Data Security and Access Control

Dr. Amit Praseed

Cryptography

- **Cryptography** is the practice and study of techniques for secure communication in the presence of adversarial behavior
  - Constructing and analyzing protocols that prevent third parties or the public from reading private messages
  - Maintaining various aspects in information security such as data confidentiality, data integrity, authentication, and non-repudiation
- Primarily consists of two operations
  - **Encryption**: Converting human readable messages (plaintext) to unintelligible messages (ciphertext)
  - **Decryption**: Unintelligible ciphertext to human readable plaintext

Cryptography

- An encryption algorithm is a means of transforming plaintext into ciphertext under the control of a secret key.
- This process is called encryption or encipherment. We write $c = E_k(m)$, where
  - m is the plaintext
  - E is the cipher function
  - k is the secret key
  - c is the ciphertext.
- The reverse process is called decryption or decipherment, and we write $m = D_k(c)$.
- The encryption and decryption algorithms E, D are public, the secrecy of m given c depends totally on the secrecy of k.

Shift Ciphers

- This is an ancient encryption technique wherein every letter in the English alphabet is substituted by a letter k positions in front of it.
  - Eg: If k=3, A will be replaced by D, B by E… and Z by C
  - HELLO will be encrypted as KHOOR
  - The shift cipher with k=3 is often called the Caesar cipher
- Relatively easy to break
  - k can have only 26 possible values - one can simply examine all possible combinations and decrypt -- Brute Force Approach
  - Shift ciphers do not hide the statistical patterns within languages, and hence can be exploited for decryption

Substitution Ciphers

- The issue with the shift cipher is that the key space is too small (only 26)
- A substitution cipher increases the key space and provides extra secrecy
  - Assign one plaintext alphabet to any (previously unmapped) alphabet in the ciphertext
  - Plaintext alphabet: ABCDEFGHIJKLMNOPQRSTUVWXYZ
  - Ciphertext alphabet: GOYDSIPELUAVCRJWXZNHBQFTMK
  - Eg: HELLO would encrypt to the ciphertext ESVVJ
- Number of possible permutations is 26!
- However, one alphabet is always mapped to a single alphabet, so frequency analysis can still be performed

Polyalphabetic Substitution Ciphers

- Polyalphabetic Cipher: One letter in the plaintext is mapped to more than one letter in the ciphertext
  - Plaintext alphabet ABCDEFGHIJKLMNOPQRSTUVWXYZ
  - Ciphertext alphabet one TMKGOYDSIPELUAVCRJWXZNHBQF
  - Ciphertext alphabet two DCBAHGFEMLKJIZYXWVUTSRQPON
  - Plaintext letters in an odd position are encrypted using the first ciphertext alphabet, whilst the plaintext letters in even positions are encrypted using the second alphabet
  - The plaintext word HELLO is encrypted to SHLJV
  - The two L's in the plaintext are encrypted differently in the ciphertext
- We usually use more than two substitutions, so the key space is $(26!)^k$

Vignere Cipher

- A popular polyalphabetic cipher in the 1900s - uses a simple keyword which is easy to remember

T	H	I	S	I	S	A	T	E	S	T	M	E	S	S	A	G	E
S	E	S	A	M	E	S	E	S	A	M	E	S	E	S	A	M	E
L	L	A	S	U	W	S	X	W	S	F	Q	W	W	K	A	S	I

- The keyword SESAME is repeated to equal the length of the plaintext
- Encryption: T(19) + S(18) = 37%26 = 11 = L
- This encryption is also relatively easy to break once the period of the keyword is found

Permutation Ciphers

- In a substitution cipher , the number and type of symbols in the plaintext and ciphertext may be different
- In a permutation cipher, the symbols in the plaintext are shuffled, but no symbol is added or removed
  - Consider a permutation: (12345) --> (24135)
  - Plaintext: Once upon a time there was a little girl called snow white
  - Break into groups: onceu ponat imeth erewa salit tlegi rlcal ledsn owwhi te
  - Padding: onceu ponat imeth erewa salit tlegi rlcal ledsn owwhi teahb.
  - Permute: coenu npaot eitmh eewra lsiat etgli crall dlsdn wohwi atheb
  - Remove Spaces: coenunpaoteitmheewralsiatetglicralldlsdnwohwiatheb.

