Berner Fachhochschule
Haute école spécialisée bernoise
Bern University of Applied Sciences

OpenSSH

Linux User Group Meeting #3

Thursday, 7th August 2014 17:00 – 18:00

BFH – Room HG 303 (Quellgasse, Biel)

Daniel Baumann <daniel.baumann@bfh.ch>

IT System Engineer, Infrastructure Team

Overview

OpenSSH Basics

OpenSSH Advanced, Part 1

Other SSH Tools, Part 1

Other Operating Systems/Environments

Aperitif

OpenSSH Basics

Introduction

- Problem: Login on a remote system
- Solution: Secure Shell (SSH)

Installation

Client

- sudo apt-get install openssh-client

Server

- sudo apt-get install openssh-server

Server (optionally)

- sudo apt-get install openssh-blacklist openssh-blacklist-extra

Configuration

- Default configurations are workable
- As always defaults are defaults: they are not bad but also not very good at the same time

Usage

Remote Login

- ssh daniel@147.87.225.31

  ssh daniel@damm-bad9.bfh.ch

  ssh bad9@ssh.bfh.ch

  ssh sysadmin@lxc1.its.bfh.ch

  ...

- (or ssh -l daniel damm-bad9.bfh.ch if you use non-US keyboard and have to use Alt-GR+2 for @ :)

OpenSSH Advanced, Part 1

Use Debian (or Debian-based if you must)

Baseline

- We assume Debian 7 (wheezy) or Debian 8 (jessie)

Use sudo instead of root login

How?

- System-wide server configuration (/etc/ssh/sshd_config):

  # Disable Root Login

  PermitRootLogin no

Why?

- root is dangerous because it can do everything
- root exists on all machines therefore tried by everyone
- just use sudo instead

Use public key authentication instead of passwords (1/2)

How?

- System-wide server configuration (/etc/ssh/sshd_config):

  # Disable Password Authentication

  PasswordAuthentication no

- Create user key:

  mkdir -p ~/.ssh/keys

  ssh-kegen -t rsa -b 4096 -f ~/.ssh/keys/daniel.baumann@bfh.ch -C daniel.baumann@bfh.ch

- Copy user key to remote system (to ~/.ssh/authorized_keys):

  ssh-copy-id -i ~/.ssh/keys/daniel.baumann@bfh.ch daniel@damm-bad9.bfh.ch

- Use key to login on remote system:

  ssh -i ~/.ssh/keys/daniel-baumann@bfh.ch daniel@damm-bad9.bfh.ch

Use public key authentication instead of passwords (2/2)

How?

- Local client configuration (~/.ssh/config):

```
Host damm-bad9.bfh.ch
    Hostname damm-bad9.bfh.ch
    User daniel
    IdentityFile ~/.ssh/keys/daniel.baumann@bfh.ch
```

Why?

- passwords are weak
- passwords need to be remembered
- passwords are inconvenient

Use good passphrases on your keys

How?

- Change passphrase on user key:

  ssh-keygen -f ~/.ssh/keys/daniel.baumann@bfh.ch -p

- Load user key:

  ssh-add ~/.ssh/keys/daniel.baumann@bfh.ch

Why?

- Good passphrases are inconvenient to type all the time

Use strong cryptography

How?

- System-wide client configuration (/etc/ssh/ssh_config):

  #Use RSA for Host Keys only

  HostKeyAlgorithms ssh-rsa-cert-v01@openssh.com,ssh-rsa

- System-wide server configuration (/etc/ssh/sshd_config):

  #Use RSA for User Keys only (needs patch currently not in Debian)

  PubkeyTypes rsa

- #Do not use DSA and ECDSA and Host Keys (openssh-server on Debian will recreate them on package upgrades)

  sudo rm -f /etc/ssh/ssh_host_*dsa_key*

  sudo sed -i -e 's|^\(HostKey .*dsa_key$\)|#\1|' /etc/ssh/sshd_config

Why?

- DSA is broken, ECDSA is obsolete, and ED25519 is not mature yet.

Trust host certificates instead of host keys (1/2)

For System Administrators

- Create a new CA key:

  ssh-keygen -N "" -t rsa -b 4096 -f CA_KEYFILE -C COMMENT

- Create a host key:

  ssh-keygen -N "" -t rsa -b 4096 -f HOST_KEYFILE -C COMMENT

- Sign host key with CA:

  ssh-keygen -s CA_KEYFILE -I CA_IDSTRING -h -n HOSTNAME,HOSTNAME2 -V +123d HOST_KEYFILE.pub

- Use host key certificate in /etc/ssh/sshd_config:

  HostKeyCertificate /etc/ssh/sshd_rsa_key-cert.pub

Trust host certificates instead of host keys (2/2)

For Users

- Local client configuration (~/.ssh/known_hosts):

  @cert-authority *.bfh.ch ssh-rsa [...] its-linux-sysadmin@lists.bfh.ch

Why?

- Host key deployments are unreliable, untimely and anoying to deploy
- Allows System Administration to change host keys as they like without user anoyance
- Users can be instructed to distrust in unsigned host keys (with DNSSEC they can be even have conections refused)

Strict host key checking (1/2)

How?

- System-wide client configuration (/etc/ssh/ssh_config):

  #Use DNS and SSHFP ressource records for Host Key Verification if available (only useful with DNSSEC)

  VerifyHostKeyDNS ask

  #Abort connection attempt if host key doesn't match

  StrictHostKeyChecking yes

- System-wide server known_hosts:

  cat > /etc/ssh/ssh_known_hosts << EOF

  @cert-authority *.example.org ssh-rsa [...]

