



Security+

Domain 1: General Security Concepts

SY0-701

Brian Olliff

Defensive Engineering Instructor

Topics

Security+ overview

Fundamental concepts

Security controls

Change management

Cryptography

Learning Objectives

- Be able to summarize and compare various security controls
- Understand difference between security control categories
- Understand fundamental security concepts
 - + CIA, AAA
 - + Non-repudiation
 - + Zero trust
- Explain importance of change management processes
- Understand cryptography fundamentals
 - + Asymmetric vs symmetric
 - + PKI
 - + Encryption methods, tools, and practices

CompTIA Security+

- General Security Concepts
 - Security controls, fundamental security concepts, change management, cryptographic solutions
- Threats, Vulnerabilities, and Mitigations
 - Threat actors & motivations, vectors & attack surfaces
 - Types of vulnerabilities and mitigation techniques
- Security Architecture
 - Architecture models & types, security architecture principles
 - Data protection, architecture resilience & recovery
- Security Operations
 - Security hardening, asset management, alerting & monitoring
 - Vulnerability management, IAM, IR & investigations
- Security Program Management and Oversight
 - Security governance, risk management, compliance
 - Third-party assessments, audits & assessments

Studying & The Exam

- Use multiple resources to study
- Take your time and revisit topics if needed
- Ask questions
- Schedule your exam and set a goal
 - Max of 90 questions
 - Combination of multiple choice and performance-based
 - 90 minute time limit
 - 750 needed to pass (scale of 100-900)
 - Cost of exam varies depending on location/currency

Security Controls



Security Control

- Procedures, activities, and tools designed to protect confidentiality, integrity, availability of system and data
- Examples
 - Security software & devices
 - Written security policies (for both users and security staff)
 - Locked doors and gates
 - Security cameras
 - Risk management programs
- Four main categories
 - Technical
 - Managerial
 - Operational
 - Physical

Technical Controls

- Control implemented as some type of system
 - Hardware, software, firmware, etc
- Examples
 - Network firewall
 - Anti-virus (AV) or Endpoint detection & response (EDR) software
 - Operating system security settings
 - Encryption software
 - Intrusion prevention system (IPS)
 - Access control lists (ACL)

Operational Controls

- Control implemented primarily by person instead of system
- With *managerial controls*, can collectively be referred to as **Administrative Controls**
- Important distinction:
 - All controls are implemented by people, does not define them as operational
- Examples
 - Security guards
 - Training programs
 - IT and security staff
 - Security awareness training for users

Managerial Controls

- Controls that provide oversight and management of security programs
- Frequently describe requirements for other categories of controls
- Examples
 - Risk assessment and management programs
 - Tools used to help identify risk
 - Systems to evaluate and select other security controls
 - Security policies
 - Acceptable Use Policies (AUP) for users
 - Policies state requirements for operational and technical controls

Physical Controls

- Controls that provide security to defined structures or areas
- Used to control and monitor access to various locations
- Examples
 - Security alarms
 - Locked doors
 - Closed-circuit surveillance cameras
 - Security lighting
 - Security ID badges
 - Biometric security locks

Control Types

- Preventive
 - Eliminate or reduce likelihood of successful attack
 - Operates before attack can take place
 - ACLs, AV/EDR
- Deterrent
 - Designed to discourage attacker from attempting attack
 - May not prevent any type of access
 - “No trespassing” signs, legal disclaimers in systems
- Detective
 - Identifies and records (logs) attempted (or successful) attacks/intrusions
 - Like deterrent, not designed to prevent attack
 - Logging systems

Control Types

- **Corrective**
 - Assists in reducing or eliminating the effect/impact of attack
 - Used after attack has concluded
 - Backup data/systems, patching exploited vulnerability
- **Compensating**
 - Secondary (or substitute) control to primary control
 - Provides same or better level of protection
 - Uses different technology or methodology (defense in depth)
- **Directive**
 - Implemented to monitor regulatory compliance
 - Provide guidance aligned specifically with organizational requirements

Fundamental Security Concepts



CIA Triad

- Confidentiality
 - Privacy and security of data & assets
 - Keeping unauthorized individual from accessing information
 - Encryption, access control, training
- Integrity
 - Ensuring data/assets free from unauthorized changes
 - Intentional or accidental
 - Malware on system / User entering incorrect information
- Availability
 - Maintain access to systems - timely and reliably
 - Systems remain online as much as possible
 - Quick and secure recovery from disruptions

