О криптографии

Елена Киршанова

https://crypto-kantiana.com/elena.kirshanova/

Кто я


arch Google Maps
e travel times, traffic and nearby places
'13: Математик
Baltic Sea
North Sea
Estonia
Riga
Latvia
Lithuania
Minsk
Belarus
Poland
Warsaw
Denmark
Hamburg
Berlin
Edinburgh
United Kingdom
Isle of Man
Dublin
Ireland
Manchester
Liverpool
London
Amsterdam
Netherlands
Brussels
Cologne
Germany
Belgium
Frankfurt
Luxembourg
Prague
Czechia
Paris
Munich
Vienna
Slovakia
Budapest
Austria
Hungary
Kyiv
Київ
Ukraine
Moldova
Chisinau
Odesa
Одеса
France
Switzerland
Milan
Slovenia
Zagreb
Croatia
Belgrade
Београд
Romania
Bucharest
Bosnia and Herzegovina
Serbia
Sarajevo
Sofia
София
Bulgaria
Podgorica
Подгорица
Kosovo
North Macedonia
Tirana
Albania
Italy
Rome
Monaco
Andorra
Barcelona
Porto
Madrid
Valencia
Bay of Biscay
Black Sea
İstanbul
Bursa
Ankara
Google
rrhenian Sea


’13: Математик
’16: PhD Math
Estonia
Baltic Sea
Riga
Latvia
North Sea
Lithuania
Edinburgh
United Kingdom
Denmark
Isle of Man
Minsk
Manchester
Dublin
Liverpool
Ireland
Belarus
Poland
Berlin
Warsaw
Amsterdam
Netherlands
London
Brussels
Germany
Belgium
Kyiv
Київ
Luxembourg
Paris
Ukraine
Slovakia
Munich
Vienna
Budapest
Austria
Moldova
Chisinau
Odesa
Одеса
Hungary
France
Switzerland
Slovenia
Zagreb
Milan
Romania
Belgrade
Београд
Croatia
Bucharest
Bosnia and Herzegovina
Serbia
Bay of Biscay
Monaco
Sarajevo
Sofia
София
Black Sea
Italy
Andorra
Podgorica
Подгорица
Kosovo
Bulgaria
Barcelona
Rome
Porto
North Macedonia
Tirana
İstanbul
Albania
Madrid
Bursa
Ankara
Valencia
Google
rrhenian Sea


rch Google Maps
travel times, traffic and nearby places
’13: Математик
’16: PhD Math
’16-’19: PostDoc
Baltic Sea
Estonia
Riga
Latvia
Lithuania
Minsk
Belarus
North Sea
Edinburgh
United Kingdom
Isle of Man
Dublin
Ireland
Manchester
Liverpool
Denmark
Berlin
Poland
Warsaw
Amsterdam
Netherlands
London
Germany
Belgium
Luxembourg
Munich
Vienna
Slovakia
Budapest
Austria
Hungary
Switzerland
France
Slovenia
Croatia
Belgrade
Београд
Romania
Bucharest
Moldova
Chisinau
Odesa
Одеса
Ukraine
Kyiv
Київ
Bay of Biscay
Sarajevo
Sofia
София
Bulgaria
Black Sea
Podgorica
Подгорица
Kosovo
North Macedonia
Tirana
Albania
Italy
Rome
Andorra
Barcelona
Porto
Madrid
Valencia
İstanbul
Bursa
Ankara
Google


rch Google Maps
travel times, traffic and nearby places
Baltic Sea
Estonia
Riga
Latvia
North Sea
Edinburgh
United Kingdom
Isle of Man
Dublin
Ireland
Manchester
Liverpool
Denmark
Lithuania
Minsk
Belarus
Poland
Warsaw
Berlin
Amsterdam
Netherlands
London
Germany
Belgium
Luxembourg
Kyiv
Київ
Ukraine
Munich
Vienna
Slovakia
Austria
Budapest
Hungary
Moldova
Chisinau
Odesa
Одеса
France
Switzerland
Slovenia
Croatia
Belgrade
Београд
Romania
Bucharest
Bay of Biscay
Sarajevo
Sofia
София
Bulgaria
Black Sea
Andorra
Barcelona
Porto
Madrid
Rome
Italy
Podgorica
Подгорица
Kosovo
North Macedonia
Tirana
Albania
İstanbul
Bursa
Ankara
Google
Valencia
'13: Математик
'16: PhD Math
'16-'19: PostDoc

Современное крипто **НЕ** про:

Современное крипто **НЕ** про:

© Wikipedia

взлом пентагона

Современное крипто **НЕ** про:

© Wikipedia

взлом

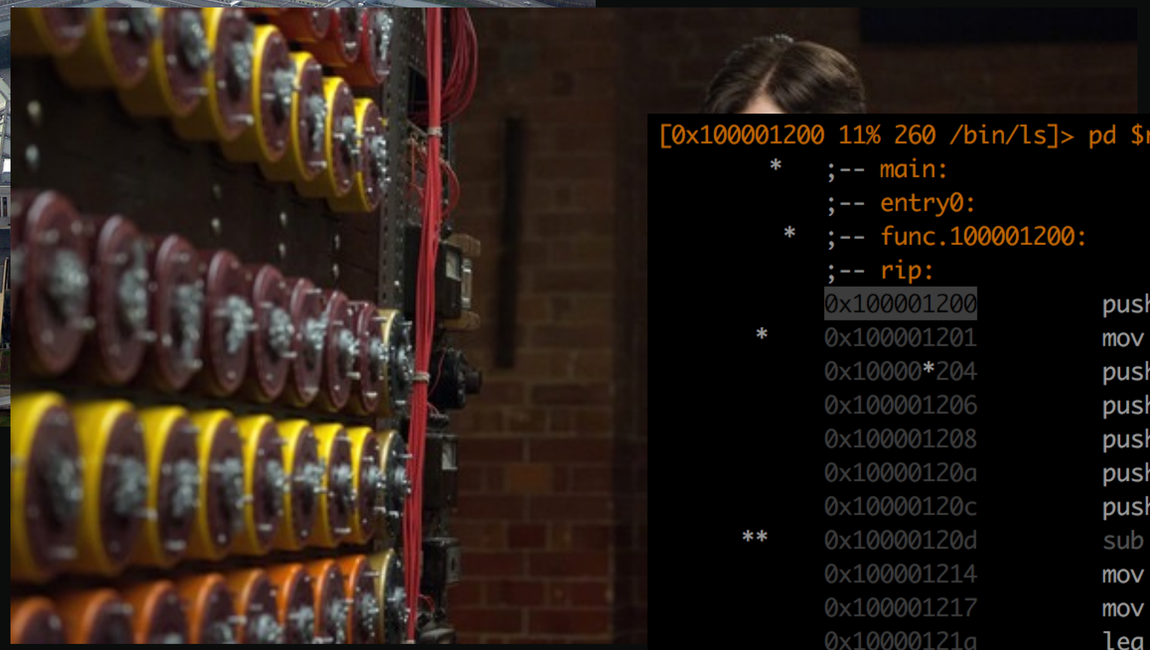
© Allstar

Энигму

Современное крипто **НЕ** про:

© Wikipedia

взлом



© Allstar

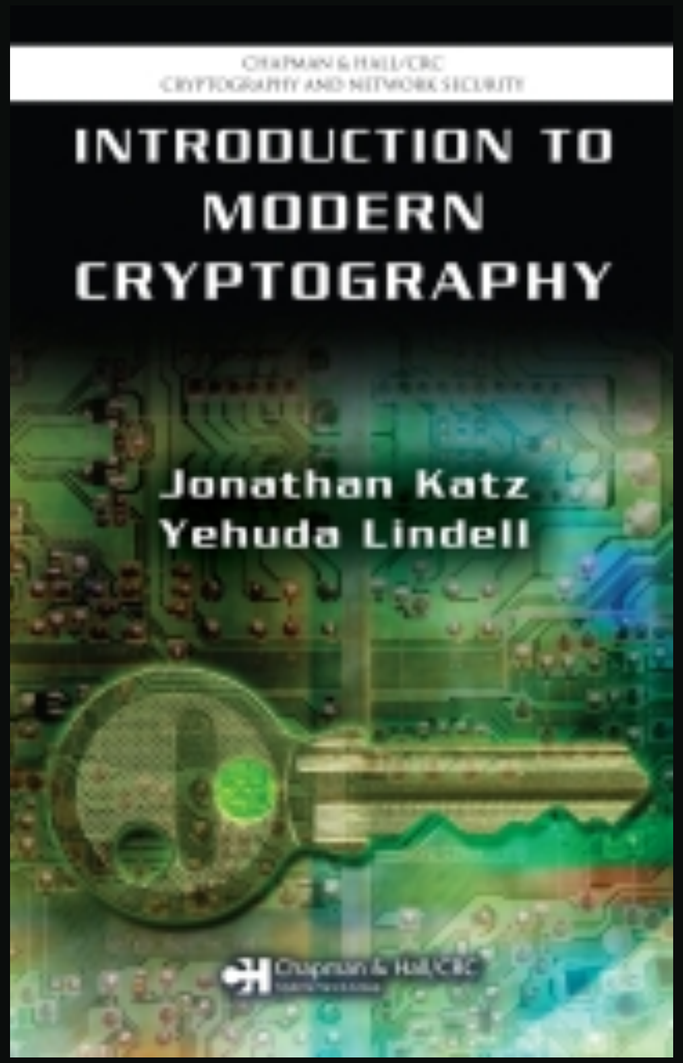
Энигм

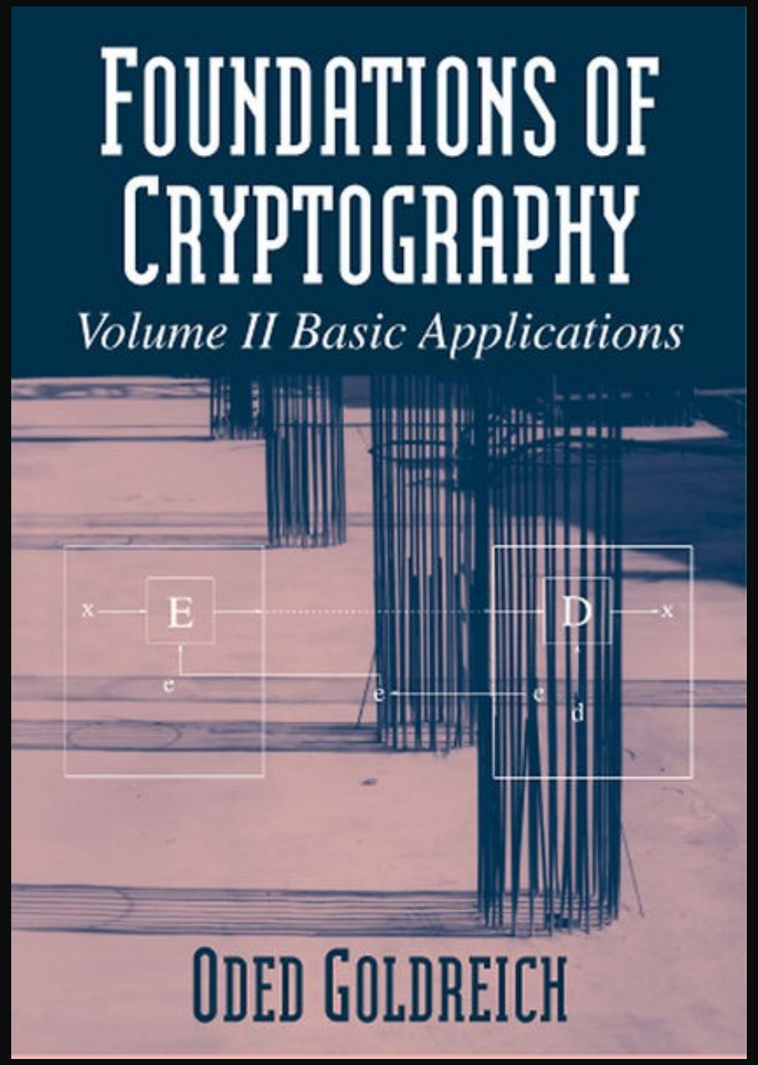
```
[0x100001200 11% 260 /bin/ls]> pd $r @ main
        ;-- main:
        ;-- entry0:
        ;-- func.100001200:
        ;-- rip:
            0x100001200          push rbp
            0x100001201          mov rbp, rsp
            0x10000*204          push r15
            0x100001206          push r14
            0x100001208          push r13
            0x10000120a          push r12
            0x10000120c          push rbx*
            0x10000120d          sub rsp, 0x618
            0x100001214          mov r15, rsi *
            0x100001217          mov r14d, edi
            0x10000121a          lea rax, rbp - 0x240
            0x100001221          mov qword [rbp - 0x30], rax
            0x100001225          test r14d, r14d
        ┌─< 0x100001228 [1]      jg 0x10000122f
        │   0x1000012*a [2]      call sym.func.100004*00
        └─>*0x10000122f          le* rsi, 0x100004af0
            0x100001236          xor *di, edi
            0x100001238 [3]      call sym.imp.setlocale
            0x10000123d          mov edi, 1
```

