jail環境を作る

2008年04月05日 13時51分32秒

変更修正

2008/04/05 otsune様よりご指摘があり/etc/rc.conf の sendmail_enable=“NONE”の記述を消します。

NONE 記述は互換の為に残してあり、既に修正はされているとのことでした 6.3RELEASEの /etc/rc.d/sendmail を見てみると確かにそうですね。

```
case ${sendmail_enable} in
[Nn][Oo][Nn][Ee])
        sendmail_enable="NO"
        sendmail_submit_enable="NO"
        sendmail_outbound_enable="NO"
        sendmail_msp_queue_enable="NO"
        ;;
esac
```

NONE→NOに変更して確認してみました。otsune様のおっしゃるとおりでございます orz

6.2RELEASEの頃、確かに当方のマシンでは NONE にしなければsendmailが動いてしまっていたのですが6.3RELEASEにアップグレードしてしまっているのでもう確認は取れません。 結構古いリリースからアップグレードし続けていたのでおそらくその際に問題があって気づかぬままだったのかもしれません。

ezJailでjail環境を作る

単純なシェル＋サービスを使ったバッチ処理回りのテスト。どうもrootを使いたいらしい rootを必要とする辺りでもう失敗作の雰囲気ムンムンだけど、現場では結構よくある話。

しかしだ、結構気にしてアップデートやら設定やらを見直してきたサーバなのだrootをおいそれと貸す訳にはいかない。

環境

FreeBSD6.2RELEASE での説明jail は FreeBSD独自の機構です chroot よりも強化されていてセキュアだと言われてます。が、本ドキュメントの作者には判断できません。

環境構築の選択肢

さて、方法としては

1. jailで仮環境を作ってその環境のrootをあげる。

2. VMソフトウエアで仮想環境を作ってその環境のrootをあげる。

がある。

VMwareとか□VirtualPCなんかが仮想環境としては有名。

最近だとこれらの製品で作った仮想環境を扱える無償のプレイヤーがある。しかし当然ながら環境を作成するためには製品購入が必要になる。

フリーのものだと□qemuが今現在お勧め。フリ□□□の作者もこれを使っている人が多い印象あり。あとはVirtualBoxが今後に期待、かな。

だが、これらを利用するに際してはネットワーク周りの部分で面倒がある。今回の用件では、その仮想環境下でネットワークサービスを動かす必要があるからだ。

qemuだと、仮想環境→実環境への通信は楽。でも実環境→仮想環境への通信は色々小細工がいる。イメージ的には

```
 | ←-- qemu環境 - - →|
[仮想環境] ←→ [NAT box] ←→ [実環境]
```

になっていて、ブロードバンドルータでよくFAQにあがっているような「外部から内部の特定PCへの接続は？」を解決する必要がある。 なので□I/Fにエリアスで IPアドレスを振って使うjailの方が楽だろうなと判断。

jailでいくことに決める。

ezjailを使って構築

マニュアルにあるjailツリーを作るのはちょっと読んだだけでゲンナリする。 これを楽にしてくれる道具として ezJailというものがある。ただしこいつは 6.0RELEASE以降でしか使えないので注意□ portsで導入するのが手っ取り早い。

```
 failsafe# cd /usr/ports/sysutils/ezjail
 failsafe# make ; make install
```

ezjail.conf を定義

/usr/local/etc/ に ezjail.conf.sample があるので、これを ezjail.conf の名前でコピーする。 デフォルトだと /usr/jails に環境を作るようなので□ezjail_jaildirのコメントをはずして環境を作るディレクトリを指定する。

```
 # ezjail.conf - Example file, see ezjail.conf(5)
 #
 # Note: If you alter some of those variables AFTER creating your first
 # jail, you may have to adapt /etc/fstab.* and EZJAIL_PREFIX/etc/ezjail/*
by
 # hand

 # Location of jail root directories
```

```
#
# Note: If you have spread your jails to multiple locations, use softlinks
# to collect them in this directory
ezjail_jaildir=/home/jails          ← ここ

# Location of the tiny skeleton jail template
```

