DNS

About

Here we list several types of DNS records that integrate quite nicely with FreeSWITCH and IP communications.

ENUM

A way to query E.164 numbers to keep calls "on-net".

SRV

These records can give you a pseudo redundant network. Allowing failover (or pseudo load-balancing) to other servers in your network.

ITAD (ISN)

E-Mail style phone numbers that more easily allow cross-domain calls to stay "on-net" without the need for a full keyboard on your telephone (i.e. analog telephones hooked up to analog adapters). A typical number will look similar to 123*543.

Updating Records

There are many possible ways to dynamically update records in running DNS zones. Here we list the ones we know.

Generating A DNS Key

```
dnssec-keygen -a HMAC-MD5 -b 512 -n USER <NAME_OF_KEY_HERE>
```

nsupdate

All of the following commands are input from the nsupdate command line. You should first have a key that you can use for updating a single zone.

ENUM

```
update add 3.2.1.enum.org 60 NAPTR 100 10 "u" "E2U+sip" "!^.*$!sip:123@freeswitch.org!" .
```

If a user dials 123, this will route the call to 123@freeswitch.org provided enum is correctly configured to use your zone.

ISN

```
update add 3.2.1.543.freenum.org 60 NAPTR 100 10 "u" "E2U+sip" "!^.*$!sip:123@freeswitch.org!" .
```

This would route 123*543 to 123@freeswitch.org

NAPTR

for bind find the example.com zone file, you should put this near your _sip._udp records.

```
$ORIGIN example.com.
; NAPTR record for SIP Secure TLS (TCP) example.com
;     priority: 50  weight: 0
;     protocol: "SIPS+D2T"  regex: ""  uri: _sips._tcp.example.com
;     example.com
@           IN      NAPTR   50 0 "s" "SIPS+D2T" "" _sips._tcp
; NAPTR record for SIP TCP example.com
;     priority: 90  weight: 0
;     protocol: "SIP+D2T"  regex: ""  uri: _sip._tcp.example.com
;     example.com
@           IN      NAPTR   90 0 "s" "SIP+D2T" "" _sip._tcp
; NAPTR record for SIP UDP example.com
;     priority: 100  weight: 0
;     protocol: "SIP+D2U"  regex: ""  uri: _sip._udp.example.com
;
@           IN      NAPTR   100 0 "s" "SIP+D2U" "" _sip._udp
```

Under Windows Server 2008, SR2 you can also add them. Take note of 'Regex'.

- SIPS or SIP Secure over TLS is represented by SIPS and the transport protocol is always going to be TCP.
- The "s" denotes SIP and can be used inside a firewall and outside, but if you have a client, like the Cisco EX60 or EX90 terminals, the software reads "s" and sets the connectivity flag in the SIP contact header to "0" or Internal. Setting this to "se" denotes SIP External and set the connectivity flag in the SIP contact header to "1" or External. This is useful if you are using the SIP Provisioning mechanism so that the system can return the proper configuration details.
- Typically it is preferable to use a secure TLS connection, a less secure but more reliable transport of TCP and lastly use an insecure and less reliable transport of UDP. Setting a priority of the transport methods to 50, 90 & 100 respectively then helps to facilitate the proper use of protocols unless you want to attempt UDP or TCP first and then failover to TLS.

test with

```
host -t naptr example.com
```

You should get something like this:

```
example.com has NAPTR record 50 0 "S" "SIPS+D2T" "" _sips._tcp.example.com.
example.com has NAPTR record 90 0 "S" "SIP+D2T" "" _sip._tcp.example.com.
example.com has NAPTR record 100 0 "S" "SIP+DTU" "" _sip._udp.example.com.
```

SRV

```
update add sip.example.com 60 SRV 100 10 5060 sip1.example.com
update add sip.example.com 60 SRV 100 10 5060 sip2.example.com
```

The preceding will allow load-balancing between sip1.example.com and sip2.example.com

```
update add sip.example.com 60 SRV 10 100 5060 sip1.example.com
update add sip.example.com 60 SRV 100 100 5060 sip2.example.com
```

This will keep sip2 on stand-by for use if sip1 fails. This works for User-Agents that properly support SRV.

Explanation of DNS/SRV/NAPTR

This explanation is from Lawrence Conroy and is part of a mailing list thread. I found it helpful to read this explanation, even though it is brief.

> *I have some sympathy, as the term domain is overloaded within FreeSWITCH.*

> *A sip address consists of a userpart and a domain part -- e.g., <sip:user@sipdomain> The sip domain is similar to an email domain -- e.g., <mailto:user@maildomain> With email, you need to do a lookup of the MX record in DNS to find the FQDN of the machine that handles mail for the domain. With SIP (see RFC 3263), you do a lookup on the SRV record (at _sip._udp.<sipdomain>) to find the machine that handles SIP registrations/incalls for the domain. That also gives you the port on which that machine is listening.*

> *(Yup, you can also have a NAPTR record in the domain to tell you where the SRV record is, but many folks don't bother -- for Best Practice, you should, but ...)*

> *There IS a "get out" clause in the SIP specs for RFC 2543 (AKA legacy) support that means most SIP clients will look for the SRV and, if it can't be found (or there's an IP address rather than a DNS -style domain, in which case the SIP client won't bother hunting the SRV), the client will guess that the domain has a machine (i.e. it will look for an A or AAAA record), and also guess it's listening on 5060 (the default port). Email is the same (mail to fred@example.com, and strictly the sender will do a check for a MX and then look for an A record for example.com, and try there).*

> *There IS a wrinkle to this special case: If you have a URL of the form <sip:user@example.com:5055> the client should NOT look for a NAPTR or SRV, but instead go straight to looking for an A or AAAA for the host example.com (i.e., there's a machine called example.com, and it is handling SIP traffic for that domainpart). Basically, if you see a colon in the domainpart, you're looking for a machine -- otherwise you're looking for a NAPTR (and/or a SRV at _sip._udp.<sipdomain>).*

> *However, relying on that default "get out" clause is definitely NOT what you should do for BCP. Using the hostname as the sip domain is a kludge -- the FQDN with A record usually works, but it's not what you want to do.*

> *SO ... get yourself a domain, put a D2U NAPTR at that domain, put a SRV at _sip._udp.<domain>, and you're done. No need for an A record at that domain at all.*

> *(RFC 3263 is not too hard to read, for a change -- it's certainly shorter than RFC 3261, and it even has an ASCII art diagram :).*

See Also

Function_sofia_dig

Enterprise deployment DNS SRV