

DANIEL MIESSLER

GET THE PODCAST ON APPLE PODCASTS (<https://podcasts.apple.com/us/podcast/unsupervised-learning/id1099711235>). ×

MASSCAN EXAMPLES: FROM INSTALLATION TO EVERYDAY USE

By [DANIEL MIESSLER \(HTTPS://DANIELMIESSLER.COM/BLOG/AUTHOR/DANIEL/\)](https://danielmiessler.com/blog/author/daniel/).

CREATED/UPDATED: MARCH 14, 2019

Basic

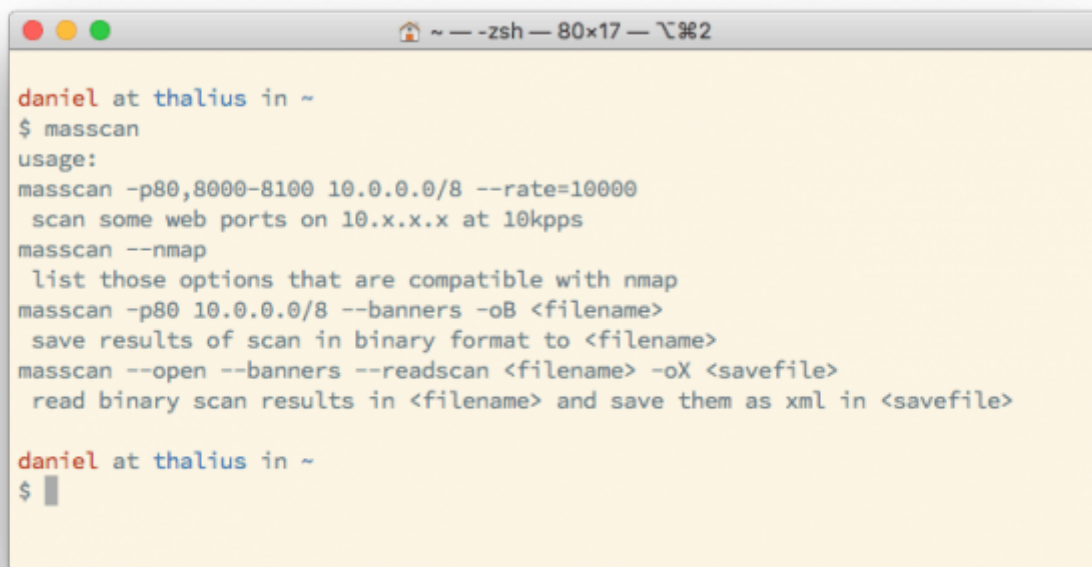
- [Background](#)
- [Installation](#)
- [Single-port Scans](#)
- [Multi-port Scans](#)
- [Scan Top Ports](#)
- [Scanning Fast](#)
- [Excluding Hosts](#)
- [Saving Your Configuration](#)

Examples

- [Output](#)
- [Nmap Functionality](#)
- [Finding Web Ports on a Network](#)
- [Finding All Ports on Network](#)
- [Finding Top 10 Ports on a Network](#)
- [Scan Internet for a Port](#)
- [Scan Internet for a All Ports](#)

BASICS

Everyone in INFORMATION SECURITY ([HTTPS://DANIELMIESSLER.COM/INFORMATION-SECURITY/](https://danielmiessler.com/information-security/)) knows NMAP ([HTTPS://NMAP.ORG](https://nmap.org)) as the rightful king of the port scanners, and it still remains the most versatile option today. But for pure speed there have some that have surpassed it, including `scanrand`, `unicornscan`, `zmap`, and now MASSCAN ([HTTPS://GITHUB.COM/ROBERTDAVIDGRAHAM/MASSCAN](https://github.com/robertdavidgraham/masscan)).

