DNS Concepts for Rage! Business Office Xchange

DNS Records

It is important to understand the difference between A records and SRV records so let's get that out of the way first.

An A record is a typical host name pointer to an IP address. So, sipx.xyzcompany.com might be an A record that points to 127.1.1.43. In this example, sipx is the host name, xyzcompany.com is the domain name and sipx.xyzcompany.com is referred to as the fully qualified domain name (fqdn). We use A records because it's easier to remember computer names than a series of numbers (for most people anyway). Most Asterisk configurations the author has seen run with an A record setup for the PBX.

SRV records (service records) are used by newer Internet protocols to locate a type of service that might be available for a domain. MX records (mail exchange) work similarly to SRV records but are used specifically for mail. SRV records can be used for any number of services. A SIP service record will look like _sip._udp.xyzcompany.com. In this example a SIP device can query DNS for domain xyzcompany.com and find an IP address or host name for SIP services running on the UDP protocol. Similarly, _sip._tcp.xyzcompany.com would return an IP address or host name for SIP services running on the TCP protocol. So, a phone registering to the xyzcompany.com domain would lookup _sip._udp.xyzcompany.com and have the host name sipx.xyzcompany.com returned. Why not just cut to the chase and use the A record? You'll see why in a bit.

sipXecs PBX configurations are typically setup with SRV records (strongly recommended). SRV records are the method utilized to locate the sipXecs PBX in a clustered environment.

Host Records (A Records)

As described above, A Records are simply name pointers to IP addresses. While A record naming is not typically recommended for sipXecs installations, A records can be utilized to locate the pbx. On a clean sipXecs system (before you add any users), under system administration, go into the System menu and select the Domain menu item. Change the domain name to the fully qualified domain name that you would like to use, click 'Apply' and then reboot.

Make sure your new A record is in your internal DNS server pointing to the internal IP address of the sipXecs pbx. Internally DNS zone file for the xyzcompany.com domain would have an A record setup pointing to the IP of the PBX: sipx A 192.168.10.2

Have your ISP (or if you can do it yourself with a self-managed DNS hosting provider) add an A record for your pbx pointing at the external IP address of your firewall/SBC.

Externally, DNS would have an A record setup pointing to the external IP which is mapped to the internal IP of the PBX: sipx A 127.1.1.43

Testing A Record Configuration

Verify that it all works properly by pinging the PBX by name from both inside the firewall and from outside the firewall using the same DNS domain name. For example: ping sipx.xyzcompany.com

Should return something like:
Pinging sipx.xyzcompany.com [192.168.10.2] with 32 bytes of data:
Reply from 192.168.10.2: bytes=32 time=1ms TTL=64
Reply from 192.168.10.2: bytes=32 time=1ms TTL=64
Reply from 192.168.10.2: bytes=32 time=1ms TTL=64

Reply from 192.168.10.2: bytes=32 time=1ms TTL=64
Ping statistics for 192.168.10.2:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:
Minimum = 1ms, Maximum = 1ms, Average = 1ms
From the Internet the results should be similar but the IP address will be the external IP address of the firewall/SBC (assuming of course that you are allowing ICMP traffic).

SRV Records

So, why are SRV records typically used by sipXecs? Well, they allow us to be contacted with the same address used to deliver e-mail and SRV records allow for server redundancy and load balancing.
Take for example the top salesperson for XYZ Company, Joe User. Joe's phone extension is 512 and he has an email address of joe.user@xyzcompany.com. SRV records allow Joe User to be contacted at sip:512@xyzcompany.com and if an alias is put on his user account he can also be called by phone at sip:joe.user@xyzcompany.com. If xyzcompany.com were using A records for their sipXecs installation, Joe's phone number and alias would have be sip:512@sipx.xyzcompany.com and sip:joe.user@sipx.xyzcompany.com respectively (unless of course the addresses are transformed by the SBC, but then we get into the whole problem of inside the network and outside the network).
SRV records also allow the administrator to have redundant and load balanced SBC computers (sipXecs also uses this method for locating PBX's in a clustered/high availability configuration). SRV records hold the following information:

Service: the symbolic name of the desired service.