Roles & Responsibilities

- Chief Information Officer (CIO)
 - Overall responsibility for IT systems and functions
- Chief Technology Officer (CTO)
 - Can be similar to CIO
- Chief Security Officer (CSO)/Chief Information Security Officer (CISO)
 - Directly responsible for information/cyber security
 - May report to CIO in some organizations
- Information Systems Security Officer (ISSO)
 - Staff with technical/specialist administrative responsibilities for security

Other Concepts

- Non-repudiation
 - Proof that individual/system performed an action, cannot deny
 - Creating/modifying files, sending email, etc
 - Often implemented with digital signatures, certificates
- Gap Analysis
 - In-depth review of all security controls, policies, procedures, etc
 - Designed to identify where areas may need improvement
 - Compared to cyber security standards & frameworks
 - Often industry specific (financial, healthcare, etc)
 - Helps with understanding risks and vulnerabilities in organization

Authentication, Authorization, and Accounting



AAA

- Authentication, Authorization, & Accounting (Auditing/Accountability)
- Authentication
 - Proof of identity
 - Password, PIN, MFA
- Authorization
 - Verification that user can access resource
 - Access control lists, various authorization models
- Accounting
 - Ensures that users are *accountable* for their actions
 - Logging & auditing

Authentication

- Method to verify & prove a claimed identity
- Multiple types (factors) of authentication
 - Knowledge-based
 - Ownership-based
 - Biometric
 - Location
 - Behavioral
- Strong authentication
 - MFA - Multi-factor authentication
- Authentication methods must be protected

Authorization

- Ensure properly authenticated users have access to their resources
 - But **NOT** resources they shouldn't access
- On-going process
 - Occurs every time resource access is attempted
- Multiple mechanisms/models
 - Discretionary access control (DAC)
 - Mandatory access control (MAC)
 - Role-based access control (RBAC)
 - Rule-based access control (RB-RBAC)
 - Attribute-based access control
 - Risk-based access control

Accountability

- Users identified & authenticated, authorized, and accessing system
- Ensure all access is correct and not abused
- Logging, monitoring, & auditing
 - Detect intrusions
 - Monitor for suspicious activity
 - Assist with incident response activities
 - Track actions back to individuals
 - Legal support
 - Deter possible malicious actions
- Requires periodic review - manual or automated

Authorization Models

- DAC - Discretionary Access Control
 - Owner assigned to resource, that owner directly assigns permissions
 - Used by majority of operating systems
- MAC - Mandatory Access Control
 - Data sensitivity labels assigned to resources and users
 - Users can access resources that correspond to their “clearance level”
- RBAC - Role-Based Access Control
 - Permissions granted based on job role
 - Users assigned to roles, roles are assigned to resources
- ABAC - Attribute-Based Access Control
 - Access based on combination of subject/object attributes
 - OS used, IP of request, patches installed, geographic location, etc

Zero Trust



What

- Trust on a network must be continually evaluated
- Architecture to protect internal network resources
 - Less easily-identified perimeter in modern networks
 - Once attacker is past perimeter controls, lateral movement unhindered
- All resources & communications are secured regardless of location
- Internal networks are no longer “implicit trust zones”
 - No resource is inherently trusted
 - Not all resources are enterprise-owned
- Zero Trust architecture commonly divided into two components
 - Control Plane - backend communication (policies, admin, etc)
 - Data Plane - application communication (users, servers, etc)

Why

- Deperimeterization
 - Shifting away from network boundaries to protect individual resources/data
- Attacks can come from anywhere on the network
- Some attackers take their time
 - Can be inactive on network for days, weeks, months
- Malicious (or careless) insiders
- Assume everything in network is malicious, unless proven otherwise
- Multiple points in/out of most networks
 - Integration with 3rd party products, cloud providers, mobile devices, etc
- Designed to reduce scope of possible threats
 - Smaller, more controlled attack surface

Constant Adaptation

- Zero trust architecture must be flexible
- Utilizes adaptive identity controls
 - Internal user accounts not inherently trusted
 - Identities and authentication/authorization constantly verified
- Multiple methods to verify identities
 - Credentials
 - Risk-based authentication
 - Digital certificates
 - Network location
- No single attribute used for authentication

Control Plane

- Primary component - **policy decision point** (PDP)
- PDP often consists of **policy engine** (PE) and **policy administrator** (PA)
- Policy engine
 - Uses defined policies and external inputs (threat intelligence, logs, identity information, etc)
 - Determines whether to grant, deny, or revoke access
- Policy administrator
 - Decisions from PE are passed to PA
 - Responsible for establishing/closing communication between subject and resource
 - Performs these actions via commands to **policy enforcement point** (PEP)
 - located in data plane

