frag6 v1.0 manual pages

Description

frag6 is a security assessment tool for attack vectors based on IPv6 fragments. This tool is part of the IPv6 Toolkit v1.2: a security assessment suite for the IPv6 protocol developed by the UK CPNI.

Options

The frag6 tool takes it parameters as command-line options. Each of the options can be specified with a short name (one character preceded with the hyphen character, as e.g. “-i”) or with a long name (a string preceded with two hyphen characters, as e.g. “--interface”).

`--interface`, `-i`

This option specifies the network interface that the tool will use. The network interface must be specified (i.e., the tool does not select any network interface “by default”).

`--src-link-address`, `-S`

This option specifies the link-layer Source Address of the probe packets (currently, only Ethernet is supported). If left unspecified, the link-layer Source Address of the packets is set to the real link-layer address of the network interface.

`--link-dst-address`, `-D`

This option specifies the link-layer Destination Address of the probe packets (currently, only Ethernet is supported). By default, the link-layer Destination Address is automatically set to the link-layer address of the destination host (for on-link destinations) or to the link-layer address of the first-hop router.

`--src-address`, `-s`

This option specifies the IPv6 source address (or IPv6 prefix) to be used for the Source Address of the probe packets. If an IPv6 prefix is specified, the IPv6 Source Address of the ICMPv6 packets will be randomized from that prefix.

`--dst-address`, `-d`

This option specifies the IPv6 Destination Address of the target node. This option cannot be left unspecified.

`--hop-limit, -A`

This option specifies the Hop Limit to be used for the IPv6 packets. By default, the Hop Limit is randomized.

`--dst-opt-hdr`, `-u`

This option specifies that a Destination Options header is to be included in the outgoing packet(s). The extension header size must be specified as an argument to this option (the header is filled with padding options). Multiple Destination Options headers may be specified by means of multiple “-u” options.

`--dst-opt-u-hdr`, `-U`

This option specifies a Destination Options header to be included in the “unfragmentable part” of the outgoing packet(s). The header size must be specified as an argument to this option (the header is filled with padding options). Multiple Destination Options headers may be specified by means of multiple “-U” options.

`--hbh-opt-hdr`, `-H`

This option specifies that a Hop-by-Hop Options header is to be included in the outgoing packet(s). The header size must be specified as an argument to this option (the header is filled with padding options). Multiple Hop-by-Hop Options headers may be specified by means of multiple “-H” options.

`--frag-size`, `-P`

This option specifies the IPv6 fragment payload size.

`--frag-type`, `-O`

This option specifies the fragment “type”. Possible types are “first”, “middle”, “last”, and “atomic”. If the selected fragment type is “first”, the Fragment Offset is automatically set to 0, and the “M” (“More fragments”) bit is set to 1. If the selected fragment type is “middle”, the Fragment Offset is set to a non-zero value, and the “M” bit is set to 1. If the selected fragment type is “last”, the

Fragment Offset is set to a non-zero value, and the “M” bit is set to 0. Finally, if the selected fragment type is “atomic”, the Fragment Offset is set to 0, and the “M” bit is set to 0.

`--frag-offset`, `-o`

This option specifies the Fragment Offset. The Fragment Offset specified by means of this option overrides the value implicitly specified by means of the “-O” option.

`--frag-id`, `-I`

This option specifies the fragment “Identification” value. If left unspecified, the “Identification” value is randomized.

`--no-timestamp`, `-T`

When assessing the fragment reassembly policy of a target, the fragment payload includes a timestamp value that is used to measure the fragment reassembly timeout. If this option is set, such timestamp will not be included in the payload (and the tool will not be able to measure the fragment reassembly timeout).

`--no-responses`, `-n`

This option instructs the frag6 tool not to display the responses to the fragments sent. This option is useful when performing a fragmentation-flooding attack, as multiple response packets (ICMPv6 errors) might be received.

`--frag-reass-policy`, `-p`

This option instructs the tool to determine the IPv6 fragment reassembly policy of the target. In order to determine the aforementioned policy, the tool performs a number of tests to determine how the target node processes overlapping fragments. The following figures illustrate the sequence of packets that correspond to each of the tests.

