

 Get unlimited access to the best of Medium for less than \$1/week. [Become a member](#)

Tryhackme Free Walk-through Room: REmux The Tmux

Daniel Schwarzentraub : [Follow](#)

8 min read · Nov 10, 2024



Listen



| Share



... More

Task 1: Tmux practice machine

Username: tux

Password: tmuxwithremux

Task 2: Starting tmux “Sessions” and default tmux “prefix”

To start a new tmux session just run the tmux command with no arguments. The first session create will have the name "0". By default, tmux status bar will be green. With session name on the left. Windows in the middle and window names in the middle. Hostname, time, and date on the right of the bottom green bar.

```
tux@tmux_remux:~$
```

Tmux doesn't allow to create of a nested tmux within a tmux unless you force it to. When running the tmux command a second time.

```
tux@tmux_remux:~$ tmux
sessions should be nested with care, unset $TMUX to force
tux@tmux_remux:~$
```

To change the session name from "0" -> "box-dev". Must first learn how tmux is called. All commands within a tmux session all start with the tmux prefix is. By default, the tmux prefix is "Ctrl b".

After the tmux prefix. To the hotkeys to change the current tmux session's name is "shift \$".

ctrl b shift \$

Retype the new name and then enter-key to save the new session name.

```
tux@tmux_remux:~$
```

(rename-session) box-dev

```
[box-dev] 0: bash*
```

If there is a need to create another tmux session within the current one. Use the -d argument with the tmux command. To spawn a new tmux session without attaching to it. In the example image below. The -s argument is used to specify the session name for the new session. Typed as "tmux new -s <new-session-name> -d".

tmux@tmux_remux:~\$ tmux new -s tryhackme -d

To list all active tmux sessions. Run tmux with list-sessions. Or the short version of list-sessions as ls. In the example below. Running tmux ls also shows the current session in use. Marked by "(attached)".

```
tux@tmux_remux:~$ tmux ls
box-dev: 1 windows (created Wed Jul 29 16:27:44 2020) [77x8] (attached)
tryhackme: 1 windows (created Wed Jul 29 16:38:11 2020) [80x24]
tux@tmux_remux:~$
```

[box-dev] 0: bash*

Exiting a tmux session without closing it can be done with the prefix. Followed by d

ctrl b d

Checking again with the tmux ls command. "(attached)" is missing from both sessions. This means the sessions are active but we are detached and are unable to interact with either session.

```
File Edit View Search Terminal Help
tux@tmux_remux:~$ tmux ls
box-dev: 1 windows (created Wed Jul 29 16:27:44 2020) [77x8]
tryhackme: 1 windows (created Wed Jul 29 16:38:11 2020) [80x24]
tux@tmux_remux:~$
```

To reattach to an active tmux session. Run tmux with the attach option and -t followed by the desired session name.

```
tux@tmux_remux:~$ tmux attach -t tryhackme
```

The tmux session name has changed to the attached session of "tryhackme". Double-checking with tmux ls. Can confirm that "(attached)" also on the "tryhackme" session name.

```
File Edit View Search Terminal Help
tux@tmux_remux:~$ tmux ls
box-dev: 1 windows (created Wed Jul 29 16:27:44 2020) [77x8]
tryhackme: 1 windows (created Wed Jul 29 16:38:11 2020) [77x8] (attached)
tux@tmux_remux:~$
```

[tryhackme0:bash*] "tmux_remux" 17:48 29-Jul-20

Delete a single session by its session name. Is done with the kill-session option with tmux. Followed by -t and the <target-session-name-to-delete>

```
tux@tmux_remux:~$ tmux kill-session -t box-dev
```

Listing sessions with tmux ls. This shows that the session name of box-dev has been removed. WARNING! By deleting the session. Anything open in that session will be lost if not saved before the tmux kill-session.

```
File Edit View Search Terminal Help
tux@tmux_remux:~$ tmux ls
tryhackme: 1 windows (created Wed Jul 29 16:38:11 2020) [77x8] (attached)
tux@tmux_remux:~$
```

[tryhackme0:bash*] "tmux_remux" 18:20 29-Jul-20

In the example below there are many sessions open. Another way to swap sessions without having to detach and reattach to another session. Is to use the prefix. Followed by the s-key to list all open sessions. Using up or down arrow keys to navigate to the desired tmux session. Then enter to select the new session.

ctrl b s

```
File Edit View Search Terminal Help
tux@tmux_remux:~$ tmux list-sessions
0: 1 windows (created Wed Jul 29 18:25:51 2020) [80x24]
is-0day-0k?: 1 windows (created Wed Jul 29 18:24:35 2020) [80x24]
notes: 1 windows (created Wed Jul 29 18:23:44 2020) [80x24]
tmux_remux: 1 windows (created Wed Jul 29 18:25:39 2020) [80x24]
tryharderer: 1 windows (created Wed Jul 29 18:25:04 2020) [80x24]
tux@tmux_remux:~$
```

Change from session name "0" to "tmux_remux" without having to leave the current tmux session.

```
File Edit View Search Terminal Help
(0) + 0: 1 windows (attached)
(1) + is-0day-0k?: 1 windows
(2) + notes: 1 windows
(3) + tmux_remux: 1 windows
(4) + tryharderer: 1 windows
[0] 0:[tmux]*
```

"tmux_remux" 18:48 29-Jul-20

From the five active sessions above. If there was a need to kill all the sessions except for a single one. When using the tmux kill-session. Use the -a argument to close all sessions except the one specified by the -t argument. Closing all tmux sessions except for the one named "is-0day-0k?". Checked with the tmux list-sessions.

```
File Edit View Search Terminal Help
tux@tmux_remux:~$ tmux kill-session -t is-0day-0k? -a
tux@tmux_remux:~$ tmux list-sessions
is-0day-0k?: 1 windows (created Wed Jul 29 18:24:35 2020) [80x24]
tux@tmux_remux:~$
```

When splitting the session into different sized panes ("which will be covered in more detail in a later task"). The new pane will spawn in the directory that the tmux session was first started in.

```
File Edit View Search Terminal Help
tux@tmux_remux:~$ █

tux@tmux_remux:~$ █

[is-0day-00: bash*] "tmux remux" 18:44 29-Jul-20
```

To change the base starting directory. Must first learn about tmux prompt or command mode. The tmux prompt allows tmux sessions to run tmux commands without the tmux binary name. Useful when the terminal has been filled with other text. Enter a tmux prompt with prefix shift :

ctrl b shift :

Followed by "attach -c /path/to/new/starting/directory"

```
File Edit View Search Terminal Help  
tux@tmux_remux:~$  
  
tux@tmux_redux:~$  
  
:attach -c /dev/shm
```

With the updated starting or base directory done above as /dev/shm. Creating a new pane start in the /dev/shm directory.

```
File Edit View Search Terminal Help  
tux@tmux_remux:~$  
  
tux@tmux_remux:/dev/shm$ █  
  
[is-0day-00: bash*] "tmux_remux" 18:46 29-Jul-20
```

Even though at the start tmux doesn't allow nested tmux within a tmux. Attaching or starting a new tmux session on another computer by an ssh connection. Can make a nested tmux. Not a problem. Just by changing the number of prefix used before the following command. Can determine which session gets the command.

prefix, prefix, and command. This will run on the second nested tmux session of the ssh ubuntu machine.

prefix and command. This will run on the first tmux session. The session running on the current machine's localhost.

