2019-exam 1

A major hardware vendor provides firmware updates as binary files on its website. To provide integrity guarantees, the SHA-224 hash of each binary file is also provided on the web-site; however, the web-site supports only the http protocol, not https.
1. Explain why this scheme is insufficient to provide an integrity guarantee
2. Propose a change so that the mechanism does in fact provide an integrity guarantee. You may assume a CA.
Answer this question in at most 500 words.

1. Why the Scheme is Insufficient for Integrity Guarantees

The current scheme, which uses HTTP to transmit firmware binary files along with their SHA-224 hash, is insufficient for providing an integrity guarantee due to several key issues:

1. **Lack of Encryption in HTTP**: HTTP transmits data in plaintext, meaning that both the firmware and its associated hash can be intercepted, modified, or replaced during transmission. A man-in-the-middle (MITM) attacker could alter the firmware binary or the hash, potentially delivering a malicious or corrupted file while making it appear legitimate.
2. **Tampering of the Hash**: Since the SHA-224 hash is transmitted alongside the firmware file over HTTP, an attacker could modify both the binary file and its hash. This could go undetected because the attacker could change the hash to match the tampered firmware, causing the user to mistakenly believe the firmware is intact.
3. **No Authentication of the Vendor**: HTTP does not provide authentication for the server. Without verifying the authenticity of the vendor's website, a malicious actor could set up a fake site, tricking the user into downloading a malicious firmware update that has a matching hash value.

2. Proposed Change for Integrity Guarantee

To address these shortcomings and ensure integrity, the mechanism should be modified as follows:

Use HTTPS for Secure Communication:
Switch from HTTP to HTTPS. HTTPS encrypts communication using SSL/TLS, which protects against MITM attacks by ensuring that both the firmware and its hash are securely transmitted. This prevents attackers from altering the data in transit.

Sign the SHA-224 Hash with a Private Key:
The vendor should cryptographically sign the SHA-224 hash of the firmware file with their private key. This ensures that the hash cannot be tampered with. The signed hash would then be provided alongside the firmware file.

Verify the Signature:
The user can verify the integrity of the firmware by calculating its SHA-224 hash and checking the signature using the vendor’s public key. If the signature matches the hash, the user can be confident that the firmware is authentic and unmodified.

Use a Certificate Authority (CA) for Authentication:
The vendor's website should be secured with an SSL/TLS certificate issued by a trusted Certificate Authority (CA). This ensures the website's authenticity, preventing attackers from impersonating the vendor's site.

Summary of the Improved Mechanism:

1. The vendor signs the SHA-224 hash of the firmware using their private key.
2. The user downloads the firmware and the signed hash via HTTPS.
3. The user verifies the hash by computing it and validating the signature with the vendor's public key.
4. HTTPS encrypts the communication, preventing tampering during transmission.
5. A CA-issued certificate ensures the authenticity of the vendor's website.

By implementing these changes, the process would ensure both the integrity and authenticity of the firmware updates, protecting users from tampered or malicious files.

2019-exam 2

Your answer to this question counts 15% towards the final grade.
A bank uses NemID as a service for authenticating its customers. NemID allows the bank's customer to login using either single-factor authentication or two-factor authentication. The customer provides username and password for single-factor authentication, while, for two-factor authentication, they provide username, password, and a onetime code from a small paper that comes with their NemID. An authenticated customer can make dispositive operations (e.g. payments, create loans, sign agreements) or documentary operations (e.g. print bank statements, overview of existing loans and agreements).
The bank wants to create an access control policy that allows
a customer who authenticates via single-factor authentication to make only documentary operations;
a customer who authenticates via two-factor authentication to make dispositive and documentary operations.
Answer the following questions:
1. Explain why Discretionary Access Control (DAC) is not the most appropriate model to enforce the bank's policy
2. Identify and motivate the adoption of a more appropriate access control model, and provide a high-level specification of the policy, which fits better the chosen access control model
Answer this question in at most 500 words.

