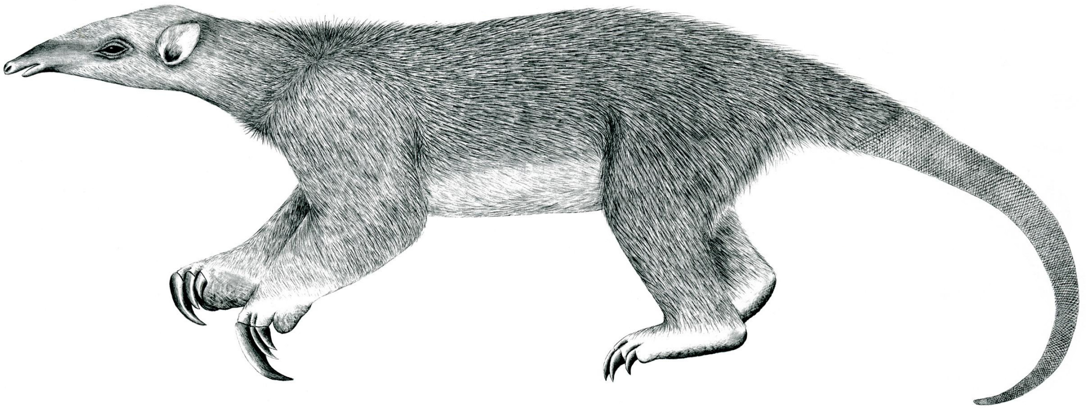
Tranalyzer2

covertChannels

Detects covert channels in IP traffic



Tranalyzer Development Team

Contents

1 covertChannels

1.1 Description

The covertChannels plugin detects various types of covert channels (CCs) in IP traffic. Currently, it detects most publicly available covert channel tools. In the future, the goal is to also detect more discreet covert channels and custom implementations based on current research (covert timing channels, SkyDe, . . .). This plugin produces only output to the flow file. Configuration is achieved by user defined compiler switches in `src/covertChannels.h`.

1.2 Required Files

1.2.1 cc_dns_whitelist.txt

The file `cc_dns_whitelist.txt` contains a domain names whitelist for the DNS CCs detection. Domains in this file will never be flagged as a covert channel.

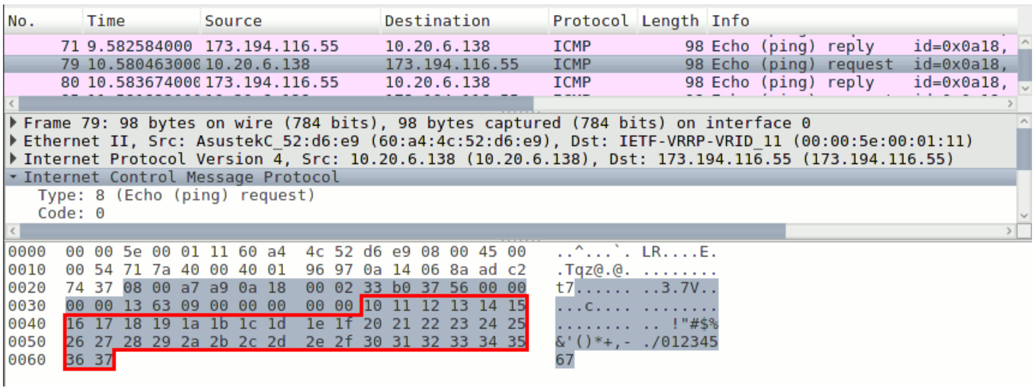
- One domain name per line.
- Lines starting with `%` are comments.
- Suffix match is used to compare domain names against the whitelist.

1.2.2 cc_ping_whitelist.txt

The file `cc_ping_whitelist.txt` contains a whitelist of PING payloads. When using the ICMP whitelist detection method, all payload patterns not in this file will be considered as a covert channel.

- Lines starting with `%` are comments.
- One hex encoded pattern per line.
- The pattern starts at the 25th byte of the ICMP payload.
- Prefix match is used to compare the payload against the whitelist patterns.

