



INSTITUTO FEDERAL

Norte de Minas Gerais

Campus Januária

Admin. Serviços de Redes

- *Acesso Remoto* -



Acesso Remoto





Telnet

TELNET



Telnet

- **Telnet** é um dos protocolos padrões da Internet para acesso remoto a hosts (p.ex. servidores).
- Acesso remoto permite que um usuário efetue comandos e altere configurações em *hosts* distantes, através da visualização do terminal remoto em sua própria estação.
- Entretanto, o TELNET não utiliza criptografia na comunicação entre a máquina local e remota, o que pode causar um grave problema de segurança.



Telnet

- Por padrão, o serviço Telnet baseia-se em conexões TCP através da Porta 23... Ou outra porta definida em:

```
# /etc/services
```

- Devido às suas limitações de segurança, é usado **somente em casos muito específicos.**





Telnet

■ Instalação

```
# apt-get install telnetd
```

■ Configuração

```
# /etc/inetd.conf
```

■ Ativação do Servidor

```
# service openbsd-inetd start
```



Segurança de Ambiente

ATENÇÃO

***Por questões de segurança
NUNCA
faça um acesso remoto
diretamente para o usuário root.***



Gestão de Usuários

- Crie um novo usuário no Server:

```
# adduser nome_usuario
```

- Conceder permissões de root ao usuário (se necessário)

```
# usermod -aG sudo nome_usuario
```

- Trocar para novo usuário (*Switch User*)

```
# su nome_usuario
```

- Trocar para usuário root

```
# su -
```



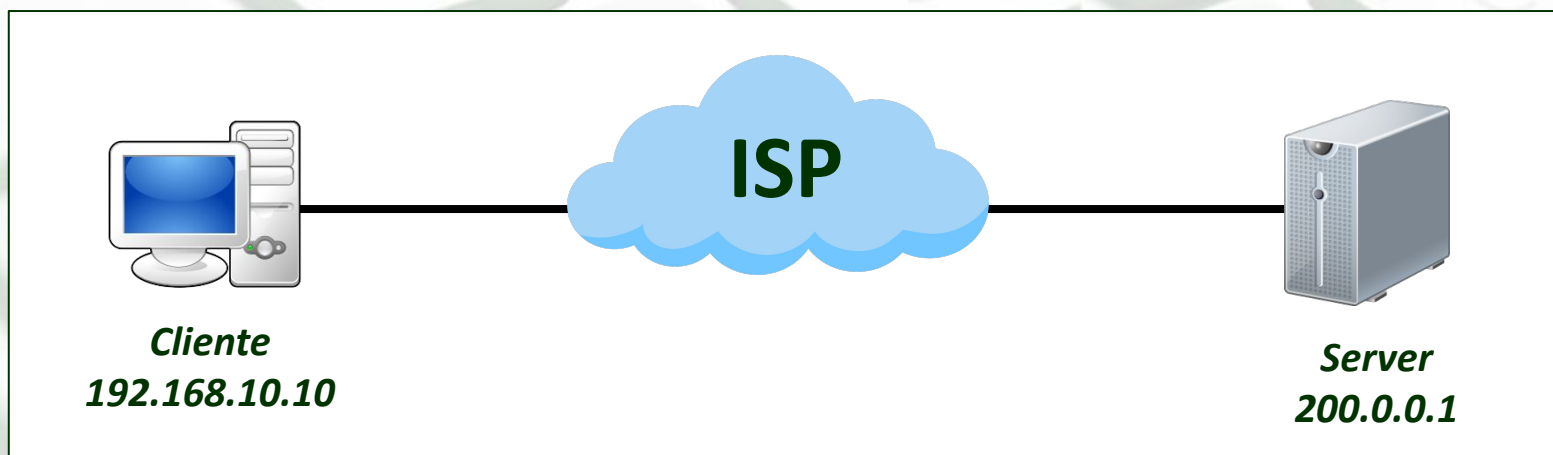

Laboratório 08-1

- Crie um novo usuário no Server:

```
# adduser nome_usuario
```

- A partir da VM Cliente, acesse o Server remotamente.

```
# telnet 200.0.0.1
```

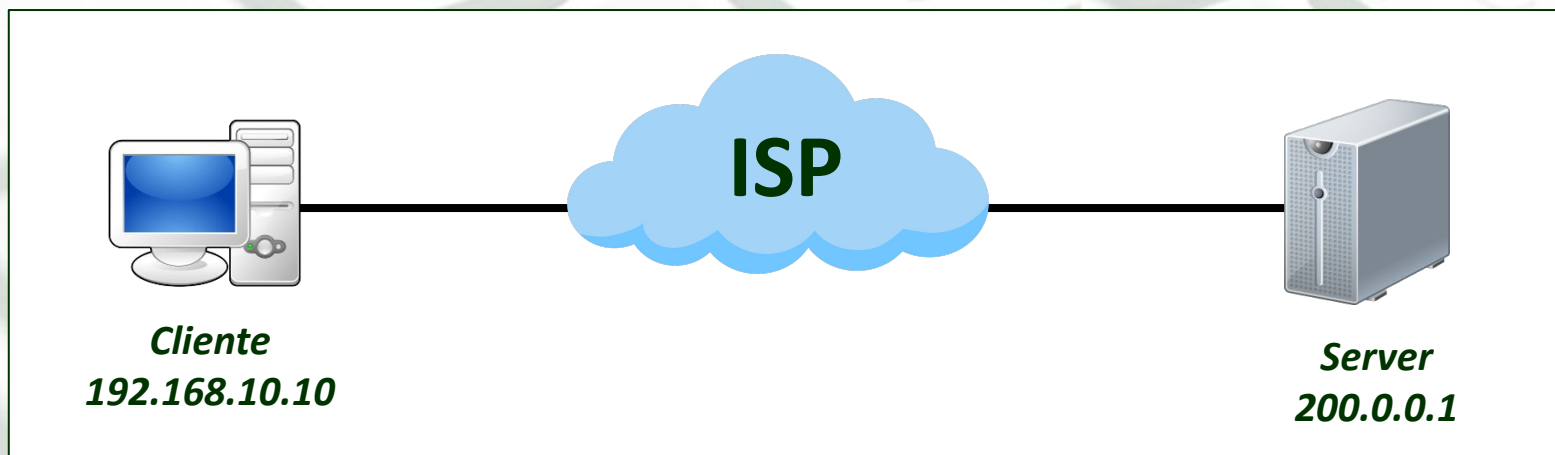




Laboratório 08-1

- No ISP, utilize um Sniffer + Analisador de Pacotes para inspecionar como as credenciais de autenticação são transmitidas entre o Cliente e o Server.

```
# tcpdump -w escutaSSH.pcap
```

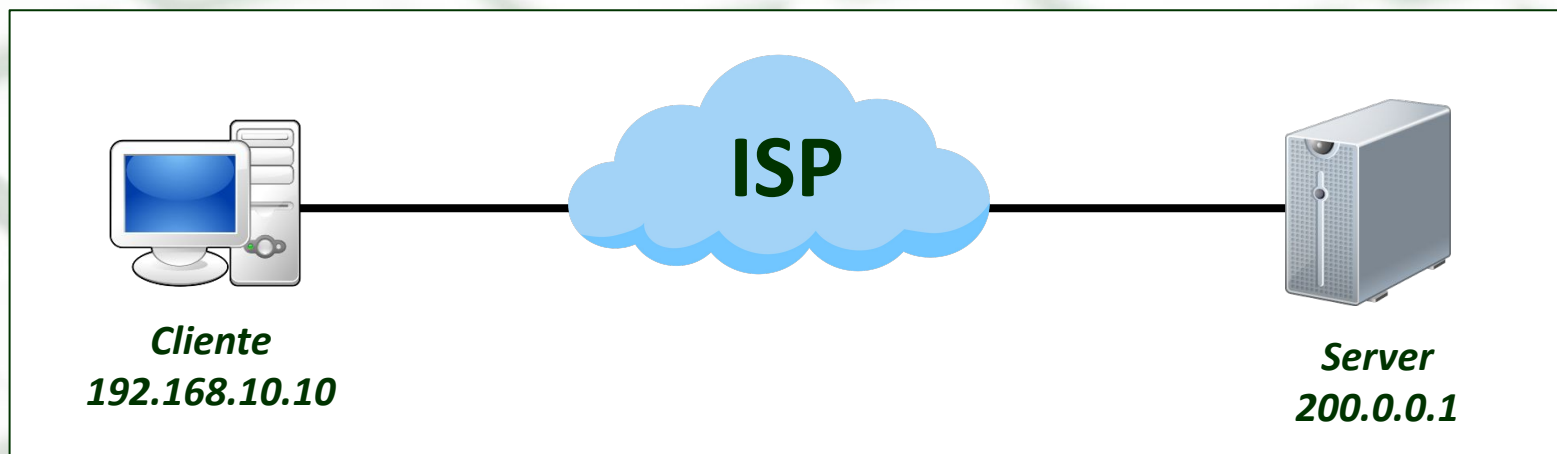




Laboratório 08-1

- Façamos outro teste...
 - Inicie o servidor HTTP (Apache) do server...

```
# /etc/init.d/apache2 start
```



- Inicie uma conexão TELNET para a porta 80, e verifique...

```
# telnet 200.0.0.1 80  
> GET /
```



