Case Studie

1. Implementation of Digital Signatures in Secure Email Communication

- **Digital Signatures in Secure Email**: Digital signatures provide secure email communication by ensuring that messages are authentic, unaltered, and cannot be repudiated by the sender. This is achieved by hashing the email content and encrypting it with the sender's private key. The recipient can verify the sender's authenticity and the message's integrity by decrypting the hash with the sender's public key.
- **Public Key Infrastructure (PKI)**: PKI is a framework that supports encryption and digital signatures by managing keys and certificates. It ensures that users can securely exchange information without risk of impersonation or tampering.
- **X.509 Certificates**: These are digital certificates that verify the identity of the certificate holder. In secure emails, X.509 certificates authenticate the sender and validate their public key, establishing trust between parties.

2. The Impact of Ransomware Attacks on Critical Infrastructure

- **Example Attack on Hospital/City Network**: A notable ransomware attack was the WannaCry incident, which impacted healthcare systems globally, including the UK’s NHS. Attackers used a vulnerability in the Windows operating system to propagate the malware quickly.
- **Malware Types and Propagation**: Ransomware often uses malicious email attachments, exploit kits, or vulnerabilities like EternalBlue (as in WannaCry) to spread through networks. Once inside, it encrypts files and demands a ransom.
- **Countermeasures**: To recover, affected systems isolated infected machines, restored files from backups, and patched vulnerabilities. Regular system updates, employee training, and robust backup systems are essential preventative measures.

3. Integer Overflow Vulnerabilities in Financial Systems

- **Vulnerability and Exploitation**: An integer overflow occurs when a calculation exceeds the storage capacity of the data type used, causing unpredictable results. In financial systems, this might lead to incorrect transaction processing or calculations, impacting financial records.
- **Impact**: If exploited, an attacker could manipulate transaction values, potentially leading to financial losses or fraudulent transactions.

- **Countermeasures**: Proper input validation, bounds checking, and choosing appropriate data types can mitigate integer overflow risks. Regular code reviews and testing for edge cases are essential practices.

4. Buffer Overflow Attack on Web Servers: A Case Study of Apache Server Vulnerability

- **Apache Vulnerability Example**: The Apache Struts vulnerability (CVE-2017-5638) involved a buffer overflow due to insufficient input validation. Attackers could send malicious payloads to overflow memory and execute arbitrary code.
- **Exploitation Technique**: Attackers would use crafted HTTP requests to exploit the vulnerability, gaining control over the server to steal data or disrupt services.
- **Impact and Security Patches**: Exploited servers could face data breaches or become launchpads for further attacks. Apache issued patches to resolve this vulnerability, and administrators were urged to apply updates promptly.

5. SQL Injection Attack on E-Commerce Websites

- **Real Incident Example**: A well-known SQL Injection case is the 2014 breach of the online retail site Magento. Attackers injected malicious SQL statements into input fields to access sensitive database information.
- **Execution and Types of SQL Injection**: Attackers typically exploit improperly sanitized input fields, using methods like Union-based or Boolean-based injections to extract data.
- **Consequences**: Compromised personal and payment information, leading to data breaches, financial loss, and damaged reputation.
- **Countermeasures**: Input validation, parameterized queries, and prepared statements can prevent SQL injections. Regular code audits and using web application firewalls (WAFs) are also effective security practices.

Case studie 2:

// submitted in class