Atividade Orientada a Ensino

Acadêmico: Diogo Leite Gomes
RGA: 2022.1907.038-4
Professor: Carlos Alberto da Silva
Atividade: Atividade Orientada a Ensino sobre Segurança computacional

1. INTRODUÇÃO

As atividades desenvolvidas na Atividade Orientada a Ensino tiveram como foco o estudo e a aplicação de técnicas de segurança computacional no contexto do Trabalho de Conclusão de Curso. O trabalho consistiu na investigação de vulnerabilidades em aplicações web de uma instituição de ensino, com ênfase na identificação de ativos expostos, tecnologias utilizadas, componentes WordPress desatualizados e vulnerabilidades conhecidas catalogadas por CVE.

Diferentemente de uma exploração ofensiva em ambiente real, as atividades foram conduzidas com caráter acadêmico, preventivo e documental. Foram utilizadas ferramentas de reconhecimento e análise para relacionar versões de plugins e temas WordPress com vulnerabilidades públicas. Por questões éticas e de confidencialidade, os nomes completos da instituição e dos domínios analisados foram omitidos ou parcialmente mascarados no relatório e no TCC.

2. OBJETIVOS DA ATIVIDADE

O objetivo geral da atividade foi documentar as etapas técnicas realizadas no TCC, demonstrando o processo de reconhecimento, validação de ativos, análise de vulnerabilidades e classificação dos riscos identificados.

- Mapear subdomínios e ativos web expostos publicamente.
- Validar quais ativos estavam acessíveis via HTTP/HTTPS.
- Identificar tecnologias utilizadas nas aplicações web analisadas.
- Selecionar aplicações baseadas em WordPress para análise específica.
- Relacionar plugins, temas e versões identificadas com CVEs públicas.
- Classificar as vulnerabilidades por severidade, com base em pontuações CVSS.
- Consolidar os resultados em planilhas e tabelas para uso no TCC.

3. ATIVIDADES REALIZADAS

3.1 Preparação do Ambiente

A primeira etapa consistiu na preparação do ambiente de análise. Para isso, foi utilizada uma máquina virtual no Oracle VM VirtualBox, executando o Kali Linux como sistema operacional principal. Esse ambiente concentrou as ferramentas utilizadas no reconhecimento e na análise das aplicações web.

- Oracle VM VirtualBox: virtualização do ambiente de análise.
- Kali Linux: sistema operacional com ferramentas voltadas à segurança.
- Terminal Linux: execução dos comandos de coleta e validação.
- Planilhas eletrônicas: organização das CVEs, quantidades, severidade e plugins afetados.

3.2 Reconhecimento de subdomínios

A etapa de reconhecimento teve como finalidade ampliar a visibilidade sobre a superfície de ataque. Para isso, foi utilizada a ferramenta Subfinder, que realiza descoberta passiva de subdomínios a partir de fontes públicas.

```
subfinder -d dominio-analisado.com.br -o subdominios.txt
```

-d = domínio alvo que será analisado.
-o = arquivo onde será salvo o resultado da enumeração.

O resultado dessa etapa foi uma lista de subdomínios potencialmente pertencentes à instituição analisada. Esses ativos foram utilizados como entrada para a etapa seguinte de validação.

3.3 Validação de ativos e identificação de tecnologias

Após a coleta de subdomínios, foi utilizada a ferramenta httpx para verificar quais ativos estavam ativos e quais tecnologias estavam expostas. Essa etapa permitiu identificar códigos de resposta HTTP, títulos das páginas, servidores web, frameworks e possíveis aplicações WordPress.

```
httpx -l subdominios.txt -title -status-code -tech-detect -o ativos.txt
```

-l = lista de URLs ou subdomínios que será utilizada como entrada.

-title = exibe o título da página encontrada.

-status-code = exibe o código HTTP retornado pela aplicação.

-tech-detect = tenta identificar tecnologias utilizadas no site, como servidor web, linguagem, CMS ou framework.

-o = arquivo onde será salvo o resultado da análise.

Com base nessa saída, foram selecionadas as aplicações que indicavam uso de WordPress, plugins ou temas conhecidos. Essa filtragem foi importante para direcionar a análise com WPScan apenas aos ativos compatíveis.

3.4 Validação de ativos e identificação de tecnologias

Nos ativos identificados como WordPress, foi executado o WPScan com token de API para obter informações sobre plugins, temas, versões e vulnerabilidades conhecidas. A ferramenta permitiu relacionar componentes detectados com registros públicos de vulnerabilidades.

```
wpscan --url https://dominio-analisado.com.br --api-token [TOKEN]
```

--url = endereço do site WordPress que será analisado.

--api-token = token da API do WPScan, utilizado para consultar a base de vulnerabilidades conhecidas.

[TOKEN] = chave de acesso fornecida pela plataforma WPScan.

A análise priorizou a identificação de versões vulneráveis, a versão corrigida informada pela ferramenta e as referências públicas das CVEs.

3.5 Organização das CVEs e classificação de severidade

Após a execução do WPScan, as vulnerabilidades foram organizadas em planilhas contendo código CVE, quantidade de ocorrências, pontuação CVSS, severidade e plugin afetado. Essa etapa permitiu calcular a quantidade de CVEs distintas, a quantidade total de ocorrências e a distribuição por severidade e por componente.