Laboratório 08-1

- Pelo navegador, acesse o site “pudim.com.br”





Laboratório 08-1

- Agora, veja pelo Telnet...

```
adriano@adriano-notebook: ~  
Arquivo  Editar  Ver  Pesquisar  Terminal  Ajuda  
adriano@adriano-notebook:~$ telnet 54.207.20.104 80  
Trying 54.207.20.104...  
Connected to 54.207.20.104.  
Escape character is '^]'.  
GET / HTTP/1.1  
Host: pudim.com.br  
[ ]
```




Laboratório 08-1

■ Agora, veja pelo Telnet...

```
adriano@adriano-notebook: ~  
Arquivo Editar Ver Pesquisar Terminal Ajuda  
GET / HTTP/1.1  
Host: pudim.com.br  
  
HTTP/1.1 200 OK  
Date: Tue, 04 Apr 2023 13:49:36 GMT  
Server: Apache/2.4.34 (Amazon) OpenSSL/1.0.2k-fips PHP/5.5.38  
Last-Modified: Wed, 23 Dec 2015 01:18:20 GMT  
ETag: "353-527867f65e8ad"  
Accept-Ranges: bytes  
Content-Length: 851  
Content-Type: text/html; charset=UTF-8  
  
<html>  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head>  
  <title>Pudim</title>  
  <link rel="stylesheet" href="estilo.css">  
</head>  
<body>  
<div>  
  <div class="container">  
    <div class="image">  
        
    </div>
```



Secure Shell

SSH



SSH

- **SSH (Secure SHell)** também é um protocolo padrão da arquitetura TCP/IP para acesso remoto a hosts.
- Ao contrário do Telnet, o SSH implementa **comunicação criptografada** entre o cliente e o servidor remoto.
- A autenticação é baseada em **Criptografia Assimétrica: Algoritmo RSA** (*Rivest, Shamir e Adleman*).

Maior Segurança



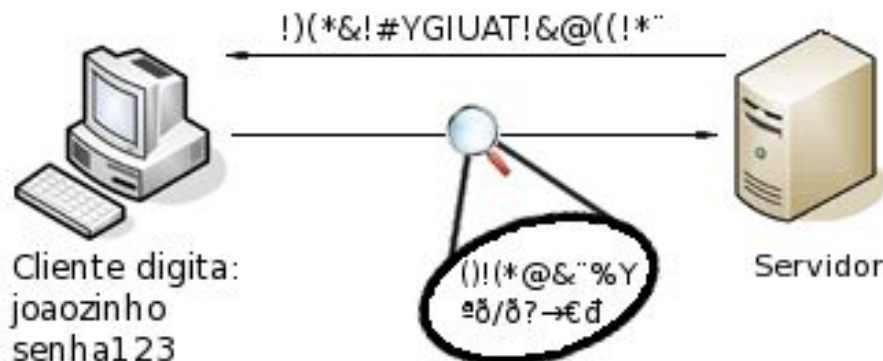


Telnet vs. SSH

Sessão de login sem criptografia como no telnet



Sessão de login criptografada como no SSH





INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Fundamentos de Criptografia



Alice

Olá Bob!



Bob



INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Fundamentos de Criptografia



Alice

Olá Bob!

Algoritmo de Criptografia: Cada letra deve avançar N posições à frente...



Bob



INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Fundamentos de Criptografia



Alice

Olá Bob!

Algoritmo de Criptografia: Cada letra deve avançar N posições à frente...

Chave da Criptografia: $N = 3$



Bob



Fundamentos de Criptografia



Alice

Olá Bob!

Algoritmo de Criptografia: Cada letra deve avançar N posições à frente...

Chave da Criptografia: $N = 3$

Rod Ere!



Bob



Fundamentos de Criptografia



Alice

Olá Bob!

Algoritmo de Criptografia: Cada letra deve avançar N posições à frente...

Chave da Criptografia: $N = 3$

Rod Ere!



Bob

O que Bob precisa saber para conseguir ler a mensagem de Alice?



Fundamentos de Criptografia



Alice

Olá Bob!

Algoritmo de Criptografia: Cada letra deve avançar N posições à frente...

Chave da Criptografia: $N = 3$

Como Alice informa a chave para Bob SEM que Darth também a veja?

Rod Ere!



Bob



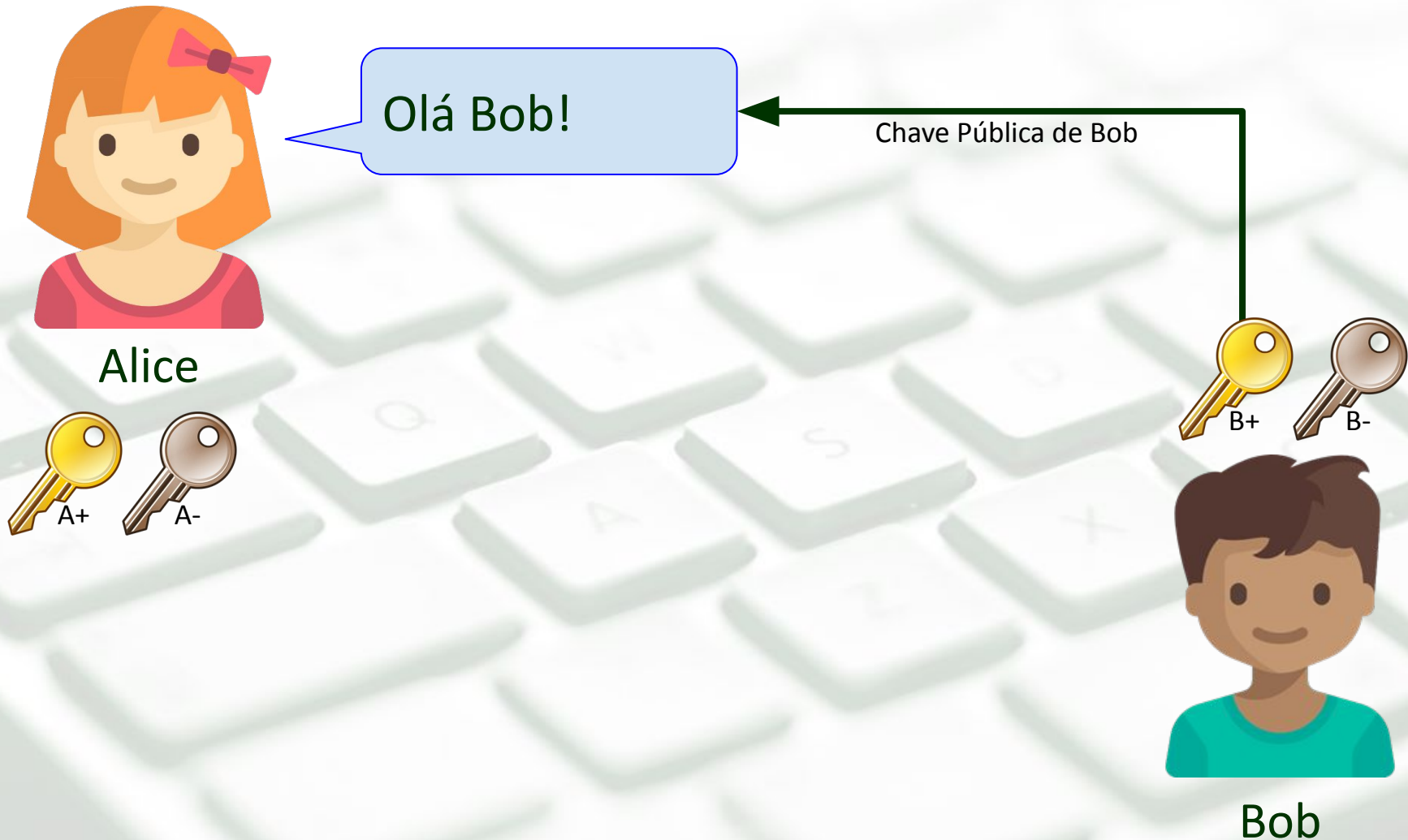
Fundamentos de Criptografia

- Existem dois modelos básicos de criptografia...
- **Criptografia Simétrica**
 - Como mostrado no exemplo anterior...
 - A chave usada para criptografar deve ser a mesma para descriptografar (*simetria*).
- **Criptografia Assimétrica**
 - Arquitetura de Chaves Públicas (e Privadas)
 - Cada ente possui um par de chaves inter-relacionadas matematicamente.



INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

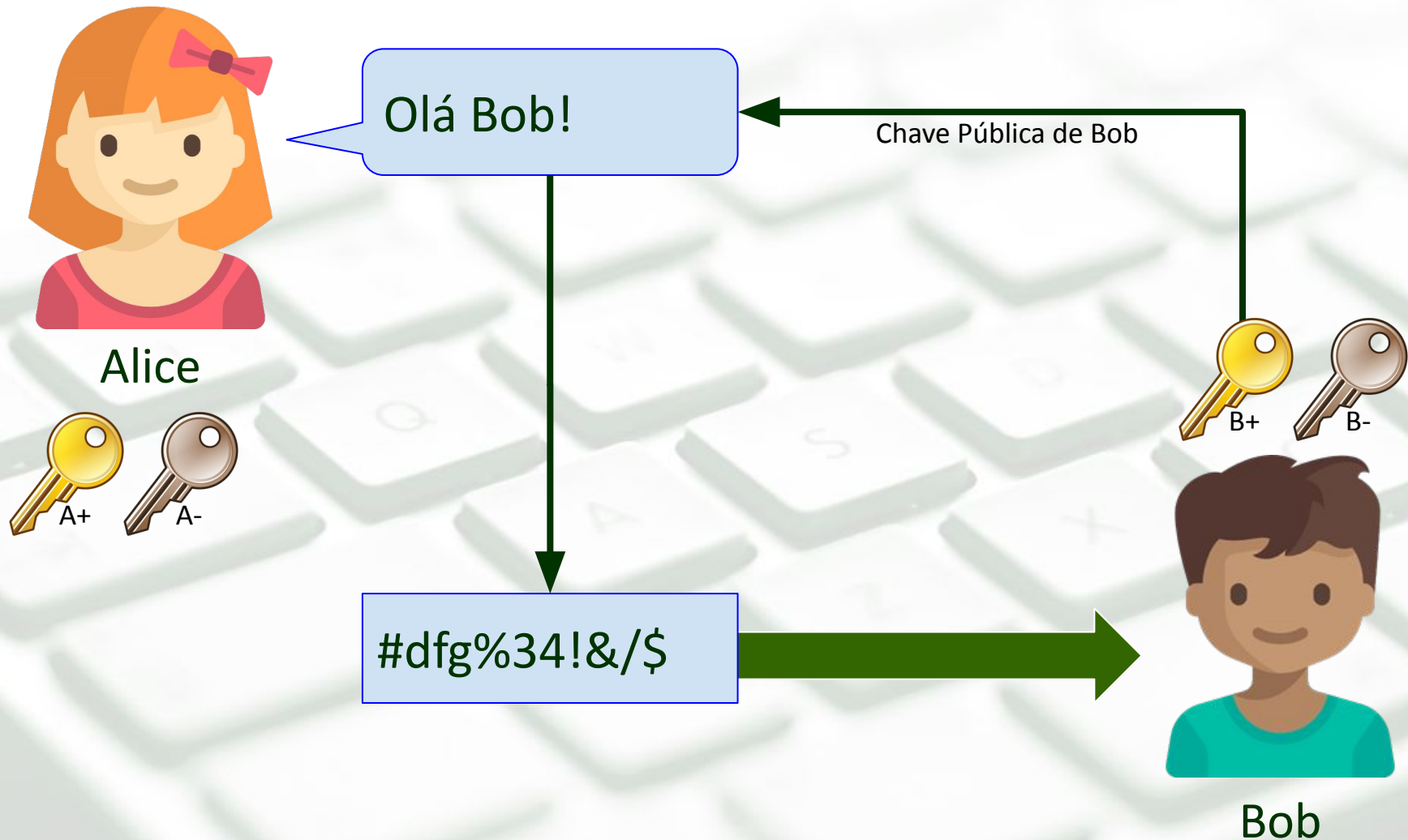
Fundamentos de Criptografia





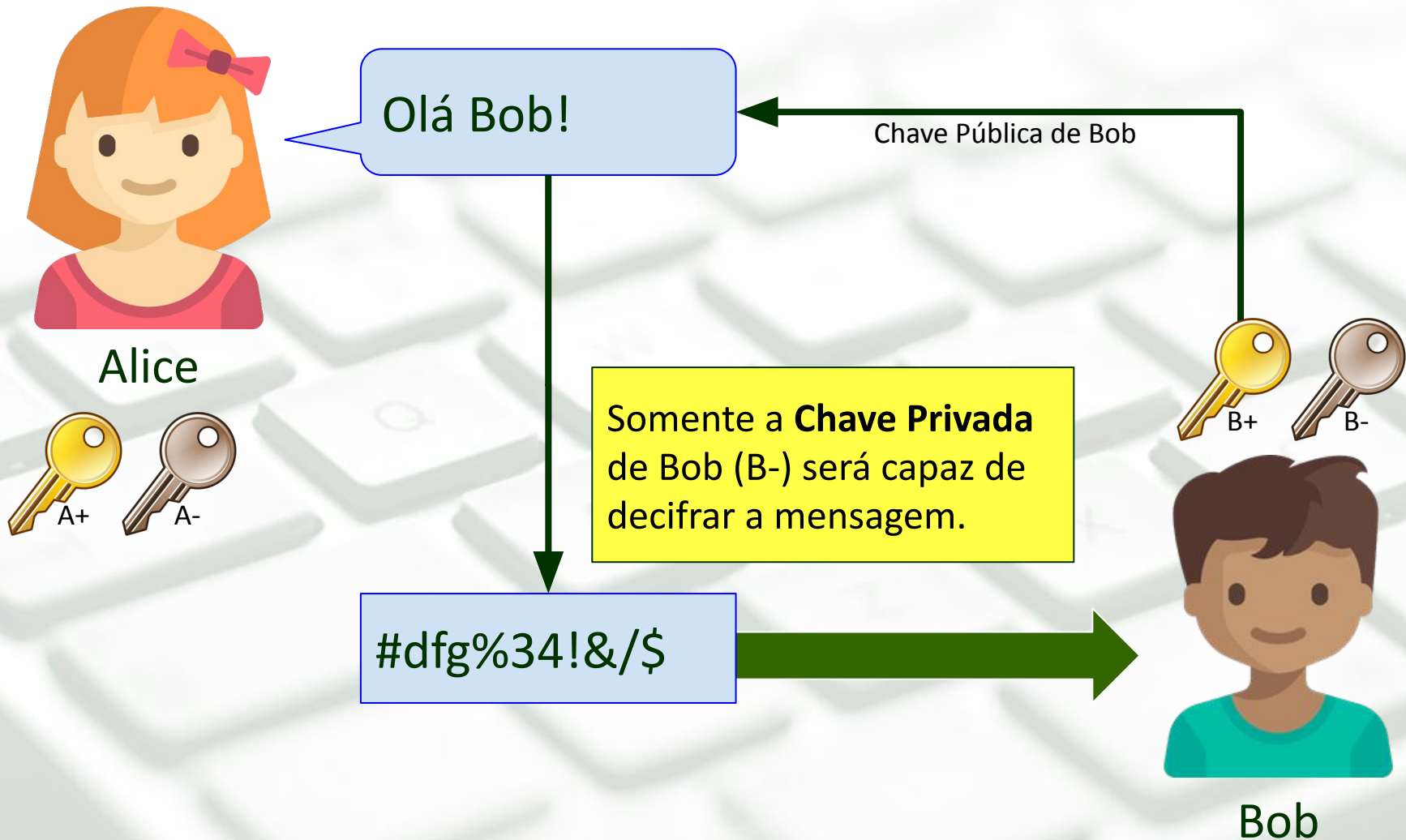
INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Fundamentos de Criptografia





