Background

Elastifile releases a new version every couple of weeks.
The Elastifile version includes customers' issues fixes, bug fixes, security improvements and more.
The Elastifile systems should be upgraded to a new version for a better experience and support.

Objective

This procedure provides step-by-step guidance for the Elastifile customers on how to upgrade an Elastifile system on GCP Marketplace to GA version **3.2.1.51**.

Note-
The procedure should not be executed without GCP support accompanied.

Overview

Elastifile offers a non-disruptive software upgrade to systems run within GCP.
The upgrade flow contains 2 main phases:

1. In the first phase the following components are upgraded:
   - Emanage service
   - ECS service
   - ECP service
   - Replication Agents
   - EPA services on all storage nodes

2. In the second phase, the cluster is doubled in the number of nodes.
   The new nodes run the new target version and will be attached to the system in addition to the existing nodes.
   Once the cluster contains both old and new nodes, data rebuild will take place and replicate the data from the old nodes to the new nodes.
   On completion, the old nodes will be removed from the system.
   As a last phase, the working version will be flipped into the new target version.

Preparations

- The upgrade requires SSH and HTTPS access to the Elastifile Management Server.
- The service account who runs the EMS should have the right permission for creating new instances in the project and network routes in the VPC.
- Make sure there are enough available quota resources for creating the new Elastifile Storage nodes as the number of the existing ones.

Detailed Procedure

Upgrade PreChecks

1. Login by SSH to the Elastifile Management Server

   a. Prepare the script environment:

```
sudo su -
source /root/elfs_admin
cd /tmp/
wget https://storage.googleapis.com/elastifile-software-repo/scripts/ecfs_health_check.tar.gz
tar xvfz ecfs_health_check.tar.gz
read -s password
<PROVIDE_PASSWORD>
export CREDENTIALS="admin:$password"
```

2. Execute the script and check all outputs are OK:

Note- If preferred/ needed, you can use the manual way of running the prechecks.
Refer to [Appendix A](#).

```
python ecfs_health_check.py -p $password -v 3.2.1.51
```

Upgrade Execution

1. If you used the '-v' option in the prechecks script, you can skip to [step 4](#)
   Download upgrade package from GCP bucket to EMS

```
wget -P /tmp/ https://storage.googleapis.com/elastifile-software-repo/ECFS/3.2.1.51/elastifile-upgrade-3.2.1.51-64365.b02d574e6085.tar


[1/1 files][  1.4 GiB/  1.4 GiB] 100% Done  81.7 MiB/s ETA 00:00:00
Operation completed over 1 objects/1.4 GiB.
```

2. Upload software package (takes a minute)

```
elfs-cli sw_package upload --content /tmp/elastifile-upgrade-3.2.1.51-64365.b02d574e6085.tar

status:  success
```

3. Verify package software version

```
elfs-cli sw_package info

version:            3.2.1.51
release_timestamp:    Thu Jan 28 23:41:57 UTC 2021
filename:            elastifile-3.2.1.51-64365.b02d574e6085.tar
compatibility:        2.0,2.1,2.5,2.7,3.0,3.1,3.2,4.0
package_type:        version
sha1sum_from_tarfile:  a6ae23b5d2cb2e931080fefdfc55a2a4dbae33e9
sha1:            a6ae23b5d2cb2e931080fefdfc55a2a4dbae33e9
verify_status:        success
```

4. Increase the verify_nodes_removed timeout

```
wget -P /elastifile/bin/upgrade/ https://storage.googleapis.com/elastifile-software-repo/scripts/ndu_after_ems_hook.sh
sh /elastifile/bin/upgrade/ndu_after_ems_hook.sh

patching file emanage/app/jobs/cloud_ndu_job.rb
```

5. Execute the upgrade

```
elfs-cli system upgrade_start --type=rolling --admin-passwd=$password --id=1

id:          18
uuid:        7028e4eb-be95-4311-bc19-6afa758a7942
handler:      rolling
created_at:    May 21, 13:59:20
updated_at:    May 21, 13:59:20
task_type:    upgrade_task
```

