Segurança - Secure Shell (ssh)

Anderson Alves de Albuquerque *aaa@cbpf.br*
Marita Maestrelli *marita@cbpf.br*

Setembro 2001

Prefácio

Muitos problemas relacionados a segurança em rede de computadores[5.68.1] começam pelo simples fato do atacante conseguir algum tipo de acesso em algum elo fraco e em seguida instalar um simples sniffer[5.72.1] que por conseqüência consegue ter acesso a outros elos mais fortes do sistema.

Para resolver ou minimizar esse problema uma das soluções é a instalação de aplicações com suporte a encriptação[5.26.1] de senhas e dados.

Atualmente, existem muitas ferramentas que suportam a encriptação[5.26.1], mas selecionamos as que: possuem código aberto (GNU/GPL)[5.35.2], suporte a diversos sistemas operacionais (cliente e servidor), escalabilidade, documentação disponível, algorítmos de criptografia[5.20] abertos e suporte a diversos serviços usados em internet/intranet[5.45]. Tendo estas características como delimitadora da escolha a solução que melhor satisfez no presente momento (1999 á 2001) é o SSH1[5.75] e SSH2[5.75].

Segurança - Secure Shell (SSH)

Índice:

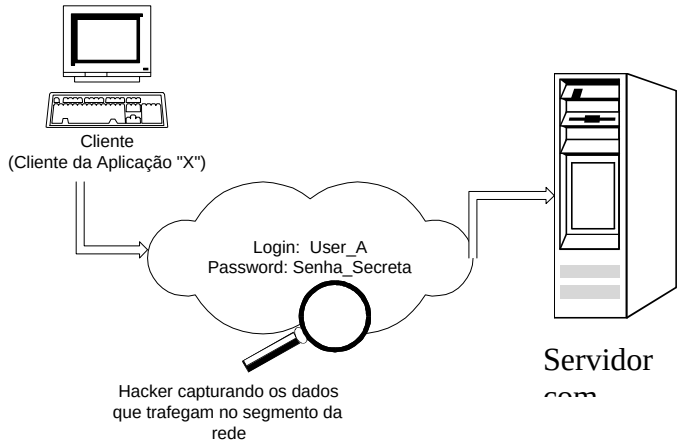
1 - Introdução

Abordaremos de forma prática a implementação do SSH1 (RSA)**[5.75]** e SSH2 (DSA)[5.75], não temos a intenção aqui de entrar em detalhes sobre criptografia.

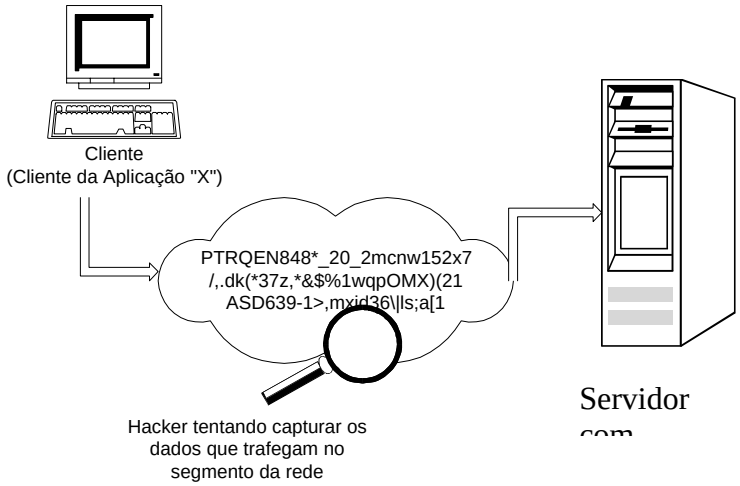
Nesta documentação utilizaremos o SSH (Secure Shell)[5.75] para acessos remotos substituindo serviços que apresentam problemas graves de segurança como o FTP[5.34], Telnet[5.79], POP3[5.64.1], X11 e outros. Estes serviços não possuem nenhum mecanismo de encriptação[5.26.1], facilitando a captura de dados por sniffers[5.72.1].

O SSH (Secure Shell)[5.75] é muito parecido com o telnet[5.79] mas possui suporte a criptografia[5.20] de dados tornando a técnica de sniffer [5.72.1] ineficiente, evitando que logins, senhas, cartões de créditos e outros dados sejam grampeados.

Esquema da situação usando aplicações inseguras como Telnet/Rlogin:



Esquema da situação usando aplicações seguras como SSL/SSH:



O SSH (Secure Shell)[5.75] possui atualmente várias implementações cliente e servidora, entre elas podemos destacar: ssh , openssh, ossh, sftp (Secure FTP) e outras. Os clientes são distribuídos para quase todas as plataformas existentes e possuem interface gráfica ou trabalham em modo texto. A grande vantagem do SSH (Secure Shell)[5.75] é o grande suporte que possui: em listas de discussão[5.54.1], profissionais, documentações, News Group[5.58] e diversas implementações.

Uma outra solução seria o SSLTelnet, pouco usado em função do pequeno suporte que existe para este software.

1.1 - SSH (Secure Shell):

Basicamente o software SSH[5.75] é Free implementado para suportar os protocolos[5.65] SSH1[5.75] e SSH2[5.75], sendo que algumas implementações suportam os dois protocolos[5.65] simultâneos, como é o caso do openssh na versão 2.3.0 que será apresentada em seguida.

As implementações de SSH usam como padrão a porta 22/tcp do TCP/IP, possuem suporte para kerberos, autenticação TIS e socks. Outro ponto positivo é que alguns equipamentos como os roteadores cisco também possuem suporte para ssh[5.75] ou kerberos.

Entre os pontos responsáveis pela popularidade do SSH[5.75] podemos destacar : a sua instalação e configuração são muito simples, grande suporte técnico em listas de discussão[5.54.1], estabilidade, escalabilidade, segurança quando bem configurado, suportado por diversos sistemas operacionais (Linux, *BSD, Solaris, IRIX, Digital UNIX, AIX, IBM OS/2, SCO UNIX, HPUX, MAC/OS, Palm, Windows CE, VAX/OPENVMS, BeOS, MS-DOS e outros).

O SSH (www.ssh.fi) é escrito em C[5.13] ANSI[5.4], também possui implementações em java[5.52], é um software open source livremente distribuído. No site do SSH[5.75] (www.ssh.fi) podemos encontrar diversas informações como: relações e correções de Bugs[5.12], listas de discussão [5.54.1], novas implementações, acessos aos diversos mirros pela internet [5.45], como comprar clientes SSH [5.75] com recursos extras e etc.. . Inicialmente abordaremos o SSH1[5.75] (www.ssh.fi) e depois o openssh (www.openssh.com) que possui suporte ao ssh1[5.75] e ssh2[5.75].

O SSH2[5.75] possui maior flexibilidade, melhor escalabilidade e uma maior segurança. O SSH2[5.75] atualmente é menos usado que o SSH1[5.75], como ambos são incompatíveis as instituições/empresas que precisam se comunicar com outras tem que utilizar o SSH1[5.75].

Principais recursos para obtermos informações sobre SSH[5.75]:

Principais sites	http://www.ssh.com http://www.ssh.org http://www.ssh.fi
Lista de discussão	ietf-ssh@clinet.fi
FAQ's	http://www.employees.org/~satch/ssh/faq/ssh-faq.html http://www.tigerlair.com/ssh/faq/ssh-faq.html

2 - Instalação do SSH (secure Shell)

2.1 - Sistema operacional e software usado

Sistema operacional UNIX[5.80.1] FreeBSD[5.30] 2.2.8-RELEASE 32 bits.

