

RNDC & TSIG

What is RNDC?

- Remote Name Daemon Controller
- Command-line control of named daemon
- Usually on same host, can be across hosts
 - Locally or remotely

Configuring RNDC

- "rndc-confgen" generates lines to be added to two files
 - rndc.conf
 - named.conf

Generating the lines: > rndc-confgen

```
key "rndc-key" {
    algorithm hmac-md5;
    secret "rXxroiejf8937Bjf_+532ktj/==";
};

Options {
    default-key "rndc-key";
    default-server 127.0.0.1;
    default-port 953;
#End of rndc.conf

# User with the followign in named.conf, adjusting the
# allow list as needed
# key "rndc-key" {
#     algorithm hmac-md5;
#     secret "rXxroiejf8937Bjf_+532ktj/==";
# };
# controls {
#     inet 127.0.0.1 port 953
#     allow { 127.0.0.1; } keys { "rndc-key" };
# }
```

Using an rndc.conf file

- /etc/rndc.conf specifies defaults for rndc
- E.g.,

```
key "rndc-key" {
    algorithm hmac-md5;
    secret "dY7/uliR0fKGvi5z50+Q==";
};

options {
    default-key "rndc-key";
    default-server 127.0.0.1;
    default-port 953;
};
```

Enabling RNDC in the server – named.conf

- key definition

```
key rndc-key {
    secret "dY7/uliR0fKGvi5z50+Q=="; algorithm
    hmac-md5;
};
```

– Warning: example secret looks good but is invalid (don't copy it!)
- controls statement

```
controls {
    inet 127.0.0.1 port 953 // for remote host, use
    allow { 127.0.0.1; } // actual IP
    keys { "rndc-key" };
};
```

What can be done with RNDC

- > rndc stop - kills server
- > rndc status - prints some information
- > rndc stats - generates stat file (named.stats)
- > rndc reload - refresh zone(s), with variations
- > rndc trace - increases debug level
- > rndc flush - removes cached data
- other commands in the ARM

TSIG

What is TSIG - Transaction Signature?

- A mechanism for protecting a message from a primary to secondary and vice versa
- 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

What is TSIG - Transaction Signature?

- TSIG (RFC 2845)
 - authorizing dynamic updates & zone transfers
 - authentication of caching forwarders
- Used in server configuration, not in zone file

Names and Secrets

- TSIG name
 - A name is given to the key, the name is what is transmitted in the message (so receiver knows what key the sender used)
- TSIG secret value
 - A value determined during key generation
 - Usually seen in Base64 encoding

Using TSIG to protect AXFR

- Deriving a secret

```
> dnssec-keygen -a <algorithm> -b
  <bits> -n host <name of the key>
e.g.
> dnssec-keygen -a HMAC-MD5 -b 128 -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
```

Using TSIG to protect AXFR

- Configuring the key
 - in named.conf file, same syntax as for rndc
 - key { algorithm ...; secret ...; }
- Making use of the key
 - in named.conf file
 - server x { key ...; }
 - where 'x' is an IP number of the other server

Configuration Example – named.conf

```
Primary server 10.33.40.46      Secondary server 10.33.50.35

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

server 10.33.50.35 {           server 10.33.40.46 {
    keys {ns1-ns2.pcx.net;};    keys {ns1-ns2.pcx.net;};
};                               };

zone "my.zone.test." {         zone "my.zone.test." {
    type master;                type slave;
    file "db.myzone";           file "myzone.backup";
    allow-transfer {            masters {10.33.40.46;};
        key ns1-ns2.pcx.net ;}; allow-transfer {
};                               key ns1-ns2.pcx.net;};
                                };
                                
```

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";`

TIME!!!

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

Other uses of TSIG

- TSIG was designed for other purposes as well
 - Protecting sensitive stub resolvers
 - This has proven hard to accomplish
 - Dynamic Update
 - Discussed later, securing this relies on TSIG

Questions ?