Upgrade Monitoring

1st phase

The first phase lasts less than 10 minutes and can be monitored through the following elfs-cli command:

- The TASK_ID can be taken from the output of step 5 in the 'Detailed Procedure' section

- Status **'success'** indicates this phase is completed successfully

- The **current_step** will have the following evolution:
    1. Preparing
    2. Check compatibility
    3. Update active emanage
    4. [Update replication agents]
    5. Update enodes
    6. Post install actions
    7. Pass task to ecs

- During the emanage activation the command can be broken due to a service restart, just wait

```
watch -n 2 'elfs-cli control_task show --id <TASK_ID>'

id:              18
uuid:            7028e4eb-be95-4311-bc19-6afa758a7942
status:          success
name:            rolling
current_step:    Pass task to ecs
step_progress:   10
step_total:      10
created_at:      May 21, 13:59:20
updated_at:      May 21, 14:04:12
task_type:       upgrade_task
```

- If the task failed, follow the steps here

Note-
There is an expected version mismatch once the first phase is done, and will be last until the second phase will be completed as well:

```
root@ndu ~# ecs-cli system | grep ver

ecs-version: 3.2.1.51-64365.b02d574e6085
working-ver: 3.2.1.39-63705.b8f33a1ebe28
```

2nd phase

Note- If preferred, this phase can be monitored through the Elastifile GUI.
Refer to Appendix B.

1. Validating new instances are created:

```
watch -n 2 'gcloud compute instances list | grep $(hostname)-elfs'

Before- Existing nodes are RUNNING:
Every 2.0s: gcloud compute instances list | grep $(hostname)-elfs          Thu May 21 14:05:13 2020

ndu-elfs-69478232      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.15.237      RUNNING
ndu-elfs-784e1ba1      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.0.21        RUNNING
ndu-elfs-e067259c      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.0.19        RUNNING

During- Existing nodes are RUNNING, new nodes are STAGING:
Every 2.0s: gcloud compute instances list | grep $(hostname)-elfs          Thu May 21 14:06:04 2020

ndu-elfs-69478232      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.15.237      RUNNING
ndu-elfs-73a6a78b      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.15.238      STAGING
ndu-elfs-784e1ba1      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.0.21        RUNNING
ndu-elfs-7ae52baf      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.0.8         STAGING
ndu-elfs-e067259c      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.0.19        RUNNING
ndu-elfs-ec62c063      europe-west4-a      custom (4 vCPU, 32.00 GiB       10.164.0.9         STAGING

After- All nodes are RUNNING:
Every 2.0s: gcloud compute instances list | grep $(hostname)-elfs          Thu May 21 14:07:08 2020

ndu-elfs-69478232      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.15.237      RUNNING
ndu-elfs-73a6a78b      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.15.238      RUNNING
ndu-elfs-784e1ba1      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.0.21        RUNNING
ndu-elfs-7ae52baf      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.0.8         RUNNING
ndu-elfs-e067259c      europe-west4-a      custom (4 vCPU, 32.00 GiB)      10.164.0.19        RUNNING
ndu-elfs-ec62c063      europe-west4-a      custom (4 vCPU, 32.00 GiB       10.164.0.9         RUNNING
```

