OpenSSH nicht-interaktiv

Robert Schulze <rob@rob-schulze.de>
17.08.2011

SSH

- **S**ecure **Sh**ell
- verschlüsselter Ersatz für telnet/rlogin/rsh/rexec
- Client (ssh/scp) und Server (sshd)
- Authentifizierung über Login/Passwort oder Schlüsselpaare
- Host-Authentifizierung

Login vs. Schlüssel

- Login oft an Systemaccount gebunden
  - Niemals Passwörter doppelt benutzen
- Bruteforce, social engineering usw.

- One key(-pair) to rule them all
- Nicht-interaktives viel simpler
- Allgemein als sicherer betrachtet

Schlüsselpaar erzeugen

```
rob@rob-desktop:~$ ssh-keygen

Generating public/private rsa key pair.

Enter file in which to save the key (/home/rob/.ssh/id_rsa):

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Your identification has been saved in /home/rob/.ssh/id_rsa.

Your public key has been saved in /home/rob/.ssh/id_rsa.pub.

The key fingerprint is:

e0:29:fe:2c:2f:59:cf:12:09:0b:aa:d3:fd:e7:37:da rob@rob-
desktop
```

Passphrase ändern

```
rob@rob-desktop:~$ ssh-keygen -p
Enter file in which the key is (/home/rob/.ssh/id_rsa):
Key has comment '/home/rob/.ssh/id_rsa'
Enter new passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved with the new
passphrase.
```

Schlüssel exportieren

- ~/.ssh/authorized_keys
- Eine Zeile pro **öffentlichem** Schlüssel
- Optionen für jeden Schlüssel möglich

Optionen

- Schlüssel darf nur von bestimmten Hosts benutzt werden

```
# ~/.ssh/authorized_keys
from="10.0.0.*" ssh-rsa ...
```

Optionen

- Schlüssel darf nur für ein bestimmtes Kommando benutzt werden
- Kein Login möglich

```
# ~/.ssh/authorized_keys
command="<cmd>" ssh-rsa ...
```

Beispiel: Systemaccouting

- Mehrere Server mit vielen Usern
- CPU-Zeit und IO-Operationen pro User auswerten können
- "pigs" identifizieren

Beispiel: Systemaccounting

- BSD: /usr/sbin/sa -m
- Benutzername, Anzahl Kommandos, CPU-Zeit, IO-Operationen, CPU-Storage-Integral
- Von einem Rechner aus mit spezifischem Schlüssel verbinden
- Schlüssel an Kommando gebunden
- Standardausgabe auswerten und speichern

Beispiel: Systemaccounting

```
user1  204    0.44cpu   0tio      76346k*sec
User2  141    0.23cpu   18tio     81464k*sec
User23 7      0.07cpu   103tio    2982k*sec
User42 125    0.63cpu   66tio     77268k*sec
User12 131    0.48cpu   3016tio   52396k*sec
```

Beispiel: Systemaccounting

- /root/.ssh/authorized_keys:

```
command="/usr/sbin/sa -smi" ssh-rsa ...
```

- Aufruf für jeden Host:

```
$ ssh -i id_sa_rsa root@host
```

- Besser: normaler Systemaccount + sudo

Literatur

- http://www.openssh.org/
- ssh(1): AUTHENTICATION
- sshd(8): AUTHORIZED_KEYS FILE FORMAT