The screenshot shows a terminal window with two tmux sessions. The left session is named "ubuntu" and the right session is named "tmux remux". Both sessions have a green border and a black background. The prompt in both sessions is "[session_name:~\$]". At the bottom of each session, there is a timestamp: "18:55 29-Jul-20" for the left session and "18:55 29-Jul-20" for the right session. The top of the window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help".

Question: Do the ctrl and b keys need to be held down the whole time with every commands to work? yea/nay

Answer: nay

Question: How to start tmux with the session with the name “thm”?

Answer: **tmux new -s thm**

Question: How to change the current tmux session name?

Answer: **ctrl b shift \$**

Question: How to quit a tmux session without closing the session? To attach back later.

Answer: **ctrl b s**

Question: How to list all tmux sessions?

Answer: **tmux ls**

Question: How to reattach to a detached tmux session with the session name of “thm”

Answer: **tmux a -t thm**

Question: How to create a new tmux session from your current tmux session with the name kali?

Answer: **tmux new -s kali -d**

Question: How to switch between two or more tmux sessions without detaching from the current tmux session?

Answer: **ctrl b s**

Question: How do you force kill the tmux session named “thm” if it’s not responsive from a new terminal window or tmux session?

Answer: **tmux kill-session -t thm**

Question: Within a nested tmux session. A second tmux session within the first one. How to change the session name of the second/internal tmux session?

Answer: **ctrl b ctrl b shift \$**

Question: How to get into a tmux prompt to run/type tmux commands?

Answer: **ctrl b shift :**

Question: Are there more than one way to exit a tmux prompt? yea/nay

Answer: **yea**

Question: Is tmux case sensitive. Will hitting the caps lock break tmux? yea/nay

Answer: **yea**

Question: Within tmux prompt or command mode how would you change the tmux directory? Where a new window or pane will start from the changed directory of /opt.

Answer: **a -c /opt**

Question: How to kill all tmux sessions accept the one currently in use? With the name “notes”.

Answer: **tmux kill-session -t notes -a**

Task 3: Manage tmux “Panes”

Within the current tmux session window. Tmux panes are used to divide the current session into multiple-sized terminals. That allows multiple commands to run within the same session window.

To split the currently selected pane horizontally. Do the prefix. This followed by a shift "

ctrl b shift "

A screenshot of a tmux session window. The window title is "tmux remux". The status bar at the bottom shows the session name "tmux remux" and the date and time "22:21 30-Jul-20". The main area of the window is black, indicating no active text or output.

ctrl b shift %

A screenshot of a tmux session window after performing the command. The window is divided into two horizontal panes. The left pane is significantly larger than the right one. Both panes are black, indicating no active text or output. The status bar at the bottom shows the session name "tmux remux" and the date and time "22:21 30-Jul-20".

The exit command can be used to close the currently selected pane.

```
File Edit View Search Terminal Help
tux@tmux_remux:~$ 
tux@tmux_remux:~$ 
tux@tmux_remux:~$ exit
tux@tmux_remux:~$ 
[0] 0:bash* "tmux_remux" 22:24 30-Jul-20
```

Moving to another pane within the same window can be done with the prefix. Followed by the arrow keys.

ctrl b <arrow key>

However, the main problem with this method is sometimes when using the arrow keys again. For example, to move the terminal cursor box could move to another tmux pane instead.

To resolve this issue can prefix followed by o instead. This can cycle through all the tmux panes. This has the added benefit of the line cursor not escaping to another pane.

ctrl b o

To swap between the most used tmux panes if more than two panes are open. "ctrl b ;" is the better option. "ctrl b o" is better when there are just two panes open.

ctrl b ;

With <arrow-keys> and or <o> in the image below to move the terminal cursor box from the left to the right.

```
File Edit View Search Terminal Help
tux@tmux_remux:~$ 
tux@tmux_remux:~$ 
tux@tmux_remux:~$ 
[0] 0:bash* "tmux_remux" 22:25 30-Jul-20
```

If a tmux pane isn't responsive and ctrl-c isn't resolving the issue. Force close or kill the currently selected tmux pane with the prefix and x. Then y to confirm.

ctrl b x y

```
tux@tmux_remux:~$ [0] 0: bash* "P" 23:43 30-Jul-20
```

If killing the pane was a mistake. Use the n option for no instead of y for yes when the kill-pane prompt is shown.

kill-pane 1? (y/n)

```
tux@tmux_remux:~$ [0] 0: bash* "tmux_remux" 23:44 30-Jul-20
```

Managing the placement of panes. Can change the layout without having to exit panes or open create new panes.

To move the currently selected pane. In a clockwise rotation. Do prefix shift }

ctrl b shift }

```
tux@tmux_remux:~$ cat cow
' _ '
(oo)-----
(--)-----)\ \
   ||---|| *  
tux@tmux_remux:~$
```

[0] 0:bash* "tmux remux" 22:41 30-Jul-20

Note. All other panes will move clockwise with the currently selected pane.

```
tux@tmux_remux:~$ cat cow
' _ '
(oo)-----
(--)-----)\ \
   ||---|| *  
tux@tmux_remux:~$
```

[0] 0:bash* "tmux remux" 22:41 30-Jul-20

To move the currently selected pane counter-clockwise. Do prefix shift {

ctrl b shift {

```
tux@tmux_remux:~$ cat cow
'--'
(oo)-----
(--)-----)\ \
| |---|| *  

tux@tmux_remux:~$
```

[0] 0:bash* "tmux_remux" 22:41 30-Jul-20

Another way to manage the pane location is with five built-in layouts. This can be done with the prefix followed by the escape OR esc key and the desired option from the number selected from 1 to 5.

ctrl b esc 4

Note that the layout result depends on how many panes are currently open.

```
tux@tmux_remux:~$ nano horse
tux@tmux_remux:~$ cat horse
'--'
(oo)-----
(--)-----)\ \
| |---|| *  

tux@tmux_remux:~$
```