2. Validating new instances are attached to the cluster:

```
watch -n 2 'ecs-cli nodes'

Before- Existing nodes are RUNNING:
Every 2.0s: ecs-cli nodes                                            Thu May 21 14:07:32 2020

[2] vHead-10.164.0.19 10.164.0.19 EOid[ENODE:a6e31627-6af4-4e9e-b76f:2] ENODE_RUNNING VALID    ECDB
[3] vHead-10.164.0.21 10.164.0.21 EOid[ENODE:2726b5ff-f634-4855-8330:3] ENODE_RUNNING VALID    ECDB SO
[4] vHead-10.164.15.237 10.164.15.237 EOid[ENODE:0e5e293a-0dbd-42a4-9f9c:4] ENODE_RUNNING VALID    ECDB

During #1- Existing nodes are RUNNING, new nodes are CONFIGURED:
Every 2.0s: ecs-cli nodes                                            Thu May 21 14:08:04 2020

[2] vHead-10.164.0.19 10.164.0.19 EOid[ENODE:a6e31627-6af4-4e9e-b76f:2] ENODE_RUNNING VALID   ECDB
[3] vHead-10.164.0.21 10.164.0.21 EOid[ENODE:2726b5ff-f634-4855:3] ENODE_RUNNING VALID  FD: 1    ECDB SO
[4] vHead-10.164.15.237 10.164.15.237 EOid[ENODE:0e5e293a-0dbd-42a4:4] ENODE_RUNNING VALID  FD: 1    ECDB
[5] vHead-10.164.0.8 10.164.0.8 EOid[ENODE:67210951-c1dd-481e-b4ca:5] ENODE_CONFIGURED VALID
[6] vHead-10.164.0.9 10.164.0.9 EOid[ENODE:3019eb52-e77a-4cec-b58d:6] ENODE_CONFIGURED VALID
[7] vHead-10.164.15.238 10.164.15.238 EOid[ENODE:ba5f6a8d-f968-4529-9862:7] ENODE_CONFIGURED VALID
During #2- All nodes are RUNNING:
```

```
Every 2.0s: ecs-cli nodes                                          Thu May 21 14:08:22 2020

[2] vHead-10.164.0.19 10.164.0.19 EOid[ENODE:a6e31627-6af4-4e9e-b76f:2] ENODE_RUNNING VALID    ECDB
[3] vHead-10.164.0.21 10.164.0.21 EOid[ENODE:2726b5ff-f634-4855-8330:3] ENODE_RUNNING VALID    ECDB SO
[4] vHead-10.164.15.237 10.164.15.237 EOid[ENODE:0e5e293a-0dbd-42a4-9f9c:4] ENODE_RUNNING VALID    ECDB
[5] vHead-10.164.0.8 10.164.0.8 EOid[ENODE:67210951-c1dd-481e-b4ca:5] ENODE_RUNNING VALID
[6] vHead-10.164.0.9 10.164.0.9 EOid[ENODE:3019eb52-e77a-4cec-b58d:6] ENODE_RUNNING VALID
[7] vHead-10.164.15.238 10.164.15.238 EOid[ENODE:ba5f6a8d-f968-4529-9862:7] ENODE_RUNNING VALID

After- old nodes status is changing from VALID to REMOVING:
Every 2.0s: ecs-cli nodes                                          Thu May 21 14:09:41 2020

[2] vHead-10.164.0.19 10.164.0.19 EOid[ENODE:a6e31627-6af4-49e6f:2] ENODE_RUNNING REMOVING    ECDB
[3] vHead-10.164.0.21 10.164.0.21 EOid[ENODE:2726b5ff-f634-48530:3] ENODE_RUNNING REMOVING    ECDB SO
[4] vHead-10.164.15.237 10.164.15.237 EOid[ENODE:0e5e293a-0dba4-9f9c:4] ENODE_RUNNING REMOVING    ECDB
[5] vHead-10.164.0.8 10.164.0.8 EOid[ENODE:67210951-c1dd1e-b4ca:5] ENODE_RUNNING VALID    ECDB
[6] vHead-10.164.0.9 10.164.0.9 EOid[ENODE:3019eb52-e77aec-b58d:6] ENODE_RUNNING VALID    ECDB
[7] vHead-10.164.15.238 10.164.15.238 EOid[ENODE:ba5f6a8d68-4529-9862:7] ENODE_RUNNING VALID    ECDB
```