Usaremos o software ssh-1.2.27.tar.gz versão 1.2.27

2.2 - Requesitos para instalar o software

- Compilador[5.19] gcc[5.35.1] (ANSI)[5.4], versão 2.7.2.1 ou superior e o GNU make;

Softwares	Site
GCC	ftp://ftp.gnu.org/gnu/make/
Make	ftp://ftp.gnu.org/gnu/make/

- Compactador[5.18.1] gunzip[5.38] e gzip[5.36];
- Programa tar[5.77];
- Comandos básicos como: **chmod**, **chown**, **chgrp** e **vi**;
- Software de compactação **gzip** e **gunzip**
- Conhecimento básico de UNIX[5.80.1] ou LINUX[5.54];
- **Zlib** (necessário no pacote openssh);

Software	Site
ZLib	http://www.freesoftware.com/pub/infozip/zlib/

- OpenSSL 0.9.5a ou maior (necessário no pacote openssh);

Softwares	Site
Ssleay 0.9.0	ftp://ftp.apache-ssl.org/SSLeay/ http://www.apache-ssl.org/ http://www2.psy.uq.edu.au/~ftp/Crypto/ssleay
Openssl 0.9.4	http://www.openssl.org

- **PAM**, o OpenSSH pode utilizar o PAM (Pluggable Authentication Modules). Isto é opcional.

Software	Site
PAM	http://www.kernel.org/pub/linux/libs/pam/

2.3 - Instalação do software SSH

2.3.1 - Descompactação do software

- Inicialmente estar no diretório onde encontra –se o software SSH[5.75] compactado[5.18.1]:

```
$  cd /usr/local/
```

- Usar o compactador "tar"[5.77] existente no UNIX[5.80.1]:

```
$  tar -xvzf ssh-1.2.27.tar.gz
$  chown –R 0 ssh-1.2.27
$ chgrp –R 0 ssh-1.2.27
```

- Entrar no diretório onde o software foi descompactado[5.18.1]

```
$ cd ssh-1.2.27
$PATH=$PATH:/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin:/usr/local/lib:/usr/lib:/usr/local/libexec:/usr/libexec:/usr/X11R6:
/usr/X11R6/bin:/usr/include:/usr/local/include
```

2.3.2 - Processo de Instalação

- Executar os "**script**" que preparará o software de acordo com o ambiente do sistema operacional:

```
$ ./configure \
--prefix=/usr/local/ssh1 \
--exec-prefix=/usr/local/ssh1 \
 --with-etcdir=/usr/local/ssh1/etc \
--disable-server-port-forwardings \
--disable-client-port-forwardings \
--disable-server-x11-forwarding \
--disable-client-x11-forwarding \
--disable-suid-ssh \
--enable-warnings \
```

Caso deseje retirar o label da versão do SSH[5.75] edite:

Para o SSH1[5.75] edite **version.h**

```
#define SSH_VERSION    "Versão clean"
```

Obs: Não podemos alterar a versão do protocolo[5.65].

Para o SSH2[5.75] edite **apps/ssh/ssh2version.h**

```
#define SSH2_VERSION "Versão clean"
```

Obs: Não podemos alterar a versão do protocolo[5.65].

- Iniciar o processo de compilação[5.19]:

```
$ make
```

- Iniciar o processo de instalação:

```
$ make install
```

2.3.3 - Listagem dos arquivos que compõe a instalação do SSH

Esta listagem mostrará a localização e permissões dos arquivos que compõem o software ssh-1.2.27 [5.75].

- Arquivos em **/usr/local/ssh1/bin/**

```
-rwxr-xr-x  1 root  wheel    21234 Sep 13 01:44 make-ssh-known-hosts1
-rwxr-xr-x  1 root  wheel    21234 Sep 13 01:44 make-ssh-known-hosts1.old
-rwxr-xr-x  1 root  wheel    70291 Sep 13 01:44 scp1
-rwxr-xr-x  1 root  wheel    70291 Sep 13 01:44 scp1.old
-rwxr-xr-x  1 root  wheel   604365 Sep 13 01:44 ssh-add1
-rwxr-xr-x  1 root  wheel   604365 Sep 13 01:44 ssh-add1.old
-rwxr-xr-x  1 root  wheel   611038 Sep 13 01:44 ssh-agent1
-rwxr-xr-x  1 root  wheel   611038 Sep 13 01:44 ssh-agent1.old
-rwxr-xr-x  1 root  wheel    67267 Sep 13 01:44 ssh-askpass1
-rwxr-xr-x  1 root  wheel    67267 Sep 13 01:44 ssh-askpass1.old
-rwxr-xr-x  1 root  wheel   580160 Sep 13 01:44 ssh-keygen1
-rwxr-xr-x  1 root  wheel   580160 Sep 13 01:44 ssh-keygen1.old
-rwx--x--x  1 root  wheel  1059554 Sep 13 01:44 ssh1
-rwxr-xr-x  1 root  wheel  1059554 Sep 13 01:43 ssh1.old
```

- Arquivos em **/usr/local/ssh1/sbin/**

```
-rwxr-xr-x  1 root  wheel  1150579 Sep 13 01:44 sshd1
-rwxr-xr-x  1 root  wheel  1150579 Sep 13 01:44 sshd1.old
```

- Arquivos em **/usr/local/ssh1/etc/**

```
-rw-r--r--  1 root  wheel  880 Sep 13 01:43 ssh_config
-rw-------  1 root  wheel  525 Sep 13 01:43 ssh_host_key
-rw-r--r--  1 root  wheel  329 Sep 13 01:43 ssh_host_key.pub
-rw-r--r--  1 root  wheel  713 Sep 13 01:43 sshd_config
```

OBS: Muito cuidado com as permissões dos arquivos que guardam a chave privada[5.16] e pública[5.15] ssh_host_key e ssh_host_key.pub, respectivamente .

```
-rw-r--r--  1 root  wheel  880 Sep 13 01:43 ssh_config
(Configuração do ssh cliente )
-rw-------  1 root  wheel  525 Sep 13 01:43 ssh_host_key
(Chave privada )
-rw-r--r--  1 root  wheel  329 Sep 13 01:43 ssh_host_key.pub
(Chave pública )
-rw-r--r--  1 root  wheel  713 Sep 13 01:43 sshd_config
(Configuração do ssh server )
```

OBS: O ssh_host_key (chave privada[5.16]) deve estar com a permissão 700. E os outros com **group** e **owner** 0 (zero).

- Manuais em **/usr/local/ssh1/man/**

```
scp1 (1)
ssh-add1 (1)
ssh-agent1(1)
ssh-keygen1 (1)
ssh1 (1)
sshd1 (8)
```

- Gerar a chave pública[5.15] do RSA para o root [5.70]:

```
$ ssh-keygen
Initializing random number generator...
Generating p:  .............................++ (distance 300)
Generating q:  .....++ (distance 44)
Computing the keys...
Testing the keys...
Key generation complete.
Enter file in which to save the key (/root/.ssh/identity):
Enter passphrase:
Enter the same passphrase again:
Your identification has been saved in /root/.ssh/identity.
Your public key is:
1024 33
122041540105743738466247270164459741254649003133478037917114363906169
255223736318200852714780222532530899068032827019319478457540300458993
919173353454503296310397555496054636454911974064081326508498333101481
485140018382098576727396395597068402554171381032119263698148383855017
118827293308030778944617748524827 root@cbpf.br
Your public key has been saved in /root/.ssh/identity.pub
```

- Sugestão para o **/usr/local/ssh1/etc/sshd_config**.

