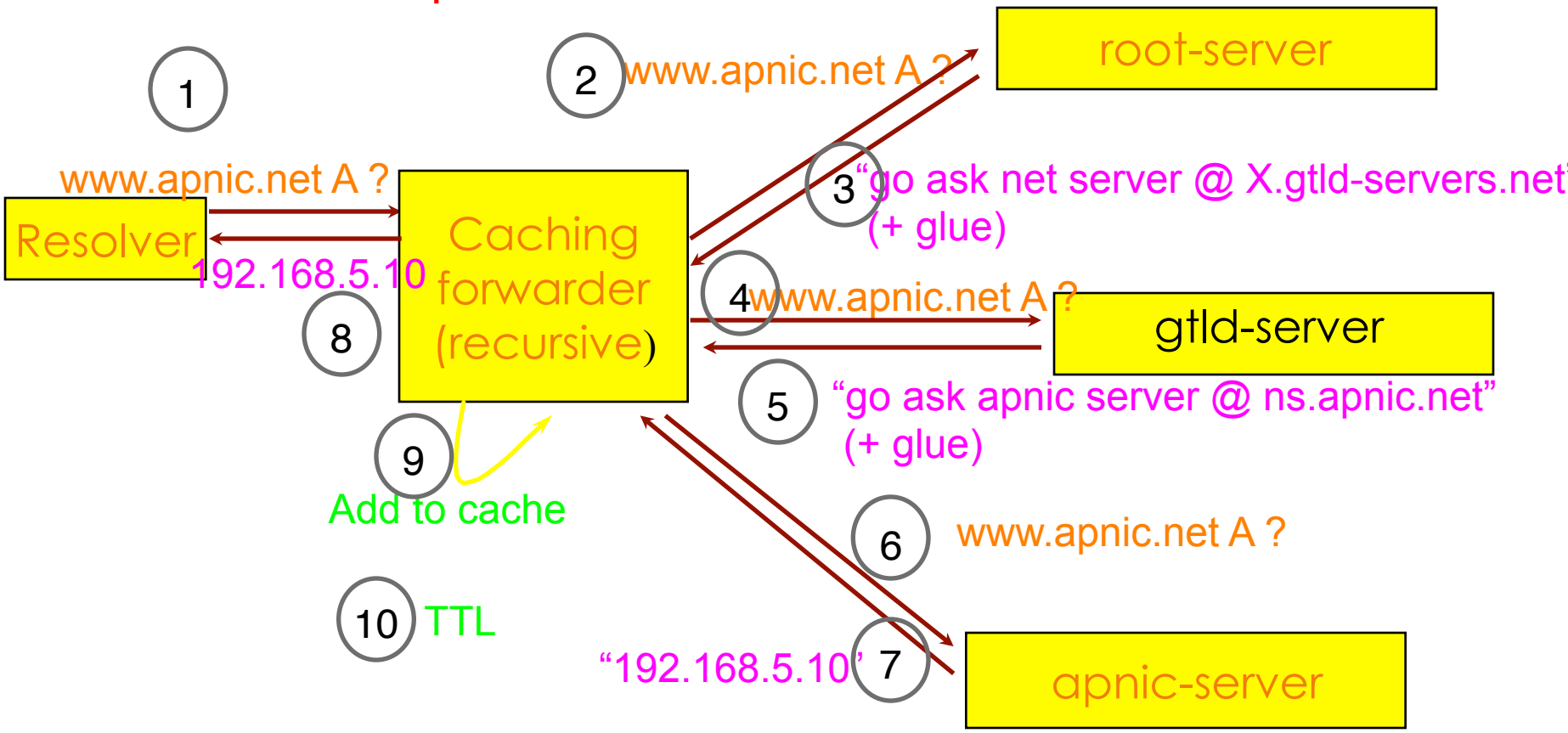
APNIC eLearning:
DNS Security

Contact: training@apnic.net

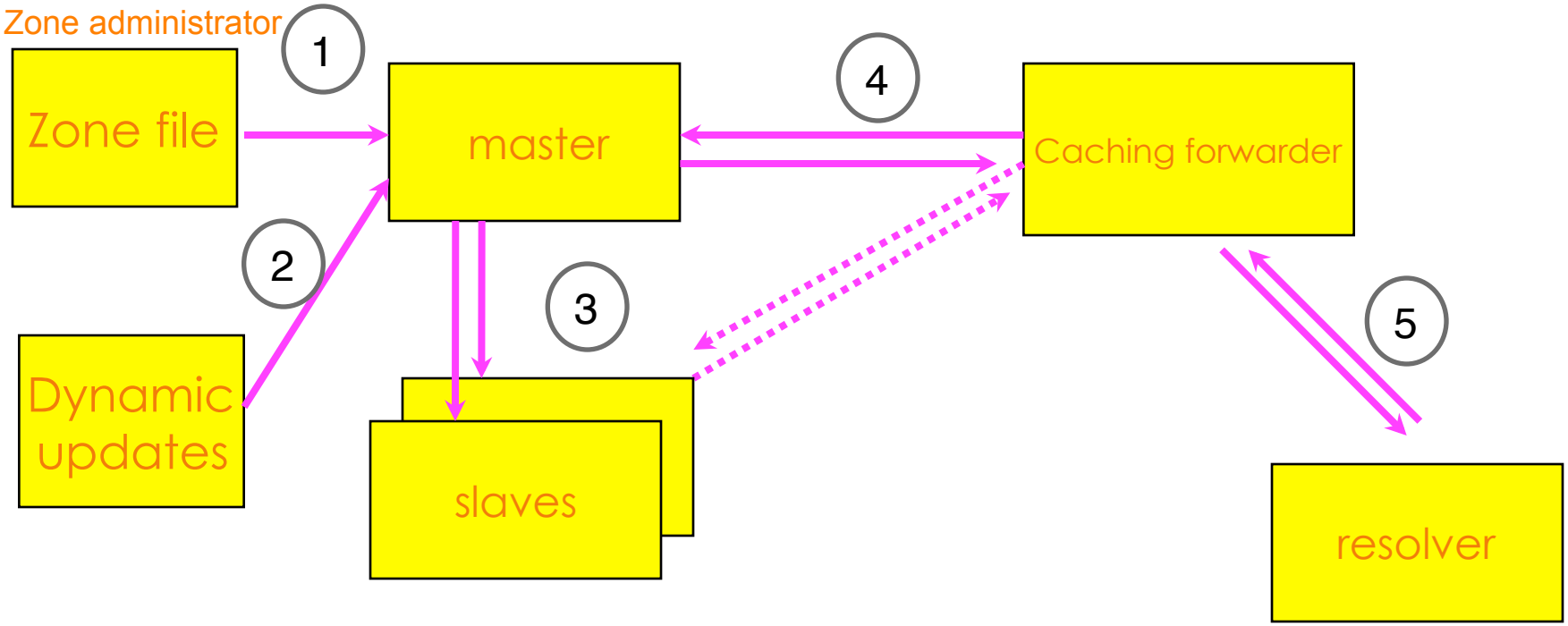
Overview

- How DNS Works
- DNS Vulnerabilities
- Securing the Nameservers
- Transaction Signature (TSIG)
- DNS Security Extensions (DNSSEC)
- DNSSEC New Resource Records
- Signing Zones

