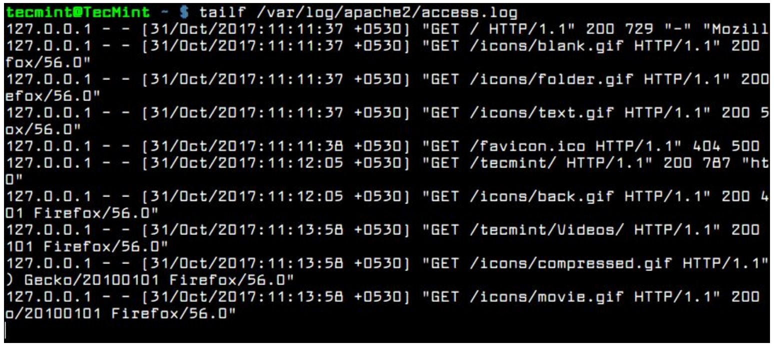
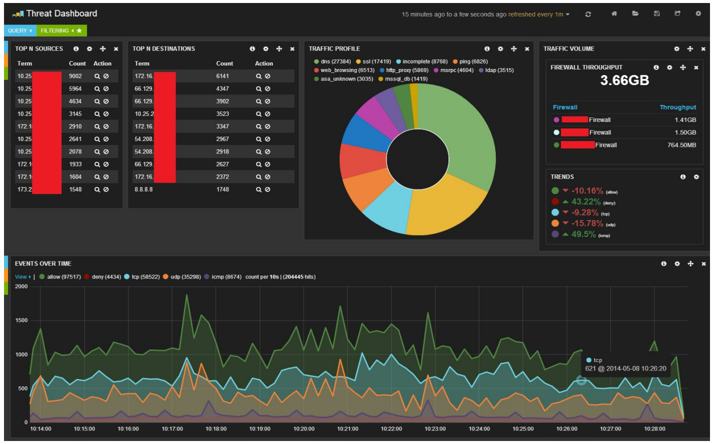


```
tecmint@TecMint ~ $ tailf /var/log/apache2/access.log
127.0.0.1 - - [31/Oct/2017:11:11:37 +0530] "GET / HTTP/1.1" 200 729 "-" "Mozill
127.0.0.1 - - [31/Oct/2017:11:11:37 +0530] "GET /icons/blank.gif HTTP/1.1" 200
fox/56.0"
127.0.0.1 - - [31/Oct/2017:11:11:37 +0530] "GET /icons/folder.gif HTTP/1.1" 200
efox/56.0"
127.0.0.1 - - [31/Oct/2017:11:11:37 +0530] "GET /icons/text.gif HTTP/1.1" 200 5
ox/56.0"
127.0.0.1 - - [31/Oct/2017:11:11:38 +0530] "GET /favicon.ico HTTP/1.1" 404 500
127.0.0.1 - - [31/Oct/2017:11:12:05 +0530] "GET /tecmint/ HTTP/1.1" 200 787 "ht
0"
127.0.0.1 - - [31/Oct/2017:11:12:05 +0530] "GET /icons/back.gif HTTP/1.1" 200 4
01 Firefox/56.0"
127.0.0.1 - - [31/Oct/2017:11:13:58 +0530] "GET /tecmint/Videos/ HTTP/1.1" 200
101 Firefox/56.0"
127.0.0.1 - - [31/Oct/2017:11:13:58 +0530] "GET /icons/compressed.gif HTTP/1.1"
) Gecko/20100101 Firefox/56.0"
127.0.0.1 - - [31/Oct/2017:11:13:58 +0530] "GET /icons/movie.gif HTTP/1.1" 200
o/20100101 Firefox/56.0"
```

=>



Threat Dashboard

15 minutes ago to a few seconds ago refreshed every 1m

TOP N SOURCES

Term	Count	Action
10.25[redacted]	9002	
10.25[redacted]	5964	
10.25[redacted]	4634	
10.25[redacted]	3145	
172.1[redacted]	2910	
10.25[redacted]	2641	
10.25[redacted]	2078	
172.1[redacted]	1933	
172.1[redacted]	1604	
173.2[redacted]	1548	

TOP N DESTINATIONS

Term	Count	Action
172.16.[redacted]	6141	
66.129.[redacted]	4347	
66.129.[redacted]	3902	
10.25.2[redacted]	3523	
172.16.[redacted]	3347	
54.208.[redacted]	2967	
54.208.[redacted]	2918	
66.129.[redacted]	2627	
172.16.[redacted]	2372	
8.8.8.8	1748	

TRAFFIC PROFILE

TRAFFIC VOLUME

FIREWALL THROUGHPUT

3.66GB

Firewall	Throughput
[redacted]Firewall	1.41GB
[redacted]Firewall	1.50GB
[redacted]Firewall	764.50MB

TRENDS

- -10.16% (allow)
- 43.22% (deny)
- -9.28% (tcp)
- -15.78% (udp)
- 49.5% (icmp)

EVENTS OVER TIME

Od sysloga do “big data”

Kamil Czekirda
kczekirda@freebsd.org

Polish BSD User Group, 13.05.2019 r.

