Análise de Vulnerabilidades Web em Universidade Privada

Jonathas Guedes Borges[1], **Carlos Alberto de Oliveira**[2]

[1]Curso de Bacharelado em Sistemas de Informação - Faculdade de Computação (FACOM) ,
[2] Faculdade de Computação (FACOM)
Universidade Federal do Mato Grosso do Sul (UFMS)
79070-900 – Campo Grande – MS – Brasil

jonathas.borges@ufms.br, carlos.silva@ufms.br

Abstract. *This article presents a web vulnerability analysis conducted at a private university, aiming to highlight the fragility of information security within organizations that manage vast amounts of devices. Utilizing tools such as Nmap, Kali Linux, Subfinder, and Httprobe, the research identified critical points of unauthorized access and potential flaws that could compromise the privacy and integrity of thousands of individuals' information. The study emphasizes the urgent need for implementing robust cybersecurity measures to protect personal and academic data, underscoring the social and academic relevance of the research by alerting to the inherent risks of neglecting information protection in digital environments.*

Resumo. *Este artigo apresenta uma análise de vulnerabilidades web realizada em uma universidade privada, com o objetivo de evidenciar a fragilidade da segurança da informação em organizações que gerenciam uma vasta quantidade de dispositivos. Utilizando ferramentas como Nmap, Kali Linux, Subfinder e Httprobe, a pesquisa identificou pontos críticos de acesso indevido e potenciais falhas que podem comprometer a privacidade e a integridade das informações de milhares de indivíduos. O estudo ressalta a urgência da implementação de medidas robustas de cibersegurança para proteger dados pessoais e acadêmicos, destacando a relevância social e acadêmica da pesquisa ao alertar sobre os riscos inerentes à negligência na proteção de informações em ambientes digitais.*

1. Introdução

No atual cenário digital, a informação tornou-se um dos ativos mais valiosos para organizações de todos os setores. Instituições de ensino superior, em particular, lidam diariamente com um volume massivo de dados sensíveis, abarcando desde registros acadêmicos e financeiros até informações pessoais de milhares de indivíduos. Contudo, a complexidade e a constante evolução das ameaças cibernéticas contrastam com a, por vezes, negligente adoção de práticas de segurança robustas. Este cenário é particularmente crítico em universidades privadas, onde a proteção de dados não apenas resguarda a instituição, mas também a privacidade de sua vasta comunidade acadêmica.

O objetivo geral deste estudo é realizar uma análise de vulnerabilidades web em uma universidade privada, visando evidenciar a falta de segurança em organizações que armazenam milhares de informações confidenciais de pessoas e, com isso, subsidiar a

adoção de ações preventivas e corretivas. Para tanto, esta pesquisa adotou uma abordagem exploratória e descritiva, focada na execução de testes de segurança ofensivos. A metodologia envolveu a aplicação de ferramentas de código aberto reconhecidas na área de segurança da informação, como Nmap, Subfinder e Httprobe, operando no ambiente Kali Linux, para a identificação de vulnerabilidades em ativos web da instituição-alvo. Foram encontradas, ao todo, 517 vulnerabilidades. Das quais, muitas permitem a execução de código arbitrário e movimentação lateral pela rede.

2. Fundamento Teórico

Este capítulo tem como objetivo apresentar os conceitos fundamentais que servem de base para a compreensão da análise de vulnerabilidades web realizada no presente estudo. Conforme a IBM [1], a varredura de vulnerabilidades, ou vulnerability scanning, é o processo de avaliar redes ou ativos de TI em busca de falhas de segurança que possam ser exploradas por ameaças. Essa prática é considerada a primeira etapa do ciclo de gerenciamento de vulnerabilidades e é crucial para identificar e corrigir fraquezas antes que sejam exploradas por cibercriminosos. Serão abordados desde os pilares da segurança da informação até os principais tipos de vulnerabilidades e as ferramentas utilizadas na pesquisa, fornecendo o embasamento teórico necessário para a discussão dos resultados.

