15. Sicherheit - Kommunikation mit SSH

Secure Shell

Merkmale:
- Verschlüsselte Login-Übertragung
- Verschlüsselte Datenübertragung
- Public-Key-Verfahren
- RSA-Schlüssel
- DSA-Schlüssel
- Nutzer-Schlüssel
- Host-Schlüssel

Protokolle:
- SSH-Protokoll Version 1
- SSH-Protokoll Version 2

Vorteile:
- Offene Quellen
- Kaum noch Sicherheitslöcher

Windows-Installation:

PuTTY (Quelle: http://www.chiark.greenend.org.uk/~sgtatham/putty/)
oder
WinSCP (Quelle: http://winscp.net/

```
UNIX-Installation:

Zentrale Konfigurationsfiles:

   in /etc/ssh oder /etc/openssh
            moduli
            ssh_config            - Konfigurationsfile für ssh-Client
            ssh_host_dsa_key      - dsa-Host-Key geheim
            ssh_host_dsa_key.pub  - dsa-Host-Key öffentlich
            ssh_host_key          - Host-Key geheim
            ssh_host_key.pub      - Host-Key öffentlich
            ssh_host_rsa_key      - Host-Key geheim
            ssh_host_rsa_key.pub  - Host-Key öffentlich
            sshd_config           - Konfigurationsfile für sshd-Server

Private Konfigurationsfiles:

 in  ~/.ssh:

   authorized_keys   - Zugelassene Keys
   authorized_keys2
   known_hosts       - bekannte Host-Keys
   known_hosts2
   id_dsa            - privater DSA-Key
   id_dsa.pub        - öffentlicher DSA-Key
   id_rsa            - privater RSA-Key
   id_rsa.pub        - öffentlicher RSA-Key
   identity          - privater Key
   identity.pub      - öffentlicher Key
```



```
ssh - OpenSSH SSH client (remote login program)

     ssh [-l login_name] hostname | user@hostname [command]

     ssh [-afgknqstvxACNTVX1246] [-b bind_address] [-c cipher_spec]
         [-e escape_char] [-i identity_file] [-l login_name] [-m mac_spec]
         [-o option] [-p port] [-F configfile] [-L port:host:hostport]
         [-R port:host:hostport] [-D port] hostname | user@hostname [command]

       Beispiel:
         ssh tbell@amsel


scp - secure copy (remote file copy program)

     scp [-pqrvBC1246] [-F ssh_config] [-S program] [-P port] [-c cipher]
         [-i identity_file] [-l limit] [-o ssh_option] [[user@]host1:]file1
         [...] [[user@]host2:]file2

       Beispiel:
         scp -r * bell@data:tmp
```

```
sftp  - secure file transfer program

     sftp [-vC1] [-b batchfile] [-o ssh_option] [-s subsystem | sftp_server]
          [-B buffer_size] [-F ssh_config] [-P sftp_server path]
          [-R num_requests] [-S program] host
     sftp [[user@]host[:file [file]]]
     sftp [[user@]host[:dir[/]]]
     sftp -b batchfile [user@]host

       Beispiel:
         sftp bell@amsel

ssh-keygen - authentication key generation, management and conversion
     ssh-keygen [-q] [-b bits] -t type [-N new_passphrase] [-C comment]
                [-f output_keyfile]
     ssh-keygen -p [-P old_passphrase] [-N new_passphrase] [-f keyfile]
     ssh-keygen -i [-f input_keyfile]
     ssh-keygen -e [-f input_keyfile]
     ssh-keygen -y [-f input_keyfile]
     ssh-keygen -c [-P passphrase] [-C comment] [-f keyfile]
     ssh-keygen -l [-f input_keyfile]
     ssh-keygen -B [-f input_keyfile]
     ssh-keygen -D reader
     ssh-keygen -U reader [-f input_keyfile]
     ssh-keygen -r hostname [-f input_keyfile] [-g]
     ssh-keygen -G output_file [-b bits] [-M memory] [-S start_point]
     ssh-keygen -T output_file -f input_file [-a num_trials] [-W generator]
```



```
ssh-agent - authentication agent

     ssh-agent [-a bind_address] [-c | -s] [-t life] [-d] [command [args ...]
     ssh-agent [-c | -s] -k

     Bespiel:
       ssh-agent -k    - beenden
       ssh-agent -s    - starten mit Shell-Ausgabe

ssh-add - adds RSA or DSA identities to the authentication agent

     ssh-add [-lLdDxXc] [-t life] [file ...]
     ssh-add -s reader
     ssh-add -e reader

     Bespiel:
          ssh-add

ssh-keyscan - gather ssh public keys

     ssh-keyscan [-v46] [-p port] [-T timeout] [-t type] [-f file]
                 [host | addrlist namelist] [...]

     Bespiel:
       ssh-keyscan -t rsa amsel
       ssh-keyscan -t dsa amsel
```

Beispiel passwortloses sichere Einloggen mit SSH:

1. **Privaten Key erzeugen**
   ```
   ssh-keygen -b 4096 -t dsa
   ssh-keygen -b 4096 -t rsa
   ```
2. **Public-Key auf das Zielsystem übertragen**
3. **Zielsystem initialisieren**
   ```
   ssh <Quellsystem>
   ```
   es entsteht ~/.ssh/known_hosts
4. **Public-Key eintragen**
   in ~/.ssh/authorized_keys oder ~/.ssh/authorized_keys2
5. **Starten eines ssh-agent auf Quellsystem**
   ```
   ssh-agent -k
   ssh-agent -s
   ```
   Shellvariable merken und setzen
6. **Privaten Key auf dem Quellsystem an SSH-Agenten uebergeben**
   ```
   ssh-add
   ```
7. **ssh ohne Passworteingabe möglich**
   (wenn Umgebungsvariable für ssh-agent gesetzt)