Курс «Администратор Linux»

DNS + DHCP

Занятие # 22

Дмитрий Молчанов
Григорий Ожегов
Алексей Цыкунов
Леонид Альбрехт

BOOTP

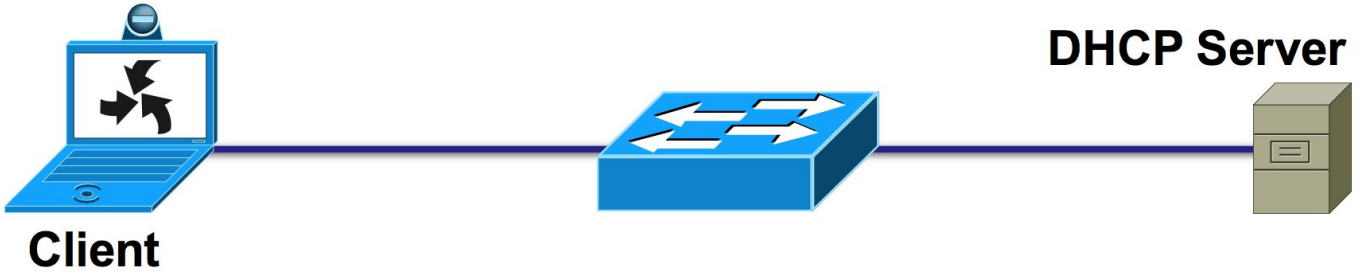
BOOTP - Bootstrap Protocol — сетевой протокол для бездисковых машин позволяющий компьютерам автоматически получать IP, gateway, DNS и другие параметры, необходимые для работы в сети TCP/IP. Используется во время загрузки, файл с конфигурацией грузится в память и запускается позже для настройки.

Раньше требовал ручной прописки MAC адресов клиентов на сервере.

DHCP

DHCP - Dynamic Host Configuration Protocol — сетевой протокол, позволяющий компьютерам автоматически получать IP, gateway, DNS и другие параметры, необходимые для работы в сети TCP/IP

DHCP



DHCP Discover (Broadcast)

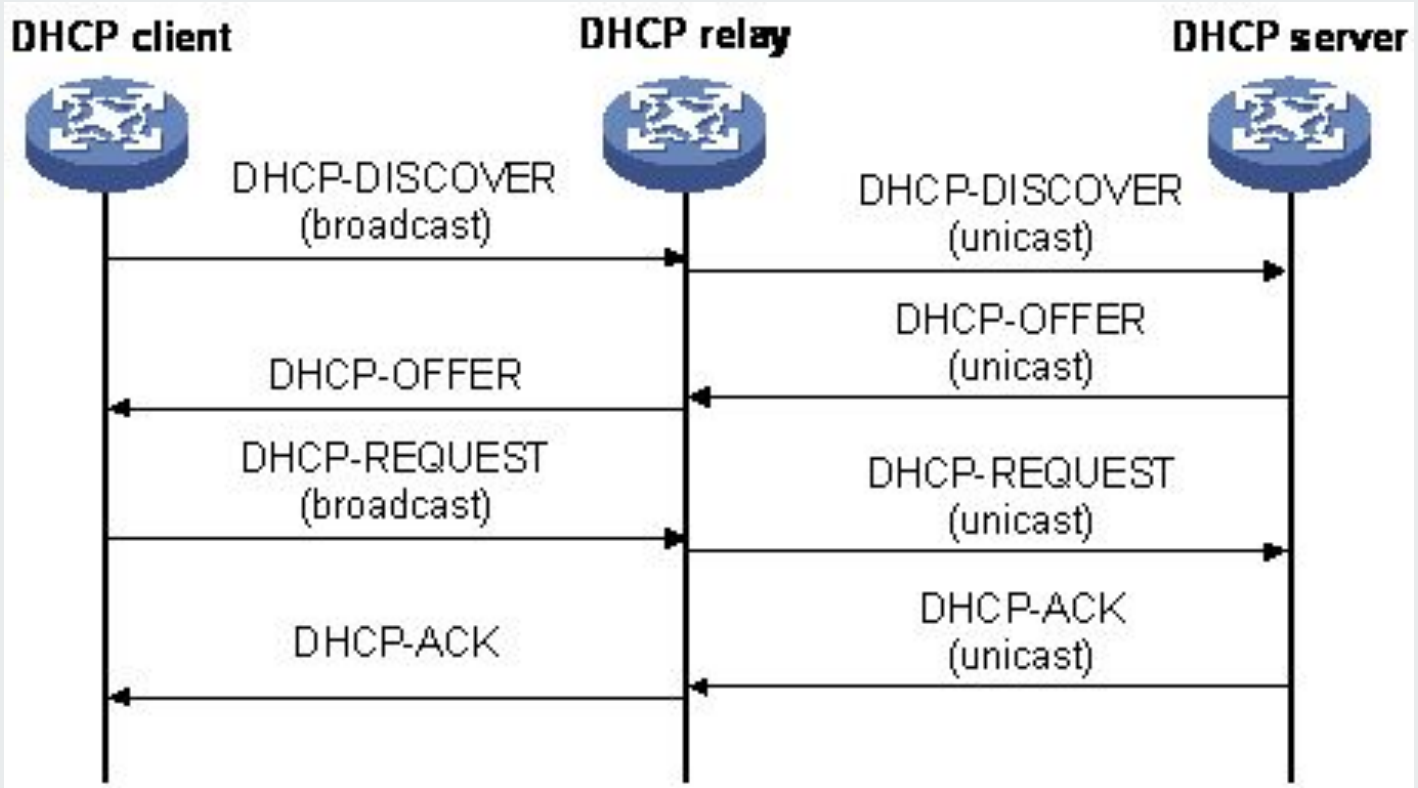
DHCP Offer (Unicast/Broadcast)

