

動態 DNS 設定技巧

這原本是 Leo 兄的來信指教，因為實在太有參考價值了，於是公諸同好。
所有對 dynamic dns 技術有興趣的朋友，都應該看的。

```
--
----- Original Message -----
From: LeoLiou
To: netman
Sent: Monday, February 11, 2002 6:59 AM
Subject: about dns
```

Hi, kenny:

我是 Leo，好久不見囉~~ 在此跟您拜個年，新年快樂喔~ :-)

放假回來翻了您的 DNS 教學，嗯，太棒囉，補充了許多許多的新東西，
昨天我回了一篇有關 "動態 IP 架站" 的討論文章，裡面是提到動態更新
DNS 的，自行架設 Dynamic DNS 與 client 端的 script 更新，不過我在
DNS 中，不是使用 allow-update，而是使用 update-policy，update-policy
比起 allow-update 更有彈性，比方，可以限定什麼 key 可以做什麼事。

舉例來說，我的 named.conf 長得像是這樣..

```
// ----- 宣告 Key 的部分-----
// P.S: 底下的 leo, mail key 是不正確的
key "leo" {
    algorithm hmac-md5;
    secret "hB/XM2eFTyxA5r/scaut0Z==";
};

key "mail" {
    algorithm hmac-md5;
    secret "ht5TkKKFP5l8u9ZTcDbStw==";
};

// ----- 宣告 ZONE 的部分-----
zone "sayya.org" {
    type master;
    file "named.sayya";

    notify yes;

    also-notify {
        140.128.78.250;
        211.23.99.147;
        61.218.164.83;
    };

    allow-transfer {
        140.128.78.250;
```

```

        211.23.99.147;
        61.218.164.83;
    };

    allow-query {
        0.0.0.0/0;
    };

    update-policy {
        // 使用 mail HOST 的 key，僅有權限變更 mail.sayya.org 與 rsync.sayya.org 的 A record
        grant mail name mail.sayya.org. A;
        grant mail name rsync.sayya.org. A;
        // 使用 leo USER 的 key，可以完全控制 sayya.org，比方新增一 subdomain，指定 DNS
        // 或是增加一 A, CNAME, TXT.. record 等等
        grant leo subdomain sayya.org. ANY;
    };
};
};

```

大概就是這個樣子，另外附上之前回覆的那篇..

作者: leoliou (暖冬) 看板: Linux
 標題: Re: 動態 IP 架站 @_@
 時間: Sun Feb 10 14:49:30 2002

※ 引述《skchen.bbs@bbs.nsysu.edu.tw (Y凱)》之銘言：

> 小弟在之前就有提過相關的問題
 > 大家用的動態IP對映的系統都不一樣
 > 我是用 DynDNS 的系統
 > 也不知道為什麼跑了一段時間就會產生一大堆殭屍程序 @_@
 > 更神的是找不到程序在哪裡(還是我比較遜腳????) @_@
 > 這好像是 DynDNS 給的 ddClient 程式有 BUG 吧
 > 我利用了 perl 和 bash shell script 寫了一些小程式
 > 可以用來偵測 PPP 介面是否存在
 > 如果不存在就連線 ^o^

嗯，看到這封信感覺是真的有討論的氣氛，超喜歡這種感覺~ :)

記得以前還用 ADSL 撥接制的时候，是使用 rp-pppoe 連線的，會在斷線後自動重新連線，這個可以解決斷線的問題。

再來是重點了，Dynamic IP 怎麼辦呢？我目前的環境是雙向 Cable，IP 也是由 DHCP Server 取得 Dynamic IP 的。當然，如果您去申請 dns2go 類似的程式，是可以解決您動態 IP 的問題，但是缺點是您不可以使用自己的網址，一定要是某一些已經律定好的網址，或者是有可能要求您把 Primary DNS 指向該公司代管，另外，就是您可能還要在 Server 中執行該公司所提供的動態 IP 更新程式.. 無論如何，就會讓我聯想到系統安全性的問題.. 對我來說，非常不方便。

於是，我找了一台固定 IP 的 Server，架上了 Bind，這台就是我自己網域的 Primary DNS，也就是 Dynamic DNS，如何設定，晚點提到。而這台 DNS 的固定 IP 假設為 61.22.33.20，Domain 為 example.com，註冊名稱為 ns.example.com :)

