PROCESSES AND JOBS

Listing Processes

`/challenge/run` is renamed to something random and `ls` cannot be used in `/challenge`

The new `/challenge/run` process is already running so using `ps -ef` would show the new name of the process.

On running `ps -ef` the output is:

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	1	0	0	20:31	?	00:00:00	/sbin/docker-init -- /nix/va
root	7	1	0	20:31	?	00:00:00	/run/dojo/bin/sleep 6h
root	68	1	0	20:31	?	00:00:00	/challenge/26423-run-32595
root	72	68	0	20:31	?	00:00:00	sleep 6h
hacker	95	1	0	20:31	?	00:00:00	/nix/store/g53lqy8w86jkp2g98
hacker	100	1	0	20:31	?	00:00:00	/nix/store/1xhds5s320nfp2022
hacker	109	100	0	20:31	?	00:00:00	/nix/store/3wb0055984n2whn44
hacker	119	1	0	20:31	?	00:00:00	/nix/store/jnz1wdn463qilzhc8
hacker	122	1	0	20:31	?	00:00:00	/nix/store/rvr416z148cjcxh2b
hacker	123	1	0	20:31	?	00:00:00	/run/current-system/sw/bin/d
hacker	128	1	0	20:31	?	00:00:00	/nix/store/3zi7zn2bbz3jpx8ds
hacker	137	1	0	20:31	?	00:00:00	/run/workspace/bin/ssh-agent
hacker	139	119	0	20:31	?	00:00:00	xfwm4
hacker	144	119	0	20:31	?	00:00:00	xfsettingsd
hacker	149	119	1	20:31	?	00:00:01	xfce4-panel
hacker	154	119	0	20:31	?	00:00:00	Thunar --daemon
hacker	162	119	0	20:31	?	00:00:00	xfdesktop
hacker	183	149	0	20:31	?	00:00:00	/nix/store/xpgf309sv9w5majf2
hacker	203	149	0	20:31	?	00:00:00	/nix/store/xpgf309sv9w5majf2
hacker	273	109	0	20:31	?	00:00:00	/nix/store/3wb0055984n2whn44
hacker	297	1	0	20:31	?	00:00:00	/run/workspace/bin/xfce4-ter
hacker	303	297	0	20:31	pts/0	00:00:00	bash
hacker	550	303	0	20:32	pts/0	00:00:00	ps -ef

The flag retrieved when running `/challenge/26423-run-32595`

`pwn.college{MmCsWPwz7zCOcX1ZbCJoE37toFw.dhzM4QDL3YjN0czW}`

Killing Processes

`/challenge/dont_run` must be killed before `/challenge/run` is executed so you get the flag.

The `PID` of `/challenge/dont_run` is found by using the `ps` command by: `ps -ef | grep /challenge/dont_run`

This gives the output:

hacker	73	71	0	20:36	?	00:00:00	/challenge/dont_run
hacker	368	301	0	20:36	pts/0	00:00:00	grep --color=auto /challenge/dont_run

The process is killed using `kill` by `kill 73`

Now running `/challenge/run` gives the flag: pwn.college{cJsbEhsfCiTctTW6wfNQQytRBhl.dJDN4QDL3YjN0czW}

Interrupting Processes

`/challenge/run` is run first and it gives the output:

```
1  I could give you the flag... but I won't, until this process exits. Remember,
2  you can force me to exit with Ctrl-C. Try it now!
3
```

After using the hotkey `Ctrl-C` it gives the flag: pwn.college{Yqw1mhbH68bt92CRs9HoUufJYUm.dNDN4QDL3YjN0czW}

SUSPENDING PROCESSES

The first instance of `/challenge/run` is run.

It is suspended using the hotkey `Ctrl-Z`.

Then after running the next instance of `/challenge/run`, the flag outputted is:

pwn.college{MOMmJw36g4i92GGsmXBIp61s-6-.dVDN4QDL3YjN0czW}

Resuming Processes

Just like the previous challenge `/challenge/run` is run and then suspended using `Ctrl-Z`

Then it is resumed using `fg`

This gives the flag: pwn.college{8kp7hE5wb95EF-7BniPtRVyU9Kh.dZDN4QDL3YjN0czW}

Backgrounding Processes

Just like the previous challenge `/challenge/run` is run and then suspended using `Ctrl-Z`

Then it is resumed in the background using `bg`.

Another instance of `/challenge/run` is run which gives the flag: pwn.college{cpmicRbb9RPsUXN_kE38IuoDbx6.ddDN4QDL3YjN0czW}

Foregrounding Processes

In this challenge, `/challenge/run` must be first run, then suspended then resumed in the background and then directly foreground the process.

This can be done using the following commands,

```
1  /challenge/run
2  Ctrl-Z
3  bg
4  fg
5
```

This gives the flag: pwn.college{0AMv3ifdWEVaCWovWVWTFTONkaq.dhDN4QDL3YjN0czW}

Starting Backgrounded Processes

`/challenge/run` needs to be backgrounded without suspending it first. This can be done by appending a `&` to the command.

`/challenge/run &` This gives the flag: `pwn.college{kJO4p7pwaZp0Wzq7ozxVETUPYV0.dlDN4QDL3YjN0czW}`

Process Exit Codes

The exit-code of `/challenge/get-code` must be accessed and used as an argument for `/challenge/submit-code` .

After running `/challenge/get-code` the exit code is stored in the shell variable `?`

Therefore `echo $?` prints the exit code which is `208`

`/challenge/submit-code` gives the flag: `pwn.college{4E9ipkQbpv47K2_RA5KiDJqJEwr.dljN4UDL3YjN0czW}`