Wie oft versuchen irgendwelche Sciptkiddies oder andere unangenehme Zeitgenossen per SSH Bruteforceatacken auf einen Linux Server. Die passende Lösung wird hier beschrieben

SSH mit Passphrase

Was ist SSH mit Passphrase? Das Grundsätzliche Prinzip ist es, den öffentlichen SSH-Schlüssel des Hosts in die Datei known_hosts des Servers zu schreiben.

Die Passphrase ist eigenlich nur dazu da, den privaten Schlüssel auf dem Host zu schützen. Diese Phrase muss nicht unbedingt per Passwort geschützt werden

Verbindung von Linux zu Linux

Bei Linux ist es ganz einfach.

Zuerst muss man den Schlüssel für den entsprechenden User generieren

```
ssh-keygen -t rsa
```

oder

```
ssh-keygen -t dsa
```

mit der Option *-b 2048* kann man noch die Schlüssellänge vergrößern.

Zum Thema DSA- oder RSA-Verschlüsselung ist folgendes zu sagen. Beide Varianten sind sicher, aber mit neuester Technik (Quantencomputer) knackbar(Bruteforce).

- Man wird gefragt, wohin man das Schlüsselpaar speichern will. Als Standardpfad wird hier ~/.ssh/id_rsa angeboten. Dies kann man so lassen
- Als nächstes wird man nach der Passphrase gefragt. Hier kann man ein Kennwort eintragen, oder aber leer lassen (Passwortloser Zugang).

Auf dem System wurde jetzt ein privater Schlüssel (~/.ssh/id_rsa) und ein öffentlicher Schlüssel (~/.ssh/id_rsa.pub) abgelegt. Der öffentliche muss jetzt dem Zielsystem übermittel werden. Dazu einfach folgenden Befehl eingeben

```
ssh-copy-id -i ~/.ssh/id_rsa.pub user@remote-system
```

Sollte ssh-copy-id nicht installiert sein, kann man sich wie folgt helfen

```
cat ~/.ssh/*.pub | ssh user@remote-system 'umask 077; cat
>>.ssh/authorized_keys'
```

Beide Methoden sind gleich sicher, da sie beide per SSH (verschlüsselte Verbindung) übermittelt

werden.

Die letzte Möglichkeit ist, den Schlüssel per Hand zu kopieren (Copy/Paste)

Jetzt kann man sich am Zielsystem ohne Passwort (wenn eingegeben wird die Passphrase abgefragt) anmelden.