1. Why Discretionary Access Control (DAC) is Not the Most Appropriate Model

Discretionary Access Control (DAC) is a model in which the owner of a resource (such as a bank account) determines who can access the resource and what type of operations are allowed. In DAC, access decisions are often made based on user identities and are flexible, allowing resource owners to freely share access with others. While this model provides ease of management and flexibility, it is not suitable for enforcing the bank's access control policy for the following reasons:

1. **Lack of Centralized Control**: In DAC, users have the discretion to control access to their own resources. This would conflict with the bank's requirement to enforce strict access rules based on the method of authentication. For example, if a customer were to authenticate using single-factor authentication, there could be a risk that they might be granted broader access to dispositive operations, contrary to the policy. The bank would have limited control over the exact permissions granted to the customer.
2. **Risk of Unauthorized Operations**: Since DAC allows users to assign permissions to other users, a user with single-factor authentication could potentially grant themselves or others access to dispositive operations, undermining the integrity of the policy. The bank cannot fully guarantee that users will comply with the specified policy without imposing additional controls.
3. **Inconsistent Enforcement**: DAC would make it difficult to uniformly enforce the access control policy across the entire system, especially when the policy is based on the method of authentication (single-factor vs. two-factor). DAC does not inherently offer a way to bind access permissions to authentication methods, which is a key requirement for the bank's policy.

2. Adoption of a More Appropriate Access Control Model

A more suitable access control model for this situation is **Role-Based Access Control (RBAC)**. RBAC is based on the idea that users are assigned roles that define their access rights, and each role has predefined permissions associated with it. The bank can create specific roles based on the authentication method used (single-factor or two-factor) and restrict access to operations accordingly.

Motivation for RBAC

- **Centralized Control**: RBAC allows the bank to centrally define roles and access rights. The bank can assign customers to specific roles based on their authentication method and ensure that access is strictly regulated according to those roles.
- **Clear Separation of Operations**: The bank's policy distinguishes between documentary and dispositive operations, which can easily be mapped to roles in RBAC. By defining roles for single-factor and two-factor authenticated users, the bank can ensure that the right operations are permitted based on the authentication method.
- **Enforcement of Security Policies**: RBAC allows for better enforcement of security policies. The bank can prevent customers who authenticate via single-factor authentication from performing dispositive operations, ensuring that sensitive operations like payments and loan creation are only available to those who authenticate with two-factor authentication.

High-Level Specification of the Policy

- **Role 1: Single-Factor Authenticated User**
  Permissions:

    - Documentary operations only (e.g., print bank statements, view loans and agreements).
    - **No access to dispositive operations** (e.g., payments, create loans, sign agreements).
- **Role 2: Two-Factor Authenticated User**
  Permissions:
    - Both documentary and dispositive operations (e.g., print bank statements, make payments, create loans, sign agreements).
- **Access Control Rules**:
    - A user who successfully authenticates via **single-factor authentication** is assigned **Role 1** and granted access to documentary operations.
    - A user who successfully authenticates via **two-factor authentication** is assigned **Role 2** and granted access to both documentary and dispositive operations.
    - The bank system enforces these roles strictly, ensuring that the permissions for each role are respected.

By implementing RBAC, the bank can ensure that its access control policy is consistently enforced, based on the customer's authentication method. This model provides clear, centralized control over access rights and reduces the risk of unauthorized operations.

2018S-EXAM 1

The IT department is considering to introduce multi-step authentication to protect their users. In the proposed multi-step authentication, each user picks any two authentication methods from the list below at each login.1. Password
2. Fingerprint recognition
3. Student card
4. SMS via phone
5. Call via phone
6. Email confirmation
7. Iris Scanning
Give a recommendation: Should this policy be adopted or not? Discuss and motivate your recommendation.

Recommendation: Do Not Adopt the Proposed Multi-Step Authentication Policy

While the introduction of multi-step authentication (MSA) can strengthen user security by requiring multiple authentication factors, the specific policy proposed here has critical flaws. It allows users to choose any two methods from a diverse list, which introduces inconsistencies, potential security vulnerabilities, and usability challenges. Below is a detailed discussion of the issues and a recommendation for a more robust approach.

