
Stratusphere™: How to backup & restore configuration data on Stratusphere database appliances

Overview

These instructions address the Disaster Recovery & Failover scenario where the main goal is to seamlessly migrate the Stratusphere agents to upload their data to a DR site when the primary site goes down or is no longer available. To achieve this goal, the amount of data to be backed up must be small and needs to be performed quickly. The backup script only backs up the lookup schema & configuration data (everything within Administration section) and leaves out the historical metrics & data (detail and summarized rollup data). The backup script can be scheduled to run every 24 hours using crontab scheduler on the production database and upload the data backup to the DR site database appliance. The restore script will run every 60 minutes using crontab on the DR database appliance and restore the lookup configuration schema and data.

Preparations

1. If default credentials have been changed, then procure updated credentials of `friend` and `root` user accounts on the Stratusphere Database appliances in production and the disaster recovery sites.
2. Download and install an SSH client like PuTTY or Windows Command Prompt, and an SCP client like WinSCP or Windows Command Prompt or similar.
3. Ensure firewall access using TCP/22 is available from the Production to the DR database appliances.
4. Prepare to request, receive, and apply a new license code for the new activation code on the DR site.

Instructions to configure backup and restore scripts on appliances.

First, we must configure the backup and restore scripts on the database appliances at the source and destination DR locations. Once we ensure the backup and restore are working, we will walk through the steps of how to configure the DR site with a Hub and Collectors for testing Disaster Recovery and then putting it into practice.

Scenarios

The instructions in this document support the following appliance combinations on the source and DR/destination side:

1. Source: Hub only appliance with or without Collectors.
 - a. Destination can be a Hub only appliance with no Collectors attached.
 - b. Destination can be a Database appliance with no Hub or Collectors attached.
2. Source: Database appliance with a Hub appliance and with or without Collectors.
 - a. Destination can be a Hub only appliance with no Collectors attached.
 - b. Destination can be a Database appliance with no Hub or Collectors attached.

On the DR / destination database appliance:

1. Open your SCP client and connect & log into the database appliance using the `friend` username and its default password `'sspassword'`. Upload the restore script from your local folder to the database appliance's `/home/friend` folder.
2. Open your SSH client and connect to the database appliance.
3. Log into the database appliance using the `friend` username and its default password `'sspassword'`. Then switch to the `root` user by executing the command: `'su -'`. Enter the default password `'sspassword'` when prompted. If these default credentials have been changed, please use your updated credentials.
4. Execute the following commands to move to the `/home/friend` folder first, and then execute the restore script that creates a backup folder, and then waits for the backup files to be uploaded:

- `cd /home/friend`
- `bash /home/friend/restoreconfigdatabase.vxxx.sh --setup`

```
[friend@10 ~]$ su -
Password:
[root@10 ~]# cd /home/friend
[root@10 friend]# bash /home/friend/restoreconfigdatabase.v662v2.sh --setup
08/11/23 11:53:25 AM EDT: Initiating setup for restoring data...
08/11/23 11:53:25 AM EDT: Completed initial setup.
[root@10 friend]#
```

5. Leave this SSH session open and begin a watch on the backups folder to see when the uploads are complete:

- `watch ls -alht /var/lib/pgsql/backups`

```
Every 2.0s: ls -alht /var/lib/pgsql/backups 10.30.50.130: Fri Aug 4 02:29:18 2023
total 0
drwxr-x---. 2 friend friend 46 Aug 4 02:29 .
-rw-r-----. 1 postgres postgres 0 Aug 4 02:20 restoreconfigdatabase.v662v2.out
drwxr-xr-x. 7 postgres postgres 236 Aug 1 21:08 ..
```

On the production / source database appliance:

6. Open your SCP client and connect & log into the database appliance using the `friend` username and its default password `'sspassword'`. Upload the backup script from your local folder to the database appliance's `/home/friend` folder.
7. Open your SSH client and connect to the database appliance.
8. Log into the database appliance using the `friend` username and its default password `'sspassword'`. Then switch to the `root` user by executing the command: `'su -'`. Enter the default password `'sspassword'` when prompted. If these default credentials have been changed, please use your updated credentials.
9. Execute the following commands to move to the `/home/friend` folder first, and then execute the backup script that creates a backup folder, saves SSH credentials to the remote DR database appliance to allow seamless, password-less transfers, and performs a backup of the configuration database, and then uploads it to the remote DR database appliance:

- `cd /home/friend`
- `bash /home/friend/backupconfigdatabase.vxxx.sh`

10. The script presents the default backup file name. Hit enter to accept and proceed.

```
-----
Liquidware Stratusphere Configuration Database Backup Script.
-----

Enter backup path and file name or hit [ENTER] to accept default below,
[/var/lib/pgsql/backups/tnt_config.db.backup.gz]:
```

11. Next, the script will present options on whether to upload the backup files or download them or simply do nothing. Select U to upload the file to a remote destination.

```
-----
Liquidware Stratusphere Configuration Database Backup Script.
-----

Enter backup path and file name or hit [ENTER] to accept default below,
[/var/lib/pgsql/backups/tnt_config.db.backup.gz]:
After the backup, what do you want to do with the backup file?
U. Upload file to a remote destination.
D. Download file.
N. Nothing. Just perform a backup and leave the file inplace.
What do you want to do? [U/D/N Default: D]: u
```

12. Next, the script will present a prompt to enter the IP/DNS address of the remote DR database appliance.

```
After the backup, what do you want to do with the backup file?
U. Upload file to a remote destination.
D. Download file.
N. Nothing. Just perform a backup and leave the file inplace.
What do you want to do? [U/D/N Default: D]: u
08/03/23 10:40:31 PM EDT: User chose to upload file after Backup is complete.
Enter IP/DNS address to upload backup file: 10.30.50.130
```

13. Next, the script will prompt the remote file path to upload the file. Hit enter to accept the default displayed.

```
08/03/23 10:46:58 PM EDT: User chose to upload file after Backup is complete.
Enter IP/DNS address to upload backup file: 10.30.50.130
Enter remote upload path [/var/lib/pgsql/backups]:
```

14. Next, it will prompt the default username to log into the remote DR database appliance, i.e., either friend or ec2-user or azureuser depending on the platform the DR Database appliance is running on. Hit enter to accept the default.

```
08/03/23 10:46:58 PM EDT: User chose to upload file after Backup is complete.
Enter IP/DNS address to upload backup file: 10.30.50.130
Enter remote upload path [/var/lib/pgsql/backups]:
Enter remote username [friend]:
```

15. Next, the script will prompt whether the remote database appliance accepts password or SSH Key Pair based authentication. Hit enter to accept the default password-based authentication. *Note: Additional instructions for key pair-based authentication are provided in another document.*

```
08/03/23 10:46:58 PM EDT: User chose to upload file after Backup is complete.
Enter IP/DNS address to upload backup file: 10.30.50.130
Enter remote upload path [/var/lib/pgsql/backups]:
Enter remote username [friend]:
Does 10.30.50.130 support Password or SSH Key pair based authentication?
Default: p [p/k] : p
```

16. Next, the script will ask if SSH credentials should be generated and stored for seamless, password-less file transfers. Enter Y.

```
08/03/23 10:46:58 PM EDT: User chose to upload file after Backup is complete.
Enter IP/DNS address to upload backup file: 10.30.50.130
Enter remote upload path [/var/lib/pgsql/backups]:
Enter remote username [friend]:
Does 10.30.50.130 support Password or SSH Key pair based authentication?
Default: p [p/k] : p
The script will prompt you to enter a password.
Do you want to generate SSH Keys for 10.30.50.130 [Y/N]: Y
```

17. Next, the script will display it is going to generate and install a new key for which you will need to enter the password for the user on the remote DR database appliance. If this is the first time, it will prompt to let you know the ECDSA key fingerprint and ask for confirmation to continue connecting to the remote database appliance. Enter yes to confirm.

```
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '10.30.50.131 (10.30.50.131)' can't be established.
ECDSA key fingerprint is SHA256:+NDM5PZzd48Ph38Pa0h41/hx+o1LI7f0qvM8kHQkSkY.
Are you sure you want to continue connecting (yes/no/[fingerprint])?
Enter 'yes' to confirm continue connecting.
Generating SSH Key for seamless transfer to 10.30.50.130...
Copying SSH Key to 10.30.50.130...
/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
friend@10.30.50.130's password:
```

Enter 'sspassword' as the password.

