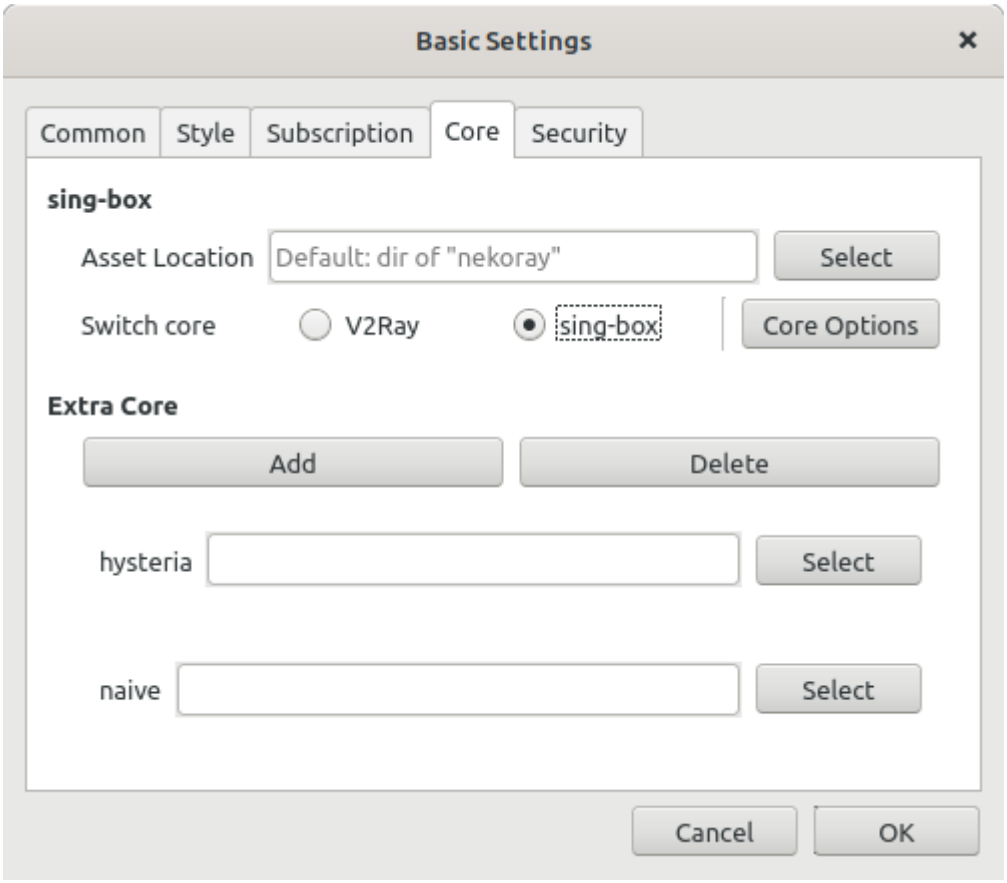
Bleeding-edge обход блокировок с полной маскировкой: настраиваем сервер и клиент XRay с XTLS-Reality быстро и просто

habr.com/ru/articles/731608

Deleted user April 25, 2023



Deleted-user 26 апр 2023 в 01:58
Простой

11 мин

573K
Туториал

В серии предыдущих статей я описывал, почему повсеместно используемые VPN- и прокси-протоколы такие как Wireguard и L2TP очень уязвимы к выявлению и могут быть легко заблокированы цензорами при желании, обозревал существующие гораздо более надежные протоколы обхода блокировок, клиенты для них, а также описывал настройку сервера для всего этого.

Но кое о чем мы не поговорили. Во второй статье я вскользь упомянул самую передовую и недетектируемую технологию обхода блокировок под названием **XTLS-Reality**, и пришло время рассказать о ней поподробнее, а именно - как настроить клиент и сервер для нее.

Кроме того, что этот протокол еще более устойчив к выявлению, приятным фактом будет и то, что настройка сервера XRay для XTLS-Reality гораздо проще, чем описанные ранее варианты - после предыдущих статей я получил довольно много комментариев типа "А что так сложно, нужен домен, нужны сертификаты, и куча всего" - теперь все будет гораздо проще.

XTLS-Reality

Коротко про **XTLS-Reality**. Это самое новое изобретение от авторов XRay. Про XRay (и его прородителя V2Ray, он же V2Fly) я рассказывал в предыдущей статье. XTLS-Reality поддерживается в последних релизах XRay, Sing-box и многих клиентах.