A terminal window titled '~ - zsh - 80x17 - 2' showing the output of the 'masscan' command. The prompt is 'daniel at thalius in ~'. The command '\$ masscan' is entered, and the output shows the usage of masscan with various options and their descriptions. The prompt '\$' is followed by a cursor.

```
daniel at thalius in ~
$ masscan
usage:
masscan -p80,8000-8100 10.0.0.0/8 --rate=10000
  scan some web ports on 10.x.x.x at 10kpps
masscan --nmap
  list those options that are compatible with nmap
masscan -p80 10.0.0.0/8 --banners -oB <filename>
  save results of scan in binary format to <filename>
masscan --open --banners --readscan <filename> -oX <savefile>
  read binary scan results in <filename> and save them as xml in <savefile>

daniel at thalius in ~
$
```

MASSCAN RUN WITH NO PARAMETERS

Asynchronous transmission means the scanner doesn't have to wait for replies before sending out probes.

`masscan` was created for the sole purpose of scanning the entire internet as fast as possible, according to its author ROBERT GRAHAM ([HTTPS://MOBILE.TWITTER.COM/ERRATAROB](https://mobile.twitter.com/erratarob)), this can be done *in less than 6 minutes at around 10 million packets per second*.

In this short tutorial we're going to learn the basics and provide some real-world examples.

If you just need syntax to run with you can jump ahead to the QUICKSTART.

INSTALLATION

Installing masscan is fairly straightforward whether you're using Linux or macOS.

This will install the binary under `bin/masscan`; you'll have to move it to run it from somewhere else.

Install on Debian/Ubuntu

```
$ sudo apt-get install clang git gcc make libpcap-dev
$ git clone https://github.com/robertdavidgraham/masscan
$ cd masscan
$ make
```

`brew` is the main command for Homebrew, which you can get [HERE](https://brew.sh).
([HTTPS://BREW.SH](https://brew.sh)).

Install on macOS

```
$ brew install masscan
```

SINGLE-PORT SCANS

Many people use `masscan` to scan very large networks (such as the internet) on one or just a few ports.

`masscan` has been designed to work much like `nmap`, which makes it instantly approachable for thousands of security professionals and enthusiasts.

Scan a class B subnet for port 443

```
$ masscan 10.11.0.0/16 -p443
```

MULTI-PORT SCANS

You can also scan multiple ports using a comma as a separator.

Scan a class B subnet for port 80 or 443

```
$ masscan 10.11.0.0/16 -p80,443
```

By default masscan only takes IP addresses as parameters. [THIS SCRIPT BY @JHADDIX \(HTTPS://GITHUB.COM/JHADDIX/SCRIPTS/BLOB/MASTER/MASS.SH\)](https://github.com/jhaddix/scripts/blob/master/mass.sh) will let you scan a domain (translated to IP) instead.

SCAN A RANGE OF PORTS

Or you can scan a range of ports using the dash.

Scan a class B subnet for ports 22 through 25

```
$ masscan 10.11.0.0/16 -p22-25
```