- co to ten syslog?
- po co zbieramy logi?
- jak zbieramy i przetwarzamy

Agenda

Syslog

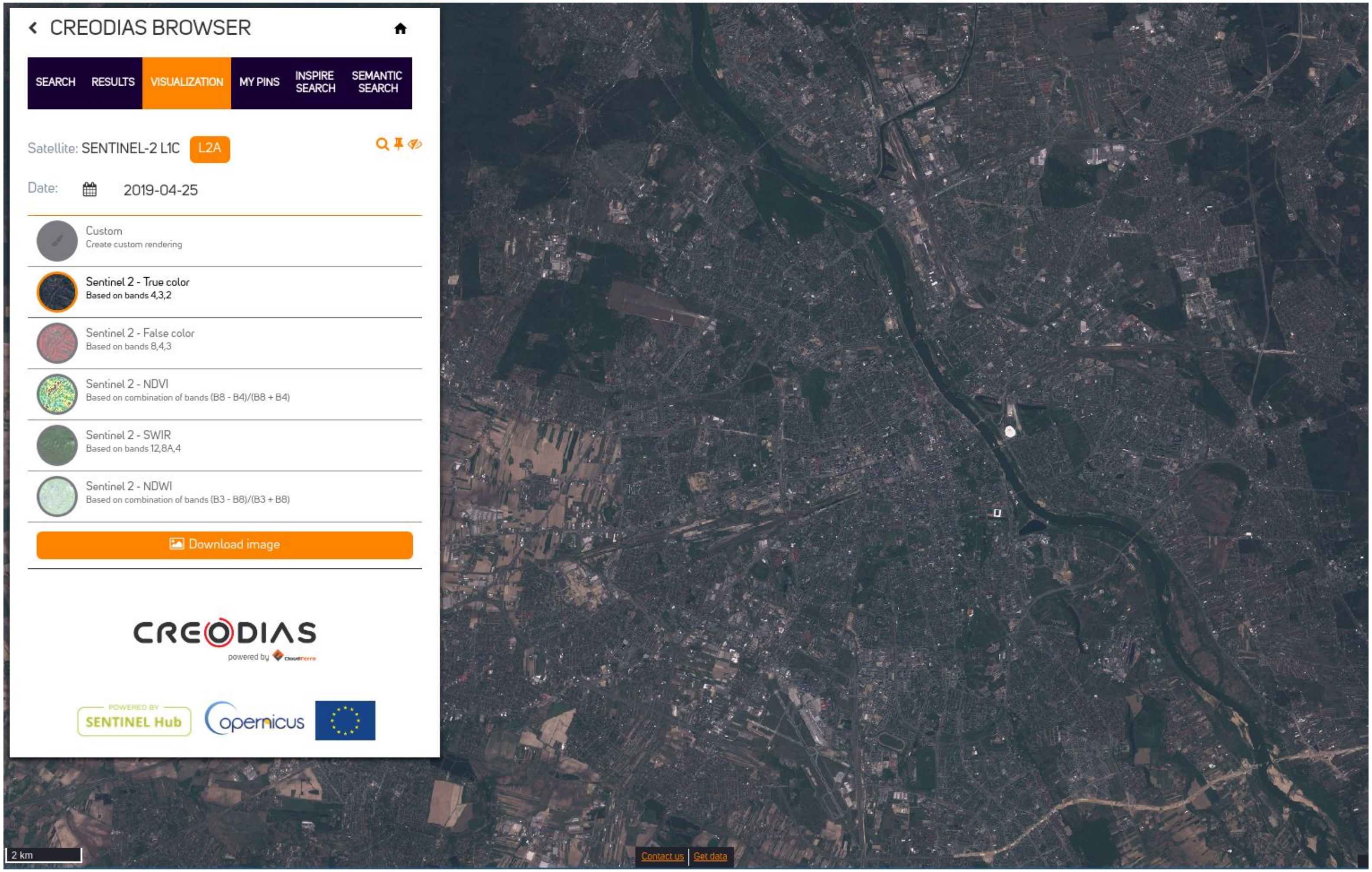
- napisany przez Erica Allmana prawie 40 lat temu
- standard logowania zdarzeń
- podział na źródło i poziom ważności
- odbieranie, obrabianie, przekierowywanie
- centralizacja
- ewolucja => rsyslog, syslog-ng...

rsyslog

- dziennik to strumień zdarzeń
- komunikaty to dane wejściowe
- rsyslog to mechanizm ich przetwarzania (filtrowanie, przekazywanie)
- każdy etap jest konfigurowalny i modułowy
- domyślnie rsyslogd czyta z /dev/log (socket)

CloudFerro

- mała firma hostingowa
- hosting obrazków



Satellite: SENTINEL-2 L1C L2A

Date: 2019-04-25

Custom
Create custom rendering

Sentinel 2 - True color
Based on bands 4,3,2

Sentinel 2 - False color
Based on bands 8,4,3

Sentinel 2 - NDVI
Based on combination of bands (B8 - B4)/(B8 + B4)

Sentinel 2 - SWIR
Based on bands 12,8A,4

Sentinel 2 - NDWI
Based on combination of bands (B3 - B8)/(B3 + B8)

hosting obrazków

Komisja Europejska
esa
European Space Agency
serco
OVH
AtoS
T · · Systems ·
creoTECH
Instruments S.A.
CloudFerro
AIRBUS
DEFENCE & SPACE
orange™

