HackerFrogs Afterschool
Network Hacking – Session 4

Class:
Network Hacking

Workshop Number:
AS-NET-04

Document Version:
1.75

Special Requirements:
Registered account
at tryhackme.com

Welcome to HackerFrogs Afterschool!

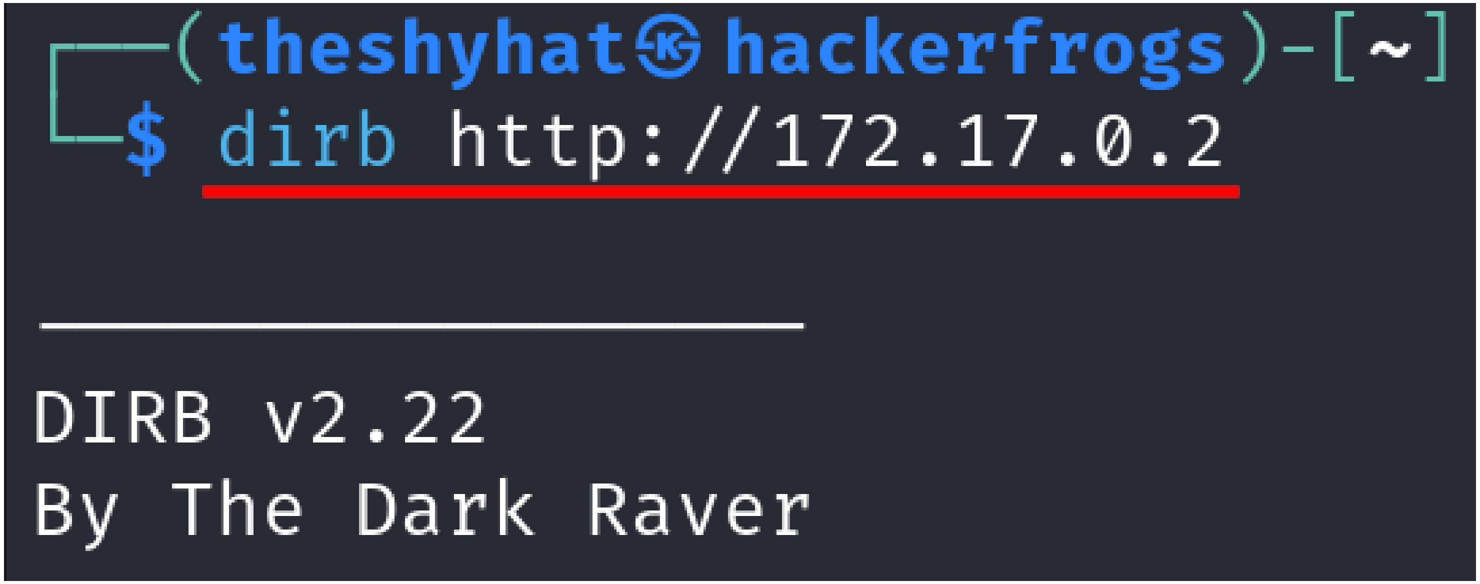
This is the fourth session for network hacking!

Let's go over the concepts we covered in the previous session!

What is Dirbusting?

Directory busting (dirbusting) is the act of determining what endpoints (files and directories) exist on a web app, by trying to access those endpoints. It is a type of brute force attack

