



**INSTITUTO FEDERAL**

Norte de Minas Gerais

Campus Januária

# Admin. Serviços de Redes

## - *Tunelamento* -



# SSH

- O **SSH** não é útil apenas para acessar hosts remotos...
- Outras funções do SSH:
  - Visualização remota de aplicativos gráficos:  
`p.ex. ssh -C -X admin@192.168.100.1`
  - Transferência segura de arquivos (SCP).
  - **Comunicação criptográfica para outros protocolos.**
  - **Túneis seguros (local e remoto).**
  - **Proxy Socks.**

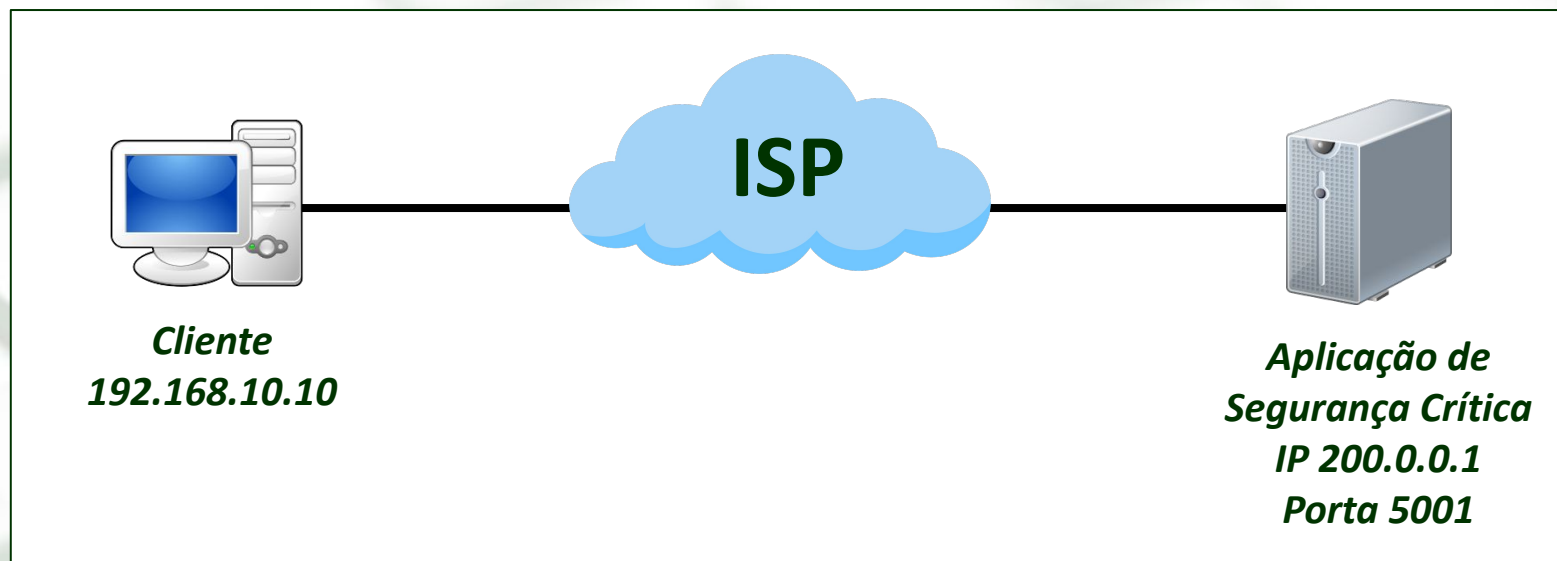
**Um verdadeiro canivete suíço!**





# Tunelamento SSH

- Considere o seguinte cenário...

