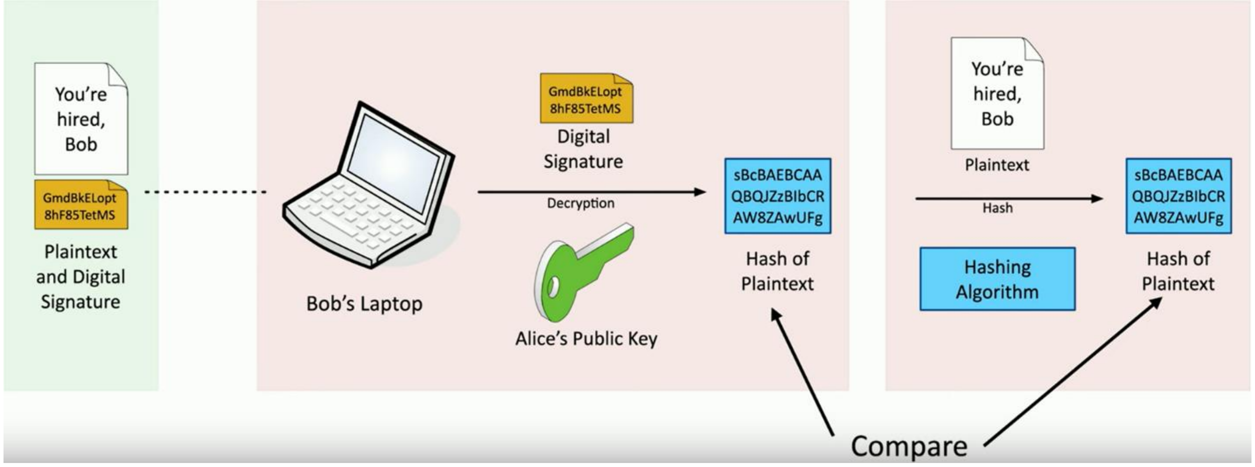
- You cannot deny what you've said, like signing a digital contract
- Proof of integrity
  - Verify data remains accurate and consistent
  - Can be attained using a hash, data as a short string of text [message digest]
    - If data changes at all, "fingerprint" changes entirely
  - However… this doesn't associate data with an individual
- Proof of origin
  - Proving the source of the message (authentication) and ensure message wasn’t changed
  - Ensure signature isn’t fake
  - Sign with digital "private key" (only known by sender of data)
    - Verified with "public key"
- Creating a digital signature



Both parties agree on a single hashing algorithm

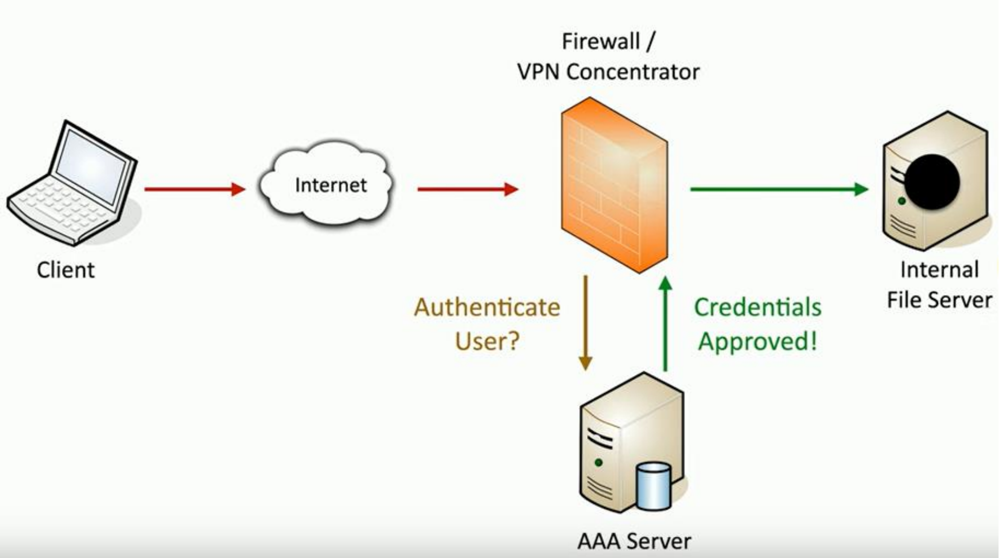
Note:

- A message digest is the output of a cryptographic hash function
- A fingerprint is a condensed representation that uniquely identifies a file or a document



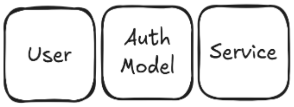
Authentication, Authorization, and Accounting

- AAA framework
  - **Identification**: who you claim to be
  - **Authentication**: prove you are who you claim to be
  - **Authorization**: what access do you have when authorized
  - **Accounting**: what resources did you use when authorized
- Authenticating people



  - The client accesses the VPN through the internet.
  - The firewall/VPN concentrator does not contain any info regarding username and password.
  - The AAA server has both the username and password stored.

- Authenticating systems
    - Many managed devices, how to authenticate without a physical user?
    - Device certificate, VPN or software validates authorized system/device
        - Digital certificates ensure the devices are valid
- Certificate authentication
    - Organization has a trusted **Certificate Authority (CA)** that signs all device certificated
        - Certificate Authorities have their own certificate signed by a root CA
- Authorization models → (Defined by roles, attributes, etc)
    - Certain users and services have access to certain data and applications
    - Put the auth model in the middle

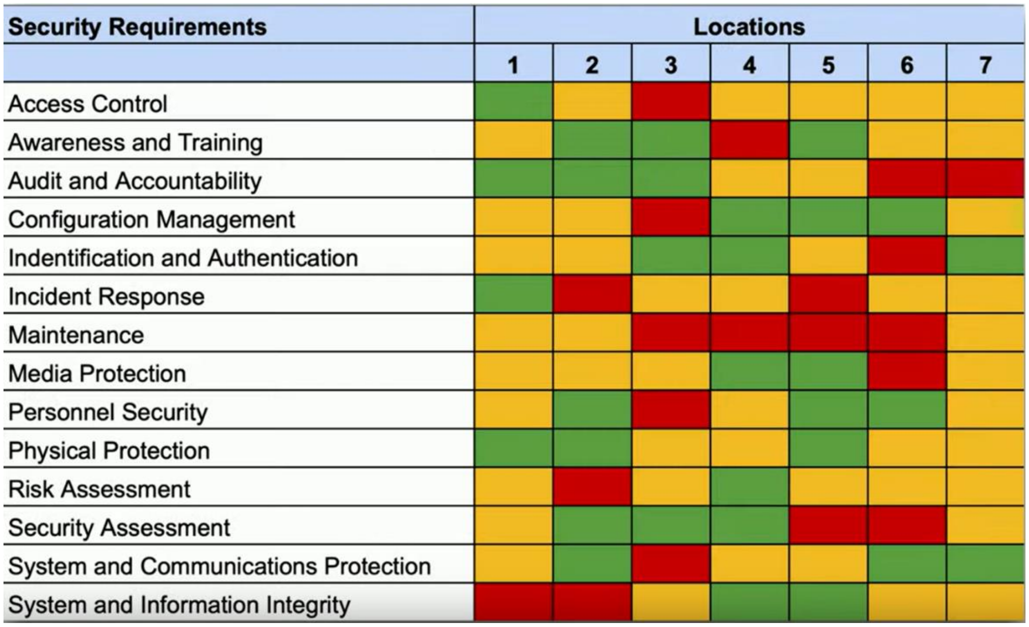


    - Without a model, you would need to profile dozens of users individually
    - With a model, you add abstractions (groups) to reduce complexity → streamlines administration

Gap Analysis

- **Where you are compared with where you want to be**, requires extensive research
- Choosing the framework
    - Work towards a known baseline and determine the end goal
    - Baseline frameworks like NIST SP 800-171 Revision 2 or ISO/IEC 27001
- Evaluate people and processes
    - Get a baseline of employees: experience, training, knowledge
    - Examine the current processes: systems and policies→use formal standards
- Analysis: compare and contrast
    - Compare, identify weaknesses, then detailed analysis → standards may not be enough/relevant to all companies all the time
- The analysis and report

  - Report includes final comparison of objectives and current state as well as path to goal



