



Health Check Guide

Technical Note

Table of Contents

1	Introduction	1
2	Health Checks	1
2.1	Global Status	1
2.2	SRE Process	4
2.3	Database Process	5
2.4	Call Admission Control Database Process	6
2.5	Statistics Database Process	6
2.6	Main Database Replication	7
2.7	Call Admission Control Database Replication	7
2.8	Statistics Production	10
2.9	SIP Stack	11
2.10	SIP Agents Monitoring	13
2.11	Accounting Generation	13
2.12	Accounting Synchronization	14

1 Introduction

This guide provides a comprehensive overview of various health checks that can be performed to assess the functionality and stability of your platform. These checks should be performed after a platform upgrade to ensure that everything works correctly and that no new issues have been introduced.

2 Health Checks

Some of these health checks rely on the use of the *sre-admin* tool, while others depend on OS commands or subsystem monitoring commands.

On some systems, the *sre-admin* tool may complain about the locales not being correctly set at the OS level. If this occurs, you can set it to a default locale with the following command: `export LC_ALL=en_US.utf8.`

2.1 Global Status

Target servers: master EM

The sre-admin tool can be used to check the global status of the system with `/opt/sre/bin/sre-admin status`. This command provides status about various information collected from all the EMs and CPs, including: - DB status - DB replication status - active connections on local DB - SRE processes status - CPU and RAM usage

Example:

```

1 [root@SRE-33-EM1 ~]# /opt/sre/bin/sre-admin status
2 Server DB status
3 Hostname      Status
4 -----
5 SRE-33-EM1   master
6
7 Server DB replication status
8 Client      Address      State      Started      WAL sent      WAL written      WAL
9 -----
10 cp2        10.0.161.183 streaming  10:20:38     F/605F7F68   F/605F7F68     F/605
11 em2        10.0.161.181 streaming  09:09:22     F/605F7F68   F/605F7F68     F/605
12 cp1        10.0.161.182 streaming  09:30:21     F/605F7F68   F/605F7F68     F/605
13
14 Server DB activity
15 Client      Address      Username    PID  DB      Last
16 -----
17
18
19      127.0.0.1   sre         5241
20      127.0.0.1   sre         5243
21      127.0.0.1   sre         18502  sre
22      127.0.0.1   sre         15258  sre
23      127.0.0.1   sre         31533  sre
24      127.0.0.1   sre         31211  sre
25      127.0.0.1   sre         18510  sre
26      127.0.0.1   sre         31431  sre
27
28
29      12:01:45 (1292118.09) idle      ROLLBACK
30      12:04:52 (82330.90)  idle      ROLLBACK
31      10:57:02 (0.31)      idle      ROLLBACK
32      127.0.0.1   sre         31211  sre
33      10:57:02 (0.56)      idle      ROLLBACK
34      127.0.0.1   sre         18510  sre
35      12:01:50 (1292112.35) idle      ROLLBACK
36      127.0.0.1   sre         31431  sre
37      10:56:06 (56.69)      idle      ROLLBACK

