Linux Cluster Monitor Manual

Chenyang Li

2021 Jan. 14th

Required Docker images:

1. **neonadia/rackobserver**

   Only one per deployment. It is the webpage of racks view. Not necessary for Linux Cluster Monitor (LCM) core functionality.

2. **neonadia/udpserver**

   One per ".csv" file. Used to send/receive short messages with UDP clients. Necessary for UDP functions and benchmark functions. Not necessary for LCM core functionality.

3. **neonadia/sysmonitorfrontend**

   One per ".csv" file. It is the webpage of systems view, display sensor reading and advanced features. Necessary for LCM core functionality.

4. **neonadia/sysmonitorsensor**

   One per ".csv" file. It is the backend of systems view, responsible for periodic sensor reading. Necessary for LCM core functionality.

5. **mongo**

   a. Database for sysmonitorsensor and sysmonitorbackend, one per ".csv" file. Necessary for LCM core functionality.

   b. Database for UDP clients, responsible for distributing ports' numbers to UDP clients. The port number for this Database is 8888. Necessary for UDP functions and benchmark functions. Noe necessary for LCM core functionality.

6. **neonadia/sysmonitorbackend**

   One per ".csv" file. It is the backend of systems view, responsible for one-time hardware/firmware specs reading. Necessary for LCM core functionality. Will exit after finishing its job.

7. **neonadia/autostart**

   One per deployment. Pre-deployment set up for LCM. Necessary for LCM core functionality. Will exit after finishing its job.

Quick deployment Steps:

1. **Install Docker, docker-compose and ncat:**

   I. Docker: https://docs.docker.com/engine/install/centos/

   II. Docker-compose: https://docs.docker.com/compose/install/

   III. Ncat: $ yum install nmap-ncat.x86_64

   IV. Configure the default bridge, create “/etc/docker/daemon.json” file and restart docker:

```
 {
   "default-address-pools":
  [
    {"base":"10.10.0.0/16","size":24}
  ]
}
```

2. **Bash shell script for deployment:**

   root@10.33.10.36: docker_deploy.sh

3. **Necessary input files for deployment:**

   “cluster1.csv” is the system access input file.

   “auto.env” is the LCM configuration file.

```
[root@localhost ~]# ls INPUTFOLDER/
auto.env    cluster1.csv
[root@localhost ~]# cat auto.env
IOPATH=INPUTFOLDER   # need to be the same as input folder name
PORTNUM=10000 # note that not only 10000 will be used
```

```
PWDSERVER=10.2.1.136 # server for password query
PWDUSERNAME=rackteam # user name for password query
PWDPWD=sMcraCK@17 # password for password query
UDPMONGOPORT=8888 # port number of mongo DB provide UDP client their UDP server port
MACNAME=eth0 # mac address used to identify UDP client and find password
```

IV. Run: $ ./docker_deploy.sh FOLDERNAME

V. Stop firewall: $ systemctl stop firewalld

Deployment tips (Must read):

1. Some projects are not using unique password, you need to add a column named “PWD” inside “cluster1.csv” file to specify the password. Otherwise, LCM will look for password from the database.
2. “PORTNUM” inside “auto.env” file is the starting number of all ports’ numbers:

   Rackobserver webpage = PORTNUM

   Sensor/System mongo DB = PORTNUM + n

   UDP mongo DB = 8888

   Sysmonitorfrontent webpage = PORTNUM + 1000 + n

   UDP port = PORTNUM + 2000 + n

   Where n is the number of clusters.
3. OOB license is necessary for the basic function of LCM. DCMS license is necessary for redfish advanced features.
4. Multiple “*.csv” files are allowed, each “*.csv” will be treated as one cluster/rack, and file name will be used as cluster/rack name.
5. Configure default bridge can help avoid IP confliction.
6. LCM can be deployed before or after UDP clients starts. UDP clients will obtained its port number once the LCM UDP mongo database has been initialized.

