Configuring BIND

Session 9261

SHARE 102

Long Beach, CA

PATRICIA EGEN CONSULTING

Abstract

If the Domain Name Server is the glue which holds the internet together, then the Berkeley Internet Name Domain (BIND) server is the brand of glue used by the majority of its users. Join us and find out how to configure your BIND server. We'll look into the types of records, and when to use them as well as a quick look into DNSSEC.

PATRICIA EGEN CONSULTING

The Speaker

Harold Pritchett

Patricia Egen Consulting

(706) 546-0692

harold@uga.edu

PATRICIA EGEN CONSULTING

Disclaimer

Everybody has lawyers:

The ideas and concepts set forth in this presentation are solely those of the respective authors, and not of the companies and or vendors referenced within and these organizations do not endorse, guarantee, or otherwise certify any such ideas or concepts in application or usage. This material should be verified for applicability and correctness in each user environment. No warranty of any kind available.

PATRICIA EGEN CONSULTING

Presentation Protocol

- Ask Questions for Understanding
  - For clarification on the current topic:
  - STICK YOUR HAND UP NOW -
- Save Questions on related issues
  - Hold for Q&A at end of session
- The only dumb question is:
  - the one you didn't ask

PATRICIA EGEN CONSULTING

BIND

- *Berkeley Internet Name Domain* (BIND) software. BIND is a client/server software system. The Client side of BIND is called the *resolver*. It generates the queries for domain name information that are sent to the server. The DNS server software answers the resolver's queries. The server side of BIND is a daemon.
- It is called 'named' (pronounced "name" "d")

PATRICIA EGEN CONSULTING

History of BIND

- The architect of the Domain Name System was Paul Mockapetris of USC's Information Sciences Institute in 1983.
- Paul then proceeded to write the first implementation of this architecture which he called "jeeves"
- Jeeves was implemented in July, 1984 on a DEC PDP-10 running TOPS-20
- Jeeves would continue to run as the "root" name servers until approximately 1988

PATRICIA EGEN CONSULTING

History of BIND (Cont.)

- BIND was written at the University of California at Berkeley for the 4.3 BSD Unix operating system.
- BIND version 4 was released in April, 1985
- Versions of BIND through 4.8.3 were maintained by Berkeley
- Versions 4.9 and 4.9.1 were released by Digital Equipment Company. Paul Vixie, a Digital employee became BIND's maintainer

PATRICIA EGEN CONSULTING

History of BIND (Cont.)

- BIND 4.9.2 was released by Vixie Enterprises. Paul Vixie became BIND's principal architect/programmer
- BIND 4.9.3 and all later releases were developed and maintained by the Internet Software Consortium (www.isc.org). Paul Vixie remained as the BIND guru.
- BIND was feature frozen at version 4.9.5 in 1995. Only Security and Bug fixes released

History of BIND (Cont.)

- BIND 8 – Initially released in May, 1997
- BIND 8 was a major rewrite of the BIND 4 code, but shared the same code base.
- BIND 8 had a new configuration file, named.conf
- BIND 8 added an impresive list of new features to BIND

New Features in BIND 8.1

- DNS Dynamic Updates (RFC 2136)
- DNS Change Notification (RFC 1996)
- Completely new configuration syntax
- Flexible, categorized logging system
- IP-address-based access control for queries, zone transfers, and updates that may be specified on a zone-by-zone basis
- More efficient zone transfers
- Improved performance for servers with thousands of zones
- The server no longer forks for outbound zone transfers
- Many bug fixes

PATRICIA EGEN CONSULTING

History of BIND (Cont.)

- BIND 9 - Initially released in September, 2000
- A complete, from scratch, rewrite of the BIND program
- Why?
- One reason was to “clean up” the BIND code.
- Paul Vixie was busy maintaining BIND 8 and took no part in the creation of BIND 9.

PATRICIA EGEN CONSULTING

Some Features of BIND Version 9

- DNS Security
  - DNSSEC (signed zones)
  - TSIG (signed DNS requests)
- IP version 6
  - Answers DNS queries on IPv6 sockets
  - IPv6 resource records (A6, DNAME, etc.)
  - Bitstring Labels
  - Experimental IPv6 Resolver Library
- DNS Protocol Enhancements
  - IXFR, DDNS, Notify, EDNS0
  - Improved standards conformance
- Views
  - One server process can provide multiple "views" of the DNS namespace, e.g. an "inside" view to certain clients, and an "outside" view to others.
- Multiprocessor Support
- Improved Portability Architecture

PATRICIA EGEN CONSULTING

Disclaimer found in BIND 4 release

The official version of ISC BIND is now 9.1.0, or failing that, 8.2.3.