SCAN N NUMBER OF nmap'S TOP PORTS

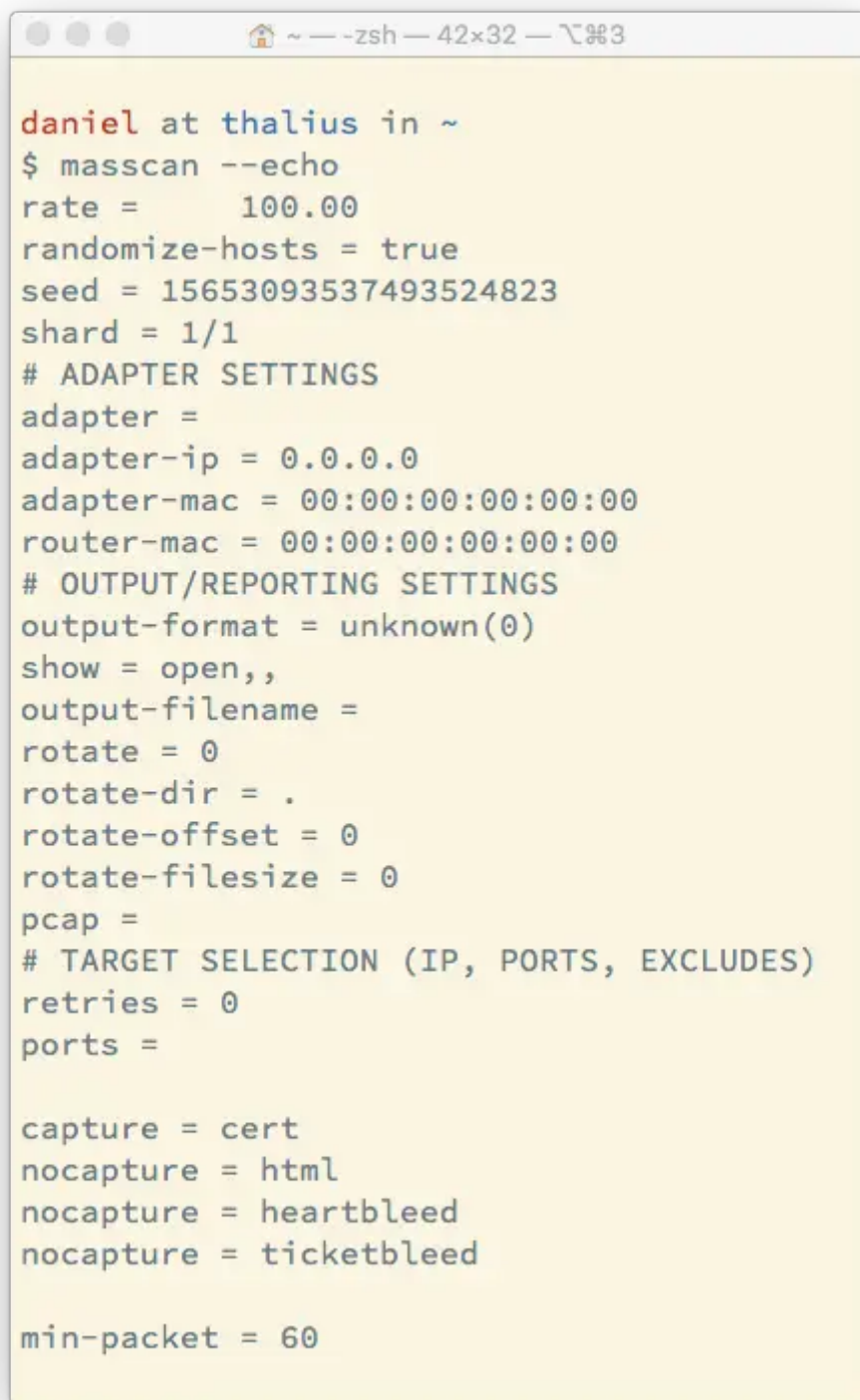
In addition you can use nmap's `--top-ports` option, which lets you specify the top n number of the most common ports to scan. So if you give it `--top-ports 100` it'll scan the top 100 most common ports discovered according to nmap.

If you don't have the `--top-ports` option available to you, make sure you have the latest version of masscan.

Scan a class B subnet for the top 100 ports

```
$ masscan 10.11.0.0/16 --top-ports 100
```

OPTIONS

A terminal window with a light yellow background. The title bar shows a home icon, a minus sign, a plus sign, and the text '~ --zsh -- 42x32 -- 100%'. The terminal content shows the output of the command 'masscan --echo'. It lists various configuration options for the masscan tool, including rate, randomize-hosts, seed, shard, adapter settings, output/reporting settings, target selection, and capture options.

```
daniel at thalius in ~  
$ masscan --echo  
rate = 100.00  
randomize-hosts = true  
seed = 15653093537493524823  
shard = 1/1  
# ADAPTER SETTINGS  
adapter =  
adapter-ip = 0.0.0.0  
adapter-mac = 00:00:00:00:00:00  
router-mac = 00:00:00:00:00:00  
# OUTPUT/REPORTING SETTINGS  
output-format = unknown(0)  
show = open,,  
output-filename =  
rotate = 0  
rotate-dir = .  
rotate-offset = 0  
rotate-filesize = 0  
pcap =  
# TARGET SELECTION (IP, PORTS, EXCLUDES)  
retries = 0  
ports =  
  
capture = cert  
nocapture = html  
nocapture = heartbleed  
nocapture = ticketbleed  
  
min-packet = 60
```

THE DEFAULT `MASSCAN` OPTIONS

You can check masscan's options with the `--echo` switch.

