Nudgis - Administrator manual

This documentation provides information regarding upgrade, maintenance and backups of Nudgis installations.

Cette documentation est aussi disponible en français.

Table of contents

1. Definitions

Nudgis: the main product, that contains all the services (portal, streaming server, monitoring service). Also referred to as the video portal (the user-facing service).

Miris Manager: web interface that is used to control, update and schedule the Miris Box, Netcapture, Campus and Studio devices.

Nudgis Worker: server that processes video, part of a compute cluster that can be extended, and controlled by the Tasks server.

Instance: the Nudgis can host multiple portals, called instances; each represent a dedicated webtv portal that is accessed through a specific URL and has a dedicated folder in /home (e.g. /home/msuser)

Monitoring service: the Nudgis includes a monitoring application that displays usage graphs. It is accessed through a specific URL.

Streaming server: the Nudgis includes a streaming server that is responsible for distributing live traffic. It listens on port tcp 1935.

Tasks server: the Nudgis solution includes a tasks execution cluster called celerity which is used to process incoming video (transcoding, trimming, etc…), which are executed by the Nudgis Worker servers.

2. Administration

2.1. Upgrade

2.1.1. Application upgrade

Please read this section completely before entering any command on the server.

On all UbiCast servers (Nudgis, Nudgis Worker, Miris Manager, Nudgis Cache, Nudgis Vault, ...) the following commands will upgrade the software:

```
sudo apt-get update
sudo apt-get dist-upgrade
```

Why **dist-upgrade** and not simply **upgrade** ?
Dist-upgrade is used to upgrade every packages even if they require installing other packages (dependencies) or if they require a reboot.

Note that it is necessary to upgrade both servers (Nudgis and Nudgis Worker) or there is a high probability that the task execution will not work anymore.

It is recommended to reboot the server after the upgrade using the "sudo reboot" command. The reboot will make the system boot on the latest kernel and will allow you to test that everything starts correctly when the server boots (unattended reboots can happen in case of a power cut for example).

Once the server has booted, you can run the UbiCast automated tests by running this command:

```
sudo /root/ubicast-tester/tester.py
```

During the upgrade, it is possible that APT asks some questions like this one:

```
Setting up skyreach (7.1.1+20190103174423-0b7c908-1) ...
```

```
Configuration file '/etc/nginx/sites-available/skyreach.conf'
 ==> Modified (by you or by a script) since installation.
 ==> Package distributor has shipped an updated version.
   What would you like to do about it ?  Your options are:
     Y or I  : install the package maintainer's version
     N or O  : keep your currently-installed version
     D     : show the differences between the versions
     Z     : start a shell to examine the situation
 The default action is to keep your current version.
*** skyreach.conf (Y/I/N/O/D/Z) [default=N] ?
```

It is recommended to answer *Y* for all questions about UbiCast maintained packages except for the `/etc/Nudgis/msconf.py` file. UbiCast packages that can raise questions on the configuration are essentially: `ubicast-Nudgis`, `ubicast-monitor`, `ubicast-skyreach`.
For other packages, it is advised to answer *N*.
If your server uses special Nginx configuration (like a non standard port), you will have to modify manually the configuration after the upgrade.
In the version 7.9.0 of Nudgis, we have changed the Nginx configuration to use fragments imported with the "include" keyword to avoid these questions from APT.

2.1.2. Operating system upgrade

The operating system (OS) upgrade requires some knowledge about Linux because some questions related to the operating system services will be asked during the procedure.

The recommended OS is Debian 10.

If your server is still using Ubuntu, please follow this procedure:

Step 1: Update Ubuntu to the 18.04 version (if not already done):
https://docs.google.com/document/d/e/2PACX-1vRqD4rHEMfpXQM--XjSYViG2NN8eMfT1r1p_8y4kx1LHL31LYcAaSDXUYPNzIgnLVxGVTuc5gBtZiYu/pub?embedded=true

Step 2: Migrate Ubuntu 18.04 to Debian 10:
https://docs.google.com/document/d/e/2PACX-1vQzVW3KleX56s88pyVPYPH98s7gQl2mzweXxPOfJqmCYBzgYdnLOjW2x6nUKjMMduJR2RAw0_z3kUfL/pub?embedded=true

2.1.3. PostGreSQL upgrade

To upgrade PostgreSQL, follow this guide:
https://docs.google.com/document/d/e/2PACX-1vTjpeBbowtqNWSU6uPm7Dh7LDFiucMQWek6cb-v1HDiWFZmYOufmqx2w6WPhWbqSZcnzACjpM-uxHnZ/pub?embedded=true

2.2. Backup

Nudgis is based on Debian and only some locations have to be backed up.

2.2.1. Backup procedure

To backup the files and in order to be able to restore them, you have to preserve all files attributes (owner, group, permissions).

2.2.2. Database backup

Nudgis

By default, a database dump is created each day by a cron script (and kept for 30 days in `/home/msuser/msinstance/dbdumps/`).

You can dump all Nudgis instances database with the following command:

```
mscontroller.py dump
```

To dump a single instance database:

```
mscontroller.py dump -u username
```

Example:

```
mscontroller.py dump -u msuser
```

Miris Manager

By default, a database dump is created each day by a cron script (and kept in `/home/skyreach/.skyreach/dbdumps/`).

To dump the Miris Manager database, run the following command:

```
service skyreach dump
```

The resulting dump will be shown in command output (a path like `/home/skyreach/skyreach_data/private/dbdumps/backup.2018-11-07_08-47-06.sql`).

Monitor

The monitoring interface has no database.