This is ISC BIND 4.9.11, hoped to be the last of 4.*, which we are releasing since it has an important security bug fixed. Other less important security bugs in BIND4 remain *unfixed*. You should not be running it. You have been warned.

PATRICIA EGEN CONSULTING

Network access to BIND

- The BIND server is accessed via the network on port 53.
- Both TCP and UPD are used.
  - Queries are made via UDP
  - Responses are made via UDP unless the response is to large to fit in a single packet
  - If the response won't fit in a single UDP packet, then the response is returned via TCP
  - MS Exchange has been reported to use TCP for queries

PATRICIA EGEN CONSULTING

A little review

- Delegation
- Zones vs Domains
- Types of Name Servers
- Zone Transfers

Delegation

- Administrators can create subdomains to group hosts:
  - According to geography, organizational affiliation or any other criterion
- An administrator of a domain can delegate responsibility for managing a subdomain to someone else
- The parent domain retains links to the delegated subdomain:
  - The parent domain “remembers” who it delegated the subdomain to

PATRICIA EGEN CONSULTING

Delegation creates zones

- Each time an administrator delegates a subdomain to someone else, a new unit of administration is created
  - The subdomain and its parent domain can now be administered independently
  - These units are called **zones**
  - The boundary between zones is a point of delegation in the name space

PATRICIA EGEN CONSULTING

What's in a zone?

- Like a domain, a zone is named after its apex node
- Unlike a domain, a zone contains only descendants of the zone's apex node that have not been delegated
  - Nodes below a delegation point are in another zone

Zones vs Domains

- A zone (of a given name) contains the same nodes as a domain (of the same name), minus those nodes that are delegated away to other zones
- For example, the zone uga.edu contains the same nodes as the domain uga.edu, minus those nodes in zones beneath uga.edu
- When do a zone and a domain (with the same name) contain the same nodes?

Name Servers

- Name servers store information about the name space in units of zones
  - The name servers that load a complete zone are said to “have authority for” or “be authoritative for” the zone
- Usually, more than one name server is authoritative for the same zone
  - This ensures redundancy and spreads the load
- Also, a single name server may be authoritative for many zones

PATRICIA EGEN CONSULTING

Name Servers (Cont)

- A name server which has been delegated authority for a zone but does not have data for that zone is referred to as a

Lame Server

- This is a very bad thing, and occurs much too often

PATRICIA EGEN CONSULTING

Types of Name Servers

- The **master** name server for a zone loads the zone's data from a file on disk
- A **slave** name server for a zone loads the zone's data from another authoritative name server (often the primary master)
  - An older term for slave was "secondary master"
  - The server the slave gets its zone data from is called its *master* server
- A Caching only name server has no local data files

PATRICIA EGEN CONSULTING

Types of Name Servers

- A single name server can be the master for some zones and a slave for other zones
  - The relationship is defined zone-by-zone
  - So, strictly speaking, you shouldn't refer to a computer as "the master name server" unless you also specify which zone you're talking about

Zone Transfers

- Slave servers retrieve zone data from other authoritative name servers using a zone transfer
- The zone transfer is initiated by the slave
  - By initiating a TCP connection to the master name server
- The master server may notify the slave server that new zone data is available

Defining the Zone Data

PATRICIA EGEN CONSULTING

Zone files

- The files which contain the data being served by the DNS system are called “Zone Files”
- They are made up of a series of “Resource Records”
- A Zone File will always contain an SOA record as well as additional records

Resource Records

Resource records have as many as five fields, most of which are optional:

- Owner: the domain name of the node to which the record is attached
- Time to live (TTL): more on this later
- Class: the kind of network this record describes
  - Only class we need is the internet class “in”
- Type: the function of this record
- RDATA: record-specific data
  - The RDATA can be further subdivided into type-specific fields

PATRICIA EGEN CONSULTING

Types of Resource Records

- SOA - start of authority, for a given zone
- NS - name server
- A - name-to-address mapping
- PTR - address-to-name mapping
- CNAME - canonical name (for aliases)
- MX – mail exchanger (host to receive mail for this name)
- TXT - textual info
- RP - contact person for this zone
- WKS - well known services
- HINFO - host information
- Comments start with ; continue to end of line