再來，比方我的 mail server 架在家裡，就是我现在雙向 Cable 的線路上，這台 mail server 的 Domain name 假設為：mail.example.com
 我只要定時向我的 Primary DNS 更新我的 mail server 的 IP 就可以囉，

Primary DNS 怎麼判斷允許哪些電腦做更新呢？有兩種方式，第一種方式為在 Primary DNS 中設定 allow-update，設定某些固定 IP 可以向 Primary DNS 更新資料，但是比較沒有彈性。另外一種就是待會要介紹的 update-policy。

allow-update 可以允許特定的 IP 或 key 來做動態更新，預設是不允許任何 IP 更新。

update-policy 在 BIND 9 才提供，不用指定某特定的 IP 才可以做動態更新，而是要憑 key 來決定更新權限。Primary DNS 有這個 key，mail server 往後只要憑這個 key 就可以動態更新 IP 了。當然了，mail server 與 Primary DNS 的 key 得一樣才行。

key 就是用 `dnssec-keygen` 工具產生出來的簽證，現在於 Primary DNS 上，產生 key，並發給 mail server。

首先，產生 mail server 用的 key：

```
# dnssec-keygen -a HMAC-MD5 -b 128 -n HOST mail
Kmail.+157+44587
#
```

查看一下，結果會產生了檔名類似 `Kmail.+157+44587.key` 與 `Kmail.+157+44587.private` 兩個檔案。

其中 `Kmail.+157+44587.key` 檔案內容長得類似這樣：

```
# less Kmail.+157+44587.key
mail. IN KEY 512 3 157 BJ7y6dZXchy3u0B4hRLksQ==
#
```

檔案內容 `BJ7y6dZXchy3u0B4hRLksQ==` 這是編碼過的，就是所謂的 key。

首先，您得將這兩個 key 複製到 mail server 上面，建議使用 `sftp` 傳輸，避免被竊。

現在，開始 Primary DNS 上的設定了。

----- /etc/named.conf -----

```
key "mail" {
    algorithm hmac-md5;
    secret "BJ7y6dZXchy3u0B4hRLksQ==";
};

zone "example.com" {
    type master;
    file "named.example";

    update-policy {
        grant mail name mail.example.com. A;
        // mail key 僅允許更新 mail.example.com 的 A record.
    };
};
```

----- End of File -----

Primary DNS 上面的設定完成後，請記得重跑 `bind`。

接下來，在 mail server 方面，要如何更新呢？

當然，最重要的，首先要取得 mail server 的 key。剛剛已經有提到，在 Primary DNS 產生後，用 `ftp` 傳輸到 mail server 來。

使用 `nsupdate` 工具向 Primary DNS 做更新動作：

```
$ nsupdate -k Kmail.+157+44587.key
> server ns.example.com           // 指定 Primary DNS
> update delete mail.example.com A // 先刪除舊資料
```

```
> update add mail.example.com 0 A 210.64.233.10 // 再新增資料
> send // 送出到 Primary DNS
$ // Ctrl-C or Ctrl-D 離開
```

驗證一下，是不是更新成功：

```
$ host mail.example.com ns.example.com
mail.example.com has address 210.64.233.10
$
```

恭喜您更新成功了，如果您沒有成功，請到 Primary DNS 上看 messages 紀錄檔。注意，key 檔案的權限，應該只有 owner 可以讀寫，應為 -rw----- 請注意。執行 nsupdate 的 user，應為 key 的 owner，否則會發生 Permission denied 的情形。

如果您要讓您的電腦自動更新，這裡有我自己寫的 script 供參考：

建立一個 tmp.txt 內容如下：

```
----- tmp.txt -----
server ns.example.com
update delete mail.example.com A
update add mail.example.com 0 A SERVER_NEW_IP
send
----- End of File -----
```

