LogRhythm MPE Rule Builder Parsing Guide

April 26, 2017 — Revision A

LogRhythm-MPE-RuleBuilderGuide-revA

LogRhythm Inc.
4780 Pearl East Circle
Boulder, CO 80301

(303) 413-8745

www.logrhythm.com

LogRhythm Customer Support
support@logrhythm.com

LogRhythm
The Security Intelligence Company

Contents

Parsing Fields and Tags

Using the Rule Builder, you can create custom parsing rules for your own log sources. The following tables provide lists of all the metadata fields LogRhythm can parse, as well as their associated parsing tag(s) and default regex. The fields are grouped by how they appear in the Web Console. If you do not see a field in the Web Console in the same tab as this document, you may have tagged the field as a favorite, in which case the field will appear in the Favorites tab instead of the main group tab as shown in this document.

NOTE: All mapping and parsing tags are lower case.

Fields denoted with † are available for parsing and investigations, and they are viewable in the Web Console. These fields will be available in all product features in the LogRhythm 7.3 release.

The Application Tab

Display Field	Description	Tag(s)	Default Regex
Application	Application derived by IANA protocol and port number or directly assigned in MPE processing settings.	N/A	N/A
Object	The resource (i.e., file) referenced or impacted by activity reported in the log.	<object>	\w+
Object Name	The descriptive name of the object. Do not use unless Object is also used.	<objectname>	\w+
Object Type †	A category type for the object (e.g., file, image, pdf, etc.).	<objecttype>	\w+
Hash †	The hash value reported in the log. Choose MD5 > Sha1 > Sha256.	<hash>	\w+
Policy †	The specific policy referenced (i.e., Firewall, Proxy) in a log message.	<policy>	\w+
Result †	The outcome of a command operation or action. For example, the result of *quarantine* might be *success*.	<result>	\w+
URL	The URL referenced or impacted by activity reported in the log. You may need to override the default regex for URLs that are not HTTP/HTTPS.	<url>	https?://.+
User Agent †	The User Agent string from web server logs.	<useragent>	\w+

Display Field	Description	Tag(s)	Default Regex
Response Code †	The explicit and well-defined response code for an action or command captured in a log. Response Code differs from Result in that response code should be well-structured and easily identifiable as a code.	<responsecode>	\w+
Subject	The subject of an email or the general category of the log.	<subject>	\w+
Version	The software or hardware device version described in either the process or object.	<version>	\w+
Command	The specific command executed that has been recorded in the log message.	<command>	\w+
Reason †	The justification for an action or result when not an explicit policy.	<reason>	\w+
Action †	Field for "what was done" as described in the log. Action is usually a secondary function of a command or process.	<action>	\w+
Status †	The vendor's perspective on the state of a system, process, or entity. Status should NOT be used as the result of an action.	<status>	\w+
Session Type †	The type of session described in the log (e.g., console, CLI, web). Unique from IANA Protocol.	<sessiontype>	\w+
Process Name	System or application process described by the log message.	<process>	\w+
Process ID	Numeric ID value for a process.	<processid>	\d+
Parent Process ID †	The parent process ID of a system or application process that is of interest.	<parentprocessid>	\w+
Parent Process Name †	The parent process name of a system or application process.	<parentprocessname>	\w+
Parent Process Path †	The full path of a parent process of a system or application process.	<parentprocesspath>	\w+
Quantity	A numeric count of something. For example, there are 4 lights (quantity is 4).	<quantity>	[0123456789\.]+

Display Field	Description	Tag(s)	Default Regex
Amount	The qualitative description of quantity (percentage or relative numbers) For example, half the lights are on (amount is .5 or 50). Amount is also used for currency.	<amount>	[0123456789\.]+
Size	Numeric description of capacity (e.g., disk size) without a specific unit of measurement. Size is generally used as a limit rather than a current measurement. Use Amount for non-specific measurements.	<size>	[0123456789\.]+
Rate	Defines a number of something per unit of time without a specific unit of measurement. Always expressed as a fraction.	<rate>	[0123456789\.]+
Duration	The elapsed time reported in a log message, derived from multiple fields. Timestart and Timeend need custom parsing patterns.	If log has start/end use: (?<timestart>*pattern)* (?<timeend>*pattern)* If log has elapsed time use: <days> <hours> <minutes> <seconds> <milliseconds> <microseconds> <nanoseconds>	[0123456789\.]+ *Note: Time Start and Time End tags must be overloaded to function properly.*
Session	Unique user or system session identifier.	<session>	\w+
Known Application	Application derived from IANA protocol and port number. If a known application cannot be derived, it is displayed as unknown.	N/A	N/A

