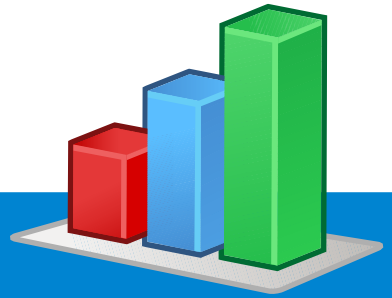
RMLL
BEAUVAIS 2015
Destination libre

Analyse OpenLDAP™ logs

with ElasticSearch, Logstash and Kibana



Savoir-faire LINUX®

Clément OUDOT
@clementoudot

LLNG

LSC
LDAP Synchronization Connector

LTB Project

Savoir-faire LINUX

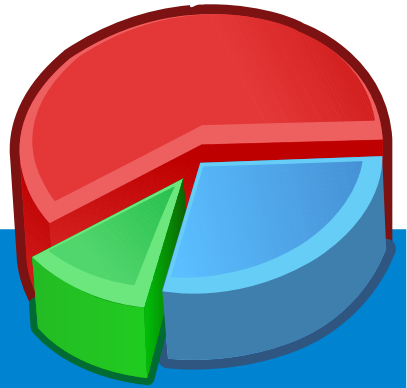
- Founded in 1999
- >100 persons
- Montréal, Quebec City, Ottawa, Paris
- ISO 9001:2004 / ISO 14001:2008
- contact@savoirfairelinux.com

Microsoft Certified Partner
redhat. Advanced Business Partner
odoo Gold Partner
LIFERAY Community Excellence 2014 • 2013 2012 • 2011
BlueMind Premium Partner
Alfresco Gold Partner
ORACLE PARTNER
talend Gold Partner
Acquia

Summary

1. The ELK stack
2. Format of OpenLDAP logs
3. OpenLDAP with ELK

1 The ELK Stack



ELASTICSEARCH

LOGTASH

KIBANA

Elasticsearch

- Written in Java
- License Apache 2.0
- Based on Lucene
- JSON
- REST
- Distributed
- Index and fields

```
clement@ader-sfl:~$ curl http://localhost:9200
{
  "status" : 200,
  "name" : "Mahkizmo",
  "cluster_name" : "elasticsearch",
  "version" : {
    "number" : "1.6.0",
    "build_hash" : "cdd3ac4dde4f69524ec0a14de3828cb95bbb86d0",
    "build_timestamp" : "2015-06-09T13:36:34Z",
    "build_snapshot" : false,
    "lucene_version" : "4.10.4"
  },
  "tagline" : "You Know, for Search"
}
```

Elasticsearch http://localhost:9200/ Se connecter **elasticsearch** **Santé du cluster: yellow (20 40)** Info

Aperçu | Index | Navigateur | Recherche Structurée [+] | Autres requêtes [+]

Navigateur Rafraîchir

Tous les index

Index

- kibana-int
- logstash-2015.06.26
- logstash-2015.06.30
- logstash-2015.07.01

Types

- _default_
- dashboard
- openldap_log_file

Champs

- @timestamp
- @version
- add_dn
- bind_dn
- bind_mech
- bind_method
- bind_ssf
- connection
- dashboard
- data
- deferring_op
- del_dn
- dst_ip
- dst_port
- error_code
- fd_number
- group
- host
- index_error_attribute_name
- index_error_filter_type
- location
- logsource
- message
- mod_attr
- mod_dn
- nentries

