

## HA 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	master	standby
ServerID	master	standby
Option	—	mandatory ※nomandatory で構築後に変更
server-service	commit-server	standby
IP	192.168.56.114	192.168.56.115
Port	1666	1666
P4ROOT	/master	/standby

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. mandatory へ変更する.....	8
7. フェイルオーバー：テスト運用—マスターサーバが起動している場合.....	9
8. フェイルオーバー：テスト運用—マスターサーバが起動していない場合.....	10
9. フェイルオーバー：テスト運用—参照サーバの変更.....	11
10. その他.....	12

## 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“standby”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 standby
```

コマンド例)

```
p4 -u super -p master:1666 server standby
```

2. 設定を修正・追加します

赤字部分が修正・追記箇所です

ServerID: standby

Type: server

Services: **standby** ● standard (default) を上書き

Options: nomandatory ●変更しない※新規サーバは mandatory にできません。後で変更します

**ReplicatingFrom: master** ●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=192.168.56.114:1666
```

```
P4TICKETS=/standby/.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 192.168.56.114:1666 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@master

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 192.168.56.114:1666 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 “service” 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 * //... ●一行追加する
```

### 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 192.168.56.114:1666 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 “standby” 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 /standby
```

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 /standby/
```

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 /standby/
```

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 standby
```

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“standby”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 standby
```

#### 8. チェックポイントファイルを展開します。※owner が“perforce”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 perforce 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. Perforce 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 standby
```

## 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=/standby/.p4tickets" login  
Enter password: (service ユーザのパスワード入力)
```

コマンド例

```
p4 -p 192.168.56.114:1666 -u service -E "P4TICKETS=/standby/.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 /standby/.p4tickets
```

実行結果

```
-r-----. 1 performe performe ## m月 dd h:mm /standby/.p4tickets
```

例) -r-----. 1 performe performe 53 10月 10 16:48 /standby/.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 192.168.56.115:1666 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 8403.
```

```
Current master journal state is:    Journal 1,      Sequence 8403.
```

```
The statefile was last modified at: 2019/10/18 14:01:11.
```

```
The replica server time is currently: 2019/10/18 14:01:42 +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. mandatory へ変更する

【マスターサーバで作業します】

1. マスターサーバとスタンバイサーバのジャーナルの整合性を確認します

実行コマンド

```
p4 servers -J
```

実行結果

```
master '2019/10/18 14:07:44' commit-server 1/8773 1/8773 wadL/1 1
standby '2019/10/18 14:07:43' standby 1/8575 1/8575 wAdI/4 1
```

■ 解説

p4 servers -J はジャーナルコピーの進行状況を確認するコマンドです。

マスターサーバとスタンバイサーバのジャーナルの数値(緑文字)が異なる場合はまだ進行中であり、一致していないこととなります。再度実行し、マスターサーバとスタンバイサーバが一致することを確認します。

2. serverID “standby” の設定を表示します。

実行コマンド

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

コマンド例)

```
p4 -u super -p master:1666 server standby
```

3. serverID “standby” の Options を変更します

赤文字が変更箇所です

```
⋮
Services:      standby
Options:       mandatory ●nomandatory からの書き換え
ReplicatingFrom: master
⋮
```

4. mandatory に変更されたことを確認します

実行コマンド

```
p4 servers -J
```

実行結果 ※末尾に「mandatory」と表示されます

```
master '2019/10/18 14:09:47' commit-server 1/9179 1/9179 wadL/1 1
standby '2019/10/18 14:10:40' standby 1/9179 1/9179 wAdI/4 1 mandatory
```



## 7. フェイルオーバー：テスト運用—マスターサーバが起動している場合

マスターサーバが起動できており、ネットワークに接続できてい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ない・ネットワーク接続ができない状況は次頁をご覧ください。

### 1. フェイルオーバーの確認

実行コマンド【スタンバイサーバで作業します】

```
p4 failover
```

#### 【表示されるメッセージ】

Checking if failover might be possible ...

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 failover -y
```

#### 【表示されるメッセージ】

Starting failover process ...

Refusing new commands on server from which failover is occurring ...

Giving commands already running time to complete ...

Stalling commands on server from which failover is occurring ...

Waiting for 'journalcopy' to complete its work ...

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

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

Checking for archive file content not transferred ...

Verifying content of recently updated archive files ...

Stopping server from which failover is occurring ...

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。

## 8. フェイルオーバー：テスト運用—マスターサーバが起動していない場合

マスターサーバが起動できない・ネットワークに接続できない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できる状況は前頁をご覧ください。

### 1. フェイルオーバーの確認

実行コマンド【スタンバイサーバで作業します】

```
p4 failover -s master-server
```

コマンド例

```
p4 failover -s master
```

#### ■ 解説

-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

“-s”オプションが無い場合は以下のメッセージが表示されます

```
Server ID must be specified in the '-s' or '--serverid' argument for a failover without the participation of the server from which failover is occurring.
```

```
Empty identifier not allowed.
```

#### 【表示されるメッセージ】

```
Checking if failover might be possible ...
```

```
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 failover -s master-server -y
```

コマンド例

```
p4 failover -s master -y
```

#### 【表示されるメッセージ】

```
Starting failover process ...
```

```
Refusing new commands on server from which failover is occurring ...
```

```
Giving commands already running time to complete ...
```

```
Stalling commands on server from which failover is occurring ...
```

```
Waiting for 'journalcopy' to complete its work ...
```

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

```
Waiting for 'pull -u' to complete its work ...
```

```
Checking for archive file content not transferred ...
```

```
Verifying content of recently updated archive files ...
```

```
Stopping server from which failover is occurring ...
```

```
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。

## 9. フェイルオーバー：テスト運用—参照サーバの変更

この項目では、フェイルオーバーを実行後、旧マスターサーバ(本マニュアルの場合は master)を参照していたサーバでの対応についてのご案内です。ここではエッジサーバを例にしています。

※マスターサーバとスタンバイサーバに対する DNS のエイリアス設定がされている場合は対応は不要です。

### 1. 状況を確認する

【エッジサーバで作業します】

実行コマンド

```
p4 pull -lj
```

※このメッセージが表示される場合は、DNS のエイリアス設定がされていません。

Replica access to P4TARGET server failed.

Remote server refused request. Please verify that service user is correctly logged in to remote server, then retry.

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

connect: **master-server:port**: Connection refused

### 2. “P4TARGET”を修正します

実行コマンド

```
sudo -u perforce p4d -r /edge1 "-cset edge1#P4TARGET=新 master-server:port"
```

コマンド例

```
sudo -u perforce p4d -r /edge1 "-cset edge1#P4TARGET=192.168.56.115:18800"
```

### 3. 変更を確認します

実行コマンド

```
sudo -u perforce p4d -r /edge1 "-cshow"
```

コマンド結果

⋮

```
edge1: P4TARGET = 新 master-server:port
```

⋮

### 4. 同期の確認

実行コマンド

```
p4 pull -lj
```

コマンド結果

Current replica journal state is: Journal 4, Sequence 6065.

Current master journal state is: Journal 4, Sequence 6065.

The statefile was last modified at: 2019/10/21 02:38:44.

The replica server time is currently: 2019/10/21 02:38:51 +0900 JST

“Sequence”が同じ値になれば同期完了です。

## 10. その他

Q1. タイムゾーンの設定方法を教えてください

A1. 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（記載は日本時間の場合）
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`

Q2. 「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
```

A2. 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