



Deployment and monitoring



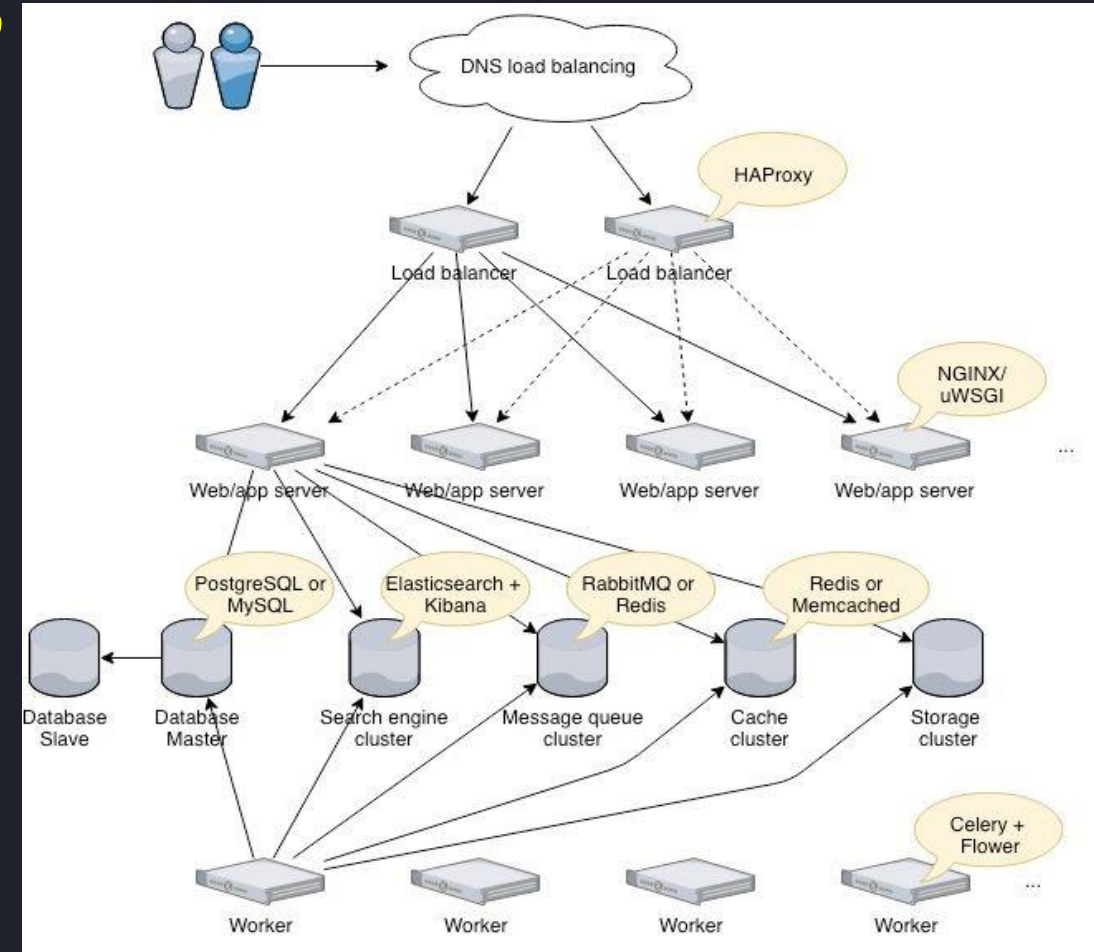
Outline

- *Deployment configurations*
- *Dimensions*
- *Monitoring*
- *Logging*

Deployment configurations

Infrastructure overview

- Load balancers: HAProxy, Nginx or others.
- Web servers: Nginx, Apache or others.
- Application servers: uWSGI, Gunicorn or mod_wsgi.
- Distributed task queue: Celery
- Database: PostgreSQL, MySQL or SQLite.
- Search engine: Elasticsearch (v5 and v6).
- Message queue: RabbitMQ, Redis or Amazon SQS.
- Cache system: Redis or Memcache.
- Storage system: Local, S3, XRootD, WebDAV and more.





