A good security architecture can kill online fraud

Objective

...to show you how to avoid single-point-of-failures (SPOFs) and prevent fraud

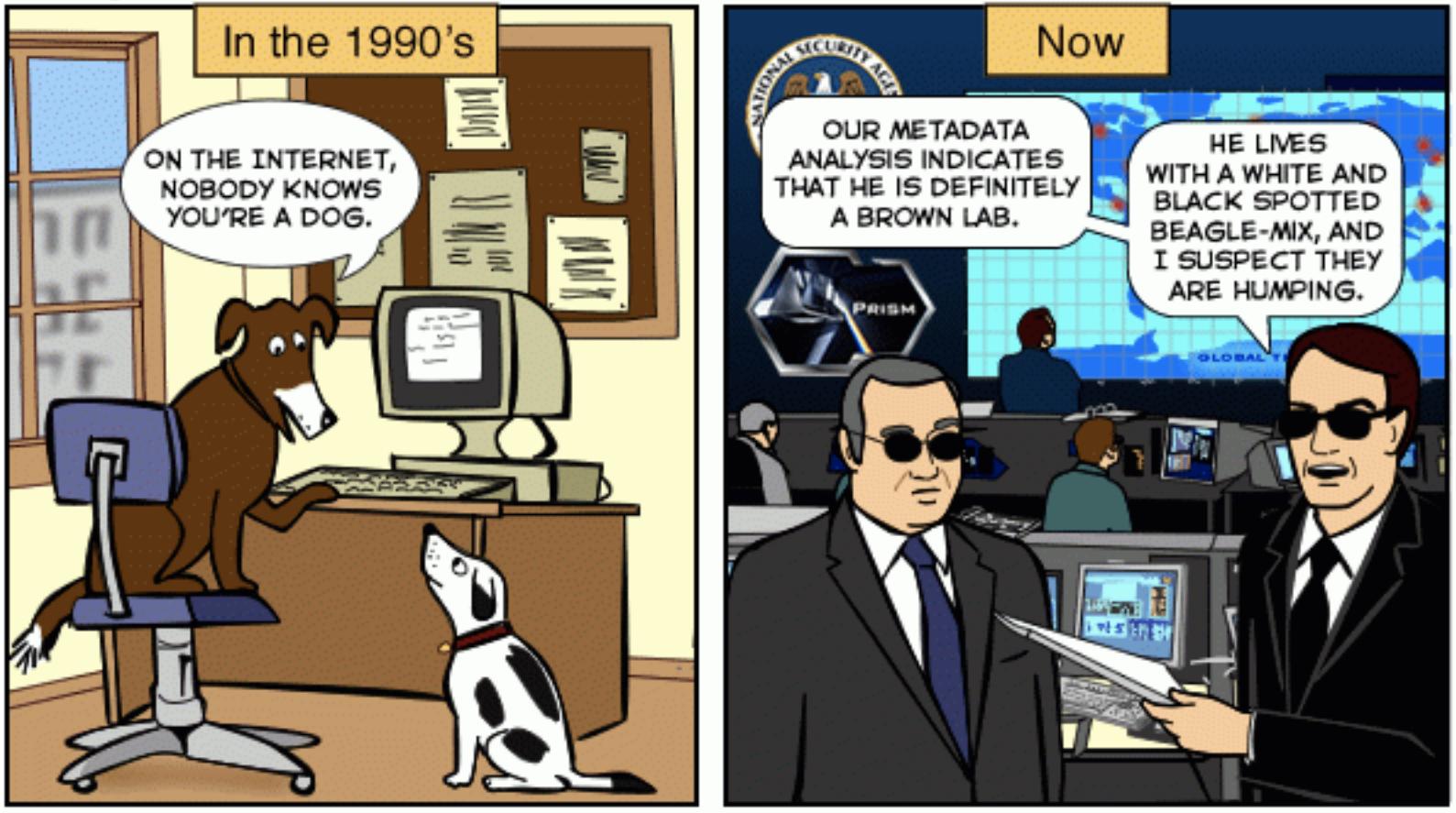
What is a single-point-of-failure?

One place, which compromises the whole security if hacked.

What's behind most fraud online?

..... a fundamentally broken nature of identity online...

US government approach to this problem: sneaky surveillance



Chinese government approach to this problem: Real ID online

What's needs to be done to kill fraud?

1. ***Identity: Our learnings from fixing the online identity***

2. ***Communication: Applying the principles to security of TLS***

Why should you even listen to me?

ADUCID (The world's most advanced authentication system)

Our background:

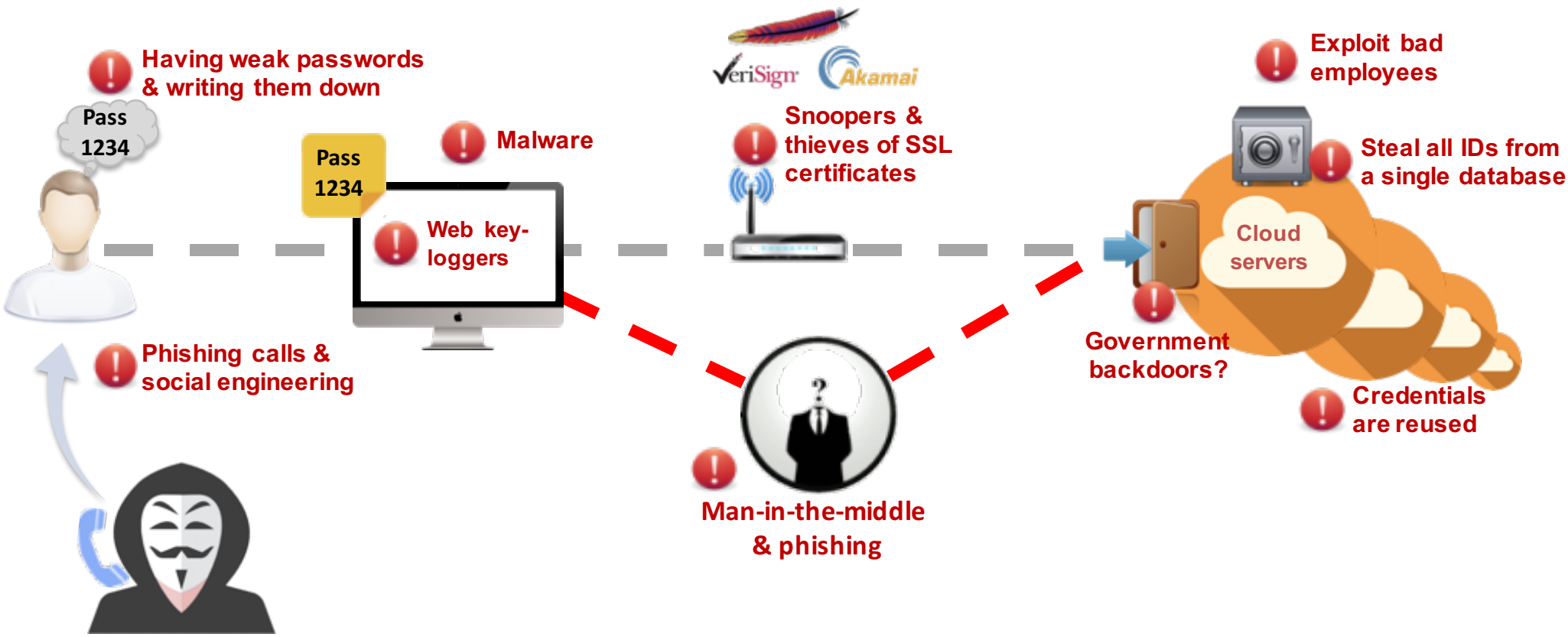
- > 20 years of experience in IT security
- We helped shape the standard for digital identities in the European Union

