OTUS

Администратор Linux

DNS: настройка и обслуживание

otus.ru

Проверить, идет ли запись

Меня хорошо видно && слышно?

Ставим “+”, если все хорошо
“-”, если есть проблемы

Тема вебинара

DNS: настройка и обслуживание

Федоров Иван Романович

Технический директор ГК “Инотех”

Опыт:

Более 10 лет в IT-сфере

Аспирант университета ИТМО по направлению “Информационная безопасность”

Многократный победитель различных конкурсов и хакатонов (команда IBI Solutions)

Эл. почта: ifedorov.devops@gmail.com

Правила вебинара

Активно
участвуем

Off-topic обсуждаем
в группе Telegram
OTUS-Linux-2022-12

Задаем вопрос
в чат или голосом

Вопросы вижу в чате,
могу ответить не сразу

Маршрут вебинара

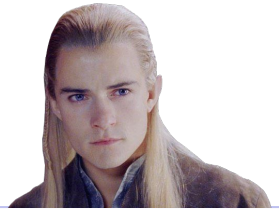
- Знакомство
- Как работает DNS
- DNS-записи
- Настройка DNS-сервера
- Split-DNS
- Рефлексия

Цели вебинара

К концу занятия вы сможете

1. Понять как работает DNS

2. Использовать утилиты для диагностики DNS



3. Управлять DNS-зонами

Смысл

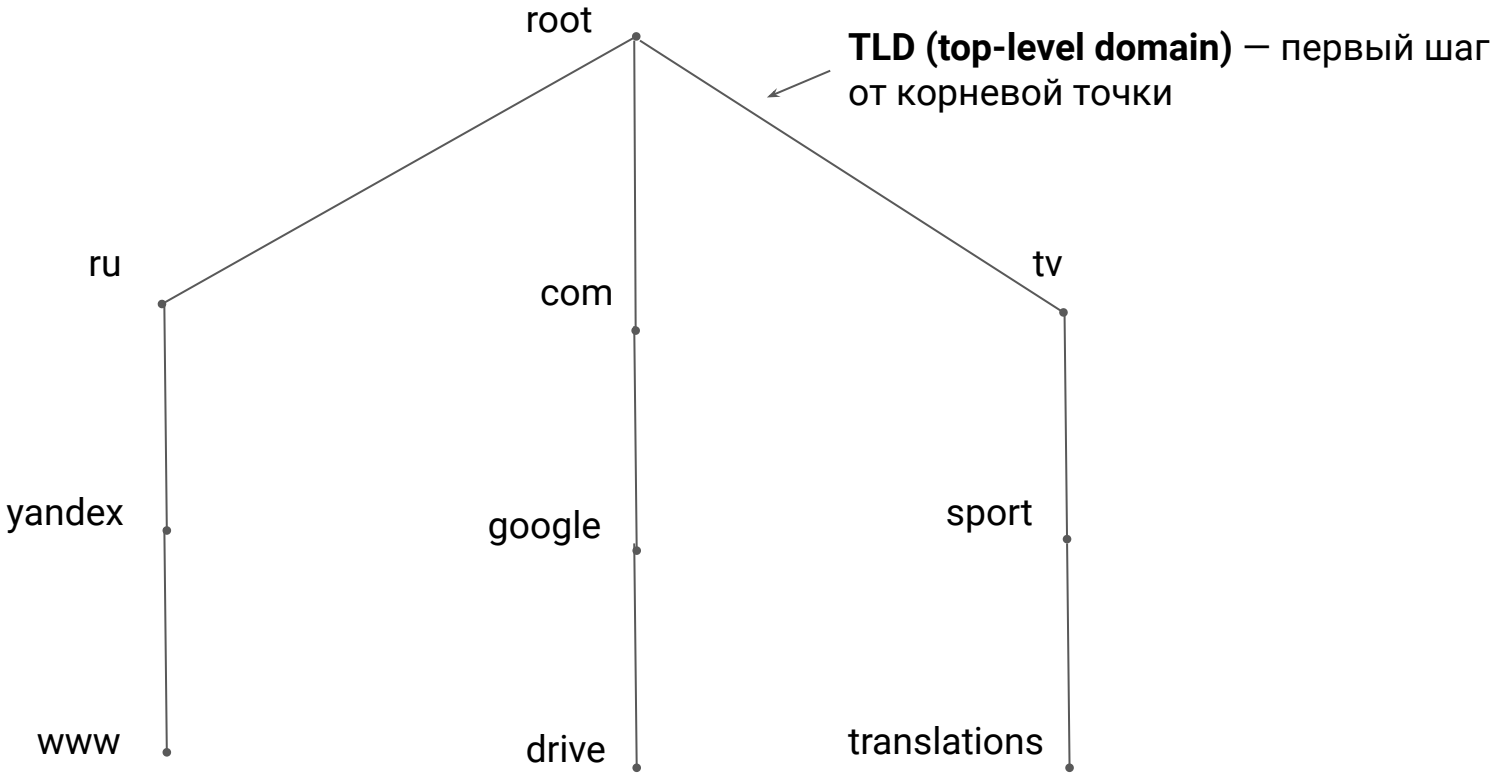
Зачем вам это уметь

1. 200% из 100, что Вы столкнетесь с DNS в реальной жизни

Как работает DNS

Зачем нужен DNS

Иерархия доменных имен



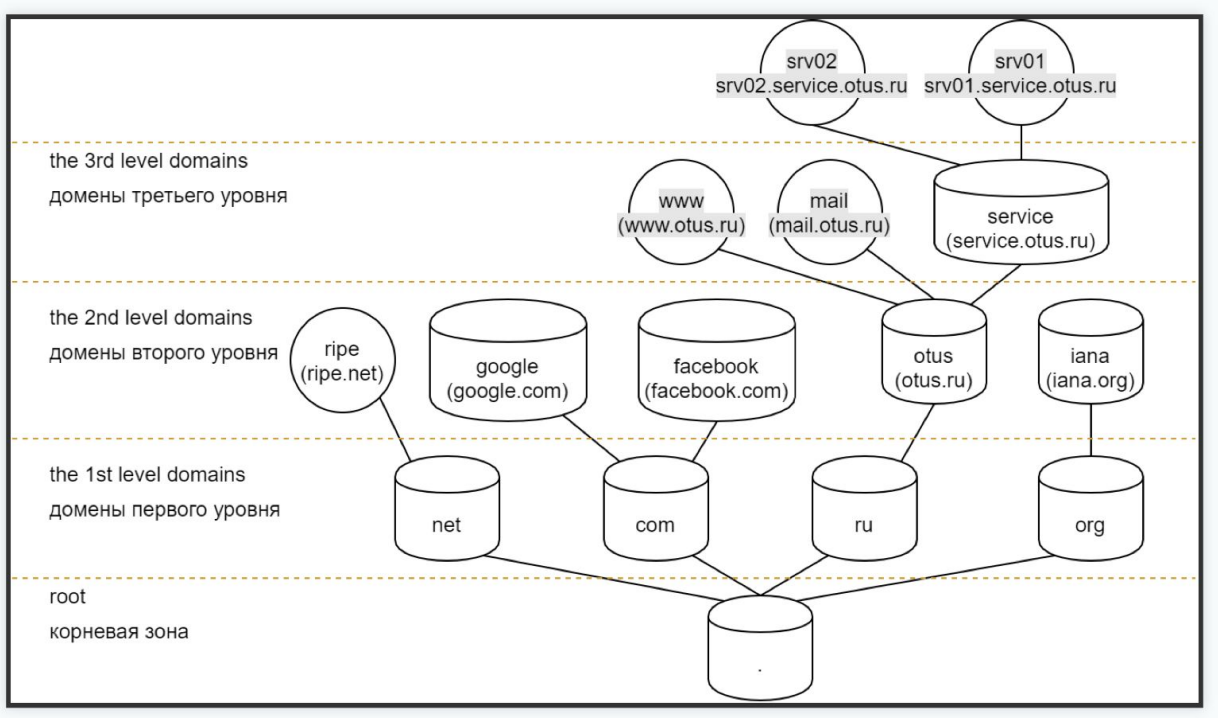
TLD (top-level domain) — первый шаг от корневой точки