3. Validating Data Rebuild (ROC transitions are in progress) is running

```
watch -n 2 'ecs-cli ha-report'

Before- ORC transitions are in progress:
Every 2.0s: ecs-cli ha-report                                      Thu May 21 14:09:56 2020

ownership-recovery: NOT_IN_RECOVERY total:0 done:0 id:0
roc-transitions: total:0 done:0 inprogress:0 rebuildid:0
roc-transitions: progress:<> src:<> dst:<>
orc-transitions: total:1 done:0 inprogress:1 rebuildid:2
orc-transitions: progress:<359073/535758B> src:<> dst:<>
ecdb-transitions: total:0 done:0 inprogress:0 rebuildid:0
ecdb-transitions: progress:<> src:<> dst:<>
ROC map calculated with keeping Failure Domains constraints: true
FD Eoid: EOid[FAILURE_DOMAIN:eea703cd-740c-4717-bdd2:1], count of running nodes: 6

During- ROC transitions are in progress (Data rebuild is RUNNING):
Every 2.0s: ecs-cli ha-report                                      Thu May 21 14:10:50 2020

ownership-recovery: NOT_IN_RECOVERY total:0 done:0 id:0
roc-transitions: total:1398 done:1 inprogress:45 rebuildid:5
roc-transitions: progress:<5B 23352/34675B 20950/34675B 20827/34675B 21934/34675B
22646/34675B 22546/34675B 21964/34675B 22492/34675B 19878/34675B 23741/34675B
23456/34675B 34675/34675B 22123/34675B 23078/34675B 23781/34675B 22734/34675B
22461/34675B 19794/34675B 24444/34675B 27143/34675B 24262/34675B 23434/34675B
19432/34675B 21584/34675B 22143/34675B 21536/34675B 25325/34675B 21786/34675B
22405/34675B 22726/34675B 21976/34675B 20170/34675B 20958/34675B 24560/34675B>
src:<> dst:<>
orc-transitions: total:0 done:0 inprogress:0 rebuildid:0
orc-transitions: progress:<> src:<> dst:<>
ecdb-transitions: total:0 done:0 inprogress:0 rebuildid:0
ecdb-transitions: progress:<> src:<> dst:<>
ROC map calculated with keeping Failure Domains constraints: true
FD Eoid: EOid[FAILURE_DOMAIN:eea703cd-740c-4717:1], count of running nodes: 6
```

4. When Data Rebuild is running based on the previous step,
   the session with GCP Elastifile Support can be wrapped up.
   The following outputs should be collected:
   - 'ecs-cli ha-report' output, indicating data rebuild is running
   - Hourly updates on the data rebuild progress.
     Can be monitored easily using the GUI.

5. Once data rebuild is completed, and old nodes are deleted,
   proceed to the process validation task completion.

Process Validation

Please execute the Elastifile Health Check script for general health status and NDU task completion.
The script should be executed after the old nodes are removed from the system and the task from the 1st phase is not 'running' anymore.

For manual process validation, refer here.

For any issue, follow here to collect the relevant logs.

Troubleshooting

```
# elfs-cli system show --id 1
# Invalid username or password.
# 401 Unauthorized

WORKAROUND:
# elfs-cli system show --id 1 --auth admin
password:
```

In case an upgrade was failed or version was not flipped,
Open a new case and upload system logs files. Contact the Elastifile TSE On-Call as well

```
# gsutil cp /elastifile/log/emanage/production.log gs://elk-cs-logs/<CUSTOMER_NAME>/upgrade/
# gsutil cp /elastifile/log/upgrade/upgrade.log gs://elk-cs-logs/<CUSTOMER_NAME>/upgrade/
# gsutil cp /elastifile/log/ecp/ecp-server.log gs://elk-cs-logs/<CUSTOMER_NAME>/upgrade/
# gsutil cp /elastifile/log/ecs/ecs.log gs://elk-cs-logs/<CUSTOMER_NAME>/upgrade/
```

Appendix A- Manual Prechecks

1. Verify SSH works to all nodes

```
[root@ndu ~(elfs_admin)]# for i in $(elfs-cli enode list | grep vHead | awk '{print $7}'); do ssh $i df -h /; done

Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1        30G  2.8G   28G  10% /
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1        30G  2.8G   28G  10% /
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1        30G  2.8G   28G  10% /
```

2. Make sure all the quotas metrics are OK
   Download the script and execute it by providing the ADMIN password

```
wget https://storage.googleapis.com/elastifile-software-repo/scripts/check_quotas.py

python check_quotas.py -p <ADMIN_PASSWORD>
Cores:  OK
Standard HDD:  OK
Persistent SSD:  OK
Instances:  OK
Internal IP Addresses:  OK
Routes:  OK
```

