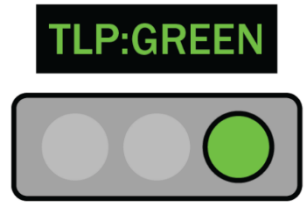


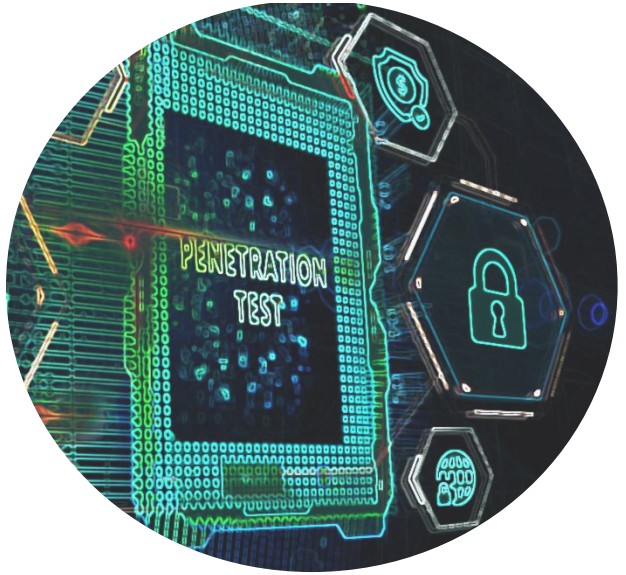
RNEP-GW

MÓDULO AUDITORIAS

fct csirt

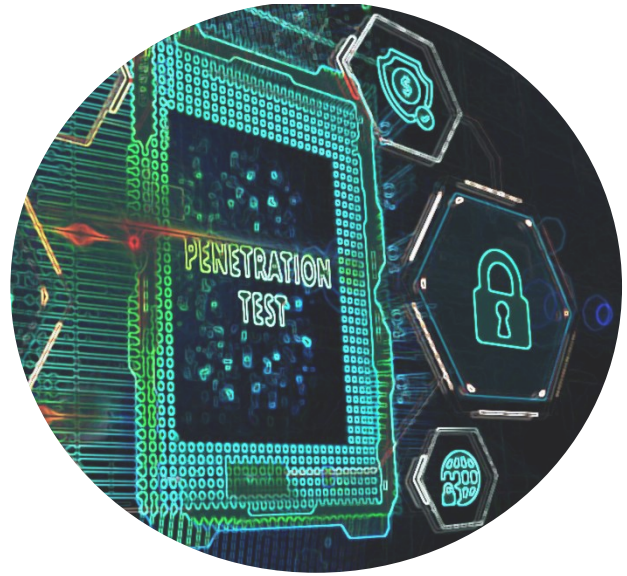
RNEP-GW

AUDITORIA



- Teste controlado de sistemas, aplicações e redes
- Simula ataques reais e identifica vulnerabilidades
- Combina ferramentas automatizadas e análise manual
- Relatório ao final propõe estratégias de mitigação