Dirb Tool

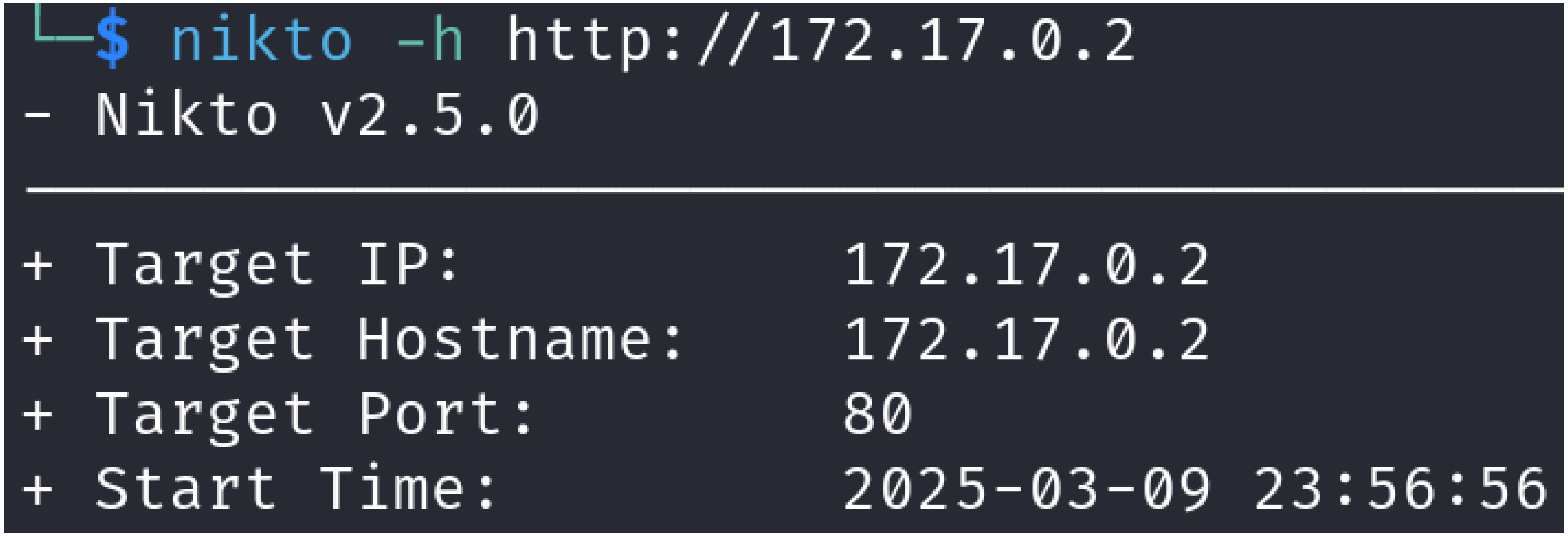


```
┌──(theshyhat㉿hackerfrogs)-[~]
└─$ dirb http://172.17.0.2

-----------------
DIRB v2.22
By The Dark Raver
```

The Dirb tool is an older directory busting tool, and but it's a good first program to run on web servers--

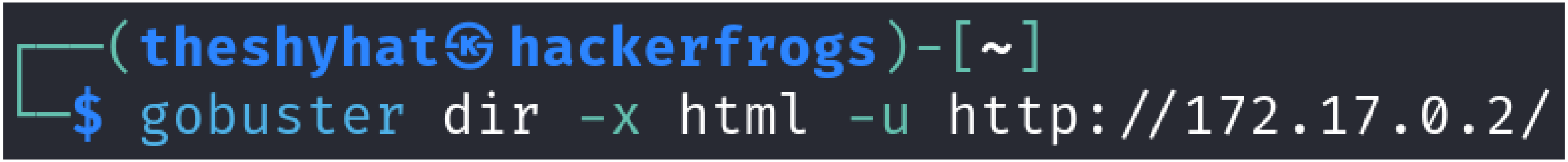
Nikto Tool



```
└─$ nikto -h http://172.17.0.2
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          172.17.0.2
+ Target Hostname:    172.17.0.2
+ Target Port:        80
+ Start Time:         2025-03-09 23:56:56
```

The Nikto tool is a web app vulnerability scanner, which attempts to ID insecure configurations and general web app settings

Gobuster Tool



```
┌──(theshyhat㉿hackerfrogs)-[~]
└─$ gobuster dir -x html -u http://172.17.0.2/
```

The Gobuster tool is a much more powerful directory busting tool than Dirb, however its command syntax is much more complex

This Session's Topics

- What is Remote Shell Access?

- Reverse Shells

- The Limitations of Netcat Shells

- Bind Shells

What is Remote Shell Access?

A remote shell is command-line interface (CLI) access to a remote server. This allows OS commands to be run through the remote shell

What is Remote Shell Access?

