SmartMEDIA. Installation guide (trial version)

1 General

The current manual describes installation procedures of the trial version of the SmartMEDIA system and its software. All system services are adjusted for an easy start, requiring no special knowledge of the product.

Important: all the commands given in the manual should be executed using authorization of a super-user.

Following the system installation:

> ▷ Packages and their associated software will be installed. The trial version includes a set of packages:
>
>   o smartmedia-recorder
>
>   o smartmedia-fasthls
>
>   o smartmedia-utils
>
>   o smartmedia-nginx
>
> ▷ MongoDB database required for the system operation will be installed.
>
> ▷ `/video`, directory will be created together with `/video/pvr`. The `/video/pvr` directory is required for the functioning of the SmartMEDIA system. Hence, if in the future a storage is added to `/video`, you will have to manually create `/video/pvr` in it and change its owner, executing the following commands:

```
mkdir -p /video/pvr      chown -R smartmedia /video
```

> ▷ Time will be synced via the NTP Time Protocol from the Internet resource clock.isc.org and the following services will be started:

```
ntpd
mongod
smartmedia-nginx
smartRecord
fastHLS
```

2 Installation and Setting of SmartMEDIA

Note: To eliminate errors which can occur in case of incorrect settings of Firewall/iptables and SELinux rules, it is recommended to turn off Firewall and SELinux before the first start of SmartMEDIA. After making the settings of Smartmedia services you need to set up Firewall/iptables and SELinux rules and turn them on, controlling the work of SmartMEDIA.

2.1 Installation Start

CentOS 6

```
service iptables stop

chkconfig iptables off
```

```
setenforce 0

rpm -ihv http://smartlabs-
public:6vTh5WVrQlgF@repo.smartlabs.tv/repo/public/centos/6/x86_64/smartlabs-
repo-1.0-1.el6.x86_64.rpm
```

yum install smartmedia

CentOS 7

```
systemctl stop firewalld

systemctl disable firewalld

setenforce 0

rpm -ihv http://smartlabs-
public:6vTh5WVrQlgF@repo.smartlabs.tv/repo/public/centos/7/x86_64/smartlabs-
repo-1.0-1.el7.centos.x86_64.rpm
```

yum install smartmedia

2.2 License Obtaining and Installation

In case you don't own a SmartMEDIA license file, after the system installation, smartRecord and fastHLS services will start and end with an error. Error information is available in the service log (see *Service Log* section):

```
[error]        [default]        Can't        read        license        file
"/opt/smartlabs/smartmedia/license.bin",        error=No        such        file        or
directory[info] [default] smartRecord license verification failed, exit
```

To add a license:

1. Send system UUID to the following email address: support@smartlabs.tv. It may be viewed in any service log after a service start.

For instance:

```
[2017-06-01  13:15:32.687637]  [0x00007f0c77d7a777]  [info]  [default]  System
UUID:  00000000-0000-0000-0000-0011101DD11
```

2. Receive a license file from SmartLabs.

3. Copy the received file to `/opt/smartlabs/smartmedia/license.bin` on the server.

2.3 Service Start

After installing the license, start SmartMEDIA services.

CentOS 6

```
service smartRecord start

service fastHLS start
```

CentOS 7

```
systemctl reset-failed

systemctl start smartRecord

systemctl start fastHLS
```

Note. After server reboot the services will start automatically.

2.3.1 Service Logs

```
/var/log/smartmedia/fastHLS

/var/log/smartmedia/smartRecord

/var/log/smartmedia/smartmedia-nginx
```

2.4 Turn on Firewall and SELinux

In case of service starting without error and turning off Firewall/SELinux previously you can turn them on again. Set Firewall/SELinux rules in the following way:

> ▷ Enable incoming queries (required for nginx service, which will be installed and set for listening at Port 80 in the course of SmartMEDIA installation).

> ▷ Enable incoming multicast-streams.

If after Firewall/SELinux turning on the SmartMEDIA services fail to start, it indicates to the incorrect setting up of rules.