Our products:

- 10 years of unique research in authentication & secure communication
- 7+ million USD in direct investments
- 7+ international patents
- several rounds of penetration tests without a single relevant weakness

PASSWORDS HAVE SOOO MANY PROBLEMS...

- Reliant on users
- Copying identities
- Centralization
- Static protection
- Physical ID online
- Lack of mutual trust
- Credentials reuse
- Reliant on HTTPS



EACH OF THEM REPRESENTS A SINGLE-POINT-OF-FAILURES

HOW TO SOLVE THESE PROBLEMS?

- Reliant on users
- Physical ID online
- Copying identities
- Lack of mutual trust
- Centralization
- Credentials reuse
- Static protection
- Reliant on HTTPS

2 key design principles & 6 main tools:

1. Remove the people from security

2. Fix the architecture instead of "plugging holes"
   - Separate cyber IDs
   - Asymmetric cryptography
   - Mutual trust
   - Distributed design
   - Dynamic protections
   - Dual authentication

bitcoin

BLOCKCHAIN

REMOVE USER

How much of security do users handle?

1. Create strong passwords
2. Never reuse passwords
3. Change password frequently
4. Make sure communication is secure
5. Recognize fake websites & phishing
6. Recognize fake calls or fraudsters
7. Make sure all apps, systems and devices are up-to-date
8. Fight against viruses
9. NEVER EVER make a mistake

Can any person do all that?

Who should be doing that?

Physical world

REMOVE USER

Online world

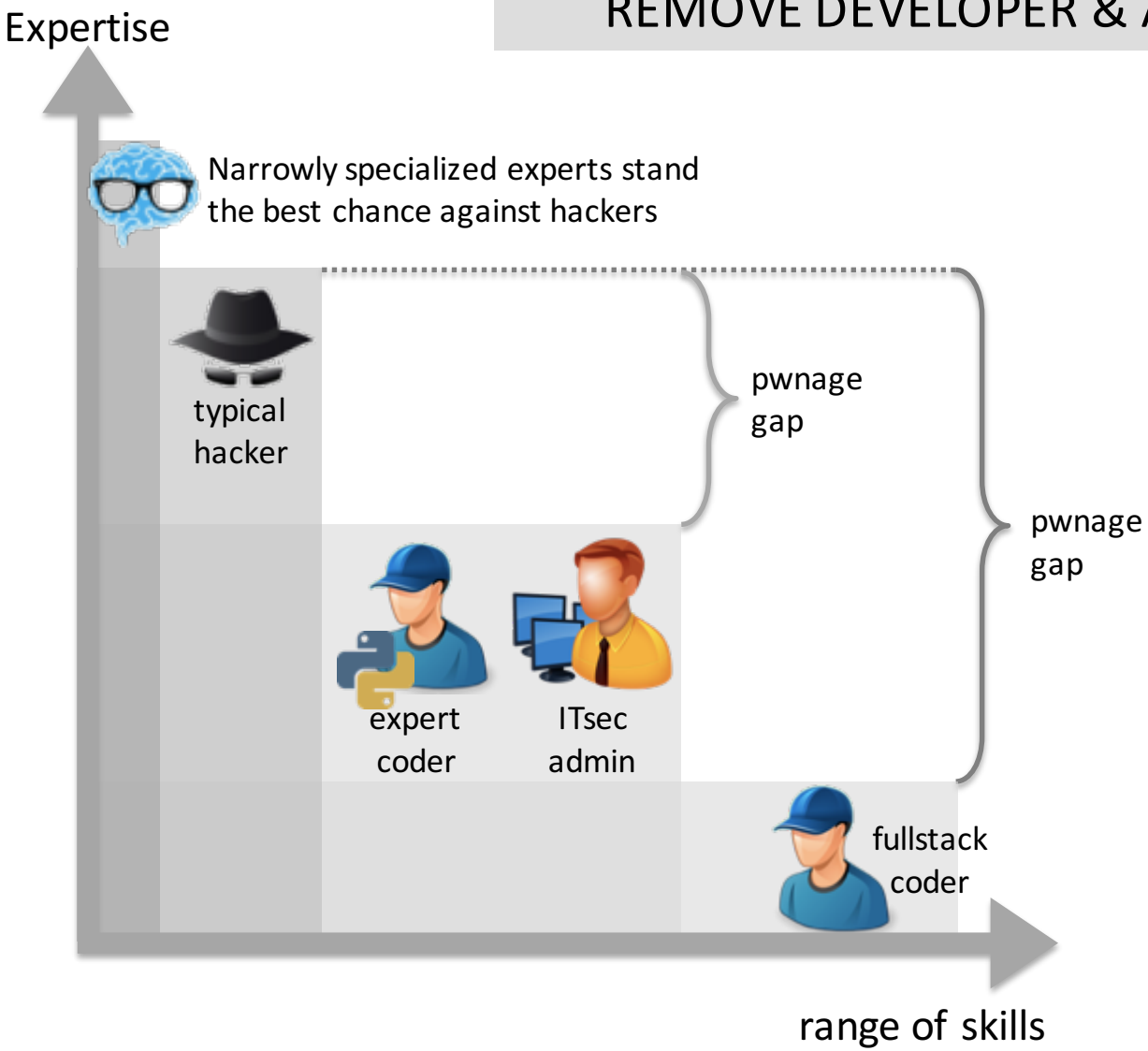
The app handles reliably all the security actions for the users

