7.2.5 Cryptographic Implementation Facts

This lesson covers the following topics:

- Hybrid models
- Digital signatures
- Encryption with steganography
- Hardware-based encryption

Hybrid Models

Operating systems, applications, and other components of information systems typically use a hybrid cryptography system. A hybrid cryptography system combines the strengths of hashing, symmetric encryption, and asymmetric encryption depending on the needs of the project or service. An example of these strengths are:

- Use symmetric encryption for fast and efficient encryption of bulk data.
- Use hashing to verify message integrity.
- Use asymmetric encryption for authentication and non-repudiation.
- Use asymmetric encryption for secure exchange of symmetric encryption keys (for example, by encrypting the key used for symmetric encryption prior to sharing the key with the recipient). Using asymmetric cryptography for encryption is best for small pieces of data.

A hybrid cryptosystem combines the efficiency of symmetric methods and the convenience of asymmetric methods. One example of this is Microsoft's Encrypting File System, or EFS. Microsoft's EFS uses the following steps to encrypt data:

1. A file is encrypted using a File Encryption Key (FEK).
2. The FEK is encrypted with the user's public asymmetric key.
3. The file is sent to the intended recipient.
4. The user's private key is used to decrypt the FEK.
5. The FEK is used to decrypt the file.

One of the biggest weaknesses of the EFS is that the user's private key is essentially their user password. If the password is weak, the encryption will also be weak.

Digital Signatures

One very common practice that combines cryptographic methods is the digital signature. A digital signature combines the hash of a file and a user’s private key to electronically sign a document. This provides authentication and non-repudiation of the file.

Signing Concept	Description
Digital Signature	A digital signature is a combination of asymmetric encryption and hashing values. A signature provides confidentiality, integrity validation, strong authentication, and non-repudiation. Typically, a digital signature works as follows: 1. A hash value is generated for a message. 2. The hash value is asymmetrically encrypted using the sender's private key. Non-repudiation

	is provided because only the sender could have encrypted the hash using the private key (only the sender knows the private key). 3. The encrypted hash value and the message are sent. 4. The recipient decrypts the hash using the sender's public key. 5. The recipient hashes the message. 6. Message integrity and sender authenticity (non-repudiation) is confirmed if the two hash values match.
Hashing	Hashing is the process of using an algorithm, like MD5 or SHA, on data and generating a fixed-length key called a hash. The three main hashing algorithms used today are: ▪ SHA-1, which generates a 160-bit key ▪ MD5, which generates a 128-bit key ▪ SHA-2 (SHA-256), which can generates a 256-bit key
Digital Envelope	In addition to digital signatures, data can be protected by using secure data transmission. This protects the message from hackers by using asymmetric encryption to secure the message before sending it to the recipient. Secure data transmission uses the following process: 1. The sender requests a copy of the recipient's public key. 2. The recipient or CA sends a digital certificate containing the public key to the sender. 3. The sender asymmetrically encrypts the message using the recipient's public key. 4. The sender sends the asymmetrically encrypted message to the recipient. 5. The recipient uses his private key to decrypt the message.

Encryption with Steganography

There has been an increased amount of research done on how to best combine encryption with steganography. The process essentially follows the steps below:

1. Encrypt plaintext with a private key to generate ciphertext.
2. The ciphertext is hidden inside of a media file, such as an image, using steganography.
3. The recipient extracts the ciphertext and decrypts it using the matching public key.
4. Because the ciphertext is hidden in the image file, someone intercepting the message would have to know its there before being able to decrypt it.

Hardware-Based Encryption

Hardware devices can be combined with software-based encryption to offer powerful cryptography options. The following table shows the two most common hardware cryptography devices - TPM and HSM.

Hardware Cryptography Device	Description
Trusted Platform Module (TPM)	Trusted Platform Module (TPM) is a hardware chip on the motherboard that can generate and store cryptographic keys. TPM version 2.0 was released in 2014. Beginning with Windows 10 version 1607, Microsoft required that TPM 2.0 be enabled by default on all new computers. ▪ A TPM is required to check the integrity of startup files and components in BitLocker implementations.

	▪ The TPM generates a hash of the startup files to verify the integrity of those files. ▪ Additionally, the TPM creates a hash of system components. This hash acts as a validation check of the system to ensure that system components have not changed. The hash can also be used to uniquely identify the system. ▪ Windows Credential Guard requires the computer to have a TPM chip installed. ▪ A TPM provides protection for virtual-based security encryption keys that are stored in the firmware. This helps protect against attacks involving a physically present user with BIOS access. ▪ A TPM can generate truly random numbers, thus preventing entropy. ▪ TPM provides full support for asymmetric encryption; therefore, it can generate public and private keys. ▪ A TPM also provides encrypted storage for user passwords, encryption keys, and digital certificates. ▪ Windows 10 can pull stored keys directly from the TPM without loading them into the RAM where they would be more vulnerable to an attack.
Hardware Security Module (HSM)	A Hardware Security Module (HSM) is a piece of hardware and associated software/firmware that is connected to a computer system to provide cryptographic functions such as: ▪ Generate and store encryption keys ▪ Generate and validate digital signatures ▪ Generate keys used in smart cards HSMs traditionally come in the form of a plug-in card or an external security device that can be attached directly to the computer system. These devices offer some benefits over TPM chips. ▪ HSMs are more powerful and can perform more powerful cryptographic functions quicker. ▪ HSMs can perform multiple cryptographic functions simultaneously. ▪ HSMs can be attached to a network and handle cryptographic functions for multiple users across the network. Hardware Security Modules are also known as: ▪ Personal Computer Security Module (PCSM) ▪ Secure Application Module (SAM) ▪ Hardware cryptographic devices ▪ Cryptographic modules