[0] 0:bash* "tmux_remux" 22:48 30-Jul-20

To cycle through the built-in pane layouts one at a time. Do the prefix with the spacebar. This can also be an alternative using a clockwise or counter-clockwise option when there are only two panes open on the current tmux window.

ctrl b spacebar

```
orse
'--'
(oo)-----
(--)-----)\ \
| |---|| *  

tux@tmux_remux:~$
```

[0] 0:bash* "tmux_remux" 22:51 30-Jul-20

Lastly when it comes to managing panes. There might be a need to swap the place between two panes. When cycling panes don't create the desired pane layout.

```
tux@tmux_remux:~$ cat parrot
-----+
 / \ /{}/o\_
 / \ \-\ /-
 / \ \-\ /-
 / \ \-\ /-
tux@tmux_remux:~$
```

```
tux@tmux_remux:~$
```

```
tux@tmux_remux:~$
```

[0] 0:bash* "tmux_remux" 23:21 30-Jul-20

Before swapping, you can identify pane numbers with `ctrl b q`. You must first check the number each pane has been assigned. This also provides the size of each pane.

```
tux@tmux_remux:~$ cat parrot 36x10 tux@tmux_remux:~$ 36x10
```

```
tux@tmux_remux:~$ 36x9 tux@tmux_remux:~$ 36x9
```

[0] 0:bash* "tmux_remux" 23:20 30-Jul-20

For the swap pane example. It will be done within tmux prompt or command mode with prefix `shift : then the commands swap-pane`. With `-s <source-pane>` and `-t <destination-pane>`. This will send the move/swaps the currently selected pane and keep the currently selected pane selected after the swap.

`:swap-pane -s <source pane final swap location> -t <currently selected destination pane>`

```
tux@tmux_remux:~$ cat parrot tux@tmux_remux:~$
```

```
tux@tmux_remux:~$
```

```
tux@tmux_remux:~$
```

:swap-pane -s 3 -t 0

With the command above in this example. The currently selected pane of the parrot has moved to the bottom right and has followed the currently selected pane. Still able to start typing within the parrot pane without having to switch to another pane.

```
tux@tmux_remux:~$ cat cow
' _ '
(oo)_____
(--)_____)\
   ||---|| *  

tux@tmux_remux:~$  

tux@tmux_remux:~$ cat parrot
' _ .
(oo)_____
(--)_____)\
   ||---|| *  

tux@tmux_remux:~$
```

[0] 0:bash* "tmux_remux" 23:33 30-Jul-20

To swap places with another pane and select the other open pane. Instead of the currently selected pane. Have the -t <destination-pane> first with the pane you want to swap with. Followed by the -s <source-pane> or currently selected pane

:swap-pane -t <pane to swap with> -s <currently selected pane>

```
tux@tmux_remux:~$ cat cow
' _ '
(oo)_____
(--)_____)\
   ||---|| *  

tux@tmux_remux:~$  

tux@tmux_remux:~$ cat parrot
' _ .
(oo)_____
(--)_____)\
   ||---|| *  

tux@tmux_remux:~$
```

"tmux_remux" 23:33 30-Jul-20

```
tux@tmux_remux:~$ cat cow
' _ '
(oo)_____
(--)_____)\
   ||---|| *  

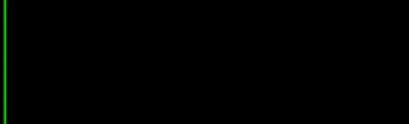
tux@tmux_remux:~$  

tux@tmux_remux:~$ :swap-pane -t 0 -s 3
```

"tmux_remux" 23:33 30-Jul-20

Started with the parrot in the bottom right pane. After the command. Have swapped places with the cow. Also, the new currently selected pane is the cow.

tux@tmux_remux:~\$ cat parrot	tux@tmux_remux:~\$
	
tux@tmux_remux:~\$	
tux@tmux_remux:~\$	tux@tmux_remux:~\$ cat cow
	
	tux@tmux_remux:~\$

Question: How to create a new pane split horizontally?

Answer: **ctrl b shift “**

Question: How to close a tmux pane like closing a ssh session?

Answer: exit

Question: How to create a new pane split vertically?

Answer: **ctrl b shift %**

Question: How to cycle between tmux pre built layout options? Starting with the number 1.

Answer: **ctrl b esc 1**

Question: How to cycle/toggle between tmux layouts, one at a time?

Answer: ctrl b spacebar

Question: How to force quit a frozen, crashed or borked pane?

Answer: ctrl b x y

Question: How to move between the two most used tmux panes for the current tmux window?

Answer: ctrl b ;

Question: Can you use the arrow to move to the desired pane? yea/nay

Answer: yea

Question: How to move the currently selected pane clockwise?

Answer: **ctrl b shift {**

Question: How to move the currently selected pane counter-clockwise

Answer: **ctrl b shift }**

Question: Before using swap-pane. How to check for which pane has what number?

Answer: **ctrl b q**

Question: How to swap two panes and move with the swapped pane? Within tmux prompt mode. 1 -> 3 location

Answer: **swap-pane -s 3 -t**

Question: How to swap two panes without changing the currently selected pane location? Within tmux prompt mode. 1 -> 4 pane number

Answer: **swap-pane -s 4 -t**

Task 4: Manage tmux “Windows”

Tmux windows are like a new terminal tab that you can easily swap from and more. To create a new empty tmux window to work on. Do prefix c. Shown with the new window number and name. Note the currently selected window will be marked with a * or wildcard star. At the end of the current window name, te must be at least 1 minute.

ctrl b c

```
tux@tmux_remux:~$
```

[0] 0:bash- 1:bash* "tmux remux" 23:50 30 Jul 20

To change the name of the current window. Do prefix ,. After retying the new window name. Hit the enter key to save changes.

ctrl b ,

```
tux@tmux_remux:~$
```

(rename-window) tmux remux

[0] 0:bash- 1:tmux remux*

Oh NO. The recon window has been cluttered with multiple panes and can't view the Nmap output. To detach a pane into its own window. move the desired pane to move to its own window. Then prefix shift !.

ctrl b shift !

```
'--'
(oo)-----)
(_)
  ||---|| *
tux@tmux_redux:~$
```

GNU nano 2.9.3 finds Modified

important finds

^G Get Help ^O Write Out ^W Where Is

^X Exit ^R Read File ^C Replace

```
//nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1.91 seconds
tux@tmux_redux:~$
```

```
tux@tmux_redux:~$ dirb http://localhost/
```

[0] 0:recon* 1:tmux_redux- "tmux remux" 23:54 30 Jul 20

The Nmap has been sent to a new window named "bash". Now able to view the whole Nmap scan without run the Nmap scan again.

```
Starting Nmap 7.60 ( https://nmap.org ) at 2020-07-30 23:54 UTC
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000080s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://
//nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1.91 seconds
tux@tmux_redux:~$
```

[0] 0:recon- 1:tmux_redux_ 2:bash* "tmux remux" 23:56 30 Jul 20

To cycle between windows can be done with **prefix n** for the next window. Or **prefix p** for the previous windows.