Issues with the Proposed Policy

1. **Security Variability Across Combinations**
   Allowing users to choose any two methods creates significant variability in security levels. For example:
   - Combining **password** (something you know) and **email confirmation** (another "something you know") is weaker than combining **password** and **fingerprint recognition** (something you have and something you are).
   - A user might select two relatively weak methods (e.g., **password** and **email confirmation**) instead of stronger combinations (e.g., **password** and **iris scanning**).
2. **User Behavior and Convenience**
   - Users often prioritize convenience over security. Many may opt for the easiest methods to use, such as **password** and **email confirmation**, leaving stronger methods like **biometrics** or **SMS via phone** unused. This undermines the goal of increasing security.
   - The diversity of methods could confuse users and increase the likelihood of errors during login, negatively impacting the user experience.
3. **Implementation Complexity**
   Supporting seven distinct authentication methods, with the flexibility for users to pick any two, significantly increases system complexity. This includes:
   - Development and integration costs for multiple authentication systems.
   - Higher maintenance overhead and potential compatibility issues.
   - Greater likelihood of security flaws due to increased system complexity.
4. **Inconsistent Assurance Levels**
   - Authentication methods like **passwords** and **email confirmation** are inherently weaker due to susceptibility to phishing or brute-force attacks.
   - Combining weak methods does not achieve the desired security level and could be exploited by attackers.
5. **Lack of Multi-Factor Authentication Principle**
   True multi-factor authentication (MFA) relies on combining factors from at least two distinct categories:
   - **Something you know** (e.g., password).
   - **Something you have** (e.g., student card, SMS).
   - **Something you are** (e.g., fingerprint, iris scan).
     The proposed system does not enforce this principle, allowing users to select two methods from the same category (e.g., password and email), which diminishes its security benefits.

Recommendation

The IT department should revise the policy to enforce **true multi-factor authentication (MFA)** rather than allowing users to freely choose methods. A recommended policy is as follows:

1. **Mandatory Use of Two Different Factors**
   Users must authenticate using two methods from different categories:
   - **Knowledge-based** (password).
   - **Possession-based** (SMS, phone call, student card).
   - **Biometric-based** (fingerprint, iris scan).

2. **Default Method Pairing**
   Define default, secure combinations (e.g., password + fingerprint or password + SMS). Allow limited flexibility for users to select alternatives, but only if the chosen methods maintain the MFA principle.
3. **Eliminate Redundant or Weak Methods**
   Remove weaker methods like **email confirmation** from the list, as they are less secure and redundant.
4. **User Training and Usability**
   Educate users about the importance of MFA and provide clear, simple instructions to ensure usability without compromising security.

Conclusion

The proposed MSA policy should **not be adopted** due to its lack of security consistency, implementation complexity, and violation of MFA principles. Instead, the IT department should enforce true MFA with predefined secure combinations, ensuring robust protection while maintaining usability. This approach balances security and practicality, safeguarding users effectively.

2018S-EXAM 2

The IT department at ITU is considering new password rules to protect their

users. Any password must satisfy all of the following rules:

1. Passwords must be longer than 8 characters

2. Passwords should not be dictionary words

3. Passwords must contain at least two non-consecutive digits, two non-

consecutive capital letters, and exactly one special character

4. Passwords must be changed at least every semester

5. “Paste” functionality is disabled in every login page related to ITU when

entering a password.

Give a recommendation: Should this policy be adopted or not? Discuss and

motivate your recommendation.

Recommendation: Do Not Adopt the Proposed Password Policy Without Revisions

While the proposed password rules aim to enhance security, they suffer from usability issues, impractical requirements, and potential security drawbacks. A more balanced approach is necessary to achieve both strong security and user compliance. Below, I discuss the challenges and propose an improved policy.

Issues with the Proposed Policy

1. Password Length and Complexity (Rules 1 and 3)

- **Strength**: Requiring passwords longer than 8 characters with complex criteria improves resistance to brute-force and dictionary attacks. However, the specific requirements for two **non-consecutive digits**, two **non-consecutive capital letters**, and exactly one **special character** are overly rigid and could lead to usability issues:
  - Users may struggle to create and remember such passwords, leading to weaker security practices (e.g., writing passwords down, reusing passwords across platforms).
  - Non-consecutive rules add unnecessary complexity, which may frustrate users without providing significant security benefits. Attackers can still brute-force passwords with complex rules given enough time.

2. Dictionary Words (Rule 2)

- While avoiding dictionary words improves security, enforcing this rule strictly may block legitimate, strong passwords like passphrases (e.g., "BlueMonkey$72"), which are easy to remember and hard to guess.

3. Frequent Password Changes (Rule 4)

- Forcing users to change passwords every semester (approximately every 6 months) may reduce security:
  - Frequent changes can cause users to adopt predictable patterns (e.g., incrementing numbers or reusing old passwords), which attackers can exploit.
  - NIST (National Institute of Standards and Technology) guidelines discourage periodic password changes unless there is evidence of compromise.