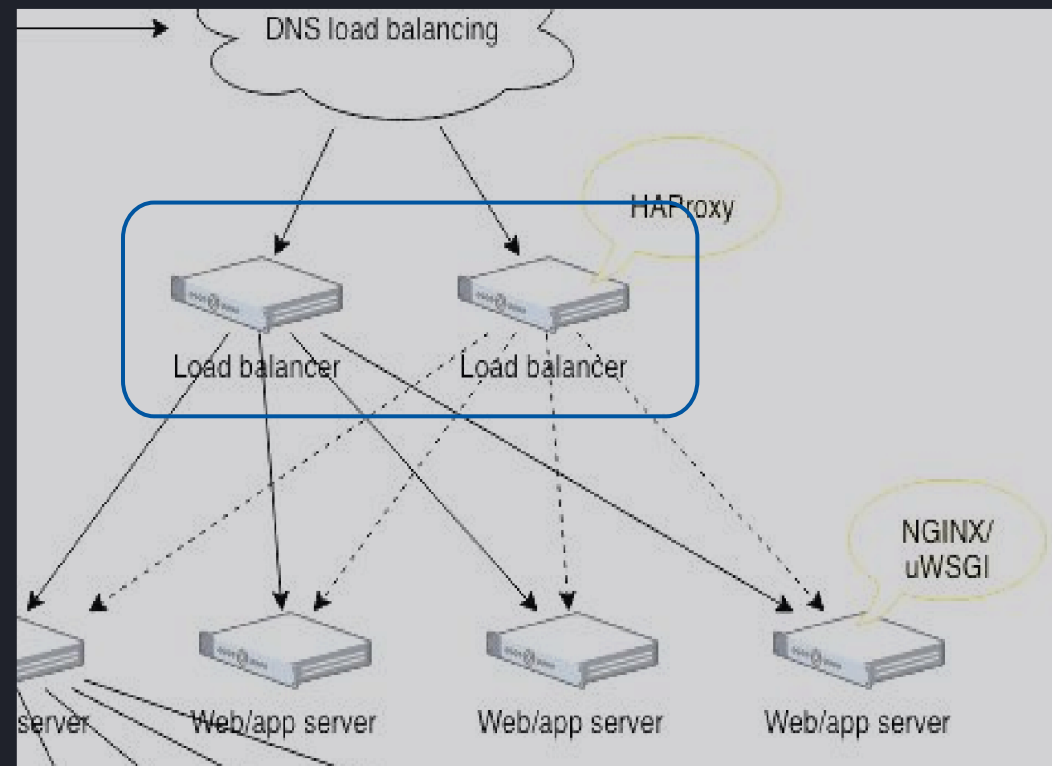
Deployment configurations

Load balancer: HAProxy

- Traffic distribution depending on its “type”
- /static, /<path>
- URL Rewriting
- SSL Termination
- Downtime static website
- More than 1: redundancy