Another method to switch to the desired window is to use **prefix w**. This will list all the tmux windows. Using the arrow keys to select the desired windows and enter-key to select the highlighted window.

```
(0) - 0: 3 windows (attached)
(1) -> + 0: recon- (3 panes)
(2) -> 1: tmux_remux (1 panes) "tmux_remux"
(3) -> 2: bash* (1 panes) "tmux_remux"

[0] 0:recon* 1:tmux_remux 2:bash*          "tmux_remux" 00:01 31-Jul-20
```

Moving back to the recon window. Shows the Nmap pane has been moved and there is more room to work.

```
tux@tmux_remux:~$ cat cow
GNU nano 2.9.3 finds Modified
# important finds
^G Get Help ^O Write Out ^W Where Is
^X Exit      ^R Read File ^Y Replace

tux@tmux_remux:~$ dirb http://localhost/
[0] 0:recon* 1:tmux_remux 2:bash*          "tmux_remux" 00:12 31-Jul-20
```

If a tmux window has been borked and needs to be terminated quickly. Killing the window will also close any panes open within the currently selected tmux window. To kill or close a tmux window do prefix shift & and followed by y to close the window. Or n to keep it open.

ctrl b shift &

```
tux@tmux_remux:~$ nano creds
tux@tmux_remux:~$ ^C
tux@tmux_remux:~$ echo 'reminder'
reminder
tux@tmux_remux:~$ ^C
tux@tmux_remux:~$ ^C
tux@tmux_remux:~$ kill-window recon? (y/n)
[0] 0:recon* 1:nmap-          "tmux_remux" 00:55 31-Jul-20
```

Prompt after the command.

```
tux@tmux_remux:~$ nano creds
tux@tmux_remux:~$ ^C
tux@tmux_remux:~$ echo 'reminder'
reminder
tux@tmux_remux:~$ ^C
tux@tmux_remux:~$ ^C
tux@tmux_remux:~$ kill-window recon? (y/n)
```

Closing the recon window. The name recon is removed from the status bar below.

```
tput localhost
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000080s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Thu Jul 30 23:54:28 2020 -- 1 IP address (1 host up) scanned in 1.91 seconds
tux@tmux_remux:~$ [0] 1:nmap* "tmux_redux" 00:56 31-Jul-20
```

Previously in this task was shown how to break off a pane into its own window. However, sometimes there is a need to fuse two windows together as a single window.

To join two different windows/panes back into one. Can be done with the window name or number. The command is run within tmux prompt or command mode. Prefix shift : then within the prompt :join-pane -t <window-name> OR :join-pane -s <window-name>. Where -s is the source name and -t is the destination window name.

First Enter tmux prompt or command mode from the tmux session.

ctrl b shift :

```
tux@tmux_redux:~$ cat fox
|\_/_|,,----,_~~`_
(._.)--_____)`-}}}
 `o/\ /---~\ \ ~}}}
 _//_ _//_ ~}}
tux@tmux_redux:~$ [0] 0:fox-ascii* 1:cat-ascii- "tmux_redux" 03:12 31-Jul-20
```

Second. Type join-pane with the desired option and window name

join-pane -s <source-window-name>

join-pane -t <destination-window-name>

```
tux@tmux_remux:~$ cat fox
|\_/_|,,----,~~`~_
(._.)--` )`-{}}
\o/\ /---~\ \ ~{}}
_//_ _//_ ~{}}
tux@tmux_remux:~$
```

Third, Hit Enter to run the join-pane command.

For the join-pane commands adding `-v` on the end fuses the two panes together horizontally. Adding `-h` on the end of the join-pane command fuses two panes together vertically.

```
:join-pane -s window-name -v
```

```
:join-pane -s window-name -h
```

Question: How to create a new empty tmux window?

Answer: ctrl b c

Question: How to change the currently select window's name?

Answer: ctrl b ,

Question: How to move the currently selected pane to its own tmux window?

Answer: ctrl b shift !

Question: How to fuse two panes together with the “source” window of “bash”? After entering a tmux prompt?

Answer: `join-pane -s bash`

Question: How to fuse two panes together with the “destination” window of “sudo”?

After entering a tmux prompt?

Answer: **join-pane -t sudo**

Question: What option can be added with question 4 and 5 to fuse together vertically?

Answer: **-v**

Question: What option can be added with question 4 and 5 to fuse together horizontally?

Answer: **-h**

Question: With join-pane can you use the window number instead of the window’s name? yea/nay

Answer: **yea**

Question: How to kill or completely close a window. Including all the panes open on that window. If it’s unresponsive?

Answer: **ctrl b shift &**

Question: How to view and cycle between all the tmux windows for the current tmux session without detaching from the current session?

Answer: **ctrl b w**

Question: How to move back to the previous tmux window?

Answer: **ctrl b n**

Question: How to move up to the next tmux window?

Answer: **ctrl b p**

Task 5: Tmux “copy” mode

Copy mode can be used to scroll up and down the page. If the text is larger than the length of the pane or window size. To start copy mode. Type **ctrl b** [and a number will show up in the top right corner of the current pane or window using copy mode. After scrolling. **Exit copy mode with the g key.**

```
[+] Finding 'username' string inside /home /var/www /var/backups /tmp /etc
/root /mnt (limit 70)
/home/tux/linpeas.sh:    for f in $tomcat; do grep "username=" "$f" 2>/dev
>null | grep "password=" | sed "s,.*,${C}[1;31m&${C}[0m,"; done
/home/tux/privilege-escalation-awesome-scripts-suite/winPEAS/winPEASexe/wi
nPEAS/InterestingFiles.cs:                                string UserName = "";
/home/tux/privilege-escalation-awesome-scripts-suite/winPEAS/winPEASexe/wi
nPEAS/InterestingFiles.cs:                                UserName = "[BLANK]";
/home/tux/privilege-escalation-awesome-scripts-suite/winPEAS/winPEASexe/wi
nPEAS/UserInfo.cs:                               WTSUserName = 5,
[+] Searching specific hashes inside files - less false positives (limit 7
0)
tux@tmux_redux:~$ cp linpeas.sh ~/^C
tux@tmux_redux:~$ █
[0] 0:linpeas enum*                                     "tmux_redux" 06:02 31-Jul-20
```

Allows scrolling through linpeas output.

```
ider                                         [796/1878]
-rw-r---- 1 root adm 31 Jul 10 20:48 /snap/core/9665/var/log/dmesg
-rw-r---- 1 root adm 31 Jul 10 20:48 /snap/core/9665/var/log/fsck/checkfs
-rw-r---- 1 root adm 31 Jul 10 20:48 /snap/core/9665/var/log/fsck/checkro
ot

[+] Modified interesting files in the last 5mins (limit 100)
/var/log/auth.log
/var/log/lxd/lxd.log
/var/log/syslog
/var/log/journal/e6e2e310d2e341a1b1887d1fbb5b0676/user-1000.journal
/var/log/journal/e6e2e310d2e341a1b1887d1fbb5b0676/system.journal
/var/log/kern.log
/home/tux/.gnupg/trustdb.gpg
/home/tux/.gnupg/crls.d/DIR.txt
/home/tux/.gnupg/pubring.kbx
/home/tux/.config/lxc/config.yml
[0] 0:linpeas enum*                                     "tmux_redux" 06:03 31-Jul-20
```

When manually searching for the block of text to copy. It can be easy to miss. This can be quickly solved by searching up or down the page.