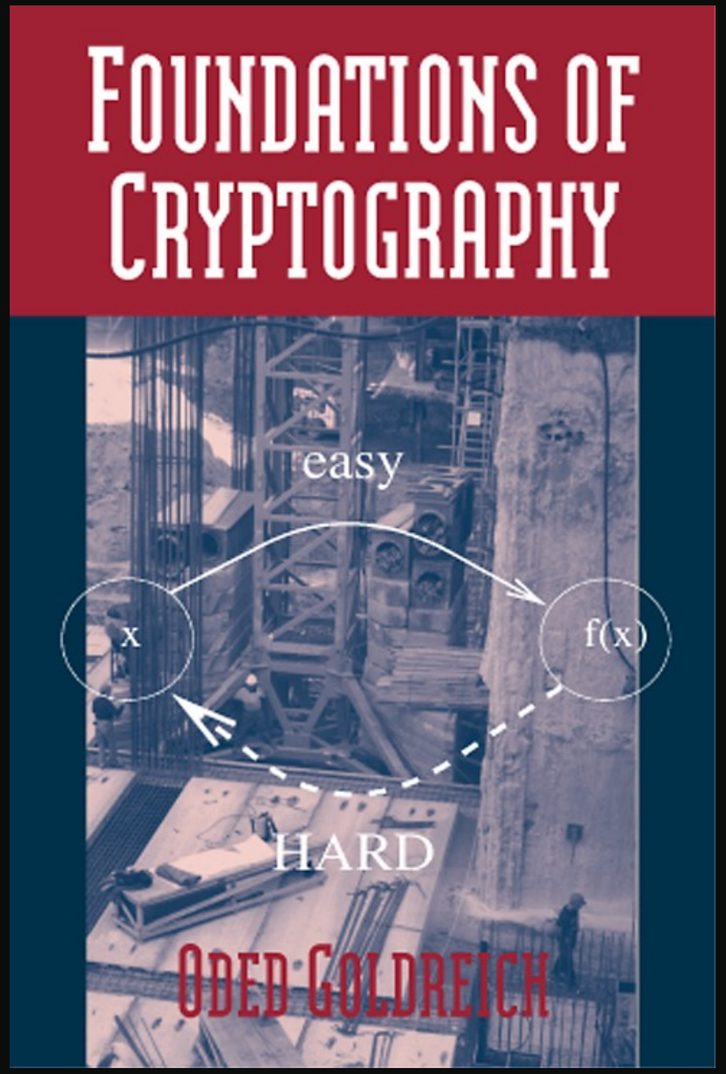
© https://thehacktoday.com/

реверс-инжиниринг

Современное крипто про:

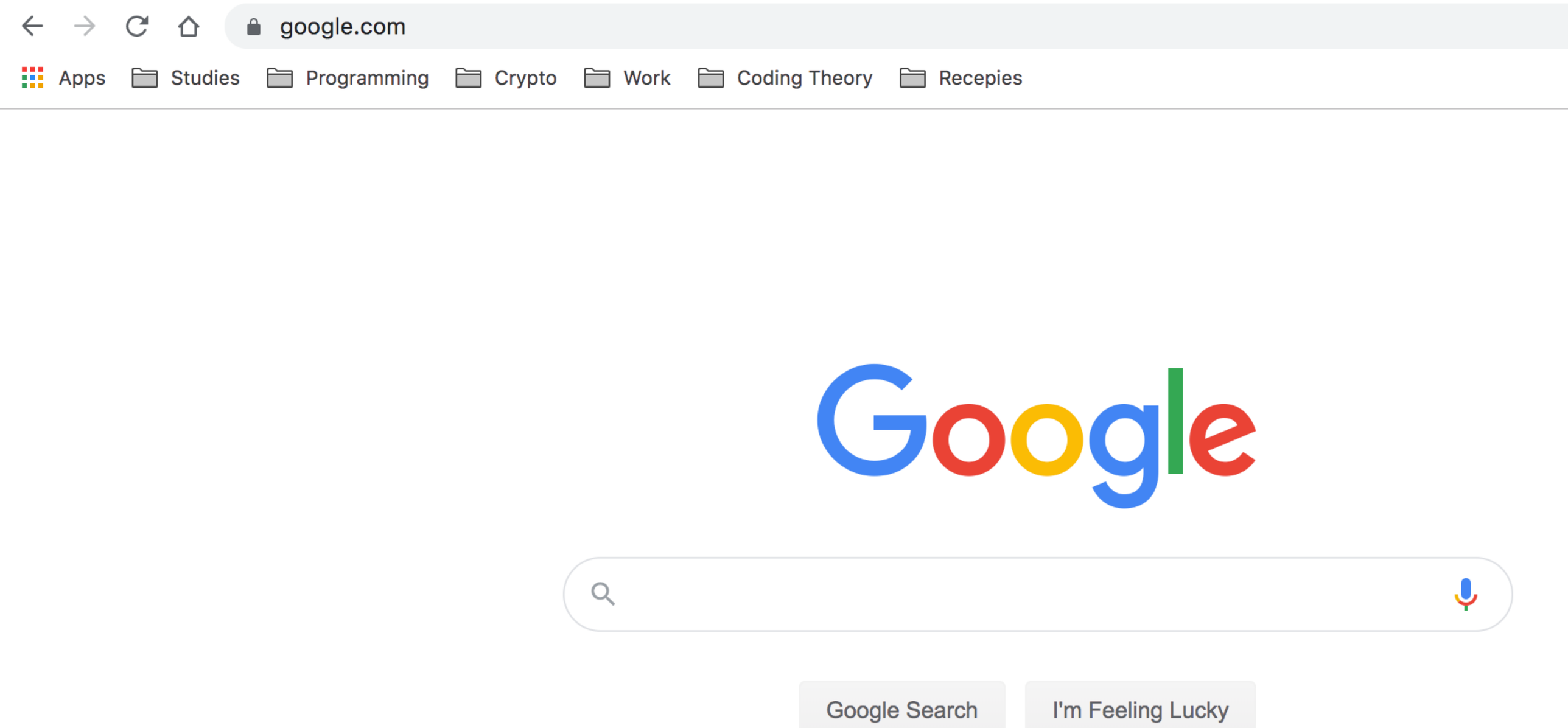




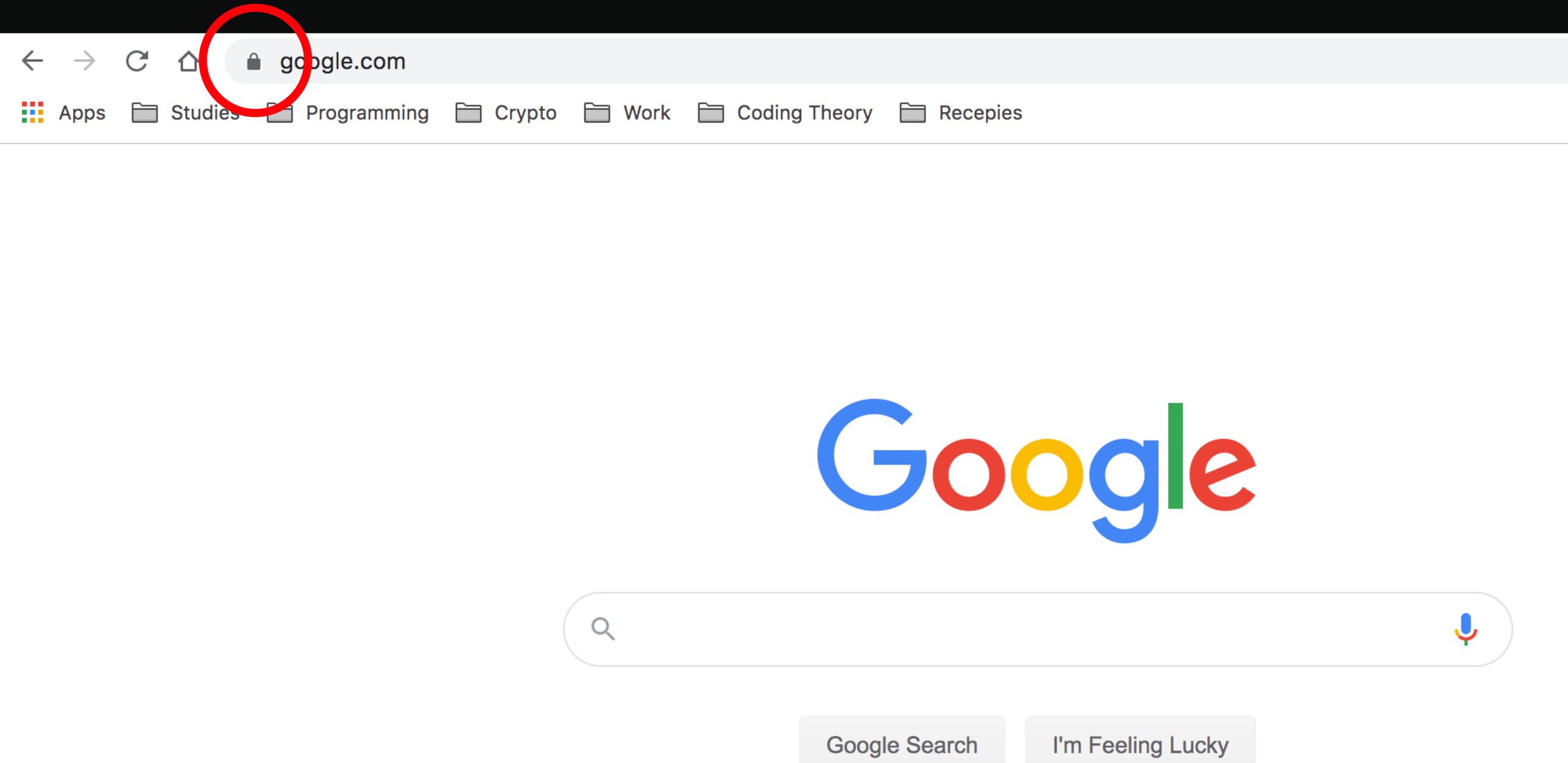



Математика
Крипто
Информатика

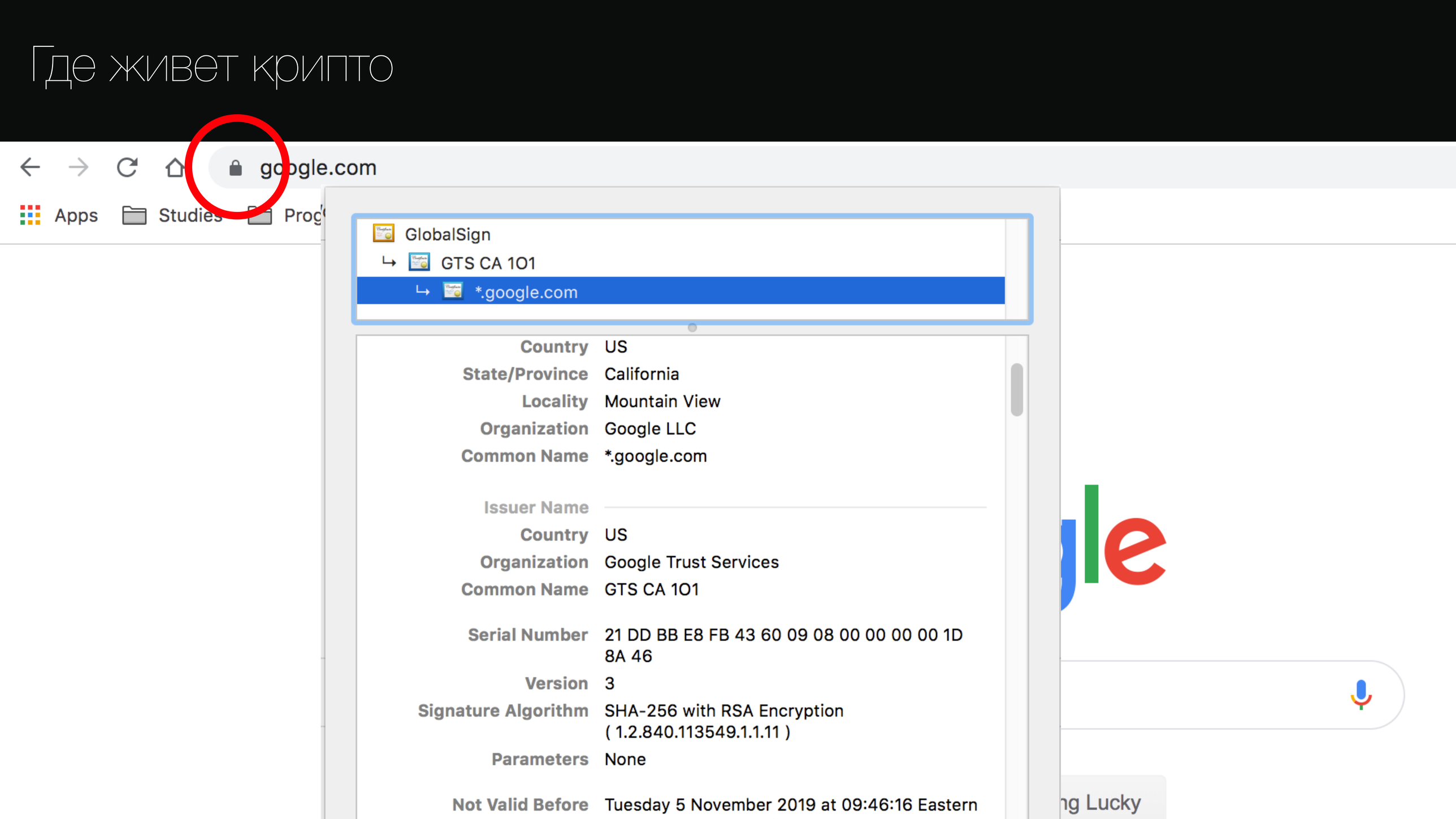