Kbytes/Packets Tab

Display Field	Description	Tag(s)	Default Regex
• **Host (Impacted) KBytes Rcvd** • **Host (Impacted) KBytes Sent** • **Host (Impacted) Kbytes Total**	The number of bytes sent or received in the context of the Impacted Host. • Rcvd – Bytes received by impacted host • Sent – Bytes sent by impacted host • Total – Total bytes in session as seen by impacted host	Use the appropriate tags based upon the units and direction represented by the log data: <bitsin>, <bitsout> <bytesin>, <bytesout> <kilobitsin>, <kilobitsout> <kilobytesin>, <kilobytesout> <megabitsin>, <megabitsout> <megabytein>, <megabyteout> <gigabitsin>, <gigabitsout> <gigabytein>, <gigabyteout> <terabitsin>, <terabitsout> <terabytesin>, <terabytesout> <petabitsin>, <petabitsout> <petabytesin>, <petabytesout>, <bits>, <bytes>, <kilobits>, <kilobytes>, <megabits>, <megabytes>, <gigabits>, <gigabytes>, <terabits>, <terabytes>, <petabits>, <petabytes>	[0123456789\.]+
• **Host (Impacted) Packets Rcvd** • **Host (Impacted) Packets Sent** • **Host (Impacted) Packets Total**	The number of packets sent or received in the context of the Impacted Host. • Rcvd – Packets received by impacted host • Sent – Packets sent by impacted host • Total – Total packets in session as seen by impacted host	<packetsin>, <packetsout>, <packets>	[0123456789\.]+

Classification Tab

Display Field	Description	Tag(s)	Default Regex
Classification	Value is determined based on the MPE Rule's assigned Common Event.	N/A	N/A
Common Event	Value is determined based on the MPE Rule's assigned Common Event	N/A	N/A
Priority	Value is determined based on the Risk-Based-Priority (RBP) calculation.	N/A	N/A
Direction	Indicates the directional flow of data between the Origin Host and the Impacted Host — Inbound, Outbound, Internal, External, or Unknown.	N/A	N/A
Severity	The vendor's view of the severity of the log.	<severity>	\w+
Vendor Message ID	Specific vendor for the log used to describe a type of event.	<vmid>	\w+
Vendor Info †	Description of a specific vendor log or event identifier for the log. Human readable elaboration that directly correlates to the VMID.	<vendorinfo>	\w+
MPE Rule Name	Name of rule that matched, assigned on rule creation.	N/A	N/A
Threat Name †	The name of a threat described in the log message (e.g., malware, exploit name, signature name). Do not overload with Policy.	<threatname>	\w+
Threat ID †	ID number or unique identifier of a threat. Note that CVE is stored separately.	<threatid>	\w+
CVE †	CVE ID (i.e., CVE-1999-0003) from vulnerability scan data.	<cve>	\w+

Host Tab

Display Field	Description	Tag(s)	Default Regex
Host (Origin)	Origin host derived from Origin IP Address and/or Origin Hostname.	N/A	N/A
Host (Impacted)	Impacted host derived from Impacted IP Address and/or Impacted Hostname.	N/A	N/A
MAC Address (Origin)	The MAC address from which activity originated (i.e., attacker, client).	<smac>	(\w{2}(:|-)?){6}
MAC Address (Impacted)	The MAC address that was affected by the activity (i.e., target, server).	<dmac>	(\w{2}(:|-)?){6}
Interface (Origin)	The network port/interface from which the activity originated (i.e., attacker, client).	<sinterface>	\w+
Interface (Impacted)	The network port/interface that was affected by the activity (i.e., target, server).	<dinterface>	\w+
IP Address (Origin)	The IP address from which activity originated (i.e., attacker, client).	<sip> (parses IPv4 and IPv6)	((?<sipv4>(?<sipv4>1??(1??\d{1,2}|2[0-4]\d|25[0-5])\.(1??\d{1,2}|2[0-4]\d|25[0-5])\.(1??\d{1,2}|2[0-4]\d|25[0-5])\.(1??\d{1,2}|2[0-4]\d|25[0-5]))))|(?<sipv6>(?<sipv6>1??((?:(?:[0-9A-Fa-f]{1,4}:){7}[0-9A-Fa-f]{1,4}|(?=(?:[0-9A-Fa-f]{1,4}:){0,7}[0-9A-Fa-f]{1,4}\z)|(([0-9A-Fa-f]{1,4}:){1,7}|:)((:[0-9A-Fa-f]{1,4}){1,7}|:))))))

