nmap Usage Example

```
root@kali:~# nmap -v -A -sV 192.168.1.1


Starting Nmap 6.45 ( http://nmap.org ) at 2014-05-13 18:40 MDT

NSE: Loaded 118 scripts for scanning.

NSE: Script Pre-scanning.

Initiating ARP Ping Scan at 18:40

Scanning 192.168.1.1 [1 port]

Completed ARP Ping Scan at 18:40, 0.06s elapsed (1 total hosts)

Initiating Parallel DNS resolution of 1 host. at 18:40

Completed Parallel DNS resolution of 1 host. at 18:40, 0.00s elapsed

Initiating SYN Stealth Scan at 18:40

Scanning router.localdomain (192.168.1.1) [1000 ports]

Discovered open port 53/tcp on 192.168.1.1

Discovered open port 22/tcp on 192.168.1.1

Discovered open port 80/tcp on 192.168.1.1

Discovered open port 3001/tcp on 192.168.1.1
```

```
root@kali:~# nping --tcp -p 22 --flags syn --ttl 2 192.168.1.1


Starting Nping 0.6.45 ( http://nmap.org/nping ) at 2014-05-13 18:43 MDT

SENT (0.0673s) TCP 192.168.1.15:60125 > 192.168.1.1:22 S ttl=2 id=54240 iplen=40  seq=1720523417 win=1480

RCVD (0.0677s) TCP 192.168.1.1:22 > 192.168.1.15:60125 SA ttl=64 id=0 iplen=44  seq=3377886789 win=5840 <mss 1460>

SENT (1.0678s) TCP 192.168.1.15:60125 > 192.168.1.1:22 S ttl=2 id=54240 iplen=40  seq=1720523417 win=1480

RCVD (1.0682s) TCP 192.168.1.1:22 > 192.168.1.15:60125 SA ttl=64 id=0 iplen=44  seq=3393519366 win=5840 <mss 1460>

SENT (2.0693s) TCP 192.168.1.15:60125 > 192.168.1.1:22 S ttl=2 id=54240 iplen=40  seq=1720523417 win=1480

RCVD (2.0696s) TCP 192.168.1.1:22 > 192.168.1.15:60125 SA ttl=64 id=0 iplen=44  seq=3409166569 win=5840 <mss 1460>

SENT (3.0707s) TCP 192.168.1.15:60125 > 192.168.1.1:22 S ttl=2 id=54240 iplen=40  seq=1720523417 win=1480

RCVD (3.0710s) TCP 192.168.1.1:22 > 192.168.1.15:60125 SA ttl=64 id=0 iplen=44  seq=3424813300 win=5840 <mss 1460>

SENT (4.0721s) TCP 192.168.1.15:60125 > 192.168.1.1:22 S ttl=2 id=54240 iplen=40  seq=1720523417 win=1480

RCVD (4.0724s) TCP 192.168.1.1:22 > 192.168.1.15:60125 SA ttl=64 id=0 iplen=44  seq=3440460772 win=5840 <mss 1460>
```

```
Max rtt: 0.337ms | Min rtt: 0.282ms | Avg rtt: 0.296ms
Raw packets sent: 5 (200B) | Rcvd: 5 (230B) | Lost: 0 (0.00%)
Nping done: 1 IP address pinged in 4.13 seconds
```

ndiff Usage Example

```
root@kali:~# ndiff yesterday.xml today.xml
-Nmap 6.45 scan initiated Tue May 13 18:46:43 2014 as: nmap -v -F -oX yesterday.xml 192.168.1.1
+Nmap 6.45 scan initiated Tue May 13 18:47:58 2014 as: nmap -v -F -oX today.xml 192.168.1.1

 endian.localdomain (192.168.1.1, 00:01:6C:6F:DD:D1):
-Not shown: 96 filtered ports
+Not shown: 97 filtered ports
 PORT   STATE SERVICE VERSION
-22/tcp open  ssh
```

ncat Usage Example

```
root@kali:~# ncat -v --exec "/bin/bash" --allow 192.168.1.123 -l 4444 --keep-open
Ncat: Version 6.45 ( http://nmap.org/ncat )
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 192.168.1.123.
Ncat: Connection from 192.168.1.123:39501.
Ncat: Connection from 192.168.1.15.
Ncat: Connection from 192.168.1.15:60393.
Ncat: New connection denied: not allowed
```

Packages and Binaries:

ncat

ncat is a reimplementation of Netcat by the NMAP project, providing most of the features present in the original implementations, along with some new features such as IPv6 and SSL support. Port scanning support has been removed.