Data Plane

- Policy enforcement point
 - Enable, monitor, terminate connections between subject and resource/system
 - Communicates with PA
 - Forwards requests for communication
 - Receives policy updates
- Trust zones
 - No universal implicit trust in network
 - Multiple logical divisions of network containing related resources
 - Referred to as “implicit trust zones”
 - PEP controls access to these zones
 - Granular microsegmented approach

Physical Security



Physical Security

- Designed to restrict/monitor access to physical locations and assets
 - Secure buildings
 - Data centers & wiring closets
- Uses same AAA concepts as other security controls
- Frequently divided into zones
 - Separated by various barriers
 - Entry/exit controlled by various mechanisms
 - More secure zones = stricter security control
 - Ex: public zone, medium security zone, high security zone, etc

General Recommendations

- Secure zones should be located far away from public areas
 - Screens and monitors should not face doors, windows, passageways
 - Minimize use of windows, or use one-way glass
- Use signs and other warnings as deterrent controls
- Entry points to secure zones should be discreet
 - Specific security mechanisms should not be obvious
- Traffic between zones should be minimized
- Public areas should be highly visible
 - Design to prevent attackers from covertly using network ports, scanners, etc

Physical Controls

- Defense in depth strategy
 - Layered defense
- Entry points
 - Bollards to block vehicle traffic and protect physical structure
 - Fencing around secure (non-public) locations
 - Transparent
 - Robust
 - Secure against climbing
 - Locked doors (physical, electronic, biometric)
- Lighting for security
 - Safety at night
 - Assists with video surveillance

Entry/Exit Points

- Access control vestibule
 - Most common type - two sets of doors, only one set can be opened/unlocked at a time (“mantrap”)
 - Designed to prevent tailgating, increase difficulty for entry
 - Turnstile used for less secure implementations
- Access cards/badges
 - Used to grant access to secure areas and to identify personnel
 - RFID, NFC, magnetic strip
 - Photo ID
 - May also show access level (number, color code, etc)
 - Requirement to always have visible
 - Anyone without one should be challenged/reported

Prevent & Deter

- Security guards
 - Armed or unarmed (depends on need)
 - Used to monitor and control access to critical checkpoints
 - ID checks, verify proper access, etc
 - Visible presence is strong deterrent, can also prevent
 - Requires proper screening and training
- Video surveillance
 - Less expensive than security guards
 - Response to potential incident may take longer
 - Requires constant monitoring
 - Ability to record events for evidence or investigation
 - Strong deterrent, but not preventative

Physical Security Sensors

- Infrared
 - Detects heat and motion in dark areas
 - Normally only detect motion from living creatures
- Microwave
 - Very sensitive sensor, used in more critical locations
 - Can detect any motion, especially through non-metallic objects
 - Function similar to sonar
- Ultrasonic
 - Emit sound waves at frequencies above human hearing range
 - Often used for automated lighting systems
- Pressure
 - Detects change in pressure over specific area
 - Weight of person, vehicle, etc

Deception & Disruption Techniques



Active Defense

- Defensive engagement with adversary
- Most common is deployment of decoy assets
 - Act as lure or bait for attackers
 - Do not hold any sensitive resources
 - Isolated from rest of infrastructure
- Specifically designed to be vulnerable and accessible
 - May be a specific vulnerability or weakness, or more broad
- Decoy assets are heavily monitored
 - Allows detection of active threats & intrusions without sacrificing security
- Can be used for research purposes or to detect internal threats

Honeypot

- System that is set up specifically to attract attackers
 - Similar to honey attracting flies
- Extensive logging enabled
 - Allows for analysis of strategies, tactics, and tools
- Designed to provide “early warning” of possible attacks
- Can divert attention away from actual sensitive assets
- Critically important to isolate from rest of environment
 - Protects rest of network
 - Prevents attacker from gaining intelligence about infrastructure
 - Completely separate network or placed in DMZ

Honeynet

- Similar to honeypot, but entire decoy network
- Consists of multiple resources (servers, workstations, network devices)
 - Depends on needs and scenario
- Can use actual network resources, or emulators
 - Emulators can be less expensive, but may not be as convincing
- Most frequently used for more extensive monitoring and campaigns
 - More advanced threats that require multiple resources
 - Compared to single-resource vulnerabilities or threats

Deceptive Files

- Honeyfile & honeypot
- Fake data that is meaningless to organization
 - Designed to look convincing to attacker
- Files contain specific, unique data that can be used to track
 - Email addresses, database data, executable files
 - Beacons - object inside file that contacts server once file opened
 - Unique, but useless AWS (or other cloud) keys
- Once data is stolen and/or leaked, can be traced to specific threat actor
- Commonly integrated with honeypot and honeynet