```

```

25      127.0.0.1      postgres      18522 postgres
      ↪ 10:57:00 (2.95)      idle      SELECT * FROM pg_stat_bgwriter
26 ...
27      127.0.0.1      sre          31216  sre
      ↪ 10:57:02 (0.56)      idle      ROLLBACK
28      127.0.0.1      sre          2322  sre
      ↪ 10:57:03 (0.01)      active   SELECT datname,pid,username,
      ↪ application_name,client_addr,backend_start,xact_start,query_start,
      ↪ state_change,state,query FROM pg_stat_activity
29
30      5239
31      5238
32      5240
33 Platform process status
34 Process/Server      SRE-33-CP1      SRE-33-EM1      SRE-33-EM2
      ↪ CP2
      ↪
      ↪ SRE-33-EM2
35 -----
      ↪ -----
      ↪ -----
      ↪ -----
36 sre-REST      STOPPED      STOPPED
      ↪
      ↪ RUNNING (pid 18463, uptime 14 days,
      ↪ 22:55:35) RUNNING (pid 28206, uptime 14 days, 22:55:05)
37 sre-agents-monitor      RUNNING (pid 17243, uptime 8 days, 1:09:08)      RUNNING (
      ↪ pid 3113, uptime 8 days, 1:04:36)      STOPPED
      ↪ STOPPED
38 sre-broker      RUNNING (pid 16207, uptime 14 days, 22:55:48)      RUNNING (
      ↪ pid 7436, uptime 14 days, 22:55:49)      STOPPED
      ↪ STOPPED
39 sre-call-processor:0      RUNNING (pid 16208, uptime 14 days, 22:55:48)      RUNNING (
      ↪ pid 7437, uptime 14 days, 22:55:49)      STOPPED
      ↪ STOPPED
40 sre-cdr-collector      STOPPED      STOPPED
      ↪
      ↪ RUNNING (pid 18464, uptime 14 days,
      ↪ 22:55:35) RUNNING (pid 28207, uptime 14 days, 22:55:05)
41 sre-cdr-postprocessor      STOPPED      STOPPED
      ↪
      ↪ STOPPED
      ↪ STOPPED
42 sre-cdr-sender      RUNNING (pid 16210, uptime 14 days, 22:55:48)      RUNNING (
      ↪ pid 7438, uptime 14 days, 22:55:49)      STOPPED
      ↪ STOPPED
43 sre-dns-updater      STOPPED      STOPPED
      ↪
      ↪ STOPPED
      ↪ STOPPED
44 sre-enum-processor      RUNNING (pid 16211, uptime 14 days, 22:55:48)      RUNNING (
    
```

```

↪ pid 7439, uptime 14 days, 22:55:49) STOPPED
↪ STOPPED
45 sre-gui STOPPED STOPPED
↪ RUNNING (pid 18465, uptime 14 days,
↪ 22:55:35) RUNNING (pid 28208, uptime 14 days, 22:55:05)
46 sre-health-monitor RUNNING (pid 16212, uptime 14 days, 22:55:48) RUNNING (
↪ pid 7440, uptime 14 days, 22:55:49) RUNNING (pid 18466, uptime 14 days,
↪ 22:55:35) RUNNING (pid 28209, uptime 14 days, 22:55:05)
47 sre-http-processor RUNNING (pid 16213, uptime 14 days, 22:55:48) RUNNING (
↪ pid 7441, uptime 14 days, 22:55:49) RUNNING (pid 18467, uptime 14 days,
↪ 22:55:35) RUNNING (pid 28210, uptime 14 days, 22:55:05)
48 sre-manager STOPPED STOPPED
↪ RUNNING (pid 26712, uptime 10:57:02)
↪ RUNNING (pid 13034, uptime 12 days, 10:56:57)
49
50 Platform system status
51 Metric/Server SRE-33-CP1 SRE-33-CP2 SRE-33-EM1 SRE-33-EM2
52 -----
53 CPU 8.8 % 16.2 % 44.7 % 13.2 %
54 Memory 43.5 % 49.3 % 48.3 % 64.4 %
55 Swap 4.0 % 8.0 % 4.9 % 4.5 %
    
```

2.2 SRE Process

Target servers: EMs and CPs

You may check that the SRE processes are running by executing the `systemctl status sre` command.

Example:

```

1 [root@sre-40-em1 ~]# systemctl status sre
2 * sre.service - SRE is a centralized, multi interface Session Routing Engine.
3   Loaded: loaded (/usr/lib/systemd/system/sre.service; enabled; vendor preset:
4     ↪ disabled)
5   Active: active (running) since Wed 2024-06-05 12:01:26 CEST; 2 weeks 1 days
6     ↪ ago
7     Docs: https://www.netaxis.be/products/session-routing-engine/
8   Main PID: 18447 (supervisord)
9   CGroup: /system.slice/sre.service
10          |-18447 /opt/sre/bin/python /opt/sre/bin/supervisord -n
11          |-18463 /opt/sre/bin/python /opt/sre/bin/sre-REST
12          |-18464 /opt/sre/bin/python /opt/sre/bin/sre-cdr-collector
13          |-18465 /opt/sre/bin/python /opt/sre/bin/sre-gui
14          |-18466 /opt/sre/bin/python /opt/sre/bin/sre-health-monitor
    
```

```
13 | -18467 /opt/sre/bin/python /opt/sre/bin/sre-http-processor
```