Now that we've covered some basics, let's look at some additional tweaks we can make.

SCANNING FAST

Using the settings above you'll definitely get results, but the speed will be quite average. As discussed already, the whole point of `masscan` is to be quick, so let's speed it up.

By default, `masscan` scans at a rate of 100 packets per second, which is quite slow. To increase that, simply supply the `--rate` option and specify a value.

Scan a class B subnet for the top 100 ports at 100,000 packets per second

```
$ masscan 10.11.0.0/16 --top-ports 100 --rate 100000
```

Scanning this fast (or even slower) is likely to cause all sorts of problems, including getting your system blocked on the internet, getting abuse complaints to your hosting provider, etc. Don't just start scanning large networks without setting groundwork first.

How fast you can scan is going to depend on a lot of factors, including your operating system (Linux scan scan far faster than Windows), the resources of your system, and—most importantly—your bandwidth. In order to scan very large networks at high speeds you'll need to use rates of a million or more (`--rate 1000000`).

EXCLUDING TARGETS

Because much of the internet can react poorly to being scanned—and also just out of sheer courtesy—you may want or need to exclude some targets from your scans. To do this, provide the `--excludefile` switch along with the name of the file that includes lists of ranges to avoid.

```
# Scan a class B subnet, but avoid the ranges in exclude.txt
```

```
$ masscan 10.11.0.0/16 --top-ports 100 --excludefile exclude.txt
```

This will produce the notification at the top of your scan that:

```
exclude.txt: excluding 1 range from file
```

SAVING YOUR CONFIGURATION

As we mentioned earlier, you can show the current `masscan` options using `--echo`, but you can also save them to a file using the standard method.

```
# Scan a class B subnet, but avoid the ranges in exclude.txt
```

```
$ masscan 10.11.0.0/16 --top-ports 100 --echo > scan.txt
```

OUTPUT

First, you can just use the standard Unix redirector to send output to a file:

```
$ masscan 10.11.0.0/16 --top-ports 100 > results.txt
```

But in addition to that you also have the following output options:

- `-oX filename`: Output to `filename` in **XML**.
- `-oG filename`: Output to `filename` in **Grepable** format.
- `-oJ filename`: Output to `filename` in **JSON** format.

NMAP FUNCTIONALITY

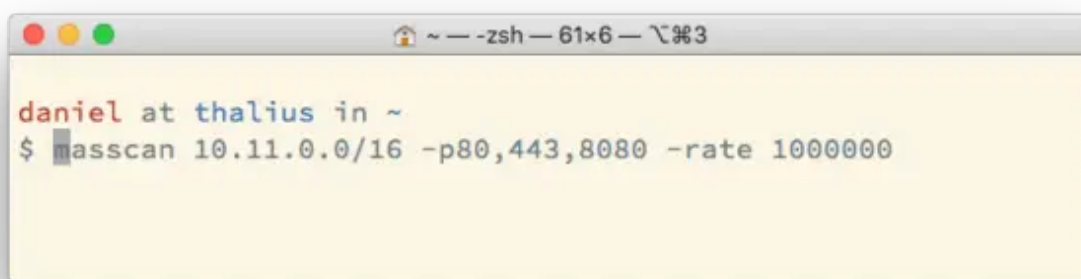
As mentioned initially, masscan is built to work much like `nmap`, which makes it familiar to many security people. Here are some of the other nmap-like options that are available:

You can see the nmap-like functionality by passing the `--nmap` switch.

- `-iL filename`: Read inputs from a file.
- `--exclude filename`: Exclude a network on the command line.
- `--excludefile`: Exclude networks from a file.
- `-S`: Spoof source IP.
- `-v interface`: Verbose output.

- `-vv interface`: *Very* verbose output.
- `-e interface`: Use specified interface.
- `-e interface`: Use specified interface.

QUICKSTART

A terminal window with a yellow background and a grey title bar. The title bar contains a home icon, a minus sign, a plus sign, and the text '~ -zsh- 61x6 - 100%'. The terminal content shows the prompt 'daniel at thalius in ~' followed by the command '\$ masscan 10.11.0.0/16 -p80,443,8080 -rate 1000000'.

```
daniel at thalius in ~  
$ masscan 10.11.0.0/16 -p80,443,8080 -rate 1000000
```

Ok, here are some quick and functional scan examples that you can start with and then tweak to your taste and requirements.