建立更新的 script: newip.sh

```
----- newip.sh -----
#!/bin/sh
#
# Written by LeoLiou.
#

# config the update dir.
exe_path=/root/nsupdate

new_IP=`/sbin/ifconfig eth0 | grep 'inet addr' | cut -d: -f 2 \
| cut -d' ' -f1`

/bin/cat $exe_path/tmp.txt | sed s/SERVER_NEW_IP/$new_IP/g \
> $exe_path/update.txt

# update now IP to name server.
/usr/bin/nsupdate -k $exe_path/Kmail.+157+44587.key -v $exe_path/update.txt
----- End of File -----
```

請將您的 key, tmp.txt, newip.sh 放置於同一目錄，並給予 newip.sh 執行權限。請執行一次 newip.sh 確定沒有問題之後，再利用 crontab 執行 newip.sh 即可。

```
0 * * * * /root/nsupdate/newip.sh
```

這樣就完成整個動態 IP 更新機制了。

關於 update-policy，請看 BIND 9 Administrator Reference Manual：

<http://www.isc.org/products/BIND/bind9.html>

dnssec-keygen 與 nsupdate 的使用方法，相信 man page 寫得已經很清楚了。
:-)

--

Leo Liou (leo@i18n.linux.org.tw)

Join i18n project: <http://i18n.linux.org.tw/>

* ~ * ~ * ~ * ~ * ~ * ~ * ~ * ~ * ~ * ~ * ~ * ~ * ~ * ~ * ~ *

補充 by netman on October 30, 2002:

我略為修改了 Leo兄 的 script , 並置於 /root/nsupdate 目錄下面。
我這裡將界面改為 ppp0 , 同時也減少不必要的傳送動作。
(註: 原本的 tmp.txt 我也改名為 nsupdate.scr 了, 請留意。)

內容如下:

```
#!/bin/bash
#
# Written by LeoLiou.
# Modified by netman on 2002/09/26
#

# set variables
if echo $0 | grep '^/' ; then
    w_dir=${0%/*}
else
    w_dir=$PWD/${0%/*}
fi
KEY_FILE=$w_dir/Kmail.+157+44587.key
UPDATE_SCR=$w_dir/nsupdate.scr
UPDATE_DATA=$w_dir/nsupdate.data
HOST_NAME=mail.example.com
NS_SERVER=ns.example.com
IF="ppp0"

# ensure key files
for file in $KEY_FILE ${KEY_FILE%key}private
do
    if [ ! -r $file ]; then
        echo "${basename $0}: ERROR: $file is not readable."
        exit 1
    fi
done

# prepare initial script
test -f $UPDATE_SCR || {
    cat >| $UPDATE_SCR <<-ENDSCR
        server NS_SERVER
        update delete HOST_NAME A
        update add HOST_NAME 0 A NEW_IP
        send
    ENDSER # <----- 請將 ENDSER 前面的空白換成 tab !
    test "$?" = "0" || {
        echo "${basename $0}: ERROR: could not create $UPDATE_SCR."
        exit 2
    }
}

# ensure the server is connectable
host $NS_SERVER $NS_SERVER | grep "$NS_SERVER" &>/dev/null || {
    echo "${basename $0}: ERROR: could not contact nameserver $NS_SERVER."
    exit 3
}

# get current ip
```

```
NEW_IP=$(ifconfig | grep "$IF " -A 1 \
| awk '/inet/ {print $2}' | sed -e 's/.*://')

# do a test then update
host $HOST_NAME $NS_SERVER | grep "$NEW_IP" &>/dev/null || {
    /bin/cat $UPDATE_SCR | sed s/NEW_IP/$NEW_IP/ \
    | sed s/NS_SERVER/$NS_SERVER/ \
    | sed s/HOST_NAME/$HOST_NAME/ \
    >| $UPDATE_DATA
    /usr/bin/nsupdate -k $KEY_FILE -v $UPDATE_DATA || rm -rf $UPDATE_DATA
}

#-- end of script --#
```

上面的 script 可於這裡下載：

<http://study-area.ks.edu.tw/linux/src/newip.sh.tgz>

目前這個 script 只能在 linux client 上執行，
但我想，透過簡單的 cgi 功能，那麼任何的 windows client 也能輕鬆的進行更新，
不過，這部份就需要其他朋友幫忙設定了。
假如您已經成功完成了，歡迎通知 leo 或 netman ，以便和大家作更進一步的分享。

謝謝！