PATRICIA EGEN CONSULTING

SOA Record

- A start of authority (type SOA) record specifies zone-specific values
- Each zone has one SOA record
- The owner is the domain name of the zone
- The RDATA is, to say the least, complicated

```
halshome.net. IN SOA  rottweiler.halshome.net. harold.halshome.net. (
                      2008071301 ; serial
                      3h ; refresh
                      1h ; retry
                      1w ; expire
                      10m ) ; negative caching TTL
```

PATRICIA EGEN CONSULTING

SOA fields

- The fields in the SOA RDATA are, in order:
  - The MNAME field, by convention the domain name of the master name server,
  - The email address of the technical contact for the zone, with a dot replacing the “@”,
  - The zone’s “serial number”
  - The zone’s “refresh interval”
  - The zone’s “retry interval”
  - The zone’s “expiration interval”
  - The “negative caching time to live (TTL)” for records in the zone

PATRICIA EGEN CONSULTING

SOA fields

- All of the times default to seconds
  - This was all that was allowed in BIND 4
  - In BIND 8 and BIND 9, times may be suffixed with a unit
    - s for seconds (259200 seconds)
    - m for minutes (4320 minutes)
    - h for hours (72 hours)
    - d for days (3 days)

NS records

- A name server (type “NS”) record lists an authoritative name server for a zone
- The owner is the domain name of the zone
- The RDATA is a single domain name (not an IP address) of a name server authoritative for the zone

halshome.net. IN NS ns1.granitecanyon.com.

halshome.net. IN NS ns2.granitecanyon.com.

PATRICIA EGEN CONSULTING

A records

- An address (type “A”) record specifies an IP address of a host
  - More generally, it specifies the IP address of a domain name
- The owner is the domain name of the host
- The RDATA is the dotted-octet format of a single IP address
- Multihomed hosts and routers can have multiple A records, one for each network interface

www.halshome.net. IN A 192.168.1.3

PATRICIA EGEN CONSULTING

PTR records

- A PTR or “Pointer Record” is the “reverse lookup” record for a host
- The owner field of the record contains the IP address of the host in reverse order with the domain name “in-addr.arpa” appended to it
- The RDATA of the record is the fully qualified Domain Name (FQDN) of the host

3.1.168.192.in-addr.arpa. IN PTR www.halshome.net.

PATRICIA EGEN CONSULTING

CNAME records

- A CNAME or “Canonical Name” record is an Internet Alias. The name in the owner field is equated as an alias to the FQDN found in the RDATA field.
- A name which appears in the owner field of a CNAME record can not appear in the owner field of any other record.

www.halshome.net. IN CNAME halshome.net.

PATRICIA EGEN CONSULTING

MX records

- A MX or “Mail Exchanger” record is used to assign e-mail delivery information for a host. The mail address in the owner field is assigned to the host in the RDATA field.
- The RDATA field contains two sub-fields, an assignment priority and host name. Multiple records are tried in priority order, lowest first

```
halshome.net. in mx 0 mail1.halshome.net.
halshome.net. in mx 1 mail2.halshome.net.
```

PATRICIA EGEN CONSULTING

TXT records

- Text (TXT) records contain data which is associated with the name in the owner field.
- The RDATA field consists of multiple strings, enclosed in quotes (“)

dawg IN TXT “Location:” “Room 608”

PATRICIA EGEN CONSULTING

RP records

- The Responsible Person (RP) record can be used to provide contact information about the name in the owner field
- The RDATA consists of two fields
  - The e-mail address of the person, in DNS format (@ replaced with .)
  - The name of a TXT record to be associated with this person

PATRICIA EGEN CONSULTING

RP and TXT records

- It looks something like this

```
dawg   IN RP   harold.dawg.halshome.net hp
hp     IN TXT “Harold Pritchett – 706-546-0692”
```

PATRICIA EGEN CONSULTING

HINFO and WKS records

- These records were created to provide information which previously occurred in the old arpanet HOSTS.TXT file. They are rarely used today.
- Most people don’t want to provide details about the hardware and software they are running

Zone file conventions

- The domain specified in the zone files is known as the origin
- It is initially set to the value from the zone statement in named.conf
- It can be represented by the @ symbol
- Origin will be appended to all hostnames that do not end with a dot

PATRICIA EGEN CONSULTING

Zone file conventions

- Comments in zone files start with a semi-colon and end at the end of line
- The parentheses “()” are used to group data that crosses a line boundary. In effect, line terminations are not recognized within parentheses.

