By Falko Timme
Published: 2008-12-04 13:16

Useful Uses Of netcat

Version 1.0
Author: Falko Timme <ft [at] falkotimme [dot] com>
Last edited 11/28/2008

This short article shows some useful netcat commands. netcat is known as the TCP/IP swiss army knife. From the netcat man page: ***netcat 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. At the same time, it is a feature-rich network debugging and exploration tool, since it can create almost any kind of connection you would need and has several interesting built-in capabilities.***

I do not issue any guarantee that this will work for you!

1 Preliminary Note

I'm using two systems in this article:

- *server1.example.com*: IP address *192.168.0.100*
- *server2.example.com*: IP address *192.168.0.101*

netcat should already be installed on your system - you can check with

```
which nc
```

To learn more about netcat, take a look at its man page:

```
man nc
```

2 Copying A File From One System To The Other

Let's say we want to copy the file *ISPConfig-2.2.27.tar.gz* from *server1* to *server2*. To do this, run

server2:

```
nc -lp 1234 > ISPConfig-2.2.27.tar.gz
```

on *server2* (*1234* is some unused port - you can replace it with another value if you like). server2 will then wait for the file *ISPConfig-2.2.27.tar.gz* on port *1234*.

On *server1*, run

server1:

```
nc -w 1 server2.example.com 1234 < ISPConfig-2.2.27.tar.gz
```

to start the file transfer.

3 Cloning Hard Drives & Partitions

You can use netcat even to clone hard drives/partitions over the network. In this example, I want to clone */dev/sda* from *server1* to *server2*. Of course, the to-be-cloned partitions must be unmounted on the target system, so if you want to clone the system partition, you must boot the target system (*server2*) from a rescue system or Live-CD such as ***Knoppix***. Please keep in mind that the target system's IP address might change under the live system (you can find out by running

```
ifconfig
```

). *server2*'s IP address in this example is *192.168.0.12* instead of *192.168.0.101*.

On *server2*, run

server2:

```
nc -l -p 1234 | dd of=/dev/sda
```

Afterwards, on server1, run

server1:

```
dd if=/dev/sda | nc 192.168.0.12 1234
```

to start the cloning process. This can take some time, depending on the size of the hard drive or partitions.

4 Port Scanning

On *server1*, you can scan for open ports on *server2* as follows:

server1:

```
nc -v -w 1 server2.example.com -z 1-1000
```

(*1-1000* means: scan ports from port number 1 to port number 1000.)

You can also scan ports on the local system:

```
nc -v -w 1 localhost -z 1-1000
```

5 Serving Web Pages

You can even use netcat to act as a web server:

```
while true; do nc -l -p 80 -q 1 < somepage.html; done
```

would serve the page *somepage.html* until you close the terminal window.

6 Spoofing HTTP Headers

You can use netcat to request web pages:

```
nc ispconfig.org 80
```

You can then type in headers as follows:

```
GET / HTTP/1.1
  Host: ispconfig.org
  Referrer: mypage.com
User-Agent: my-browser
```

As you see, this allows you to make up your own referrers and browser (*User-Agent*). After you've typed in your headers, press *ENTER* twice, and the requested page will appear (including the headers sent back by the remote server):

```
server2:~# nc exampple.com 80
  GET / HTTP/1.1
  Host: example.com
  Referrer: mypage.com
User-Agent: my-browser

HTTP/1.1 200 OK
  Date: Fri, 28 Nov 2008 14:11:49 GMT
  Server: Apache/2.2.3 (Debian) mod_ssl/2.2.3 OpenSSL/0.9.8c
```

```
  Last-Modified: Wed, 26 Nov 2008 19:34:17 GMT
  ETag: "228c707-21b1-b6b7e040"
  Accept-Ranges: bytes
  Content-Length: 8625
  Content-Type: text/html

[...]
```

7 Chatting

You can even use netcat to chat from one system to the other on the command line.

Type

server2:

```
nc -lp 1234
```

on *server2*. *server2* will then wait until *server1* connects on port *1234*.

On *server1*, run

server1:

```
nc server2.example.com 1234
```

Now you can type in messages on either system and press *ENTER*, and they will appear on the other system. To close the chat, press *CTRL+C* on either system.

8 Links

- netcat: ***http://netcat.sourceforge.net/***