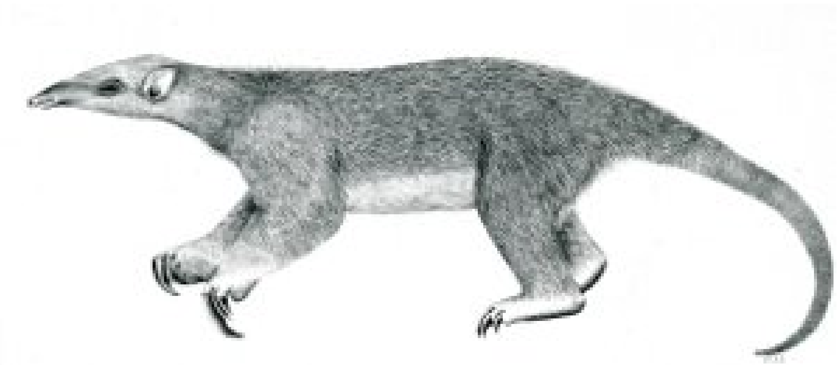
# Tranalyzer2

## tp0f

tp0f



Tranalyzer Development Team

# Contents

# 1 tp0f

## 1.1 Description

The tp0f plugin classifies IP addresses according to OS type and version. It uses initial TTL and window size and can also use the rules from p0f. In order to label non-TCP flows, the plugin can store a hash of already classified IP addresses.

### 1.1.1 Required Files

If `TP0FRULES=1`, then the file `tp0fL34.txt` is required.

## 1.2 Configuration Flags

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

| Name | Default | Description |
|---|---|---|
| `TP0FRULES` | 1 | 0: standard OS guessing; 1: OS guessing and p0f L3/4 rules |
| `TP0FHSH` | 1 | 0: no IP hash; 1: IP hash to recognize IP already classified |
| `TP0FRC` | 0 | 0: only human readable; 1: tp0f rule and classifier numbers |
| `TP0FL34FILE` | `"tp0fL34.txt"` | file containing converted L3/4 rules |

In *tp0flist.h*:

| Name | Default | Description |
|---|---|---|
| `MAXLINELN` | 4096 | maximal line input buffer size for *tp0fL34.txt* |
| `TCPOPTMAX` | 40 | maximal TCP option byted codes being stored and processed |

## 1.3 Flow File Output

The p0f plugin outputs the following columns:

| Column | Type | Description |
|---|---|---|
| `tp0fStat` | H8 | status |
| `tp0fDis` | U8 | initial ttl distance |
| `tp0fRN` | U16 | rule number that triggered |
| `tp0fClass` | U8 | OS class of rule file |
| `tp0fProg` | U8 | Program category of rule file |
| `tp0fVer` | U8 | version category of rule file |
| `tp0fClName` | SC | OS class name |
| `tp0fPrName` | SC | OS/Program name |
| `tp0fVerName` | SC | OS/Program version name |

### 1.3.1 tp0fStat

The `tp0fStat` column is to be interpreted as follows:

| tp0fStat | Description |
|---|---|
| `0x01` | SYN tp0f rule fired |
| `0x02` | SYN-ACK tp0f rule fired |
| `0x04` | — |
| `0x08` | — |
| `0x10` | — |
| `0x20` | — |
| `0x40` | IP already seen by tp0f |
| `0x80` | TCP option length or content corrupt |

## 1.4 Plugin Report Output

The number of packets which fired a tp0f rule is reported.

## 1.5 TODO

- Integrate TLS rules
- Integrate HTTP rules

## 1.6 References

- http://www.netresec.com/?page=Blog&month=2011-11&post=Passive-OS-Fingerprinting
- http://lcamtuf.coredump.cx/p0f3/