Authentication versus Encryption

https://meetup.com/Internet-Security-and-Privacy
Vancouver

by Boris Reitman

Who am I?

- Internet Security & Privacy Meetup (Vancouver)
- Secure webmail solution
- Advocate of Capitalism
- Speaker at B|Sides

Agenda

- WhatsApp
- Byzantine General’s Problem
- No auth — Shortened URLs
- OTP
- Biometric Auth + KDF
- Digital Signatures: RSA and ECC
- PGP
- Email phishing, DomainKeys
- DNSSEC, Tack
- OPAQUE

Encryption vs. Authentication

- Encryption: Guarantees “what” is said

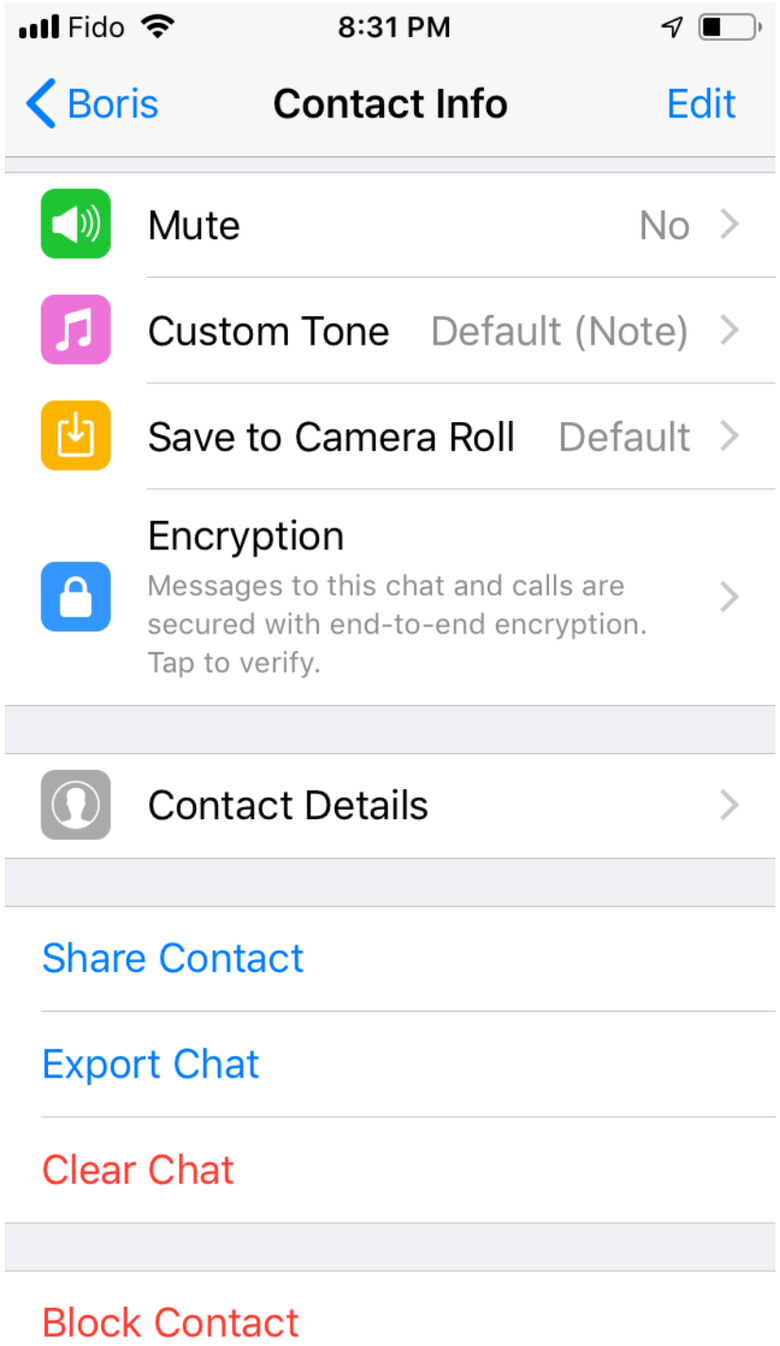
- Authentication: Guarantees “who” said it