2.1. Segurança da Informação

A segurança da informação (InfoSec) pode ser definida como a proteção de informações importantes contra acesso não autorizado, divulgação, uso, alteração ou interrupção. Conforme a IBM, a InfoSec ”ajuda a garantir que os dados organizacionais confidenciais estejam disponíveis para usuários autorizados, permaneçam confidenciais e mantenham sua integridade”[2]. Seu principal objetivo é, portanto, garantir a **integridade**, a **confidencialidade** e a **disponibilidade** dos dados, independentemente do seu formato ou meio de armazenamento. Essas três características são amplamente reconhecidas como os pilares da segurança da informação, também conhecidos como Tríade CID.

3. Ferramentas

Para a identificação e mitigação de vulnerabilidades, a análise de segurança da informação faz uso de ferramentas especializadas que auxiliam no mapeamento e análise de sistemas e aplicações. Neste estudo, foram empregadas as seguintes ferramentas de código aberto, amplamente reconhecidas na comunidade de cibersegurança:

3.1. Kali Linux

O Kali Linux [3] é uma distribuição Linux de código aberto, baseada em Debian, desenvolvida especificamente para testes de penetração e auditoria de segurança. Essa distribuição se destaca por integrar centenas de ferramentas, configurações e scripts pré-instalados, permitindo que profissionais e pesquisadores se concentrem em atividades como computação forense, pentest, engenharia reversa e detecção de vulnerabilidades. Para o presente estudo, o Kali Linux serviu como a plataforma operacional principal, proporcionando um ambiente robusto e completo para a execução das demais ferramentas utilizadas na análise.

3.2. Nmap

O Nmap [4] (”Network Mapper”) é uma ferramenta de código aberto para exploração de redes e auditoria de segurança. Embora tenha sido projetado para escanear rapidamente grandes redes, o Nmap também é eficaz para a análise de hosts individuais. Ele opera utilizando pacotes IP brutos para determinar quais hosts estão ativos na rede, quais serviços estão sendo oferecidos por esses hosts, quais sistemas operacionais estão em execução, e outras características relevantes para a segurança. Além disso, o Nmap possui um poderoso Motor de Scripts (NSE - Nmap Scripting Engine), que permite estender suas funcionalidades, possibilitando a detecção de vulnerabilidades específicas, exploração de falhas conhecidas, e a realização de tarefas avançadas de descoberta e auditoria. No contexto deste trabalho, o Nmap foi fundamental tanto para o mapeamento inicial da superfície de ataque da universidade quanto para a varredura por vulnerabilidades utilizando seus scripts especializados, identificando dispositivos e serviços expostos com potenciais falhas de segurança.

3.3. Subfinder

O Subfinder [5] é uma ferramenta de descoberta de subdomínios rápida e passiva. Sua principal função é enumerar o maior número possível de subdomínios associados a um domínio-alvo, utilizando diversas fontes de dados, como mecanismos de busca e bases de dados de DNS. A identificação de subdomínios é crucial em uma análise de vulnerabilidades, pois expande a superfície de ataque conhecida de uma organização, revelando ativos que podem estar menos protegidos ou serem desconhecidos pela equipe de segurança. No estudo, o Subfinder foi empregado para identificar ativos digitais da universidade que poderiam não estar evidentes inicialmente.

3.4. Httprobe

O Httprobe [6] é uma ferramenta que complementa a enumeração de subdomínios, sendo utilizada para verificar a disponibilidade de portas HTTP/S em massa. Ele processa uma lista de domínios ou subdomínios e retorna apenas aqueles que possuem um servidor web respondendo ativamente nas portas 80 (HTTP) ou 443 (HTTPS). No presente trabalho, o Httprobe foi essencial para validar quais dos subdomínios identificados pelo Subfinder estavam de fato ativos e acessíveis via web, permitindo focar a análise nas aplicações que poderiam apresentar vulnerabilidades.