- Protocol: this is usually either TCP or UDP.
- Domain name: the domain for which this record is valid.
- TTL: standard DNS time to live field.
- Class: standard DNS class field (this is always IN).
- Priority: the priority of the target host.
- Weight: A relative weight for records with the same priority.
- Port: the TCP or UDP port on which the service is to be found.
- Target: the hostname of the machine providing the service.

An example SRV record might look like this in a DNS zone file:
_sip._udp.example.com 86400 IN SRV 0 5 5060 sipx.xyzcompany.com

This SRV record points to a server named sipx.xyzcompany.com listening on UDP port 5060 for SIP protocol connections. The priority given here is 0, and the weight is 5.

The priority field is similar to an MX record's priority value. Clients always use the SRV record with the lowest priority value first, and only fall back to other records if the connection with this record's host fails. Thus a service may have a designated "fallback" server, which will only be used if the primary server fails. Only another SRV record, with a priority field value higher than the primary server's record, is needed.

If a service has multiple SRV records with the same priority value, clients use the weight field to determine which host to use. The weight value is relevant only in relation to other weight values for the service, and only among records with the same priority value.

In the following example, both the priority and weight fields are used to provide a combination of load balancing and backup service.

_sip._udp.example.com 86400 IN SRV 10 60 5060 sipx1.xyzcompany.com.
_sip._udp.example.com 86400 IN SRV 10 20 5060 sipx2.xyzcompany.com.
_sip._udp.example.com 86400 IN SRV 10 20 5060 sipx3.xyzcompany.com.
_sip._udp.example.com 86400 IN SRV 20 0 5060 sipx4.xyzcompany.com.

The first three records share a priority 10, so the weight field's value will be used by clients to determine which host to contact. The sum of all three values is 100, so sipx1.xyzcompany.com will be used 60% of the time. The other two hosts, sipx2 and sipx3, will be used for 20% of requests each. If sipx1.xyzcompany.com is unavailable, these two remaining machines will share the load equally, since they will each be selected 50% of the time.

If all three hosts with priority 10 are unavailable, the record with the next highest priority value will be chosen, which is sipx4.xyzcompany.com. This might be a machine in another physical location, presumably not vulnerable to anything that would cause the first three hosts to become unavailable Let's look at what these records look like on the internal DNS server and then on the external DNS server. We'll just concentrate on a single SBC and not a redundant configuration.

So, internally DNS zone file for the xyzcompany.com domain would have an A record setup pointing to the IP of the PBX: sipx A 192.168.10.2

And then the SRV records would be setup as follows:
_sip._udp.xyzcompany.com 86400 IN SRV 10 100 5060 sipx.xyzcompany.com
_sip._tcp.xyzcompany.com 86400 IN SRV 10 100 5060 sipx.xyzcompany.com

Externally, DNS would have an A record setup pointing to the external IP of the PBX (which would have a 1:1 NAT mapping to the internal IP address of the PBX and be firewalled appropriately):
sipx A 127.1.1.43

And then the SRV records would be setup exactly as they were internally (note, if you are using OpenSBC it only needs the UDP record):

_sip._udp.xyzcompany.com 86400 IN SRV 10 100 5060 sipx.xyzcompany.com
_sip._tcp.xyzcompany.com 86400 IN SRV 10 100 5060 sipx.xyzcompany.com

Testing SRV Configuration

The dig command in Linux lets us test DNS.

At the command line enter: dig –t SRV _sip._udp.xyzcompany.com
The following will be displayed:
; <<>> DiG 9.3.4-P1 <<>> -t SRV _sip._udp.xyzcompany.com
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 48387
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;_sip._udp.xyzcompany.com. IN SRV
;; ANSWER SECTION:
_sip._udp.xyzcompany.com. 0 IN SRV 0 0 5060 sipx.xyzcompany.com
;; Query time: 3 msec
;; SERVER: 172.16.1.254#53(172.16.1.254)
;; WHEN: Sat Jan 3 15:46:12 2009
;; MSG SIZE rcvd: 81

From the ANSWER: 1 in the returned information and the sipx.xyzcompany.com in the ANSWER SECTION we know that the SRV record is working properly.

