

paramiko

Capitoul 22 juin 2023

Lightning talk

Laurent GUERBY

IMT Mines Albi

Paramiko python SSH

- <https://www.paramiko.org/>
- Librairie LGPL autour du protocole SSH
- Python 3.6+
- `pip install paramiko`
- `apt-get install python3-paramiko`

Outils basés sur paramiko

- <https://github.com/saltstack/salt/blob/master/DEPENDENCIES.md>
- https://docs.ansible.com/ansible/latest/collections/ansible/builtin/paramiko_ssh_connection.html
- Fabric! <https://www.fabfile.org/> Fabric is a high level Python (2.7, 3.4+) library designed to execute shell commands remotely over SSH, yielding useful Python objects in return
- <https://github.com/ktbyers/netmiko> Multi-vendor library to simplify CLI connections to network devices
- <https://github.com/topics/paramiko>

Example 1

- <https://docs.paramiko.org/en/stable/api/client.html>

```
client = SSHClient()
```

```
client.load_system_host_keys()
```

```
client.connect('ssh.example.com')
```

```
stdin, stdout, stderr = client.exec_command('ls -l')
```

Example 2 fs.com S2805

```
ssh=paramiko.SSHClient()
```

```
ssh.set_missing_host_key_policy(paramiko.AutoAddPolicy())
```

```
ssh.connect(hostname="192.168.3.31",port=22,username="ad  
min",password="admin",look_for_keys=False,allow_agent=False,  
disabled_algorithms=dict(pubkeys=["rsa-sha2-512","rsa-sha2-  
256"]))
```

Example 2 suite

```
transport = ssh.get_transport()
channel = transport.open_session()
channel.get_pty()
channel.invoke_shell()
while True:
    if channel.exit_status_ready():
        print("ready")
        break
    rl, wl, xl = select.select([channel], [channel], [], 0.0)
    if len(rl) > 0:
        tmp=channel.recv(1024)
        print(tmp)
```

Example 3 timeout

```
def do_ssh(hostname,command):
    ssh=paramiko.SSHClient()
    ssh.set_missing_host_key_policy(paramiko.AutoAddPolicy())
    try:
        ssh.connect(hostname=hostname,port=22,username="root",timeout=5,banner_timeout=5,auth_timeout=5)
        stdin,stdout,stderr=ssh.exec_command(command)
        lines=stdout.read()
        return lines.decode()
    except:
        return None
```

Usage a IMT Mines Albi

- Se connecter a notre parc de VM, commutateurs, “devices”
- Executer quelques commandes
- Sans avoir de “grosse” dépendance

Usage futur

- Outil de génération, suivi, cohérence de règle pare-feu
- Analyse de “ss -tulpn” (ex netstat) pour avoir la réalité des ports en écoute sur le parc de VM
- Choisir le zonage firewall du service
- “diff” quotidiens et archivage git

Usage futur 2

- Rancid a un peu de mal avec les fs.com
- Pas un pro d'expect tel qu'utilisé par rancid
- Au final peu de modèles de commutateurs sur notre parc donc paramiko

Usage futur 3

- Créer un équivalent iftop sur tout le parc VM et commutateurs
- Paramiko + select + boucle

Conclusion

- Questions ?