Universidade de Brasília – UnB

Departamento de Ciência da Computação - CIC

Disciplina: Tópicos Avançados em Segurança Computacional – 2025/1

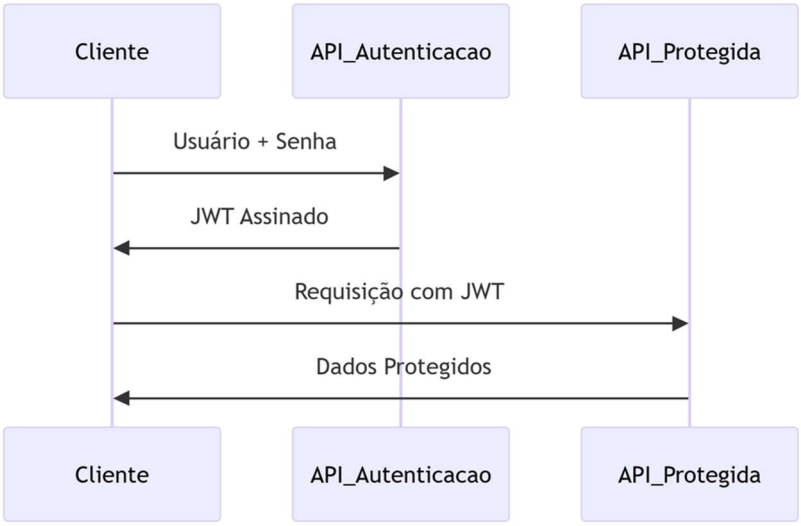
Professora: Lorena Borges

Trabalho Prático 01

REST API com Autenticação Segura e Criptografia

1. Objetivo

Desenvolver uma aplicação cliente-servidor que realize comunicação segura entre aplicações através de autenticação segura, via padrão REST e tokens JWT assinados digitalmente, com o uso dos algoritmos criptográficos e de autenticação, HMAC e RSA (PKCS#1 v1.5 e PSS).



2. Implementação

API Autenticação

- As credenciais dos usuários deverão ser armazenadas no servidor de maneira protegida, persistindo apenas o hash da senha;
- O envio das credenciais para o servidor pelo módulo cliente deverá ser feito de maneira segura, ou seja, mantendo o sigilo das informações em caso de interceptação;
- Deverá ser gerado um token JWT (JSON Web Token) para futuras autenticações seguras, assinado através do algoritmo RSA, com data ou tempo de expiração e retornado para o cliente.

API Protegida com Verificação de Assinatura

- O cliente fará uma requisição a dados secretos utilizando o Token JWT enviado pelo servidor;
- O servidor fará a verificação da assinatura do Token JWT para validação e permissão de acesso aos dados protegidos;
- Caso a validação seja realizada com sucesso e o token não tenha expirado, os dados devem ser retornados ao cliente.

Cenário 1:

Autenticar o Token JWT com o algoritmo HMAC.

Cenário 2:

Assinar o Token JWT com o algoritmo RSA (RSA-PSS ou PKCS#1 v1.5);

3. Análises

- Adaptar a dinâmica de acesso da API_Protegida para cada algoritmo de segurança usado nos cenários anteriores (**Cenários 1 e 2**), realizando análises sobre os mecanismos de segurança, aplicabilidade e vantagens de cada algoritmo;
- Testar assinaturas de JWT inválidas, alteradas ou expiradas, demonstrando a falha no acesso aos dados protegidos através da rejeição pelo servidor;
- Verificar possíveis vulnerabilidades e ataques conhecidos aos **cenários 1 e 2**, destacando métodos e procedimentos que poderiam tratar as respectivas vulnerabilidades;
- Utilizar o Wireshark para analisar os pacotes trocados através dos métodos padrão REST, assim como as informações transferidas entre cliente e servidor para embasar as análises de segurança.

Observações:

- É permitida a utilização de bibliotecas públicas para primitivas criptográficas de cifração e decifração simétrica, assimétrica, hash e geração de chaves.
- Não será permitida a utilização de frameworks REST API, utilizar bibliotecas padrão REST, json, HTTP/HTTPS etc.
- A avaliação será mediante a verificação das funcionalidades e inspeção do código.
- Implementação individual ou em dupla.
- Linguagens C++, Java e Python.

O que deve ser entregue: o código fonte e seu executável, relatório descritivo (6 pg max) contendo contextualização teórica, implementação, comparativos, análises e considerações.