TIPOS DE AUDITORIA



Auditoria de Rede

- Infraestrutura, firewall, switches roteadores

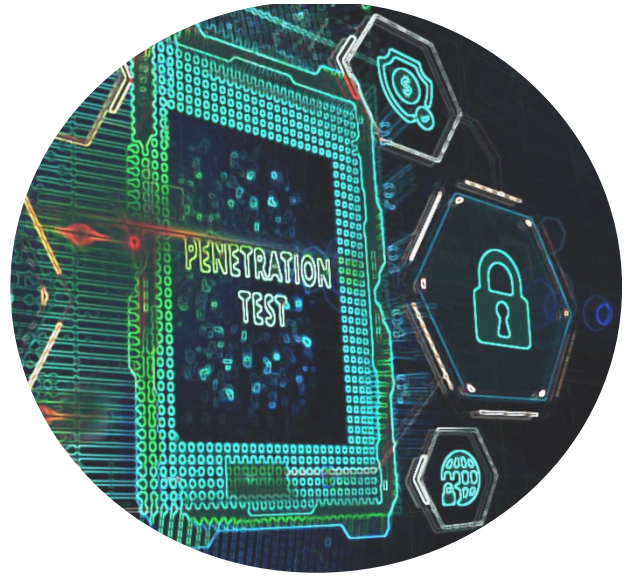
Auditoria de Hardware

- Dispositivos físicos, IoTs, servidores

Auditoria de Aplicações Web

- Sites, APIs, serviços online

TIPOS DE AUDITORIA



Auditoria de Dispositivos Móveis

- Apps Android/iOS, integração com backend

Auditoria Wireless

- Redes Wi-Fi, autenticação, segmentação

Auditoria física

- Acesso não autorizado a edifícios e salas de servidores

ABORDAGENS DE AUDITORIA

Auditorias Black Box

- Nenhuma informação prévia

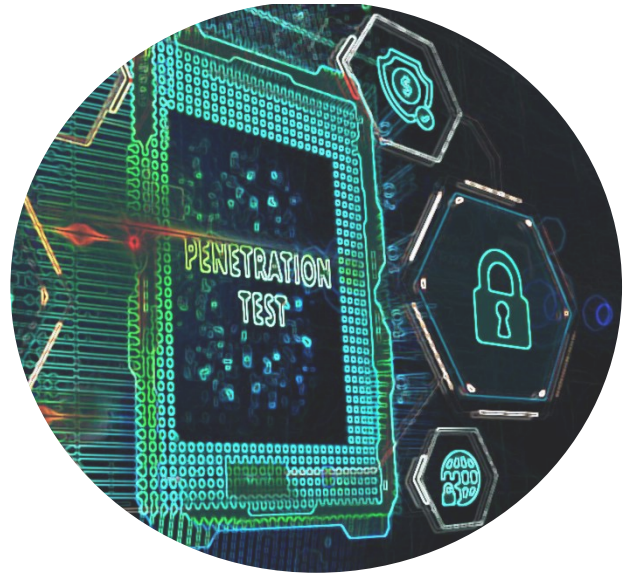
Auditorias White Box

- Informação detalhada

Auditorias Gray Box

- Informação parcial

FASES DA AUDITORIA (PENTESTING)



Planeamento e Reconhecimento

- Coleta de informações públicas, footprinting

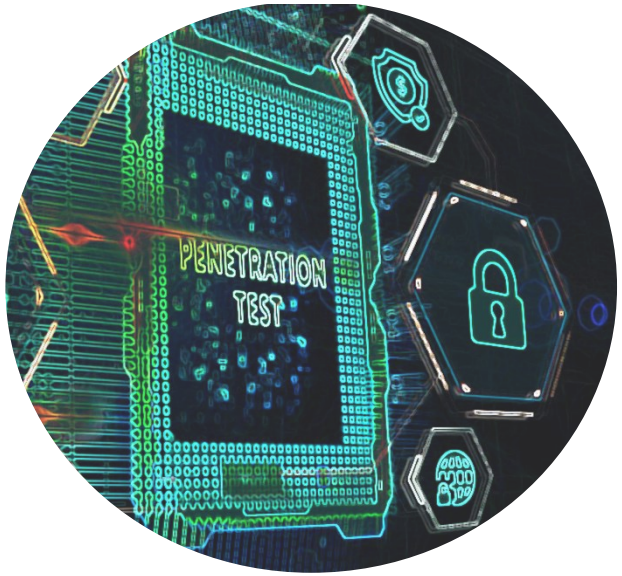
Scanning e Enumeração

- Identificação de hosts, portas, serviços, usuários

Análise de Vulnerabilidades

- Mapeamento de falhas conhecidas

FASES DA AUDITORIA (PENTESTING)



Exploração (Exploitation)

- Tentativa controlada de explorar vulnerabilidades

Pós-exploração (opcional)

- Elevação de privilégios, movimentação lateral

Análise e Relatório

- Documentação detalhada e recomendações

AUDITORIAS WEB

Auditoria web é imprescindível

- Superfície exposta 24/7
- Alvo preferido dos atacantes

Vulnerabilidades mais comuns (OWASP Top 10)

- Falha no controle de acesso (Broken Access Control) | Falhas criptográficas | Injections | Design Inseguro | Configuração incorreta de segurança (Security Misconfiguration) | Componentes vulneráveis e desatualizados | Falhas de identificação e autenticação | Falhas de software e de integridade de dados | Falhas de registo (logging) e de monitoramento de segurança | Server-Side Request Forgery (SSRF)

<https://owasp.org/www-project-top-ten/>

OWASP TOP 10 VULNERABILIDADES

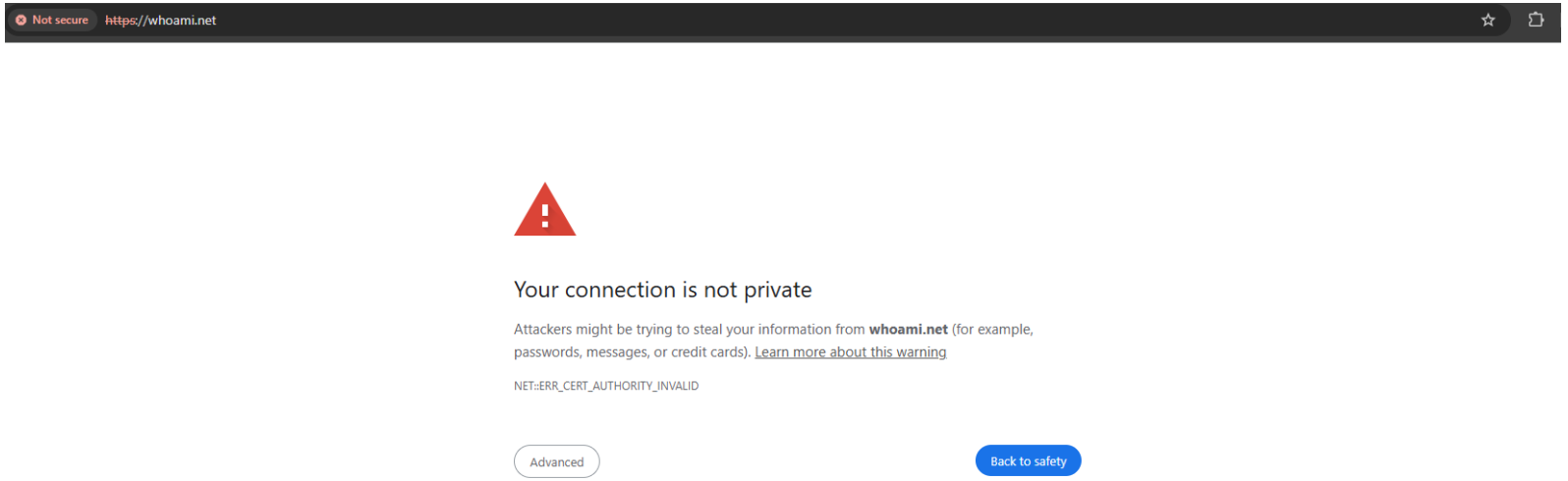
Falha no controle de acesso (Broken Acess Control)