Backup data:

1. LCM can automatically backup sensor readings and system information inside a created directory. The directory name should be looks like “cluster1_MongoDB_2021-01-04-18-05-43”.
2. After cleaning all the containers, the folder will not be removed. If you want to access the data, simply run a mongo DB mount with this folder:

```
docker run -d -p XXXX:XXXX -v ~/cluster1/cluster1_MongoDB_2021-01-04-18-05-43:/data/db --name cluster1backup mongo mongod --port XXXX
```

Redeployment steps:

Adding a new LCM without completely turning off the previous one is possible:

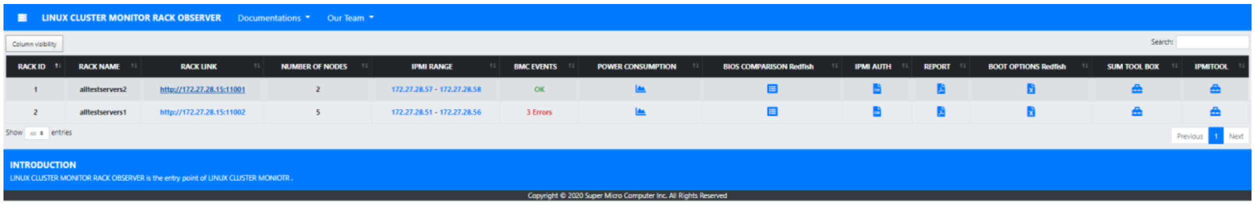
1. Stop and remove UDP config mongo (port number 8888).
2. Change PORTNUM and MACNAME.
3. Deploy a new LCM.

LCM playground:

- **172.27.28.15** LCM development server 1
- **172.27.28.17** LCM development server 2
- **172.31.32.198** LCM development server on engineering network
- **10.33.10.36** LCM server for Facebook Cluster monitoring.

LCM Key Functions:

1. Cluster view page：
   - Entry point of LCM, the port number of this page can be found in “auto.env” file.
   - Container neonadia/rackobserver is hosting this page.
   - Power Consumption buttons can be used to monitor the power usage of every server for a single cluster.



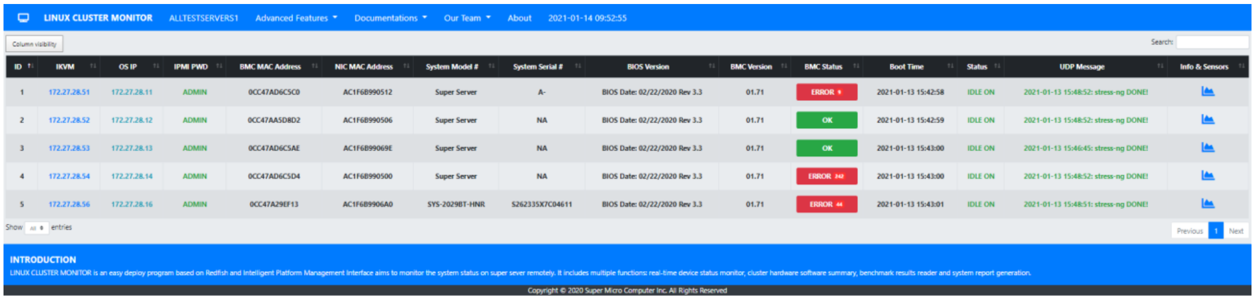
RACK ID	RACK NAME	RACK LINK	NUMBER OF NODES	IPMI RANGE	BMC EVENTS	POWER CONSUMPTION	BIOS COMPARISON Redfish	IPMI AUTH	REPORT	BOOT OPTIONS Redfish	SUM TOOL BOX	IPMITOOL
1	alltestservers2	http://172.27.28.15:11001	2	172.27.28.57 - 172.27.28.58	OK	📈	☰	📄	📄	📄	🧰	🧰
2	alltestservers1	http://172.27.28.15:11002	5	172.27.28.51 - 172.27.28.56	3 Errors	📈	☰	📄	📄	📄	🧰	🧰

INTRODUCTION

LINUX CLUSTER MONITOR RACK OBSERVER is the entry point of LINUX CLUSTER MONIOTR .

Copyright © 2020 Super Micro Computer Inc. All Rights Reserved

2. System view page
    - Entry point of systems info and key functions.
    - BMC events can be found here.
    - UDP messages are used to confirm the connections between LCM and server OS and run benchmarks.
    - Info & Sensors buttons are linked with Details page.
    - Useful functions can be found in Advanced Features, including BIOS update, Firmware Update, change BIOS settings, compare BIOS settings and so on.