2.2.3. Locations to backup

Here is the list of all locations that must be backed up to be able to restore the system.

```
/home
```

This directory contains all data (videos, jpeg slides, photos, images, database dumps, ...). This location exists on: Nudgis, Miris Manager, Worker.

```
/data
```

If the storage is not local, the remote storage like NFS is mounted on `/data`, (so that you can still log onto the system in case the NFS is down) and then a symbolic link `/home/msuser/msinstance` folder points to `/data` (so that the home folder is available for login even if the NFS share is unavailable). In this case, you should back it too, but the `/home` backup should not follow the symlinks (but it should preserve links).

To save space on the backup, you can skip HLS resources (m3u8 and ts below; note that you will need to re-run transcoding on all resources if you restore from this backup); you can use the rsync exclusion list below as example:

```
- .zfs/
- *.log
- *.sock
- *.pyc
- *.swp
- *.pid
- *.part
- __pycache__/
- apt-cacher-ng/
- */msinstance-disabled/
- *.lock
- .nfs*
- *.m3u8
- *.ts
- *.tmp/
```

```
/etc/nginx
```

This location contains the configuration of the Nginx server. It isn't useful to backup this every day (once per week or month should be enough).
This location exists on: Nudgis, Miris Manager.

```
/etc/Nudgis
```

This location contains the shared configuration files for all Nudgis instances of the server. As for the Nginx configuration directory, it isn't useful to backup this every day (once per week or month should be enough).
This location exists on: Nudgis.

2.2.4. Backup restoration

NOTE: Never attempt to restore a backup on a system that has a Nudgis version older than the one that was used at the time of the backup. Always upgrade (to a version >= of the version used in the backup) Nudgis before restoring.

All following commands should be run as root.

A) Before restoring, you should stop all services.

```
systemctl stop Nudgis
systemctl stop msmonitor
systemctl stop skyreach
```

B) Restore files to their original locations with their original permissions.

C) Restore the database by typing :

```
mscontroller.py restore -u username -t
/path_to_last_Nudgis_database_dump

service skyreach restore /path_to_last_skyreach_database_dump
```

D) Once you have restored files and databases, you can start services :

```
systemctl restart Nudgis
systemctl restart msmonitor
systemctl restart skyreach
```

2.2.5. Full OS backup

In order to prepare for a full system restoration, it might be a good idea to perform a full copy of the OS from time to time, using a VM snapshot (virtualization), or by performing a full rsync copy of the following folders (hardware).

```
for filename in /dev/sd*; do sudo sfdisk -d "${filename}"; done >
/path/to/backup/folder/partitions_dump
rsync -aAXv
--exclude={"/dev/*","/proc/*","/sys/*","/tmp/*","/run/*","/mnt/*","/m
edia/*","/lost+found"} / /path/to/backup/folder
```

Source: https://wiki.archlinux.org/index.php/full_system_backup_with_rsync

To restore, you will have to
- re-create the same partitioning using the dump (note that hardware units come with a RAID6 array + LVM).
- restore with the same command, yet reverted
- reinstall grub

```
rsync -aAXv
--exclude={"/dev/*","/proc/*","/sys/*","/tmp/*","/run/*","/mnt/*","/m
edia/*","/lost+found"} /path/to/backup/folder /newroot
```

However, this is not recommended as the data and the full backup will diverge: the backed-up data and database contents might be more recent than the software present in the OS backup, hence incompatible. This can be solved by first updating the last backup, then injecting the data, then relaunching database patches using

```
dpkg-reconfigure ubicast-Nudgis
dpkg-reconfigure ubicast-skyreach
```

2.3. Logs location

2.3.1. Nudgis logs

- /home/msuser/mstmp/
  - mediaserver.log : general, user-facing interaction logs
  - celery-interactions.log celery_monitor.log : worker-related logs
  - django.log : generic, low level logs
  - resources_checker.log resources_deleter.log : media resources related logs (detection, deletion)
  - restarter.log : restarter daemon log (application restart after configuration changes)
  - services_checker.log : FTP service checking logs (FTP is deprecated)
  - statistics_checker.log : statistics data computation logs
  - uwsgi.log : requests log
  - api.log : API requests logs
- Webserver logs: /var/log/nginx/
  - access_msuser.log and error_msuser.log : webserver access and error logs
- Streaming server logs: /var/log/nginx/
  - rtmp.log: live RTMP ingest logs
  - access.log: live HLS output logs

2.3.2. Miris Manager log

- /home/skyreach/skyreach_data/logs/
  - django.log : generic, low level logs
  - uwsgi.log : requests log
  - skyreach.log : general log
  - stations_calendars_updater.log : external calendar parsing logs (e.g. Syllabus+)
  - stations_calendars_parser.log : calendar to commands extraction logs
  - stations_profiles_manager.log : system profiles caching log
  - stations_support_dates_checker.log : license/support verification logs
  - stations_tasks_manager.log : remote control logs
  - synchronizations_launcher.log : manual package synchronization logs (push or pull)
  - twisted.log : remote control daemon logs (long-polling)
  - synchronizations_checker.log : automatic package synchronization logs

- /var/log/nginx/access_skyreach.log and /var/log/nginx/error_skyreach.log : webserver access and error logs

2.3.3. Nudgis Worker logs

Server side: /var/lib/celerity/*.log (usually, the server side is the system in which Nudgis is running)
Client side: /home/celerity/celerity/*.log (usually, the client side is the system in which the tasks are executed)

2.3.4. Monitoring logs

- /home/msmonitor/msmonitor/logs
  - django.log : generic, low level logs
  - uwsgi.log : requests log
  - site.log : general log
  - daemons_manager.log : monitoring daemons logs
  - instances_checker.log : Nudgis instance monitoring logs
  - latency/ : Nudgis instance http latency logs
- /var/log/nginx/access_msmonitor.log and /var/log/nginx/error_msmonitor.log : monitoring interface access and error logs

2.3.5. PostgreSQL logs

PostgreSQL is the database server.
Logs location:
    /var/log/postgresql/

2.3.6. Nginx logs

Nginx is the web server. It handles incoming HTTP(S) requests.
Logs location:
    /var/log/nginx/
The default configuration of the Ubicast Nginx package enables logs buffering to avoid frequent writing to the disk. By default the logs are written every minute or every 10 MB (the setting is defined in /etc/nginx/nginx.conf in the l

2.4.1. Nudgis commands

All commands should be run as root.