- Acesso indevido. Ex.: Utilizador comum consegue aceder a dados de um administrador apenas mudando o valor de um parâmetro no URL

OWASP TOP 10 VULNERABILIDADES

Falhas Criptográficas (Cryptographic Failures)

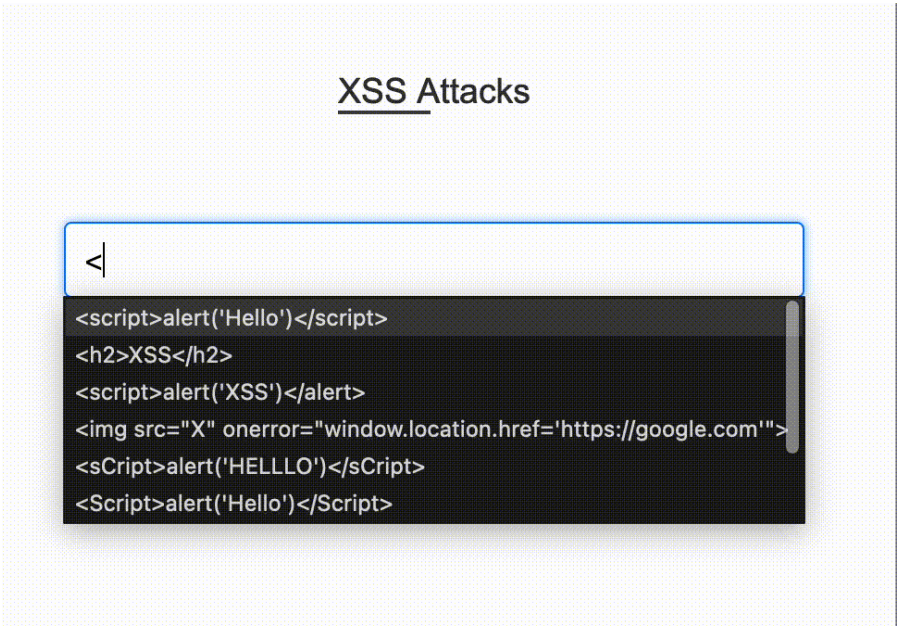
- Informações sensíveis não são protegidas. Ex.: Um site que ainda usa HTTP sem TLS e expõe credenciais de login



OWASP TOP 10 VULNERABILIDADES

Injection

- Dados não confiáveis são enviados a um intérprete de código por meio de uma entrada de formulário ou algum outro envio de dados a um aplicativo web



OWASP TOP 10 VULNERABILIDADES

Design Inseguro (Insecure Design)

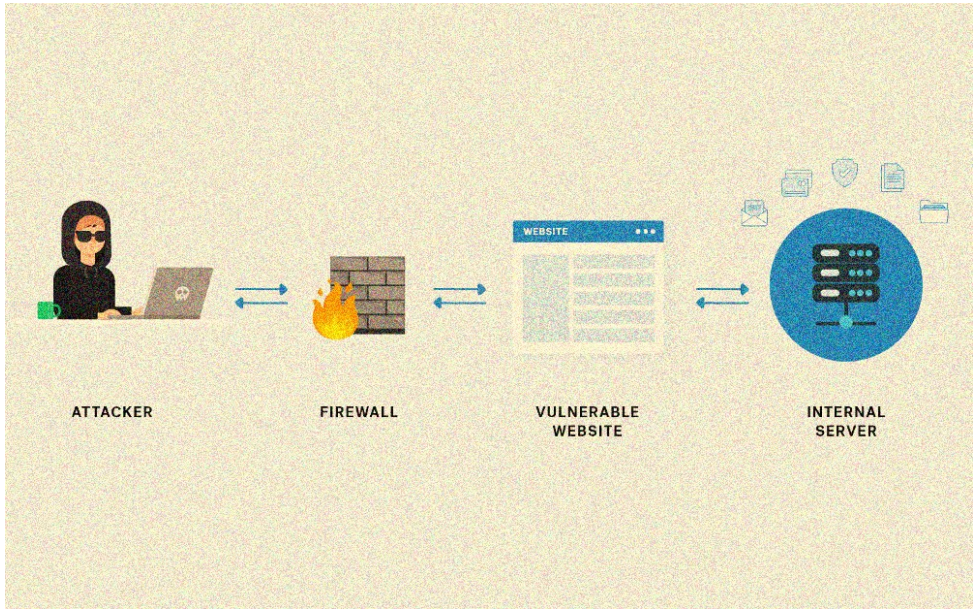
- Foca na falta de princípios de segurança desde o início do desenvolvimento de uma aplicação. Ex.: Uma aplicação que armazena senhas em *plaintext*

