FortiSIEM - Disaster Recovery Procedures - NFS

Version 5.4.0

FORTINET DOCUMENT LIBRARY
https://docs.fortinet.com

FORTINET VIDEO GUIDE
https://video.fortinet.com

FORTINET BLOG
https://blog.fortinet.com

CUSTOMER SERVICE & SUPPORT
https://support.fortinet.com

FORTINET TRAINING & CERTIFICATION PROGRAM
https://www.fortinet.com/support-and-training/training.html

NSE INSTITUTE
https://training.fortinet.com

FORTIGUARD CENTER
https://fortiguard.com/

END USER LICENSE AGREEMENT
https://www.fortinet.com/doc/legal/EULA.pdf

FEEDBACK
Email: techdoc@fortinet.com

10/13/2020
FortiSIEM 5.4.0 Disaster Recovery Procedures - NFS

TABLE OF CONTENTS

Change Log

Date	Change Description
04/25/2018	Initial version of FortiSIEM - Disaster Recovery Procedures
08/19/2019	Revision 1: Updated the location of the image download site.
11/25/2019	Revision 2: Updated the recovery procedures.
03/30/2020	Release of Disaster Recovery Procedures for 5.3.0.
08/15/2020	Revision 3: All new content for Disaster Recovery.

Disaster Recovery

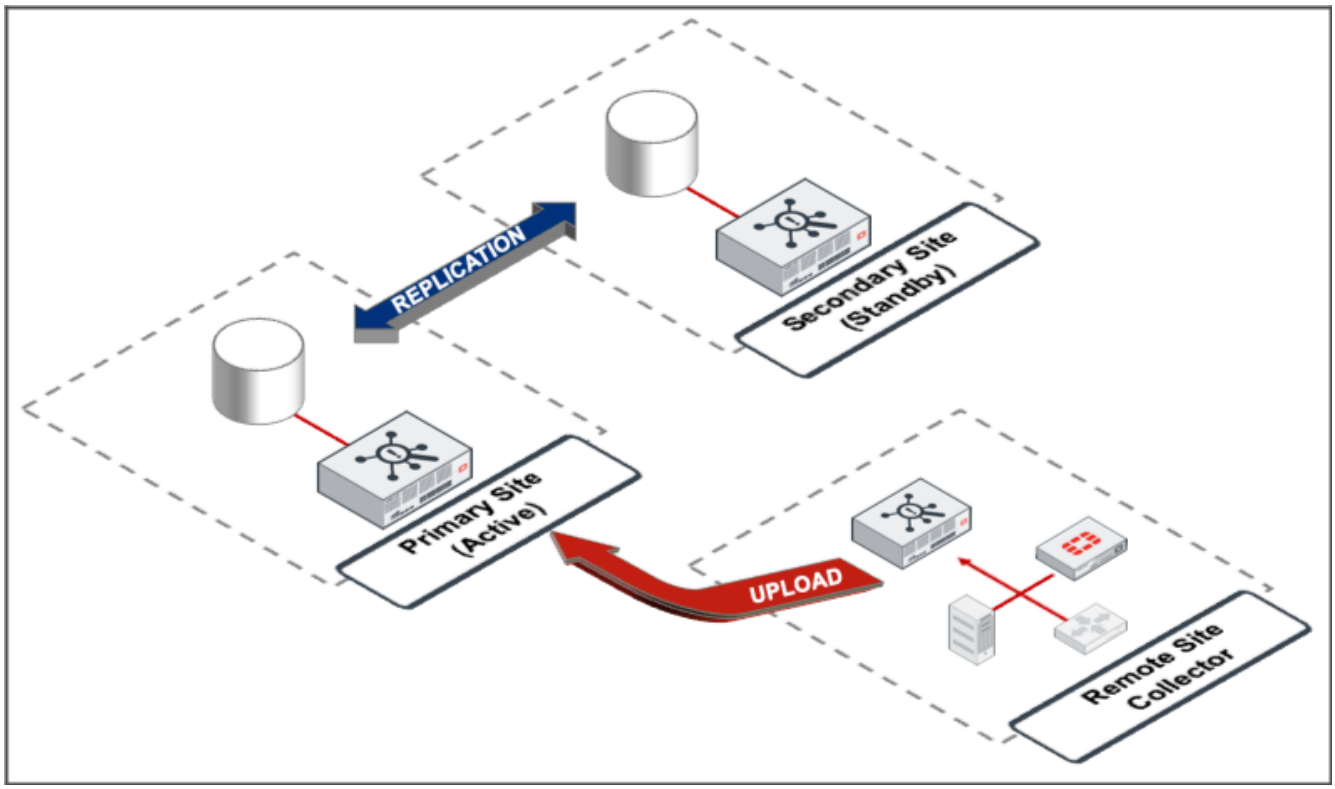
The following sections describe how to enable and work with the FortiSIEM Disaster Recovery (DR) feature.

Introduction

Understanding the FortiSIEM DR Feature

FortiSIEM has a replication feature, designed for those customers who require full disaster recovery capabilities, where one site is designated to be the Primary (active) and the other the Secondary (standby) site. The two systems replicate the Primary sites databases.

This requires a second fully licensed FortiSIEM system, where the Primary and Secondary Sites are identically setup in terms of Supervisor, Workers, and event storage.

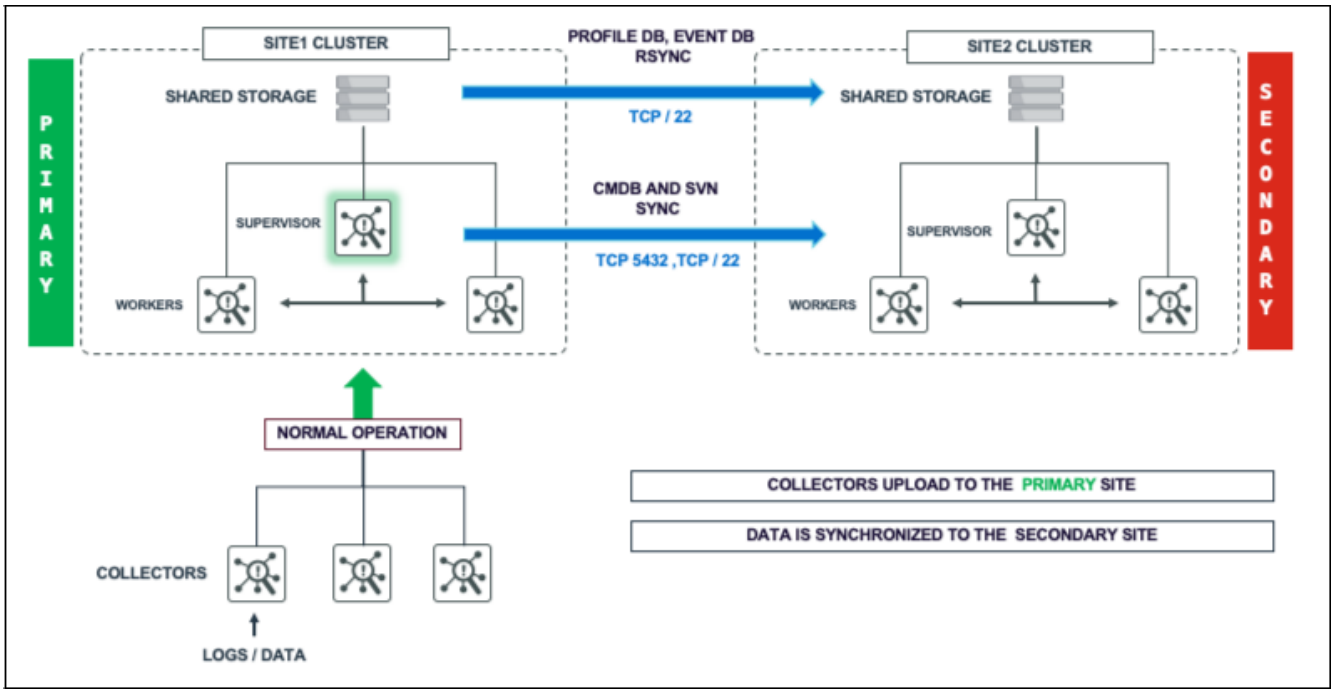


Under normal operations, if collectors are being used, these upload to the Primary site and will buffer by design when this site is not available. If DR is used, and a disaster occurs, then these same collectors will revert to uploading to the Secondary site which will now be designated as the Primary/Active site.

FortiSIEM runs as a cluster (or single node for a SMB) with Super, Worker, Report Server, and Collectors nodes.

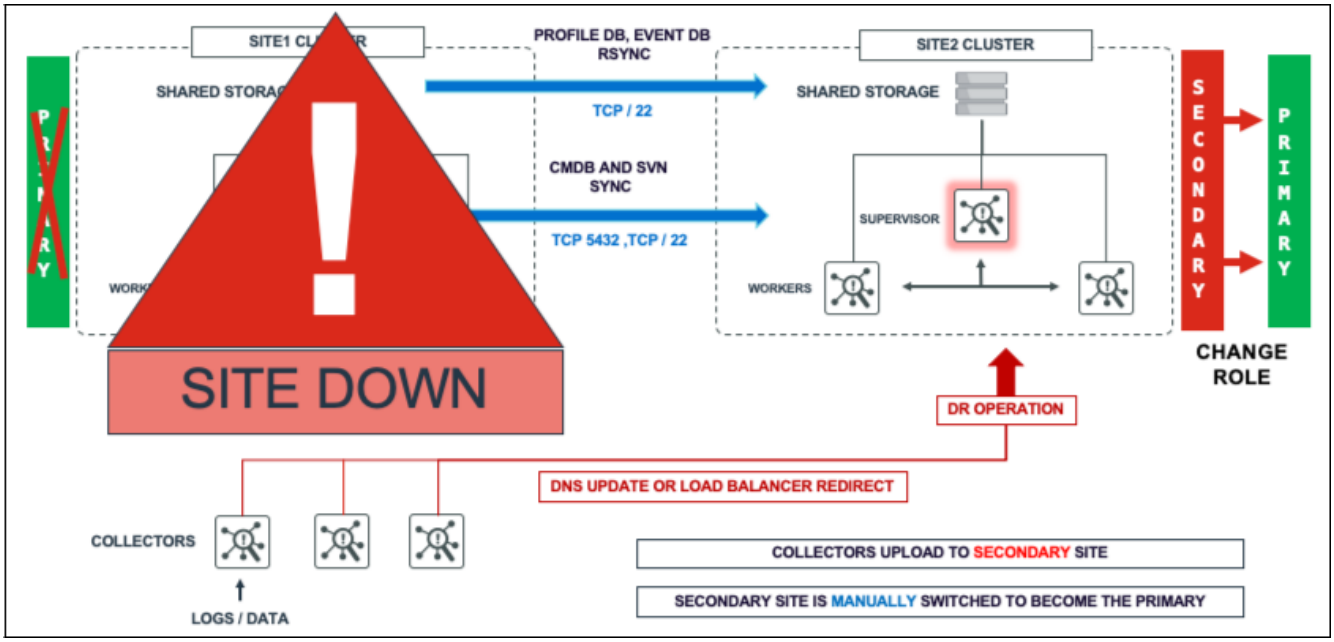
To provide DR features, FortiSIEM must have a Secondary system ready on standby to take over operations, with the following databases replicated from the Primary site:

- The CMDB residing in a PostGreSQL database.
- Device configurations residing in SVN on the Supervisor node.
- Profile data residing on SQLite databases on the Supervisor node.
- Event DB can be on a local disk (for small single node deployments) or on external storage - NFS Event DB or Elasticsearch for cluster deployments.



When disaster strikes:

1. The Secondary must become the Primary FortiSIEM.
2. DNS Changes must be made so that users will logon to Secondary Supervisor, and that Collectors will send events to Secondary Workers.



When the Old Primary is recovered and powered up, it will sync missing data with the Secondary site (the Active Primary FortiSIEM).

When the user decides to return to the pre-disaster setup, the user can switch the roles of Primary and Secondary.

Prerequisites for a Successful DR Implementation

- Two separate FortiSIEM licenses - one for each site.
- The installation at both sites must be identical - workers, storage type, archive setup, report server setup, hardware resources (CPU, Memory, Disk) of the FortiSIEM nodes.
- DNS Names are used for the Supervisor nodes at the two sites. Make sure that users, collectors, and agents can access both Supervisor nodes by their DNS names.
- DNS Names are used for the Worker upload addresses.
- TCP Ports for HTTPS (TCP/443), SSH (TCP/22) and PostGreSQL (TCP/5432) are open between both sites.

Understanding the Requirements for DNS Names

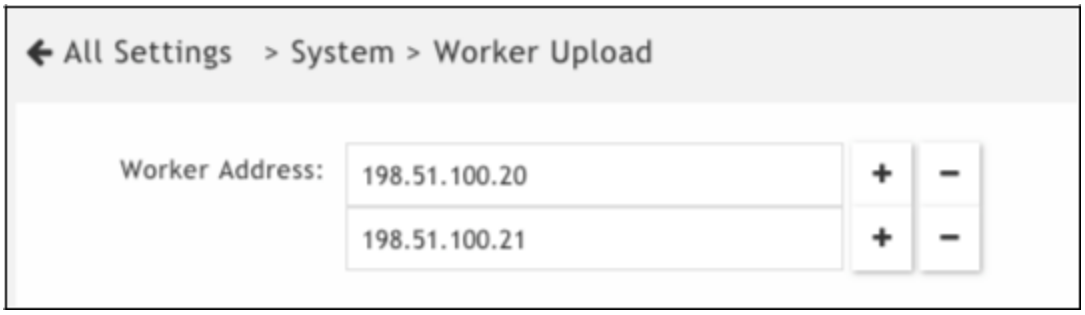
It is important to understand your FortiSIEM environment and plan ahead in terms of communications from users, agents and collectors.

Worker Upload

- Performing Collector Registration
- Agent Communications

Each entry in the **Worker Upload** address list is given to Collectors at registration (and periodically in communication to the Supervisor) to instruct where to upload customer event data.

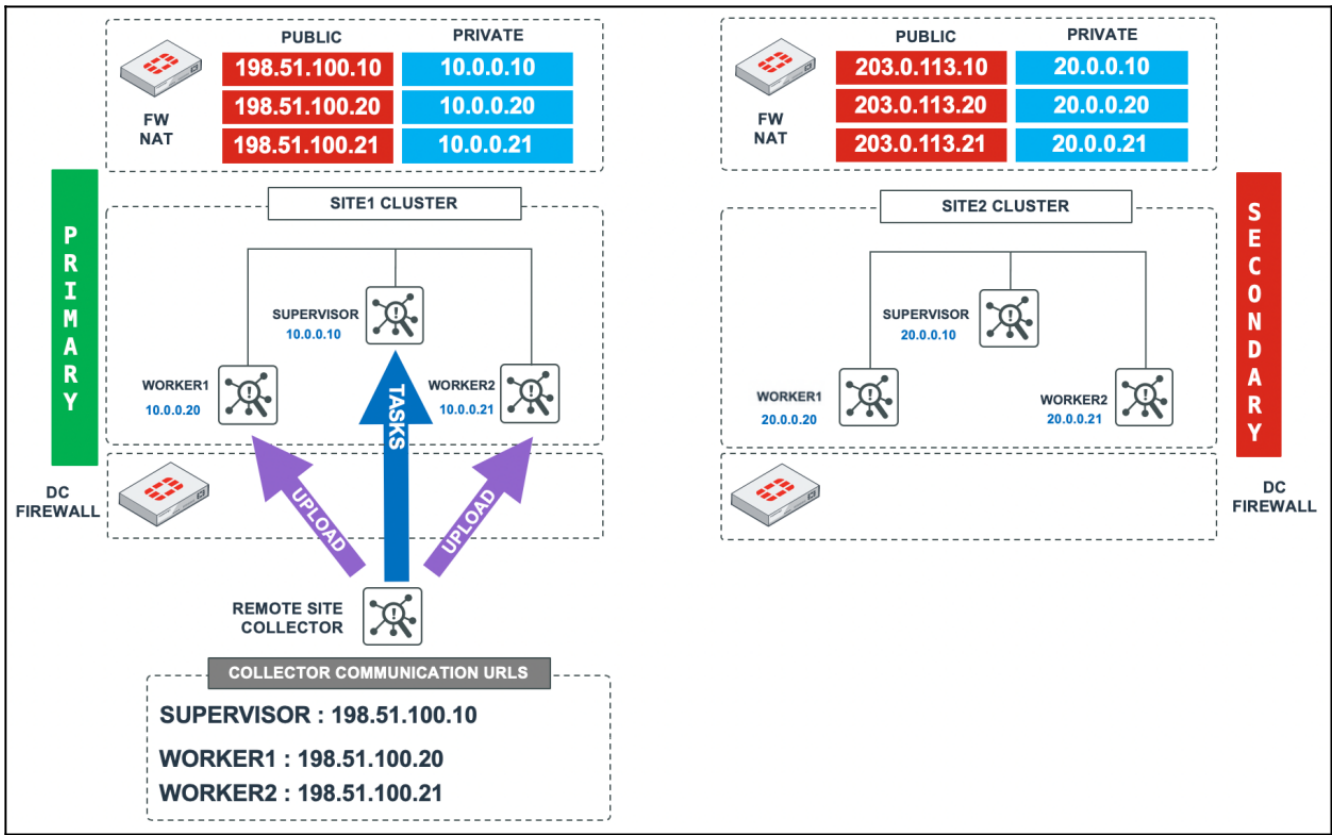
An example is shown below, where the customer has *not* followed best practice advice and used IP Addresses and not FQDNs.



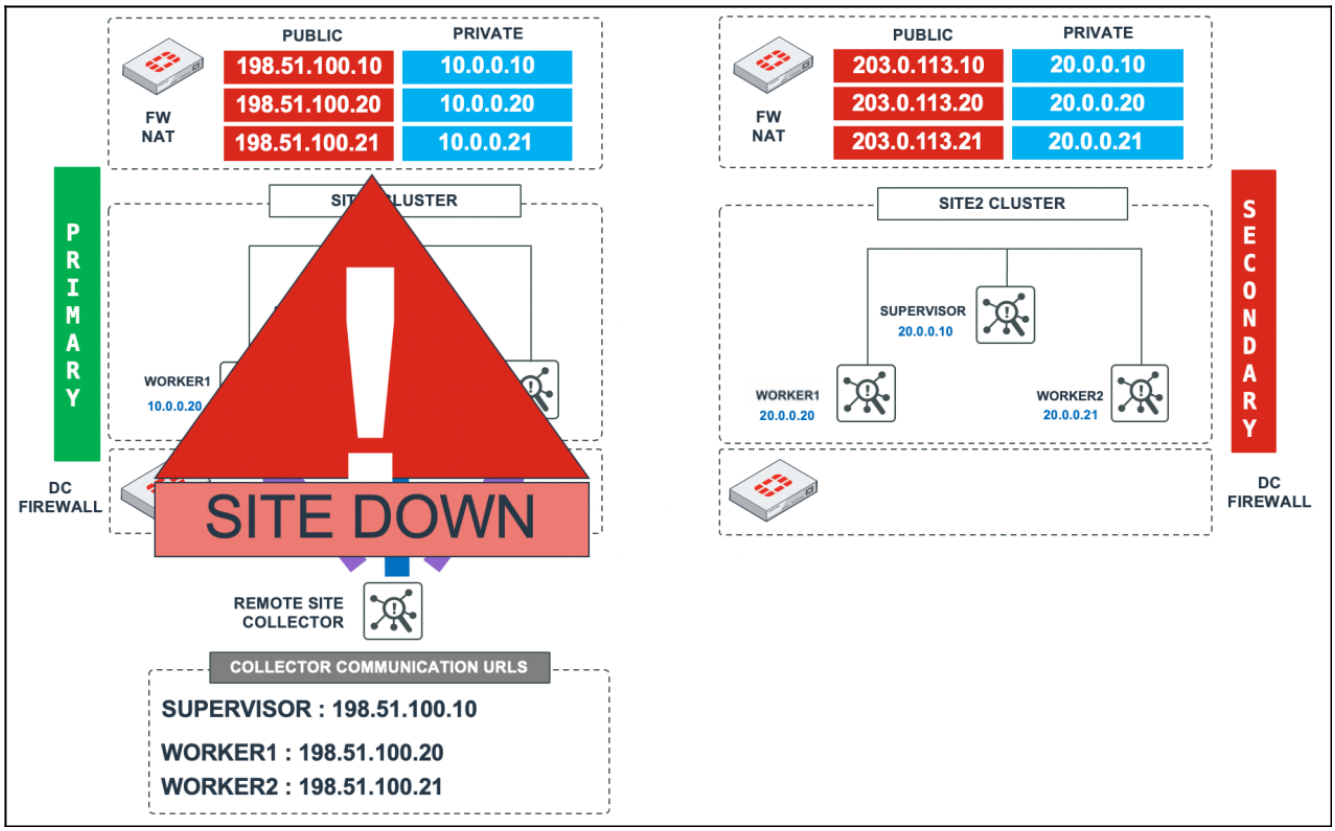
In addition to the Worker Upload entries, Collectors also maintain communication with the Supervisor node, to receive jobs/tasks and report Collector health data. When Collectors register for the first time with the Supervisor node, these communication addresses are stored for this purpose.

