Information Security Assignment 3

Aryan Nath

12th March 2025

1 System Overview:

The unified payment interface (UPI) is a real-time payment system developed by the National Payments Corporation of India (NPCI) to enable money transfer directly between the bank accounts of the payer and the payee.
The process for payment transcations in UPI is as follows:

1. **Initiation:**
   Payer opens UPI app such as GPay, Paytm, etc. and selects recipient VPA/scans QR code, enters amount, adds optional remarks.

2. **Payload to PSP:**
   Payer authenticates via UPI PIN/biometrics. Transaction details (payer/payee VPAs, amount) sent to PSP from the payer's device, which verifies VPA syntax and transaction limits.

3. **PSP to NPCI:**
   PSP forwards validated request to NPCI, which routes via IMPSv3 protocol:

   (a) Global identifiers (internal resolution):

      i. IFSC+Account: Direct bank account linking

      ii. Aadhar@NPCI: Resolution via Aadhar-linked banking details

      iii. Mobile@NPCI: Identification through registered mobile numbers

   (b) Local identifiers (Userid@PSP): NPCI forwards to recipient's PSP, which resolves to actual bank account.

4. **Payee Validation:**
   Recipient's PSP verifies account existence and KYC compliance.

5. **Debit:**
   NPCI sends debit request to payer's bank, which verifies authentication, sufficient funds, and transaction limits. If sufficient, bank updates balance and sends signed confirmation with transaction ID to NPCI.

6. **Credit:**
   NPCI generates credit request to payee's bank, which credits amount in real-time and confirms to NPCI.

7. **PSP Update:**
   NPCI sends pay response to both PSPs. Payer's PSP notifies payer's device of completion.

2 Threat Actors and Adversaries

Threat Actors (Everyone involved in the UPI protocol):

1. Payer (user).
2. Payee (user).
3. Payer's PSP.
4. Payee's PSP.
5. Payer's bank.
6. Payee's bank.
7. Third-party UPI-enabled app developers.
8. Communication providers (for forwarding transaction messages and SMS for payment confirmation).
9. Mobile device manufacturers.

Adversaries (Entities who may try to actively exploit vulnerabilities in the UPI protocol for their own gain):

1. Payee denying receiving payment.
2. Developers of malicious third-party UPI-enabled app developers.
3. An adversary who compromises the device of the payer/payee for fraudulent activities.
4. An adversary who compromises the PSP server.
5. Malicious NPCI insider.
6. An adversary with access to all network communication lines used for UPI responses.
7. Social engineering attacker.
8. An adversary who can compromise the payer's/payee's bank's server.
9. Competitors of PSPs/Banks.

3 Capabilities of the Adversaries:

1. **Payee denying receiving payment:**

   (a) Resources: Knowledge of the dispute resolution process.

   (b) Techniques: Exploiting lack of immediate proof of delivery and exploiting trust in the system.

   (c) Attack vectors: the payee falsely claims to not have received the payment while hoping that payer won't be able to disprove the claim.

2. **Developers of malicious third-party UPI-enabled apps:**

   (a) Resources: Malicious apps that request unnecessary permissions or exploit OS vulnerabilities.

   (b) Techniques: Development and distribution of these apps.

   (c) Attack vectors: unauthorized registration using cell number, unauthorized transactions using cell number and partial debit card number, unauthorized transactions without debit card number (learning authentication factors). Screen overlay attacks, denial of service, SMS interception, and unauthorized data access.

3. **An adversary who compromises the device of the payer/payee for fraudulent activities:**

   (a) Resources: Compromised device of the payer/payee with full access to all installed apps.

   (b) Techniques: Malware installation and social engineering to steal passwords.

   (c) Attack vectors: altering transaction amounts and recipients, stealing UPI PIN and other sensitive information from the device, bypassing UPI authentication from the UPI app of the compromised device after stealing the UPI PIN.

4. **An adversary who compromises the PSP server:**

   (a) Resources: unauthorised accessed to the PSP server, knowledge of server side vulnerabilities and attack techniques, and tools for unauthorized data access.

   (b) Techniques: SQL injection and remote code execution.

   (c) Attack vectors: VPA mapping manipulation to divert funds to attacker-controlled accounts; steal sensitive data such as VPA, transaction history, and user credentials; denial of service attack to disrupt UPI services.

5. **Malicious NPCI insider:**

(a) Resources: acceess to core UPI infrastructure and data, knowledge of the UPI security mechanisms, and ability to bypass security controls of the UPI systems.