O arquivo sshd_config possui definições de segurança importantes que serão mostradas abaixo.

O servidor usado como exemplo possui o ip[5.48] 10.10.10.2/24 .

```
Port 22
( Porta do ssh )
ListenAddress 10.10.10.2
( endereço ip onde o socket estará funcionando )
HostKey /usr/local/ssh1/etc/ssh_host_key
(Chave privada do sshd)
RandomSeed /usr/local/ssh1/etc/ssh_random_seed
ServerKeyBits 768
( Define o número de bits da chave do server )
LoginGraceTime 600
( tempo de espera do sshd até o sucesso do login )
KeyRegenerationInterval 3600
( Define o tempo em segundos de regeneração da chave de encriptação, diminuindo esse tempo podemos evitar que a secção seja capturada e desencripitada )
PermitRootLogin no
( Não permite o login do root )
IgnoreRhosts yes
( Ignora ou não a autenticação do rhosts e shosts )
StrictModes yes
```

QuietMode no
X11Forwarding no
(Habilita o uso de X11 forwarding)
X11DisplayOffset 10
FascistLogging no
(Habilita o logging verbose)
PrintMotd yes
(Define se o sshd mostra o /etc/motd)
KeepAlive yes
SyslogFacility DAEMON
(habilita o código usado no syslog, podemos usar o código AUTH)
RhostsAuthentication no
(Habilita a autenticção pelo rhosts ou /etc/hosts.equiv)
RhostsRSAAuthentication no
(Habilita a autenticção pelo rhosts ou /etc/hosts.equiv usando o método de autenticação RSA)
RSAAuthentication yes
(Especifica se o método de autenticação RSA é aceito)
PasswordAuthentication yes
PermitEmptyPasswords no
(permite logon com senhas nulas ou não, recomendação "NO")
UseLogin no
 # CheckMail no
 # PidFile /u/zappa/.ssh/pid
(Path do arquivo que armazenara o número do processo sshd)
#AllowUsers aaa
(Define que o user "aaa" tem permissão de efetuar login)
DenyUsers anderson
(Nega o login do user anderson)
#AllowGroups user
(Permite o acesso do grupo user definido em /etc/group)
#DenyGroups user
(nega o acesso do grupo user definido em /etc/group)
*AllowHosts 10.10.10.**
(permite o acesso da rede 10.10.10./24)*
*#DenyHosts *.*.*.* lowsecurity.theirs.com *.evil.org evil.org*
(Nega acesso dos hosts e redes especificados)
 # Umask 022
 # SilentDeny yes

2.3.4 - Como iniciar o sshd[5.75]:

```
# ------------  inicio do script ----------------
#!/bin/sh
#   start SSHD
if [ ! -d /usr/local/ssh1/sbin/sshd ]
# Se o daemon sshd existir, este será executado
then
        /usr/local/ssh1/sbin/sshd &
fi
# ------------  fim  do script ----------------
```

OBS: No freebsd[5.30] podemos adicionar este script em **/etc/rc.local** .

Outra forma de iniciar é através do **/etc/inetd.conf** adicionando esta linha:

```
Ssh  stream  tcp nowait  root  /usr/local/ssh1/sbin/sshd sshd –i
```

Onde:

SSH: é a porta e protocolo (tcp/udp) configurado no /etc/services;
TCP: é o protocolo;
Stream: modo de transmissão, “paralela”;
Nowait: Não aguarda o fechamento de um open_passivo de um cliente para receber outro open_passivo;
Root: é o user que iniciará o servidor;
/usr/local/ssh1/sbin/sshd: *linha executa para iniciar o Daemon;*
-i: *opções.*

OBS: Neste exemplo podemos usar o **tcp-wrapper** .

No **/etc/services** devemos colocar a linha:

```
Ssh          22/tcp   #Secure Shell Login
Ssh          22/udp   #Secure Shell Login
```

2.3.5 - Utilização do SSH cliente

Após ter configurado corretamente o servidor sshd[5.75] podemos testar o **login** através de um cliente ssh[5.75] no unix[5.80.1]. Apresentaremos as formas básicas de utilização do ssh[5.75] cliente:

Exemplo 1:

```
Ssh –l <user> <host>
# /usr/local/ssh1/bin/ssh -l aaa 10.10.10.2
Host key not found from the list of known hosts.
Are you sure you want to continue connecting (yes/no)? Yes
Host '10.10.10.2' added to the list of known hosts.
aaa@10.10.10.2's password:
Last login: Mon Sep 13 05:21:06 1999 from 10.10.10.1
Copyright (c) 1980, 1983, 1986, 1988, 1990, 1991, 1993, 1994
        The Regents of the University of California.  All rights reserved.
No mail.
$
```

Exemplo 2:

```
Ssh <user>@<host>
$ /usr/local/ssh1/bin/ssh  aaa@10.10.10.2
aaa@10.10.10.2's password:
Last login: Mon Sep 13 05:26:15 1999 from server1.aaa.com.
Copyright (c) 1980, 1983, 1986, 1988, 1990, 1991, 1993, 1994
        The Regents of the University of California.  All rights reserved.
No mail.
$
```

2.3.6 - Utilização do SCP cliente

- Uso do utilitário scp[5.75] usado para transferência de arquivos usando a autenticação encriptada.

Exemplo do comando “put” no FTP[5.34]:

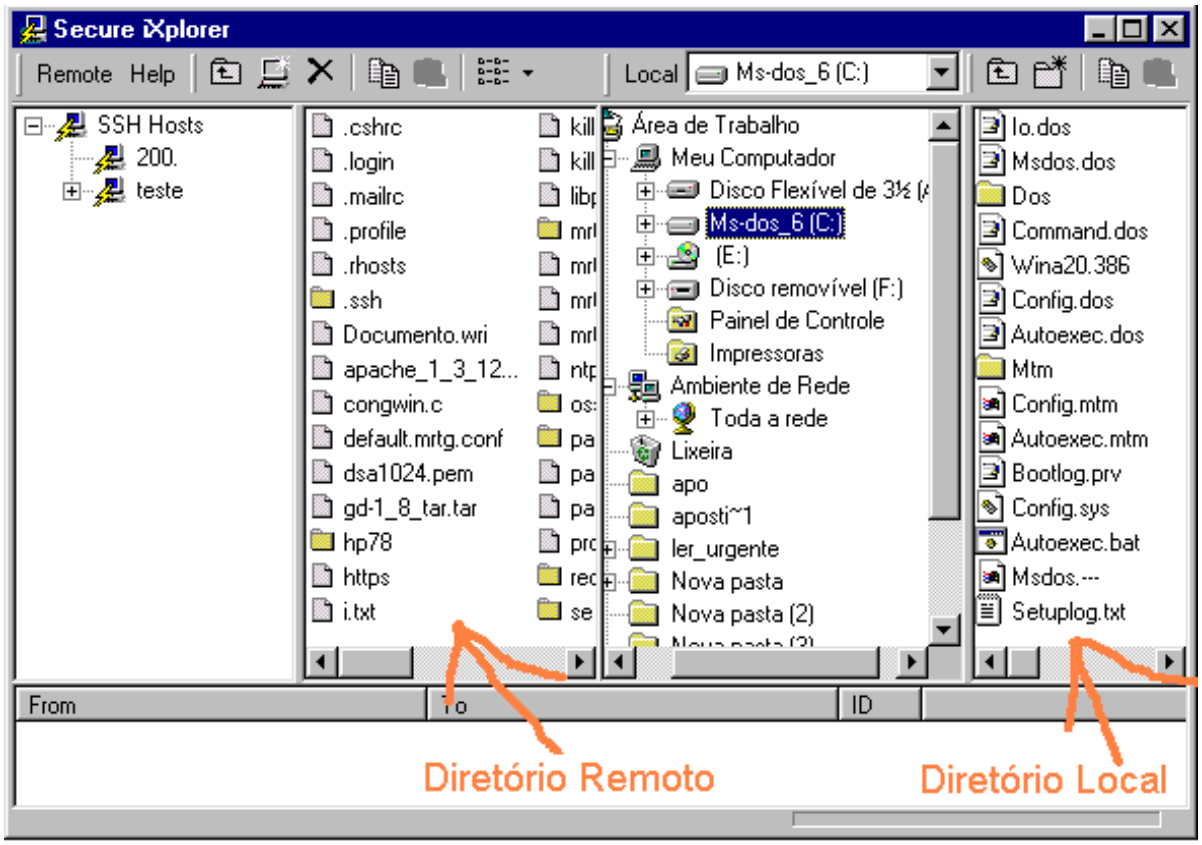
```
Scp <nome_completo_do_arquivo_de_transferência>
<login>@<maquina_remota_que>: / <diretorio_onde_o_arquivo será_gravado>

Scp /home/aaa/relatorio.txt aaa@10.10.10.2:/home/aaa/arquivos/
(Neste exemplo o arquivo /home/aaa/relatorio.txt da máquina local será transferido
para o diretório /home/aaa/arquivos/ da maquina remota 10.10.10.2 )
```