Username	Email	Password
admin	admin@gmail.com	admin
budi	budi@gmail.com	admin
ani	ani@gmail.com	Ani19900703
asep	asep@gmail.com	Asep#6969
udin	udin@gmail.com	P@ssw0rd

OWASP TOP 10 VULNERABILIDADES

Server-Side Request Forgery (SSRF)

- Permite que um atacante abuse do servidor para fazer requisições internas, muitas vezes acedendo a metadados ou informações que deveriam estar isoladas



FERRAMENTAS

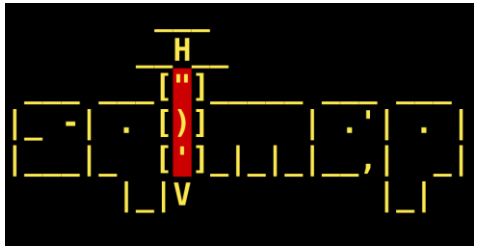
- Varredura e Enumeração

- Análise de Vulnerabilidades

- Foco em Aplicações Web

FERRAMENTAS

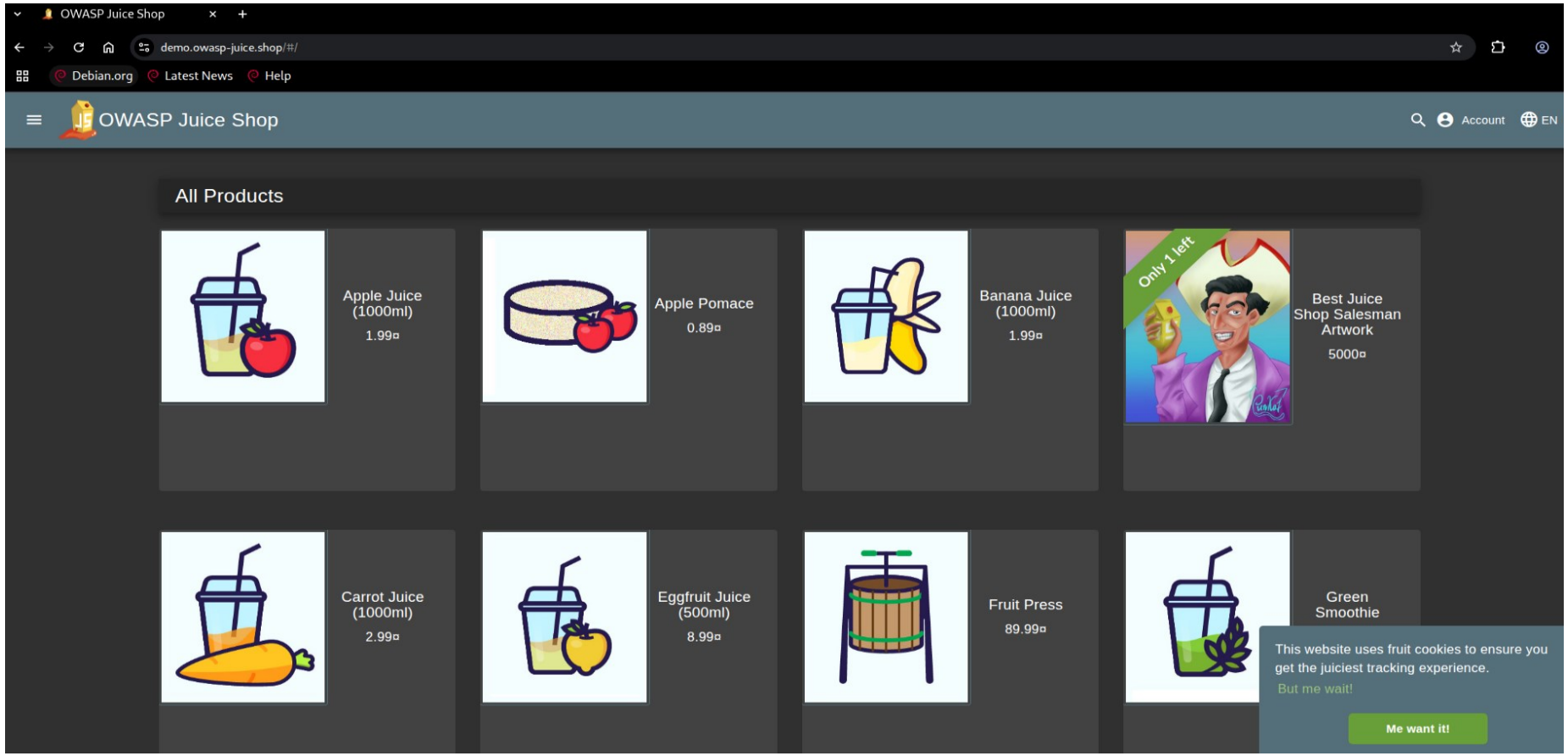
- Exploitation



(sqlmap)

