

Apache on CLUSTERPRO for Linux HOWTO

1 はじめに

この文章は、Linuxの標準HTTPサーバであるApacheとCLUSTERPRO for Linuxを組み合わせることでフェイルオーバークラスタを構成するのに必要な情報を記述したものです。HTTPサーバをフェイルオーバークラスタ化することにより、単一Linuxサーバでは達成し得ない可用性・信頼性の高いHTTPサーバを構築することができます。

この文章を書くにあたって次のバージョンのソフトウェアを使用しました。この他のバージョンのソフトウェアを使用した場合でも、いくつかの設定項目の読み替えでクラスタ化できると思われます。

- TurboLinux Server 日本語版6.1
- CLUSTERPRO for Linux Ver1.0
- Apache-1.3.12-5

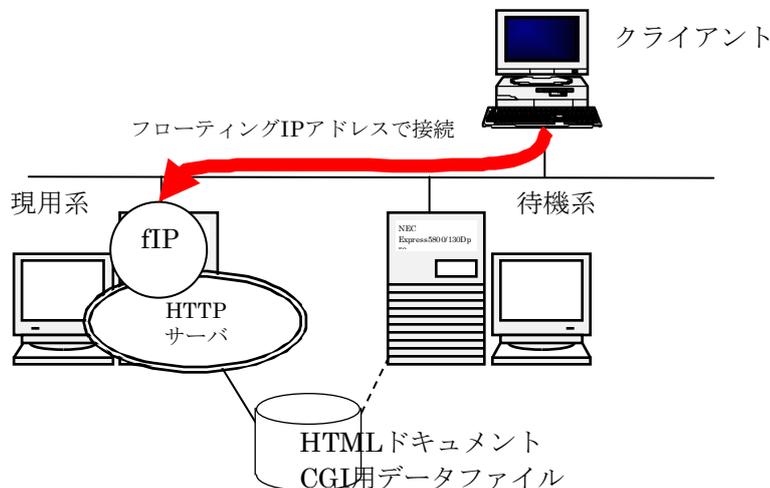
2 責任範囲

この文章は、Apacheをクラスタ化するための注意点や設定例を参考情報として示すものであり、これらの動作保証を行うものではありません。

3 どのようにクラスタ化するか

(1) HTMLドキュメント、およびCGIで用いるデータファイルを共有ディスク上のパーティションに配置・設定を行うことで、HTTPデータの引き継ぎを行います。これによって、現用系サーバの障害時に、待機系サーバでHTTPサービスの提供が可能になります。

(2) この文章で扱っているのは、HTTPサーバが1ノードあたり1つのみ動作する、片方向スタンバイシステムの構築例です。



図はHTTPサーバを2ノードクラスタ化したときの構成イメージです。

CLUSTERPROのフェイルオーバーグループには、フローティングIPアドレス、HTMLドキュメントを格納する切り替えパーティション、Apacheの起動・終了スクリプトを設定します。また、Apacheが起動するサーバ上では、HTMLドキュメントを置く共有ディスク内のディレクトリを設定します。

これによって、現用系サーバに障害が発生したときに、待機系でフェイルオーバーグループを起動することで、クライアントからはサービスノードがどちらかを気にせずにHTTPサービスを受受できます。

4 事前準備

1. ユーザnobodyのユーザID、グループnobodyのグループIDが、クラスタを構成する全てのサーバで

同一であることを確認してください。

(Apacheがnobodyのユーザアカウントを用いるため)

2. HTMLドキュメントを共有ディスク内に置く必要があります。

まず、ディレクトリを作成します。

以下に、共有ディスクのマウントポイントを/mnt/sdb2に置いた場合のコマンド入力例を示します。

```
umask 022
mkdir /mnt/sdb2/httpd
mkdir /mnt/sdb2/httpd/html
mkdir /mnt/sdb2/httpd/cgi-bin
mkdir /mnt/sdb2/httpd/icons
```

次に、HTMLドキュメントの各ファイルを、共有ディスクのマウントポイントの下のhttpdディレクトリ配下に配置します。

5 クラスタ化のための設定

5.1 Apacheコンフィグファイルの編集

クライアント側からHTMLデータの要求を受けたときに、共有ディスク内のファイルが読み込まれるようにするため、httpd.conf,access.conf,srm.conf等の変更が必要です。