DHCP Request (Broadcast)

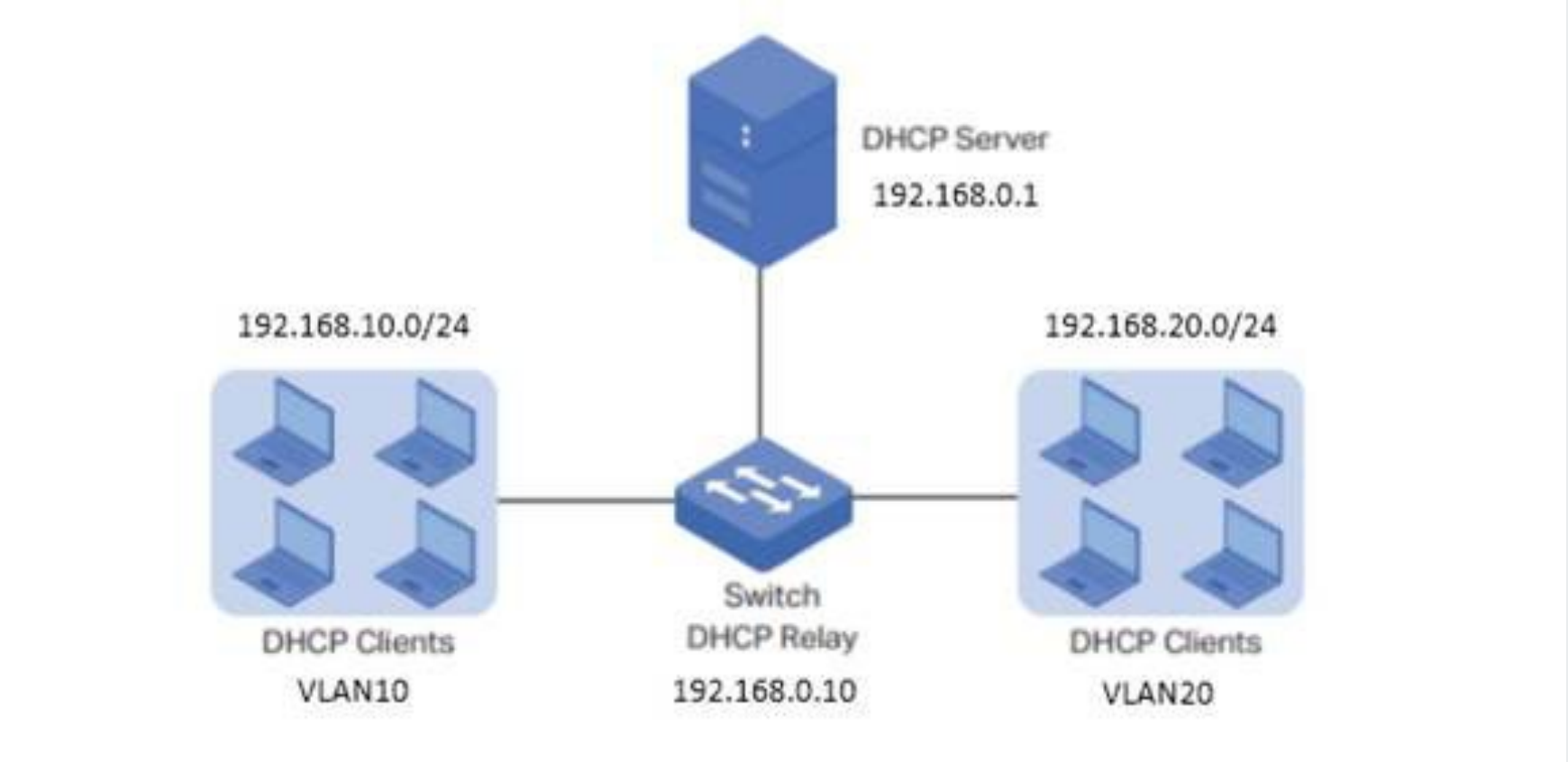
DHCP Ack (Unicast/Broadcast)

See: DHCP defined by RFC 2131

DHCP: relay


DHCP client
DHCP relay
DHCP server
DHCP-DISCOVER
(broadcast)
DHCP-DISCOVER
(unicast)
DHCP-OFFER
(unicast)
DHCP-OFFER
DHCP-REQUEST
(broadcast)
DHCP-REQUEST
(unicast)
DHCP-ACK
(unicast)
DHCP-ACK

DHCP: relay


DHCP Server
192.168.0.1
192.168.10.0/24
192.168.20.0/24
Switch
DHCP Relay
192.168.0.10
DHCP Clients
VLAN10
DHCP Clients
VLAN20

