ITENOS
IT's us.

SSH PUBLIC KEY AUTHENTICATION

ANLEITUNG

ZUM GENERIEREN DES SCHLÜSSELPAARES

UND

ZUR ANMELDUNG AN LINUX SERVERN

Deutsche Telekom Gruppe

Inhaltsverzeichnis **Seite**

1 Einleitung

Im Rahmen der Standardisierten Datenschutz und Sicherheits Konzepte (*SDSK*) des Telekom Konzerns exisitiert hinsichtlich des Zugriffs per SSH die Anforderung einer Zwei-Faktor-Authentifizierung.

ITENOS hat diese Anforderung mit der dafür zugelassenen Public-Key-Authentication implementiert. Zugriffe auf (Server-)Systeme erfordern daher für jeden Benutzer ein individuelles Schlüsselpaar.

Dieses Dokument unterstützt Sie dabei, dieses Schlüsselpaar mit den erforderlichen Parametern zu erzeugen.

1.1 Public Key Authentication

Bei der Public Key Authentication werden zwei Schlüssel verwendet, welche der Benutzer im Vorfeld erzeugen muss (dazu im Folgenden mehr).

Der eine Schlüssel ist öffentlich (*public key*), welcher frei verteilt werden kann. In der Regel verteilt man ihn auf die Systeme, auf die man zukünftig zugreifen möchte. Der zweite Schlüssel ist privat (***private key***) und darf nicht in fremde Hände gelangen, da man sich mit diesem Schlüssel eindeutig identifiziert.

Die Kombination aus privatem und öffentlichem Schlüssel ermöglich dann den Zugriff auf die Systeme. Der Zugriff ist nur auf Systeme möglich, auf denen der *public key* hinterlegt ist und er ist nur möglich, wenn man den passenden *private key* verwendet.

2 Schlüsselpaar erstellen

Die folgenden Abschnitte zeigen die notwendigen Schritte, um ein Schlüsselpaar auf einem Windows System mittels der Anwendung *Putty* bzw. *Puttygen* zu erstellen sowie eine Möglichkeit dieses auf einem Linux System zu tun.

2.1 Schlüsseltyp

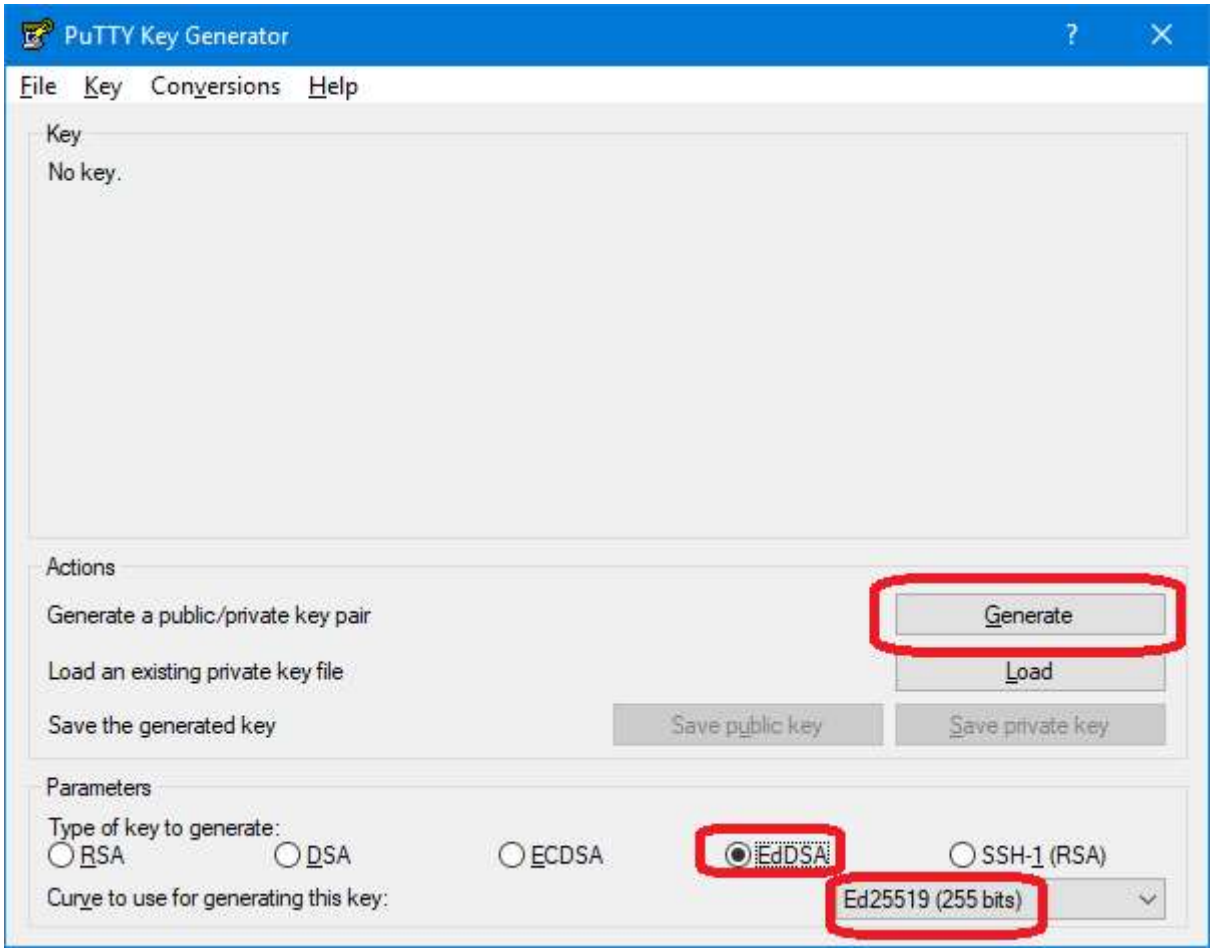
Mit der Einführung von OpenSSH 9 wird beim Zugriff über die *ITENOS Zentrale Jumzone* aus Kompatibilitätsgründen nur noch der Schlüsseltyp *ED25519* unterstützt. *RSA*-Schlüssel waren lange Zeit der Standard, funktionieren heute aber nicht mehr zuverlässig.