Disruption Strategies

- Adopted from obfuscation techniques that attackers commonly use
- Purpose is to make attacks more difficult and “expensive” for attackers
 - Requires more resources and time for adversaries
- Commonly used in production environments, instead of honeynet
- Examples
 - False DNS records that do not exist
 - Web servers with decoy directories
 - Falsified returns for port scans (open ports that are not accessible)
 - DNS sinkholes
 - May route to honeypot or honeynet for analysis

Change Management Operations



Change Management

- Organizations and systems are constantly evolving
 - Changes to information systems always needed
- Systematic approach to managing all changes in IT infrastructure
- Goal to minimize disruption and downtime
 - Maximize efficiency and value of changes
- Requires planning, testing, approvals, oversight, etc
- Process works with organization
 - Determine potential impacts, dependencies
 - Contingency plans and rollback plans
- Documentation required

Business Operations

- CM impacts more than just security operations
- Changes may be simple or complex
 - Routine software updates or deployments
 - New product deployments
 - Major network architecture modifications
- Structured approval process required to ensure minimal disruptions
 - New vulnerabilities introduced
 - Unexpected downtime
 - Introduce compliance issues
- After changes completed, reviews and audits ensure outcomes

Stakeholders

- Anyone with vested interest in change or project
 - Working with and performing changes (Server, network, security, dev, etc)
 - Leadership overseeing operations
 - Compliance teams
- Stakeholder involvement ensures:
 - Proposed changes receive input from multiple viewpoints
 - Helps to avoid non-obvious risks & problems
 - Minimizes unplanned disruptions
- Help with acceptance and adoption of changes
 - Involvement and input from multiple areas of organization

Ownership

- Individuals/groups responsible for implementing change
 - Project managers, team leaders, etc
- Accountable for ensuring change is completed as planned
 - Deadlines met
 - If modifications needed - appropriate and approved
- Manage risks related to change
 - Any potential downtime
 - Additional personnel needs
- Plan and implement communication and training as needed
- Ensure stakeholders are informed and have approved

Change Management Concepts

- Impact analysis
 - Process of identifying/assessing implications of change
 - How affects users, processes, systems, etc
- Test results
 - Changes first tested in test environment before production systems
 - Can identify unexpected issues or impacts
- Backout plans
 - Plan for reversing change, returning systems to previous configurations
 - Minimizes risk of downtime if change doesn't go as planned
- Maintenance windows
 - Predefined timeframe for implementing changes
- SOPs (Standard operating procedures)
 - Written, detailed instructions for routine operations or changes

Change Management Technical Considerations



Allow & Block Lists

- Lists can be used in multiple contexts
 - Allowed/denied applications
 - Firewall rules
 - Change-related
- Change management context
 - Software, hardware, change types not required to go through entire process
 - Software, hardware, changes not permitted/must always go through process
 - Individuals permitted to approve changes
- Existing restrictions/allowances in systems that may affect changes
 - Software allowed based on hash value - different value after change

Downtime

- Changes often involve restarts
 - Server reboots
 - Service/process restart during upgrades, patches
 - Applications upgraded stop running during updates
- Restarts often result in some length of downtime
 - Should be minimized if possible
 - Scheduled & announced to users with adequate notice
- Goal - reduce impact on users and business processes
- Systems may have dependencies on other software/systems
 - Database server/service restart affects any applications using that server
 - Simple changes may have larger impacts if not carefully planned
 - May affect time required for change implementation & resulting downtime

Legacy Applications

- Many organizations still run systems/software that are “legacy”
 - May no longer receive updates
 - Can contain security issues/vulnerabilities
- Legacy applications may be incompatible with newer software
- Can create complications with other changes
 - Older application with database on server that needs updates
 - DB server updates create issues with legacy application
- May require specialized solutions
 - Virtualization, emulation, custom software, extra systems, etc
- Often lack sufficient documentation, support from vendor

Documenting Changes



Documentation

- All changes need careful, intentional documentation
 - Request, notes, approval, discussion, implementation, etc
- Changes often require other documentation to be updated
 - Policies and procedures
 - Network/structure diagrams
 - Software-specific user instructions
- Frequency of documentation updates depends on org and changes
 - Typically updated whenever major changes, updates occur
 - Applications, systems, processes, etc
- Updated documentation should be noted with new version information
 - Archive older versions, but still keep accessible for reference if needed