For instance, to whitelist the PING packet shown in Figure 1, the whitelist should contain the following pattern: `101112131415161718191a1b1c1d1e1f202122232425262728292a2b2c2d2e2f3031323334353637`



```
No.    Time          Source           Destination      Protocol Length Info
    71 9.582584000   173.194.116.55   10.20.6.138      ICMP         98 Echo (ping) reply    id=0x0a18,
    79 10.58046300(  10.20.6.138      173.194.116.55   ICMP         98 Echo (ping) request  id=0x0a18,
    80 10.58367400(  173.194.116.55   10.20.6.138      ICMP         98 Echo (ping) reply    id=0x0a18,

▸ Frame 79: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
▸ Ethernet II, Src: AsustekC_52:d6:e9 (60:a4:4c:52:d6:e9), Dst: IETF-VRRP-VRID_11 (00:00:5e:00:01:11)
▸ Internet Protocol Version 4, Src: 10.20.6.138 (10.20.6.138), Dst: 173.194.116.55 (173.194.116.55)
▾ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0

0000  00 00 5e 00 01 11 60 a4  4c 52 d6 e9 08 00 45 00   ..^...`. LR....E.
0010  00 54 71 7a 40 00 40 01  96 97 0a 14 06 8a ad c2   .Tqz@.@. ........
0020  74 37 08 00 a7 a9 0a 18  00 02 33 b0 37 56 00 00   t7...... ..3.7V..
0030  00 00 13 63 09 00 00 00  00 00 10 11 12 13 14 15   ...c.... ........
0040  16 17 18 19 1a 1b 1c 1d  1e 1f 20 21 22 23 24 25   ........ .. !"#$%
0050  26 27 28 29 2a 2b 2c 2d  2e 2f 30 31 32 33 34 35   &'()*+,- ./012345
0060  36 37                                              67
```

Figure 1: *Wireshark view of whitelisted PING pattern.*

1.3 Configuration Flags

The following flags can be used to control the output of the plugin:

Name	Default	Description
`CC_DETECT_DNS`	1	detect CCs in DNS traffic
`CC_DETECT_ICMP_ASYM`	1	detect CCs in ICMP traffic (using flow asymmetry)
`CC_DETECT_ICMP_WL`	0	detect CCs in ICMP traffic (using payload whitelist)
`CC_DETECT_ICMP_NP`	0	detect CCs in ICMP traffic (bidirectional non-ping flow)
`CC_DETECT_HCOVERT`	1	detect CCs in HTTP GET requests (hcovert)
`CC_DETECT_DEVCC`	1	detect CCs in TCP timestamp field (devcc)
`CC_DETECT_IPID`	1	detect CCs in the IP Identification field (covert_tcp)
`CC_DETECT_RTP_TS`	0	detect CCs in the RTP timestamp field
`CC_DETECT_SKYDE`	0	detect CCs in Skype silent packets (SkyDe)
`CC_DEBUG_MESSAGES`	0	activate debug output

1.4 Flow File Output

The covertChannels plugin outputs the following column:

Column	Type	Description	Flags
`covertChannels`	H16	Detected covert channels bitfield	

1.4.1 covertChannels

The covertChannels column is to be interpreted as follows:

covertChannels	Description
2^0 `(=0x0001)`	DNS CC (iodine, dnstunnel, nstx, . . .)
2^1 `(=0x0002)`	ICMP CC: asymmetric flow (hans, itun, loki, icmptx, . . .)
2^2 `(=0x0004)`	ICMP CC: non-whitelisted payload (hans, itun, loki, icmptx, . . .)
2^3 `(=0x0008)`	ICMP CC: bidirectional non-PING flow
2^4 `(=0x0010)`	HTTP GET URL-encoded CC (hcovert)
2^5 `(=0x0020)`	TCP timestamp CC (devcc)
2^6 `(=0x0040)`	IP Identification CC (covert_tcp)
2^7 `(=0x0080)`	RTP timestamp CC
2^8 `(=0x0100)`	Skype silent packets CC (SkyDe)

1.5 Plugin Report Output

The following information is reported:

- Aggregated `covertChannels`
- Number of covert channels packets

1.6 TODO

- Smarter IPID covert channels detection (stegtunnel)
- SSH/Telnet based covert timing channels detection