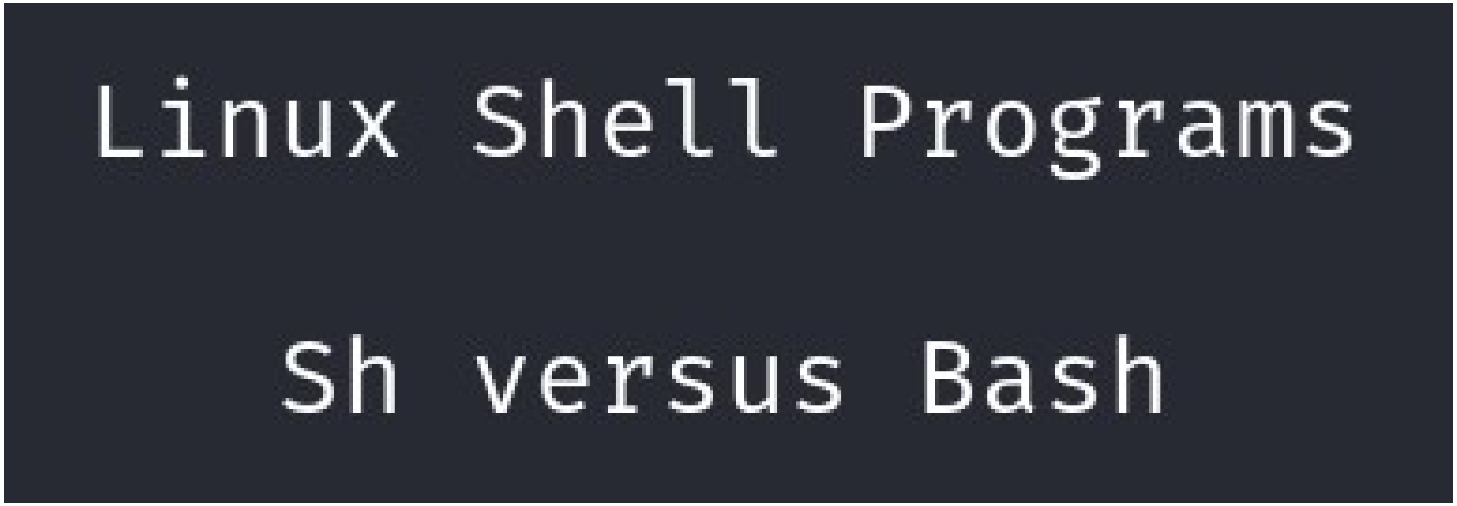
Typically remote shell access is gained through programs like SSH, but security testers typically use other programs to establish remote shell access, such as Netcat

Accessing TryHackMe

Let's access this TryHackMe room to learn about remote shell access:

https://tryhackme.com/room/c2carnage

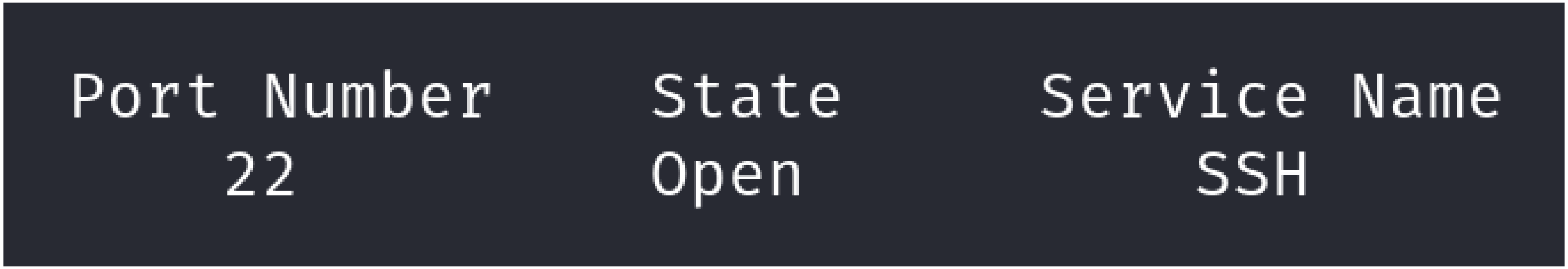
Linux Shell Programs


Linux Shell Programs

Sh versus Bash

For Linux, there are a few common shell programs: the most typical ones being Bash and Sh

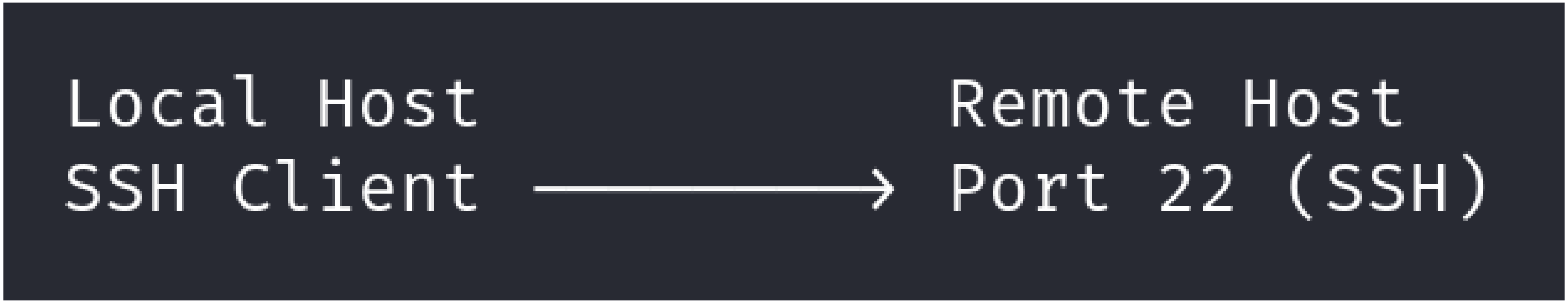
Reverse Shells



```
Port Number    State      Service Name
    22         Open           SSH
```