4. Metodologia

Este capítulo detalha a metodologia de pesquisa empregada para a análise de vulnerabilidades web na universidade privada. A abordagem adotada foi de natureza exploratória e descritiva, e, de acordo com Harper et al. [7], pautou-se na realização de uma avaliação de vulnerabilidades (vulnerability assessment), com o objetivo de simular o comportamento de um atacante para identificar e mapear potenciais falhas de segurança. A pesquisa foi conduzida seguindo etapas sequenciais de reconhecimento, varredura e análise, utilizando ferramentas de código aberto amplamente reconhecidas no campo da cibersegurança.

4.1. Ambiente de Testes e Ferramentas

Os testes foram realizados utilizando a distribuição **Kali Linux**, que forneceu o ambiente operacional necessário e um vasto repositório de ferramentas de segurança. Para a

execução das varreduras e coletas de informação, empregou-se o **Proxychains**, garantindo o anonimato da máquina de origem e a continuidade dos testes em caso de bloqueio por *Web Application Firewalls* (WAFs). As ferramentas específicas utilizadas em cada etapa da metodologia estão detalhadas a seguir.

4.2. Etapas da Análise de Vulnerabilidades

A análise de vulnerabilidades foi dividida em fases distintas, conforme descrito abaixo:

4.2.1. Reconhecimento e Coleta de Subdomínios

A fase inicial consistiu no reconhecimento da superfície de ataque da instituição. Para tal, foi utilizada a ferramenta **Subfinder** para a busca abrangente por subdomínios associados ao domínio principal da universidade. Paralelamente, foi realizada uma consulta exploratória no domínio da universidade utilizando a ferramenta Censys [8]. Essa busca permitiu a identificação de endereços IP correlacionados ao domínio e dos serviços que estavam em execução em cada um deles. Durante esta etapa, foi possível identificar um painel de *login* do setor financeiro da instituição.

4.2.2. Verificação de Ativos Web Ativos

Com a lista de subdomínios obtida pelo Subfinder, a etapa seguinte foi a verificação de quais desses subdomínios estavam ativos e respondendo a requisições web. Para isso, a ferramenta **Httprobe** foi empregada para testar as principais portas utilizadas por servidores web. A consulta resultou na identificação de **2586 domínios e subdomínios**, dos quais **1019 foram confirmados como ativos** pelo Httprobe.

4.2.3. Filtragem de Endereços IP Ativos

Após a identificação dos domínios ativos, foi necessário filtrar o arquivo de saída do Httprobe para extrair apenas os endereços IP dos domínios que retornaram um *status code* `200 OK`, indicando uma resposta bem-sucedida do servidor. O arquivo resultante continha a lista dos endereços IP dos ativos web que estavam operacionais e prontos para a fase de varredura de vulnerabilidades.

4.2.4. Varredura de Vulnerabilidades com Nmap Scripting Engine (NSE)

A etapa final da metodologia consistiu na varredura detalhada dos endereços IP ativos em busca de vulnerabilidades conhecidas. Para isso, foi utilizado o **Nmap** em conjunto com seu **Motor de Scripts (NSE - Nmap Scripting Engine)**. Especificamente, o script `vulscan/vulscan.nse` foi empregado para verificar a existência de CVEs (Common Vulnerabilities and Exposures) nos serviços expostos pelos *hosts* da universidade.

5. Análise e Discussão dos Resultados

Nesta seção, são apresentados e discutidos os resultados obtidos a partir da análise de vulnerabilidades web realizada com as ferramentas Nmap e Subfinder sobre o domínio

avaliado da universidade privada. A metodologia detalhada no Capítulo 3 permitiu identificar uma série de vulnerabilidades associadas a códigos CVE (Common Vulnerabilities and Exposures).