Restarting the Nudgis instances

This script is used to control the daemons to run a Nudgis instance.
Mostly, you will need this script to restart the instances. To do that the command is:

```
mscontroller.py restart or systemctl restart Nudgis
```

To restart a specific instance:

```
mscontroller.py restart -u username
```

Example:

```
mscontroller.py restart -u msuser
```

This command may restore the service if the service does not work anymore (e.g. 502 HTTP error).

Dumping the database contents

To dump all instances databases:

```
mscontroller.py dump
```

The database dumps will be saved in /data/backup/dbdump (if using a network volume) or /home/backup/dbdump.

To dump a single instance database:

```
mscontroller.py dump -u username
```

Example:

```
mscontroller.py dump -u msuser
```

Restarting the monitoring service

This script is used to control the daemons for the monitoring service.
To restart the monitoring service:

```
systemctl restart msmonitor
```

This command may restore the service if the service does not work anymore (e.g. 502 HTTP error).

Restarting the webserver and RTMP server

nginx is used as webserver and live RTMP streaming server; after configuration changes, you can restart it using:

```
systemctl restart nginx
```

Restarting the database

PostgreSQL is used as database; you can restart it using:

```
systemctl restart postgresql
```

Restarting the tasks server

Nudgis includes a task server, which is responsible for dispatching the processing tasks to the Nudgis Workers. It can be restarted using:

```
systemctl restart celerity-server
```

2.4.2. Nudgis Worker commands

Restart the service

If the Nudgis Worker is not visible in the Nudgis tasks overview, it is possible to restart the Nudgis Worker service by using this command:

```
systemctl restart celerity-workers
```

2.4.3. Miris Manager commands

Restart the service

It is possible to restart the Miris Manager service by running:

```
systemctl restart skyreach
```

Dumping the database

It is possible to manually dump the Miris Manager database using :

```
service skyreach dump
```

Example:

```
$ service skyreach dump
```

```
Searching for old database dump to remove
Dumping database contents to
/home/skyreach/skyreach_data/private/dbdumps/backup.2016-09-08_15-51-
44.sql
```

Other dependencies

The Miris Manager also uses postgresql and nginx, so restarting them might be necessary as well.

```
systemctl restart postgresql
systemctl restart nginx
```

2.5. Stopping all services

All services should stop cleanly when pressing the power button (or running `shutdown -h`), but if you prefer to shut down each service separately, you can as well:

```
systemctl stop Nudgis
systemctl stop skyreach
systemctl stop msmonitor
systemctl stop postgresql
systemctl stop nginx
systemctl stop celerity-server
systemctl stop celerity-workers
```

3. Specific operations

3.1. Adding HDD space

3.1.1. No LVM

If LVM is not used, you have to add a new bigger hdd, create a new partition and move /home to it.

- Add the new disk
- Part it with fdisk/cfdisk or parted. Don't forget to set the type (Linux)
- Inform the system with partprobe

```
$ partprobe
```

- You should be able to see the new disk now with fdisk -l
- Create a FS

```
$ mkfs.ext4 /dev/sdaX
```

- Mount it on a temporary location

```
$ mount /dev/sdaX /media/
```

- Copy /home

```
$ rsync -avz /home/* /media/
```

- Umount /media

```
$ umount /media
```

- Change /etc/fstab by adding

```
/dev/sdaX /home    ext4    defaults      0       2
```

- Reboot

3.1.2. LVM

If LVM is used, things are a bit tricky but nicer.

Add a new disk

If your system is a VM, prefer the next method, expand.

- Add (physically or from your hypervisor) a disk, size isn't a problem since it will be added to the actual pool; to verify it's correctly installed run

```
fdisk -l
```

You should see it in the results (device /dev/sdxX)

- Part it and select the proper type

```
fdisk /dev/disk
```

or

```
cfdisk (type LVM : 8e)
```

- Extend LVM

```
vgextend [logical_volume] /new_partition
```

- Extend logical volume

```
lvextend -l +100%FREE /dev/volume/
```

- Extend file system

```
resize2fs /dev/volume/
```

Expand the initial disk

Only for VMs.

- Shutdown the VM and expand the virtual disk
- Boot the VM and resize the disk with cfdisk
    - delete the concerned partition (/dev/sdaX) - yes delete ! Don't worry, data still remains !
    - create it with the new size (/dev/sdaX)
    - write modifications (press w)
- Reboot the VM
- Resize the partition

```
pvresize -v -d /partition
```

- Extend logical volume

```
lvextend -l +100%FREE /dev/volume/
```

- Extend file system

```
resize2fs /dev/volume/
```

3.2. Triggering LDAP group synchronization

The LDAP group synchronization runs nightly (at 0:30) ; you can force a sync manually by running the following command:

```
$ sudo su msuser -c "python3 /usr/lib/python3/dist-packages/Nudgis/services/daemons/ldap_synchronizer.py start -n"
```

The log is here:

```
$ tail /home/msuser/mstmp/ldap_synchronizer.log
Synchronizing users.
Found 80 users.
Synchronizing groups.
Found 20 groups.
```

To enable the users synchronization, the following settings should be enabled: "Enable LDAP service", "Allow authentication" and "Synchronize each night".

