

DR 構築マニュアル

【前提条件】

下記設定を前提にマニュアル化しています

マニュアル内の**太字部分**は、実環境に合わせて変更してください

※マスターサーバとスタンバイサーバの**タイムゾーン**が一致している必要があります。

※p4d の**バージョン**が一致している必要があります。

| | マスター | スタンバイ |
|----------------|----------------|----------------|
| サーバ名 | commit | nomandatory |
| ServerID | commit | nomandatory |
| Option | - | nomandatory |
| server-service | commit-server | standby |
| IP | 192.168.56.102 | 192.168.56.104 |
| Port | 1666 | 1666 |
| P4ROOT | /commit | /p4root_st |

| | |
|---------------------|---------|
| Perforce super user | super |
| レプリケーション用ユーザ名 | service |

マニュアルの流れ

| | |
|--------------------------|---|
| 1. スタンバイサーバのサーバ仕様作成..... | 2 |
| 2. サービスユーザの作成と設定..... | 3 |
| 3. チェックポイントの作成と移動..... | 4 |
| 4. スタンバイサーバの設定..... | 5 |
| 5. レプリケーションの設定と確認..... | 7 |
| 6. フェイルオーバー：テスト運用..... | 8 |
| 7. その他..... | 9 |

1. スタンバイサーバのサーバ仕様作成

【マスターサーバで作業します】

1. p4 コマンドで“nomandatory”というサーバ仕様を作成します

実行コマンド

```
p4 -u super -p master-server:port server nomandatory
```

コマンド例)

```
p4 -u super -p commit:1666 server nomandatory
```

2. 設定を修正・追加します

赤文字部分が修正・追記箇所です

```
ServerID:      nomandatory
Type:          server
Services:      standby ● standard (default) を上書き
Options:      nomandatory
ReplicatingFrom: commit ● 一行追記する(マスターサーバの server ID)
Description:
Created by super.
DistributedConfig:
    any#P4JOURNAL=../journals/journal
    any#P4LOG=../logs/log
    :
    any#unicode=1
    P4LOG=log
    P4TARGET=192.168.56.102:1666
    P4TICKETS=/p4root_st/.p4tickets
    db.replication=readonly
    lbr.replication=readonly
    monitor=2
    rpl.journalcopy.location=1
    serviceUser=service
    startup.1=journalcopy -i 1
    startup.2=pull -L -i 1
    startup.3=pull -u -i 1
```

■ 解説

startup.1=journalcopy -i 1 …… 起動時に journalcopy (マスターサーバの journal をスタンバイに移送) を実行
startup.2=pull -L -i 1 …… 移送したローカルの journal から情報を読み取る
startup.3=pull -u -i 1 …… アーカイブファイルを取得する

※より詳細な情報は『[Helix Core P4 コマンドリファレンス \(2019.1\)](#)』をご参照ください

2. サービスユーザの作成と設定

【マスターサーバで作業します】

1. p4 コマンドで“service”というレプリケーション用サービスユーザを作成します

実行コマンド

```
p4 -u super -p master-server:port user -f service
```

コマンド例)

```
p4 -u super -p 192.168.56.102:1666 user -f service
```

2. 設定を修正・追加します

赤字部分が修正箇所です

User: service

Type: **service** ●一行追記する

Email: service@commit

FullName: service

3. サービスユーザ “service” のパスワードを設定します

実行コマンド

```
p4 -u super -p master-server:port passwd service
```

コマンド例)

```
p4 -u super -p 192.168.56.102:1666 passwd service
```

パスワード設定

Enter new password: (任意のパスワードを入力します)

Re-enter new password: (確認のため同じパスワードを入力します)

4. サービスユーザのアクセスレベルの設定を追加します

実行コマンド

```
p4 protect
```

表示例)

```
write user * * //...
```

```
list user * * -//spec/...
```

```
super user super * //...
```

```
super user service * //... ●一行追加
```

