Patrick
Haller

Critique of The Secure Shell

SSH Prep

People with wireless-enabled laptops are encouraged to follow along

How SSH Works

U

S

How SSH Kills Valkyries

```
$ cat ~/.ssh/known_hosts
```

```
$ ssh server_IP
```

Check mail, use w3m, or if the box has X setup, run an X app.

Remember to set up a VPN for when this happens next time. ;)

How SSH Ki^H^H^H Works

With public and private keys

Private keys are hard to fake

SSH only requires working TCP/IP

No other external network services

Crypto Math

Treasure Hunt Race:

Find house with key nailed to door

When you get back, you win

Private key has the house address

SSH Setup

```
$ ssh-keygen
  choose RSA 2048

$ eval `ssh-agent`

$ ssh-add

$ ssh-copy-id \
  -i ~/.ssh/id_rsa server
```

SSH Setup Verification

```
$ ssh-keygen -l \
  -f ~/.ssh/known_hosts
$ ssh server "ssh-keygen -l
  -f /etc/ssh/ssh_host_rsa_key.pub"
```

The fingerprints MUST match

Ideally, verify fingerprints off-line via PGP-signing the key

SSH Truck

Securing Services with SSH

CVS / Subversion

```
svn+ssh://server/path/to/repo
```

Rsync / Unison

```
unison this ssh://server/that
```

POP / SMTP

```
ssh server “sendmail -oi -oem”
```

Music Streaming

```
ssh server “cat mp3/*” | mplayer -
```

Tunnels with SSH

Through FW to internal server

```
ssh fw -L 1024:billing:443 sleep 3600&
firefox https://localhost:1024
```

SOCKS5 Proxy

```
ssh -D 5000 server sleep 3600 &
export HTTP_PROXY=localhost:5000
w3m msdn.microsoft.com/vbasic/
```

VPN can be set up with tun* devices

For SSL - use stunnel instead

Maintaining SSH

Unmanaged key churn is BAD

- backup sshd keys, put 'em back after upgrading!

Use good passphrases

- check out Randall William's Passphrase FAQ
- `$ ssh-keygen -p`

Use ssh-agent carefully!!!

- root can access your agent
- `# su -l -c “ssh secure_box” YOU`

Maintaining SSH

```
$ cat ~/.ssh/config
# don't accidentally connect to evil
StrictHostKeyChecking=yes

# verify DNS not poisoned
CheckHostIP=yes

# only use Protocol 2 (v1 is unsafe!!!)
Protocol 2

# compress while encrypting
Compression
```

~/.ssh/config

```
# hide where we connect
HashKnownHosts yes

# keep our sessions alive
TCPKeepAlive no
ServerAliveInterval 60
ServerAliveCountMax 1440

# send “~.” to break a session
EscapeChar ~
```

~/.ssh/config

```
# turn off unused features
ForwardAgent no
ForwardX11 no
ForwardX11Trusted no
HostbasedAuthentication no
RhostsRSAAuthentication no
VerifyHostKeyDNS no
GatewayPorts no
```

~/.ssh/config

```
# per host configuration follows
Host eisner.decus.org
        User me
        Port 2200
        HostKeyAlgorithms ssh-dss
        PreferredAuthentications publickey,password

# everyone else
Host *
Port 22
```

~/.ssh/config

```
# use only good ciphers and methods
Ciphers blowfish-cbc,aes256-cbc,aes256-ctr
HostKeyAlgorithms ssh-rsa
MACs hmac-md5,hmac-sha1,hmac-ripemd160
PreferredAuthentications \
    publickey,keyboard-interactive

# do not use weak crypto
# ssh-dsa is limited by FIPS to 1024 bits
# hmac-sha1-96, hmac-md5-96
# aes-small-cbc, 3des-small-cbc, cast-small-cbc....
```

www.OpenSSH.com

Putting an end to unencrypted network logins