Fundamentos de Redes de Computadores: Servidor DNS com Extensões DNSSEC

Arthur Jahn, Áulus Diniz, Gabriel Araújo, Jonatas Lenon

26 de Novembro de 2015

Sumário

Capítulo 1

Problema e Motivação

O DNS provê serviços de resolução de *hosts* por meio da tradução de nomes de domínios em números de endereçamento IP. Como um mesmo domínio pode estar vinculado a vários endereços IP, esse serviço pode ser responsável também por distribuição de carga entre os *hosts* de destino, alternando o endereço fornecido ao cliente quando uma solicitação de resolução de domínio é feita.

Quando o DNS foi projetado, no início dos anos 80, não foram pensadas questões de segurança. No momento de uma requisição DNS, o cliente simplesmente confia que a informação recebida é válida e legítima. Essa forma de requisição cria vulnerabilidades com o envenenamento de *cache*.

Para se corigir tais vulnerabilidades, foram definidas extensões de segurança para prover confiabilidade nos registros trocados via DNS: o DNSSEC. Para que não fosse modificada a forma como o DNS opera, o DNSSEC simplemente adiciona novos tipos de registros ao DNS como o RRSIG e o DNSKEY e podem ser requisitados da mesma forma como registros do tipo A, CNAME ou MX.

Neste trabalho, pretende-se apresentar como configurar um DNS com as extensões de segurança segundo o DNSSEC e apresentar o passo-a-passo para que um cliente faça requisições seguras. O processo como um todo para configuração das ferramentas utilizadas e como executar o sistema é apresentado a seguir.

Capítulo 2

DNS - *Domain Name System*

Uma consulta DNS comum consiste basicamente de uma requisição enviada pelo cliente seguida de uma resposta devolvida pelo servidor. Nesse sentido, é interessante a utilização de um protocolo de transporte sem estabelecimento de conexão como o UDP para não acarretar em um atraso na requisição, como no caso do protocolo TCP que necessita de um *handshake* de três vias. O processo para configuração do servidor DNS utilizado neste trabalho está descrito a seguir.

2.1 Configuração do Servidor DNS

O servido DNS utilizado nesse experimento foi o Bind9 - *Berkeley Internet Name Domain* - comumente encontrado em distribuições linux.

Para a instalação do servidor Bind, caso não esteja instalado, pode-se instalar seguindo o seguinte passo:

```
sudo apt-get install bind9 dnsutils
```

Após a instalação do servidor DNS é necessario configurar as zonas de endereçamento. Cada zona consiste em um mapeamento de uma URL para um endereço de IP e vice-versa. Para configurar uma zona no Bind, deve-se criar uma arquivo de zona.

Navegue até a pasta referente ao servidor Bind e crie uma pasta para armazenar os arquivos de zona.

```
cd /etc/bind

mkdir -p zones/master
```

O arquivo de zona criado é referênte ao site www.tcpdump.org. Portando o nome do arquivo deve ser: *db.tcpdump.org* e deve conter o seguinte conteudo:

```
;
; BIND data file for tcpdump.org
;
$TTL    3h
@       IN      SOA     ns1.tcpdump.org. admin.tcpdump.org. (
```

```
                            1         ; Serial
                            3h        ; Refresh after 3 hours
                            1h        ; Retry after 1 hour
                            1w        ; Expire after 1 week
                            1h )      ; Negative caching TTL of 1 day
;
@       IN      NS      ns1.tcpdump.org.
@       IN      NS      ns2.tcpdump.org.


tcpdump.org.    IN      MX      10      mail.tcpdump.org.
tcpdump.org.    IN      A       192.139.46.66
ns1                     IN      A       192.139.46.66
ns2                     IN      A       192.139.46.66
www                     IN      CNAME   tcpdump.org.
mail                    IN      A       192.139.46.66
ftp                     IN      CNAME   tcpdump.org.
```

Com o arquivo de zona configurado é necessário informar ao ao servidor DNS a localização desse arquivo.

```
zone "tcpdump.org" {
       type master;
       file "/etc/bind/zones/master/db.linuxconfig.org";
};
```

Com esses arquivos é possivel a partir de uma URL descobrir seu IP.