To enable the groups synchronization, the following settings should be enabled: "Enable LDAP service", "Import users groups" and "Synchronize each night".

3.3. Changing domain names

To be able to change any application domain, you must ensure that your server is **up to date on the latest stable version**.
You also need to have the envsetup repository on the server.

Script usage:

```
/root/envsetup/tools/set_app_domain.py -h
USAGE: set_app_domain.py [-d] [-f] [-h] [app] <domain>
    -d: Debug mode (can be started with non root users).
    -f: Force mode (to force replacement of configuration even if
there are warnings).
    -h: Show this message.
    app: The application for which the new domain should be set.
         Possible values:
           "ms" (Nudgis), "mm" (Miris Manager), "mon" (Monitor).
         It is possible to specify which MS instance should be
targetted
         by using this format: ms-<instance name> (for example
ms-msuser).
    domain: The new domain.
```

For example for Nudgis:

```
/root/envsetup/tools/set_app_domain.py ms new.domain.com
```

3.4. Performing a manual health check

Nudgis comes with a built-in health check, that is executed every day (`/etc/cron.daily/Nudgis`) to ensure that the core features are not at risk. It is strongly advised, after any modification, to run a new check:

```
root@beta:~# /root/ubicast-tester/tester.py
--------------------------------
- UbiCast applications tester -
--------------------------------
Updating envsetup: Already up-to-date.
-- Test "test_email.py" --
Test start: 2017-06-14 07:35:45 UTC.
Postfix is listening port 25 correctly.
Checking if SMTP relay conforms to conf.
STMP relay is properly set.
Sending test email to "noreply+1497425745.9890726-312@ubicast.eu".
Waiting 118 seconds for email sending status.
Email sent.
Test end: 2017-06-14 07:35:48 UTC (duration: 0:00:02.083289).
-- Test "test_Nudgis Worker.py" --
Test start: 2017-06-14 07:35:48 UTC.
...
Expected NTP server ntp.debian.com found in configuration (total
servers: 5)
Test end: 2017-06-14 07:35:58 UTC (duration: 0:00:00.073414).

Tests results:
--------------------------------------------------
Test  Description
  Criticality  Result        Duration
--------------------------------------------------
```

```
test_email.py  Check that emails can be sent.
  High         success       0:00:02.083289
--------------------------------------------------
test_Nudgis Worker.py  Checks that Nudgis Worker can be reached using SSH and that it can reach the tasks server
  High         not testable  0:00:00.033271
...
```

If health check emails have been disabled due to too many consecutive failures, you can re-enable it by doing:

```
rm /root/envsetup/tests/logs/tests_history.txt
```

3.5. Trigger a calendar update

If the refresh interval is too long, you can force-trigger a calendar update by running:

```
$ su skyreach -c "python3 /home/skyreach/skyreach_site/daemons/stations_calendars_updater.py restart"
```

3.6. Setting the server timezone

```
timedatectl list-timezones
sudo timedatectl set-timezone desired_timezone
```

Also define the timezone in `/home/msuser/msinstance/conf/mssettings.py`:

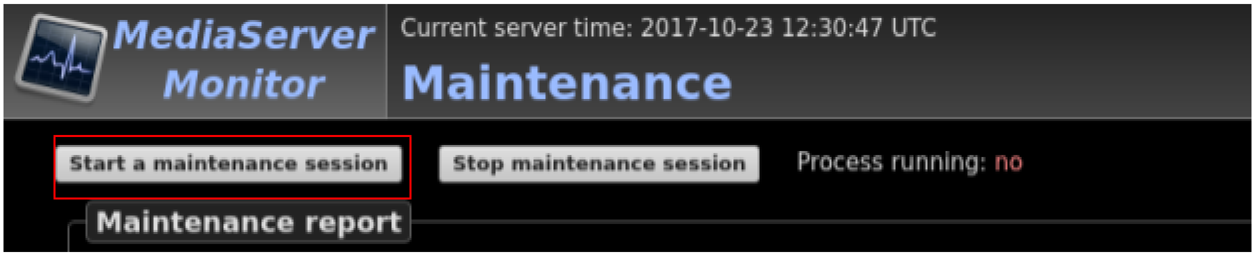
```
TIME_ZONE = 'Europe/London'
```

Then restart the service.

3.7. Starting an outbound maintenance session

Sometimes UbiCast will need to access your system remotely for advanced analysis; if you are not willing to open port 22 from the outside for UbiCast IP addresses, you can initiate a maintenance session by yourself (the only requirement is that the server can reach support2.ubicast.eu using ssh, i.e. tcp 22).

1) Connect to the included maintenance portal included in your Nudgis (usually, reachable over msmonitor.mydomain.net, see panel for the exact url or `/etc/nginx/sites-available/msmonitor.conf`)
2) open the Maintenance tab
3) start the tunnel (it should say “connection successful”)
4) notify UbiCast through panel.ubicast.eu on the related ticket



If the monitoring web interface is not available anymore on your server, you will have to log in in your server using SSH and start the following command:

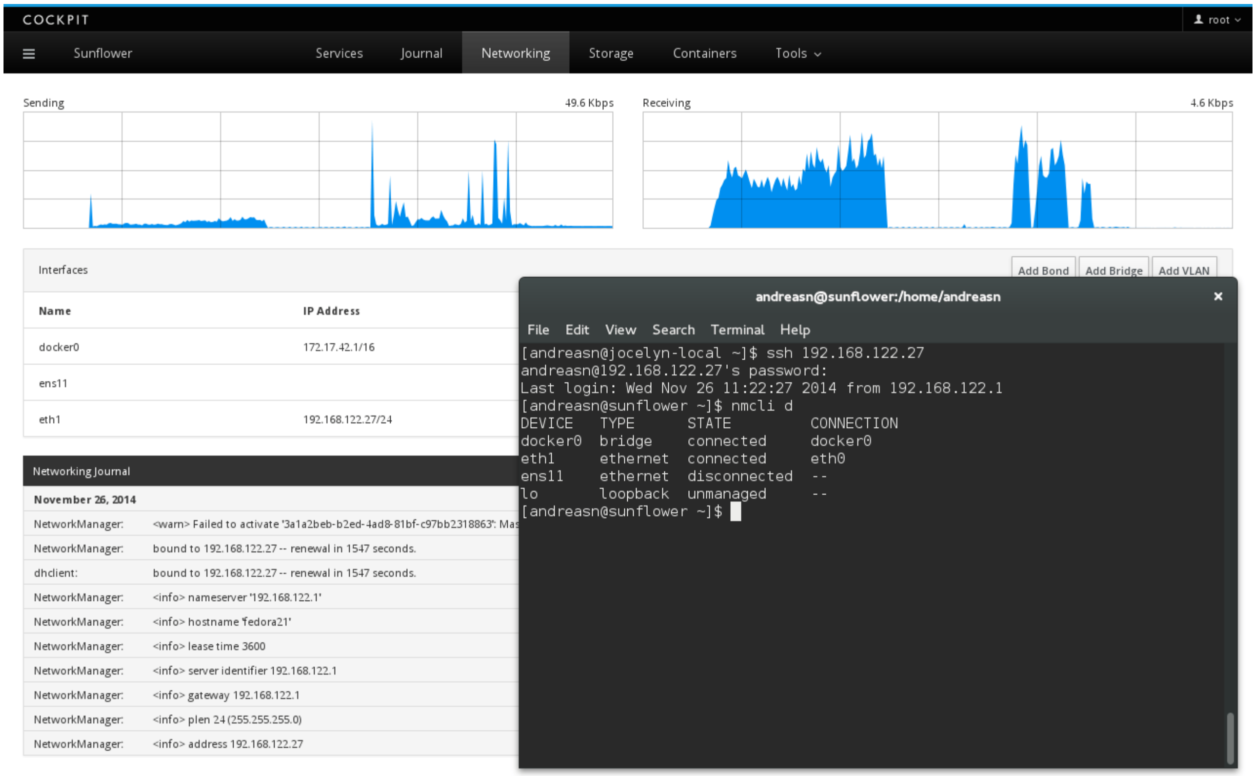
```
$ sudo su msmonitor -c 'python3
/home/msmonitor/msmonitor/daemons/ssh_maintenance.py restart'
```

You can find the credentials for the SSH access in https://panel.ubicast.eu.

A more detailed guide is available here.

3.8. Configuring the network

The physical servers sold by UbiCast are delivered with cockpit installed, which provides a web interface on port 9090: https://ip-address-of-machine:9090 ; cockpit provides a point-and-click web frontend to NetworkManager:



Interfaces

Name	IP Address
docker0	172.17.42.1/16
ens11	
eth1	192.168.122.27/24

Networking Journal

```
November 26, 2014
NetworkManager:   <warn> Failed to activate '3a1a2beb-b2ed-4ad8-81bf-c97bb2318863': Mas
NetworkManager:   bound to 192.168.122.27 -- renewal in 1547 seconds.
dhclient:         bound to 192.168.122.27 -- renewal in 1547 seconds.
NetworkManager:   <info> nameserver '192.168.122.1'
NetworkManager:   <info> hostname 'fedora21'
NetworkManager:   <info> lease time 3600
NetworkManager:   <info> server identifier 192.168.122.1
NetworkManager:   <info> gateway 192.168.122.1
NetworkManager:   <info> plen 24 (255.255.255.0)
NetworkManager:   <info> address 192.168.122.27
```

```
[andreasn@jocelyn-local ~]$ ssh 192.168.122.27
andreasn@192.168.122.27's password:
Last login: Wed Nov 26 11:22:27 2014 from 192.168.122.1
[andreasn@sunflower ~]$ nmcli d
DEVICE   TYPE      STATE         CONNECTION
docker0  bridge    connected     docker0
eth1     ethernet  connected     eth0
ens11    ethernet  disconnected  --
lo       loopback  unmanaged     --
[andreasn@sunflower ~]$
```

If you prefer using /etc/network/interfaces instead, port your configuration into /etc/network/interfaces, then disable NetworkManager with:

```
$ systemctl disable NetworkManager
$ systemctl stop NetworkManager
```

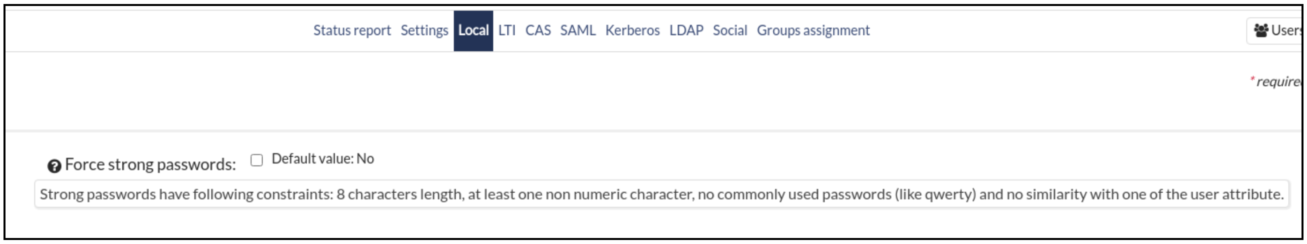
You can also disable Cockpit permanently with

```
$ systemctl disable cockpit
$ systemctl stop cockpit
```

3.9. Disabling password complexity policy

3.9.1. Nudgis

The setting to enable / disable password complexity policy is available in this page: administration > authentication settings and services > local (url: /authentication/services/local/)



Force strong passwords: □ Default value: No

Strong passwords have following constraints: 8 characters length, at least one non numeric character, no commonly used passwords (like qwerty) and no similarity with one of the user attribute.