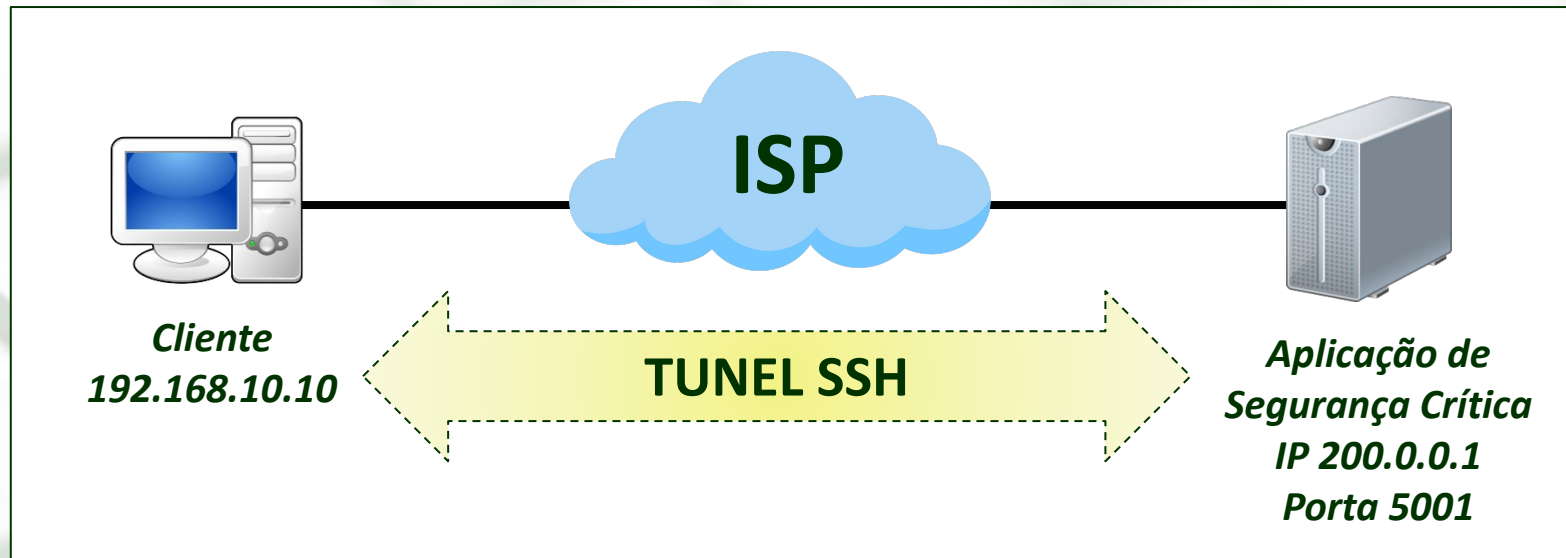


Como **garantir** que as informações desta aplicação irão trafegar de **forma sigilosa** na rede pública **Internet**?



# Tunelamento SSH

- Considere o seguinte cenário...



Podemos usar o **protocolo SSH** para criar um **túnel criptografado** entre os processos.



# Túnel Local SSH

- Criando um Túnel Seguro com Redirecionamento de Porta

```
# ssh -NL PRT_L:IP_R:PRT_R user@SRV_SSH
```

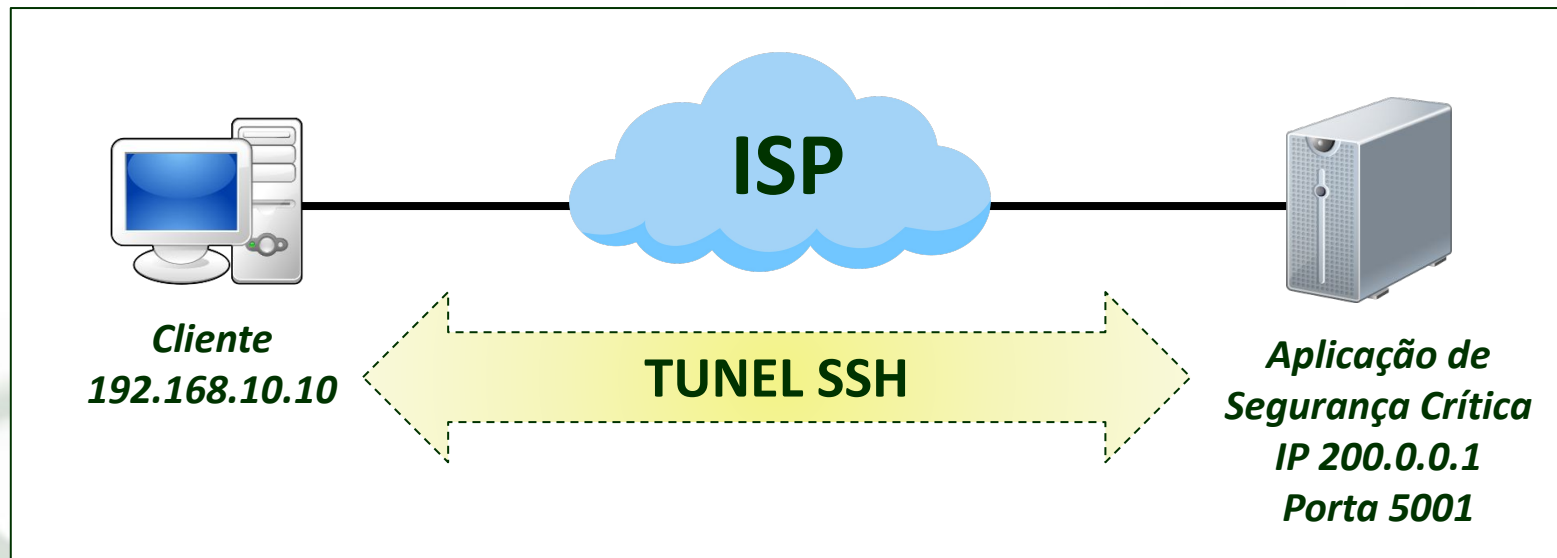
Onde...

-N: Será aberto um túnel SSH, sem a necessidade de abrir o terminal remoto.  
-L: O túnel será criado a partir de uma porta local para um socket remoto.  
PRT\_L: Número da porta local que será usada para criar o túnel.  
IP\_R: Endereço IP da máquina remota que receberá os pacotes.  
PRT\_R: Número da porta remota que receberá os pacotes.  
user: Usuário para autenticação SSH.  
SRV\_SSH: End. IP do Servidor para autenticação SSH (Não precisa ser, necessariamente, o mesmo IP\_R).