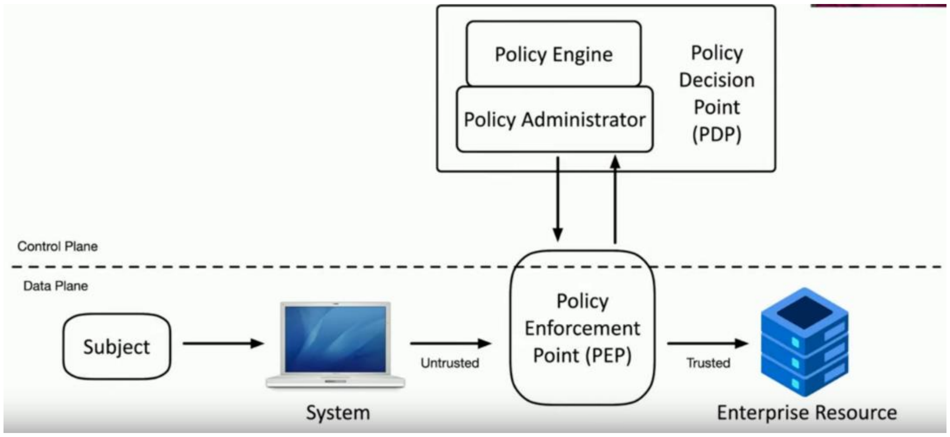
Security Requirements	Locations						
	1	2	3	4	5	6	7
Access Control	🟩	🟨	🟥	🟨	🟨	🟨	🟨
Awareness and Training	🟨	🟩	🟩	🟥	🟩	🟨	🟨
Audit and Accountability	🟩	🟩	🟩	🟨	🟨	🟥	🟥
Configuration Management	🟨	🟨	🟥	🟩	🟩	🟩	🟨
Indentification and Authentication	🟨	🟨	🟩	🟩	🟨	🟥	🟩
Incident Response	🟩	🟥	🟨	🟨	🟥	🟨	🟨
Maintenance	🟨	🟨	🟥	🟥	🟥	🟥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

Green = Good, Yellow = Moderate, Red = Bad

Zero Trust

- Many networks are open on the inside, zero trust says everything must be verified and secured at every step (security in depth)
  - Nothing is inherently trusted
- Planes of operation
  - Split the network into "functional planes"
  - **Data plane**: frames, packets, network data
    - performs the actual security process
  - **Control plane**: manage actions of data plane (policies, rules, etc.)
    - Determines how packets are forwarded
- Controlling trust
  - Adaptive identity
    - Checks the identity of the user based on not just what the user is telling us but also other security measures

        - Consider source and requested resources, multiple risk indicators, make auth stronger
    - Threat scope reduction: reduce number of entry points
    - Policy-driven access control: combine adaptive identity with predefined rules
- Security zones
    - Broad categorization of where a user is connecting from and where you are going
    - Trusted vs. untrusted, internal vs. external, departments, etc.
    - Zones may be enough by itself to deny access or provide implicit trust
- Policy enforcement point
    - Multiple devices working together to auth users and traffic
    - Any subjects and systems that communicate using a certain secure network is subject to evaluation by PEP
    - Subjects and systems must pass through a "gatekeeper" to access resources
    - Allow, monitor, and terminate connections -- hands over to the policy decision point
- Policy decision point: responsible for examining authentication and deciding authenticity - Policy engine: evaluates each decision based on policy -- grant, deny, or revoke - Policy administrator: generates access tokens or credentials, main function is to hands back the decision to PEP
- Full zero trust model:



Physical Security

- Barricades / bollards
  - Prevent access
  - Channel people through an access point
  - Identify safety concerns
- Access control vestibules: isolated rooms for access point authentication
  - All doors either normally unlocked or locked depending on level of security
  - When one door is open, other cannot be unlocked -- allows for one-at-a-time or controlled groups to access an area
- Fencing
  - Building a very obvious perimeter
  - May be transparent or opaque
  - Must be robust to prevent bending or cutting
  - Add razor wire or height to prevent climbing for more secure areas (some fences may be solely deterrents)
- Video surveillance
  - CCTV (closed circuit television): can replace physical guards
  - Newer features include motion detection and object recognition
    - Different angles, single storage/ control point
- Guards and access badges
  - Security guard: physical protection at reception area
    - Two-person integrity/control: minimize exposure to attack, no single person has access to a physical asset
  - Access badge: identifying details, worn at all times, electronically logged
- Lighting → IR cameras exist but nothing beats raw footage
  - More light means more security: attackers avoid light and exposed areas
  - Lighting and angles may be important for facial recognition