3.9.2. Miris Manager

Edit the `/home/skyreach/skyreach_data/private/settings_override.py` file to include

```
AUTH_PASSWORD_VALIDATORS = []
```

Finish by restarting the service (here, Miris Manager):

```
systemctl restart skyreach
```

3.10. Disabling resources download protection

Edit Nginx configuration:

```
vi /etc/nginx/sites-enabled/mediaserver-msuser.conf
```

Define an empty secret:

```
server {
     listen 443 ssl http2;
     server_name nudgis;
     root /var/www/msuser;
```

```
	#set $secret "mysecret";
	set $secret "";
	include /etc/mediaserver/nginx/serve-media.conf;

	location / {
		uwsgi_pass unix:///home/msuser/mstmp/uwsgi.sock;
		include /etc/nginx/uwsgi_params;
	}
}
```

Restart Nginx:

```
nginx -t && systemctl restart nginx
```

It is not necessary to empty the secret key of the resources in the Nudgis player settings (/admin/player/).

3.11. Using fail2ban

Configuration options that can be used in `/root/envsetup/conf.sh` (adapt to your needs):

```
FAIL2BAN_ENABLED='1'
FAIL2BAN_SEND_EMAIL='0'  # set to 1 if you want to receive an email at each ban
FAIL2BAN_DEST_EMAIL='john.doe@example.net'
FAIL2BAN_MAXRETRY='6'
FAIL2BAN_BANTIME='30'
```

After that you can install fail2ban with this command:

```
python3 /root/envsetup/envsetup.py 28
```

You can see enabled jails with the command:

```
fail2ban-client status
```

And for a specific jail:

```
fail2ban-client status <jail-name>  # ex: sshd
```

To unban an IP:

```
fail2ban-client set <jail-name> unbanip <ip-address>
```

To disable fail2ban temporarily:

```
systemctl stop fail2ban
```

3.12. Resetting the admin account password

If you have lost the admin account password and if there is no email linked to the account to let you use the "forgotten password" procedure, you use the following commands to set a new password.

Nudgis

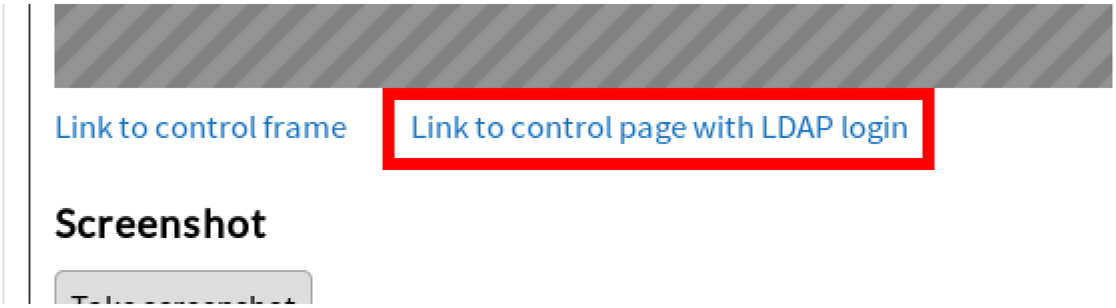
```
$ sudo su - msuser -c "python3 /home/msuser/msinstance/manage.py
changepassword admin"
```

Miris Manager

```
$ sudo su - skyreach -c "python3
/home/skyreach/skyreach_site/manage.py changepassword admin"
```

3.13. Configuring LDAP for the Miris Capture access in Miris Manager

The link to the control page can be found below the control interface of the system page in Miris Manager:



The LDAP settings for the control page can only be configured in the file system of Miris Manager.
The configuration should be added in
`/home/skyreach/skyreach_data/private/settings_override.py`

Example of minimal configuration:

```
AUTH_LDAP_SERVER_URI = 'ldaps://ldap.domain.com'
CONTROL_LDAP_USER_SEARCH_SCOPE = 'ou=People,dc=domain,dc=com'
CONTROL_LDAP_USER_SEARCH_FILTER =
'(&(objectClass=inetOrgPerson)(|(eduPersonAffiliation=staff)(eduPersonAffiliation=faculty)))'
CONTROL_LDAP_USER_ID_FIELD = 'uid'
CONTROL_LDAP_USER_EMAIL_FIELD = 'mail'
```

A more complete example with LDAP login allowed in Miris Manager:

```
AUTH_LDAP_ENABLED = True
AUTH_LDAP_AUTHENTICATION = True
AUTH_LDAP_SERVER_URI = 'ldaps://ldap.domain.com
AUTH_LDAP_START_TLS = False
AUTH_LDAP_USE_SASL = False
AUTH_LDAP_USER_SEARCH_SCOPE = 'ou=People,dc=domain,dc=com'
AUTH_LDAP_USER_ID_FIELD = 'uid'
AUTH_LDAP_BIND_DN = ''
AUTH_LDAP_BIND_PASSWORD = ''
CONTROL_LDAP_USER_SEARCH_SCOPE = 'ou=People,dc=domain,dc=com'
CONTROL_LDAP_USER_ID_FIELD = 'uid'
CONTROL_LDAP_USER_EMAIL_FIELD = 'mail'
CONTROL_ALLOWED_IPS = ('129.168.104.25', '192.168.0.0/23')
```

To allow access from all IP addresses, remove "CONTROL_ALLOWED_IPS" from your settings.