FQDN (Fully Qualified Domain Name)

- **Полностью указанное доменное имя, т.е. от корневого домена.**
  Ключевой индикатор — точка в конце имени.
  FQDN www.otus.ru. состоит из:
  - домена 3-го уровня, входящего в состав otus.ru;
  - домена 2-го уровня, входящего в состав ru;
  - домена 1-го уровня, входящего в состав корневого домена

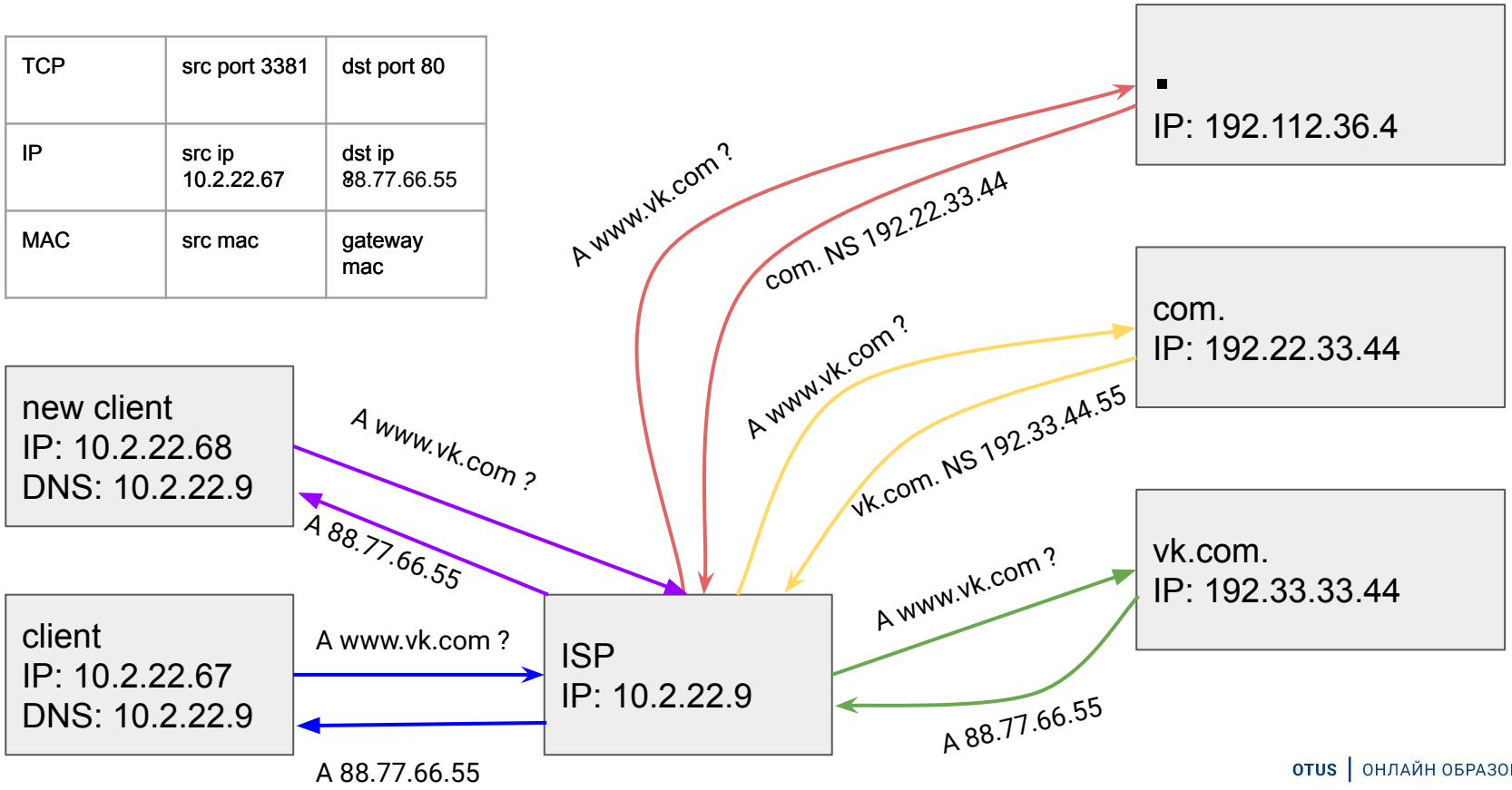
- **Корневой домен не имеет названия и обозначается точкой “.”**

