The *Bro* 0.8 User Manual

Vern Paxson

Contents

7.18.1 ident

List of Figures

List of Tables

Chapter 1

them by simply changing variables associated with the analyzer. Furthermore, Bro supports a notion of *refining* the initialization of a variable, so that, in a *separate*

Important note: some versions of

prints

```
128.3.192.0
```

and even event handlers:

```
event bro_done()
    {
    print "all done!";
    }
```

```
redef ftp_guest_ids = { "anonymous", "guest", "visitor", "student" };
```

`-s` *signaturefile*

2.2.2 The *hf* utility

The *hf* utility reads text on *stdin* and attempts to convert any “dotted quads” it sees to hostnames. It is very convenient for running on Bro log files to generate human-readable f887(t)Seethe manual page included w “” 1.66397(“)-he distribution for details.

- function

3.3 Numeric Types

`int`, `count`, and `double`

```
/foo/ | /bar/
```

but

```
/^oob/ in "foobar"
```

does not, since the text “oob” does not appear the beginning of the string “foobar”. Note, though, that the $ regular

Accordingly, `net` values hold either an 8-bit class A prefix, a 16-bit class B pre

and if it corresponded to all there, then:

```
function nothing_special(): string
    {
    if ( panic_mode )
        return "look out!";
    else
        return "nothing special";
    }

global a: table[.24962(c)-2.24962(i)-ount] of string &default = nothing_specia
```

You specify multiple attributes by listing one after the oth

3.16 Event handlers

3.17 The any type

```
log expr-list ;
```

The expressions are converted to a list of strings, which are then logged as a comma-separated list. “Logging” means recording the values to `bro_log_file`

Yields the boolean §

Compares two values for equality or inequality, yielding a `bool` value. Defined for all non-compound types except `pattern`.

relational

Syntax:

```
expr₁ < expr₂
expr₁ <= expr₂
expr₁ > expr₂
expr₁ >= expr₂
```

Compares two values for magnitude ordering, yielding a `bool`

Defines an *anonymous function*, which, in abstract terms, .66393(b)-5s how you specify a constant of ty

though to do so you need to enclose the function in parenthese

Returns the sub-value of *expr*$_1$ indexed by the value of *expr-list*$_{2ę}$ which must be compatible with the index type of *expr*$_1$

Chapter 5

Global and Local Variables

5.1 Overview

```
global a = 5;
```

also declares a

5.1.6 Refinement

To do.

- `&redef`
- `&add_func`
- `&delete_func`

Chapter 6

```
partial_connection_ok : bool
```

`inactivity_timeout : interval`

If a connection is inactive, time it out after this interval. If 0 secs, then don’t time it out.

last_stat : net_stats

`distinct_rejected_PTR_requests : table[addr] of count`

How many DNS PTR requests from the given source adrmss we5(m)-209.68e 5(m)-209.68e r rejected. An nprrt is generated if this number crsses a threshold, namely, `report_rejected_PTR_thresh`.

`distinct_answered_PTR_requests : table[addr] of count`

How many DNS PTR requests from the given source adrmss we5(m)-209.68e 5(m)-209.68e r answered.

`report_rejected_PTR_thresh : count`

I this many DNS requests from a host a5(m)-209.68e 5(m)-209.68e rejected PTR requests we generate an event.

`report_rejected_PTR_f962(a)-2.24962(d)-2.2ctor : double`

6.1.13 hot.bro

`same_local_net_is_spoof : bool`

`flag_successful_service : table[port] of string`

Successful connections to any of the specified ports are flagg

`http_sessions : table[addr, addr] of http_session_info`

A [source, destination] indexed table of `http_session_info` records07974.5(t)-2.24962(a)-2.249601p

6.1.18 icmp.bro

`icmp`

`interconn_default_pkt_size : count`

The estimated packet size used to calculate the number of packets missed when we see an ack above a hole. *Fix me: Please verify. NOTE: This variable is const*

, so may only be changed via the config file.

`_stat_period : interval`

interconn

full_

non_ASCII_hosts : set[addr]

6.1.25 portmapper.bro

`rpc_`

report_

skip_

```
external_orig: bool;
in_data: bool;
num_cmds: count;
num_replies: count;
```

```
smtp_session_by_message_id : table[string] of smtp_session_info
```

6.1.34 tftp.bro

`tftp_`

6.1.38 Uncategorized

Fix me: These need categorization.

`bro_log_file : file`
: Used to record the messages logged by

6.2 Predefined Functions

Bro provides a number ofbuilt-in functions:

`active`

`connection_record (id: conn_id): connection`

Returns the `connection` record corresponding to the given connection identifier. *Note: If the connection does not exist, then exits with a fatal run-time*

Deficiency: If Bro had an exception mechanis

A format specifier of "`.`**`n`**" (coming after **`m`**, if present) instructs `fmt` to use a precision of **`n`** digits. You can only specify a precision for the `e`, `f` or `g` formats. (`fmt` generates a run-time error if either **`m`** or **`n`** exceeds 127.)