Recherche sur 20 des 20 shards. 7430890 résultats. 0.035 secondes

_index	_type	_id	_score ▼	message	@version	@timestamp
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUdI	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37743 op=413 SEARCH RESULT tag=101 err=0 nentries=50 text=	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUdN	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37749 op=104 SRCH attr=ou	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUdS	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37746 op=151 SEARCH RESULT tag=101 err=0 nentries=50 text=	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUdX	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37743 op=420 SEARCH RESULT tag=101 err=0 nentries=4 text=	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUdb	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37742 op=448 SRCH base="ou=groups,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUdg	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37749 op=105 SRCH base="ou=groups,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUdl	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37746 op=152 SRCH attr=ou	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUdq	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37744 op=324 SEARCH RESULT tag=101 err=0 nentries=4 text=	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUdv	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37741 op=493 SRCH attr=ou	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUd0	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37750 op=28 SEARCH RESULT tag=101 err=0 nentries=4 text=	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUd5	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37750 op=29 SRCH base="ou=groups,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUeA	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37745 op=315 SRCH attr=ou	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUeF	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37748 op=111 SEARCH RESULT tag=101 err=0 nentries=4 text=	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUeK	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37746 op=153 SRCH attr=uid	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUeP	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37746 op=153 SEARCH RESULT tag=101 err=0 nentries=50 text=	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUeU	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37747 op=122 SEARCH RESULT tag=101 err=0 nentries=4 text=	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUeZ	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37745 op=318 SRCH attr=ou	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUed	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37741 op=495 SRCH base="ou=groups,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUei	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37746 op=154 SRCH base="ou=groups,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUen	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37741 op=496 SRCH base="ou=users,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUes	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37743 op=423 SEARCH RESULT tag=101 err=0 nentries=50 text=	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUex	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37741 op=497 SRCH base="ou=users,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUe2	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37746 op=156 SRCH attr=uid	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUe7	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37743 op=425 SRCH base="ou=groups,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUe-	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37741 op=498 SRCH base="ou=groups,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUe_	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37741 op=498 SRCH attr=ou	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUfC	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37745 op=320 SRCH base="ou=users,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUfH	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37747 op=125 SRCH base="ou=groups,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUfM	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37745 op=321 SRCH attr=uid	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUfR	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37747 op=126 SRCH attr=uid	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUfW	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37741 op=499 SRCH attr=uid	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUfa	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37745 op=322 SRCH attr=uid	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUff	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37741 op=500 SRCH base="ou=groups,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUfk	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37747 op=127 SEARCH RESULT tag=101 err=0 nentries=50 text=	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUfp	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37746 op=158 SRCH base="ou=groups,dc=example,dc=com" scope=1 deref=0 filter="(objectClass=*)"	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUfu	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37745 op=324 SEARCH RESULT tag=101 err=0 nentries=4 text=	1	2015-06-26T1
logstash-2015.06.26	openldap_log_file	AU4wUbvQTI6hfpoDwUfz	1	Jun 26 16:22:31 ader-sfl slapd[1386]: conn=37741 fd=25 closed	1	2015-06-26T1

Logstash

- JRuby
- License Apache 2.0
- Multiple inputs (file, syslog, ...)
- Multiple outputs (console, redis, elasticsearch, ...)
- Filters

```
root@ader-sfl:~# /opt/logstash/bin/logstash -e 'input { stdin
{ } } output { stdout { codec => rubydebug } }'

Logstash startup completed

RMLL 2015
{
       "message" => "RMLL 2015",
      "@version" => "1",
    "@timestamp" => "2015-07-02T08:29:09.363Z",
          "host" => "ader-sfl"
}
```

Grok

- Grok allows to parse message and store content in fields
- Grok comes with standard patterns (Syslog, Apache, ...)
- You can also define you own patterns
- Grok debugger: http://grokdebug.herokuapp.com/

Example of grok patterns

```
# Log formats

SYSLOGBASE %{SYSLOGTIMESTAMP:timestamp} (?:%{SYSLOGFACILITY} )?%
{SYSLOGHOST:logsource} %{SYSLOGPROG}:

COMMONAPACHELOG %{IPORHOST:clientip} %{USER:ident} %{USER:auth} \
[%{HTTPDATE:timestamp}\] "(?:%{WORD:verb} %{NOTSPACE:request}(?:
HTTP/%{NUMBER:httpversion})?|%{DATA:rawrequest})" %
{NUMBER:response} (?:%{NUMBER:bytes}|-)

COMBINEDAPACHELOG %{COMMONAPACHELOG} %{QS:referrer} %{QS:agent}
```

Using grok in logstash

```
55.3.244.1 GET /index.html 15824 0.043
```

```
input {
  file {
    path => "/var/log/http.log"
  }
}
filter {
  grok {
    match => { "message" => "%{IP:client} %{WORD:method} %
{URIPATHPARAM:request} %{NUMBER:bytes} %{NUMBER:duration}" }
  }
}
```

