LIS-3353

Algorithms and Milkshakes

From this…

BLUE BELL
ICE CREAM
Peaches & Homemade Vanilla

GENUINE
CHOCOLATE FLAVOR
HERSHEY'S
SYRUP

...to this. (and back?)

Milkshake Model

Incidentally:

Source Code → Binary

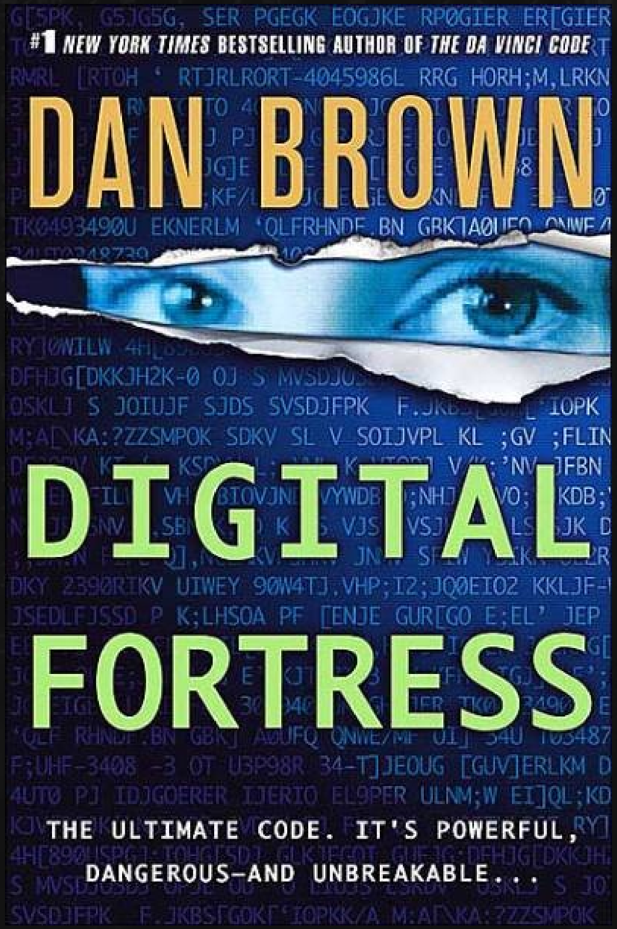
(human ish language goes in,
computer gibberish comes out)

Milkshake Model

DELIBERATELY:

Encryption and Hashing

Dan Brown is fun..but I threw this book across the room..


#1 NEW YORK TIMES BESTSELLING AUTHOR OF THE DA VINCI CODE
DAN BROWN
DIGITAL FORTRESS
THE ULTIMATE CODE. IT'S POWERFUL, DANGEROUS—AND UNBREAKABLE...

We (probably, if not certainly) still have unbreakable encryption.

yep. Even given all this NSA stuff.

But you have to be SUPER careful.

The purpose of today:

The **basic** tools for encryption are mostly available to all…

…even if the social factors, companies, corporations, and governments aren’t down.

(remember, you gotta have it on BOTH SIDES)

Encryption

Alice needs to send a verifiable message to Bob, but Carol is trying to listen in.

“This is a conversation between A and B so you can C your way out!”

Old school

Caesar Cipher. (yes, this really used to fool people)

DWWDFN WKH HDVW ZLQJ RI WKH IRUW DW WKUHH RQ WKXUVGDB..

attack the east wing of the fort at three on thursday

Also, hiding the message itself?

- Shave a guy's head, write the message on it.

….and wait.

Steganography

- Hiding the fact that the message (or payload) exists at all

  Examples:
  - fake personal ad to say something else
- - having a safe but hiding valuables in a shoebox
- - weird bits in a jpg

Steganography as (online) strategy?

Bad, because: Robots and radio

Better to be like “Yeah, you can have a copy. Too bad you STILL can't read it. HA!”

Other old school strategies

- Navajo Code Talkers
- Harriet Tubman “Wade in the water”

Languages, dialects, patois'…

CULTURAL ENCRYPTION.

CULTURAL AUTONOMY.

(ever heard of “code switching?”)