それには、コンフィグファイルを直接編集する方法と、Management Consoleを用いて設定する方法の2通りの方法があります。

なお、設定はクラスタを構成する全てのサーバで行ってください。

※以下説明内で、網掛け斜体字の部分は、環境に合わせて変更してください。

(1) Management Consoleを用いて設定を行う場合

1. 仮想サーバの基本設定

「サービス」 - 「Webサーバ (Apache)」で、「仮想サーバ設定」の[編集]ボタンを押します。

「■仮想サーバの基本設定」で、以下のように設定します。

サーバ名: _default_ 又は フローティングIPアドレス

ドキュメントルート: /mnt/sdb2/httpd/html



2. ディレクトリ設定

「ディレクトリ設定」で、追加ボタンを押します。

以下の画面が表示されます。

■ ディレクトリの設定

ディレクトリ名:

デフォルトファイル名:

実行権:

- CGIの実行を有効にする
- SSIを有効にする
- ディレクトリリストを表示する
- シンボリックリンクの参照を有効にする
- .htaccessの使用を許可する

ここで、つぎのように設定してください。

ディレクトリ名: `/mnt/sdb2/httpd/html`
 デフォルトファイル名: `index.html`
 CGIの実行を有効にする: **OFF**
 SSIを有効にする: **OFF**
 ディレクトリリストを表示する: **OFF**
 シンボリックリンクの参照を有効にする: **OFF**
 .htaccessの使用を許可する: **OFF**

最後に「設定ボタン」を押し、設定を確定します。

同様に、以下の二つのディレクトリを設定します。

ディレクトリ名: `/mnt/sdb2/httpd/icons`
 デフォルトファイル名: (※空欄のまま)
 CGIの実行を有効にする: **OFF**
 SSIを有効にする: **OFF**
 ディレクトリリストを表示する: **OFF**
 シンボリックリンクの参照を有効にする: **OFF**
 .htaccessの使用を許可する: **OFF**

ディレクトリ名: `/mnt/sdb2/httpd/cgi-bin`
 デフォルトファイル名: (※空欄のまま)
 CGIの実行を有効にする: **ON**
 SSIを有効にする: **OFF**
 ディレクトリリストを表示する: **OFF**
 シンボリックリンクの参照を有効にする: **OFF**
 .htaccessの使用を許可する: **OFF**

最終的に、「ディレクトリ設定」の表示は、以下のようになります。

■ ディレクトリ設定		
操作	実ディレクトリ名	
追加		
編集 削除	/mnt/sdb2/httpd/html	
編集 削除	/mnt/sdb2/httpd/cgi-bin	
編集 削除	/mnt/sdb2/httpd/icons	

3. 仮想パスの設定

仮想パス設定で、以下の項目を追加してください。

仮想パス名: /cgi-bin

実ディレクトリ名: /mnt/sdb2/httpd/cgi-bin

仮想パス名: /icons

実ディレクトリ名: /mnt/sdb2/httpd/icons

■ 仮想パス設定		
操作	仮想パス名	実ディレクトリ名
追加		
編集 削除	/cgi-bin	/mnt/sdb2/httpd/cgi-bin
編集 削除	/icons	/mnt/sdb2/httpd/icons

(2) コンフィグファイルを直接編集する場合

/etc/httpd/conf内の設定ファイルを編集します。

※ 以下に設定例を示します。これを参考に、httpd.conf,srm.conf,access.conf等の設定ファイルを編集してください。(注:本サンプルをそのまま使用しないでください)

なお、ここでは以下の環境を想定しています。

共有ディスクパーティションのマウントポイント: /mnt/sdb2

フローティングIPのサーバ名: www.clusterpro.co.jp

```
ServerName www.clusterpro.co.jp
DocumentRoot </mnt/sdb2/httpd/html>
Alias /icons/ /mnt/sdb2/httpd/icons/
ScriptAlias /cgi-bin/ /mnt/sdb2/httpd/cgi-bin/
AddHandler cgi-script .cgi

<Directory "/mnt/sdb2/httpd/html">
  DirectoryIndex index.html
  Options -ExecCGI -Includes -Indexes -FollowSymLinks
  AllowOverride None
  Order allow,deny
  Allow from all
</Directory>
<Directory "/mnt/sdb2/httpd/cgi-bin">
  DirectoryIndex index.cgi
  Options +ExecCGI -Includes -Indexes -FollowSymLinks
  AllowOverride None
  Order allow,deny
  Allow from all
</Directory>
<Directory "/mnt/sdb2/httpd/icons">
  Options -ExecCGI -Includes -Indexes -FollowSymLinks
  AllowOverride None
  Order allow,deny
  Allow from all
</Directory>
```