2.3 Database Process

Target servers: EMs and CPs

You may check that the Database processes are running by executing the `systemctl status postgresql-14` command.

```
1 [root@sre-40-em1 ~]# systemctl status postgresql-14
2 * postgresql-14.service - PostgreSQL 14 database server
3   Loaded: loaded (/usr/lib/systemd/system/postgresql-14.service; enabled;
4           ↪ vendor preset: disabled)
5   Active: active (running) since Wed 2024-06-05 08:47:27 CEST; 2 weeks 1 days
6           ↪ ago
7     Docs: https://www.postgresql.org/docs/14/static/
8   Main PID: 5234 (postmaster)
9   CGroup: /system.slice/postgresql-14.service
10          |- 5234 /usr/pgsql-14/bin/postmaster -D /var/lib/pgsql/14/data/
11          |- 5236 postgres: logger
12          |- 5238 postgres: checkpointer
13          |- 5239 postgres: background writer
14          |- 5240 postgres: walwriter
15          |- 5241 postgres: autovacuum launcher
16          |- 5242 postgres: stats collector
17          |- 5243 postgres: logical replication launcher
18          |- 7353 postgres: walsender repmgr 10.0.161.181(40814) streaming F
19           ↪ /60DFDC00
20          |- 8854 postgres: walsender repmgr 10.0.161.182(47168) streaming F
21           ↪ /60DFDC00
22          |-11921 postgres: walsender repmgr 10.0.161.183(44226) streaming F
23           ↪ /60DFDC00
24          |-17337 postgres: sre sre 127.0.0.1(43842) idle
25          |-17339 postgres: sre sre 127.0.0.1(43884) idle
26          |-17606 postgres: sre sre 127.0.0.1(49254) idle
27          |-17687 postgres: sre sre 127.0.0.1(50974) idle
28          ...
29          |-19049 postgres: sre datamodel_a 127.0.0.1(46566) idle
30          |-19050 postgres: sre datamodel_b 127.0.0.1(46568) idle
```

Note

The number of running processes for the database can fluctuate. For instance, processes managing connections to specific databases may only appear after a call processor has received some

requests.

The presence of the `walsender` process indicates that this instance is sending replication information to a standby DB, along with its IP address. On a standby DB instance, the presence of the process `walreceiver` indicates the receiving process side.

2.4 Call Admission Control Database Process

Target servers: EMs (if activated) and CPs

This check is relevant only if the CAC database (MongoDB-based) was installed on the deployment.

You may check that the CAC DB is running by executing the `systemctl status mongod` command:

```

1 [root@sre-40-em1 ~]# systemctl status mongod
2 * mongod.service - MongoDB Database Server
3   Loaded: loaded (/usr/lib/systemd/system/mongod.service; enabled; vendor
4   ↪ preset: disabled)
5   Active: active (running) since Tue 2024-06-18 09:41:58 UTC; 2 days ago
6     Docs: https://docs.mongodb.org/manual
7   Main PID: 23435 (mongod)
8   CGroup: /system.slice/mongod.service
           └─23435 /usr/bin/mongod -f /etc/mongod.conf
  
```

2.5 Statistics Database Process

Target servers: EMs

You may check that the statistics DB is running by executing the `systemctl status influxdb` command.

```

1 [root@sre-40-em1 ~]# systemctl status influxdb
2 * influxdb.service - InfluxDB is an open-source, distributed, time series
3   ↪ database
4   Loaded: loaded (/usr/lib/systemd/system/influxdb.service; enabled; vendor
5   ↪ preset: disabled)
6   Active: active (running) since Mon 2024-05-27 14:14:48 CEST; 3 weeks 3 days
7   ↪ ago
8     Docs: https://docs.influxdata.com/influxdb/
9   Main PID: 943 (influxd)
10  CGroup: /system.slice/influxdb.service
        └─943 /usr/bin/influxd
  
```


the list of different members belonging to the *replica set* as well as their current state. There should be only one PRIMARY node and several SECONDARY nodes. Optionally, there may be one or more ARBITER nodes. No nodes should be in other states, such as RECOVERING or others.