Another note on strategy

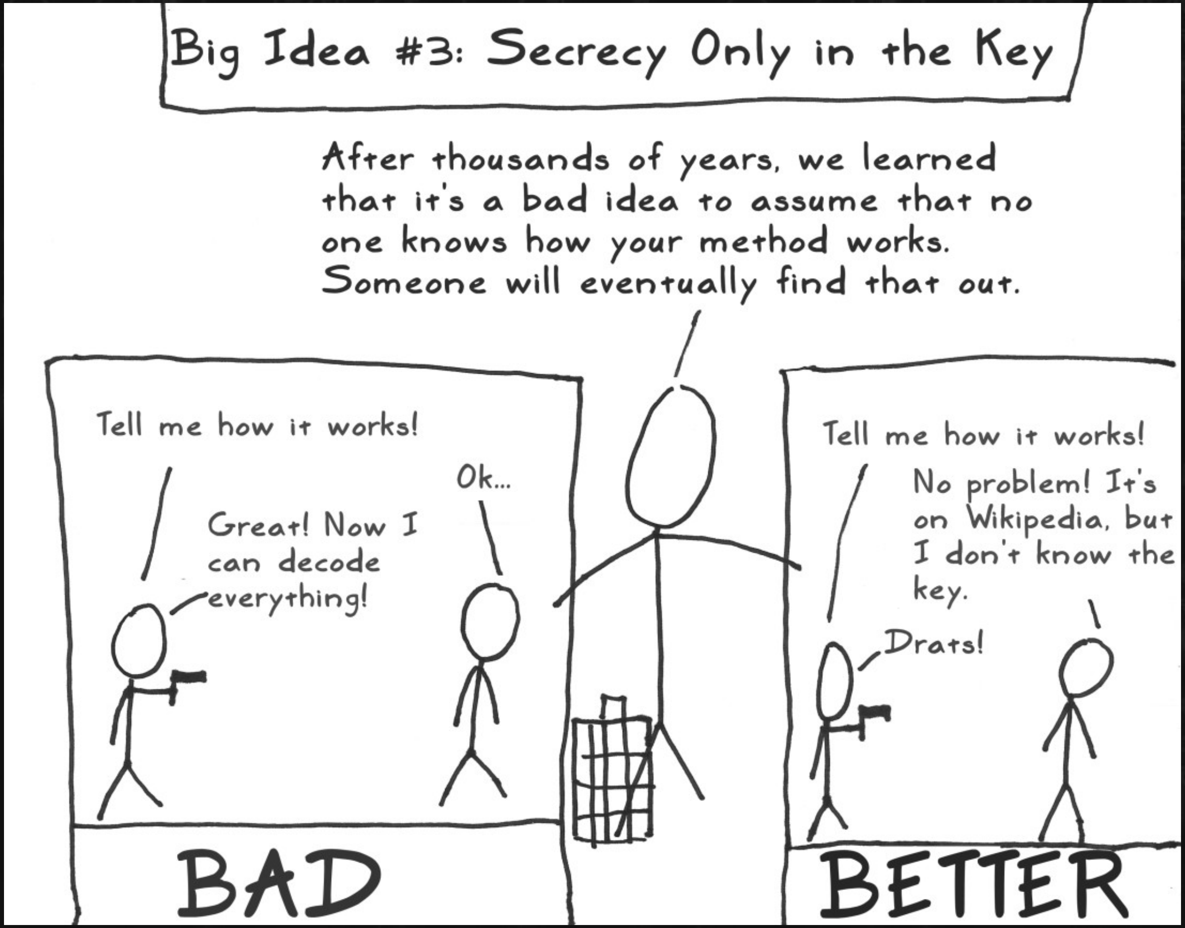
- Secret method vs.

  Public method (but secret key)

Why not interrogate the creators?

- We find Diffie and Hellman and MAKE THEM TELL US HOW IT WORKS!!
- (Free / Open Source v. closed?)

(score one more for open-source)


Big Idea #3: Secrecy Only in the Key
After thousands of years, we learned that it's a bad idea to assume that no one knows how your method works. Someone will eventually find that out.
Tell me how it works!
Ok...
Great! Now I can decode everything!
BAD
Tell me how it works!
No problem! It's on Wikipedia, but I don't know the key.
Drats!
BETTER

Newer strategies (if you can meet)

The bookstore strategy

OR

The One-Time Pad

but what if you CAN'T meet each other...