Display Field	Description	Tag(s)	Default Regex
IP Address (Impacted)	The IP address that was affected by the activity (i.e., target, server).	<dip> (parses IPv4 and IPv6)	((?<dipv4>(?<dipv4>1??(1??\d{1,2}|2[0-4]\d|25[0-5])\.(1??\d{1,2}|2[0-4]\d|25[0-5])\.(1??\d{1,2}|2[0-4]\d|25[0-5])\.(1??\d{1,2}|2[0-4]\d|25[0-5])))|(?<dipv6>(?<dipv6>1??((?:(?:[0-9A-Fa-f]{1,4}:){7}[0-9A-Fa-f]{1,4}|(?=(?:[0-9A-Fa-f]{1,4}:){0,7}[0-9A-Fa-f]{1,4}\z)|(([0-9A-Fa-f]{1,4}:){1,7}|:)((:[0-9A-Fa-f]{1,4}){1,7}|:))))))
NAT IP Address (Origin)	The Network Address Translated (NAT) IP address from which activity originated (i.e., attacker, client).	<snatip>	Same as IP Origin (<sip>)
NAT IP Address (Impacted)	The Network Address Translated (NAT) IP address that was affected by the activity (i.e., target, server).	<dnatip>	Same as IP Impacted (<dip>)
Hostname (Origin)	The hostname from which activity originated (i.e., attacker, client).	<sname> (or DNS resolved from IP)	([^\s\.]+\.?)+
Hostname (Impacted)	The hostname that was affected by the activity (i.e., target, server).	<dname> (or DNS resolved from IP)	([^\s\.]+\.?)+
Known Host (Origin)	A value determined by mapping parsed origin host identifiers, such as IP address or hostname, to a LogRhythm host record.	N/A	N/A
Known Host (Impacted)	A value determined by mapping parsed impacted host identifiers, such as IP address or hostname, to a LogRhythm host record.	N/A	N/A
Serial Number †	The hardware or software serial number in a log message. This value should be a permanent unique identifier.	<serialnumber>	\w+

Identity Tab

Display Field	Description	Tag(s)	Default Regex
User (Origin)	The originating user or system account of the activity reported in the log.	<login>	\w+
User (Impacted)	The user or system account impacted by activity reported in the log.	<account>	\w+
Sender	The sender of an email or the "caller number" for a VOIP log. This value must relate to a specific user or unique address in the case of a phone call or email.	<sender>	[^\s]+@[^\s]+
Recipient	The recipient of an email or the dialed number for a VOIP log.	<recipient>	[^\s]+@[^\s]+
Group	The user group or role impacted by activity reported in the log. Do not use for entity group (zone or domain).	<group>	\w+

Location Tab

Display Field	Description	Tag(s)	Default Regex
Entity (Origin)	A value determined based on the origin host's assigned entity.	N/A	N/A
Entity (Impacted)	A value determined based on the impacted host's assigned entity.	N/A	N/A
Zone (Origin)	A value determined based on the zone of the origin host — Internal, External, DMZ, or Unknown.	N/A	N/A
Zone (Impacted)	A value determined based on the zone of the impacted host — Internal, External, DMZ, or Unknown.	N/A	N/A
Location (Origin)	A value determined by resolving the parsed origin IP address against a Geo-IP database.	N/A	N/A
Location (Impacted)	A value determined by resolving the parsed impacted IP address against a Geo-IP database.	N/A	N/A
Country (Origin)	The country in which the determined origin location exists.	N/A	N/A
Country (Impacted)	The country in which the determined impacted location exists.	N/A	N/A