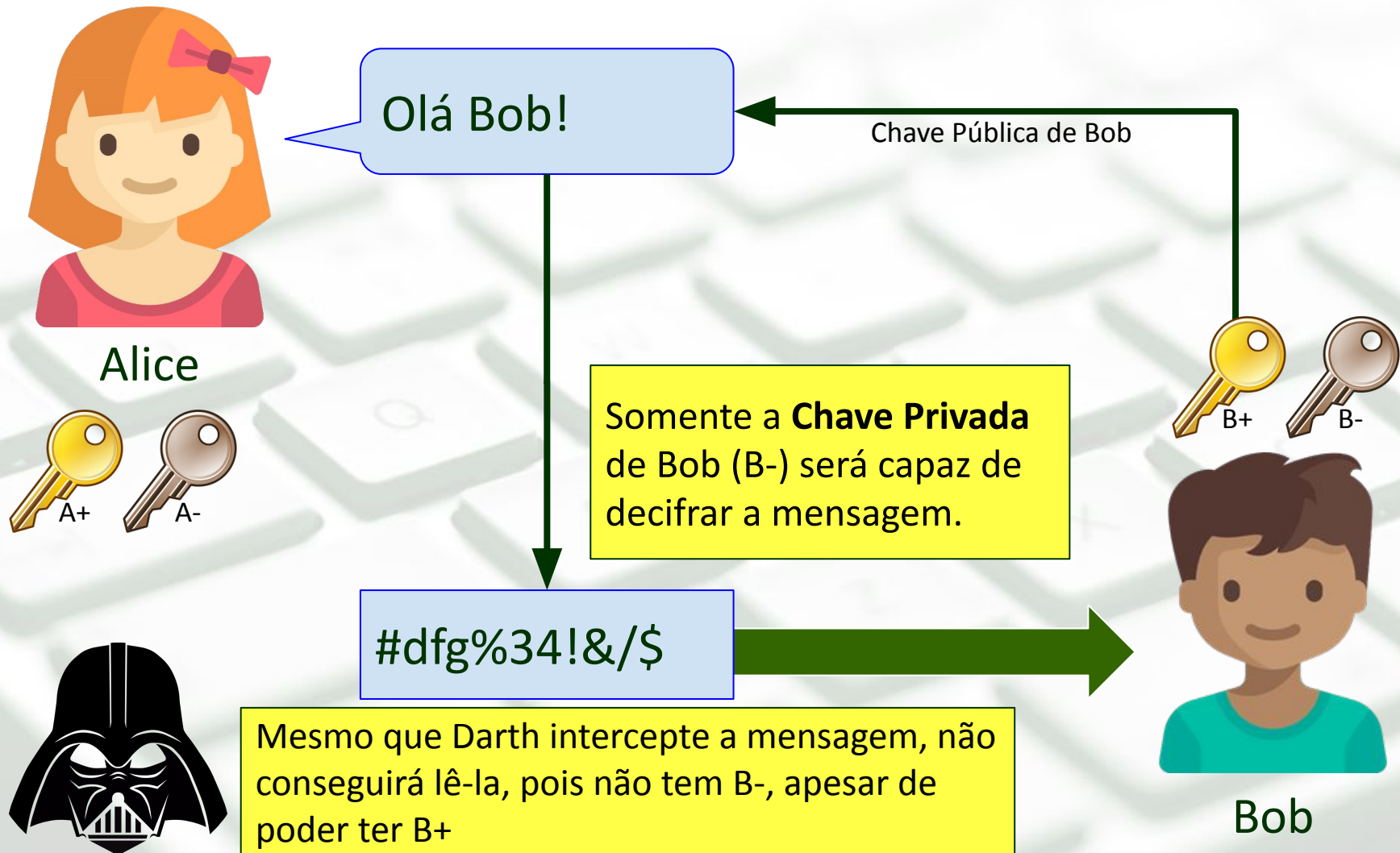
Fundamentos de Criptografia





INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Fundamentos de Criptografia