Zone name

- In configuration file “named.conf”:

```
zone “halshome.net" in {
type master;
file “halshome.db”; };
```

- In Zone File “halshome.db”:

```
@       IN SOA rottweiler harold (
                    2000071200 ; serial
                    3h ; refresh
                    1h ; retry
                    1w ; expire
                    10m ) ; negative caching TTL
```

PATRICIA EGEN CONSULTING

Zone file conventions

- Repeat last name
  - If a resource record starts with a space or tab it assumes the same name as the previous resource record

Zone files

- Created by the domain administrator for each zone for which this server is authoritative
- There are two zone files created for each zone
  - One for the forward lookup (lookup by hostname – contains “A” records)
  - One for the reverse lookup (lookup by IP address – contains “PTR” records)

Examples of zone files

PATRICIA EGEN CONSULTING

localhost file

```
$ORIGIN    localhost.
$TTL  86400
@  IN  SOA localhost. root.localhost. (
                    1          ; serial
                 1800          ; refresh
                  900          ; retry
                86400          ; expire
                 1200)         ; negative cache ttl

          NS    localhost.
          A     127.0.0.1
```

PATRICIA EGEN CONSULTING

localhost.rev file

```
$ORIGIN      0.0.127.in-addr.arpa.
$TTL   86400
@      IN    SOA localhost. root.localhost. (
                    1  ; serial
                  30m  ; refresh
                  15m  ; retry
                   1d  ; expire
                  20m) ; negative cache ttl

       NS           localhost.
1      PTR          localhost.
```

PATRICIA EGEN CONSULTING

halshome file

```
$TTL    86400
@       IN              SOA     mickey harold.mickey (
                          2004022201 ; serial
                                1800  ; refresh
                                 900  ; retry
                               69120  ; expire
                                1080) ; negative cache ttl

                        NS  ns1.granitecanyon.com.
                        NS  ns2.granitecanyon.com.
                        A   192.168.1.1
rottweiler              A   192.168.1.1
mickey                  A   192.168.1.3
stitch                  A   192.168.1.6
jr                      A   192.168.1.11
```

PATRICIA EGEN CONSULTING

Halshome.rev file

```
$TTL  86400
@ IN   SOA     mickey.halshome.net. harold.mickey.halshome.net. (
                2004022201 ; serial
                      1800  ; refresh
                       900  ; retry
                     69120  ; expire
                      1080) ; negative cache ttl

1                     ptr   rottweiler.halshome.net.
3                     ptr   mickey.halshome.net.
6                     ptr   stitch.halshome.net.
11                    ptr   jr.halshome.net.
```

PATRICIA EGEN CONSULTING

Configuring your Name Server

PATRICIA EGEN CONSULTING

Configuring DNS Name Service

- Configuring the BIND resolver

- Configuring the BIND name server (named)
  - The name server configuration file (named.conf)
  - The name server database files, (called the *zone files*)

BIND Configurations

- There are four levels of service which can be defined in a BIND configuration
  - Resolver-only
  - Caching-only servers
  - Master servers
  - Slave servers

Resolver configuration

- The resolver is the code that makes requests to name servers for domain information
- On UNIX systems, it is implemented as a library (libresolv.a, libresolv.so)
- Single configuration file

  /etc/resolv.conf

Configuring the Resolver

- The resolver is configured in the /etc/resolv.conf file
  - It allows you to identify up to three name servers, the default domain name, and various other processing options

Resolver statements

- nameserver address
- domain name
- search domain ...
- sortlist network ...
- options option ...

Example of /etc/resolv.conf

*Resolver configuration file

domain halshome.net

nameserver 192.168.1.3

nameserver 128.192.1.9

nameserver 128.192.1.193

PATRICIA EGEN CONSULTING

Configuring named

- Several files are used to configure named
- Only one is hard coded

  /etc/named.conf
- All other files are defined in this one
- File names are completely arbitrary
- The name named.conf can be changed on the command line when named is started

  named -c /my/config/file

Configuration Files

- named.boot
- named.conf
- root.hints
- localhost
- localhost.rev
- Other zone files

- Remember, File names are arbitrary

named.boot

- If you find this file, you have a real problem
- named.boot is the configuration file for BIND 4 which is completely deprecated
- You MUST de-install BIND 4 and install either BIND 8 or BIND 9!
- This is NOT optional!