4. Disabling “Paste” Functionality (Rule 5)

- Disabling “paste” functionality undermines security:
  - It prevents users from using password managers, which are highly effective at generating and storing strong passwords. Password managers reduce the likelihood of reused or weak passwords.
  - This rule increases the chance of typos and user frustration, especially for long and complex passwords.

Recommendation for an Improved Policy

To balance security and usability, the IT department should revise the policy as follows:

1. Password Length and Complexity

- Require a minimum password length of 12 characters. Encourage the use of passphrases (e.g., "Coffee&Skyline2025") for easier memorization and strong entropy.
- Enforce at least three of the following criteria to ensure flexibility:
    - At least one uppercase letter.
    - At least one lowercase letter.
    - At least one digit.
    - At least one special character.

2. Avoiding Weak Passwords

- Use automated password strength checks to block common passwords (e.g., "password123") or those found in breach databases, rather than strictly banning dictionary words.

3. Password Expiration

- Remove the requirement for periodic password changes. Instead, require password changes only if there is evidence of compromise or suspicious activity.

4. Enable Password Managers

- Allow users to paste passwords to support the use of password managers, which significantly enhance security and usability.

5. Multi-Factor Authentication (MFA)

- Strengthen security by implementing MFA. Even a strong password can be compromised, but MFA provides an additional layer of protection.

Conclusion

The proposed policy is overly rigid, impractical, and counterproductive in some aspects. It should **not be adopted without revision**. A revised policy emphasizing passphrase use, password managers, and MFA will provide better security while maintaining user compliance and usability. This approach aligns with modern security best practices, such as those recommended by NIST.

2018S-EXAM 3

A vendor of software solutions for practicing doctors ("GPs", "praktiserende læger") offers an on-line appointment module. Patients authenticate themselves to the system with their CPR-number and an automatically-generated password that the doctor provides to the patient over the phone or during a consultation. Once authenticated, the patient is shown a list of open slots in the immediate

future, and a link to PDF-version of his patient records.

Write an abbreviated risk analysis for the on-line appointment module.

Abbreviated Risk Analysis for the Online Appointment Module

1. Assets

- **Patient Data**: Includes medical records, personal information, and CPR-number (highly sensitive).
- **Appointment Slots**: Scheduling data.
- **Authentication Credentials**: CPR-number and auto-generated password.

2. Threats

1. **Unauthorized Access**:
   - Attackers could guess or intercept CPR numbers and passwords to gain access to sensitive data.
2. **Insecure Authentication**:
   - The use of CPR numbers (publicly known in some cases) and simple passwords provided over the phone or during consultations introduces vulnerabilities.
   - Lack of multi-factor authentication (MFA) increases the risk of compromise.
3. **Data Breach**:
   - Compromise of the database storing patient data or appointment slots could lead to large-scale exposure of sensitive information.
4. **Man-in-the-Middle (MITM) Attacks**:
   - If the system uses insecure communication channels (e.g., HTTP instead of HTTPS), attackers could intercept login credentials or sensitive data during transmission.
5. **Unauthorized Access to Patient Records**:
   - Links to PDF versions of records could be accessed by attackers if not properly protected by session controls or authorization mechanisms.
6. **Denial of Service (DoS) Attacks**:
   - Attackers could overload the system, preventing patients from booking appointments.
7. **Social Engineering**:
   - An attacker could impersonate a patient and request login credentials from the doctor.

3. Vulnerabilities

1. **Weak Authentication Mechanism**:
   - CPR numbers are often predictable, and passwords provided over the phone may not meet complexity requirements, making them easier to compromise.
2. **Session Management Issues**:
   - Poorly implemented session controls could allow session hijacking, unauthorized access, or data leakage.
3. **Lack of Encryption**:

    - If communication or data storage is not encrypted, sensitive information is exposed to interception.
4. **Poor Authorization Controls**:
    - Weak authorization mechanisms could allow patients to access other users' data or records.

4. Impact Assessment

1. **Patient Privacy Violation**:
    - Unauthorized access to medical records could lead to legal, reputational, and financial damages for the doctors and the vendor.
2. **Loss of Trust**:
    - Patients may lose trust in the system, reducing its adoption and effectiveness.
3. **Regulatory Penalties**:
    - Non-compliance with data protection laws (e.g., GDPR) could result in substantial fines and legal actions.