18. The script will add credentials to log into the remote DR database appliance.

```
Do you want to generate SSH Keys for 10.30.50.130 [Y/N]: y
Generating SSH Key for seamless transfer to 10.30.50.130...
Copying SSH Key to 10.30.50.130...
/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
friend@10.30.50.130's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'friend@10.30.50.130'"
and check to make sure that only the key(s) you wanted were added.
```

19. It will then display the basic settings, then proceed to perform the actual configuration database backups, and upload them to the remote DR database appliance.

```
Backup schema only to : /var/lib/pgsql/backups/portal.schema.backup.gz
Backup config data to : /var/lib/pgsql/backups/tnt_config.db.backup.gz
Upload backup files to : friend@10.30.50.130:/var/lib/pgsql/backups

08/03/23 11:02:58 PM EDT: Backup database schema...
08/03/23 11:03:23 PM EDT: Backup configuration data...
08/03/23 11:03:39 PM EDT: Uploading backups to 10.30.50.130...
tnt_config.db.backup.gz          100% 80MB 169.3MB/s 00:00
portal.schema.backup.gz         100% 1171KB 114.3MB/s 00:00
restore                          100% 0 0.0KB/s 00:00
```

20. The uploaded files should appear on the earlier SSH session's watch window.

21. Next, the script will prompt to delete backup files created. Enter Y for both.

```
Delete backup /var/lib/pgsql/backups/tnt_config.db.backup.gz file? [Y/N]: y
08/03/23 11:13:11 PM EDT: Removing /var/lib/pgsql/backups/tnt_config.db.backup.gz file...
Delete backup /var/lib/pgsql/backups/portal.schema.backup.gz file? [Y/N]: y
08/03/23 11:13:14 PM EDT: Removing /var/lib/pgsql/backups/portal.schema.backup.gz file...
08/03/23 11:13:14 PM EDT: Removing /var/lib/pgsql/backups/restore file...
08/03/23 11:13:14 PM EDT: Removing /var/lib/pgsql/backups/backupconfigdatabase.v662v2.out file...
08/03/23 11:13:14 PM EDT: Done.
-----
[friend@10 ~]$
```

22. By default, a log file is created under `/home/friend/backupconfigdatabase.vxxx.log` – monitor it for status or any error messages. There is functionality to rotate the log if it gets above 5MB and to keep last 5 log files so that the logs do not take up too much space on the appliance.
23. Once adequate testing has been completed, we can add an entry into crontab for running the backup script once every day. As `root` enter the following command for scheduling a backup every day at 9a UTC / 5a EDT / 4a EST and run the backup script automatically.
- On the command line enter:


```
➤ crontab -e
```
 - Enter the `i` key to go into insert mode. Enter the following information:


```
0 9 * * * /bin/bash
/home/friend/backupconfigdatabase.vxxx.sh -b
/var/lib/pgsql/backups/tnt_config.db.backup.gz -r
<IPorDNS.of.dr.db> -p /var/lib/pgsql/backups -u friend
```
 - Hit the `ESC` key and then enter `“:wq”` to write and quit from the editor.
 - The scheduler now will run the backup script every day at 9a UTC or 4a ET and upload the backup files to the remote DR destination location. Depending on your time zone, please an appropriate UTC based time when your database appliance may be at its least amount of load for the day.

On the DR / destination database appliance:

24. After a successful backup, the following files should be uploaded and visible in the `watch` window:

```
Every 2.0s: ls -alht /var/lib/pgsql/backups 10.30.50.130: Fri Aug 4 03:18:52 2023
total 81M
-rw-r--r--. 1 postgres postgres 1.7K Aug 4 03:17 restoreconfigdatabase.v662v2.out
drwxr-x---. 2 postgres postgres 148 Aug 4 03:15 .
-rw-r--r--. 1 root root 0 Aug 4 03:15 restoreinprogress
-rw-r-----. 1 friend friend 0 Aug 4 03:03 restore
-rw-r-----. 1 postgres postgres 1.2M Aug 4 03:03 portal.schema.backup.gz
-rw-r-----. 1 postgres postgres 80M Aug 4 03:03 tnt_config.db.backup.gz
drwxr-xr-x. 7 postgres postgres 236 Aug 1 21:08 ..
```

