# InfraGard Technology Tool Kit Disclaimer Notice:

# Created by Deborah Watson, Technology Tool Kit Program Development Manager, InfraGard - Houston Chapter; adapted from content created by Dave Armstrong, dave@bindshell.net

# IN NO EVENT SHALL INFRAGARD BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, ARISING OUT OF THE USE OF THIS SOFTWARE AND/OR DOCUMENTATION, EVEN IF INFRAGARD HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

INFRAGARD SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE AND/OR ACCOMPANYING DOCUMENTATION, IF ANY, PROVIDED HEREUNDER IS PROVIDED "AS IS". INFRAGARD HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS

# GenIP

A small utility, based on the NMap target specification code, for quickly and easily generating lists of IP addresses.

Developer: [Dave Armstrong](mailto:dave@bindshell.net)

## Usage

genip [ -h ] [ -i filename ] [ <target-spec> ... ]  
genip -r <ipaddress> <ipaddress>

**Options**

-h

Display usage information.

-i

Read target specifications from the give filename. If a filename of "-" used, target specifications are read from standard in. Target specifications read from input files are processed in NMap mode regardless of the presence of the -r option.

-r

Specify range mode (see below).

**Modes**

GenIP has two modes of operation that are detailed below:

*NMap Mode (Default)*

In this mode genip will expand all target specifications listed on the command line. Since genip is essentially just the NMap target parsing code it functions in exactly the same way. Here is what the NMap documentation has to say about target specification:

Everything that isn't an option (or option argument) is treated as a target host specification. The simplest case is listing single hostnames or IP addresses on the command line. If you want to scan a subnet of IP addresses, you can append /mask to the hostname or IP address. mask must be between 0 (scan the whole Internet) and 32 (scan the single host specified). Use /24 to scan a class "C" address and /16 for a class "B". There is also a more powerful notation which lets you specify an IP address using lists/ranges for each element. Thus you can scan the whole class "B" network 192.168.\*.\* by specifying "192.168.\*.\*" or "192.168.0-255.0-255" or even "192.168.1-50,51-255.1,2,3,4,5-255". And of course you can use the mask notation: "192.168.0.0/16". These are all equivalent. If you use asterisks ("\*"), remember that most shells require you to escape them with back slashes or protect them with quotes.

*Range Mode*

In this mode two (and only two!) IP addresses must be specified, in dotted quad notation, and the output is all the addresses between the two (inclusive). This mode can be used to cross class boundaries.

**Example**

In its most basic form genip simply echos the IP addresses listed on the command line:

> genip 10.1.1.1 10.3.4.5

10.1.1.1

10.3.4.5

By use one of the many expansion methods detailed above a large number of IP addresses can be generated from simple command line specifications:

> genip 10.1.1.1-3

10.1.2.0

10.1.2.1

10.1.2.2

By selecting range mode (with the use of the -r flag) it is a simple matter to generate address lists that cross class boundaries:

> genip -r 10.1.1.254 10.1.2.2

10.1.1.254

10.1.1.255

10.1.2.0

10.1.2.1

10.1.2.2

**Download**

A safe version of GenIP can be obtained from the InfraGard site at: [insert link][genip.tgz](http://bindshell.net/tools/genip/genip.tgz) (md5sum: 9e5c82204487c46cbab32057d711177e)

**License**

Target specification code (and related documentation) was extracted from nmap (by Fyodor <fyodor@insecure.org> ). GenIP is therefore a "derived work" and release under the GNU General Public License (a copy of which should be contained within the distribution archive).