Kibana

- Javascript
- License Apache 2.0
- Connect to elasticsearch
- Lucene queries
- On the fly graphics

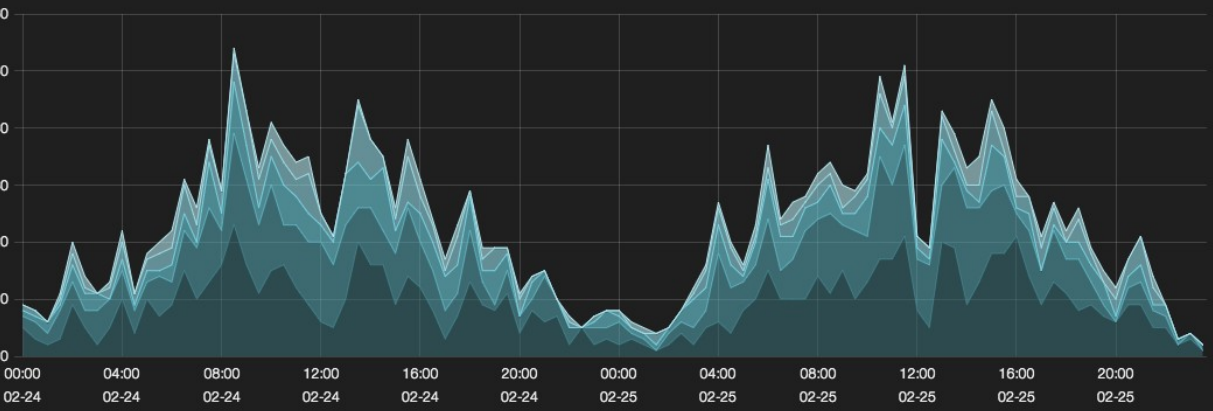
Logstash Search

2 days ago to 3 minutes ago

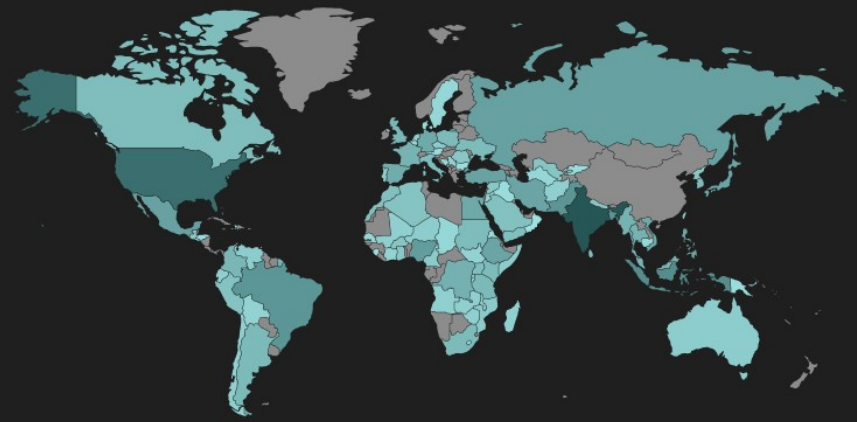
bytes:[0 TO 4000000] AND @tags:success

EVENTS OVER TIME

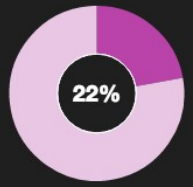
Zoom Out | html (ext) (888) | php (ext) (659) | png (ext) (317) | gif (ext) (229) | css (ext) (121) count per **30m** | (**2214** hits)