Exemplo do comando “get” no FTP[5.34]:

Scp <login>@<maquina_remota_que>: / <nome_completo_do_arquivo_de_transferência> <diretorio_onde_o_arquivo será_gravado>

Scp aaa@10.10.10.2:/home/aaa/relatorio.txt /home/aaa/arquivos/
(Neste exemplo o arquivo /home/aaa/relatorio.txt que está na máquina remota 10.10.10.2 será transferido para o diretório /home/aaa/arquivos/ da maquina local)

- Uso do software ixplorer.zip (http://www.i-tree.org/) scp[5.75] gráfico para windows[5.83]:



2.3.7 - Forward de portas

Uma das opções do SSH[5.75] é criar um túnel encriptado entre o local host[5.41], e o host[5.41] remoto, e estabelecer a conexão de outro serviço entre os dois hosts[5.41] em cima deste túnel encriptado.

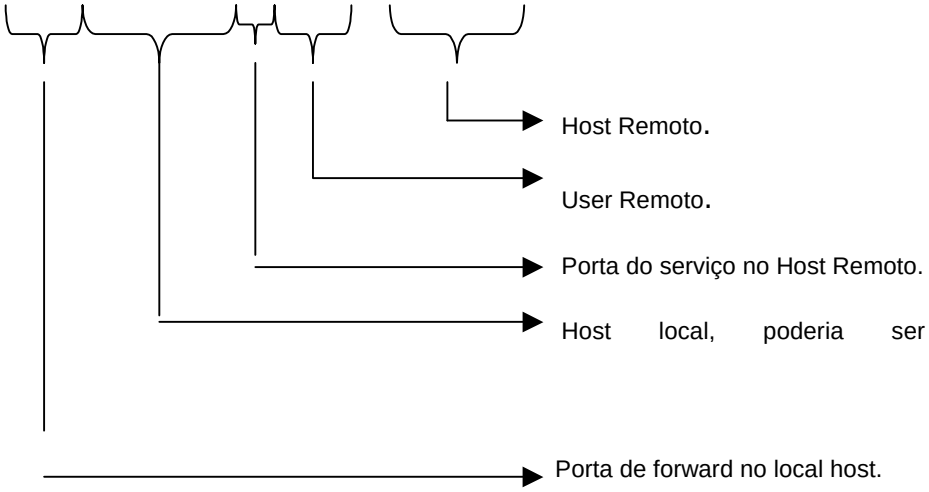
Isto pode ser usado para acessos usando serviços como: Telnet[5.79], FTP[5.34], POP3[5.64.1], **Xwindows**, **imap** e outros.

O exemplo abaixo será com o telnet[5.79], neste exemplo usaremos os dados:

Local host	10.10.10.1 ou 127.0.0.1
Host Remoto	10.10.10.2
Serviço Remoto	23 (telnet)
Porta local host	2233 (deve ser maior que 1024)

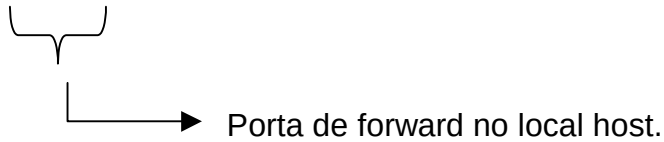
O primeiro passo é criar o telnet[5.79] encriptado, inicialmente escolheremos a porta local 2233 para o forward da conexão de telnet[5.79]. A senha do host[5.41] remoto será solicitada como se estivesse-mos em uma conexão ssh comum.

$ ssh -L 2233:10.10.10.1:23 aaa@10.10.10.2



O segundo passo é conectar na porta local escolhida para o forward (2233).

$ telnet 10.10.10.1 2233



Feito isso a transmissão de dados será encriptada, isto pode ser observado com sniffers[5.72.1]. Este método pode ser usado para outros serviços ditos inseguros, bastando apenas apontar a conexão para este serviço.

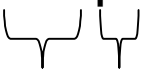
2.4 - Instalação do OpenSSH

Como foi dito acima o openssh é uma alternativa derivada do ssh 1.2.12 [5.75], basicamente o openssh possui uma pequena diferença na instalação que será mostrada abaixo. A configuração e utilização não possui grandes diferenças.

O desenvolvimento do openssh é baseado em rigorosos processos de segurança coordenado pelo grupo do OpenBSD[5.60]. Inicialmente o openssh foi desenvolvido pelo "the OpenBSD Project". Ele traz outros recursos de encriptação além do RSA e DES, como: **Blowfish**, **3DES**, **arcfour** e **cast128-cbc**.

E iniciado na versão 2.1.0 o openssh foi portado para outras plataformas, este pacote tar.gz é caracterizado pela letra p (portado) após a versão. Exemplo:

openssh-2.1.1p1.tar.gz



Versão Caracteriza um pacote portado para outras plataformas

Lista de discussão:

Listas	Subscribe mail
Geral	openssh-unix-announce@mindrot.org
Desenvolvedores	openssh-unix-dev@mindrot.org

News Group: comp.security.ssh
Archive List: http://marc.theaimsgroup.com/?l=secure-shell&r=1&w=2

Teremos inicialmente que instalar os pacotes: **Ssleay**, **openssl** e **zlib**. Esta instalação foi realizado em um linux[5.54] Slackware kernel 2.0.35.

2.4.1 - Instalação do Ssleay

```
$ tar -xvf SSLeay-0.9.0b.tar
$ cd SSLeay-0.9.0b
$ ./Configure  linux-elf
$ make
$ make install
$ find . -name  ssleay –print
  ./apps/ssleay
$ cp apps/ssleay /usr/local/bin/
# Copiar o binário ssleay para
# /usr/local/bin/ caso este
# exista na versão ssleay usada
```

2.4.2 - Instalação do Zlib

```
$ tar –xvzf zlib.tar.gz
$ cd zlib-1.1.3
$ ./configure
$ make
$ make install
$ make install
```

2.4.3 - Instalação do OpenSSL

```
$ tar -xvzf openssl-0.9.6.tar.gz
$ cd openssl-0.9.6
$ ./config
$ make
$ make test
$ make install
```

2.4.4 - Instalação do OpenSSH

```
$ tar –xvzf openssh-2.3.0p1.tar.gz
$ cd openssh-2.3.0p1
$./configure --enable-suid-ssh
$make
$make install
```

Caso deseje retirar o label da versão edite a linha abaixo em **version.h**:

```
#define SSH_VERSION     "Versão clean"
```

Obs: Não podemos alterar a versão do protocolo[5.65].