5.1. Visão Geral das Vulnerabilidades Identificadas

A varredura e o mapeamento dos ativos digitais da instituição revelaram a presença de diversas vulnerabilidades, indicando potenciais brechas na segurança da informação. A análise dos dados coletados permitiu quantificar e classificar essas vulnerabilidades.

Para padronizar a identificação e comunicação dessas falhas, utiliza-se o CVE (Common Vulnerabilities and Exposures), que, de acordo com Harper et al. [7], é uma compilação de vulnerabilidades conhecidas publicamente. Esse dicionário público de vulnerabilidades e exposições de segurança cibernética fornece um identificador único para cada falha conhecida. A severidade dessas vulnerabilidades é avaliada através do CVSS (Common Vulnerability Scoring System), que, conforme o National Vulnerability Database (NVD) [9], é um framework aberto para comunicar as características e impactos de vulnerabilidades de TI. Ele atribui uma pontuação numérica e um nível (Baixo, Médio, Alto ou Crítico) com base nas características e impacto da falha, auxiliando na priorização de correções.

Para uma compreensão inicial das falhas de maior criticidade, a Tabela 1 a seguir ilustra as vulnerabilidades de alta severidade (CVSS High/Critical) identificadas durante a varredura e que representam os maiores riscos para a instituição.

Tabela 1: Vulnerabilidades Identificadas

Código CVE	Qtde	Nível CVSS	Breve Descrição
CVE-2015-7292 [10]	61	CVSS: 9.8 (Critical)	Vulnerabilidade crítica de execução de código remoto em Amazon Fire OS (Driver Havok).
CVE-2018-1169 [11]	61	CVSS: 8.8 (High)	Vulnerabilidade de execução de código arbitrário em Amazon Music Player 6.1.5.1213. Falta de validação de string.
CVE-2017-9450 [12]	61	CVSS: 7.8 (High)	AWS CloudFormation bootstrap tools (aws-cfn-bootstrap) antes de 1.4-19.10 permite execução de código arbitrário com privilégios de root.
CVE-2017-17069 [13]	61	CVSS: 7.8 (High)	ActiveSetupN.exe em Amazon Audible for Windows Nov/2017 permite execução de código DLL arbitrário via DLL hijacking com dwmapi.dll maliciosa.
CVE-2005-2046 [14]	61	CVSS: 7.5 (High)	Vulnerabilidades de SQL Injection em DUware DUamazon Pro 3.0 e 3.1 permitem a execução de comandos SQL arbitrários.

Continua na próxima página

Tabela 1: Vulnerabilidades Identificadas

Código CVE	Qtde	Nível CVSS	Breve Descrição
CVE-2006-6593 [15]	61	CVSS: 7.5 (High)	AMAZONIA MOD para phpBB (zufallscodepart.php) vulnerável a remote file inclusion PHP.
CVE-2005-3976 [16]	61	CVSS: 7.5 (High)	Múltiplos produtos DUware (incluindo DUamazon 3.1): SQL Injection em type.asp permite execução de comandos SQL arbitrários via parâmetro iType.
CVE-2012-5817 [17]	61	CVSS: 7.4 (High)	Codehaus XFire 1.2.6 (em Amazon EC2 API Tools Java) falha na validação de hostname no certificado X.509.
CVE-2017-6189 [18]	61	CVSS: 7.3 (High)	Vulnerabilidade de caminho de busca não confiável em Amazon Kindle for PC 1.19 permite execução de código arbitrário e DLL hijacking.

5.2. Descrição Detalhada das CVEs mais severas

Nesta subseção, são descritas em profundidade as três vulnerabilidades de maior severidade, apresentadas na **Tabela 1**, destacando suas características técnicas, o impacto potencial para a universidade e as recomendações de mitigação, com base nas informações da **National Vulnerability Database (NVD)** [9]. A relevância dessas falhas reside no fato de que sua exploração pode levar a consequências graves, como acesso não autorizado a dados confidenciais, negação de serviço, ou comprometimento total do sistema, exigindo atenção e correção imediatas.