HAProxy configuration

- Frontend
- Backend
- Headers: X-Forwarded-For

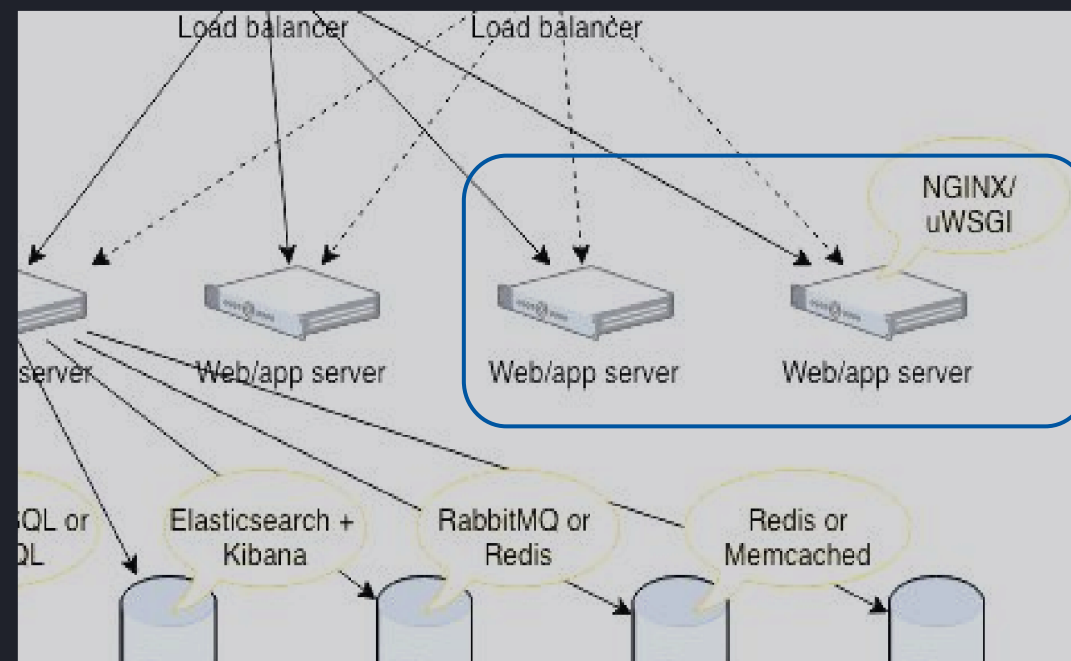




Deployment configurations

Web app: uWSGI (behind Nginx)

- [PEP 333 \(TL;DR\)](#)
- Invenio UI and REST
- processes and threads

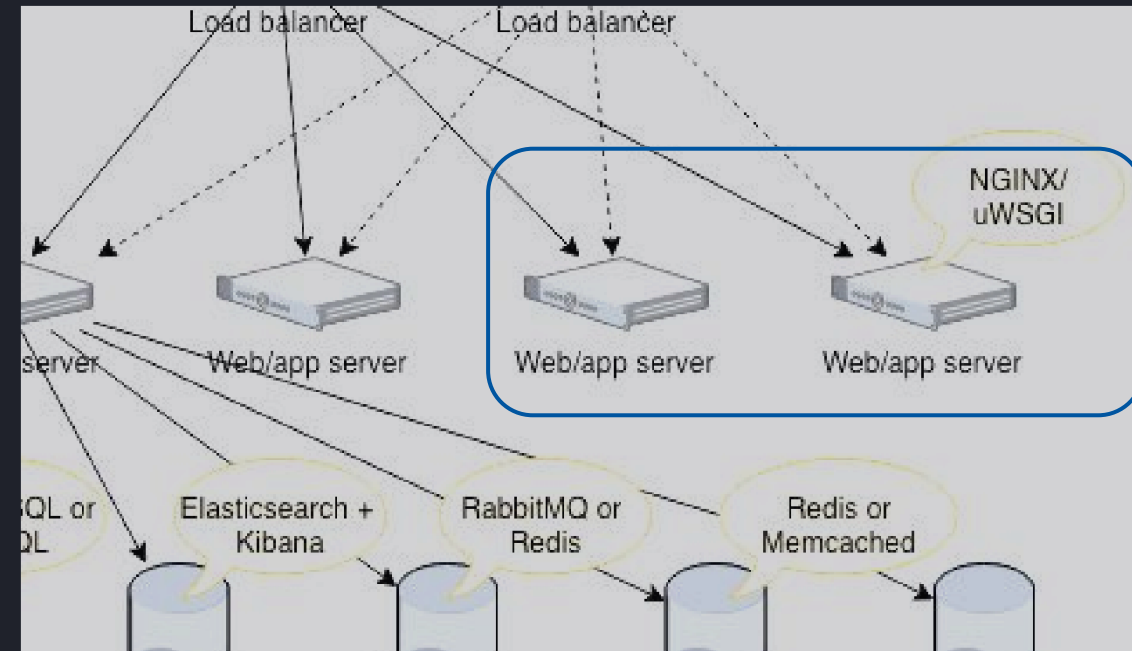


Deployment configurations



Web app: Nginx

- ngx_http_uwsgi_module: translate HTTP -> WSGI
- Reverse proxy to distribute traffic
- ui, api, static, files
- protocols and ciphers
- logging
- headers: X-Request-ID





Dimensions

Processes and Threads

Example: 200 max connections

```
$ less /var/lib/pgsql/data/postgresql.conf  
max_connections = 200  
  
$ mysql  
  
> SET GLOBAL max_connections = 200;
```

Max processes per web node

12

- SQLAlchemy: pool of db connections `pool_size` (default 5) per process
- Each uWSGI process -> 5 db connections
- 8 process per web node -> $5 * 8 = 40$ connections
- 200 max connections on db / 40 per web node = ???

5 web nodes max

BUT

- You have to count all Invenio apps (your workers, celery apps)!