5.2 CLUSTERPROの設定

5.2.1 フェイルオーバーグループの作成

HTTPサービスを実現するフェイルオーバーグループを一つ作成します。

フェイルオーバーグループに設定するリソースは次のとおりです。

- ・ 切り替えディスクパーティション
- ・ フローティングIPアドレス
- ・ スクリプト

5.2.2 スクリプトの作成

クラスタの起動時やフェイルオーバー時に、Apacheのサービスを適切に起動・停止させるために、以下のような手順で、フェイルオーバーグループの開始・終了スクリプトを編集する必要があります。

1. CLUSTERPROのマネージャ内のグループを右クリックし、プロパティを選択します。



2. 「処理選択」ウインドウで、「参照・更新」を選択し、OKボタンを押します。
3. 「スクリプト」タブを選択し、START.BATとSTOP.BATを編集します。

以下に、スクリプトの一例を示します。

※下線が引かれたところが、デフォルトのスクリプトから追加された部分です。

START.BAT

```

#!/bin/sh
#*****
#*           START.BAT           *
#*****

if [ "$ARMS_EVENT" = "START" ]
then
  if [ "$ARMS_DISK" = "SUCCESS" ]
  then
    armlog "NORMAL1"
    armlog "/etc/rc.d/init.d/httpd start`"
    if [ "$ARMS_SERVER" = "HOME" ]
    then
      armlog "NORMAL2"
    else
      armlog "ON_OTHER1"
    fi
  else
    armlog "ERROR_DISK from START"
  fi
elif [ "$ARMS_EVENT" = "RECOVER" ]
then
  armlog "RECOVER"
elif [ "$ARMS_EVENT" = "FAILOVER" ]
then
  if [ "$ARMS_DISK" = "SUCCESS" ]
  then
    armlog "FAILOVER1"
    armlog "/etc/rc.d/init.d/httpd start`"
    if [ "$ARMS_SERVER" = "HOME" ]
    then
      armlog "FAILOVER2"
    else
      armlog "ON_OTHER2"
    fi
  else
    armlog "ERROR_DISK from FAILOVER"
  fi
else
  armlog "NO_ARM"
fi
armlog "EXIT"
exit 0

```

STOP.BAT

```

#!/bin/sh
#*****
#*           STOP.BAT           *
#*****

if [ "$ARMS_EVENT" = "START" ]
then
  if [ "$ARMS_DISK" = "SUCCESS" ]
  then
    armlog "NORMAL1"
    armlog "/etc/rc.d/init.d/httpd stop`"
    if [ "$ARMS_SERVER" = "HOME" ]
    then
      armlog "NORMAL2"
    fi
  fi

```

```

else
    armlog "ON_OTHER1"
fi
else
    armlog "ERROR_DISK from START"
fi
elif [ "$ARMS_EVENT" = "FAILOVER" ]
then
    if [ "$ARMS_DISK" = "SUCCESS" ]
    then
        armlog "FAILOVER1"
        armlog "`/etc/rc.d/init.d/httpd stop`"
        if [ "$ARMS_SERVER" = "HOME" ]
        then
            armlog "FAILOVER2"
        else
            armlog "ON_OTHER2"
        fi
    else
        armlog "ERROR_DISK from FAILOVER"
    fi
else
    armlog "NO_ARM"
fi
armlog "EXIT"
exit 0

```

5.2.3 注意事項

CGI等を用いた動的なページが存在する場合には、クライアント側から受信したデータなどを保存するファイルを共有ディスク上に置く必要があります。もしそうっていない場合は、HTMLドキュメントを書き換えるなどして対応してください。

6 動作が確認されたページ

下記のようなページで、動作を確認しました。

1. 静的なページ（サーバ側からクライアントに情報を提供するだけのページ）
（Apacheのデフォルトページで動作を確認しました）
2. CGIを用いたページ
下記のようなカウンタプログラムで、正常動作、フェイルオーバー時の適切なデータの引継ぎを確認しました。

sample.cgi

```

#!/usr/bin/perl

print "Content-type: text/html¥n";
print "¥n";

open(IN,"data/number.dat");
$number = <IN> + 0;
$number++;
close(IN);
open(OUT, ">data/number.dat");
print OUT $number;
close(OUT);

```

```
print "<html>¥n";
print "$number¥n";
open(IN, "/hostname");
print <IN>;
close(IN);
print "</html>¥n";
```

3. Cookieを用いたページ

下記のようなカウンタプログラムで、正常動作時、フェイルオーバ時における動作確認を行いました。

cookie.html

```
<html><head>
<title>cookie sample</title></head>
<body>

<script language="JavaScript">

<!--
cookie = parseInt(document.cookie.substring(5,8));
if (isNaN(cookie)) {
    cookie=0;
} else {
    cookie++;
}
document.write(cookie + "<BR>");
document.cookie = "NAME=" + cookie;
// -->
</script>
</body>
</html>
```