2.4.5 – Arquivos de configuração e binários

- Em **/usr/local/etc**:

```
$ ls -l /usr/local/etc/
total 7
-rw-r--r--   1 root     root          895 Nov 12 17:49 ssh_config
-rw-------   1 root     root          668 Nov 12 17:50 ssh_host_dsa_key
-rw-r--r--   1 root     root          600 Nov 12 17:50 ssh_host_dsa_key.pub
-rw-------   1 root     root          525 Nov 12 17:49 ssh_host_key
-rw-r--r--   1 root     root          329 Nov 12 17:49 ssh_host_key.pub
```

```
-rw-r--r--  1 root     root         1292 Nov 12 17:49 sshd_config
```

OBS: Cuidado com as permissões dos arquivos ssh_host_dsa_key e ssh_host_dsa_key.pub são os arquivos que armazenam as chaves privadas[5.16]e públicas[5.15] para o SSH2[5.75] que usa o DSA, respectivamente. E os arquivos ssh_host_key (chave privada[5.16]) e ssh_host_key.pub (chave pública[5.15]) para o ssh1[5.75] que usa o RSA.

- Binários gerados:

Daemon ssh[5.75], Pode ser iniciado standalone[5.74] ou através do inetd[5.47]. O ítem 2.3.4 descreve como inicia–lo.

```
$ ls -l /usr/local/sbin/sshd*
-rwxr-xr-x  1 root     root       630172 Nov 12 17:49 /usr/local/sbin/sshd
$
```

Cliente ssh[5.75], o item 2.3.5 descreve como usa–lo. O 2.3.6 descreve como usar o scp[5.75] para transferência de arquivos.

```
$ ls -l /usr/local/bin/ssh*   /usr/local/bin/scp /usr/local/bin/sftp*
-rwx--x--x   1 root     root       672556 Nov 12 17:48 /usr/local/bin/ssh
-rwxr-xr-x   1 root     root       529800 Nov 12 17:48 /usr/local/bin/ssh-add
-rwxr-xr-x   1 root     root       197448 Nov 12 17:48 /usr/local/bin/ssh-agent
-rwxr-xr-x   1 root     root       533664 Nov 12 17:48 /usr/local/bin/ssh-keygen
-rwxr-xr-x   1 root     root       4  Nov 12 17:48 /usr/servers/bin/scp -> scp2
-rwxr-xr-x   1 root     root       1153356 Nov 12 17:48 /usr/servers/bin/scp2
lrwxr-xr-x  1 root  wheel        5 Nov 12 17:48 /usr/local/bin/sftp -> sftp2
lrwxr-xr-x  1 root  wheel       12 Nov 12 17:48 /usr/local/bin/sftp-server -> sftp-server2
-rwxr-xr-x  1 root  wheel   550538 Nov 12 17:48 /usr/local/bin/sftp-server2
-rwxr-xr-x  1 root  wheel   696335 Nov 12 17:48 /usr/local/bin/sftp-server2.static
-rwxr-xr-x  1 root  wheel  1316683 Nov 12 17:48 /usr/local/bin/sftp2
$
```

- Arquivo de configuração do sshd[5.75]:

Arquivo de configuração **sshd_config** localizado em **/usr/local/etc/** .

```
$ more /usr/local/etc/sshd_config
Port 22
# Define a porta usada pelo sshd
#Protocol 2,1
# Define o protocolo ssh1 ou ssh2
ListenAddress 192.10.10.10
# Define o ip onde será “montado“ o servidor sshd
HostKey /usr/local/etc/ssh_host_key
# Localização da chave privada
ServerKeyBits 768
# Tamanho da chave privada
LoginGraceTime 600
```

```
# Tempo de espiração do login sem autenticação
KeyRegenerationInterval 3600
# Tempo de espiração da chave
PermitRootLogin no
# Não permite que o root log remotamente
IgnoreRhosts yes
# ignora as definições nos arquivos .rhosts
RhostsRSAAuthentication no
# Não permite autenticação RSA para as definições nos .rhosts
StrictModes yes
X11Forwarding no
# Não permite forward de sessões xwindows
X11DisplayOffset 10
PrintMotd yes
KeepAlive yes
SyslogFacility AUTH
# Define que serão logados via “syslogd” os acessos e autenticações
LogLevel INFO
# Define nível de log “syslogd”
RhostsAuthentication no
# Não permite autenticação Rhosts
/usr/local/etc/ssh_known_hosts
# arquivo que guardará as chaves públicas do servidores
RhostsRSAAuthentication no
# Não permite rhosts rsa autenticação
RSAAuthentication yes
# Permite autenticação usando o algorítimo RSA
PasswordAuthentication yes
PermitEmptyPasswords no
# Não permite autenticação com user contendo senhas em branco
#SkeyAuthentication no
# Habilita suporte a S/Key
#KbdInteractiveAuthentication yes
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#AFSTokenPassing no
#KerberosTicketCleanup no
# Kerberos TGT Passing does only work with the AFS kaserver
#KerberosTgtPassing yes
# Habilita suporte ao  kerberos
CheckMail no
#UseLogin no
# Uncomment if you want to enable sftp
Subsystem      sftp    /usr/local/libexec/sftp-server
# habilita o acesso via sftp (Secure ftp)
#MaxStartups 10:30:60
$
```

A restrição de ips[5.27] acessando o seu openssh server pode ser feita através do **/etc/hosts/deny** e **/etc/hosts.allow**:

```
$ more /etc/hosts.deny
all:all
$ more /etc/hosts.allow
sshd:10.10.10.1 10.10.10.2
```

- Arquivo de configuração do ssh[5.75]:

Arquivo de configuração **ssh_config** localizado em **/usr/local/etc/** .

```
$ more /usr/local/etc/ssh_config
# Host *
#   ForwardAgent yes
  ForwardX11 no
# Permite foward de Xwindows
  RhostsAuthentication no
# Suporte a acesso .rhosts
 RhostsRSAAuthentication no
# Suporte a acesso .rhosts via algorítimo RSA
 RSAAuthentication yes
# Suporte ao algorítimo RSA
 PasswordAuthentication yes
 FallBackToRsh no
 UseRsh no
# Habilita uso de "rsh" via ssh
#   BatchMode no
 CheckHostIP yes
#   StrictHostKeyChecking no
 IdentityFile ~/.ssh/identity
# Guarda em ~/.ssh/identity as chaves públicas dos servidores
 Port 22
# Define a porta de acesso do ssh client
  Protocol 2,1
# Deifne o suporte aos protocolos ssh1 e ssh2
#   Cipher blowfish
# Suporte ao algorítimo blowfish
  EscapeChar ~
# Define a Tecla de escape, "exit"
$
```