- `tnt_config.db.backup.gz`: This file contains all the configuration and lookup information within the Stratusphere database.
 - `portal.schema.backup.gz`: This file contains the full schema of the database – no data, just the schema.
 - `restore`: This file is a control file that gets uploaded to the DR database appliance when a new backup is uploaded. When the restore begins, this file is deleted to ensure that another scheduled restore event does not start if the current one is still running. If the `restore` file is not found, then the restore script does not proceed and exits immediately.
 - `restoreinprogress`: The `restoreinprogress` file is created when the restore script begins restoring the database. This file is deleted on completion of a restore. It is created to prevent another restore starting when the previous restore is still in progress.
25. CTRL+C out of the watch command.

26. Execute the command line to restore the backup:

```
➤ bash /home/friend/restoreconfigdatabase.vxxx.sh -r
   /var/lib/pgsql/backups/tnt_config.db.backup.gz
```

```
-----
Liquidware Stratusphere Configuration Database Restore Script.
-----

Restore schema from:: /var/lib/pgsql/backups/portal.schema.backup.gz
Restore config from:: /var/lib/pgsql/backups/tnt_config.db.backup.gz

08/08/23 04:09:12 PM EDT: Asking lwl-backend-priv service to stop...
08/08/23 04:09:13 PM EDT: Asking lwl-backend service to stop...
08/08/23 04:09:13 PM EDT: Asking lwl-smx service to stop...
08/08/23 04:09:13 PM EDT: Asking lwl-cidd service to stop...
08/08/23 04:09:13 PM EDT: Restarting Postgres 12 service...
08/08/23 04:09:13 PM EDT: Waiting 5s for database to restart...
08/08/23 04:09:19 PM EDT: Backup permissions for /var/lib/pgsql/backups/ folder...
08/08/23 04:09:19 PM EDT: Backup permissions for /var/lib/pgsql/backups/tnt_config.db.backup.gz folder...
08/08/23 04:09:19 PM EDT: Setting owners of /var/lib/pgsql/backups/ to postgres:postgres...
08/08/23 04:09:19 PM EDT: Setting owners of /var/lib/pgsql/backups/tnt_config.db.backup.gz to postgres:postgres...
08/08/23 04:09:19 PM EDT: Setting owners of /var/lib/pgsql/backups/portal.schema.backup.gz to postgres:postgres...
08/08/23 04:09:19 PM EDT: Backing up existing license...
08/08/23 04:09:19 PM EDT: Warn: License is empty.
08/08/23 04:09:19 PM EDT: Restarting Postgres 12 service...
08/08/23 04:09:20 PM EDT: Waiting 5s for database to restart...
08/08/23 04:09:25 PM EDT: Dropping portal database now...
08/08/23 04:09:25 PM EDT: Restoring database schema now...
08/08/23 04:11:51 PM EDT: Restoring configuration data now...
08/08/23 04:13:31 PM EDT: Restarting Postgres 12 service...
08/08/23 04:13:32 PM EDT: Waiting 5s for database to restart...
08/08/23 04:13:37 PM EDT: Restore permissions on /var/lib/pgsql/backups/ folder...
08/08/23 04:13:37 PM EDT: Restore permissions on /var/lib/pgsql/backups/tnt_config.db.backup.gz file...
08/08/23 04:13:37 PM EDT: Restore permissions on /var/lib/pgsql/backups/portal.schema.backup.gz file...
08/08/23 04:13:37 PM EDT: Asking lwl-backend-priv service to start...
08/08/23 04:13:47 PM EDT: Asking lwl-backend service to start...
08/08/23 04:13:57 PM EDT: Asking lwl-smx service to start...
08/08/23 04:14:08 PM EDT: Asking lwl-cidd service to start...
08/08/23 04:14:08 PM EDT: Done.
-----

08/08/23 04:14:08 PM EDT: Rebooting appliance in 60 seconds...
```

27. By default, a log file is created under

`/home/friend/restoreconfigdatabase.vxxx.log` – monitor it for status or any error messages. There is functionality to rotate the log if it gets above 5MB and to keep last 5 log files so that the logs do not take up too much space on the appliance.

