Insurance & Banking: BlingBank

Network and Computer Security Project

Niklas Herbster, Vuk Jurisic, Kevin Ross

Table of contents

01
Secure Document Format

Ensuring Confidentiality, Integrity and Freshness

Design Principles for Secure Document Format

Confidentiality

- AES for document encryption
- RSA for secret key encryption

Integrity

- Creation of a digital signature

Freshness

- Timestamp usage

Cryptographic Libraries and Tools

- Java 21
- Custom Cryptographic Library

- `java.security.Signature:` For generating and verifying digital signatures for document integrity.
- `java.security.SecureRandom:` For generating IVs to ensure integrity.
- `javax.crypto.Cipher:` For encryption and decryption operations.
- `javax.crypto.KeyGenerator:` For generating symmetric keys(AES).
- `KeyFactory, PublicKey, PrivateKey:` For handling public and private keys in RSA encryption.

Sender

Pub Key Encryption

Generate iv and encrypt public key using symmetric key and the iv.

Encrypt Document

Generate a new iv and encrypt the document using session key and iv.

Create Signature

Encrypting object generate in step 2 using RSA with our private key.

Adding the timestamp

Timestamp is added to the final payload which gets sent to the server.

Receiver

Pub Key Decryption

Using the iv and the symmetric key, public key is retrieved.

Signature Verification

Signature is verified using RSA and timestamp is checked.

Document Decryption

Using the 2nd iv and the session key, document is decrypted

02

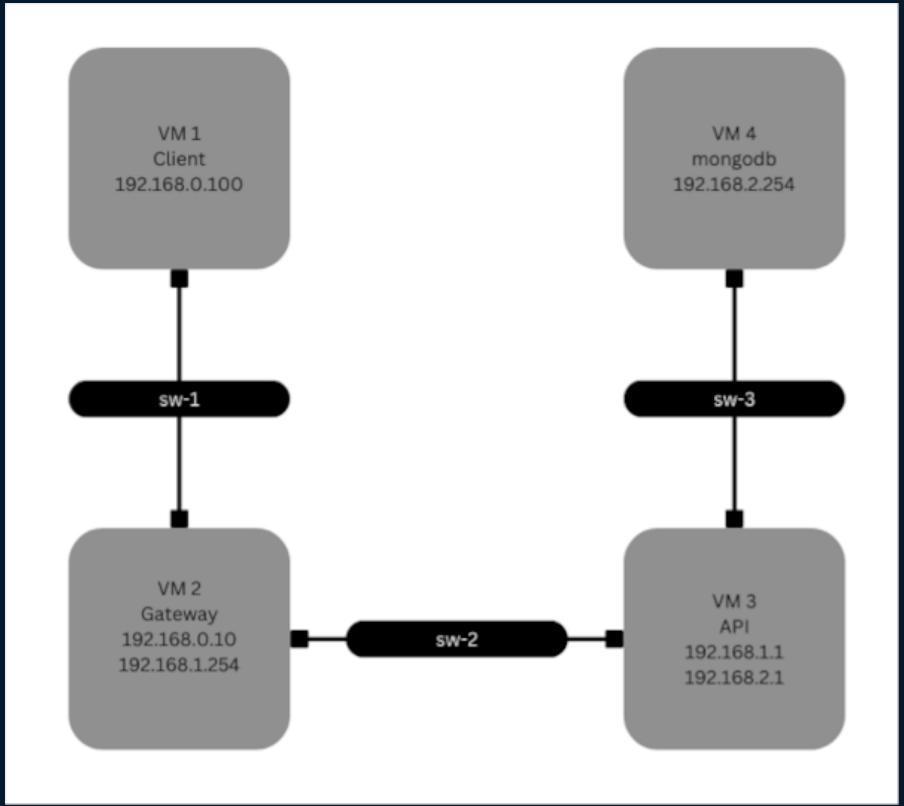
Infrastructure

Implementation of our infrastructure

02 Infrastructure

Client

Database



Gateway

Server

03

Secure Channel and Key Distribution

Implementation and details of our cryptographic library

03 Secure Channel and Key Distribution

Secure Channel

- Client and Server communicate via HTTPS
- Server and Database communicate via SSL/TLS

Key Distribution

- Public Keys are encrypted and exchanged when a User creates an account
- Session Keys are encrypted and exchanged when a User logs in (PFS)

04

Security Challenge

Enhancing Security: Non-Repudiation, Freshness, and Multi-Owner Accounts

Non-repudiation

It is ensured through the usage of the digital signature.

Freshness

Achieved through the usage of the timestamp.

Multi-Owner Accounts

Transactions are processed once any of the owners initiates the transaction.

Results

- Distributed secure Infrastructure
- Confidentiality, Integrity and Authenticity are ensured
- Perfect Forward Secrecy

Conclusion

- Fundamentals of secure communication between all parties
- Possible improvements using nonces
- Implement Https between client and server

Demo

https://drive.google.com/file/d/1MHrQyi1eSxr_tzU_vSVFUcGWePZF4Quh/view?usp=sharing