DNS Server with DNSSEC

Tony Perez
SUNY Polytechnic Institute
100 Seymour Rd
Utica, New York
pereztr@sunyit.edu

ABSTRACT

This project will show how I have setup a Domain Name System (DNS) server with DNS Security Extension (DNSSEC). I have setup a local DNS server in the CS Department Network. It is on a virtual machine (VM) running CentOS minimal with BIND9 as the DNS server and I have then implemented DNSSEC to work with the server for extra security.

Besides showing DNSSEC, I have also implemented other things to show how a local DNS server can be beneficial in a network that deals with internal IP address and also wants a better control of where users can and cannot go to. In my project I have blocked facebook.com and replaced it with another page to show that they are not allowed to go there. I also show how I made custom domains to use internally that point to other servers. This DNS server could be also used to block domains that belong to advertisers, malicious websites, and any other sites you do not want others to go to on your network.

I also show how to test DNSSEC and see what websites are actually using it. I will also talk about the security issues DNSSEC resolves but also why it is still not widely used. There are also other security features that can be used to help protect your DNS server like Hashed Authenticated Denial of Existence (NSEC3) and DNS-based Authentication of Named Entities (DANE) that I will disscuss.

Categories and Subject Descriptors

C.5 [**Computer Systems Organization**]: Computer System Implementation; C.5.5 [**Computer Systems Organization**]: Servers

Keywords

DNS, DNSSEC, VM, Domain Name

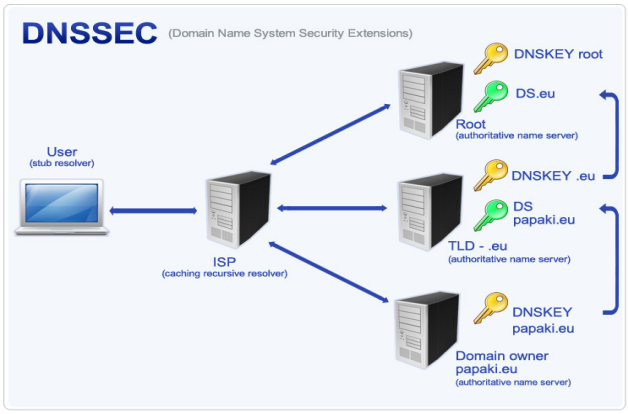

Figure 1: Chain of Trust
Source: DNSSEC-graphix.jpg

1. INTRODUCTION

There are many businesses and users that have a local DNS server setup but do not use all its great features. They also do not know about the security issues DNS servers face. Some things you can do with a DNS server is block unwanted websites; this includes site you do not want users to go to, malicious sites, and ad pages. You can setup a DNS server to also cache sites and so loading time will be quicker.

In today's world there are many that try to point users to the wrong direction. In the case of domain names, they try to match users with the wrong IP address to get them to go to another site which could be unsafe and used to steal their information. DNSSEC is used to help prevent the misdirection. DNSEC does not encrypt data but it does make sure it matches the user to the correct IP address. DNSSEC creates a chain of trust from the root the Top Level Domain (TLD), like .com, to the DNS server that the current records are being stored as shown in Figure 1.

The big problem I am trying to show in my project is that although a local DNS server is great to have, it can be used against you if you use your DNS outside of your network. This problem only exists if you host your DNS server on the web and use it to get to domains in your internal network or maybe you have them point to other servers only your DNS server knows.

2. RELATED WORK

Table 1: Network Configuration

Master DNS	10.103.67.80
Slave DNS	10.103.67.81

In my previous lab we have already learned to setup a DNS server. I will be using the same two VM's, master and slave, and expand on it. As I looked into DNS server and what they can do I have found other that have implemented them to block many unwanted sites. I found one article on how someone used their DNS server to block ad pages [4].

I have also looked into how others have setup their DNS servers with DNSSEC [2]. I will not be using my DNS server on the web and so I needed to figure the steps as if I was, just to show how someone would implement DNSSEC on a DNS server. The reason for this was because I needed to demonstrate how the zone file files were signed and the type of keys they used, which is similar to the SSH key process.

3. SETTING UP SERVER

For my setup I needed:

- Primary DNS
- Secondary DNS
- Client

I used two server, one for the master DNS and the other for the slave DNS, and then need a client machine to test the DNS servers with. Since I had already setup a DNS server previously I did not need do much for that part but I will do a quick overview of how it is done. Table 1 show the network configuration I used for my servers

To install on CentOS is was really simple. You just had to use `yum` and install the `bind` and `bind-utils` packages. Once that is done you just need to edit the `named.conf` file with the zones you wish to add. In my configuration I had a zone to block facebook.com, one with my internal addresses, and one to redirect users from google.com to another. The site redirection was used in both the case of facebook.com and google.com but in the case of facebook.com it was used show that we can block a site. In the case of google.com it was to show how it would seem when you were directed to a bad website.