Он предназначен для защиты от выявления методом active probing. В отличие от старых протоколов (Shadowsocks, VMess, VLESS, и транспорта XTLS-Vision), определение “свой/чужой” здесь происходит еще на этапе TLS-хендшейка в момент чтения ClientHello. Если клиент опознан как “свой”, сервер работает как прокси, а если нет - вжух! - и TLS подключение передается на какой-нибудь другой абсолютно реальный хост с TLS (например, google.com или gosuslugi.ru), и таким образом клиент (или цензор, желающий методом active probing проверить, а что же прячется на том конце) получит настоящий TLS-сертификат от google.com или gosuslugi.ru и настоящие данные с этого сервера. Полное соответствие. Механизм определения "свой/чужой" во многом схож с механизмом работы Cloak, и позволяет достоверно определить подлинность клиента, но вместе с тем не вызывает подозрения у цензоров и устойчив к replay-атакам - со стороны систем анализа трафика это выглядит как подключение к настоящему популярному сайту, сервер отдает настоящий TLS-сертификат этого сайта, и вообще все (включая TLS fingerprint сервера) выглядит до предела аутентично и не вызывает подозрений. Еще XTLS-Reality может оказаться вариантом для обхода суровых корпоративных прокси с Man-in-the-Middle, которые перешифровывают весь трафик из сети своим сертификатом (нередко подобные прокси имеют список исключений для ресурсов с HSTS и certificate pinning, либо для экономии ресурсов, и подобрав правильный домен можно пролезть во внешнюю сеть без расшифровки трафика). Бонусом еще XTLS-Reality обычно используется в паре с XTLS-Vision, то есть мы имеем очень достоверно выглядящие паттерны трафика из-за отсутствия двойного шифрования TLS-in-TLS (и заодно еще очень высокую производительность, у меня между хостами в Москве и в центральной Европе XRay легко выдает >100 мегабит).

Единственный минус подобного решения - в отличие от более старых протоколов (VLESS без XTLS) нет возможности работать через websocket-транспорт, и, соответственно, через CDN типа Cloudflare. **Upd**: такая возможность есть если

использовать многоуровневую схему с SNI-proxy (типа haproxy) или второй IP-адрес - см. Особенности проксирования через CDN/Websocket/gRPC для обхода блокировок

Установка сервера XRay

А теперь настало время все это настроить. Дано: VPS на Linux (Debian или Ubuntu, на других дистрибутивах плюс-минус то же самое) с IPv4 или IPv6-адресом.

Установку XRay я уже описывал в предыдущей статье, поэтому здесь буду краток.

Можно установить XRay руками:

```
wget https://github.com/XTLS/Xray-core/releases/download/v1.8.1/Xray-linux-64.zip
mkdir /opt/xray
unzip ./Xray-linux-64.zip -d /opt/xray
chmod +x /opt/xray/xray
nano /usr/lib/systemd/system/xray.service
systemctl enable xray
```

▶ xray.service

А можно установить скриптом от разработчиков (почему-то по умолчанию он ставит старую версию 1.7.5, которая не поддерживает Reality, поэтому нужно явно указать более свежую):

```
bash -c "$(curl -L https://raw.githubusercontent.com/XTLS/Xray-install/046d9aa2432b3a6241d73c3684ef4e512974b594/install-release.sh)" @ install --version 1.8.1
```

Скрипт установит XRay и создаст для него systemd-юнит.

Настройка сервера XRay

Для настройки нам понадобится ряд параметров. Часть из них нам может сгенерировать сам XRay:

```
/usr/local/bin/xray uuid # /opt/xray/xray если устанавливали вручную
/usr/local/bin/xray x25519 # /opt/xray/xray если устанавливали вручную
```

На выходе вы получите UUID (идентификатор пользователя для протокола аутентификации VLESS), а также приватный и публичный ключи - запишите их, они вам понадобятся.

Еще один параметр, который нужен - short ID, он представляет собой просто шестнадцатиричное число (символы 0-9, a-g) длиной до 8 байт (16 символов) - можно набрать любую абракадабру типа "aabbccdd" или запустить `openssl rand -hex 8`

А вот дальше начинается самое интересное. Нам нужно найти сайт, под который мы будем маскироваться.

Требования довольно простые:

это должен быть иностранный сервер (вне РФ), не забаненный по домену Роскомнадзором, поддерживающий подключения по TLSv1.3 и HTTP/2, имеющий заглавную страницу, которая *не* переадресовывает на какой-нибудь другой домен. Если совсем упарываться, то неплохо было бы если бы IP-адрес был из диапазона того же облачного хостера, что и у вас, и чтобы сервер поддерживал Online Certificate Status Protocol (OCSP). Если вы не знаете, что вся эта фигня значит - не заморачивайтесь, выбирайте что-нибудь простое, например

- www.samsung.com:443
- www.googletagmanager.com:443
- www.asus.com:443
- www.amd.com:443
- www.cisco.com:443
- www.microsoft.com:443
- dl.google.com:443
- www.linksys.com:443
- www.nvidia.com:443

  и т.д.