28. Once adequate testing has been completed, we can add an entry into crontab for running the restore script at the bottom of every hour. As `root`, enter the following command to schedule a restore at the bottom of every hour and run the restore script automatically.

a. On the command line enter:

```
➤ crontab -e
```

b. Enter the `i` key to go into insert mode. Enter the following information:

```
30 * * * * /bin/bash
/home/friend/restoreconfigdatabase.vxxx.sh -r
/var/lib/pgsql/backups/tnt_config.db.backup.gz
```

c. Hit the `ESC` key and then enter `:"wq"` to write and quit from the editor.

d. The scheduler now will run the restore script at the bottom of every hour and restore the database automatically.

29. The entire backup and restore scripting process is now completely configured. Monitor the log files mentioned on both source and destination appliances to ensure the backup and restore are performing successfully.

High Level Plan

In case of a disaster, these scripts **would have already performed a backup and uploaded it to the DR site database appliance** every 24 hours. On the DR site, **the database has already been restored based on the last successful upload** from the production appliance. The high-level plan **after** this is to:

1. Attach a Hub to the DR database.
2. Since the restored configuration database is from the production appliance, the license check may fail on the DR Hub on the first restore. Once a new license is applied with its own new Activation Code, the restore script will keep restoring this new license on every subsequent restoration of the production database.
3. Remove the old Collectors using the Web UI on the new DR Hub and add new Collectors to the installation using the same DNS resolvable host names but new local IP addresses.
4. Update the DNS Servers to point existing DNS names from the production Hub & Collectors to the DR Hub and Collectors IP addresses thus facilitating the CID Keys to connect to the DR site's Hub and Collectors.
5. The Hub & Collectors will recognize each registered CID Key uploading data and be able to store their data received within the database.

Complete installation, Verification and DR Testing

1. Stratusphere Hub:
 - a. Attach a Hub to the DR database and verify if the production/source and DR/destination database have the same information under the ADMINISTRATION section of the Stratusphere Web UI.
 - b. The INVENTORY, VM DIRECTORY, USER DIRECTORY, etc. should all be the same between the two installations.
 - c. Under the HUB ADMINISTRATION > CONFIGURATION page, validate whether the HOST NAME, IP ADDRESS, NETWORK MASK, DEFAULT GATEWAY, & DNS SERVERS configuration, CID KEY CALLBACK DNS NAME are accurate and resolve to the right DR network-based IP addresses if using the DNS names.
 - d. If adding Collectors, please note down the HOST NAME & CID KEY CALLBACK DNS NAME field under HUB ADMINISTRATION > CONFIGURATION page which will be used within the Collectors section below.
 - e. Apply the new license to the DR Stratusphere Hub installation since the appliances are different from the original production source and would thus fail the license check. Apply the new license based on the new Activation code and license code received from Liquidware.
 - f. If using **Enhanced Security**, please enable Enhanced Security **BEFORE** adding any Collectors.
2. Stratusphere Collectors:
 - a. If any Collectors were installed within the production source installation, please note down all the relevant DNS host names of existing Collectors first. Then delete them from the Web UI under the Administration product within the COLLECTOR ADMINISTRATION > COLLECTORS tab.
 - b. Before adding brand new Collectors, please ensure that the DR Hub's Host Name and CID Key Callback DNS Names resolve to the DR Hub's local IP address. If this is a test of DR, please add the Hub's Host Name and CID Key