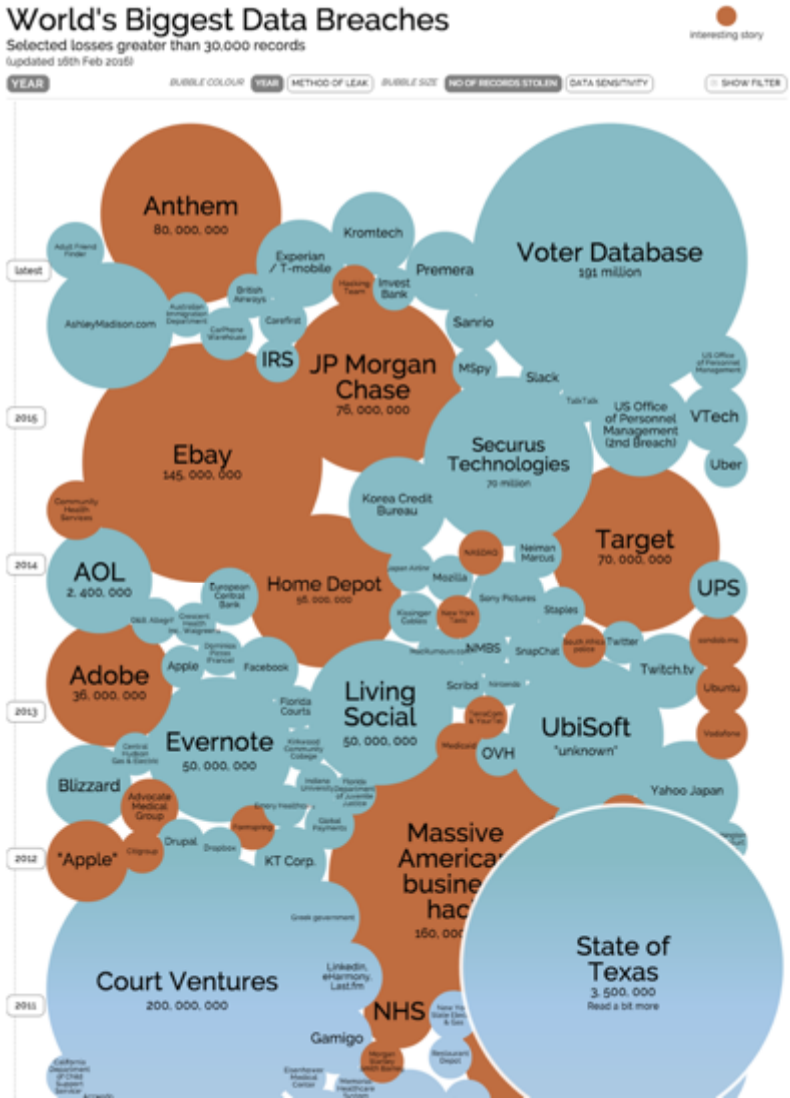
REMOVE DEVELOPER & ADMIN

Developers and admins must never make a mistake in...

... and they have to face a huge number of attack vectors

REMOVE DEVELOPER & ADMIN





Physical world

REMOVE DEVELOPER & ADMIN

Online world

I don't handle security

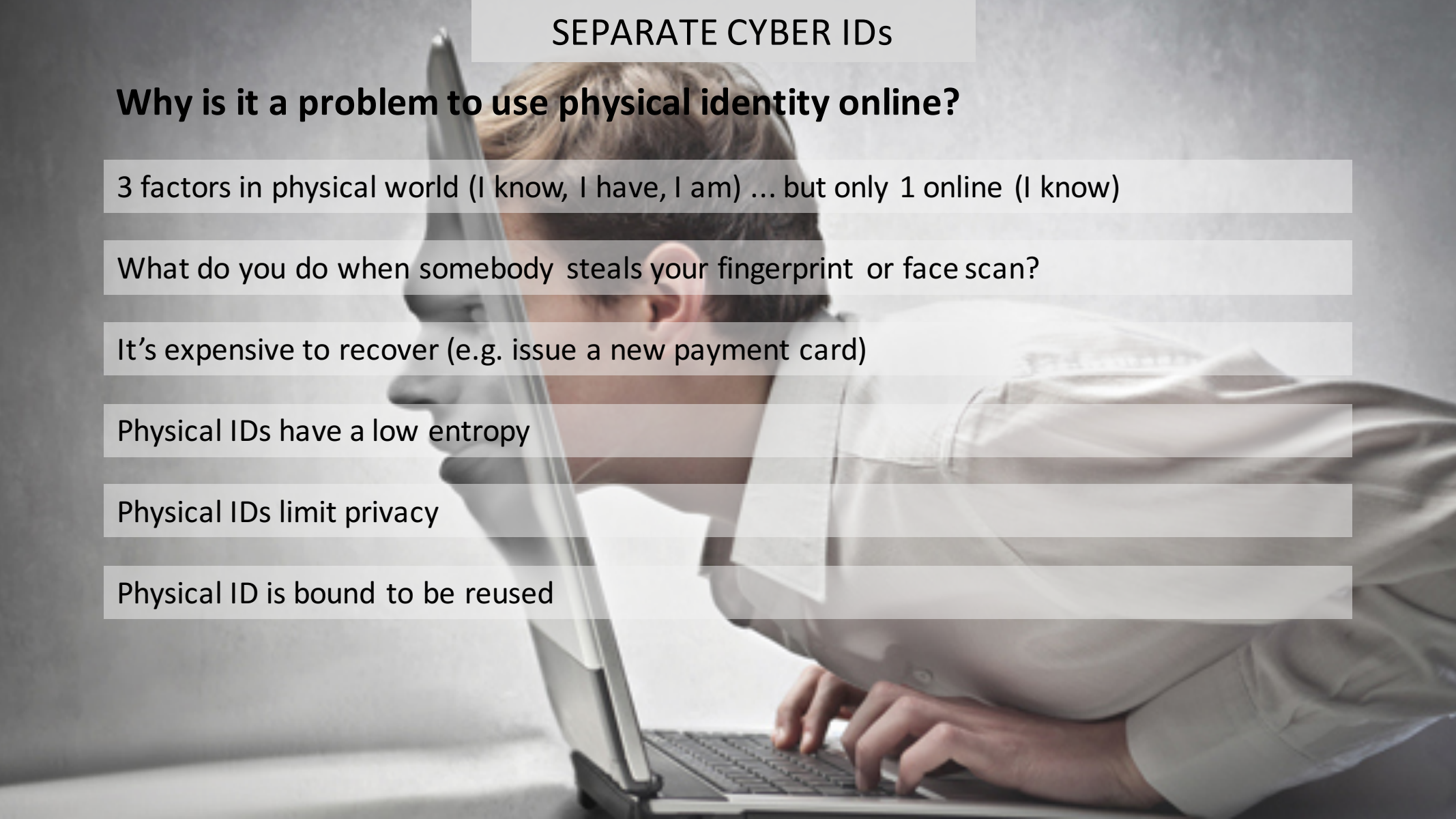
... I just have my device

My app handles security

Security as a service

Security should be developed by experts & work as a service

SEPARATE CYBER IDs



Why is it a problem to use physical identity online?