Once that was done you had to created the zone files. In my case I had made a dummy-zone to like Pameranz did to point any blocked sites to [4]. Once that was done I made a zone for a domain called dontattack which I used to show the zone signing process.

The step-by-step instructions I used to setup the basic DNS servers can by viewed from here[1].
The step-by-step guide to getting DNSSEC running can be viewed here[2].

I will run through the modified steps I have taken to show how to sign the keys but not put them on a Top Level Domain servr because my DNS server is not accessible outside of the CS Network.

After setting up my basic DNS server and setting the domains and zone file I wanted to use I began to sign the zone files and then test them locally. To do this I went into the directory where my zone files are located on my master DNS server. The next step was to sign them using the `dnssec-keygen` command on the zone file you want to use. We need

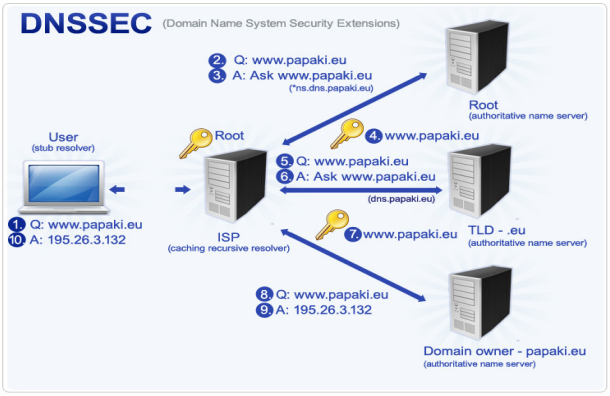

Figure 2: Process of Chain of Trust
Source: dnssec-grafix.jpg

to make a Zone Signing Key (ZSK) and a Key Signing Key (KSK). The ZSK is used as a long term key to sign the zone file and the KSK is used as a short term key to sign the ZSK. To sign the zone file you must first sign it with the ZSK and then the KSK. This offers a better protection because you must always resign your zone files in a certain time because both jeys expire. The KSK is used to sign the keys on your DNS server while the ZSK is used to sign the zone files which then create a public key that must put on a Top Level Domain.

In my case I used my `dontattack.zone` file. I generated the ZSK set with the `dnssec-keygen` command and the KSK with the `dnssec-keygen -f KSK`. Now there will be 4 key files that have been generate. Both the ZSK and KSK have public and private keys. We must put the public keys of the ZSK and KSK in the zone file to then sign them. After the keys are generated for the zone file we must now sign them using the `dnssec-signzone` command in this format `dnssec-signzone -3 <salt> -A -N INCREMENT -o <zonename> -t <zonefilename>` Once this is done the zone files have been signed and we can see a file with the end `.signed` has been created with the same name as our original zone file name. That is the file we must use from now on in our `named.conf` file. This is as far as I went since my DNS server is not accessible from the outside.

4. USE OF A LOCAL DNS SERVER

If you are using a DNS server locally then you can use it for things like caching, blocking domains of unwated sites like advertisement or malicious sites. A local DNS server can also make it easier for you if you use local static IP address and so you can use domain names instead of the IP address and make it easier for you to remember and type.

When using a DNS server to block ad pages it can improve the quality of your network by alot because they would instantly be pointed to your own server and not have to load any pages from outside the network. To figure out which ad sites to block you could look up common ad domains or do some research based on the sites your go to and figure out what site link to where like Pomeranz has done [4].

5. TESTING DNS & DNSSEC

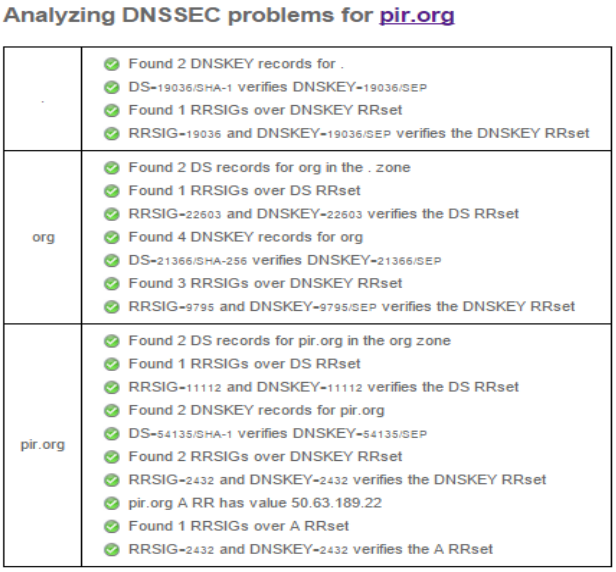

Figure 3: Results of pir.org

To test my DNS server and see if the files have been signed I can use the `dig` locally to see if the domain `dontattack` had the keys and if they are signed. You can use `dig A www.dontattack. @localhost +noadditional +dnssec +multiline` and `dig DNSKEY dontattack @localhost +multiline` to test and see if you get a answer with your server info and information on your keys. A example can be seen in Jesin's example[2].