# Laboratório 10-1



## Cliente da Aplicação && SSH

```
# tmux [multiplex. do terminal]
# CTRL+B C [Cria nova janela]
# ssh -NL 8001:200.0.0.1:5001
#   user@200.0.0.1
# CTRL+B N [alternar janelas]
# nc 127.0.0.1 8001 [conecta-se à
#   porta 8001 localhost]
```

## Servidor da Aplicação && SSH

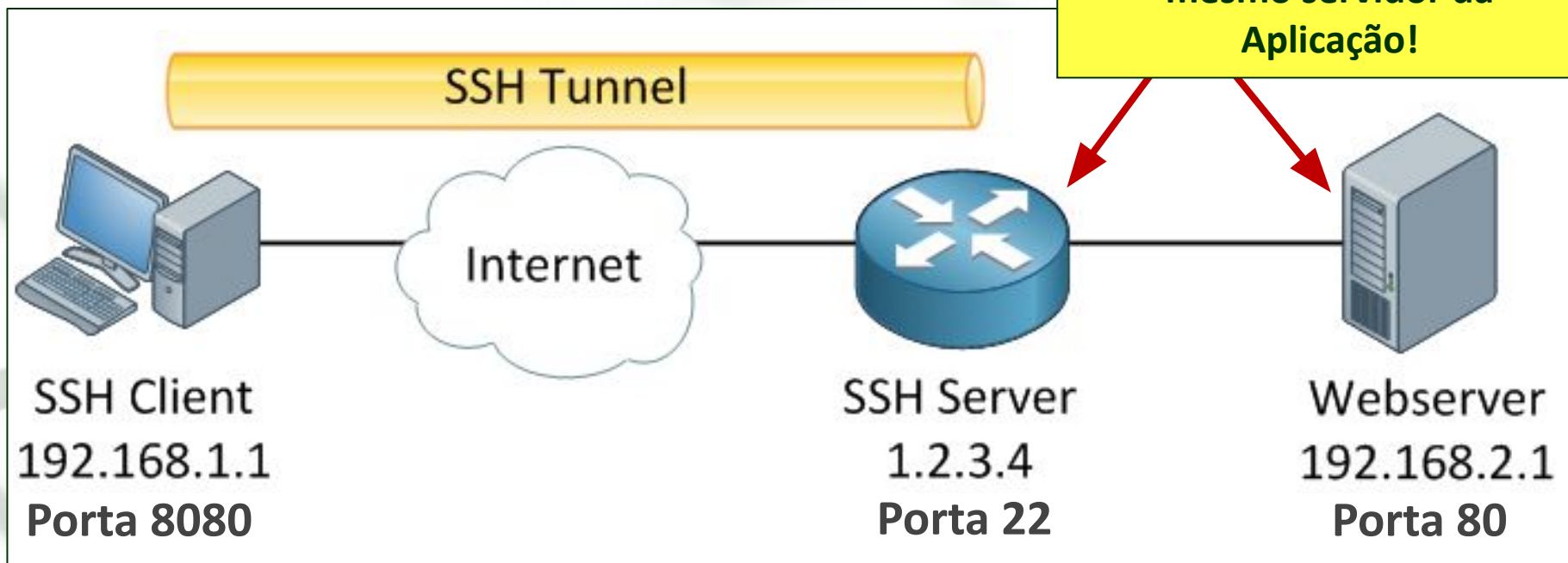
```
# /etc/init.d/ssh start
# nc -lp 5001 [simula o app]
```

*Faça a captura do tráfego no ISP e veja no Wireshark o conteúdo trocado nas mensagens da aplicação.*