Autenticação SSH



Chave Privada C-

**Cliente
SSH**

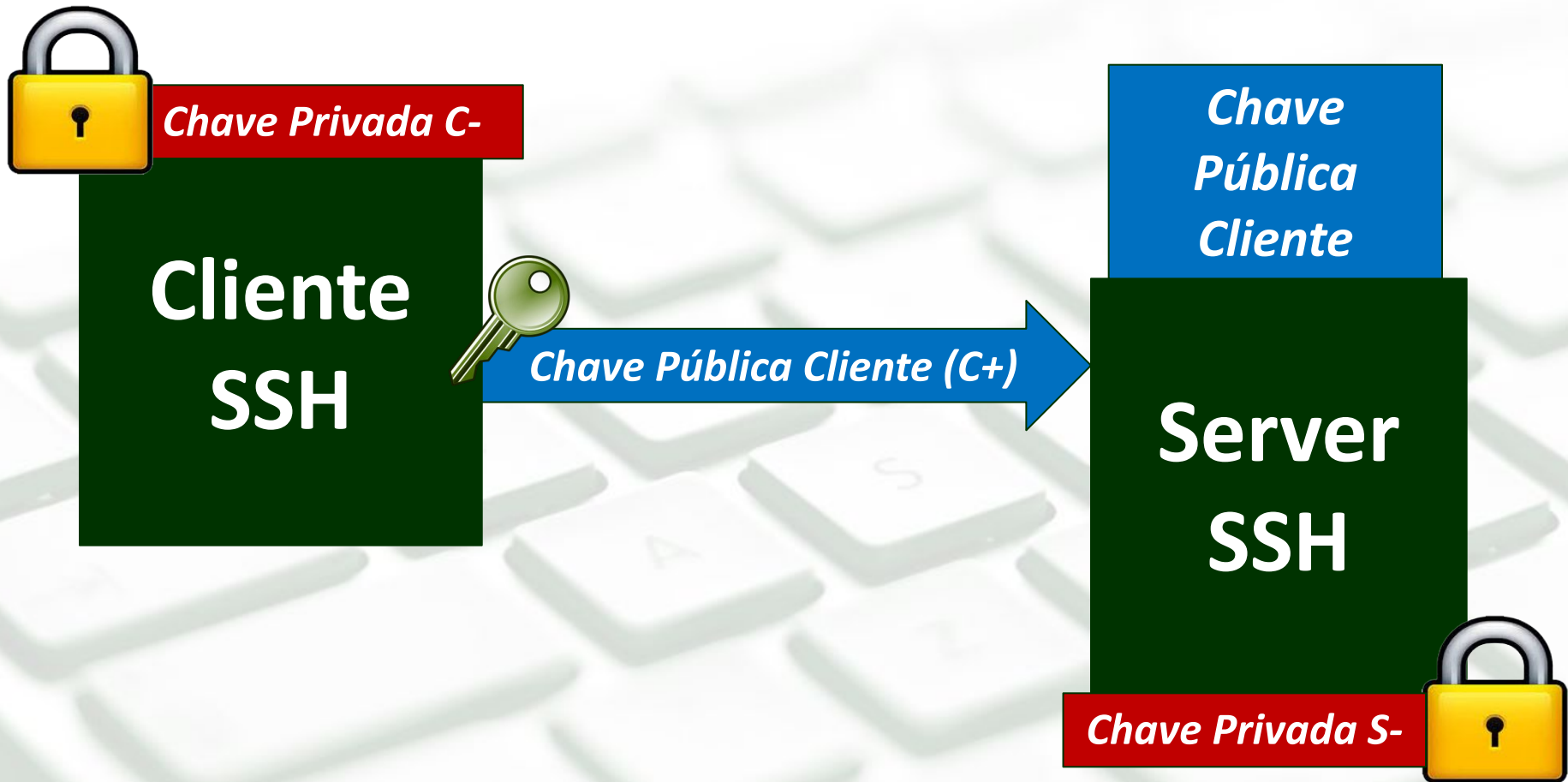
**Server
SSH**

Chave Privada S-



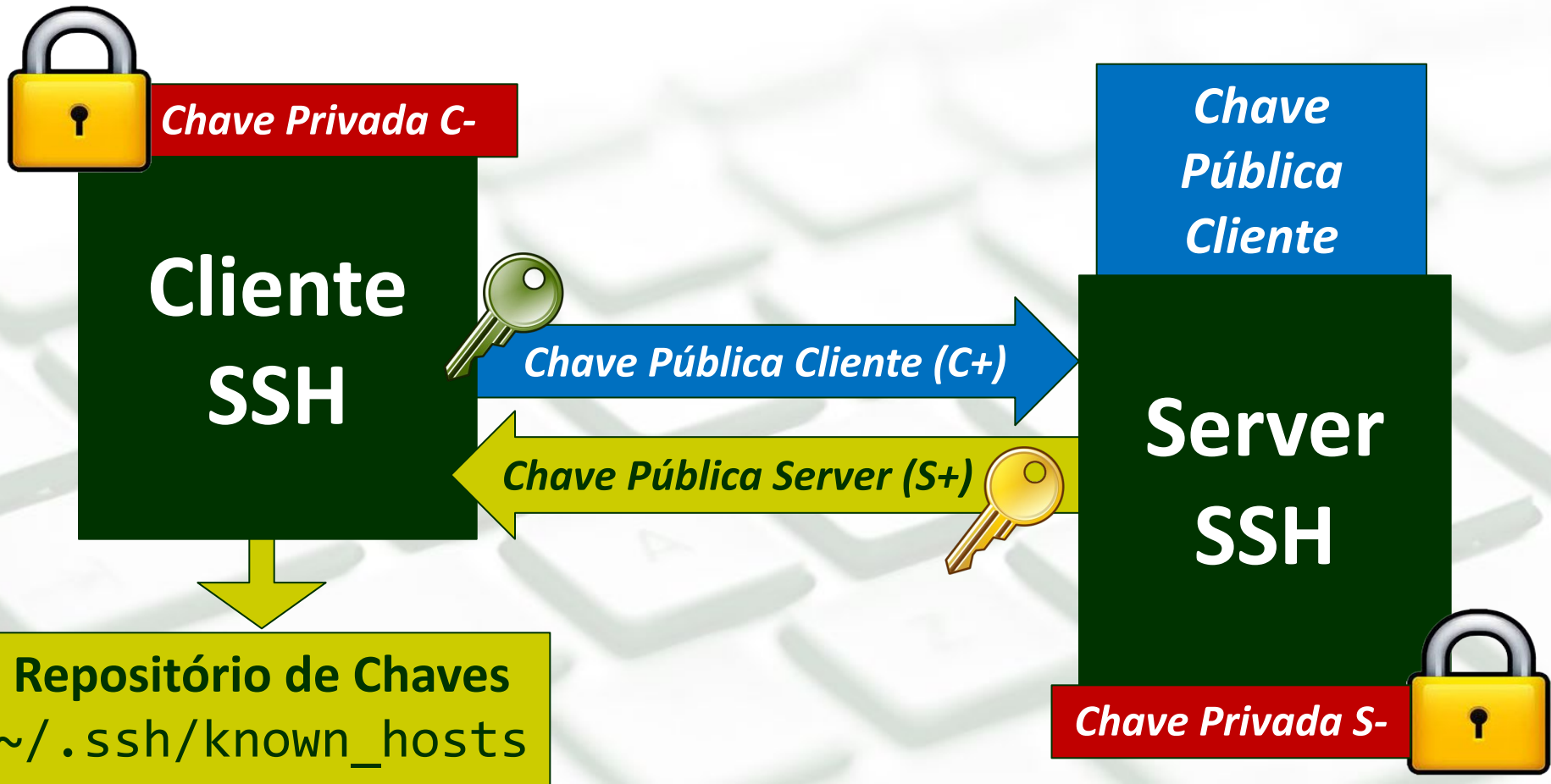


Autenticação SSH



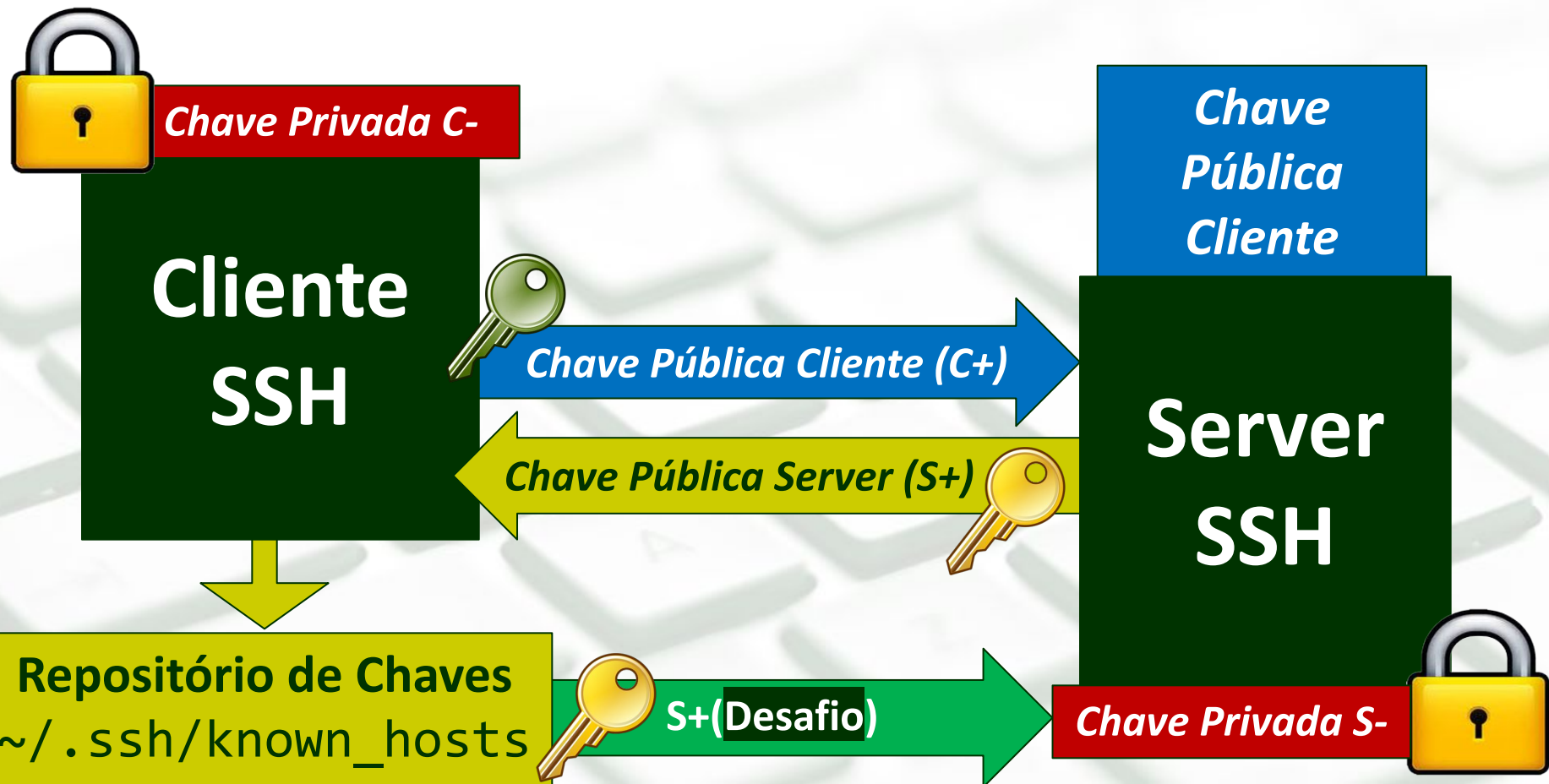


Autenticação SSH



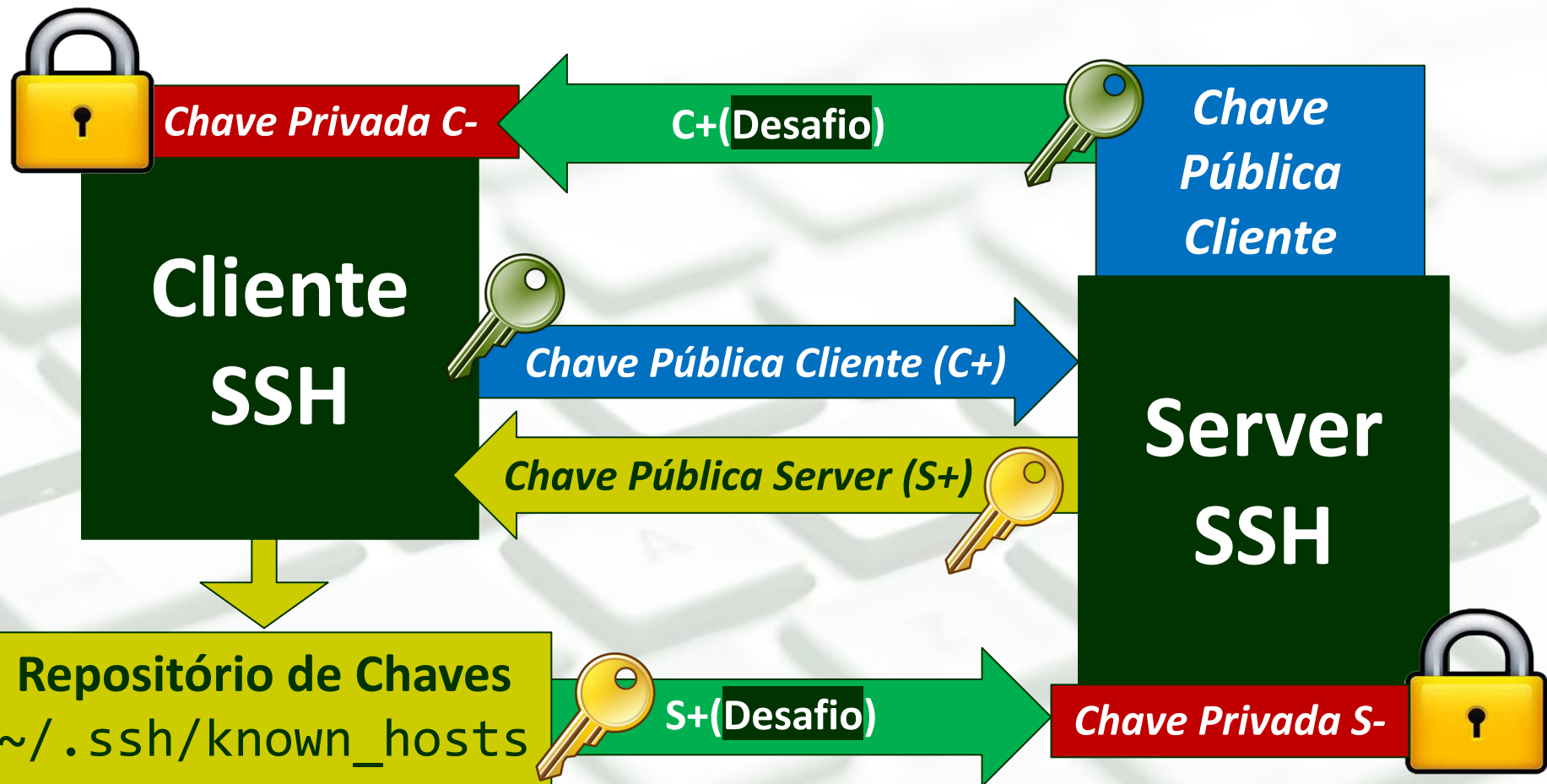


Autenticação SSH



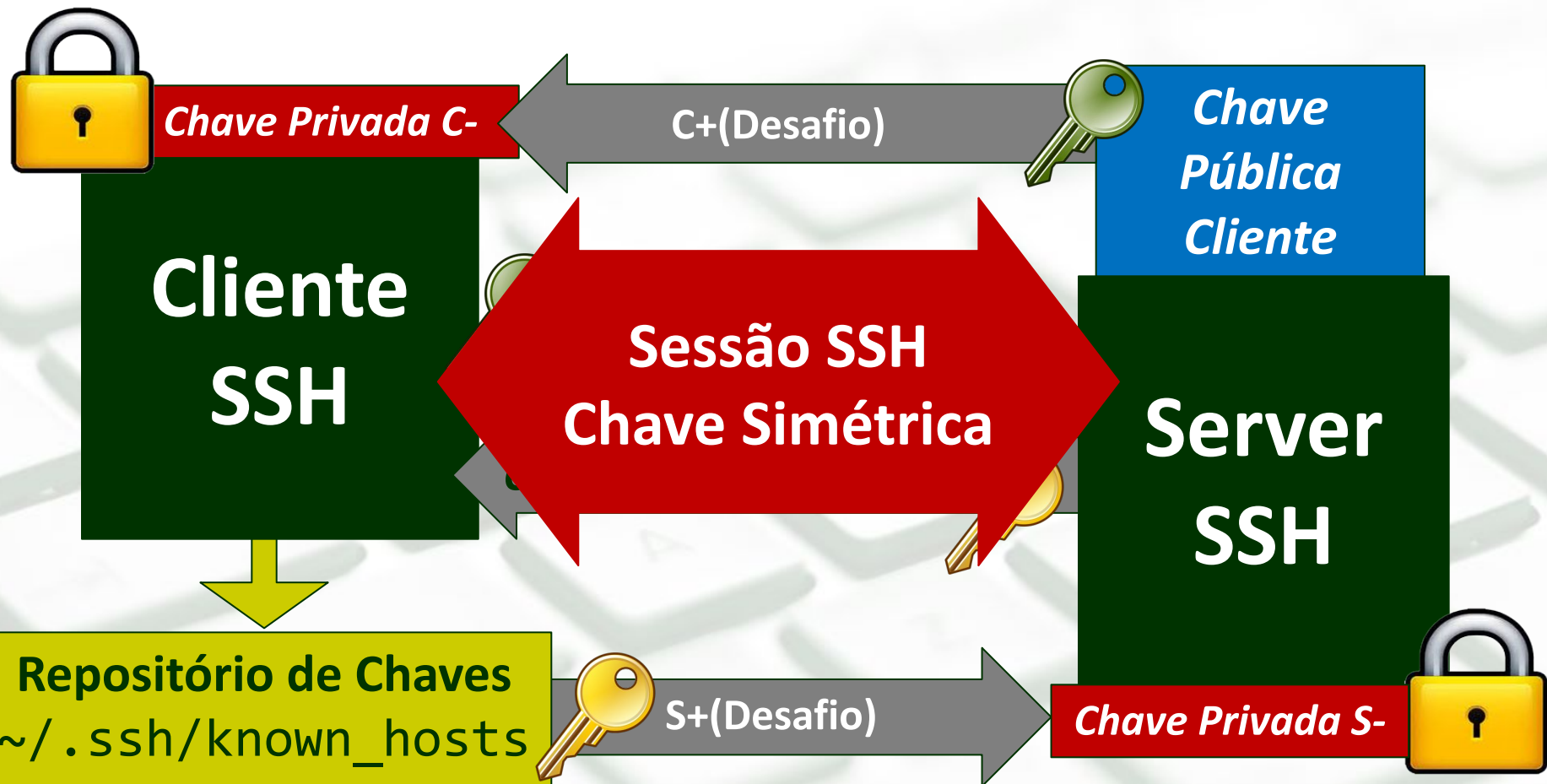


Autenticação SSH





Autenticação SSH





SSH

■ Instalação:

```
# apt-get install openssh-server  
# apt-get install openssh-client
```

■ Configuração:

```
# /etc/ssh/sshd_config (server)  
# /etc/ssh/ssh_config (client)
```

■ Ativação

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



Laboratório 08-2

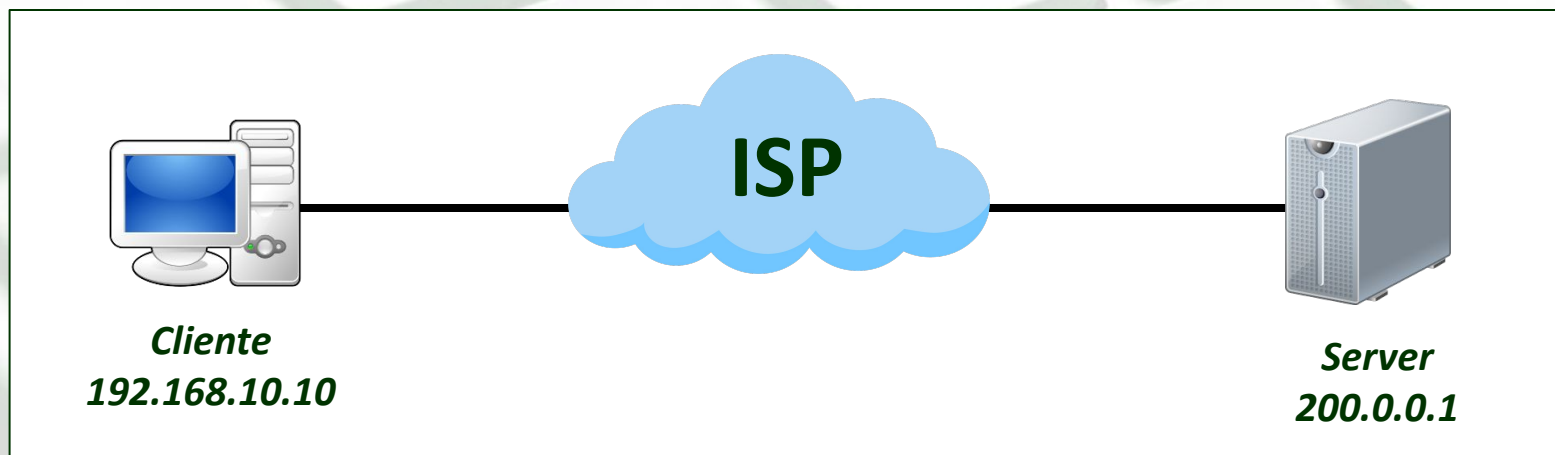
- Crie um usuário no Server.

```
# adduser nome_usuario
```

- Acesse remotamente o Server.

```
# ssh nome_usuario@200.0.0.1
```

```
# su -
```





Chaves de Autenticação

- Abra o arquivo abaixo no **Cliente** e veja a identificação da chave pública do **Server**:


```
# nano ~/.ssh/known_hosts
```

- Exclua uma chave pública do repositório do Cliente:

```
# ssh-keygen -R 200.0.0.1
```

- As chaves dos **hosts** estão localizadas em:

```
# /etc/ssh/ssh_host_rsa_key  
# /etc/ssh/ssh_host_rsa_key.pub
```



Sempre mantenha as chaves privadas bem protegidas!



Autenticação por Chaves

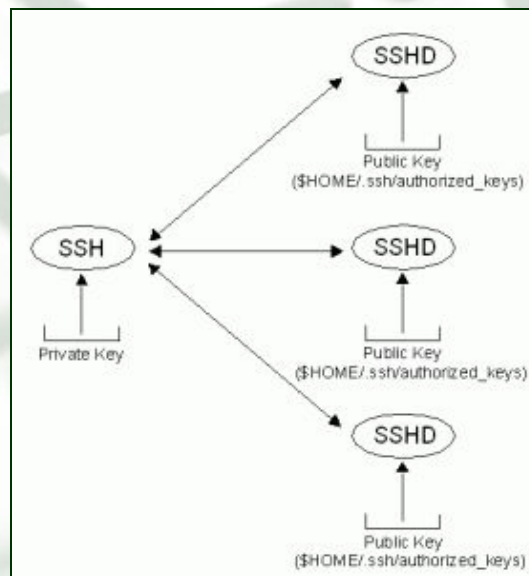