Why is using IP addresses for Collector registration and Worker Upload settings bad when it comes to DR planning?

Consider the environment below where only IP addresses have been used. During normal operations Collector traffic flows to the Workers at the Primary site and the Collector maintains communications with the Supervisor. This all works fine until the Primary site has a disaster.



At this point, when the Primary node is unavailable. The remote Collector nodes are essentially hard-coded (by IP) to talk to the Primary site only. Even if the Secondary node is up and operational and promoted to be the Primary node, Collectors are unable to upload logs or get any tasks from the Supervisor node due to the old Primary sites IPs being used.



A much better approach is to utilize DNS.

This allows name resolution to control which Supervisor, Primary, or Secondary is currently active and which worker addresses to attempt to upload customer data to. DNS “A” records are created for the Supervisor nodes at both sites, and a “CNAME” is used to determine which is active, which has a small time to live (TTL) value.

The Worker Upload settings reference DNS addresses:

External DNS Example

Node	DNS Record Type	Name	IP/Alias
Supervisor (Primary)	A	site1.fsm-mssp.com	198.51.100.10
Supervisor (Secondary)	A	site2.fsm-mssp.com	203.0.113.10
Active Supervisor	CNAME	site.fsm-mssp.com	site1.fsm-mssp.com

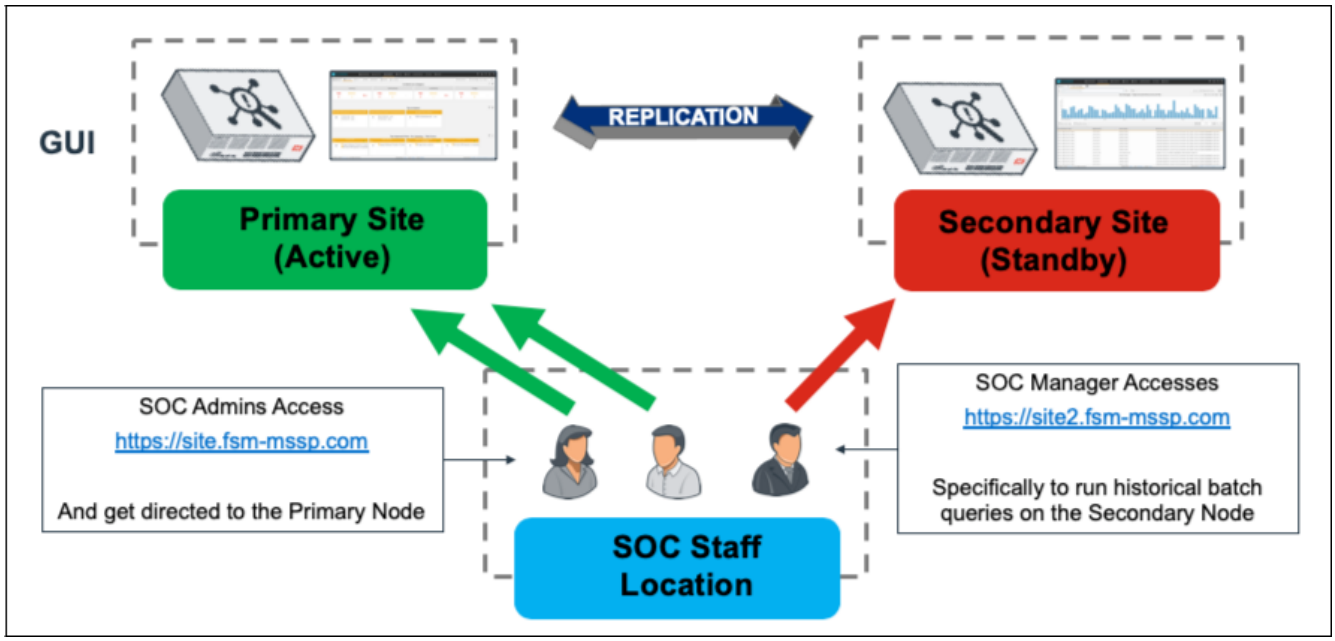
Node	DNS Record Type	Name	IP/Alias
Worker1 (Primary)	A	worker1.fsm-mssp.com	198.51.100.20
Worker2 (Primary)	A	worker2.fsm-mssp.com	198.51.100.21

For the internal DNS records, again both internal Supervisor addresses are listed with a CNAME to determine the current Primary GUI to logon to for SOC operators. (If public certificates are being used, then a Wildcard cert should be used to achieve this).

Internal DNS Example

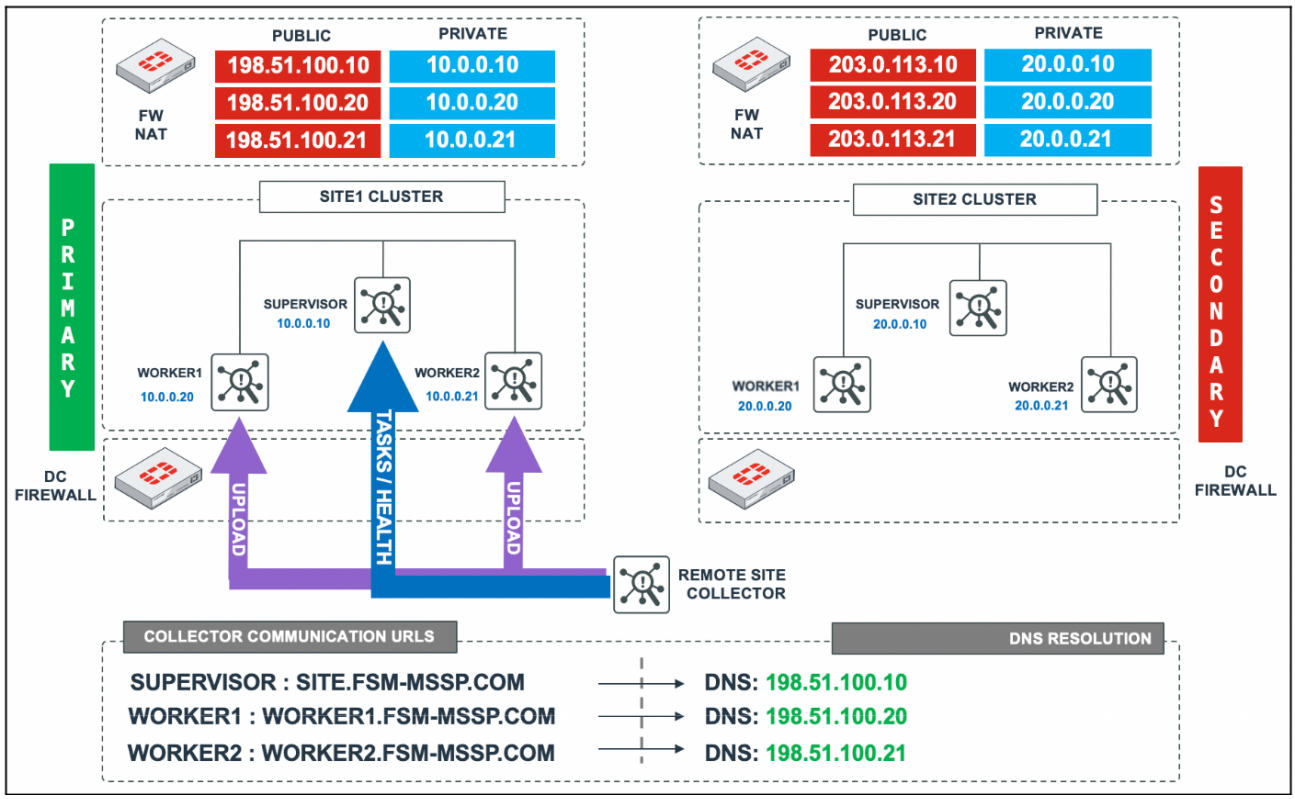
Node	DNS Record Type	Name	IP/Alias
Supervisor (Primary)	A	site1.fsm-mssp.com	10.0.0.10
Supervisor (Secondary)	A	site2.fsm-mssp.com	20.0.0.10
Active Supervisor	CNAME	site.fsm-mssp.com	site1.fsm-mssp.com

By utilizing internal DNS, then SOC operators can always access the active Supervisor GUI via `site.fsm-mssp.com`, but as will be discussed later, the Secondary Standby Supervisor can always be accessed if required.