- Sensors
    - Infrared: detects radiation in light and dark, common in motion detectors
    - Pressure: detects change in force, floor and window sensors
    - Microwave: detects movement across large areas
    - Ultrasonic: sends signals and receives reflected sound waves to detect motion or collision

Deception and Disruption

- Honeypots → create virtual systems and environments to trick bad actors
    - Attract threat actors and trap them
    - Probably a machine, create a virtual world (can be done with open-source software)
    - Constant battle to discern real from fake (honeypot effectiveness)
- Honeynets → Dedicated servers, routers, workstations and files
    - A collection of honeypots on a large deception network
    - https://projecthoneypot.org
- Honeyfiles → False but authentic looking files
    - More honey, more believable -- increases chance of deception (passwords.txt, etc.)
    - Alert sent if file accessed (virtual bear trap)
- Honeytokens
    - Track malicious actors by adding traceable data to honeynet
        - Fake API credentials that don't actually provide access
        - Fake email addresses whose inbox is monitored
    - Can be any kind of data like cookies, DB records, etc

1.3

Full objective: "Explain the importance of change management processes and the impact to security."

Change Management → Helps avoid confusion, down time and mistakes

- Upgrading software, patching an application, changing firewall configuration, modifying switch ports, etc.
- One of the most common and frequent risks and enterprise, must be standardized
- **Change approval process**: a formal process for making change
  - Complete request forms → ensures org has logs and documentation
  - Determine purpose of change → Why is the change happening?
  - Identify scope of change → What systems will be affected
  - Schedule a date and time →Ensures people are prepared
  - Determine affected systems and impact → Possible impact of change
  - Analyze risk associated with change → What are the risks?
  - Get approval of the board
  - Get end user acceptance

Inform all required parties of any change taking place

- Ownership
  - Individual or entity needs to make a change - they own the process but don't perform the actual change
  - Owner manages the change process, is informed of updates and must test their system afterward [Ensures acceptability]
    - Eg: Shipping Printer is to be updated
      - The shipping department owns the process
      - IT handles actual change
- Stakeholders - who is impacted by this change?
  - Stakeholders may want to input their views
  - Small things may affect the entire company

- Impact analysis
  - Determine a risk value (high, medium, low)
  - What risks can occur upon making the change? What about NOT making the change?
- Test results - sandbox testing environment
  - Technological "safe space"
  - Use before making change to production → Test in production environment
  - Confirm a backout (revert) plan → Even if the plan works in the sandbox, ensure your backup remains functional
- Backout plan
  - ALWAYS HAVE A STABLE BACKUP
  - Change might not go well, always have a way to revert your changes
  - Some changes may be very difficult to revert
- Maintenance window - when is the change happening?
  - Might be the most difficult part of the process
  - Time of day, time of month, time of year → Ensure it doesn't affect the company workflow as much as possible
- Standard operating procedure
  - Change management is critical and must be well-documented
  - Changes to process itself must fit within standards
    - Nobody should be able to make changes without prior approval

Technical Change Management

- Put change management process into action
- Change management concerned with what, technical team handles the how
- Allow list / deny list - security policy controlling app execution
  - Allow list - nothing runs unless it's approved
  - Deny list - everything runs unless it's disapproved (i.e. anti-malware)

- Restricted activities
    - Scope of a change is critical
        - Some scopes are nuanced
            - They maybe small enough to make the decision yourself or not
    - Change approval is permission only for specific changes
    - Scope may need to be expanded
        - Because one change may lead to many changes being required
    - Change management process determines next steps
- Downtime
    - Services will be eventually unavailable, usually done during non-business hours
    - If possible, prevent any downtime, switch to secondary system
        - Usually for sensitive platforms
        - Should be as automated as possible, part of backout plans
    - Send email notifications and calendar updates
- Restarts - common requirement
    - Reboot OS, power cycle switch, bounce service
    - Services - stop and restart service or daemon
    - Application - close completely
- Legacy applications
    - Been around a while, often no longer supported
    - Document the system, may be quirky
- Dependencies - may add complexity
    - To complete A, you must complete B
    - Libraries, other applications, etc.
    - Modifying one component may require modifying/restarting several others