There is another way to test if a domain name is on a DNS server that has DNSSEC implemented. You can use verisignlab's tool to search a domain and see if it does or does not have DNSSEC running. I used pir.org to show DNSSEC being used on all Domain levels as shown in figure 3. When a chain of trust is made the process can be seen in figure 2.

To show that my DNS server worked I used a my laptop as the client but because it was not on the same network as my DNS server I had to use SSH SOCKS Proxy in Firefox to show that it works. To do this you can run the use a SSH client and run the following command `ssh -d 8080 user@host.com`. The `-d` flag is what sets up the port you wish to go through and creates a SSH tunnel. In a browser you then configure it to go through a SOCKS proxy and use your locahost and the IP address and 8080 as the port.

I showed how I blocked facebook.com and redirected it to go to a page that tells the user why it was blocked. The page was made by me and I used an Apache web server to display the page.

6. ISSUES & DELAYS

If you host a DNS server in the open you cannot just rely on DNSSEC to secure it. As shown in figure 4.

Some issues I faced was figuring out how to test and see if my DNS server had DNSSEC running. I also thought that I had to use a client that was on the same network as my DNS server to test it. My biggest issue was that I could not demonstrate DNSSEC on my DNS server because my DNS server could not reach the outside.

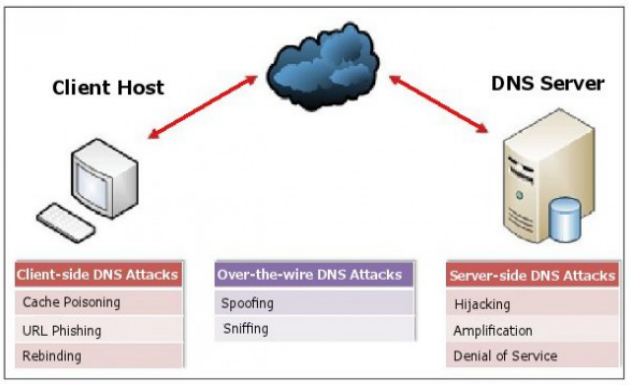

Figure 4: Security Issues from different points
Source: dns1-590x344.jpg

6.1 Solutions

To solve these issues I had to find another way on how show DNSSEC working on a DNS server. I came across verisignlab's tool to search domains and see if they DNSSEC implemented and where the chain of trust ends. DNSSEC was only implemented on the root and TLD for most domains. For my presentation I was able to find a domain, pir.org, that had DNSSEC fully implemented. I did not need a client on the same network after figuring out how to use a SOCKS Proxy.

7. FUTURE WORK

The implementation of DNSSEC is still not widely used because most host a DNS Server for internal use only. DNSSEC also does not encrypt data only verifies integrity. To secure a DNS server that is in the open there are other things that can be used like NSEC3 and DANE. These two extensions were made in order to better secure a DNS server because when implementing DNSSEC it actually makes that DNS server vulnerable in another way. A DNS server will normally keep its zone data private but "Due to a deliberate design choice, DNSSEC does not provide" confidentiality [3]. That is why NSEC3 was created. Later came DANE which actually encrypts data when using DNSSEC. These other two security extensions have also not been widely adopted but with further research can be used to eliminate vulnerabilities in a DNS server on the web.

8. CONCLUSIONS

In my project I was able to learn a great amount about DNS servers than what I had learned in class. In class I only setup a basic DNS server and nothing else. I realized how they work when they are actually used in the open. I see to maintatin a DNS server in the open it takes alot of work and there are many secuirty isssues that it faces.

DNSSEC is just one of the many other security extensions that have been created to help secure DNS servers. There are others like NSEC3 and DANE. For a local DNS server you do not need to worry about most problems stated in figure 4.

9. REFERENCES

[1] How to install the bind dns server on centos 6, June 2013.

[2] A, J. How to setup dnssec on an authoritative bind dns server, March 2014.

[3] Arends, R., Austein, R., Larson, M., Massey, D., and Rose, S. DNS Security Introduction and Requirements. RFC 1, Mar. 2005.

[4] Pomeranz, H. A simple dns-based approach for blocking web advertising.