3. チェックポイントの作成と移動

【マスターサーバで作業します】

1. スタンバイサーバ用のチェックポイントファイルを作成します

作成したチェックポイントファイル名は“checkpoint”としています

実行コマンド

```
p4 -u super -p master-server:port admin checkpoint filename
```

コマンド例)

```
p4 -u super -p 192.168.56.102:1666 admin checkpoint checkpoint
```

チェックポイントファイル(ファイル名 : checkpoint.ckp.1)が作成されます

2. チェックポイントファイルを転送します

FTP ツール(任意)を使用して、スタンバイサーバへチェックポイントファイル(checkpoint.ckp.1)を転送します。

※このマニュアルではスタンバイサーバの /tmp に格納しています

4. スタンバイサーバの設定

【スタンバイサーバでの作業です】

1～4の作業については、サーバ構築方法によって対応が異なります。実環境をご確認の上、必要に応じて作業してください。

1. P4ROOT 用のディレクトリ “p4root_st” を作成します ※既にルートフォルダを作成済みであれば、作業不要です

実行コマンド

```
mkdir /p4root_st
```

2. 作成したディレクトリの owner を“root”から“perforce”に変更します ※owner が perforce であれば作業不要です

実行コマンド

```
chown perforce:perforce -R /p4root_st/
```

3. FTP 転送したチェックポイントファイルを\$P4ROOT へ移動します ※\$P4ROOT へ転送済みであれば作業不要です

実行コマンド

```
mv /転送場所/filename /$P4ROOT/
```

コマンド例

```
mv /tmp/checkpoint.ckp.1 /P4root_st/
```

4. チェックポイントファイルの owner を“root”から“perforce”に変更します
※owner が perforce であれば、作業不要です

実行コマンド

```
chown perforce:perforce filename
```

5. Perforce サービスを停止します

実行コマンド

```
p4dctl stop server-name
```

コマンド例)

```
p4dctl stop nomandatory
```

6. 不要なファイルを削除します
構築したスタンバイサーバに存在するデータベース等を削除します

\$P4ROOT に移動し、以下のファイルを削除します

- ・ファイル名が[db.]で始まるファイル(データベース)すべて
- ・server.locks フォルダ：フォルダごと削除

実行コマンド例

```
rm -f db.*
```

```
rm -fr server.locks/
```

7. ServerID を指定します

サーバ“standby”の serverID を“nomandatory”に指定します

※既に指定されている場合は作業不要です

実行コマンド

```
p4d -r /$P4ROOT -xD nomandatory
```

8. チェックポイントファイルを展開します。※owner が“performer”になるように展開してください。

実行コマンド

```
p4d -r /$P4ROOT -jr filename
```

または

```
sudo -u performer p4d -r /$P4ROOT -jr filename
```

実行後、db.ファイルが作成されたことを確認します

9. Performer サービスを開始します

実行コマンド

```
p4dctl start server-name
```

コマンド例

```
p4dctl start nomandatory
```

5. レプリケーションの設定と確認

【スタンバイサーバでの作業です】

1. ユーザ“service”が使用するレプリケーション用チケットを作成します

実行コマンド

```
p4 -p master_server:port -u service -E "P4TICKETS=/p4root_st/.p4tickets" login  
Enter password: (service ユーザのパスワード入力)
```

コマンド例

```
p4 -p 192.168.56.102:1666 -u service -E "P4TICKETS=/p4root_st/.p4tickets" login
```

2. レプリケーション用チケットの確認をします

実行コマンド

```
ls -la /p4root_st/.p4tickets
```

実行結果

```
-r----- . 1 performce performce ## m月 dd h:mm /p4root_st/.p4tickets  
例) -r----- . 1 performce performce 53 10月 10 16:48 /p4root_st/.p4tickets
```