/usr/src 以下を展開する

OSのソースを展開しておく。普通なら /usr/src に展開されているはず。 無ければ sysinstall コマンド等々で展開しておく。 ソースからのOSアップグレードと同じように□ezJailではここでベースとなる環境をコンパイルし、そのコンパイル結果を ${ezjail_jaildir}/ へインストールするようだ。

※jailのマニュアルにある「 フルツリー」の構築をやってる？

ベースの作成

以下のコマンドラインで /home/jails にベースを作らせる□ failsafe# ezjail-admin update 長々コンパイルが始まるのでお茶でも飲んで待つ。
…
…
…あちゃあ。手間的には qemuでOSインストールするのと変わらんかもしれん…

コンパイル初期では□/home/jails/fulljail ディレクトリにフルビルドした内容が格納される。 ビルドが完了すると、

```
 Note: a non-standard /etc/make.conf was copied to the template jail in
order to get the ports collection running inside jails.
 failsafe#
```

のメッセージが出力され、

```
drwxr-xr-x   9 root  wheel  512  1 18 17:16 basejail
drwxr-xr-x   3 root  wheel  512  1 18 17:18 flavours
drwxr-xr-x  12 root  wheel  512  1 18 17:18 newjail
```

の３ディレクトリが最終的に残る。

親環境の設定

親環境で必要な設定を行っておく。 というか、

- 親環境でネットワークI/Fに対してIPアドレスのエリアスを切る
- 親環境でエリアスを切ったIPアドレスを利用しないように設定を変更する

を実施する。

ネットワークI/Fにエリアスを切る

このエリアスを切って指定したIPアドレスをjail環境で利用することになる。 例えばrl0 のI/Fに もう一つ 192.168.1.200 を名乗らせたければ(エリアスを切りたければ）次のコマンドを実行する。

```
failsafe# ifconfig rl0 alias 10.204.187.200 netmask 255.255.255.255
```

/etc/rc.conf の修正

また、このままだとマシンのリブートで定義が消えるので /etc/rc.conf にも記述を入れておく。

```
ifconfig_rl0_alias0="inet 192.168.1.200  netmask 255.255.255.255"
```

もっと追加したければ次のように

```
ifconfig_rl0_alias0="inet 192.168.1.200  netmask 255.255.255.255"
ifconfig_rl0_alias1="inet 192.168.1.201  netmask 255.255.255.255"
ifconfig_rl0_alias2="inet 192.168.1.202  netmask 255.255.255.255"
```

ifconfig_rl0_aliasX のXは、ゼロから始めてエリアスの数だけ増えていく。

親環境での使用アドレス限定

このままだとtelnet等のサービスが 192.168.1.200 でも反応してしまう。
例えばrl0 に付与されているIPアドレスが 192.168.1.100 だった場合、
telnet で 192.168.1.100 にアクセスしても 192.168.1.200 にアクセスしても親環境にログインできてしまう。
つまり、せっかくjail環境用に準備したIPアドレスがjailで使えない、ということに。

なので、親環境で使うIPアドレスを限定する。 親環境でどの様なサービスを起動しているかによる。以下にドキュメント作成者の環境での例を挙げておく。 定義を書き換えたら、各サービスの再起動を忘れぬこと。

apache

httpd.conf の Listen に親自身のIPアドレスを指定する。

```
#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses (0.0.0.0)
#
#Listen 12.34.56.78:80
```

```
 Listen 192.168.1.100:80  ← 明示的に 192.168.1.100:80 である旨指定する
```

clamAV

clamd.conf の TCPAddr に親自身のIPアドレスを指定する。

```
 # TCP address.
 # By default we bind to INADDR_ANY, probably not wise.
 # Enable the following to provide some degree of protection
 # from the outside world.
 # Default: disabled
 #TCPAddr 127.0.0.1
 TCPAddr 192.168.1.100  ← 明示的に 192.168.1.100 である旨指定する
```