- 3 factors in physical world (I know, I have, I am) ... but only 1 online (I know)
- What do you do when somebody steals your fingerprint or face scan?
- It's expensive to recover (e.g. issue a new payment card)
- Physical IDs have a low entropy
- Physical IDs limit privacy
- Physical ID is bound to be reused

PHYSICAL & CYBER IDs

Physical world

Online world

I have my device

Physical ID: Ivo Toman

My app handles security

Cyber ID:
6F_j7+hl$ps&6aF3^9f;k?fs
dK97jkJ975u....R.;'u&5dh8
K3l;9asd-k)*2@'',96&(8o3

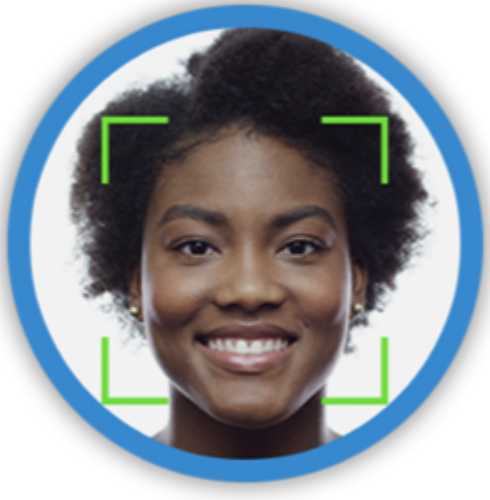
Your physical ID cannot be misused online.

Your cyber ID cannot be misused in physical world.

Would you give the credit card and walk away?


NEW YORK STATE
ENHANCED
DRIVER LICENSE
ID: 012 345 678
CLASS D
DOCUMENT
SAMPLE, LICENSE
2345 ANYPLACE AVE
ANYTOWN NY 12345
DOB: 06-09-85
Would you give the driving license and drive away?

... but that's exactly what you do online today when you enter your password, face-scan or payment card info....



After how many days from a security breach do Asian companies discover that they have been hacked?

A typical Western company discovers a breach after ~200 days

A typical Asia company discovers a breach after ~500 days

ASYMMETRIC IDs

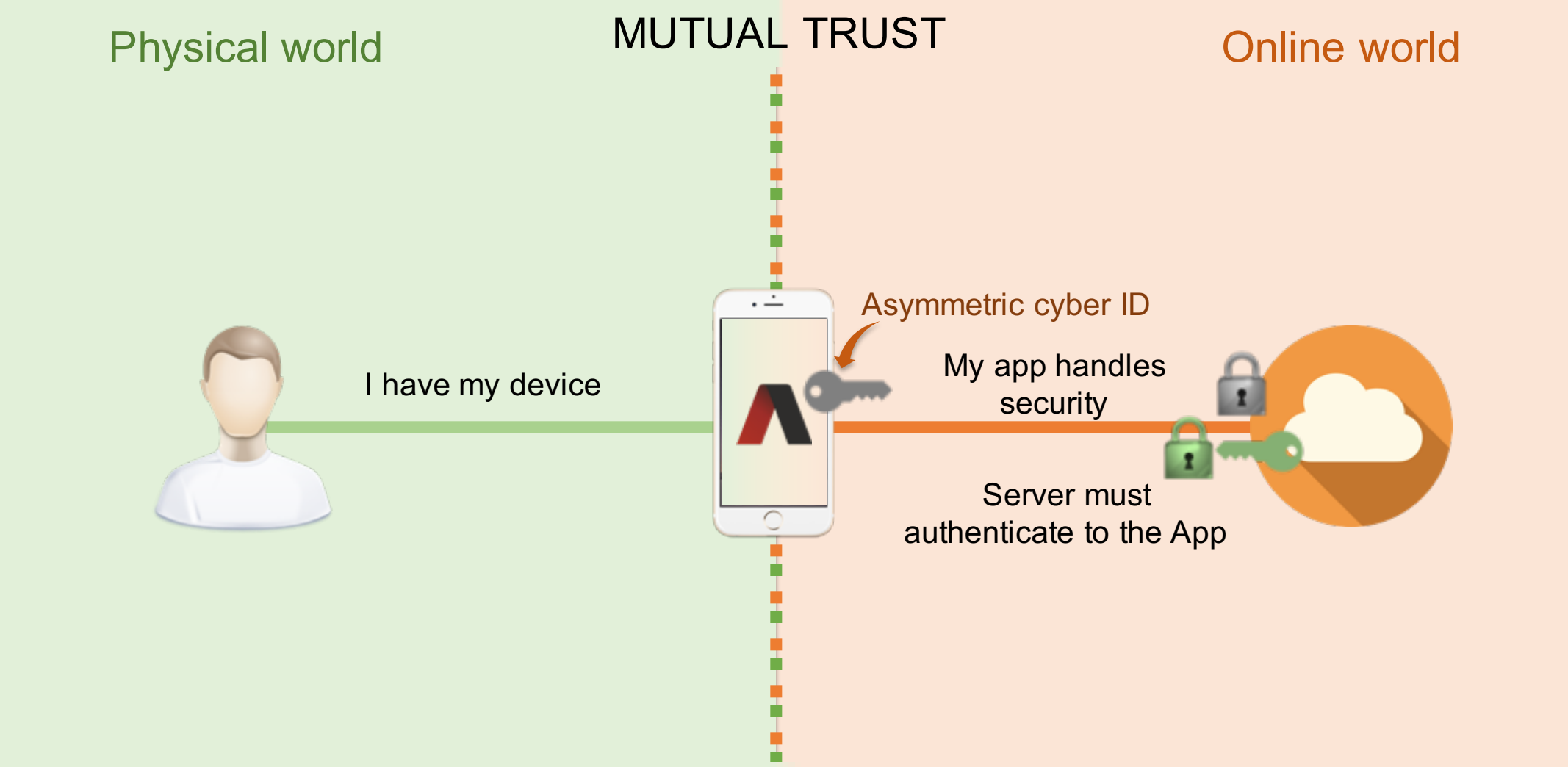
Physical world

Online world

Cyber IDs are asymmetric and never leave user's hands.