Лучше всего выбирать что-нибудь из сети того же хостера, каким пользуетесь вы. Для этого есть специальный инструмент: https://github.com/XTLS/RealiTLScanner

Скачиваете его под Windows/Linux со страницы Releases, или собираете сами (go build).

Далее, запускаете как-то так:

```
./RealiTLScanner -addr IP_вашего_VPS -showFail
```

и ждете.

Сканер будет перебирать IP-адреса из той же подсети, что и ваш сервер, и пытаться к ним подключиться по TLS. Если он что-то найдет - вы это увидите. Пример (я сканирую рандомный IPшник):

```
89.116.243.206:443      TLS handshake failed: EOF
89.116.243.207:443      TLS handshake failed: EOF
89.116.243.208:443      ----- Found TLS v1.3    ALPN
CN=caprover.com,O=CapRover.com,L=Vancouver,ST=British
Columbia,C=CA,1.2.840.113549.1.9.1=#0c11696e666f40636170726f7665722e636f6d
89.116.243.209:443      TLS handshake failed: EOF
89.116.243.210:443      ----- Found TLS v1.3    ALPN    CN=patentpath.io
89.116.243.211:443      ----- Found TLS v1.3    ALPN    CN=vps3.gecon.pl
89.116.243.212:443      TLS handshake failed: EOF
89.116.243.213:443      TLS handshake failed: EOF
89.116.243.214:443      TLS handshake failed: EOF
89.116.243.215:443      TLS handshake failed: read tcp 192.168.136.132:55142-
>89.116.243.215:443: i/o timeout
89.116.243.216:443      ----- Found TLS v1.3    ALPN
CN=localhost,OU=none,O=none,L=Sometown,ST=Someprovince,C=US,1.2.840.113549.1.9.1=#
0c137765626d6173746572406c6f63616c686f7374
89.116.243.217:443      TLS handshake failed: EOF
89.116.243.218:443      TLS handshake failed: EOF
89.116.243.219:443      TLS handshake failed: EOF
89.116.243.220:443      TLS handshake failed: EOF
89.116.243.221:443      TLS handshake failed: EOF
89.116.243.222:443      ----- Found TLS v1.3    ALPN
89.116.243.223:443      ----- Found TLS v1.3    ALPN
CN=milapanel.milahosting.com
89.116.243.224:443      ----- Found TLS v1.3    ALPN    CN=vps-us.workx.dev
89.116.243.225:443      ----- Found TLS v1.3    ALPN    CN=www.google.com
89.116.243.226:443      ----- Found TLS v1.3    ALPN    CN=www.bookifynow.com
89.116.243.227:443      ----- Found TLS v1.3    ALPN    CN=next.tasosvl.cc
89.116.243.228:443      TLS handshake failed: EOF
89.116.243.229:443      ----- Found TLS v1.3    ALPN    CN=alpaca-dreams.com
89.116.243.230:443      TLS handshake failed: EOF
```

Если сканер нашел какие-то домены - попробуйте сходить на них браузером - должен открыться соответствующий сайт без каких-либо ошибок сертификатов. Если не открывается, или лезут ошибки - такой домен нам не подходит, а если открывается и ошибок нет - можно попробовать маскироваться под него.

Сервер выбрали, настало время редактировать конфиг. Если вы ставили XRay вручную то он будет лежать в /opt/xray/config.json, если скриптом - то в /usr/local/etc/xray/config.json.

Приводим его к следующему виду:

```
{
  "log": {
    "loglevel": "info"
  },
  "routing": {
    "rules": [],
    "domainStrategy": "AsIs"
  },
  "inbounds": [
    {
      "port": 23,
      "tag": "ss",
      "protocol": "shadowsocks",
      "settings": {
        "method": "2022-blake3-aes-128-gcm",
        "password": "aaaaaaaaaaaaaaaabbbbbbbbbbbbbbbb",
        "network": "tcp,udp"
      }
    },
    {
      "port": 443,
      "protocol": "vless",
      "tag": "vless_tls",
      "settings": {
        "clients": [
          {
            "id": "4c3fe585-ac09-41df-b284-70d3fbe18884",
            "email": "user1@myserver",
            "flow": "xtls-rprx-vision"
          }
        ],
        "decryption": "none"
      },
      "streamSettings": {
        "network": "tcp",
        "security": "reality",
                "realitySettings": {
                        "show": false,
                        "dest": "www.microsoft.com:443",
                        "xver": 0,
                        "serverNames": [
                                "www.microsoft.com"
                        ],
                        "privateKey":
"GOTPj_klK7_j_IvjxiCtyBL80RYotYSOdBBBSfFOMH4",
                        "minClientVer": "",
                        "maxClientVer": "",
                        "maxTimeDiff": 0,
                        "shortIds": [
                                "aabbccdd"
                        ]
                }
      },
      "sniffing": {
        "enabled": true,
        "destOverride": [
```

```
          "http",
          "tls"
        ]
      }
    }
  ],
  "outbounds": [
    {
      "protocol": "freedom",
      "tag": "direct"
    },
    {
      "protocol": "blackhole",
      "tag": "block"
    }
  ]
}
```

