[Skip to main content](https://lms.alnafi.com/xblock/block-v1:alnafi+DCCS102+2025_DCCS+type@vertical+block@e70a59a186844a1c884a2a55856b630e?exam_access=&recheck_access=1&show_bookmark=0&show_title=0&view=student_view#main)

**Map the Network2**

Everything on the Nmap command line that isn't an option (or option argument) is treated as a target host specification. The simplest case is to specify a target IP address or hostname for scanning.

When a hostname is given as a target, it is resolved via the Domain Name System (DNS) to determine the IP address to scan. If the name resolves to more than one IP address, only the first one will be scanned. To make Nmap scan all the resolved addresses instead of only the first one, use the --resolve-all option.

Sometimes you wish to scan a whole network of adjacent hosts. For this, Nmap supports CIDR-style addressing. You can append /<numbits> to an IP address or hostname and Nmap will scan every IP address for which the first <numbits> are the same as for the reference IP or hostname given. For example, 192.168.10.0/24 would scan the 256 hosts between 192.168.10.0 (binary: 11000000 10101000 00001010 00000000) and 192.168.10.255 (binary: 11000000 10101000 00001010 11111111), inclusive. 192.168.10.40/24 would scan exactly the same targets. Given that the host scanme.nmap.org is at the IP address 64.13.134.52, the specification scanme.nmap.org/16 would scan the 65,536 IP addresses between 64.13.0.0 and 64.13.255.255. The smallest allowed value is /0, which targets the whole Internet. The largest value for IPv4 is /32, which scans just the named host or IP address because all address bits are fixed. The largest value for IPv6 is /128, which does the same thing.

CIDR notation is short but not always flexible enough. For example, you might want to scan 192.168.0.0/16 but skip any IPs ending with .0 or .255 because they may be used as subnet network and broadcast addresses. Nmap supports this through octet range addressing. Rather than specify a normal IP address, you can specify a comma-separated list of numbers or ranges for each octet. For example, 192.168.0-255.1-254 will skip all addresses in the range that end in .0 or .255, and 192.168.3-5,7.1 will scan the four addresses 192.168.3.1, 192.168.4.1, 192.168.5.1, and 192.168.7.1. Either side of a range may be omitted; the default values are 0 on the left and 255 on the right. Using - by itself is the same as 0-255, but remember to use 0- in the first octet so the target specification doesn't look like a command-line option. Ranges need not be limited to the final octets: the specifier 0-255.0-255.13.37 will perform an Internet-wide scan for all IP addresses ending in 13.37. This sort of broad sampling can be useful for Internet surveys and research.

IPv6 addresses can be specified by their fully qualified IPv6 address or hostname or with CIDR notation for subnets. Octet ranges aren't yet supported for IPv6.

IPv6 addresses with non-global scope need to have a zone ID suffix. On Unix systems, this is a percent sign followed by an interface name; a complete address might be fe80::a8bb:ccff:fedd:eeff%eth0. On Windows, use an interface index number in place of an interface name: fe80::a8bb:ccff:fedd:eeff%1. You can see a list of interface indexes by running the command netsh.exe interface ipv6 show interface.

Nmap accepts multiple host specifications on the command line, and they don't need to be the same type. The command nmap scanme.nmap.org 192.168.0.0/8 10.0.0,1,3-7.- does what you would expect.

While targets are usually specified on the command lines, the following options are also available to control target selection:

-iL <inputfilename> (Input from list)

Reads target specifications from <inputfilename>. Passing a huge list of hosts is often awkward on the command line, yet it is a common desire. For example, your DHCP server might export a list of 10,000 current leases that you wish to scan. Or maybe you want to scan all IP addresses except for those to locate hosts using unauthorized static IP addresses. Simply generate the list of hosts to scan and pass that filename to Nmap as an argument to the -iL option. Entries can be in any of the formats accepted by Nmap on the command line (IP address, hostname, CIDR, IPv6, or octet ranges). Each entry must be separated by one or more spaces, tabs, or newlines. You can specify a hyphen (-) as the filename if you want Nmap to read hosts from standard input rather than an actual file.

The input file may contain comments that start with # and extend to the end of the line.

-iR <num hosts> (Choose random targets)

For Internet-wide surveys and other research, you may want to choose targets at random. The <num hosts> argument tells Nmap how many IPs to generate. Undesirable IPs such as those in certain private, multicast, or unallocated address ranges are automatically skipped. The argument 0 can be specified for a never-ending scan. Keep in mind that some network administrators bristle at unauthorized scans of their networks and may complain. Use this option at your own risk! If you find yourself really bored one rainy afternoon, try the command nmap -Pn -sS -p 80 -iR 0 --open to locate random web servers for browsing.

--exclude <host1>[,<host2>[,...]] (Exclude hosts/networks)

Specifies a comma-separated list of targets to be excluded from the scan even if they are part of the overall network range you specify. The list you pass in uses normal Nmap syntax, so it can include hostnames, CIDR netblocks, octet ranges, etc. This can be useful when the network you wish to scan includes untouchable mission-critical servers, systems that are known to react adversely to port scans, or subnets administered by other people.

--excludefile <exclude\_file> (Exclude list from file)

This offers the same functionality as the --exclude option, except that the excluded targets are provided in a newline-, space-, or tab-delimited <exclude\_file> rather than on the command line.

The exclude file may contain comments that start with # and extend to the end of the line.

-n (No DNS resolution)

Tells Nmap to never do reverse DNS resolution on the active IP addresses it finds. Since DNS can be slow even with Nmap's built-in parallel stub resolver, this option can slash scanning times.

-R (DNS resolution for all targets)

Tells Nmap to always do reverse DNS resolution on the target IP addresses. Normally reverse DNS is only performed against responsive (online) hosts.

--resolve-all (Scan each resolved address)

If a hostname target resolves to more than one address, scan all of them. The default behavior is to only scan the first resolved address. Regardless, only addresses in the appropriate address family will be scanned: IPv4 by default, IPv6 with -6.

--unique (Scan each address only once)

Scan each IP address only once. The default behavior is to scan each address as many times as it is specified in the target list, such as when network ranges overlap or different hostnames resolve to the same address.

--system-dns (Use system DNS resolver)

By default, Nmap reverse-resolves IP addresses by sending queries directly to the name servers configured on your host and then listening for responses. Many requests (often dozens) are performed in parallel to improve performance. Specify this option to use your system resolver instead (one IP at a time via the getnameinfo call). This is slower and rarely useful unless you find a bug in the Nmap parallel resolver (please let us know if you do). The system resolver is always used for forward lookups (getting an IP address from a hostname).

--dns-servers <server1>[,<server2>[,...]] (Servers to use for reverse DNS queries)

By default, Nmap determines your DNS servers (for rDNS resolution) from your resolv.conf file (Unix) or the Registry (Win32). Alternatively, you may use this option to specify alternate servers. This option is not honored if you are using --system-dns. Using multiple DNS servers is often faster, especially if you choose authoritative servers for your target IP space. This option can also improve stealth, as your requests can be bounced off just about any recursive DNS server on the Internet.

This option also comes in handy when scanning private networks. Sometimes only a few name servers provide proper rDNS information, and you may not even know where they are. You can scan the network for port 53 (perhaps with version detection), then try Nmap list scans (-sL) specifying each name server one at a time with --dns-servers until you find one which works.

This option might not be honored if the DNS response exceeds the size of a UDP packet. In such a situation our DNS resolver will make the best effort to extract a response from the truncated packet, and if not successful it will fall back to using the system resolver. Also, responses that contain CNAME aliases will fall back to the system resolver.