2.5 Channel Source Indication

After installing SmartMEDIA, you need to set source addresses of channels that should be recorded on a video server. This can be done in a configuration file of smartmedia-recorder package:

```
/opt/smartlabs/smartmedia/smartRecord/config/smartRecordChannels.json
```

By default this file contains sample addresses:

> ▷ For a channel broadcast using adaptive bitrate technology. The stream will be saved in three different bitrates:
> 239.65.40.1:5001 (битрейт 1758000 bps)
> 239.65.41.1:5002 (битрейт 1002000 bps)
> 239.65.42.1:5003 (битрейт 462000 bps)

> ▷ For a channel broadcast in a single bitrate:
> 239.65.40.142:5001 (bitrate 1758000 bps)

SmartMEDIA takes streams from addresses indicated in the file and saves them on a video server twice:

1. First time – in the TS container for further preparation of the HLS content.
2. Second time – in the MP4 container for DASH and HLS protocols with an MP4 container.

By default the recorded content is stored for four hours. This value is indicated in the configuration file `/opt/smartlabs/smartmedia/smartRecord/config/config.json`. You may change the storage time depending on available content space and the required DVR window.

More details on the configuration file and setting of the file with channels are given in the "SmartRecord" section in the "SmartMEDIA. User guide".

Adding Channels to File

Add the required channels to the file in accordance with the example. If necessary, you may edit the list of channels and sources at any time. The changes are applied automatically. You don't have to restart the service. Channel update process can be monitored via the smartRecord log.

3 Installation and Use of ffmpeg

ffmpeg is a free tool enabling channel broadcast via adaptive broadcast technology. The repository contains packages and scripts developed by Smartlabs to simplify the start and transcoder setting with the help of ffmpeg.

Note: the current section describes the basic principles of working with ffmpeg. More detailed information is given here: https://www.ffmpeg.org/ffmpeg.html.

To install ffmpeg and its associated features, execute the following command:

```
yum install ffmpeg-transcoder
```

After the installation is complete, the following features will be created:

- ▷ start script `/etc/init.d/ffmpeg`

- ▷ directory with stream settings `/etc/ffmpeg-channels.d/`. It includes the preset test file STREAM1 for transcoding one stream in three bitrates with scaling and deinterlacing.

Control Commands
Start:

```
/etc/init.d/ffmpeg start
```

Status Check:

```
/etc/init.d/ffmpeg status
```

Pause:

```
/etc/init.d/ffmpeg stop
```

Transcoder Parameters

You may set the following transcoding parameters in the `STREAM1` test file:

Parameter	Description
CHANNEL_input_mcast	Source stream multicast-group address
CHANNEL_input_port	Source stream port
CHANNEL_input_filter	Stream filter
CHANNEL_input_opts	Additional input options
CHANNEL_output_count	Output stream number
CHANNEL_outputN	Output stream multicast-group address having an N number
CHANNEL_output_portN	Output stream port of an N number
CHANNEL_setN	Parameters of an output stream port of an N number

`CHANNEL_input_opts`

To transcode one program from MPTS-stream you need to indicate corresponding settings in the `CHANNEL_input_opts` variable. For example, to select a program 1010:

`CHANNEL_input_opts="-map 0:p:1010 -ignore_unknown"`

`CHANNEL_setN`

`CHANNEL_setN` is a set of parameters for a specific profile (each profile has its own corresponding bitrate) of an output stream, where N is the profile number.

For example:

```
-map [r1] -map 0:a -acodec copy -vcodec libx264 -sc_threshold 0 -g 75 -
keyint_min 75 -f mpegts -b:v 1500k -maxrate 1500k -bufsize 7500k -crf 18 -f
mpegts
```

This means that all audio paths (`-map 0:a`) of the output stream [`r1`] will be taken and the following parameters will be set:

- ▷ codec for a video path - libx264,
- ▷ number of stills to be followed by a key still - 75 stills (-g 75 -keyint_min 75),
- ▷ bitrates (-b:v 1500k -maxrate 1500k),
- ▷ VBV-buffer (-bufsize 7500k).

