

Perforce Software, Inc.

510.864.7400

[consulting@perforce.com](mailto:consulting@perforce.com)

[www.perforce.com](http://www.perforce.com)

Disaster Recovery Failover Plan

Table of Contents

1 Overview 3

2 Fail Over Procedure 3

2.1 Check replica status 3

2.2 Stop access to master server – done on master server 3

2.3 Check replica status 3

2.4 Change the replica to become the master 3

2.5 Update the crontab (Run all commands as perforce user) 4

2.6 Change the original master to become the replica 4

# Overview

The DR server is being replicated using the Perforce built in server to server replication via the “p4 pull” command and a rsync script to replicate checkpoints. This document provides the steps to fail over from the main server to the DR server.

NOTE: This is a generic document that needs to be customized for the specific set up in your environment.

# Fail Over Procedure

## Check replica status

This is just a verification that replication is up to date before we start the whole process.

On the replica machine, run p4 pull -lj to check the status of the replication. The command should return two matching journal sequence numbers.

## Stop access to master server – done on master server

Change the /p4/1/bin/p4broker.cfg file to block all user access to the server. There is an entry in the config file to do this vi search for words “block all access”, just uncomment that entry in the config file, and save the file. The broker does not have to be restarted to pick up the change. It will take effect immediately.

If you are not running a broker, just validate the replica is up to date and then shut down the master, then the replica.

## Check replica status

On the replica machine, run p4 pull -lj to check the status of the replication. You want to wait for the replica to fully catch up before shutting down the master and replica instances. Once you have verified the replication is up to date:

stop the replica by running /p4/1/bin/p4d\_1\_init stop on the replica server

stop the master by running /p4/1/bin/p4d\_1\_init stop on the master server

## Change the replica to become the master

Change the DNS entry or move the VIP from the master to the new master if you are using a VIP address.

On the replica server, edit /p4/common/bin/p4\_vars and change the P4REPLICA= value to FALSE and save the file. Now you can restart the server as the master by running /p4/1/bin/p4d\_1\_init start .

*After this is done, the former replica is now the master.*

Next run:

rm /p4/1/root/state

rm /p4/1/root/rdb.lbr

## Update the crontab (Run all commands as perforce user)

On the new master server, originally the replica, run crontab /p4/crontab to reset the crontab to run the master server crontab settings.

On the original master server, now the replica, run crontab /p4/crontab.replica to change it to run the replica scripts.

## Change the original master to become the replica

On the original master, edit /p4/1/root/instance\_vars and change the P4REPLICA= value to TRUE and save the file.

Move /p4/1/logs/journal, e.g. to /p4/1/logs/journal.orig.master

Delete the following files if they exist: /p4/1/root/state or /p4/1/root/rdb.lbr . You also need to login into the master server as the service user on the replica so that replication can run.

Run:

p4 -p ssl:master:port -u service trust (Only if your site is using SSL)

p4 -p ssl:master:port -u service login < /p4/common/bin/servicepass

Now, start the server as a replica by running /p4/1/bin/p4d\_1\_init start. Edit /p4/1/bin/p4broker.cfg and comment out the lines that were blocking all user access (search for “block all access”) and save the file if you are using a broker.

Check replication by running:

p4 login < /p4/common/bin/adminpass

p4 pull -lj

on the master server. You should see two matching journal sequence numbers. The p4 pull –lj will only succeed when run on a replica pulling from a master. So this proves that the roles have been switched.

Finally, run these final two steps:

On the new replica server, run the sync\_replica.sh command that cron normally runs in order to make sure it is working properly.

/p4/common/bin/p4master\_run 1 /p4/common/bin/sync\_replica.sh

NOTE 1: *If the master server has been down for more than 7 days*, you will have to reset the replica as follows before starting it. The reason for the 7 days value is that the number of old checkpoints and journals to keep is set to 7 in the p4\_vars file. The replication uses the old journals in order to catch up from where it last stopped, so if the old ones have rotated off, you have to reset the replica with a new checkpoint and updated versioned files.

To reset the replica, from the replica run:

If your servers are set up to rsync without a password, you can reset the replica by running:

/p4/common/bin/p4master\_run 1 /p4/common/bin/weekly\_sync\_replica.sh

weekly\_sync\_replica.sh performs the following steps:

rsync -avz --delete perforce@master\_server:/p4/1/checkpoints/ /p4/1/checkpoints

rm -r -f /p4/1/root/db.\*

rm -r -f /p4/1/offline\_db/db.\*

rm -r -f /p4/1/logs/journal

ls -lah /p4/1/checkpoints

Make note of the highest numbered p4\_1.ckp.#.gz file in that directory. Assume it is 10 for this example:

/p4/1/bin/p4d\_1 -r /p4/1/root -jr -z /p4/1/checkpoints/p4\_1.ckp.10.gz

/p4/1/bin/p4\_1 -p master\_server:port -u service login

/p4/1/bin/p4d\_1\_init start

/p4/1/bin/p4d\_1 -r /p4/1/offline\_db -jr –z /p4/1/checkpoints/p4\_1.ckp.10.gz

Optionally, you may want to run:

rsync -avz --delete perforce@master\_server:/p4/1/depots/ /p4/1/depots

or

/p4/common/bin/p4master\_run 1 /p4/common/bin/p4verify.sh

to make sure the depot files are up to date.

NOTE 2: If the master server is inoperable, follow these steps:

1. On the replica server, run /p4/1/bin/p4d\_1\_init stop
2. Do the steps in Section 2.4 above