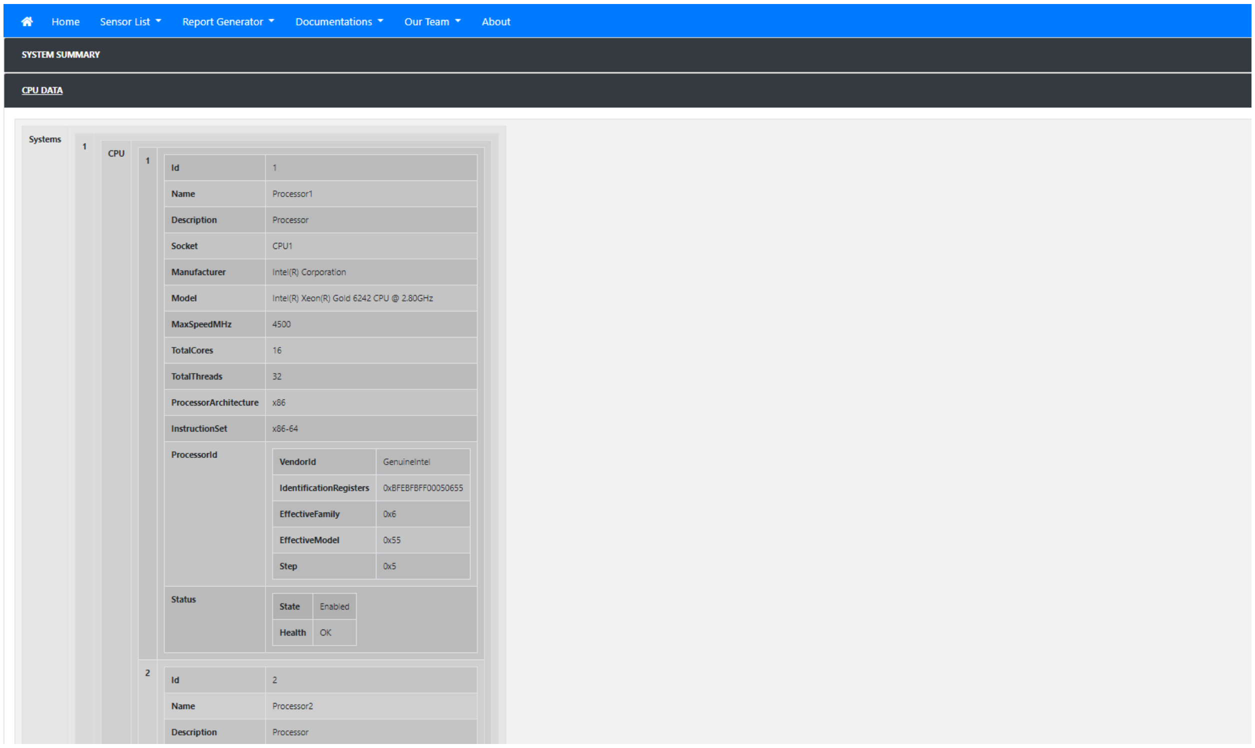
ID	IKVM	OS IP	IPMI PWD	BMC MAC Address	NIC MAC Address	System Model #	System Serial #	BIOS Version	BMC Version	BMC Status	Boot Time	Status	UDP Message	Info & Sensors
1	172.27.28.51	172.27.28.11	ADMIN	0CC47AD6C5C0	AC1F6B990512	Super Server	A-	BIOS Date: 02/22/2020 Rev 3.3	01.71	ERROR 9	2021-01-13 15:42:58	IDLE ON	2021-01-13 15:48:52: stress-ng DONE!	📈
2	172.27.28.52	172.27.28.12	ADMIN	0CC47AA5D8D2	AC1F6B990506	Super Server	NA	BIOS Date: 02/22/2020 Rev 3.3	01.71	OK	2021-01-13 15:42:59	IDLE ON	2021-01-13 15:48:52: stress-ng DONE!	📈
3	172.27.28.53	172.27.28.13	ADMIN	0CC47AD6C5AE	AC1F6B99069E	Super Server	NA	BIOS Date: 02/22/2020 Rev 3.3	01.71	OK	2021-01-13 15:43:00	IDLE ON	2021-01-13 15:46:45: stress-ng DONE!	📈
4	172.27.28.54	172.27.28.14	ADMIN	0CC47AD6C5D4	AC1F6B990500	Super Server	NA	BIOS Date: 02/22/2020 Rev 3.3	01.71	ERROR 242	2021-01-13 15:43:00	IDLE ON	2021-01-13 15:48:52: stress-ng DONE!	📈
5	172.27.28.56	172.27.28.16	ADMIN	0CC47A29EF13	AC1F6B9906A0	SYS-2029BT-HNR	S262335X7C04611	BIOS Date: 02/22/2020 Rev 3.3	01.71	ERROR 44	2021-01-13 15:43:01	IDLE ON	2021-01-13 15:48:51: stress-ng DONE!	📈

