Vinícius Miranda Andrade Piovesan RA: 23025544

Felipe Ribeiro Almeida RA: 23024683

Sérgio Ricardo Pedote Junior RA:23747441

MATHEUS DE MEDEIROS TAKAKI RA: 23025143

Escolha da Criptografia: bcrypt

A biblioteca utilizada para criptografar senhas foi o **bcrypt**, por apresentar características fundamentais para proteger as credenciais dos usuários.

Motivos da escolha do bcrypt:

1. **Proteção contra força bruta**
   O bcrypt é propositalmente **lento**, dificultando tentativas de adivinhação (brute force) em massa.
2. **Uso automático de "salt"**
   O bcrypt aplica um **salt aleatório automaticamente** a cada senha, o que impede ataques com tabelas pré-calculadas (rainbow tables).
3. **Adaptável com o tempo**
   Ele permite ajustar o **fator de custo (work factor)**, aumentando a complexidade conforme o poder computacional evolui.
4. **Reconhecido e consolidado**
   O bcrypt é amplamente utilizado e recomendado em aplicações web modernas e por especialistas em segurança.

```python
@app.route( rule: "/registrar", methods=["POST"])
def registrar():
    data = request.json
    usuario = data.get("usuario")
    senha = data.get("senha")

    if not usuario or not senha:
        return jsonify({"erro": "Campos obrigatórios"}), 400

    usuarios = carregar_usuarios()
    for u in usuarios:
        if u["usuario"] == usuario:
            return jsonify({"erro": "Usuário já existe"}), 400

    senhaHash = bcrypt.generate_password_hash(senha).decode("utf-8")
    usuarios.append({"usuario": usuario, "senhaHash": senhaHash})

    with open(USUARIOS_PATH, "w") as f:
        json.dump(usuarios, f, indent=2)

    return jsonify({"mensagem": "Usuário registrado com sucesso"}), 201

/login
@app.route( rule: "/login", methods=["POST"])
def login():
    data = request.json
    usuario = data.get("usuario")
    senha = data.get("senha")

    usuarios = carregar_usuarios()
    for u in usuarios:
        if u["usuario"] == usuario and bcrypt.check_password_hash(u["senhaHash"], senha):
            return jsonify({"mensagem": "Login bem-sucedido"}), 200

    return jsonify({"erro": "Usuário ou senha inválido"}), 401
```

Comparação com outras abordagens

Método	Seguro para senhas?	Usa salt?	Lento/adaptável?	Recomendado?
SHA-256	❌ Não	❌ Manual	❌ Rápido demais	Não
SHA-256 + salt	⚠️ Parcialmente	✅ Manual	❌	Só para estudo
bcrypt	✅ Sim	✅ Automático	✅ Sim	✅ Sim
PBKDF2 / Argon2	✅ Sim	✅ Sim	✅ Sim	✅ Sim

Conclusão

A escolha do bcrypt foi feita com foco em **segurança**, **boas práticas de criptografia moderna** e **conformidade com padrões reais de sistemas web**. Ele garante que mesmo que a base de dados de usuários seja comprometida, as senhas continuem protegidas contra ataques comuns.