Где живет крипто



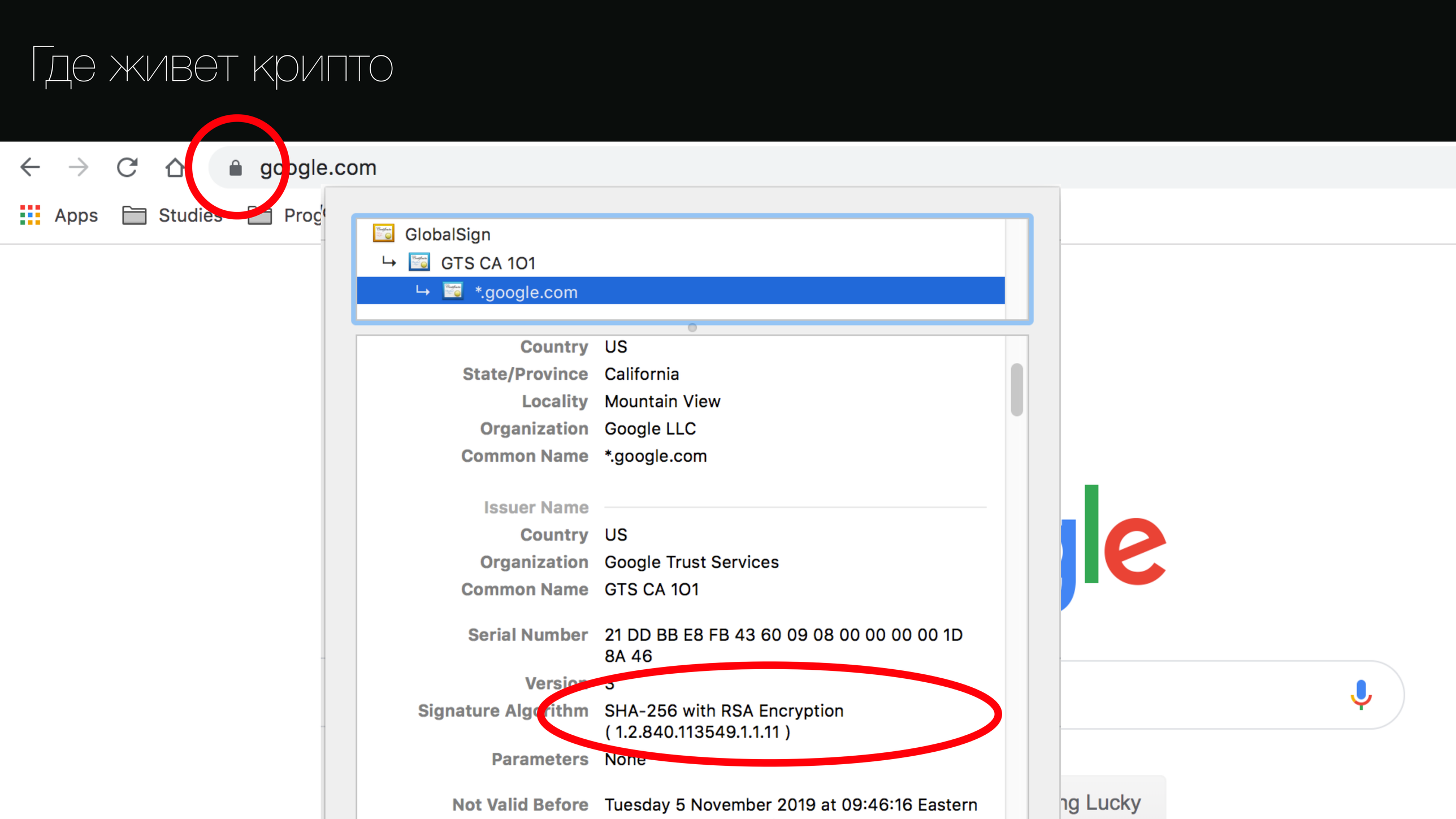
Где живет крипто



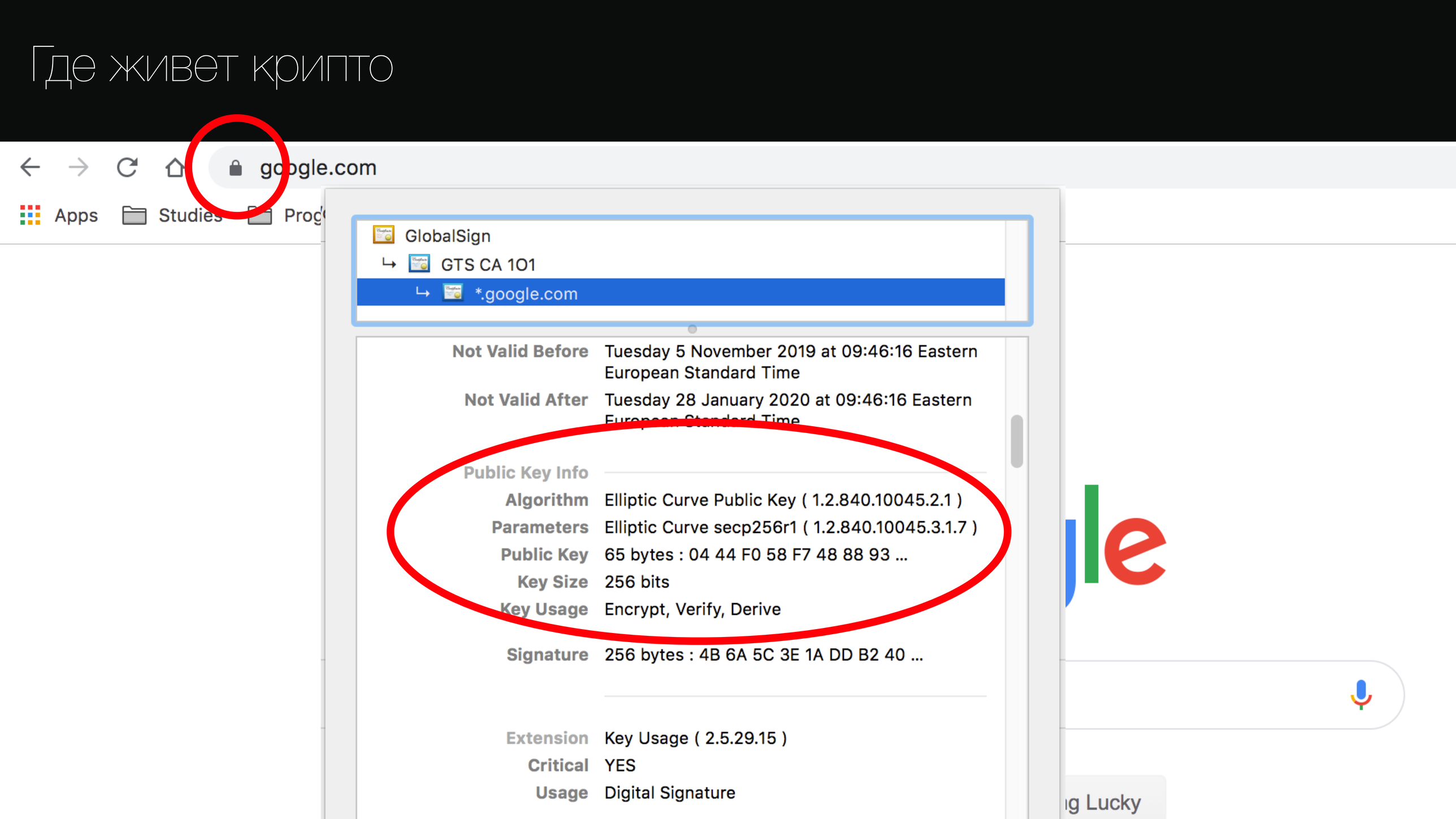
Где живет крипто



Где живет крипто



Где живет крипто



Симметрическая vs. ассиметрическая криптография

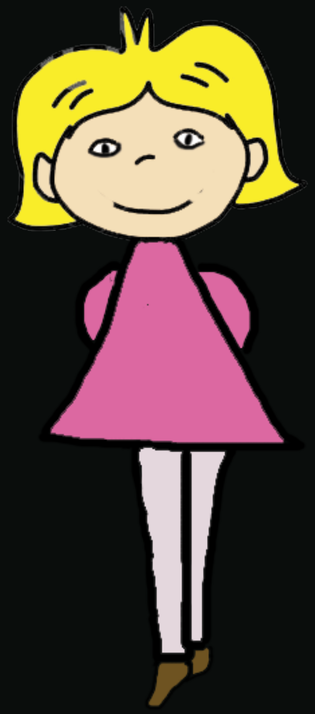
Симметрическая vs. ассиметрическая криптография

