tcp6 v1.3 manual pages

Description

This tool allows the assessment of IPv6 implementations with respect to a variety of attack vectors based on TCP/IPv6 segments. This tool is part of the IPv6 Toolkit v1.3: a security assessment suite for the IPv6 protocols.

Modes of Operation

This tool has two modes of operation: active and listening. In active mode, the tool attacks a specific target, while in listening mode the tool listens to TCP traffic on the local network, and launches an attack in response to such traffic. Active mode is employed if an IPv6 Destination Address is specified. Listening mode is employed if the "-L" option (or its long counterpart "--listen") is set. If both an attack target and the "-L" option are specified, the attack is launched against the specified target, and then the tool enters listening mode to respond incoming packets with TCP segments.

The tool supports filtering of incoming packets based on the Ethernet Source Address, the Ethernet Destination Address, the IPv6 Source Address, and the IPv6 Destination Address. There are two types of filters: "block filters" and "accept filters". If any "block filter" is specified, and the incoming packet matches any of those filters, the message is discarded (and thus no TCP segments are sent in response). If any "accept filter" is specified, incoming packets must match the specified filters in order for the tool to respond with TCP segments.

Options

The tcp6 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").

If the tool is instructed to e.g. flood the victim with TCP segments from different sources ("--flood-sources" option), multiple packets may need to be generated.

tcp6 supports IPv6 Extension Headers, including the IPv6 Fragmentation Header, which might be of use to circumvent layer-2 filtering and/or Network Intrusion Detection Systems (NIDS). However, IPv6 extension headers are not employed by default, and must be explicitly enabled with the corresponding options.

```
--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-address,-s`

This option specifies the IPv6 source address (or IPv6 prefix) to be used for the Source Address of the attack packets. If the "-F" ("--flood-sources") option is specified, this option includes an IPv6 prefix, from which random addresses are selected. See the description of the "-F" option for further information on how the "-s" option is processed in that specific case.

Note: When operating in "listening" mode, the Source Address is automatically set to the Destination Address of the incoming packet.

`--dst-address,-d`

This option specifies the IPv6 Destination Address of the victim. It can be left unspecified only if the "-L" option is selected (i.e., if the tool is to operate in "listening" mode).

Note: When operating in "listening" mode, the Destination Address is automatically set to the Source Address of the incoming packet.

`--hop-limit, -A`

This option specifies the Hop Limit to be used for the IPv6 packets. It defaults to 255.

`--frag-hdr, -y`

This option specifies that the resulting packet must be fragmented. The fragment size <u>must</u> be specified as an argument to this option.

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

This option specifies that a Destination Options header is to be included in the resulting packet. The extension header size <u>must</u> 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 resulting packet. The header size <u>must</u> 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. This option is only valid if the "-y" option is specified (as the concept of "unfragmentable part" only makes sense when fragmentation is employed).

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

This option specifies that a Hop-by-Hop Options header is to be included in the resulting packet. The header size <u>must</u> 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.

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

This option specifies the link-layer Source Address of the TCP segments (currently, only Ethernet is supported). If left unspecified, the link-layer Source Address is randomized.

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

This option specifies the link-layer Destination Address of the TCP segments (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 of the first-hop router.

`--payload-size,-P`

This options specifies the size of the TCP payload. It defaults to 0 (i.e., empty TCP segments).

`--src-port,-o`

This option specifies the Source Port of the TCP segment.

`--dst-port,-a`

This option specifies the Destination Port of the TCP segment.

`--tcp-flags,-X`

This option is used to set specific the TCP flags. The flags are specified as "F" (FIN), "S" (SYN), "R" (RST), "P" (PSH), "A" (ACK), "U" (URG), "X" (no flags).

If this option is not set, and the tool operates in listening mode, the flags of the generated TCP segments are automatically set as follows: TCP segments elicited by SYNs have both the SYN and ACK flags set. All other TCP segments have the ACK bit set.

`--tcp-seq, -q`

This option specifies the Sequence Number of the TCP header. If left unspecified, the Sequence Number is randomized.

If this option is left unspecified and the tool is operating in listening mode, the TCP Sequence Number is set to the Acknowledgement Number of the packet that elicited the TCP segment.

`--tcp-ack, -Q`

This option specifies the Acknowledgment Number of the TCP segment. If left unspecified, the Acknowledgment Number is randomized.

If this option is left unspecified and the tool is operating in listening mode, the TCP Sequence Number is set to the Acknowledgement Number of the packet that elicited the TCP segment.

`--tcp-urg, -V`

This option specifies the Urgent Pointer of the TCP segment. If left unspecified, the Urgent Pointer is set to 0.

`--tcp-win, -w`

This option specifies the value of the TCP Window. If left unspecified, the Window is randomized.

`--not-ack-data, -N`

This option instructs tcp6 not to acknowledge the TCP payload of incoming segments (when operating in listening mode).

Note: By default, tcp6 will acknowledge both the payload and the flags of the incoming TCP segments.

`--not-ack-flags, -f`

This option instructs tcp6 not to acknowledge the TCP flags (SYN and/or FIN) of incoming segments (when operating in listening mode).

Note: By default, tcp6 will acknowledge both the payload and the flags of the incoming TCP segments.

`--block-src, -j`

This option sets a block filter for the incoming packets, based on their IPv6 Source Address. It allows the specification of an IPv6 prefix in the form "-j prefix/prefixlen". If the prefix length is not specified, a prefix length of "/128" is selected (i.e., the option assumes that a single IPv6 address, rather than an IPv6 prefix, has been specified).

`--block-dst, -k`

This option sets a block filter for the incoming packets, based on their IPv6 Destination Address. It allows the specification of an IPv6 prefix in the form "-k prefix/prefixlen". If the prefix length is not specified, a prefix length of "/128" is selected (i.e., the option assumes that a single IPv6 address, rather than an IPv6 prefix, has been specified).

`--block-link-src, -J`

This option sets a block filter for the incoming packets, based on their link-layer Source Address. The option must be followed by a link-layer address (currently, only Ethernet is supported).

`--block-link-dst, -K`

This option sets a block filter for the incoming packets, based on their link-layer Destination Address. The option must be followed by a link-layer address (currently, only Ethernet is supported).

`--accept-src, -b`

This option sets an accept filter for the incoming packets, based on their IPv6 Source Address. It allows the specification of an IPv6 prefix in the form "-b prefix/prefixlen". If the prefix length is not specified, a prefix length of "/128" is selected (i.e., the option assumes that a single IPv6 address, rather than an IPv6 prefix, has been specified).

`--accept-dst, -g`

This option sets a accept filter for the incoming packets, based on their IPv6 Destination Address. It allows the specification of an IPv6 prefix in the form "-g prefix/prefixlen". If the prefix length is

not specified, a prefix length of "/128" is selected (i.e., the option assumes that a single IPv6 address, rather than an IPv6 prefix, has been specified).

`--accept-link-src, -B`

This option sets an accept filter for the incoming packets, based on their link-layer Source Address. The option must be followed by a link-layer address (currently, only Ethernet is supported).

`--accept-link-dst, -G`

This option sets an accept filter for the incoming packets, based on their link-layer Destination Address. The option must be followed by a link-layer address (currently, only Ethernet is supported).

`--flood-sources, -F`

This option instructs the tool to send multiple TCP segments with different Source Addresses. The number of different source addresses is specified as "-F number". The Source Address of each TCP segment is randomly selected from the prefix specified by the "-s" option. If the "-F" option is specified but the "-s" option is left unspecified, the Source Address of the packets is randomly selected from the prefix ::/0.

`--flood-ports, -T`

This option instructs the tool to send multiple TCP segments with different Source Ports. The number of different sources is specified as "-T number". The Source Port of each TCP segment is randomly selected from the whole port number space (0-65535).

`--loop, -l`

This option instructs the tcp6 tool to send periodic TCP segments to the victim node. The amount of time to pause between sending TCP segments can be specified by means of the "-z" option, and defaults to 1 second. Note that this option cannot be set in conjunction with the "-L" ("--listen") option.

`--sleep, -z`

This option specifies the amount of time to pause between sending TCP segments (when the "--loop" option is set). If left unspecified, it defaults to 1 second.