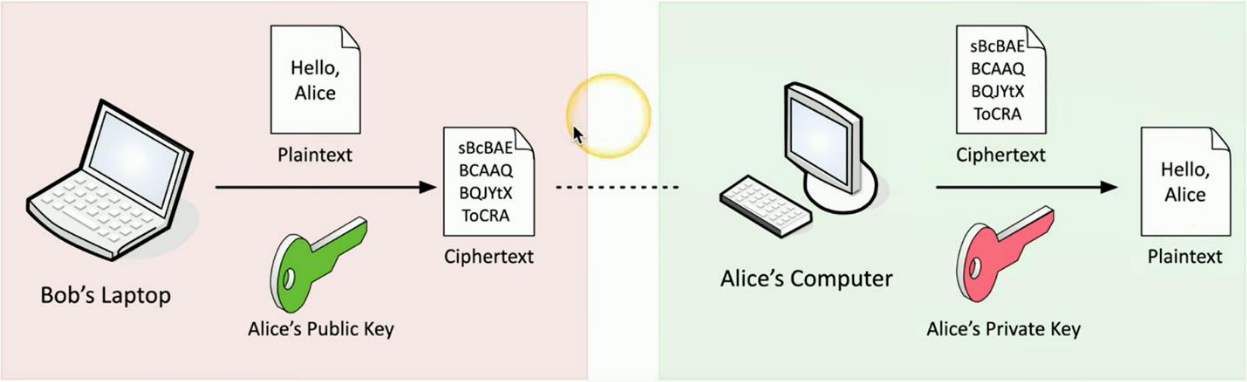
- Documentation → must be always be updated
    - New changes require new processes and documentation
    - Updating diagrams (network, etc.)
    - Updating policies and procedures, configurations, etc
- Version control
    - Track changes to a file or config data over time
    - Router configs, Windows patches, application registry entries, etc.
    - Some devices/suites provide this, some need additional software
        - If possible ensure there is a method to revert to previous versions

1.4

Full objective: "Explain the importance of using appropriate cryptographic solutions."

Public Key Infrastructure (PKI)

- A PKI is a framework that comprises of hardware, software, plans, policies, people, etc, which are required to create, manage, store, distribute and revoke digital certs and keys
- Also refers to the binding of public keys to people or devices, in association with certificate authority
- Extremely important for the organization. Plan accordingly
- Symmetric encryption
    - A single, shared key
        - Used for the encryption and decryption of data
    - Secret key algorithm
    - Doesn't scale very well, but very fast
    - Often combined with asymmetric encryption
- Asymmetric encryption
    - Public key cryptography - two mathematically related keys
    - Private key and public key
        - Alice encrypts with Bob's public key
        - Bob decrypts with his private key
    - Cannot derive private key from public key
        - Anybody can encrypt data using the public key but data can only be decrypted using the private key
- The key pair
    - Key generation involves randomization and very large prime numbers
    - Everyone has public key, only individual has private key



- Key escrow
    - Someone else holds your decryption keys (third-party)
        - May be within your organization
    - Business/government might need to decrypt partner data
    - Controversial, but can be legitimate business arrangement

Encrypting Data

- Encrypting stored data
    - Protect data on stored data (data at rest)
        - Data is stored physically on storage devices
    - Full-disk and partition/volume encryption – BitLocker (Windows), FileVault(MacOS), etc.
    - File encryption - EFS, etc. → File Level Encryption built into NTFS (New Technology File System)
- Database encryption
    - Protecting and transmitting stored data
    - Transparent encryption - encrypt all database info with symmetric key
    - Record-level encryption - encrypt individual columns and use separate keys
- Transport encryption
    - Protect data traversing the network
    - Encrypting in the application (HTTPS, etc.)