Log Tab

Display Field	Description	Tag(s)	Default Regex
Log Date	Timestamp when the log was generated or received, corrected to UTC.	N/A	N/A
Log Count	The number of identical log messages received.	N/A	N/A
Log Source Entity	The entity to which the log source belongs.	N/A	N/A
Log Source Type	The device or application type from which a log was received.	N/A	N/A
Log Source Host	The origin host from which the log was received.	N/A	N/A
Log Source	The assigned name of a log source.	N/A	N/A
Log Sequence Number	The sequence in which a log was collected, generated by the Agent.	N/A	N/A
Log Message	The raw log message.	N/A	N/A
First Log Date	Timestamp when the first identical log message was received.	N/A	N/A
Last Log Date	Timestamp when the last identical log message was received.	N/A	N/A

Network Tab

Display Field	Description	Tag(s)	Default Regex
Network (Origin)	A value determined by mapping the origin IP address to a LogRhythm network record.	N/A	N/A
Network (Impacted)	A value determined by mapping the impacted IP address to a LogRhythm network record.	N/A	
Domain (Impacted) †	The Windows or DNS domain name referenced or impacted by activity reported in the log.	<domain>	\w+
Domain (Origin) †	The Windows or DNS domain where the logged activity originated.	<domainorigin>	\w+
Protocol	The IANA protocol name or number.	<protnum>, <protname>	1??\d{1,2}|2[0-4]\d|25[0-5] \w+
TCP/UDP Port (Origin)	The port from which activity originated (i.e., client, attacker port).	<sport>	\d+
TCP/UDP Port (Impacted)	The port to which activity was targeted (i.e., server, target port).	<dport>	\d+
NAT TCP/UDP Port (Origin)	The Network Address Translated (NAT) port from which activity originated (i.e., client, attacker port).	<snatport>	\d+
NAT TCP/UDP Port (Impacted)	The Network Address Translated (NAT) port to which activity was targeted (i.e., server, target port).	<dnatport>	\d+

Special Sub-Rule Tags (Tag1-Tag5)

Five additional tags are available for identifying data in the log specifically for sub-rules. These tags do not parse text into metadata fields, they are only used to identify portions of the log message that should be used in the development of sub-rules.

Tag	Field Type	Default Regex
<tag1>	Text	.*
<tag2>	Text	.*
<tag3>	Text	.*
<tag4>	Text	.*
<tag5>	Text	.*

Best Practices and Working with Rules

This section contains several best practices for modifying MPE rules.

Override Default Regex

You can override the default regex if the source data does not conform to the default pattern. You only need to override the default regex when the default:

- will not properly parse the correct data out of the log message.
- is not the optimal regex from a performance perspective.

To override the default regex, the following syntax should be used.

(?<[tagname]>[regex])

For example, suppose your regex needs to match file names with a specific extension such as the sample log message below:

User john.doe opened AnnualReport.pdf

If the base-rule was written as:

User <login> opened <object>

The value parsed for login would john and the value for object would be AnnualReport. This is due to the fact that a period is not a word character and the default regex of "\w+" would only match up to the period. Instead, the default expressions should be overridden and the base-rule should be:

User (?<login>\w+\.?\w) opened (?<object>\w+\.pdf)*

Now, the base-rule will parse anything for login starting with a word character that optionally contains a period followed be additional word characters.

Do not override/overload <sip>, <dip>, <snatip>, or <dnatip>

Rule Names

When naming a rule, follow these accepted best practices:

- When the matching log message contains a vendor message ID such as an event ID in Windows Event Logs, it is good to include the ID in the name of the rule. This makes searching for the rule easier and also makes the rule more descriptive of the log that it matches.
- If the rule matches a log from a logging system that generates logs for a wide variety of services, such as the Windows Application Event Log, the service that generated the log message should be included in the rule name.
- All rule names should contain a brief description of the action described by the log.
  For example: EVID 528 : Failed Authentication : Bad Username or Password

Common Event Names

Using the Rule Builder Common Event Browser, you can view the complete list of more than 40,000 common events. Use the predefined common events wherever possible. If you need to create a new common event, use the following guidelines:

- Common events should be generically named so that they can be re-used for a wide variety of devices. For example, if a common event is being created for a log message that describes a successful connection to an FTP server, the common event should be named so that the FTP server type is irrelevant.
  - Good Name: FTP Connection Succeeded
  - Bad Name: Gene6 FTP Connection Succeeded
- Common event names should always have the first letter of each word capitalized to make viewing common events in analysis tools more consistent.

