Linux and more.
Ssh as proxy

Artem Trunov for Ozon Masters

Сегодня на занятии

- Использование SSH для проброса трафика
  - Проброс одного порта
  - Socks proxy

Internet vs. Intranet

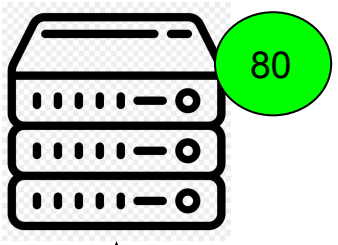
- Интранет - сегмент сети организации, закрытый для доступа снаружи.
- Разные сервисы для сотрудников, базы данных, кластеры для обработки данных, анутренней облако
- Веб сервер - уязвим
  - Сам код сервера (nginx, apache)
  - Фреймворк приложения (Django, Wordpress)
  - Само приложение
- Некоторые сервисы не предназначены для открытого размещения
  - Кластер Hadoop

Доступ в Интранет

дома

80

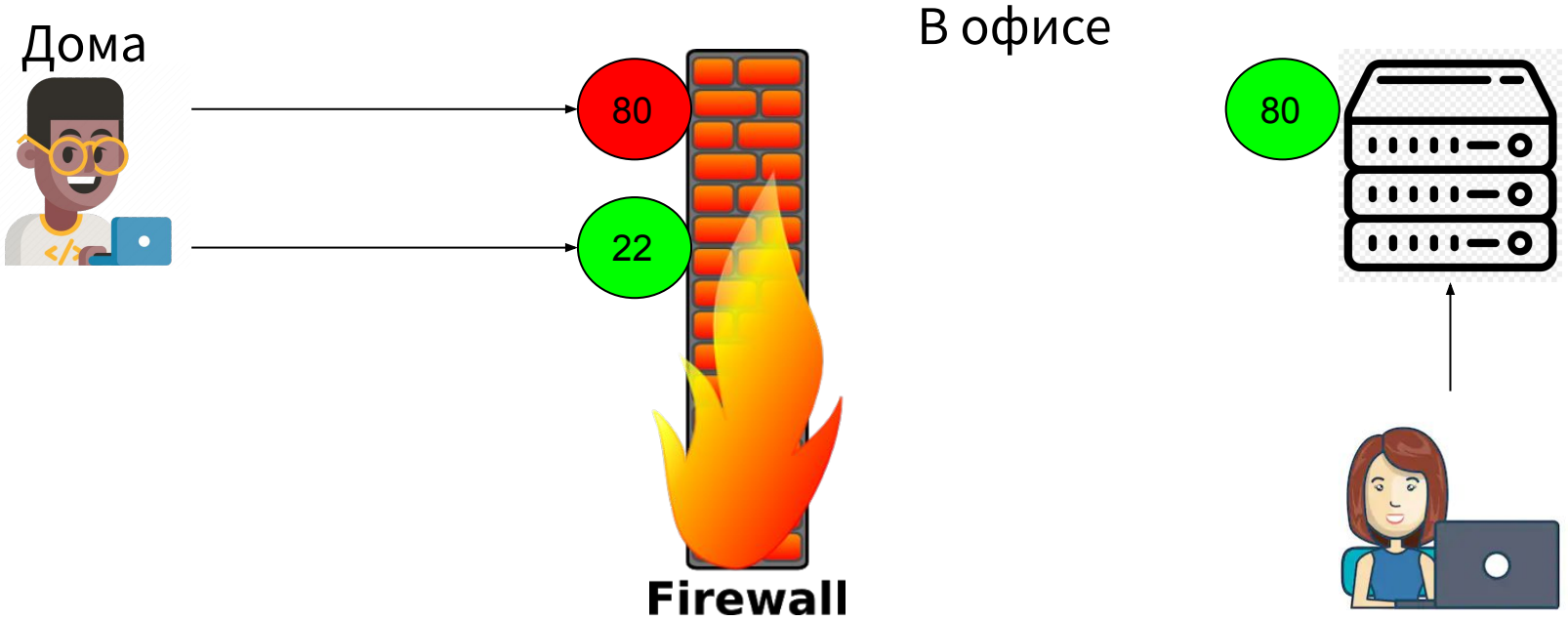
В офисе



shutterstock.com • 512458387

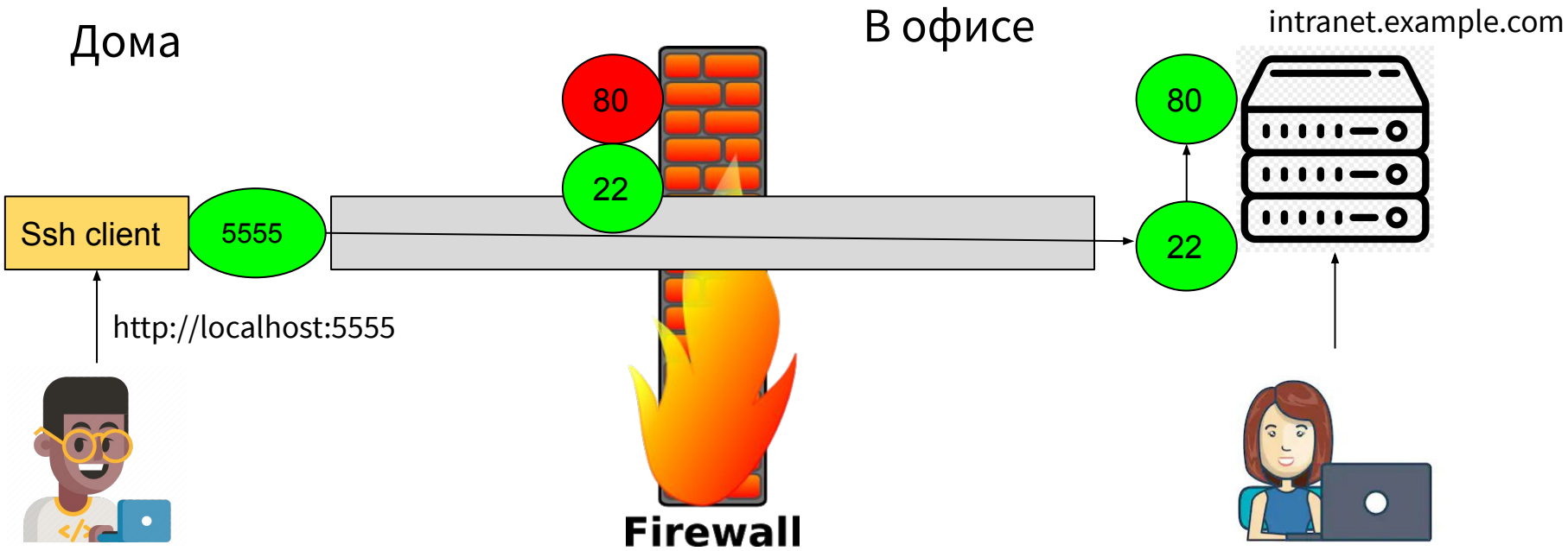
Доступ извне запрещен

Ssh - безопасен (с оговорками)



Доступ извне может быть разрешен, так как ssh предоставляет и хороший механизм аутентификации и шифрование трафика

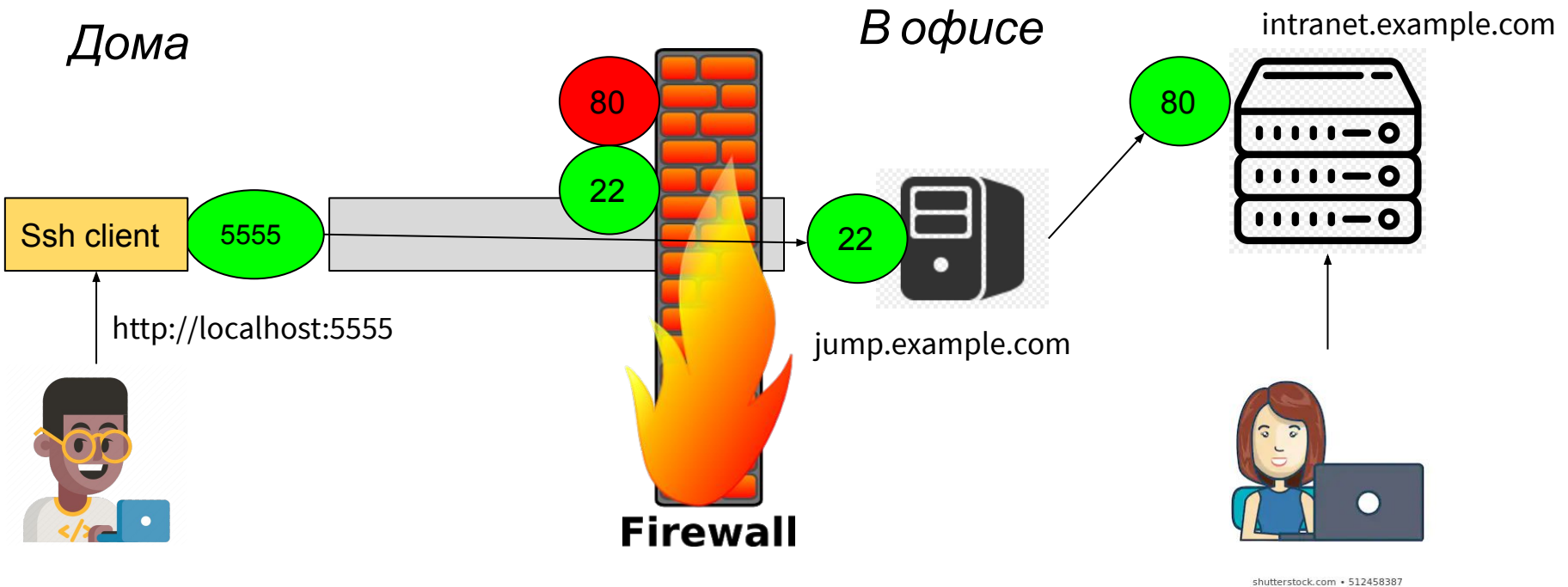
Ssh туннель на веб-сервер в интранете




```
ssh -L 5555:localhost:80 user@intranet.example.com
```

