Reverse DNS

Reverse Mapping

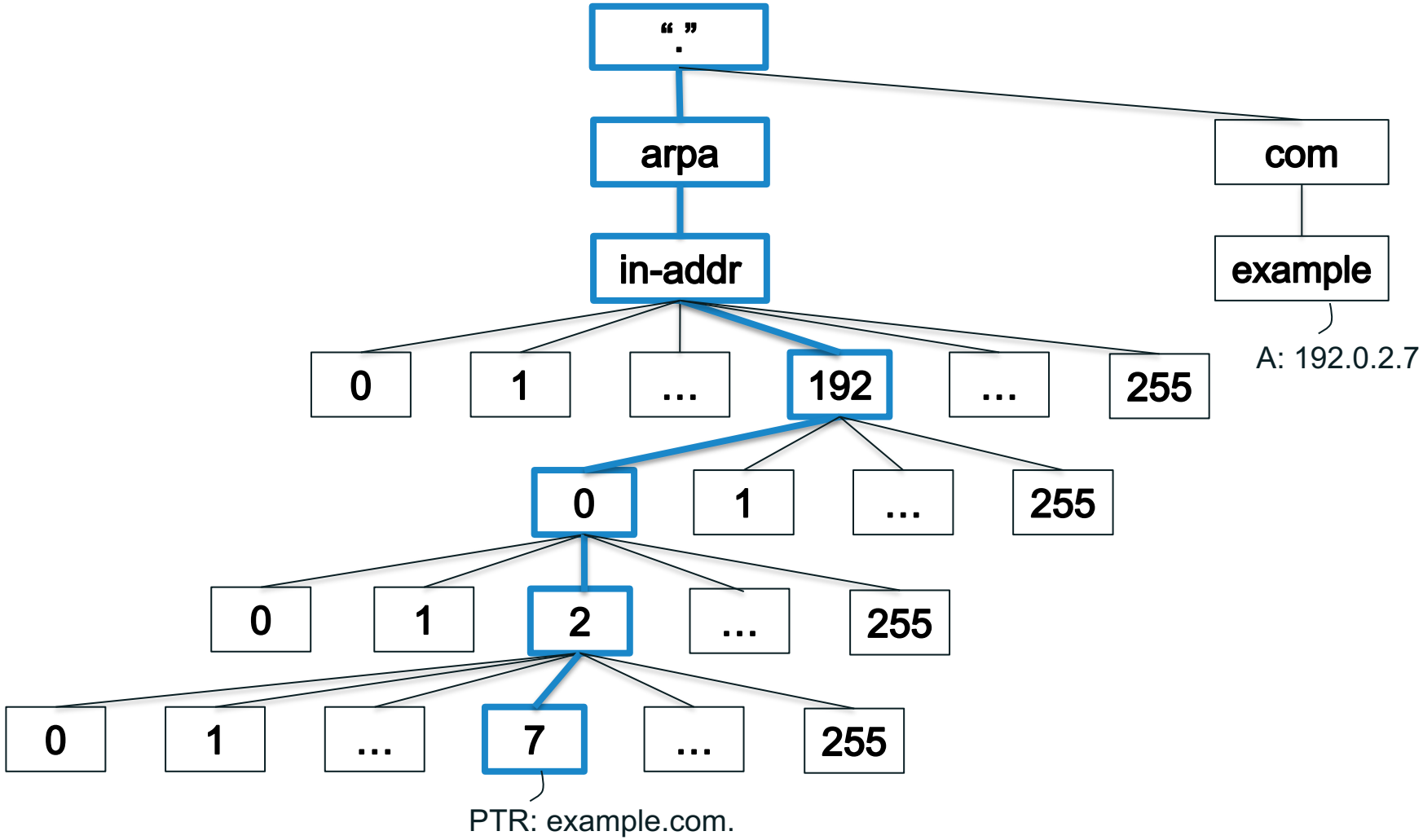
- Name-to-IP is “forward” mapping
- IP-to-name is “reverse” mapping
- Reverse mapping accomplished by mapping IP address space to the DNS name space
  - IPv4 addresses under *in-addr.arpa*
  - IPv6 addresses under *ip6.arpa*
- Uses PTR (pointer) records

```
7.2.0.192.in-addr.arpa.    PTR      example.com.
```

- Corresponds to this A record:

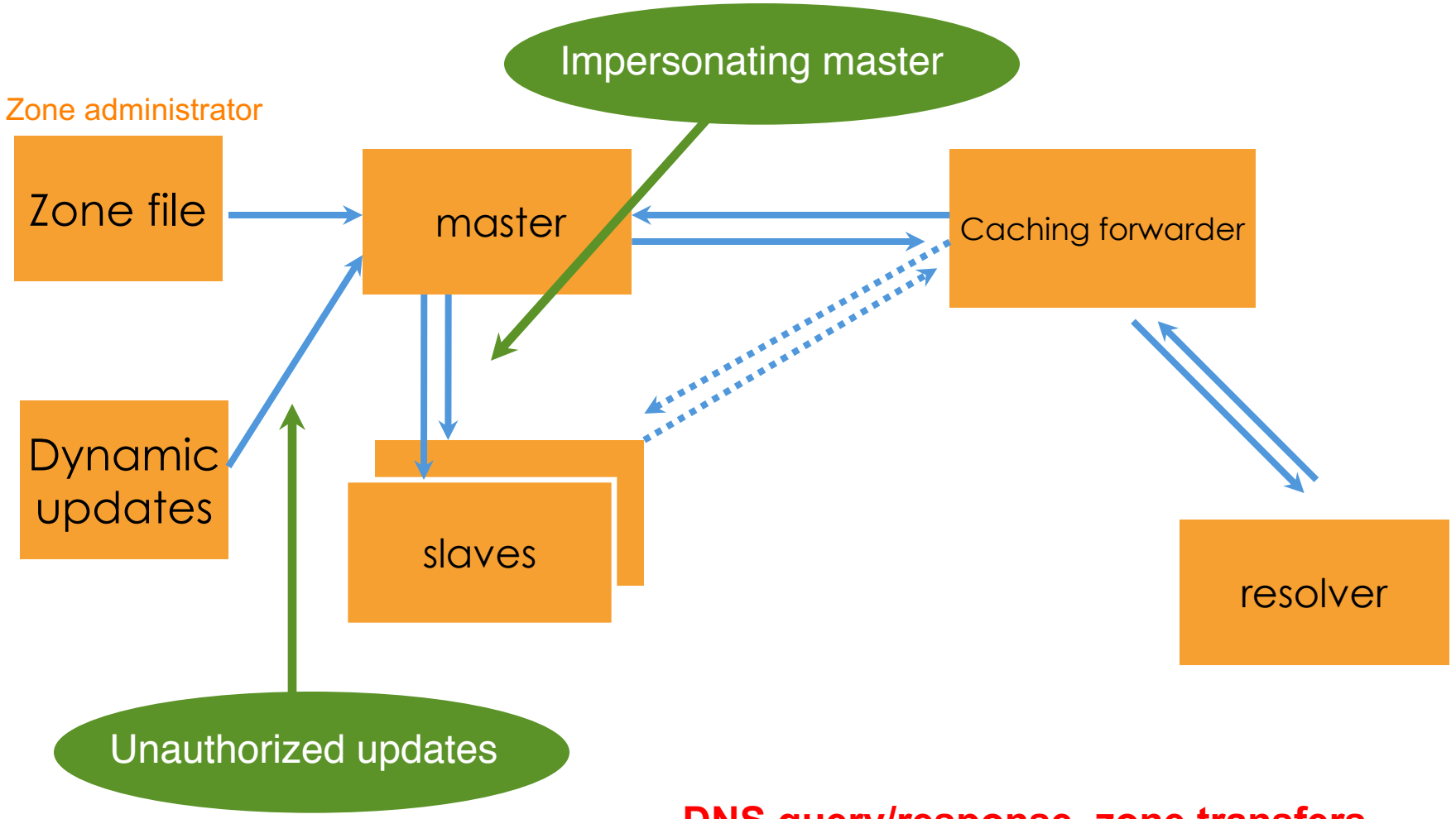
```
example.com.                A       192.0.2.7
```

Reverse Mapping



Transaction Signatures

Transactions - Protected Vulnerabilities



DNS query/response, zone transfers, Dynamic updates

TSIG steps

1. Generate secret
2. Communicate secret
3. Configure servers
4. Test

TSIG – Generating a Secret

- dnssec-keygen
  - A simple tool to generate keys
  - Used here to generate TSIG keys

```
dnssec-keygen -a <algorithm> -b <bits> -n
  host <name of the key>
```

TSIG – Generating a Secret

- Example

```
> dnssec-keygen –a HMAC-SHA256 –b 256 –n HOST ns1-
  ns2.pcx.net

This will generate the key

  Kns1-ns2.pcx.net.+157+15921

>ls
Kns1-ns2.pcx.net.+157+15921.key
Kns1-ns2.pcx.net.+157+15921.private
```

TSIG – Generating a Secret

- TSIG is used in server configuration, not in zone file
- Could be confusing because it looks like RR

```
ns1-ns2.pcx.net. IN KEY 128 3 157 nEfRX9…bbPn7lyQtE=
```

TSIG – Configuring Servers

- Configuring the key

```
key { algorithm ...; secret ...;}
```

- Making use of the key

```
server x { key ...; }
```

where x is the IP address of the other server

Configuration Example – named.conf

Primary server 192.168.1.100

```
key ns1-ns2.pcx. net {
    algorithm hmac-md5;
    secret "APlaceToBe";
};
server 192.168.1.200 {
    keys {ns1-ns2.pcx.net;};
};
zone "my.zone.test." {
    type master;
    file “db.myzone”;
    allow-transfer {
    key ns1-ns2.pcx.net ;};
};
```

Secondary server 192.168.1.200

```
key ns1-ns2.pcx.net {
    algorithm hmac-md5;
    secret "APlaceToBe";
};
server 192.168.1.100 {
  keys {ns1-ns2.pcx.net;};
};
zone "my.zone.test." {
    type slave;
    file “myzone.backup”;
    masters {192.168.1.100;};
};
```

You can save this in a file and refer to it in the 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 @localhost example.net AXFR \
   -k Kns1-ns2.pcx.net.+157+15921.key
```

- A wrong key will give “Transfer failed” and will be logged on the server’s using the security-category

TSIG Testing - Time

- TSIG is time sensitive
- Message protection expires in 5 minutes
  - Make sure time is synchronized
  - For testing, set the time
  - In operations, (secure) NTP is needed