We need something “milkshake-y”

What, even for a computer, is VERY FAST in one direction..

..and IMPOSSIBLY SLOW in the other?

(remembering,
all computers do is math?

Hey, remember this?

```
    24
    / \
   4   6
  /\   /\
 2  2 2 3
```

What about 17?

PRIME!

Shortcuts to factoring?

(Let's hope not.....)

But, we need MATH, since online = numbers.

Hey, remember that prime number stuff? (warning, fake numbers below)

92348203942................................(random prime)

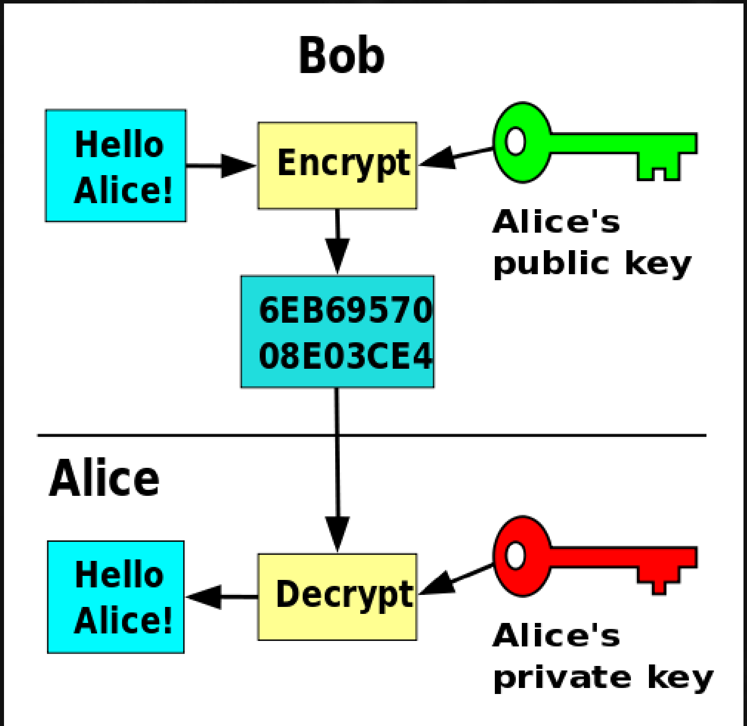
x 28059273729................................(random prime)

____

189808591765...(big ol' composite)

Public Key Encryption

- Usually, with a physical key, you just have one key, right? The key that locks your door also unlocks it....but what if you separate those two? One does the locking, the other unlocking?



On the
“key exchange?”

Look up Diffie-Hellman Key Exchange

It’s crazy-sounding, but it is possible to exchange the keys without a MITM attack:

(Think boxes locked inside other boxes)

Recap

- To encrypt:

  big composite number + clear message = coded text

  (public key)

  To decrypt

  two primes + coded text = clear message
  (private key)

Phil Zimmerman invents this and says “hey, this is pretty good privacy. I'll open source it...”

Eben Moglen calls him up a few hours later: “Cool idea, bro.”