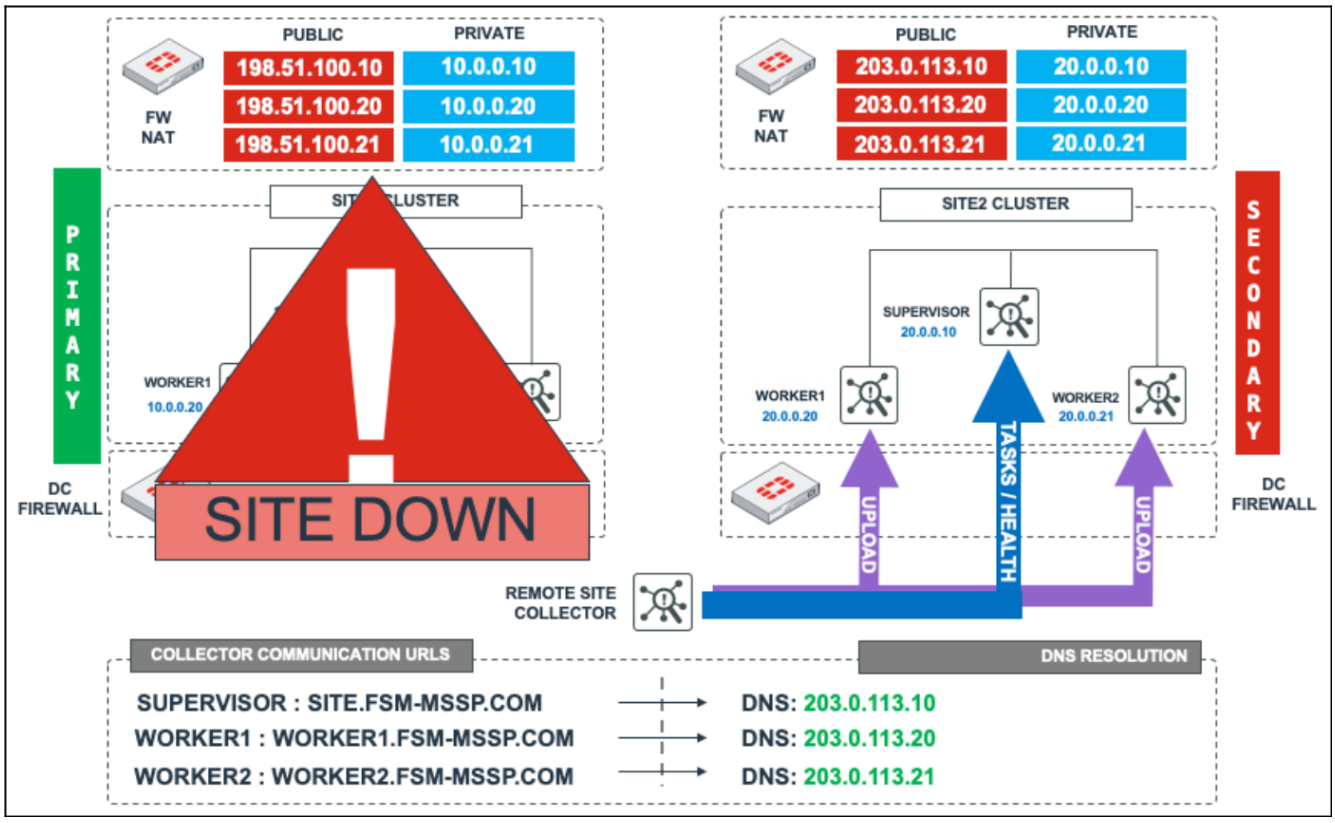


Note: Any DNS changes, are made **manually** in the event of a failover.

As can be seen below, using DNS the Collectors are instructed to talk to the Active site.



And in the event of a failure at the Primary Site, they can be easily instructed to communicate with the Supervisor and Workers at the Secondary site which will be manually switched to be the Primary Role site.



Note : In addition to DNS changes being made manually, the process for promoting the Secondary Supervisor to be the Primary Role Supervisor node is also made manually in the FortiSIEM GUI.

Performing Collector Registration

When registering Collectors, you should ignore the Supervisor-IP requirement, and instead use the CNAME for the Active Supervisor node.

```
[root@collector ~]# phProvisionCollector

Usage: phProvisionCollector --add <Organization-user-name> <Organization-user-password>
     <Supervisor-IP> <Organization-name> <Collector-name>
```

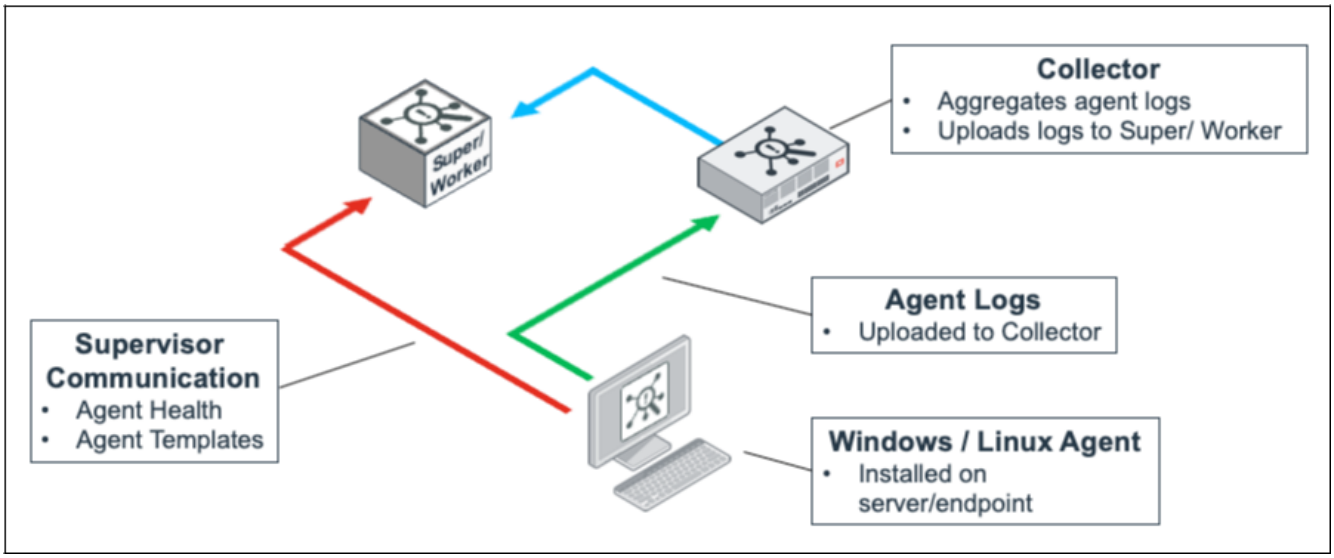
An example using `site.fsm-mssp.com` is shown below. Since Collectors always communicate with the Supervisor node, communications can be easily restored to the Primary via a simple DNS change.

```
[root@collector ~]# phProvisionCollector --add admin admin*1 site.fsm-mssp.com super
     collector.fsm-mssp.com

Continuing to provision the Collector
Adding Collector (collector.fsm-mssp.com) to Super (site.fsm-mssp.com) with Organization
     (super)
This collector is registered successfully, and will be rebooted soon.
```

Agent Communications

The communications for FortiSIEM Windows and Linux agents follow a similar path to the above. Agents register with the Supervisor node, and maintain this communication to receive updated templates and report health. One or more

Collectors are assigned to each agent as the node or nodes to deliver event data.



For best practice, agent registration should use the Supervisor CNAME. This way, if the Primary Site is a totally destroyed, you can still easily ensure agent communication to the DR site Supervisor via a simple DNS change and still make template changes etc.

The Windows installation file `installSettings.xml` is shown:

```
<?xml version="1.0" encoding="utf-8"?>
<InstallConfig Version="1">
  <Org>
    <ID>1</ID>
    <Name>Super</Name>
  </Org>
  <Super>
    <Name>site.fsm-mssp.com</Name>
    <Port>443</Port>
  </Super>
  <Registration>
    <Username>super/agent_admin</Username>
    <Password>admin*2</Password>
  </Registration>
  <Proxy/>
  <SSLCertificate>ignore</SSLCertificate>
</InstallConfig>
```

The same concept also applies to deploying Linux agents.

Configuring Disaster Recovery

The following sections describe how to configure FortiSIEM primary and secondary nodes for disaster recovery.

- FortiSIEM Primary Node
- FortiSIEM Secondary Node

FortiSIEM Primary Node

On the Primary FortiSIEM node in the GUI:

1. Navigate to **Admin > Settings > Database > Replicate** (or **Replication** in 5.3+).
2. Select **Enable Replication**.
3. For the **Primary**, enter the **Host** and **IP** information.
4. For the **UUID**, obtain the **Hardware ID** value through an **SSH session** on the Primary by entering the following command:
   `/opt/phoenix/bin/phLicenseTool --show`
   For example:

```
[root@site1 ~]# /opt/phoenix/bin/phLicenseTool --show
License Information:
Attribute                       Value                                   Expiration Date
Serial Number                   FSMS0100[redacted]
Hardware ID                     564[redacted]C-0247-87C2-[redacted]3B56EFFF
License Type                    Enterprise
Devices                         1500                                    Mar 17, 2021
Endpoint Devices                N/A                                     N/A
Additional EPS                  N/A                                     N/A
```

5. For the **CMDB Replication** mount point, enter `/something` (this can be any fake mount point). (**Note:** this value is not actually used today).
6. Under **Configuration and Profile Replication**, generate the **SSH Public Key** and **SSH Private Key Path** by entering the following in your SSH session:

```
su - admin
ssh-keygen -t rsa -b 4096

#Leave the file location as default, and press enter at the passphrase prompt.
```

   The output will appear similar to the following:

```
Generating public/private rsa key pair.
Enter file in which to save the key (/opt/phoenix/bin/.ssh/id_rsa):
Created directory '/opt/phoenix/bin/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /opt/phoenix/bin/.ssh/id_rsa.
Your public key has been saved in /opt/phoenix/bin/.ssh/id_rsa.pub.
The key fingerprint is:
a9:43:88:d1:ed:b0:99:b5:bb:e7:6d:55:44:dd:3e:48 admin@site1.fsmtesting.com
The key's randomart image is:
+--[ RSA 4096]----+
| ....|
| . . E. o|
```

7. For the **SSH Public Key** enter the following command, and copy **all** of the output into the field:
   `cat /opt/phoenix/bin/.ssh/id_rsa.pub`
8. For the **SSH Private Key Path**, enter the following into the field: `/opt/phoenix/bin/.ssh/id_rsa`.
9. Exit the `admin` user in the SSH session by entering the following command:
   `exit`
10. Select a **Replication Frequency**, with a minimum of 10 minutes.
    Note: For Local/NFS Event DB installs, this value is used for SVN and ProfileDB synchronization.
11. Select the EventDB Replication check box if you would also like the Event Database to be replicated.
    Note: For Local/NFS Event DB installs, `rsync` is used and this runs continually in the background.