5. Risk Mitigation Recommendations

1. **Strengthen Authentication**:
    - Replace CPR-number and password with a secure login mechanism, such as:
        - A unique username/password combination.
        - Multi-factor authentication (e.g., SMS codes, email confirmation).
2. **Encrypt Communication and Data**:
    - Use HTTPS for all communications and encrypt sensitive data in storage and transit.
3. **Enhance Password Security**:
    - Enforce strong password policies for auto-generated passwords, including minimum complexity and expiration rules.
4. **Implement Robust Authorization Controls**:
    - Ensure patients can access only their own appointment slots and medical records.
5. **Secure Patient Records Links**:
    - Use time-limited, session-bound links to ensure PDF access is restricted and secured.
6. **Monitor and Log Activity**:
    - Implement logging and monitoring to detect suspicious activities and unauthorized access attempts.
7. **Educate Staff and Patients**:
    - Train doctors and administrative staff to verify the identity of patients when issuing credentials. Educate patients about secure password usage and the risks of sharing credentials.

6. Conclusion

The online appointment module poses significant risks due to its weak authentication mechanism and potential for data breaches. By implementing stronger authentication, encryption, and authorization controls, and by addressing other identified vulnerabilities, the vendor can significantly reduce the risk of compromise while ensuring patient data privacy and compliance with legal obligations.

2018F-EXAM 1

A big European car manufacturer wishes to add wireless firmware update capabilities to its
top-line models. Each car comprises two distinct but communicating computers:
The Driving Controller, which controls engine and brakes during actual driving, taking inputs from pedals, steering wheel, etc.
The Infotainment System, which among other functions contains a GPS. Only the Infotainment System has wireless capabilities.
The manufacturer plans on having the Infotainment System of each car wirelessly querying a
central server for firmware updates. Such an update comprises new programming for both
the Driving Controller and the Infotainment System. If there is an update, the Infotainment
System downloads the update, forwards it to the Driving Controller, and both systems replace their firmware with the updated one. The query and update is scheduled to happen automatically every night at 23:55 provided the car is at a stop and the engine is turned off.
The manufacturer plans on neither confidentiality nor integrity checks for the update process.

1. Assume the firmware is transmitted in the clear and not integrity protected. Outline a

denial-of-service attack on this system.

2. Assume that the firmware itself is not confidential. Describe how to add integrity

protection to the firmware update system using cryptographic primitives.

3. Assume that the firmware itself is confidential. Describe how to additionally add

confidentiality protection to the firmware update system.

Report your answers to these three questions. Submit at most 500 words in total.

Answer to Question D1

1. Denial-of-Service (DoS) Attack

If the firmware is transmitted in cleartext and lacks integrity protection, attackers can exploit this vulnerability for a DoS attack:

- **Attack Method**:
    - The attacker intercepts the firmware update communication between the car and the central server using techniques such as packet sniffing or a Man-in-the-Middle (MITM) attack.
    - The attacker corrupts the firmware file during transmission by altering or replacing it with invalid data.
    - When the Infotainment System or Driving Controller attempts to install the corrupted firmware, the process fails, rendering the system inoperable.
- **Impact**:
    - The Driving Controller, responsible for critical safety functions, may become unresponsive, potentially immobilizing the vehicle.
    - Repeated corruption of updates prevents the system from receiving valid firmware, leading to long-term disruption.

---

2. Adding Integrity Protection to the Firmware Update System

To ensure integrity, cryptographic mechanisms such as hashing and digital signatures should be implemented:

1. **Generate Hash**:
    - The manufacturer calculates a cryptographic hash (e.g., SHA-256) of the firmware file before transmission.
    - The hash ensures any alteration to the firmware during transmission can be detected.
2. **Sign the Hash**:
    - The manufacturer signs the hash using its private key, generating a digital signature. This signature binds the hash to the firmware and authenticates its origin.
3. **Transmit Firmware and Signature**:
    - The firmware file and its corresponding digital signature are sent to the vehicle.
4. **Verify on the Car**:
    - Upon receiving the update, the vehicle computes the hash of the received firmware.
    - The vehicle uses the manufacturer’s public key to verify the digital signature.
    - If the signature verification succeeds and the calculated hash matches the signed hash, the firmware is deemed authentic and unaltered.