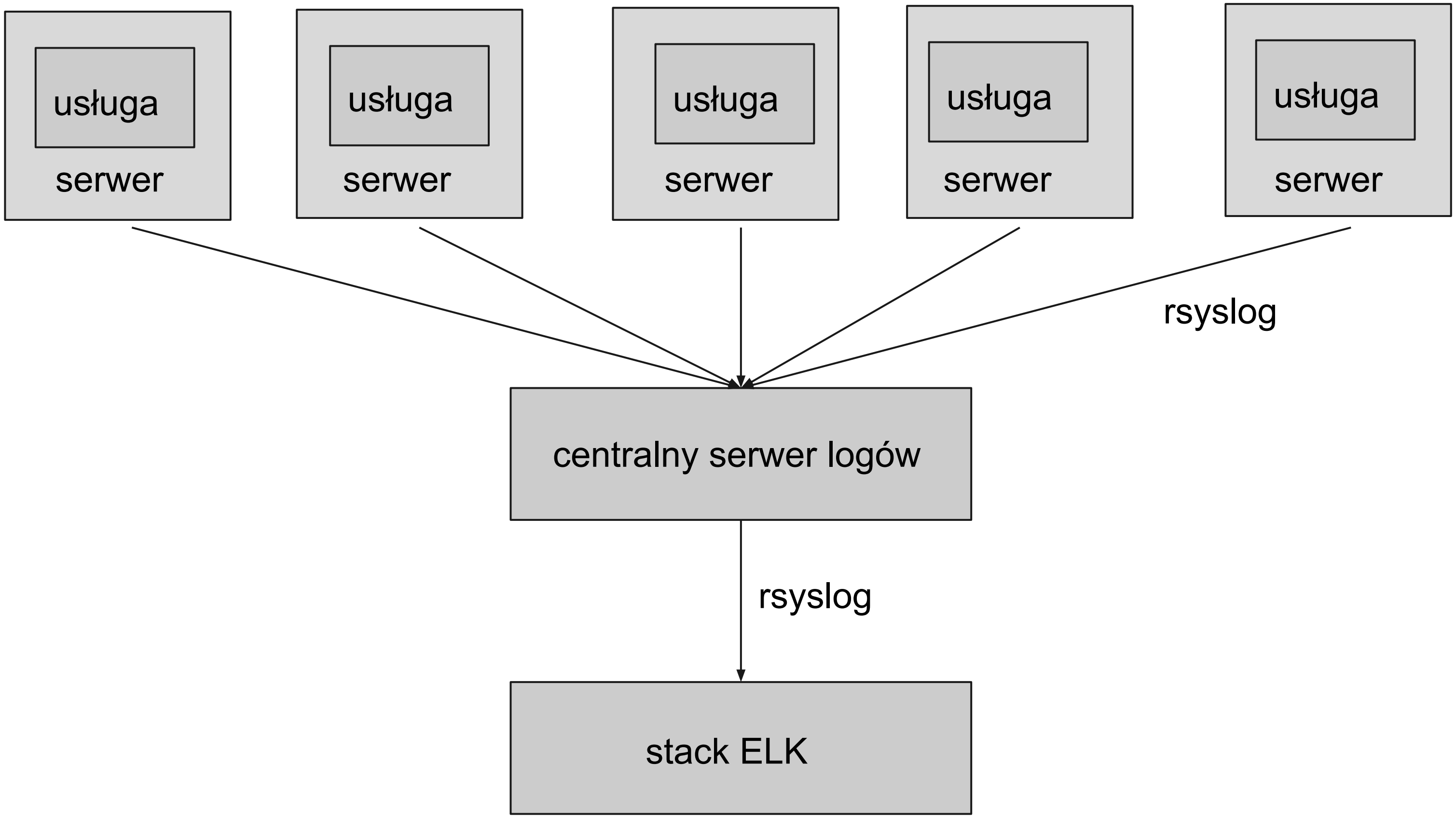
mała firma

- ~12 PB danych satelitarnych dostępnych dla użytkowników
- ~16 TB dziennej produkcji

mała firma

- billing
- monitoring
- analiza użycia repozytorium

po co zbieramy logi?



jak zbieramy logi?

/etc/rsyslog.conf

```
:syslogtag, isequal, "s3endpoint:" @logi.int.cloudferro.com:514
& stop
```