На что обратить внимание: в "serverNames" указан домен, под сервер которого вы маскируетесь (в данном случае www.microsoft.com), "id" в секции "clients" - это тот самый UUID, что мы сгенерировали выше. "privateKey" и первый элемент в массиве "shortIds" - это приватный ключ и short ID, что мы тоже сгенерировали выше. Публичный ключ не теряйте, он будет нужен на клиенте.

В этом конфиге так же на 23 порту настроен Shadowsocks-2022, на всякий случай, вдруг пригодится. Если не надо, или хочется полной маскировки - можно удалить этот элемент из "inbounds".

Перезапускаем еще раз xray:

```
$ systemctl restart xray
```

Проверяем что все нормально запустилось:

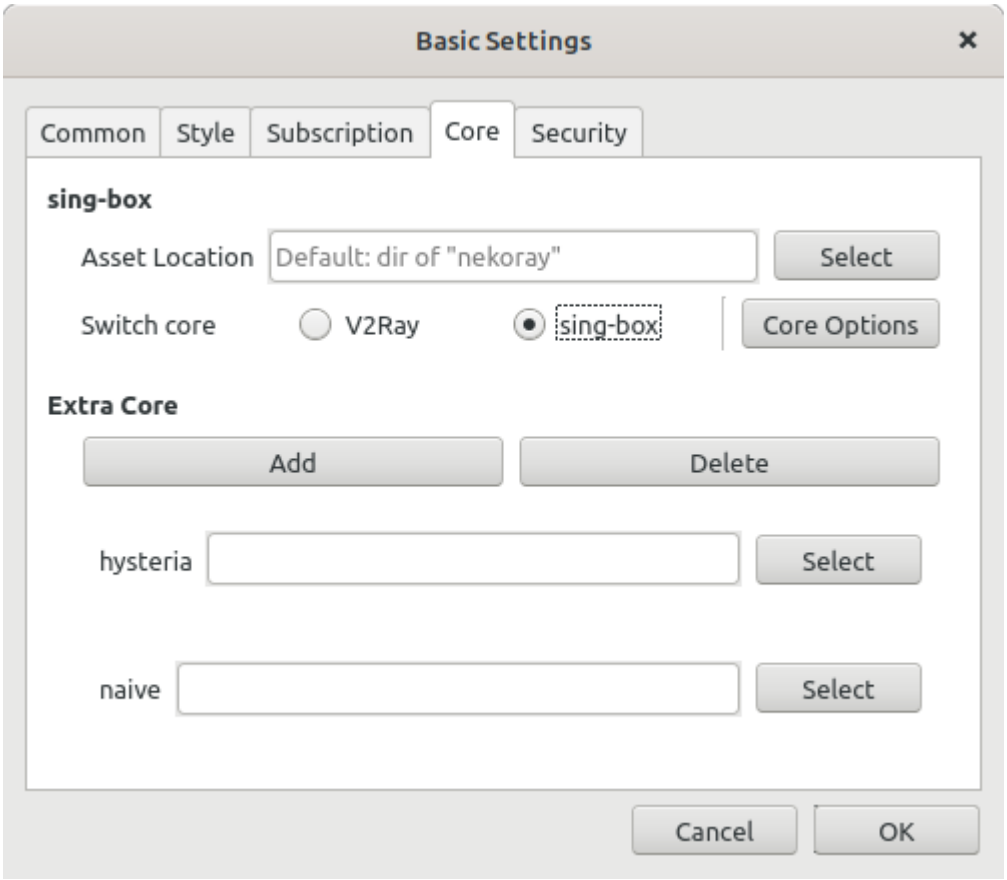
```
$ journalctl -u xray
```

Например, XRay может ругнуться что не удается распарсить JSON-файл, обычно это связано с лишними запятыми в конце {} блока, в этом случае он укажет, на какой строке ошибка. Исправляем ошибки, перезапускаем еще раз, и переходим к настройке клиентов.