# Túnel com Redirecionamento

## ■ Túnel com Redirecionamento de Porta

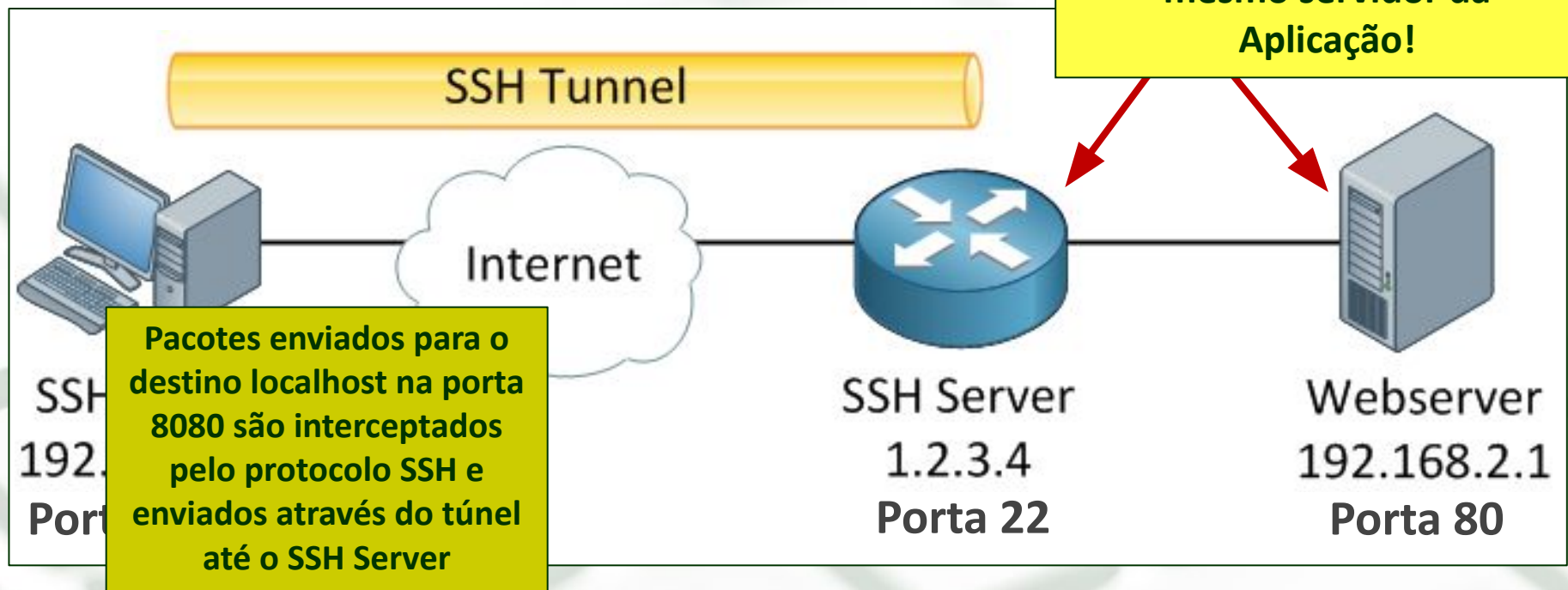


```
# ssh -NL 8080:192.168.2.1:80 user@1.2.3.4
```



# Túnel com Redirecionamento

## ■ Túnel com Redirecionamento de Porta



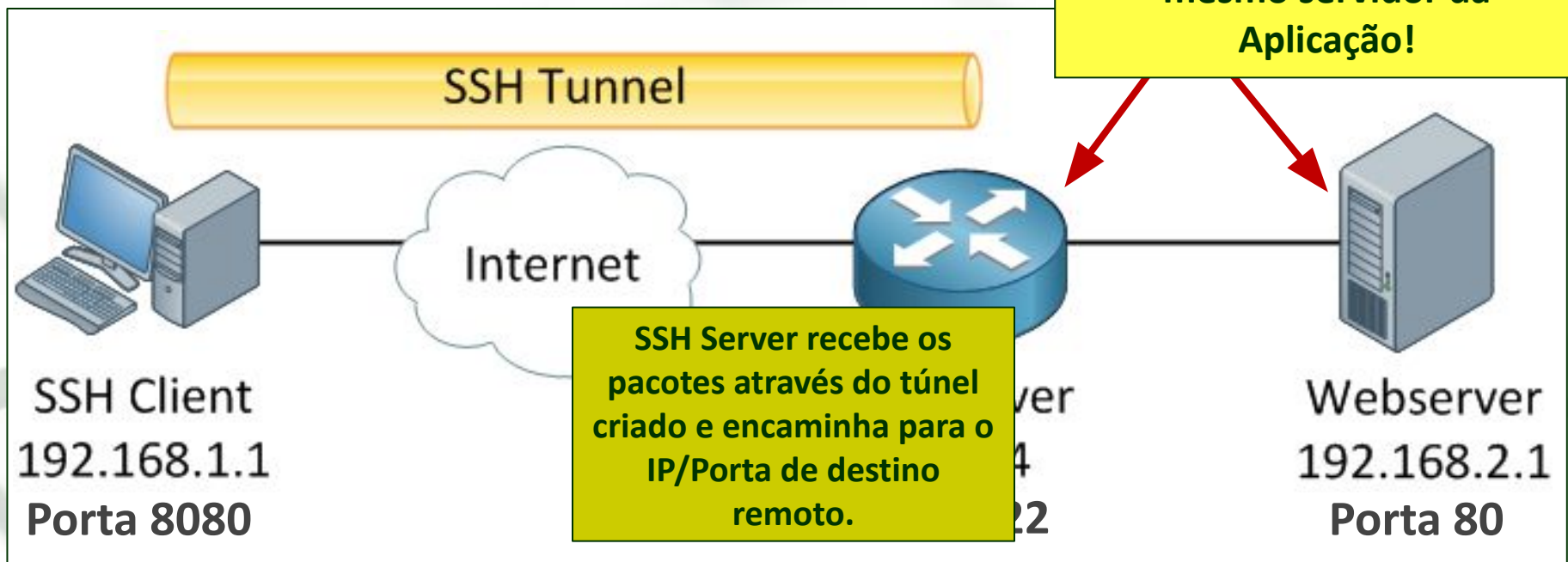
```
# ssh -NL 8080:192.168.2.1:80 user@1.2.3.4
```