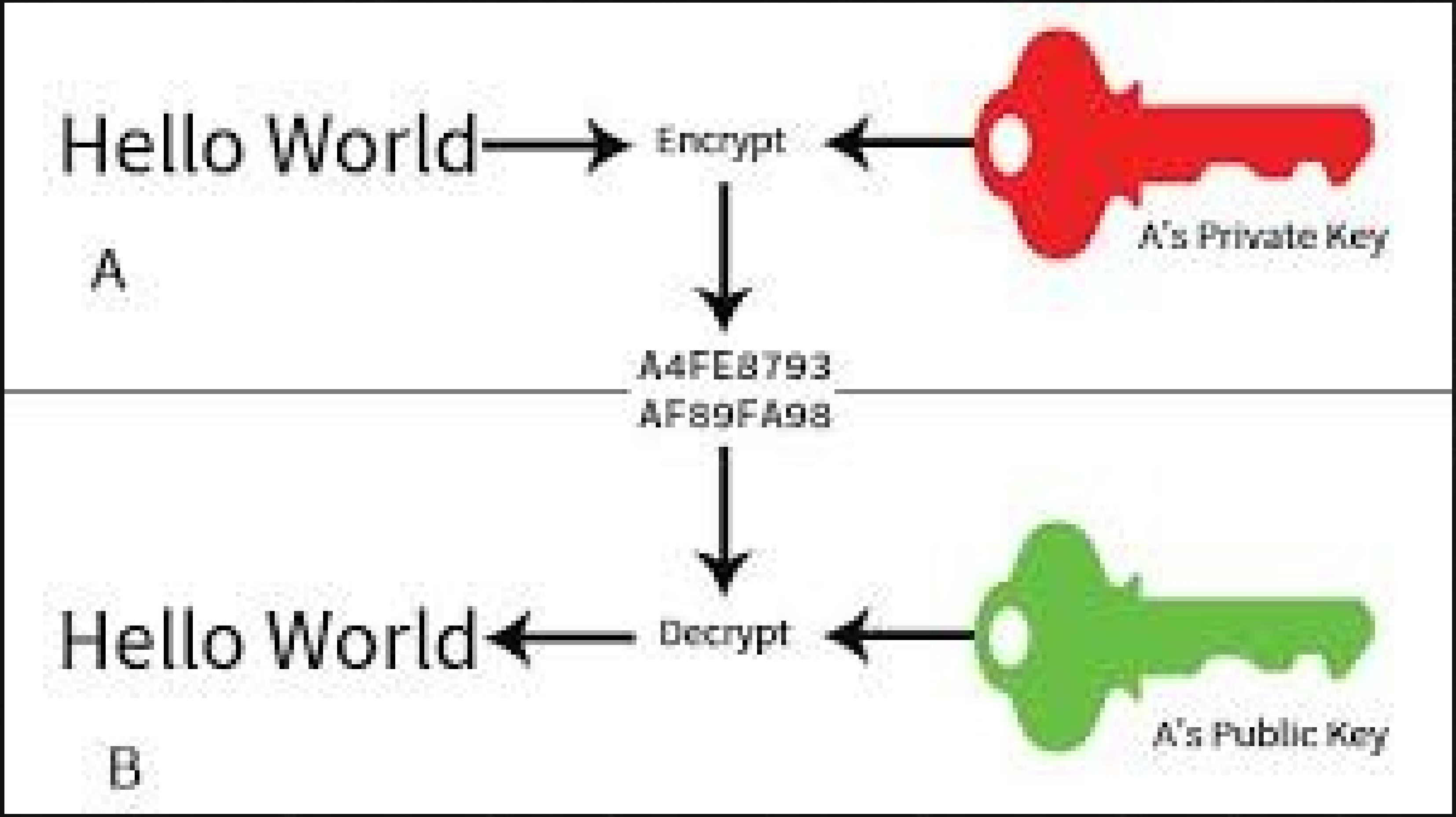
“also, in 5 hours when the FBI kicks down your door and throws you in jail, holler at me.”

-

Phil's a free man today. Why?

- ...he did something else at the same time, with the same technology.

Let's do something weird.
Let's do the opposite.



Who's more powerful than the government?

Even a government that is trying to stop terrorism?

Gotta sign those checks and credit cards…

- Encryption and digital signatures are two sides of the same coin.

  You need digital signatures to send money, so we also have encryption. (mostly)

But…the NSA?

…will discuss further :)

Other Milkshakey Topics

- File/Message Verification
- Smart Password Storage
- Cryptocurrencies (e.g. Bitcoin)

But…the NSA?

- All known NSA attacks/backdoors aren't against the actual method itself…

  …but its implementation.

Random Numbers:

We need primes, but obviously we can't do them in order.

We need RANDOM NUMBERS.

but, can computers do random numbers?

Sure, you could bang on a calculator, but...

Good RNG's..

Pull from a number of different places..

- typing

- cameras

..etc.

..bad ones, aren't really random.

“Random number generator? Sure, why not try OURS? Wink wink.

So, what's using this now? A lot, with different levels of “reliability”

- SSL
- Bitlocker
- Whatsapp
- Signal

But the most interesting is:

Truecrypt!

- You can use it now.

- Makes you wiggle your mouse. Why?