/etc/rsyslog.conf

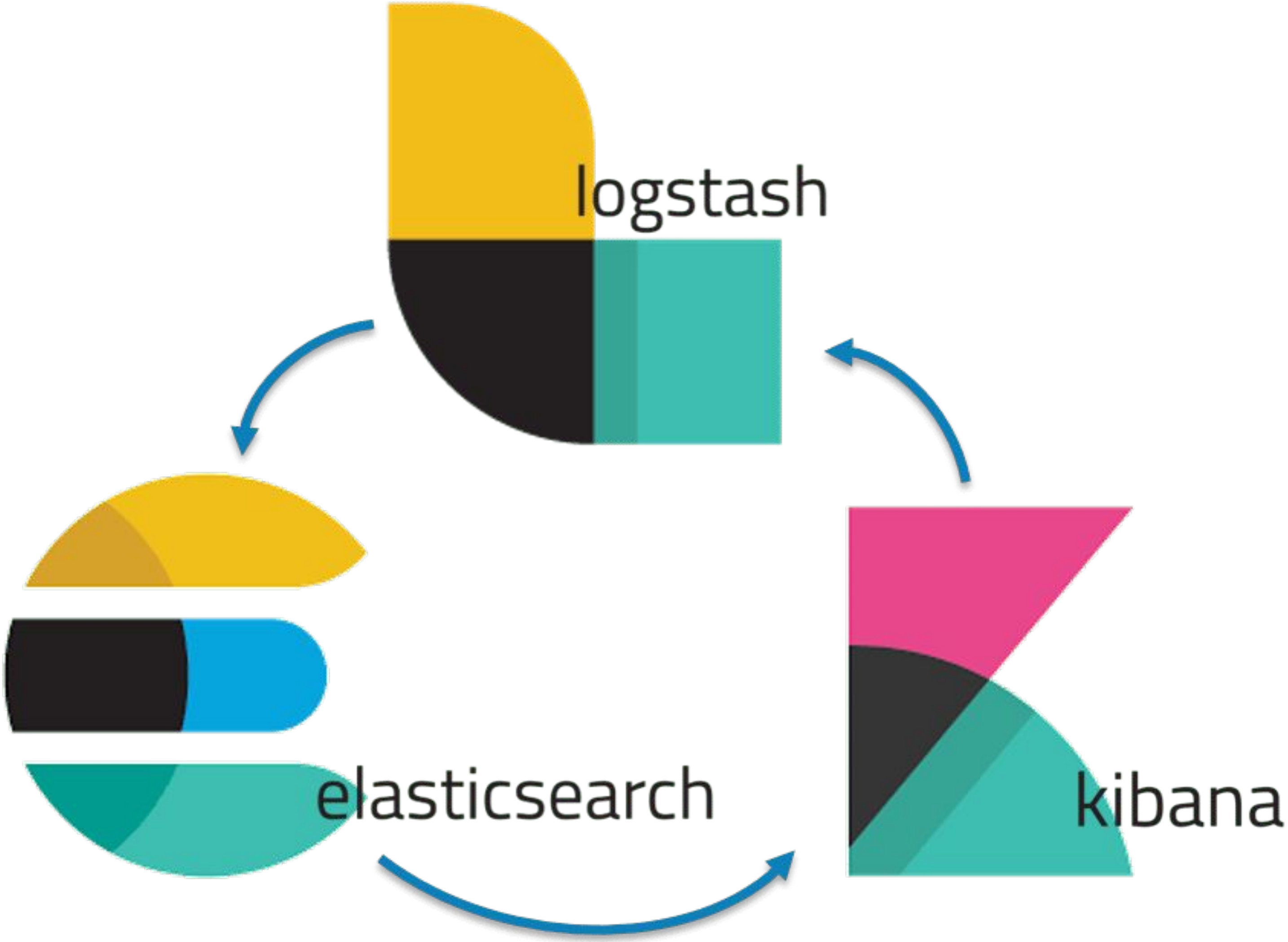
```
module(load="imudp")
input(type="imudp" port="514")
$InputUDPServerBindRuleset remote514
$UDPServerRun 514

#s3endpoint
$template DYNs3endpoint,"/logs/%fromhost%/s3endpoint.log"
:syslogtag, startswith, "s3endpoint"            ?DYNs3endpoint
:syslogtag, startswith, "s3endpoint" @elk.int.cloudferro.com:10514;json-template
& stop
```

centralny serwer logów

/etc/logrotate.conf

```
/logs/*/*.log
{
	rotate 365
	daily
	missingok
	copytruncate
	notifempty
	compress
	dateext
	sharedscripts
	postrotate
		reload rsyslog >/dev/null 2>&1 || true
	endscript
}
```

centralny serwer logów



elk stack

/etc/logstash/conf.d/remote.conf

```
input {
  udp {
    host => "10.11.12.13"
    port => 10514
    codec => "json"
    type => "rsyslog"
  }
}
```

logstash - input

```
grok {
     match => { "message" => "%{SYSLOG5424SD}
%{IP:client_ip} \(\) \{%{DATA} vars in %{DATA} bytes\}
%{SYSLOG5424SD:syslog_date} %{WORD:request}
%{URIPATHPARAM:path} => generated
%{NUMBER:t_bytes:int} bytes in
%{NUMBER:execution_time:int} msecs \(%{DATA}
%{NUMBER:response_code}\)" }
    }
```

logstash - filter

```
if "_grokparsefailure" not in [tags] {
  grok {
    match => { "path" => "/DIAS/%{DATA:collection}/" }
    match => { "path" => "/EODATA/%{DATA:collection}/" }
  }
  mutate {
    remove_field => [ "message" ]
  }
}
```