DHCP: сообщения

DHCPDISCOVER - широковещательный запрос с целью обнаружения DHCP сервера

DHCPOFFER - ответ сервера с предложением о варианте конфигурации

DHCPREQUEST - клиент соглашается на предложенный сервером вариант

DHCPACK - подтверждение от сервера о выделенной конфигурации

DHCP

DHCPNAK - сервер не подтвержает реквест клиента или клиентский lease истек,

DHCPDECLINE - клиент сообщает серверу, что запрашиваемый адрес используется

DHCPRELEASE - клиент сообщает серверу что он закончил использовать адрес и освобождает lease

DHCPINFORM - Запрос на конфигурацию без выделения IP адреса

DHCP: conf

```
  subnet 239.252.197.0 netmask 255.255.255.0
{
     range 239.252.197.10 239.252.197.107;
     default-lease-time 600;
     max-lease-time 7200;
   }

pool {
    option domain-name-servers
bogus.example.com;
    max-lease-time 300;
    range 10.0.0.200 10.0.0.253;
    allow unknown-clients;
  }
```

```
host haagen {
     hardware ethernet 08:00:2b:4c:59:23;
     fixed-address 239.252.197.9;
     filename "/tftpboot/haagen.boot";
   }
```

DHCP: опции

option subnet-mask *ip-address*;
Задает маску подсети клиента. В случае если параметр не указан, dhcpd использует значение маски подсети в которой находится используемый сервером адрес.
option routers *ip-address* [**,** *ip-address*...]**;**
Список IP адресов маршрутизаторов для клиентской сети.
option ntp-servers *ip-address* [, *ip-address*...]**;**
Список серверов NTP
option domain-name-servers *ip-address* [**,** *ip-address*...]**;**
Список DNS серверов доступных клиенту.

DHCP: leases

Файл **dhcpd.leases** - база данных(БД), хранящая информацию выделенных демоном DHCP адресам. Каждая запись включает в себя единственный IP-адрес, тот что был выделен клиенту. Инструкции в скобках определяют кому и на какой срок выдан адрес.

DHCP: leases

```
lease 10.1.2.48 {
  starts 2 2017/07/11 03:50:22;
  ends 3 2024/05/15 03:50:22;
  tstp 3 2024/05/15 03:50:22;
  cltt 2 2017/07/11 03:50:22;
  binding state active;
  next binding state free;
  hardware ethernet 00:1c:c0:c3:13:68;
  client-hostname "bl-celeron";
}
```

PXE

- **pxe — Загрузка PXE-загрузчика**. Прошивкой PXE, встроенной в сетевую карту, выполняется загрузка загрузчика **pxelinux**.
- **dhcp1 — Получение IP-адреса и адреса TFTP-сервера.** Загрузчик pxelinux делает DHCP-запрос и с его помощью получает IP-адрес и адрес TFTP-сервера. По умолчанию адрес TFTP-сервера равен адресу DHCP-сервера.
- **tftp — Обращение к TFTP-серверу для получения ядра Linux.** Загрузчик pxelinux обращается к TFTP-серверу и запрашивает у него ядро Linux (и, при необходимости, образ initrd).
- **kernel — Запуск ядра Linux.** После того как ядро Linux загружено, управление передаётся ему.

PXE

- **dhcp2 — Получение IP-адреса DHCP.** Ядро Linux делает запрос DHCP, с помощью которого получает свой IP-адрес, адрес NFS-сервера, на котором находится корневая файловая система, а также путь к местоположению этой файловой системы на диске.
- **nfs — Монтирование корневой файловой системы**. Корневая файловая система монтируется.
- **init — Вызов процесса init.** На корневой файловой системе находится файл /sbin/init, которому передаётся управление.

DNS

DNS - Domain Name Service. Один из важнейших сервисов в сетях вообще и Интернет в целом. Без DNS Интернет в том виде в котором он есть сейчас - невозможен.
Для своей работы DNS использует протокол udp и порт 53.

Здесь и далее мы будем говорить только о ISC-BIND как о ПО для организации сервера DNS. В то время как это не единственное ПО для этих целей. Есть еще powerdns, dnsmasq

DNS: Предназначение

Основное предназначение DNS - сопоставление адресов и имен.

Сопоставления бывают:

- Прямые: Имя -> IP-адрес
- Обратные: IP-адрес -> Имя

Так же DNS можно использовать для

- хранения дополнительной информации:
  - Архитектуры сети
  - Конфигурация приложения
  - Текстовая информация которая используется антиспам сервисами
- Балансировки нагрузки

