## **DNSSEC**

DNSSEC (Domain Name System Security Extensions) — набор расширений протокола DNS, позволяющих минимизировать атаки, связанные с подменой DNS - адреса при разрешении доменных имен.

Для чего нужен?

Злоумышленник может изменить ответ DNS или отравить кэш DNS и перевести пользователя на вредоносный сайт с допустимым доменным именем в адресной строке.

Таким образом злоумышленники получают доступ к паролям, номерам кредитных карт и другой конфиденциальной информации. Пользователь может даже не заметить подмены - запись в строке браузера и сам сайт в точности такие, какими их и ожидает увидеть пользователь.

Принцип работы DNSSEC тот же, что и у цифровой подписи. То есть закрытым ключом подписываем, открытым сверяем.

DNSSEC использует два типа ключей — одним подписывается зона (ZSK, zone signing key), другим подписывается набор ключей (KSK, key signing key).

## ZSK

С помощью этого ключа подписываются все наборы записей в зоне (RRSET), кроме точек делегирования.

#### KSK

Этим ключом подписывается набор DNSKEY записей. Кроме того, от открытой части KSK берется хэш, который в дальнейшем отправляется в родительскую зону.

О подробном механизме работы DNSSEC и ключей можно прочитать статью на хабре (ссылка приведена ниже).

# Настройка среды

Доменное имя: example.com

### **Master Nameserver:**

IP Address: 1.1.1.1

Hostname: master.example.com

OS: Debian 7

## **Slave Nameserver:**

IP Address: 2.2.2.2

Hostname: slave.example.com

OS: CentOS

## Расположение файлов

Debian/Ubuntu Сервис: bind9

Основной файл конфигурации: /etc/bind/named.conf.options

Файл имен зон: /etc/bind/named.conf.local

Расположение файла зон по умолчанию: /var/cache/bind/

CentOS/Fedora Сервис: named

Основной файл конфигурации и имен зон: /etc/named.conf

Расположение файла зон по умолчанию: /var/named/

# **DNSSEC Master Configuration**

# Включение DNSSEC Добавить директивы, внутри options { } nano /etc/bind/named.conf.options dnssec-enable yes; dnssec-validation yes; dnssec-lookaside auto; Перейти к расположению файлов вашей зоны cd /var/cache/bind Создание ключа подписи зоны (ZSK) dnssec-keygen -a NSEC3RSASHA1 -b 2048 -n ZONE example.com Генерация может занять много времени Вывод: root@master:/var/cache/bind# dnssec-keygen -a NSEC3RSASHA1 -b 2048 -n ZONE example.com Generating key pair.....+++ Kexample.com.+007+40400 Создание ключа подписи (KSK) dnssec-keygen -f KSK -a NSEC3RSASHA1 -b 4096 -n ZONE example.com Вывод: root@master:/var/cache/bind# dnssec-keygen -f KSK -a NSEC3RSASHA1 -b 4096 -n ZONE example.com Generating key pair....+ + .....++ Kexample.com.+007+62910 В каталоге будет 4 ключа — приватные/публичные пары ZSK и KSK. Теперь надо добавить открытые ключи, которые содержат запись DNSKEY в файл зоны. Скрипт:

for key in `ls Kexample.com\*.key`
do
echo "\\$INCLUDE \$key">> example.com.zone
done

## Подпись зоны:

dnssec-signzone -3 <salt> -A -N INCREMENT -o <zonename> -t <zonefilename>

## Вывод:

root@master:/var/cache/bind# dnssec-signzone -A -3 \$(head -c 1000 /dev/random | sha1sum | cut -b 1-16) -N INCREMENT -o example.com -t example.com.zone Verifying the zone using the following algorithms: NSEC3RSASHA1.

Zone signing complete:

Algorithm: NSEC3RSASHA1: KSKs: 1 active, 0 stand-by, 0 revoked

ZSKs: 1 active, 0 stand-by, 0 revoked

example.com.zone.signed

Signatures generated: 14
Signatures retained: 0
Signatures dropped: 0
Signatures successfully verified: 0
Signatures unsuccessfully verified: 0
Signing time in seconds: 0.046
Signatures per second: 298.310
Runtime in seconds: 0.056

head -c 1000 /dev/random | sha1sum | cut -b 1-16

Создастся новый файл с именем example.com.zone.signed, который содержит записи RRSIG для каждой записи DNS.

### Загрузка подписанной зоны

nano /etc/bind/named.conf.local

## Перезагрузить bind

service bind9 reload

<sup>\*</sup>вместо <salt> используйте команду, она выдаст рандомную строку из 16 символов

# **DNSSEC Slave Configuration**

Ha slave серверах необходимо только включить DNSSEC и изменить местоположение файла зоны.