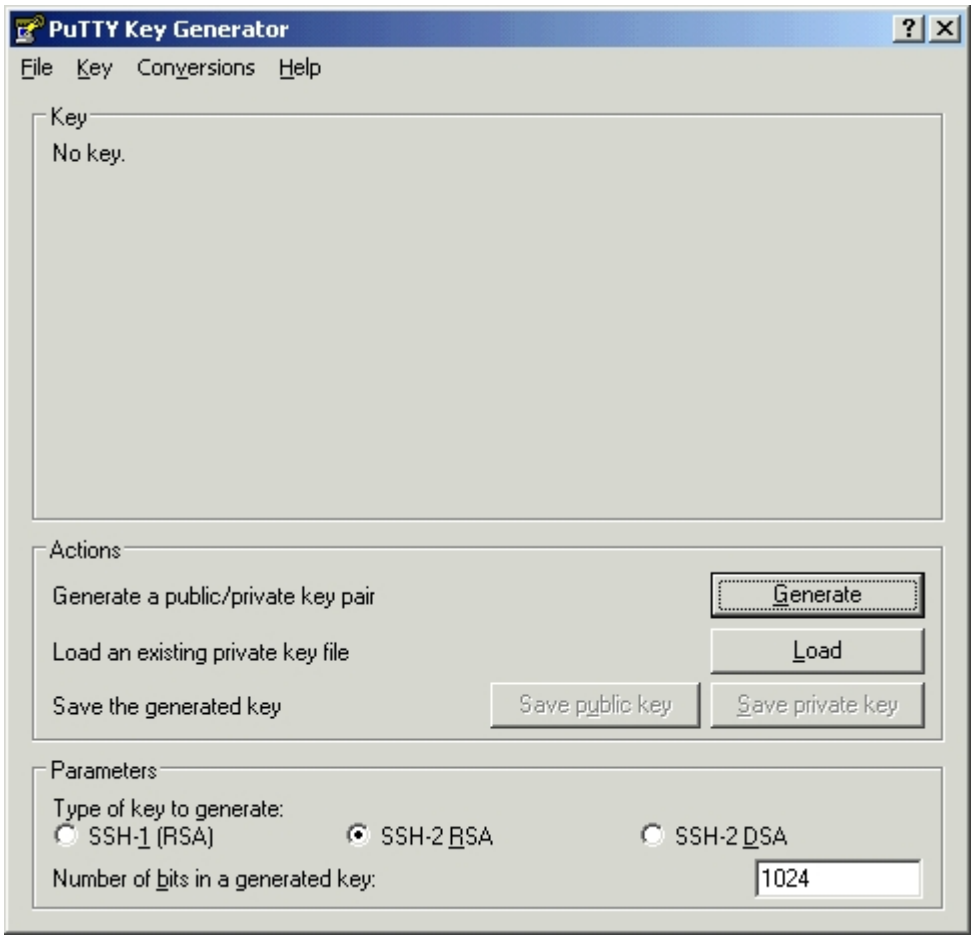
Verbindung von Windows zu Linux

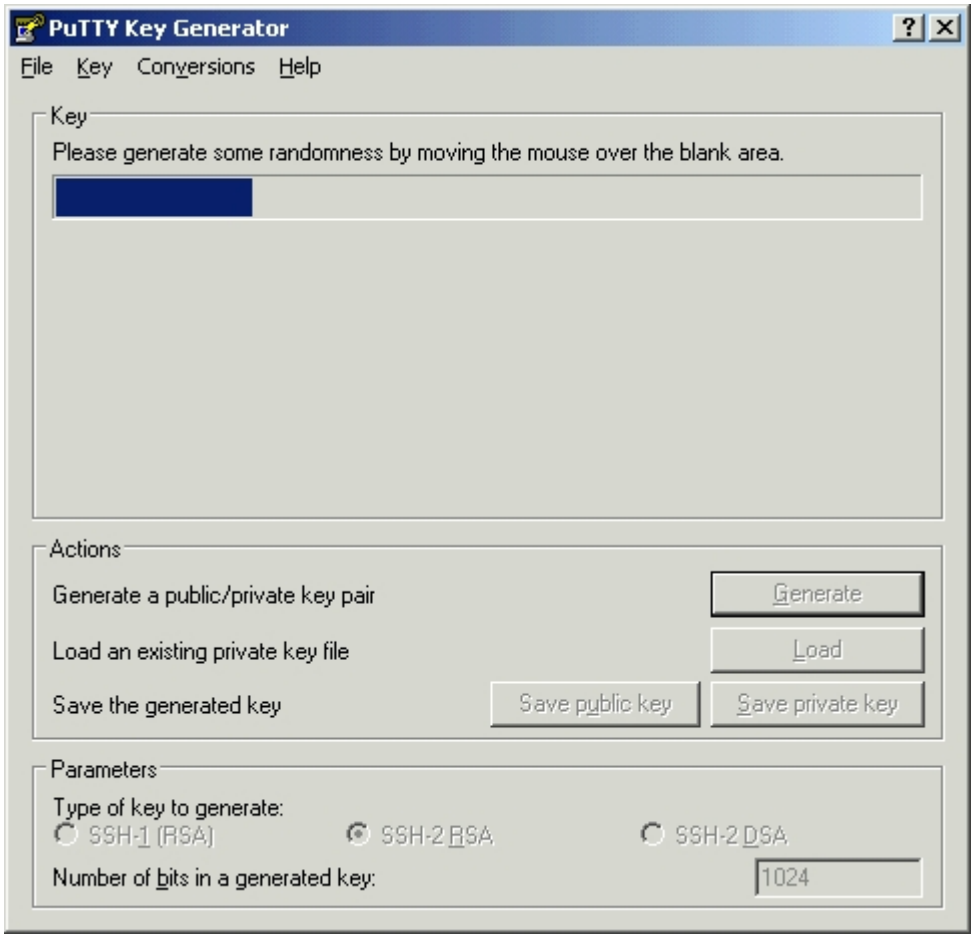
Im Prinzip ist es der gleiche Vorgang wie die Verbindung von Linux zu Linux. SSH-Key erstellen und beim Server eintragen. Da es in Windows kein Standardverzeichnis für SSH-Schlüssel gibt (bei Linux ~/.ssh/), muss man PuTTy noch sagen, wo der Private Schlüssel liegt.