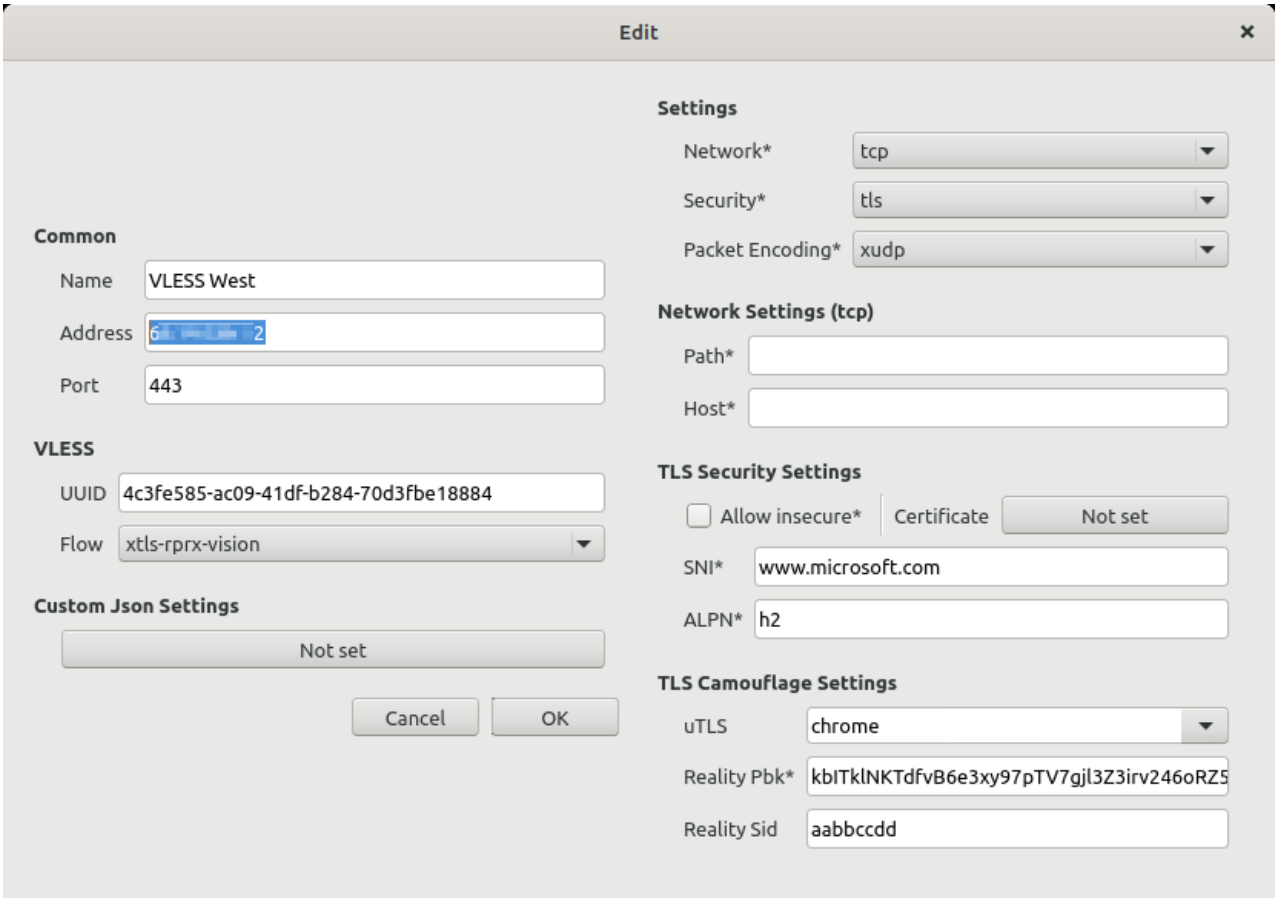
Настройка клиентов

Сначала **Nekobox** на десктопе (Windows, Linux, и есть неофициальные билды под MacOS).

Если вы раньше им не пользовались, нужно переключить его на использование движка sing-box, Preferences -> Basic Settings -> Core:



Идем в Server -> New profile и заполняем все вот так:

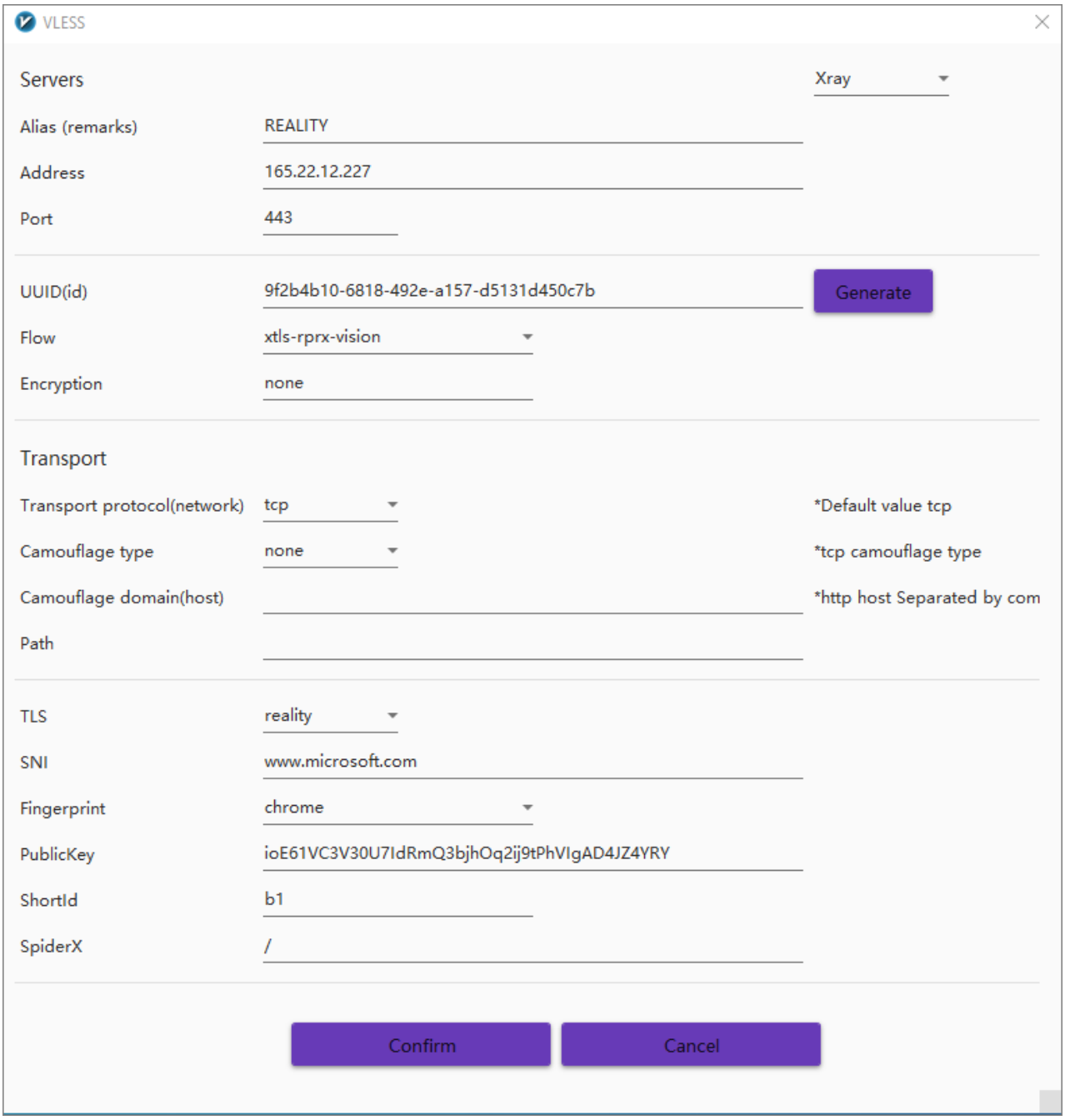


Address - IP-адрес вашего сервера, UUID - соответственно, UUID, SNI должен соответствовать домену, под который вы маскируетесь (один из списка "serverNames" из конфига сервера), uTLS - я выбираю Chrome (это маскировка клиента под обычный браузер), Reality Pbk - публичный ключ (не приватный, а второй, публичный), Reality Sid - shortId из конфига выше.

Сохраняем, кликаем правой кнопкой мыши на новый сервер в списке, жмем Start, и проверяем подключение выбрав там же Current Select -> URL test.

Если все нормально, то галочками "VPN Mode" или "System proxy" можно завернуть трафик всех приложений на прокси.

Настройка v2rayN под Windows аналогична, набор параметров тот же, вот скриншот (не мой, из гугла):



Автор Nekobox перестал собирать версии под macOS, поэтому я рекомендую использовать Wings X / FoXray. Настройки точно такие же.

Если вдруг вам нравятся Clash-based клиенты (например, Clash Verge под Windows, Linux, MacOS или для мобильных устройств), то нужно использовать ядро Clash.Meta и специальный конфиг для Clash. В случае с Clash Verge можно сделать так:

1. Settings -> Clash Core -> Выбрать Meta
2. Сохранить конфиг в какой-нибудь локальный файл:

▶ clash-reality.yml

3. Profiles -> New - Type : Local -> выбрать ваш файл и кликнуть по нему в окне Clash
4. Proxies -> выбрать ваш новый прокси на вкладке Global -> кликнуть по полю справа чтобы протестировать подключение, должно появиться значение пинга
5. Settings -> System proxy: вкл - после этого трафик всей системы пойдет через прокси. Можно использовать и TUN, но для этого надо запускать Clash Verge от рута.