5.2.1. CVE-2015-7292

- **Severidade**: CVSS 9.8 (Crítica)
- **Descrição**: Vulnerabilidade de **estouro de buffer baseado em pilha (*stack-based buffer overflow*)** na função `havok write` localizada em `drivers/staging/havok/havok.c`, no sistema operacional **Amazon Fire OS** (versões anteriores a 15 de janeiro de 2016). Essa falha permite que atacantes causem uma condição de negação de serviço (*panic* do sistema) ou, possivelmente, tenham outros impactos não especificados, enviando uma *string* longa para o dispositivo `/dev/hv`.
- **Impacto Potencial à Universidade**: Dada a natureza específica desta vulnerabilidade, que afeta o **Amazon Fire OS**, o impacto direto à universidade dependerá da utilização de dispositivos ou servidores com este sistema operacional em sua infraestrutura. Se a universidade possuir **dispositivos baseados em Amazon Fire OS** (como tablets Amazon Fire, ou servidores/dispositivos embarcados que utilizem esta versão do OS, especialmente em ambientes de IoT ou gerenciamento de tela/conteúdo), a exploração desta falha pode levar a:

  - **Negação de Serviço**: Causar o *travamento* (*panic*) ou reinicialização forçada dos dispositivos afetados, tornando-os indisponíveis para o uso acadêmico ou administrativo (ex: tablets usados em laboratórios, quiosques de informação, sistemas de apresentação).
  - **Comprometimento Inespecífico**: A menção de "possivelmente ter outros impactos não especificados"em uma vulnerabilidade de execução de código/estouro de buffer de alta severidade (CVSS 9.8) geralmente indica o potencial para **execução de código arbitrário**. Isso significa que um atacante poderia, teoricamente, obter controle sobre o dispositivo, acessando informações nele armazenadas, instalando *malware* (como *ransomware*), ou usando o dispositivo como um ponto de acesso para explorar outros sistemas na rede da universidade.
  - **Riscos a Dados Confidenciais**: Se os dispositivos afetados lidarem com dados sensíveis de alunos, professores ou da administração, há um risco de comprometimento da confidencialidade e integridade dessas informações.
- **Recomendações de Mitigação**:
  - **Identificação**: Primeiramente, é crucial que a universidade **identifique todos os dispositivos que operam com Amazon Fire OS** em sua rede, verificando as versões do sistema operacional.
  - **Atualização Urgente**: Se dispositivos vulneráveis forem encontrados, eles devem ser **imediatamente atualizados** para a versão mais recente do Amazon Fire OS que contenha a correção para esta CVE.
  - **Isolamento de Rede**: Dispositivos baseados em Amazon Fire OS, se utilizados, devem ser **segmentados em redes isoladas** (VLANs), minimizando o risco de acesso não autorizado a sistemas críticos em caso de comprometimento.
  - **Monitoramento e Hardening**: Implementar monitoramento contínuo de segurança e aplicar práticas de *hardening* (endurecimento de configurações) nesses dispositivos para reduzir a superfície de ataque e detectar comportamentos anormais.

5.2.2. CVE-2018-1169

- **Severidade**: CVSS 8.8 (Alta)
- **Descrição**: Vulnerabilidade de execução de código arbitrário em **Amazon Music Player 6.1.5.1213**. A falha existe no processamento de *URI handlers*, resultando da falta de validação adequada de uma *string* fornecida pelo usuário antes de usá-la para executar uma chamada de sistema. Exige interação do usuário (visitar página maliciosa ou abrir arquivo).
- **Impacto Potencial à Universidade**: A exploração desta vulnerabilidade exigiria que um atacante já tivesse acesso local à máquina ou convencesse um usuário a baixar e executar um arquivo/link malicioso. O impacto potencial inclui:
  - **Execução de Código Arbitrário no Contexto do Usuário**: Um atacante pode executar código malicioso no computador do usuário que interagir com o artefato comprometido, comprometendo a estação de trabalho.
  - **Roubo de Credenciais/Dados**: Se o computador afetado for utilizado para acessar sistemas ou armazenar dados confidenciais (ex: por funcionários