ED25519-Schlüssel werden von OpenSSH seit 2014 implementiert und bieten eine ähnliche Sicherheitsstufe wie 2048-Bit *RSA*-Schlüssel. Sie lassen sich jedoch mit weniger Rechenzeit erstellen und prüfen und die ASCII-Darstellung ist auch kürzer.

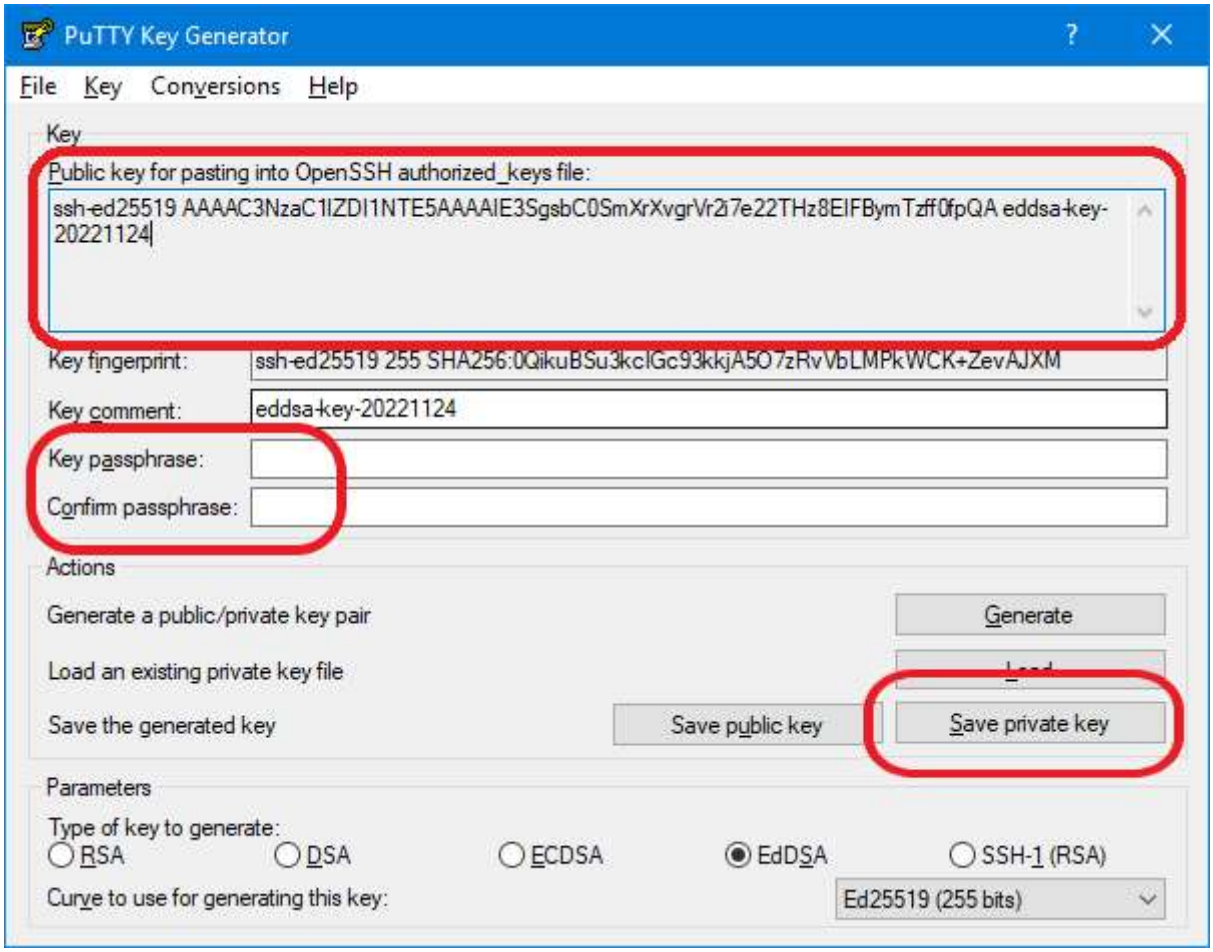
2.2 Windows

Unter Windows bietet es sich an, das Schlüsselpaar mit der Anwendung *Putty* bzw. der mit *Putty* ausgelieferten Anwendung *Puttygen* zu erstellen.

Dazu wird das Programm *Puttygen.exe* auf einem Windows Client oder Server gestartet. In der Anwendung wird *EdDSA* gewählt und der Generate-Butten gedrückt:



Die Generierung benötigt eine größere Menge an Zufallszahlen, die man durch das Bewegen des Mauszeigers über dem Puttygen-Fenster erzeugt. Dabei wird ein Fortschrittsbalken angezeigt.



Vor dem Speichern wird dem Schlüssel ein Name gegeben und ein persönliches Passwort für den Schlüssel festgelegt. **(Laut Anforderung aus dem SDSK muss das Passwort mindestens 12 Zeichen lang sein.)**

Dann wird der **private Schlüssel bzw. private key abgespeichert**.

