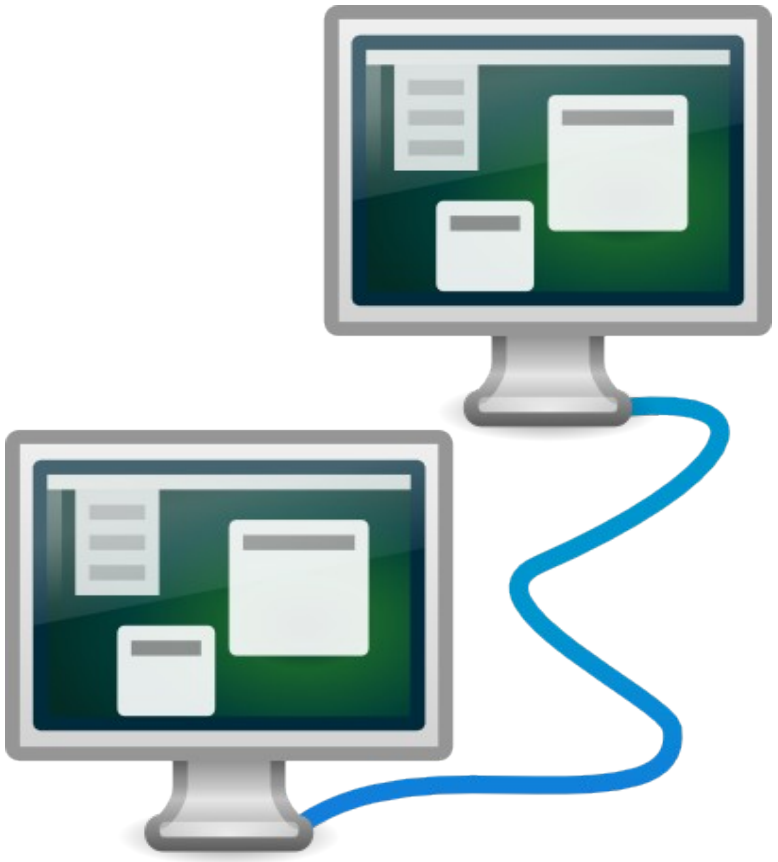
www.4LINUX.com.br

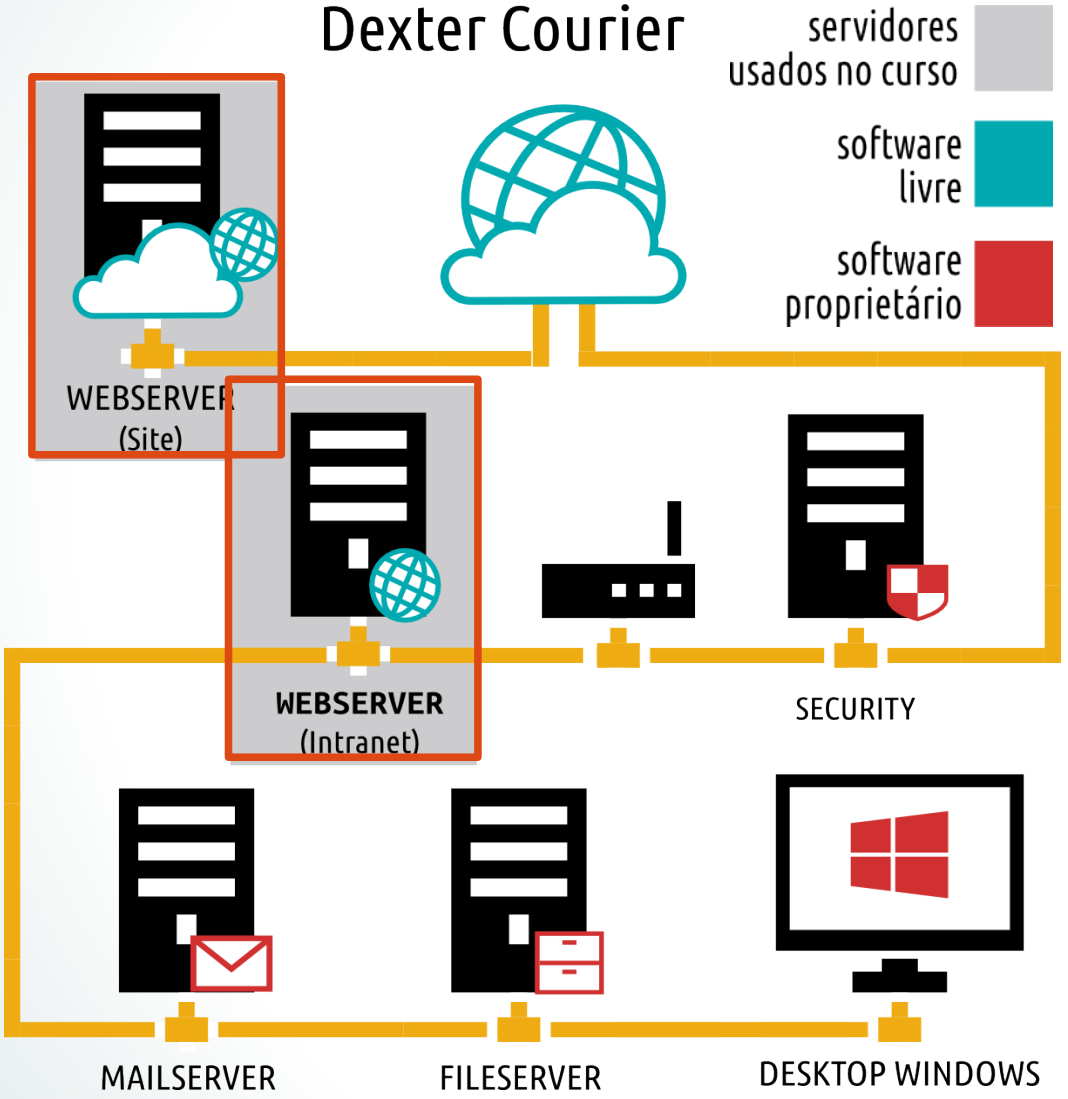
Versão 2.0

Só na 4Linux você aprende MUITO MAIS!

Servidor SSH



IT Experience



Nesta Aula:

- Usaremos os dois Servidores da Dexter:
- WebServerCloud
- WebServerInterno

Objetivos da Aula

- Realizar acesso e cópias através do SSH;
- Ajustar configurações do servidor SSH;
- Configurar acesso com uso de chaves entre os servidores WebServerInterno e WebServerCloud;
- Criar tunelamento SSH.

Serviço SSH

- O Serviço SSH é usado para realizar Acesso Remoto de forma Segura. Ele oferece as seguintes proteções:
  - Após a primeira conexão ele armazena a identidade do Servidor (know_hosts) para garantir que você sempre irá acessar o servidor correto. Caso a identidade seja alterada, ele irá te alertar;
  - O cliente transmite as informações de autenticação usando criptografia forte de 128bits;
  - Todo os dados recebidos e enviados usa uma criptografia de 128bits tornando praticamente impossível decifrar os dados;
  - O cliente pode enviar aplicações X11 de forma segura;
  - Como SSH criptografa tudo, ele pode servir de tunelamento para outros protocolos inseguros (Tunelamento).

Porque SSH?

- Existem uma variedade de ferramentas que podem ser usadas para romper ou interceptar dados de uma comunicação com o objetivo de conseguir acesso a um sistema, como por exemplo, usar um sniffers para capturar dados que estão trafegando na rede.

Com o SSH essa ameaça é quase nula, isso porque o cliente e o servidor SSH usam assinaturas digitais para verificar a sua identidade. Além disso, toda a comunicação entre eles é criptografada. As tentativas para falsificar a identidade de cada lado de uma comunicação não funciona, já que cada pacote é criptografado utilizando uma chave conhecida apenas pelo cliente e o servidor.

Conexão SSH


Solicitação de Conexão: Porta 22/tcp
Confirmação de Identidade
(know_hosts)
Solicitação de Autenticação
Fornecimento de Senha ou Chave
Liberação do Shell
bash
Troca de Dados (Criptografada)
DESKTOP LINUX
WebServerCloud
200.100.1.X

SSH & Cloud

Importante

É praticamente impossível falar de Cloud sem SSH.