From a Windows workstation the nslookup command can be used:
At the command prompt enter: nslookup
Default Server: UnKnown
Address: 172.16.1.254
> set type=srv (enter this)
> _sip._udp.xyzcompany.com (enter the SRV record to test)
Server: UnKnown
Address: 172.16.1.254
_sip._udp.xyzcompany.com SRV service location: (expected results)
priority = 0
weight = 0
port = 5060
svr hostname = sipx.xyzcompany.com

Scenario 1 – sipXecs PBX on the Data Network

Ok, so most companies aren't hiding under rocks and they are already using DNS internally on their networks. Like many, the internal network DNS domain name may not be the same as their Internet facing domain. Consider the following network scenario:

Internet Connection (DSL/Cable Modem/T1) Firewall with NAT Internal Network 192.168.10.x/24 sipx PBX 192.168.10.2/24 IP Phones Laptop w/ Softphone IP Phones 192.168.10.1/24 127.1.1.43/24

Outside world: Company Domain = xyzcompany.com Web Ste = www.xyzcompany.com eMail = *@xyzcompany.com DNS Hosted at ISP

Inside world: Internal Network Domain = xyzcompany.corp File server = server.xyzcompany.corp DNS Server = server.xyzcopany.corp File Server 192.168.10.10/24 server.xyzcompany.corp

With our fictitious company (sorry if it's not), the internal computer network has an existing DNS domain of xyzcompany.corp which is being used by Active Directory. DHCP and DNS are being handled by the file server at 192.168.10.10. The simplest approach to integrating sipXecs into this network is to leave DNS and DHCP on the file server and add the required SRV records and DHCP options to it.

There are a few problems with this scenario:
1. The operation of your phone system is dependent on DNS working. If DNS or your File Server has problems, your phone system is also down.
2. Mobile users that need to roam to different networks outside of the organization will not be able to resolve the xyzcompany.corp domain from the internet.
3. Users outside of your organization can not call your system via a SIP URL (sip:ext@xyzcompany.com).

Problems 2 and 3 can be worked around, but problem 1 is the bigger issue. Making sure that you have a robust DNS configuration is important to your computer network and your phone network.

Important Tip: In this scenario DNS and DHCP should be configured properly BEFORE installing sipXecs.

Scenario 1 - Configure DNS

There are four DNS records that need to be configured for sipXecs to function properly. They are:

- A Record for sipXecs PBX host name

- SRV Record for SIP UDP Signaling Traffic
- SRV Record for SIP TCP Signaling Traffic
- SRV Record for sipXecs PBX Resource locating (Resource Record)

Add a new host record for the PBX (in the above example it would be something like sipx.xyzcompany.corp) pointing to the IP address of the PBX.

Add SRV records for _sip._udp.xyzcompany.corp, _sip._tcp.xyzcompany.corp and _sip._tcp.rr.sipx.xyzcompany.corp all pointing to sipx.xyzcompany.corp. (if you are doing this in a Microsoft Windows environment, see document mentioned in beginning of this whitepaper).

In a bind (Linux DNS) configuration file these records would look something like this:
_sip._tcp.xyzcompany.corp. IN SRV 1 0 5060 sipx.xyzcompany.corp.
_sip._udp.xyzcompany.corp. IN SRV 1 0 5060 sipx.xyzcompany.corp.
_sip._tcp.rr.sipx.xyzcompany.corp. IN SRV 1 0 5070 sipx.xyzcompany.corp.
sipx.xyzcompany.corp. IN A 192.168.10.2

Scenario 1 - Configure DHCP

DHCP is used by phones and PC's alike in this scenario to get IP addresses as well as other information needed about the network to operate properly.

In addition to an IP address, to operate properly phones need a Default Gateway (DHCP option 3), DNS Domain Name (DHCP option 15), DNS Server IP address (DHCP option 6) and a TFTP server address (DHCP option 66). If you are using a different provisioning method you're on your own here.

