Мужчины...

Продолжим.

SSH. Ключи.

- Аутентификация по ключам намного безопаснее, чем пароль
- есть ключ публичный и закрытый (создаются парой)

пропащая женщина

твоя мечта

пригодятся обе

Это твой компьютер

private key

он тут и больше нигде и никому!

(можно даже зашифровать, так надежнее)

server1

public key

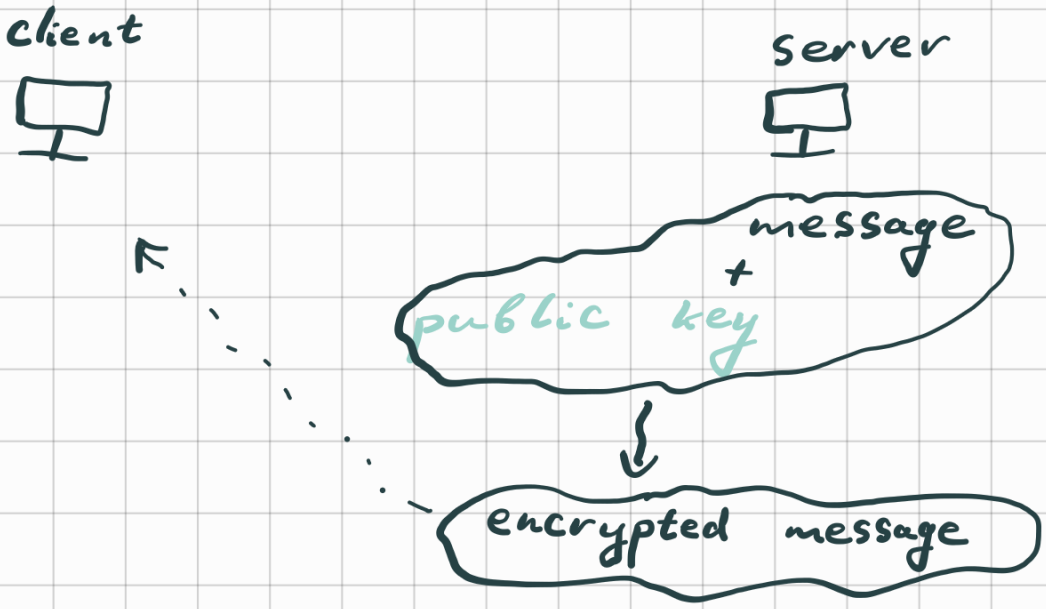
server2

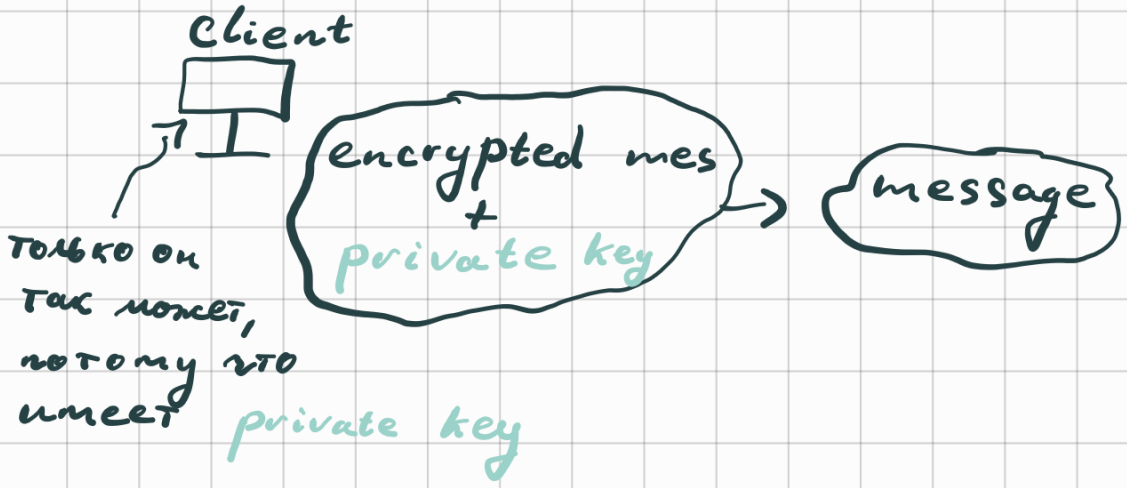
public key

server3

public key

парный к приватному публичный ключ располагаем на всех компах, к которым имеем доступ в папку .ssh/authorized_keys





безопаснее (использует этот алгоритм)

создание публичного и приватного ключей:

```
mkdir ~/.ssh
chmod 700 ~/.ssh
ssh-keygen -t rsa
```

