

Projeto Final: Sistema de Mensagens Seguras com Registo de Utilizadores e Troca de Chaves

1. Objetivo Geral

Desenvolver um sistema simplificado de mensagens seguras entre clientes, suportado por um servidor central responsável pelo registo de utilizadores, gestão de chaves públicas, e distribuição dessas chaves mediante solicitação.

O sistema deve permitir que dois clientes comuniquem de forma confidencial e autêntica, utilizando primitivas criptográficas modernas.

2. Descrição do Sistema

O sistema deverá ser composto por dois módulos principais:

A. Servidor de Registo e Distribuição de Chaves

O servidor funciona como uma **Autoridade de Registo (AR)**, mantendo uma base de dados com:

- ID de cada cliente (ex.: nome, NIF, número de estudante, ou outro identificador);
- Chave pública correspondente;
- Data/hora de registo.

O servidor deve permitir:

1. Registo de um novo cliente, recebendo:

- ID do cliente
- Chave pública do cliente

2. **Consulta da chave pública de um cliente específico**, mediante solicitação.
3. (Opcional) Registo seguro, protegendo a comunicação entre cliente e servidor.

B. Cliente de Mensagens Seguras

Cada cliente deve implementar:

1. **Geração local do par de chaves (pública/privada)** usando uma biblioteca à escolha.
2. **Envio do ID e chave pública ao servidor para registo.**
3. **Pedido ao servidor da chave pública de outro cliente.**
4. **Início de uma sessão de chat seguro:**
 - O cliente A solicita a chave pública de B.
 - A deriva uma **chave de sessão** (simétrica), que será cifrada com a chave pública de B.
 - A e B passam a comunicar cifrando todas as mensagens com essa chave simétrica.
5. **Verificação de integridade e autenticidade das mensagens**, por exemplo:
 - MAC/HMAC;
 - Assinatura digital de cada mensagem.

3. Requisitos Criptográficos

Os grupos podem implementar os mecanismos criptográficos:

- Usando **bibliotecas padrão** (OpenSSL, libsodium, BouncyCastle, Python cryptography, Java JCE, etc.);
- **OU** usando ferramentas externas como **OpenSSL CLI** para geração, armazenamento e manipulação de chaves e certificados, integrando-as no fluxo do sistema.

Mínimos necessários:

- Cifra simétrica segura (AES-GCM, AES-CBC + HMAC, ChaCha20-Poly1305, etc.);
- Cifra assimétrica para troca de chaves (RSA, ECC, ou DH/ECDH);
- Hash criptográfico (SHA-256 ou superior);
- Opcional: assinatura digital.

4. Requisitos Funcionais

O sistema deve permitir:

- Registo de utilizadores no servidor.
- Pedidos de chave pública por ID.
- Troca de mensagens cifradas ponto-a-ponto.
- Implementação de pelo menos **uma** das seguintes funcionalidades avançadas:
 1. Assinatura digital das mensagens.
 2. Mecanismo de autenticação inicial entre cliente e servidor.
 3. Rotação periódica de chaves de sessão.
 4. Lista de utilizadores registados.

5. Entregáveis

1. **Código-fonte completo** do sistema (cliente + servidor).
2. **Relatório técnico** (10–15 páginas) contendo:
 - Arquitetura do sistema;
 - Esquema criptográfico adotado;
 - Diagrama do fluxo de mensagens;
 - Justificativa das escolhas criptográficas;
 - Instruções para executar o sistema;
 - Uma análise sucinta das limitações e possíveis melhorias.
3. **Demonstração prática** (10–15 minutos) em sala online.

6. Critérios de Avaliação

- Correção e segurança da implementação (40%)
- Clareza do relatório e da documentação (25%)
- Qualidade da arquitetura e design do sistema (20%)
- Demonstração funcional e explicação técnica (15%)

7. Observações Finais

- O sistema não precisa ter interface gráfica; linha de comando é suficiente.
- Não é necessário implementar protocolos complexos (TLS, STS, Signal, etc.).
- O foco está em demonstrar compreensão prática das primitivas criptográficas estudadas na disciplina.

8. Outras Informações

- Data da **apresentação** online: **29 Jan 2026**
- Data da **entrega** de todo o material: **28 Jan 2026 via moodle**
- Trabalho em grupo de 2 ou 3 alunos.