INTRODUCTION

LINUX CLUSTER MONITOR is an easy deploy program based on Redfish and Intelligent Platform Management Interface aims to monitor the system status on super sever remotely. It includes multiple functions: real-time device status monitor, cluster hardware software summary, benchmark results reader and system report generation.

Copyright © 2020 Super Micro Computer Inc. All Rights Reserved

3. Details page
    - Firmware and hardware information can be found here.
    - Sensor List is the entry point of sensor readings
    - Report Generator can be used to generate system level report.



Systems 1 CPU 1

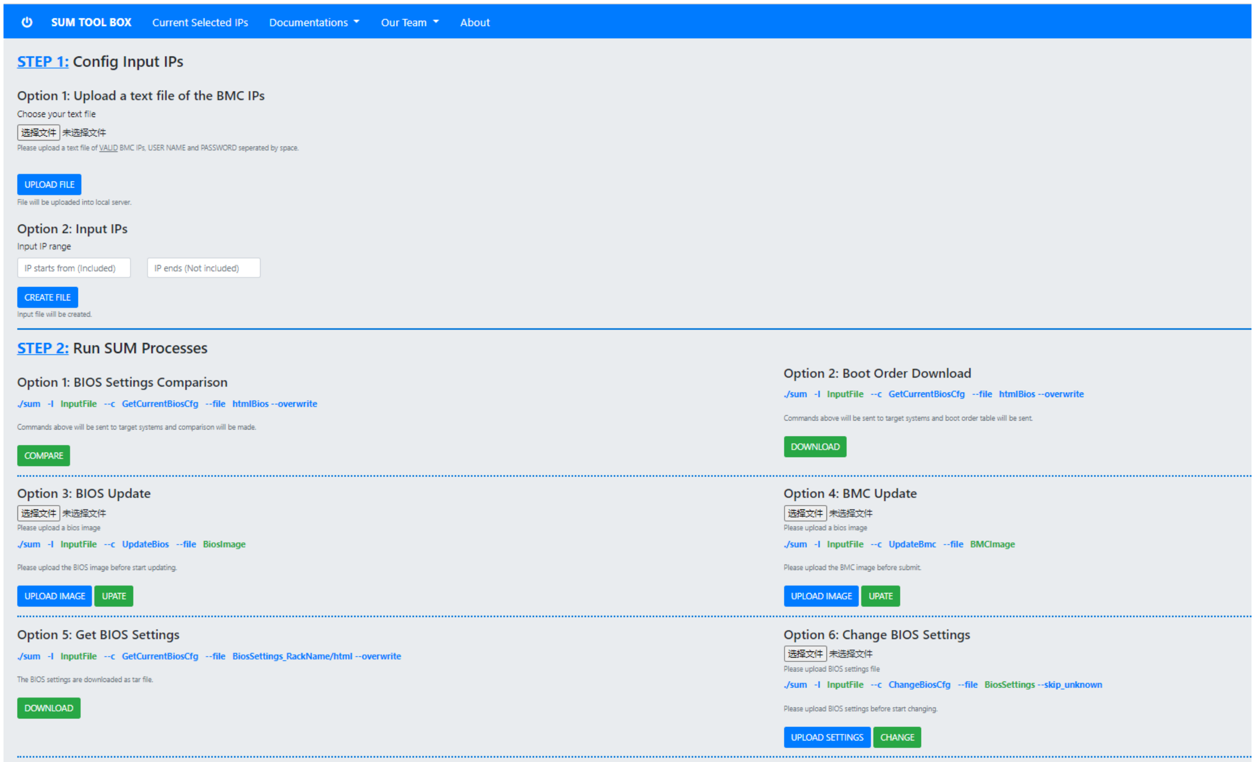
Id	1
Name	Processor1
Description	Processor
Socket	CPU1
Manufacturer	Intel(R) Corporation
Model	Intel(R) Xeon(R) Gold 6242 CPU @ 2.80GHz
MaxSpeedMHz	4500
TotalCores	16
TotalThreads	32
ProcessorArchitecture	x86
InstructionSet	x86-64
ProcessorId	VendorId: GenuineIntel IdentificationRegisters: 0xBFEBFBFF00050655 EffectiveFamily: 0x6 EffectiveModel: 0x55 Step: 0x5
Status	State: Enabled Health: OK