  - VPN (Virtual Private Network)
    - Encrypts all data transmitted over the network
    - Client-based using SSL/TLS, site-to-site using IPSec
- Encryption algorithms
  - Many different options, both sides decide on algorithm
  - Algorithms differentiate in security, speed, complexity
  - (his graphic here for DES vs. AES is too vague/inaccurate)
- Cryptographic keys
  - To be secure, cryptographic solutions must be able to withstand the algorithm/process being revealed (only cracked if key is in possession)
  - <u>Kerckhoff's principle - **the security of a cryptographic system shouldn't rely on the secrecy of the algorithm**</u>
  - Key determines the output (encrypted data, hash value, etc.)
- Key lengths
  - Larger key = more secure
  - Symmetric encryption is commonly 128-, 192-, or 256-bit (AES)
  - Asymmetric encryption requires larger keys (3,072-bit or larger)

Making keys larger isn't always convenient

- Key stretching - make a key stronger by encrypting data multiple times
  - Also called key strengthening

Key Exchange

- Large and popular logistical challenge
- One example is "out-of-band" (not over network)
- In-band key exchange - over network, protect key with additional encryption
- Real-time encryption/decryption
  - Share a symmetric session key using asymmetric encryption (i.e. Diffie-Hellman)

Encryption Technologies

- Trusted Platform Module (TPM) → Present in motherboard
    - **Cryptography hardware on a device,** specification for cryptographic functions
    - Cryptographic processor (RNG, keygen)
    - Persistent memory - unique keys burned during manufacturing
    - Versatile memory - storage keys and hardware config info (BitLocker, etc.)
    - Password protected → No Dictionary Attacks
- Hardware Security Module (HSM)
    - Used in large environments - securely store thousands of keys
        - Data is clustered so there are redundancies
    - High-end cryptographic hardware → Performs cryptographic functions in-device
    - Cryptographic accelerators - offload high computing requirement(CPU overhead)
- Key management system
    - Centralized management system, on-premises or cloud-based
    - Separate keys from encrypted data
    - All key management from one console
    - Specific keys for specific services
    - Rotate keys frequently
    - Log key use
- Keeping data private
    - Many, many different locations, wide attack vector
        - Most private and critical data is often physically closest
- Secure enclave - a protected area, hardware isolated from main processor
    - Solely dedicated to data privacy

    - Manufacturers may have different names for it
  - Provides extensive security features
    - Has own boot ROM, monitors bot process
    - Has TRNG, real-time memory encryption, and root keys
  - Ensures data is private even if the devices falls into the hands of somebody else

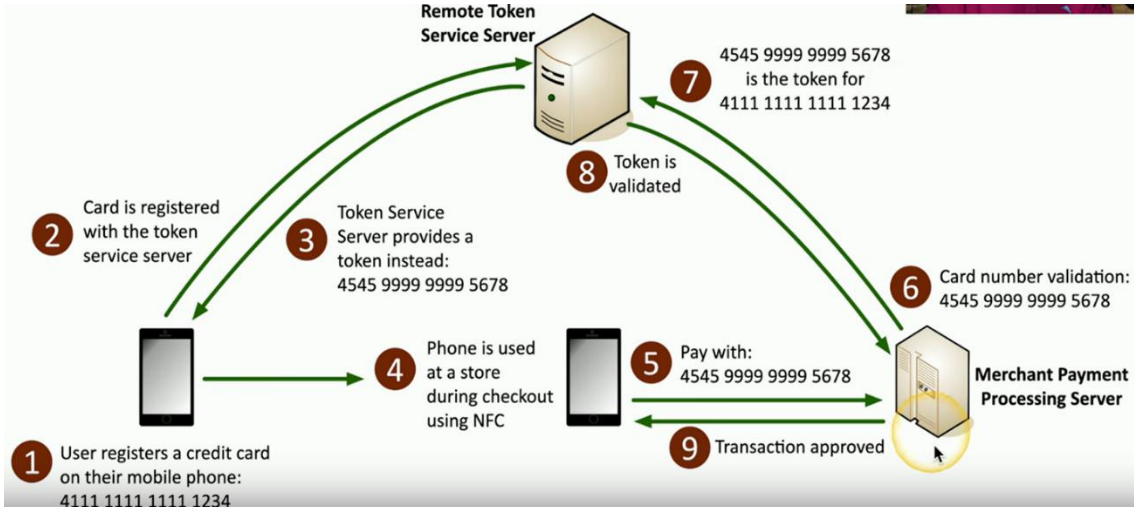
Obfuscation → Security through obscurity

