LOGRIP - Manual

8/1/2025

Document Version history:
8/1/2025 ver 1.0 Rama Hoetzlein

USAGE:
> logrip {access_log} {config_file.conf}

Both are required. There are no other arguments to logrip.
The access_log must be either .txt or .log, and should contain raw server log page hit data which will be parse and analyzed.
The config_file must be .conf, and contains a list of key-value settings described below.
Operation of logrip is fully controlled from the .conf file.

HOW TO USE LOGRIP
1. Generate or retrieve access logs
2. Modify or use an existing config .conf file to control parsing and policy settings
3. Run logrip on an access log with a given .conf file
4. Examine the output products

I. GENERATING LOGS

Logrip takes a historic server access log as input.
To generate logs you would typically use journalctl, or others server tools that output logs.
Here are examples of how to generate logs for apache2 or ruby-on-rails.
How to generate **Apache2 logs:**
> cd /var/log/apache2
> ls -l -a
> zcat access.log.*.gz > apache.log
> cat access.log.1 access.log >> apache.log
Now run logrip the apache.log input file along with the apache2.conf config file.

How to generate **Ruby-on-Rails logs:**
> journalctl | grep {project} | grep 'Started GET' > ruby.log
Now run logrip with the ruby.log input file along with the ruby.conf config file.

II. CONFIG – PARSER SETTINGS

Two settings control the dynamic log parser.
The **format string** specifies the format of each get/post line in the input log.
Format strings may contain the following:

{X.X.X.X}	IP address
{AAA}	Name of user/client
{GET}	Type of HTTP request. Will match to GET, POST or HEAD
{PAGE}	Page retrieved
{PLATFORM}	Platform details
{DD/MMM/YYYY}	Date in 12/Jan/2025 format
{YYYY-MM-DD}	Date in 2025-01-12 format
{HH:MM:SS}	Time in 24-hour HH:MM:SS format
{RETURN}	Return code
{BYTES}	Number of bytes in return
{NNN}	Any captured number
*	Match any number of characters. Not captured.
literal	Match specific words or numbers. Not captured.

The {...} strings are capture groups and will be recorded as part of the log hit.
The literals can be any other characters, numbers, symbols or whtespaces that are matched but not captured. The following are examples of the format strings for capturing logs for Apache2 or Ruby-on-Rails.

Apache2 format string:
{X.X.X.X} {AAA} {AAA} [{DD/MMM/YYYY}:{HH:MM:SS} +{NNN}] "{GET} {PAGE}HTTP/*" {RETURN} {BYTES} "*" {PLATFORM}

Ruby-on-Rails format string:
* Started {GET} "{PAGE}" for {X.X.X.X} at {YYYY-MM-DD} {HH:MM:SS}

The other parser setting is **debugparse**, which is set to either 0 or 1, indicating if you wish to debug/troubleshoot the parser. Set this to 1 when you are trying to test new format strings.
debugparse: 1

III. CONFIG – POLICY SETTINGS

Several settings allow for control over blocking policy.

The following table is a brief summary, with more details below.

Setting	Unit	Filter Type	Default value
min_ip_b	# machines	B-subnet filter	1024
min_ip_c	# machines	C-subnet filter	3
max_ip_c	# machines	C-subnet filter	80
max_robot	# hits per IP	robots filter	10
max_daily_hits	# hits per day	daily hits	100
max_daily_range	minutes per day	range check	360
max_consec_days	days	consecutive range	5
max_consec_range	minutes per day	consecutive range	240
max_daily_ave	# ave hits per day	smart throttling	40
max_daily_ppm	# daily hits per min	smart throttling	40

Detailed list of policy settings and value ranges:

min_ip_b

Unit: # machines in subnet

Purpose: B-subnet allow filter

Short Desc: Allow B-subnets with machines < given #, otherwise apply scoring

Details: Controls B-subnet filtering.
If the number of machines in a B-subnet is *less* than this number, then it should not be blocked (allowed). B-subnets are 65536 machines. If the B-subnet uses more machines, then we apply the scoring metrics to the subnet (does not mean we just block them).

Aggressive: 0, lower numbers

Turn off: 65536

Default: 1024

Example: B-subnet hits server with 80 machines, that is OK (not blocked) because it is less than 1024. 80 machines out of 65536 doesn't quality as B-level attack.

min_ip_c

Unit: # machines in subnet
Purpose: C-subnet allow filter
Short Desc: Allow C-subnets with machines < given #, otherwise apply scoring
Details: Controls C-subnet filtering.
Aggressive: 0, lower numbers
Turn off: 256
Default: 3
Example: C-subnet hits server with 2 machines, that is OK (not blocked) because 2 machines doesn't qualify as a C-level data center attack.

max_ip_c

Unit: # machines in subnet
Purpose: C-subnet block filter
Short Desc: Block C-subnets with machines > #
Details: Controls C-subnet blocking.
Aggressive: 0, lower numbers
Turn off: 256 – a C subnet can't use more than 256 machines
Default: 80
Example: C-subnet hits server with 104 machines (greater than 80).
104 out of 256 is 40% of the subnet hit the server.
This is probably a C-level data center attack, block them.

max_daily_hits

Unit: maximum # hits per day
Purpose: daily hit blocking
Short Desc: Block IPs which queries server > # hits per day
Details: Controls for aggressive hosts that get too many pages per day.
Aggressive: 0, lower numbers
Turn off: 10000 – even machines can't hit 10000 pages/day
Default: 200
Example: One IP tries to get 300 pages a day, regardless of when. Block them.

max_daily_range

Unit: minutes per day
Purpose: range blocking
Short Desc: Block IPs that query the server > # minutes per day
Details: Controls for hosts that spend to much time on the server throughout the day.
If a host hits a few times per hour, but keeps hitting for many hours,

its probably a machine.