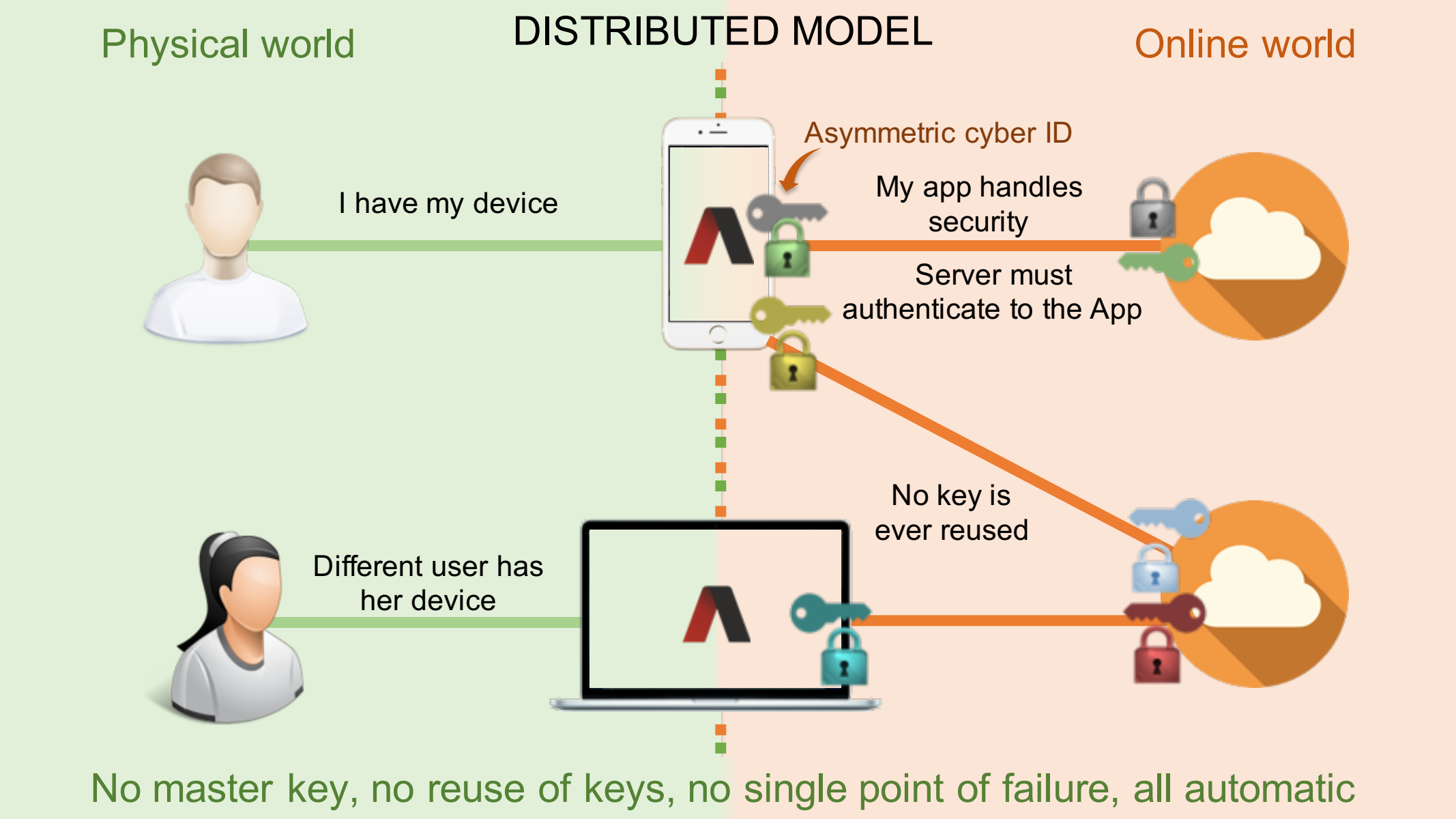
No hack on server or communication can compromise user's cyber ID


Is this service who it claims to be?
Phishing
Man-in-the-middle
Theft of user's identity
Is this user who she claims to be?
FRAUD



No user or App proves its identity until the server proves its identity the user/App.
AND! This is HTTPS independent and largely quantum resistant.

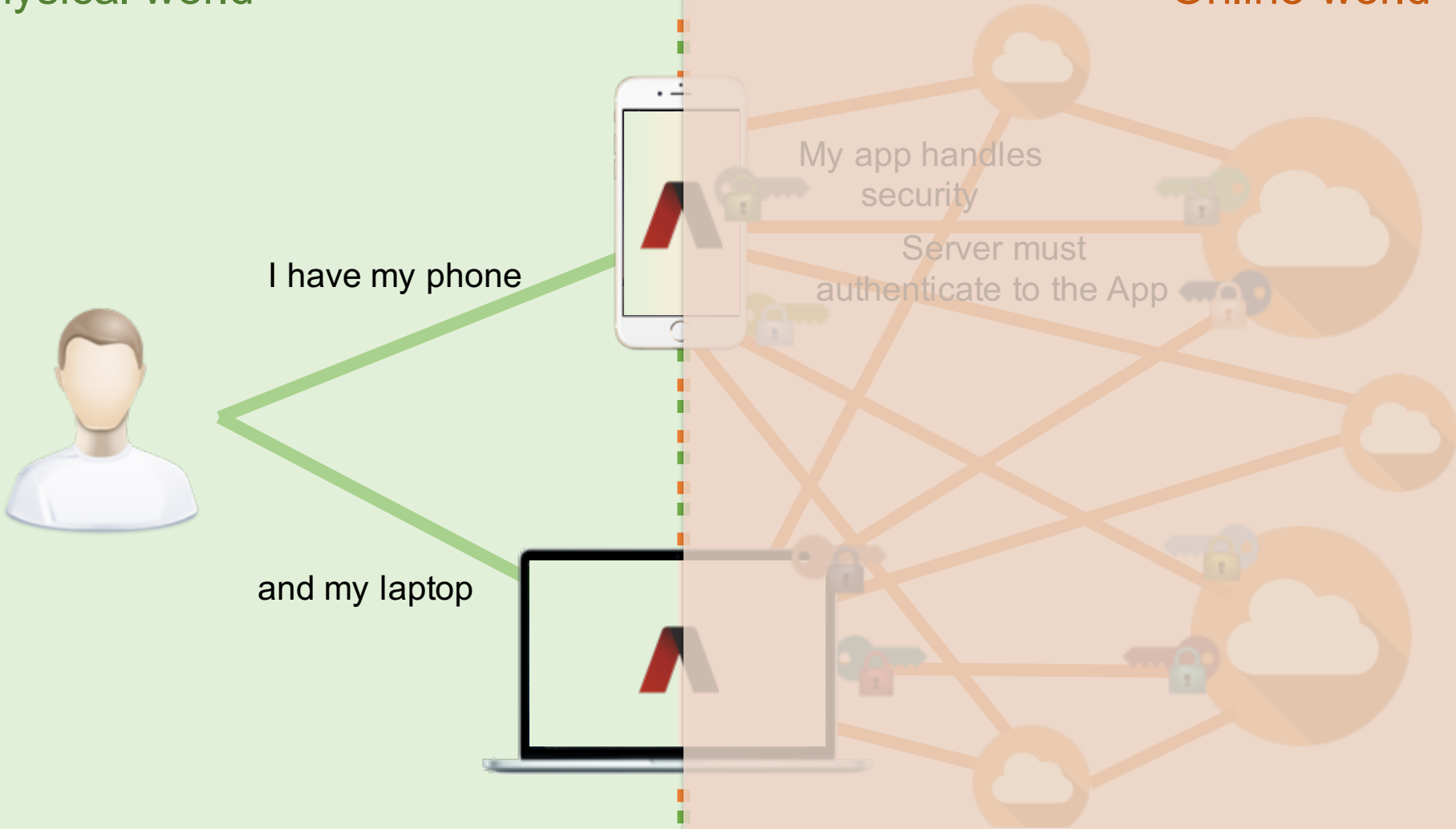

Internet Map


Physical world
DISTRIBUTED MODEL
Online world
Asymmetric cyber ID
I have my device
My app handles security
Server must authenticate to the App
No key is ever reused
Different user has her device
No master key, no reuse of keys, no single point of failure, all automatic

Physical world

USER CENTRIC UX

Online world



Absolute user simplicity. Ultimate network security. Foundation of accountability.