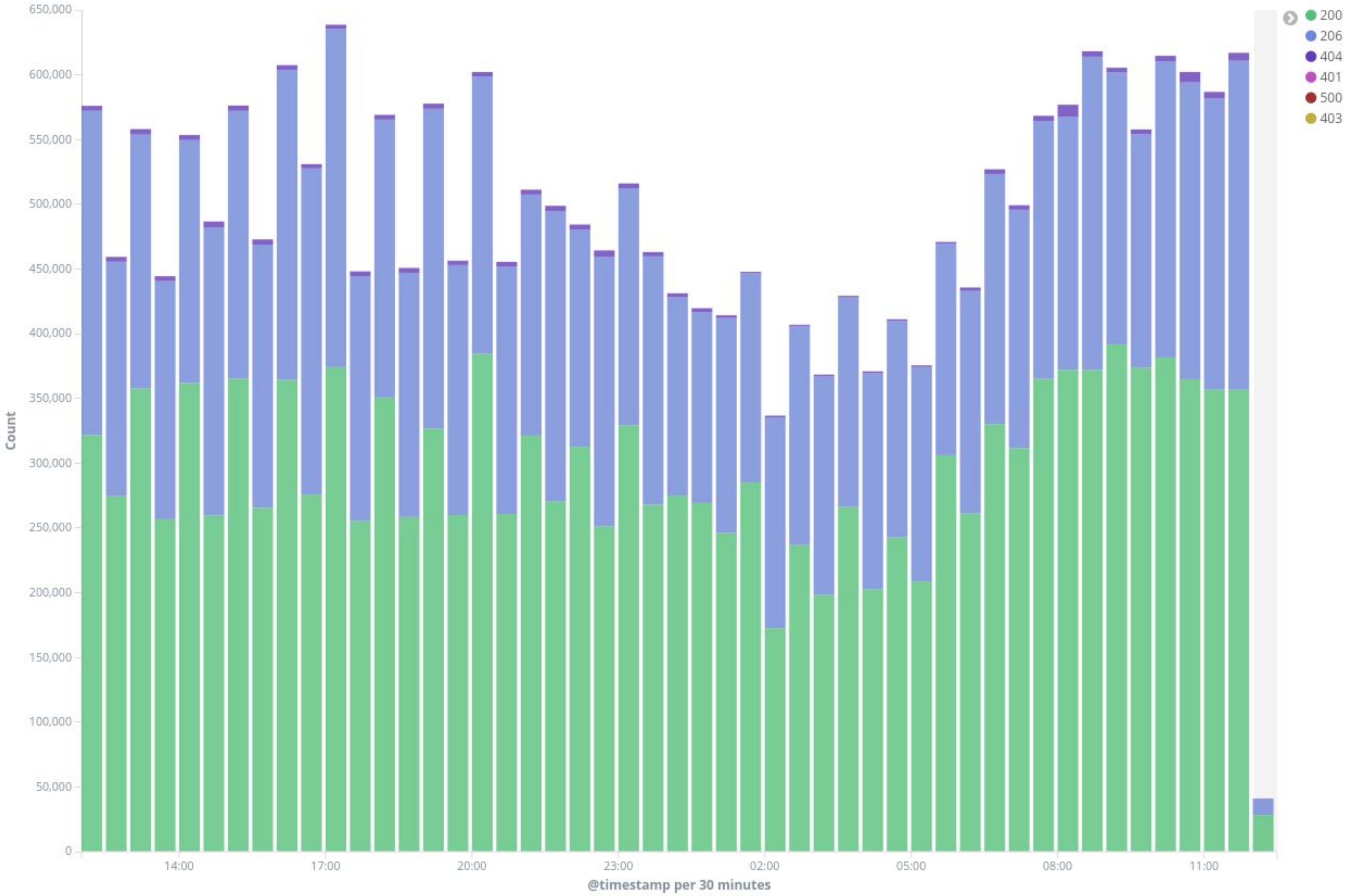
logstash - mutate

```
translate {
  regex => true
  field => "[product]"
  destination => "[collection]"
  dictionary => {
    "S1" => "Sentinel-1"
    "S2" => "Sentinel-2"
    "S3" => "Sentinel-3"
    "S5" => "Sentinel-5"
    "LC" => "Landsat-8"
    "MER" => "Envisat"
  }
    fallback => "Other"
}
```