In a typical networking environment, servers open ports to be connected to for remote shell access, and this is fine for legitimate shell access

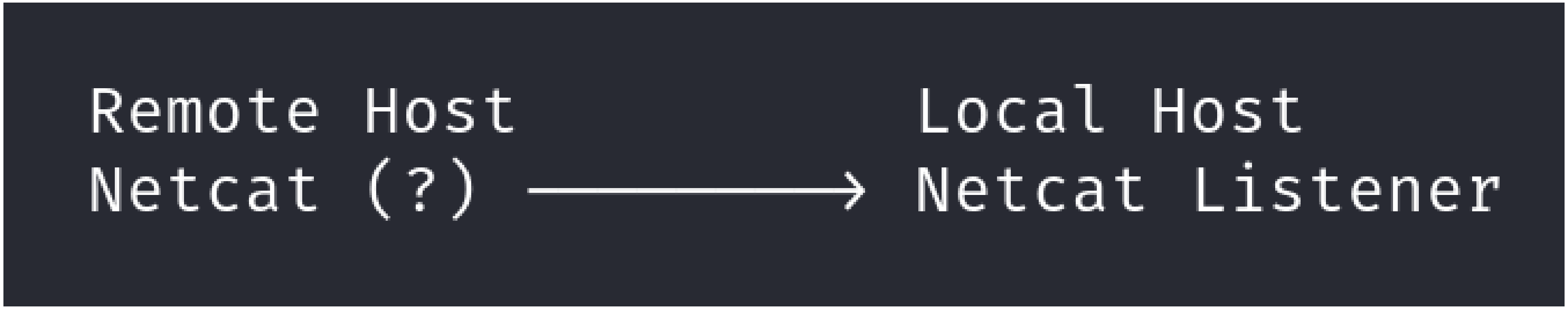
Reverse Shells



```
Local Host                 Remote Host
SSH Client  ───────────→   Port 22 (SSH)
```

Normally, the local host (your device) will connect to the remote host to create the remote shell access

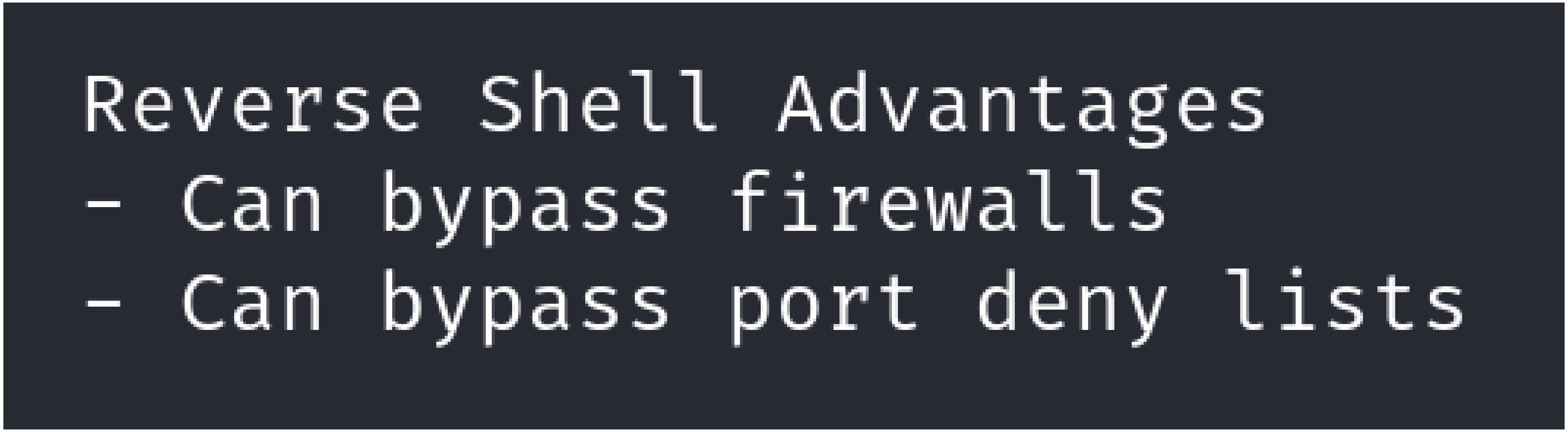
Reverse Shells



```
Remote Host                Local Host
Netcat (?) ──────────→ Netcat Listener
```