To search up the page. Within copy mode. Type **ctrlr** and the search term or string to search for. To continue searching the same string. Hold down ctrl and press r again to jump to the next string location. Resume scrolling by hitting the enter key. Will stop at the search string/term.

```
/home/tux/privilege-escalation-awesome-scripts-su(3/3 results) [1836/1878]
tux@tmux_remux:~/privilege-escalation-awesome-scripts-suite/linPEAS$ ls
images linpeas.sh README.md
tux@tmux_remux:~/privilege-escalation-awesome-scripts-suite/linPEAS$ cp linpeas.sh ~/
tux@tmux_remux:~/privilege-escalation-awesome-scripts-suite/linPEAS$ cd ~/
tux@tmux_remux:~$ ls
another_omt horse plugins_for_tmux
armadillo linpeas.sh privilege-escalation-awesome-scripts-suite
cat.txt oreo.sh tmux-config.bak
cow output
fox parrot
tux@tmux_remux:~$ ./linpeas.sh

[search up] ./linpeas.sh
```

To search down the page. Is the same basic idea as search up. However, this is done with s instead of r as **ctrls** and then the search term or string. While holding down ctrl retype s to jump to the next string location. Again resume scrolling by hitting the enter key. Will stop at the search string/term.

```
(66/66 results) [1234/1878]
===== ( Users Information ) =====
=====
[+] My user
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#groups
uid=1000(tux) gid=1000(tux) groups=1000(tux),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),108(lxd)

[+] Do I have PGP keys?

[+] Clipboard or highlighted text?
xsel and xclip Not Found

[+] Testing 'sudo -l' without password & /etc/sudoers
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#commands-with-sudo-and-suid-commands

[search down] password
```

To copy and paste within tmux copy-mode takes 4 steps. Note that this method will only apply to the tmux clipboard as follows.

0) **ctrl b** [# enter copy mode]

1) scroll to the start of the block of text you would like to copy

```
===== ( Network Information ) ===== [1309/1878]
=====
[+] Hostname, hosts and DNS
tmux_remux
127.0.0.1 localhost
127.0.1.1 tmux_remux

::1      ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

nameserver 127.0.0.53
options edns0

[+] Content of /etc/inetd.conf & /etc/xinetd.conf
[0] 0:linpeas enum*                                     "tmux_remux" 06:08 31-Jul-20
```

2) enable highlighting with **ctrl spacebar**. Then use the arrow keys to up to select all the text. Down if you start from the top instead.

```
===== ( Network Information ) ===== [1309/1878]
=====
[+] Hostname, hosts and DNS
tmux_remux
127.0.0.1 localhost
127.0.1.1 tmux_remux

::1      ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

nameserver 127.0.0.53
options edns0

[+] Content of /etc/inetd.conf & /etc/xinetd.conf
[0] 0:linpeas enum*                                     "tmux_remux" 06:09 31-Jul-20
```

3) copy all the highlighted text to the tmux clipboard with **alt w**. Note! Even though the highlight will disappear, the text still copied to the tmux clipboard.

4) create a new file to paste the final text with **ctrl b**].

```
GNU nano 2.9.3                               network linpeas                               Modified

fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

nameserver 127.0.0.53
options edns0
```

To double-check what the text that is currently copied to the tmux clipboard. Do prefix followed by shift #

ctrl b shift #

Note. Even though the text might look broken don't worry it will paste as it was copied within a terminal file editor. To quit back to the terminal. Hit the **q** key once.

```
buffer0000: 217 bytes: ".....\012.....:::///,::-[0/0]
,:'////////`:::/|\012...../\012.....:://``\012.....'
,||| |`/(\ e \\\\012---_-\`\\_X_`\\_\\_/_/`_`...."
```

Pasting again with prefix] shows that it pasted corrected from the origin ASCII image.

ctrl b]

Question: How to start copy mode?

Answer: ctrl b [

Question: While in copy mode. How to search/grep up the wall of terminal text?

Answer: ctrl r

Question: While in copy mode. How to search/grep down the wall of terminal text?

Answer: ctrl s

Question: How to exit search up or search down within copy mode?

Answer: esc

Question: What single key can also be used to exit out of copy mode.

Answer: q

Question: After starting copy mode. How do you enable text highlighting to select for text copying?

Answer: ctrl spacebar

Question: After selecting the text you want to copy. How do copy it?

Answer: alt w

Question: When in a terminal text editor. How to paste from the tmux clipboard?

Answer: ctrl b]

Question: How to double check what is currently copied to the tmux clipboard

Answer: ctrl b shift #

Task 6: Oh My Tmux and beyond

The last four tasks have covered commands of how to use tmux with its defaults. Tmux by default doesn't have a default configuration file. That doesn't mean you can make one.

Before crafting a custom configuration file for tmux. It might be a good idea to show all the options of the defaults. With tmux show -g argument for global.

```
tux@tmux_remux:~$ tmux show -g [35/35]
activity-action other
assume-paste-time 1
base-index 0
bell-action any
default-command ""
default-shell "/bin/bash"
destroy-unattached off
detach-on-destroy on
display-panes-active-colour red
display-panes-colour blue
display-panes-time 1000
display-time 750
history-limit 2000
key-table "root"
lock-after-time 0
lock-command "lock -np"
[0] 0:[tmux]* "tmux_remux" 06:15 31-Jul-20
```

When creating the tmux configuration script it is important that. The file name is .tmux.conf and .tmux.conf is saved within the user's home directory /home/username/.tmux.conf

```
GNU nano 2.9.3 .tmux.conf
set -g status-style bg='#44475a',fg='#8be9fd'
setw -g window-status-current-style fg=black,bg=white

set -g history-limit 9001

set -g prefix C-a

bind-key | split-window -h -c "#{pane_current_path}"
bind-key - split-window -v -c "#{pane_current_path}"
[ Wrote 9 lines ]

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify
^X Exit ^R Read File ^N Replace ^U Uncut Text ^T To Spell
```

`set -g status-style bg='#44475a',fg='#8be9fd'`

This line is to modify the color of the status bar.