Are the Apps/devices a single-point-of-failure for individual users?

what should we do about it?

Local locks or data encryption?

- still a single-point-of-failure
- bad for user experience
- bad for recovery process
- static protection reliant on the user

What about specialized HW chips, smartcards or USB keys?

Maginot line
马奇诺防线

The Hacker News™
Security in a serious way

How to Crack Android Full Disk Encryption on Qualcomm Devices

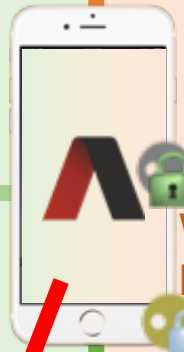
Friday, July 01, 2016 Mohit Kumar

DYNAMIC PROTECTIONS

Physical world

Online world

Sandboxing &
protections by OS



Window of opportunity limited
by the user's next login

Sharing of
security events
between services

Multi-level
HW footprint

Dynamic protections make misuse very hard

We need a security solution, which:

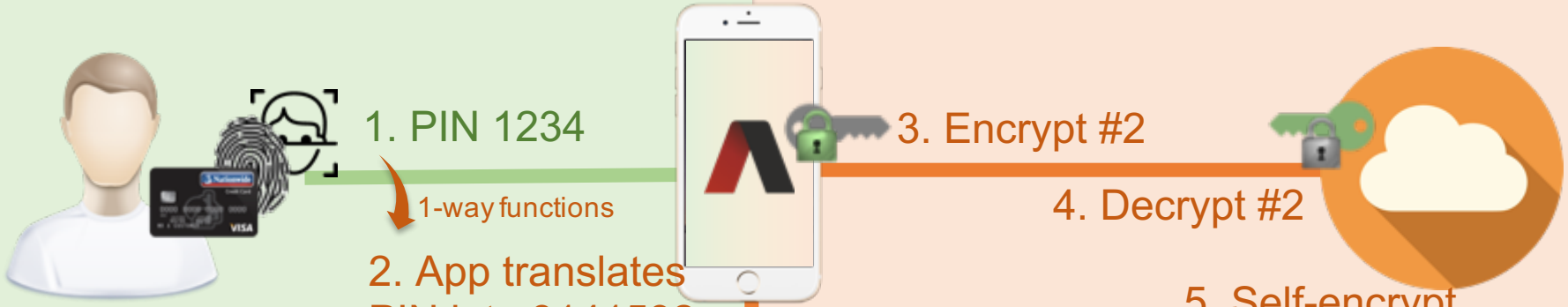
- is dynamically enforced
- dynamic in time
- not limited by user's abilities
- never stores secret on the device
- cannot be stolen from the server

Sounds impossible?

DUAL AUTHENTICATION

Physical world

Online world

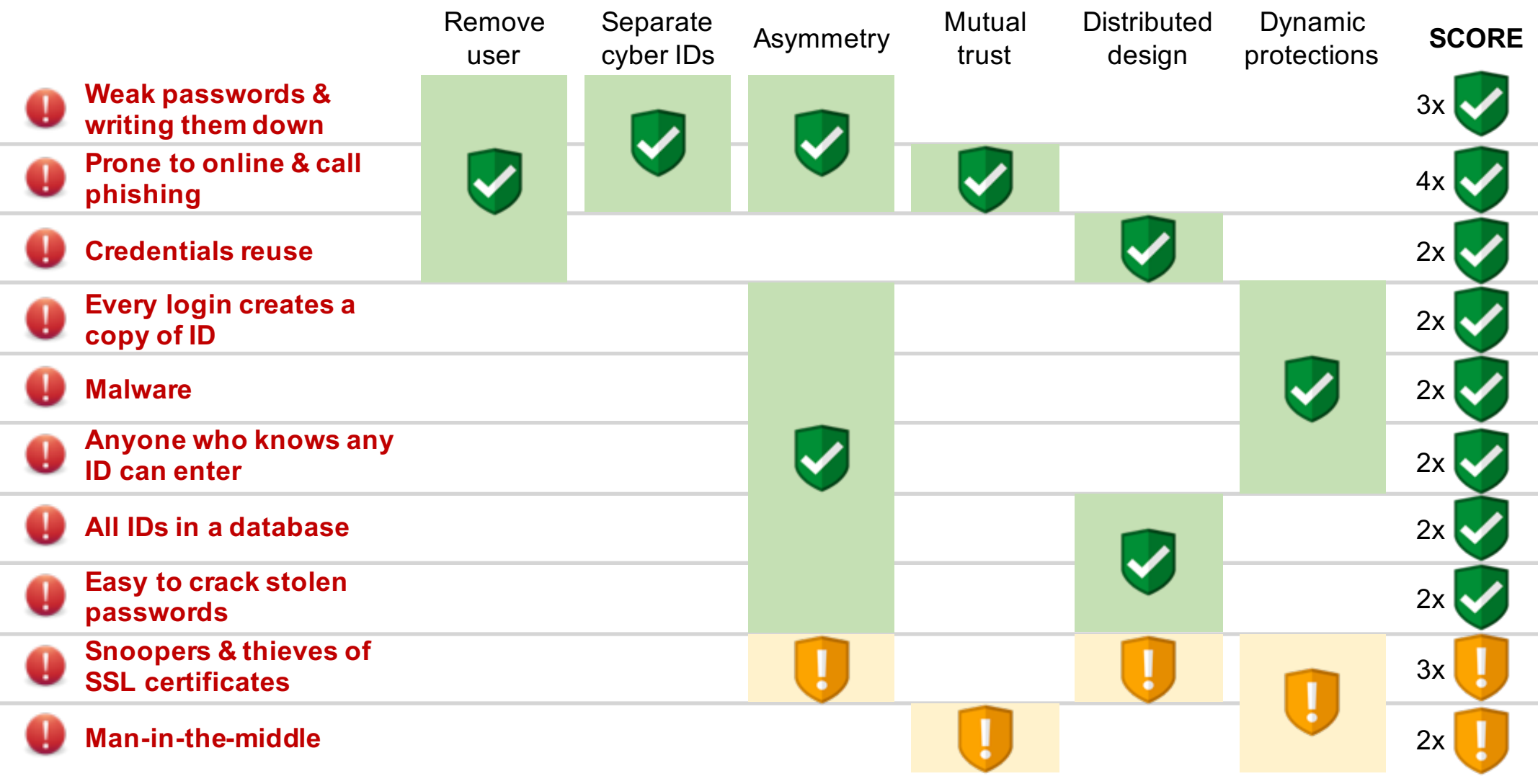


Factors are never stored and never sent.