However, network security testers find it easier to have the remote host connect to their local host, creating what is called a “reverse shell” connection

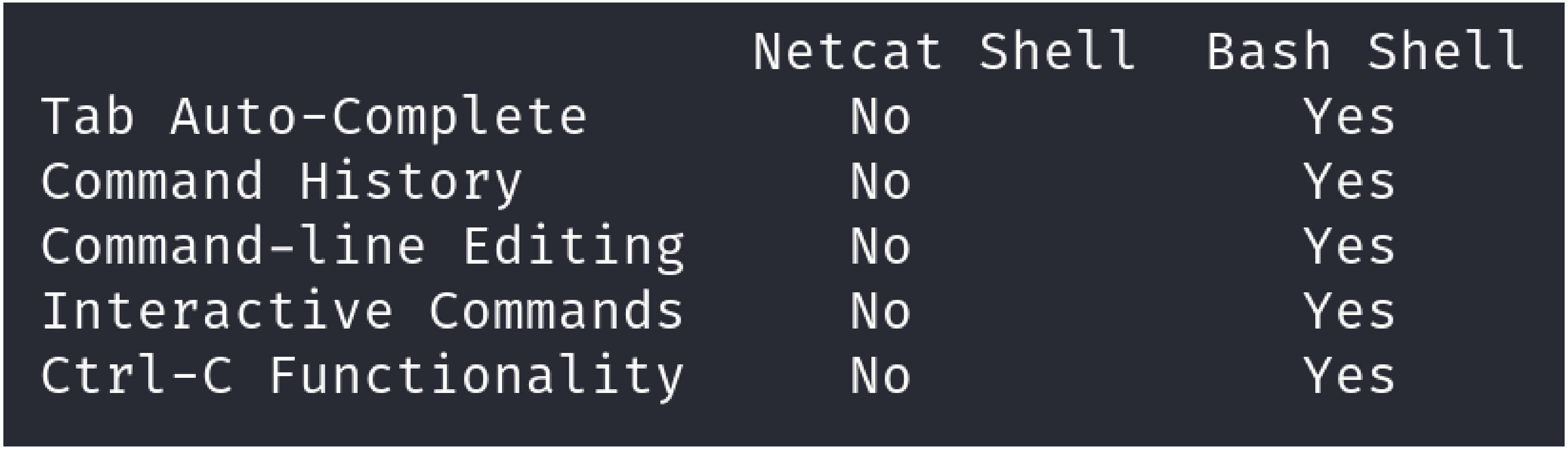
Reverse Shells



```
Reverse Shell Advantages
- Can bypass firewalls
- Can bypass port deny lists
```

Reverse shells are often preferable because they can bypass certain restrictions in the remote host environment

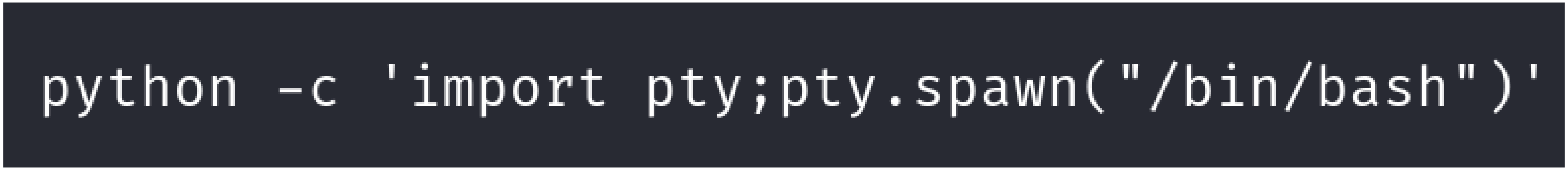
The Limitation of Netcat Shells



	Netcat Shell	Bash Shell
Tab Auto-Complete	No	Yes
Command History	No	Yes
Command-line Editing	No	Yes
Interactive Commands	No	Yes
Ctrl-C Functionality	No	Yes