freshclam.conf の LocalIPAddress に親自身のIPアドレスを指定する。

```
 # Use aaa.bbb.ccc.ddd as client address for downloading databases. Useful
for
 # multi-homed systems.
 # Default: Use OS'es default outgoing IP address.
 #LocalIPAddress aaa.bbb.ccc.ddd
 LocalIPAddress 192.168.1.100  ← 明示的に 192.168.1.100 である旨指定する
```

inetd

sambaでswatを使うような場合には inetd.conf を使うので□/etc/rc.conf への指定を行う。

```
 inetd_enable="YES"
 inetd_flags="-wW -C 60 -a 192.168.1.100" ← 明示的に 192.168.1.100 である旨指定す
る
```

sshd

sshd_config の ListenAddress に親自身のIPアドレスを指定する。

```
 #Port 22
 #Protocol 2
 #AddressFamily any
 #ListenAddress 0.0.0.0
 #ListenAddress ::
 ListenAddress 192.168.1.100 ← 明示的に 192.168.1.100 である旨指定する
```

jail環境（インスタンス）の作成

jailインスタンスを作成する。

```
failsafe# ezjail-admin create instanceName 192.168.1.200
```

instanceName にjail環境の名称(インスタンス名)を入れる。IPアドレスは、エリアスを切ったIPアドレスで他のインスタンスやサービスと競合しないものを充てる。 もし、この時にIPアドレスがらみでおかしなことがあれば、以下のように警告してくれる。

```
 Warning: Some services already seem to be listening on IP 192.168.1.200
   This may cause some confusion, here they are:
 root     ntpd       612   7  udp4   192.168.1.200:123     *:*
 Warning: Some services already seem to be listening on all IP, (including
192.168.1.200)
   This may cause some confusion, here they are:
 pgsql    postgres   685   4  tcp4   *:5432                *:*
 root     ntpd       612   4  udp4   *:123                 *:*
 root     syslogd    518   7  udp4   *:514                 *:*
 failsafe#
```

この例だと、

- ntpd が既に 192.168.1.200:123 を使ってますよー
- PostgreSQL、ntp、syslog のサービスが全IPアドレスで待ち受けしてますよー

と教えてくれる。

ezjail-adminコマンド自体はシェルスクリプトで、この表示部分を調べてみるとsockstatコマンドの表示であった。なので、以下のようなコマンドで同じことができる。

- 既に192.168.1.200でListenしているものを確認

```
failsafe# sockstat -4 -l | grep "192.168.1.200:[[:digit:]]"
root     ntpd       627   10 udp4   192.168.1.200:123       *:*
bind     named      514   28 udp4   192.168.1.200:53        *:*
bind     named      514   29 tcp4   192.168.1.200:53        *:*
failsafe#
```

- 全アドレスでListenしているものを確認

```
failsafe# sockstat -4 -l | grep "*:[[:digit:]]"
root     nmbd       39738 7  udp4   *:137                 *:*
root     nmbd       39738 8  udp4   *:138                 *:*
root     smbd       39731 19 tcp4   *:445                 *:*
root     smbd       39731 20 tcp4   *:139                 *:*
root     Xorg       900   3  tcp4   *:6000                *:*
www      httpd      660   3  tcp46  *:80                  *:*
www      httpd      659   3  tcp46  *:80                  *:*
www      httpd      658   3  tcp46  *:80                  *:*
www      httpd      657   3  tcp46  *:80                  *:*
www      httpd      656   3  tcp46  *:80                  *:*
root     httpd      649   3  tcp46  *:80                  *:*
root     ntpd       627   4  udp4   *:123                 *:*
root     nfsd       578   3  tcp4   *:2049                *:*
root     mountd     576   4  udp4   *:699                 *:*
```

```
root     mountd     576   5  tcp4   *:702                 *:*
root     rpcbind    524   9  udp4   *:111                 *:*
root     rpcbind    524   10 udp4   *:870                 *:*
root     rpcbind    524   11 tcp4   *:111                 *:*
bind     named      514   32 udp4   *:51974               *:*
root     syslogd    449   9  udp4   *:514                 *:*
failsafe#
```

