

# Настройка DNS-зоны

## Содержание

### Настройка DNS-зоны

- Запись SOA (Start of Authority)

- Записи NS (name server)

- Запись A (address record)

- Записи MX (mail exchange)

- Записи TXT

  - Запись TXT для DKIM

    - Генерация приватного ключа

    - Генерация публичного ключа

  - Запись TXT для SPF

  - Запись TXT для DMARC

- Обратная зона DNS

### Инструменты диагностики зон DNS

- Прямая доменная зона

- Обратная доменная зона

- Он-лайн сервисы для проверки зон DNS

### Автонастройка календарей и адресных книг с помощью DNS

- Что такое CalDAV и CardDAV

- Поиск сервиса

  - Использование ресурсных записей

  - Использование ожидаемого хорошо известного URL-адреса

- Использование сервиса

  - CalDAV

    - Поиск сервиса

    - Изучение коллекций (календарей)

  - CardDAV

    - Поиск сервиса

    - Исследование коллекции

- Вывод

DNS - это один из языков, которым написан интернет.

DNS (Domain Name System) - это глобальная распределенная база данных, хранящая записи для каждого домена интернет.

Задача этой статьи:

- Определить, как настроить зону DNS для корректной работы почтового сервера;
- Как проверить сделанные настройки.

Рассмотрим минимально необходимую конфигурацию доменной зоны:

```
$TTL 3600
$ORIGIN tegu.online.
```

```

tegu.online.      3600    IN      SOA      ns3.nic.ru.      support.nic.ru. (
                    2020092144 ; serial
                    14400 ; refresh
                    3600 ; retry
                    2592000 ; expire
                    600 ; minimum
                    )

tegu.online.      IN      NS      ns8-cloud.nic.ru.
tegu.online.      IN      NS      ns3-l2.nic.ru.
tegu.online.      IN      NS      ns4-l2.nic.ru.
tegu.online.      IN      NS      ns8-l2.nic.ru.
tegu.online.      IN      NS      ns4-cloud.nic.ru.
tegu.online.      IN      A      185.215.4.56
dav.tegu.online.  IN      A      89.208.226.195
mail.tegu.online. IN      A      79.137.210.127
tegu.online.      IN      MX      10      mail.tegu.online.
_caldavs._tcp.tegu.online. IN      SRV      0      0      8809      mail.tegu.online.
_carddavs._tcp.tegu.online. IN      SRV      0      0      8809      mail.tegu.online.
_imaps._tcp.tegu.online. IN      SRV      0      0      993      mail.tegu.online.
_smtpps._tcp.tegu.online. IN      SRV      0      0      465      mail.tegu.online.
_ssmtp._tcp.tegu.online. IN      SRV      10     0      25      mail.tegu.online.
tegu.online.      IN      TXT      "v=spf1 mx -all"
_caldavs._tcp.tegu.online. IN      TXT      "path=/caldav"
_carddavs._tcp.tegu.online. IN      TXT      "path=/carddav"
_dmarc.tegu.online. IN      TXT      "v=DMARC1; p=quarantine; rua=mailto:abuse@tegu.online"
mail._domainkey.tegu.online. IN      TXT      "v=DKIM1; h=sha256; k=rsa; p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDRKjngoDbWD3Cj1NHc/GB9YB4i3bjmJV2F

```

Рассмотрим каждую запись подробнее.

## Запись SOA (Start of Authority)

Описывает основные/начальные настройки зоны.

Для каждой зоны должна существовать только одна запись SOA, и она должна быть первая.

- Поле Name содержит имя домена/зоны,
- поля TTL, CLASS — стандартное значение,
- поле TYPE принимает значение SOA,
- а поле DATA состоит из нескольких значений, разделенных пробелами: имя главного DNS (Primary Name Server), адрес администратора зоны, далее в скобках — серийный номер файла зоны (Serial number). При каждом внесении изменений в файл зоны данное значение необходимо увеличивать, это указывает вторичным серверам, что зона изменена, и что им необходимо обновить у себя зону.

## Записи NS (name server)

Указывают сервера, на которые делегирован данный домен.

## Запись A (address record)

Отображает имя хоста (доменное имя) на адрес IPv4.

Для каждого сетевого интерфейса машины должна быть сделана одна A-запись.

## Записи MX (mail exchange)

Указывают хосты для доставки почты, адресованной домену.

- При этом поле NAME указывает домен назначения,
- поля TTL, CLASS — стандартное значение,
- поле TYPE принимает значение MX,
- а поле DATA указывает приоритет и через пробел - доменное имя хоста, ответственного за прием почты.