The fields `lastAppliedWallTime` and `lastHeartbeat` should be recent for all members.

```
1 [root@sre-cp1 ~]# mongo --eval 'rs.status()'
2 MongoDB shell version v5.0.26
3 connecting to: mongod://127.0.0.1:27017/?compressors=disabled&
   ↪ gssapiServiceName=mongod
4 Implicit session: session { "id" : UUID("59806e01-1811-4d4d-988e-3b696e683316")
   ↪ }
5 MongoDB server version: 5.0.26
6 {
7   "set" : "sre_location",
8   "date" : ISODate("2024-06-20T10:20:42.319Z"),
9   "myState" : 2,
10  "term" : NumberLong(1),
11  "syncSourceHost" : "10.0.161.64:27017",
12  "syncSourceId" : 3,
13  "heartbeatIntervalMillis" : NumberLong(2000),
14  "majorityVoteCount" : 3,
15  "writeMajorityCount" : 3,
16  "votingMembersCount" : 4,
17  "writableVotingMembersCount" : 4,
18  "optimes" : {
19    "lastCommittedOpTime" : {
20      "ts" : Timestamp(1718878840, 1),
21      "t" : NumberLong(1)
22    },
23    "lastCommittedWallTime" : ISODate("2024-06-20T10:20:40.383Z"),
24    "readConcernMajorityOpTime" : {
25      "ts" : Timestamp(1718878840, 1),
26      "t" : NumberLong(1)
27    },
28    "appliedOpTime" : {
29      "ts" : Timestamp(1718878840, 1),
30      "t" : NumberLong(1)
31    },
32    "durableOpTime" : {
33      "ts" : Timestamp(1718878840, 1),
34      "t" : NumberLong(1)
35    },
36    "lastAppliedWallTime" : ISODate("2024-06-20T10:20:40.383Z"),
37    "lastDurableWallTime" : ISODate("2024-06-20T10:20:40.383Z")
38  },
39  "lastStableRecoveryTimestamp" : Timestamp(1718878830, 1),
```

```
40   "members" : [  
41     {  
42       "_id" : 0,  
43       "name" : "10.0.161.60:27017",  
44       "health" : 1,  
45       "state" : 1,  
46       "stateStr" : "PRIMARY",  
47       "uptime" : 175121,  
48       "optime" : {  
49         "ts" : Timestamp(1718878840, 1),  
50         "t" : NumberLong(1)  
51       },  
52       "optimeDurable" : {  
53         "ts" : Timestamp(1718878840, 1),  
54         "t" : NumberLong(1)  
55       },  
56       "optimeDate" : ISODate("2024-06-20T10:20:40Z"),  
57       "optimeDurableDate" : ISODate("2024-06-20T10:20:40Z"),  
58       "lastAppliedWallTime" : ISODate("2024-06-20T10:20:40.383Z"),  
59       "lastDurableWallTime" : ISODate("2024-06-20T10:20:40.383Z"),  
60       "lastHeartbeat" : ISODate("2024-06-20T10:20:41.142Z"),  
61       "lastHeartbeatRecv" : ISODate("2024-06-20T10:20:40.352Z"),  
62       "pingMs" : NumberLong(0),  
63       "lastHeartbeatMessage" : "",  
64       "syncSourceHost" : "",  
65       "syncSourceId" : -1,  
66       "infoMessage" : "",  
67       "electionTime" : Timestamp(1712329875, 1),  
68       "electionDate" : ISODate("2024-04-05T15:11:15Z"),  
69       "configVersion" : 1,  
70       "configTerm" : 1  
71     },  
72     {  
73       "_id" : 1,  
74       "name" : "10.0.161.62:27017",  
75       "health" : 1,  
76       "state" : 2,  
77       "stateStr" : "SECONDARY",  
78       "uptime" : 175124,  
79       "optime" : {  
80         "ts" : Timestamp(1718878840, 1),  
81         "t" : NumberLong(1)  
82       },  
83       "optimeDate" : ISODate("2024-06-20T10:20:40Z"),  
84       "lastAppliedWallTime" : ISODate("2024-06-20T10:20:40.383Z"),  
85       "lastDurableWallTime" : ISODate("2024-06-20T10:20:40.383Z"),
```

```

86         "syncSourceHost" : "10.0.161.64:27017",
87         "syncSourceId" : 3,
88         "infoMessage" : "",
89         "configVersion" : 1,
90         "configTerm" : 1,
91         "self" : true,
92         "lastHeartbeatMessage" : ""
93     },
94     ...
95 ],
96 "ok" : 1,
97 "$clusterTime" : {
98     "clusterTime" : Timestamp(1718878840, 1),
99     "signature" : {
100         "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
101         "keyId" : NumberLong(0)
102     }
103 },
104 "operationTime" : Timestamp(1718878840, 1)
105 }