`setw -g window-status-current-style fg=black,bg=white`

This line adds highlight in the color of white for the currently selected window.

set -g prefix C-a

Allow changing the prefix from ctrl b to ctrl a. If there is any need to use the alt key instead of ctrl. Then set prefix as M-b or another second character, such as M-a.
Changing the prefix from ctrl b to alt a.

#IMPORTANT NOTE

bind or bind-key options allow adding extra hotkey options without overwriting the default hotkeys
set options overwrite the default tmux hotkeys for the updated ones written within the .tmux.conf file

Using bind-key with | and - to add extra ways to split-window and create a new pane. The -c argument with "#{pane_current_path}" on the end. Allows any new pane within a different directory to keep that directory path. Even if it is not the directory that session had started in.

ctrl b shift | # will split horizontally

ctrl b - # will split vertically

```
tux@tmux_redux:~$ [0] 0: bash 1: bash- 2: bash* [0] 0: [tmux]* 2: bash- "tmux_redux" 07:00 31-Jul-20
```

The set history limit allows copying mode to have a higher max character limit. Useful for copying output that goes beyond the default tmux history limit.

```
/tmp /etc /root /mnt (limit 70) [0/5354]
[+] Finding 'username' string inside /home /var/www /var/backups /tmp /etc /root /mnt (limit 70)
/home/tux/linpeas.sh: for f in $tomcat; do grep "username=" "$f" 2>/dev/null | grep "password=" | sed "s,.*,$C[1;31m&${C}[0m,"; done
/home/tux/privilege-escalation-awesome-scripts-suite/winPEAS/winPEASexe/winPEAS/InterestingFiles.cs: string UserName = "";
/home/tux/privilege-escalation-awesome-scripts-suite/winPEAS/winPEASexe/winPEAS/InterestingFiles.cs: UserName = "[BLANK]";
/home/tux/privilege-escalation-awesome-scripts-suite/winPEAS/winPEASexe/winPEAS/UserInfo.cs: WTSUserName = 5,
[+] Searching specific hashes inside files - less false positives (limit 70)
tux@tmux_redux:~$ [0] 0: [tmux]* 2: bash- "tmux_redux" 07:11 31-Jul-20
```

The left side of the status bar can be customized with status-left within set or set-option. In the example below the first one status-left-length 15. Set the total amount of characters that can be shown on the left. The following line displays 'Try Harderer' with the echo Linux command.

```
set-option -g status-left-length 15
set -g status-left "#[fg=\"green,bold\"]#(echo 'Try Harderer ')"
```

~
~
~

".tmux.conf" 13L, 355C written 13,0-1 All

:source-file ~/.tmux.conf

After reloading the tmux configuration file within the session ctrl b shift : followed by source-file /home/username/.tmux.conf OR source-file ~/tmux.conf

NOTE the .tmux.conf can also be reloaded by closing and restarting a new session. Or with the tmux command as

```
tmux source /home/username/.tmux.conf
```

Try Harderer 0:vim*

When adding the output of Linux commands to the status bar. Sometimes commands will not function correctly. To resolve the issue. Put the desired one-liner or script into a file. Then make it executable with chmod or it won't work.

```
tux@tmux_remux:~$ nano script.sh
tux@tmux_remux:~$ chmod +x script.sh
tux@tmux_remux:~$ cat script.sh
#!/bin/bash
whoami
tux@tmux_redux:~$
```

Try Harderer 0:vim- 1:bash* "tmux_redux" 17:08 31-Jul-20

Instead of running a Linux command. The following Linux includes the path to the script.

```
set-option -g status-left-length 15
set -g status-left "#[fg=\"green,bold\"]#(echo 'Try Harderer ')"
```

set -ga status-right "#[fg=\"red,bold\"]#(~script.sh) "

".tmux.conf" 15L, 412C written 15,0-1 Bot

:source-file ~/.tmux.conf

Reloading the .tmux.conf file again shows the output of whoami from the script.sh

```
set -ga status-right "#[fg=\"red,bold\"]#(~script.sh) "
```

".tmux.conf" 15L, 412C written 15,0-1 Bot

Try Harderer 0:vim- 1:bash- "tmux_redux" 17:12 31-Jul-20 tux

In addition to calling script though .tmux.conf is calling custom plugins. The plugins are shown were sourced from GitHub.

<https://github.com/tmux-plugins/tmux-resurrect.git> # save sessions to resume after a system reboot.

<https://github.com/tmux-plugins/tmux-logging.git> # plugin for creating command history to review later.

<https://github.com/tmux-plugins/tmux-yank.git> # plugin copy text from outside the tmux clipboard to use with other programs.

For the example below add the two following lines for tmux-logging. A plugin that can take pane screenshots, per pane history, and complete history logging.
NOTE. Double-check install instructions from their Github page if something doesn't follow up.

```
set -ga status-right "#[fg='red,bold']#(-/script.sh) "
set -g @plugin '/home/tux/plugins_for_tmux/tmux-logging'
run-shell /home/tux/plugins_for_tmux/tmux-logging/logging.tmx
".tmux.conf" 18L, 533C written           18.0-1      Bot
:source-file ~/.tmux.conf
```

Another .tmux.conf reload. To start command history logging for the currently selected pane do prefix shift p. This will start a log file within the current user's home directory.

ctrl b shift P

```
tux@tmux_remux:~$ tmux source ~/.tmux.conf
tux@tmux_redux:~$ nano ~/.tmux.conf
tux@tmux_redux:~$ tmux source ~/.tmux.conf
tux@tmux_redux:~$
```

Ended logging to /home/tux/tmux-0-1-0-20200803T032730.log

Although there a problem. When reloading the .tmux.conf file. The whoami script.sh from before prints a duplicate string.

```
tux@tmux_redux:~$ 
Try Harder 0:vim- 1:bash* "tmux_redux" 03:29 03-Aug-20 tux tux
```

This is because the first set status-right had used the -a or append argument. Append give to option without having to redo everything. However when reloading the tmux configuration .tmux.conf file. It prints duplicates. This can be resolved by only using the -g or global argument.

```
#set -ga status-right "#[fg='red,bold']#(-/script.sh) "
set -g status-right "#[fg='red,bold']#(-/script.sh) "
```

Reloading the .tmux.conf file. Can see that only using the global argument -g alone removes the hostname, time, and date from the status-right. Whoops.

```
tux@tmux_remux:~$
```

Resolving the issue will be done by value in the same order as was with -ga.