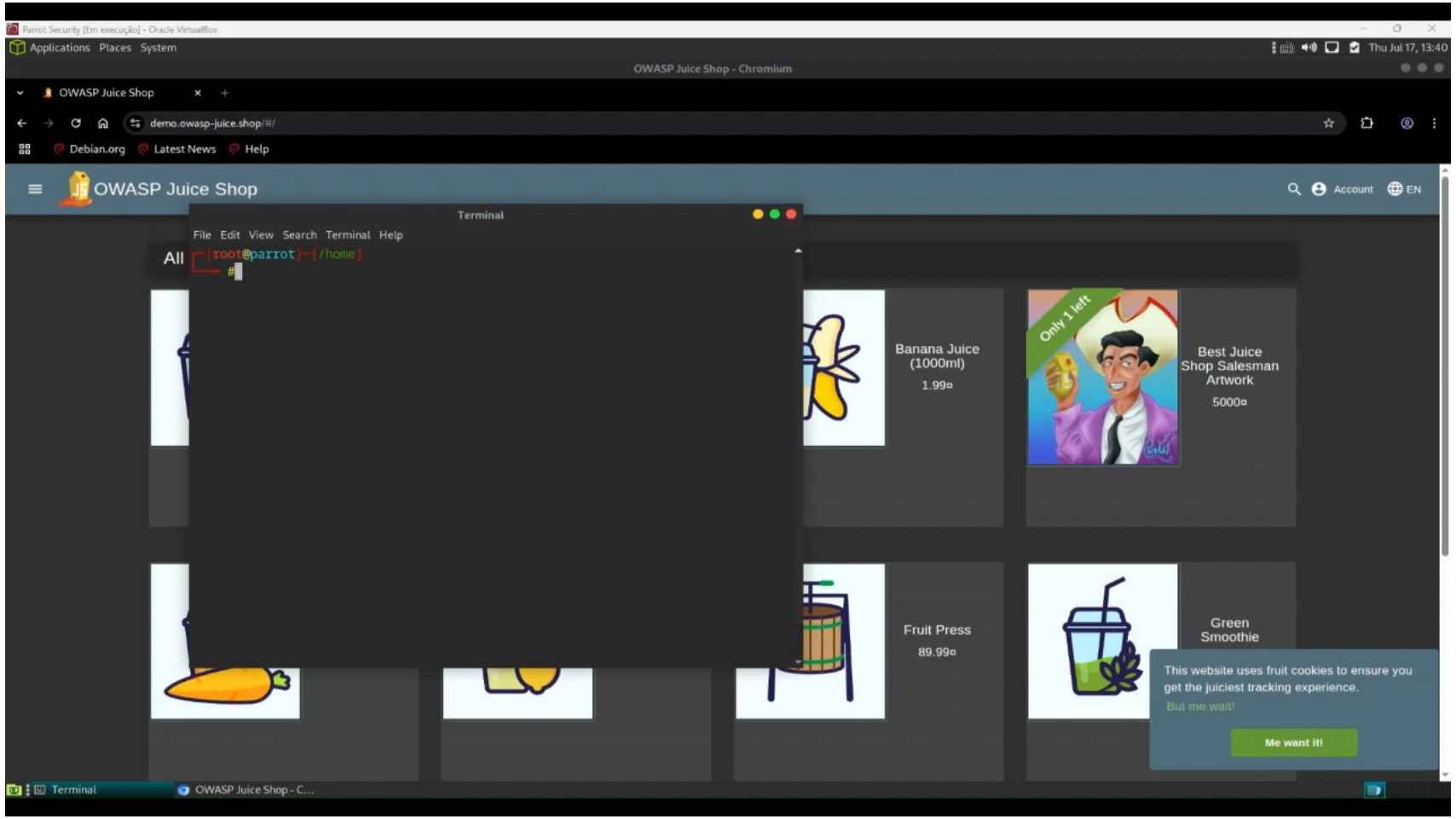
- Sistemas Operativos para Pentesting

DEMONSTRAÇÃO

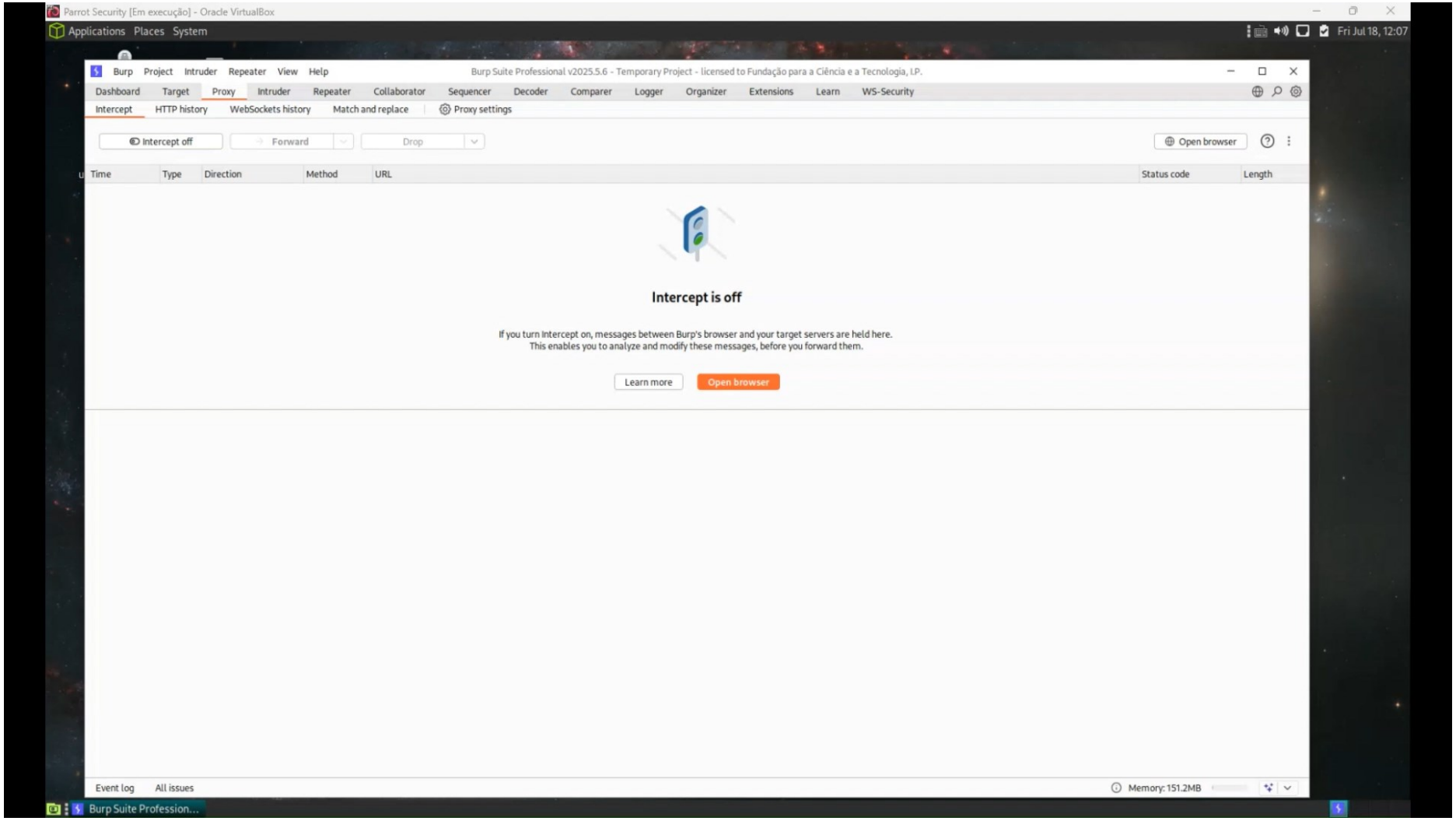


https://demo.owasp-juice.shop/

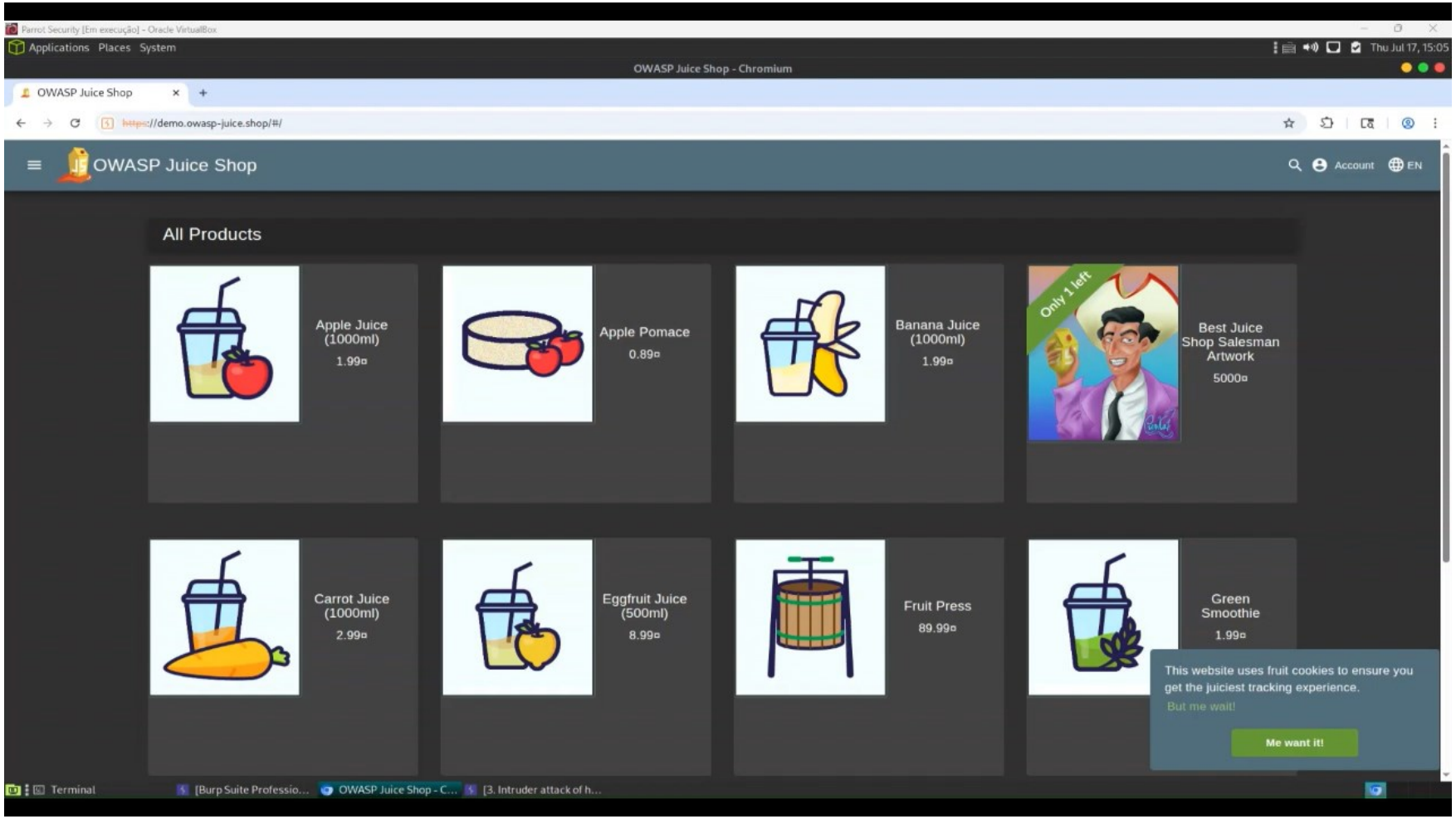
DEMONSTRAÇÃO



DEMONSTRAÇÃO



DEMONSTRAÇÃO



RELATÓRIO

Estrutura comum:

- Metodologia e escopo / Vulnerabilidades encontradas (com grau de severidade) / Recomendações

Linguagem adaptada ao público (gestores vs. técnicos)

Importância de ser objetivo e priorizar riscos

RELATÓRIO

1 Informações Gerais

Auditoria	
Início:	04/11/2022 --:--
Final:	04/11/2022 --:--
Organização:	Pobre Segurança
Contactos:	Sheila Souza <sheilasouza@pobreseguranca.tk>

Documento	
Autor(es) :	

2 Introdução

2.1 Objetivos