Aggressive: 0, lower numbers

Turn off: 1440 (max value)

Default: 360 mins (= 6 hrs)

Example: One IP tries to access the server for 7 hours/day, regardless of how often. Block them.

max_consec_day

Unit: days

Purpose: consecutive filter

Short Desc: Block IPs that query the server > # days... AND max_consec_range

Details: Controls for hosts that access over many consecutive days. If your host tries to access for many days, AND the range over that day is also high, then you're probably a machine. Block.

Aggressive: 0, lower numbers

Turn off: 365 – logs should never go for a full year

Default: 5 days

Example: One IP tries to access the server for over 5 days in a row, AND also for 4 hours each day. Block them.

max_consec_range

Unit: minutes per day

Purpose: consecutive filter

Short Desc: Blocks IPs that query server > minutes per day.. AND max_consec_days

Details: Controls for hosts that access over many consecutive days. If your host tries to access for many days, AND the range over that day is also high, then you're probably a machine. Block.

Aggressive: 0, lower numbers

Turn off: 1440 (max value) – nothing operates >24 hrs a day (1440 mins)

Default: 240 min/day (=4 hrs/day)

Example: One IP tries to access the server for over 5 days in a row, AND also for 4 hours each day. Block them.

max_daily_ave

Unit: # hits per day

Purpose: smart throttling

Short Desc: Block IPs who average hits > # hits per day.. AND max_daily_ppm

Details: Controls for hosts that access faster than the maximum pages per minute,

AND also that average more than # hits per day. If this was only by pages per min, it would be classic throttling (set max_daily_ave = 0). By also checking max_daily_ave, we can lower the ppm throttling value an still get meaningful results. A human may be faster than 1 ppm, but they won't be as steady and persistent throughout the day (lower average).

Aggressive: 0, lower numbers
Turn off: 10000 – not even machines will average >10000 hits/day
Default: 100 hits/day
Example: One IP tries to access the server faster than 1 hit per min (a fast human could), and ALSO hits on average >100 pages per day. Block them.

max_daily_ppm

Unit: # hits per min (same as ppm)
Purpose: smart throttling
Short Desc: Blocks IPs that query > # hits per min.. AND max_daily_ave

IV. CONFIG – LOAD SETTINGS

Two settings control the load visualization, out_load.png
For example:
load_duration: 80
load_scale: 40

load_duration

Estimates the impact, in seconds, of a single request to the server including network traffic, servicing the request, and responding to host. In a real world scenario the response time will vary widely as some requests are easy while others take time to process. The value is used to *estimate* the server load, so an average between these extremes can be used.

load_scale

Vertical scaling of the y-axis. The unit for the y-axis, representing load, is the average number of simultaneous requests being handled. This setting controls how the plot is scaled on y-axis.

V. CONFIG – VISUALIZATION SETTINGS

Two settings control the visualization output products, which are generated for the original (pre-filtered) data, the blocking actions taken, and the filtered output. Each plot shows hits plotted as time (x-axis) versus host IP (y-axis). These are all output as .png image files.
The default settings are:
vis_res: 2048, 1024
vis_zoom: 0, 0, 1000, 224

vis_res

This setting is the resolution of the visualization images. The default is 2048, 1024. Larger resolutions will result in finer dots, while lower resolutions will give bigger dots.

vis_zoom

This controls the zoomed area to plotted.
It is a 4-component vector, with the following meaning:
x = left = Starting day to plot (x-axis)
y = bottom = Lowest Class A mask to plot (y-axis)
z = right = Ending day to plot (x-axis)
w = top = Highest Class A mask to plot (y-axis)
The default settings of 0, 0, 1000, 224, corresponds to the entire data. If the ending day (z) is too large, it is reduced to the total maximum number of days in the access log. The highest that the Class A mask could be is 224, since there are generally no viable host IP addresses above 224.x.x.x. Therefore, the setting 0,0, 1000, 224 will plot the entire data.

Notice that the vertical axis is extremely dense, since the full IP range is 256^4 = 4.294 billion, which is compressed into just 1024 vertical pixels. So zooming is helpful. One can zoom into a sub-region of the data by modifying the visible time range (x and z) or the IP range (y and w). For example, to look at IP ranges 30.x.x.x to 50.x.x.x for only days 1 thru 3, one would use: 1, 30, 3, 50