PATRICIA EGEN CONSULTING

named.conf

- Comments: /* */ , // , #
- Each statement begins with a keyword
- An address match list can include:
  IP/IP with netmask/acl name/key/!
  e.g. { ! 1.2.3.13; 1.2.3.24; };
  { 140.113/16; 127.0.0.1; };
- Uses a “first match” algorithm
- All statements end with a semicolon (;)

PATRICIA EGEN CONSULTING

Some statement types

include	inserts an external file
options	Sets global name server configuration
acl	Defines access control lists
logging	Specifies logging categories and destinations
zone	Define a zone of resource records

PATRICIA EGEN CONSULTING

The include statement

- include “path”;

- Put different portion of the configuration in separate files
- The path is relative...
  - To the directory listed in the “options” statement
- Protect cryptographic keys by making included files not world-readable

PATRICIA EGEN CONSULTING

The options statement

- options {
  option;
  option;
  ...
  };
- BIND 8 had 30
- BIND 9 has over 50

PATRICIA EGEN CONSULTING

The options statement

- version “string”; [actual version of server]
- directory “path”; [where server started]
- notify yes | no; [yes]
- also-notify svrs_ips; [empty]
- recursion yes | no; [yes]
- allow-recursion { add_list }; [all hosts]
- Many others - see references

PATRICIA EGEN CONSULTING

The acl statement

- Acl acl_name {
  address_match_list
  };
- Must be defined before it is used (one pass processing of the config file)
- Predefined lists:
  any, localnets, localhost, none

The logging statement

- Logging {
  channel_def;
  channel_def;
  ...
  category category_name {
    channel_name;
    channel_name;
    ...
  };

The zone statement

```
zone “domain_name” {
        type master|slave|stub|hint|forward;
        file “path”;
        allow-query {address_match_list; };
        allow-transfer {address_match_list; };
        allow-update {address_match_list; };
};
```

PATRICIA EGEN CONSULTING

root.hints

- This file is used to find the root name servers.
- The current version is always available for ftp from ftp.rs.internic.net in the /domain directory. It is named named.root
- Rename it to root.hints or to whatever name you call it in your named.conf file
- It only has to be reasonably current

PATRICIA EGEN CONSULTING

Configuring a caching-only server

- Why?
- Resolvers are dumb
- They don’t remember anything!
  - Well, the WinDoze one does, but that’s a different problem
- If we could remember previously requested addresses we could speed up internet access

PATRICIA EGEN CONSULTING

Configuring a caching-only server

- A caching-only name server
  - Is authoritative for no domain or zone
  - Provides DNS services for users, and remembers the results.
  - Only configuration files are
    - named.conf
    - localhost
    - localhost.rev
    - root.hints

PATRICIA EGEN CONSULTING

Configuring a caching-only server

```
# named.conf for caching-only server
options { directory "/var/named"; };
zone “localhost" {
    type master;
    file “localhost"; };
zone "0.0.127.IN-ADDR.ARPA" {
    type master;
    file “localhost.rev"; };
 zone "." {
    type hint;
    file “root.hints"; };
```

PATRICIA EGEN CONSULTING

Master and Slave servers

- Both are authoritative for the zones they serve. This means that they have the data for the zones locally and don't have to query other servers for the data.
- Master servers actually contain zone data in local files
- Slave servers get their data from the master server

SOA Fields and Zone Transfers

- Most of the SOA fields are related to zone transfers
- A slave server for a zone checks with its master server once during each refresh interval to see if the zone's serial number has gone *up* (not just changed)
  - If the master has a higher serial number than the slave for the zone, the slave transfers the zone
  - If the serial number is the same, the slave resets its refresh timer
  - If the serial number on the master is *lower,* the slave complains

PATRICIA EGEN CONSULTING

SOA Fields and Zone Transfers

- If the slave's check of the master's serial number fails, it tries again within the retry interval until it gets the serial number
- If the slave still can't get the serial number within the expire interval, it stops giving out answers about the zone
  - Queries for data in the zone return an error (SERVFAIL)

Notify

- Allows the master name server for a zone to notify the slave servers of changes to the zone by sending them specially formatted messages
- Upon receipt of these messages, NOTIFY-capable slaves
  - Verify that the message came from one of their master servers
  - Immediately check the zone's SOA record on their configured master server(s)