Com o avanço do Mercado de Cloud o SSH passou a ser uma ferramenta vital para o SysAdmin quando o assunto é administrar servidores remotos. É exatamente por isso que é extremamente importante configurar esse Serviço de forma correta visando sempre a segurança do acesso ao Sistema.

Acesso Remoto

- Verificando o Servidor SSH:

```
1# dpkg -l | grep openssh
2# netstat -ntlp | grep 22
3# /etc/init.d/ssh status
4# /etc/init.d/ssh start
5# /etc/init.d/ssh status
6# netstat -ntlp | grep 22
```

Para usar o SSH é necessário ter o pacote instalado, tanto o pacote do Servidor, quanto o pacote do Cliente.

A porta padrão do SSH é a 22, se o Serviço está ativo essa porta é liberada para aceitar conexões SSH no Servidor.

Iremos ativar o Serviço SSH no WebServerCloud para permitir conexão SSH através do WebServerInterno.

Acesso Remoto

- Acessando o WebServerCloud:

```
1# rm /root/.ssh/know_hosts
2# ssh -l root 200.100.1.X
```

ou

```
3# ssh root@200.100.1.X
4# hostname
5# exit
6# cd .ssh
7# ls
8# cat know_hosts
```

O comando SSH precisa receber basicamente duas informações:

1) Usuário do Servidor Remoto;

2) IP do Servidor Remoto;

No primeiro acesso será solicitado que você aceite a chave do servidor que será armazenada no arquivo **know_hosts**.

Caso seja omitido o Usuário que irá realizar a autenticação, o ssh irá usar o nome do usuário logado na máquina cliente.

```
# ssh 200.100.1.X
```

Você também poderá especificar uma porta específica caso não seja a porta padrão do SSH (22):

```
# ssh root@200.100.1.X -p 2222
```

Acesso Remoto

- Executando Comandos Remotamente:

```
1# ssh root@webservercloud free -m

2# ssh root@200.100.1.X hostname
```

Lembre-se

Na aula de redes configuramos no /etc/hosts a resolução de nomes dos Servidores da Dexter portanto é possível usar o nome ao invés de IP.

Ao invés de se conectar no Servidor Remoto para receber o bash e executar comandos, também é possível executar comando remotamente, bastando informar na sintaxe do comando ssh:

```
# ssh root@200.100.1.X free -m
```

Realizando Cópias Remotas

- Copiando Arquivos do WebServerCloud para o WebServerInterno:

```
1# scp root@webservercloud:/etc/hostname  /tmp/

2# cat /tmp/hostname

3# scp -r root@webservercloud:/etc   /tmp/

4# scp -P 22 -r root@webservercloud:/home /tmp/

5# ls /tmp/home
```

Opções do SCP

-r → Copiar recursivamente. Usado para enviar ou copiar diretórios completos.

-P → Usado para especificar porta, diferente do ssh, o -P precisa ser obrigatoriamente MAÍUSCULO.

Realizando Cópias Remotas

- Enviando Arquivos do WebServerInterno para o WebServerCloud:

```
1# scp /etc/hostname root@webservercloud:/opt/
2# ssh root@webservercloud ls /opt
3# scp -r  /etc  root@webservercloud:/opt
4# scp -P 22 -r /home root@webservercloud:/opt
5# ssh root@webservercloud ls /opt
```

Configurando o SSH

- O Serviço SSH possui 2 arquivos de Configuração:
  - **sshd_config** → Configurações do Servidor SSH
  - **ssh_config** → Configurações do Cliente SSH

Servidor → Máquina que recebe um acesso Remoto

Cliente → Máquina que realiza um acesso Remoto

Todos os Servidores podem ser Cliente e Servidor!

SSHD_CONFIG

- Vamos realizar a configuração do Servidor SSH no WebServerCloud:

```
1# vim /etc/ssh/sshd_config
 5 Port 22                          (Alterar para 2222)
 8 #ListenAddress 0.0.0.0           (Limitar 200.100.1.X)
 9 Protocol 2
27 PermitRootLogin yes              (Alterar para “no”)
67 TCPKeepAlive yes                 (Manter descomentada)
68 ServerAliveInterval 60           (Adicionar)
72 Banner /etc/issue.net            (Manter descomentada)
```

Salve o Arquivo e Reinicie o Serviço para atualizar as configurações!