Структура

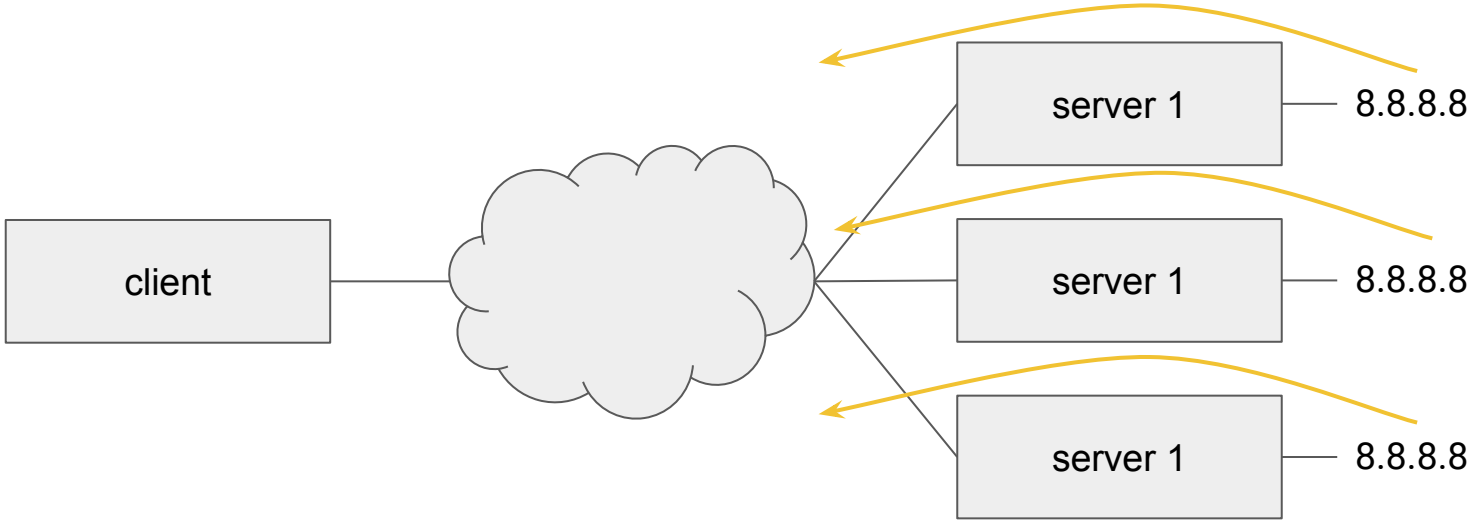

srv02
srv02.service.otus.ru
srv01
srv01.service.otus.ru
the 3rd level domains
домены третьего уровня
www
(www.otus.ru)
mail
(mail.otus.ru)
service
(service.otus.ru)
the 2nd level domains
домены второго уровня
ripe
(ripe.net)
google
(google.com)
facebook
(facebook.com)
otus
(otus.ru)
iana
(iana.org)
the 1st level domains
домены первого уровня
net
com
ru
org
root
корневая зона

Как работает DNS

TCP	src port 3381	dst port 80
IP	src ip 10.2.22.67	dst ip 88.77.66.55
MAC	src mac	gateway mac



ANYCAST



Все хорошо?
Есть ли вопросы?