# Túnel com Redirecionamento

## ■ Túnel com Redirecionamento de Porta

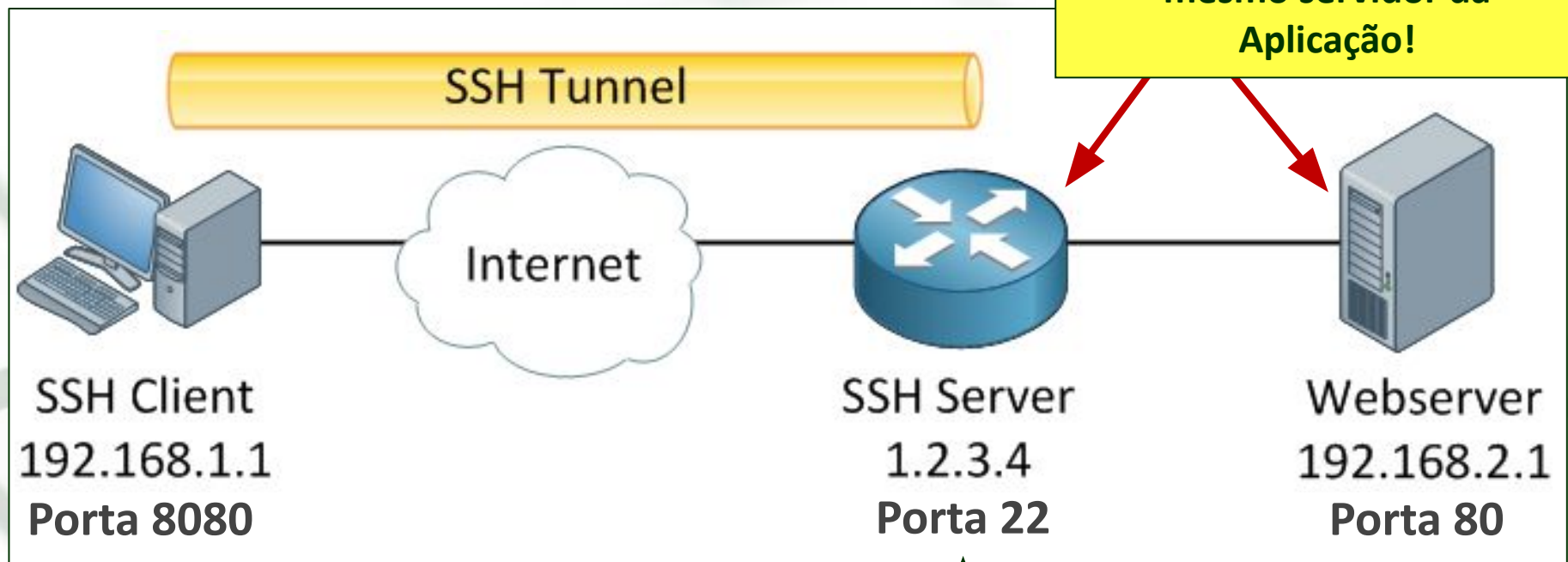


```
# ssh -NL 8080:192.168.2.1:80 user@1.2.3.4
```



# Túnel com Redirecionamento

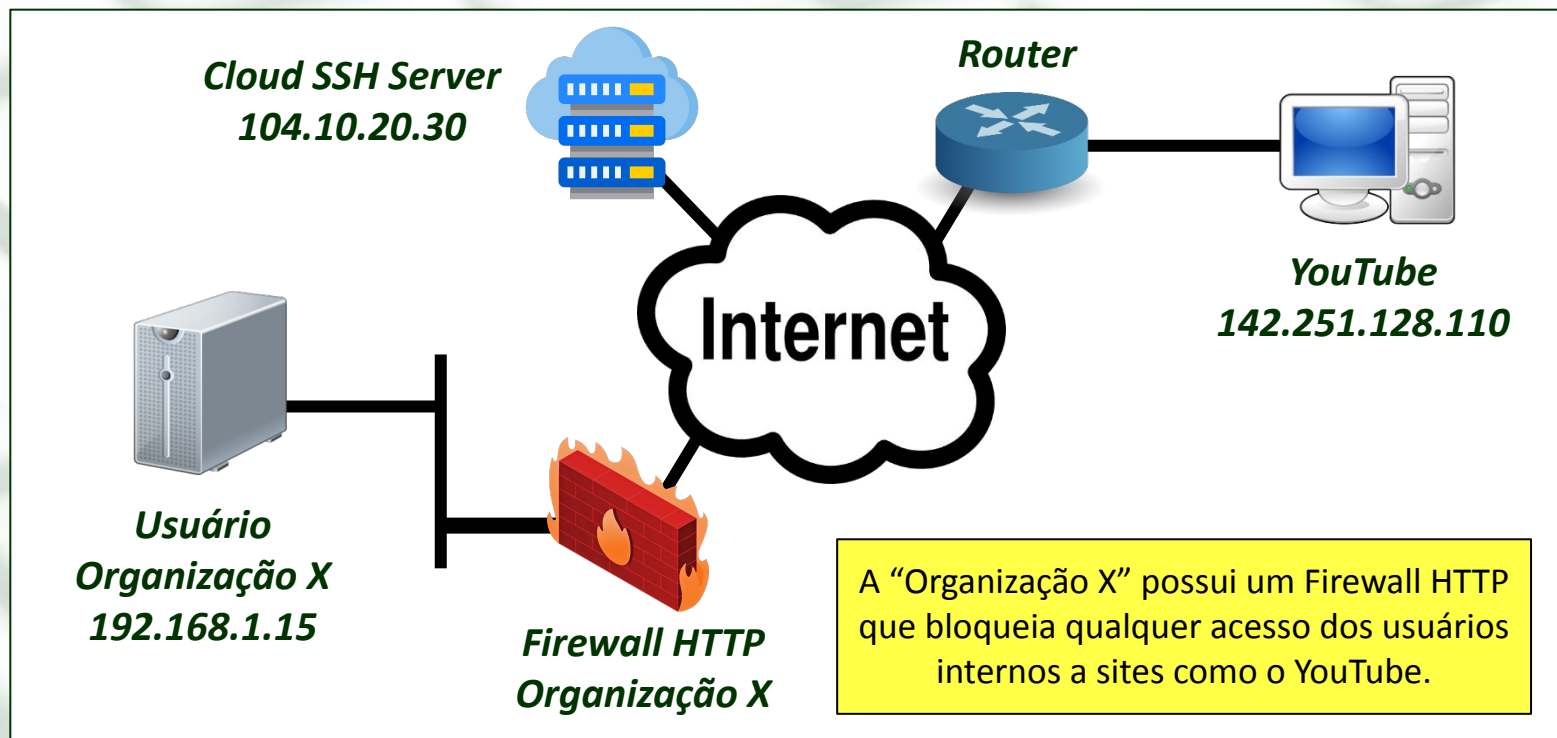
## ■ Túnel com Redirecionamento de Porta





