Relatório Técnico - Chat Seguro

Integrantes:

- **Daniel Merib Lisboa [RA: 1125729]**
- **Luis Henrique Freitas Giotto [RA: (adicionar)]**

Disciplina: Criptografia e Segurança Professor: Rafael Basso Reis

---

Tecnologias e Algoritmos Utilizados

- **Linguagem de Programação: Python**
- **Bibliotecas:**
  - `socket` **- para comunicação via rede.**
  - `cryptography` **- para criptografia assimétrica e simétrica.**
  - `hashlib` **- para geração e verificação de hashes.**
- **Criptografia Simétrica: AES (Advanced Encryption Standard)**
- **Criptografia Assimétrica: RSA (Rivest-Shamir-Adleman)**
- **Função Hash: SHA-256**
- **Assinatura Digital: RSA com SHA-256**

Descrição do Funcionamento do Sistema

O sistema desenvolvido é um chat seguro, utilizando a troca de mensagens criptografadas via sockets TCP/IP. Os principais passos de funcionamento são:

1. **Inicialização:**
   - **Cada usuário gera seu par de chaves RSA (pública e privada).**

2. **Troca de chaves:**
    - **As chaves públicas são trocadas no momento da conexão inicial entre os pares.**
3. **Autenticação:**
    - **As mensagens iniciais de apresentação contêm a assinatura digital da chave pública para garantir a autenticidade dos participantes.**
4. **Envio de mensagens:**
    - **Antes de enviar, a mensagem é criptografada usando AES, cuja chave simétrica é trocada previamente criptografada com RSA.**
    - **É gerado um hash SHA-256 da mensagem e enviado junto.**
5. **Recebimento de mensagens:**
    - **Ao receber, a mensagem é descriptografada.**
    - **A integridade é verificada comparando o hash recebido com o hash calculado.**

Justificativa das Escolhas

- **Uso de RSA:**
    - **RSA foi escolhido pela facilidade de implementação e pela segurança na troca de chaves simétricas.**
- **Uso de AES:**
    - **AES foi escolhido para a criptografia das mensagens pela sua eficiência e forte padrão de segurança, ideal para dados em trânsito.**
- **Uso de SHA-256:**
    - **SHA-256 é um padrão amplamente utilizado para garantir integridade, sendo robusto contra colisões.**
- **Sockets TCP:**
    - **Utilizamos sockets TCP pela confiabilidade no transporte de dados, essencial para garantir que as mensagens não se percam ou cheguem**

corrompidas.

- **Troca de Chaves e Assinatura Digital:**
    - **Para garantir autenticidade, implementamos a troca de chaves públicas assinadas, prevenindo ataques de man-in-the-middle.**

Divisão de Tarefas

- **Daniel:**
    - **Implementação da criptografia simétrica e assimétrica.**
    - **Geração e validação de hashes.**
    - **Lógica de encriptação/desencriptação.**
- **Luis:**
    - **Desenvolvimento da comunicação com sockets.**
    - **Implementação da assinatura digital.**
    - **Testes de transmissão segura e validação de integridade.**

Testes Realizados

- **Teste de Envio e Recebimento de Mensagens:**
    - **Foram enviados vários tipos de mensagens (curtas e longas) para verificar a criptografia/descriptografia.**
- **Teste de Integridade:**
    - **Alteramos propositalmente uma mensagem em trânsito e confirmamos que o sistema detectou a alteração.**
- **Teste de Autenticação:**
    - **Simulamos trocas de chaves sem assinatura digital e o sistema corretamente rejeitou conexões não confiáveis.**

Conclusões

O chat seguro desenvolvido atendeu aos requisitos de confidencialidade, integridade e autenticação. A escolha das tecnologias e algoritmos se mostrou adequada, com resultados satisfatórios nos testes realizados. Como sugestão para trabalhos futuros, poderíamos aprimorar a interface gráfica e incluir suporte para grupos de conversa.