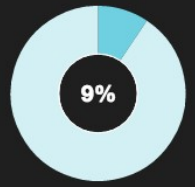
MAP



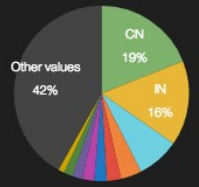
REVENUE



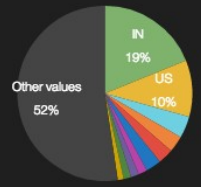
VOLUME



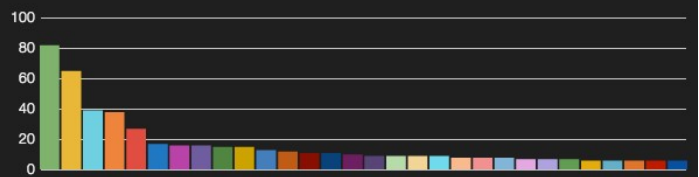
TOP DESTINATIONS



TOP SOURCES



GEO PAIRS



ALL EVENTS

0 to **100** of 500 available for paging

Fields

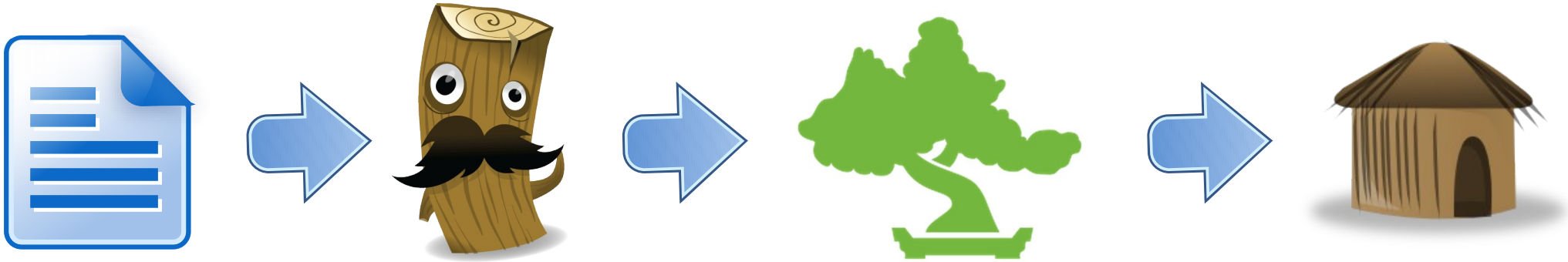
Type to filter...

@message

@tags

@tags	geo.srcdst	extension	clientip	bytes	id	phpmemory	response
success,security	MY:VN	html	167.12.22.189	8540	1066		200
success,info	IT:MM	png	164.87.170.73	2045	1903		200
success,info	AR:ES	html	222.23.102.238	1801	1133		200
success,info	IN:DZ	html	138.226.66.81	7029	1801		200

Working together



Format of OpenLDAP logs

Logging in OpenLDAP

- Relies on syslog
- Log levels are cumulative
- Log level is configured in olcLogLevel parameter in cn=config

```
dn: cn=config
objectClass: olcGlobal
cn: config
olcLogLevel: stats
```

Level	Keyword	Description
-1	any	enable all debugging
0		no debugging
1	(0x1 trace)	trace function calls
2	(0x2 packets)	debug packet handling
4	(0x4 args)	heavy trace debugging
8	(0x8 conns)	connection management
16	(0x10 BER)	print out packets sent and received
32	(0x20 filter)	search filter processing
64	(0x40 config)	configuration processing
128	(0x80 ACL)	access control list processing
256	(0x100 stats)	stats log connections/operations/results
512	(0x200 stats2)	stats log entries sent
1024	(0x400 shell)	print communication with shell backends
2048	(0x800 parse)	print entry parsing debugging
16384	(0x4000 sync)	syncrepl consumer processing
32768	(0x8000 none)	only messages that get logged whatever log level is set

Information in the “stats” loglevel

- Connections (opening, closing, TLS, …)
- Operations and results
- Errors and warnings:
  - Missing index
  - Connection lost
  - Password policy messages
  - …

Connections

```
Jul  1 15:27:39 ader-sfl slapd[1377]: conn=162707 fd=12 ACCEPT
from IP=127.0.0.1:44678 (IP=0.0.0.0:389)

...

Jul  1 15:27:39 ader-sfl slapd[1377]: conn=162707 fd=12 closed
```

Operation: BIND and UNBIND

```
Jul  1 15:27:39 ader-sfl slapd[1377]: conn=162709 op=3 BIND
dn="cn=admin,dc=example,dc=com" method=128
Jul  1 15:27:39 ader-sfl slapd[1377]: conn=162709 op=3 BIND
dn="cn=admin,dc=example,dc=com" mech=SIMPLE ssf=0
Jul  1 15:27:39 ader-sfl slapd[1377]: conn=162709 op=3 RESULT
tag=97 err=0 text=
```

```
Jul  1 15:27:39 ader-sfl slapd[1377]: conn=162708 op=7 UNBIND
```