3. Make sure all the ELFS LB routes are exist with the right LB VIP
   ● In case of shared VPC, HOST_PROJ_ID is required

```
gcloud compute routes list [--project <HOST_PROJ_ID>] | grep $(hostname) | grep elfs

elfs-route-ndu-elfs-69478232     snir-network      10.255.255.1/32      10.164.15.237
elfs-route-ndu-elfs-784e1ba1     snir-network      10.255.255.1/32      10.164.0.21
elfs-route-ndu-elfs-e067259c     snir-network      10.255.255.1/32      10.164.0.19
```

4. Run the following commands and check the output accordingly:

● **ecs-cli system**
  ○ System state is IN_SERVICE
  ○ ecs state is ELECTED_RUNNING
  ○ ecs-upgrade-state & upgrade-state are both IDLE
  ○ ecs version and working-ver are matched

```
[root@ndu ~(elfs_admin)]# ecs-cli system
ecs-version: 3.2.1.20-63424.23f95979e015
ecs-state: internal=ECS_STATE_ELECTED_RUNNING
:
ecs-upgrade-state: IDLE
upgrade-state=IDLE
:
system-state: IN_SERVICE NORMAL CO_LOCKDOWN_NONE
working-ver: 3.2.1.20-63424.23f95979e015
```

- **ecs-cli ha-report**
  - Make sure ownership-recovery is not running
  - There are no in progress transitions

```
[root@ndu ~(elfs_admin)]# ecs-cli ha-report
ownership-recovery: NOT_IN_RECOVERY total:0 done:0 id:0
roc-transitions: total:0 done:0 inprogress:0 rebuildid:0
roc-transitions: progress:<> src:<> dst:<>
orc-transitions: total:0 done:0 inprogress:0 rebuildid:0
orc-transitions: progress:<> src:<> dst:<>
ecdb-transitions: total:0 done:0 inprogress:0 rebuildid:0
ecdb-transitions: progress:<> src:<> dst:<>
ROC map calculated with keeping Failure Domains constraints: true
```

- **ecs-cli nodes**
  - Number of nodes is correct
  - All nodes states are ENODE_RUNNING
  - There are 3 ECDB nodes

```
[root@ndu ~(elfs_admin)]# ecs-cli nodes
[1] vHead-10.154.0.X EOid[ENODE:d3e350da-99f9-4484:1] ENODE_RUNNING VALID ECDB     ||
[2] vHead-10.154.0.X EOid[ENODE:c6317a68-8dca-432d:2] ENODE_RUNNING VALID ECDB     ||
[3] vHead-10.154.0.X EOid[ENODE:852fbd49-7bf7-48ce:3] ENODE_RUNNING VALID ECDB SO ||
```

- **elfs-cli host list -t**
  - All the converged hosts are active
  - The number of converged hosts match the number of node from 'ecs-cli nodes'
  - <u>Note-</u> The None hosts are either EMS or RAs

```
[root@ndu ~(elfs_admin)]# elfs-cli host list -t
id  name           status  host_type  role       enode_setup_status  power_state
-----------------------------------------------------------------------------------
1   10.154.0.X             physical   None                           poweredOn
2   10.154.0.X             physical   converged  active              poweredOn
3   10.154.0.X             physical   converged  active              poweredOn
4   10.154.0.X             physical   converged  active              poweredOn
5   10.154.0.X             physical   None                           poweredOn
6   10.154.0.X             physical   None                           poweredOn
```

- **df -h /**
  - EMS local disk has enough space (use% < 80%)

```
[root@ndu ~(elfs_admin)]# df -h /
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1        30G  4.6G   26G  16% /
```

- **df -i /**
  - Make sure there are no Inodes leaks

```
[root@ndu ~(elfs_admin)]# df -i /
Filesystem       Inodes   IUsed    IFree IUse% Mounted on
/dev/sda1      15727104  167227 15559877    2% /
```

Appendix B- Monitoring 2nd Phase in GUI

The second phase can be monitored in the GUI,
using the very last option in the left menu called SYSTEM VIEW.