Permutation Ciphers

- A permutation cipher can be broken using a **chosen plaintext attack**
- For example, assume an attacker forces the system to encrypt a chosen plaintext
- Plaintext: abcdefghijklmnopqrstuvwxyz
- Ciphertext: cadbehfigjmknlorpsqtwuxvyz
- The sequence repeats (modulo 5) after every five steps and so the value of n is probably equal to five.
- We can recover the key by simply taking the first five columns of the above permutation
  - 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
  - 2 4 1 3 5 7 9 6 8 10 12 14 11 13 15

Types of Attacks against Encryption

- **Ciphertext-only attack**: The cryptanalyst has the ciphertext of several messages, all of which have been encrypted using the same encryption algorithm. The cryptanalyst's job is to recover the plaintext of as many messages as possible, or better yet to deduce the key (or keys) used to encrypt the messages, in order to decrypt other messages encrypted with the same keys.

  Given: $C_1 = E_k(P_1)$, $C_2 = E_k(P_2)$,...$C_i = E_k(P_i)$

  Deduce: Either P_1, P_2,...P_i ; k; or an algorithm to infer P_{i+1} from $C_{i+1} = E_k(P_{i+1})$

Types of Attacks against Encryption

- **Known-plaintext attack**: The cryptanalyst has access not only to the ciphertext of several messages, but also to the plaintext of those messages. His job is to deduce the key (or keys) used to encrypt the messages or an algorithm to decrypt any new messages encrypted with the same key (or keys)

  Given: $P_{1,}\ C_1 = E_k(P_1)$, P_2, $C_2 = E_k(P_2)$,...P_i , $C_i = E_k(P_i)$

  Deduce: Either k, or an algorithm to infer P_{i+1} from $C_{i+1} = E_k(P_{i+1})$

- This method was used to decipher German communication encrypted using the Enigma.
  - A daily weather report was transmitted by the Germans at the same time every day.
  - Due to the standardized style of military reports, it would contain the word Wetter (German for "weather") at the same location in every message.
  - Knowing the local weather conditions helped guess other parts of the plaintext as well

Types of Attacks against Encryption

- **Chosen-plaintext attack**: The cryptanalyst not only has access to the ciphertext and associated plaintext for several messages, but he also chooses the plaintext that gets encrypted. His job is to deduce the key (or keys) used to encrypt the messages or an algorithm to decrypt any new messages encrypted with the same key (or keys).

  Given: P_1, $C_1 = E_k(P_1)$, P_2, $C_2 = E_k(P_2)$,...P_i , $C_i = E_k(P_i)$, where the cryptanalyst gets to choose P_1, P_2,...P_i

  Deduce: Either k, or an algorithm to infer P_{i+1} from $C_{i+1} = E_k(P_{i+1})$

Types of Attacks against Encryption

- **Chosen-ciphertext attack**: The cryptanalyst can choose different ciphertexts to be decrypted and has access to the decrypted plaintext. For example, the cryptanalyst has access to a tamper-proof box that does automatic decryption. His job is to deduce the key.