Transcoder settings

Conversions in the course of transcoding are defined by an integrated filter.

By default the test file includes the following filter:

```
[0:v]yadif=0,split=3[r1][pr2][pr3];[pr2]scale=640:trunc(ow/a/2)*2[r2];[pr3]sc
ale=320:trunc(ow/a/2)*2[r3], herein:
```

Designation in filter	Conversion
`[0:v]`	Use video path 0
`yadif=0`	Apply `yadif` filter for deinterlacing, reducing the still frequency by half
`split=3[r1][pr2][pr3]`	Divide the stream into three. [ ] contains the names of the output paths
`[pr2]scale=640:trunc(ow/a/2)*2[r2]`	Apply the scale filter to path [`pr2`] to scale the still along the width of 640 points, retaining its proportions. Name the scaled stream [`r2`]
`[pr3]scale=320:trunc(ow/a/2)*2[r3]`	Apply the scale filter to path [`pr3`] to scale the still along the width of 320 points, retaining its proportions. Name the scaled stream [`r3`]

System Log

All ffmpeg messages of the warning level and above are entered in the system log.

To view the log, execute the following command:

```
journalctl -xe
```

4   Utilities

The repository contains the multicat and ssmping packages with utilities multicat and mcfirst, correspondingly.

With the help of the multicat utility you may:

- save a multicast-stream to file,
- broadcast a stream to another address.

With the help of the mcfirst utility you may:

- Check availability of the multicast-stream
- Evaluate the bitrate.

To install utilities execute the following command:

```
yum install ssmping multicat
```

Examples

multicat

Save a stream to file:

```
multicat -u -U @239.39.39.39:1234 out.ts
```

Broadcast to another address:

```
multicat -u -U @239.39.39.39:1234 239.39.49.1:1234
```

mcfirst

Check stream availability:

```
mcfirst 239.39.49.1 1234
```

Evaluate bitrate in five seconds:

```
mcfirst -t 5 239.39.49.1 1234
```

5 Useful Information

5.1 Rotation of SmartMedia Service Logs

Compression and deletion of old logs are scheduled with the help of crond. The schedule is kept in /etc/cron.d/smartmedia-tasks in the following way:

```
9  1 * * *           root find /var/log/smartmedia/ -mtime +14 -type f -exec
rm {} \;

12 */4  * * *        root find /var/log/smartmedia/smartRecord/ -name
"smartRecord*.log" -a -not -name "*.gz" -mtime +1 -exec gzip {} \;

15 */4  * * *        root find /var/log/smartmedia/fastHLS/ -name
"fastHLS*.log" -a -not -name "*.gz" -mtime +1 -exec gzip {} \;
```

If necessary, you may change the storage period and the schedule start.

5.2 Index Creation in MongoDB

```
db.mp4_index.createIndex( { npvr_id: 1, npt_usec: 1, npt_end_usec: 1}, {
background: true } )

db.ts_index.createIndex( { npvr_id: 1, npt_usec: 1, npt_end_usec: 1}, {
background: true } )

db.mp4_index.createIndex( { header_id: 1}, { background: true } )

db.ts_index.createIndex( { header_id: 1}, { background: true } )
```

5.3 Rotation of MongoDB Logs

Rotation of `/var/log/mongodb/mongod.log` is performed on a daily basis with the help of the (`/etc/cron.daily/mongorotate`) script.

```sh
#!/bin/sh



MAXAGE=14

kill -USR1 $(cat /var/run/mongodb/mongod.pid)

find  /var/log/mongodb  -name  "mongod.log.????-??-??T??-??-??"  |  xargs  -n  1
gzip

find /var/log/mongodb -type f -name *.gz -mtime +$MAXAGE -delete
```

If necessary you may change duration of the log storage.

5.4 Restriction on MongoDB Cash Memory

By default MongoDB allocates half of the free memory at MongoDB start time, and at least 256 MB for cash.

If necessary you may change the cash size by adding the following lines to the `storage` section of the `/etc/mongod.conf file`:

```
 wiredTiger:

   engineConfig:

     cacheSizeGB: [cache size]
```