Types of Documentation

- Change requests
 - Information about needed change
 - Updates to reflect status changes, modifications, approvals, etc
- Policies and procedures
 - Review and update as necessary after/during changes
- Systems and processes
 - Updated to reflect system changes
 - Architecture changes, diagrams, user manuals, etc
- Configuration management
 - Up to date configuration information, updated after each change
- Training material
- Incident response/recovery plans

Version Control

- Tracking and managing changes to documents, code, etc
- Keeps a history of all changes
 - Request, approvals, etc
- Allows timely reversal of changes if needed
 - Instead of consulting documentation and manually reverting
- Updated documentation
 - Diagrams, processes, procedures, policies, instructions, etc
 - Provides easy method to see updates
 - Helps avoid confusion and potentially using outdated documents

Encryption Types



Cryptography Terminology

- Plaintext - original, readable data
- Algorithm/Cipher - method of encrypting and decrypting plaintext
- Ciphertext - encrypted data, appears random and unreadable
- Key - data passed through algorithm used to encode and decode
- Keyspace - range of values that can be used to construct key
- Key length - # of bits used in key
- Symmetric key cryptography - using the same key to encrypt and decrypt
- Asymmetric key cryptography - separate keys for encrypting and decrypting
 - Public key and private key

Encryption & Hashing Purposes

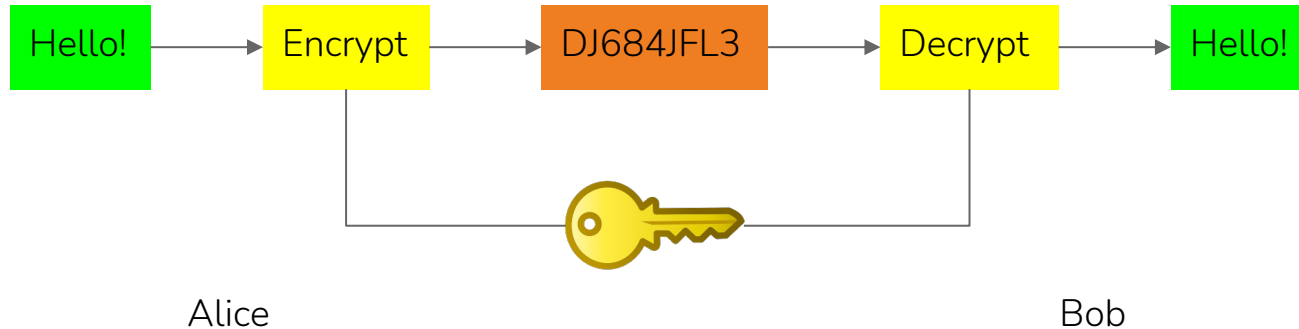
- Confidentiality
 - Encrypted text unreadable without decryption
- Integrity
 - Hash verification ensures data has not been altered
- Authenticity
 - Public/private keypairs and certificates verifies source of data
- Nonrepudiation
 - Certificates and unique keys/signatures prove activity
- Authorization
 - Certificates used for proving identity and granting access

Encryption Strength

- Difficulty in discovering the algorithm and/or key
 - Whichever is not publicly known
- Time required to break encryption
- Amount of processing power and resources required
- Determined by
 - Encryption algorithm used
 - Length and complexity of key
 - Secrecy of key
- Stronger the encryption, the longer it should take to crack (if ever)