```

2.8 Statistics Production

Target servers: master EM

The *sre-manager* process writes the statistics collected from different nodes to CSV files every minute. Check the content of the CSV file `/var/log/sre/counters.csv` to ensure that calls are being processed and statistics are being generated. The format of this file is:

- hostname
- metric name
- timestamp (human-readable format)
- timestamp (EPOCH timestamp)
- 15 fields with the metric value, one per minute, from the most recent to the oldest

For an SRE with active traffic, the metric `request.INVITE` should indicate the presence of calls being processed. The presence of `response.relay` metrics indicate that calls have been relayed by the Service Logic.

Example:

```

1 [root@sre-em1-ams-dune ~]# cat /var/log/sre/counters.csv
2 sre-cp1-ams-dune,request.INVITE,2024-06-20T10
   ↪ :39:00,1718879940,16,20,23,19,18,20,24,23,21,21,26,24,21,21,19

```

```

3 sre-cp2-ams-dune,request.INVITE,2024-06-20T10
  ↳ :39:00,1718879940,0,0,1,0,0,0,0,0,0,0,0,1,0,0,0,0
4 sre-cp1-ams-dune,request.OPTIONS,2024-06-20T10
  ↳ :39:00,1718879940,7,6,6,6,6,6,7,7,7,5,6,7,6,7,7
5 sre-cp1-ams-dune,request.REGISTER,2024-06-20T10
  ↳ :39:00,1718879940,2,1,1,3,0,5,0,1,0,3,2,0,1,3,2
6 sre-cp1-ams-dune,response.200 (OPTIONS),2024-06-20T10
  ↳ :39:00,1718879940,7,6,6,6,6,6,7,7,7,5,6,7,6,7,7
7 sre-cp1-ams-dune,response.authenticate,2024-06-20T10
  ↳ :39:00,1718879940,5,2,1,2,1,5,2,1,1,2,2,0,2,2,1
8 sre-cp1-ams-dune,response.locationServiceRelay,2024-06-20T10
  ↳ :39:00,1718879940,1,2,0,0,1,0,2,1,0,0,0,1,0,2,2
9 sre-cp1-ams-dune,response.locationServiceSave,2024-06-20T10
  ↳ :39:00,1718879940,2,1,1,1,0,3,0,1,0,2,1,0,1,1,1
10 sre-cp1-ams-dune,response.proxySIPError,2024-06-20T10
  ↳ :39:00,1718879940,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0
11 sre-cp1-ams-dune,response.relay,2024-06-20T10
  ↳ :39:00,1718879940,15,19,23,19,17,20,22,23,21,21,26,22,20,19,16
12 sre-cp2-ams-dune,response.relay,2024-06-20T10
  ↳ :39:00,1718879940,0,0,1,0,0,0,0,0,0,0,0,0,1,0,0,0
    
```

2.9 SIP Stack

Target servers: CPs

The command `kamctl stats` provide statistics about the SIP stack processing. This command should be run twice to verify the increase of the counters. Among these, `core:rcv_requests`, `core:rcv_requests_invite`, `core:rcv_replies` and `core: fwd_requests` should increase under traffic conditions.

