Análise de Vulnerabilidades em Infraestrutura de Provedor de Internet: Uma Abordagem Prática de Teste de Penetração

Pedro Augusto França Lima Dutra[1], **Carlos Alberto da Silva**[1]

[1]Faculdade de Computação – Universidade Federal de Mato Grosso do Sul (UFMS)
Campo Grande – MS – Brasil

pedro.dutra@ufms.br

Abstract. *This paper presents a practical study of penetration testing applied to the infrastructure of an Internet Service Provider (ISP). Through systematic reconnaissance, service enumeration, and vulnerability exploration, critical misconfigurations were identified in network equipment, database servers, and web applications. The methodology follows a structured approach based on frameworks such as OWASP and PTES, covering phases from passive reconnaissance to controlled exploitation. The findings include exposed management interfaces, outdated software stacks, missing transport-layer encryption, and default credentials on perimeter devices. The work reinforces the importance of Defense in Depth and centralized log monitoring pipelines as countermeasures. All tests were performed in a controlled environment with prior authorization, and all identifying information has been anonymized to protect the organization.*

Resumo. *Este artigo apresenta um estudo prático de testes de penetração aplicados à infraestrutura de um Provedor de Serviços de Internet (ISP). Por meio de reconhecimento sistemático, enumeração de serviços e exploração controlada de vulnerabilidades, foram identificadas configurações críticas inadequadas em equipamentos de rede, servidores de banco de dados e aplicações web. A metodologia segue uma abordagem estruturada baseada em frameworks como OWASP e PTES, cobrindo as fases de reconhecimento passivo até a exploração controlada. Os achados incluem interfaces de gerenciamento expostas, pilhas de software desatualizadas, ausência de criptografia na camada de transporte e uso de credenciais padrão em dispositivos de borda. O trabalho reforça a importância da Defesa em Profundidade e de pipelines de monitoramento centralizado de logs como contramedidas. Todos os testes foram realizados em ambiente controlado com autorização prévia, e as informações identificadoras foram anonimizadas para proteger a organização.*

1. Introdução

O crescimento acelerado da conectividade no Brasil tem ampliado a superfície de ataque das infraestruturas de Provedores de Serviços de Internet (ISPs). De acordo com o Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil (CERT.br), o número de incidentes reportados no setor de telecomunicações cresce a cada ano, com destaque para falhas de configuração e exposição indevida de serviços [CERT.br 2023].

Nesse contexto, a prática de *Penetration Testing* (Pentest) surge como uma metodologia essencial para a identificação proativa de falhas de segurança antes que agentes maliciosos as explorem. O Pentest é definido como "um método de avaliação da segurança de um sistema de informação ou rede, simulando um ataque de um agente malintencionado" [EC-Council 2021].

Este Trabalho de Conclusão de Curso tem como objetivo documentar e analisar as vulnerabilidades identificadas durante um processo de teste de penetração autorizado realizado na infraestrutura de um ISP regional. O estudo cobre desde a fase de reconhecimento passivo até a exploração controlada de falhas críticas, incluindo: exposição de interfaces de gerenciamento, software obsoleto, ausência de criptografia no transporte de dados e uso de credenciais padrão em equipamentos de borda.

A relevância do trabalho reside não apenas no levantamento técnico das falhas, mas também na proposta de um pipeline de monitoramento centralizado baseado em tecnologias como Apache Kafka e OpenSearch, que mitiga os riscos de destruição de evidências forenses mesmo em cenários de comprometimento parcial da infraestrutura.

O restante deste artigo está organizado da seguinte forma: a Seção 3 apresenta o referencial teórico; a Seção 4 descreve a metodologia adotada; a Seção 5 detalha os resultados obtidos; a Seção 6 apresenta as propostas de mitigação; e a Seção 8 conclui o trabalho.