Uma última coisa é adicionar um endereço de IP de um servidor DNS estável no arquivo named.conf.options. Este endereço de IP é usado quando o servidor DNS local não sabe a resposta para a consulta de resolução de nome. Para este experimento foi utilizado o servidor de DNS do google - 8.8.8.8 ou 8.8.4.4.

```
 forwarders {
     8.8.4.4;
};
```

Com as configurações realizadas liga-se o servidor Bind da seguinte forma:

```
 cd /etc/init.d/
 sudo ./bind9 start
```

Com os servidor DNS Bind é possivel obter o endereço de IP de um site através de um requisição, utilizando o comando dig, no servidor DNS local. Nesse experimento o IP da máquina com o servidor DNS é: 192.168.1.23

```
 dig @192.168.1.23 tcpdump.org

 ; <<>> DiG 9.8.3-P1 <<>> @192.168.1.23 tcpdump.org
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 30817
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2

;; QUESTION SECTION:
;tcpdump.org.                   IN      A

;; ANSWER SECTION:
```

```
tcpdump.org.            10800   IN      A       192.139.46.66

;; AUTHORITY SECTION:
tcpdump.org.            10800   IN      NS      ns1.tcpdump.org.
tcpdump.org.            10800   IN      NS      ns2.tcpdump.org.

;; ADDITIONAL SECTION:
ns1.tcpdump.org.        10800   IN      A       192.139.46.66
ns2.tcpdump.org.        10800   IN      A       192.139.46.66

;; Query time: 165 msec
;; SERVER: 192.168.1.23#53(192.168.1.23)
;; WHEN: Sat Nov 14 16:18:10 2015
;; MSG SIZE  rcvd: 113
```

2.2 Vulnerabilidades

Com a utilização de protocolo UDP, como não há estabelecimento de conexão apenas um QID (i.e. *query ID*), um atacante pode tentar utilizar vários pacotes para simular respostas de um servidor DNS conhecido e eventualmente enviar um pacote com um QID válido para o cliente. Nesse momento, o atacante conseguirá inserir no *cache* do cliente um redirecionamento de uma página conhecida para uma outra controlada por ele. Essa é apenas uma vulnerabilidade conhecida como envenenamento de *cahe*. Esse e outros tipos de ataques são melhor explicados a seguir.

Envenenamento de Cache (Cache Poisoning): este é um dos mais conhecidos e difundidos ataques a segurança do DNS, ocorrendo quando um cliente realiza uma consulta para um determinado dominio, esta consulta é feita a uma zona DNS, porém esta zona não é fornecida por um servidor autoritativo os quais possuem os registros originais que associam aquele domínio a seu endereço de IP, assim um outro servidor de Cache DNS que encontra-se configurado nos resolvers do computador do cliente responde primeiro a requisição do cliente, levando o este a um servidor que não é o servidor de origem dodomínio, assim fornecendo informações forjadas ao cliente. Este cenario é comumente utilizado para levar o utilizador as denominadas páginas de Phishing, onde os dados submetidos pelo resolver são capturadas pelo hacker.

Ataque de negação de serviço ou DDoS (Distributed Denial of Service): Acontece quando um invasor tenta negar a disponibilidade de serviços DNS, através de consultas recursivas, consumindo todos os recursos como memória e processador do servidor, tornando-o indisponível.

Modificação de dados: Após ocupar uma rede usando DNS, O invasor tenta usar os endereços IP válidos, em pacotes IP criados pelo invasor, assim esses pacotes passam a ter a aparência de um endereço IP válido na rede. Normalmente isso é denominado falsificação de IP.

man-in-the-middle (Homem no meio): Nesse ataque, os dados trocados entre duas partes são interceptados, e assim essas informações são registrados e possivelmente alteradas pelo atacante, sem que as vítimas tomem conhecimento dessa ação, fazendo com que os envolvidos acreditem que estão se comunicando diretamente em uma conexão privada, quando de fato a conversa é controlada pelo atacante.