Dimensions



HTTP Connections

How many connections per web node? How do I distribute traffic?

uWSGI: Start small

1. Predict/Estimate the n. of request/sec you need
2. Take one web node
3. Configure number of uWSGI processes
4. Benchmark by increasing number of requests

```
locust --host=https://www.google.com --no-web -c 1000 -r 100
```

Total Requests per Second



Dimensions



HTTP Connections

HAProxy: distribute load

Since we now know how many connections per node, we can distribute the load

```
backend ssl_invenio_app
...
server web-backup 127.0.0.1:444 backup check check-ssl fall 2 inter 5000 maxconn 255 rise 1 ssl verify none weight 2
server web1 frontend:443 check check-ssl fall 2 inter 5000 maxconn 40 rise 1 ssl verify none weight 2
server web2 frontend:443 check check-ssl fall 2 inter 5000 maxconn 40 rise 1 ssl verify none weight 2

backend ssl_invenio_files
...
server web5 frontend:443 check check-ssl fall 2 inter 5000 maxconn 100 rise 1 ssl verify none weight 2
server web6 frontend:443 check check-ssl fall 2 inter 5000 maxconn 100 rise 1 ssl verify none weight 2

backend ssl_invenio_static
...
server web3 frontend:443 check check-ssl fall 2 inter 5000 maxconn 255 rise 1 ssl verify none weight 2
server web4 frontend:443 check check-ssl fall 2 inter 5000 maxconn 255 rise 1 ssl verify none weight 2
```

Monitoring

Load balancer: HAProxy

http://127.0.0.1:8080

HAProxy

Statistics Report for pid 20749

> General process information

pid = 20749 (process #1, nbproc = 1)
uptime = 93d 18h25m07s
system limits: memmax = unlimited; ulimit-n = 8057
maxsock = 8057; maxconn = 4000; maxpipes = 0
current conns = 3; current pipes = 0/0; conn rate = 1/sec
Running tasks: 1/38; idle = 100 %

active UP backup UP
active UP, going down backup UP, going down
active DOWN, going up backup DOWN, going up
active or backup DOWN not checked
active or backup DOWN for maintenance (MAINT)
active or backup SOFT STOPPED for maintenance
Note: "NOLB"/"DRAIN" = UP with load-balancing disabled.

in-cds_cdslabs_http																		
	Queue			Session rate			Sessions						Bytes				Denied	
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	F		
Frontend				0	68	-	0	67	3 000	16 535			14 945 003	20 964 447	0			

In-cds_cdslabs_https																	
	Queue			Session rate			Sessions						Bytes				Re
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out			
Frontend				0	43	-	2	47	3 000	801 526			330 327 397 112	2 265 959 457 193			

In-cds_cdslabs_state																	
	Queue			Session rate			Sessions						Bytes			Denied	
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	
Frontend				1	2	-	1	2	3 000	37			108 078	11 547 457	0		

cds_cdslibs_app																			
	Queue			Session rate						Sessions						Bytes			Rec
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out					
videos-cds-lb2.cern.ch-backup	0	0	-	0	0	0	0	0	0	255	0	0	?	0	0				
videos-cds-web1.cern.ch	0	0	-	0	16	0	0	2	40	12 904	12 904	40m8s	3 738 807	5 208 625					
videos-cds-web2.cern.ch	0	0	-	0	19	0	0	2	40	12 908	12 908	19m52s	3 680 512	5 211 161					
videos-cds-web3.cern.ch	0	0	-	0	16	0	0	2	40	12 903	12 903	39m44s	3 768 591	5 205 159					
videos-cds-web4.cern.ch	0	0	-	0	17	0	0	2	40	12 904	12 904	9m5s	3 658 629	5 203 619					
Backend	0	0		0	68	0	0	8	300	51 619	51 619	9m5s	14 846 539	20 826 564					

cds_cdslabs_files																		
	Queue			Session rate			Sessions					Bytes			Denied			P
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp		
videos-cds-web5.cern.ch	0	0	-	0	1		0	1	256	8	8	25d10h	1 199	3 486		0		
videos-cds-web6.cern.ch	0	0	-	0	1		0	1	256	7	7	28d20h	897	3 052		0		
Backend	0	0		0	2		0	1	300	15	15	25d10h	2 096	6 538	0	0		