## Laboratório 10-2

- Implemente um túnel SSH para “furar” o Firewall WEB implementado pela Organização X.



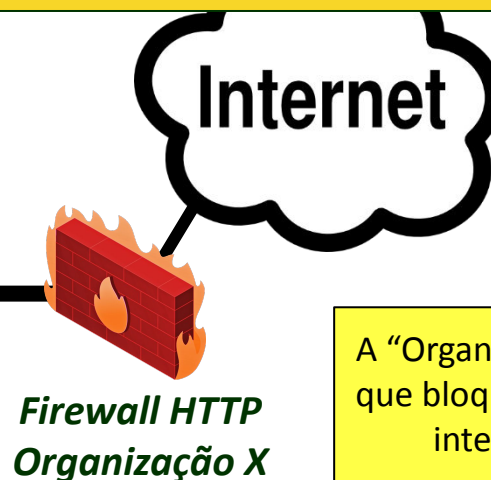
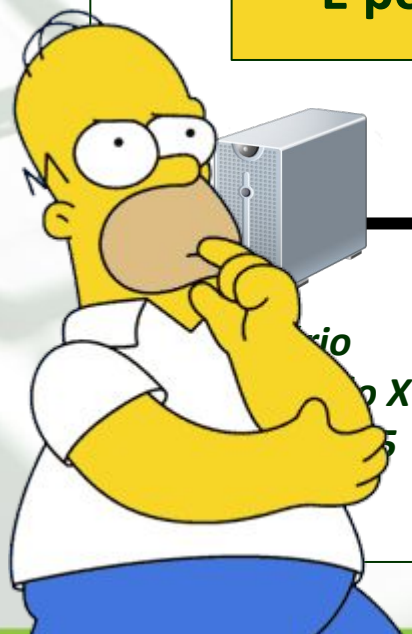


## Laboratório 10-2

- Implemente um túnel SSH para “furar” o Firewall WEB implementado

Mas esse túnel sempre conduz para um local único e muito específico (p.ex. o YouTube neste Lab)...

É possível fazer algo similar mas para qualquer destino?