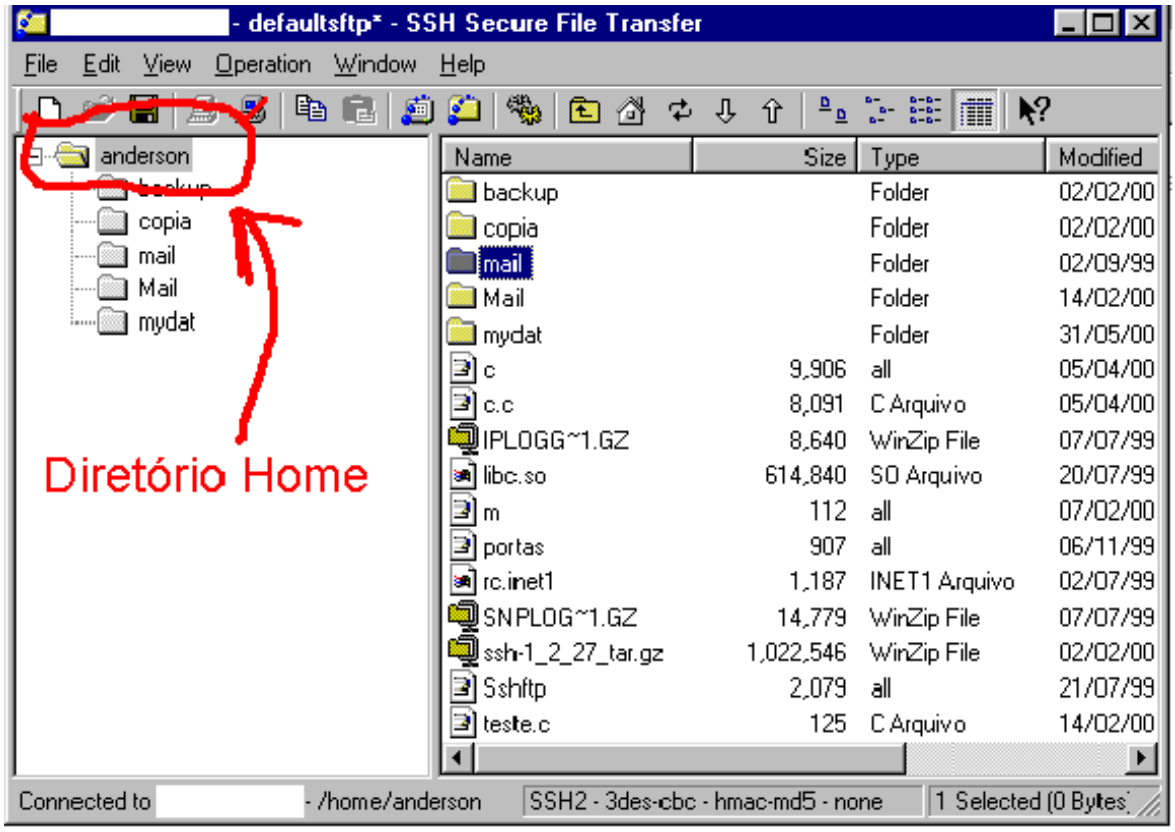
2.4.6 – Recursos do scp gráfico no protocolo SSH2

Este software é o ***SSHWin-versao.exe*** encontrado no site www.ssh.com, ele possui licença para uso acadêmico.

Basta descomentar a linha abaixo em **sshd_config**:

```
Subsystem      sftp    /usr/local/libexec/sftp-server
```

Após conectar o servidor notaremos que o usuário não pode descer de nível no seu diretório **home** do usuário, ficando restrito ao **home** diretório como acontece no FTP[5.34]. Outra diferença é a flexibilidade com o windows que o software **SSHWin-versao.exe** possui, podemos transferir arquivos como se estivesse-mos usando as facilidades de um software de FTP[5.34] tradicional para windows[5.83].



3 - Soluções de acesso remoto encriptado

3.1 - Servidor

- LSH/psst (implementação do **SSH2[5.75])**

http://www.net.lut.ac.uk/psst/

- **OpenSSH [5.75]**

Um projeto iniciado para o openbsd, compatível com a versão ssh1[5.75].

Este é um dos melhores SSH server[5.75], pois, possui suporte ao protocolo[5.65] SSH1[5.75]e SSH2[5.75], também é desenvolvido pelo mesmo grupo que coordena o desenvolvimento do OpenBSD[5.60]. Quando a transferências de arquivos é vital este software é o mais aconselhado ao lado do SSH2[5.75], também é aconselhado a utilização dos clientes F-Secure para SSH2[5.75] ou do site www.ssh.com para SSH2[5.75].
Site: http://www.openssh.com/

- OSSH

ftp://ftp.pdc.kth.se/pub/krypto/ossh/

- SSH1

Sites:
http://www.ssh.fi/
http://www.ssh.org/
http://www.datafellows.com

- **SSH2**

Sites:
http://www.ssh.fi/
http://www.ssh.org/

Quando a transferências de arquivos é vital este é o mais aconselhável ao lado do OpenSSH (com suporte a SSH2[5.75]), também é aconselhado a utilização dos clientes F-Secure para SSH2[5.75] ou do site www.ssh.com para SSH2[5.75].

- Telnet – SSL com SSLtelnet and MZtelnet

  ftp://ftp.uni-mainz.de/pub/internet/security/ssl/
  ftp://ftp.zedz.net/pub/replay/linux/redhat/.

- Windows NT Server

http://marvin.criadvantage.com/caspian/Software/SSHD-NT/default.php

3.2 - Clientes

- BeOS: The BeOS R4 port of SSH1 for Intel and PowerPC is

http://www.be.com/beware/Network/ssh.html

- "BetterTelnet 2.0bX with SSH patch"

http://www.cstone.net

- Cédric Gourio also developed a Java based SSH for his diploma

http://www.cl.cam.ac.uk/~fapp2/software/java-ssh

- DataFellows's "F-Secure SSH Client for Macintosh and Windows":

  Um dos melhores clientes SSH1[5.75] e SSH2[5.75] para windows[5.83], porém este software é pago.

http://www.datafellows.com

- DOS Cliente

http://www.vein.hu/~nagyd/#ssh

- Fresh Free FiSSH

http://www.massconfusion.com/ssh/.

- Fsh "Fast remote command execution"

http://www.lysator.liu.se/fsh/

- Gnome SSH Cliente

  Cliente SSH[5.75] com interface gráfica para **Gnome**.

http://zephyr.webhop.net/gnome-ssh.html

- Ixplorer – scp gráfico para windows[5.83]

http://www.i-tree.org/

- MacSSH (implementação do SSH2)

http://www.macssh.com/

- Mindterm (escrito em JAVA)

http://www.mindbright.se/mindterm/

- NSH

http://www.networkshell.com/

- "NiftyTelnet 1.1 SSH" (implemetação do SSH1[5.75])

http://www.lysator.liu.se
http://andrew2.andrew.cmu.edu

- OS/2

ftp://hobbes.nmsu.edu/pub/os2/apps/internet/telnet/client/ssh-1.2.27-b1.zip

- Palm Pilot

http://www.isaac.cs.berkeley.edu/pilot/

ftp.zedz.net/pub/crypto

- Putty

Este talvez seja o mais flexível cliente para SSH1[5.75] permite conectar nos mais diversos ssh1 servidor, somente superado pelo F-Secure que é pago. Porém não possui uma interface gráfica para o scp (para arquiteturas intel i386), ao contrário dos clientes ssh2 da F-Secure ou do site www.ssh.com. O software que implementa o scp (para arquiteturas intel i386) ainda é executado via linha de comandos e a sua versão gráfica atualmente não é muito agradável.