  Given: C_1, $P_1 = D_k(C_1)$, C_2, $P_2 = D_k(C_2)$,...C_i , $P_i = D_k(C_i)$

  Deduce: k

Stream and Block Ciphers

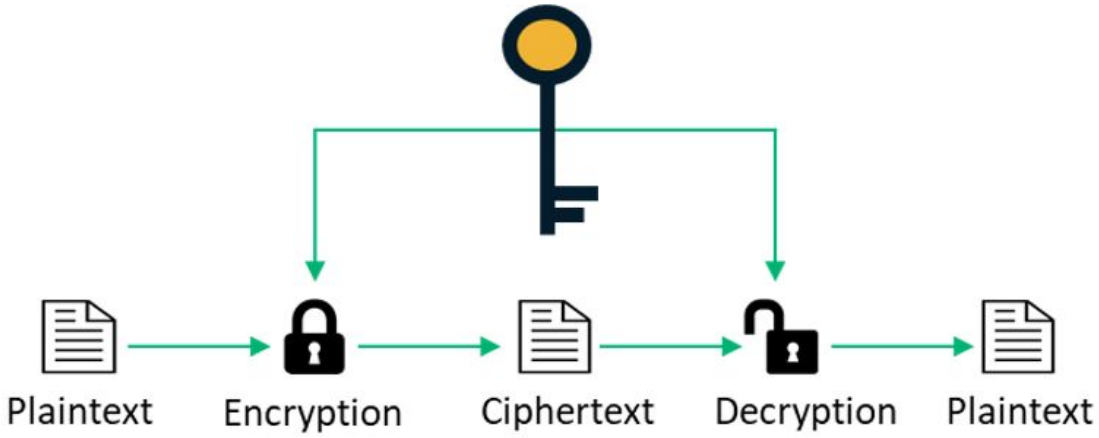
- The basic operation of encryption involves some function/operation involving the message and the key
  - Example, let the encryption operation be the XOR operation
  - Message: 101000101
  - Key: 001101011
  - Ciphertext: $101000101 \oplus 100101110 = 001101011$
- A block cipher is one that allows you to use a key to encrypt data in groups (blocks) of a pre-determined size (such as 128 bits, 256 bits, etc.)
  - If the last block is not full, padding bits are added to make it the same length as the block size
  - Ciphertext blocks may be XORed with the next plaintext block to create even stronger encryption through cipher chaining
- Eg: DES, AES etc.

Stream and Block Ciphers

- Stream ciphers encrypt data in long, pseudorandom streams
  - Process one bit of data at a time instead of waiting for a data block to form
- A stream of pseudorandom bits are generated based on an encryption key and a seed (nonce)
  - Together, they create a keystream that gets XORed with your plaintext input, which encrypts it and results in your ciphertext output.
- Eg: Salsa20, ChaCha20, RC4 (for wireless networks), A5 (for GSM cellular networks).