administrativos ou pesquisadores), o atacante poderia roubar credenciais de acesso ou exfiltrar informações sensíveis.
    - **Propagação de Malware**: O computador comprometido poderia ser usado para espalhar *malware* (como *ransomware* ou vírus) para outros sistemas na rede da universidade.
- **Recomendações de Mitigação**:
    - **Atualização de Software**: É fundamental que todas as instalações do **Amazon Music Player** na rede da universidade sejam imediatamente atualizadas para a versão mais recente que corrija esta CVE.
    - **Conscientização do Usuário**: Educar usuários sobre os riscos de clicar em links suspeitos ou abrir arquivos de fontes não confiáveis.
    - **Princípio do Menor Privilégio**: Garantir que os usuários operem com os menores privilégios necessários, limitando o impacto de uma execução de código.
    - **Monitoramento de Endpoint**: Utilizar soluções de segurança de *endpoint* que monitorem e detectem atividades suspeitas, como chamadas de sistema incomuns.

5.2.3. CVE-2017-9450

- **Severidade**: CVSS 7.8 (Alta)
- **Descrição**: Vulnerabilidade em **AWS CloudFormation bootstrap tools** (também conhecido como `aws-cfn-bootstrap`, versões anteriores a 1.4-19.10). Permite que usuários locais executem código arbitrário com privilégios de *root* (administrador) ao criar arquivos em um diretório não especificado (e vulnerável).
- **Impacto Potencial à Universidade**: Se a universidade utiliza o AWS CloudFormation para gerenciar sua infraestrutura na nuvem e algum servidor ou estação de trabalho na rede tiver uma versão vulnerável desta ferramenta, um usuário com acesso local (mesmo que limitado) poderia:
    - **Escalada de Privilégios para Root**: Obter controle total sobre o sistema operacional subjacente ao adquirir privilégios de *root*, ignorando as proteções de segurança.
    - **Comprometimento da Infraestrutura em Nuvem**: Usar o acesso de *root* para manipular, roubar dados ou comprometer recursos críticos na nuvem AWS da universidade.
    - **Persistência e Movimento Lateral**: Instalar *backdoors* ou outras ferramentas para manter o acesso e se mover lateralmente para outros sistemas na rede.
- **Recomendações de Mitigação**:
    - **Atualização Urgente**: Todas as instalações do pacote **AWS CloudFormation bootstrap tools (`aws-cfn-bootstrap`)** devem ser **imediatamente atualizadas** para a versão 1.4-19.10 ou superior.
    - **Auditorias de Permissões**: Realizar auditorias regulares de permissões de arquivos e diretórios em servidores e estações de trabalho, garantindo que não existam diretórios com permissões excessivas que possam ser explorados.

- **Princípio do Menor Privilégio**: Assegurar que os usuários e processos operem com o menor conjunto de privilégios possível.
- **Monitoramento de Integridade de Arquivos**: Implementar sistemas que monitorem alterações em arquivos e diretórios críticos do sistema.

6. Considerações Finais da Análise

A análise detalhada das vulnerabilidades web na universidade privada, conforme apresentada nas subseções anteriores, indicou a possível presença de falhas de segurança de alta criticidade, com destaque para a CVE-2015-7292, que possui classificação crítica (CVSS 9.8). A prevalência de vulnerabilidades potencialmente relacionadas a produtos Amazon e DUware, com diversos tipos de falhas como execução de código remoto, DLL hijacking e injeção de SQL, sugere que a instituição pode estar exposta a riscos significativos.