Test #1

Frag. #1: `AAAAAAAAAAA`
Frag. #2: `BBBBBBBBBBB`

Test #2

```
Frag. #1: AAAAAAAAAA
Frag. #2:                   BBBBBBBBBBB
Frag. #3:         CCCCCCCCCCC
```

Test #3

```
Frag. #1: AAAAAAAAAA
Frag. #2:                   BBBBBBBBBBB
Frag. #3:             CCCCCCCCCCC
```

Test #4

```
Frag. #1: AAAAAAAAAA
Frag. #2:                   BBBBBBBBBBB
Frag. #3:             CCCCCCCCCCCCCCCCCCCCCCCCCC
```

Test #5

```
Frag. #1: AAAAAAAAAA
Frag. #2:                   BBBBBBBBBBB
Frag. #3:                             CCCCCCCCCCC
Frag. #4:             DDDDDDDD
```

For each of the aforementioned tests, the tool reports which copy of the data is used by the target host. If there is no response from the host, the tool informs whether the host silently dropped the fragments, or sent an ICMPv6 Time Exceeded error message.

`--frag-id-policy`, `-W`

This option instructs the tool to determine the fragment “Identification” generation policy. The tool sends a number of probe packets to the target node, and samples the “Identification” values of the corresponding response packets. Based on the sampled values, it tries to infer the fragment Identification generation policy of the target.

The tool will first send a number of fragments from single IPv6 address, such that the per-destination policy is determined. The tool will then send a number of fragments from random IPv6 addresses (from the same prefix as the first fragments) such that the “global” fragment Identification generation policy can be inferred.

The tool computes the expected value and the standard deviation of the difference between consecutive-sampled Identification values ($ID_n - ID_{n-1}$), with the intent of inferring the fragment Identification algorithm at the target node.

For small values of the standard deviation, the fragment Identification is assumed to be a monotonically-increasing function with increments of the “expected value”. For large values of the standard deviation, the fragment Identification is assumed to be randomized, and the expected value and standard deviation are informed to the user, as indicators of the “quality” of the fragment Identification generation algorithm.

`--pod-attack`, `-X`

This option instructs the tool to perform a “Ping of Death” attack against the specified target.

`--flood-frags`, `-F`

This option instructs the tool to send the specified number of fragments back-to-back to the target node. This option is likely to be used in conjunction with the “-l” option, such that the process is repeated in a loop.

`--loop, -l`

This option instructs the frag6 tool to periodically send IPv6 fragments to the target node. The amount of time to pause between sending a batch of fragments can be specified by means of the “-z” option, and defaults to 1 second.

`--sleep`, `-z`

This option specifies the amount of time that the tool should pause between sending btaches of IPv6 fragments (when the “--loop” option is set). If left unspecified, it defaults to 1 second.

`--verbose`, `-v`

This option instructs the frag6 tool to be verbose.

`--help`, `-h`

Print help information for the frag6 tool.

Examples

Example #1

```
# frag6 -i eth0 --frag-id-policy -d fc00:1::1 -v
```

Assess the fragment Identification generation policy of the host “fc00:1::1”, using the network interface “eth0”. Be verbose.

Example #2

```
# frag6 -i eth0 --frag-reass-policy -d fc00:1::1 -v
```

Assess the fragment reassembly policy of the host “fc00:1::1”, using the network interface “eth0”. Be verbose.

Example #3

```
# frag6 -i eth0 –frag-type atomic -d fc00:1::1 -v
```

Send an IPv6 atomic fragment to the host “fc00:1::1”, using the network interface “eth0”. Be verbose.

Example #4

```
# frag6 -i eth0 -s ::/0 --flood-frags 100 -l -z 5 -d fc00:1::1 -v
```

Send 100 fragments (every 5 seconds) to the host fc00:1::1, using a forged IPv6 Source Address from the prefix ::/0. The aforementioned fragments should have an offset of 0, and the M bit set (i.e., be first-fragments). Be verbose.

Credits

The frag6 tool version 1.0 and related manuals were produced by Fernando Gont <fgont@si6networks.com> on behalf of the UK Centre for the Protection of National Infrastructure (CPNI) <http://www.cpni.gov.uk>.

License