Monitoring



Web app: uWSGI

- [uWSGI Stats Server](#)
- [uwsgitop](#)

node: 127.0.0.1 - cwd: /opt/zenodo - uid: 1001 - gid: 1001 - masterpid: 28932


WID	%	PID	REQ	RPS	EXC	SIG	STATUS	AVG	RSS	VSZ	TX	ReSpwn	HC	RunT	LastSpwn
7	10.1	28968	120274	0	2528	0	idle	776ms	0	0	52.0G	1	0	1278m	11:45:37
10	10.1	28971	120126	0	2559	0	idle	1ms	0	0	57.0G	1	0	1394m	11:45:37
2	10.0	28946	120067	0	2564	0	idle	1ms	0	0	54.0G	1	0	1447m	11:45:37
1	10.0	28942	119980	1	2526	0	idle	1ms	0	0	57.0G	1	0	1546m	11:45:37
8	10.0	14707	119859	0	2548	0	idle	2ms	0	0	49.0G	2	0	1552m	10:13:16
5	10.0	28958	119785	0	2616	0	idle	846ms	0	0	89.0G	1	0	1655m	11:45:37
3	10.0	28952	119741	0	2500	0	busy	5ms	0	0	61.0G	1	0	1589m	11:45:37
4	10.0	8635	119062	0	2643	0	idle	1ms	0	0	67.0G	2	0	2230m	12:07:51
9	9.9	28970	118862	0	2623	0	busy	1ms	0	0	63.0G	1	0	2199m	11:45:37
6	9.8	28967	117497	0	2505	0	busy	12ms	0	0	62.0G	1	0	3222m	11:45:37



Monitoring

ElasticSearch: Kibana

<http://127.0.0.1:5601>

**kibana**

Discover

Visualize

Dashboard

Timelion

Logout

Indices

Own Home

Management

Elasticsearch index status

Sort by

name

Index name	health	# replicas	# shards	# docs	size (including replicas)
index name ...	health...	replicas ...	shards ...	docs ...	size ...
.kibana_cdsbooksqa	green	1	1	1	6.8kb
cdsbooks-documents-document-v1.0.0	green	1	5	19	259.3kb
cdsbooks-internal_locations-internal_location-v1.0.0	green	1	5	8	64.4kb
cdsbooks-items-item-v1.0.0	green	1	5	49	489.6kb
cdsbooks-loans-loan-v1.0.0	green	1	5	99	337.4kb
cdsbooks-locations-location-v1.0.0	green	1	5	1	12.6kb
Selected: 6		Selected: 6		Selected: 26	
Total: 6		Total: 6		Total: 177	
		Total: 26		Total: 1.1mb	

Monitoring

Job queue

<http://127.0.0.1:15672> - guest:guest



Overview

Connections

Channels

Exchanges

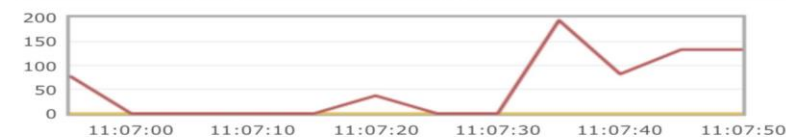
Queues

Admin

Overview

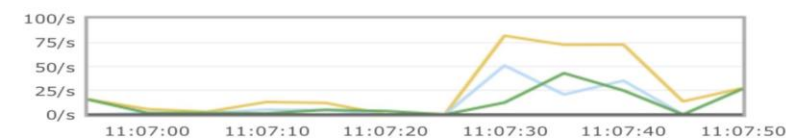
Totals

Queued messages (chart: last minute) (?)



Ready 0 msg
Unacknowledged 0 msg
Total 0 msg

Message rates (chart: last minute) (?)



Publish 27/s
Deliver 0.00/s
Redelivered 0.00/s
Acknowledge 27/s
Get 0.00/s
Get (noack) 0.00/s

Celery Flower

Dashboard

Tasks

Broker

Monitor

Docs

Code

Active: 0

Processed: 19781

Failed: 759

Succeeded: 18687

Retried: 361



Shut Down




	Worker Name	Status	Active	Processed	Failed	Succeeded	Retried	Load Average
<input type="checkbox"/>	celery@zenodo-task7.cern.ch	Online	0	4929	202	4633	93	0.04, 0.03, 0.05
<input type="checkbox"/>	celery@zenodo-task5.cern.ch	Online	0	4954	255	4607	101	0.08, 0.08, 0.05
<input type="checkbox"/>	celery@zenodo-task6.cern.ch	Online	0	4953	133	4759	80	0.0, 0.01, 0.05
<input type="checkbox"/>	celery@zenodo-task4.cern.ch	Online	0	4945	169	4688	87	0.0, 0.01, 0.05