- The process of making something unclear, hiding info in plain sight
- Steganography - hiding information in images
  - Message is invisible but extractable
- Common steganography techniques
  - Network-based - embed messages in TCP packets
  - Use and image - embed message in the image itself
  - Invisible watermarks
  - Audio and video steganography

Not the strongest security measure since if you know how it works, the data isn't secure anymore

(violates Kerchoff's principle)

- Tokenization - replace sensitive data with a non-sensitive placeholder
  - Common with credit card processing - temporary token during payment
  - NOT encryption or hashing, original data and token not related
  - Data is transferred into another format that is useless to the attackers even if captured
  - After the data reaches it destination, the tokens are converted into the sensible data


Remote Token Service Server
1 User registers a credit card on their mobile phone: 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
4 Phone is used at a store during checkout using NFC
5 Pay with: 4545 9999 9999 5678
6 Card number validation: 4545 9999 9999 5678
7 4545 9999 9999 5678 is the token for 4111 1111 1111 1234
8 Token is validated
9 Transaction approved
Merchant Payment Processing Server

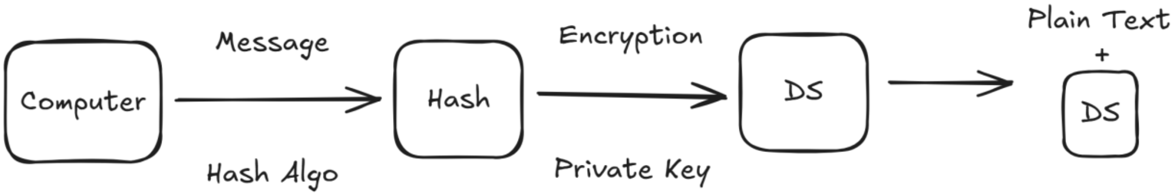
- Data masking → Data can be masked by any method like substitution, shuffling, etc
  - Data obfuscation - hide some of the original data - protects S/PII (Eng Banking Appliances where the bank numbers are partially hidden)
  - May only be hidden from view (still in storage)

Hashing and Digital Signatures

- Hashes - represent data as a short string of text
  - One-way function, impossible to recover data from hash
  - Used for integrity and digital signatures → verify that the content is the same as the original
- Collision
  - Hash functions may produce an identical hash from different input (i.e. MD5)
  - Modern hash functions (SHA256, etc.) do not have this problem
- Practical hashing
  - Verify a downloaded file by comparing hash with one provided on source
  - Storing passwords as a salted hash and use those during authentication
- Adding some salt - random data added to a password when hashing
  - Every user gets their own random salt, changes hash completely
  - Breaks rainbow tables (pre-computed hashes of common passwords)

Note: Hashing is not Encrypting

- Digital signatures



  - Prove integrity, authentication, non-repudiation
  - Sign with private key, verify with public key
  - Any change in message will invalidate the signature