3. レプリケーションの確認をします

実行コマンド

```
p4 -u super -p standby_server:port pull -lj
```

コマンド例)

```
p4 -u super -p 192.168.56.104:1666 pull -lj
```

■ 解説

pull -lj コマンド : レプリカとマスターにおけるカレントジャーナルの状態を表示するコマンドです

以下の内容が含まれます

- ・カレントジャーナルの状態
- ・ステートファイルの最終更新時刻
- ・サーバのローカル時刻
- ・タイムゾーン

表示されるメッセージ例

```
Current replica journal state is:   Journal 1,   Sequence 4878.  
Current master journal state is:   Journal 1,   Sequence 4878.  
The statefile was last modified at: 2019/10/16 15:16:27.  
The replica server time is currently: 2019/10/16 15:19:11 +0900 JST
```

※大規模なサイトの場合、同期に時間がかかる場合がありますので、時間を置いて再度コマンドを実行してください。

“Sequence”が同じ数値になれば同期完了です。

※エラーが表示される場合は、作業を停止し、弊社サポートまでご連絡ください。

6. フェイルオーバー : テスト運用

マスターサーバが起動できない・完全にダウンしていると判断できる状況を想定した運用のご案内です。
マスターサーバが起動している・ネットワーク接続が可能な状況は別の手順となりますのでご注意ください。

1. フェイルオーバーの確認【スタンバイサーバで作業します】

実行コマンド

```
p4 -u super -p standby-server:port failover -i -s master-serverID -w 10 -F
```

コマンド例)

```
p4 -u super -p 192.168.56.104:1666 failover -i -s commit -w 10 -F
```

■ 解説-----

- i フェイルオーバー動作中にマスターサーバを無視するオプションです
- s serverID 切り替えるサーバ（今回はマスターサーバ）の serverID を指定します
- w 10 フェイルオーバー実行までの待機時間を“秒”で指定します。今回は 10 秒としています
- F 強制的にフェイルオーバーを実行します(※マニュアルに記載のないオプションです)

【表示されるメッセージ】

Checking if failover might be possible ...

This failover requires that this server have the 'mandatory' option set since the server from which failover is to occur is not participating in the failover.

Checking for archive file content not transferred ...

Verifying content of recently updated archive files ...

After addressing any reported issues that might prevent failover, use --yes or -y to execute the failover.

※異なるメッセージ・エラーが表示される場合は、テストを中止し、弊社サポートまでご連絡ください。

2. フェイルオーバーの実行

実行コマンド

```
p4 -u super -p standby-server:port failover -i -s master-serverID -w 10 -F -y
```

コマンド例)

```
p4 -u super -p 192.168.56.104:1666 failover -i -s commit -w 10 -F -y
```

【表示されるメッセージ】

Attempting unsupported forced failover; attempting to continue through any errors encountered. This server might not be as expected after the forced failover.

Starting failover process ...

This failover requires that this server have the 'mandatory' option set since the server from which failover is to occur is not participating in the failover.

Waiting for 'pull -L' to complete its work ...

Checking for archive file content not transferred ...

Verifying content of recently updated archive files ...

Moving latest journalcopy'd journal into place as the active journal ...

Updating configuration of the failed-over server ...

Restarting this server ...

Perforce Server re-starting...

自動的に Perforce サーバが再起動され、マスターサーバへと切り替わります。

7. その他

Q 1. タイムゾーンの設定方法を教えてください

A 1. 以下の方法で設定してください

①Perforce の設定ファイルを開きます

実行コマンド `less /etc/perforce/p4dctl.conf.d/server-name.conf`

②タイムゾーンを設定ファイルに追記します

```
⋮  
P4USER    =    super  
P4SSLDIR  =    ssl  
TZ       =    Asia/Tokyo    ●一行追加する（実際に使用するタイムゾーンを入力）  
PATH      =    /bin:/usr/bin:/usr/local/bin:/opt/perforce/bin:/opt/perforce/sbin  
⋮
```

③Perforce を再起動します

実行コマンド `p4dctl restart server-name`

Q 2. 「接続できない」というエラーメッセージが表示されました。対応を教えてください

表示例

Connect to server failed; check \$P4PORT.

TCP connect to **server:port** failed.

connect: **server:port**: No route to host

A 2. 記載されているサーバの firewall が原因の場合があります。firewall を停止して接続できるかご確認ください