Example:

```

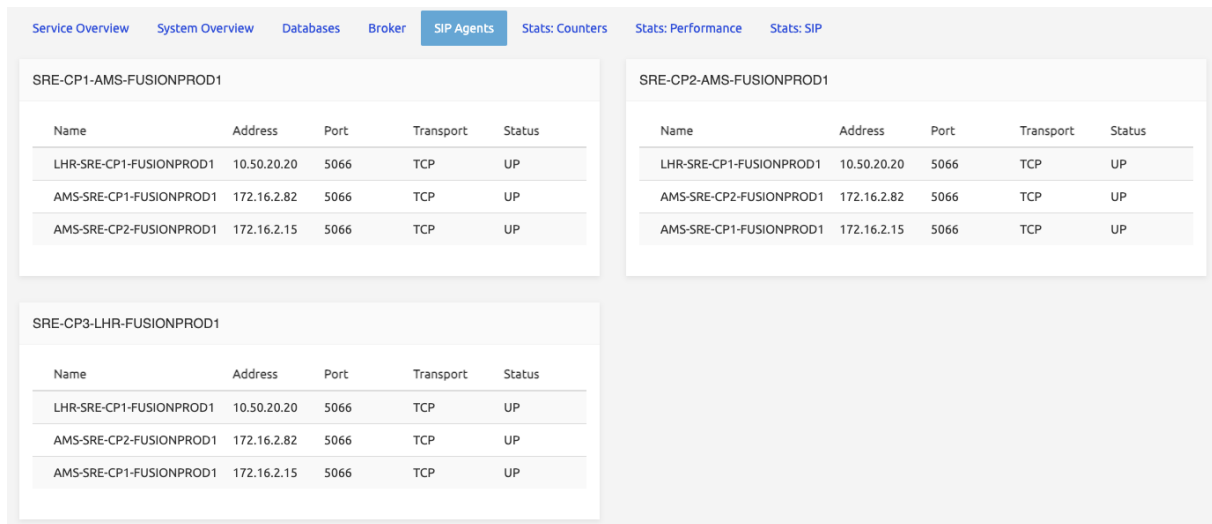
1 [root@sre-cp1-ams-dune /]# kamctl stats
2 {
3   "jsonrpc": "2.0",
4   "result": [
5     "app_python3:active_dialogs = 167",
6     "app_python3:early_dialogs = 0",
7     "app_python3:expired_dialogs = 0",
8     "app_python3:failed_dialogs = 709",
9     "app_python3:processed_dialogs = 21047",
10    "core:bad_URIs_rcvd = 0",
11    "core:bad_msg_hdr = 0",
12    "core:drop_replies = 0",
13    "core:drop_requests = 465",
    
```

```
14     "core:err_replies = 0",
15     "core:err_requests = 0",
16     "core:fwd_replies = 2",
17     "core:fwd_requests = 24173",
18     "core:rcv_replies = 98268",
19     "core:rcv_replies_18x = 18432",
20     ...
21     "core:rcv_replies_6xx_reg = 0",
22     "core:rcv_replies_6xx_update = 0",
23     "core:rcv_requests = 109766",
24     "core:rcv_requests_ack = 25120",
25     "core:rcv_requests_bye = 20026",
26     "core:rcv_requests_cancel = 382",
27     "core:rcv_requests_info = 0",
28     "core:rcv_requests_invite = 25584",
29     "core:rcv_requests_message = 0",
30     "core:rcv_requests_notify = 0",
31     "core:rcv_requests_options = 21065",
32     "core:rcv_requests_prack = 13620",
33     "core:rcv_requests_publish = 0",
34     "core:rcv_requests_refer = 0",
35     "core:rcv_requests_register = 3911",
36     "core:rcv_requests_subscribe = 0",
37     "core:rcv_requests_update = 58",
38     "core:unsupported_methods = 0",
39     "dns:failed_dns_request = 0",
40     "dns:slow_dns_request = 0",
41     "registrar:accepted_regs = 2088",
42     "registrar:default_expire = 3600",
43     "registrar:default_expires_range = 0",
44     "registrar:expires_range = 0",
45     "registrar:max_contacts = 1",
46     "registrar:max_expires = 3600",
47     "registrar:rejected_regs = 0",
48     "shmem:fragments = 101",
49     "shmem:free_size = 63602864",
50     "shmem:max_used_size = 4053816",
51     "shmem:real_used_size = 3506000",
52     "shmem:total_size = 67108864",
53     "shmem:used_size = 3077096",
54     ...
55   ],
56   "id": 22673
57 }
```

2.10 SIP Agents Monitoring

Target servers: master EM

The status of the SIP agents monitored by SRE can be viewed from the GUI dashboard under the SRE tab. Unless there are known issues with the probed agents, the status of all agents should be *UP*.