2. Trabalhos Relacionados

Trabalhos anteriores desenvolvidos na Faculdade de Computação da UFMS investigaram contextos relacionados ao pentest e à análise de vulnerabilidades. Fernandes [Fernandes 2025] conduziu um estudo conceitual sobre o papel dos testes de intrusão com o Metasploit Framework em ambientes de laboratório controlados. Fernandes e Silva [Fernandes e Silva 2025] aplicaram técnicas de varredura automatizada para identificar vulnerabilidades em aplicações web de empresas de leilões e instituições EAD. O presente trabalho se diferencia dessas abordagens ao conduzir um pentest autorizado sobre uma infraestrutura de rede real de um ISP, cobrindo não apenas a camada de aplicação web, mas também equipamentos de borda, servidores de banco de dados e interfaces de gerenciamento de containers, propondo adicionalmente um pipeline centralizado de logs como camada de resiliência forense.

3. Referencial Teórico

3.1. Teste de Penetração e Frameworks

O Pentest é estruturado em fases que variam conforme o framework adotado. O *Penetration Testing Execution Standard* (PTES) define sete fases principais: pré-engajamento, coleta de inteligência, modelagem de ameaças, análise de vulnerabilidades, exploração, pós-exploração e geração de relatório [PTES Technical Guidelines 2012].

O framework OWASP (*Open Web Application Security Project*) complementa essa visão com foco em aplicações web, mapeando os dez riscos mais críticos — o OWASP Top 10 — amplamente utilizados como base de referência em avaliações de segurança [OWASP Foundation 2021b].

3.2. Reconhecimento e Enumeração

A fase de reconhecimento é dividida em passiva e ativa. O reconhecimento passivo consiste na coleta de informações sem interação direta com o alvo, enquanto o reconhecimento ativo envolve o envio de pacotes e sondas aos sistemas-alvo [Stallings e Brown 2018].

Ferramentas como Nmap (*Network Mapper*) são amplamente utilizadas para enumeração de portas e serviços. O *Banner Grabbing*, técnica de captura de cabeçalhos de serviço, permite identificar versões de software e sistemas operacionais, possibilitando o mapeamento de CVEs (*Common Vulnerabilities and Exposures*) aplicáveis [Lyon 2023].

3.3. Vulnerabilidades em Infraestrutura de Rede

Equipamentos de borda, como roteadores MikroTik RouterOS [MikroTik 2024], são alvos frequentes por combinarem alta conectividade com configurações padrão inseguras. Pesquisas recentes demonstram que uma parcela significativa dos dispositivos MikroTik expostos à internet ainda opera com credenciais de fábrica [Matherly 2022], vulnerabilidade registrada no CVE-2018-14847 [NIST — National Institute of Standards and Technology 2018].

A exposição de serviços de banco de dados (MySQL, PostgreSQL, MariaDB) diretamente à internet representa uma falha grave, pois tais sistemas não foram projetados para operação sem uma camada de proteção de rede [OWASP Foundation 2021a].

3.4. Information Disclosure e CGNAT

O vazamento de informações (*Information Disclosure*) é classificado como uma vulnerabilidade de alta severidade no contexto de ISPs, especialmente quando envolve dados de infraestrutura CGNAT (*Carrier-Grade Network Address Translation*). A Lei 12.965/2014 (Marco Civil da Internet) e a Lei 13.709/2018 (LGPD) impõem obrigações de sigilo e proteção de dados de navegação dos usuários [Brasil 2014, Brasil 2018].

4. Metodologia

Esta pesquisa caracteriza-se como um estudo de caso de natureza aplicada, com abordagem qualitativa e quantitativa. Os testes foram realizados com autorização formal da organização-alvo, seguindo um escopo definido em contrato (*Rules of Engagement*). Todos os endereços IP e nomes de domínio foram anonimizados neste documento.

4.1. Ambiente e Escopo

O ambiente avaliado consiste na infraestrutura de rede de um ISP regional brasileiro. O escopo incluiu:

- Servidores web e de aplicação na DMZ;
- Equipamentos de roteamento de borda (concentradores CGNAT);
- Servidor de monitoramento de logs;
- Interfaces de gerenciamento de containers.