PATRICIA EGEN CONSULTING

DNS Security

PATRICIA EGEN CONSULTING

TSIG

- Transaction Signatures
- Introduced in BIND 8.2
- Codified in RFC 2845
- Uses shared secret keys to sign transactions
- Uses a special MD5 hash which includes a secret key (HMAC-MD5)
- Main problem is the shared secret keys

PATRICIA EGEN CONSULTING

DNSSEC

- The DNS Security Extensions
- Defined in RFC 2535
- Still under development by the DNSEXT working group of the IETF. They may change before the final standard is defined
- Uses public key cryptography to digitally sign data, thereby proving it's authenticity
- Current implementation is BIND 9

PATRICIA EGEN CONSULTING

DNSSEC

- Private key is kept secret
  - Usually in a file on the server somewhere
  - Included into the named.conf file
  - File access can be limited
- Public key is published to the world
  - It is kept in a new dns record type called a “key” record
  - This “key” record is attached to the domain name of the zone

PATRICIA EGEN CONSULTING

Key signing

- Keys can be signed by a “higher authority” to guarantee their authenticity
  - The uga.edu key would be signed by the .edu zone key
  - The .edu key would be signed by the root zone key
  - None of this is currently in place, but will be when DNSSEC becomes more widely used
- Signatures are stored in “sig” records

PATRICIA EGEN CONSULTING

Signing of Data

- Data is signed at the RRset level
- An RRset consists of all the resource records with:
  - The same owner
  - The same class
  - The same type
- They are always returned together when a query is made to the name server

PATRICIA EGEN CONSULTING

DNSSEC

- There are many more details than I have shown here. For more information about DNSSEC, see Albitz and Liu, Chapter 11, or read the applicable RFCs
- Also please note that the DNS security implemented with MS W2K DNS products is NOT compatible with the DNS security implemented in BIND