localhost = такое специальное имя для доступа к сервисам хоста с него самого

Ssh туннель на jump host с пробросом на другой




```
ssh -L 5555:intranet.example.com:80 user@jump.example.com
```

Практика

- Сделаем ssh-туннель с вашего пк или лаптопа на bigdatamasters.ml
- С помощью браузера на вашем пк/лаптопе зайдем на веб-сервер, запущенный на bigdatamasters.ml
- Сделаем другой тоннель по схеме с jump host
- Зайдем на веб-сервер на втором нашем сервере.

Задание

Попробуйте зайти браузером на вашем пк/лаптопе по адресу:

http://bigdatamasters.ml:5000

Получается?

Пробросьте порт на наш сервер. Запустите на вашем пк/лаптопе:

```
ssh -L 5555:localhost:5000 ваш_логин@bigdatamasters.ml
```

Зайдите браузером (на вашем пк) на http://localhost:5555

Введите ваш персональный аккаунт (Github nick), чтобы посмотреть время самого первого логина на сервер.

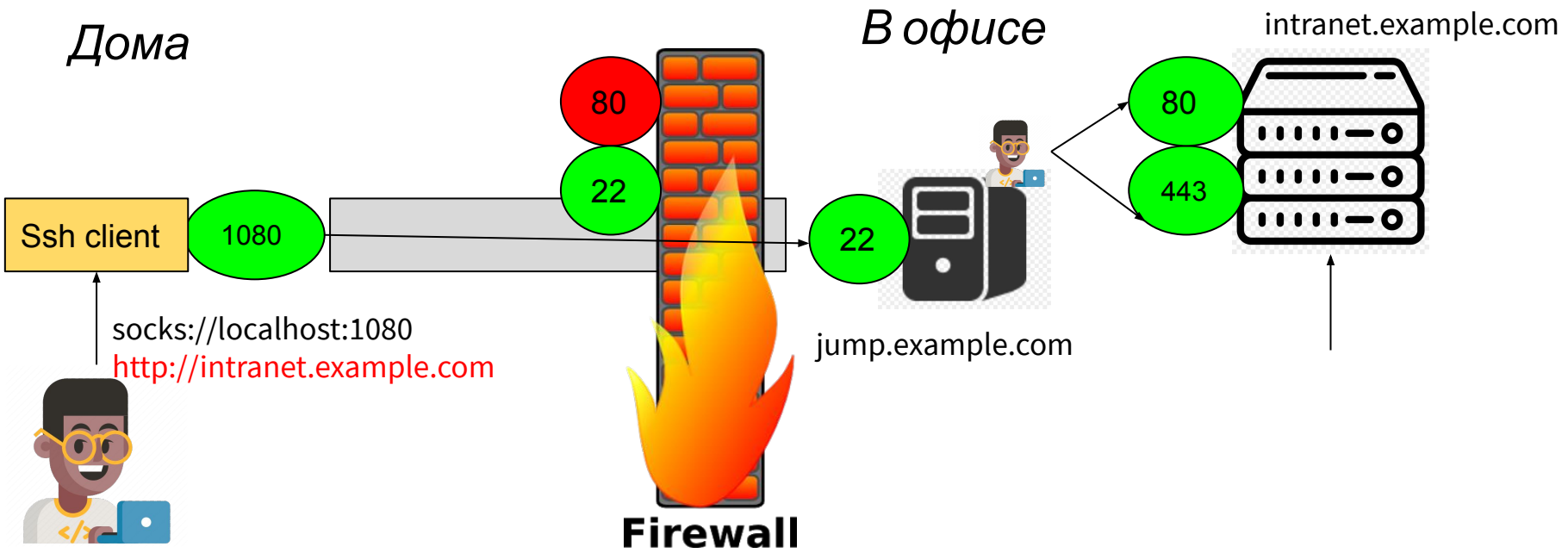
Проброс порта

- ssh -L local_port:remote_host:remote_port user@example.com
  - -L 80 - эквивалентно -L 80:localhost:80
- Открывает **local_port** на вашем ПК/лаптопе
- Перенаправляет трафик с него на **remote_host:remote_port**
  - **remote_host** может быть именем сервера в интранете, ip-адресом, или localhost
- Авторизация по ssh
- Канал шифруется
- Если надо пробросить дополнительный порт
  - Используйте -L несколько раз в одной команде.
- А если неизвестно какой порт? А если неизвестно на какой-сервер? Если все порты надо и все серверы в интранете?

Socks прокси

- Проброс всего трафика приложения по тоннелю.
- Позволяет обращаться к серверам интранета по их настоящим именам и их реальным портам
  - А не localhost и какой-то локальный порт как при пробросе порта
- Конфигурируется для каждого приложения отдельно, и это приложение должно поддерживать socks прокси
  - Браузер
  - Телеграм-бот
  - pip install -U requests[socks]
- У socks прокси авторизация опциональна, некоторые клиенты поддерживают только локальный сокс-сервер

Ssh в качестве прокси



1. `ssh -D 1080 user@jump.example.com`
2. `chromium-browser --proxy-server=socks://localhost:1080`

Конфигурация приложения на пк/лаптопе

- `chromium-browser --proxy-server=socks://localhost:1080`
  - **chrome.exe** в командной строке Windows
- Firefox, IE требуют изменения системных настроек