#H is the place holder for the hostname.

%H:%M is the hour and minutes of the clock.

%d-%b-%y will follow last for the day, month, and year.

```
#set -ga status-right "#[fg="red,bold"]#(~/.script.sh) "
#set -g status-right "#[fg="red,bold"]#(~/.script.sh) "
set -g status-right "#H %H:%M %d-%b-%y #[fg="red,bold"]#(~/.script.sh) "
```

A .tmux.conf reload later and everything is back to normal. Any attempt to reload the .tmux.conf file doesn't print duplicate status-right from the script.sh file.

```
tux@tmux_remux:~$ tmux source ~/.tmux.conf
tux@tmux_remux:~$ tmux source ~/.tmux.conf
tux@tmux_remux:~$ tmux source ~/.tmux.conf
tux@tmux_remux:~$
```

Not a fan of copy modes hotkeys. Well by default they are set to emacs.

```
status-justify left
status-keys emacs
status-left "#%c"
```

To changes, this adds the set -g mode-keys vi. Save the .tmux.conf script and reload. To copy with this setting.

- 1) **ctrl b** [# enter copy mode
- 3) # arrow key to the start or end of what you would like to copy
- 2) **spacebar** # followed by the arrow keys to highlight all the text
- 3) **enter** # copy the highlighted text to the tmux clipboard
- 4) **ctrl b]** # paste the tmux clipboard

```
set -g mode-keys vi
~
~
".tmux.conf" 21L, 684C written
```

To make copy mode feel more like as if copying blocks of text in vi file editor. The code between mouse on and the last two lines.

This allows has mouse mode on. That allows us the use of the mouse to resize tmux panes or scroll without needing to use the hotkeys to enter copy mode.

The last two lines below allow anything though vi y key to be also saved secondary Linux keyboard. The secondary keyboard is the ctrl c ctrl v. This means that the copy text will be pasted from the ctrl v buffer. Instead of the copied text being trapped within tmux.

```
set-option -g mouse on
set-option -s set-clipboard off
bind P paste-buffer
bind-key -T copy-mode-vi v send-keys -X begin-selection
bind-key -T copy-mode-vi y send-keys -X rectangle-toggle
unbind -T copy-mode-vi Enter

bind-key -T copy-mode-vi Enter send-keys -X copy-pipe-and-cancel 'xclip -se c -i'
bind-key -T copy-mode-vi MouseDragEnd|Pane send-keys -X copy-pipe-and-cancel 'xclip -se -c -i'
```

The alternative to building a custom .tmux.conf configuration script it utilize a open-source .tmux.conf script. That can be easily found by searching online with Oh My Tmux. For this example will use the tmuxconfig repository from github. that can be found at the link below.

<https://github.com/chaosma/tmuxconfig.git>

First. Change the .tmux.conf to something new. For example, changing the name from .tmux.conf to custom-tmux.bak

```
tux@tmux_redux:~$ mv .tmux.conf custom-tmux.bak
tux@tmux_redux:~$
```

Second. Install the tmuxconfig repository with git clone and copy the hidden .tmux.conf file to the home directory of /home/username/

```
tux@tmux_redux:~/plugins_for_tmux/tmuxconfig$ ls -al
total 112
drwxrwxr-x 3 tux tux 4096 Aug  1 01:04 .
drwxrwxr-x 7 tux tux 4096 Aug  3 00:56 ..
drwxrwxr-x 8 tux tux 4096 Aug  1 00:51 .git
-rw-rw-r-- 1 tux tux 1057 Aug  1 00:51 LICENSE.MIT
-rw-rw-r-- 1 tux tux 479 Aug  1 00:51 LICENSE.WTFPLv2
-rw-rw-r-- 1 tux tux 2264 Aug  1 00:51 PowerlineSymbols.otf
-rw-rw-r-- 1 tux tux 11804 Aug  1 00:51 README.md
-rw-rw-r-- 1 tux tux 55425 Aug  1 00:51 .tmux.conf
-rw-rw-r-- 1 tux tux 11454 Aug  1 00:51 .tmux.conf.local
-rw-rw-r-- 1 tux tux 5230 Aug  1 00:51 .zshrc
tux@tmux_redux:~/plugins_for_tmux/tmuxconfig$ cp .tmux.conf /home/tux/
tux@tmux_redux:~/plugins_for_tmux/tmuxconfig$
```

Third run a new tmux session or reload the tmux configuration .tmux.conf file to see changes

To change scripts back replace the .tmux.conf file again in the home directory or call the desired .tmux.conf file when starting a new tmux session with the -f argument for configuration file

tmux -f /home/username/custom-tmux.bak

NOTE. That using the tmuxconfig or other tmux.conf configuration files is the only one that loads in. Regardless if the .tmux.conf had been deleted or replaced with another .tmux.conf file. Tmux can be reset back to its default configuration by killing the tmux server. with kill-server

IMPORTANT WARNING. Before running the kill-server command backup and save everything. On every session, window, and pane because this command will stop everything.

tmux kill-server

Question: Does tmux have a default tmux.conf config file? yea/nay

Answer: nay

Question: Where can you find examples for custom tmux.conf config files?

Answer: /usr/share/doc/tmux

Question: Can you use Hex color codes in place of the color name? yea/nay

Answer: yea

Question: What directory must the .tmux.conf be put in to work with the next tmux session

Answer: home

Question: How would you update tmux changes without quitting the tmux session from a tmux prompt?

Answer: source-file~/.tmux.conf

Question: How to completely reset tmux to its default and kill all sessions? If the .tmux.conf is borked.

Answer: :tmux kill-server

Question: How would you select addition hotkeys. Without overwriting the default hotkey?

Answer: bind

Question: How would you change the prefix to Ctrl a?

Answer: set -g prefix C-a

Question: Can you display shell command output. From a script or one line command? yea/nay

Answer: yea

Question: How would you load a plugin into a tmux config file?

Answer: set -g @ plugin

Question: How can you run the desired plugin after loading it?

Answer: run-shell

Task 7: Oreo's open-source .tmux.conf file

Hope you enjoyed my first room. This script was created while testing the Tmux Remux room. Where the network interfaces will be checked for the OpenVPN network interface name of "tun0". If "tun0" is up. The user's TryHackMe localhost address will be shown at the bottom right of their tmux. If "tun0" is down. An error message will be shown "OpenVPN Not Active". In the bottom right of their tmux.