- **Autenticação por Chaves** ou **Autenticação de Duas Vias** é um método ainda mais seguro para fazer a autenticação entre duas máquinas remotas.
- Nesse método, a autenticação é feita através de **chaves assimétricas geradas pelo usuário** - ao invés de usar a sua própria senha de acesso.
 - *Evita roubo de senhas por “olhudos” de plantão e Ataques de brute-force.*
- A **chave pública** gerada pelo usuário deve ser instalada no servidor, e a **chave privada** (armazenada localmente) pode ser (ou não) protegida por uma ***passphrase***.



Autenticação por Chaves

- Porque a autenticação por chaves é mais segura?

Para que um invasor consiga ter acesso indevido a um servidor é necessário que ele roube a chave privada do usuário, e ainda conheça a passphrase que a decodifica.





Chaves de Autenticação

- Para gerar um par de chaves utilize o comando:

```
# ssh-keygen
```

- As chaves serão salvas no diretório “home” do usuário:

```
# ~/.ssh/id_rsa
```

```
# ~/.ssh/id_rsa.pub
```

- Instale a chave pública no servidor:

```
# ssh-copy-id -i ~/.ssh/id_rsa.pub login@server
```



Chaves de Autenticação

- Recomenda-se (por simplificação) que o nome do usuário no servidor remoto seja o mesmo nome de usuário do cliente.
- As chaves públicas autorizadas a acessar uma determinada conta de usuário no servidor, são instaladas no arquivo:

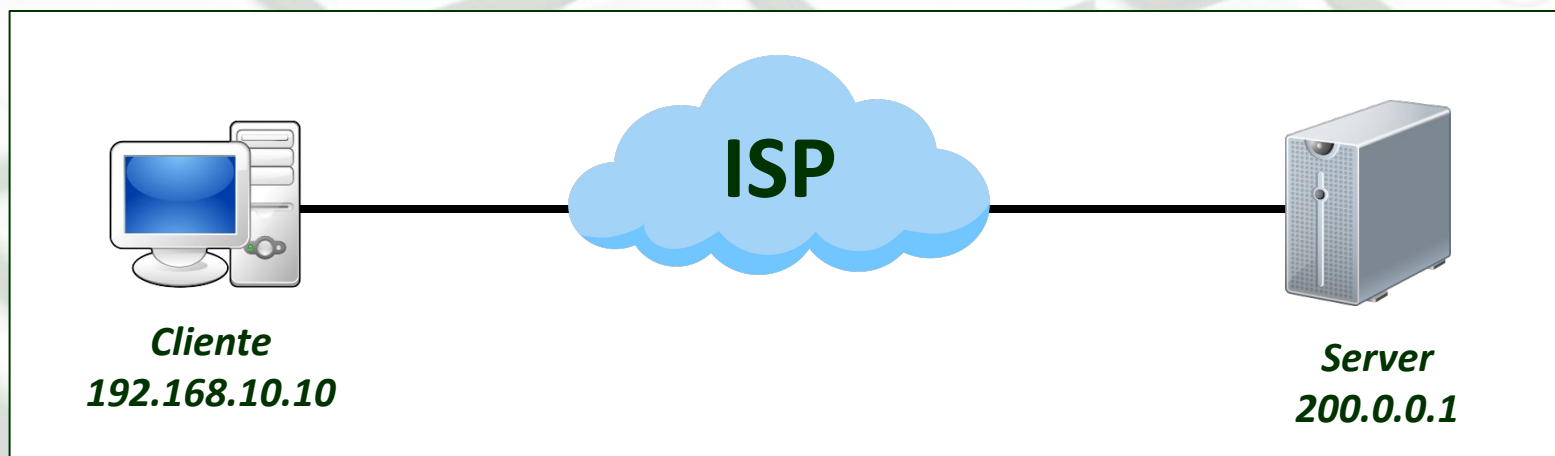
```
# ~/.ssh/authorized_keys
```

** Também é possível instalar (copiar) a chave pública do cliente diretamente no arquivo `authorized_keys`, ao invés de usar o comando `ssh-copy-id`.*



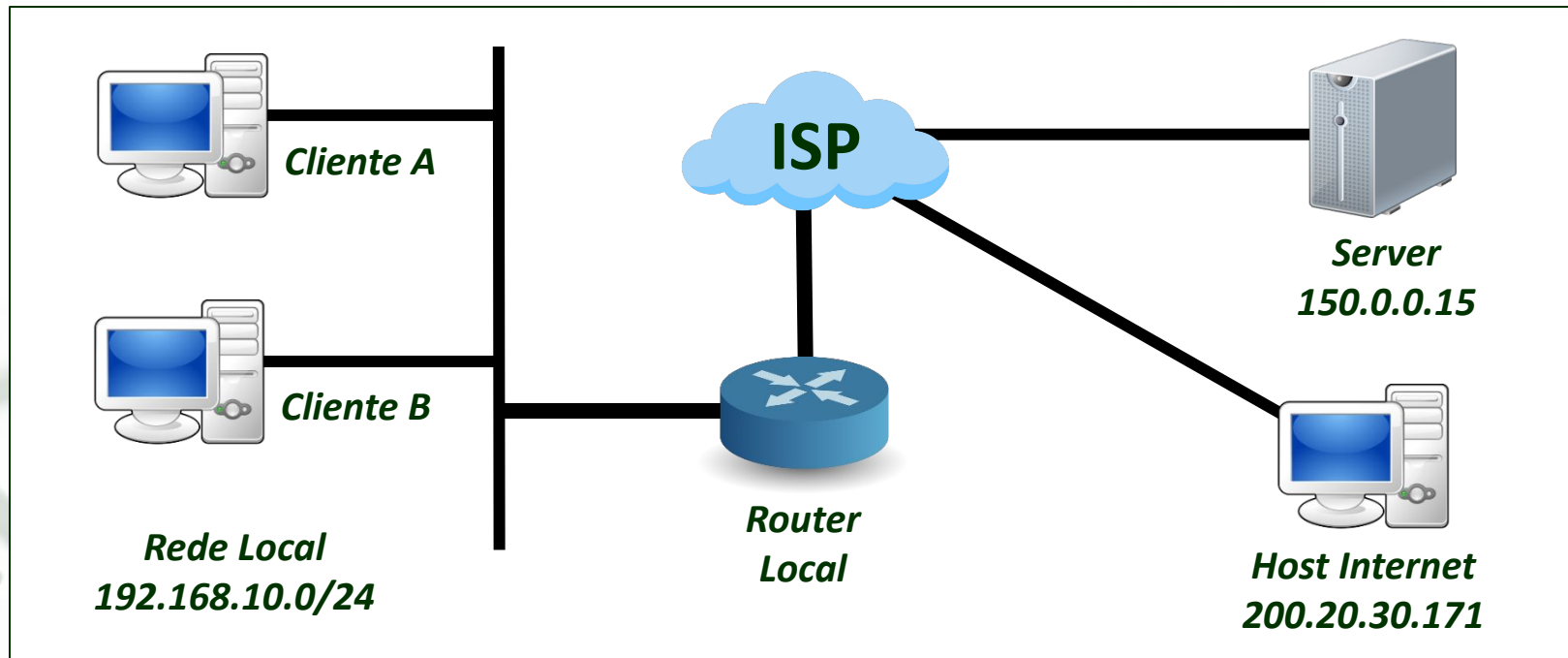
Laboratório 08-3

- Crie um par de chaves de autenticação SSH para o usuário "Admin".
- Quando este usuário tentar acessar remotamente sua conta no Server, não deverá ser solicitada nenhuma senha (a autenticação será via chaves assimétricas).





Laboratório 08-4



- A partir do “Cliente A”, use o SSH (com autenticação via chaves) para subir a porta 80 em modo escuta no Server (use o netcat).
- A partir de um “Host qualquer na Internet”, use o SSH (login+senha) para “invadir” o “Router Local” e criar um sniffer (captura de tráfego) para gerar e salvar em arquivo todos os pacotes que trafegam ali.
- A partir do “Cliente B”, conecte-se a porta 80 do server, enviando e recebendo informações.
- Através do Wireshark, inspecione o arquivo gerado pelo *sniffer* do Router Local.



Boas Práticas SSH

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

```
# Alterar a porta padrão (22) do serviço
```

```
Port 1025
```

```
# Endereço de escuta da conexão
```

```
ListenAddress 192.168.10.1
```

```
# Desabilite login do usuário root (apenas por chaves já é padrão!)
```

```
PermitRootLogin prohibit-password || no
```

```
# Desabilite login de usuários por senha (força que a autenticação  
aconteça apenas por chaves)
```

```
PasswordAuthentication no
```

```
# Nº Tentativas de conexão sem sucesso (5), % de recusa de chamadas  
após as 5 iniciais, Nº Máximo Total até bloqueio total.
```

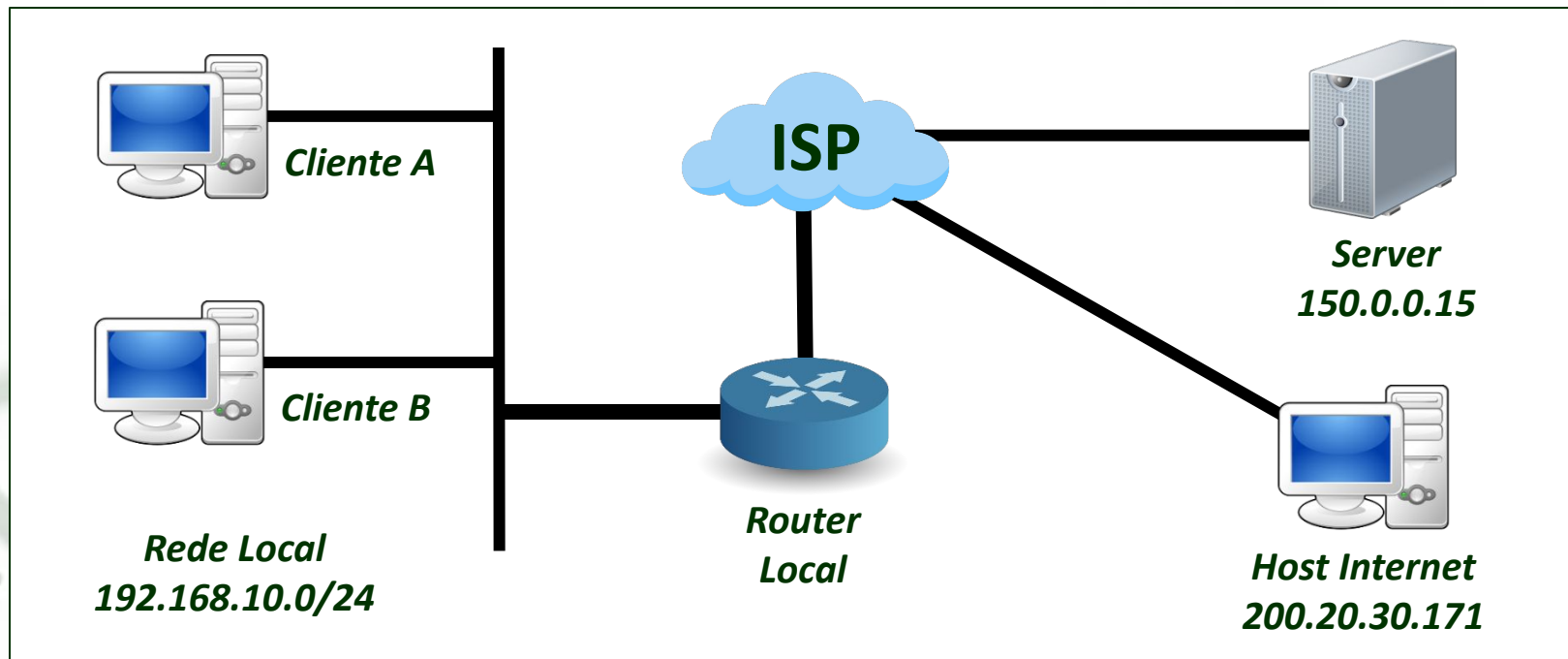
```
MaxStartups 5:70:8
```

```
# Permite apenas conexões para os usuários explicitamente indicados.
```

```
AllowUsers adminIfnmg
```