-
- Callback DNS Name to each Collector's `/etc/hosts` file so that they resolve properly onboard the Collectors. If this step is skipped and the DNS servers are not pointing to the DR Hub's addresses, the Collectors will NOT register.
- c. Once they resolve to the Hub's DR IP addresses, add new Collectors to the DR Hub using the same DNS host name, if possible, but using new local IP addresses within the DR network subnet.
 - d. Please ensure that individual Collectors are part of the Collector Groups as designed at the source production Hub installation.
 - e. The addition of Collectors will now rebrand the `mgrcert.pem` with list of the new Collectors.
 - f. Once these Collectors are added, please update your DNS servers to point all existing DNS Names of the source production Hub & Collectors to the DR IP addresses.
3. Stratusphere CID Key:
 - a. The CID Keys will continue collecting data onboard the machines they are deployed on all through this time. They will try uploading to their list of Collectors in product which may NOT be available anymore due to the disaster. The CID Key will store all the data collected in a Last-In-First-Out (LIFO) queue.
 - b. The CID Keys will try uploading data every callback, and after 4 consecutive failures to upload data to its list of Collectors, they will fallback to the Hub just to check if something has changed and receive a new `cert.txt` with a new list of Collectors. So, if there are 2 Collectors in the source production environment, and your callback interval is 60minutes, the CID Keys will try uploading data for 4 callbacks i.e., 240 minutes or 4 hours before falling back to the Hub to receive any new `cert.txt` settings.
 - c. Depending on when the DNS servers are switched to point to the DR IP addresses, the CID Keys will callback to either their DR Collectors or the DR Hub. They will be recognized by the DR Stratusphere installation, and a new `cert.txt` will be provided based on the new list of DR Hub's CID Key Callback DNS Name and list of DR Collectors.
 4. Validate DR Migration:
 - a. Log into the DR Hub's Web UI under the ADMINISTRATION section's HUB ADMINISTRATION > OVERVIEW tab to see how machines start calling back to the DR Hub and Collectors.
 - b. Monitor the EVENT LOG tab for any errors and to see machines calling back and uploading data.
 - c. Switch to STRATUSPHERE UX and check for CID Key Level callbacks within the DIAGNOSTICS tab's MACHINE INSPECTION REPORTS view or under ADVANCED > INSPECTORS > MACHINES tab with the CID LEVEL resolution to see the latest data being inserted into the database.
 5. If CID Key data is being uploaded and inserted into the Stratusphere Database, the DR process for Stratusphere is now complete.

Troubleshooting & Error Checking

Ensure scheduler is running on Production & DR database appliances.

The backup and restore scripts require the built-in scheduler, cron to be running reliably all the time. So, here are some areas that need to be monitored:

1. Is the cron service is running?

```
/etc/init.d/crond status
crond (pid 4291) is running..
```

2. Are there any cron errors?

```
/var/log/cron
```

Error Checking

As part of daily operations, there can be errors on execution of the backup and restore scripts. Liquidware recommends monitoring the log files for errors so that action can be taken sooner rather than later. Here is a list of error statements that can be output by the scripts while executing the backup and restore. These statements should be monitored as part of determining if a backup or restore failed.

Backup Script

On the source production appliance that hosts the Stratusphere Database, the backup script creates a log output every time it executes a backup. It is available under `/home/friend/backupconfigdatabase.vxxx.log` or `/home/[username]/backupconfigdatabase.vxxx.log` where [username] could be `ec2-user` or `azureuser` or similar depending on the platform the appliance is running on. Here is a list of errors that the backup script can generate:

```
Error: Failed to remove file $file.
Error: Failed to restore permissions on $backupPath folder.
Error: Failed to restore ownership to postgres for $backupPath folder.
Error: Failed to set ownership of $logFilePath to '$username' user.
Error: Failed to grant read access of $logFilePath to '$username' user.
Error: Failed to set $postgresOwners ownership on $backupPath.
Error: Failed to restore $userOwners ownership on $backupPath.
Error: Control file 'restoreinprogress' found on $remoteIpDNS.
Error: Control file 'restoreinprogress' found on $remoteIpDNS.
Error: Upload of control file restore to $remoteIpDNS failed.
Error: Failed to create $restore file.
Error: Upload of configuration backups to $remoteIpDNS failed.
Error: Generation of ECDSA SSH Keys failed.
Error: Generation of SSH Keys failed.
Error: Copying SSH Keys failed.
Error: Failed to grant read access to '$username' user to $backupPath folder.
Error: Failed to change ownership of $configBackupFile to '$username' user.
Error: Failed to change ownership of $schemaOnlyBackupFile to '$username' user.
Error: Failed to delete $filePath.
Error: Failed to delete $schemaOnlyPathFile.
Error: $filePath is invalid. Please enter a valid file under $backupPath.
Error: $backupPath could not be created. Exiting now...
```

```

Error: Could not set a+rx permissions on $baseBackupPath.
Error: Could not $defaultBackupFolderOwners ownership on $backupPath.
Error: Version compatibility check failed. This script works with Stratusphere 6.6.x only.
Error: Contact Liquidware for a script that works with $appVersion.
Error: This script is meant to run only on appliances that host the Stratusphere database.
Error: This script is meant to run only on appliances that host the Stratusphere database.
Error: $configBackupPathFile not in $backupPath!
Error: $configBackupFile does not have expected $dbExt extension.
Error: $schemaOnlyPathFile not in $backupPath!
Error: $schemaOnlyBackupFile does not have expected $schemaExt extension.
Error: You must be logged in as root to run this script.

