

paramiko

avagy SSH Python-ból szépen

<https://github.com/herczy/paramiko-presentation>

Hello, world!

Hercinger Viktor

Senior Consultant

@ Sigma Technology

github.com/herczy

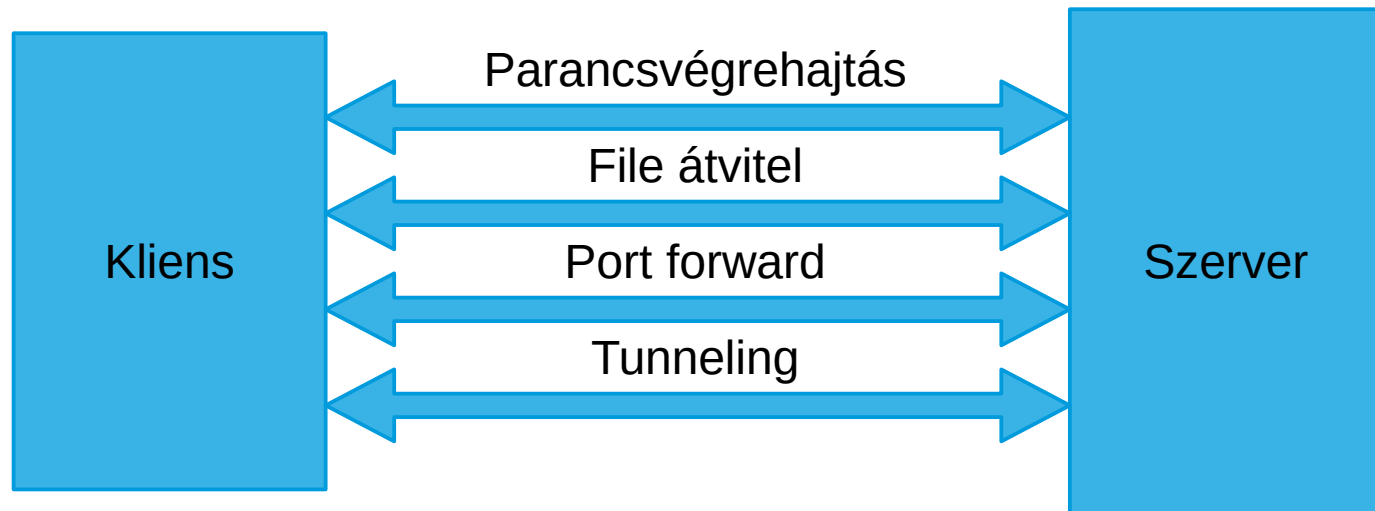
Mi is az SSH

- Secure Shell
- Protokoll, több implementáció
 - OpenSSH
 - Tectia SSH
- Aktuális verzió: SSHv2

Mit tud az SSH

- Shell access szerverhez
- File átvitel
 - SFTP
 - SCP
- Remote- és local port forward
- Tunneling

SSH és csatornák



Paramiko

- Python library
- Paranoid (paranoja) és barát (amiko) összevonása Esperanto nyelven
- Kompatibilis interpreterek:
 - Python 2.7, 3.4+
 - PyPy
- Minimum elvárások:
 - OpenSSL 1.0.1 vagy újabb
 - Pip 8.0 vagy újabb

Felépítés

- Transport – Rögton socket-re épül, alacsonyszintű kommunikációt kezel.
- Channel – Egyes SSH csatornákat (file átvitel, parancsvégrehajtás, stb.) kezel.
- SSHClient – Transport-ra és Channel-re épülve kliensfunkcionalitást nyújt.
- ServerInterface – Interfész szerverek létrehozására

Első lépések

`simple-command.py` példa

Mi történt?