In network CTF environments, reverse shells are often created with Netcat, but Netcat shells are functionally limited compared to standard shells

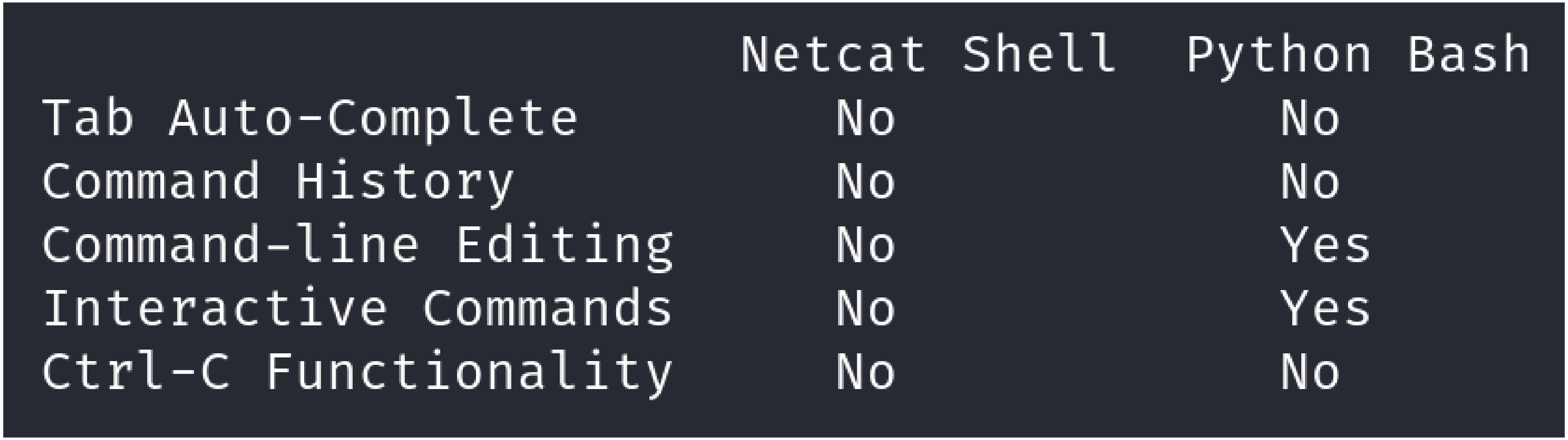
The Limitation of Netcat Shells



```
python -c 'import pty;pty.spawn("/bin/bash")'
```

We often use Python to upgrade Netcat shells

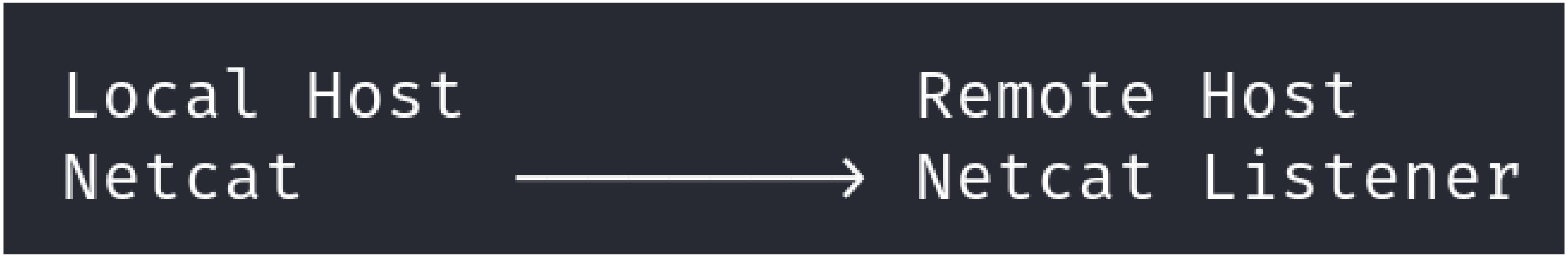
The Limitation of Netcat Shells



```
                         Netcat Shell  Python Bash
Tab Auto-Complete           No              No
Command History             No              No
Command-line Editing        No              Yes
Interactive Commands        No              Yes
Ctrl-C Functionality        No              No
```