CPU 2

Id	2
Name	Processor2
Description	Processor

4. SUM Toolbox
    - SUM has been implemented into LCM
    - Multi-node support

- Require OOB license.
- DCMS license is not required.



STEP 1: Config Input IPs

Option 1: Upload a text file of the BMC IPs

Choose your text file

未选择文件

Please upload a text file of VALID BMC IPs, USER NAME and PASSWORD seperated by space.

File will be uploaded into local server.

Option 2: Input IPs

Input IP range

Input file will be created.

STEP 2: Run SUM Processes

Option 1: BIOS Settings Comparison

```
./sum  -I  InputFile  --c  GetCurrentBiosCfg  --file  htmlBios --overwrite
```

Commands above will be sent to target systems and comparison will be made.

Option 2: Boot Order Download

```
./sum  -I  InputFile  --c  GetCurrentBiosCfg  --file  htmlBios --overwrite
```

Commands above will be sent to target systems and boot order table will be sent.

Option 3: BIOS Update

未选择文件

Please upload a bios image

```
./sum  -I  InputFile  --c  UpdateBios  --file  BiosImage
```

Please upload the BIOS image before start updating.

Option 4: BMC Update

未选择文件

Please upload a bios image

```
./sum  -I  InputFile  --c  UpdateBmc  --file  BMCImage
```

Please upload the BMC image before submit.

Option 5: Get BIOS Settings

```
./sum  -I  InputFile  --c  GetCurrentBiosCfg  --file  BiosSettings_RackName/html --overwrite
```

The BIOS settings are downloaded as tar file.