A potencial exploração dessas vulnerabilidades, especialmente as de alta severidade, poderia acarretar consequências graves como:

- Acesso não autorizado a dados confidenciais de alunos, professores e funcionários (registros acadêmicos, financeiros, dados pessoais).
- Interrupção de serviços essenciais, resultando em negação de serviço e indisponibilidade de plataformas acadêmicas e administrativas.
- Comprometimento total de servidores e dispositivos, levando à instalação de *malware* (ex: *ransomware*), manipulação de dados ou uso da infraestrutura para outros ataques.
- Dano severo à reputação da universidade e perda de confiança da comunidade acadêmica.

A detecção dessas potenciais vulnerabilidades por meio de ferramentas automatizadas de varredura demonstra a eficácia da metodologia empregada neste estudo para identificar riscos cibernéticos. Os resultados reforçam a premente necessidade de a organização adotar uma postura proativa em segurança da informação. A correção ou mitigação dessas potenciais falhas é essencial para fortalecer a postura de segurança da universidade, prevenir incidentes cibernéticos e garantir a integridade, confidencialidade e disponibilidade das informações críticas.

Este estudo reitera a importância de práticas regulares de auditoria de segurança, da adoção de protocolos atualizados, da implementação de políticas de *hardening*, e da configuração adequada de *firewalls* e outros controles de segurança. A contínua vigilância e a resposta rápida a potenciais vulnerabilidades são etapas fundamentais para a proteção de milhares de informações confidenciais em um cenário digital em constante evolução.

7. Trabalhos Futuros

A presente pesquisa realizou uma análise inicial de vulnerabilidades web na infraestrutura de uma universidade privada, identificando a possível presença de diversas vulnerabilidades de severidade alta e crítica. Contudo, dada a complexidade e a constante evolução do cenário de cibersegurança, bem como as limitações intrínsecas a qualquer estudo, diversas linhas de investigação podem ser exploradas para aprofundar e complementar os achados deste trabalho.

Sugere-se, portanto, para trabalhos futuros, as seguintes frentes de pesquisa:

- **Validação e Exploração Controlada das Vulnerabilidades**: Realizar testes de intrusão controlados (Pentest), com o devido consentimento e acompanhamento da instituição, para validar a explorabilidade das vulnerabilidades identificadas. Isso permitiria confirmar o real impacto e o nível de risco de cada falha, fornecendo dados mais precisos para a priorização de correções.
- **Análise de Segurança de Aplicações Específicas**: Aprofundar a análise em aplicações web de maior criticidade, como sistemas acadêmicos, portais de acesso a dados pessoais ou plataformas financeiras. Utilizar técnicas de análise de segurança de aplicações (SAST - Static Application Security Testing e DAST - Dynamic Application Security Testing) mais específicas.
- **Implementação e Teste de Controles de Mitigação**: Após a correção das vulnerabilidades pela universidade, realizar uma nova bateria de testes para verificar a eficácia das medidas de mitigação implementadas. Isso inclui a avaliação de Web Application Firewalls (WAFs) e outras defesas.
- **Análise de Vulnerabilidades em Outras Camadas da Rede**: Expandir a análise para outras camadas da infraestrutura da universidade, incluindo redes internas, sistemas de endpoints (estações de trabalho), servidores de e-mail e outros serviços que não sejam apenas web. Isso proporcionaria uma visão mais abrangente da postura de segurança da instituição.
- **Estudo de Conscientização e Engenharia Social**: Investigar a componente humana da segurança da informação na universidade, por meio de simulações de ataques de engenharia social (Phishing, Vishing) e avaliação do nível de conscientização dos usuários sobre as ameaças cibernéticas.
- **Análise Comparativa com Outras Instituições**: Realizar um estudo comparativo da postura de segurança da informação em relação a outras universidades (públicas ou privadas), utilizando metodologias similares, para identificar padrões e melhores práticas no setor educacional.
- **Desenvolvimento de Ferramentas ou Scripts Personalizados**: Propor ou desenvolver ferramentas e scripts personalizados que possam automatizar a detecção de tipos específicos de vulnerabilidades frequentemente encontradas no ambiente universitário ou que otimizem o processo de varredura.