Далее, мобильные клиенты. Вариант раз: в Nekobox или в v2ray кликнуть правой кнопкой мыши на ваш сервер из списка, выбрать Share -> QR code или Link, и получить ссылку или QR-код, которые потом можно отсканировать/вставить в мобильные клиенты. Либо вбить все те же данные руками, вот как это выглядит в андроидовском v2rayNG (версия из Google Play еще не обновилась и не умеет работать с Reality, скачиваем APK с Гитхаба):

▶ скриншоты

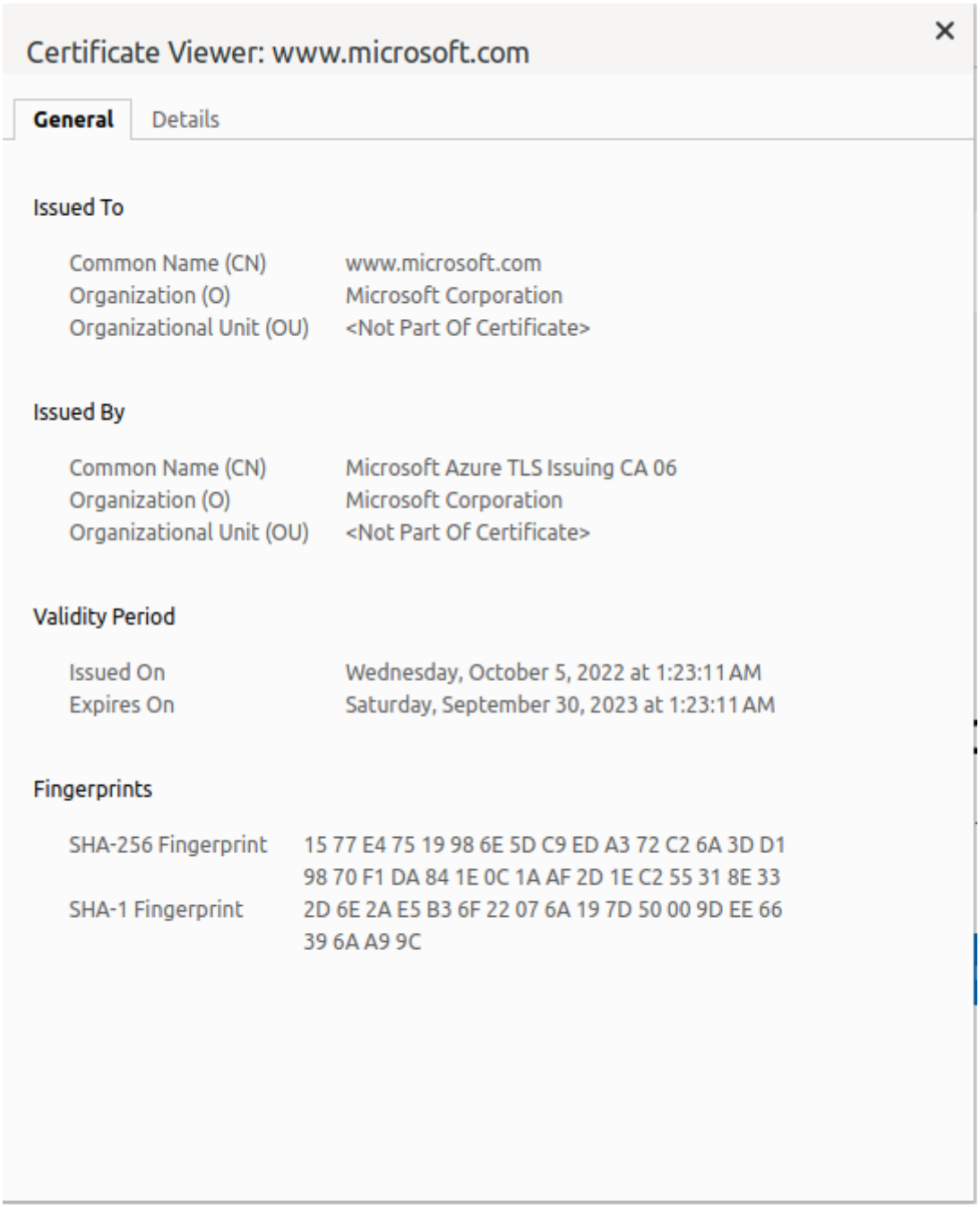
Под iOS я рекомендую использовать Shadowrocket (3$) или Wings X / FoXray (он бесплатный). Настройки подключения полностью аналогичны описанному выше.