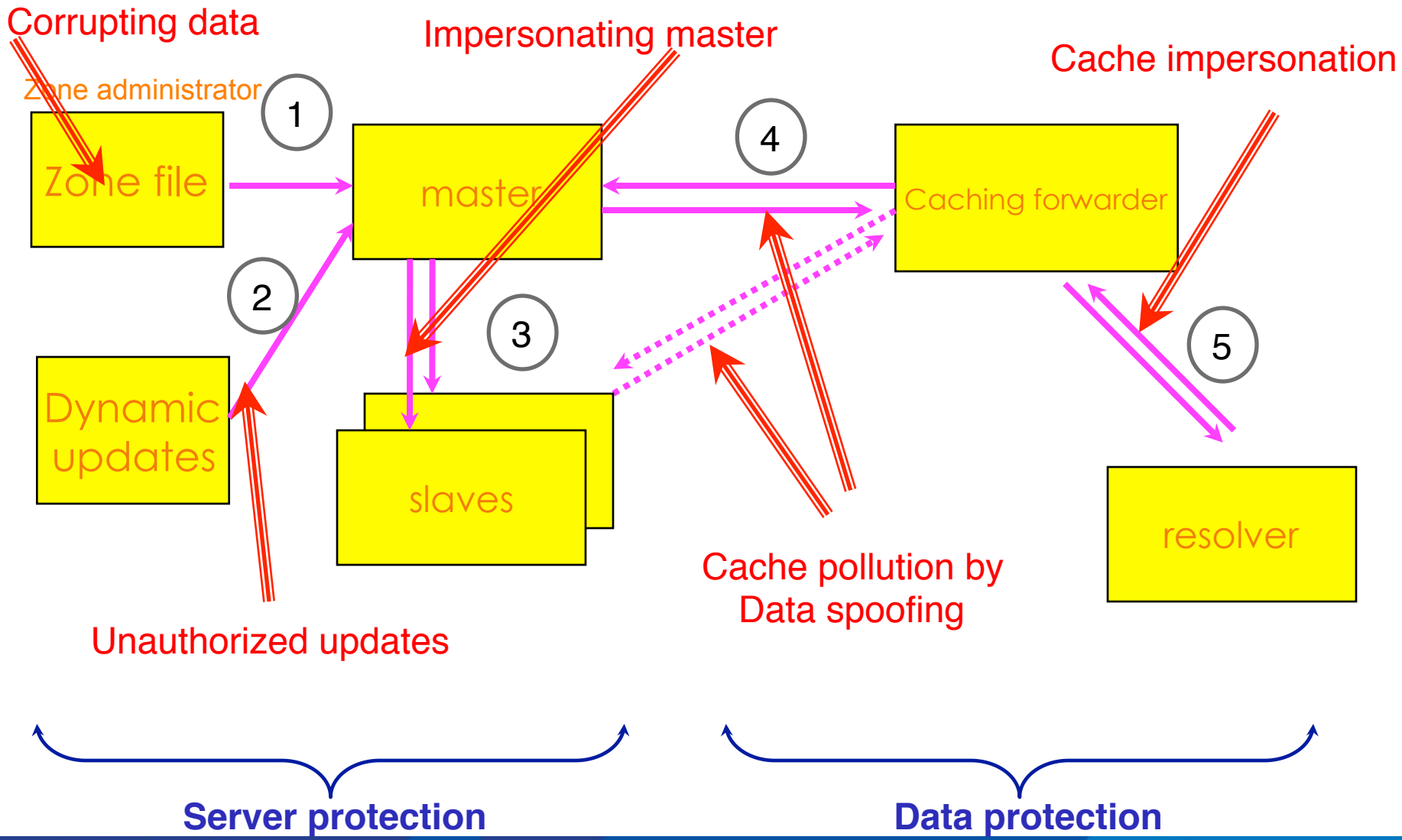
Overview: How DNS Works



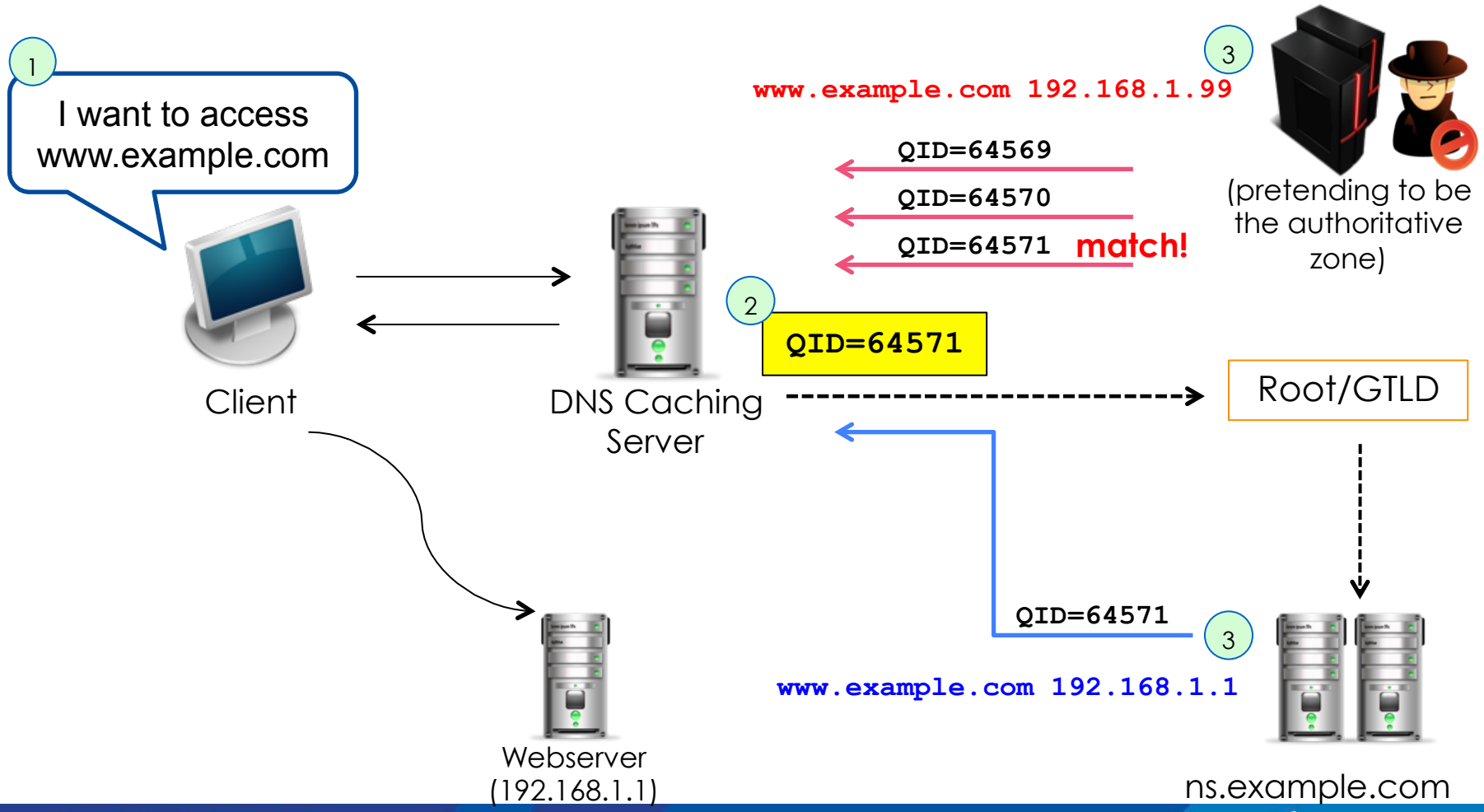
DNS Vulnerabilities



DNS Vulnerabilities



DNS Cache Poisoning



RFC 4033: DNS Security Introduction and Requirements

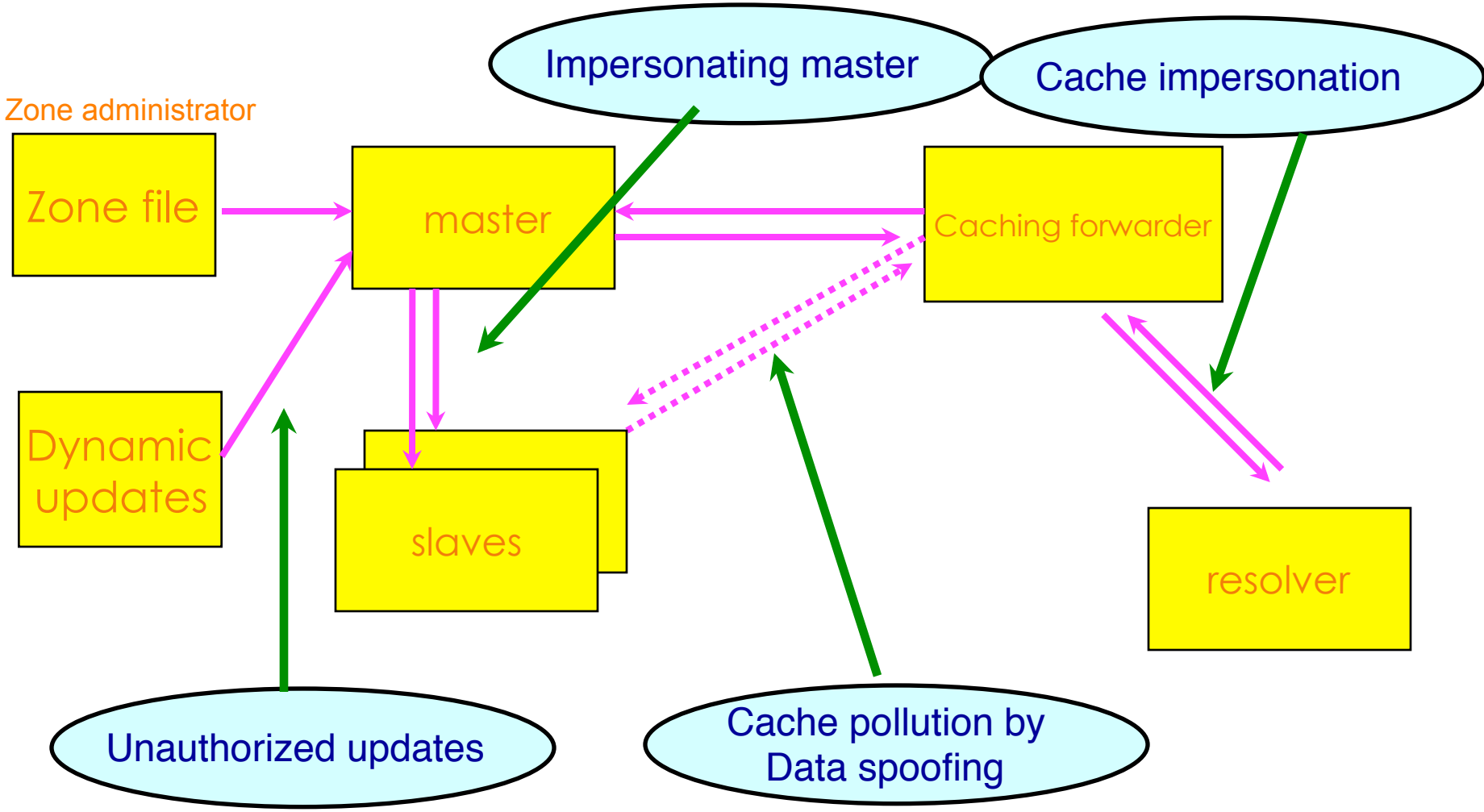
Securing the Nameserver

- Run the most recent version of the DNS software
  - Bind 9.9.1 or Unbound 1.4.16
  - Apply the latest patches
- Hide version
- Restrict queries
  - `Allow-query { acl_match_list; };`
- Prevent unauthorized zone transfers
  - `Allow-transfer { acl_match_list; };`
- Run BIND with the least privilege (use `chroot`)
- Randomize source ports
  - don't use `query-source` option
- Secure the box
- Use TSIG and DNSSEC