DNS: Стуктура

DNS

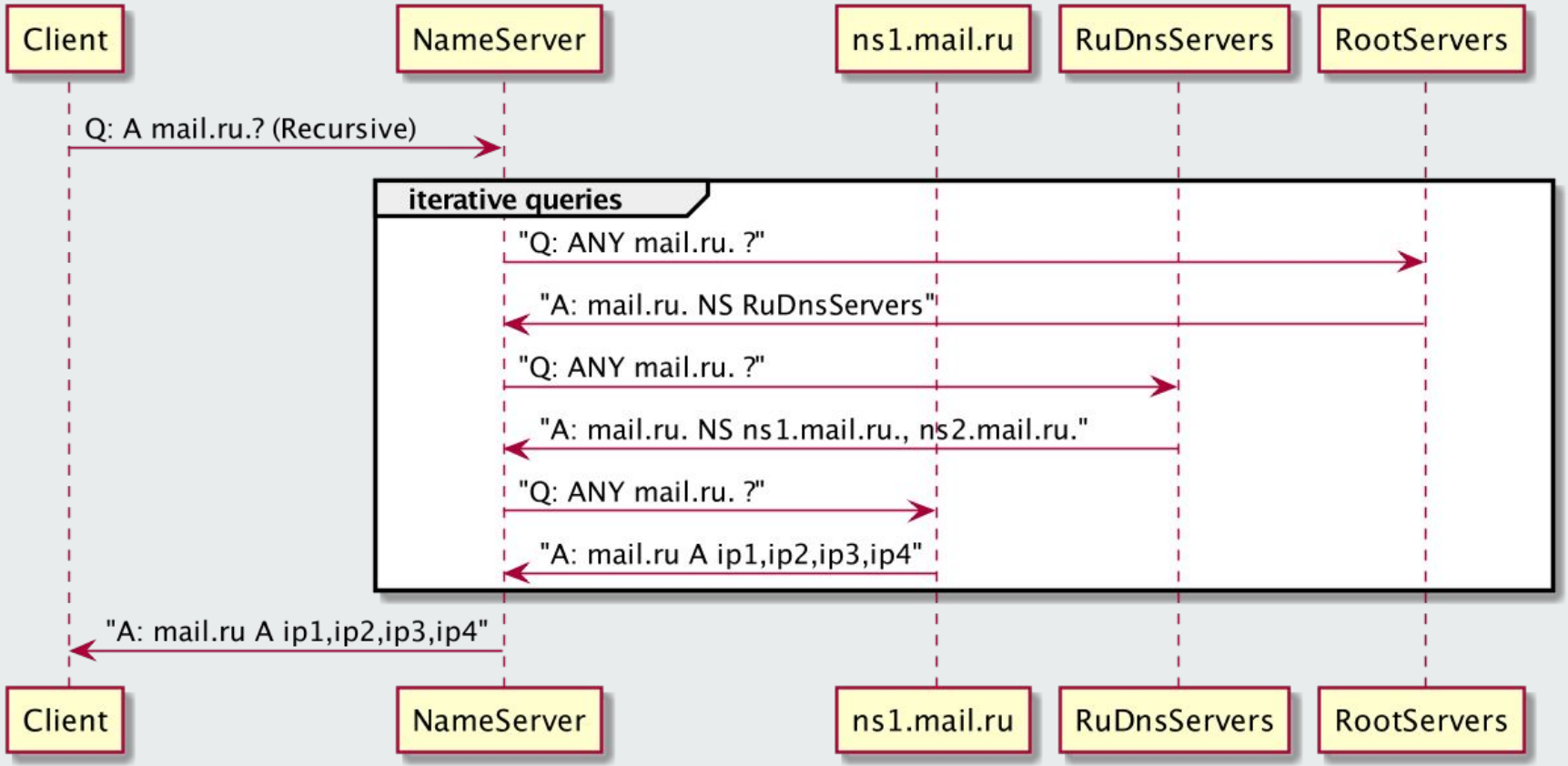
FQDN

FQDN - Fully qualified domain name. Полностью указанное доменное имя - от корневого домена. Ключевым индикатором fqdn является точка в конце имени.

FQDN www.otus.ru состоит из:

- домена 3-го уровня **www**, входящего в состав **otus.ru.**
- домена 2-го уровня **otus**, входящего в состав **ru**
- домена 1-го уровня **ru,** который входит в корневой домен - он не имеет названия и обозначается в FQDN точкой - “.”

DNS: Запросы


Client
NameServer
ns1.mail.ru
RuDnsServers
RootServers
Q: A mail.ru.? (Recursive)
iterative queries
"Q: ANY mail.ru. ?"
"A: mail.ru. NS RuDnsServers"
"Q: ANY mail.ru. ?"
"A: mail.ru. NS ns1.mail.ru., ns2.mail.ru."
"Q: ANY mail.ru. ?"
"A: mail.ru A ip1,ip2,ip3,ip4"
"A: mail.ru A ip1,ip2,ip3,ip4"
Client
NameServer
ns1.mail.ru
RuDnsServers
RootServers

DNS: типы серверов

- Главный (primary/master) - авторитативный, хранит главную копию файла зоны
- Вторичный (slave) - получает копию файла зоны от главного или другого вторичного серверов
- Кэширующий - кэширует ответы на запросы пользователя

DNS: Записи

Запись DNS обладает следующими ключевыми атрибутами:

- Имя
- TTL
- Класс
- Тип
- Значение

Атрибут TTL регулирует время жизни записи в DNS-кэше.
Некоторые записи могут иметь в значении массивы данных.

Типы записей

A	ipv4-адрес соответствующий имени.
AAAA	ipv6-адрес соответствующий имени.
CNAME	(Canonical NAME) имя соответствующее имени.
MX	Массив (prio, name) почтовых серверов для домена.
TXT (SPF)	Некая текстовая информация соответствующая имени
SOA	Start Of Authority - ключевая запись домена
NS	имя name-server’a для домена
PTR	PoinTeR - имя соответсвующее ip-адресу (только для in-addr.arpa и ip6.arpa)
SRV	Описание сервиса

SOA

SOA - единственная запись, которая уникальна для домена (именованного пространства имен), остальные записи могут встречаться более одного раза.
Описывает “точку отсчета” для домена:

- Имя первичного авторитетного name server’а
- Адрес администратора домена, в этом месте “@” заменен на “.” поэтому имя в адресе эл.почты лучше иметь без знаков препинания, например hostmaster
- **Serial** - порядковый номер версии файла. Это отправная точка для решения о синхронизации между серверами. это целое число (int) 4 байта, знаковое(?) - велика вероятность “переполнения” - надо быть аккуратными.