  EOF

Strict host key checking (2/2)

Avoid fail2ban/ssh-guard/etc traps

- Local client configuration (~/.ssh/config):

```
# Use configured Identities only
IdentitiesOnly yes
```

What if I test stuff a lot and don't care about anything?

- Local client configuration (~/.ssh/config):

```
Host *.test
    StrictHostKeyChecking no
    UserKnownHostsFile /dev/null
    VerifyHostKeyDNS no
```

Why?

- Keep connections binary – either they are good or they are bad

Use persistent connections

Local Configuration (~/.ssh/config)

- Use shared connections:

mkdir -p ~/.ssh/sockets

```
Host *
    ControlMaster auto
    ControlPath ~/.ssh/sockets/%r@%h
    ControlPersist 30m
```

SSH Gateway

Local Configuration (~/.ssh/config)

- SSH Gatway:

```
Host ssh.bfh.ch
    Hostname ssh.bfh.ch
    User bad9
    IdentityFile ~/.ssh/keys/daniel.baumann@bfh.ch

Host damm-bad9.bfh.ch_extern
    Hostname damm-bad9.bfh.ch
    User daniel
    ProxyCommand ssh -W %h:%p ssh.bfh.ch
```

Forwarding aka „You don't need VPN for that“

- **X11**

  ssh -X damm-bad9.bfh.ch

- **Firefox**
- Socks Proxy: ssh -TN -D 8080 ssh.bfh.ch
- Install FoxyProxy Basic from addons.mozilla.org
- Add a new proxy: Host: localhost; Port: 8080; SOCKS v5 proxy

Thunderbird (or any Mail Client)

- Port Forwarding Tunnel: ssh -TN -L 2525:ssh.bfh.ch:25 ssh.bfh.ch
- Configure your mailclient to use localhost:2525 with the same credentials you'd use otherwise
- Note: not sure if that actually works with the BFH mailserver setup, but will work with your own mailserver for sure :)

Other Stuff (1/3)

System-wide server configuration (/etc/ssh/sshd_config)

- # Disable Message of the Day

  PrintMotd no

- # Disable distribution-specified extra version suffix

  DebianBanner no

- # Reject user specified locales

  AcceptEnv

- **Local client configuration (~/.ssh/config)**

  # Don't send user specific locales

  SendEnv

Other Stuff (2/3)

System-wide Configuration (/etc/ssh/sshd_config)

- Useful on servers with untrustworthy users or central authentication,

  system-wide server configuration file (/etc/ssh/sshd_config):

  # Use system-wide authorized_keys

  AuthorizedKeysFile /etc/ssh/authorized_keys/%u %h/.ssh/authorized_keys

Local Configuration (~/.ssh/config)

- mkdir -p ~/.ssh/configs
- # Use Configuration File Includes (requires a patch not included in Debian)

  Include ~/.ssh/configs/bfh.ch

Local Configuration (~/.ssh/authorized_keys)

- command="COMMAND",no-port-forwarding,no-X11-forwarding,no-agent-forwarding,no-pty ssh-rsa [...]

Other Stuff (3/3)

Logging User Logins

- System-wide server configuration (/etc/ssh/sshd_config):

  LogLevel VERBOSE

  Rsyslog configuration:

  cat > /etc/rsyslog.d/openssh-server.conf << EOF

  :msg, regex, "Accepted publickey for .*" -/var/log/openssh-server.log

  :msg, regex, "Found matching .* key:" -/var/log/openssh-server.log

  EOF

SSH Tools, Part 1

SSH Tools

Auto SSH

- Reopens connections automatically

Bash Completion

- Command and hostname completion with <Tab><Tab>

- **Screen**

  Terminal Multiplexor

- **SSHFS**

  Access files on remote system through a (FUSE-based) filesystem mount, alternative to scp

Other Operating Systems/Environments

Android

OpenSSH

- Connectbot: http://code.google.com/p/connectbot/

OS X

OpenSSH

- Server and Client: openssh from http://macports.org

SSH Tools

- SSH „GUI“ Client: putty from http://macports.org
- SFTP Client: http://cyberduck.io
- SFTP Integration: sshfs from http://macports.org, Macfusion http://macfusionapp.org/, or http://expandrive.com (commercial)

Windows

OpenSSH

- Server and Client: http://www.mls-software.com/opensshd.html

SSH Tools

- SSH „GUI“ Client: http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
- SFTP Client: http://winscp.net
- SFTP Integration: http://swish-sftp.org
- X11 Integration: http://www.straightrunning.com/XmingNotes

Future

- Currently SSH user keys need to be send once to its-linux-sysadmin@lists.bfh.ch for various services (login, Git, etc.)

- We'd like to have user keys in LDAP via DirX (https://selfhelp.bfh.ch)
- We'd like to have two-factor auth for servers (yubikey OTP)

- We can have a „OpenSSH, part 2“ meeting if desired

Thank You for Your Attention.

❤ Source Code is freely available

Further Information

BFH

- https://linux.bfh.ch/software/openssh
- https://linux.bfh.ch/services/openssh
- https://linux.bfh.ch/software/screen
- ...abd bfh-linux-users@lists.bfh.ch as usual ;)

Manual pages

- man ssh
- man ssh_config, sshd_config