The different format specifiers are:`nt ara-5.88993(e)-1.66308(a)-1.66638(t)0.965521(er)55.663(7)-5.89`

`is_tcp_port (p: port): bool`

Returns true if the given `port` value corresponds to a TCP port, false otherwise (i.e., it belongs to a UDP port).

`length (args: any): count`

```
to_lower (s: string): string
```

Chapter 7

Analyzers and Events

In this chapter we detail the different analyzers that Bro provides. Some analyzers look at traffic in fairly generic terms, such as at the level of TCP or UDP connections. Others delve into the specifics of a particular application that is carried on top of TCP or UDP.

As we use the term here, *analyzer* primarily refers to Bro's event engine. We use the term *script* to refer to a set of event handlers (and related functions and variables) written in the Bro lang42.561(t)042.561(f)0*module* to a script that serves

```
event bro_init()
    {
    if ( restrict_filter == "" && capture_filter == "" )
        print "tcp or not tcp";  # Capture everything.

    else if ( restrict_filter == "" )
        print capture_filter;

    else if ( capture_filter == "" )
        print restrict_filter;

    else
        print fmt("(%s) and (%s)", capture_filter, restrict_filter);

    exit();
    }
```

```
type conn_id: record {
    orig_h: addr;  # Address of originating host.
```

For ICMP, Bro likewise creates a connection the first time it s

Note: Per the discussion above, a client attempting to connect to a server will result in one *of*

The analyzer invokes `check_hot` with a mode of `CONN_ESTABLISHED` and then again with a mode of `CONN_FINISHED`

Symbol	Name	

```
log_hot_conn (c: connection)
```

local_24_

hot_src_

flag_rejected_

Sta

7.6 The `scan` Anayer

The `scan` analyJ /R13 9.9(r)-329.473(d)-5.88993(e)- /66516(t)0.965521(e)-/66516(c)-/66393(t)0.964296(s)-333.698(c)- /66516(o)-5.88 (across scanning, connection attempts to
many diffJ /R13 9.9(r)-4.2603(e)- /66516(n)-5.88993(t)-264.021(s)3.55944(J /R13 9.9(r)-4.2603(v)-5.88993(i)0.965521(c)- /66516(J /R13

skip_accounts_tried_hosts::set[addr] 997]TJ -252f962]TJ /R14 9.96264 Tp6251.6 -11.88 Td [(D)-0.70024

Do not do bookkeeping for account attempts for the given hosts.

Default: empty.

skip_outbound_services : set[port]

Do not do outbound-scanning bo: 8993(-)-4.2603(s)3.55944(c)-1.66393(a)-1.8267(o)-5.89115e(u)-5.8887g for ahesetoemftf

ou s.dbuu71280.12 -13129f -24.29115e(a)-1.66516oasJ -25a15 T3129

7.6.2 `scan` functions

7.7 The `port-name` Module

The `port-name` utility module provides one redefinable variable and one cal

7.10 The `active` Module

7.14 The `frag` Modole

The `frag`

```
type ftp_session_info: record {
    id: count;              # unique number associated w/ session
```

```
972499885.784104 #26 131.243.70.68/1899 > 64.55.26.206/ftp start
972499886.685046 #26 response (220 tuvok.ooc.com FTP server
    (Version wu-2.6.0(1) Fri Jun 23 09:17:44 EDT 2000) ready.)
972499886.686025 #26 USER anonymous/IEUser@ (logged in)
972499887.850621 #26 TYPE I (ok)
972499888.421741 #26 PASV (227 64.55.26.206/2427)
972499889.493020 #26 SIZE /pub/OB/4.0/JOB-4.0.3.zip (213 1675597)
972499890.135706 #26 *RETR /pub/OB/4.0/JOB-4.0.3.zip, ABOR (complete)
972500055.491045 #26 response (225 ABOR command successful.)
```

In addition, `ftp_log`

7.16.4

```
972482763.371224 %1596 start 200.241.229.80 > 131.243.2.12
%1596 GET /ITG.hm.pg.docs/dissect/portuguese/dissect.html
```

Figure 7.17.1 shows an example of what entries in this file look like. Here we see a transcript of the 1596th HTTP session seen since Bro started running. The first line gives its start time and the participating hosts. The next five lines all correspond to `GET`

Type of confusion	Meaning
`"excessive_typeahead"`	The user has typed ahead 12 or more lines. *Deficiency: The upper bound*

login_

7.19.3 `login` functions

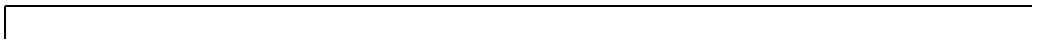
The standard `login` script provides the following functions for external use:

`is_login_conn (c: connection): bool`
: Returns true if the given connection is one analyzed by `login` (currently, Telnet or Rlogin), false otherwise.