---

3. Adding Confidentiality Protection to the Firmware Update System

To protect firmware confidentiality, encryption should be incorporated in addition to integrity protection:

1. **Encrypt the Firmware**:

  - The manufacturer encrypts the firmware file using a symmetric encryption algorithm such as AES-256.

2. **Secure Key Distribution**:
   - A unique symmetric encryption key is generated for each update.
   - This key is encrypted with the vehicle's public key (from a pre-shared public/private key pair) and sent alongside the encrypted firmware.
3. **Transmit Encrypted Firmware**:
   - The encrypted firmware and the encrypted symmetric key are transmitted to the vehicle.
4. **Decrypt on the Car**:
   - The vehicle uses its private key to decrypt the symmetric key.
   - The symmetric key is then used to decrypt the firmware file.
5. **Verify Integrity**:
   - After decryption, the vehicle performs the integrity verification steps outlined above (hash comparison and signature verification).

Summary

1. A DoS attack can corrupt firmware during transmission, leading to failed installations and system disruptions.
2. Adding **integrity protection** via hashing and digital signatures ensures firmware authenticity and prevents tampering.
3. Adding **confidentiality protection** through encryption and secure key distribution ensures the firmware remains confidential during transmission.

These measures together enhance the safety and security of the wireless firmware update process.

2018F-EXAM 2

Write an abbreviated risk analysis for the firmware update system including both the integrity
and confidentiality protections.
You may hypothesise details of the system and must stipulate yourself its security requirements. Be sure to cover System, Stakeholders, Assets, Vulnerabilities, Threats, and
Risk. You may find Chapter 8 of the course book helpful.
Report your analysis as the answer to this question. Your answer is expected to consume at
most 500 words

Abbreviated Risk Analysis for the Firmware Update System

1. System Overview

The firmware update system allows wireless updates for two critical components in the car:

1. **Driving Controller**: Manages engine, brakes, and other safety-critical driving functionalities.
2. **Infotainment System**: Includes GPS, user interface, and wireless communication capabilities.

Updates are downloaded by the Infotainment System, which forwards them to the Driving Controller. Integrity and confidentiality protections have been implemented using cryptographic techniques.

---

2. Stakeholders

- **Car Manufacturer**: Responsible for firmware updates and system security.
- **Car Owners/Drivers**: Rely on secure and functional systems for safety and convenience.
- **Third-Party Service Providers**: May participate in maintenance or provide cloud hosting for the update server.
- **Attackers**: Individuals or groups seeking to compromise the system for malicious purposes, financial gain, or reputational damage.

---

3. Assets

1. **Firmware Files**: Critical software ensuring the safe and optimal operation of the Driving Controller and Infotainment System.
2. **Encryption Keys**: Ensure confidentiality and integrity during firmware transmission.
3. **Communication Channel**: Wireless link between the car and central server.
4. **System Functionality**: Driving Controller and Infotainment System rely on valid firmware to operate.

---

4. Security Requirements

1. **Integrity**: Ensure firmware is authentic and unaltered.
2. **Confidentiality**: Protect firmware from unauthorized access during transmission.
3. **Authentication**: Verify the legitimacy of updates and their origin.
4. **Availability**: Prevent attacks that disrupt the update process.

---

5. Vulnerabilities

1. **Key Management Issues**: Compromise of private keys or poor key rotation could expose encrypted firmware.

2. **System Isolation**: Inadequate separation between the Infotainment System and the Driving Controller increases the attack surface.
3. **Physical Access**: Attackers with physical access to vehicles could tamper with stored firmware.
4. **Implementation Flaws**: Bugs in cryptographic implementations may weaken protections.

---

6. Threats

1. **Man-in-the-Middle (MITM) Attacks**: Intercept and tamper with updates during transmission.
2. **Unauthorized Access**: Attackers could exploit weak authentication mechanisms to upload malicious firmware.
3. **Replay Attacks**: Resending old but valid firmware updates to cause functional disruptions.
4. **Denial-of-Service (DoS)**: Disrupting the update process to delay critical fixes.

---

7. Risk Analysis

Threat	Impact	Likelihood	Mitigation
MITM Attack	High	Medium	Use TLS, mutual authentication, and PKI.
Malicious Firmware	Critical	Low	Digital signatures with robust verification.
Key Compromise	High	Medium	Key rotation and hardware security modules.
Physical Access Attack	Medium	Medium	Tamper-resistant hardware and encrypted storage.
Replay Attack	Medium	Low	Include nonces or versioning in firmware updates.