This gives the Netcat shell a little bit more functionality

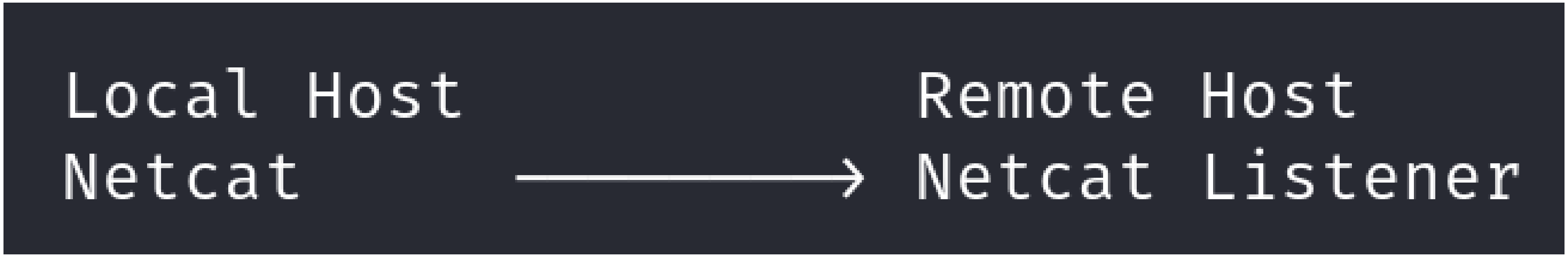
Bind Shells



```
Local Host                   Remote Host
Netcat       ─────────────>  Netcat Listener
```

The alternative to reverse shells is bind shells, where an open port is created on the remote host and allows the local host to connect to it for access

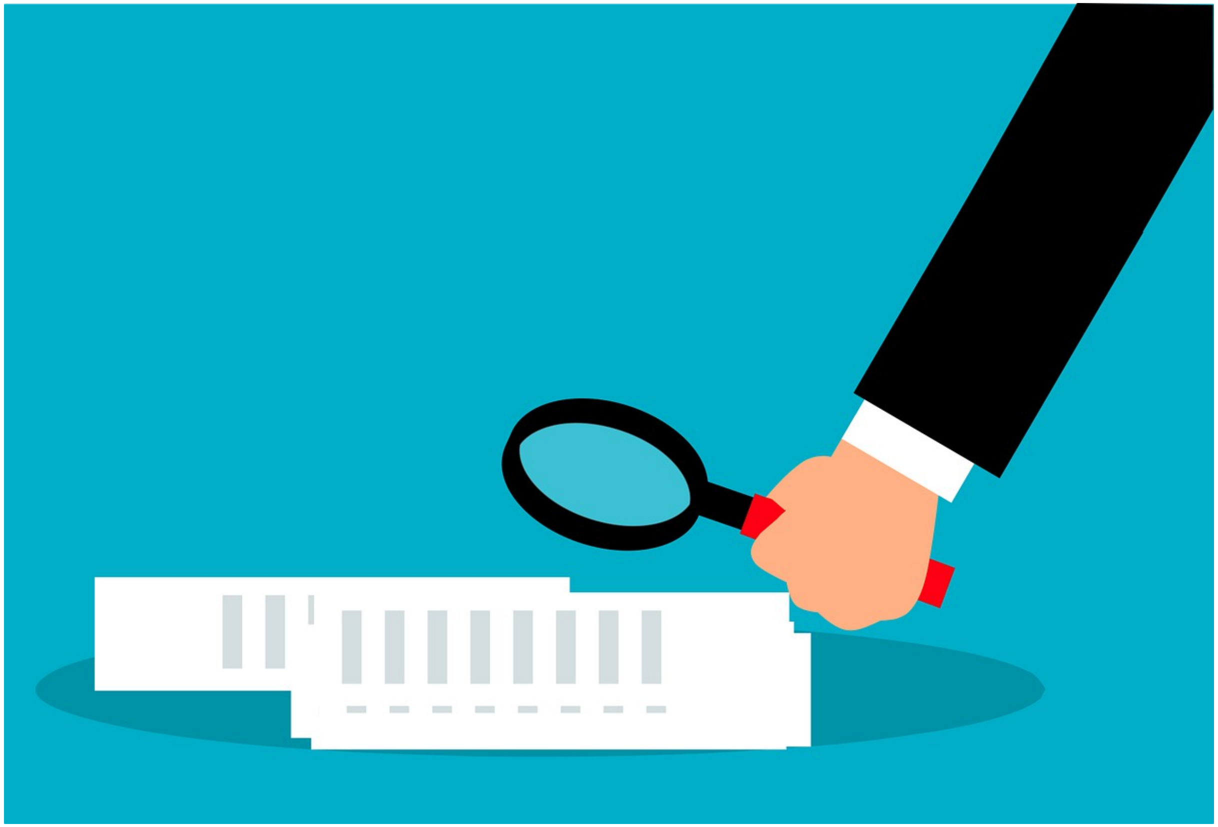
Bind Shells



```
Local Host                   Remote Host
Netcat       ──────────→     Netcat Listener
```