- paramiko – Paramiko Python module
- paramiko.SSHClient – SSH kliens objektum
 - set_missing_host_key_policy – Mit csináljunk, ha nem ismerjük a host key-t?
 - connect – csatlakozás szerverhez
 - exec_command – parancs végrehajtása
 - recv_exit_status – parancs visszatérési értéke

Autentikáció

- connect – csatlakozás szerverhez
 - Példában: felhasználónév/jelszó kombináció
 - Lehetséges még:
 - `connect(pkey=..., passphrase=...)` – Privát kulcs objektum
 - `connect(key_filename=..., passphrase=...)` – Privát kulcs file neve
 - `connect()` – “none” autentikációval
- Alacsony szinten:
 - `Transport.auth_none`
 - `Transport.auth_password`
 - `Transport.auth_publickey`

Parancsvégrehajtás

- `exec_command` – egy csatorna – több file
 - `stdin`, `stdout` és `stderr` adatai ugyanazon a `Channel` objektumon dolgoznak.
 - `Channel`-en belül dől el, hogy egy adott bemenet melyikre jött
 - Alacsonyabb szinten is használható lenne
 - `Transport.open_session`
 - `Channel.exec_command`
 - `Channel.send` / `Channel.recv` / `Channel.recv_stderr`
- `SSHClient` tényleg csak wrapper, minden varázslat “lejjebb” történik a `Channel` és `Transport` szintjén.

host-keys.py példa

Host-kulcsok kezelése

- Alapértelmezetten: nincsenek ismert hostok
- Betölthető létező file-ból (`client.load_host_keys()`)
 - Visszamenthető `client.save_host_keys()`-szel
- `client.set_missing_host_key_policy:`
 - `paramiko.AutoAddPolicy()` – automatikus elfogadás és listához adás
 - `paramiko.RejectPolicy()` – automatikus visszautasítás
 - `paramiko.WarningPolicy()` – elfogadás, de figyelmeztetés kibocsátása
 - Lehet sajátot is csinálni – `paramiko.MissingHostKeyPolicy`-ből származtatással

Fájlátvitel SFTP-vel

copy-from-host.py
copy-to-host.py
list-files.py

Fájlvitel SFTP-vel

- `cli.open_sftp()` – SFTP kapcsolat létrehozása
- `sftp.put` / `sftp.get` – File fel- és letöltése
- `sftp.listdir` – Könyvtár listázása szerveren
- `sftp.stat` – File információ lekérdezése
 - Tulajdonos- és csoportazonosító, hozzáférési jogok, módosítás ideje, stb.
- És még sok más
- Kb. ugyanazok a metódusok, amik lokális file rendszerénél vannak
 - `chmod`, `chdir`, `stat`, `rmdir`, `mkdir`, stb.

Szerver kód

`greeting-server.py`

Szerver kód

- SSH szerver is készíthető vele
- Teljes funkcionalitás
 - Shell, parancsfuttatás
 - SFtp és egyéb alrendszerek
- Egyszerű példa: üdvözli a felhasználót, majd kilép

Amikre nem tértünk ki

- Port forwarding
- SSH tunneling
- Agent forward
- X11 forward
- Saját subsystem
- GSSAPI autentikáció

Példa: teszt log letöltése hosztról

- Adottak gépek egy IP tartományban (pl. 10.20.30.0/24)
- Várjuk meg, amíg egy adott file elkészül
- A file-ban lévő útvonal alapján töltsük le a logot
- Jelenítsük meg az eredményt (pl. \$EDITOR értékével)

Példa: teszt feltöltése hosztra

- Adottak gépek egy IP tartományban (pl. 10.20.30.0/24) – ugyanazok a gépek, mint az előző körben
- Lefordított futtathatók/library-k, stb.
- Fel kell tölteni ezeket a megfelelő helyre a hoszton
- Futtassuk le a teszteket
- Előző dián lévő programmal töltsük le az eredményeket
- Tervezett volt, proof-of-concept elkészült, de nem lett megvalósítva

Példa: teszteléshez szerver

- Tesztfuttatáshoz szükség van SSH szerverre?
- paramiko szerver tud tesztadatokkal kezelni
- Előnye, hogy nem kell “igazi” szervert használni
- Hátránya, hogy komplex kódot igényelhet időnként

Linkek

- Paramiko dokumentáció
- SSH & SCP in Python with Paramiko
- Github repo for examples

Köszönöm a figyelmet

Q&A