- The FBI tried/tries to force people to give up their passwords. This is an interesting legal question. Is a password..

  - a "thing" like a key (then it can be forced)

  – Or "words?" (plead the ____th amendment)

  (also, what does this tell us about truecrypt)

But then, suddenly...

“WARNING: Using TrueCrypt is not secure as it may contain unfixed security issues”

“So use this different Windows one. Don't use the linux one at all. “

-hmmmm

LIS-3353

Other Milkshakey Topics…

Truecrypt (and similar)

- You can use it now.

- Makes you wiggle your mouse. Why?

- The FBI tried/tries to force people to give up their passwords. This is an interesting legal question. Is a password..

  - a “thing” like a key (then it can be forced)
  - Or “words?” (plead the ____th amendment)

  (also, what does this tell us about truecrypt)

But then, suddenly…

“WARNING: Using TrueCrypt is not secure as it may contain unfixed security issues”

“So use this different Windows one. Don't use the linux one at all. “

-hmmmm

Also, passwords

- When you lose your password, what does the website do?

  Make you change it?

  Or

  Send you a copy of it.

If they're nice enough to send you your password, you can rest assured..

If they're nice enough to send you your password, you can rest assured..

that they suck at security:

Passwords

- If they send you a copy...they suck at security.

  Good websites CANT send you your password.

- Because they don't actually “know”* it.

Obviously, if you're storing passwords, you want them "encrypted"

One way to do this:

1) Get their password

2) Save it on your computer

3) Then, encrypt it for safety.

...but wait

What do we REALLY need?

The actual password

or

simply:

Proof that they typed in the same thing both times?

What do we REALLY need?

Remember: “encrypting” something always yields:

GIBBERISH

“MyPassword123” ›
ab18db351a3ed3849cca9839d98381ee63
92eeb391baa39d766290082812d9eceab

What do we REALLY need?

Remember: “encrypting” something
always yields:

UNIQUE GIBBERISH

“MyPassword123” >
ab18db351a3ed3849cca9839d98381ee6392eeb391baa39d766
290082812d9eceab
means that
”DifferentPassword456” ≠
ab18db351a3ed3849cca9839d98381ee6392eeb391baa39d766
290082812d9eceab

So, let’s just switch it:

ANOTHER way to do this:

1) Get their password

2) ENCRYPT IT FIRST for safety.
3) Then save the ENCRYPTED password.
→
next time they log in?

So, let’s just switch it:

next time they log in?

1) Get password
2) Encrypt it the same way, then compare the gibberish!

~~“MyPassword123” = “MyPassword123”~~

but instead..

So, let’s just switch it:

“ab18db351a3ed3849cca9839d98381ee639
2eeb391baa39d766290082812d9eceab”

=

“ab18db351a3ed3849cca9839d98381ee639
2eeb391baa39d766290082812d9eceab”?

And “MyPassword123” IS NOT ON THE SERVER

But wait: Let’s do one more thing? Do we really need all of this?

“ab18db351a3ed3849cca9839d98381ee639 2eeb391baa39d766290082812d9eceab”

=

“ab18db351a3ed3849cca9839d98381ee639 2eeb391baa39d766290082812d9eceab”?

And “MyPassword123” IS NOT ON THE SERVER

But wait: Let’s do one more thing?
Do we really need all of this?

“ab18db351a3”= “ab18db351a3”*

As long as

- we use ALL the data in the original to get this number
- And it’s STILL mathematically unliklely that two different passwords will yield the same short gibberish, we’re good to go.

*you don’t quite just cut a chunk off, but more on that in a bit… i

But wait: Let's do one more thing? Do we really need all of this?

"ab18db351a3"= "ab18db351a3"*

ADVANTAGES:

- It's shorter
- Now you LITERALLY CANNOT "decrypt" it because you're missing some information. This is good!
- And, now – we can use this verification method on things other than passwords as well.

To Illustrate, first something slightly dumber…

File Verification

Presumption: The network (or person) is imperfect. The bytes we receive may not always be the exact ones that were sent.

Also: The network or verification is “slow”

We need a shorter, but verifiable, version of the data.