Symmetric Encryption

- Symmetric uses the same key/cipher to encrypt and decrypt

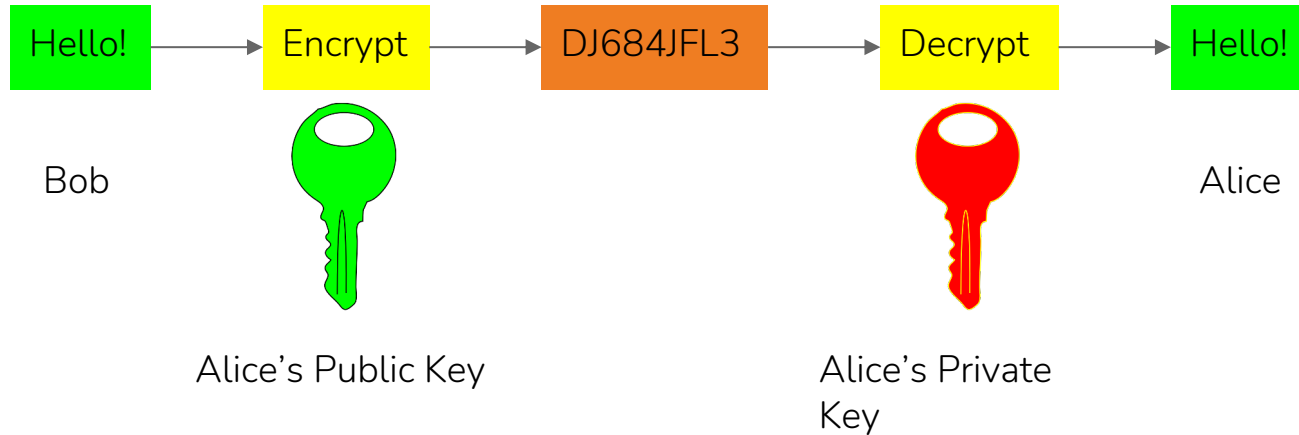


Symmetric Encryption

- Algorithms tend to be more secure than asymmetric algorithms
- Faster than asymmetric algorithms
- Provides confidentiality, but not authenticity or integrity
- Hundreds of different algorithms available
 - DES - Data Encryption Standard
 - 3DES - Triple DES
 - AES - Advanced Encryption Standard
 - RC4 - Rivest Cipher 4
 - RC5 - Rivest Cipher 5
- Requires shared key between communication endpoints
 - How can this key be easily shared without being compromised?

Asymmetric Encryption

- Asymmetric uses different keys to encrypt and decrypt



Asymmetric Encryption

- Asymmetric encryption is the “key” to PKI
- Public key - publicly available, anyone can have and use
 - Bob can use Alice’s public key to encrypt secret data to send to Alice
- Private key - known only to individual/entity that creates it
 - MUST NOT BE SHARED
 - Bob can use to decrypt private data sent to him using his public key
 - Alice can use her private key to digitally sign data sent to Bob
 - Bob decrypts (and verifies signature) using Alice’s public key
- Each key can only encrypt OR decrypt
 - Data encrypted with a public key can **NOT** be decrypted with that same key
- **Key escrow** can be used to ensure keys are not lost
 - Archived key(s) with a *TRUSTED* third party

Asymmetric Encryption Algorithms

- More complex algorithms than symmetric
- Much slower than symmetric algorithms
- Provides confidentiality, authenticity, and nonrepudiation
- Requires two keys - private and public
- Two main algorithms in use
 - Diffie-Hellman - mostly used for key exchange
 - RSA - Rivest Shamir Adleman