Installed size: 799 KB

How to install: sudo apt install ncat

Dependencies:

- libc6
- liblua5.4-0
- libpcap0.8t64
- libssl3t64

```
root@kali:~# ncat -h

Ncat 7.95 ( https://nmap.org/ncat )

Usage: ncat [options] [hostname] [port]


Options taking a time assume seconds. Append 'ms' for milliseconds,

's' for seconds, 'm' for minutes, or 'h' for hours (e.g. 500ms).

 -4                  Use IPv4 only

 -6                  Use IPv6 only

 -U, --unixsock          Use Unix domain sockets only

    --vsock             Use vsock sockets only

 -C, --crlf             Use CRLF for EOL sequence

 -c, --sh-exec <command>    Executes the given command via /bin/sh

 -e, --exec <command>      Executes the given command

    --lua-exec <filename>  Executes the given Lua script

 -g hop1[,hop2,...]       Loose source routing hop points (8 max)

 -G <n>                 Loose source routing hop pointer (4, 8, 12, ...)
```

```
-m, --max-conns <n>      Maximum <n> simultaneous connections

-h, --help               Display this help screen

-d, --delay <time>       Wait between read/writes

-o, --output <filename>  Dump session data to a file

-x, --hex-dump <filename> Dump session data as hex to a file

-i, --idle-timeout <time> Idle read/write timeout

-p, --source-port port   Specify source port to use

-s, --source addr        Specify source address to use (doesn't affect -l)

-l, --listen             Bind and listen for incoming connections

-k, --keep-open          Accept multiple connections in listen mode

-n, --nodns              Do not resolve hostnames via DNS

-t, --telnet             Answer Telnet negotiations

-u, --udp                Use UDP instead of default TCP

  --sctp                 Use SCTP instead of default TCP

-v, --verbose            Set verbosity level (can be used several times)

-w, --wait <time>        Connect timeout

-z                       Zero-I/O mode, report connection status only

  --append-output        Append rather than clobber specified output files

  --send-only            Only send data, ignoring received; quit on EOF

  --recv-only            Only receive data, never send anything

  --no-shutdown          Continue half-duplex when receiving EOF on stdin

  --allow                Allow only given hosts to connect to Ncat

  --allowfile            A file of hosts allowed to connect to Ncat

  --deny                 Deny given hosts from connecting to Ncat

  --denyfile             A file of hosts denied from connecting to Ncat

  --broker               Enable Ncat's connection brokering mode

  --chat                 Start a simple Ncat chat server

  --proxy <addr[:port]>  Specify address of host to proxy through

  --proxy-type <type>    Specify proxy type ("http", "socks4", "socks5")

  --proxy-auth <auth>    Authenticate with HTTP or SOCKS proxy server

  --proxy-dns <type>     Specify where to resolve proxy destination

  --ssl                  Connect or listen with SSL

  --ssl-cert             Specify SSL certificate file (PEM) for listening

  --ssl-key              Specify SSL private key (PEM) for listening
  --ssl-verify          Verify trust and domain name of certificates
  --ssl-trustfile       PEM file containing trusted SSL certificates
  --ssl-ciphers         Cipherlist containing SSL ciphers to use
  --ssl-servername      Request distinct server name (SNI)
  --ssl-alpn            ALPN protocol list to use
  --version             Display Ncat's version information and exit

See the ncat(1) manpage for full options, descriptions and usage examples
```