- Before the second phase is started
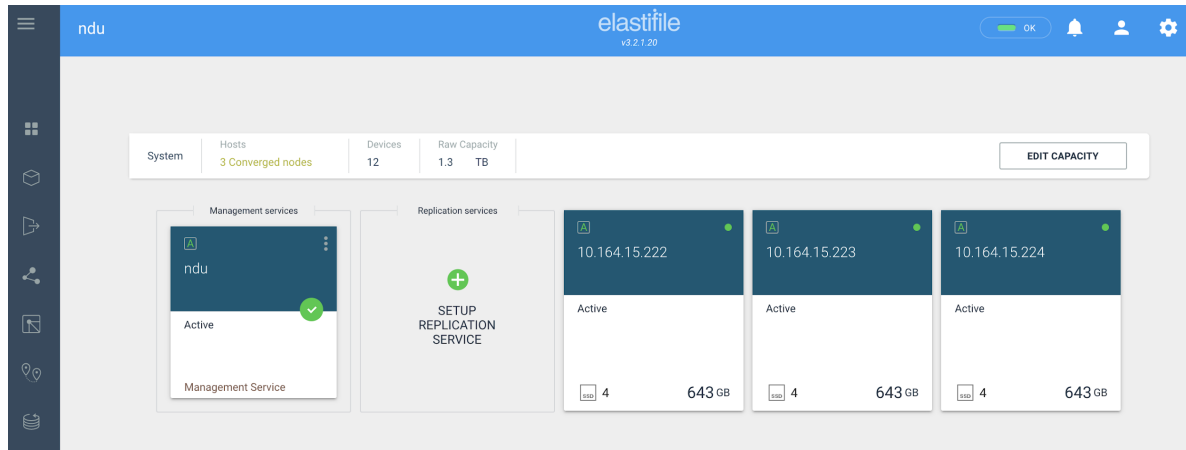
- The version in the top-center banner will be the new one.
  System status is 'Upgrading'.
