



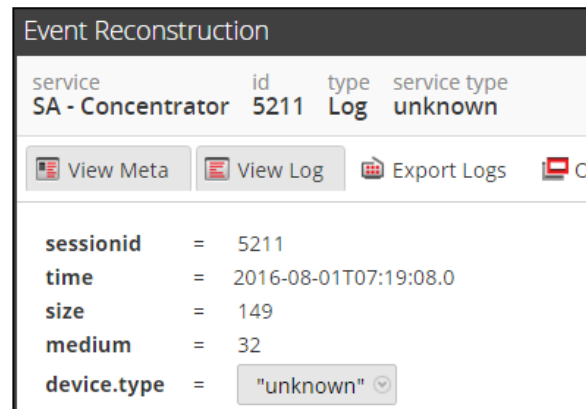
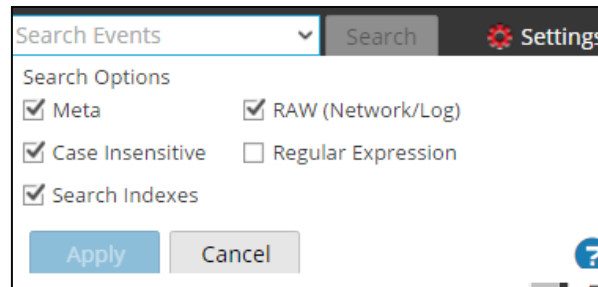
# SECURITY ANALYTICS 10.6

RAW LOG SEARCH IMPROVEMENTS

RSA

# What is an unknown log message?

- ▶ Security Analytics applies all the configured log parsers for incoming logs to:
  - ▶ Identify the device type
  - ▶ Extract the relevant meta data
- ▶ If no log parser recognizes the message, this will be «unknown»:
  - ▶ Cannot be queried in Investigator
  - ▶ Can be retrieved in Events with a free-text search



# Raw log search: before and after

## ▶ Security Analytics <10.6:

- ▶ The log decoder retrieves from the storage **every single raw log** in the given timeframe
- ▶ The content is searched for the provided value (brute-force)
- ▶ If there is a match the raw log is presented

## ▶ When executed on a large timeframe may be slow

## ▶ Security Analytics 10.6:

- ▶ For every unknown log, the decoder tokenizes the content
- ▶ By default the first 5 letters of every word are extracted and a meta is generated
- ▶ Upon a search, the decoder goes through these «word» meta first
- ▶ ONLY if there is a match the raw log is retrieved and the full search is executed

## ▶ Resulting in much better and optimized performance

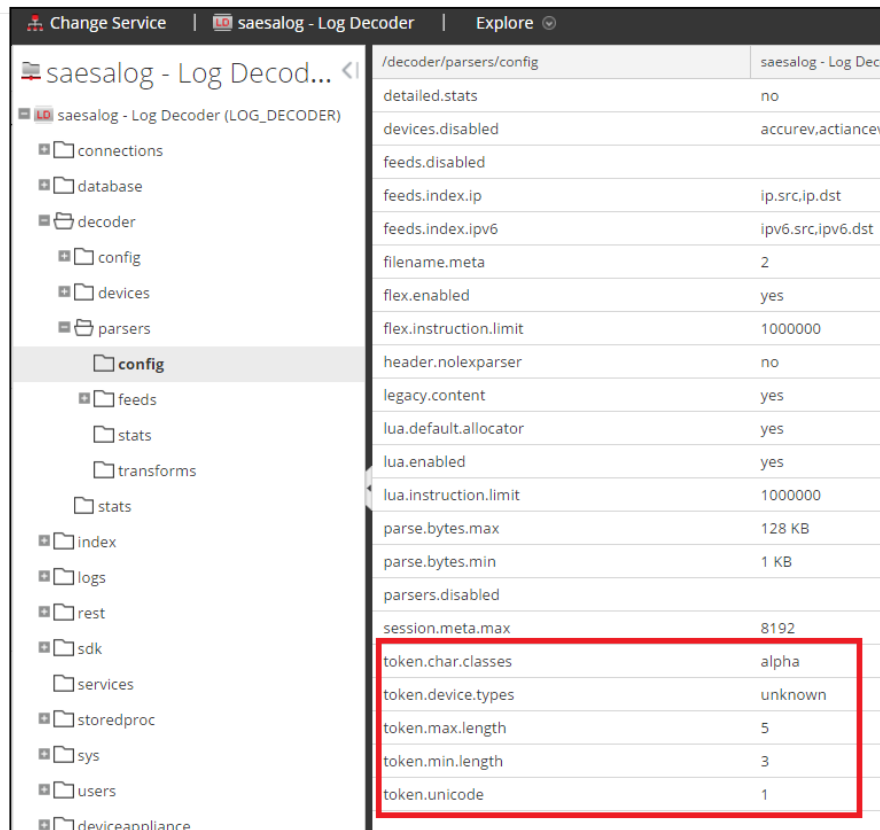
# Unknown log messages “tokenized”

- The Log Decoder tokenizes unparsed logs and creates meta data that form a full-text index
- When searching, this meta will be used first, before brute-forcing all the raw logs

| device.type = 'unknown'   did = 'logdecoder' Cancel |                     |            |             |  |
|---|---------------------|------------|-------------|--|
| <input type="checkbox"/>                            | Event Time          | Event Type | Event Theme | Details  |
| <input type="checkbox"/>                            |                     |            |             | <div>sessionid : 7</div> <div>device.ip : 127.0.0.1</div> <div>medium : 32</div> <div>device.type : unknown</div> <div>word : audit</div> <div>word : runti</div> <div>word : com</div> <div>word : rsa</div> <div>word : ims</div> <div>word : authn</div> <div>word : impl</div> <div>word : authe</div> <div>word : error</div> <div>word : login</div> <div>word : event</div> <div>word : fail</div> <div>word : metho</div> <div>word : faile</div> <div>word : sxecw</div> <div>word : admin</div> <div>word : passw</div> <div>word : ldap</div> <div>did : logdecoder</div> <div>rid : 7</div> <div>Hide Additional Meta View Details</div> |
| <input type="checkbox"/>                            | 2016-02-15T12:58:38 | Log        |             | 564 bytes  |

# Customizing the tokenizer

- ▶ The way tokens are generated can be customized under `/parsers/config`
- ▶ Before applying any change, please consider the impact it could have in terms of storage utilization



The screenshot shows the 'saesalog - Log Decoder' interface. On the left is a tree view of the configuration structure, with 'config' selected under 'parsers'. On the right is a table of configuration parameters. A red rectangle highlights the 'token' section of the configuration.

| Parameter              | Value             |
|------------------------|-------------------|
| detailed.stats         | no                |
| devices.disabled       | accurev,actiancev |
| feeds.disabled         |                   |
| feeds.index.ip         | ip.src,ip.dst     |
| feeds.index.ipv6       | ipv6.src,ipv6.dst |
| filename.meta          | 2                 |
| flex.enabled           | yes               |
| flex.instruction.limit | 1000000           |
| header.noexparser      | no                |
| legacy.content         | yes               |
| lua.default allocator  | yes               |
| lua.enabled            | yes               |
| lua.instruction.limit  | 1000000           |
| parse.bytes.max        | 128 KB            |
| parse.bytes.min        | 1 KB              |
| parsers.disabled       |                   |
| session.meta.max       | 8192              |
| token.char.classes     | alpha             |
| token.device.types     | unknown           |
| token.max.length       | 5                 |
| token.min.length       | 3                 |
| token.unicode          | 1                 |