  - Весь трафик браузера пойдет через прокси
- Совет - используйте chrome только для прокси, остальные браузеры без прокси
- Другие приложения - смотрите документацию
  - Python requests: https://requests.readthedocs.io/en/master/user/advanced/#socks

Задание. Socks proxy

- На вашем пк запустите тоннель с сокс-прокси
  - **`ssh -D 1080 bigdatamasters.ml`**
- Сконфигурируйте ваш браузер для использования сокс прокси на порту 1080
  - Проще всего хром, но надо закрыть все его окна, потом запустить с опцией сокс

В браузере запустите поиск “my ip”, получите внешний адрес нашего сервера.

На нашем сервере запустите команду **whois** с IP-адресом нашего сервера в качестве аргумента и посмотрите, какому провайдеру принадлежит этот адрес.

Далее, запустите команду **whois** с именем нашего сервера (bigdatamasters.ml) и посмотрите, кто регистратор домена.

Запуск браузера в терминале Windows

"C:\Program Files\Google\Chrome\Application\chrome.exe"
--proxy-server=socks://localhost:1080

Запуск браузера в терминале Mac OS

/Applications/Google\ Chrome.app/Contents/MacOS/Google\ Chrome

Что может пойти не так

- Убедитесь, что запущено только одно окно Хрома
  - Иногда необходимо убить последнее окно через диспетчер задач
- Иногда необходимо убить телеграм-десктоп

Задание. Ssh config

- Ssh config для проброса порта

```
LocalForward 5555 наш_сервер:80
```

- Ssh config для сокс-прокси

```
DynamicForward 1080
```

Socks proxy vs. обычный http proxy vs. VPN

- Простой http(s) proxy - без особых мер безопасности.
  - Хорошо подходит для корпоративной сети, для ограничения доступа сотрудников в интернет
  - Можно настроить на телефоне (отдельно для wi-fi, мобильного трафика)
  - Детектируется (=> блокируется)
- VPN - перенаправляет **весь** трафик через VPN сервер
  - А вы доверяете вашему провайдеру VPN? Риторический вопрос. :)
  - Подходит для телефонов, легко конфигурируется, в том числе на каждое приложение
  - Самостоятельно - OpenVPN на арендованном сервере
  - Детектируется (=> блокируется)
- Socks proxy
  - На каждое приложение в отдельности
  - Хорошо подходит для браузера
  - Самостоятельно - арендованный Linux сервер, минимальная конфигурация.
  - На телефоне - not so easy. Privoxy? (http-to-socks)