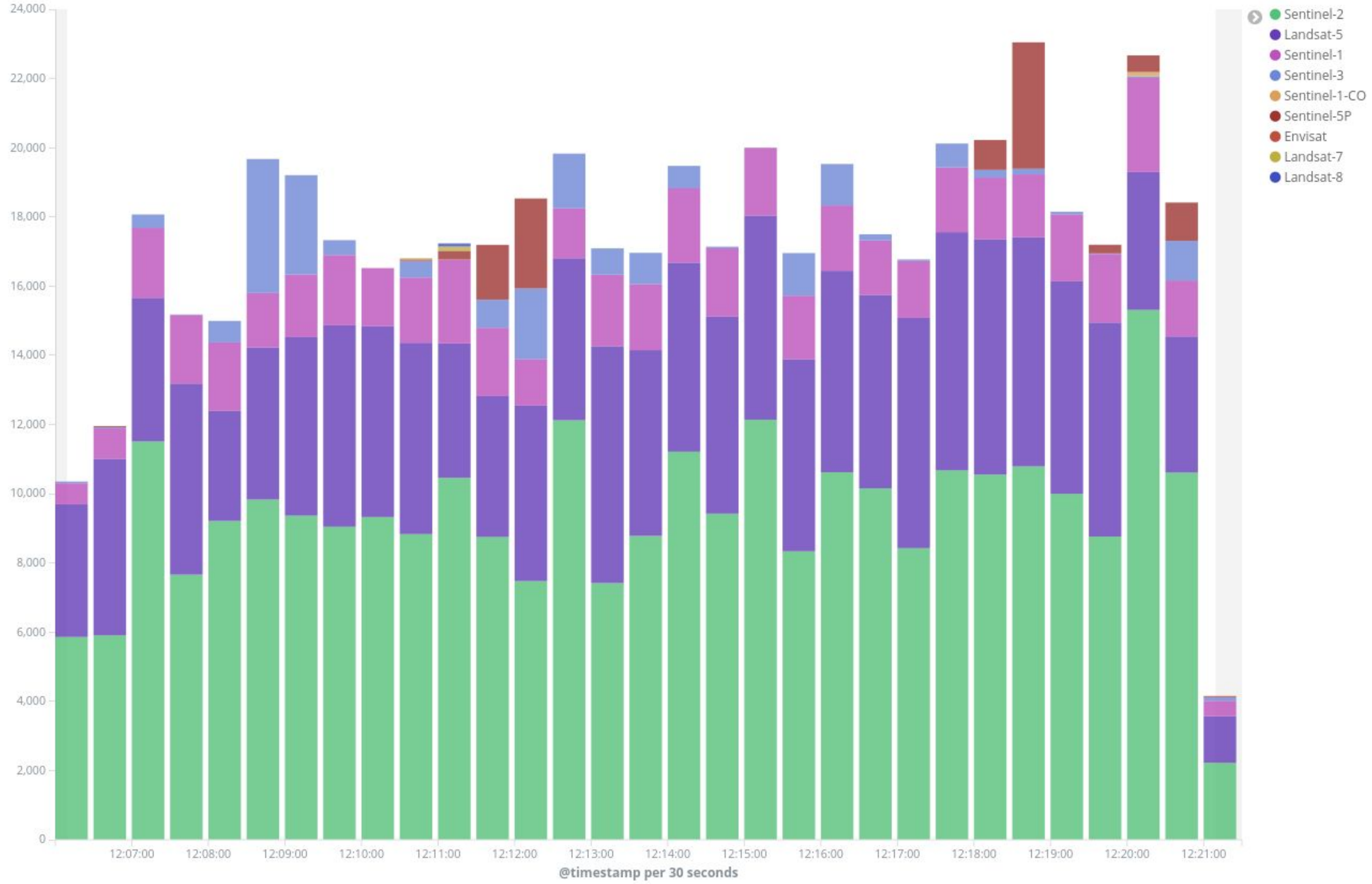
logstash - translate

```
output {
  if [type] == "rsyslog" {
        elasticsearch {
          hosts => [ "127.0.0.1:9200" ]
          document_id => "%{fingerprint}"
        }
      }
}
```