## Редактирование файла конфигурации:

```
nano /etc/named.conf

Внутри options { } добавить эти строки:

dnssec-enable yes;
dnssec-validation yes;
dnssec-lookaside auto;

Изменить параметр file внутри zone { }

zone "example.com" IN {
  type slave;
  file "example.com.zone.signed";
  masters { 1.1.1.1; };
  allow-notify { 1.1.1.1; };
};
```

## Перезапуск bind

service named reload

# Создание DS записей в регистраторе

Когда мы запустили dnssec-signzone отдельно от .signed файла зоны, был создан файл dsset-example.com, содержащий записи DS

root@master:/var/cache/bind# cat dsset-example.com.

example.com. IN DS 62910 7 1

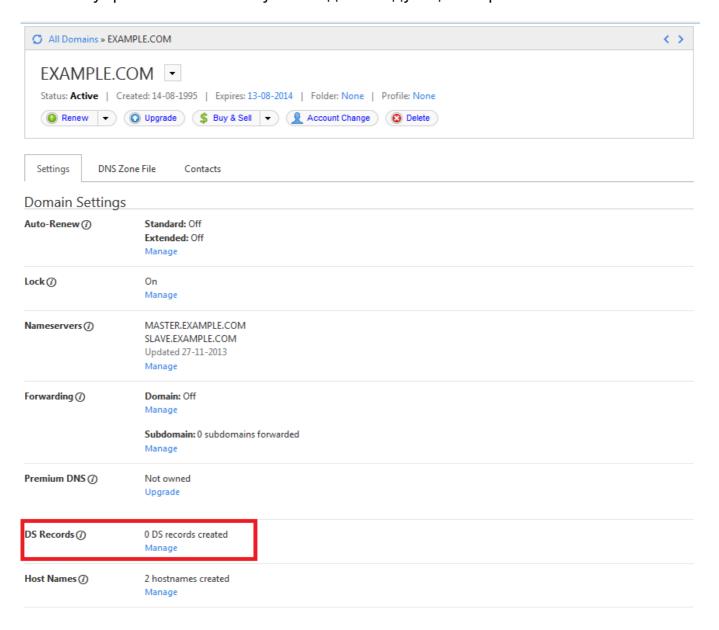
1D6AC75083F3CEC31861993E325E0EEC7E97D1DD

example.com. IN DS 62910 7 2

198303E265A856DE8FE6330EDB5AA76F3537C10783151AEF3577859F FFC3F59D

Они должны быть введены в панели управления регистратора доменов. digest

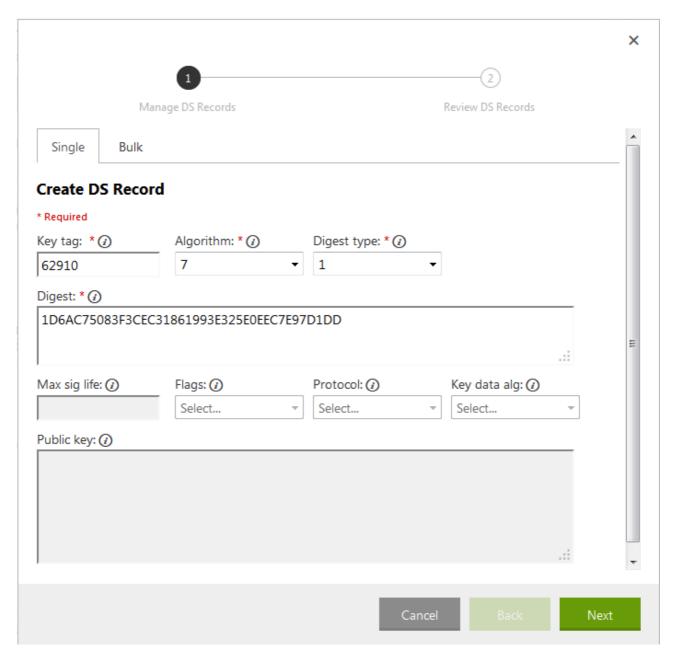
Панель управления GoDaddy выглядит следующим образом:



## DS запись 1:

Key tag: 62910 Algorithm: 7 Digest Type: 1

Digest: 1D6AC75083F3CEC31861993E325E0EEC7E97D1DD

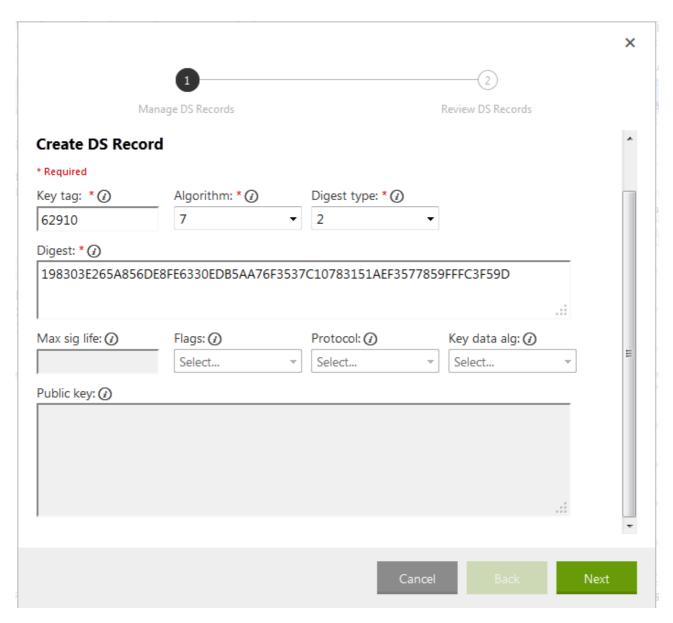


## DS запись 2:

Key tag: 62910 Algorithm: 7 Digest Type: 2

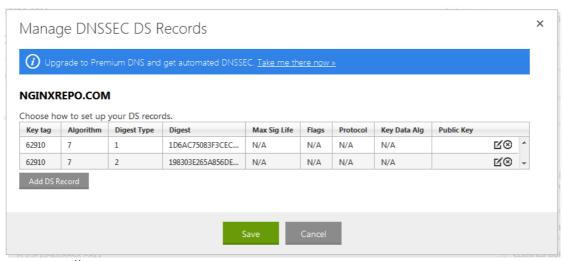
Digest:

198303E265A856DE8FE6330EDB5AA76F3537C10783151AEF3577859FFFC3F59D



<sup>\*</sup>вторая запись DS в dsset-example.com. файле содержала пробел в digest, но при вводе ее в форму вы должны опустить его

## Next → Finish → Save



## Сохранение займет несколько минут

Теперь можно проверить, работает ли DNSSEC, с помощью одного из сервисов:

http://dnssec-debugger.verisignlabs.com

http://dnsviz.net/

## Analyzing DNSSEC problems for example.com

-	<ul> <li>✓ Found 2 DNSKEY records for .</li> <li>✓ DS=19036/SHA1 verifies DNSKEY=19036/SEP</li> <li>✓ Found 1 RRSIGs over DNSKEY RRset</li> <li>✓ RRSIG=19036 and DNSKEY=19036/SEP verifies the DNSKEY RRset</li> </ul>
com	<ul> <li>✓ Found 1 DS records for com in the . zone</li> <li>✓ Found 1 RRSIGs over DS RRset</li> <li>✓ RRSIG=59085 and DNSKEY=59085 verifies the DS RRset</li> <li>✓ Found 2 DNSKEY records for com</li> <li>✓ DS=30909/SHA256 verifies DNSKEY=30909/SEP</li> <li>✓ Found 1 RRSIGs over DNSKEY RRset</li> <li>✓ RRSIG=30909 and DNSKEY=30909/SEP verifies the DNSKEY RRset</li> </ul>
example.com	<ul> <li>Found 2 DS records for example.com in the com zone</li> <li>Found 1 RRSIGs over DS RRset</li> <li>RRSIG=22625 and DNSKEY=22625 verifies the DS RRset</li> <li>Found 2 DNSKEY records for example.com</li> <li>DS=62910/SHA256 verifies DNSKEY=62910/SEP</li> <li>Found 2 RRSIGs over DNSKEY RRset</li> <li>RRSIG=40400 and DNSKEY=40400 verifies the DNSKEY RRset</li> <li>example.com A RR has value 93.184.216.119</li> <li>Found 1 RRSIGs over A RRset</li> <li>RRSIG=40400 and DNSKEY=40400 verifies the A RRset</li> </ul>

#### P. S.

Каждый раз, когда редактируете зону, добавляя или удаляя записи, она должна быть подписана.

Скрипт для того, чтобы не вводить каждый раз команды:

root@master# nano /usr/sbin/zonesigner.sh

#!/bin/sh
PDIR=`pwd`
ZONEDIR="/var/cache/bind" #location of your zone files
ZONE=\$1
ZONEFILE=\$2
DNSSERVICE="bind9" #On CentOS/Fedora replace this with "named"
cd \$ZONEDIR
SERIAL=`/usr/sbin/named-checkzone \$ZONE \$ZONEFILE | egrep -ho '[0-9]{10}'`
sed -i 's/'\$SERIAL'/'\$((\$SERIAL+1))'/' \$ZONEFILE
/usr/sbin/dnssec-signzone -A -3 \$(head -c 1000 /dev/random | sha1sum | cut -b 116) -N increment -o \$1 -t \$2
service \$DNSSERVICE reload
cd \$PDIR

root@master# chmod +x /usr/sbin/zonesigner.sh

Каждый раз, когда хотите добавить или удалить записи, изменяйте example.com.zone и HE изменяйте .signed файл. После редактирования запустите скрипт:

root@master# zonesigner.sh example.com example.com.zone

Подготовил: Ваулин Данил, 18204

#### Ресурсы:

https://www.digitalocean.com/community/tutorials/how-to-setup-dnssec-on-an-authoritative-bind-dns-server--2

https://ru.wikipedia.org/wiki/DNSSEC

https://habr.com/ru/post/120620/