We're assuming here that you want to scan quickly.

SCAN A NETWORK FOR WEB PORTS

```
$ masscan 10.11.0.0/16 -p80,443,8080 -rate 1000000
```

SCAN A NETWORK FOR THE TOP 10 PORTS

```
$ masscan 10.11.0.0/16 --top-ten --rate 1000000
```

SCAN A NETWORK FOR ALL PORTS

```
$ masscan 10.11.0.0/16 -p0-65535 --rate 1000000
```

SCAN THE INTERNET FOR A PORT

We've increased the speed to 10 million per second, which will max you out.

```
$ masscan 0.0.0.0/0 -p443 --rate 10000000
```

SCAN THE INTERNET FOR ALL PORTS

In general you should expect bad and/or amazing things to happen if you try this.

```
$ masscan 0.0.0.0/0 -p0-65535 --rate 10000000
```

Summary

There are other options available that you can get from following the

[README.MD \(HTTPS://GITHUB.COM/ROBERTDAVIDGRAHAM/MASSCAN/BLOB/MASTER/README.MD\)](https://github.com/RobertDavidGraham/masscan/blob/master/README.md) for the source code repository, but this primer should get you up and running nicely.

Happy (speed) scanning!

NOTES

1. There are number of defaults that are enabled with masscan that need to be defined with nmap simply because the scanners work in different ways. For example, masscan always treats all hosts as online, scans are always randomized, it's a SYN-based scan, it never does DNS resolution, and scans are performed using raw `libpcap`.
2. One thing that's fairly unique to masscan is that you can easily pause and resume scans. When you press `ctrl-c` a file is created called `paused.conf` that has all the settings and progress from the scan. You can resume that scan with `--resume paused.conf`.
3. The project [README \(HTTPS://GITHUB.COM/ROBERTDAVIDGRAHAM/MASSCAN/BLOB/MASTER/README.MD\)](https://github.com/RobertDavidGraham/masscan/blob/master/README.md).

DANIEL MIESSLER

[HTTPS://DANIELMIESSLER.COM](https://danielmiessler.com)

© Daniel Miessler 1999-2020

Recommended

[Most Popular \(https://danielmiessler.com/popular/\)](https://danielmiessler.com/popular/)
[Blog \(https://danielmiessler.com/blog/\)](https://danielmiessler.com/blog/)
[Tutorials \(https://danielmiessler.com/study/\)](https://danielmiessler.com/study/)
[Information Security \(https://danielmiessler.com/information-security/\)](https://danielmiessler.com/information-security/)
[Technology \(https://danielmiessler.com/technology/\)](https://danielmiessler.com/technology/)

Tutorials

[Recommended Tutorials \(https://danielmiessler.com/study/\)](https://danielmiessler.com/study/)
[A Vim Primer \(https://danielmiessler.com/study/vim/\)](https://danielmiessler.com/study/vim/)
[A Tcpdump Primer \(https://danielmiessler.com/study/tcpdump/\)](https://danielmiessler.com/study/tcpdump/)
[Security Assessment Types \(https://danielmiessler.com/study/security-assessment-types/\)](https://danielmiessler.com/study/security-assessment-types/)
[URLs vs. URIs \(https://danielmiessler.com/study/difference-between-uri-url/\)](https://danielmiessler.com/study/difference-between-uri-url/)

Projects

[Unsupervised Learning \(https://danielmiessler.com/podcast/\)](https://danielmiessler.com/podcast/)
[Reading \(https://danielmiessler.com/reading/\)](https://danielmiessler.com/reading/)
[Concepts \(https://danielmiessler.com/projects/concepts/\)](https://danielmiessler.com/projects/concepts/)
[Ideas \(https://danielmiessler.com/projects/ideas/\)](https://danielmiessler.com/projects/ideas/)
[Book Summaries \(https://danielmiessler.com/projects/reading/book-summaries/\)](https://danielmiessler.com/projects/reading/book-summaries/)