Напоминаю базу:

ssh user@hostname -p port

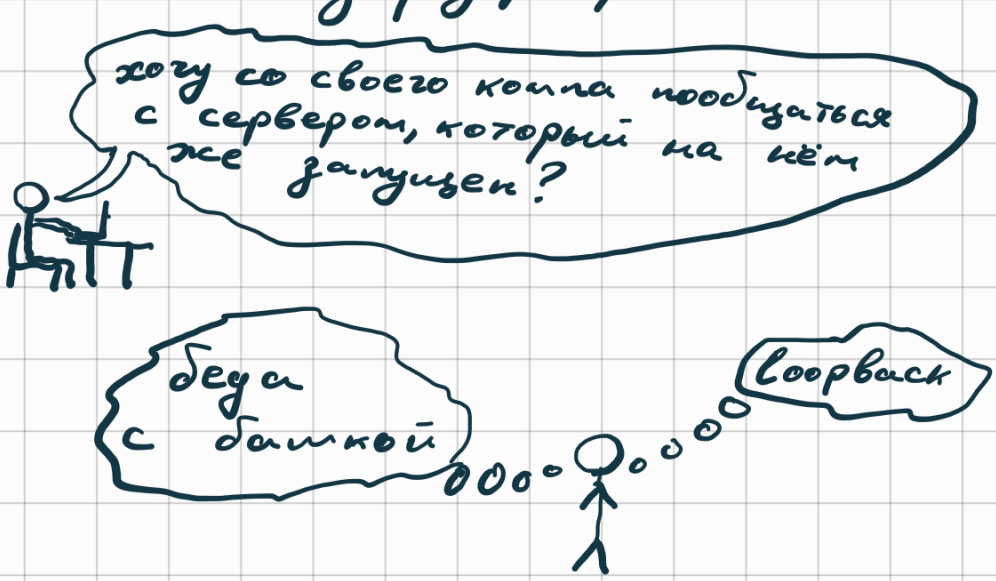
-i identity_file – выбирает файл, из к-го читается приватный ключ для RSA или DSA аутентификации

How to : telnet

telnet host [port] – коммуникация с другим хостом, используя протокол TELNET.

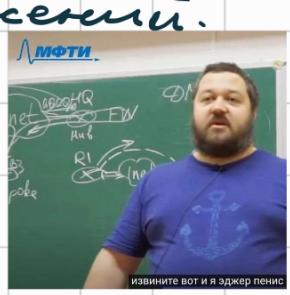
Замечание: протокол никак не шифруется!

Справка: localhost - это имя хоста, которое относится к текущему компьютеру. Имя localhost зарезервировано для loopback целей.



SSL TLS (Transport Layer Security)

Кратко: протокольный уровень приложений.



TLS handshake:

- Договариваемся о используемых шифрах
- Сервер светит своим сертификатом, в к-ом есть его публичный ключ
- Клиент смотрит, что сертификат не лажа
- клиент: генерирует random number
  шифрует публичным ключом
  шлёт серверу

  сервер: расшифровывает random number своим приватным ключом
- обе стороны используют это число, чтобы сгенерировать сессионный ключ для encryption/decription

или

Алгоритм Диффи-Хеллмана

Как:

openssl command [command-opt] [cmd-args]

OpenSSL - это набор криптографических инструментов, реализующий протоколы SSL и TLS. Посвящён криптографическим стандартам, к-е в них используются.

How to: openssl s_client

SSL/TLS клиент. Подключается к удаленному хосту, используя SSL/TLS

на OTW целая книга(хоть и небольшая) по поводу теста сетей юзая OpenSSL ахуеть

Пример:

openssl s_client -connect servername:port

Port scanner

Приложение для обнаружения открытых портов на сервере.

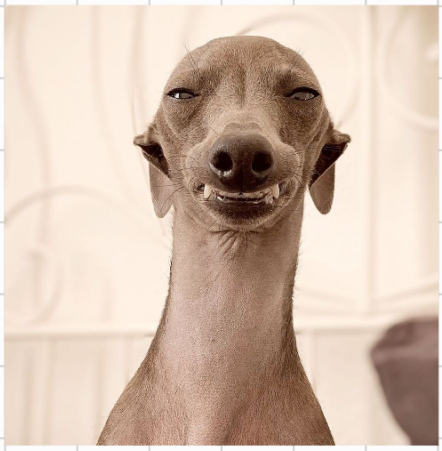
portscan - процесс, к-ый посылает клиентские

запросы на ряд портов хоста в поиске активного

(это не nefarious process 😑)

подлый

собака-подозревака ⤳



В большинстве случаев port scan исп-ся не для атак, а чтобы понять, какие сервисы работают на сервере

port sweep - сканирование с целью нахождения специфичного номера слушающего порта.

Результат сканирования:

1. Open or Accepted: хост слушает этот порт
2. Closed or Denied or Not Listening: хост послал ответ, что соединение к порту отвергнуто
3. Filtered, Dropped or Blocked: нет ответа от хоста

Виды сканирования:

- TCP. Пытается провести TCP handshake. Если получается, сразу закрывает соед-е
- SYN. Шлём SYN пакет ждем SYN-ACK ответ

Посылает RST, завершая связь до окончания рукопожатия

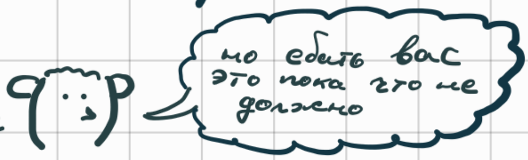
• UDP. Если порт закрыт, то получаем ICMP об этом

неточны:
• ACK
• Window

почитай про TCP

• FIN. Закрытые порты отвечают на FIN пакет RST пакетом, открытые порты их игнорируют

Есть и другие виды



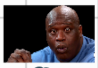
Как?

Давайте потрогаем ~~женские титьки~~ nmap.

быстрый скан портов:

nmap <target> -p portrange

↑ (1 - 35000)

Если внезапно  заинтересовал вопрос, а как же проверить, есть ли ssl на данном порте, то

ssl-cert скрипт позволяет проверить серти-

фикат для определённого сервера:

```
nmap --script ssl-cert -p 443 google.com
```