*YouTube*  
**142.251.128.110**

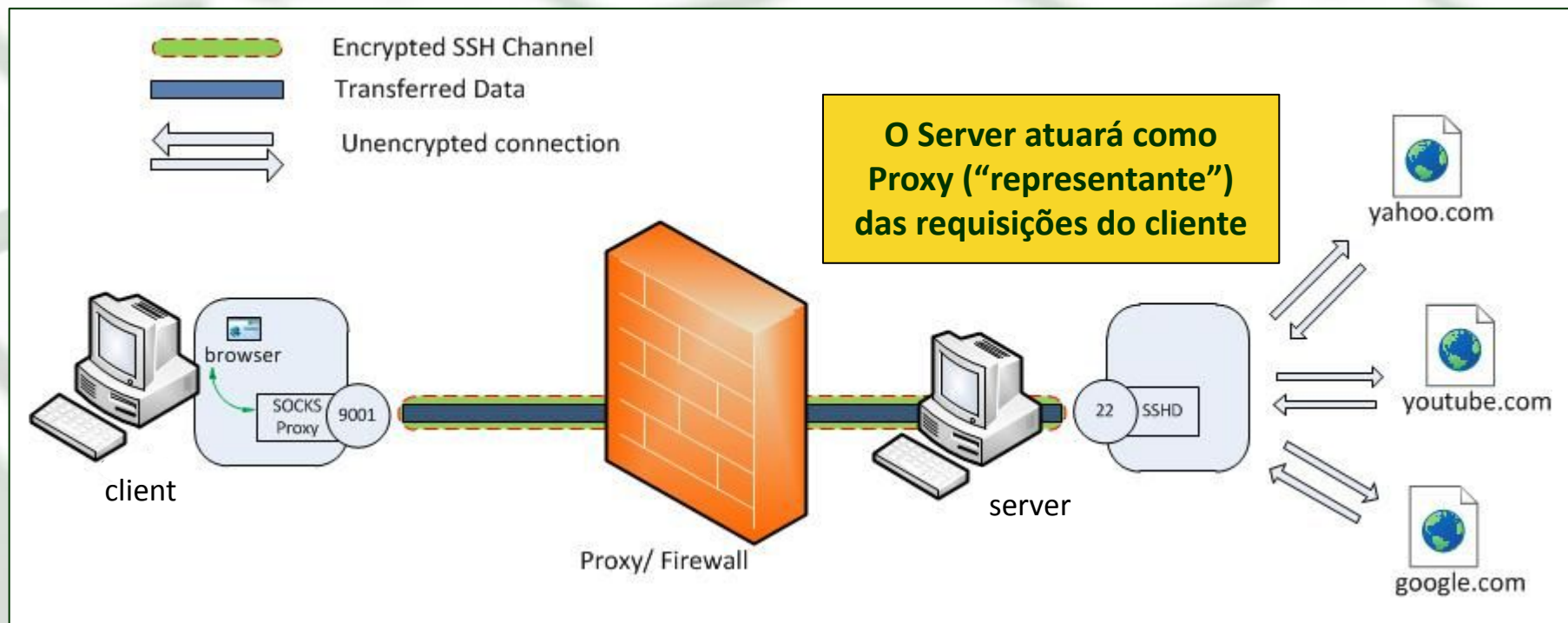
A “Organização X” possui um Firewall HTTP que bloqueia qualquer acesso dos usuários internos a sites como o YouTube.





# Proxy SSH

- Neste caso, precisamos criar um Proxy SSH.
- Também conhecido como Proxy Socks







# Proxy SSH

## ■ Criando um Proxy SSH

```
# ssh -ND PRT_L user@SRV_SSH
```

onde...

-N: Será aberto um túnel SSH, sem a necessidade de abrir o terminal remoto.  
-D: (*Dynamic*) O túnel criado será para destinos diversos.  
PRT\_L: Número da porta local que será usada para criar o túnel.  
user: Usuário para autenticação SSH.  
SRV\_SSH: End. IP do Servidor que atuará como Proxy.



# Proxy SSH

Após a criação do proxy, é necessário configurar o seu navegador para utilizá-lo...

# ssh -l

onde...

-N: Será aberto em modo não interativo  
-D: (Dynamic) Allocated a dynamic port  
PRT\_L: Número da porta local  
user: Usuário  
SRV\_SSH: Endereço do servidor SSH

Configuração de conexão

Configuração do proxy de acesso à internet

☐ Sem proxy

☐ Detectar automaticamente as configurações de proxy desta rede

☐ Usar as configurações de proxy do sistema

☒ Configuração manual de proxy

Proxy HTTP  Porta

☐ Usar este proxy também para HTTPS

Proxy HTTPS  Porta

Domínio SOCKS  Porta

☐ SOCKS v4 ☒ SOCKS v5



## Laboratório 10-3

- Configure uma Instância EC2 AWS para servir como Proxy Socks para acesso da sua Máquina à Internet.

Acesse

<https://ipinfo.io>

antes...

aws



...e depois de  
configurar o Proxy.  
Analise as diferenças



# Tunelamento Remoto

- **Túnel SSH Remoto ou Remote Forwarding**
- Além da criação de Túneis Locais e Proxy, o SSH permite outra importante funcionalidade: **Encaminhamento Remoto**.
- Com essa técnica é possível, a partir de uma **rede privada**, se conectar a um **servidor SSH público** e utilizar este servidor para encaminhar os pacotes **para dentro da sua rede privada**, mesmo atravessando NATs.
- **Ou seja, podemos nos conectar de qualquer lugar do mundo diretamente ao PC da sua casa, por exemplo!**



# Tunelamento Remoto

- Antes, precisamos habilitar o recurso de encaminhamento remoto no servidor SSH...

```
# nano /etc/ssh/sshd_config
```

```
# AllowTcpForwarding yes
```

```
# GatewayPorts no
```



```
AllowTcpForwarding yes
```

```
GatewayPorts yes
```





# Tunelamento Remoto

- Criando um Túnel para Encaminhamento Remoto.

```
# ssh -NR PRT_R:IP_L:PRT_L user@SRV_SSH
```

Onde...