Esta auditoria tem como principal objetivo identificar e documentar possíveis vulnerabilidades existentes no website da empresa Pobre Segurança (http://pobreseguranca.tk) passíveis de serem exploradas por atacantes externos.

2.2 Tipo de testes

A auditoria foi executada através de testes "black-box" e a informação previamente fornecida foi a seguinte:

- http://pobreseguranca.tk

RELATÓRIO

4 METODOLOGIA

4.1 TESTES AUTOMATIZADOS DE VULNERABILIDADES

Software	Resultados
WPScan	A ferramenta WPScan encontrou os seguintes resultados: o Server: Apache/2.4.52 (Ubuntu) o http://pobreseguranca.tk/xmlrpc.php o http://pobreseguranca.tk/readme.html o http://pobreseguranca.tk/wp-cron.php o WordPress version 5.2 identified o WordPress theme in use: Neve - Version: 3.4.1 (80% confidence) < pode ser confirmado aqui http://pobreseguranca.tk/wp-content/themes/neve/style.css> o Plugin(s) Identified: • otter-blocks| Location: http://pobreseguranca.tk/wp-content/plugins/otter-blocks/ | Version: 1.3.1 (Versão certa 2.0.14 confirmável através de http://pobreseguranca.tk/wp-content/plugins/otter-blocks/readme.txt & http://pobreseguranca.tk/wp-content/plugins/otter-blocks/CHANGELOG.md)
Idyllum Labs	A ferramenta Idyllum Labs encontrou os seguintes resultados: o SSL Issues found. Traffic to site is not properly encrypted o IP: [redacted] o Open ports: 22 (tcp) – ssh o Port 80 (tcp) – http o CMS • Wordpress 6.0 o Other • Open-Graph-Protocol (website)