`hot_login (c: connection, msg: string, tag: string)`

and *DEL* keystroke editing (§ 7.19.2, page 131). `successful` should be true if the user has successfully

Note: As bor



```
NFS_world_servers : set[addr]
```

pm

Status description	Meaning
"ok"	

Field	

Action	Meaning
SIG_IGNORE	Ignore the signature completely.
SIG_QUIET	Process for scan detection but don't report individually.
SIG_FILE	Write matches to

`client_cert`
: The information from the client certificate, if available.

`server_cert`
: The information from the server certificate, if available.

`id_index`
: Index into associated `SSL_sessionID_record` table.

`handshake_cipher`

Invoked upon the client side of connection `c` when the analyzer sees a `CLIENT-HELLO` of SSL version `version` including the cipher suites the client offers `cipherSuites`.

The version can be `0x0002`, `0x0300` or `0x0301`. A new entry is generated inside the SSL connection table

Invoked when the analyzer receives an SSL alert. The `level`

Default: empty.

`weird`

`active_connection_reuse`
: A new connection attempt (initial SYN) was seen for an already-established connection that has n5.89115(n)-t yet terminated.

`bad_HTTP_reply`
: The first line of a reply from an HTTP server did n5.89115(n)-t include *HTTP/version*.

`bad_HTTP_version`
: The first line of a reques from an HTTP client did n5.89115(n)-t include *HTTP/version*.

`bad_`

bad_

fragment_size_inconsistency

7.24.8 Events generated by the standard scripts

The following events are generated by the standard scripts themselves:

`bad_pm_`

Chapter 8

Signatures

8.1 *verview

In addition to the policy language, Bro provides another lan

8.2.1 Conditions

Header conditions

Content conditions

Content conditions are defined by regular expressions. We differentiate two kinds of content conditions: first, the expression may be declared with the `payload`

Chapter 9

```
[4] policy/demux.bro:40
[5] policy/login.bro:496
[a] All of the above
[n] None of the above
Enter your choice: 1
Breakpoint 1 set at connection_finished at policy/conn.bro:268
```

Now we resume execution; when the breakpoint is reached, execution stops and the debugger prompt returns.

```
(Bro [1]) continue
Continuing.
Breakpoint 1, connection_finished(c = ’[id=[orig_h=1.0.0.163,
orig_p=2048/tcp, resp_h=1.0.0.6, resp_p=23/tcp], orig=
[size=0, state=5], resp=[size=46, state=5], startm-2.24962(e)962(63962(a)-2.24962(t)-2.24962(e)-2.249
```

Command	Shortcut	Description
help		Get help with debugger commands
quit		Exit Bro
next	n	Step to the followi478(x)-5.88993(i)0.965521(t)-239.931(B)4.52619(r)-4.26153(o)-5.8887]TJ ET Q q 4.8

Example. Lastly, execution tracing may be combined with the debugger. Here we send outp8993(m)1.89428(p)4-t(d)64.01915(t)0.

delete (d) With no arguments, permanently delete all breakpoints. If n

10.7 Timer management

10.8 SYN-FIN filtering

10.9 Split routing

10.17 Demultiplexing

10.18 Bro init file

10.19 Hostnames vs. addresses

10.20 The hot-report script

10.21 Use of libpcap/BPF

[MJ93, MLJ94]

10.22 The problem of evasion

[PN98]

10.23 Backscatter

10.24 Playing back traces

10.25 Discarders

10.26 Differences between this release and the previous one

10.27 Alert cascade

10.28 The need for subtyping

E.g., `src addr` vs. `dst addr`, perhaps using attributes.

10.29 The need for CIDR masks

10.30 The wish list

10.31 Known bugs

Bibliography

[RFC2373] R. Hinden and S. Deering, “IP Version 6 Addressing Architecture,” RFC 2373, Jul. 1998.

[MJ93] S. McCanne and V. Jacobson, “The BSD Packet Filter: A N

[RFC1831] R. Srinivasan, “RPC: Remote Procedure Call Protocol Specification Version 2,” RFC 1831, Aug. 1995.

[RFC1832] R. Srinivasan, “XDR: External Data Representation Standard,” RFC 1832, Aug. 1995.

[RFC1939] J. Myers and M. Rose, “Post Office Protocol - Version 3,” RFC 1939, May 1996.

[RFC1945] T. Berners-Lee, R. Fielding and H. Frystyk, “Hype

Index

code_red_list1

telnet 10.88879rs152.2-231.799taosPn50 28 231.799mbom1A50 28 231.799mntom1

repeat