(b) Techniques: abuse of privilege, data manipulation for transaction records and user data, and system disruption by disabling security controls or launching DoS attack.

(c) Attack vectors: redirect funds to attack controlled accounts, steal sensitive data such as transaction history and user credentials, and disrupt the entire UPI infrastructure.

6. **An adversary with access to all network communication lines used for UPI responses:**

   (a) Resources: ability to intercept network traffic, tools to sniff and manipulate packets, and knowledge of networking protocols and security mechanisms.

   (b) Techniques: packet sniffing to steal sensitive information, traffic interception to read and modify transaction data, DNS spoofing to redirect users to fake UPI server to steal their credentials.

   (c) Attack vectors: altering transaction amount and recipient, replay previously captured transaction data to execute unauthorized transactions, and cause DoS attack to disrupt UPI service.

7. **Social engineering attacker:**

   (a) Resources: social skills, knowledge of UPI and banking procedures, tools for creating fake websites and emails, access to leaked phone numbers or bank account details.

   (b) Techniques: Phishing, vishing, QR code scams for phishing or initiating fraudulent transactions, and masquerading as a legitimate service and earning the user's trust.

   (c) Attack vectors: stealing UPI PINs and passwords, tricks users into making unauthorized transactions, and persuading users to install malware.

8. **An adversary who can compromise the payer's/payee's bank's server:**

   (a) Resources: unauthorized access to a bank's internal systems, knowledge of the bank's security mechanisms, tools for extracting and manipulating UPI system data.

   (b) Techniques: SQL injection, privileged server access, and data manipulation.

   (c) Attack vectors: unauthorized transactions, denial of service, and stealing sensitive data.

9. **Competitors of PSPs/Banks:**
    (a) Resources: capital to hire one of the above adversaries.
    (b) Techniques: same as one of the above adversaries.
    (c) Attack vectors: disrupt services and reduce public trust.

4 Trust vs Verifiability:

The UPI protocol depends on trust on the following entities to run successfully:

1. Trust that the PSP and bank servers correctly maintain transaction logs.
2. Trust in the payer/payee's device for device to user binding without malicious exploitation. This can be exploited through malicious apps downloaded on the user's device.
3. Trust in the PSP servers. This trust can be exploited as a compromised PSP server can be used to manipulate VPA mappings, steal sensitive data, or disrupt UPI services.
4. Trust in NPCI to act as the trusted central authority in the protocol. Can be exploited by a malicious insider to compromise the entire UPI system.
5. Trust in the Payer and Payee's banks to correctly debit and credit accounts, verify user authentication, and adhere to security protocols. Can be exploited by a compromised bank server to authorize fraudulent transactions, steal sensitive data, or manipulate account balances.
6. Trust in networking channels to securely transmit SMSs and UPI responses. Can be exploited by an attacker to intercept, spoof, and modify messages.

Mechanisms that can be used to enhance verifiability in transactions are as follows:

1. Digital signatures of transaction logs maintained at both PSP and bank servers verify that payees cannot deny receiving payments and ensure log integrity, as coordinated attacks on both servers are highly improbable.
2. Hardware-backed attestation mechanisms, such as Android Keystore attestation, verify device and UPI app integrity before authorizing transactions.
3. Strengthen security of PSP servers to verify the VPA-to-bank account resolution process, with similar verification needs applying to payer's/payee's bank servers. The same verification follows with the payer's/payee's bank servers.

4. MPC techniques for secret recovery verify against malicious NPCI insider threats by preventing single NPCI member transaction modifications.

5. Secure symmetric key exchange between all parties using Needham-Schroeder or Diffie-Hellman protocols with user IDs in signatures verifies against MITM attacks, while message authentication codes verify against bogus messages.

5 Threat Analysis:

I use the STRIDE framework to identify and categorize potential threats to the UPI system by covering:

- Spoofing: Pretending to be something or someone else.
- Tampering: Modifying data on disk, network, or memory.
- Repudiation: Claiming you didn't do something or weren't responsible. The key question: what evidence exists?
- Information disclosure: Providing information to unauthorized parties.
- Denial of Service: Consuming resources needed for service.
- Elevation of Privilege: Allowing unauthorized actions.

The descriptions of each threat category for the UPI protocol is as follows:

1. **Spoofing:**
   (a) **Potential Threats:**
      i. SMS/response spoofing: Adversary spoofs SMS from bank/UPI app to extract sensitive information.
      ii. VPA spoofing (via social engineering): Adversary creates fake VPA resembling legitimate one to misdirect payments.
      iii. Device spoofing (via malicious app/compromised device): Malicious app spoofs device fingerprint for unauthorized UPI access.
      iv. Server spoofing: Adversary spoofs PSP/bank server to intercept transaction data.