Their “cyber ID” (#2) can change over time.

Only the server can evaluate the correctness of a factor.

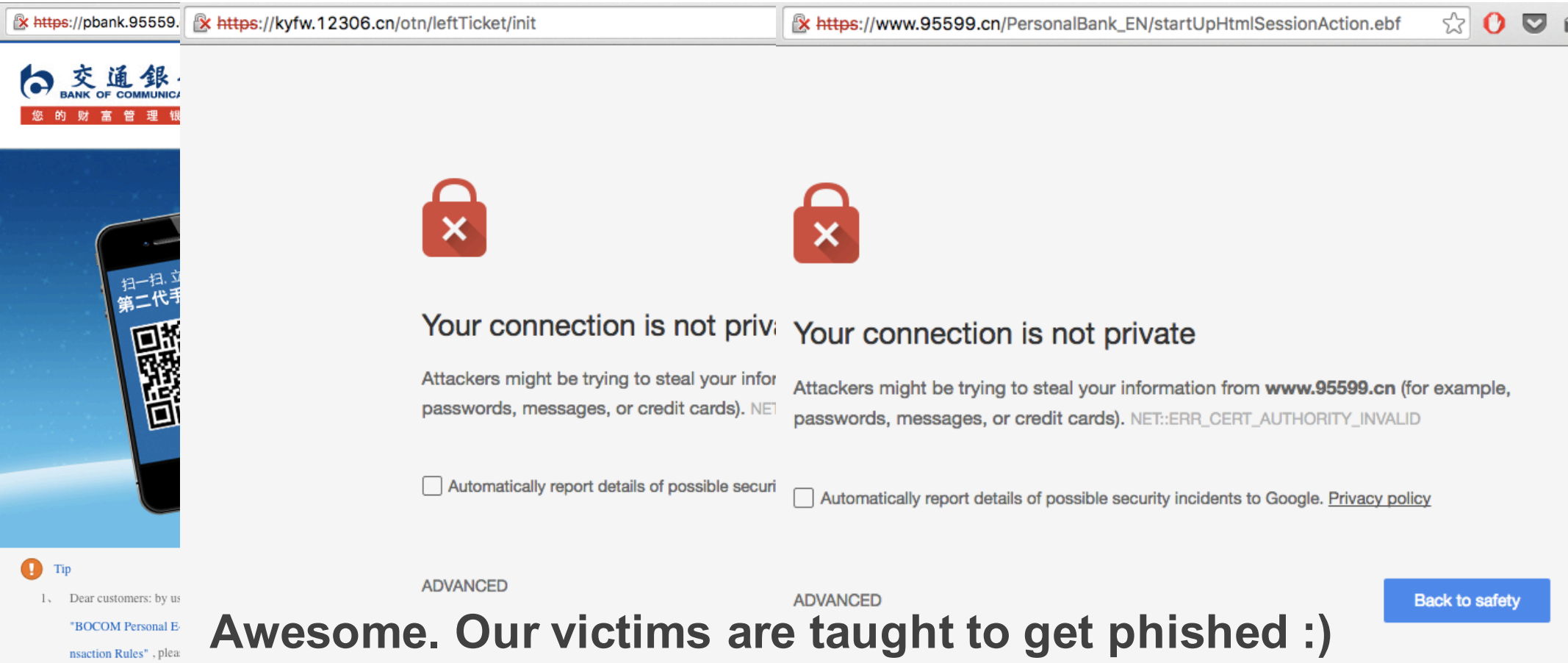
THE MOMENT OF TRUTH



Do you ever visit these pages?

What's the problem?

Any more problems?

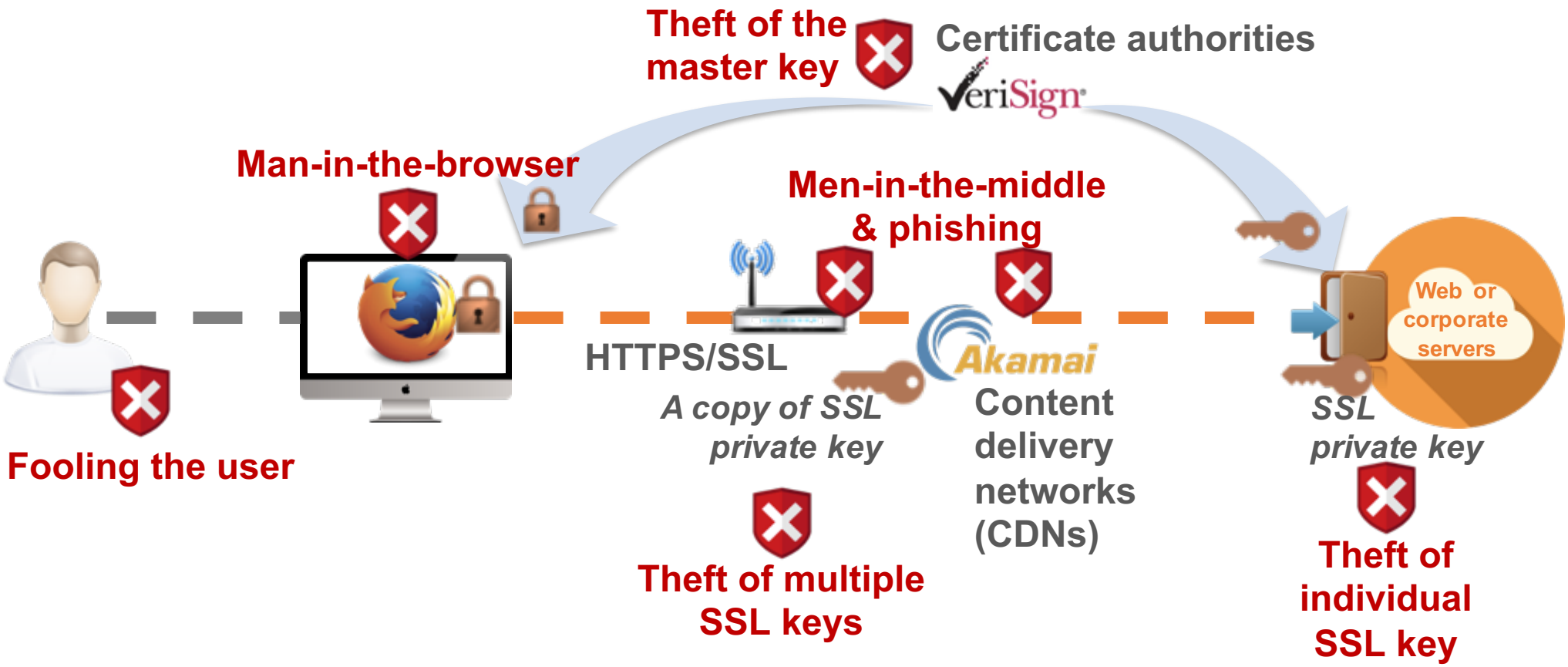


Awesome. Our victims are taught to get phished :)

These must be anomalies and the worst possible examples, right?