## Записи TXT

### Запись TXT для DKIM

DKIM [RFC6376](#) - механизм, позволяющий проверить является ли отправитель достоверным или нет. Проверка осуществляется с помощью цифровой подписи, публичная часть которой находится в DNS соответствующей зоны. DKIM защищает от отправки сообщения с подменой адреса отправителя.

Чтобы настроить DKIM необходимо:

1. Создать пару (публичный/приватный) RSA-ключей;
2. Публичный ключ опубликовать в DNS-зоне;
3. Приватный ключ отдать своему серверу (он знает как с ним поступить).

### Генерация приватного ключа

```
openssl genrsa -out private.pem 1024
```

### Генерация публичного ключа

```
rsa -pubout -in private.pem -out public.pem
```

Но в случае с почтовым сервером Tegu все обстоит намного проще.

Генерировать вручную RSA-ключ нет необходимости. Достаточно зайти в опцию меню **DKIM**, выбрать один из обслуживаемых сервером интернет-доменов и кнопкой **Создать** сгенерировать PSA-ключ. Приватную часть ключа сервер оставит себе, а публичную вернет вам для использования в описании DNS-зоны. В дальнейшем, ключ можно пересоздать или удалить.

### Запись TXT для SPF

SPF [RFC7208](#) - механизм для проверки подлинности сообщения, путем проверки фактического адреса сервера отправителя со списком разрешенных адресов

серверов, указанных в соответствующей зоне DNS. SPF не позволяет случиться ситуации, когда от имени вашего домена будут рассылаться мошеннические письма.

Поля записи:

- v=spf1 - является версией, всегда spf1;
- a - разрешает отправлять письма с адреса, который указан в A и\или AAAA записи домена отправителя;
- mx - разрешает отправлять письма с адреса, который указан в mx записи домена;
- all - означает то, что будет происходить с письмами, которые не соответствуют политике:
  - "-" — отклонять;
  - "+" — пропускать;
  - "~" — дополнительные проверки;
  - "?" — нейтрально.

## Запись TXT для DMARC

DMARC [RFC7489](#) — механизм снижения количества спамовых и фишинговых писем. DMARC описывает действие, которое должен совершить сервер для писем, которые не прошли проверку DKIM и SPF. А также описывает адрес, на который раз в сутки будет отправляться отчет об этих действиях.

Поля записи:

- v - версия, принимает значение v=DMARC1 (обязательный параметр);
- p - правило для домена. Может принимать значения none, quarantine и reject, где:
  - p=none не делает ничего, кроме подготовки отчетов;
  - p=quarantine добавляет письмо в СПАМ;
  - p=reject отклоняет письмо;
- rua - позволяет отправлять ежедневные отчеты на email.

Если ваш сервер не удовлетворяет политикам безопасности, то как минимум ваше сообщение будет отвергнуто, как максимум - на ваш сервер в автоматическом режиме будет отправлена "жалоба". Достать домен из списков неблагожелательных бывает долго и трудно.

## Обратная зона DNS

Обратная зона DNS — это особая доменная зона, которая позволяет совершить обратное разрешение доменных имен относительно IP-адресов. Для обратной зоны фактически разрешена только запись одного типа A, называемая PTR.

В большинстве случаев PTR-запись может быть прописана только владельцем IP-адреса, как правило (исключая автономные системы, AS) это операторы связи или ЦОДов. Поэтому в практическом смысле это означает, что вы должны обратиться к вашему провайдеру услуг с просьбой прописать обратную зону для выбранного хоста.

**ВАЖНО!** Пожалуйста не используйте почтовый сервер, для которого не прописана обратная зона даже для тестирования. Начав свое общение с другими серверами, но не умея подтвердить свое имя в обратной зоне, ваш тестовый сервер скомпрометирует и доменное имя, и свой IP-адрес. Восстановление может занять весьма значительное время.

## Инструменты диагностики зон DNS

Для тестирования зон DNS нам потребуется консольный DNS-клиент.

Среди популярных известно два:

Среди популярных известны два.

- dig (domain information groper);
- nslookup (name server lookup).

Ниже мы приведем команды для каждой из этих утилит.