RELATÓRIO

4.2 TESTES MANUAIS

- Foi possível fazer a enumeração de plugins usados através do seguinte link: http://pobreseguranca.tk/index.php?rest_route=/
- E, seguindo o padrão do WordPress de localizações de informações sobre plugins, foi possível encontrar a versão de alguns dos plugins através do link: https://pobreseguranca.tk/wp-content/plugins/nome-plugin/readme.txt

Plugins:

- oembed
- redux 4.3.19
- extendify-gutenberg-patterns-and-templates
- seopress 6.0.2
- nv
- elementor 3.6.0
- otter 2.0.14
- wp-site-health
- wp-block-editor

* Também foi possível encontrar a versão do plugin elementor através de link obtido por meio do scan feito pelo Idyllum Labs - http://pobreseguranca.tk/?p=174 .

- Existe divulgação de informação sensível em vários lugares:
  - No código fonte da página principal existe um comentário que divulga um endereço de email;
  - Existe um post de blog com o nome "Teste" onde há divulgação do que parece ser nomes de usuários do wordpress;

RELATÓRIO

5 Resultados

5.1 Enumeração

Sistema	Âmbitos
Hostname	pobreseguranca.tk
Endereço IPV4	[redacted]
Endereço IPV6 (se aplicável)	0[redacted]
Versão do Wordpress	6.0
Plugins instalados	Gutenberg, Otter (2.0.14), Elementor (3.6.0), SEOPress (6.0.2)

5.2 Vulnerabilidades Detetadas

Serviços e Comunicações

VulnID	Vuln/Info	Impacto	Descrição
01	Divulgação de informação Sensível	Alto	É possível obter informação sensível através do código fonte da página principal e a enumeração dos usuários do WordPress através de post do blog. Estas informações facilitam um ataque de brute force.
02	Insufficient Access Control leading to Subscriber+ Remote Code Execution	Alto	Versão do plugin Elementor (3.6.0) utilizada pelo website está exposta a uma vulnerabilidade que, se explorada, permite a execução não autorizada de várias ações AJAX que tornam possível que um atacante modifique dados do site além de ser capaz de fazer o upload de ficheiros maliciosos que podem ser usados para obter remote code execution (RCE).
03	Uso do username 'admin'	Médio	Através da enumeração dos usuários do WordPress, que foi possível fazer por meio do descobrimento de divulgação de informação sensível, foi possível verificar que existe um usuário 'admin', o que facilita a execução de um ataque de brute force.
04	WordPress Admin Page	Médio	A página de login do WordPress está disponível

RELATÓRIO

6 RECOMENDAÇÕES

Recomendamos as seguintes tomadas de ação, por forma a mitigar ou eliminar as falhas de segurança identificadas no capítulos anterior:

Vulnerabilidades	
VulnID	**Recomendação**
01	Apagar qualquer informação sensível, esconder ou deletar post "Test".
02	Houve um patch para esta vulnerabilidade na versão 3.6.3 do plugin Elementor. Contudo, a versão mais atualizada do plugin é a 3.7.8. Recomendamos atualizar o plugin à versão mais recente sempre que possível, ou no mínimo à versão 3.6.3.
03	Evitar usar nomes de usuário como 'admin'. Mas como o nome dos usuários não pode ser mudado no WordPress, garantir que usa uma password segura de forma a diminuir as chances de haver uma intrusão através de brute-force.
04	A página de login da framework está acessível. A mesma deveria ser limitada ao host ou existir algum tipo de ACL de modo a restringir o acesso à mesma.
05	Implementar o Header X-Frame-options ou Content-Security-Policy, através das configurações do Apache, em todas as páginas web do website. Se for necessário que a página possa ser usada em frames por outras páginas dentro do mesmo servidor, então o header deve ser configurado como SAMEORIGIN, caso contrário, deve ser configurado como DENY. Se for implementada uma Content Security Policy, a diretiva "frame-ancestors" deve ser usada.
06	O porto ssh só deve estar aberto para IPs específicos dos administradores.

A IMPORTÂNCIA CONTÍNUA DAS AUDITORIAS

- Novos riscos surgem constantemente
- O resultado de uma auditoria é um retrato momentâneo do sistema
- Mudanças em códigos, infraestrutura ou utilizadores impactam a segurança
- Auditorias devem ser recorrentes, não pontuais

IDEIAS CHAVE

- Existem diversos tipos de auditorias, dependendo da informação que é fornecida ao auditor
- Aplicações web são alvos críticos e exigem atenção especial
- A validade dos resultados de uma auditoria reporta-se apenas ao momento em que os testes são realizados

Questões?

Obrigado.

csirt.fct.pt/podcast

csirt.fct.pt/mooc

csirt.fct.pt/etc