Regular Expression Characters and Practices

This section provides an overview of regular expression characters and recommend practices.

Match Characters

Notation	Characters Matched	Example
\d	Any digit from 0 to 9	\d\d\d matches 101 but not 10a
\D	Any character that is not a numeric digit (0 to 9)	\D\D\D matches abc but not 101
\w	Any word character, for example, a-z, A-Z, 0-9, and the underscore character _ (will also match Unicode based word characters from non-Latin alphabets and scripts)	\w\w\w matches abc but not &@#
\W	Any non-word character	\W\W\W matches $#! but not abc
\s	Matches any whitespace character	\s\s\s matches (three spaces) but not abc
\S	Matches any non-whitespace character	\S\S\S matches a1_ but not (three spaces)
.	Matches any character	. matches any character except line breaks
[ ]	Any character between the square brackets	[abc] matches a or b or c but no other character
[^]	Matches any character except the characters appearing after the ^ and before the]	[^abc] matches def but not abc

Repetition Characters

Notation	Characters Matched	Example
{n}	Matches n of the previous item	\w{4} matches AAAA but not A
{n, }	Matches n or more of the previous item	\w{4, } matches AAAAAA but not A
{n,m}	Matches at least n and at most m of the previous item if n is 0 that makes the character optional ({,9})	A{2,3} matches AA and AAA but not A or AAAA
?	Matches the previous item 0 or 1 times	A? matches A or nothing but not AA
+	Matches the previous item 1 or more times	A+ matches A, AA, AAA but not nothing
*****	Matches the previous item 0 or more times	A* matches nothing, A or any number of A characters

Positional Characters

Notation	Description
^	The following pattern must be at the start of the string, or for a multi-line string, at the beginning of a line. For multi-line text (string containing a carriage return), the multi-line flag option needs to be set.
$	The preceding pattern must be at the end of the string, or for a multi-line string, at the end of a line.
\A	The preceding pattern must be at the start of the string; the multi-line flag is ignored.
\Z	The preceding pattern must be at the end of the string; the multi-line pattern is ignored.
\b	Matches a word boundary, essentially the point between a word character (a-z, A-Z, 0-9, _) and a non-word character (the start of a word).
\B	Matches a position that is not a word boundary (not the start of a word).

Grouping

Notation	Characters Matched	Example
()?	Matches the pattern inside the brackets 0 or 1 times.	(Error)? Matches Error or nothing
()+	Matches the pattern inside the brackets 1 or more times.	(\w+\s)+ Matches AA AA
()*	Matches the pattern inside the brackets 0 or more times.	(\w+\s)* Matches nothing or AA AA

The Non-Greedy Qualifier (?)

The non-greedy qualifier is a question mark (?) following a repetition character (*+?). The non-greedy qualifier is used to tell the regex engine that it should stop matching the current match as soon as the next match criterion is met. The non-greedy qualifier is used in combination with a repetition qualifier in order to create a non-greedy match. The non-greedy qualifier improves performance when you want to match any text value up to a specific text value where the specific text value can be uniquely specified within the regex.

For example, suppose your regex needs to match the following log:

> 02/28/2007 16:55:22 MsgID=1590 : Failed authentication for user john.doe user account locked out

If you use the following regex, incorrect values will be parsed for the login field due to the fact that user occurs twice in the log message. Using this regex will cause "account" to be parsed into the login field.

> MsgID=1590.*user (?<login>\w+\.?\w*)

This is because ".*" will match everything to the end of the log message. When the regex engine reaches the end of the log message it will begin looking backwards in the log message for the next match. As soon as it finds the last occurrence of "user" it will match for that portion of the log message. Since the specified regex for "login" will match account, it will use that match and continue.