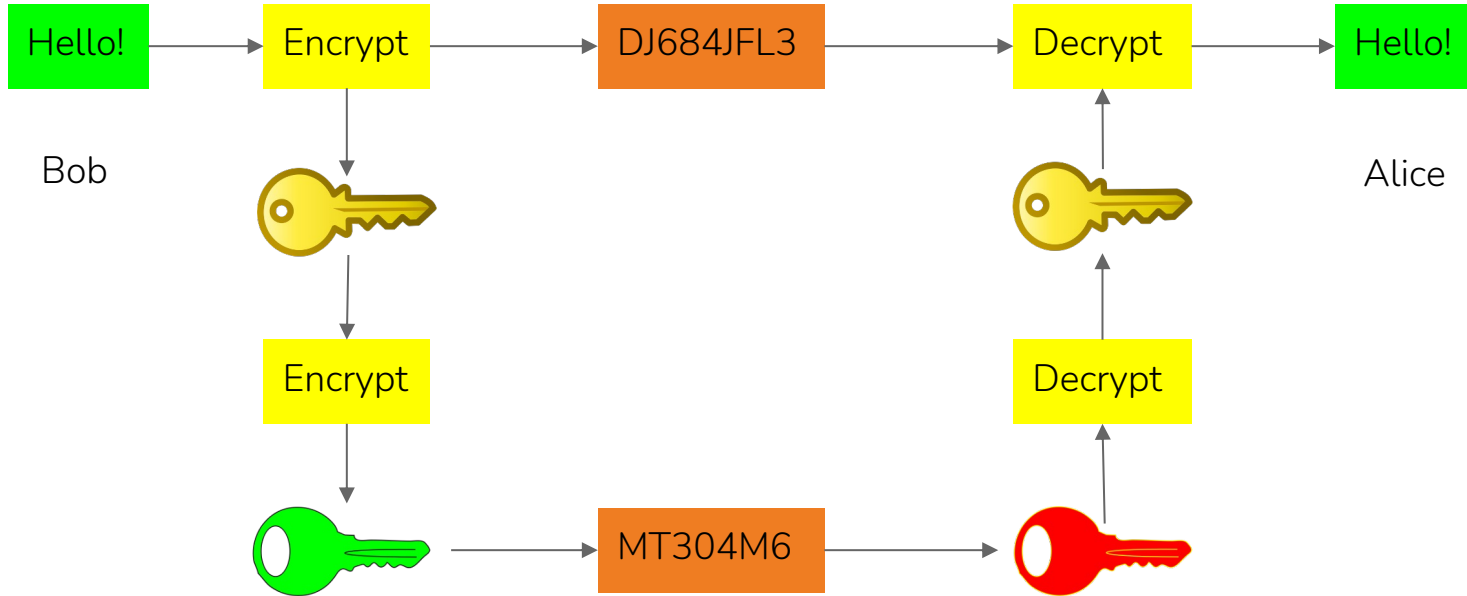
PKI



Symmetric & Asymmetric

- Symmetric is fast - but key exchange is tricky
- Asymmetric solves key exchange - but algorithms are slow
 - Potential issue with trusting source of public key
- Asymmetric encryption used to protect key(s) for symmetric encryption
 - Data to be protected is encrypted with key
 - Sender finds public key of recipient and encrypts first key, sends to recipient
 - Recipient uses private key to decrypt message, and receives first key
 - Decrypts original message using that key
- Basis of most secure communications that are used today
 - HTTPS websites

Encryption and Key Exchange



Public Key Cryptography

- Sharing keys required for encryption workflows
 - Either secret key must be shared OR
 - Public/private key pairs used
- Sharing public keys universally is problematic
 - How do you reliably share these keys?
 - How can you trust that the key is legitimate?
 - Anyone can create key pairs
 - Anyone can claim a public key is theirs

Public Key Infrastructure

- Systems designed to make secure communication easy and accessible
 - Standards
 - Communication protocols
 - Security policies
 - Systems of cryptography
- Purpose is to establish and maintain high level of trust in keys
- Functions on a hybrid encryption model
 - Both symmetric and asymmetric encryption
- PKI is not public key cryptography
 - Infrastructure that enables trust and ease of use
 - Built partially on public key cryptography

Public Key Infrastructure

- Provides the needed trust for public keys
 - Network of trust
- Computers and browsers trust certain Root CAs (Trusted Root CAs)
- Trusted CAs issue digital certificates
 - Web servers (and others) install those certificates
 - Encrypts communication
 - Verifies identity of server
- Certificates signed by Trusted CAs
 - Public key verified and identified on certificates
- Computer/browser implicitly trusts site

Certificate Authorities (CA)

- Trusted entity (organization or server)
 - All devices trust organization CAs
 - Issued certificates have same level of trust
- Maintains and issues certificates
- Root CA
 - Primary CA for organization
 - Verifies subordinate CAs
 - Heavily secured
 - Normally kept offline
- Intermediate CA (or subordinate CA)
 - Trusted & certified by root CA
 - Processes requests and issues certificates

Digital Certificates

- Central concept of PKI
- Associates public key with a claimed owner
- Most common standard is X.509
 - Requires certain fields to be used in certificate
 - Algorithm, validity dates, signature of issuer
- Specifies “subject” - who owns the certificate
- Also “issuer” - what entity generated and issued the certificate
 - Issuer certifies the identity of the subject
- Self-signed certificates
 - Created by the owner (subject) and not issued by third party

Certificate Types

- Self-signed
 - Not issued by a trusted CA
 - Created by system/application that is using the certificate
 - Common with default certificates on new appliances
- Third-party
 - Certificates (usually) purchased from a specialized vendor
 - Publicly trusted CA
- Wildcard
 - Certificates that are valid for multiple subdomains
 - Ex: *.example.com - valid for domain1.example.com, domain2.example.com

Certificate Revocation

- Issued certificates need to be revoked occasionally
 - Compromised keys
 - Replacements for various reasons
- Certificate Revocation List (CRL)
 - Each CA maintains their own
 - List of every certificate that has been revoked
- Online Certificate Status Protocol (OCSP)
 - Returns single cert status instead of entire CRL
 - Does not rely on browser or other system
 - OCSP automatically checks with CAs CRL

Certificate Signing Request (CSR)

- Used when requesting a certificate from CA
 - Web servers - system administrators
 - Email - end user
- PKCS #10 - Public Key Cryptography Standard #10
 - Defines the format for all certificate requests
- System/individual requesting generates keypair, keeps private key
- Public key passed to CA in request
- Additional information included
 - Common name
 - Org name
 - Locality (city), State, County
 - Email address

Encryption Tools



Trusted Platform Module (TPM)