Пример ошибок файла зоны (домена).

```
$TTL 3600
@       600 IN      SOA     ns.domain.tld. hostmaster.domain.tld. (
                                0 ; Serial
                                28800   ; Refresh (8h)
                                7200    ; Retry (2h)
                                604800  ; Expire (7day)
                                86400 ); NegTTL (1day)

                60 IN      NS      ns1.domain.tld
                60 IN      NS      ns2.domain.tld.
                60 IN      MX      10 mx.domain.tld


@               60 IN      A  1.2.4.5
;this record was added
mx          60 IN A     1.2.4.5
imap        60 IN A     1.2.4.5
smtp        60 IN A     1.2.4.5
pop3        60 IN A     1.2.4.5
_imap._tcp  60 IN SRV   5 0 143 imap.
_pop3._tcp. 60 IN SRV   5 0 110 pop3
_submission._tcp  60 IN SRV 5 0 25 smtp
```

Пример файла зоны(домена).

```
ORIGIN domain.tld.
$TTL 3600
@       600 IN      SOA     ns.domain.tld. hostmaster.domain.tld. (
                                2017113001 ; Serial
                                28800   ; Refresh (8h)
                                7200    ; Retry (2h)
                                604800  ; Expire (7day)
                                86400 ; NegTTL (1day)
                                )

                60 IN      NS      ns1.domain.tld.
                60 IN      NS      ns2.domain.tld.
                60 IN      MX      10 mx.domain.tld.
                           MX      20 mx1


@           60 IN  A    1.2.4.5
mx          60 IN A     1.2.4.5
mx1         60 IN A     1.2.4.6
imap        60 IN A     1.2.4.5
smtp        60 IN A     1.2.4.5
pop3        60 IN A     1.2.4.5
www            IN CNAME h1
_imap._tcp  60 IN SRV   5 0 143 imap
_pop3._tcp  60 IN SRV   5 0 110 pop3
_submission._tcp  60 IN SRV 5 0 25 smtp
```

DNS: обратное разрешение имен

в ip-адресе общая и частная части располагаются слева направо, а в доменном имени - наоборот.

1. ip-адрес надо развернуть.
   `192.168.10.1 -> 1.10.168.192.`

В таком виде он станет соответствовать “направлению” доменных имен.

2. централизованный домен для хранения информации об ip-адресах:
   - in-addr.arpa - для ipv4
   - ip6.arpa - для ipv6

3. - тип записи в котором хранится имя соответствующее адресу - PTR.

DNS: обратное разрешение имен

Для, того, чтобы узнать какое имя соответствует адресу 1.2.3.4 небоходимо:

- “развернуть” адрес: 1.2.3.4 -> 4.3.2.1
- сделать запрос PTR для записи 4.3.2.1.in-addr.arpa
- В ответ может быть получено ноль или более записей типа PTR которые будут говорить какие имена указывают на этот адрес.

DNS: Репликация

В протокол DNS встроена возможность репликации зон. Это происходит с помощью запросов

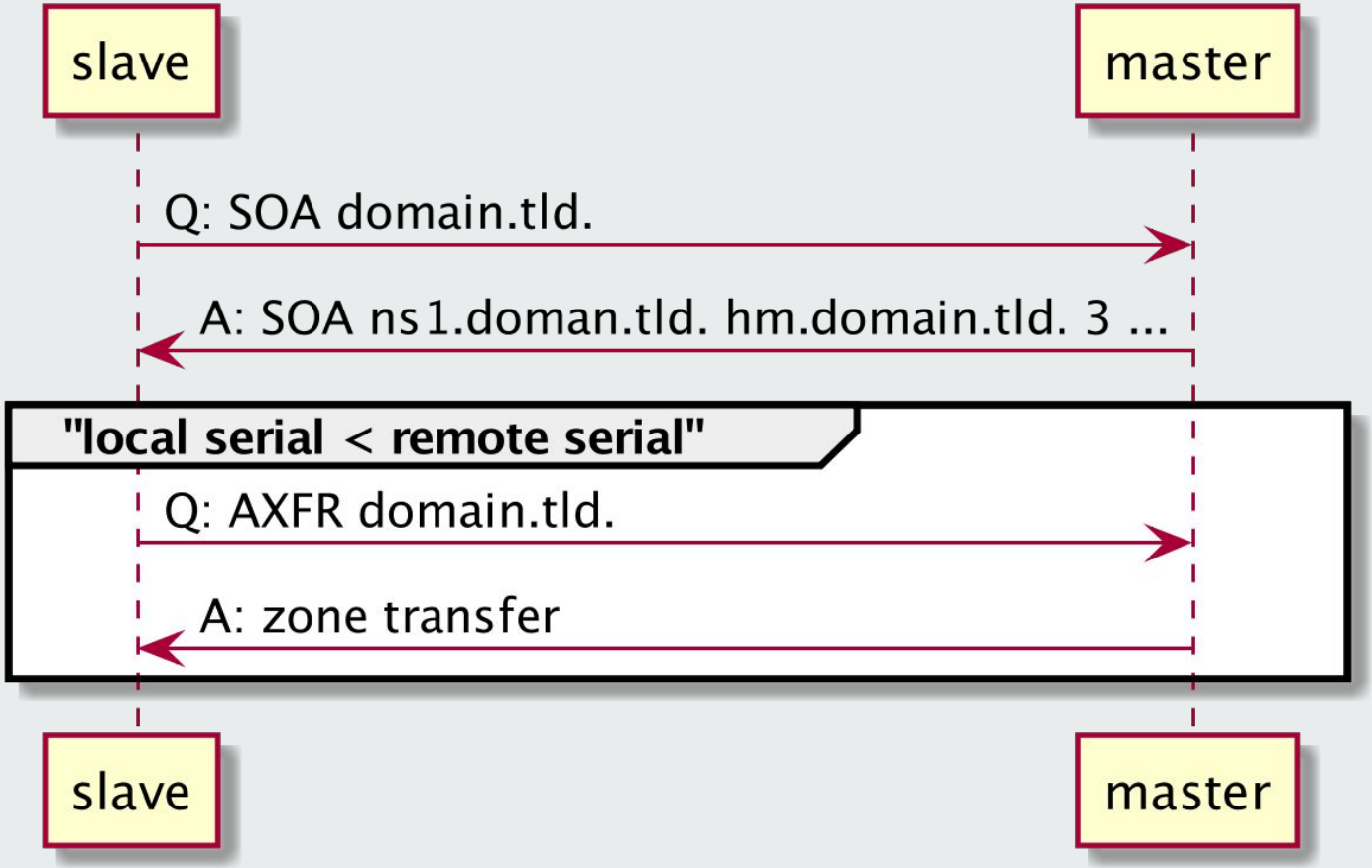
- AXFR (transfer all records)
- IXFR (incremental transfer).