Цель: передать конфиденциальную информацию от *A* к *B*

Симметрическая vs. ассиметрическая криптография

Цель: передать конфиденциальную информацию от *A* к *B*

Алиса



key

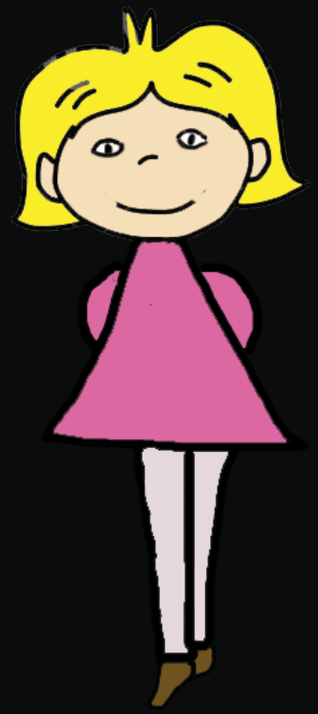
Боб

key

Симметрическая vs. ассиметрическая криптография

Цель: передать конфиденциальную информацию от *A* к *B*

Алиса



key

c = mes+key

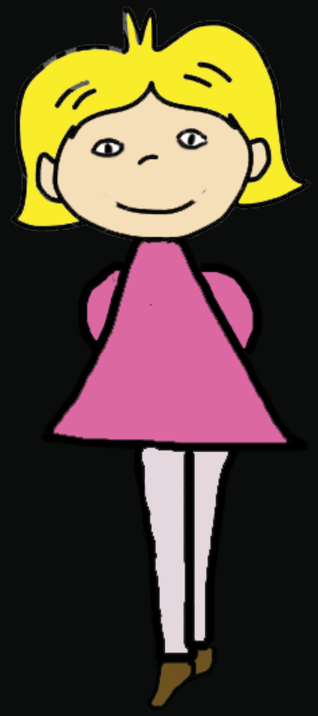
Боб

key

Симметрическая vs. ассиметрическая криптография

Цель: передать конфиденциальную информацию от *A* к *B*

Алиса



key

c = mes+key

Боб

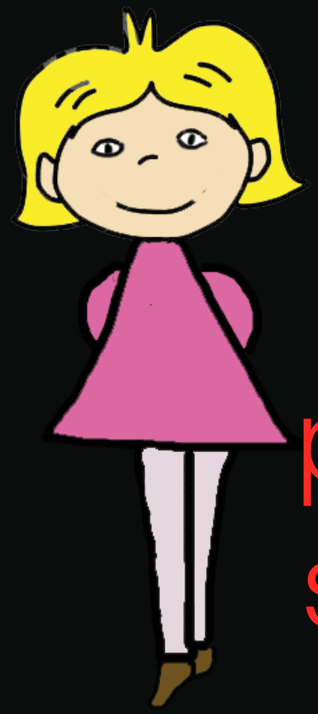
key

c — key = mes

Симметрическая vs. ассиметрическая криптография

Цель: передать конфиденциальную информацию от *A* к *B*

Алиса



pk_A
sk_A

Боб

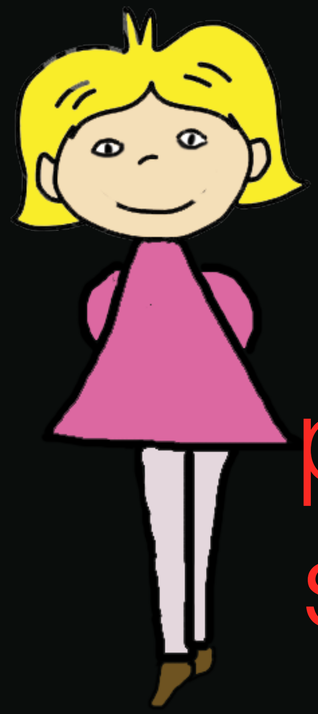
pk_B
sk_B

Симметрическая vs. ассиметрическая криптография

Цель: передать конфиденциальную информацию от *A* к *B*

Алиса

Боб



c= f(mes,pk_B)

pk_A
sk_A

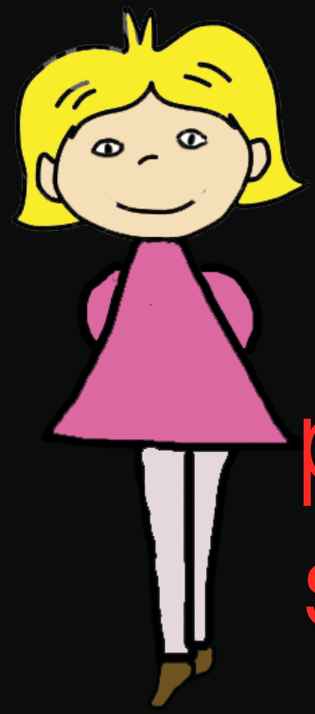
pk_B
sk_B

Симметрическая vs. ассиметрическая криптография

Цель: передать конфиденциальную информацию от *A* к *B*

Алиса

Боб



c= f(mes,pk_B)

pk_A
sk_A

pk_B
sk_B

mes=g(c, sk_B)

Криптографические алгоритмы

Хэш-функции

Цифровые подписи

Схемы идентификации

Генерации ключей

Шифрование

А ещё?

А ещё?

Multi-party computation

Электронное голосование

Машинное обучение на конфиденциальных данных

А ещё?

Проблема миллионеров

© Wikipedia

X $

X > Y ?

© Wikipedia

Y $

А ещё?

Проблема миллионеров

© Wikipedia

X $

X > Y ?

F(X)

F(Y)

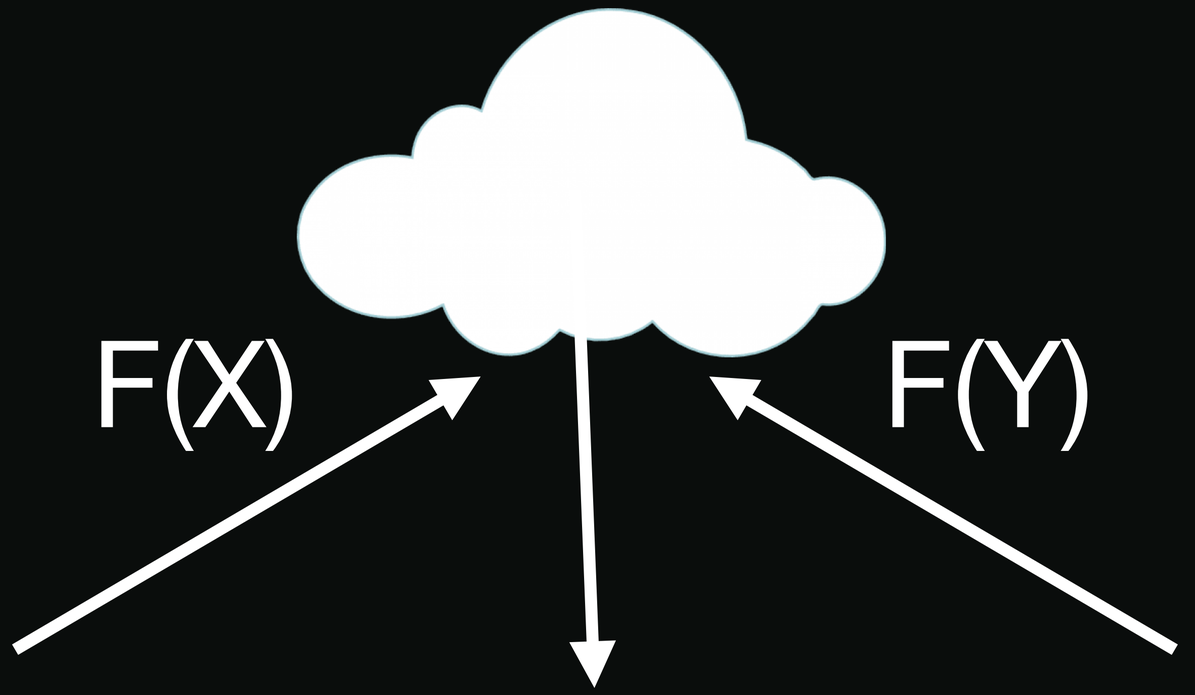
© Wikipedia

Y $

А ещё?