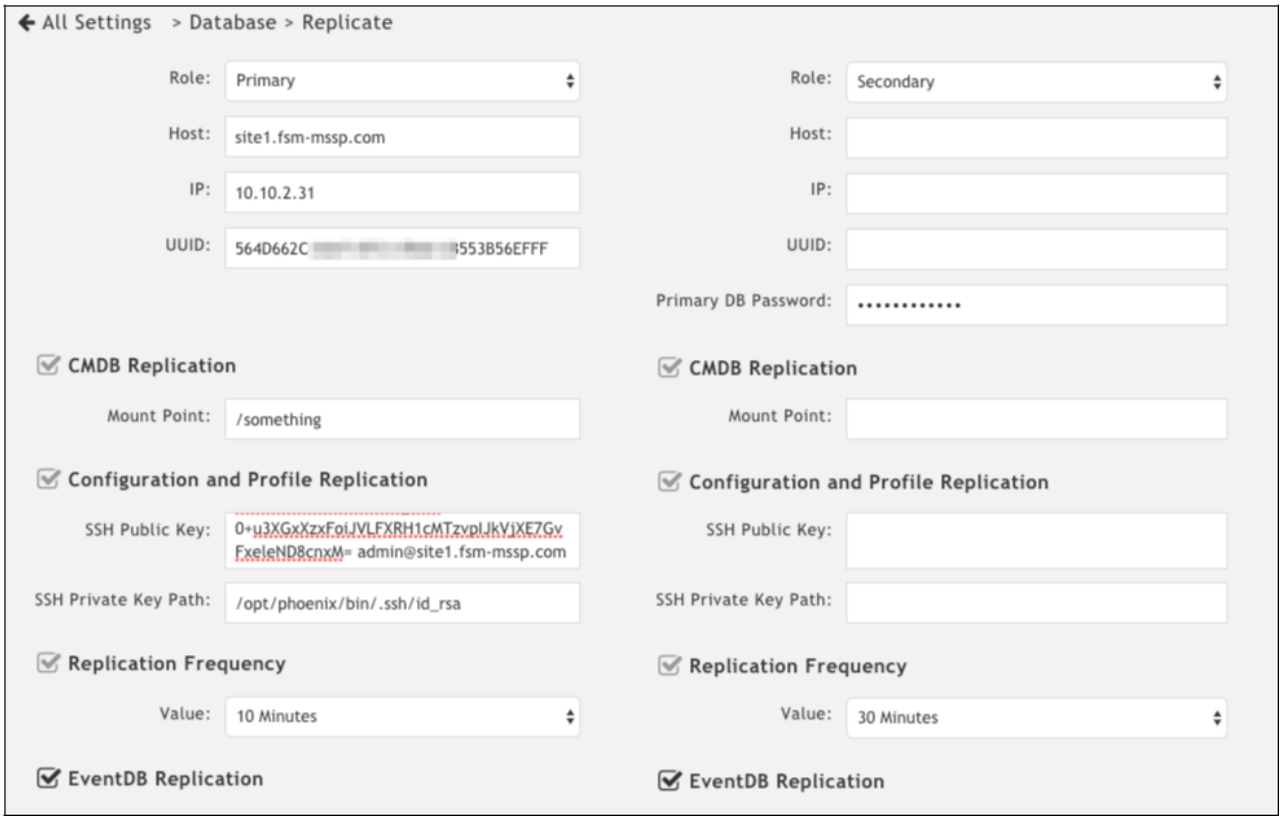
12. Finally, run the following command in the primary SSH session and enter the output under the Role: Secondary, **Primary DB Password** field.
Note: The **Primary DB Password** field initially looks like it has a populated value. This is **false**, and the following step must be completed.


```
/opt/phoenix/bin/phLicenseTool –showDatabasePassword
```

```
[root@site1 ~]# /opt/phoenix/bin/phLicenseTool --showDatabasePassword
EPicip@8CORi
```

Keep a copy of this password for Step 4 under FortiSIEM Secondary Node.

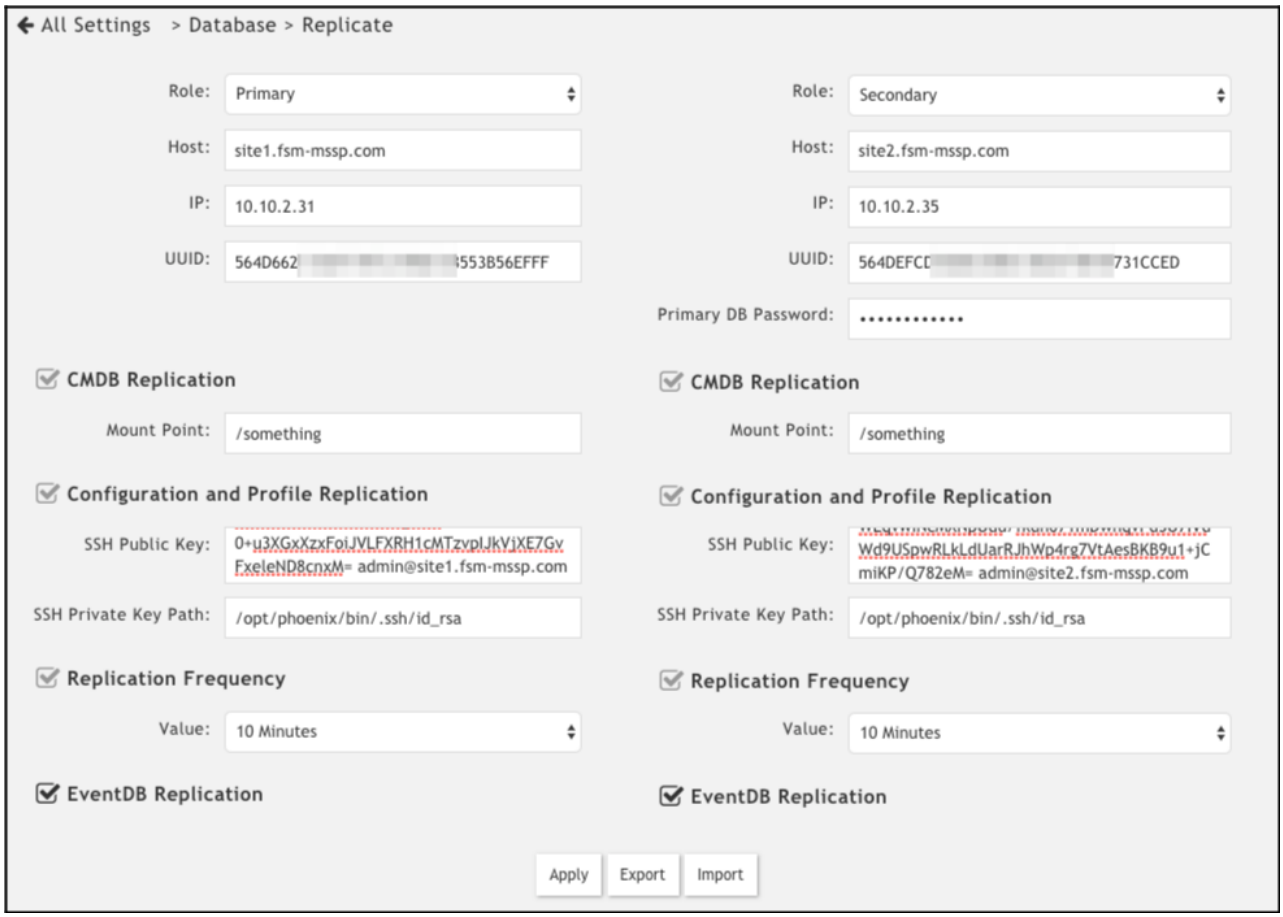
The completed Primary role details will appear similar to the following:



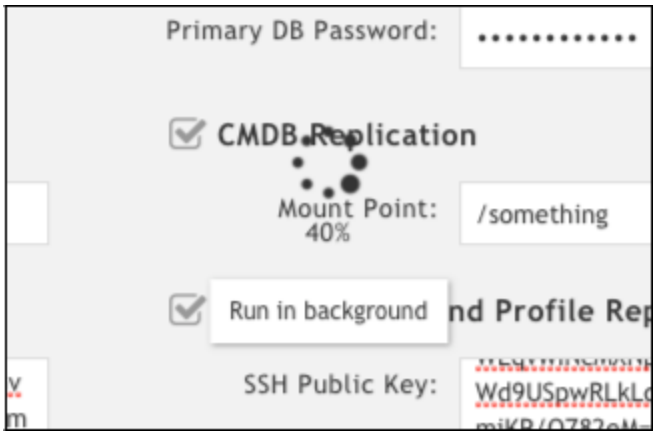
Now move on to configuring the Secondary nodes details.

13. For the **Secondary**, enter the **Host** and **IP** information.
14. For the **UUID**, obtain the **Hardware ID** value through an SSH session on the secondary node by entering the following command:
`/opt/phoenix/bin/phLicenseTool --show`
15. For the **CMDB Replication** mount point enter `/something` (this can be any fake mount point). **Note:** this value is not actually used today.