Capítulo 3

DNSSEC - *Domain Name System Security Extensions*

O DNSSEC é uma solução que expande a segurança do serviço DNS, permitindo validar a origem dos dados, assegurar que a informação recebida de um servidor DNS é autêntica, confirmando que a informação não foi alterada durante a passagem pelos vários "nodos" da Internet e confirmar a inexistência de um domínio.

O sistema DNSSEC introduz apenas alguns novos tipos de Registos nas zonas DNS, nomeadamente DNSKEY, RRSIG e DS, sendo compatível com servidores ou clientes que não tenham implementado o DNSSEC. As extensões de segurança DNSSEC utilizam criptografia assimétrica, permitindo que a verificação e troca de informação DNS entre o servidor e o cliente seja privada e entregue sem alterações, recorrendo a uma assinatura digital e um conjunto de chaves privadas e publicas.

O DNSSEC é baseado numa cadeia de confiança, onde inicialmente é validada a raiz ".", assinada digitalmente pela ICANN e IANA. Posteriormente a raiz assina a zona ".pt" utilizando a sua chave privada e anexa-lhe a chave publica, permitindo a partir desse momento que quando um servidor de nomes local fizer uma consulta relacionada com um domínio ".pt", este possa validar a autenticidade da raiz ".pt" recorrendo à chave publica. O processo é repetido, até chegar à zona final correspondente ao domínio que está a ser assinado e validado. As extensões de segurança do DNS são totalmente compatíveis com IPv6, uma vez que apenas é validada a integridade dos registos.

3.1 Configuração do Servidor DNS Seguro

Passos para a configuração segura

Exemplo de código

```
sudo aptitude install tshark

sudo aptitude install texlive

sudo aptitude install texlive-lang-portuguese
```

3.2 Requisições seguras

Cada zona no DNSSEC tem um par de assinatura de chave(ZSK): a parte privada da chave assina digitalmente cada RRset na zona, enquanto que a porção pública verifica a assinatura. Para habilitar o DNSSEC, um operador de zona cria assinaturas digitais para cada RRset usando o ZSK privado e os armazena em seu servidor DNS como registros RRSIG. Isso é como dizer: "Estes são os meus registros de DNS, eles vêm do meu servidor, e eles devem ser assim", como mostra a figura 3.4.

Figura 3.1: Chave de assinatura para RRSet.

Quando um cliente DNSSEC solicita um tipo de registro em particular (por exemplo, AAAA), o servidor DNS também retorna o RRSIG correspondente. O cliente pode em seguida puxar o registro DNSKEY contendo a ZSK pública a partir do servidor de nomes. Juntos, o RRset, RRSIG, e ZSK público podem validar a resposta.

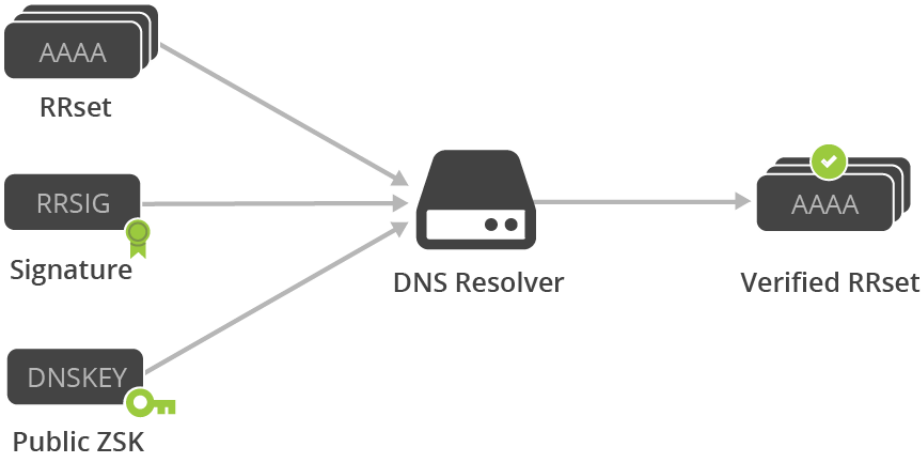

Figura 3.2: Validação de registros RRSIG.