Утилиты

Клиент DNS. Утилиты.

```
# Для работы с утилитами необходимо установить пакет bind-utils
$ yum install bind-utils

# Диагностика
$ dig www.otus.ru
$ host www.otus.ru
$ nslookup www.otus.ru

# Пример получения ресурсных записей
$ dig MX www.otus.ru
$ dig TXT www.otus.ru
```

DNS-записи

Ресурсные записи (RR)

- Записи DNS обладают следующими атрибутами:
  - имя
  - TTL (время жизни в кеше)
  - класс
  - тип
  - значение (или массив значений)

```
www.otus.ru.            60      IN      A       1.2.4.5
mail                    60      IN      MX      10      mx1.otus.ru.
                                        MX      20      mx1.otus.ru.
```

Типы записей

- **A** — адрес IPv4, соответствующий имени
- **AAAA** — адрес IPv6, соответствующий имени
- **CNAME** — имя, соответствующее имени (canonical name)
- **MX** — массив (приоритет и имя) почтовых серверов для домена
- **TXT** — текстовая информация
- **SOA** — ключевая запись домена (start of authority)
- **NS** — имя сервера имён для домена (nameserver)
- **PTR** — имя, соотвествующее IP-адресу (pointer для in- addr.arpa и ip6.arpa)
- **SRV** — указание на расположение сервиса

PTR-запись

- Представляет собой “обратную A-запись”
- A-запись нужна для преобразования доменного имени в IP-адрес, PTR — для преобразования IP-адреса в доменное имя
- В PTR-записи IP-адрес записывается в обратном порядке.
  То есть IP-адрес 11.22.33.44 в обратной зоне будет записан как 44.33.22.11.
- Зачастую прямая и обратная зоны находятся на разных DNS-серверах, так как ресурсы принадлежат разным компаниям (доменное имя покупает компания, а IP-адрес принадлежит провайдеру)
- Пример:
  Обратная зона: 10.168.192.in-addr.arpa
  Запись: 1 IN PTR www.otus.ru
  Результат: 1.10.168.192.in-addr.arpa IN PTR www.otus.ru

Все хорошо?
Есть ли вопросы?

DNS-сервер

Типы серверов

Типы серверов (по свойствам и функциям):

- **главные (primary или master)** — авторитетные, хранят главную копию информации о зоне;
- **вторичные (secondary или slave)** — получают копию информации о зоне с главного или вторичного сервера и работают с ней;
- **кеширующие** — кешируют ответы на запросы пользователя;
- **рекурсивные** — выполняют полный поиск по иерархии DNS;
- **нерекурсивные** — не выполняют полный поиск (не умеют или им запрещено)

Установка и настройка

```
# Ставим сервер bind9
$ yum install bind

# Конфигурационный файл
$ vim /etc/named.conf

# В конфигурационном файле задаем роль сервера, расположение файлов зоны
# Делаем рестарт сервиса
$ systemctl restart named

# Добавляем адрес созданного сервера имен в /etc/resolv.conf
$ vim /etc/resolv.conf
nameserver 1.1.1.1
```

Пример файла зоны

```
# Описание зоны dns.lab
$ cat /etc/named/zones/db.dns.lab
$TTL 3600
; описание зоны dns.lab.
$ORIGIN dns.lab.
@               IN      SOA     ns01.dns.lab. root.dns.lab. (
                            2711201407 ; serial
                            3600       ; refresh (1 hour)
                            600        ; retry (10 minutes)
                            86400      ; expire (1 day)
                            600        ; minimum (10 minutes)
                        )

                IN      NS      ns01.dns.lab.

; DNS Servers
ns01            IN      A       10.0.0.23

; Web
web1            IN      A       10.0.0.23
web2            IN      A       10.0.0.23
```

Пример файла обратной зоны

```
# Описание обратной зоны 0.0.10.in-addr.arpa.
$ cat /etc/named/zones/db.0.0.10
$TTL 604800
@       IN      SOA     ns01.dns.lab. root.dns.lab. (
                        20210806 ; Serial
                        604800 ; Refresh
                        86400 ; Retry
                        2419200 ; Expire
                        604800 ) ; Negative Cache TTL
;
; name servers
@       IN      NS      ns01.dns.lab.

; PTR Records
23      IN      PTR     ns01.dns.lab.   ;10.0.0.23
24      IN      PTR     testptr.dns.lab.   ;10.0.0.24
```

Split-DNS

Split-DNS

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

Важно: в случае, когда определены **views**, не должно быть зон находящихся вне view.

Клиент может попасть (**match-clients**) во **view** основываясь на:

- адресе источника;
- адресе назначения;
- DNS TSIG-ключе.