16. Under **Configuration and Profile Replication**, generate the **SSH Public Key** and **SSH Private Key Path** by entering the following in your SSH session on your secondary node:
```
su - admin
ssh-keygen -t rsa -b 4096

#Leave the file location as default, and press enter at the passphrase prompt.
```
17. For the **SSH Public Key** enter the following command, and copy **all** of the output into the field:
`cat /opt/phoenix/bin/.ssh/id_rsa.pub`
18. For the **SSH Private Key Path**, enter the following into the field: `/opt/phoenix/bin/.ssh/id_rsa`.
19. Exit the admin user in the SSH session by entering the following command:
`exit`
20. Select the same **Replication Frequency** as were set on the Primary node.



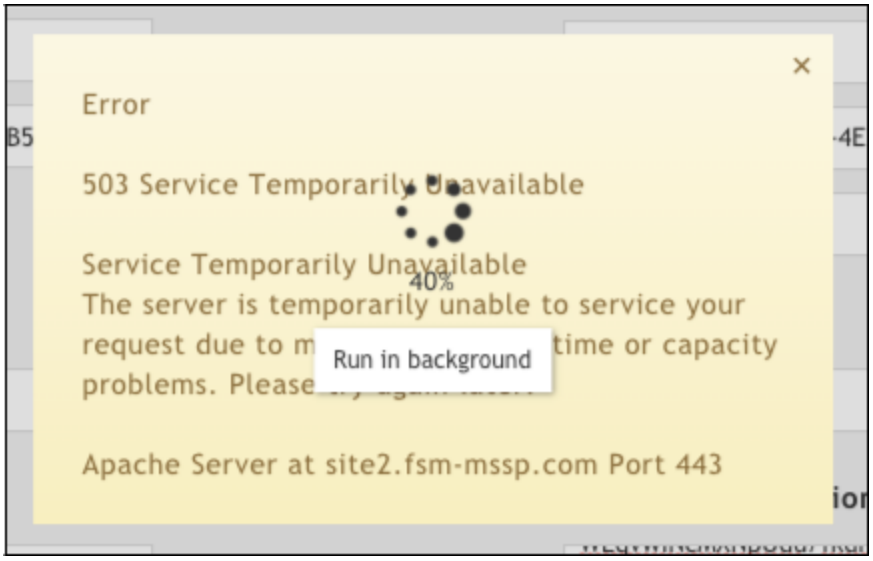
21. Click **Export** and download a file named `replicate.json` . **Note:** This file contains all of the DR settings, except the Primary DB Password.
22. Click **Apply**.
Note: This should result in the following message in the GUI, where it will stick at 40% until the Secondary node configuration is completed.



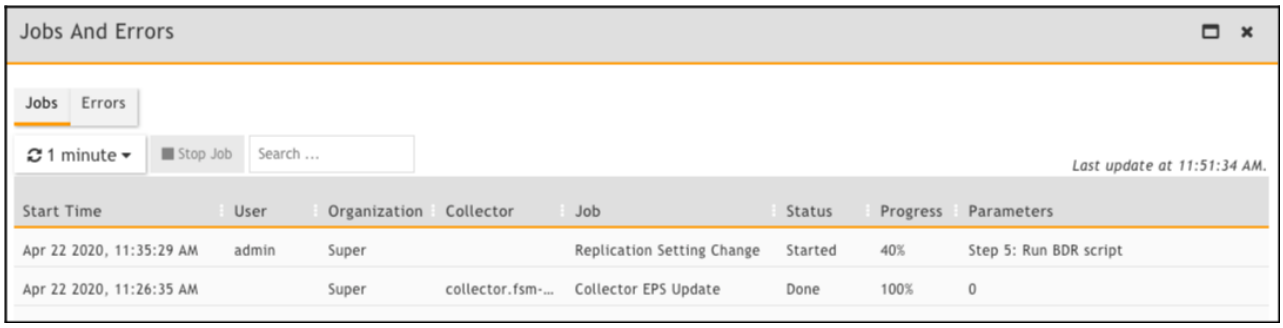
FortiSIEM Secondary Node

On the Secondary FortiSIEM node, log into the FortiSIEM GUI:

1. Navigate to **Admin > Settings > Database > Replicate** (or **Replication** in 5.3+).
2. Select **Enable Replication**.
3. Click **Import**, and select the `replicate.json` file downloaded from the Primary node.
4. Copy the **Primary DB Password**, from Step 12 in FortiSIEM Primary Node.
   If you do not have the password handy, run the following command on the Primary node's SSH session and enter the output under the **Primary DB Password** field.
   ```
   #On the PRIMARY node
   /opt/phoenix/bin/phLicenseTool -showDatabasePassword
   ```
5. Click **Apply**.
   At this point, the Secondary node will display the following while the backend scripts are disabling services, etc.



Note: There will be disruption of services on both nodes, while the setup is taking place behind the scenes. While initial replication is taking place, you can view the status on the Primary node, Jobs, and Errors (Red Alert Symbol, top right of GUI) on what Step (out of 10) the process is currently at.



Backend logs will better display the current status of the replication and DR scripts being run.

Troubleshooting Disaster Recovery Setup

- Backend Logs
- Alternative Logs
- FortiSIEM Services Status on Primary and Secondary Node
- Understanding FortiSIEM Operations in DR Mode
- Verify Elasticsearch Snapshots for Data Replication

Backend Logs

On both the Primary and Secondary nodes, use the `cat` command to view the backend logs:

```
cat /opt/phoenix/config/pgMasterRep/bdrlog
```

Note: This process can take a while. The output below was a new installation with minimal test data and it took around 5 minutes to complete, For a live system it will take a lot longer. (It is recommended to `tail -f` the log).

Successful Enablement of Disaster Recovery on the Primary node

```
[root@site1 ~]# cat /opt/phoenix/config/pgMasterRep/bdrlog
bdr_connection_count for 10.10.2.31 is
back up pg_hba.conf and postgresql.conf
setting bdr configuration ...
inserting pg_hba records ...
finished setting bdr configuration
restart postgresql9.4
ext_btree_gist_count is 0
ext_bdr_count is 0
bdr_node1_count is 0
please wait the bdr building ...
no primary file exist, add primary file
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
```

```
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Waiting for Secondary 10.10.2.35 to finish up synch Primary CMDB
Secondary 10.10.2.35 finished synch Primary CMDB
```

Successful Enablement of Disaster Recovery on the Secondary node

```
[root@site2 ~]# cat /opt/phoenix/config/pgMasterRep/bdrlog
slave - bdr_connection_count for 10.10.2.31 is
Backup unsynchable system properties from ph_sys_conf before replicating CMDB from Primary
     CMDB
dump file ph_sys_server.sql and ph_sys_conf.sql ...
Shutdown App Server to preparing synch CMDB from primary Stopping crond: [ OK ]
Stopping postgresql-9.4 service: [ OK ]
wait port 5432 to stop...
port 5432 stopped
join connection according cmdb buffer ... master ip = 10.10.2.31, slave ip = 10.10.2.35 bdr_
     init_copy: starting ...
Getting remote server identification ...
Detected 1 BDR database(s) on remote server
Updating BDR configuration on the remote node:
phoenixdb: creating replication slot ...
phoenixdb: creating node entry for local node ...
Creating base backup of the remote node...
194081/194081 kB (100%), 1/1 tablespace
Creating restore point on remote node ...
Bringing local node to the restore point ...
Transaction log reset
Initializing BDR on the local node:
phoenixdb: adding the database to BDR cluster ...
All done
please wait the connection building ...
synching CMDB from Primary, status= c
Done synching CMDB from Primary
DELETE 1
DELETE 8
DELETE 0
DELETE 58
DELETE 6
DELETE 361
import sql ph_sys_server.sql ...
COPY 1
COPY 1
Restoring non-replicable system properties
```

```
COPY 3
Stop running all quartz jobs on secondary
restart App Server ...
Starting crond: [ OK ]
ALTER ROLE
Done replication CMDB
```

Alternative Logs

It is also possible to track the DR scripts by examining the `phoenix.log` file. Use the grep command on both Primary and Secondary nodes to track progress.

grep "521-ReplicationRoleChange" /opt/phoenix/log/phoenix.log

```
[root@site1 log]# grep "521-ReplicationRoleChange" /opt/phoenix/log/phoenix.log 2020-04-
     15T20:04:55.143563+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=6866,[phLogDetail]=521-
     ReplicationRoleChange, Step 1.1: check command type
2020-04-15T20:04:55.143667+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=6875,[phLogDetail]=521-
     ReplicationRoleChange, Step 1.2: check command data
2020-04-15T20:04:55.143729+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=6882,[phLogDetail]=521-
     ReplicationRoleChange, Step 2: load replication setting
2020-04-15T20:04:55.183173+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr
ocess.cpp,[lineNumber]=6897,[phLogDetail]=521-ReplicationRoleChange, Step 3: handle
     replication
role change
2020-04-15T20:04:55.183344+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=6916,[phLogDetail]=521-
     ReplicationRoleChange, Step 3.1: handle replication role change on super
2020-04-15T20:04:55.183442+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=6919,[phLogDetail]=521-
     ReplicationRoleChange, Step 3.2: prepare role info
2020-04-15T20:04:55.218565+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=6942,[phLogDetail]=521-
     ReplicationRoleChange, Step 3.3: update SSH keys
2020-04-15T20:04:55.265239+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr
ocess.cpp,[lineNumber]=6955,[phLogDetail]=521-ReplicationRoleChange, Step 3.4: update SSH
configurations
2020-04-15T20:04:55.312994+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=6970,[phLogDetail]=521-
     ReplicationRoleChange, Step 3.5: run database replication script
2020-04-15T20:19:39.991395+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=6992,[phLogDetail]=521-
     ReplicationRoleChange, Step 3.6: wait appsvr back
```

```
2020-04-15T20:19:40.056744+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=7001,[phLogDetail]=521-
     ReplicationRoleChange, Step 3.7: update service and SVN password for the first time
2020-04-15T20:19:40.542801+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=7198,[phLogDetail]=521-
     ReplicationRoleChange, Step 3.7.1: get sevice user
2020-04-15T20:19:40.542861+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=7206,[phLogDetail]=521-
     ReplicationRoleChange, Step 3.7.2: get secondary host
2020-04-15T20:19:40.543375+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=7225,[phLogDetail]=521-
     ReplicationRoleChange, Step 3.7.3: update secondary
2020-04-15T20:19:40.670656+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr
ocess.cpp,[lineNumber]=7013,[phLogDetail]=521-ReplicationRoleChange, Step 3.8: restart
     processes
on super
2020-04-15T20:19:40.711471+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=7021,[phLogDetail]=521-
     ReplicationRoleChange, Step 3.9: notify processes on super
2020-04-15T20:19:40.751225+02:00 site1 phMonitorSupervisor[4874]:
[PH_GENERIC_INFO]:[eventSeverity]=PHL_INFO,[procName]=phMonitorSupervisor,
     [fileName]=phMonitorPr ocess.cpp,[lineNumber]=7031,[phLogDetail]=521-
     ReplicationRoleChange, Step 3.10: finish role change on super
```

FortiSIEM Services Status on Primary and Secondary Node

On the Primary node, all FortiSIEM `ph*` services will be in an "up" state. (They will all restart, but it may take up to 3 to 5 minutes to restart.)