4.2. Ferramentas Utilizadas

O conjunto de ferramentas seguiu o padrão de distribuições de segurança ofensiva (Kali Linux 2024.1):

Tabela 1. Ferramentas utilizadas por fase do pentest

Fase	Ferramenta	Finalidade
Reconhecimento	Nmap 7.95	Enumeração de portas e serviços
Reconhecimento	WhatWeb	Fingerprint de tecnologias web
Análise de Vulnerabilidades	Nikto	Auditoria de cabeçalhos HTTP
Análise de Vulnerabilidades	FFUF	Fuzzing de diretórios e arquivos
Análise de Vulnerabilidades	DIRB	Enumeração de recursos web
Exploração	Hydra	Teste de força bruta
Exploração	cURL	Análise manual de respostas HTTP

4.3. Fases da Metodologia

A metodologia foi estruturada em quatro fases, conforme ilustrado:

Fase 1 — Reconhecimento Passivo e Ativo: Identificação de hosts ativos, portas abertas e versões de serviços via Nmap. Varredura de blocos /23 com flag `-open` para otimizar tempo.

Fase 2 — Análise de Vulnerabilidades: Correlação dos serviços identificados com CVEs conhecidos. Análise de cabeçalhos HTTP via Nikto e WhatWeb. Verificação de configurações padrão em equipamentos de rede.

Fase 3 — Exploração Controlada: Tentativas de autenticação com credenciais padrão. Análise de *Information Disclosure* em arquivos de diagnóstico. Verificação de SQL Injection e LFI em aplicações web identificadas.

Fase 4 — Documentação e Relatório: Compilação dos achados com classificação de severidade (Crítico, Alto, Médio, Baixo) baseada no CVSS (*Common Vulnerability Scoring System*) v3.1 [FIRST.org 2019].

5. Resultados e Análise

5.1. Fase 1: Descoberta de Ativos e Enumeração de Serviços

A varredura inicial do bloco de endereços do ISP identificou 64 hosts ativos com serviços variados. A Tabela 2 sintetiza os principais ativos descobertos e seus serviços associados (endereços anonimizados).

Tabela 2. Principais ativos identificados durante o reconhecimento

Host (Anon.)	Serviço	Porta	Observação
ISP-MON-01	PostgreSQL	5432/TCP	Banco de dados exposto
ISP-MON-01	HTTP/HTTPS	80,443/TCP	Nginx 1.18.0
ISP-EDGE-01	MikroTik API	8728/TCP	RouterOS exposto
ISP-EDGE-01	Winbox	8291/TCP	Interface proprietária
ISP-APP-01	MySQL	3306/TCP	MariaDB 11.8.3 exposto
ISP-SRV-01	FTP/Telnet	21,23/TCP	Protocolos inseguros
ISP-SRV-02	RDP	3389/TCP	Acesso remoto Windows
ISP-CAM-01	RTSP	554/TCP	Câmera IP exposta
ISP-WEB-01	HTTPS	443/TCP	Gophish na porta 9000

O comando utilizado para a varredura inicial foi

```
sudo nmap -Pn -F --open <BLOCO_ANONIMIZADO>/23
```

Listing 1. Varredura inicial do bloco de rede do ISP

5.2. Fase 2: Vulnerabilidades Identificadas

A análise do ambiente avaliado resultou na identificação de um total de 45 vulnerabilidades, distribuídas por severidade conforme a escala CVSS v3.1: 8 vulnerabilidades classificadas como **Críticas** (CVSS entre 9,0 e 10,0), 15 classificadas como **Altas** (CVSS entre 7,0 e 8,9) e 22 classificadas como **Médias** (CVSS entre 4,0 e 6,9). As subseções a seguir detalham as principais vulnerabilidades identificadas, com suas respectivas evidências, classificações e recomendações de mitigação.

5.2.1. Information Disclosure — Arquivo de Diagnóstico PHP

O servidor de aplicação web (ISP-APP-WEB) expunha o arquivo `info.php` publicamente, exibindo a saída completa da função `phpinfo()`. Esta é uma falha classificada como **Crítica** segundo o CVSS, pois vaza metadados sensíveis da infraestrutura interna.

Os dados expostos incluíam:

- Versão do PHP: 5.6.40 (sem suporte desde dezembro de 2018);
- *Hostname* interno do banco de dados e nome do banco;
- Caminho absoluto de arquivos no servidor (*path disclosure*);
- Sistema operacional do container (Debian 11);
- Variáveis de ambiente, potencialmente incluindo credenciais.