## Прямая доменная зона

Проверка записей NS:

```
$ host -t ns tegu.online
tegu.online name server ns4-cloud.nic.ru.
tegu.online name server ns4-l2.nic.ru.
tegu.online name server ns8-cloud.nic.ru.
tegu.online name server ns3-l2.nic.ru.
tegu.online name server ns8-l2.nic.ru.
```

Проверка записей CNAME:

```
$ host mail.tegu.online
mail.tegu.online has address 79.137.210.127
```

Проверка записей MX:

```
$ host -t mx tegu.online
tegu.online mail is handled by 10 mail.tegu.online.
```

Проверка записи TXT для SPF:

```
$ host -t txt tegu.online
tegu.online descriptive text "v=spf1 mx -all"
```

Проверка записи TXT для DMARC:

```
$ host -t txt _dmarc.tegu.online
_dmarc.tegu.online descriptive text "v=DMARC1; p=quarantine; rua=mailto:abuse@tegu.online"
```

Проверка записи TXT для DKIM:

```
$ host -t txt mail._domainkey.tegu.online
mail._domainkey.tegu.online descriptive text "v=DKIM1; h=sha256; k=rsa; p=MIGfMA0GCsqGSIb3DQEBAQUAA4GNADCBiQKBgQDRKjngoDbWD3Cj1NHc/GB9YB4i3bjmJV2R0HfV
```

## Обратная доменная зона

```
$ host 79.137.210.127
127.210.137.79.in-addr.arpa domain name pointer mail.tegu.online.
```

## Он-лайн сервисы для проверки зон DNS

- <https://2whois.ru/>
- <https://dns.nettools.ru/>

## Автонастройка календарей и адресных книг с помощью DNS

Эта статья поможет вам использовать методы автонастройки почтовых клиентов для использования календарей и адресных книг.

Обратите внимание, что не все клиенты почты или календарей поддерживают описанные RFC.

Тем не менее исследуемые нами на десктопах Outlook, Thunderbird, Evolution, а также на мобильных ОС - K9 Mail (и многие другие), полностью поддерживают данные механизмы.

## Что такое CalDAV и CardDAV

CalDAV и CardDAV являются открытыми протоколами, которые используются для доступа к календарям и адресным книгам, хранящимся на сервере. Они позволяют нескольким пользовательским устройствам синхронизировать контакты адресных книг и события календарей.

## Поиск сервиса

Существует два метода, используемых для обнаружения служб CalDAV и CardDAV.

- Первый и предпочтительный способ - через записи DNS SRV.
- Второй способ заключается в использовании "ожидаемого" хорошо известного URL-адреса для обнаружения сервисов.

## Использование ресурсных записей

Использование записей DNS SRV, определенных в [RFC 6764](#), определяет следующий формат записей:

```
_caldav._tcp: Идентификатор CalDAV-сервера без TLS
_caldavs._tcp: Идентификатор CalDAV-сервера с поддержкой TLS
_carddav._tcp: Идентификатор CardDAV-сервера без TLS
_carddavs._tcp: Идентификатор CardDAV-сервера с поддержкой TLS
```

Форматы этих записей определены в [RFC 2782](#):

```
SRV priority weight port target
```

Где:

- **Priority:** приоритет этого целевого хоста. Клиент ДОЛЖЕН попытаться связаться с целевым хостом с наименьшим приоритетом, которого он может достичь; целевые хосты с тем же приоритетом следует опробовать в порядке, определенном полем **weight**. Диапазон составляет 0-65535.
- **Weight:** В поле вес указывается относительный вес для записей с одинаковым приоритетом. Чем больше вес, тем пропорционально выше вероятность того, что они будут выбраны. Диапазон этого числа составляет 0-65535.
- **Port:** Порт на этом целевом узле данной службы. Диапазон составляет 0-65535.
- **Target:** The domain name of the target host. There MUST be one or more address records for this name, the name MUST NOT be an alias. A target of "." means that the service is decidedly not available at this domain.

Будем использовать TLS и отключим незащищенные протоколы:

```
$ host -t srv _carddavs._tcp.tegu.online
_carddavs._tcp.tegu.online has SRV record 0 0 8443 mail.tegu.online.
$ host -t srv _caldavs._tcp.tegu.online
_caldavs._tcp.tegu.online has SRV record 0 0 8443 mail.tegu.online.

$ host -t txt _carddavs._tcp.tegu.online
_carddavs._tcp.tegu.online descriptive text "path=/carddav"
$ host -t txt _caldavs._tcp.tegu.online
_caldavs._tcp.tegu.online descriptive text "path=/caldav"
```

Просматривая доменную часть `tegu.online`, мы видим, что службы размещены на сервере под названием `mail.tegu.online`. Теперь, зная сервер, определим пути к сервисам CalDAV и CardDAV (согласно [RFC 6764](#)):

```
$ host -t TXT _caldavs._tcp.tegu.online
_caldavs._tcp.tegu.online descriptive text "path=/calendars"
$ host -t TXT _carddavs._tcp.tegu.online
_carddavs._tcp.tegu.online descriptive text "path=/addressbooks"
```

Теперь начало сеанса должно быть направлено на сервер и путь, которые мы нашли в записях DNS. Если комбинированный поиск SRV + TXT приводит к ошибке, клиент должен вернуться ко второму методу "ожидаемого хорошо известного URL-адреса".