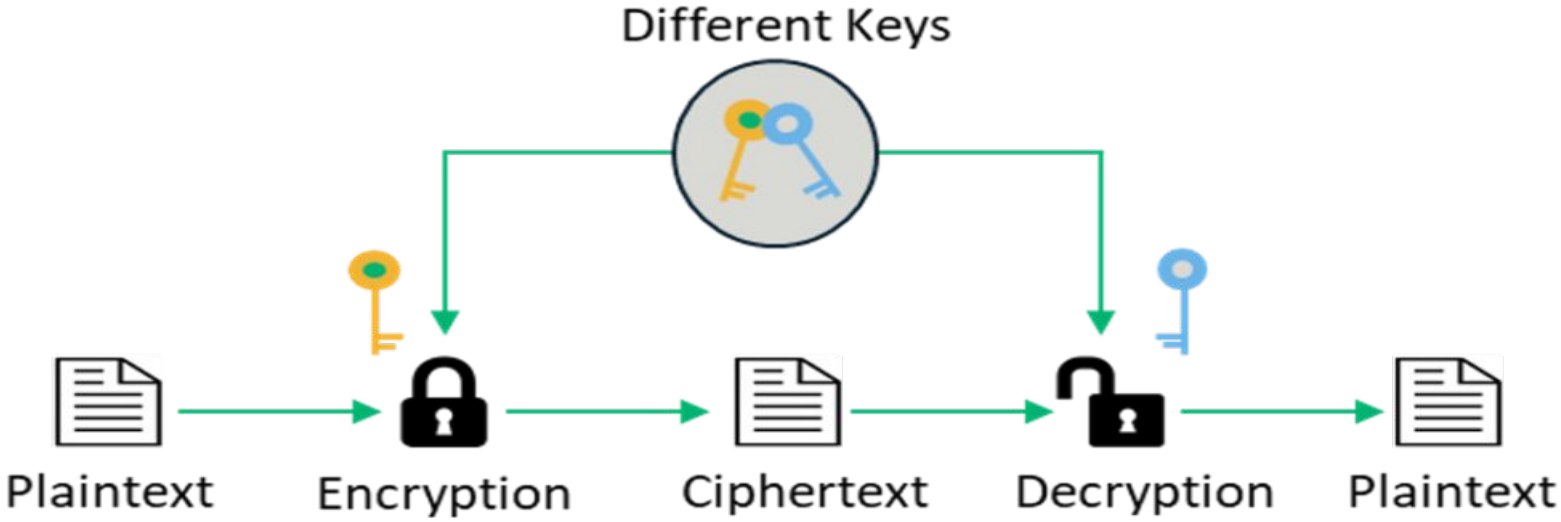
Symmetric and Asymmetric Encryption

- In **symmetric encryption**, the same key is used for both encrypting and decrypting messages
  - Entire mechanism is dependent on keeping the key a shared secret
  - It does not scale well



Symmetric and Asymmetric Encryption

- Asymmetric encryption uses a pair of related keys — a public and a private key
  - The public key is accessible to everyone and is used to encrypt a plaintext message before sending it
  - To decrypt and read this message, you need to hold the private key



Symmetric and Asymmetric Encryption

- Asymmetric encryption involves the use of two mathematically related keys.
  - The public key (the one that’s known to everybody) and the private key (which is only known by you) are required for encrypting and decrypting the message.
  - The private key cannot be derived from the public key.
- The public key is used by others to encrypt the messages they send to you, but to decrypt and read these messages, one needs access to the private key.

Feature	Symmetric Encryption	Asymmetric Encryption
Number of Keys	Only one key	Public and Private Key
Complexity	Simple, Quick	Complicated, Slower
Key Size	Typically 128 or 256 bits	Recommended RSA key size is 2048 bits or higher
Usage	Used when large chunks of data need to be transferred.	Used in smaller transactions, primarily to authenticate and establish a secure communication channel prior to the actual data transfer.
Security	The secret key is shared, hence risk of leakage	The private key is not shared, hence more secure
Examples	RC4, AES, DES, 3DES, etc.	RSA, Diffie-Hellman, ECC, etc.

Access Control Principles

- Access control specifies who or what may have access to each specific system resource and the type of access that is permitted in each instance
- This context involves the following entities and functions:
  - Authentication: Verification that the credentials of a user or other system entity are valid
  - Authorization: The granting of a right or permission to a system entity to access a system resource. This function determines who is trusted for a given purpose
  - Audit: An independent review and examination of system records and activities

Access Control Policies

- Discretionary access control (DAC): Controls access based on the identity of the requestor and on access rules (authorizations) stating what requestors are (or are not) allowed to do.
- Mandatory access control (MAC): Controls access based on comparing security labels with security clearances
- Role-based access control (RBAC): Controls access based on the roles that users have within the system and on rules stating what accesses are allowed to users in given roles
- Attribute-based access control (ABAC): Controls access based on attributes of the user, the resource to be accessed, and current environmental conditions.