- Standard for hardware-based storage for encryption data
- Normally integrated chip with motherboard or CPU
- Each is encoded with static private key - cannot be changed
 - Subkeys can be reset through “ownership” of TPM
- Purposes:
 - Secure storage for encryption keys
 - Can authenticate hardware platform devices
 - Allows for cryptographic verification of hardware configuration
 - Useful during boot process to verify integrity of system
- Commonly used with Windows BitLocker

Hardware Security Module (HSM)

- Separate physical hardware device for key management
 - Different from TPM (not integrated into other hardware)
 - Dedicated purpose with smaller attack surface
- Usually hardened device, with dedicated security features
 - Certified hardware, security-focused OS, etc
 - Tamper detection features
- Performs encryption operations
 - Digital signatures, authentication, etc
- Can be integral part of PKI environment
 - Key storage, archive, escrow

Key Management System

- System responsible for complete lifecycle of cryptographic keys
 - Generating, distributing, revoking, expiration, and renewal
- Commonly integrated with PKI environments
 - Standalone available
- May include key escrow functionality
 - Holding “backup” keys in secure manner
 - Depends on where keys generated, purpose for keys
 - TPM, HSM can also be used

Secure Enclave

- Isolated memory location
- Allows running code within trusted execution environment (TEE)
 - Protects sensitive, secure code
- Protected and secured by dedicated hardware
- Cannot be altered or written by outside processes
- Examples:
 - Intel SGX (Software Guard Extensions)
 - built into modern Intel CPUs
 - Arm Trustzone

Cryptographic Solutions



Data Encryption

- Data can be in one of three states
 - Data at rest - on storage media
 - Data in transit (motion) - being transferred over network
 - Data in use (processing) - data present in RAM (or other volatile memory)
- File encryption
 - Single (or multiple) files on disk are encrypted
 - Can also be folder level
- Database encryption
 - Entire (or partial) database is encrypted
 - Drive and/or volume may or may not be encrypted
 - Can also encrypt specific tables, columns, records

Disk-Level Encryption

- Full disk encryption (FDE)
 - Entire contents of drive are encrypted
 - For OS drive, decryption occurs on boot
 - Keys normally stored in TPM
- Partition & volume encryption
 - Partition is raw “chunk” of a disk
 - Volume is formatted partition with drive letter assigned (in Windows)
 - Single partition or volume on disk encrypted

Hashing

- One-way function
 - Hashing cannot be reversed
 - Can be brute forced
- Algorithms are public information
- Used to verify integrity of files, messages, etc
 - Digital signature
- Strong hashing algorithms avoid “collisions”
 - Two different inputs result in same hash output
- MD5 (Message Digest 5) - 128-bit hash
- SHA (Secure Hash Algorithm) - 160-bit hash
- SHA-1, SHA-256, SHA-384, SHA-512
 - 160-bit - 512-bit hashes

Digital Signatures

- Uses asymmetric encryption
 - Public and private keys
- Author generates a hash of the created item
 - File, document, image, email, application, etc
- Author uses their own private key to digitally “sign”
 - Encrypts hash with private key
- Recipient uses author’s public key to decrypt hash and verify
- Provides authenticity, integrity verification, non-repudiation
 - Only author’s public key can decrypt the hash
 - Verified hash indicates that document was not modified

Salting

- Typically used to help secure stored password hashes
 - Hash cannot be reversed, but can be brute forced to determine original value
- “Salt” is added to original password value before hashing
 - Random characters
- Prevents attackers from using hash tables as brute force
 - Tables not computed with specific salt used
 - Cannot be used to brute force
- Different salt for each password -> identical passwords, different value
- Key stretching
 - Key generated from password/salt, repeatedly converted to longer keys
 - Not necessarily stronger key, but can slow attacks down

Obfuscation

- Steganography
 - Embedding information inside another source
 - Typically data hidden in an image file
 - Message can be encrypted before embedding
- Data masking
 - Redacting data (substituting with different character, “x”)
 - Frequently performed in database fields
- Tokenization
 - All or part of data replaced with randomly generated token
 - Token stored with original value in separate location
 - Authorized services/applications can query token vault for original data

Blockchain

- List of transactional records stored using cryptography
- Each record is a “block” and is run through a hash function
 - Hash of previous block added to hash calc of next block in chain
 - Ensure that each successive block is cryptographically linked
- One block validates hash of previous block, and so on
 - Ensures records of transactions were not tampered with
- Recorded on **open public ledger**
 - Decentralized record - mitigates risk of single point of failure/compromise
- Can be used to ensure integrity and transparency of
 - Financial transactions, legal contracts
 - Copyright and IP protections
 - Online voting, identity management

EXPERTS AT MAKING YOU AN EXPERT