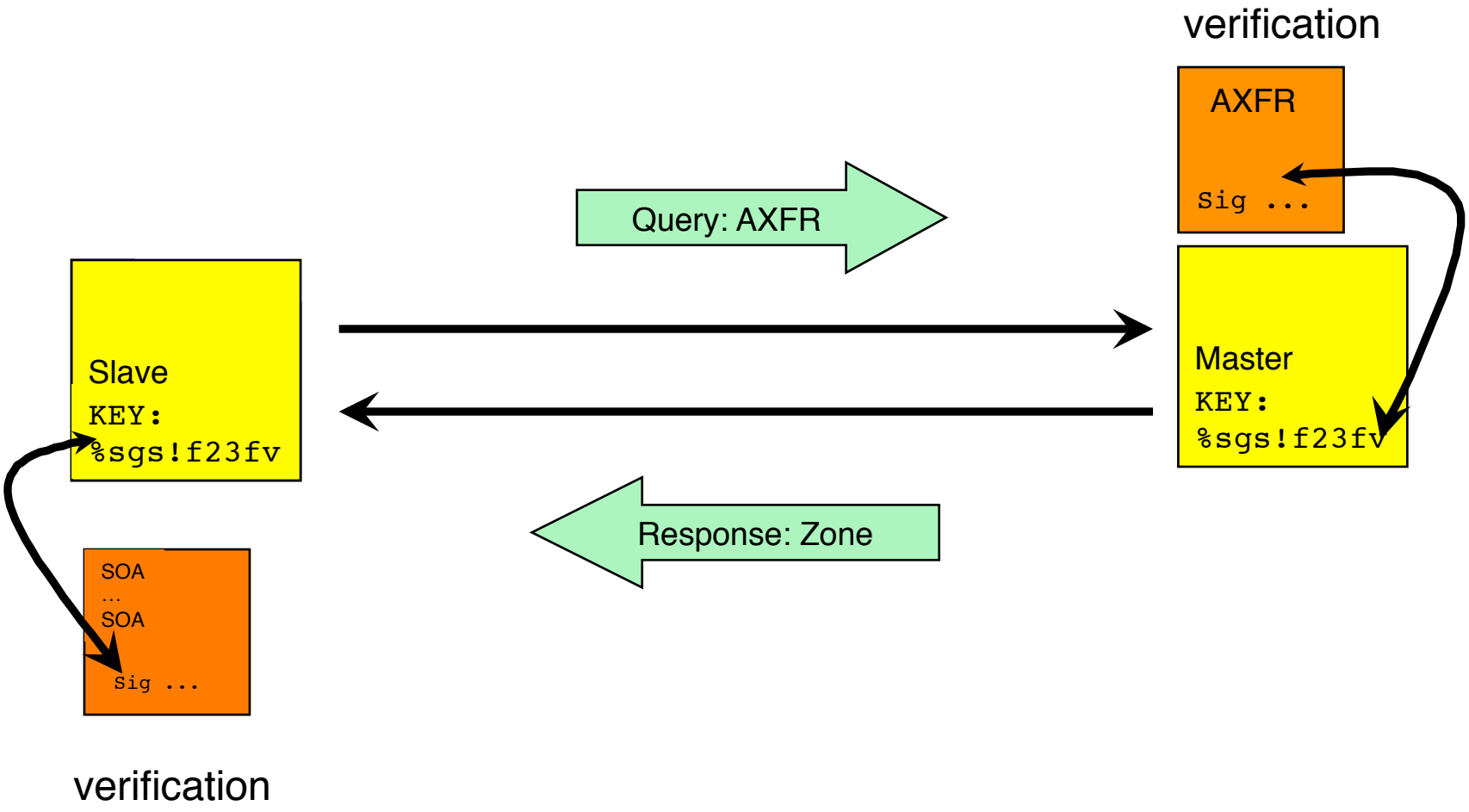
Transaction Signature (TSIG)

- A mechanism for protecting a message from a primary to secondary and vice versa (i.e. transactions)
- A keyed-hash is applied (like a digital signature) so recipient can verify message
  - DNS question or answer & the timestamp
  - Based on a shared secret - both sender and receiver are configured with it
- RFC 2845

TSIG Protected Vulnerabilities



TSIG Example



TSIG Steps

- Generate secret
  - `dnssec-keygen -a <algorithm> -b <bits> -n host <name of the key>`
- Communicate secret
  - Transfer the key securely (ex. SSH/SCP)
- Configure the servers
  - Edit configuration file for primary and secondary
- Test
  - dig @<server> <zone> AXFR -k <TSIG keyfile>

TSIG Configuration – named.conf

Primary server 10.33.40.46

```
key ns1-ns2.pcx. net {
  algorithm hmac-md5;
  secret "APlaceToBe";
};
server 10.33.50.35 {
  keys {ns1-ns2.pcx.net;};
};

allow-transfer {
  key ns1-ns2.pcx.net ;};
```

