Communications Research Centre Canada
An Agency of Industry Canada
Centre de recherches sur les communications Canada
Un organisme d'Industrie Canada

Flow Profiling Using NetMate

A. De Montigny-Leboeuf, M. Couture, F. Massicotte
Communications Research Centre Canada, Ottawa, Ontario K2H 8S2 • e-mail: networksystems-security@crc.gc.ca, http://www.crc.gc.ca/networksystems_security

Abstract

We are proposing a practical implementation of flow behaviour indicators with NetMate.
The output provides valuable insights on traffic activities and can be particularly helpful to security analysts investigating suspicious flows.
The tool also has great potential for other use cases such as intrusion detection, insider threat detection, and traffic classification.

Introduction

In our approach, the flow attributes have discriminative power and provide insight into the traffic behaviour.
The set of flow attributes includes indicators of interactivity (human control), conversation, transaction, data transfer, and more.
The analysis is confined to headers at the network and transport layers and does not depend on access to payload.

Implementation in NetMate

NetMate is an Open Source Packet Processing Framework for Traffic Measurements.
It is distributed under the GNU license.
http://www.ip-measurement.org/tools/netmate

CRC's extended version of NetMate is available at:
http://www.crc.gc.ca/networksystems_security

It Includes several additional Processing Modules to calculate flow behaviour attributes; and one additional Export Module to describe/recognize flows.

The Export Module includes a rule engine that compares flow attributes with signature rules.

This module processes two types of rules:
–Description Rules: the output provides a description of the traffic activities.
–Recognition Rules: the output contains the list of predefined traffic profiles that matched.

These two types of rules are independent from one another.

Simple to Use

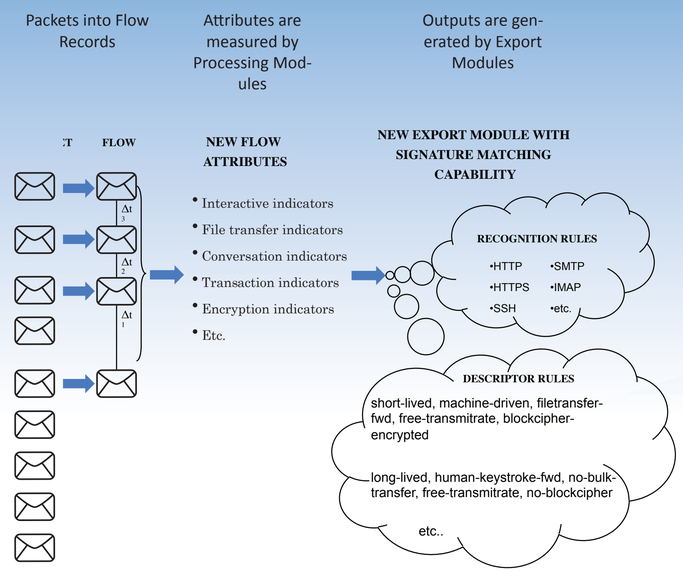
Installation and use are simple and similar to other common packet processing tools such as tcpdump and snort.
We have NOT changed how NetMate is used. The following command is a simple example for processing a pcap file:

```
netmate -r <rule_file.xml> -f
      <input_file.pcap>
```

The NetMate <rule_file.xml> specifies which Processing Modules and Export Modules to use, with their configurable parameters. Here is an example of the syntax:

```
<ACTION NAME="tp_filetransfer">
      <PREF
NAME="maxInterval">50000</PREF>
      <PREF
NAME="minBigPacketSize">225</PR
EF>
</ACTION>
```