1. Schritt wie oben: Schlüssel generieren

```
C:\Programme\PuTTY\puttygen.exe
```

starten. Bei erstmaliger Verwendung einfach auf *Generate* klicken und im oberen Feld die Maus Bewegen. Dadurch werden per Zufallsprinzip Werte ermittelt, die in den Schlüssel mit einfließen





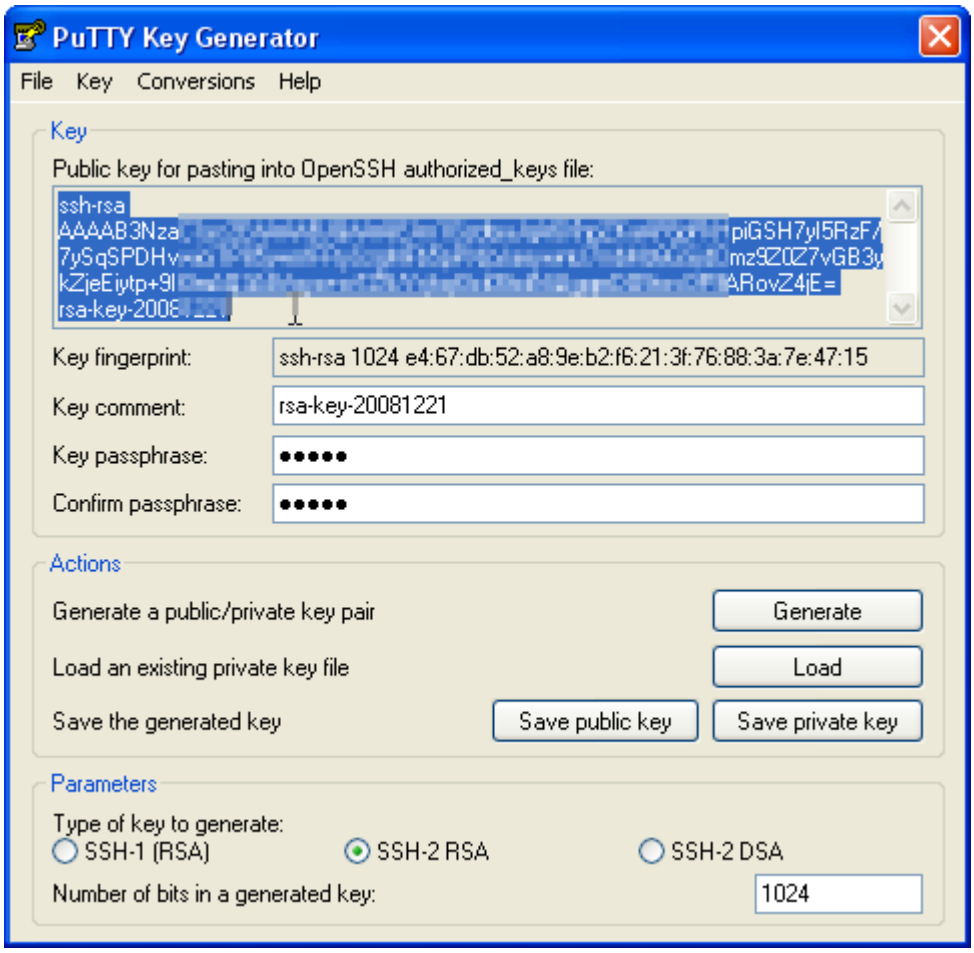
Hat man den Schlüsselgenerierungsprozess schon einmal durchlaufen, kann man die gespeicherten Keys nutzen. Dazu unter File –> Load private Key auswählen und Privaten Schlüssel auswählen

2. Schritt: Schlüssel speichern

Wie bei Linux werden die Schlüssel gespeichert. Leider wird das bei Windows nicht automatisch gemacht, sondern man muss es per Hand machen. Dazu auf die Button *Save public key* und *Save private Key* klicken und einen Ort zum speichern festlegen