```

Note: The **highlighted** parts will be substituted with the actual data such as, remote IP address of the DR database appliance, or the actual backup path, or appliance version, etc.

Restore Script

On the target DR appliance that hosts the DR Stratusphere Database, the restore script creates a log output every time it tries to run a restore. It is available under

/home/friend/restoreconfigdatabase.vxxx.log or

/home/[username]/restoreconfigdatabase.vxxx.log where [username] could be ec2-user or azureuser or similar depending on the platform the appliance is running on.

Here is a list of errors that the restore script can generate:

```

Error: Failed to schedule reboot of the appliance in 60 seconds.
Error: Failed to backup existing license.
Error: Failed to restore existing license.
Error: Failed to remove file $file.
Error: Failed to drop portal database.
Error: $connections Active database connections found! Stop Hub/Collectors prior to restore.
Error: Failed to stop/start database service.
Error: Failed to set $owners as owner:group of $filePath.
Error: Failed to set $owners as owner:group of $filePath.
Error: $filePath not found.
Error: $backupPath not found.
Error: Failed to restore permissions $defaultBackupFolderPerms on $backupPath folder.
Error: Failed to restore ownership $defaultBackupFolderOwners for $backupPath folder.
Error: Failed to restore permissions $defaultBackupFilePerms on $configBackupPathFile file.
Error: Failed to restore ownership $defaultBackupFileOwners for $configBackupPathFile file.
Error: Failed to restore permissions $defaultBackupFilePerms on $schemaOnlyPathFile file.
Error: Failed to restore ownership $defaultBackupFileOwners for $schemaOnlyPathFile file.
Error: $schemaOnlyPathFile not found!"
Error: $filePath not found!"
Error: $filePath not found!"
Error: Failed to set ownership of $logFilePath to '$username' user.
Error: Failed to grant read access of $logFilePath to '$username' user.
Error: Failed to $action tnt-backend-priv service.
Error: Failed to $action tnt-backend service.
Error: Failed to $action servicemix service.
Error: Failed to $action lwl-inventoryd service.
Error: Failed to $action lwl-backend-priv service.
Error: Failed to $action lwl-backend service.
Error: Failed to $action lwl-smx service.
Error: Failed to $action lwl-cidd service.
Error: $configBackupPathFile not in $backupPath!"
Error: $configBackupPathFile not found!"
Error: $schemaOnlyPathFile not in $backupPath!"
Error: $schemaOnlyPathFile not found!"

```

Error: A restore is still in progress.
Error: `$backupPath` could not be created. Exiting now...
Error: Could not set a+rx permissions on `$baseBackupPath`.
Error: Could not `$defaultBackupFolderOwners` ownership on `$backupPath`.
Error: Version compatibility check failed. This script works with Stratusphere 6.6.x only.
Error: Contact Liquidware for a script that works with `$appVersion`.
Error: This script is meant to run only on appliances that host the Stratusphere database.
Error: This script is meant to run only on appliances that host the Stratusphere database.
Error: A prior restore appears to be in progress right now. Exiting.
Error: If error persists, remove `restoreinprogress` file manually and try again.
Error: You must be logged in as root to run this script.

Here is a list of Information only entries that the restore script can generate:

Info: Exiting as restore file not found.

The restore script is scheduled to run every hour whereas the backup script is scheduled to run once a day. So, the restore script will NOT have something to restore every time it runs.

Note: The highlighted parts will be substituted with the actual remote IP address of the DR database appliance, or the actual restore file path or folder name or appliance version.

©2023 Liquidware Labs Inc. All rights reserved. Stratusphere, ProfileUnity, FlexApp, FlexDisk, and ProfileDisk are trademarks of Liquidware Labs. All other products are trademarks of their respective owners. 23-0811