Precisa-se de uma forma de validar a ZSK pública, uma vez que não se sabe se houve alteração. Para isso, verifica-se a "validade da validade".

Além de uma chave de assinatura de zona, servidores DNSSEC também possuem uma chave de assinatura (KSK). O KSK valida o registro DNSKEY exatamente da mesma maneira como nosso ZSK garantiu o resto de nossas RRsets na seção anterior: Ele assina a ZSK público (que é armazenado em um registro DNSKEY), criando uma RRSIG para o DNSKEY.

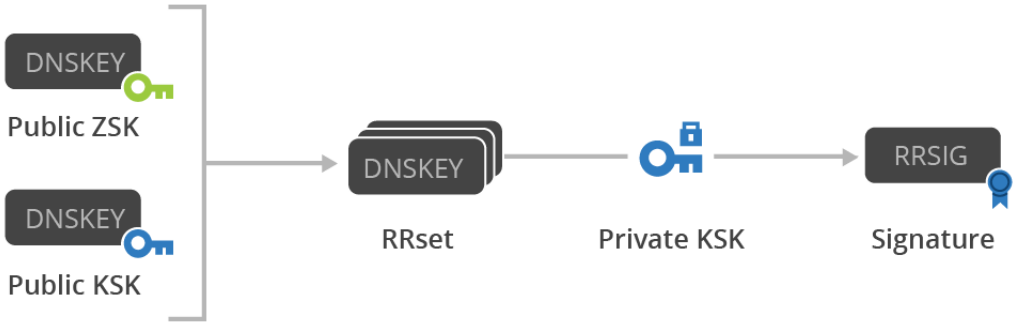

Figura 3.3: Verificação da validade da chave.

Assim como o ZSK público, o servidor de nomes publica o KSK pública em outro registro DNSKEY, que nos dá a RRset DNSKEY mostrado acima. Tanto o KSK e ZSK pública público são assinados pelo KSK privada. Clientes pode então utilizar a KSK pública para validar a ZSK pública.