3.14. SSL certificates setup

Please follow instructions in this document :

https://docs.google.com/document/d/1XPJ7QmKwk3bi_4RjjT9Z4_Vz5NjZevigYyi_ztFZokU/edit

3.15. Configuring emails sending

Help on postfix SMTP relay configuration:
https://www.linode.com/docs/email/email-services/postfix-smtp-debian7/#configuring-the-relay-server

3.16. Software raid array management

Software RAID is handled by the mdadm service. If mdadm is not installed, it means that your server doesn't have software RAID.

3.16.1. Getting RAID status

To get the RAID status, run the following command:

```
$ cat /proc/mdstat
```

It will output something like this:

```
$ cat /proc/mdstat
Personalities : [raid6] [raid5] [raid4] [linear] [multipath] [raid0]
[raid1] [raid10]
md1 : active raid5 sda2[0] sdb2[1] sdc2[2]
     15987712 blocks super 1.2 level 5, 512k chunk, algorithm 2
[3/3] [UUU]

md0 : active raid5 sda3[0] sdb3[1] sdc3[2]
     7797771264 blocks super 1.2 level 5, 512k chunk, algorithm 2
[3/3] [UUU]

unused devices: <none>
```

If something is wrong, you will get something like:

```
$ cat /proc/mdstat
Personalities : [raid6] [raid5] [raid4] [linear] [multipath] [raid0]
[raid1] [raid10]
md1 : active raid5 sda2[0] sdb2[1] sdc2[2]
     15987712 blocks super 1.2 level 5, 512k chunk, algorithm 2
[3/3] [UUU]

md0 : active raid5 sdb3[1] sdc3[2]
     7797771264 blocks super 1.2 level 5, 512k chunk, algorithm 2
[3/2] [_UU]

unused devices: <none>
```

Note the content for md0: [_UU], it means that a disk is missing / defective. In this case, the RAID array is marked as degraded.

3.16.2. Repairing a degraded RAID array

To repair a degraded RAID array, you have to replace the defective disk with a new one with the same size and do the following steps:

A. Identify what disk is missing / broken

```
$ cat /proc/mdstat
Personalities : [raid6] [raid5] [raid4] [linear] [multipath] [raid0]
[raid1] [raid10]
md1 : active raid5 sda2[0] sdb2[1] sdc2[2]
     15987712 blocks super 1.2 level 5, 512k chunk, algorithm 2
[3/3] [UUU]

md0 : active raid5 sdb3[1] sdc3[2]
     7797771264 blocks super 1.2 level 5, 512k chunk, algorithm 2
[3/2] [_UU]

unused devices: <none>
```

In this case, the disk /dev/sda is broken (the partition /dev/sda3 is missing from md0).

B. Replace the defective disk

With the above, example /dev/sda should be replaced (disk in slot 0).

C. Create the same partition table as in other disks used in the RAID

To do that, you can either copy the old disk content with dd in the new disk if it was not entirely defective, else you should use tools like parted or fdisk to do that.

The dd command is:

```
$ dd if=/dev/sdX of=/dev/sdX bs=10M conv=noerror,sync
```

if means input (the defective disk) and of output (the new disk).

D. Rebuild the RAID with the new disk

With the same example as above, sda3 needs to be added to md0:

```
$ mdadm --manage /dev/md0 --add /dev/sda3
mdadm: added /dev/sda3
```

The RAID status will then show the rebuild progress:

```
$ cat /proc/mdstat
Personalities : [raid6] [raid5] [raid4] [linear] [multipath] [raid0]
[raid1] [raid10]
md1 : active raid5 sda2[0] sdb2[1] sdc2[2]
     15987712 blocks super 1.2 level 5, 512k chunk, algorithm 2
[3/3] [UUU]

md0 : active raid5 sda3[3] sdb3[1] sdc3[2]
     7797771264 blocks super 1.2 level 5, 512k chunk, algorithm 2
[3/2] [_UU]
     [>....................]  recovery =  0.0% (355684/3898885632)
finish=365.3min speed=177842K/sec

unused devices: <none>
```

3.17. Anonymize statistics

If you don't want your Nudgis to store user accounts in statistics data, you will have to disable the setting "Store user in statistics" in "admin" → "authentication settings".
This setting is available in the authentication settings frontend since Nudgis 9.6.0 (or manually by adding STATS_STORE_USER = False in mssettings.py).

If you already have statistics data in your Nudgis, you will have to anonymize them by running the following commands (for instance "msuser"):

```
$ runuser -u msuser -- python3 /home/msuser/msinstance/manage.py
shell --command='from Nudgis.statistics.models import
VideoOnDemandStatEntry as ve, LiveSessionStatEntry as le;
ve.objects.all().update(user_id=0);
le.objects.all().update(user_id=0)'

$ sed -i 's/user_id = [0-9]*/user_id = 0/'
/home/msuser/msinstance/stats/*/*/*/*
```

3.18. Maximizing Nudgis Worker load

Nudgis Workers are supposed to use as much resources as possible (i.e. load should peak at ~ thread count), to increase the number of concurrent processes, increase the QUEUES_PER_WORKER number in /etc/celerity/config.py **on each Nudgis Worker** and restart the service.

```
$ cat /etc/celerity/config.py
QUEUES_PER_WORKER = 2
$ systemctl restart celerity-workers.service
```

After increasing this, the average load should increase. Do not increase too much or processing delays will increase.

4. Monitoring

You can consult the specific documentation about monitoring. It lets you know whose ports/services to monitor.