One important reason why bind shells are avoided is because they expose an open port to the network which any device could connect to

Summary

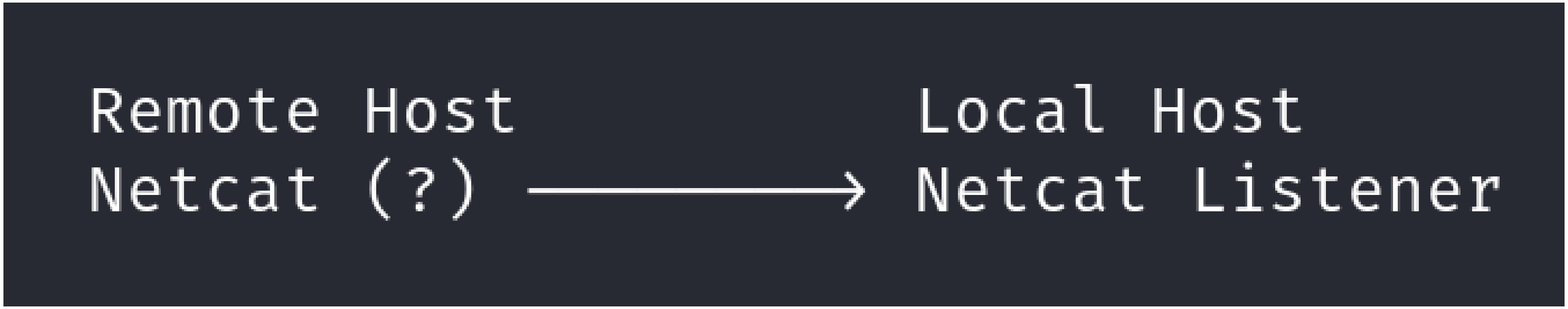


Let's review the network hacking concepts we learned in this workshop:

What is Remote Shell Access?

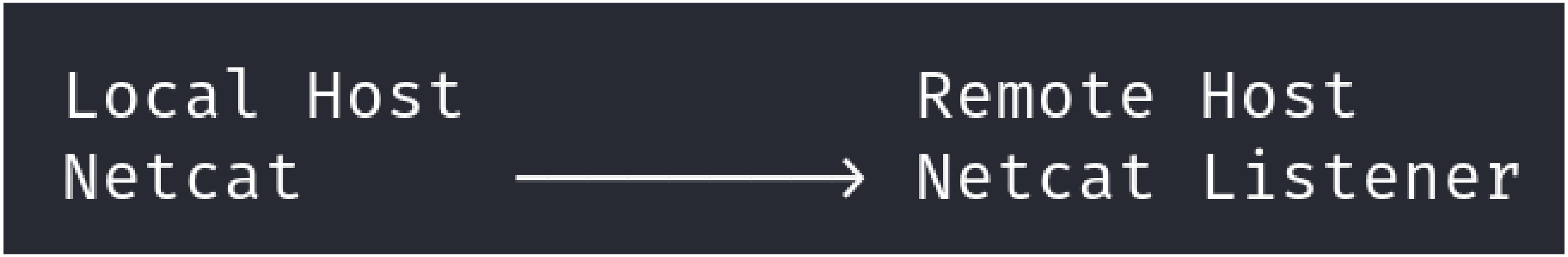
A remote shell is command-line interface (CLI) access to a remote server. This allows OS commands to be run through the remote shell

Reverse Shells



Reverse shell access is where a listening port is created on the local host and the connection is established by the remote host connecting to that port

Bind Shells



```
Local Host                Remote Host
Netcat     ──────────→    Netcat Listener
```

And bind shells access is created when the remote host opens a networking port which is then connected to by the local host

What's Next?

In the next HackerFrogs Afterschool Network Hacking workshop, we'll be learning about how to crack password hashes!