---

8. Conclusion

The integrity and confidentiality protections significantly reduce risks associated with tampering, eavesdropping, and unauthorized access. However, vulnerabilities in key management and system isolation remain critical areas to address. By incorporating additional safeguards like tamper-resistant hardware, robust key management, and periodic security audits, the system can meet its security requirements and ensure reliable and secure firmware updates.

2017 EXAM 1

A hospital wants to introduce an automated medication-dispenser device to patients in treatment for pain. Traditionally, nurses dispense medication to patients on the patients requests, observing dosage and frequency limitations. The hospital wishes to leave this responsibility with a machine.
The dispenser is programmed by hospital nurses using an ipad, which connects to the dispenser through a wire, while at the same time retrieving data on dosage etc. from the central hospital database of patient records using the wireless network.
The patient operates the dispenser device by connecting to it via bluetooth; the patient can then request medication using an app on his iOS or Android device. The device will refuse to dispense medication above dosage or frequency as programmed by the nurse.

Design, using appropriate cryptographic primitives, an authentication scheme for the above system. Be sure to note (a) who has access to which keys, (b) what happens if a key is compromised, and (c) how keys a distributed/revoked.
Report your design as the answer to this question. Your answer is expected to consume ~1 page.
Hint: This question can be answered satisfactorily using either simple or advanced machinery from the course.

Authentication Scheme Design for the Medication-Dispenser System

The proposed authentication scheme ensures secure interaction between the stakeholders (nurses, patients, and the dispenser), using cryptographic primitives such as public-key cryptography, symmetric encryption, and digital signatures. The scheme provides authentication, integrity, and non-repudiation, while also addressing key compromise and revocation.

---

1. Key Distribution and Access

- **Hospital Certification Authority (CA)**:
  - Maintains the public-private key pair for issuing digital certificates to all entities.

- **Nurses**:
    - ○ Each nurse is issued a unique public-private key pair (Nurse_Public, Nurse_Private) with a corresponding digital certificate from the hospital CA.
- **Dispensers**:
    - ○ Each dispenser has a unique public-private key pair (Disp_Public, Disp_Private) and a certificate issued by the hospital CA.
- **Patients**:
    - ○ Each patient uses a secure app on their device that generates a symmetric session key (Patient_Key) for communication with the dispenser.
- **Hospital Database**:
    - ○ Uses a unique key pair (DB_Public, DB_Private) for secure communication with dispensers and nurses.

---

2. Authentication Process

Step 1: Nurse-Dispenser Programming

1. **Establish Secure Channel**:
    - ○ The nurse connects their iPad to the dispenser via a wired connection.
    - ○ Mutual authentication is performed using the nurse's certificate and the dispenser's certificate, validated by the hospital CA.
2. **Data Retrieval**:
    - ○ The iPad retrieves dosage and frequency information from the hospital database using TLS (Transport Layer Security), authenticated by the database's digital certificate.
3. **Programming**:
    - ○ The nurse signs the programming instructions with their private key (Nurse_Private). The dispenser verifies the signature using the nurse's public key (Nurse_Public).

Step 2: Patient-Dispenser Interaction

1. **Bluetooth Pairing**:
    - ○ The dispenser generates a random challenge and sends it to the patient’s app via Bluetooth.
    - ○ The app computes a hash of the challenge using the patient’s symmetric key (Patient_Key) and sends it back.
    - ○ The dispenser verifies the hash to authenticate the patient.
2. **Request Medication**:
    - ○ The app sends an encrypted medication request using the shared Patient_Key.
    - ○ The dispenser decrypts the request and verifies dosage and frequency compliance.
3. **Dispense Medication**:

  - If the request is valid, the dispenser logs the transaction and dispenses medication.

---

3. Key Compromise and Revocation

Key Compromise Mitigation:

1. **Nurse/Dispenser Private Key**:
   - If a private key is compromised, the hospital CA revokes the associated certificate and issues a new key pair.
   - Communication logs can identify misuse, as all actions are digitally signed.
2. **Patient Key**:
   - If Patient_Key is compromised, the app generates a new symmetric key during the next interaction.