A typical DHCP configuration file from a Linux system would look like:
subnet 192.168.10.0 netmask 255.255.255.0 {
range 192.168.10.20 172.168.10.254; #IP Range
default-lease-time 21600;
max-lease-time 43200;
option routers 192.168.10.1; # Default gateway
option subnet-mask 255.255.255.0; # Subnet mask
option domain-name "xyzcompany.corp"; #DNS Domain Name
option domain-name-servers 192.168.10.10; #DNS Server IP
option time-offset -18000; # Eastern Standard Time
option tftp-server-name "sipx.xyzcompany.corp"; #phone provisioning
option ntp-servers 192.168.10.10; #get time from file server

Scenario 1 - Testing

It is important to test your configuration and verify that it is operating as it should before you install your sipXecs system! Refer to the testing section above.
Boot a computer and make sure it receives an IP address just as a phone would. Check that it is receiving the proper DNS domain name (on a Windows machine you can use 'ipconfig /all' from the command prompt to verify this).
Make sure the computer can Ping the PBX by its host name and us NSLOOKUP to verify that the SRV records are working as they should.

Once DNS and DHCP are working properly you are ready to install your sipXecs system.

Scenario 1 - Remote Users

Assuming now that your sipXecs system is up and running correctly, the next challenge with DNS is configuring it so that remote users can connect to your PBX. If those users are connecting via a VPN tunnel or wide area network, simply configure DNS on the far end to have the same DNS records and DHCP records we setup above.
If users are connecting from the Internet they are never going to be able to resolve xyzcompany.corp because it isn't a valid DNS domain name. Your organization more than likely has something like xyzcompany.com that is registered with an organization like Network Solutions and hosted either there or at another DNS hosting provider.
The following host record and SRV records must be configured at your DNS hosting provider:
_sip._tcp.xyzcompany.com. IN SRV 1 0 5060 sipx.xyzcompany.com.
_sip._udp.xyzcompany.com. IN SRV 1 0 5060 sipx.xyzcompany.com.
sipx.xyzcompany.com. IN A 127.1.1.43 # Change to outside IP of your FW

Configure your firewall to allow and NAT ports 5060 udp, 5060 tcp and ports 30000 – 31000 udp from 127.1.1.43 (again, this would be changed to YOUR external IP address) to 192.168.10.2 (this would be changed to the internal IP address of YOUR PBX).

Just as you tested your internal DNS, make sure you test the external DNS from outside your network.
One last step needs to be completed. sipXecs allows for domain name aliases (System Menu -> Domains). Add a domain alias for xyzcompany.com. An alias allows the sipXecs system to respond to requests made to domains other than the domain it is setup with. It thus should be possible to setup your mobile users differently than your fixed position hard phone users and do a translation of xyzcompany.com to xyzcompany.corp.

Scenario 1 - Dynamic DNS

Wondering what about the case where the external IP address may change like with a Cable Modem or DSL connection? Usually the only way you will be able to deal with SRV records is by owning your own domain. Drop $20 a year with a hosting provider like GoDaddy.com or similar (just make sure they let you have SRV records) and get yourself a domain name.

Once you have an domain name, get setup with DynDNS or one of the other dynamic DNS providers (I use DynDNS because it works with Vyatta firewall). If you don't have a firewall that does dynamic DNS updates, you can usually run software on an internal machine that helps the dynamic DNS provider figure out your external IP address.
The dynamic DNS provider will let you determine your own host name and tag it to one of their domain names. For instance, xyzpbx.dyndns.net might be a host name you could specify. We can then point to this dynamic DNS name from our own domain.

For pointing a host name at another host name we'll use a CNAME record (canonical name). So, externally the DNS would have a CNAME record setup pointing to the dynamic DNS name:
sipx CNAME xyzpbx.dyndns.net

And then the SRV records would be setup also pointing to the dynamic DNS name as follows:
_sip._udp.xyzcompany.com 86400 IN SRV 10 100 5060 xyzpbx.dyndns.net
_sip._tcp.xyzcompany.com 86400 IN SRV 10 100 5060 xyzpbx.dyndns.net