Laboratório 08-5

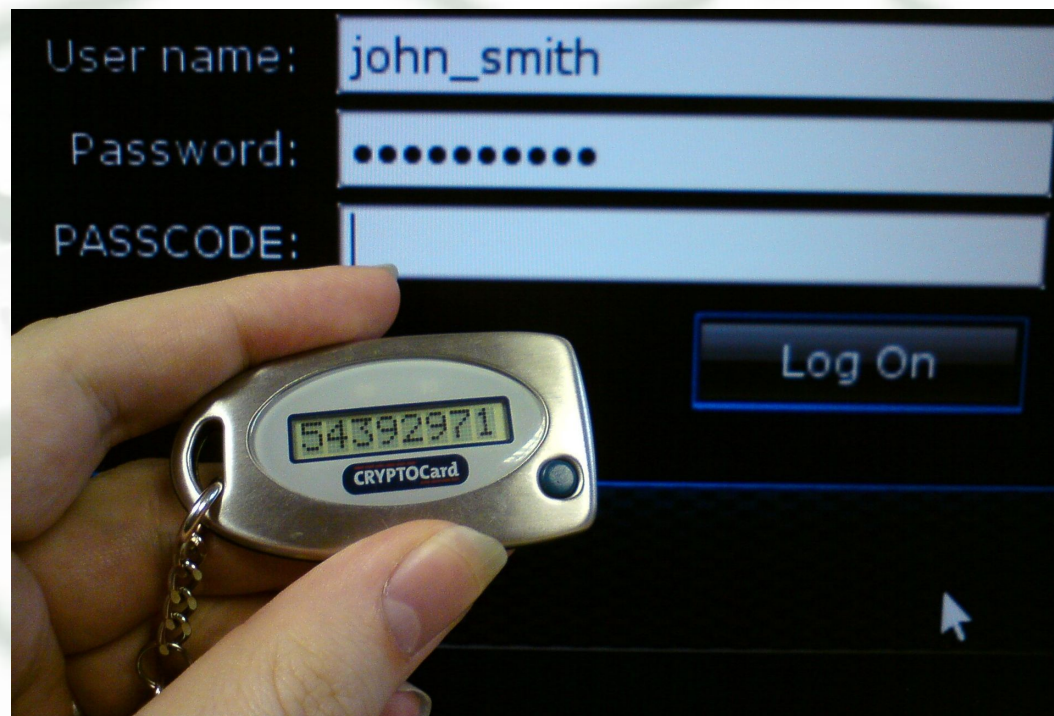


- Aprimore a segurança do Lab08-4:
 - Conexões SSH para o “Router Local” devem ser aceitas APENAS provenientes da Rede Local, e específicas para um usuário bem definido (p.ex.:adminIfnmg)
 - Conexões SSH para o “Server” devem ser feitas EXCLUSIVAMENTE via chaves assimétricas, porta padrão deve ser a 5001, e sendo 5 tentativas no máximo.



Token Authentication

- Podemos implementar uma outra camada de segurança para autenticação de usuários através de **Tokens mutáveis**.
- Essa técnica é conhecida como MFA (Multi-Factor Authentication)
- Combinações MFA:
 - Senha + Token
 - Chave + Token





Laboratório 08-6

- Vamos implementar a autenticação multifator por **Chave+Token em uma instância AWS.**



- Crie uma nova instância Debian
- Acesse a instância via SSH
- Instale a ferramenta de token: ***Google-Authenticator***

```
# apt update
```

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

- Altere os arquivos a seguir...



Laboratório 08-6

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

```
# ATENÇÃO! Se for usar Chave+Token, comente a linha  
abaixo, se for usar Senha+Token, deixe como está...
```

```
@include common-auth
```

```
# Token Authentication via Google PAM
```

```
auth required pam_google_authenticator.so
```




Laboratório 08-6

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

```
# Habilita a autenticação via token no SSH  
ChallengeResponseAuthentication yes
```

```
UsePAM yes
```

```
# SE o modelo for CHAVE+TOKEN, adicionar essa  
instrução no final do arquivo...
```

```
AuthenticationMethods publickey,keyboard-interactive
```



Laboratório 08-6

- Acesse a conta do usuário que fará a autenticação por token, e configure o APP externo...

```
# su admin
```

```
# Instale um APP para visualizar os tokens gerados, p.ex.: Google Authenticator
```

```
# Através deste APP, escaneie o QR code gerado pelo comando abaixo...
```

```
# google-authenticator
```

```
# Recomendado (Y)es para todas as opções seguintes...
```



Laboratório 08-6

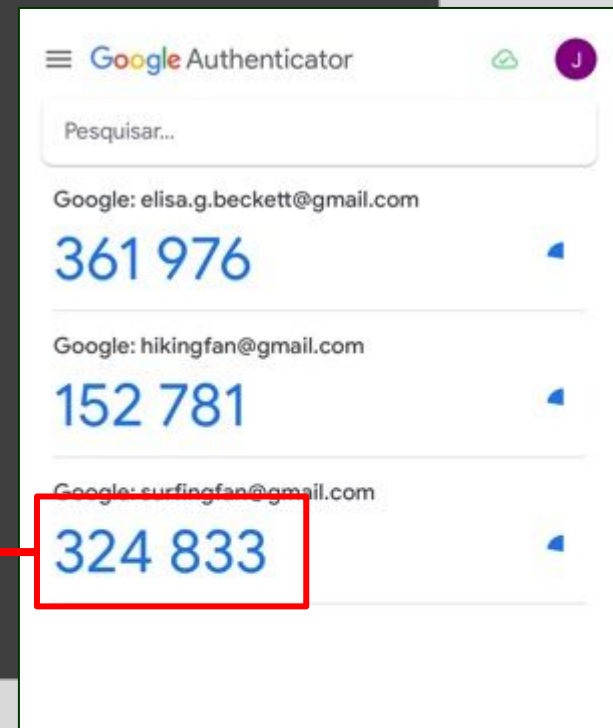
FA, e

g

p

#

```
adriano@adriano-notebook: ~/Dropbox/aws
Arquivo  Editar  Ver  Pesquisar  Terminal  Ajuda
adriano@adriano-notebook:~/Dropbox/aws$ ssh -i chaveAWS.pem admin@18.234.125.54
Verification code: 
```



seguintes...



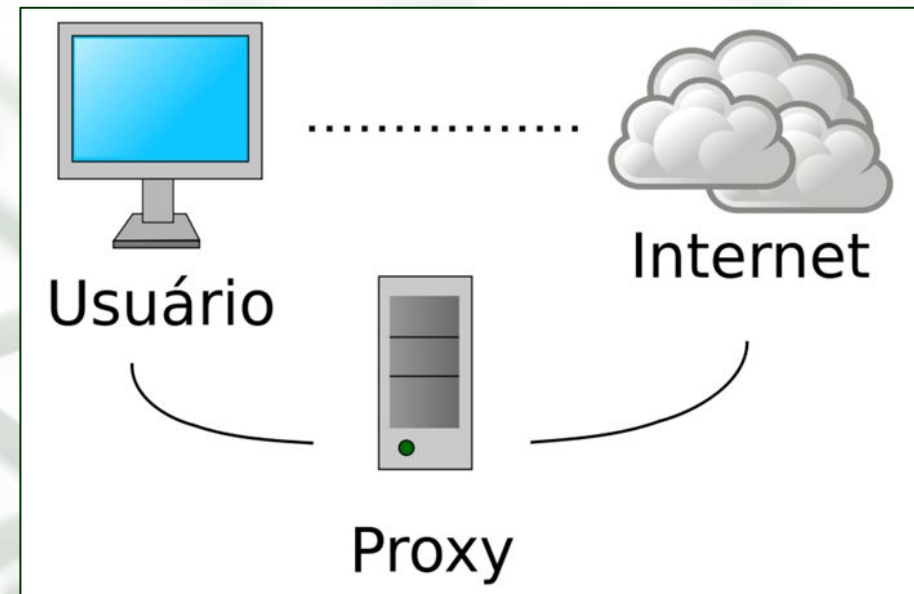
Seminário Individual

PROXYs

O que são e para que servem?

Proxy Socks vs. Proxy HTTP

Proxy vs. VPNs



LINK para Vídeo Introdutório



Referências

- **Guia Foca GNU/Linux.**
Disponível em <http://www.guiafoca.org/>
- **MORIMOTO, Carlos E; Servidores Linux – Guia Prático.**
- [Set Up SSH Two-Factor Authentication \(2FA\) on Debian 11 Server.](#)