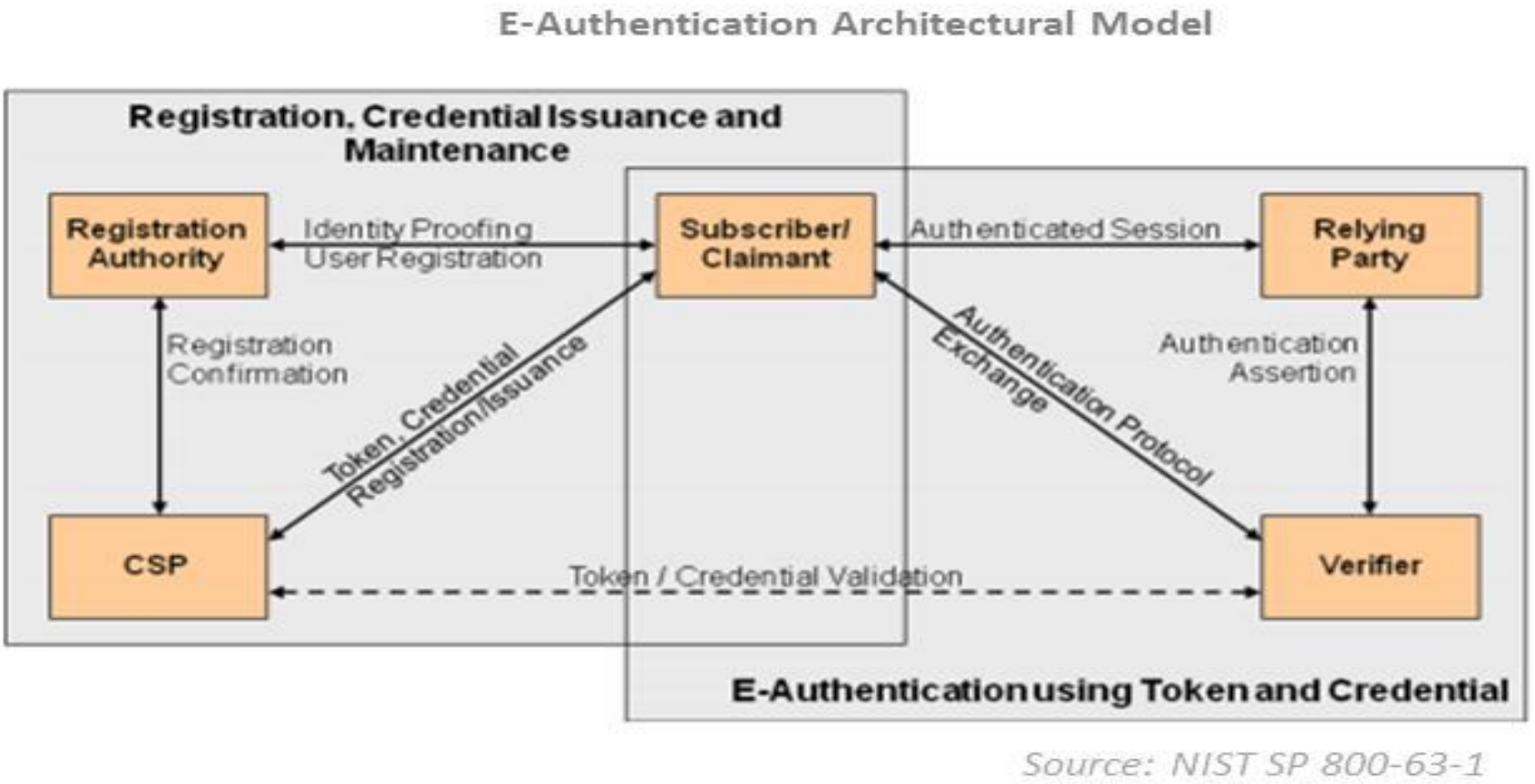
Subjects, Objects and Rights

- A subject is an entity capable of accessing objects.
- A subject is usually characterized as belonging to a class or group
  - Owner: This may be the creator of a resource, such as a file
  - Group: In addition to the privileges assigned to an owner, a named group of users may also be granted access rights, such that membership in the group is sufficient to exercise these access rights
  - World: The least amount of access is granted to users who are able to access the system but are not included in the categories owner and group for this resource
- An object is a resource to which access is controlled

