Nome: Gabriel Brito Bitencourt

Nome: Mateus Brito Bitencourt

Nome: Cecília Weselowsky da Cunha

Nome: Gabriel Marques Costa

Plano de Testes de Sistema: DockShield

Projeto: DockShield - Sistema de Análise de Vulnerabilidades em Containers **Disciplina:** Ferramentas de Desenvolvimento Web (FDW)

1. Introdução

Este documento apresenta o roteiro de testes para a validação funcional do MVP (Produto Mínimo Viável) do projeto **DockShield**. O escopo abrange a verificação da integração entre os microsserviços, a eficácia dos mecanismos de segurança e a persistência de dados.

O objetivo é demonstrar, de forma reprodutível, que o sistema atende aos requisitos de autenticação segura, controle de acesso e visualização de relatórios de vulnerabilidade.

2. Escopo do Teste

Este plano foca na validação da seguinte funcionalidade crítica:

- **Autenticação e Acesso Seguro:** Verificação do cadastro, login via token JWT, redirecionamento entre microsserviços e visualização de dados protegidos.

3. Pré-requisitos do Ambiente

Para a execução deste plano, o ambiente de teste deve atender aos seguintes requisitos mínimos:

- **Sistema Operacional:** Linux (Ubuntu 20.04/22.04 LTS recomendados) ou Windows com suporte a WSL2.
- **Software:** Docker Engine e Docker Compose instalados e configurados.
- **Rede:** Acesso à internet para *pull* das imagens e portas 3000, 5000 e 80 disponíveis.

4. Preparação do Ambiente (Setup)

Antes da execução dos testes, é necessário configurar a infraestrutura. Siga os passos abaixo para garantir que os microsserviços se comuniquem corretamente na rede local.

4.1. Obtenção dos Artefatos

Clone o repositório oficial do projeto contendo os códigos-fonte e arquivos de orquestração:

git clone https://github.com/SEU_USUARIO/fatec-scs-fdw-2025-2-DockShield.git

cd fatec-scs-fdw-2025-2-DockShield/P2

4.2. Configuração de Rede (Crucial)

O sistema utiliza redirecionamentos HTTP entre o serviço de autenticação (Porta 3000) e a aplicação principal (Porta 5000). Para que o navegador consiga seguir esses redirecionamentos, é necessário configurar o endereço IP da máquina host.

1. **Identifique o IP da máquina de teste:** Execute no terminal: ip addr (Linux) ou ipconfig (Windows). *Anote o endereço IPv4 da interface de rede principal (ex: 192.168.0.28).*
2. **Atualize o Frontend (Node.js):** Edite o arquivo loginprogweb/.env:

nano loginprogweb/.env

Altere a variável FLASK_EXTERNAL_URL:

FLASK_EXTERNAL_URL=http://<SEU_IP_REAL>:5000/

3. **Atualize o Backend (Flask):** Edite o arquivo de configuração do servidor web:

nano web_server/web_config.ini

Na seção [LOGIN], altere a chave url_node:

url_node = http://<SEU_IP_REAL>:3000/login.html

4.3. Inicialização dos Serviços

Execute o comando abaixo para construir as imagens (ou baixá-las) e iniciar os containers em modo de orquestração:

sudo docker-compose up --build --force-recreate

Aguarde a estabilização dos serviços. O terminal exibirá logs indicando que o servidor Node.js e o Apache estão ouvindo nas portas designadas.

5. Roteiro de Testes (Casos de Teste)

CT-01: Verificação da Integridade da Infraestrutura

Objetivo: Garantir que todos os componentes da arquitetura de microsserviços estão ativos.

Passo	Ação	Resultado Esperado
1	Abrir um novo terminal e executar sudo docker ps.	Devem ser listados três containers ativos: fatec-frontend-auth, fatec-backend-app e fatec-mongo-db.
2	Verificar logs do banco de dados no terminal principal.	O MongoDB deve indicar "Waiting for connections" na porta 27017.

CT-02: Cadastro e Autenticação de Usuário

Objetivo: Validar a criação de identidade e a geração do token de acesso.

Passo	Ação	Resultado Esperado

1	Abrir o navegador e acessar http://<SEU_IP>:3000/login.html.	A interface de login deve ser carregada.
2	Clicar em "Cadastrar-se" e preencher o formulário com um novo usuário.	O sistema deve confirmar o cadastro e redirecionar para a tela de login.
3	Inserir as credenciais recém-criadas e confirmar.	O sistema deve validar o hash da senha, gerar o cookie auth_token e redirecionar automaticamente o navegador para a porta 5000.

CT-03: Acesso e Visualização de Relatórios (Backend)

Objetivo: Validar a integração entre os serviços e a renderização dos dados de vulnerabilidade.

Passo	Ação	Resultado Esperado
1	Após o redirecionamento do CT-02, observar a página carregada.	Deve ser exibida a página inicial do Dashboard (Flask) listando as imagens Docker disponíveis no banco.
2	Clicar em uma das imagens listadas (ex: ubuntu:latest).	O sistema deve exibir o relatório detalhado da análise, convertendo o conteúdo Markdown armazenado no banco para HTML visualizável.

CT-04: Validação de Segurança (Controle de Acesso)

Objetivo: Verificar a eficácia do middleware de proteção contra acesso não autorizado.

Passo	Ação	Resultado Esperado
1	Abrir uma janela anônima no navegador (sem cookies).	Navegador limpo, sem sessão ativa.
2	Tentar acessar diretamente a URL do backend: http://<SEU_IP>:5000/.	O sistema deve interceptar a requisição, identificar a ausência do token JWT e redirecionar forçadamente o usuário para a tela de login (:3000).

6. Procedimento de Encerramento (Teardown)

Após a conclusão dos testes, recomenda-se a execução dos procedimentos abaixo para garantir a limpeza do ambiente e a preservação dos dados persistentes.

1. **Parar a execução:** No terminal onde o Docker está rodando, pressione Ctrl + C.
2. **Remover containers e redes:**

```
sudo docker-compose down
```

Nota: Este comando não remove o volume de dados (mongo_data), garantindo que os registros do banco de dados sejam preservados para a próxima execução.

7. Informações Adicionais

Conforme requisito 2.7 do projeto, as imagens Docker oficiais do sistema encontram-se disponíveis publicamente no DockerHub sob os seguintes repositórios, permitindo a execução sem necessidade de *build* local:

- **Frontend:** docker pull avaliacaon2/fatec-scs-imagem-dockshield-frontend
- **Backend:** docker pull avaliacaon2/fatec-scs-imagem-dockshield-backend
- **Database:** docker pull avaliacaon2/fatec-scs-imagem-dockshield-db