Der angezeigte **Schlüssel/Key** (öffentliche Schlüssel bzw. *public key*) wird **dem Server-Administrator** zur Verfügung gestellt, um ihn auf den Servern in die jeweiligen *authorized_key*-Dateien einzutragen. (siehe 2.4)

2.3 Linux

Unter Linux kann der Befehl **ssh-keygen -t ed25519** für die Erstellung eines geeigneten Schlüsselpaares benutzt werden. Eine **Passphrase** ist zwingend festzulegen **(mindestens 12 Zeichen lang)**.

```
user@server:~/$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/user/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase): ***
Enter same passphrase again: ***
Your identification has been saved in /home/user/.ssh/id_ed25519
Your public key has been saved in /home/user/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:jQcewGPNQuHGcoRxzSYvBxVDdUpzZdTU79XB39VmyBU user@server
The key's randomart image is:
+--[ED25519 256]--+
|     .=*O=o+ o+=E*|
|     .=B.*o = .ooB|
+----[SHA256]-----+
```

Der öffentliche Schlüssel ist in diesem Fall **/home/user/.ssh/id_ed25519.pub**, der private Schlüssel **/home/user/.ssh/id_ed25519**

Die Dateinamen lassen sich bei der Erstellung des Schlüsselpaares selbstverständlich individuell festlegen.

Der **private Schlüssel bzw. *private key* verbleibt bei Benutzer** und wird lokal abgelegt.