jail環境起動

起動前に、/etc/rc.conf に次の一行を追加する。

```
ezjail_enable="YES"
```

jail起動スクリプトである /usr/local/etc/rc.d/ezjail.sh はこの定義を見てjailインスタンスをたたき起こす。rc.conf への追加が終わったら以下のコマンドラインでjailを起動する。

```
failsafe# /usr/local/etc/rc.d/ezjail.sh start
 ezjailConfiguring jails:.
Starting jails: instaceName.
failsafe#
```

起動したインスタンスの一覧を確認するには、jls コマンドを使う。

```
failsafe# jls -la
   JID  IP Address      Hostname                      Path
     1  192.168.1.200   instanceName
/home/jails/instanceName
failsafe#
```

起動したインスタンスにログインするには、jexec コマンドを使う。

```
failsafe# jexec 1 /bin/sh
#
```

1 はJID で、jls で表示されるインスタンスにヒモ付く。上記例で言うと、インスタンス instanceName にログインするためには、instanceName に割り振られたJID “1” を使ってログインすることになる。

ログインしたら、あとは通常のサーバ管理に従った設定を行えばいい。

jail環境内でのサービス起動

起動直後のjail環境内には syslog、cron、sendmailぐらいしかサービスが動いていない。 ここでは例として sshd のサービスを起動させる方法を記述する。

rc.conf のベース

まずは/etc/rc.confの雛形。

```
inetd_enable="YES"
inetd_flags="-wW -C 60 -a xxx.xxx.xxx.xxx"
syslogd_flags="-s -a xxx.xxx.xxx.xxx"
rpcbind_enable="NO"
network_interfaces=""
hostname="hogehoge.hgotoh.jp"
sendmail_enable="NO"
```

xxx.xxx.xxx.xxx はそのJail環境に割り当てるIPアドレス inetdを使わない場合は " NO"を入れておく。
sendmailは " NO"ではなく"NONE"に。これ、昔からこうなんだよなぁ。直す気無いのかしらん？
ネットワークは " " にしておく。

sshd

jail環境内の /etc/rc.conf に次の一行を追加する rc.conf が無ければ新規に作る。

```
sshd_enable="YES"
```

次に /etc/rc.d/ へ移動し、 sshd スクリプトを起動すれば良い。次回からは、インスタンス起動時に自動でsshdサービスが起動する。

```
# cd /etc/rc.d
# ./sshd start
Generating public/private rsa1 key pair.
Your identification has been saved in /etc/ssh/ssh_host_key.
Your public key has been saved in /etc/ssh/ssh_host_key.pub.
The key fingerprint is:
ea:c7:fb:10:7b:f7:79:6d:8e:ca:ea:55:16:a6:84:4b root@instanceName
Generating public/private dsa key pair.
Your identification has been saved in /etc/ssh/ssh_host_dsa_key.
Your public key has been saved in /etc/ssh/ssh_host_dsa_key.pub.
The key fingerprint is:
a7:a9:13:aa:72:98:c4:dd:3e:bb:3a:d7:9d:3c:f0:7a root@instanceName
Generating public/private rsa key pair.
Your identification has been saved in /etc/ssh/ssh_host_rsa_key.
Your public key has been saved in /etc/ssh/ssh_host_rsa_key.pub.
The key fingerprint is:
56:2e:84:da:a6:a9:df:b8:e4:9c:97:10:a0:2d:07:66 root@instanceName
Starting sshd.
```

以降まだ作業中です

参考

- 悪魔茶屋 - ezjail
- otsune's FreeBSD memo :: jailの作り方

- KMsWiki: FreeBSD/jail - □jailを利用すると□FreeBSDの中で、もう一つ(あるいは複数)のF...

jail, FreeBSD, 技術資料