On the Secondary node, most `ph*` services will be "down" except for `phQueryMaster`, `phQueryWorker`, `phDataPurger`, and `phMonitor`.

This can be seen in the following images. They illustrate the Primary Node and Secondary Node after a full CMDB sync:

```
Every 1.0s: /opt/phoenix/bin/phstatus.py

System uptime:  18:11:06 up 52 min,  1 user,  load average: 0.52, 0.44, 0.27
Tasks: 25 total, 0 running, 24 sleeping, 0 stopped, 0 zombie
Cpu(s): 8 cores, 3.1%us, 0.9%sy, 0.0%ni, 96.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 8060332k total, 7628060k used, 432272k free, 77548k buffers
Swap: 25165820k total, 0k used, 25165820k free, 1832908k cached


PROCESS                 UPTIME          CPU%            VIRT_MEM        RES_MEM

phParser                01:58           0               1837m           222m
phQueryMaster           00:48           0               910m            71m
phRuleMaster            01:55           0               591m            53m
phRuleWorker            01:55           0               1338m           321m
phQueryWorker           01:55           0               1377m           320m
phDataManager           01:55           0               1133m           67m
phDiscover              01:58           0               423m            44m
phReportWorker          01:55           0               1429m           96m
phReportMaster          01:55           0               496m            50m
phIpIdentityWorker      01:55           0               938m            50m
phIpIdentityMaster      01:55           0               398m            31m
phAgentManager          01:58           0               1504m           45m
phCheckpoint            01:55           0               117m            22m
phPerfMonitor           01:58           0               756m            55m
phReportLoader          01:55           0               736m            320m
phBeaconEventPackager   01:58           0               1046m           57m
phDataPurger            01:55           0               516m            50m
phEventForwarder        01:58           0               476m            38m
phMonitor               47:53           0               1228m           582m
Apache                  49:04           0               224m            6088
Node.js-charting        48:57           0               922m            73m
Node.js-pm2             47:56           0               0               114m
AppSvr                  51:30           1               11170m          2907m
DBSvr                   52:03           0               376m            28m
Redis                   51:35           0               130m            7608
```

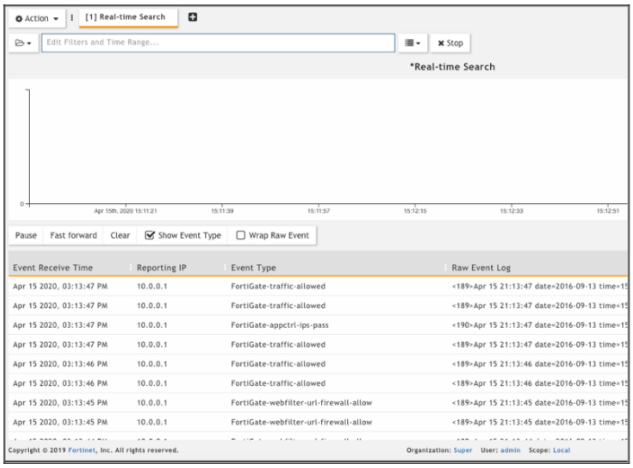
```
Every 1.0s: /opt/phoenix/bin/phstatus.py

System uptime:  18:11:24 up 52 min,  2 users,  load average: 0.30, 0.49, 0.33
Tasks: 25 total, 0 running, 9 sleeping, 15 stopped, 0 zombie
Cpu(s): 8 cores, 1.0%us, 0.5%sy, 0.0%ni, 98.6%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 8060332k total, 6192344k used, 1867988k free, 84284k buffers
Swap: 25165820k total, 0k used, 25165820k free, 1905052k cached


PROCESS                 UPTIME          CPU%            VIRT_MEM        RES_MEM

phParser                DOWN
phQueryMaster           00:49           0               910m            68m
phRuleMaster            DOWN
phRuleWorker            DOWN
phQueryWorker           00:49           0               1366m           317m
phDataManager           DOWN
phDiscover              DOWN
phReportWorker          DOWN
phReportMaster          DOWN
phIpIdentityWorker      DOWN
phIpIdentityMaster      DOWN
phAgentManager          DOWN
phCheckpoint            DOWN
phPerfMonitor           DOWN
phReportLoader          DOWN
phBeaconEventPackager   DOWN
phDataPurger            00:49           0               516m            50m
phEventForwarder        DOWN
phMonitor               00:30           0               999m            27m
Apache                  48:53           0               224m            6088
Node.js-charting        48:47           0               923m            74m
Node.js-pm2             48:14           0               0               102m
AppSvr                  04:15           0               11046m          2644m
DBSvr                   05:40           0               374m            28m
Redis                   51:26           0               130m            7612
```

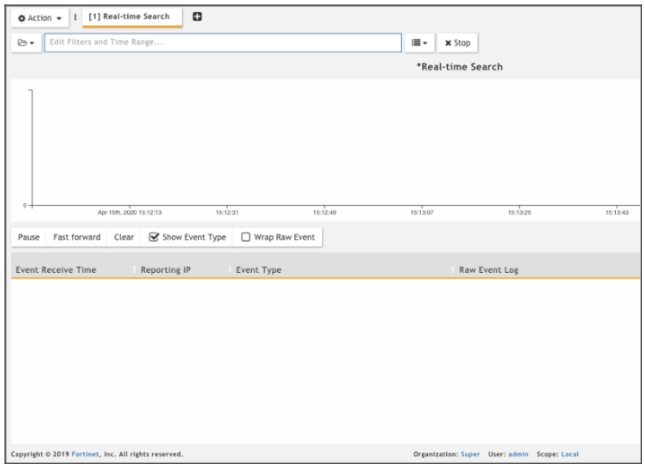
Understanding FortiSIEM Operations in DR Mode

When operating in DR Replication mode, there are a few things to bear in mind:

- Both the Primary and Secondary nodes GUI are available for login.
- The CMDB is set in a multi-master mode, so any changes on the Secondary are replicated over to the Primary.
- Although the CMDB can be edited from either site, it is recommended to do all edits on the Primary site.
- Analytical queries and reports can be run from either node.
- Performing Real-Time queries: You will see results only on the Primary node, as this is done in memory before storage.

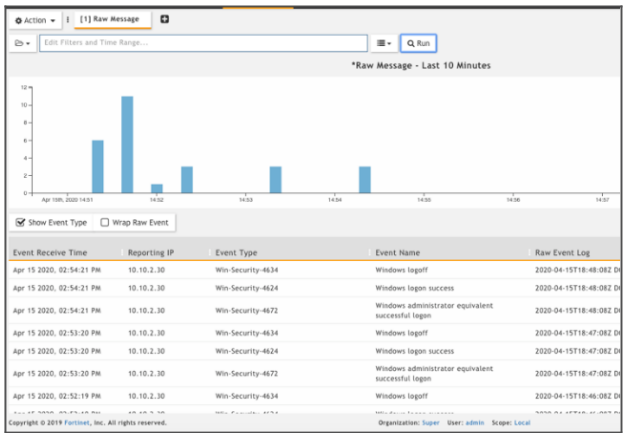
  Primary vs Secondary – Real-Time Search

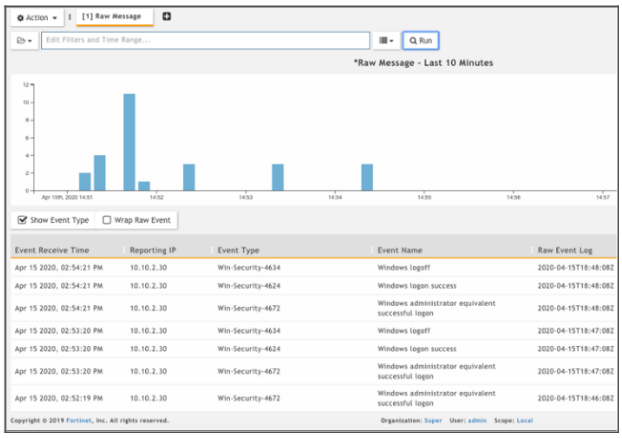




- Performing Historical Queries: Bear in mind the data on the Secondary node will be slightly out of date, dependent upon how much data is being replicated, but this is ideal for running large complex queries on the Secondary without impacting the Primary's performance.

  Primary vs Secondary – Historical Search (Last 10 Minutes)





- Any notifications or scheduled report deliveries are performed on the Primary node only. (Since most of the required `ph*` processes are down on the Secondary).