Secondary server 10.33.50.35

```
key ns1-ns2.pcx.net {
  algorithm hmac-md5;
  secret "APlaceToBe";
};
server 10.33.40.46 {
  keys {ns1-ns2.pcx.net;};
};
zone "my.zone.test." {
  type slave;
  file “myzone.backup”;
  masters
       {10.33.40.46;}; };
```

You can save this in a file and refer to it in the config file (named.conf) using ‘include’ statement:

```
include “/var/named/master/tsig-key-ns1-ns2”;
```

TSIG Testing - dig

- You can use dig to check TSIG configuration

```
dig  @<server> <zone> AXFR -k <TSIG keyfile>
```

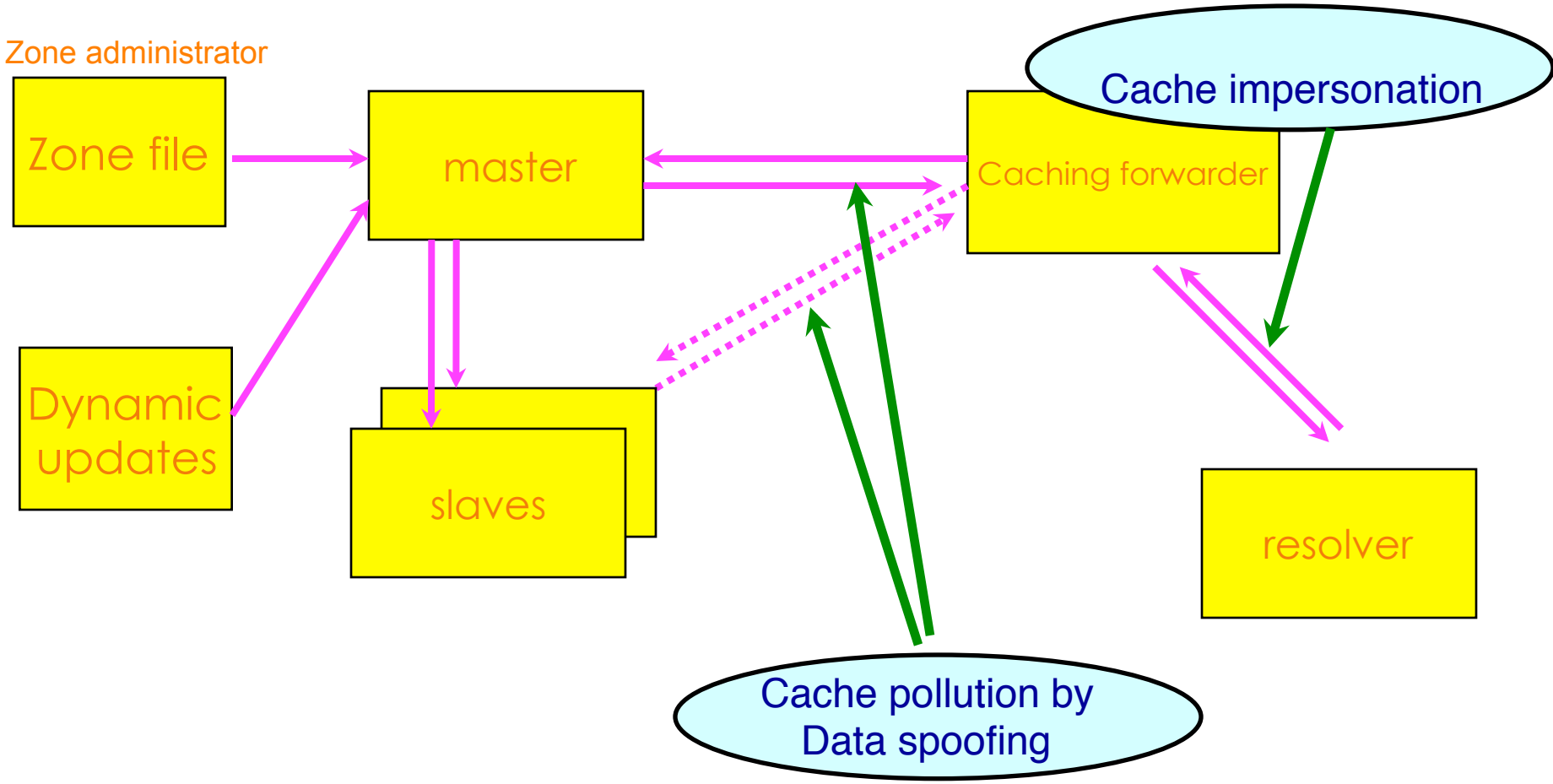
```
$ dig @127.0.0.1 example.net AXFR \
-k Kns1-ns2.pcx.net.+157+15921.key
```

- A wrong key will give “Transfer failed” and on the server the security-category will log this.
- Note: TSIG is time-sensitive

DNS Security Extensions (DNSSEC)

- Protects the integrity of data in the DNS by establishing a chain of trust
- A form of digitally signing the data to attest its validity
- RFC 4033, 4034, 4035
- DNSKEY/RRSIG/NSEC: provides mechanisms to establish authenticity and integrity of data
- DS: provides a mechanism to delegate trust to public keys of third parties