Key Distribution and Revocation:

- **Certificate Revocation List (CRL)**:
  - Maintained by the hospital CA and distributed regularly to all dispensers and devices.
- **Periodic Key Rotation**:
  - Keys are rotated every 3 months to minimize exposure.

---

4. Security Benefits

- **Authentication**: Nurses, dispensers, and patients are authenticated using certificates and symmetric keys.
- **Integrity and Non-repudiation**: All programming instructions and actions are digitally signed and verifiable.
- **Confidentiality**: Medication requests and programming data are encrypted during transmission.
- **Key Management**: Revocation and rotation mechanisms ensure security even after key compromise.

This scheme ensures robust security while maintaining usability and accountability in the medication-dispenser system.

2017 EXAM 2

Write an abbreviated risk analysis for the medication dispenser system including your authentication scheme.

You may hypothesise details of the system and must stipulate yourself its security

requirements. Be sure to cover System, Stakeholders, Assets, Vulnerabilities, Threats, and

Risk. You may find Chapter 8 of the course book helpful.

Report your analysis as the answer to this question. Your answer is expected to consume ~1

page.

Abbreviated Risk Analysis for the Medication Dispenser System

1. System Overview

The automated medication dispenser system enables patients to request medication via Bluetooth and nurses to program the dispenser using an iPad connected via a wired connection. The system retrieves dosage and frequency data from a hospital database over a wireless network. The authentication scheme ensures secure communication between stakeholders using cryptographic mechanisms, including digital signatures, symmetric encryption, and certificates issued by a hospital Certification Authority (CA).

---

2. Stakeholders

1. **Patients**: Operate the dispenser to request medication.
2. **Nurses**: Program dosage and frequency limits on dispensers.
3. **Hospital IT Staff**: Maintain the central database, CA, and key management infrastructure.
4. **Attackers**: Potential adversaries seeking unauthorized access or disruption.

---

3. Assets

1. **Dispenser Firmware**: Controls medication dispensing.
2. **Patient Data**: Dosage, frequency, and treatment details stored in the hospital database.
3. **Authentication Keys**: Public-private key pairs for dispensers, nurses, and the hospital database, along with symmetric session keys for patient-device communication.
4. **Medication Logs**: Records of all dispensing activities for compliance and accountability.

---

4. Security Requirements

1. **Authentication**: Verify the identities of patients, nurses, and dispensers.
2. **Confidentiality**: Protect sensitive patient data and medication requests.

3. **Integrity**: Ensure programming instructions and requests are unaltered.
4. **Non-repudiation**: Enable traceability of actions to prevent denial of responsibility.
5. **Availability**: Ensure the system is operational and resistant to disruption.

---

5. Vulnerabilities

1. **Key Compromise**: Exposure of private keys or symmetric keys can compromise communication.
2. **Physical Access**: Attackers with access to the dispenser could tamper with hardware or firmware.
3. **Bluetooth Communication**: Susceptible to eavesdropping or replay attacks if not properly secured.
4. **Database Connection**: Exploitation of wireless communication could lead to data leakage.

---

6. Threats

1. **Unauthorized Access**: Adversaries impersonate patients or nurses to manipulate the dispenser.
2. **Data Breaches**: Interception of patient data during communication with the database.
3. **Replay Attacks**: Attackers resend previously valid requests to dispense medication illicitly.
4. **Denial-of-Service (DoS)**: Disrupting communication channels to prevent system functionality.

---

7. Risk Analysis

Threat	Impact	Likelihood	Mitigation
Key Compromise	High	Medium	Key rotation, hardware security modules, CRL checks
Bluetooth Exploitation	Medium	Medium	Encrypted session keys, mutual authentication

Unauthorized Access	Critical	Low	Digital signatures, robust authentication scheme
Replay Attacks	Medium	Low	Nonces in requests, session validation
Database Exploitation	High	Medium	TLS, database firewall, access logging
Physical Tampering	Medium	Medium	Tamper-resistant hardware, regular audits

---

8. Conclusion

The authentication scheme significantly mitigates risks by ensuring secure programming and operation of the dispenser. Key management, encryption, and robust authentication prevent unauthorized access and data breaches. However, vulnerabilities like key compromise and physical tampering require additional safeguards, such as tamper-resistant hardware and stringent key management policies. With proper implementation, the system can securely and reliably manage medication dispensing, ensuring patient safety and compliance.