Intro to djbdns

A DNS server besides BIND

Nathan Straz

nate@techie.com

Today's Topics

- Overview of DNS
- Overview of djbdns
- djbdns by example
  - local cache
  - network cache
  - private domain
- Comparison to BIND

What is DNS?

- Domain Name System
- System used to bind names to IP addresses
- Provides additional information about domains such as mail servers and name servers
- Documented in RFC1034 and RFC1035
- Circa November 1987
- A distributed database

DNS Records

SOA Start of Authority, defines the primary name server and contact for a domain.

A Address, binds a name to an IP address

PTR Pointer, used to look up the name of an IP address

NS Name Server, defines the IP address and name of a name server for a domain

MX Mail Exchanger, defines the IP address and name of a mail server responsible for a domain

- Each record has a Time To Live which dictates how long a record can be cached.

DNS Organization

root The top of the DNS food chain. Delegates authority of TLDs to other name servers.

tld Top Level Domains. Delegates authority of domain names to other name servers.

domains Company and institution level addresses. *real-time.com*, *mn-linux.org*, *etc*

cache A name server without any authority. It looks up records for clients and caches records. It's placed on a network to speed up DNS queries by reducing the number of queries the need to go outside of the network.

How a DNS query works

- What is the address of `network-surveys.cr.yp.to`?
- Does this server know anything about this name? No, ask a root name server.
- A root name server looks at the address and says, "All I know about this name is that the name server for `to` is 198.6.1.82."
- So we ask 198.6.1.82 and get, "All I know about this name is that the name server for `yp.to` is 131.193.178.181."
- So we ask 131.193.178.181 and get, "I know `network-surveys.cr.yp.to` and the address is 131.193.178.100."

What is djbdns?

- An alternative to BIND
- Written by Dan Bernstein, author of Qmail
- A collection of servers that handle different aspects of DNS
- Handles DNS caching, authorative servers, zone transfers, and open relay lists.
- Homepage: http://cr.yp.to/djbdns.html
- Useful site: http://djbdns.org/

Why use djbdns?

- There is a $500 reward for the first person to publish a security hole in djbdns. All servers run as a non-root user in a chroot jail.
- The servers are much easier to configure than BIND.
- Less than 7,000 instructions in djbdns, compared to nearly 100,000 instructions in BIND 9.
- Uses a fixed about of memory for the cache which is easily tuned.
- No waiting for the DNS database to load before queries can be answered.

daemontools

- daemon management tools written by DJB
- restarts daemons if they die or disappear
- uses the file system as the configuration database
- They don’t follow any established standards
- a symlink in `/service` is used to register services
- `svc` is used to start and stop services

Pieces of djbdns

dnscache A local DNS cache. It handles recursive queries from local clients like web browsers, mail servers, ftp servers, etc. It collects responses from remote DNS servers. It caches the responses to save time later. It can be configured as a local or external cache.

tinydns A DNS server. It does not handle recursive queries. It only ansswers queries using local data.

rbldns A specialized DNS server for handling lists of IP addresses. It's a more efficient server for lists like RBL or ORBS.

Pieces of djbdns (cont.)

walldns A reverse DNS wall. It's used for giving out generic responses to reverse DNS queries. This makes sure IP addresses get names while not revealing private host names.

axfrdns, axfr-get A zone transfer server and client to handle DNS-over-TCP requests.

Initial Installation

- Install daemontools with your favorite packaging system
- Install djbdns with your favorite packaging system
- Create users for use by the daemons

  dnslog User to handle the DNS server logs

  dnscache User to run dnscache as

  tinydns User to run tinydns as

  axfrdns User to run axfrdns as
- Start `svscan`

Creating servers

`dnscache-conf` ***`acct logacct D ip`***

`acct` Account to run dnscache as, usually dnscache

`logacct` Account used to handle the logs

`D` The root directory of this server instance

`ip` The IP address that dnscache will listen on

- `ln -s` *`D /service`*
- *`D/root/ip`* tells dnscache which addresses to accept queries from
- *`D/root/servers`* tells dnscache which name servers to look at for specific domains.

dnscache config options

These files are in *D/env*

IP IP address that dnscache listens to

IPSEND IP address that dnscache sends packets from. Default is 0.0.0.0, which is an alias for the machine's primary IP.

FORWARDONLY Treats servers/@ as a list of other caches and forwards all requests to those servers.

CACHESIZE The amount of memory dnscache will use for caching. Default 1 megabyte.

ROOT The directory dnscache will use as its chroot jail.

Example: local cache

A caching name server on the loopback device.

1. `dnscache-conf dnscache dnslog /var/dnscache/local 127.0.0.1`
2. `ln -s /var/dnscache/local /service/`

Example: external cache

A caching name server on a network address.

1. `dnscache-conf dnscache dnslog /var/dnscachex 192.168.42.2`
2. `ln -s /var/dnscachex /service/`
3. `touch /var/dnscachex/root/ip/192.168.42`

Example: use local DNS servers

Tell your caching name server to consult internal DNS servers for certain names.

- For addresses ending in .straz, ask 192.168.42.2
- For reverse lookups in 192.168.42.0/24, ask 192.168.42.2

1. `echo "192.168.42.2" > D/root/servers/straz`
2. `echo "192.168.42.2" > D/root/servers/42.168.192.in-addr.arpa`

Example: private DNS

Create your own private TLD for home

- Create DNS server for straz TLD on 192.168.42.2
- Add host candle.straz as 192.168.42.2
- Add host marvin.straz as 192.168.42.32

Example: private DNS (cont.)

1. `tinydns-conf tinydns dnslog /var/tinydns 127.0.0.1`
2. `ln -s /var/tinydns /service/`
3. `cd /var/tinydns/root`
4. `./add-ns straz 192.168.42.2`
5. `./add-host candle.straz 192.168.42.2`
6. `./add-host marvin.straz 192.168.42.32`
7. `make`

Example: Change the cache size

Change the cache size to 5MB

1. `svc -d /service/dnscache`
2. `echo "5000000" > /service/dnscache/env/CACHESIZE`
3. `echo "15000000" > /service/dnscache/env/DATALIMIT`
4. `svc -u /service/dnscache`

That’s all folks

Questions?