Monitoring

Alerting: UpTimeRobot





[Upgrade](#)[Dashboard](#)[My Settings](#)

+ Add New Monitor

(Bulk Actions)

Sort Monitors ▾

(Export Monitors - Shorten Monitor Names)

⌚ Last 24 Hours

14

18

22

2

6

10

99.67%	http	CDS Videos - Streaming	<div></div>	⚙
100%	http	CDS Videos - Search	<div></div>	⚙
100%	http	CDS Videos - Record	<div></div>	⚙
100%	http	CDS Videos - EOS	<div></div>	⚙
94.76%	http	CDS Legacy - Search	<div></div>	⚙
100%	http	CDS Legacy - MediaArchive	<div></div>	⚙
97.87%	http	CDS Legacy - Home	<div></div>	⚙
82.84%	http	CDS Legacy [SSL] - Home	<div></div>	⚙

Account Dashboard

details about the account

A place to find all the details about your monitors

Quick Stats

You are currently using 8 of your 50 monitors

UP MONITORS

8

show "up" monitors



Logging

Load balancer: HAProxy

```
backend ssl_app
  balance leastconn
  http-check disable-on-404
  option http-server-close
  option forwardfor except 127.0.0.0/8
  option httpchk HEAD /ping HTTP/1.0
  server web1 frontend:443 check check-ssl fall 2 inter 20000 maxconn 30 rise 1 ssl verify none weight 2

backend ssl_static
  balance leastconn
  http-check disable-on-404
  option http-server-close
  option forwardfor except 127.0.0.0/8
  option httpchk HEAD /ping HTTP/1.0
  server web1 frontend:443 check check-ssl fall 2 inter 5000 maxconn 255 rise 1 ssl verify none weight 2
```


Logging



Web app: Nginx

Infrastructure IPs

```
172.22.0.12 my-site_lb_1
172.22.0.11 my-site_frontend_1
172.22.0.10 my-site_web-ui_1
172.22.0.7 my-site_worker_1
172.22.0.9 my-site_web-api_1
172.22.0.6 my-site_flower_1
172.22.0.8 my-site_kibana_1
172.22.0.4 my-site_mq_1
172.22.0.5 my-site_db_1
172.22.0.3 my-site_cache_1
172.22.0.2 my-site_es_1
```

Nginx log config

```
log_format trace '$remote_addr - [$time_local] "$request" '
                 '$status $body_bytes_sent "$http_referer" '
                 '"$http_user_agent" '
                 '"$http_x_forwarded_for" $request_id '
                 '$msec $request_time '
                 '$upstream_http_x_session_id $upstream_http
```

Log example

```
172.22.0.12 - [15/Mar/2019:14:56:27 +0000] "GET /api/records
200 209 "https://127.0.0.1/search?q="
"Mozilla/5.0 (Macintosh; Intel Mac OS X 10.14; rv:65.0) Gecko
"172.22.0.1" 00fdb97e91fe18238733045e9c23a331
1552661787.662 2.242
84e4ee86efbf0e53_5c8bb8cc -
```


Logging



Web app: Nginx

Nginx passes
params to the
uWSGI server

```
location /api/files {  
    gzip off;  
    uwsgi_pass api_server;  
    include uwsgi_params;  
    uwsgi_buffering off;  
    uwsgi_request_buffering off;  
    uwsgi_param Host $host;  
    uwsgi_param X-Forwarded-For $proxy_add_x_forwarded_for;  
    uwsgi_param X-Forwarded-Proto $scheme;  
    # Pass request id to api server  
    uwsgi_param X-Request-ID $request_id;  
    # X-Session-ID / X-User-ID is read by nginx and included in the logs,  
    # however we don't want to expose them to clients so we are hiding them.  
    uwsgi_hide_header X-Session-ID;  
    uwsgi_hide_header X-User-ID;  
    # Max upload size for files is set to 50GB (configure as needed).  
    client_max_body_size 50G;  
}
```



Logging

Web app: uWSGI

- [invenio-app](#) can read the header X-Forwarded-For and the request_id
- It will contain the list of all IPs (HAProxy, Nginx, etc...)

Error reporting

- [invenio-logging](#)
- Sentry

Invenio Logging

- Log to file
- Configure logging level
- Configure file rotation
- Configure Sentry integration

Logging

Sentry