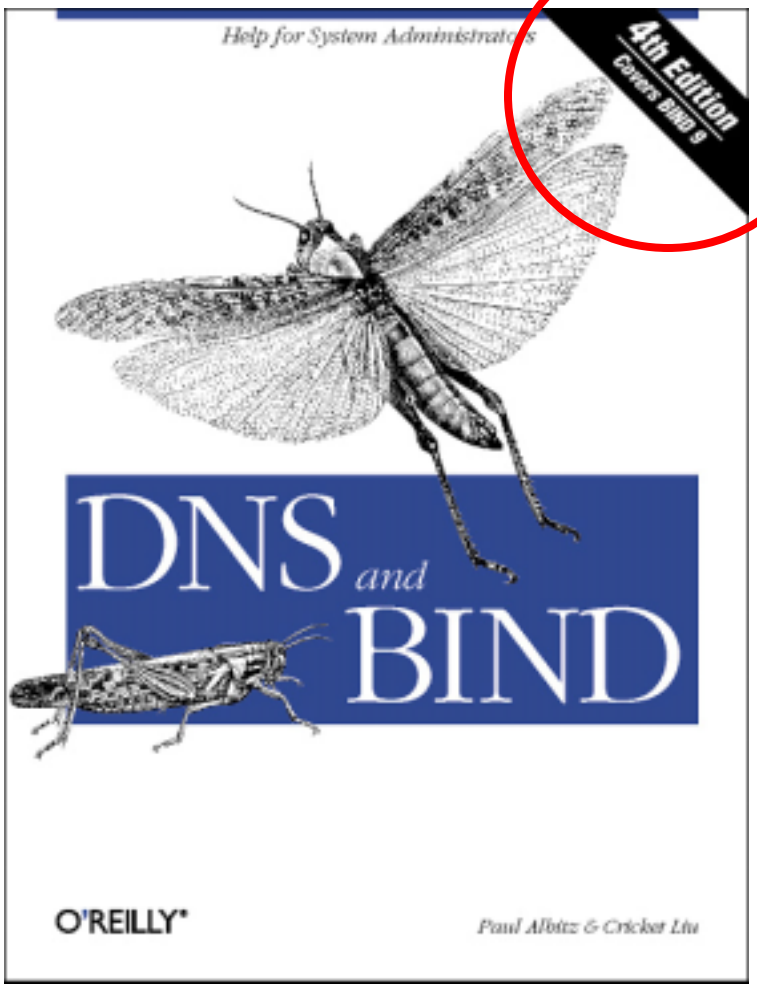
PATRICIA EGEN CONSULTING

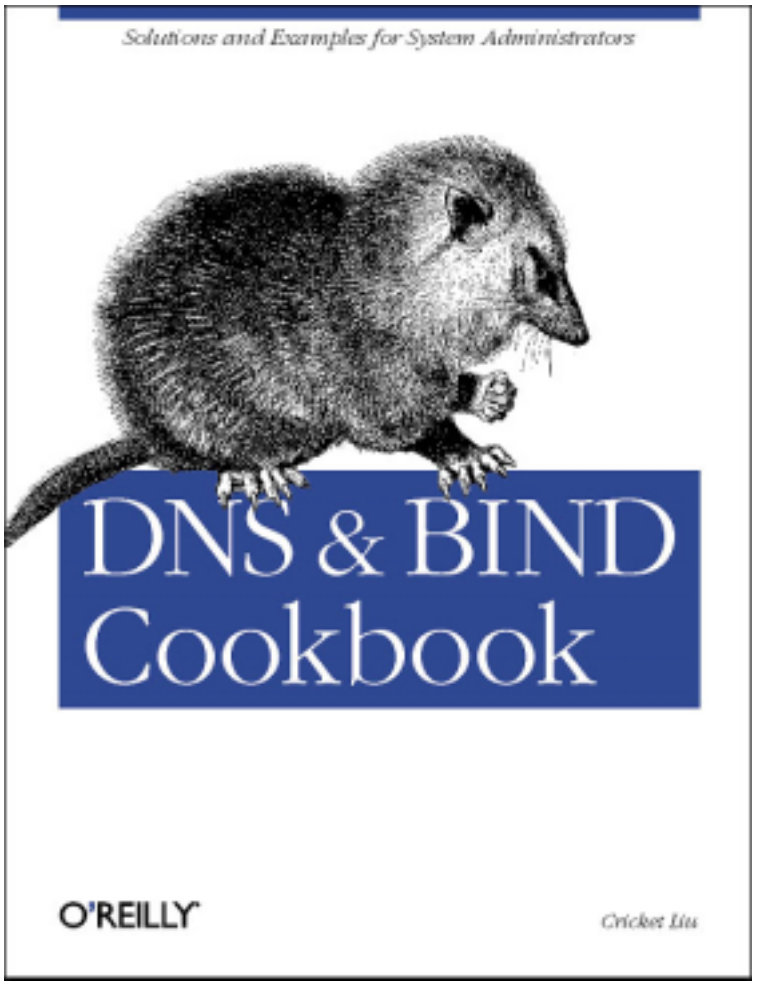
Tools for troubleshooting

- nslookup
  - Deprecated and may be removed from the bind distribution at some date
- dig
  - Domain Information Groper
  - General purpose DNS lookup tool
    - Performs lookups and returns results
    - Very flexible (lots of options)
- host
  - Another general purpose DNS lookup tool

PATRICIA EGEN CONSULTING

Primary Reference Materials





www.oreilly.com

PATRICIA EGEN CONSULTING

Other References

- Internet Security Consortium
  - www.isc.org
- Internet RFC Archives
  - www.faqs.org/rfcs
- ICANN Home Page
  - www.icann.org
- DNS Resource Directory
  - www.dns.net/dnsrd

PATRICIA EGEN CONSULTING

Other References

Men and Mice DNS Glossary

www.menandmice.com/online_docs_and_faq/glossary/glossarytoc.htm

Rob Thomas' secure bind template

www.cymru.com/Documents/secure-bind-template.html

DNS Security Extensions

www.dnssec.net

BIND 9 Administrator Reference Manual

www.nominum.com/content/documents/bind9arm.pdf

PATRICIA EGEN CONSULTING

Mailing Lists

- bind-announce
- bind-users
- bind9-users
  - All lists are hosted at isc.org along with their archives
  - To subscribe go to

  www.isc.org/services/public/lists/bind-lists.html

Always check the archives BEFORE asking for help

PATRICIA EGEN CONSULTING

Tools to maintain your domain

- DNS Builder

  www.ewas.net/tools/dnsbuilder

- WEBMIN

  www.webmin.com

- Lots of others
  - Some free
  - Some not

PATRICIA EGEN CONSULTING

Session 9261
Th-th-th-that's all folks

Questions?

PATRICIA EGEN CONSULTING