Vulnerabilities protected by DNSSEC



DNSSEC New Resource Records

- 3 Public key crypto related RRs
  - RRSIG = Signature over RRset made using private key
  - DNSKEY = Public key, needed for verifying a RRSIG
  - DS = Delegation Signer; ‘Pointer’ for building chains of authentication
- One RR for internal consistency
  - NSEC = Next Secure; indicates which name is the next one in the zone and which typecodes are available for the current name
    - authenticated non-existence of data

Types of Keys

- Zone Signing Key (ZSK)
  - Sign the RRsets within the zone
  - Public key of ZSK is defined by a DNSKEY RR
- Key Signing Key (KSK)
  - Signed the keys which includes ZSK and KSK and may also be used outside the zone
- Trusted anchor in a security aware server
- Part of the chain of trust by a parent name server
- Using a single key or both keys is an operational choice (RFC allows both methods)

DNSSEC - Setting up a Secure Zone

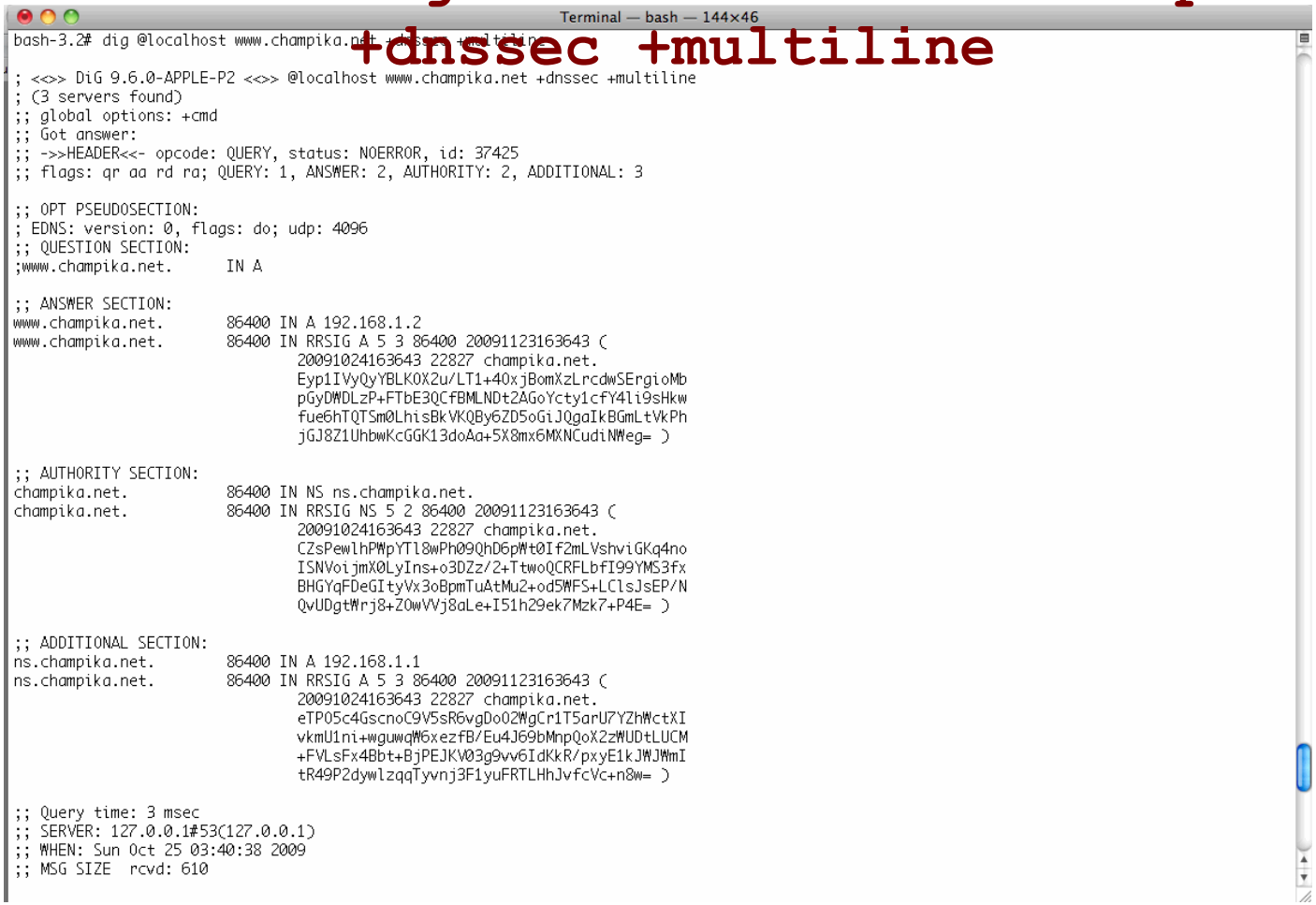
- Enable DNSSEC in the configuration file (named.conf)
  - `dnssec-enable yes; dnssec-validation yes;`
- Create key pairs (KSK and ZSK)
  - `dnssec-keygen -a rsasha1 -b 1024 -n zone champika.net`
- Publish your public key
- Signing the zone
- Update the config file
  - Modify the zone statement, replace with the signed zone file
- Test with dig