```
--listen, -L
```

This instructs the tcp6 tool to operate in listening mode (possibly after attacking a given node). Note that this option cannot be used in conjunction with the "-l" ("--loop") option.

```
--verbose, -v
```

This option instructs the tcp6 tool to be verbose. When the option is set twice, the tool is "very verbose", and the tool also informs which packets have been accepted or discarded as a result of applying the specified filters.

```
--help, -h
```

Print help information for the tcp6 tool.

Examples

Example #1

```
# tcp6 -i eth0 -s fc00:1::/64 -d fc00:1::1 -a 22 -X S -F 100 -l -z 1 -v
```

In this example the tcp6 tool is essentially employed to perform a SYN-flood attack against port number 22 of the host fc00:1::1. The tool uses the network interface "eth0" (as specified by the "-i" option), and sends SYN segments (as specified by the "-X" option) from the prefix fc00:1::/64 (as specified by the "-s" option) to port 22 (specified by the "-a" option) at the destination address fc00:1::1 (specified by the "-d" option). The tool sends TCP segments from 100 different addresses (as specified by the "-F" option) every one second (as specified by the "-l" and "-z" options). The tool will be verbose (as specified by the "-v" option).

Example #2

```
# tcp6 -i eth0 -L -X RA -v
```

In this example, the tcp6 tool is employed to perform a TCP connection-reset attack against all active TCP connections in the local network. The tool listens ("-L") on the interface eth0 ("-i eth0"), and responds to any TCP segments with a RST packet (with both the RST and ACK bits set). The tool will be verbose.

Credits

The tcp6 tool and related manuals were produced by Fernando Gont <fgont@si6networks.com> for SI6 Networks <http://www.si6networks.com>.

License

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with the Invariant Sections being just "Credits", with no Front-Cover Texts, and with no Back-Cover Texts. A copy of the license is available at <http://www.gnu.org/licenses/fdl.html>.