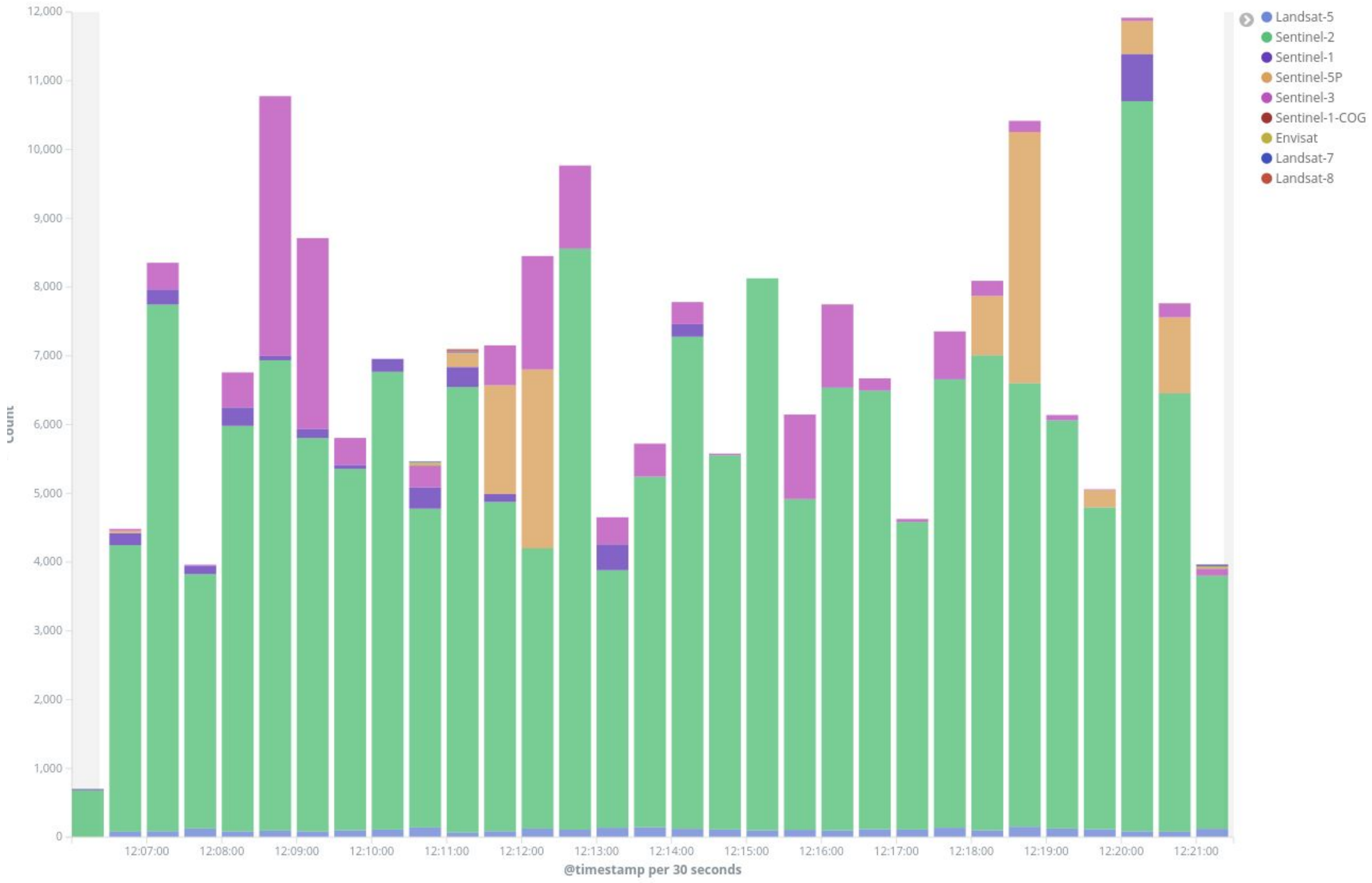
logstash - output



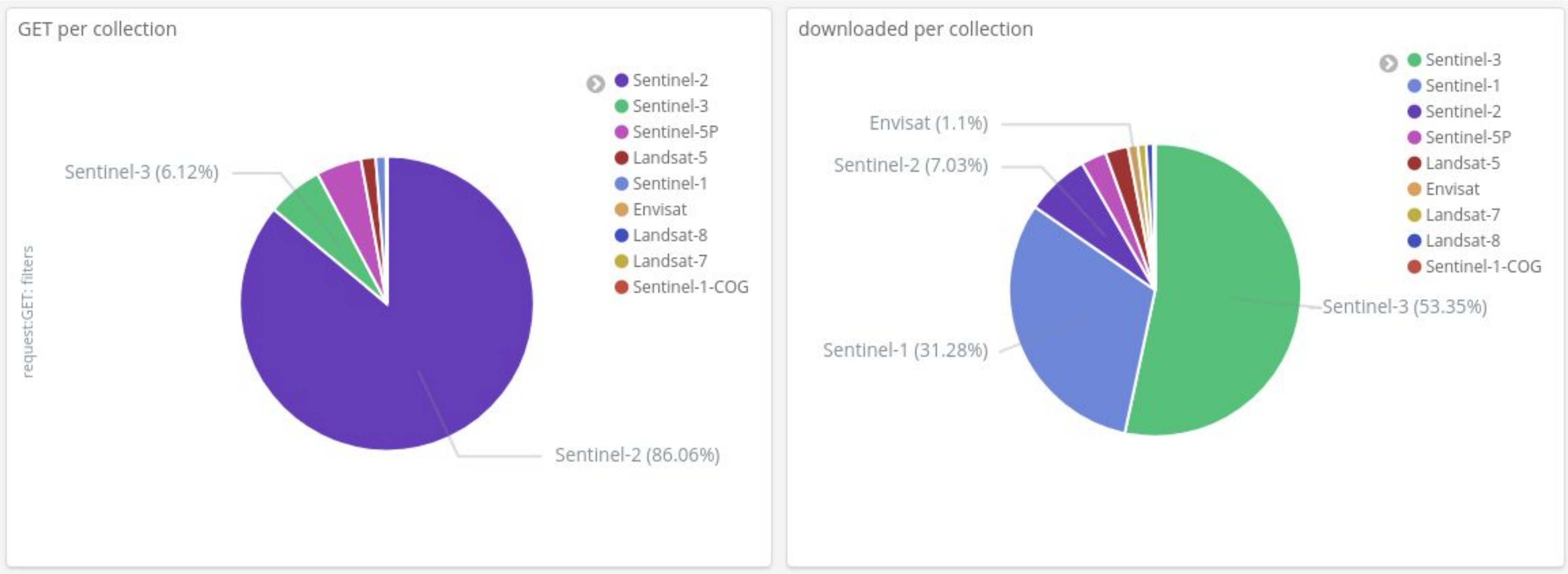
monitoring jakościowy

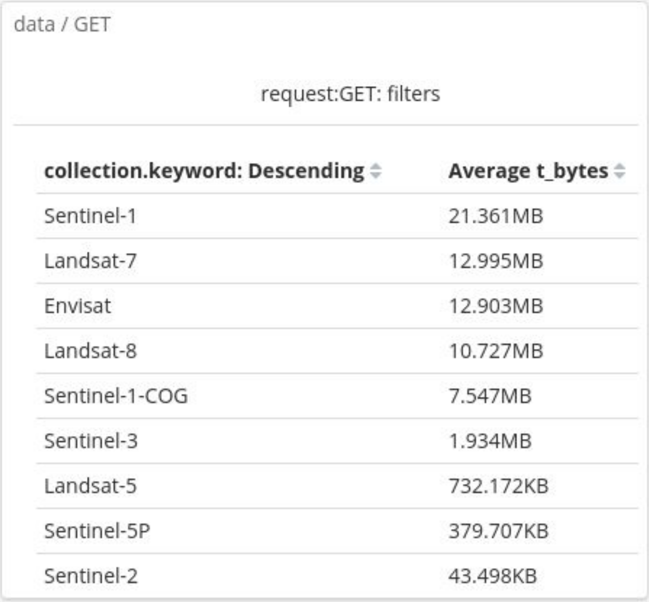

24,000
22,000
20,000
18,000
16,000
14,000
12,000
10,000
8,000
6,000
4,000
2,000
0
12:07:00
12:08:00
12:09:00
12:10:00
12:11:00
12:12:00
12:13:00
12:14:00
12:15:00
12:16:00
12:17:00
12:18:00
12:19:00
12:20:00
12:21:00
@timestamp per 30 seconds
Sentinel-2
Landsat-5
Sentinel-1
Sentinel-3
Sentinel-1-CO
Sentinel-5P
Envisat
Landsat-7
Landsat-8

requesty



requesty - tylko GET

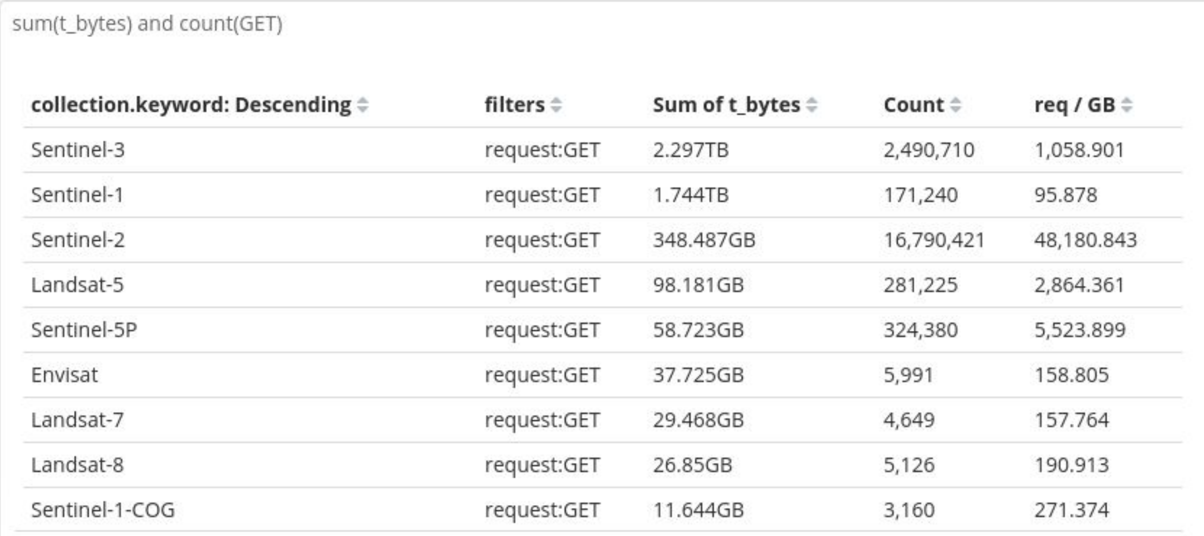

GET per collection
request:GET: filters
Sentinel-3 (6.12%)
Sentinel-2 (86.06%)
● Sentinel-2
● Sentinel-3
● Sentinel-5P
● Landsat-5
● Sentinel-1
● Envisat
● Landsat-8
● Landsat-7
● Sentinel-1-COG
downloaded per collection
Envisat (1.1%)
Sentinel-2 (7.03%)
Sentinel-1 (31.28%)
Sentinel-3 (53.35%)
● Sentinel-3
● Sentinel-1