A exposição de variáveis de ambiente via `phpinfo()` é particularmente grave em ambientes Docker, onde senhas de banco de dados são frequentemente passadas como variáveis de ambiente no `docker-compose.yml`. O trecho ilustra o padrão de configuração vulnerável:

```
services:
  web:
```

```
    image: php:5.6-apache
    environment:
      - DBHOST=<hostname_interno>
      - DBNAME=<nome_banco>
      - MYSQL_ROOT_PASSWORD=<senha_exposta_via_phpinfo>
```

Listing 2. Padrão de configuração vulnerável identificado

5.2.2. Uso de Software Obsoleto (PHP 5.6 / PHP EoL)

A versão PHP 5.6.40, identificada no servidor de aplicação, atingiu seu *End of Life* (EoL) em dezembro de 2018. Esta versão possui vulnerabilidades críticas sem correção oficial disponível. Os CVEs mais relevantes para este contexto são listados na Tabela 3.

Tabela 3. CVEs identificados aplicáveis ao ambiente avaliado

CVE	CVSS v3	Descrição
CVE-2019-11043	9.8 (Crítico)	RCE via PHP-FPM com Nginx
CVE-2012-1823	7.5 (Alto)	Injeção de argumentos via PHP-CGI
CVE-2024-4577	9.8 (Crítico)	Injeção de argumentos em PHP/CGI

5.2.3. Credenciais Padrão em Equipamento de Borda (MikroTik)

O equipamento de roteamento de borda identificado como ISP-EDGE-01 (MikroTik RouterOS) permitiu autenticação bem-sucedida com o usuário padrão `admin` e senha vazia — configuração de fábrica não alterada. A verificação foi realizada via `curl` e módulo do Hydra

```
curl -s http://<IP_ANONIMIZADO>/webfig/ \
  --user admin: | grep -i "title"
# Retorno: acesso bem-sucedido a interface WebFig
```

Listing 3. Verificação de credenciais padrão no equipamento de borda

Esta falha é classificada como **Crítica** pois um equipamento de borda comprometido permite interceptação de tráfego, manipulação de tabelas de roteamento e, no contexto de CGNAT, alteração dos destinos de exportação de logs — destruindo evidências forenses em investigações judiciais.

5.2.4. Exposição de Banco de Dados sem Proteção de Rede

O servidor ISP-MON-01 expunha o PostgreSQL 13.13 na porta padrão 5432 diretamente à internet, sem restrição de origem por firewall. De forma similar, o host ISP-APP-01 expunha uma instância MariaDB 11.8.3 na porta 3306.

A gravidade desta exposição está na possibilidade de:

1. Ataques de força bruta diretos ao sistema de autenticação do SGBD;

2. Exploração de vulnerabilidades específicas das versões identificadas;
3. Injeção ou deleção em massa de registros de log já processados.

O *banner grabbing* do MariaDB também revelou o sistema operacional subjacente (Debian), aumentando a superfície de ataque ao permitir a busca de CVEs específicos da combinação OS + SGBD.

5.2.5. Protocolos Inseguros — Telnet e FTP

Três hosts identificados apresentavam os protocolos Telnet (porta 23/TCP) e FTP (porta 21/TCP) ativos e acessíveis via rede pública. Estes protocolos transmitem credenciais e comandos em texto claro (*plain text*), tornando-os vulneráveis a ataques de interceptação (*sniffing*) em qualquer ponto da rota de rede.

```
nmap --script ftp-anon -p 21 <IP_ANONIMIZADO>
# PORT   STATE SERVICE
# 21/tcp open  ftp
```

Listing 4. Verificação de FTP anônimo no host ISP-SRV-01

O impacto no contexto de logs forenses é particularmente crítico: um atacante com acesso ao equipamento via Telnet pode manualmente deletar arquivos de log locais antes que eles sejam encaminhados ao servidor de monitoramento centralizado.