   (b) **Adversaries:**
      i. Malicious Third-Party App Developer.
      ii. Network Attacker.
      iii. Social Engineering Attacker.
      iv. Compromised Device Owner.

   (c) **Affected Components:**
      i. Payer/Payee Device.

ii. PSP Servers.

iii. NPCI.

iv. Bank Servers.

2. **Tampering:**

(a) **Potential Threats:**

i. Transaction Tampering (Malicious App/Network Attacker): Adversary modifies transaction details before authorization.

ii. VPA Mapping Manipulation (Compromised PSP Server/NPCI Insider): Adversary alters VPA-bank account mappings to divert funds.

iii. Log Tampering (Compromised PSP Server/Bank Server): Attacker modifies transaction logs to hide fraud.

iv. Code Tampering (Malicious App Developer): Malicious app modifies UPI app code to bypass security or steal data.

(b) **Adversaries:**

i. Malicious Third-Party App Developer.

ii. Network Attacker.

iii. Compromised PSP Server Attacker.

iv. Malicious NPCI Insider.

v. Compromised Bank Server Attacker.

(c) **Affected Components:**

i. Payer/Payee Device.

ii. PSP Servers.

iii. NPCI Database.

iv. Bank Servers.

v. Network Communication Channels

3. **Repudiation:**

(a) **Potential Threats:**

i. Payer Denies Transaction (Fraudulent Payer): Payer claims they didn't authorize a transaction to reverse payment.

ii. Payee Denies Receiving Payment (Fraudulent Payee): Payee claims non-receipt despite successful credit.

iii. PSP Denies Processing Transaction (Malicious PSP Insider): PSP denies processing to avoid fraud responsibility.

(b) **Adversaries:**

i. Fraudulent Payer.

ii. Fraudulent Payee.

iii. Malicious PSP Insider.

iv. Malicious Bank Insider.

(c) **Affected Components:**

i. Payer.
ii. Payee.
iii. PSP Servers.
iv. Bank Servers.

4. **Information Disclosure:**

(a) **Potential Threats:**

i. Data Breaches (Compromised PSP Server/NPCI/Bank): Adversary accesses databases containing sensitive user information.
ii. Network Sniffing (Network Attacker): Adversary intercepts network traffic to steal sensitive data.
iii. Malicious Apps (Malicious App Developer): Malicious app extracts sensitive data from device.
iv. Social Engineering (Social Engineering Attacker): Attacker tricks users into revealing sensitive information.

(b) **Adversaries:**

i. Malicious Third-Party App Developer.
ii. Network Attacker.
iii. Compromised PSP Server Attacker.
iv. Malicious NPCI Insider.
v. Compromised Bank Server Attacker.
vi. Social Engineering Attacker.

(c) **Affected Components:**

i. Payer/Payee Device.
ii. PSP Servers.
iii. NPCI Database.
iv. Bank Servers.
v. Network Communication Channels.

5. **Denial of Service:**

(a) **Potential Threats:**

i. DDoS Attacks (Network Attacker/Competitors): Adversary floods UPI infrastructure with traffic, overwhelming servers.
ii. Resource Exhaustion (Compromised PSP Server/NPCI): Adversary exploits vulnerabilities to exhaust system resources.

(b) **Adversaries:**

i. Network Attacker.

   ii. Competitors of PSPs/Banks.
   iii. Malicious NPCI Insider.
   iv. Compromised PSP Server Attacker.

   (c) **Affected Components:**

   i. PSP Servers.
   ii. NPCI.
   iii. Bank Servers.
   iv. Network Infrastructure.

6. **Elevation of Privilege:**

   (a) **Potential Threats:**

   i. Privilege Escalation (Compromised PSP Server/Bank Server): Adversary exploits vulnerabilities for administrative access.
   ii. Unauthorized Access to Sensitive Data (NPCI Insider): Malicious insider accesses sensitive NPCI data.

   (b) **Adversaries:**

   i. Compromised PSP Server Attacker.
   ii. Malicious NPCI Insider.
   iii. Compromised Bank Server Attacker.

   (c) **Affected Components:**

   i. PSP Servers.
   ii. NPCI.
   iii. Bank Servers.