● Sentinel-2
● Sentinel-5P
● Landsat-5
● Envisat
● Landsat-7
● Landsat-8
● Sentinel-1-COG



data / GET

request:GET: filters

collection.keyword: Descending	Average t_bytes
Sentinel-1	21.361MB
Landsat-7	12.995MB
Envisat	12.903MB
Landsat-8	10.727MB
Sentinel-1-COG	7.547MB
Sentinel-3	1.934MB
Landsat-5	732.172KB
Sentinel-5P	379.707KB
Sentinel-2	43.498KB



sum(t_bytes) and count(GET)

collection.keyword: Descending	filters	Sum of t_bytes	Count	req / GB
Sentinel-3	request:GET	2.297TB	2,490,710	1,058.901
Sentinel-1	request:GET	1.744TB	171,240	95.878
Sentinel-2	request:GET	348.487GB	16,790,421	48,180.843
Landsat-5	request:GET	98.181GB	281,225	2,864.361
Sentinel-5P	request:GET	58.723GB	324,380	5,523.899
Envisat	request:GET	37.725GB	5,991	158.805
Landsat-7	request:GET	29.468GB	4,649	157.764
Landsat-8	request:GET	26.85GB	5,126	190.913
Sentinel-1-COG	request:GET	11.644GB	3,160	271.374

Dziękuję