5.2.6. Interface de Gerenciamento de Containers Exposta (Portainer)

A interface Portainer — sistema de gerenciamento visual de containers Docker — foi identificada exposta publicamente na porta 443 sem restrição de acesso por IP ou VPN. A API retornou o seguinte JSON sem autenticação prévia:

```
{
  "AuthenticationMethod": 1,
  "RequiredPasswordLength": 12,
  "hideFileUpload": false
}
```

Listing 5. Resposta da API do Portainer sem autenticação

O campo `"hideFileUpload": false` indica que, após uma eventual autenticação (via força bruta ou credencial comprometida), seria possível realizar *Remote Code Execution* (RCE) através do upload de imagens Docker maliciosas ou scripts de inicialização de container, obtendo acesso a todos os serviços gerenciados pela instância Portainer.

5.2.7. Ausência de Criptografia na Camada de Transporte (MySQL)

Durante os testes de acesso ao banco de dados MySQL do servidor, o cliente reportou o erro

```
ERROR 2026 (HY000): SSL is required, but the server
does not support it
```

Listing 6. Erro de conexão SSL no cliente MySQL

Paradoxalmente, embora o servidor exija SSL do cliente, ele mesmo não suporta conexões criptografadas — uma configuração inconsistente que indica erro de *hardening*. Na prática, isso significa que credenciais e dados transitam em texto claro pela rede interna, vulneráveis a ataques de interceptação (*Man-in-the-Middle*).

Adicionalmente, os testes de força bruta com o script `mysql-brute` do Nmap revelaram a ausência de *Rate Limiting*: o servidor aceitou mais de 1.000 tentativas de autenticação por segundo sem bloquear a origem, demonstrando que não há mecanismo de proteção contra ataques automatizados de dicionário.

5.2.8. Enumeração de Arquivos Sensíveis via Fuzzing

A técnica de *fuzzing* com a ferramenta FFUF revelou a existência de arquivos com nomenclatura indicativa de sistemas críticos, embora todos retornassem código HTTP 403 (acesso negado). A existência confirmada de nomes como `replication.php` e arquivos `.sql` indica a presença de rotinas de replicação de banco de dados e possíveis backups acessíveis mediante bypass das ACLs.

5.3. Fase 3: Consolidação e Mapa de Risco

A Tabela 4 sintetiza todas as vulnerabilidades identificadas com sua classificação de severidade segundo o CVSS v3.1:

Tabela 4. Consolidação das vulnerabilidades e classificação de risco

Vulnerabilidade	Severidade	Impacto Principal
Information Disclosure (phpinfo)	Crítica	Vazamento de credenciais e topologia interna
Credenciais padrão (admin/-vazio)	Crítica	Comprometimento total do roteador de borda
Portainer exposto	Crítica	RCE em todos os containers gerenciados
PostgreSQL exposto (5432)	Alta	Acesso direto ao banco de logs
PHP 5.6 EoL	Alta	RCE via CVEs sem correção
Ausência de SSL/TLS no MySQL	Alta	Interceptação de credenciais
Protocolos FTP/Telnet	Alta	Captura de credenciais em texto claro
Winbox/API MikroTik expostos	Alta	Força bruta e escalada de privilégios
RDP exposto (3389)	Média	Vetor de entrada para Ransomware
Headers HTTP ausentes	Média	Clickjacking e MIME sniffing
Ausência de Rate Limiting	Média	Ataques de dicionário não bloqueados

6. Propostas de Mitigação

6.1. Ações Imediatas (Prazo: 72 horas)

1. **Remoção do arquivo info.php**: Excluir imediatamente de todos os ambientes de produção. Implementar regras de WAF para bloquear requisições a arquivos de diagnóstico.
2. **Alteração de credenciais padrão**: Todos os equipamentos MikroTik devem ter as senhas de fábrica substituídas por senhas complexas (mínimo 16 caracteres, seguindo a política de senhas da organização).
3. **Restrição de acesso ao Portainer**: Fechar o acesso externo, permitindo conexão apenas via VPN ou túnel SSH autenticado por chave.