Der **öffentliche Schlüssel bzw. *public key*** wird **dem Server-Administrator** zur Verfügung gestellt, um ihn auf den Servern in die jeweiligen *authorized_key*-Dateien einzutragen. (siehe 2.4)

2.4 Schlüssel übermitteln

Der **öffentliche Schlüssel bzw. *public key*** wird **dem Server-Administrator** in folgender Form übermittelt. Alle Angaben sind Pflichtangaben:

user: Benutzername für den der key hinterlegt werden soll.

type: Schlüsseltyp, in der Regel ssh-ed25519 (siehe 2.1)

key: Der eigentliche öffentliche Schlüssel (siehe 2.2 und 2.3)

owner: Eigentümer des keys bzw. Ansprechpartner

email: Email-Adresse des Eigentürmers/Ansprechpartners

Beispiel 1

```
user: mustermax
type: ssh-ed25519
key:
AAAAC3NzaC3NzaC1IZDI1NTE5AAAAIE3SgsbC0SmXrXvgrVr2i7e22THz8EIFBymTzff0fpQA
owner: Max Muster
email: max.muster@noreply.local
```

Beispiel 2

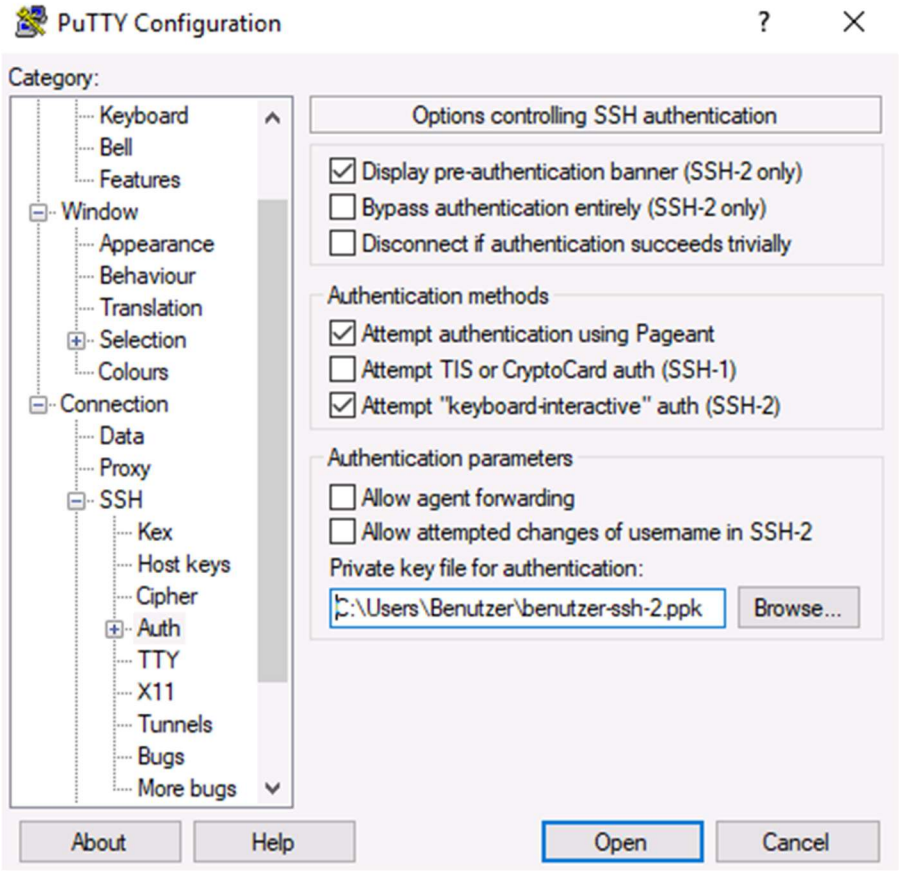
```
user: www_op
type: ssh-ed25519
key:
AAAAC3NzaC3NzaC1IZDI1NTE5AAAAIE3SgsbC0SmXrXvgrVr2i7e22THz8EIFBymTzff0fpQA
owner: Max Muster
email: max.muster@noreply.local
```

3 Anmeldung an einem System per Putty und Schlüsselpaar

Zur Anmeldung an einem Server per SSH wird zunächst durch einen Administrator der *public key* für ein Benutzerkonto hinterlegt. Anschließend kann *Putty* ggf. mit *Pageant* verwendet werden, um sich am System anzumelden. Dazu wird die Applikation gestartet und der private Schlüssel in *Putty* oder *Pageant* geladen.