Basic Digest/Checksum

The grocery list:

Cheese

- Crackers
- Eggs
- Ham
- Koala
- Mangoes
- Salt
- Underwear

Send the following…

- Cheese
- Ham
- Eggs
- Crackers
- Koala
- Salt
- Underwear
- Mangoes
- CHECKSUM.45

CHECKSUM.45 = CHECKSUM.45

If the reciever gets

- Cheese
- Ham
- Eggs
- Crackers
- Salt
- Underwear
- Mangoes
- CHECKSUM.45

CHECSUM.40 = CHECKSUM.45? *NO, SEND AGAIN.*

Hashing

Error checking/Checksumming.

One tiny change in the original still means BIG changes in the gibberish.

(MD5, which is fast, but not super-secure) is good for this)

Hashing

“Used to map data of arbitrary (big) size,

to data of fixed (small) size.”

Verification:

Hashing Uses

- Error Checking/Checksums
- Password “Storage”
- Bitcoin/Cryptocurrencies

Passwords

They don't store your password (your secret ingredient)

They just store the entire milkshake....and calculate/mix it every time.

(don't use MD5, use something deliberately slow, like bcrypt)

To abuse some more analogies..

Consider your mom’s ________ recipe?
(milkshake?)

Even if you don’t know the ingredients..
… you know when it’s WRONG :)

Horrible – storing the password

Better but still bad – storing the password hashed

Decent – storing “userid+password” hashed

Best – storing “userid+password+salt” hashed

Login: jmarks

password: g00dpassword

(salt): b00gab00ga

jmarks+g00dpassword+b00gab00ga

==HASHED==>

02f39aae85ad73e162b446e918597e89

Hey, so these hashes

They look like--

- 02f39aae85ad73e162b446e9

What are the odds that it would look like, say..

- 00000ae85ad73e162b446e9

Not IMPOSSIBLE, just VERY UNLIKELY.

A bit on banks and money

What is most money "made of?" How is it stored? Coins and little green pieces of paper?

A bit on banks and money

What is most money “made of?” How is it stored? Coins and little green pieces of paper?

NOPE.

Just (trusted) lists. Ledgers in banks and such.

Usually “digital”

In fact, lists are older than “money” itself.

Not dollars, but a list somewhere that says

“Ug owes Oof two cows”

“Oof owes Grok a stick”

or more accurately

“everybody owes the king taxes”

theorem: any system that involves writing down “ownership” and “what you've paid” for is (possibly) a bank = (gamestop, even)

But, you also might want "pieces/tokens"

GOLD AND DIAMONDS HA HA HA

(wait, seriously. Why are diamonds more expensive than water?)

Bitcoin and most other “Cryptocurrencies”

(note, some others don’t work exactly this way)

A huge encoded/distributed online ledger/list, also called a “blockchain”

Powered/driven by “mining” (which is more like a slot machine, pull the lever, power the thing, and see if you “win”)

Mining Bitcoin?

- Randomly trying to find “nice looking” hashes.

......4E9BB99 nope.

.......OOOOOO yep! $$$$

Mining

When you download a bitcoin (node) wallet program, you literally have to get a copy of every single transaction ever.

Transactions are computationally expensive.

The “Mining” also powers the “hashed transactions..eg.”

02b23 gave bf239 .005 bitcoins. I can prove it because the hash of this transaction is =>

081ee23

Add this to the chain and spread it around.

Bitcoin transaction.

- You “add your new or old hash movement” to the ledger. By making another special hash. Which is “expensive.”

  This work powers the blockchain and “proves” that you’ve put in work. By design, the system “rewards” you for it.

Bitcoin transaction.

Now, why is it valuable?

Because people believe it is.

See also: gold, beanie babies, crappy companies, virtual swords, pokemon cards, sneakers, whatevs.

Bitcoin transaction.

PS: Crypto addresses? Just numbers in different bases (sometimes with a prefix): Functionally similar to public keys

BTC: 18L1qxHaN1i8ihrLMX8sxrQLmfoaTWud9R

ETH: 0xD54b6C55A8aEc0bec04Cb6b3eB6F84F6BcF03619

(these are 100% real, please feel free to send me money :))

“Blockchain”

It’s a large public database, in which everyone can see every transaction. That’s all.

Now give me a bunch of money because I just said “blockchain” :)