A continuidade da pesquisa nesta área é fundamental para fortalecer a cibersegurança das instituições de ensino, que são alvos valiosos devido à grande quantidade de informações confidenciais que detêm.

Referências

[1] Matthew Kosinski e Amber Forrest. *What is Vulnerability scanning?* URL: https://www.ibm.com/think/topics/vulnerability-scanning. Acesso em: 17 jun. 2025.

[2] Jim Holdsworth e Matthew Kosinski. *O que é segurança da informação (InfoSec)?* URL: https://www.ibm.com/br-pt/topics/information-security. Acesso em: 2 mai. 2025.

[3] Offensive Security. *Kali Linux Documentation*. URL: https://www.kali.org/docs/. Acesso em: 5 mar. 2025.

[4] Fyodor. *Nmap: the Network Mapper*. URL: https://nmap.org/. Acesso em: 5 mar. 2025.

[5] ProjectDiscovery. *Subfinder*. URL: https://github.com/projectdiscovery/subfinder. Acesso em: 5 mar. 2025.
[6] Tomnomnom. *httprobe*. URL: https://github.com/tomnomnom/httprobe. Acesso em: 5 mar. 2025.
[7] Allen Harper et al. *Gray Hat Hacking: The Ethical Hacker's Handbook*. Fourth Edition. McGraw-Hill Education, 2015.
[8] Inc. Censys. *Censys: Attack Surface Management*. URL: https://censys.com/. Acesso em: 5 mar. 2025.
[9] National Institute of Standards e Technology (NIST). *National Vulnerability Database (NVD)*. URL: https://nvd.nist.gov/. Acesso em: 1 mai. 2025.
[10] National Institute of Standards e Technology (NIST). *CVE-2015-7292*. 2017. URL: https://nvd.nist.gov/vuln/detail/CVE-2015-7292. Acesso em: 1 mai. 2025.
[11] National Institute of Standards e Technology (NIST). *CVE-2018-1169*. 2018. URL: https://nvd.nist.gov/vuln/detail/CVE-2018-1169. Acesso em: 1 mai. 2025.
[12] National Institute of Standards e Technology (NIST). *CVE-2017-9450*. 2017. URL: https://nvd.nist.gov/vuln/detail/CVE-2017-9450. Acesso em: 1 mai. 2025.
[13] National Institute of Standards e Technology (NIST). *CVE-2017-17069*. 2017. URL: https://nvd.nist.gov/vuln/detail/CVE-2017-17069. Acesso em: 1 mai. 2025.
[14] National Institute of Standards e Technology (NIST). *CVE-2005-2046*. 2005. URL: https://nvd.nist.gov/vuln/detail/CVE-2005-2046. Acesso em: 1 mai. 2025.
[15] National Institute of Standards e Technology (NIST). *CVE-2006-6593*. 2006. URL: https://nvd.nist.gov/vuln/detail/CVE-2006-6593. Acesso em: 1 mai. 2025.
[16] National Institute of Standards e Technology (NIST). *CVE-2005-3976*. 2005. URL: https://nvd.nist.gov/vuln/detail/CVE-2005-3976. Acesso em: 1 mai. 2025.
[17] National Institute of Standards e Technology (NIST). *CVE-2012-5817*. 2012. URL: https://nvd.nist.gov/vuln/detail/CVE-2012-5817. Acesso em: 1 mai. 2025.
[18] National Institute of Standards e Technology (NIST). *CVE-2017-6189*. 2017. URL: https://nvd.nist.gov/vuln/detail/CVE-2017-6189. Acesso em: 1 mai. 2025.