-N: Será aberto um túnel SSH, sem a necessidade de abrir o terminal remoto.  
-R: O túnel será criado a partir de porta remota para um socket local.  
PRT\_R: Número da porta remota que será usada para criar o encaminhamento.  
IP\_L: Endereço IP da máquina local que receberá os pacotes.  
PRT\_L: Número da porta local que receberá os pacotes.  
user: Usuário para autenticação SSH do lado do servidor.  
SRV\_SSH: End. IP do Servidor para autenticação SSH.



# Alguns Serviços na Web

**localhost.run**

<3 local dev

connect a tunnel to your web application running on port 8080 now with

**Telebit**



LocaltoNet

CRPROXY



`ssh srv.us :`  
expose local HTTP services online

`# ssh -NR 1:localhost:80 srv.us`



## Laboratório 10-4

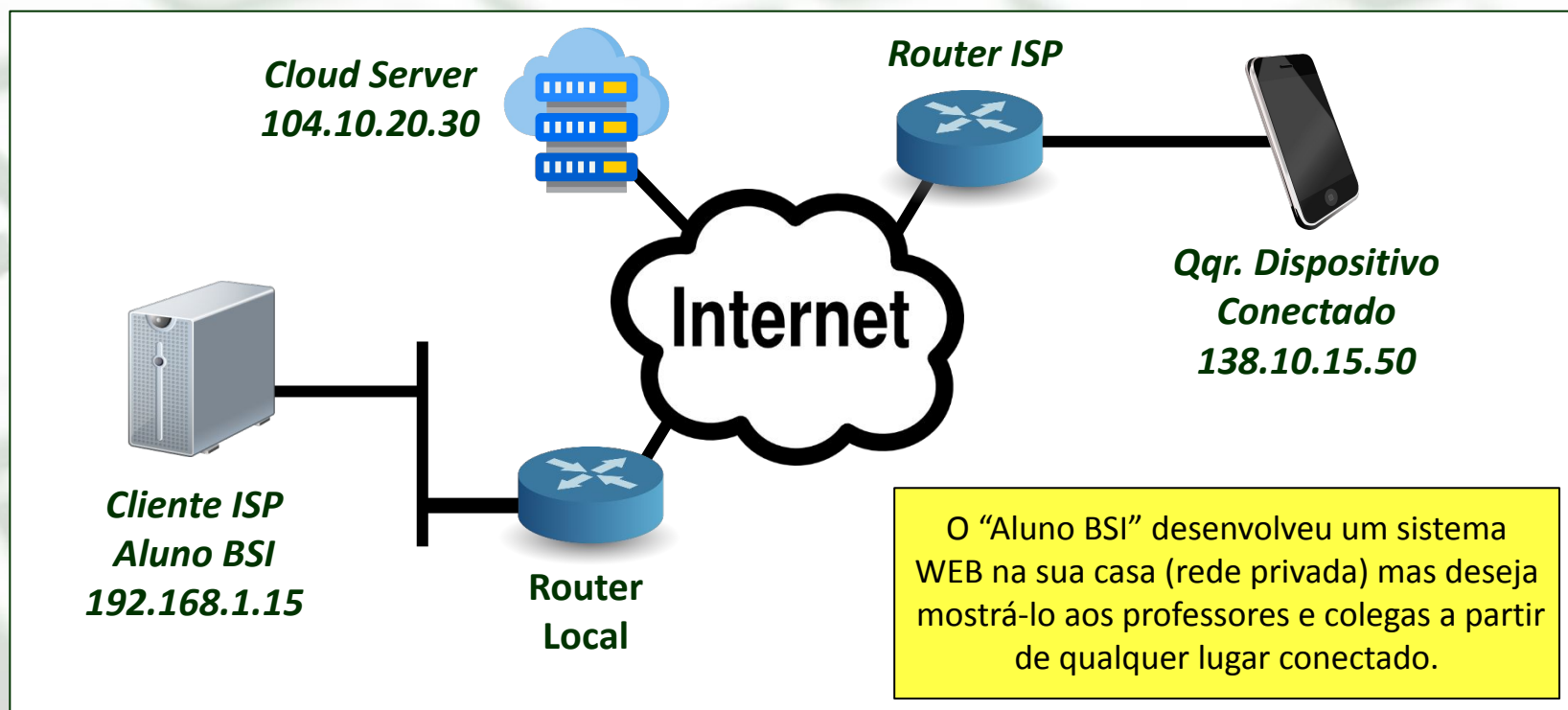
- Configure uma Instância EC2 AWS para conseguir acessar remotamente uma aplicação (p.ex.. nc) rodando em seu PC na Rede Privada.





## Laboratório 10-5

- Implemente no Kathará um túnel de redirecionamento remoto conforme o cenário abaixo.





# Seminário Individual

## **IPTables**

*O que é e para que serve?*

*Sintaxe Básica IPtables*

*Chains IPtables*







# Referências

- **Guia Foca GNU/Linux.**

Disponível em <http://www.guiafoca.org/>

- **MORIMOTO, Carlos E; Servidores Linux – Guia Prático.**