Operation: SEARCH

```
Jul  1 15:27:39 ader-sfl slapd[1377]: conn=162709 op=5 SRCH
base="ou=users,dc=example,dc=com" scope=1 deref=0
filter="(objectClass=*)"
Jul  1 15:27:39 ader-sfl slapd[1377]: conn=162709 op=5 SRCH
attr=uid
Jul  1 15:27:39 ader-sfl slapd[1377]: conn=162709 op=5 SEARCH
RESULT tag=101 err=0 nentries=50 text=
```

Operation: MODIFY

```
Jul  1 16:10:36 ader-sfl slapd[1377]: conn=162711 op=17 MOD
dn="uid=user1,ou=users,dc=example,dc=com"
Jul  1 16:10:36 ader-sfl slapd[1377]: conn=162711 op=17 MOD
attr=userPassword
Jul  1 16:10:36 ader-sfl slapd[1377]: conn=162711 op=17 RESULT
tag=103 err=0 text=
```

Errors and warnings

```
Jul  1 17:18:48 ader-sfl slapd[1377]: conn=162711 fd=12 closed
(connection lost)

Jul  1 09:28:40 ader-sfl slapd[1377]: connection_input: conn=93309
deferring operation: binding

Jul  2 08:41:02 ader-sfl slapd[1377]: <= mdb_equality_candidates:
(objectClass) not indexed

Jul  1 16:23:46 ader-sfl slapd[1377]: ppolicy_bind: Setting
warning for password expiry for
uid=user1,ou=users,dc=example,dc=com = 589 seconds
```

Logfile alternatives

- Overlay accesslog : log events in an LDAP backend
- Overlay auditlog : log events in an LDIF file

3 OpenLDAP with ELK

Configure logstash

- Input:
  - OpenLDAP logfile
  - Syslog
- Filters:
  - Syslog Grok pattern
  - Date
  - OpenLDAP specific log patterns
- Output: elasticsearch

Main grok pattern

```
%{SYSLOGBASE} (?:(?:<= (?:b|m)db_%
{DATA:index_error_filter_type}_candidates: \(%
{WORD:index_error_attribute_name}\) not indexed)|(?:ppolicy_%
{DATA:ppolicy_op}: %{DATA:ppolicy_data})|(?:connection_input: conn=%
{INT:connection} deferring operation: %{DATA:deferring_op})|
(?:connection_read\(%{INT:fd_number}\): no connection!)|(?:conn=%
{INT:connection} (?:(?:fd=%{INT:fd_number} (?:(?:closed(?: \(connection
lost\)|)))|(?:ACCEPT from IP=%{IP:src_ip}\:%{INT:src_port} \(IP=%
{IP:dst_ip}\:%{INT:dst_port}\))|(?:TLS established tls_ssf=%
{INT:tls_ssf} ssf=%{INT:ssf}))))|(?:op=%{INT:operation_number} (?:(?:(?:
(?:SEARCH )|(?:))RESULT (?:tag=%{INT:tag}|oid=(?:%{DATA:oid}(?:))) err=%
{INT:error_code}(?:(?: nentries=%{INT:nentries})|(?:)) text=(?:(?:%
{DATA:error_text})|(?:)))|(?:%{WORD:operation_name}(?:(?: %{DATA:data})|
(?:))))))))%{SPACE}$
```

Conditional grok pattern

```
  if [operation_name] == "SRCH" {
    grok {
      match => [ "data", "(?:(?:base=\"%{DATA:search_base}\" scope=%
{INT:search_scope} deref=%{INT:search_deref} filter=\"%
{DATA:search_filter}\")|(?:attr=%{DATA:search_attr}))%{SPACE}$" ]
    }
  }

  if [operation_name] == "ADD" {
    grok {
      match => [ "data", "dn=\"%{DATA:add_dn}\"%{SPACE}$" ]
    }
  }
```