Blockchain Technology

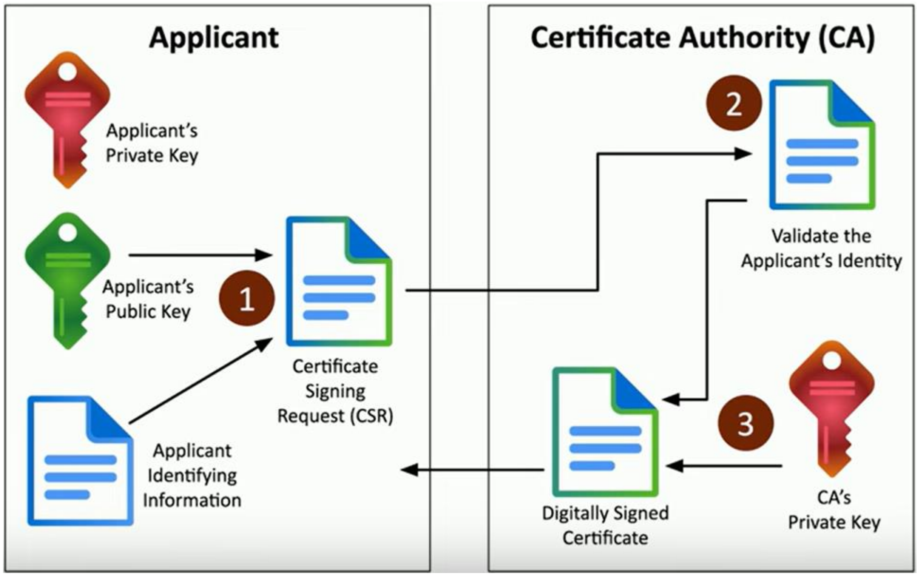
- Blockchain - a distributed ledger(which is public) to keep track of transactions
  - Everyone maintains the ledger
- The blockchain process → All users will gain information about all transactions
  i. Transaction is requested
  ii. Transaction is sent to every node in a decentralized network to be verified
  iii. Verified transaction is added to a new block of data with other transactions
  iv. Hash is calculated from previous blocks of transaction data and added to new block of verified transactions
  v. Block is added to end of Blockchain, updated to all nodes in the network
  vi. If any blocks altered, all hashes recalculated, altered chain now rejected

Certificates

- Digital certificates - a public key certificate → file that contains a public key and digital cerificate
  - Binds a public key with a digital signature
  - Adds trust using Certificate Authority
  - Web of Trust adds other users for additional trust
  - Can also be built into OS (Windows Domain services, etc.

Web of Trust is when individual trusted organizations/components sign (trust) each other’s security

- What's in a digital certificate?
  - X.509 - standard format
  - Details - serial number, version, algorithm, issuer, cert holder, public key, etc.
- Root of trust - an inherently trusted component to build trust → If this component trusts something, It can be trusted
  - Hardware, software, firmware, etc.
  - HSM, Secure Enclave, CA, etc.
- Certificate Authorities
  - Digitally signs website certificates to provide real-time verification/ validation
  - Third-party trust for an unknown entity (website)
  - Hundreds of third-party CAs, built in to browser
  - When we purchase a certificate from a CA, we are purchasing the processes and verification steps involved in certifying the website
- Certificate signing requests
  - Create key pair, send public key to CA to be signed
    - Certificate signing request (CSR)
  - CA validates request (confirm DNS emails and website ownership)
  - CA digitally signs cert and returns to 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're your own CA, devices must trust internal CA
        - You would need to install your own certs in your servers and internal browsers
    - Needed for medium to large organizations with lots of web servers
    - Windows Certificate Services, OpenCA, etc.
- Self-signed certificates
    - Internal certificates don't need to be signed by a public CA
    - Build your own CA to issue your own certificates
    - Install CA certificate on all internal devices
- Wildcard certificates
    - Subject Alternative Name (SAN) - *.domain.com
        - Extension to X.509
        - Allows certificate to support many domains
    - Wildcard domain
        - Certificates based on the name of the server
        - Applies to all server names in a domain

Wildcard certs secures the many sub domains of a parent domain.

Eg: info.example.com, home.example.com, etc

- Key revocation
    - Certificate Revocation List (CRL)
        - Maintained by CA
        - List of all revoked certs are kept in the CA itself
        - CVE-2014-0160 (Heartbleed) - OpenSSL flaw that required every web server certificate be replaced
- OCSP stapling - Online Certificate Status Protocol
    - Provides scalability for OCSP checks
    - CA responsible for responding to all client OCSP requests
    - Instead of this, have certificate holder verify their own status

  - OCSP status is "stapled" into the SSL/TLS handshake
  - We cannot fully trust a 3rd party web server to truthfully tell us the status of the certificate and relying on the CA to provide and go through the list of all revoked certs is not efficient
- Getting revocation details to the browser
  - OCSP - browser can check certificate revocation
    - Browser checks for cert revocation when you visit a 3rd party website
  - Messages usually sent to OCSP responder via HTTP
    - More efficient than downloading a CRL
  - Not all browsers/apps support OCSP (most modern do)
  - If you are not using OSCP to staple the status of the certs into the SSL/TLS handshake, you can use a trusted third party server to provide the OSCP info. This info is then added to the certificate