Signing the Zone

- `dnssec-signzone -o champika.net db.champika.net Kchampika.net.+005+33633`
- Once you sign the zone a file with a .signed extension will be created
  - `db.champika.net.signed`
- Note that only authoritative records are signed NS records for the zone itself are signed
  - NS records for delegations are not signed
  - DS RRs are signed!
  - Glue is not signed
- Difference in the file size
  - db.champika.net vs. db.champika.net.signed

Testing with dig: an example

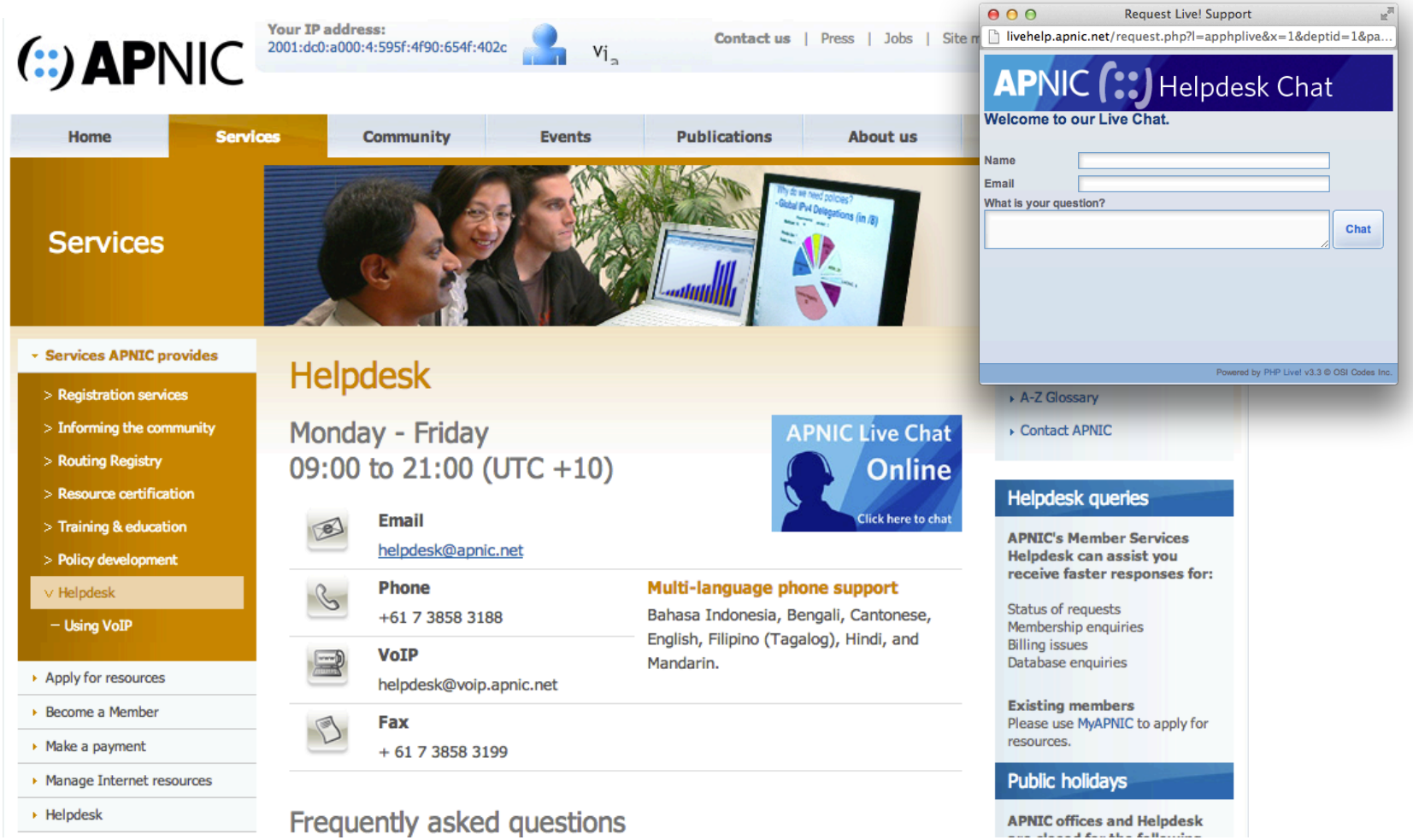
```
dig @localhost www.champika.net
+dnssec +multiline
```



Questions

- Please remember to fill out the feedback form
  - <survey-link>
- Slide handouts will be available after completing the survey

APNIC Helpdesk Chat



Thank you!

End of Session

APNIC