Acesso Remoto Seguro

- Acesse novamente o Servidor WebServerCloud após as novas configurações:

```
1# ssh root@webservercloud
```

Como foi alterado a Porta Padrão e Bloqueado o Acesso do Root como medidas de segurança, não será possível realizar acesso da maneira anterior!

```
2# ssh suporte@webservercloud -p 2222
3$ su -
4# whoami
```

Acesso Remoto sem Senha

- Uma possibilidade que temos com o SSH é a autenticação por chaves, que pode ter ou não uma senha diferente da senha do usuário, que chamamos na verdade de passphrase (Frase Chave).
- Vamos agora garantir que nosso Servidor WebServerInterno consiga acessar o Servidor WebServerCloud sem a necessidade de senha.

Isso será útil mais pra frente no curso para rotinas de backup do Servidor em Cloud para a Rede Local da Dexter.

Laboratório Dexter

- Gere uma chave sem senha no WebServerInterno:

```
1# ssh-keygen
```

```
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):   (Apenas Digite ENTER)
Enter passphrase (empty for no passphrase):                (Apenas Digite ENTER)
Enter same passphrase again:                               (Apenas Digite ENTER)
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
bb:9e:4a:a8:e0:0f:2b:4f:89:12:05:43:77:bd:52:0f root@webserverinterno.dexter.com.br
```

```
2# cd /root/.ssh
3# ls
```

id_rsa → Chave Privada. Não pode ser Compartilhada!

id_rsa.pub → Chave Pública que deve ser enviada ao servidor que deseja acessar sem senha.

Laboratório Dexter

- Envie a chave pública para o WebServerCloud:

```
1# ssh-copy-id suporte@webservercloud
2# ssh suporte@webservercloud
3# cd /home/suporte/.ssh
4# ls
```

authorized_keys → Arquivo que armazena as chaves públicas de todas as máquinas que possuem autorização de se conectar nesse Servidor com palavra-chave ou sem senha como é o nosso caso.

Tunelamento SSH

- O Servidor SSH tem um recurso muito interessante que é capacidade de criar túneis criptografados para que o dado seja trafegado.
- Vamos fazer um exemplo, da máquina física do laboratório iremos acessar o Sistema da Dexter (WebServerInterno) através de um tunelamento SSH dessa forma, embora o Site ainda não esteja com SSL (https) conseguiremos trafegar de forma criptografada.

Laboratório Dexter

- Crie o Túnel SSH na Máquina Física da Sala:

```
1# ssh -f -N -L12345:192.168.200.X:80 root@192.168.200.X
```

-f e **-N** → Usamos essa opção para o SSH devolver o shell local já que nosso interesse é apenas criar o túnel.

-L → Usado para criar o túnel onde você precisa especificar a porta de origem e destino.

- Acesse o Navegador da Máquina Física:

http://127.0.0.1:12345

Limitando Acesso ao SSH

- Alterando a opção “PermitRootLogin yes” para “no” no arquivo do Servidor WebServerCloud nós limitados o Root de poder logar via SSH, porém todos os demais usuários do Servidor ainda possuem acesso a realizar uma conexão SSH. Vamos limitar esse acesso apenas ao usuário suporte.

```
1# vim /etc/ssh/sshd_config
   AllowUsers suporte
2# /etc/init.d/ssh restart
```

Vá no WebServerInterno e teste o acesso:

```
3# ssh -l helpdesk webservercloud
4# ssh -l suporte webservercloud
```

Pergunta LPI

Qual arquivo de configuração você precisará editar para alterar as opções padrões do cliente SSH?

A. /etc/ssh/sshd_config

B. /etc/ssh/ssh_client

C. /etc/ssh/client

D. /etc/ssh/ssh

E. /etc/ssh/ssh_config

Pergunta LPI

Qual arquivo de configuração você precisará editar para alterar as opções padrões do cliente SSH?

A. /etc/ssh/sshd_config

B. /etc/ssh/ssh_client

C. /etc/ssh/client

D. /etc/ssh/ssh

E. /etc/ssh/ssh_config

Resposta: E


4LINUX
FREE SOFTWARE SOLUTIONS