Для репликации DNS использует протокол tcp, т.к. важна гарантия доставки.

Репликация происходит только при одном условии - **local_serial < remote_serial**.
Проверка serial является частью процесса репликации.
Репликация может быть инициирована следующими событиями:

- ручной запуск (reload)
- истечение timeout указанного в SOA
- NOTIFY-запрос.

DNS: репликация



Аспекты безопасности

- ограничение адресов которым разрешены рекурсивные запросы (anti DDoS).
- ограничение адресов которые могут делать запросы (per zone).
- ограничение адресов которые могут присылать NOTIFY.
- ограничение адресов с которых могут приходить обновления.

BIND (ACL)

- **Acl (access control list)** - позволяет задать именованный список сетей. Формат раздела: **acl "имя_сети" {ip; ip; ip; };**

BIND: раздел Options

- **allow-query {*список_ip*}** - Разрешает ответы на запросы только из *список_ip*. При отсутствии - сервер отвечает на все запросы.
- **allow-recursion {*список_ip*}** - На запросы из список_ip будут выполняться рекурсивные запросы. Для остальных - итеративные. Если не задан параметр, то сервер выполняет рекурсивные запросы для всех сетей.
- **allow-transfer {*список_ip*}** - Указывает список серверов, которым разрешено брать зону с сервера (в основном тут указывают slave сервера)
- **directory /path/to/work/dir** - указывает абсолютный путь к рабочему каталогу сервера. Этот оператор допустим только в разделе options.

BIND: раздел Options

- **forwarders {*ip порт, ip порт...*}** - указывает адреса хостов и если нужно порты, куда переадресовывать запросы (обычно тут указываются DNS провайдеров ISP).
- **forward *ONLY*** или **forward *FIRST*** - параметр **first** указывает, DNS-серверу пытаться разрешать имена с помощью DNS-серверов, указанных в параметре forwarders, и лишь в случае, если разрешить имя с помощью данных серверов не удалось, то будет осуществлять попытки разрешения имени самостоятельно.
- **notify *YES*|*NO*** - *YES* - уведомлять slave сервера об изменениях в зоне, *NO* - не уведомлять.
- **recursion *YES*|*NO*** - *YES* - выполнять рекурсивные запросы, если просит клиент, *NO* - не выполнять (только итеративные запросы). Если ответ найден в кэше, то возвращается из кэша. (может использоваться только в разделе Options)

BIND: раздел Zones

Формат раздела: **zone {*операторы_раздела_zone*};**

Операторы, которые наиболее часто используются:

- **allow-update {*список_ip*}** - кому разрешено динамически обновлять данную зону.
- **file *"имя_файла*"** - указывает путь файла параметров зоны (должен быть расположен в каталоге, определенном в разделе **options** оператором **directory**)
- **masters {*список_ip*}** -указывает список мастер-серверов. (допустим только в подчиненных зонах)
- **type "*тип_зоны*"** - указывает тип зоны, описываемой в текущем разделе,тип_зоны может принимать следующие значения:
  - *forward* - указывает зону переадресации
  - *hint* - указывает вспомогательную зону (информация о корневых серверах)
  - *master* - мастер сервер для текущей зоны.
  - *slave* - подчиненный сервер для текущей зоны

Использование TSIG (transaction signatures)

Для защиты от искажений и подделок ответов сервера, передачи зоны и обновлений зоны (update) поддерживается использование расширения TSIG протокола DNS.