## Использование ожидаемого хорошо известного URL-адреса

[RFC 6764](#) дополнительно определяет некоторые предопределенные URL-адреса, которые можно попробовать в случае, если поиск по SRV + TXT, как описано выше, не работает (.well-known Discovery). Хорошо известный адрес соответствует [RFC 8615](#).

Если поиск по SRV работает, а поиск по TXT - нет, клиент должен использовать адрес и порт, возвращенные в записи SRV. Однако, если поиск по SRV не сработал, то во многих случаях, хотя этого и нет в RFC, клиент попытается получить доступ к .well-known URL в домене пользователя. При этом запросы отправляются только по стандартному номеру порта (HTTP/80) или HTTPS/443).

Предварительно определенными URL-адресами являются:

- `/.well-known/caldav` для CalDAV;
- `/.well-known/carddav` для CardDAV.

Запросы к этим URL-адресам должны привести к перенаправлению туда, где находится служба. Вот пример с нашего сайта, где вы можете увидеть возвращаемый URL-адрес наших сервисов:

```
$ curl -sD - -o /dev/null https://mail.tegu.online:8443/.well-known/caldav
HTTP/1.1 302 Moved Temporarily
Server: nginx/1.16.1
Date: Sat, 09 May 2020 02:27:13 GMT
Content-Type: text/html
Content-Length: 145
Location: https://mail.tegu.online:8443/calendars
Connection: keep-alive
X-Content-Type-Options: nosniff
```

```
$ curl -sD - -o /dev/null https://mail.tegu.online:8443/.well-known/carddav
HTTP/1.1 302 Moved Temporarily
Server: nginx/1.16.1
Date: Sat, 09 May 2020 02:27:17 GMT
Content-Type: text/html
Content-Length: 145
Location: https://mail.tegu.online:8443/addressbooks
Connection: keep-alive
X-Content-Type-Options: nosniff
```

Если клиент не может определить имя сервера и путь к службе ни с помощью SRV + TXT, ни с помощью метода .well-known, то у него остается единственный способ - запросить у пользователя адрес сервера и информацию о пути.

## Использование сервиса

Итак, с помощью DNS мы нашли сервер и путь к сервисам. Как CalDAV, так и CardDAV являются расширениями протокола WebDAV, как описано в [RFC 4918](#), то это подразумевает наличие папок (коллекций), файлов (ресурсов) и их метаданных.

Внимание! Все команды далее будут аутентифицированы от имени пользователя, пытающегося получить доступ к сервису.

## CalDAV

Начнем общение со службой, как описано в [RFC 4791](#).

### Поиск сервиса

Во-первых, мы должны действительно подтвердить, что по найденному URL-адресу отвечает услуга CalDAV. Чтобы сделать это, мы ищем свойства в URL-адресе, используя метод PROPFIND:

```
$ curl --user "user@tegu.online:ThePassword" -s -X PROPFIND -H "Content-Type: application/xml" -sD /dev/stderr https://mail.tegu.online:8443/calendars
HTTP/1.1 207 Multi-Status
Server: nginx/1.16.1
```



```
Server: nginx/1.10.1
Date: Sat, 09 May 2020 05:49:47 GMT
Content-Type: application/xml; charset=utf-8
Transfer-Encoding: chunked
Connection: keep-alive
Vary: Brief,Prefer
DAV: 1, 3, extended-mkcol, access-control, calendarserver-principal-property-search, 2, resource-sharing, calendar-access, calendar-proxy, calendar-au

<?xml version="1.0"?>
<d:multistatus xmlns:d="DAV:" xmlns:s="https://sabredav.org/ns" xmlns:cal="urn:ietf:params:xml:ns:caldav" xmlns:cs="https://calendarserver.org/ns/" xm
  <d:response>
    <d:href>/calendars/</d:href>
    <d:propstat>
      <d:prop>
        <d:resourcetype>
          <d:collection/>
        </d:resourcetype>
        <d:getcontenttype>application/octet-stream</d:getcontenttype>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
  </d:response>
  <d:response>
    <d:href>/calendars/user@tegu.online/</d:href>
    <d:propstat>
      <d:prop>
        <d:resourcetype>
          <d:collection/>
        </d:resourcetype>
        <d:getcontenttype>application/octet-stream</d:getcontenttype>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
  </d:response>
</d:multistatus>
```

Поле "calendar-access" в заголовке DAV в ответе убеждает нас в том, что по данному адресу предлагаются услуги CalDAV:

```
DAV: 1, 3, extended-mkcol, access-control, calendarserver-principal-property-search, 2, resource-sharing, calendar-access, calendar-proxy, calendar-au
```

Но в теле запроса нам также была возвращено имя коллекции:

```
/calendars/user@tegu.online/
```

## Изучение коллекций (календарей)

В приведенном выше разделе мы нашли коллекцию, которая была помечена как календарь, так что давайте посмотрим, что там находится.