4. RESULTADOS OBTIDOS

A análise identificou vulnerabilidades associadas principalmente a plugins WordPress desatualizados. Foram identificadas 49 CVEs distintas e, considerando repetições entre as aplicações analisadas, 97 ocorrências no total. As vulnerabilidades foram classificadas conforme sua severidade, utilizando como referência a pontuação CVSS.

Severidade	Quantidade de CVEs distintas
Crítico	3
Alto	4
Médio	42
Baixo	0
Total	**49**

A maior parte das vulnerabilidades foi classificada como média. Entretanto, a presença de vulnerabilidades críticas e altas reforça a necessidade de priorização na correção dos componentes vulneráveis.

Plugin	Crítico	Alto	Médio	Baixo	Total
Elementor	3	2	32	0	37
Page Builder by SiteOrigin	0	1	4	0	5
Smart Slider 3	0	1	6	0	7
Total	**3**	**4**	**42**	**0**	**49**

O plugin Elementor concentrou a maior parte dos achados, com 37 CVEs distintas, incluindo todas as vulnerabilidades críticas identificadas. Também foram identificadas vulnerabilidades relevantes nos plugins Page Builder by SiteOrigin e Smart Slider 3.

5. PRINCIPAIS VULNERABILIDADES ANALISADAS

Entre as vulnerabilidades priorizadas no TCC, destacaram-se as CVEs com maiores pontuações CVSS e maior potencial de impacto. A tabela a seguir resume as principais vulnerabilidades selecionadas para análise detalhada.

CVE	Descrição resumida	CVSS	Qtde
CVE-2020-7055	Upload arbitrário de arquivos no Elementor.	Crítico (9.9)	1
CVE-2020-7109	Cross-Site Scripting (XSS) no Elementor.	Crítico (9.8)	2
CVE-2023-47504	Leitura arbitrária de anexos no Elementor.	Crítico (9.8)	2
CVE-2020-13642	CSRF e XSS no Page Builder by SiteOrigin.	Alto (8.8)	2
CVE-2022-3357	PHP Object Injection no Smart Slider 3.	Alto (8.8)	2
Total			9

5.1 CVE-2020-7055 - Upload arbitrário de arquivos

A CVE-2020-7055 está associada a versões vulneráveis do plugin Elementor e envolve a possibilidade de upload indevido de arquivos. Como consequência, em cenários específicos, a falha pode permitir o envio de arquivos maliciosos ao servidor, comprometimento da aplicação ou alteração indevida de conteúdo. A mitigação consiste em atualizar o plugin para a versão corrigida, restringir permissões de upload, validar extensões e tipos de arquivos permitidos e monitorar diretórios públicos.

5.2 CVE-2020-7109 - Cross-Site Scripting

A CVE-2020-7109 corresponde a uma falha de Cross-Site Scripting em versões vulneráveis do Elementor. Esse tipo de vulnerabilidade pode permitir a execução de scripts

maliciosos no navegador do usuário, com impactos como roubo de sessão, redirecionamento indevido ou manipulação de conteúdo. Como mitigação, recomenda-se atualizar o plugin, sanitizar entradas, escapar saídas HTML e aplicar políticas como Content Security Policy.

5.3 CVE-2022-3357 - PHP Object Injection

A CVE-2022-3357 afeta versões vulneráveis do plugin Smart Slider 3 e está relacionada à injeção de objetos PHP. Dependendo do ambiente e dos componentes disponíveis, a falha pode permitir manipulação de objetos internos, alteração do comportamento da aplicação ou comprometimento parcial do sistema. A mitigação envolve atualizar o plugin para versão corrigida, evitar desserialização de dados não confiáveis e validar entradas recebidas pela aplicação.

6. MEDIDAS DE MITIGAÇÃO RECOMENDADAS

Com base nos achados, foram identificadas medidas de mitigação aplicáveis às aplicações analisadas e a ambientes semelhantes que utilizam WordPress.

- Atualizar continuamente o WordPress, plugins e temas para versões corrigidas.
- Manter inventário dos componentes utilizados em cada aplicação.
- Remover plugins e temas não utilizados.
- Revisar permissões de usuários e aplicar o princípio do menor privilégio.
- Validar entradas de usuários e restringir uploads de arquivos.
- Aplicar mecanismos de segurança como Content Security Policy em aplicações suscetíveis a XSS.
- Monitorar periodicamente novas CVEs associadas aos componentes utilizados.
- Executar auditorias recorrentes de segurança em ativos web expostos.

7. CONTRIBUIÇÃO DAS ATIVIDADES PARA O TCC

As atividades realizadas contribuíram diretamente para a construção do TCC, especialmente nas seções de metodologia, resultados, discussão e conclusão. A execução das ferramentas permitiu obter evidências técnicas sobre ativos expostos, tecnologias utilizadas e vulnerabilidades associadas a componentes WordPress.

A organização das CVEs em planilhas possibilitou a análise quantitativa dos achados, permitindo identificar a quantidade de CVEs distintas, a recorrência total das vulnerabilidades, a distribuição por severidade e os plugins mais afetados. Essas informações foram utilizadas para priorizar as vulnerabilidades críticas e altas discutidas no trabalho.