- Генерация ключа
  - dnssec-keygen -a HMAC-MD5 -b 128 -n HOST имя-ключа
- **key** имя-ключа **{ algorithm hmac-md5; secret "секретная-строка-в-base-64"; };**
  - определяет ключ для аутентификации и авторизации: rndc и TSIG; определение ключа для TSIG можно описывать внутри утверждения view; использовать ключ можно в утверждениях server, controls и в списке-шаблонов-адресов)

SplitDNS

Иногда возникает необходимость отдавать для одной и той же зоны разные данные для одних и тех же записей. Для этого существует техника SplitDNS в isc-bind это реализовано с помощью views.

В случае когда определены views не должно быть зон находящихся вне view.
Клиент может попасть во view основываясь на:

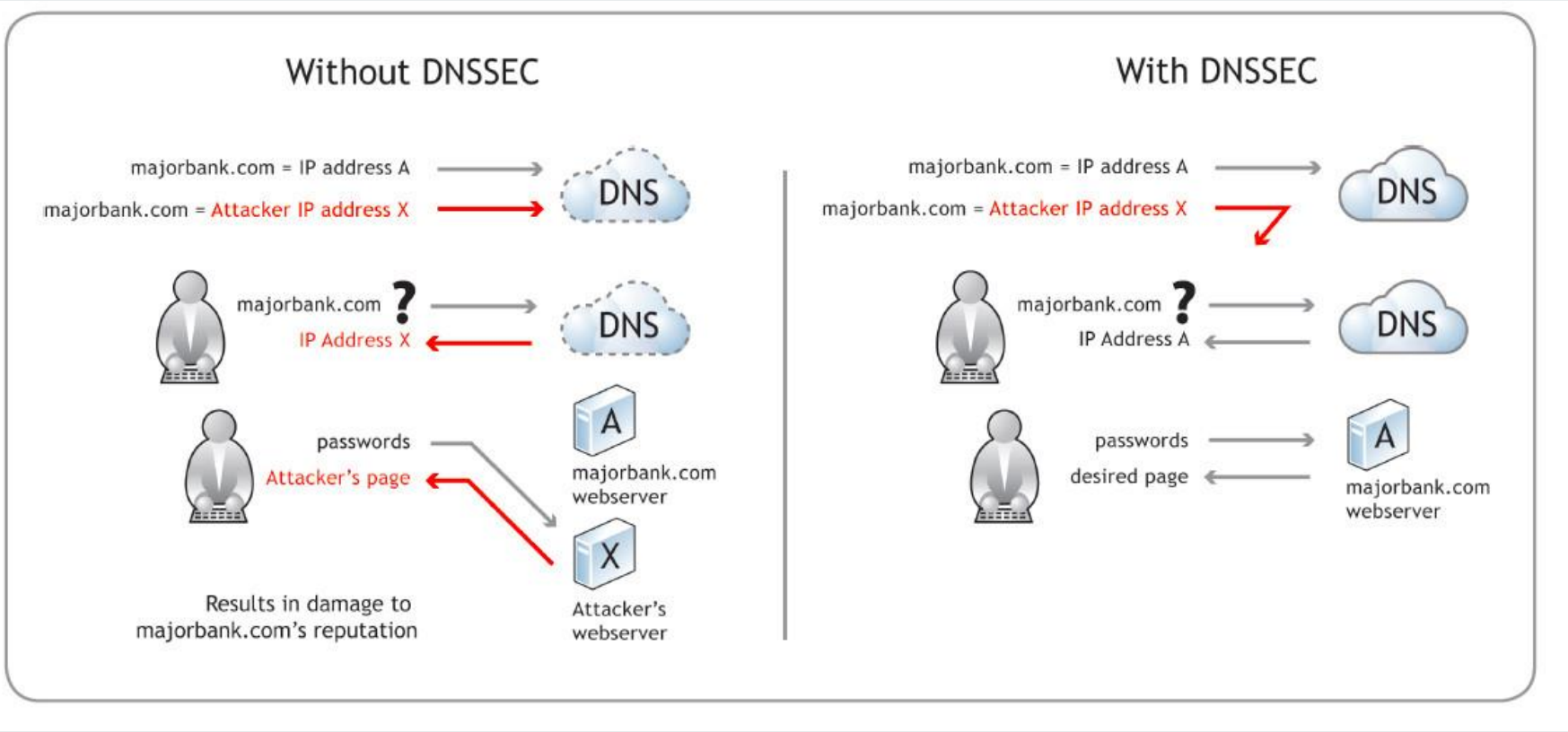
- адресе источника
- адресе назначения
- dns tsig-ключе