Мы выполним рекурсию вниз, насколько это возможно, с глубиной, установленной на бесконечность, но передадим некоторые данные запроса, чтобы указать, что нас интересуют только объекты календаря, и вернуть только URL, тип ресурса и отображаемое имя коллекции:

```
$ curl --user "user@tegu.online:ThePassword" -sD /dev/stderr -H "Content-Type: application/xml" -X PROPFIND -H "Depth: infinity" --data '<d:propfind x
HTTP/1.1 207 Multi-Status
Server: nginx/1.16.1
Date: Sat, 09 May 2020 05:48:57 GMT
Content-Type: application/xml; charset=utf-8
Transfer-Encoding: chunked
Connection: keep-alive
Vary: Brief,Prefer
DAV: 1, 3, extended-mkcol, access-control, calendarserver-principal-property-search, 2, resource-sharing, calendar-access, calendar-proxy, calendar-au

<?xml version="1.0"?>
<d:multistatus xmlns:d="DAV:" xmlns:s="https://sabredav.org/ns" xmlns:cal="urn:ietf:params:xml:ns:caldav" xmlns:cs="https://calendarserver.org/ns/" xm
  <d:response>
    <d:href>/calendars/user@tegu.online/</d:href>
    <d:propstat>
      <d:prop>
        <d:resourcetype>
          <d:collection/>
        </d:resourcetype>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
    <d:propstat>
      <d:prop>
        <d:displayname/>
      </d:prop>
      <d:status>HTTP/1.1 404 Not Found</d:status>
    </d:propstat>
  </d:response>
  <d:response>
    <d:href>/calendars/user@tegu.online/02f29dd9516f1f0c415527e0a60cb3f7/</d:href>
    <d:propstat>
      <d:prop>
        <d:resourcetype>
          <d:collection/>
          <cal:calendar/>
          <cs:shared-owner/>
        </d:resourcetype>
        <d:displayname>Birthdays</d:displayname>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
  </d:response>
</d:multistatus>
```

```
</u:response>
<d:response>
  <d:href>/calendars/user@tegu.online/0f9263536b9fc61ada745644735bfd8f/</d:href>
  <d:propstat>
    <d:prop>
      <d:resourcetype>
        <d:collection/>
        <cal:calendar/>
        <cs:shared-owner/>
      </d:resourcetype>
      <d:displayname>Work</d:displayname>
    </d:prop>
    <d:status>HTTP/1.1 200 OK</d:status>
  </d:propstat>
</d:response>
<d:response>
  <d:href>/calendars/user@tegu.online/edfab414adccd8b67f7727e3ae03b85b/</d:href>
  <d:propstat>
    <d:prop>
      <d:resourcetype>
        <d:collection/>
        <cal:calendar/>
        <cs:shared-owner/>
      </d:resourcetype>
      <d:displayname>Important Dates</d:displayname>
    </d:prop>
    <d:status>HTTP/1.1 200 OK</d:status>
  </d:propstat>
</d:response>
<d:response>
  <d:href>/calendars/user@tegu.online/default/</d:href>
  <d:propstat>
    <d:prop>
      <d:resourcetype>
        <d:collection/>
        <cal:calendar/>
        <cs:shared-owner/>
      </d:resourcetype>
      <d:displayname>Family</d:displayname>
    </d:prop>
    <d:status>HTTP/1.1 200 OK</d:status>
  </d:propstat>
</d:response>
<d:response>
  <d:href>/calendars/user@tegu.online/inbox/</d:href>
  <d:propstat>
    <d:prop>
      <d:resourcetype>
```

```

        <d:collection/>
        <cal:schedule-inbox/>
    </d:resourcetype>
</d:prop>
<d:status>HTTP/1.1 200 OK</d:status>
</d:propstat>
<d:propstat>
    <d:prop>
        <d:displayname/>
    </d:prop>
    <d:status>HTTP/1.1 404 Not Found</d:status>
</d:propstat>
</d:response>
<d:response>
    <d:href>/calendars/user@tegu.online/outbox/</d:href>
    <d:propstat>
        <d:prop>
            <d:resourcetype>
                <d:collection/>
                <cal:schedule-outbox/>
            </d:resourcetype>
        </d:prop>
        <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
    <d:propstat>
        <d:prop>
            <d:displayname/>
        </d:prop>
        <d:status>HTTP/1.1 404 Not Found</d:status>
    </d:propstat>
</d:response>
</d:multistatus>