Option 6: Change BIOS Settings

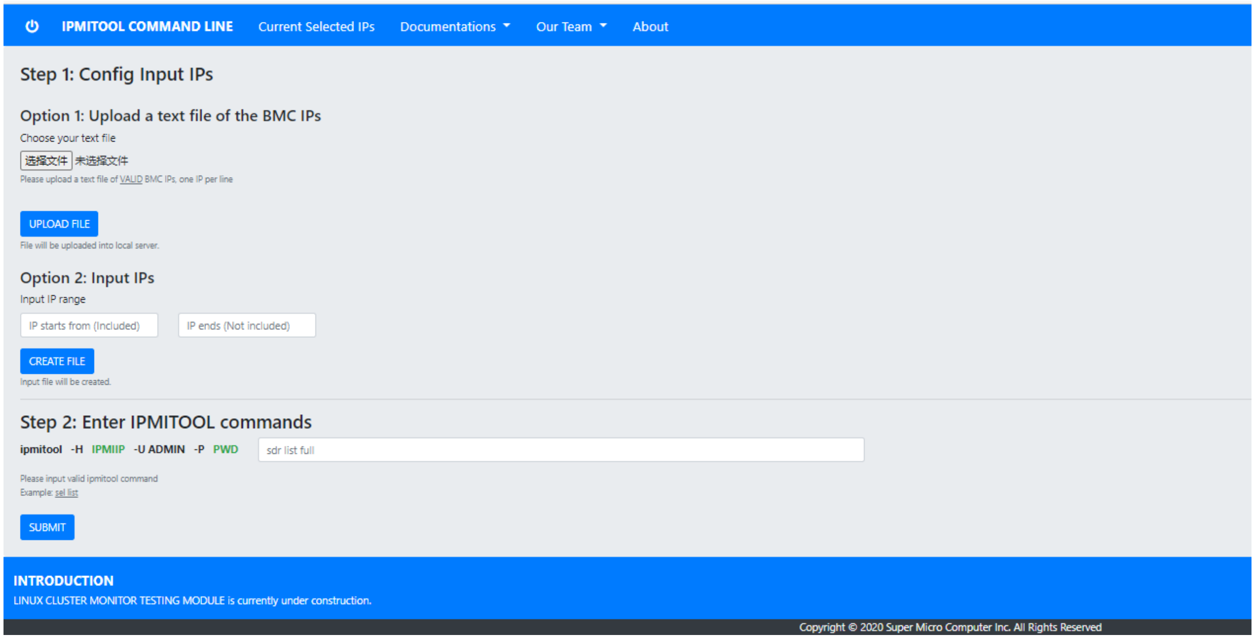
未选择文件

Please upload BIOS settings file

```
./sum  -I  InputFile  --c  ChangeBiosCfg  --file  BiosSettings --skip_unknown
```

Please upload BIOS settings before start changing.

5. IPMI Tool
   - Customized IPMI command can be ran here.
   - Multi-node support.



Step 1: Config Input IPs

Option 1: Upload a text file of the BMC IPs

Choose your text file

未选择文件

Please upload a text file of VALID BMC IPs, one IP per line

File will be uploaded into local server.

Option 2: Input IPs

Input IP range

Input file will be created.

Step 2: Enter IPMITOOL commands

```
ipmitool  -H  IPMIIP  -U ADMIN  -P  PWD   sdr list full
```

