PentestFlow - SISTEMA DE GERENCIAMENTO DE UM PROCESSO DE PENTEST

RONCA, José Renato Ferreira; SOUZA, Kleber Márcio de.

Palavras-chave: Pentest, Software, Segurança

INTRODUÇÃO

A realização de testes de penetração (pentests) representa um desafio significativo tanto para empresas quanto para profissionais de segurança da informação. A ausência de sistemas comerciais amplamente disponíveis para gerenciar essas atividades gera dificuldades na organização do fluxo de trabalho, resultando em atrasos na comunicação, documentação dispersa e relatórios pouco padronizados.

As empresas, muitas vezes, enfrentam obstáculos para acompanhar o progresso dos testes de maneira clara e objetiva, enquanto os profissionais de segurança dedicam tempo excessivo em tarefas manuais na hora de reunir as informações, gerar relatórios e manter um feedback com o cliente.

Com a crescente demanda por soluções tecnológicas que otimizem processos de cibersegurança, torna-se essencial implementar plataformas que simplifiquem o gerenciamento de pentests e ofereçam maior transparência. De forma geral, os softwares de apoio contribuem para reduzir falhas humanas e aumentar a consistência dos resultados, garantindo que as organizações mantenham um nível adequado de proteção.

Nesse contexto, o PentestFlow está sendo desenvolvido com o intuito de centralizar informações, organizar escopos, documentar vulnerabilidades e gerar relatórios técnicos e executivos de forma estruturada.

Essa solução visa não apenas facilitar a documentação de processos de pentest, mas também proporcionar ao profissional de segurança uma fonte única da verdade para reunir as informações de forma automatizada e integrada com ferramentas corriqueiras, ao mesmo passo que permite uma maior transparência entre

o profissional e o cliente durante todo o processo, otimizando o tempo gasto e aumentando a confiabilidade do mesmo.

A Assim, o desenvolvimento do **PentestFlow** surge como uma resposta eficaz para atender às necessidades de gestão e acompanhamento de testes de penetração no mercado de segurança da informação.

OBJETIVO

O objetivo do **PentestFlow** é servir como uma plataforma integrada para o gerenciamento de testes de penetração (pentests), atendendo tanto empresas quanto profissionais de segurança da informação. O sistema permitirá a definição clara de escopos, o registro de vulnerabilidades identificadas de forma manual e automatizada, o anexo de evidências e a geração de relatórios técnicos e executivos de forma estruturada, centralizada e acessível.

A solução contará com uma interface web destinada às empresas (clientes), possibilitando o acompanhamento do progresso dos testes, a visualização de descobertas e o controle do escopo contratado. Para os profissionais de segurança, será disponibilizado um ambiente dedicado para organizar as etapas do pentest, documentar resultados de forma padronizada e otimizar a comunicação com os clientes.

MÉTODO

Para levantar as informações necessárias ao desenvolvimento do software, foi realizada uma análise de mercado com foco em empresas que oferecem serviços de pentest. Identificou-se que, em geral, as maiores utilizam soluções próprias para gerenciar seus processos, mas não existem plataformas comerciais amplamente acessíveis que atendam empresas menores. Essa ausência de soluções abertas cria lacunas no fluxo de trabalho, como a falta de centralização de dados, padronização na documentação e agilidade na comunicação entre as partes envolvidas.

Com base nesse cenário, foram definidos casos de uso (use cases) que descrevem as principais funcionalidades esperadas do sistema. Esses casos de uso foram organizados em um documento markdown e serviram como guia para o desenvolvimento. A implementação seguiu os princípios descritos no livro “Domain-

Driven Design” (EVANS, 2017), visando manter uma organização clara entre domínio, aplicação e infraestrutura, e a evolução do sistema foi acompanhada por meio de testes descritivos de funcionalidade, realizados a cada implementação concluída, garantindo que as etapas fossem cumpridas de forma incremental e consistente até a entrega do protótipo.

DESENVOLVIMENTO

Para garantir uma estrutura de código legível e de fácil manutenção, foram aplicados princípios descritos no livro *"Código Limpo"* (MARTIN, 2011), visando facilitar futuras alterações e ajustes no sistema. Na autenticação, foram utilizados diferentes métodos: para a API, a autenticação é realizada via tokens JWT gerados usando NestJS/JWT e NestJS/passport enquanto na versão web foi implementado o controle da sessão por meio de cookies armazenando os tokens enviados pelo backend.

As tecnologias utilizadas no backend incluem o Node 22.18 com o framework NestJS em sua versão 11 e a lib Jest na versão 29.7 para testes automatizados. O banco de dados escolhido foi o PostgreSQL, o Docker foi incorporado ao ambiente de desenvolvimento para facilitar a configuração e execução em outras máquinas, além de simplificar o processo de hospedagem, já que a imagem contém todas as dependências necessárias, foi também utilizado o github actions para garantir a padronização e segurança do código além da cobertura de testes.

Também está sendo utilizado o AWS S3 como storage para evidências e relatórios e a AWS lambda com Python para a geração de relatórios juntamente com IA. No desenvolvimento do aplicativo web, optei por utilizar o React na versão 18.0.0, juntamente com o Vite na versão 6.3.5.

CONCLUSÃO

O desenvolvimento do PentestFlow está em um estágio avançado, com várias funcionalidades essenciais já implementadas e em funcionamento. Embora o sistema ainda não esteja completamente finalizado, os progressos feitos até o momento indicam que o projeto está caminhando para atender integralmente às necessidades de empresas e profissionais de segurança na gestão de pentests.

As regras implementadas incluem: toda o controle de cadastro e autenticação de usuários, cadastro de escopos e criação de projetos a partir dos mesmos, cadastros das informações e evidências, geração de relatórios a partir das evidências coletadas e integração com IA, e inserção de informações a partir de integração com ferramentas terceiras.

REFERÊNCIAS

MARTIN, Robert C. **Código Limpo**: habilidades práticas do agile software. 1. ed. Rio de Janeiro: Alta Books, 2011.

EVANS, Eric. **Domain-Driven Design**: atacando as complexidades no coração do software. 3. ed. Rio de Janeiro: Alta Books, 2016.

KUROSE, James F. **Redes de computadores e a Internet**: uma abordagem top-down. 8. ed. Porto Alegre: Pearson, 2021.

LYON, Gordon Fyodor. **Exame de Redes com Nmap**. Rio de Janeiro: Ciencia Moderna, 2009.

SEITZ, Justin. **Black Hat Python**: programação python para hackers e pentesters. 2. ed. São Paulo: Novatec Editora, 2024.

MOTA FILHO, João Eriberto. **Análise de Tráfego em Redes TCP/IP**: utilize tcpdump na análise de tráfegos em qualquer sistema operacional. São Paulo: Novatec Editora, 2013.