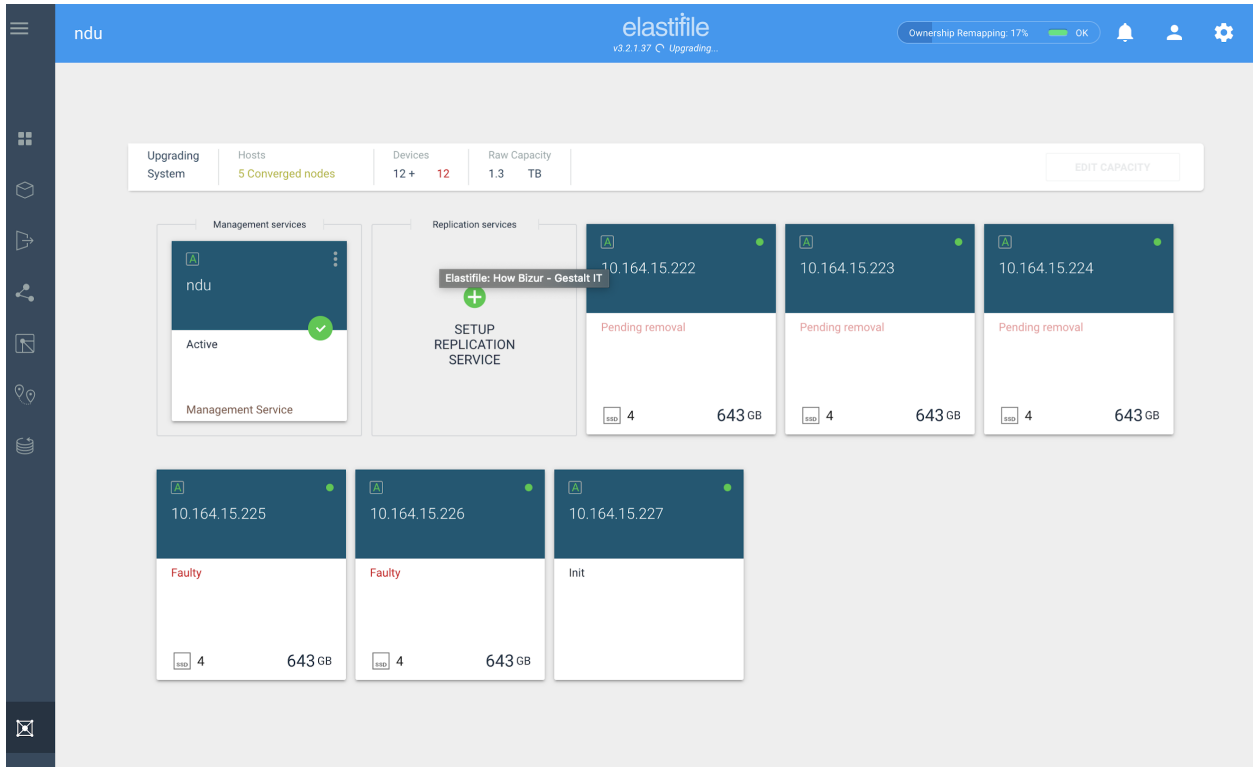 All nodes are active and 'Pending upgrade'.
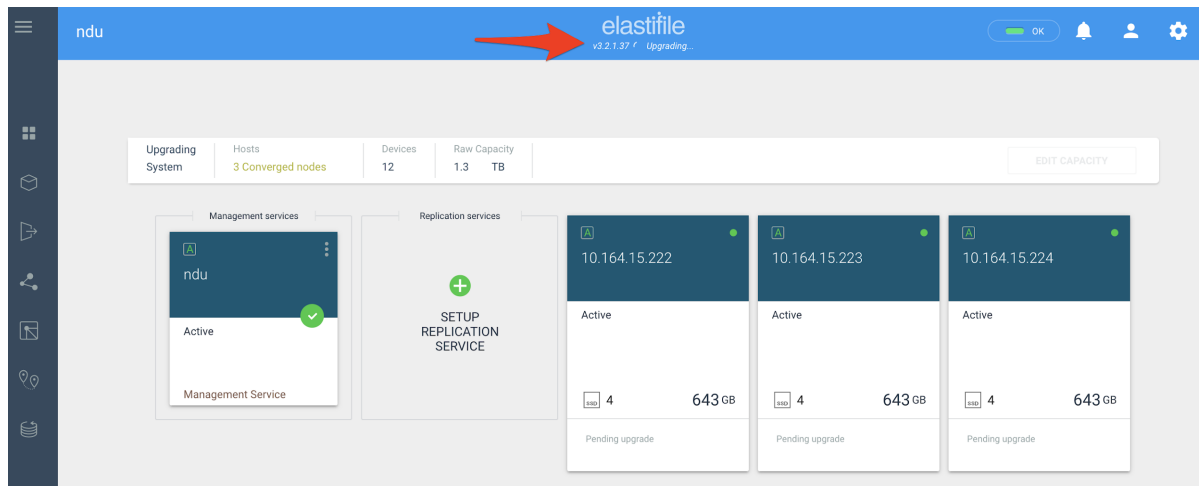
- The new nodes will be attached to the system.
  Still in 'Ready to deploy' status.