Este software também possui facilidade e rapidez na instalação, economizando bastante trabalho do suporte técnico.

http://www.chiark.greenend.org.uk/~sgtatham/putty/
ftp://ftp.chiark.greenend.org.uk/users/sgtatham/putty-latest/

scp para windows[5.83]

http://www.chiark.greenend.org.uk/~sgtatham/putty/
ftp://ftp.chiark.greenend.org.uk/users/sgtatham/putty-latest/

- Secure CRT

http://www.vandyke.com/

- SRP

http://www.kermit-project.org/k95.html
http://srp.stanford.edu/srp/

- SSH Plugin (escrito em JAVA)

http://www.mud.de/se/jta/doc/plugins/SSH.html

- SSH Win32 ports

http://guardian.htu.tuwien.ac.at/therapy/ssh/

- Telneat (implementação do SSH1[5.75] para windows[5.83])

http://telneat.lipetsk.ru/

- Tera Term Pro

www.zip.com.au/~roca/download.html
http://hp.vector.co.jp/authors/VA002416/teraterm.html

- TTSSH (implementação do SSH1 para windows)

http://www.zip.com.au/~roca/ttssh.html

- UNIX Client

http://www2.wiwi.uni-marburg.de/~leich/soft/secpanel/

- VAX/OpenVMS

http://www.er6.eng.ohio-state.edu/~jonesd/ssh
SSH1 client www.free.lp.se/fish

- Windows CE

http://www.movsoftware.com/sshce.htm

- Windows Client

http://www.pragmasys.com/SecureShell/

4 - Bibliografia:

Sites:

- BSD - “http://www.bsd.org”;
- Conectiva - “http://www.conectiva.com.br”;
- FreeBSD - “http://www.freebsd.org”;
- GNU – “http://www.gnu.org”;
- ISS - “http://www.iss.net”;
- LDP - “http://ldp.conectiva.com.br”;
- Linuxsecurity - “http://www.linuxsecurity.com.br”;
- Linux.com – “http://www.linux.com”;
- Linux.org - “http://www.linux.org”;
- Linux Unicamp - “http://www.linux.unicamp.br”;
- Linux Usp - “http://www.linux.usp.br”;
- NetBSD - “http://www.netbsd.org”;
- Net-Security - “http://www.net-security.org”;
- Olinux - “http://www.olinux.com.br”;
- OpenBSD – “http://www.openbsd.org”;
- OpenSSH - “http://www.openssh.org” e “http://www.ssh.com”;
- OpenSSL - “http://www.openssl.org”;
- RNP – “http://www.rnp.br”;
- SSH - “http://www.ssh.org”, “http://www.ssh.fi” e “http://www.ssh.com”;
- SSL - “http://www.ssl.org”;
- Unicamp - “http://www.security.unicamp.br”;

Livros:

- Computer Networks; Andrew S. Tanenbaum, Prentice Hall PTR, 1996;
- Interligação em rede com TCP/IP (Volume 1); Douglas E. Comer; Editora Campus, 1998;
- Sistemas Operacionais Modernos; Andrew S. Tanenbaum, Livros técnicos e científicos editora, 1992;
- Rede de computadores das: Lans, Mans e Wans às Redes ATM; Luiz Fernando Gomes Soares; Editora Campus, 1997;
- Segurança de dados e criptografia; José Ricardo Campelo Arruda; CEPUERJ, 1993;

Revistas:

- Security Magazine - “http://www.securitymagazine.com.br”;
- Revista do Linux - “http://www.revistadolinux.com.br”;

5 - Glossário:

[5.1] 802.3
Padrão IEEE para ethernet.

[5.2] ACK
Abreviatura para acknowledgement, confirmação.

[5.3] Active X
Ambiente de Programação baseado em programação por objetos, arquitetura compatível com o Microsoft Windows, criado pela microsoft; objetivando criar mais agilidades e opções de novas aplicações mais pesadas.

[5.4] ANSI (American National Standards Institute)
Grupo que define padrões nos EUA.

[5.5] ARP (Address Resolutions Protocol)
Vínculo entre o endereço IP e o endereço de nível físico.

[5.6] Apache
Servidor de HTTP desenvolvido pela "The Apache Software Foundation". Atualmente é o servidor de web mas usado na internet. Suportado por plataformas Linux, Unix, OS/2, Windows e etc....

[5.7] Applet
Uma página da www pode fazer referência a um programa escrito em *Java*. Desta forma, o programa é trascrito junto com a página. Chama-se este programa de *"Applet"*.

[5.8] Binário
Sistema numérico que utiliza a base 2.

[5.9] Bit
A menor unidade dentro de uma posição de memória.

[5.10] Browsers
São programas que possibilitam a visualizam de Home Pages.

[5.11] BSD
Linha (sabor) de unix desenvolvido na universidade de Berkley da California EUA. Possui basicamente quatro distribuições: BSDI, FreeBSD, NetBSD e OpenBSD.

[5.12] BUG
Expressão que caracteriza falha de programação em software ou em um sistema operacional.

[5.13] C
Linguagem de programação muito utilizada no meio científico, usada para escrever 99% do código do UNIX, Linux e outros sistemas operacionais. Pode ser usada em computadores de pequeno porte ou em super computadores.

[5.14] CGI (*Common Gaytway Interface)*
Programas que possibilitam real possibilidade de interagir com o usuário.

[5.15] Chave privada
Chave mantida em sigilo no sistema de criptografia assimétrico.

[5.16] Chave pública
Chave distribuida para o mundo.

[5.17] chroot
Possibilita a alteração da raiz do sistema tendo como referencial um o software que foi startado pelo chroot.

[5.18] Conectiva Linux
Empresa brasileira que distribui uma versão do linux baseada no linux Red Hat.

[5.18.1] Compactador
Software que dada uma entrada de dados "X" diminui a saída em "x" menos "y" de dados. E possuí a função inversas para restabelecer os dados de entrada posteriormente. Fator de compactação é ([X-Y]/X)*100 = Z%, onde desejamos que Z% esteja entre o intervalo aberto de (0,100).

[5.19] Compilador
Programa usado para traduzir uma informação em linguagem de máquina

[5.20] Criptografia
Arte de codificar e decodificar mensagens.

[5.21] Criptografia chave pública
Técnica de criptografia onde uma chave deve ser mantida em sigilo e a outra distribuida.

[5.22] Criptografia assimétrica
Algorítmo onde chaves diferentes são utilizadas no processo de criptografia de descriptografia.

[5.23] Criptografia simétrica
Algorítmo onde a mesma chave é utilizada no processo de criptografia de descriptografia.

[5.24] Decimal
Sistema numérico que utiliza a base 10.

[5.25] DNS (Domain name system)
Base de dados que converte endereço ips em nomes ou endereços ips em nomes. Porta 53/UDP. Também pode fornecer outras informações como: localização do servidor de e-mail, função dos servidores, sistema operacional dos servidores e outras informações.

[5.26] EAD
Vide ensino a distância.

[5.26.1] Encriptação
Processo pelo qual acontece a entrada de dados "X", este sofre alterações aplicando –se senhas e/ou chaves, obtém-se uma saída "Y" sem aparente significado para terceiros. A função inversa pode restabelecer o dado de entrada inicial "X".

[5.27] Endereço IP
Endereço com 32 bits (IPV4) , este caracteriza um host na rede. Também é dividido em redes classe A, B, C, D e E.

[5.28] Ensina a Distância
Utilização de recursos da internet para prover ensino, abramgendo desde o ensino fundamental até a pós-graduação.