Display data in Kibana

- Access to all logged messages
- Query data, for example:
  - Follow a connection: connection=162738
  - Find missing index: _exists_:index_error_attribute_name
- Create dashboards, for example:
  - LDAP operations
  - LDAP error codes

ldap_error_codes

15 minutes ago to a few seconds ago

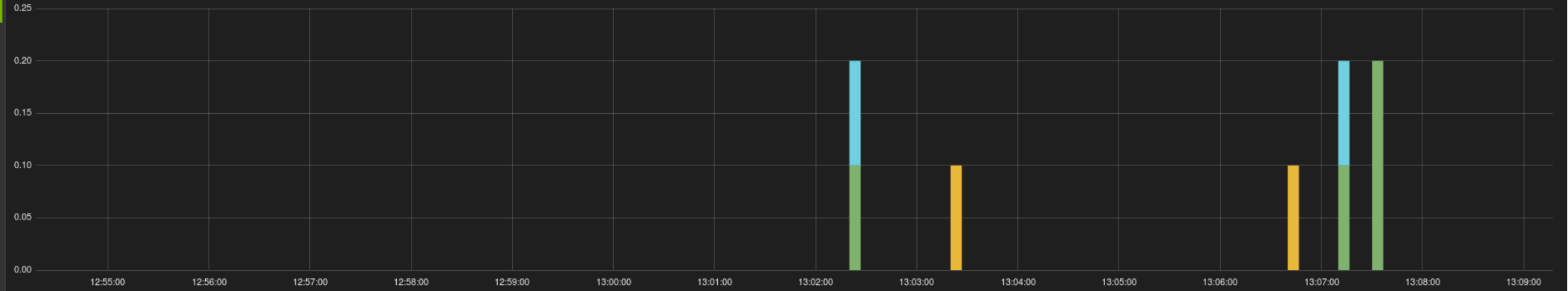
QUERY

error_code:"0" | error_code:"49" | error_code:"32" | error_code:"5" | error_code:"6"

FILTERING

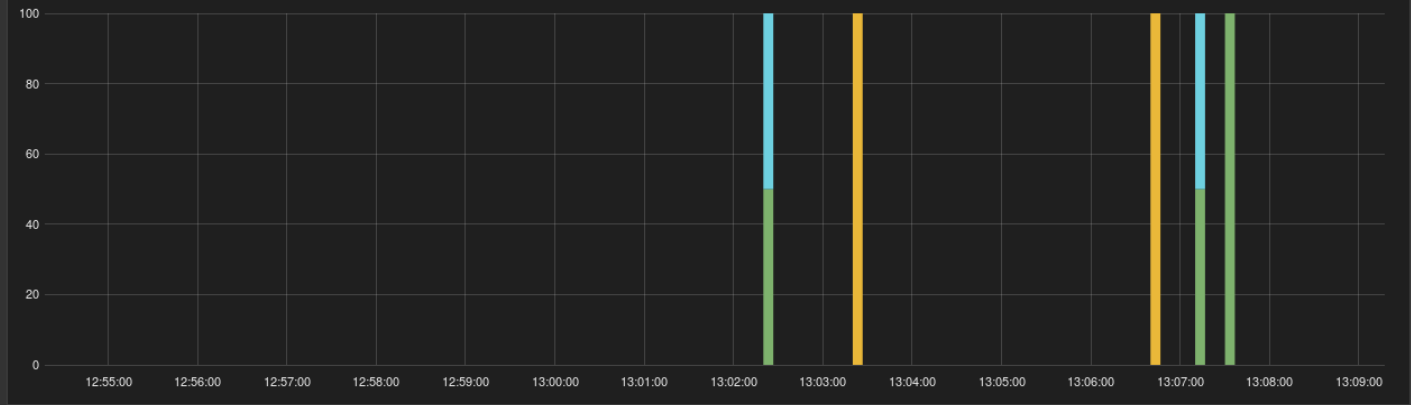
EVENTS OVER TIME

View | Zoom Out | Success (4) | Invalid credentials (2) | No such object (2) | Compare FALSE (0) | Compare TRUE (0) count per **1s** | (**8** hits)