DDNS

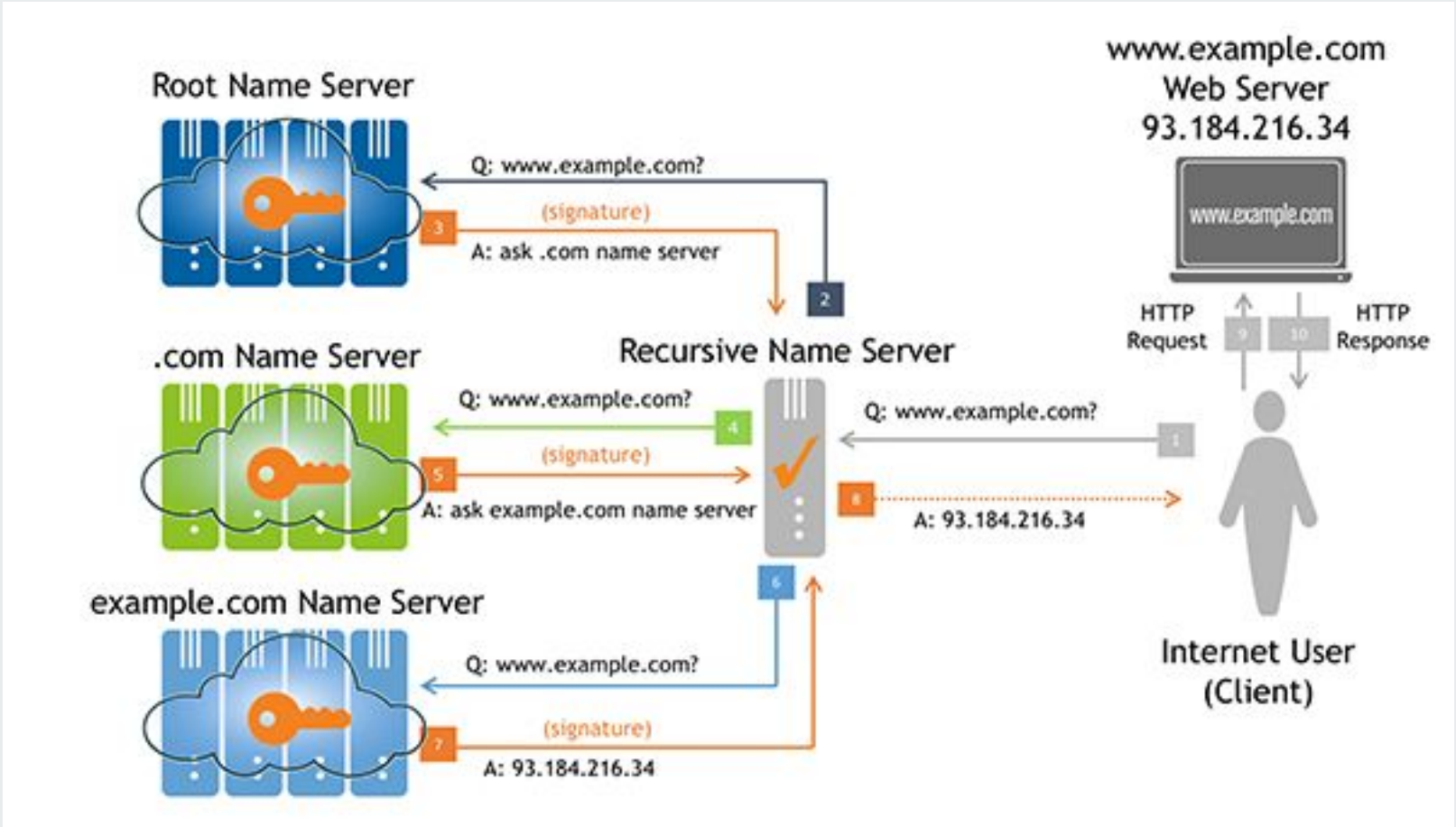
Расширение протокола DNS позволяет клиентам отправлять запросы на изменение записей о ресурсах (RR) первичному уполномоченному серверу непосредственно или с помощью вторичного уполномоченного сервера (предложение a**llow-update-forwarding** в утверждении **options** или **zone**).

Утилита **nsupdate** позволяет сформировать пакет изменений и отослать его первичному уполномоченному серверу (имя сервера извлекается из **SOA** зоны).

DNSSEC


Without DNSSEC
majorbank.com = IP address A
majorbank.com = Attacker IP address X
DNS
majorbank.com ?
IP Address X
DNS
passwords
Attacker's page
A
majorbank.com
webserver
X
Attacker's
webserver
Results in damage to
majorbank.com's reputation
With DNSSEC
majorbank.com = IP address A
majorbank.com = Attacker IP address X
DNS
majorbank.com ?
IP Address A
DNS
passwords
desired page
A
majorbank.com
webserver

DNSSEC


Root Name Server
Q: www.example.com?
(signature)
A: ask .com name server
www.example.com
Web Server
93.184.216.34
www.example.com
HTTP
Request
HTTP
Response
.com Name Server
Recursive Name Server
Q: www.example.com?
(signature)
A: ask example.com name server
Q: www.example.com?
A: 93.184.216.34
example.com Name Server
Q: www.example.com?
(signature)
A: 93.184.216.34
Internet User
(Client)

Балансировка нагрузки

Распределение по ROUND-ROBIN

- Несколько CNAME записей:

```
www1    IN    A      123.45.67.81
www2    IN    A      123.45.67.82
www     IN    CNAME  www1.example.net.
        IN    CNAME  www2.example.net.
```

- Несколько A записей:

```
www.example.net  60  IN  A  123.45.67.81
www.example.net  60  IN  A  123.45.67.82
```

ДЗ

Добавить еще один сервер client2. Завести в зоне dns.lab имена:
web1 - смотрит на клиент1
web2 - смотрит на клиент2

завести еще одну зону newdns.lab
завести в ней запись
www - смотрит на обоих клиентов

Настроить split-dns:
клиент1 - видит обе зоны, но в зоне dns.lab только web1
клиент2 видит только dns.lab

*) настроить все без выключения selinux
ddns тоже должен работать без выключения selinux

Спасибо за внимание

Дмитрий Молчанов
Григорий Ожегов