3.1 Privater SSH Schlüssel mit Putty

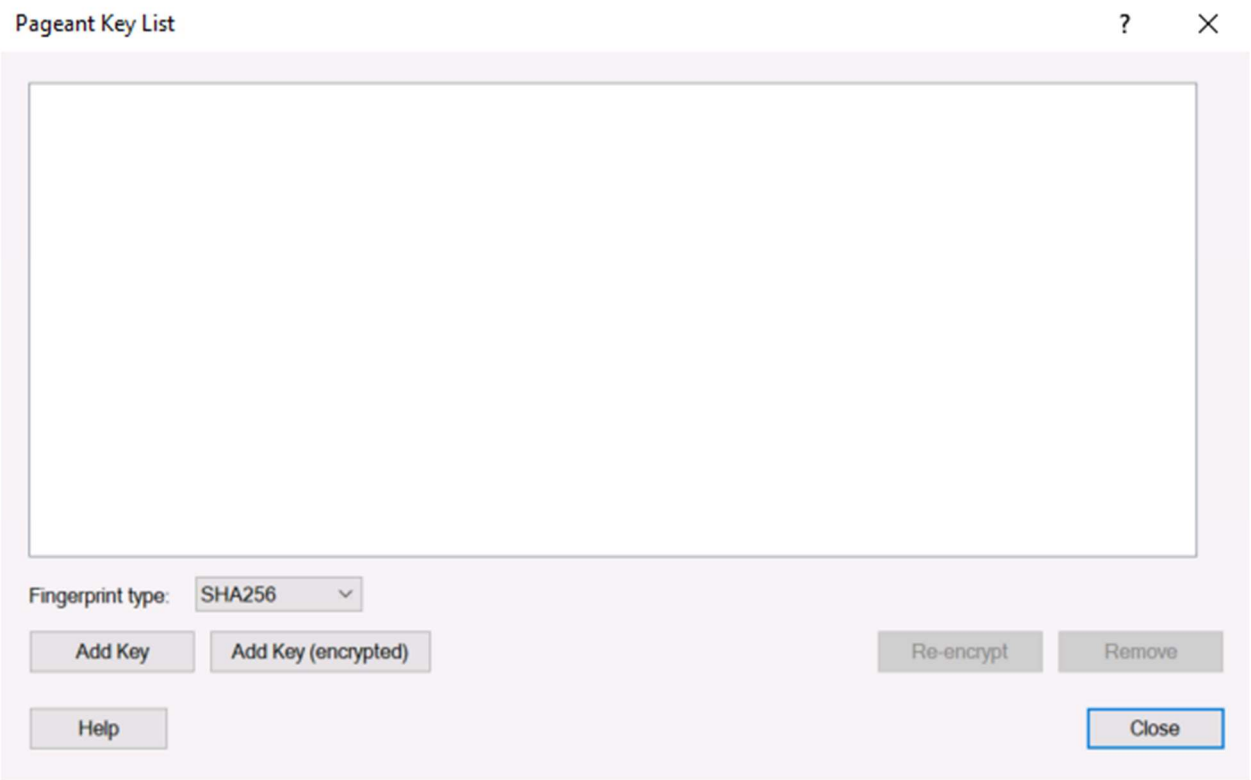
Möglichkeit eins: der Pfad und die Datei mit dem privaten Schlüssel wird in die *Putty*-Session eingetragen. Speichert man diese Session, wird auch der Pfad zum Schlüssel gespeichert.



Bei einer Anmeldung an einen Server gibt man am Server Prompt den Benutzernamen und das Passwort für den privaten Schlüssel (vergleiche 2.2 & 2.3) ein.

3.2 Privater SSH Schlüssel mit Pageant

Möglichkeit zwei: Man startet das Programm *pageant.exe* und lädt den Schlüssel mittels *Add Key*-Button:



Bevor der Schlüssel hier erfolgreich eingeladen wird, muss man das Passwort für den privaten Schlüssel zur Bestätigung eingeben (vergleiche 2.2 & 2.3).

Bei dieser Methode braucht man bei einer Anmeldung an einen Server außer dem Benutzernamen kein weiteres Passwort eingeben, da man sich bereits beim Eintragen des Passworts in *Pageant* authentifiziert hat.

Kontakt

Team ITP

servicedesk@itenos.de

0228 7293 0

Stand 04/2023