	Configuration score (overall)	HSTS	HPKP
中国银行 BANK OF CHINA	B	✗	✗
中国建设银行 China Construction Bank	C+	✗	✗
支付宝 ALIPAY	C+	✗	✗
中信银行 CHINA CITIC BANK	C	✓	✗
中国农业银行 AGRICULTURAL BANK OF CHINA	C	✗	✗
ICBC	F	✗	✗
兴业银行 INDUSTRIAL BANK CO.,LTD.	F	✗	✗
招商银行 CHINA MERCHANTS BANK	F	✗	✗
浦发银行 SPD BANK	F	✗	✗
中国邮政储蓄银行 POSTAL SAVINGS BANK OF CHINA	F	✗	✗
中国民生银行 CHINA MINSHENG BANKING CORP.,LTD.	F	✗	✗
交通银行 BANK OF COMMUNICATIONS	F	✗	✗
中国平安 平安银行 PING AN BANK	F	✗	✗

Case in point: Can you spot any way how you would break it?



The SPOF root-causes

- Reliant on users
- Centralization
- Static protections
- Lack of mutual trust
- Credentials reuse

CASE IN POINT: HTTPS/SSL

Physical world

Online world

I have my device

My app handles security

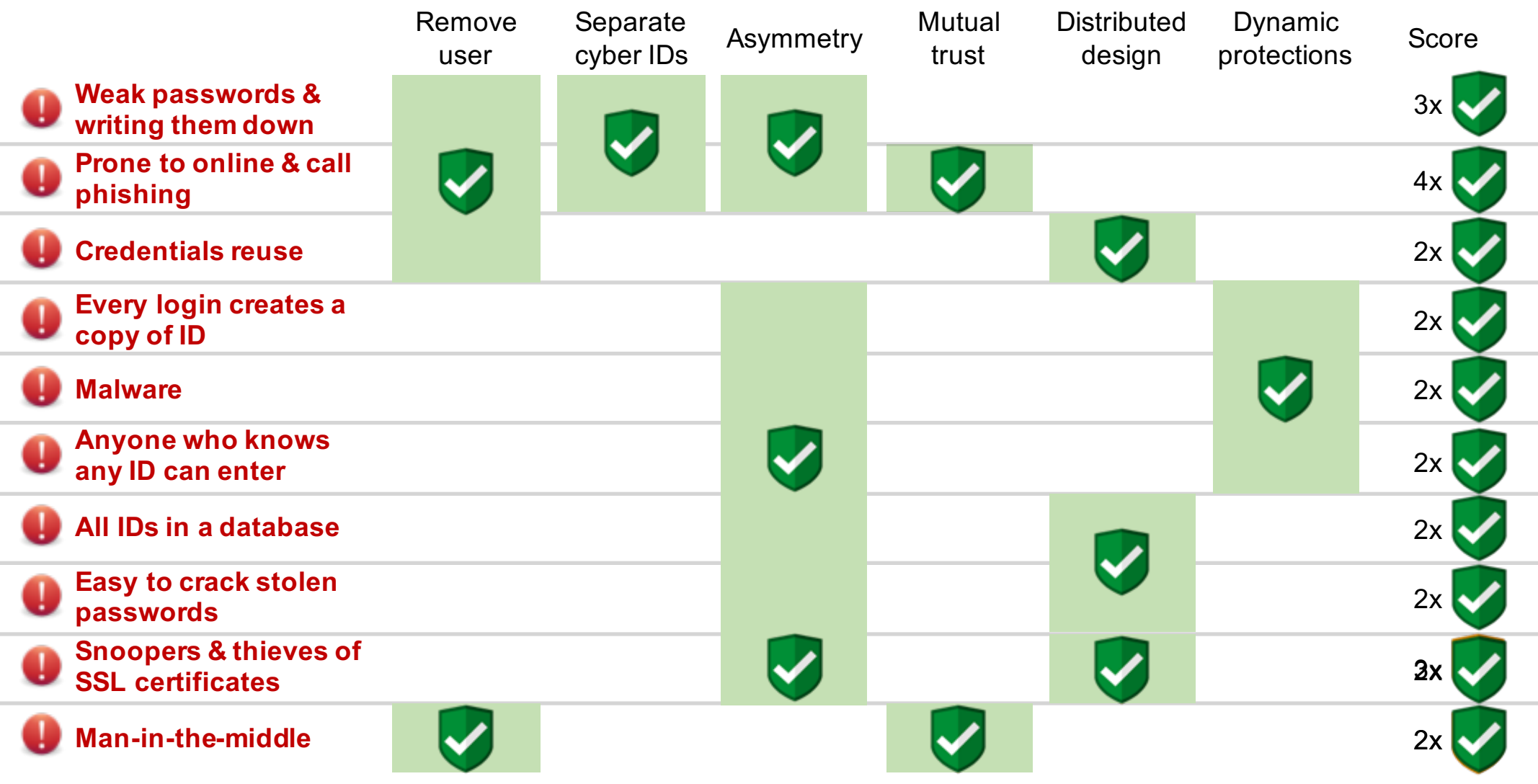
Compromised connection

Regular HTTPS connection

- No user actions (or mistakes)
- Mutual authentication builds end-to-end trust
- Distributed architecture prevents mass scale key theft
- Completely avoids Certificate Authorities and never re-uses the same SSL key

Centralization & credentials reuse

THE MOMENT OF TRUTH v2



Electronic payments

- Centralization
- Reliant on users
- Credentials reuse
- Lack of mutual trust
- Copying identities
- Physical ID online
- Static protection

Electronic signatures

- Centralization
- Reliant on users
- Credentials reuse
- Copying identities
- Static protection

Multi-party operations (e.g. online elections, real ID verification etc.)

- Accumulation of weak-points of the individual transactions

BLOCKCHAIN

Blockchain distributed database & smart contracts

- Separate cyber IDs
- Asymmetry
- Distributed design
- Reliant on users
- Credentials reuse
- Lack of mutual trust

Conclusions:

- Systematically identify single-point-of-failure
- Use following tools:
  - Remove people from security (especially users and developers)
  - Separate cyber IDs
  - Asymmetry
  - Mutual trust
  - Distributed design
  - Dynamic protections
  - Dual authentication
- These measures can be transferred to solutions of different problems