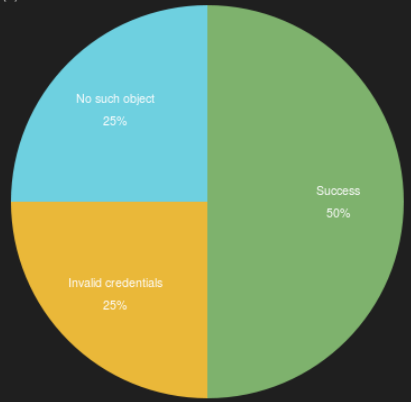
ERROR CODES

View | Zoom Out | Success (4) | Invalid credentials (2) | No such object (2) | Compare FALSE (0) | Compare TRUE (0) count per **10s** | (**8** hits)



ERROR CODES

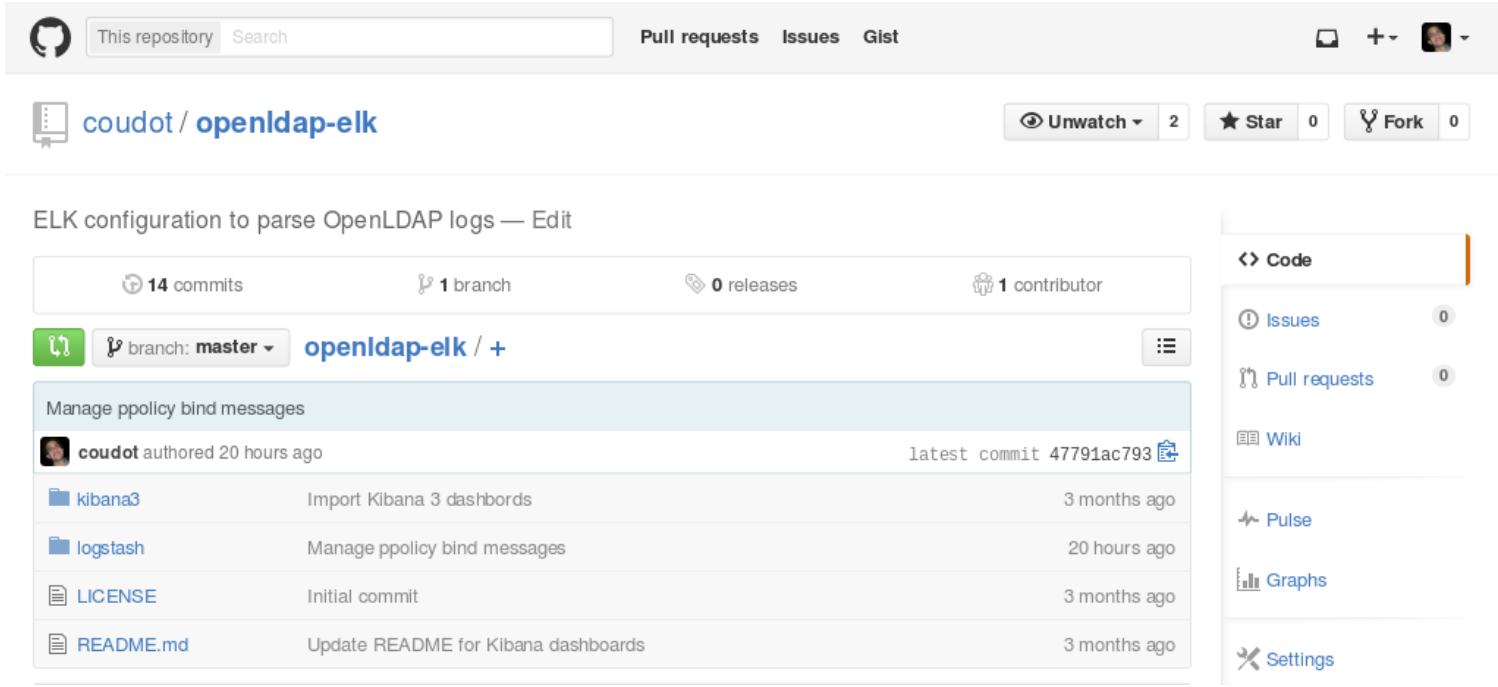
Success (4) | Invalid credentials (2) | No such object (2) | Compare FALSE (0) | Compare TRUE (0)



EVENTS

Use and improve!

https://github.com/coudot/openldap-elk



Seems all clear

Any question?