The screenshot shows a web dashboard with a navigation bar at the top containing: Service Overview, System Overview, Databases, Broker, SIP Agents (selected), Stats: Counters, Stats: Performance, and Stats: SIP. Below the navigation bar, there are three panels, each containing a table of SIP agent status.

SRE-CP1-AMS-FUSIONPROD1				
Name	Address	Port	Transport	Status
LHR-SRE-CP1-FUSIONPROD1	10.50.20.20	5066	TCP	UP
AMS-SRE-CP1-FUSIONPROD1	172.16.2.82	5066	TCP	UP
AMS-SRE-CP2-FUSIONPROD1	172.16.2.15	5066	TCP	UP

SRE-CP2-AMS-FUSIONPROD1				
Name	Address	Port	Transport	Status
LHR-SRE-CP1-FUSIONPROD1	10.50.20.20	5066	TCP	UP
AMS-SRE-CP2-FUSIONPROD1	172.16.2.82	5066	TCP	UP
AMS-SRE-CP1-FUSIONPROD1	172.16.2.15	5066	TCP	UP

SRE-CP3-LHR-FUSIONPROD1				
Name	Address	Port	Transport	Status
LHR-SRE-CP1-FUSIONPROD1	10.50.20.20	5066	TCP	UP
AMS-SRE-CP2-FUSIONPROD1	172.16.2.82	5066	TCP	UP
AMS-SRE-CP1-FUSIONPROD1	172.16.2.15	5066	TCP	UP

Figure 1: SIP Agents

2.11 Accounting Generation

Target servers: master EM

Run the command `/opt/sre/bin/sre-admin monitor accounting summary` to get the count of currently active calls from an accounting point of view.

Example:

```

1 [root@SRE-33-EM1 ~]# /opt/sre/bin/sre-admin monitor accounting summary
2 Hostname      Open calls
3 -----
4 total                7
  
```

List files in the directory `/data/sre/accounting/` to ensure that accounting files are properly generated. There should be exactly one CDR file suffixed with `.inprogress` and a recent timestamp in the filename and possibly several CDR files that are not updated anymore.

Example:

```

1 [sre@SRE-33-EM1 ~]$ ls -lrt /data/sre/accounting/
2 total 124216
3 drwxrwxr-x. 3 sre      sre          19 Mar 19 14:09 http
4 drwxrwxr-x. 2 sre      sre           6 Mar 19 14:09 events
5 drwxrwxr-x. 3 sre      sre          19 Mar 19 14:09 enum
6 -rw-r--r--  1 sre      sre    46866966 Jun 20 13:04 cdr-2024-06-20T13:00:00
7 -rw-r--r--  1 sre      sre    46682213 Jun 20 13:09 cdr-2024-06-20T13:05:00
8 drwxrwxr-x. 2 sre      sre           71 Jun 20 13:13 state
9 -rw-r--r--  1 sre      sre    28939111 Jun 20 13:13 cdr-2024-06-20T13
   ↪ :10:00.inprogress
  
```

2.12 Accounting Synchronization

Target servers: EMs

The accounting synchronization mechanism between the two EMs can be verified by listing the files in `/data/sre/accounting/state`. The file suffixed with `.checkpoint` represents the last checkpoint of the accounting status at that moment and should be identical on both servers. The file suffixed with `.oplog` contains the delta of accounting information relative to the last checkpoint file. This file is continuously synchronized between both EMs and should be approximately the same size.

Example:

```

1 [sre@SRE-33-EM1 ~ ~]$ ls -lrt /data/sre/accounting/state
2 total 45500
3 -rw-r--r--  1 sre sre 29615256 Jun 20 13:15 00000000000003ef3.checkpoint
4 -rw-r--r--  1 sre sre 13976487 Jun 20 13:15 00000000000003ef3.oplog
5
6 [sre@SRE-33-EM2 ~ ~]$ ls -lrt /data/sre/accounting/state
7 total 59388
8 -rw-r--r--  1 sre sre 29615256 Jun 20 13:15 00000000000003ef3.checkpoint
9 -rw-r--r--  1 sre sre 25596460 Jun 20 13:16 00000000000003ef3.oplog
  
```