```

В ответе мы видим три различных типа ресурсов календаря:

```

cal:calendar
cal:schedule-inbox
cal:schedule-outbox

```

Как "cal:schedule-inbox", так и "cal:schedule-outbox" являются специальными типами ресурсов, определенными в [RFC 6638](#), которые используются для планирования. Для этого поста нас интересуют только возвращаемые ресурсы "calendar", а именно:

```

/calendars/user@tegu.online/02f29dd9516f1f0c415527e0a60cb3f7/ (Display Name: Birthdays)
/calendars/user@tegu.online/0f9263536b9fc61ada745644735bfd8f/ (Display Name: Work)
/calendars/user@tegu.online/edfab414adccd8b67f7727e3ae03b85b/ (Display Name: Important Dates)
/calendars/user@tegu.online/default/ (Display Name: Family)

```

Давайте выберем календарь "Birthdays" и посмотрим, что там есть:

```
$ curl --user "user@tegu.online:ThePassword" -sD /dev/stderr -H "Content-Type: application/xml" -X PROPFIND -H "Depth: infinity" https://mail.tegu.online/
HTTP/1.1 207 Multi-Status
Server: nginx/1.16.1
Date: Sat, 09 May 2020 05:47:23 GMT
Content-Type: application/xml; charset=utf-8
Transfer-Encoding: chunked
Connection: keep-alive
Vary: Brief,Prefer
DAV: 1, 3, extended-mkcol, access-control, calendarserver-principal-property-search, 2, resource-sharing, calendar-access, calendar-proxy, calendar-a

<?xml version="1.0"?>
<d:multistatus xmlns:d="DAV:" xmlns:s="https://sabredav.org/ns" xmlns:cal="urn:ietf:params:xml:ns:caldav" xmlns:cs="https://calendarserver.org/ns/" xm
  <d:response>
    <d:href>/calendars/user@tegu.online/02f29dd9516f1f0c415527e0a60cb3f7/</d:href>
    <d:propstat>
      <d:prop>
        <d:resourcetype>
          <d:collection/>
          <cal:calendar/>
          <cs:shared-owner/>
        </d:resourcetype>
        <d:getcontenttype>application/octet-stream</d:getcontenttype>
        <cs:getctag>http://sabre.io/ns/sync/3</cs:getctag>
        <s:sync-token>3</s:sync-token>
        <cal:supported-calendar-component-set>
          <cal:comp name="VEVENT"/>
          <cal:comp name="VTODO"/>
        </cal:supported-calendar-component-set>
        <cal:schedule-calendar-transp>
          <cal:opaque/>
        </cal:schedule-calendar-transp>
        <d:displayname>Birthdays</d:displayname>
        <x1:calendar-order xmlns:x1="https://apple.com/ns/ical/">0</x1:calendar-order>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
  </d:response>
  <d:response>
    <d:href>/calendars/user@tegu.online/02f29dd9516f1f0c415527e0a60cb3f7/33992bd8-d3fe-4b07-baaf-c43d0042fae8.ics</d:href>
    <d:propstat>
      <d:prop>
        <d:getlastmodified>Sat, 09 May 2020 01:24:06 GMT</d:getlastmodified>
        <d:getcontentlength>554</d:getcontentlength>
        <d:resourcetype/>
```

```
<d:getetag>"d34b216dac0dcf327a3cf2d79f95a226"</d:getetag>
<d:getcontenttype>text/calendar; charset=utf-8; component=vevent</d:getcontenttype>
</d:prop>
<d:status>HTTP/1.1 200 OK</d:status>
</d:propstat>
</d:response>
</d:multistatus>
```

Мы можем видеть только один ресурс или файл:

```
/calendars/user@tegu.online/02f29dd9516f1f0c415527e0a60cb3f7/33992bd8-d3fe-4b07-baaf-c43d0042fae8.ics
```

В данном случае это событие календаря, и мы можем сообщить об этом с помощью свойства:

```
<d:getcontenttype>text/calendar; charset=utf-8; component=vevent</d:getcontenttype>
```