Please input valid ipmitool command

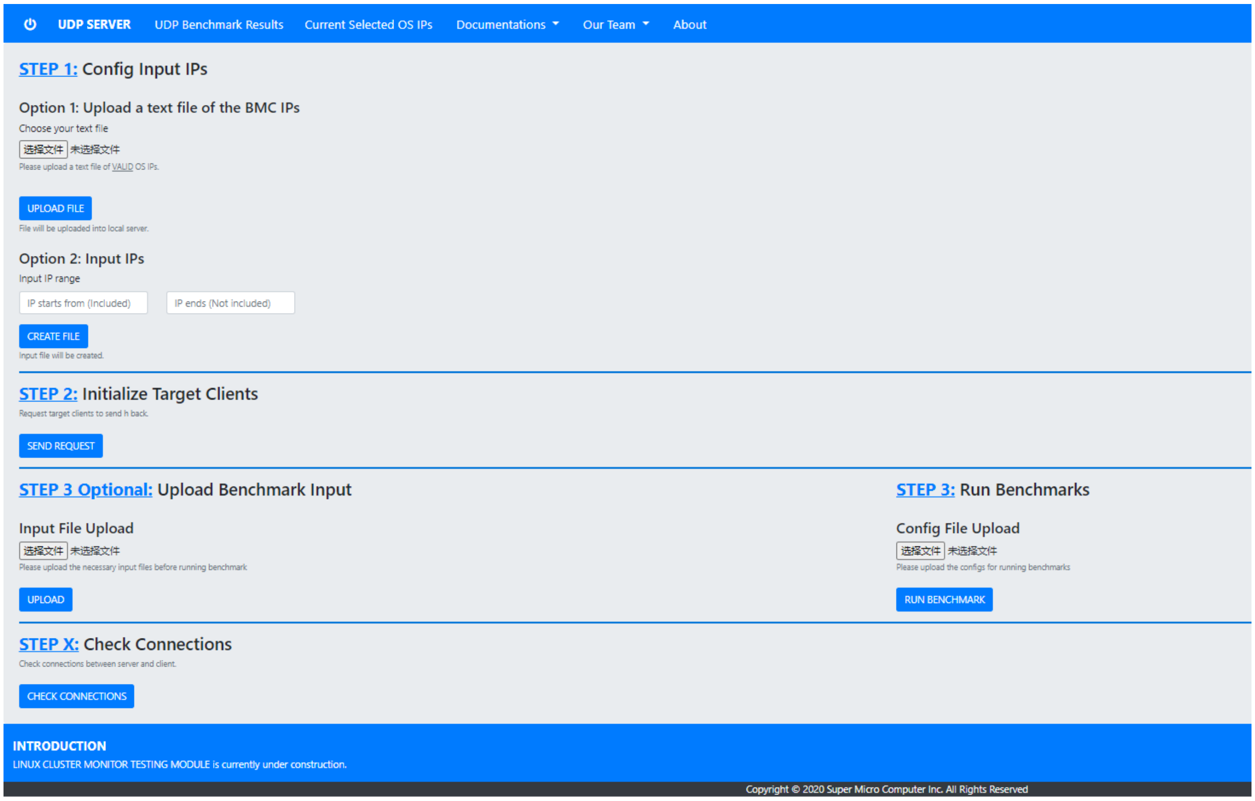
Example: sel list

INTRODUCTION

LINUX CLUSTER MONITOR TESTING MODULE is currently under construction.

6. UDP Server Controller
   - Used to control the systems' OS level.

- Used to run benchmarks on systems.
- Multi-node support.
- Results will be saved into database and LCM can display the historical results.



STEP 1: Config Input IPs

Option 1: Upload a text file of the BMC IPs

Choose your text file

未选择文件

Please upload a text file of VALID OS IPs.

File will be uploaded into local server.

Option 2: Input IPs

Input IP range

Input file will be created.

STEP 2: Initialize Target Clients

Request target clients to send h back.

STEP 3 Optional: Upload Benchmark Input

Input File Upload

未选择文件

Please upload the necessary input files before running benchmark

STEP 3: Run Benchmarks

Config File Upload

未选择文件

Please upload the configs for running benchmarks

STEP X: Check Connections

Check connections between server and client.

INTRODUCTION

LINUX CLUSTER MONITOR TESTING MODULE is currently under construction.

Copyright © 2020 Super Micro Computer Inc. All Rights Reserved

Possible errors and debug:

```
Error: UDP server problem

Created new file: /app/RACK/alltestservers1-host.json

Exception in thread Thread-1:

Traceback (most recent call last):

  File "/usr/local/lib/python3.7/threading.py", line 926, in _bootstrap_inner

    self.run()

  File "udpserver-1.5.py", line 137, in run

    self.write_log('msg', data[1])

  File "udpserver-1.5.py", line 42, in write_log

    if load_json[self.mac]['ip'] == self.ip:

TypeError: 'NoneType' object is not subscriptable
```

Restart udpserver container will resolve this issue.

Error: Failed to Setup IP tables: Unable to enable SKIP DNAT rule: (iptables failed: iptables --wait -t nat -I DOCKER -i br-79c12f098a37 -j RETURN: iptables: No chain/target/match by that name. (exit status 1))

Start the firewall will resolve the issue, after successfully deployed you can stop the firewall again.

Error: no response from RAKP 1 message

It happens sometime when redfish cannot get the IKVM address. This problem can be automatically resolved when LCM running the next round data query. So, it is not critical.

Seems Boot Up Not Completed Yet (Error 500)

This error could be due to one of the following reasons:

1. LCM does not boot up completely.
2. Wrong Redfish API.

To resolve this error, you need to check the log information of the container.

Redfish login failed

This error could be due to one of the following reasons:

1. IPMI did not activated.
2. Request confliction (Not critical).

To activate IPMI, one can use sum to activate it, check “DCMS Single Key Activate Guide” for details.

For example:

```
$ ./sum -i 172.27.28.51 -u ADMIN -p ADMIN -c ActivateProductKey –key xxxxxxxxxxxxxxxxxxxxx
```

Firmware updating cannot started, error code: 503

This error is due to something block the firmware update, please make sure the following pages have been closed:

1. Correspond IPMI page.
2. Correspond IKVM page.
3. Correspond remote console.

To resolve this issue, simply cancel the update will work.