6 Mitigations and Countermeasures:

1. **Spoofing Mitigations:**

   (a) Strong Device Binding:

   i. Combine device ID, IMEI number, and unique identifiers for device fingerprinting. Store securely on device and server. Utilize hardware-backed security features. Limit binding attempts.
   ii. Feasibility:
   Technically feasible with current technology. Requires coordination between UPI apps, device manufacturers, and OS vendors.
   iii. Trade-offs:
   May create friction for users with multiple or frequently changed devices. Requires addressing privacy concerns.

   (b) Digital Signatures for SMS Messages:

   i. Use digital signatures to verify SMS authenticity. Implement verification mechanism using trusted public key.

ii. Feasibility:
Technically feasible, but requires significant SMS infrastructure changes.

iii. Trade-offs:
May increase SMS cost and complexity. Requires user education on signature verification.

(c) Mutual Authentication:

i. Implement mutual authentication between payer/payee device, PSPs, and NPCI. Use TLS/SSL certificates. Require certificate verification before connection.

ii. Feasibility:
Technically feasible and widely adopted in secure communication protocols.

iii. Trade-offs:
May increase connection overhead. Requires careful certificate management.

2. **Tampering Mitigations:**

(a) End-to-End Encryption:

i. Encrypt transaction data using shared symmetric keys. Employ strong algorithms (AES-256) and secure key exchange (Diffie-Hellman).

ii. Feasibility:
Technically feasible, but requires careful key management.

iii. Trade-offs:
May increase computational overhead.

(b) Digital Signatures

i. Use digital signatures for transaction data integrity. Payers sign transaction data and their identity with private key, payees verify with public key.

ii. Feasibility:
Technically feasible, but requires careful key management.

iii. Trade-offs:
May increase transaction complexity. Requires standardization across UPI participants.

3. **Repudiation Mitigations:**

(a) Detailed Transaction Logs:

i. Maintain comprehensive transaction logs (timestamp, VPA, amount, transaction ID, digital signature). Ensure tamper-proof, auditable logs.

ii. Feasibility:
Technically feasible and essential for dispute resolution.

iii. Trade-offs:
Requires significant storage and careful log management.

4. **Information Disclosure Mitigations:**

   (a) Least Privilege:

   i. Grant minimum access necessary for job duties. Regularly review access control policies.

   ii. Feasibility:
   Technically feasible and a fundamental security principle.

   iii. Trade-offs:
   Requires careful planning and implementation of access controls.

   (b) Data Loss Prevention (DLP):

   i. Implement DLP to prevent sensitive data exfiltration. Use classification and monitoring tools to prevent unauthorized transfers.

   ii. Feasibility:
   Technically feasible, but requires careful configuration and monitoring.

   iii. Trade-offs:
   May impact productivity. Requires addressing privacy concerns.

   (c) Use MPC techniques:

   i. Use MPC to require multi-user approval for sensitive data access, preventing single-user exploitation.

   ii. Feasibility:
   Technically feasible, requires careful MPC protocol selection and secure key management. Requires organizational changes for multi-user approval processes.

   iii. Trade-offs:
   May increase complexity and overhead. Requires coordination between users for timely approvals.

5. **Denial of Service Mitigations:**

   (a) DDoS Mitigation Techniques:

   i. Implement traffic filtering, rate limiting, and CDNs. Use combined hardware/software solutions to protect infrastructure.

   ii. Feasibility:
   Technically feasible, requires ongoing monitoring and adaptation.

   iii. Trade-offs:
   Potentially expensive. May impact legitimate traffic performance.

   (b) Intrusion Detection Systems (IDS):

   i. Deploy IDS for DoS attack detection and response. Use signature-based and anomaly-based detection.

    ii. Feasibility:
    Technically feasible and widely adopted in secure systems.

    iii. Trade-offs:
    Requires careful configuration and monitoring.

  (c) Incident Response Plan:

    i. Develop plan for rapid DoS attack recovery. Include procedures for identification, containment, and eradication.

    ii. Feasibility:
    Requires careful planning and team coordination.

    iii. Trade-offs:
    Requires ongoing training and plan testing.

6. **Elevation of Privilege Mitigations:**

  (a) Least Privilege:

    i. Grant minimum access necessary for job duties. Regularly review access control policies.

    ii. Feasibility:
    Technically feasible and a fundamental security principle.

    iii. Trade-offs:
    Requires careful planning and implementation of access controls.

  (b) Intrusion Detection Systems (IDS):

    i. Implement IDS to detect unauthorized access attempts. Use signature-based and anomaly-based detection.

    ii. Feasibility:
    Technically feasible and widely adopted in secure systems.

    iii. Trade-offs:
    Requires careful configuration and monitoring.