5. Installation

While in most cases UbiCast provides deployment services, we also allow third parties to deploy the service autonomously. **This relies on an activation key that is provided by UbiCast** after reception of the Pre-deployment form which contains all technical information required for the deployment.

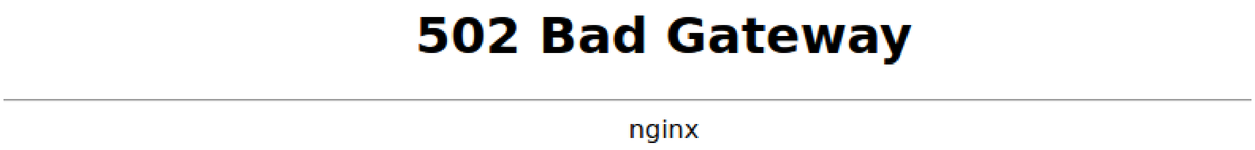
The following instructions describe how to achieve the deployment on an Debian server deployed according to the deployment instructions described in the Server Deployment Requirements documentation.

The installation instructions are detailed in the git repository documentation.

6. Troubleshooting

6.1. 502 Bad Gateway

If the Nudgis or Miris Manager display the following page, restart the services



502 Bad Gateway

nginx

Nudgis:

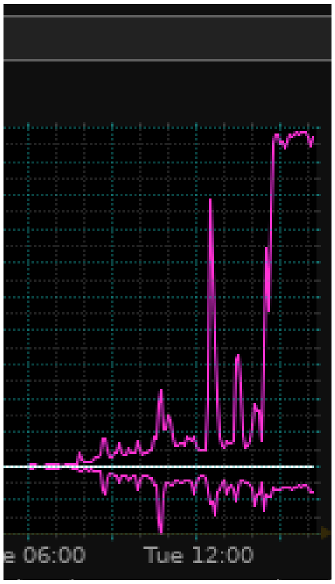
```
systemctl restart Nudgis
```

Miris Manager:

```
systemctl restart skyreach
```

6.2. Live streaming hiccups / interruptions

Check the bandwidth on the Nudgis interfaces, you may have hit the network bottleneck. This is the case if the bandwidth tops at a fixed value (e.g. 1 Gbps).



In the graph above, the network seems to be the limiting factor.

To improve this you can

- either upgrade the bandwidth of your server
- reduce the bitrate of your streams
- setup a CDN for live streams by adding a line to /home/msuser/msinstance/conf/lives.json (restart the services on the servers after change)

```
RTMP_HLS_PLAYBACK_URL =
'https://mycdn.com/%(rtmp_app)s/%(stream_id)s/chunklist.m3u8
```

6.3. PostgreSQL

6.3.1 Errors detected with and high availability database cluster

When the databases are deployed in a high availability setup (two databases balanced through a HAProxy service installed on the Nudgis frontend), if there is a failure, it is advised to not rush things and to not reboot the database servers and services.
If the database cluster is failing despite the high availability, it is unlikely due to a failure of the balancing mechanism. Restarting the database servers or services can lead to data corruptions and greatly complexify the debugging.

6.3.2 Investigate running queries

In postgreSQL, statistics on the currently running queries are logged by default in the pg_stat_activity table.
One of the columns is showing the SQL query that is being run, but by default, only 1024 bytes of the queries are shown. This can cause some queries from the application to be truncated.
To extend the length of such queries, you can change the “track_activity_query_size” parameter in “/etc/postgresql/11/main/postgresql.conf” from the default 1024 value to 16384.
You will then need to restart the postgresql service.

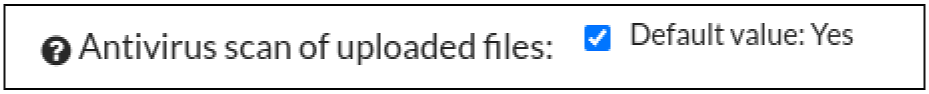
6.4. Nudgis Cache

6.4.1. nginx not caching at all

Ensure that all servers are NTP synchronized, otherwise nginx might not cache at all.

6.5. Antivirus

NB: note that the antivirus can be disabled using the UI in Nudgis at /admin/settings/ and in MirisManager too.



Antivirus scan of uploaded files: ☑ Default value: Yes

6.5.1. Handling false positives

It may happen that some valid files are detected by the antivirus (clamav), hence rejected by the API; these are called false positives.

```
# clamscan --max-filesize=500M --max-scansize=500M myfile
/root/myfile: Win.Virus.Triusor-9950253-0 FOUND

----------- SCAN SUMMARY -----------
Known viruses: 8616509
Engine version: 0.103.5
Scanned directories: 0
Scanned files: 1
Infected files: 1
Data scanned: 467.58 MB
Data read: 437.47 MB (ratio 1.07:1)
Time: 41.846 sec (0 m 41 s)
Start Date: 2022:05:18 14:09:04
End Date:   2022:05:18 14:09:46
```

After ensuring that other antivirus do not detect the virus, to allow this file, you need to add it’s hash into a false positive file:

```
sigtool --sha256 myfile >> /var/lib/clamav/false-positives.sfp
```

You can check that it is now being ignored:

```
$ clamscan --max-filesize=3000M --max-scansize=3000M myfile
/root/myfile: OK

----------- SCAN SUMMARY -----------
Known viruses: 8616510
Engine version: 0.103.5
Scanned directories: 0
Scanned files: 1
Infected files: 0
Data scanned: 0.00 MB
Data read: 2207.28 MB (ratio 0.00:1)
Time: 18.227 sec (0 m 18 s)
Start Date: 2022:05:18 14:07:37
End Date:   2022:05:18 14:07:55
```