To make the regex take the first occurrence of the next match you use the non-greedy qualifier. The following regex will parse the correct value into the login field because it will stop the previous match (.*) as soon as "user" is encountered.

> MsgID=1590.*?user (?<login>\w+\.?\w*)

Reserved Characters

The regex engine used by LogRhythm has 12 reserved characters that have special meaning. If any of these characters need to be used as a literal character they will need to be escaped using the backslash (\) character, otherwise known as the escape character. The reserved characters are:

- The opening square bracket **[**
- The opening round bracket **(**
- The closing round bracket **)**
- The backslash ****
- The caret **^**
- The dollar sign **$**
- The period **.**
- The vertical bar or pipe symbol **|**
- The question mark **?**
- The asterisk or Kleene star *****
- The plus sign **+**
- The opening curly bracket **{**
- The closing curly bracket **}**

The following regex, which is meant to match any IPv4 address (a.b.c.d), is a simple example of how to escape reserved characters:

> \d+\.\d+\.\d+\.\d+

As you can see each of the periods of the IP address are escaped meaning the regex engine will look for the actual period (.) character in the string instead of looking for any character. Without the escape slash, the period refers to any character, which would radically change the meaning of the expression.

Other Special Characters

Other special characters that match special cases and cannot normally be typed into a regular expression:

Special Character	Description
\n	Matches newline
\r	Matches carriage return
\t	Matches tab
\nnn	Matches ASCII character specified by octal number nnn Ex: \103 matches C
\xnn	Matches ASCII character specified by hexadecimal number nn Ex: \x43 matches C
\unnnn	Matches the Unicode character specified by the four hexadecimal digits replaced by nnnn
\cD	Matches a control character Ex: \cD matches end of transmission

Regex Recommended Practices

The following are some recommended practices for regex development. All regex examples use the following log.

02/28/2007 16:55:22 MsgID=1590 : Failed authentication for user "any.user" user account locked out

Name	Recommended Pattern	Description
Negative Character Class	"[^"]*" for double quote delimiters, ,[^,]*, for comma delimiters, or |[^|]*| for pipe delimiters. Can be used for any type of predictable delimiter.	Use negative character classes in log messages with clear delimiters, such as quotation marks, commas, or pipes. This will match any character that is not the delimiter. This can greatly improve parsing performance vs a more generic match, such as .*?. Example: MsgID=1589.*?user\s"(?<account>[^"]*)"
Non-greedy Match	.*?	If you need to match any characters until a specific set of characters appears, use this pattern. Example: MsgID=1590.*?user\s"(?<account>[^"]*)"
Overloading Map Tags	(?<[map tag]>[regex])	Map tags should almost always be overloaded. The default regex for map tags is .* which will match everything to the end of the log. Example: MsgID=1590.*?user\s"(?<account>[^"]*)"\s(?<tag1>.*?)$
Preceding and trailing values	N/A	Always match as much constant text as possible. The more information the regex has to evaluate, the faster it will be at identifying non-matching logs. For any parsed field, it is best to search for a constant value before and after the value being parsed. Example: MsgID=1590.*?user\s"(?<account>[^"]*)"\s(?<tag1>.*?)$
Look Aheads	(?=[regex])[regex] (?![regex])[regex]	Positive and negative look ahead allows for an initial check in the regex to see if a case is satisfied in the log messages. These are useful for finding a value later in a log to reduce extraneous processing for non-matching logs. **Do not** use for values that appear very early in a log message, such as just past a Syslog header. Look ahead is more costly than regular expressions if the match is always found early. Example: (?=.*?match contains this phrase)\s<sip>\s
Multiline character match pattern	[\r\n]	Using a character class containing both \r (return) and \n (newline) allows for either multiline character to appear as well as both in either order. Some log messages vary in the order of these new lines and others only contain a newline.
Narrow Character Classes	[a-z0-9_]+	The shorthand character class \w matches Latin alphabet characters, Hindu-Arabic numerals (0-9), underscores as well other scripts supported in Unicode. Narrowing the match to only the relevant character set will yield better performance.