Subjects, Objects and Rights

- An access right describes the way in which a subject may access an object. Access rights could include the following
  - Read: User may view information in a system resource (e.g., a file, selected records in a file, selected fields within a record, or some combination). Read access includes the ability to copy or print
  - Write: User may add, modify, or delete data in system resource (e.g., files, records, programs). Write access includes read access
  - Execute: User may execute specified programs
  - Delete: User may delete certain system resources, such as files or records
  - Create: User may create new files, records, or fields
  - Search: User may list the files in a directory or otherwise search the directory

User Authentication



User Authentication

- An applicant applies to a registration authority (RA) to become a subscriber of a credential service provider (CSP)
- RA is a trusted entity that establishes and vouches for the identity of an applicant to a CSP
- The CSP then engages in an exchange with the subscriber
  - The CSP issues some sort of electronic credential to the subscriber
  - The credential is a data structure that authoritatively binds an identity and additional attributes to a token possessed by a subscriber, and can be verified when presented
- Once a user is registered as a subscriber, the actual authentication process can take place between the subscriber and one or more systems that perform authentication and, subsequently, authorization.

User Authentication

- The party to be authenticated is called a claimant and the party verifying that identity is called a verifier. W
- hen a claimant successfully demonstrates possession and control of a token to a verifier through an authentication protocol, the verifier can verify that the claimant is the subscriber named in the corresponding credential.
- The verifier passes on an assertion about the identity of the subscriber to the relying party (RP)
    - Includes identity information about a subscriber, such as the subscriber name, an identifier assigned at registration, or other subscriber attributes that were verified in the registration process
    - The RP can use the authenticated information provided by the verifier to make access control or authorization decisions.

Means of User Authentication

- Something the individual knows: Examples includes a password, a personal identification number (PIN), or answers to a prearranged set of questions
- Something the individual possesses: Examples include electronic keycards, smart cards, and physical keys. This type of authenticator is referred to as a token
- Something the individual is (static biometrics): Examples include recognition by fingerprint, retina, and face
- Something the individual does (dynamic biometrics): Examples include recognition by voice pattern, handwriting characteristics, and typing rhythm

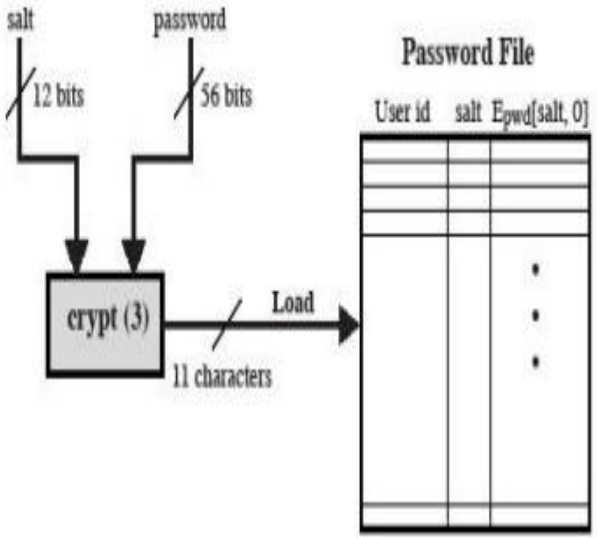
Password based Authentication

- User provides an identifier and password
- The system compares the password to a previously stored password for that user ID, maintained in a system password file
  - The password serves to authenticate the ID of the individual logging on to the system
  - ID determines whether the user is authorized to gain access to a system
  - The ID determines the privileges accorded to the user
  - The ID is used in discretionary access control

Vulnerabilities of Password based Authentication

- Offline dictionary attack
- Specific account attack
- Popular password attack
- Password guessing against single user
- Workstation hijacking
- Exploiting user mistakes
- Exploiting multiple password use
- Electronic monitoring

UNIX Password Mechanism



(a) Loading a new password