How it Works



Output Examples

```
2009-07-08 13:06:22.204097, tcp_handshake, 4, 6, 192.168.0.2, 192.168.0.3, 57475,
22, directional-fwd, numerous-packets, short-lived, machine-driven, filetransfer-
fwd, free-transmitrate, blockcipher-encrypted, , , , , , ssh, , , , , , , , ,

2009-07-08 13:06:43.385757, tcp_handshake, 4, 6, 192.168.0.1, 192.168.0.2, 3570, 22,
directional-bkwd, not-so-many-packets, long-lived, human-keystroke-fwd, no-bulk-
transfer, free-transmitrate, blockcipher-encrypted, , , , , , ssh, , , , , , , , ,
```
SSH

```
2009-07-08 13:27:48.409318, tcp_handshake, 4, 6, 192.168.0.1, xx.xx.xx.xx, 3761, 21,
directional-bkwd, not-so-many-packets, long-lived, human-keystroke-fwd, no-bulk-
transfer, free-transmitrate, no-blockcipher , , , , , , , , , ftp-command, , , , ,

2009-07-08 13:28:57.281993, tcp_handshake, 4, 6, xx.xx.xx.xx, 192.168.0.1, 63146,
5003, bidirectional, few-packets, instantaneous, machine-driven, no-bulk-transfer,
free-transmitrate, no-blockcipher, , , , , , , , , ftp-data, , , ,

2009-07-08 13:28:58.936410, tcp_handshake, 4, 6, xx.xx.xx.xx, 192.168.0.1, 61655,
5004, bidirectional, numerous-packets, short-lived, machine-driven, filetransfer-
fwd, free-transmitrate, no-blockcipher, , , , , , , , , ftp-data, , , ,
```
FTP

```
2009-07-08 13:29:03.11508, tcp_handshake, 4, 6, 192.168.0.1, yy.yy.yy.yy, 3582, 80,
directional-bkwd, not-so-many-packets, short-lived, machine-driven, filetransfer-
bkwd, free-transmitrate, no-blockcipher, http-web-browsing, , , , , , , , , , , ,
,

2009-07-08 13:29:35.590982, tcp_handshake, 4, 6, 192.168.0.20, xx.yy.xx.yy, 59771,
80, directional-bkwd, numerous-packets, long-lived, machine-driven, filetransfer-
bkwd, regular-transmitrate-bkwd, no-blockcipher, , , , , , , , , , , tcp-
audiostream, , ,
```
HTTP

```
2009-07-08 13:30:37.228348, no_syn, 4, 17, 192.168.0.1, 192.168.63.255, 138, 138,
unidirectional-fwd, few-packets, instantaneous, machine-driven, no-bulk-transfer,
free-transmitrate, no-blockcipher, , , , , , , , , , , , , ,

2009-07-08 13:32:53.860279, no_syn, 4, 17, 192.168.0.1, 192.168.0.32, 53748, 53,
directional-bkwd, few-packets, instantaneous, machine-driven, no-bulk-transfer,
free-transmitrate, no-blockcipher, , , , , , , , , , , , , , dns-udp,
```
UDP

```
2009-07-08 13:32:56.382509, tcp_handshake, 4, 6, 192.168.0.1, 192.168.0.2, 3571,
5901, directional-bkwd, numerous-packets, long-lived, human-keystroke-fwd,
filetransfer-bkwd, free-transmitrate, blockcipher-encrypted, , , , , , , , , , , ,
, ,

2009-07-08 13:33:37.299356, tcp_handshake, 4, 6, 192.168.0.1, 192.168.0.8, 3572,
3389, directional-bkwd, numerous-packets, persistent, possibly-human-cmdline,
filetransfer-bkwd, free-transmitrate, no-blockcipher, , , , , , , , , , , , , , ,
```
R-Ctrl

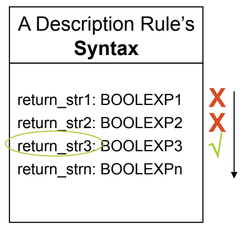
Highlights

-Intuitive outputs,
-Based entirely on behaviour (not on port numbers nor on protocol decoding),
-Provide insights on traffic activities even if flows are running on non-standard ports or if the payload is obfuscated/encrypted.

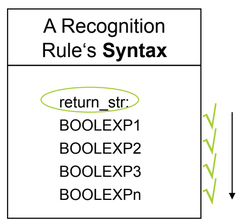
Customizable Rules

The rules are stored in signature files. The user can modify them without having to recompile.

Each rule consists of a block of consecutive non-empty lines.



A Description Rule returns the string corresponding to the first Boolean expression that evaluates to TRUE.



A Recognition Rule returns the string at the beginning of the rule if all expressions evaluate to TRUE.

Examples of Description Rules

```
#Test on duration (microseconds)
instantaneous: tp_basic_noseppaths.duration < 1000000
short-lived: tp_basic_noseppaths.duration < 60000000
long-lived: tp_basic_noseppaths.duration < 600000000
persistent: 1

#Test on file transfer
filetransfer-bidir: tp_filetransfer.isFiletransfer_fwd &
tp_filetransfer.isFiletransfer_bkwd
filetransfer-fwd: tp_filetransfer.isFiletransfer_fwd
filetransfer-bkwd: tp_filetransfer.isFiletransfer_bkwd
no-bulk-transfer: 1
```

Examples of Recognition Rules

```
telnet:
#test payload
tp_payloaddistribution.payload[0-10]_fwd > 0.8 &
tp_payloaddistribution.payload[350-Inf]_fwd = 0
#test data byte ratio
tp_basic_noseppaths.forwardOverBackwardPayloadSizeRatio < 0.2 &
tp_basic.nonEmptyPacketCount_fwd < tp_basic.nonEmptyPacketCount_bkwd
#test first non-empty packet size
tp_basic_noseppaths.firstNonEmptyPacketSize < 30
#test on direction of first 2 non-empty packet: first=Server, second=Client
tp_basic_noseppaths.firstFewNonEmptyPacketDirections % 100 = 12
#test non-empty packet ratio
tp_basic.nonEmptyPacketCount_bkwd > 0.4 * tp_basic.packetCount_bkwd
#test transaction
tp_pingpong_noseppaths.transactionIndicator > 0.7

####################
# The profiles do not have to define a protocol, they can also define general
types
# of flows you would be interested in detecting
####################

foo:
#potential encrypted exfiltration???
tp_basic_noseppaths.forwardOverBackwardPayloadSizeRatio > 1 &
tp_basic.payloadSizeSum_fwd > 10000
tp_basic.nonEmptyPacketCount_fwd > 10 & tp_basic.nonEmptyPacketCount_bkwd > 10
& (tp_cipherblock.isCipherblock_fwd | tp_cipherblock.isCipherblock_bkwd)
```