6.2. Ações de Curto Prazo (Prazo: 30 dias)

1. **Migração do PHP**: Atualizar a pilha de aplicação para PHP 8.2 ou superior, eliminando os CVEs críticos identificados.
2. **Firewall de banco de dados**: Configurar regras de *iptables*/UFW para aceitar conexões nas portas 3306 e 5432 exclusivamente a partir do IP do servidor de aplicação correspondente.
3. **Substituição de Telnet/FTP**: Migrar para SSH/SFTP em todos os equipamentos que ainda utilizam protocolos em texto claro.
4. **Implementação de Rate Limiting**: Configurar *fail2ban* ou módulos equivalentes nos servidores de banco de dados e web para bloquear IPs após um número configurável de tentativas falhas.

6.3. Ações de Médio Prazo (Prazo: 90 dias)

1. **Atualização de firmware RouterOS**: Migrar dispositivos MikroTik da série 6.x para a versão estável mais recente da série 7.x, que corrige as vulnerabilidades CVE identificadas.
2. **Segmentação de rede**: Isolar os servidores de banco de dados em VLANs dedicadas, sem rota direta para a internet.
3. **Pipeline centralizado de logs**: Implementar a arquitetura Kafka/OpenSearch para garantir a imutabilidade forense dos logs de eventos de rede.

6.4. O Pipeline de Logs como Mitigação Estratégica

A principal contribuição proposta por este trabalho vai além da correção pontual de vulnerabilidades: a implementação de um pipeline centralizado e imutável de logs representa uma mudança de paradigma na postura de segurança do ISP.

A arquitetura proposta funciona da seguinte forma: os equipamentos MikroTik encaminham logs via Syslog/IPFIX para o Apache Kafka, que os distribui para ingestão no OpenSearch. Esta camada de log centralizado possui duas propriedades fundamentais:

- **Redundância Forense**: Mesmo que um atacante comprometa um roteador via Telnet (Seção 4.2.5) e apague os logs locais, os eventos já terão sido transmitidos em tempo real para o cluster Kafka, preservando a cadeia de custódia forense exigida pelo Marco Civil da Internet [Brasil 2014].
- **Detecção de Intrusão**: Os dashboards do OpenSearch permitem criar alertas automáticos sobre anomalias — como picos de tentativas de login nas portas vulneráveis identificadas (21, 23, 8291, 3389) — possibilitando uma resposta a incidentes muito mais ágil do que a verificação manual em cada equipamento.

7. Discussão

Os resultados demonstram um padrão recorrente na segurança de ISPs regionais brasileiros: a adoção de tecnologias adequadas para o negócio principal (roteamento, CGNAT, hospedagem) sem o correspondente investimento em práticas de segurança da informação. Este fenômeno é descrito na literatura como *security debt* — a acumulação de débitos técnicos relacionados à segurança [Williams et al. 2018].

A coexistência de um servidor com configuração de *hardening* parcial (host ISP-WEB-02, com X-Frame-Options configurado) e equipamentos com credenciais de fábrica no mesmo ambiente demonstra a ausência de uma política de segurança sistematizada. Uma política coerente garantiria que os mesmos padrões de configuração segura fossem aplicados de forma homogênea em toda a infraestrutura.

A exposição do protocolo RTSP em uma câmera IP integrada à infraestrutura de rede do provedor representa um vetor de ataque frequentemente negligenciado: dispositivos IoT (*Internet of Things*) integrados a redes corporativas sem segmentação adequada. Pesquisas recentes indicam que câmeras IP e outros dispositivos IoT são frequentemente utilizados como ponto de pivotamento em ataques à infraestrutura principal [Kaspersky Lab 2023].

Por fim, é importante destacar que a conformidade com a LGPD (Lei Geral de Proteção de Dados) e com o Marco Civil da Internet não é apenas uma questão técnica,

mas também legal. Um ISP que não adota medidas técnicas adequadas de proteção de dados pode ser responsabilizado por vazamentos, com penalidades que chegam a 2% do faturamento anual, limitado a R$ 50 milhões por infração [Brasil 2018].

8. Conclusão

Este trabalho documentou e analisou um conjunto de vulnerabilidades críticas identificadas na infraestrutura de um ISP regional por meio de um processo estruturado de teste de penetração. Os achados confirmam a hipótese inicial: a segurança periférica baseada exclusivamente em firewall de borda é insuficiente quando combinada com configurações inadequadas internas — o que a literatura chama de falha na *Defesa em Profundidade*.