[5.29] FDQN
Nome completo do host, composto de host mais dmain.

[5.30] FreeBSD
Sistema operacional baseado nos BSD de Berkley, voltado para as plataforma I386.

[5.31] Filtro de pacotes
Equipamento que encaminham ou descartam pacotes em função de regras pré-definidas.

[5.32] Filtro de pacotes Inteligente
Filtro de pacotes aprimorado, onde alguns problemas do filtro de pacotes são sanados.

[5.33] Firewall
Equipamento usado para proteger uma rede, pode ser dividido basicamente em firewall de filtro de pacotes, NAT, proxy (gateway de aplicação) e gateway de circuito.

[5.34] FTP (File Transfer Protocol)
Protocolo para transferência de arquivos, porta 21/tcp.

[5.35] Gateway
Equipamento responsável pelo roteamento de pacotes entre duas ou mais redes.

[5.35.1] Gcc
Vide linguagem "C".

[5.35.2]GNU
Software Livre, da Free Software Foudation (FSF).

[5.36] Gzip
Compactador muito usado no unix.

[5.37] Gopher
Serviço precursor do WWW.

[5.38] Gunzip
Descompactador muito usado no unix.

[5.39] Handshake Sequence
Sequência de comunicação do protocolo(s) que caracteriza(m) estabelecimento da conexão entre cliente e servidor.

[5.40] HOME PAGE
Um local na web que representa a presença de uma organização.

[5.41] HOST
Qualquer computador, equipamento, de usuário final que conecta à rede.

[5.42] HTTP (Hyper Text Transfer Protocol)
Protocolo da World Wide Web.

[5.43] ICMP (Internet Control Message Protocol)
Integrante do protocolo IP e lida com mensagens de controle e erro.

[5.44] IEEE 802.3
Vide 802.3 .

[5.45] Internet
Conjunto de hosts, redes e roteadores que utiliza o protocolo TCP/IP unindo mais de 61 paises. Formando uma única rede. **Intranet** – uma Internet local

[5.46] INTERNIC
Organização que fornece informações sobre serviços e protocolos. Também é responsável pelo registro de IP e nomes na internet.

[5.47] Inetd
Software que gerencia o processo inicializado por um servidor. Chamado de super servidor.

[5.48] IP (Internet Protocol)
Protocolo usado para identificar os computadores com endereços ips de origem e destino.

[5.49] IPV4
Versão atual do ip.

[5.50] IPV6
Versão da nova versão do IP.

[5.51] ISP
Internet Service Provider, Provedor de Serviço Internet.

[5.52] JAVA
Linguagem de script que independe da plataforma. Esta é uma poderosa linguagem orientada a objeto criada pela SUN.

[5.53] JAVA SCRIPT
É uma adaptação da *Java*, de forma a ser interpretada pelo Browser, sem necessidade de ser compilada pelo autor.

[5.54] Linux
Sistema operacional baseado no minix desenvolvido por Linus Trovald no início da década de 90. Hoje este sistema operacional possui diversas distribuições pelo mundo.

[5.54.1] Lista de discussão
Software que administra grupos de discussão. Este software atua em conjunto com servidor de E-mail.

[5.55] LOG
Armazenamento de ocorrências para realização de auditorias.

[5.56] Mapa Sensitivo
São figurar onde são definidas zonas através de coordenadas de forma a selecionar áreas que ao serem acionadas funcionam como links.

[5.57] NetBSD
Sistema operacional baseado nos BSD de Berkley, voltado aspectos de segurança e multiplataforma.

[5.58] NEWS
É composto por informações agrupadas por categorias

[5.59] NFS (Network file system)
Possibilita a compartilhamento de *file system* entre duas ou mais máquinas UNIX/Linux. Porta 2049/tcp e udp.

[5.60] OpenBSD
Sistema operacional baseado nos BSD de Berkley, voltado para aspectos de segurança. Possui grande suporte a criptografia, VPN e firewall. Considerado um dos sitemas mais seguros entre os sistemas free e pagos em ambiente standard.

[5.61] OS/2
Sistema operacional desenvolvido pela IBM voltado para o mercado coorporativo e doméstico.

[5.62] Pacote
Um conjunto de informações, dados e cabeçlhos, que são usados na transmissão de dados.

[5.63] Perl
Linguagem interpletada voltada para o processamento de texto, muito utilizada em CGI's na WEB. Também facilita a depuração de logs do sistema.

[5.64] Plugins
São módulos que podem ser conectados (instalados) ao browser, de forma que este passe a reconhecer os mais diversos formatos de informação. Diversas empresas criaram *Plugins*, permitindo assim ao browser visualizar gráficos vetoriais, vídeos, som, planilhas, *VRML, Flash* e outros.

[5.64.1] POP3
Serviço usado para o usuário coletar mensagens recebidas pelo servidor de E-Mail. Porta 110/TCP.

[5.65] Protocolo
Descrição formal das regras que possibilitam a troca de informações entre os dispositivos de rede.

[5.66] Provedores de Backbone Internet
São instituições que disponibilizam acesso a internet para redes locais.

[5.67] Raiz
Em UNIX é a conta que detêm o controle do sistema.

[5.68] Red Hat
Uma Distribuição do Linux.

[5.68.1] Rede de computadores
Coleção de computadores interligadas e autônoma.

[5.69] rlogin
Serviço de acesso remoto.

[5.70] Root
Raiz do sistema. Conta que possui o controle sobre o sistema.

[5.71] Sendmail
Implementação mais comum do SMTP.

[5.72] SMTP (Simple Mail Transport Protocolo)
Serviço de transmissão de E-mails. Porta 25/TCP.

[5.72.1] Sniffer
Software que coloca a placa de rede em modo promíscuo capturando todos os pacotes que passam no mesmo domínio de colisão.

[5.73] SSL (Secure Socks Layer)
Protocolo desenvolvido pela netscape que oferece serviço de segurança acima do protocolo TCP.

[5.74] Standalone
Um servidor é iniciado, ou processo, sem auxílio de algum software “gerenciador” como o inetd.

[5.75] SSH (Secure Shell)
Protocolo de acesso remoto que possui um canal encriptado. Possui as versões ssh1 e ssh2 estas versões de ssh utilizam algorítimo de encriptação RSA (SSH versão 1) e DAS (SSH versão 2) respectivamente. Porta 22/tcp.

[5.76] SQUID
Software proxy cache para acessos a WEB.

[5.77] Tar
Programa para concatenar arquivos muito usado em UNIX/Linux.

[5.78] TCP (Transmission Control Protocol)
Protocolo orientado a comunicação e atua na camada de transporte .

[5.79] TELNET
Protocolo que possibilita a emulação de terminal entre duas maquinas. Porta 23/tcp.

[5.80] UDP (User Datagram Protocol)
Protocolo não orientado a comunicação e atua na camada de transporte .

[5.80.1] UNIX
Sistema Operacional inicialmente desenvolvido por Ken Thompson no Bell Lab para o computador DEC-PDP7, na terceira geração (1965-1980). Marca registrada da AT&T.

[5.81] VPN (Virtual Private Netwaork)
Canal de comunicação encriptado gerado através de uma rede de ip pública não confiável.

[5.82] Wais
Sistema de banco de dados de acesso remoto usado na internet.

[5.83] Windows
Sistema operacional desenvolvido pela microsoft, possuidor de vários problemas de estabilidade e segurança. Mas, possui recursos que tornam o sistema amigável.

[5.84] WWW (Word Wide Web)
Recurso de hipermídia para acessar a internet. Desenvolvido pelo CERN.