Читаем это событие с помощью способа HTTP GET:

```
curl --user "user@tegu.online:ThePassword" -i -X GET https://mail.tegu.online:8443/calendars/user@tegu.online/02f29dd9516f1f0c415527e0a60cb3f7/33992bd8-d3fe-4b07-baaf-c43d0042fae8.ics
HTTP/1.1 200 OK
Server: nginx/1.16.1
Date: Sat, 09 May 2020 05:28:59 GMT
Content-Type: text/calendar; charset=utf-8; component=vevent
Content-Length: 554
Connection: keep-alive
Last-Modified: Sat, 09 May 2020 01:24:06 GMT
ETag: "d34b216dac0dcf327a3cf2d79f95a226"
X-Content-Type-Options: nosniff

BEGIN:VCALENDAR
VERSION:2.0
PRODID:-//apiserver//API Server DAV//EN
BEGIN:VEVENT
UID:33992bd8-d3fe-4b07-baaf-c43d0042fae8
DTSTART;VALUE=DATE:20200506
DTEND;VALUE=DATE:20200507
CREATED:20190124T104704Z
DTSTAMP:20190128T234044Z
LAST-MODIFIED;X-VOBJ-FLOATINGTIME-ALLOWED=TRUE:20200503T203005
RRULE:FREQ=YEARLY;INTERVAL=1;WKST=MO
SEQUENCE:30
STATUS:CONFIRMED
SUMMARY:Dad's Birthday
TRANSP:OPAQUE
BEGIN:VALARM
```

```
ACKNOWLEDGED:20190128T234044Z
ACTION:DISPLAY
DESCRIPTION:Default Description
TRIGGER:-P1D
END:VALARM
END:VEVENT
END:VCALENDAR
```

Ответ представляет собой событие в формате календаря. Что и требовалось доказать.

## CardDAV

Использование CardDAV очень похоже на использование CalDAV.

CalDAV описан в [RFC 6352](#) и последующих исправлениях, которые расширяют функциональность.

## Поиск сервиса

Здесь мы ищем параметр "addressbook", возвращаемый в заголовке ответа DAV:

```
$ curl --user "user@tegu.online:ThePassword" -s -X PROPFIND -H "Content-Type: application/xml" -sD /dev/stderr https://mail.tegu.online:8443/addressbook
HTTP/1.1 207 Multi-Status
Server: nginx/1.16.1
Date: Sat, 09 May 2020 05:40:00 GMT
Content-Type: application/xml; charset=utf-8
Transfer-Encoding: chunked
Connection: keep-alive
Vary: Brief,Prefer
DAV: 1, 3, extended-mkcol, access-control, calendarserver-principal-property-search, 2, resource-sharing, calendar-access, calendar-proxy, calendar-auto-discovery

<?xml version="1.0"?>
<d:multistatus xmlns:d="DAV:" xmlns:s="https://sabredav.org/ns" xmlns:cal="urn:ietf:params:xml:ns:caldav" xmlns:cs="https://calendarserver.org/ns/" xmlns:card="urn:ietf:params:xml:ns:carddav">
  <d:response>
    <d:href>/addressbooks/</d:href>
    <d:propstat>
      <d:prop>
        <d:resourcetype>
          <d:collection/>
        </d:resourcetype>
        <d:getcontenttype>application/octet-stream</d:getcontenttype>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
  </d:response>
  <d:response>
    <d:href>/addressbooks/user@tegu.online/</d:href>
    <d:propstat>
      <d:prop>
```

```

    <d:prop>
      <d:resourcetype>
        <d:collection/>
      </d:resourcetype>
      <d:getcontenttype>application/octet-stream</d:getcontenttype>
    </d:prop>
    <d:status>HTTP/1.1 200 OK</d:status>
  </d:propstat>
</d:response>
</d:multistatus>

```

## Исследование коллекции

Здесь мы можем увидеть коллекцию для этого пользователя. Исследуем ее:

```

$ curl --user "user@tegu.online:ThePassword" -s -X PROPFIND -H "Content-Type: application/xml" -sD /dev/stderr https://mail.tegu.online:8443/addressbooks/user@tegu.online/
HTTP/1.1 207 Multi-Status
Server: nginx/1.16.1
Date: Sat, 09 May 2020 05:40:17 GMT
Content-Type: application/xml; charset=utf-8
Transfer-Encoding: chunked
Connection: keep-alive
Vary: Brief,Prefer
DAV: 1, 3, extended-mkcol, access-control, calendarserver-principal-property-search, 2, resource-sharing, calendar-access, calendar-proxy, calendar-authentication

<?xml version="1.0"?>
<d:multistatus xmlns:d="DAV:" xmlns:s="https://sabredav.org/ns" xmlns:cal="urn:ietf:params:xml:ns:caldav" xmlns:cs="https://calendarserver.org/ns/" xmlns:card="urn:ietf:params:xml:ns:carddav">
  <d:response>
    <d:href>/addressbooks/user@tegu.online/</d:href>
    <d:propstat>
      <d:prop>
        <d:resourcetype>
          <d:collection/>
        </d:resourcetype>
        <d:getcontenttype>application/octet-stream</d:getcontenttype>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
  </d:response>
  <d:response>
    <d:href>/addressbooks/user@tegu.online/default/</d:href>
    <d:propstat>
      <d:prop>
        <d:resourcetype>
          <d:collection/>
          <card:addressbook/>
        </d:resourcetype>
        <d:getcontenttype>application/octet-stream</d:getcontenttype>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
  </d:response>