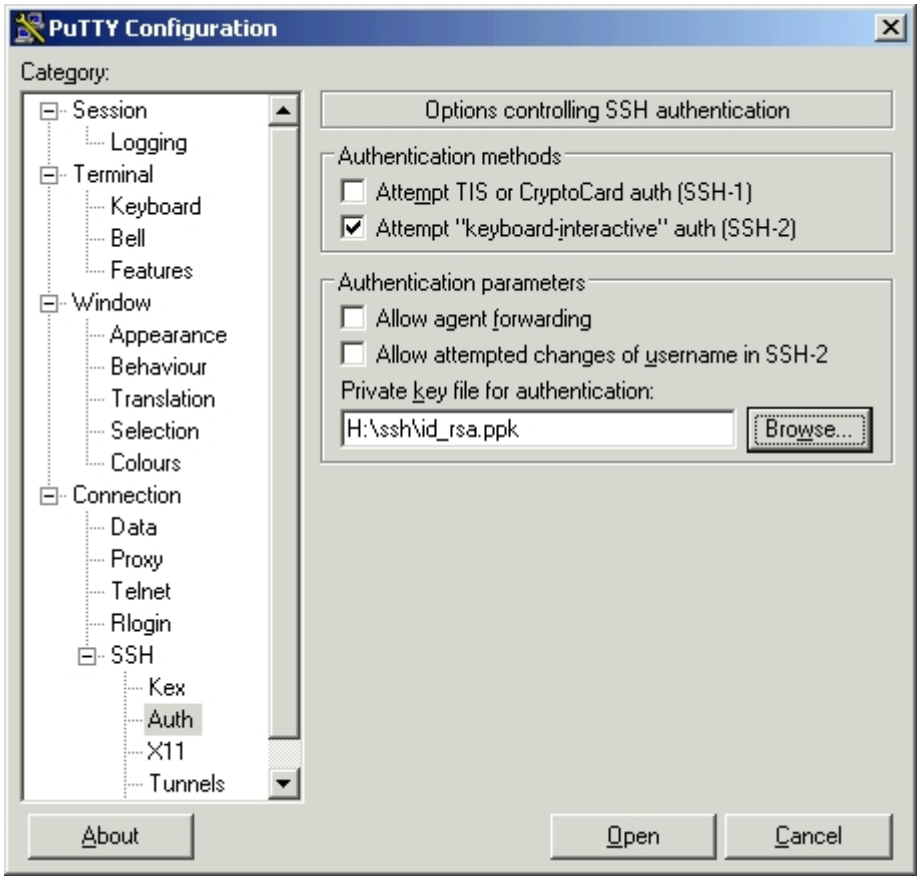
3. Schritt: Schlüssel übertragen

den Schlüssel, der oben in der Box abgebildet ist muss jetzt kopiert und beim Server unter dem jeweiligen User in die Datei *~/.ssh/authorized_keys* einfügt werden.



4. und letzter Schritt: PuTTy einrichten

Jetzt muss man PuTTy nur noch sagen, wo der private Schlüssel hinterlegt ist. Dazu muss man im Configuration-Bildschirm unter *Connection → SSH → Auth* bim Punkt *Private key file for authentification* den Pfad eingeben oder über Browse die Datei suchen.



5. Schritt: ausprobieren

Auf den entsprechenden Rechner einloggen und bei *login:* den User eingeben. wenn eine Passphrase eingegeben wurde, wird diese jetzt abgefragt. Hat man keine Phrase angegeben, wird kein Passwort abgefragt.

Fertig

6. Schritt: Fleißaufgabe

jedes mal die Passphrase eingeben ist doof. Deshalb gibt es den Pageant (PuTTY authentification agent). Dieser muss einfach gestartet werden und über Add Key der Private key eingetragen werden. Hatt der Schlüssel eine Passphrase, wird diese abgefragt und gespeichert. Bei jeder Verbindung zum Zielsystem wird nach der Usereingabe die Passphrase von Pageant „beantwortet“.

Einziger „Nachteil“ ist, dass Pageant bei der Authentifizierung gestartet sein muss. Das lässt sich aber über die Autostart-Funktion ganz einfach handhaben

SSH-Server absichern

Da man jetzt den ganzen Aufwand betrieben hat, private und öffentliche Schlüssel zu generiert und auszugetauscht, ist es sinnvoll, den Server so abzuhärten, dass dieser nur noch auf Verbindung mit Passphrase reagiert. Das macht auch die angreifbarkeit des Servers schwieriger

Dazu muss man in der Datei /etc/ssh/sshd_config folgende Werte ändern

```
PermitRootLogin without-password
```

Wenn auch andere Nutzer von der Schlüsselnutzung überzeugt werden sollen, dann sieht es etwas komplexer aus:

```
PasswordAuthentication no
ChallengeResponseAuthentication no
UsePAM yes
```

Zum Schluss muss man noch den SSH-Server neustarten

```
/etc/init.d/ssh restart
```

Interessante Links

- SSH ohne Passwort -- eine kurze Anleitung
- Anleitung für PuTTy