Проблема миллионеров

© Wikipedia



© Wikipedia

X $

X > Y ?

Y $

Зачем матчасть?

Электронное голосование

Зачем матчасть?

8 сентября: выборы в московскую Думу

Зачем матчасть?

17 июля: публикация кода электронного голосования

`https://github.com/moscow-technologies/blockchain-voting`

8 сентября: выборы в московскую Думу

Зачем матчасть?

17 июля: публикация кода электронного голосования

`https://github.com/moscow-technologies/blockchain-voting`

8 августа: атака П. Годри

8 сентября: выборы в московскую Думу

Зачем матчасть?

17 июля: публикация кода электронного голосования

`https://github.com/moscow-technologies/blockchain-voting`

8 августа: атака П. Годри

20 августа: обновление I

8 сентября: выборы в московскую Думу

Зачем матчасть?

17 июля: публикация кода электронного голосования

`https://github.com/moscow-technologies/blockchain-voting`

8 августа: атака П. Годри

20 августа: обновление I

24 августа: атака А. Головнева

8 сентября: выборы в московскую Думу

Зачем матчасть?

17 июля: публикация кода электронного голосования

`https://github.com/moscow-technologies/blockchain-voting`

8 августа: атака П. Годри

20 августа: обновление I

24 августа: атака А. Головнева

6 сентября: обновление II

8 сентября: выборы в московскую Думу

Суть атаки

Реализация на SOLIDITY — язык умных контрактов в Etherium

SOLIDITY_MAX_INT = 256 bits

Суть атаки

Реализация на SOLIDITY — язык умных контрактов в Etherium

SOLIDITY_MAX_INT = 256 bits

256 бит — ничтожно малый размер ключа для использованной криптосистемы

Почему тогда другие протоколы безопасны?

Почему тогда другие протоколы безопасны?

15 =

Почему тогда другие протоколы безопасны?

15 = 3 * 5

Почему тогда другие протоколы безопасны?

703 =

Почему тогда другие протоколы безопасны?

703 = 19 * 37

Почему тогда другие протоколы безопасны?

**12926802428524402920285950675467980784177641067886193612852138171009862055547156357278880564609165385475487184368759207797647823660196368438035260954579313248252350946920398436700079100155860842718423055353627027310716887457047902464735237735390468188232658340814522017155030356616426323443020959649572154264656012918736738539578088296256606766125374689446840169540534495671499385013633563619169036682173795656620846739400937839157079585322786294787745231105820161538888339675007456292422914718183191125834906835490972705799446047612909305525796133698962968303114626011171964657726153715324658450734624324595122787245 =

Почему тогда другие протоколы безопасны?

12926802428524402920285950675467980784177641067886193612852138171009862055547156357278880564609165385475487184368759207797647823660196368438035260954579313248252350946920398436700079100155860842718423055353627027310716887457047902464735237735390468188232658340814522017155030356616426323443020959649572154264656012918736738539578088296256606766125374689446840169540534495671499385013633563619169036682173795656620846739400937839157079585322786294787745231105820161538888339675007456292422914718183191125834906835490972705799446047612909305525796133698962968303114626011171964657726153715324658450734624324595122787245912345678 =
17976931348623159077293051907890247336179769789423065727343008115773267580550096313270847732240753602112011387987139335765878976881441662249284743063947412437776789342486548527630221960124609411945308295208500576883815068234246288147391311054082723716335051068458629823994724593847971630483535632962422413785912345678 *
71907725394492636309172207631560989344719079157692262909372032463093070322200385253083390928963014408448045551948557343063515907525766648997138972255789649751107157369946194110520887840498437647781233180834002307535260272936985152589565244216330894865340204273834519295978898375391886521934142531849689654940112345678

Почему тогда другие протоколы безопасны?

В основе криптографии — сложные задачи

Почему тогда другие протоколы безопасны?

В основе криптографии — сложные задачи

Например, задача факторизации больших чисел

Почему тогда другие протоколы безопасны?

В основе криптографии — сложные задачи

Например, задача факторизации больших чисел

Аргумент безопасности:
Взлом криптосистемы ⇒ решение сложной задачи

Понятие “сложности” зависит от модели вычислений

Понятие “сложности” зависит от модели вычислений

Задача факторизации — не “сложная” для квантовой модели вычислений

Факторизация vs. квантовый компьютер

Понятие “сложности” зависит от модели вычислений

Задача факторизации — не “сложная” для квантовой модели вычислений

Но…

Факторизация vs. квантовый компьютер

1. Криптографически значимого квантового компьютера пока нет

2. Даже если он появится, у нас есть “пост-квантовые” альтернативы

Классическая vs. квантовая vs. пост-квантовая криптография

RSA

DIFFIE-HELLMAN

ГОСТ 34.10-2012

Классическая vs. квантовая vs. пост-квантовая криптография

RSA

DIFFIE-HELLMAN

ГОСТ 34.10-2012

Квантовый канал связи
для получения генерации
квантового ключа

Классическая vs. квантовая vs. пост-квантовая криптография

RSA

DIFFIE-HELLMAN

ГОСТ 34.10-2012

Квантовый канал связи для получения генерации квантового ключа

Классические алгоритмы, стойкие к атакам на квантовом компьютере

Я придумал(а) **крутую** систему шифрования. Что делать?

Я придумал(а) **крутую** систему шифрования. Что делать?

1. **НЕ** патентовать

Я придумал(а) **крутую** систему шифрования. Что делать?

1. **НЕ** патентовать

2. Найти хорошего математика-криптографа

Я придумал(а) **крутую** систему шифрования. Что делать?

1. **НЕ** патентовать

2. Найти хорошего математика-криптографа

3. Найти грамотного инженера

Я придумал(а) **крутую** систему шифрования. Что делать?

1. **НЕ** патентовать

2. Найти хорошего математика-криптографа

3. Найти грамотного инженера-программиста

4. Обратиться в центры стандартизации (ITF, ISO, ГОСТ)