```



```
<d:getcontenttype>application/octet-stream</d:getcontenttype>
</d:prop>
<d:status>HTTP/1.1 200 OK</d:status>
</d:propstat>
</d:response>
</d:multistatus>
```

Здесь мы видим единственную коллекцию адресных книг под названием `default`, потому, что в свойствах типа ресурса указано "<card:addressbook/>":

```
$ curl --user "user@tegu.online:ThePassword" -s -X PROPFIND -H "Content-Type: application/xml" -sD /dev/stderr https://mail.tegu.online:8443/addressbook
HTTP/1.1 207 Multi-Status
Server: nginx/1.16.1
Date: Sat, 09 May 2020 05:53:23 GMT
Content-Type: application/xml; charset=utf-8
Transfer-Encoding: chunked
Connection: keep-alive
Vary: Brief,Prefer
DAV: 1, 3, extended-mkcol, access-control, calendarserver-principal-property-search, 2, resource-sharing, calendar-access, calendar-proxy, calendar-auth

<?xml version="1.0"?>
<d:multistatus xmlns:d="DAV:" xmlns:s="https://sabredav.org/ns" xmlns:cal="urn:ietf:params:xml:ns:caldav" xmlns:cs="https://calendarserver.org/ns/" xmlns:card="urn:ietf:params:xml:ns:carddav">
  <d:response>
    <d:href>/addressbooks/user@tegu.online/default</d:href>
    <d:propstat>
      <d:prop>
        <d:resourcetype>
          <d:collection/>
          <card:addressbook/>
        </d:resourcetype>
        <d:getcontenttype>application/octet-stream</d:getcontenttype>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
  </d:response>
  <d:response>
    <d:href>/addressbooks/user@tegu.online/default/49e6be92-e235-4372-91cd-145673caed4f.vcf</d:href>
    <d:propstat>
      <d:prop>
        <d:getlastmodified>Tue, 04 Feb 2020 03:17:33 GMT</d:getlastmodified>
        <d:getcontentlength>207</d:getcontentlength>
        <d:resourcetype/>
        <d:getetag>"66098840ac4759f5bf12871c85899b33"</d:getetag>
        <d:getcontenttype>text/vcard; charset=utf-8</d:getcontenttype>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
  </d:response>
</d:multistatus>
```

```
</d:multistatus>
```

Видим единственную запись:

```
/addressbooks/user@tegu.online/default/49e6be92-e235-4372-91cd-145673caed4f.vcf
```

благодаря этому свойству:

```
<d:getcontenttype>text/vcard; charset=utf-8</d:getcontenttype>
```

Читаем файл:

```
$ curl --user "user@tegu.online:ThePassword" -s -X GET -i https://mail.tegu.online:8443/addressbooks/user@tegu.online/default/49e6be92-e235-4372-91cd-
HTTP/1.1 200 OK
Server: nginx/1.16.1
Date: Sat, 09 May 2020 05:56:19 GMT
Content-Type: text/vcard; charset=utf-8
Content-Length: 207
Connection: keep-alive
Last-Modified: Tue, 04 Feb 2020 03:17:33 GMT
ETag: "66098840ac4759f5bf12871c85899b33"
X-Content-Type-Options: nosniff

BEGIN:VCARD
VERSION:3.0
PRODID:-//apiserver//API Server DAV//EN
UID:36baeaba-08df-466b-8f10-84a9b3570613
ORG:atmail;;
EMAIL;TYPE=OTHER;PREF=1:Stan@atmail.com
N:Smith;Stan;;;
FN:Stan Smith
END:VCARD
```

## Вывод

Как видите, в среде стандартных протоколов есть все механизмы, чтобы сделать вашу почтовую систему безопасной, а также свести к минимуму процедуру настройки почтовых клиентов пользователей.