Советы бывалых

1. Очень рекомендуется настраивать на клиентах правила маршрутизации (пример в комментариях), чтобы трафик до .ru-доменов и хостов с российскими IP шел напрямую, а не через прокси (в клиентах для такого поставляется GeoIP база данных).
2. Обязательно используйте uTLS на клиентах, выставляя правильный TLS fingerprint (например, Chrome).
   Если при использовании XTLS вы почему-то не можете подключиться, в логах сервера видна ошибка типа "failed to use xtls-rprx-vision, found outer tls version 771", попробуйте сменить версию uTLS. У меня, например, при выборе "android" клиент не подключается, а при выборе "chrome" все окей.

3. Для увеличения производительности можно настроить на сервере Bottleneck Bandwidth и Round-trip propagation time (BBR) congestion control algorithm:
   echo "net.core.default_qdisc=fq" >> /etc/sysctl.conf
   echo "net.ipv4.tcp_congestion_control=bbr" >> /etc/sysctl.conf
   sysctl -p

4. Чтобы проверить, что маскировка работает как надо, добавьте IP-адрес вашего сервера и домен, под который вы маскируетесь, в hosts-файл (на Linux это /etc/hosts, на Windows это c:\windows\system32\drivers\etc\hosts), например, "38.25.63.10 www.microsoft.com", и после этого попробуйте зайти на этот адрес браузером - должна открыться настоящая страница этого домена с настоящим TLS-сертификатом:



Другой вариант: использовать CURL.
`curl -v --resolve www.microsoft.com:443:151.101.65.69 https://www.microsoft.com` (вместо 151.101.xx.xx должен быть IP вашего сервера)

О VLESS

Мы настроили протокол VLESS. Иногда в интернете можно встретить утверждения, что, мол VLESS не шифрует данные, а значит он небезопасен. Это не так. То, что VLESS не предусматривает шифрования на уровне протокола, не значит, что данные передаются в нешифрованном виде. VLESS всегда работает поверх TLS, трафик шифруется именно механизмами TLS, а не самого VLESS. Никакой проблемы с безопасностью тут нет, все секьюрно :) То же самое с XTLS. XTLS отключает свой слой шифрования только в случае, если определяет, что обмен между пользователем и конечным сервером *уже* зашифрован TLS v1.3.

О надежности

Как сделать хорошую, правильную маскировку для XTLS-Reality? Внимание к мелочам.

1. Выбирайте домен для маскировки от сайта, который хостится у того же хостера, что и вы (см. начало статьи).

2. Перевесьте SSH на вашем сервере с 22 порта на какой-нибудь другой сильно повыше, а то слишком палевно

3. Если вы используете панель типа X-UI или 3X-UI - то перевесьте ее тоже со стандартного порта на какой-нибудь нестандартный сильно повыше. В идеале стоит вообще заставить ее слушать на 127.0.0.1 (localhost), а подключаться к ней через SSH: например, если панель у вас на 127.0.0.1 и порту 48888, то сделав
   `ssh -L 8080:127.0.0.1:48888 user@serveradd -p <ssh_port>`
   вы сможете попасть на панель пройдя браузером по адресу
   http://127.0.0.1:8080

4. Сделайте проброс порта не только на 443/TCP-порт (его делает XTLS-Reality), а еще на 443/UDP и 80/TCP до сервера, под который вы маскируетесь. Например, если вы маскируетесь под www.microsoft.com, то отрезолвте его IP-адрес (с помощью nslookup, ping или какого-нибудь онлайн-сервиса), а потом добавьте правила iptables (можно засунуть в /etc/rc.local, если он у вас есть - см. инструкции для вашего Linux-дистрибутива):
   iptables -t nat -A PREROUTING -i eth0 -p udp --dport 443 -j DNAT --to-destination fake_site_ip:443
   iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j DNAT --to-destination fake_site_ip:80
   (вместо eth0 должен быть ваш сетевой интерфейс, иногда бывает ens3, например).

5. Если ваш хостер позволяет менять PTR-записи для IP-адресов (так называемые "обратные DNS"), то поменяйте ее на такую, какая есть у IP-адреса сайта, под который вы маскируетесь, или хотя бы просто на сам этот домен.

О клиентах

Весьма важно настроить на клиентских устройствах правила, чтобы доступ к внутренним (российским, если вы в РФ) ресурсам не шел через прокси-сервер.

Примеры настроек разных клиентов для этого есть в FAQ:
FAQ по Shadowsocks/XRay/XTLS/Reality/Nekobox/etc. для обхода блокировок

Там же есть ответы на многие другие частые вопросы и советы по устаранению проблем.

На этом всё.

Удачи, и да прибудет с вами сила.

Если вы хотите сказать спасибо автору — сделайте пожертвование в один из благотворительных фондов: "Подари жизнь", "Дом с маяком", "Антон тут рядом".