[Example screenshot of the tmux script printing the error since THM.vpn isn't running.](#)

```
tmux remux 05:54 29-Jul-20 Broken-VPN-Connection OpenVPN Not Active
```

[Example screenshot of the tmux script printing the current THM.vpn user's IP address.](#)

```
tmux remux 05:58 29-Jul-20 THM-VPN-Connected 10.9.13.234 THM-Tun0 10.9.13.234
```

[NOTE! Using these not on THM or with a different network interface that isn't tun0 will print the error messages.](#)

[Here is a quick example script of installing my custom .tmux.conf script.](#)

```
#!/bin/bash
# also comes with the repo as oreo-setup-script.sh
cd ~/
git clone https://github.com/OreoByte/another_omt.git
chmod +x ~/another_omt/thm_vpn
chmod +x ~/another_omt/curl_thm_vpn
cp ~/another_omt/simple_tmux.conf ~/tmux.conf
```

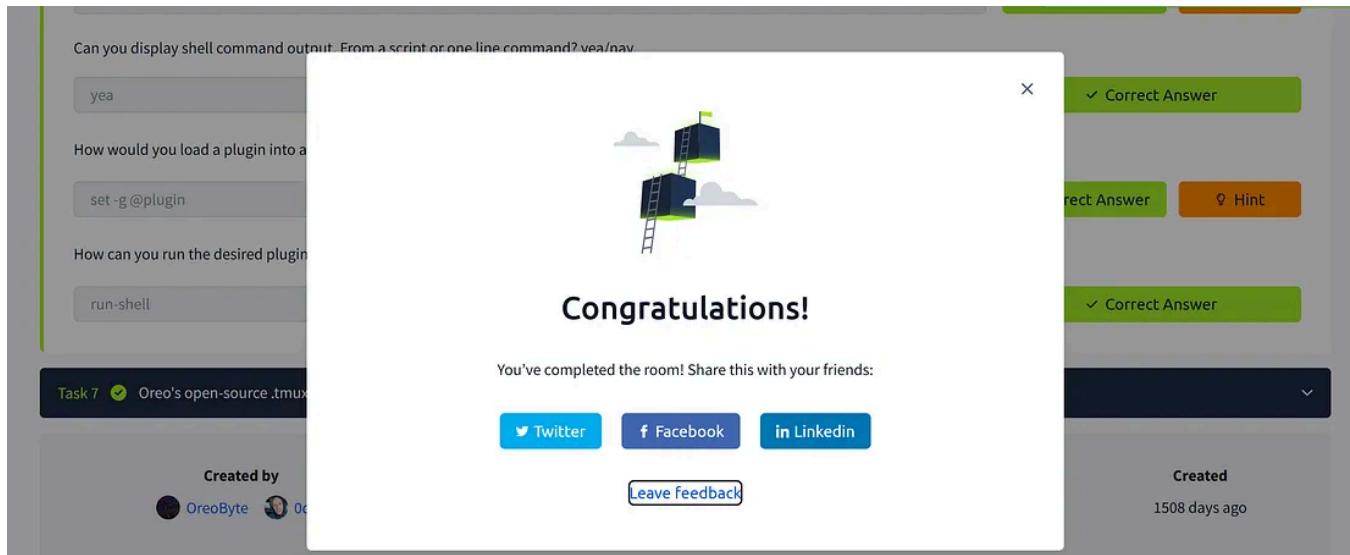
[#Great thanks to 0day, alanfoster, and briskets for helping me with this room.](#)

Answer the questions below

git clone or fork it

No answer needed

◀ Complete



Tryhackme Walkthrough

Tryhackme Writeup

Ethical Hacking Training

[Follow](#)

Written by Daniel Schwarzenraub

116 Followers · 4 Following

PNW_Hacker

No responses yet



What are your thoughts?

[Respond](#)

More from Daniel Schwarzenraub

A dark-themed screenshot of a terminal window. The visible text includes "r a few minutes until all machine r" and "{"status": "running"} when visiting".

 Daniel Schwarzenraub

HTB—Tier 1 Starting Point: Three

HTB—Tier 1 Starting Point: Three

Jul 20, 2023  4  2



...

s to introduce users to basic cryptography concepts such as:

n, such as AES

on, such as RSA

xchange

a message that no one can understand except the intended recip

 Daniel Schwarzenraub

Tryhackme: Introduction to Cryptography

Tryhackme: Introduction to Cryptography

Sep 26, 2023  2



...

```
./.../HackTheBox/Starting_Point  
9.124.107 -T 4 -VV  
( https://nmap.org ) at 202  
an at 20:56  
4.107 [2 ports]  
n at 20:56, 0.09s elapsed (1  
1 DNS resolution of 1 host)
```



Daniel Schwarzenraub

HTB—Tier 2 Starting Point: Archetype

HTB—Tier 2 Starting Point: Archetype

Jul 21, 2023



...

erative that we understand and can protect against common attacks.

mon techniques used by attackers to target people online. It will also teach some of the best wa



Daniel Schwarzenraub

Tryhackme Free Walk-through Room: Common Attacks

Tryhackme Free Walk-through Room: Common Attacks

Sep 13, 2024



...

See all from Daniel Schwarzenraub

Recommended from Medium



 Francesco Pastore

THM - Lookup

A writeup for the room Lookup on TryHackMe.

Nov 25, 2024  2



...



 Jawstar

Advent of Cyber 2024 {DAY - 18} Tryhackme walkthrough

Author : - Jawstar

◆ Dec 21, 2024  3



...

Lists



Staff picks

796 stories . 1560 saves



Stories to Help You Level-Up at Work

19 stories . 912 saves



Self-Improvement 101

20 stories . 3195 saves



Productivity 101

20 stories . 2707 saves



 In T3CH by Axoloth

TryHackMe | FlareVM: Arsenal of Tools| WriteUp

Learn the arsenal of investigative tools in FlareVM

Nov 28, 2024  50



...

 **Advent of Cyber 2024**

Dive into the wonderful world of cyber security by engaging in festive beginner-friendly exercises every day in the lead-up to Christmas!

 Easy  1440 min

Maybe SOC-mas music, he thought, doesn't come from a store



**Day 1
Answers**

cyberw1ng.medium.com

 In System Weakness by Karthikeyan Nagaraj

Advent of Cyber 2024 [Day 1] Writeup with Answers | TryHackMe Walkthrough

Maybe SOC-mas music, he thought, doesn't come from a store?

Dec 1, 2024 906 1



In T3CH by Axoloth

TryHackMe | Training Impact on Teams | WriteUp

Discover the impact of training on teams and organisations

Nov 5, 2024 60



Sticker Shop

Get the sticker shop in order to capture the flag?

20 min

Start AttackBox Help Save Room Options

Room completed (100%)

Shakhawat Hossain - OxShakhawat

The Sticker Shop | TryHackMe | Walkthrough

How I Solved The Sticker Shop CTF: Exploiting Blind XSS to Capture the Flag. This writeup walks you through the steps of exploiting a Blind...

Nov 30, 2024

112

4



...

See more recommendations