- Old nodes will change their status from 'Active' to 'Pending removal'.
  New nodes will be in 'init' state.

- It is **OK** if nodes will become **Faulty** after init state
  Ownership Remapping is running in the background

- After the init state, the new nodes will become 'Active'.
  Data Rebuild is running in the background to replicate the data from old nodes to new nodes.
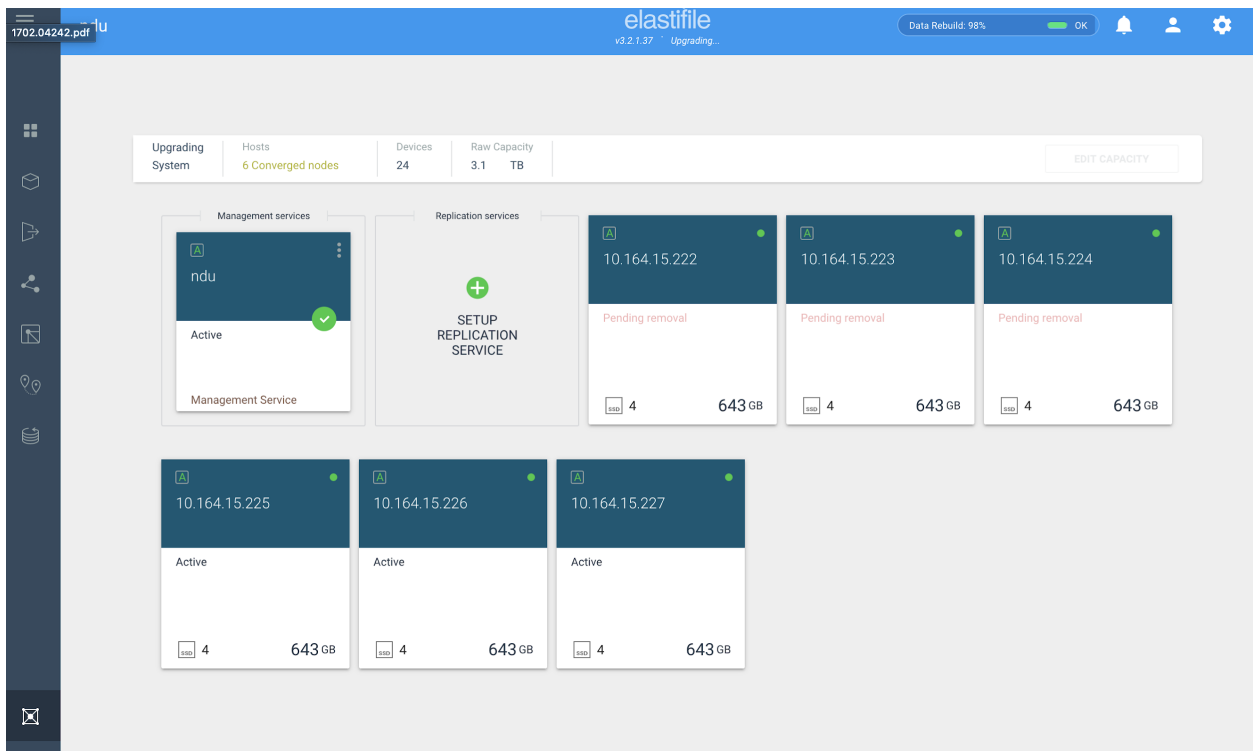
- Once data rebuild will be completed, the old nodes will be removed

(can be a long process, ~1 hour)

Appendix C- Manual Process Validation

1. Validate the control_task job has '**success**' status
   (if the status is still running, please wait)

```
elfs-cli control_task list --search type=cloud_ndu_job -t

id  uuid                                     status   name           current_step   step_progress  step_total    created_at        updated_at        ...
---------------------------------------------------------------------------------------------------------------------------------------------------------
14  866839be-5094-47c3-865c-955cbdbe097b    success  Cloud ndu job  cleanup             34             34        May 12, 08:04:38  May 12, 08:15:58  ...
```

2. Validate that ecs-version and working-ver are matched and running the target version

```
ecs-cli system | grep ver

ecs-version: 3.2.1.51-64365.b02d574e6085
working-ver: 3.2.1.51-64365.b02d574e6085
```

3. Modify and validate the ownership_policy is set to '**round_robin**'
   a. Find what is the current ownershop_policy:

   ```
   elfs-cli system show --id 1 | grep ownership_policy

   ownership_policy:              local_node
   ```

   b. Set the ownershop_policy to round_robin:

   ```
   elfs-cli system update --ownership_policy round_robin --id 1
   ```

   c. Validate the ownershop_policy is set to round_robin:

   ```
   elfs-cli system show --id 1 | grep ownership_policy

   ownership_policy:              round_robin
   ```