As vulnerabilidades mais graves identificadas — credenciais padrão em roteadores de borda, exposição de banco de dados e Information Disclosure via phpinfo — representam vetores de comprometimento total da infraestrutura que poderiam ser explorados por atacantes com nível de habilidade intermediário, utilizando ferramentas de acesso público.

A proposta do pipeline Kafka/OpenSearch demonstra que a resposta adequada a estas ameaças não é apenas reativa (corrigir vulnerabilidades individualmente), mas estratégica: construir uma camada de observabilidade que torne a infraestrutura resiliente mesmo em cenários de comprometimento parcial, garantindo a integridade forense dos logs e a capacidade de detecção precoce de intrusões.

Como trabalhos futuros, sugere-se: (i) a implementação e avaliação da eficácia do pipeline proposto em ambiente de produção; (ii) o desenvolvimento de um módulo de correlação de eventos baseado em aprendizado de máquina para detecção de anomalias nos logs CGNAT; e (iii) a condução de um estudo comparativo com outros ISPs regionais para validar a generalidade dos achados.

Referências

Brasil (2014). Lei n. 12.965, de 23 de abril de 2014: Marco civil da internet. Diário Oficial da União. Brasília, DF.

Brasil (2018). Lei n. 13.709, de 14 de agosto de 2018: Lei geral de proteção de dados (lgpd). Diário Oficial da União. Brasília, DF.

CERT.br (2023). Estatísticas de incidentes reportados ao cert.br. `https://www.cert.br/stats/`. Acesso em: mai. 2026.

EC-Council (2021). *Certified Ethical Hacker (CEH) v11: Complete Training Guide*. EC-Council Press, Albuquerque, NM.

Fernandes, H. L. e Silva, C. A. d. (2025). Investigação de vulnerabilidades em aplicações web utilizadas por empresas de leilões online e instituições de ensino a distância (EAD). Trabalho de Conclusão de Curso, FACOM–UFMS. Disponível em: `https://repositorio.ufms.br/retrieve/535e2267-50c6-4bfc-ae8e-5af6a30658e5/24719.pdf`.

Fernandes, J. V. d. A. (2025). Segurança computacional: Um estudo com foco no metasploit framework. Trabalho de Conclusão de Curso, FACOM–UFMS. Disponível em: `https://repositorio.ufms.br/retrieve/7036314f-aed5-4afa-92c1-f514493cf221/31705.pdf`.

FIRST.org (2019). Common vulnerability scoring system v3.1: Specification document. https://www.first.org/cvss/specification-document. Acesso em: mai. 2026.

Kaspersky Lab (2023). Iot threat report 2023. https://securelist.com/iot-threat-report-2023. Acesso em: mai. 2026.

Lyon, G. (2023). Nmap network scanning: The official nmap project guide to network discovery and security scanning. https://nmap.org/book/. Acesso em: mai. 2026.

Matherly, J. (2022). Shodan: The search engine for internet-connected devices. https://www.shodan.io. Acesso em: mai. 2026.

MikroTik (2024). Routeros documentation. https://help.mikrotik.com/docs/display/ROS/RouterOS. Acesso em: 04 mar. 2026.

NIST — National Institute of Standards and Technology (2018). Cve-2018-14847: Mikrotik routeros vulnerability. https://nvd.nist.gov/vuln/detail/CVE-2018-14847. Acesso em: 04 mar. 2026.

OWASP Foundation (2021a). Database security cheat sheet. https://cheatsheetseries.owasp.org/cheatsheets/Database_Security_Cheat_Sheet.html. Acesso em: mai. 2026.

OWASP Foundation (2021b). Owasp top ten 2021. https://owasp.org/Top10/. Acesso em: mai. 2026.

PTES Technical Guidelines (2012). Penetration testing execution standard. http://www.pentest-standard.org. Acesso em: mai. 2026.

Stallings, W. e Brown, L. (2018). *Computer Security: Principles and Practice*. Pearson, Hoboken, NJ, 4 edition.

Williams, L., Meneely, A., e Shipley, G. (2018). Protection poker: The new software security game. *IEEE Security & Privacy*, 16(3):14–21.