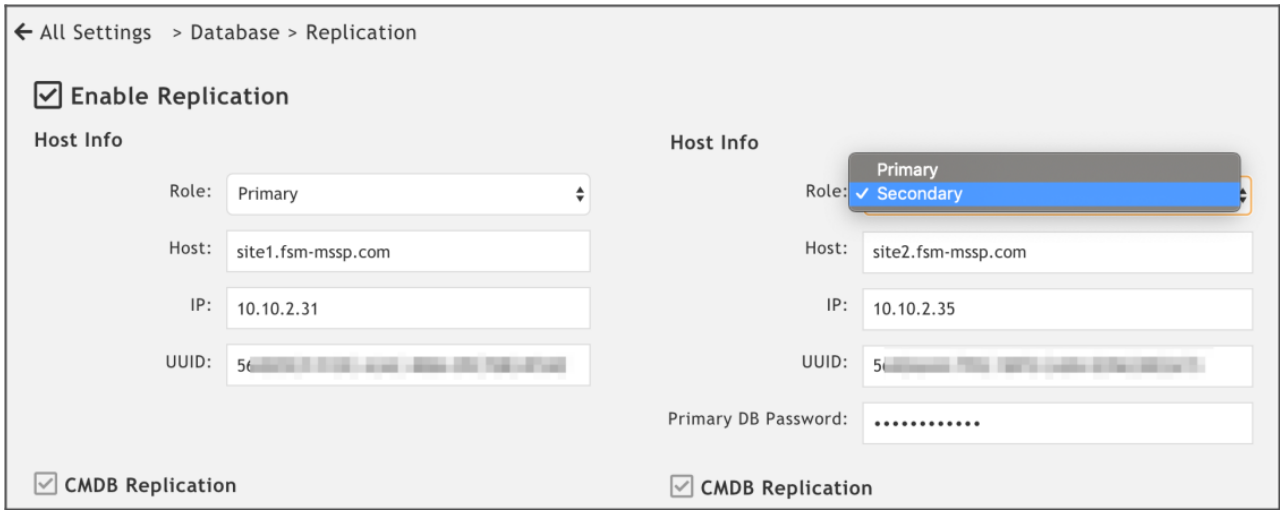
DR Change When the Primary site is Unavailable

It is important to note that it is a manual process to promote the Secondary node to be the Primary.

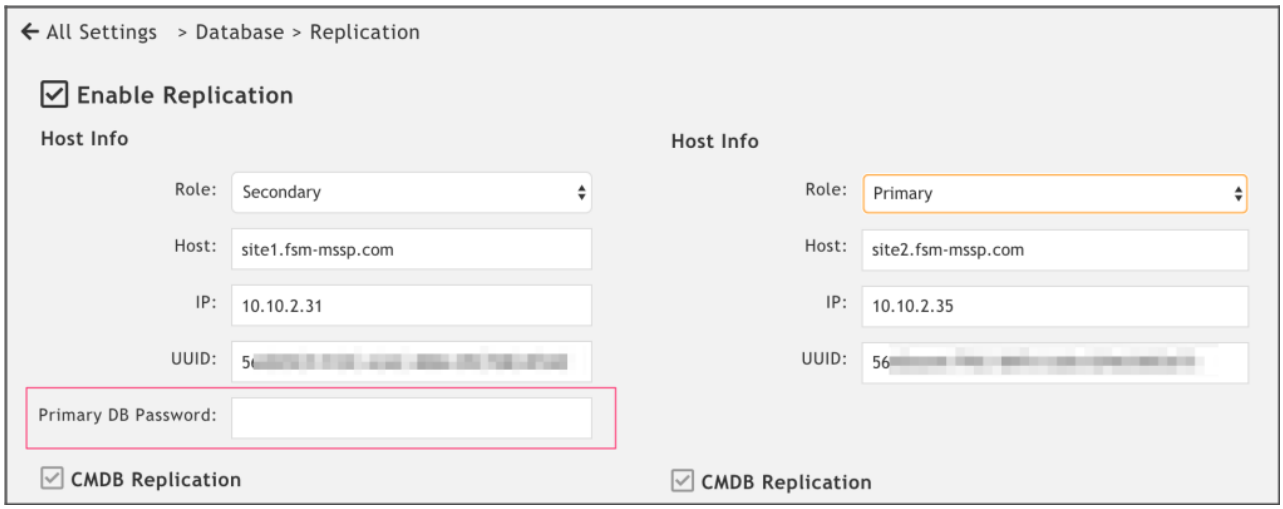
As soon as the Primary node is unavailable (that is, down/unavailable), any collector nodes will start to buffer their uploads, as the Worker Upload addresses they deliver to will be unavailable.

On the Secondary FortiSIEM node, log into the GUI:

1. Navigate to **Admin > Settings > Database > Replicate** (or **Replication** in 5.3+).
2. Change the **Role selector** for the Secondary node to be **Primary**.



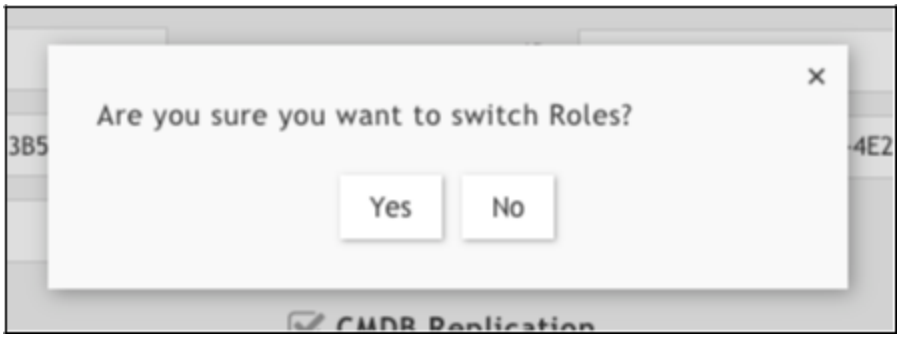
3. Notice how the original Primary Role has now switched to Secondary, and the **PrimaryDB** Password field moves across to the left.



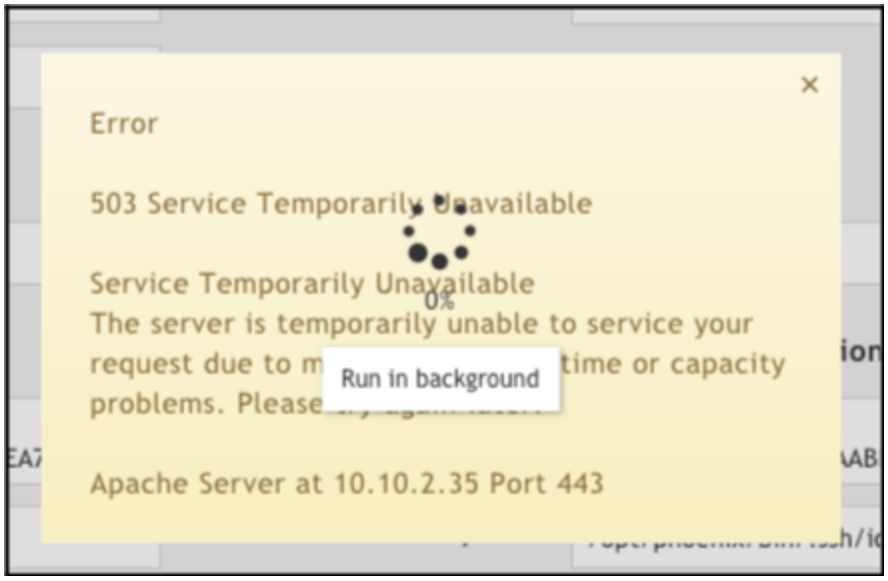
This field must be input again, but it can be obtained from an SSH session to the Secondary now, as it now has the same database as the Primary. Run the following command and paste the output into the **Primary DB Password** field.

```
#On the SECONDARY node
/opt/phoenix/bin/phLicenseTool -showDatabasePassword
```

4. Click **Apply**.
5. Click **Yes** to the warning, `Are you sure you want to switch Roles?`.



At this time, the following will appear in the GUI and it will seem to disconnect and the DR scripts will be run in the background.



After a short period of time, all the backend processes will start and the GUI will return to the login page.

If you run a Real-Time search you will probably find no data is still being received. This is because a DNS change is now required for the shared DNS addresses for the Supervisor node and the Worker upload settings, as in this example case:

DNS Address	Old Value	New Value
site.fsm-mssp.com	CNAME -> site1.fsm-mssp.com	CNAME -> site2.fsm-mssp.com
worker1.fsm-mssp.com	198.51.100.20	203.0.113.20
worker2.fsm-mssp.com	198.51.100.21	203.0.113.21

Change the DNS addresses and data will start to flow in normally.

Note: When the original Primary is recovered and powered back on, it will detect this and take on the Secondary role automatically.

Change-Over Where Both Systems are Operational

Operationally, there may be a need to perform a DR change over while both nodes are actually up and running.

Again, to note, this is a manual process of promoting the Secondary node to be the Primary.

On the Primary FortiSIEM node, log into the GUI:

1. Navigate to **Admin > Settings > Database > Replicate** (or **Replication** in 5.3+).
2. Change the **Role selector** for the Primary node to be **Secondary**.
3. Populate the **Primary DB Password** field.
   Run the following command on either the Primary or Secondary node via SSH:
   ```
   #On the PRIMARY or SECONDARY node
   /opt/phoenix/bin/phLicenseTool --showDatabasePassword
   ```
4. Click **Apply**, and respond **Yes** to the warning, "`Are you sure you want to switch Roles?`".
   Note: The extra steps below are very important. You will have a cluster which thinks it has two Primary nodes if you do not follow the two steps below.
5. Switch to the Secondary node GUI, and navigate to **Admin > Settings > Database > Replicate** (or **Replication** in 5.3+).
6. Change the Roles (unless the CMDB sync has already updated).
7. Click **Apply**.

Remember to change the DNS addresses after the migration.

Turning Off the Disaster Recovery Feature

There are cases where the DR Replication feature needs to be disabled, such as performing upgrades.

On the Primary FortiSIEM node, log into the GUI:

1. Navigate to **Admin > Settings > Database > Replicate** (or **Replication** in 5.3+).
2. Deselect the **Enable Replication** check box.
3. Respond **Yes** to the warning regarding disabling the Replication.

4. Click **Apply**.
5. Wait for the response `Replicate settings applied`.

Since the database is shared, this only needs to be performed on one node.

But, due to a bug in 5.2.8, it can only be re-enabled from the opposite node, Secondary in this case.