Пример настройки

```
# Пример настройки split-dns
$ cat /etc/named.conf
acl "client1" {
  10.0.0.12;
};
acl "client2" {
  10.0.0.13;
};

options {
  ...
}

view "client1" {
    match-clients { client1; };
    zone "dns.lab" {
        type master;
        file "/etc/named/zones/db.dns.lab.client1";
    };
};

view "client2" {
    match-clients { client2; };
    zone "dns.lab" {
        type master;
        file "/etc/named/zones/db.dns.lab.client2";
    };
};

view "default" {
  match-clients { any; };
  zone "." IN {
        type hint;
        file "named.ca";
  };
  include "/etc/named.rfc1912.zones";
  include "/etc/named.root.key";
};
```

Как настроение?
Есть ли вопросы?

Дополнительный материал

Репликация

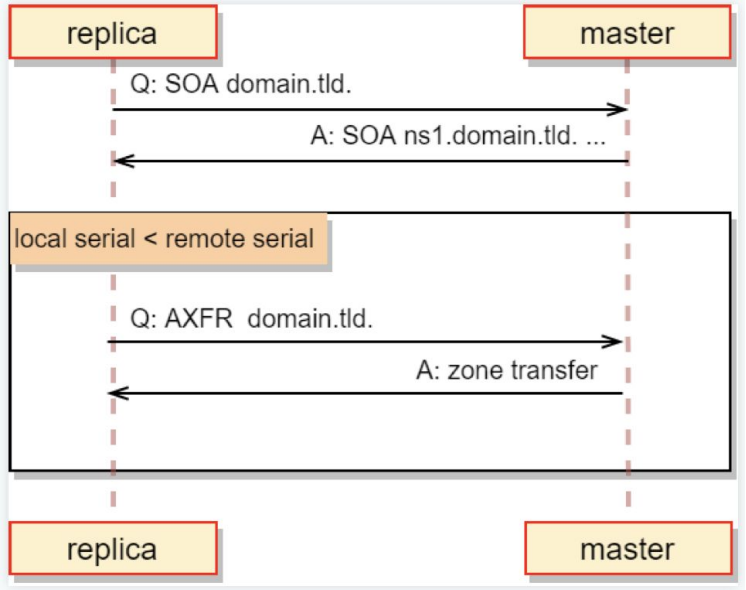
В протокол DNS встроена возможность репликации зон с помощью запросов:

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

Репликация происходит только при одном условии — local_serial < remote_serial. Проверка serial является частью процесса репликации. Для репликации DNS используется протокол tcp, т.к. важна гарантия доставки.

Репликация может быть инициирована следующими событиями:

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



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.

DNSSEC (DNS Security Extensions)

- Нужен для обеспечения безопасности клиентов от фальшивых DNS-данных (DNS cache poisoning, к примеру).
- Все ответы от DNSSEC имеют цифровую подпись.
- При проверке цифровой подписи DNS-клиент проверяет верность и целостность информации.
- DNSSEC не шифрует данные и не обеспечивает конфиденциальность данных; только аутентификация.

TSIG (Transaction Signatures)

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

Генерация ключа: dnssec-keygen -a HMAC-MD5 -b 128 -n HOST имя-ключа

Определение ключа: key имя-ключа { algorithm hmac-md5; secret "секретная-строка-в-base-64"; };

Может использоваться для аутентификации и авторизации:

- view;
- server;
- controls (например, для rndc);
- acl и прочих списках.

Простейшая балансировка

Распределение по принципу **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
```

Вопросы?

Ставим “+”,
если вопросы есть

Ставим “–”,
если вопросов нет

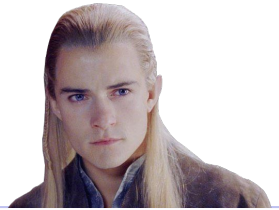
Рефлексия

Цели вебинара

К концу занятия вы сможете

1. Понять как работает DNS

2. Использовать утилиты для диагностики DNS



3. Управлять DNS-зонами

Рефлексия

С какими впечатлениями уходите с вебинара?

Как будете применять на практике то,
что узнали на вебинаре?

Заполните, пожалуйста, опрос о занятии по ссылке в чате