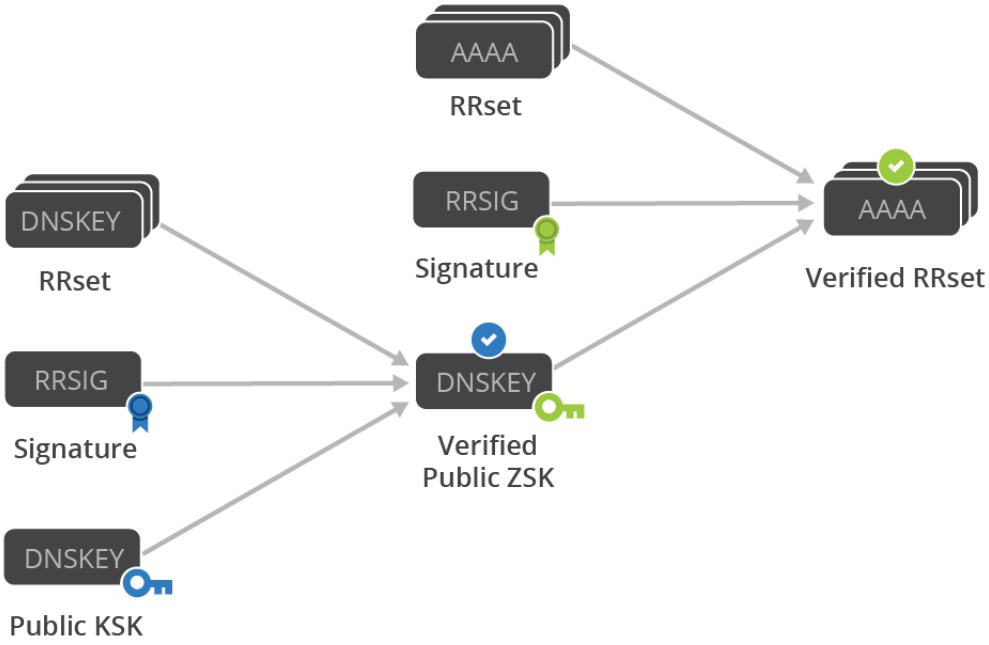

Figura 3.4: Verificação completa da autenticidade do registro.

Resumidamente, a resolução de nome de forma segura é realizada pelo cliente da seguinte forma:

- Solicite o RRset desejado, que também retorna o registro RRSIG correspondente.
- Solicite a ZSK e KSK pública, que também retornam o RRSIG para o DNSKEY RRset.
- Verifique se o RRSIG do RRset solicitado com a ZSK público.
- Verifique se o RRSIG do DNSKEY RRset com o KSK público.

É importante ressaltar que o serviço de DNSSEC configurado neste trabalho não provê registro no *delegation signer* (DS) para uma KSK, isso devido ao custo atrelado à aquisição de um registro em servidores autorizados. Entretanto, o processo de configuração do DS foi estudado e pode ser encontrado por completo na *wiki* do projeto no *github*[1].

[1] documentação sobre configuração do DS: https://github.com/ArthurJahn/FRC-Final/wiki.

Capítulo 4

Resultados

4.1 Implementação da Solução

O ecosistema desenvolvido nesse projeto, tem como base a utilização de máquinas *Vagrant*, que provê serviços de virtualização de ambientes para facilitar configuração de diferentes ambientes para vários sistemas operacionais. O provedor de máquinas virtuais utilizado neste trabalho foi o VirtualBox. Para a configuração do ambiente, é necessário a instalação do vagrant [1] e do VirtualBox [2].

Foram desenvolvidos *scripts* para inicialização das máquinas e configurações dos serviços, todos os passos para subir o ambiente completo foi passado para o script executável *quick-start* que sobe as máquinas e gera um arquivo de *log* para uma requisição a um serviço DNS seguro provido pelo servidor DNS configurado. Todo o ambiente pode ser desligado por meio do *script* de *quick-exit* que destrói os ambientes criados e remove arquivos temporários.

Execute os seguintes passos para subir o ambiente: Com as dependências instaladas, execute o arquivo quick-start encontrado na pasta vagrant. Esse executável irá inicializar um servidor DNS seguro configurando a zona *redesfga.com*, uma servidor apache para onde a zona configurada aponta e um cliente que realizará uma requisição de resolução de nome ao servidor DNS. O resultado da requisição é encontrado no arquivo dnssec.log na pasta vagrant, e mostra todo o processo executado para fazer a requisição segura do domínio passado.

4.2 Considerações

O DNS é um serviço crucial para o funcionamento da internet atualmente. Entretanto não foi idealizado provendo serviços de segurança, o que gera várias vulnerabilidades que são utilizadas para atacantes gerarem comportamentos idesejados da rede.

[1] Vagrant disponível em: https://www.vagrantup.com/
[2] VirtualBox disponível em: https://www.virtualbox.org/

Com esse trabalho foi possível identificar vulnerabilidades do serviço de DNS, configurar extensões de segurança DNSSEC, verificar os passos para a aquisição de respostas de requisições DNS de forma segura e compreender como os serviços de verificação de autenticidade da internet funcionam e quem é capaz de prover tais serviços.

Nesse sentido todos os objetivos propostos foram alcançados e o aprendizado obtido por meio da atividade prática mostrou a complexidade das operações relacionadas ao provimento de serviços seguros na internet.