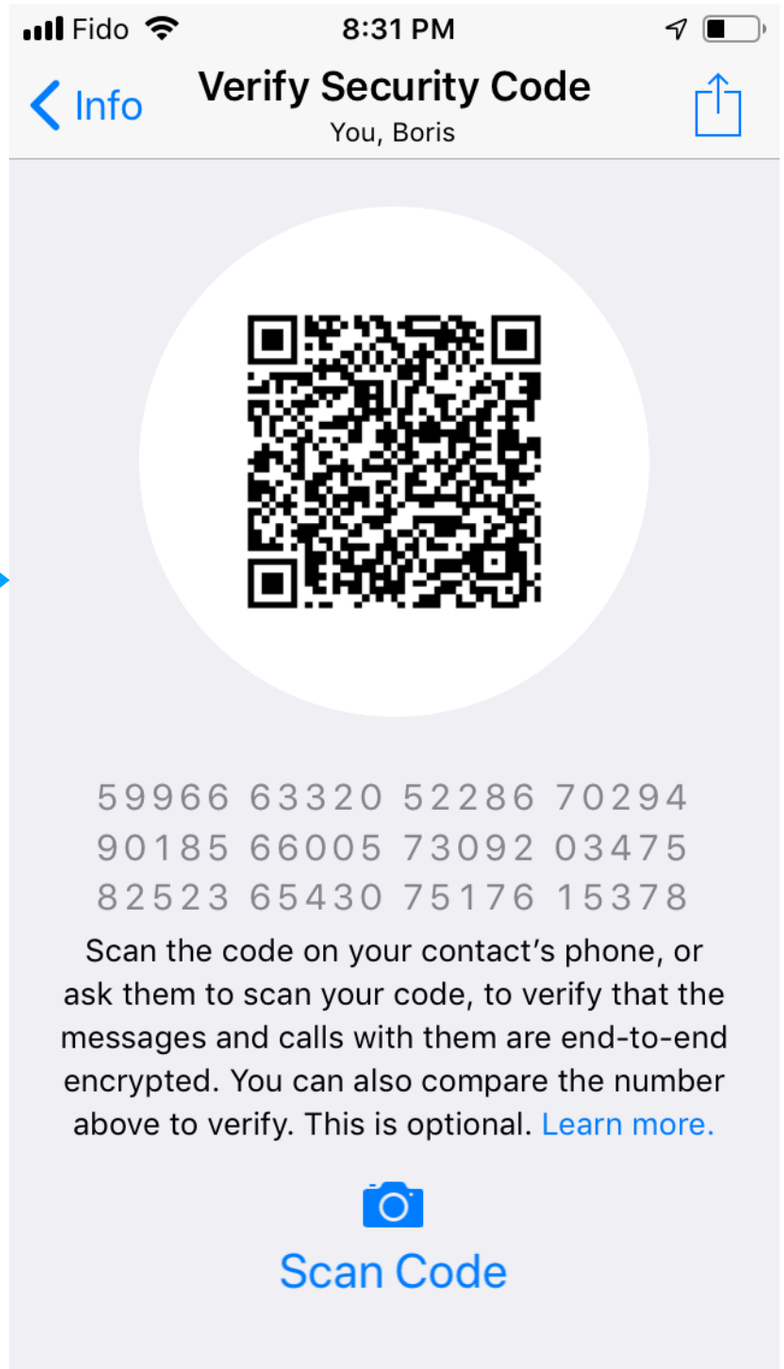
Example: WhatsApp

WhatsApp Security

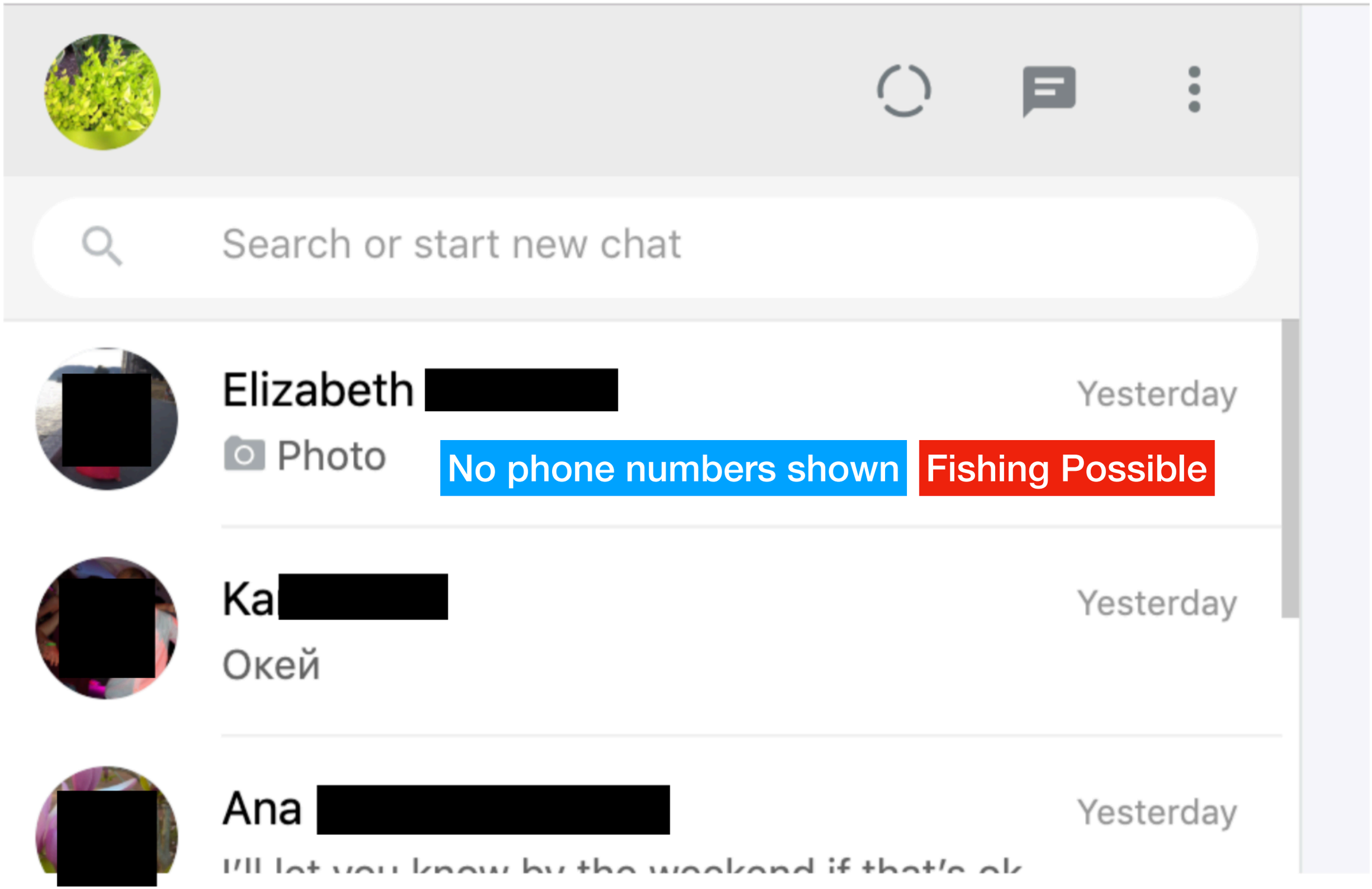
- Information is guaranteed to be end-to-end encrypted.
- Are the public keys matching ?
- Who are the people behind the phone numbers?
- What are the phone numbers behind the names?

WhatsApp verify keys





No phone numbers under contacts



Shoulder Surfing

attacker

victim

Phone Numbers

- People take over phone numbers.

- The new owner of the phone number can impersonate the person.

“Auth”

- Authentication (identification, ID)
- Authorization (ACL rules)
- OAuth2, Google Sign-In

User Identification

- Identify user by cookies
- By localStorage and sessionStorage
- By window.name
- By auto completion of (hidden) form fields.
- By browser “fingerprint”

Canvas Fingerprint

- Different graphics cards implement Canvas interface differently.

- Javascript draws text with the font and size of its choice
- Adds background colours.
- ToDataURL method to get the canvas pixel data
- Hash the result

Canvas Fingerprint

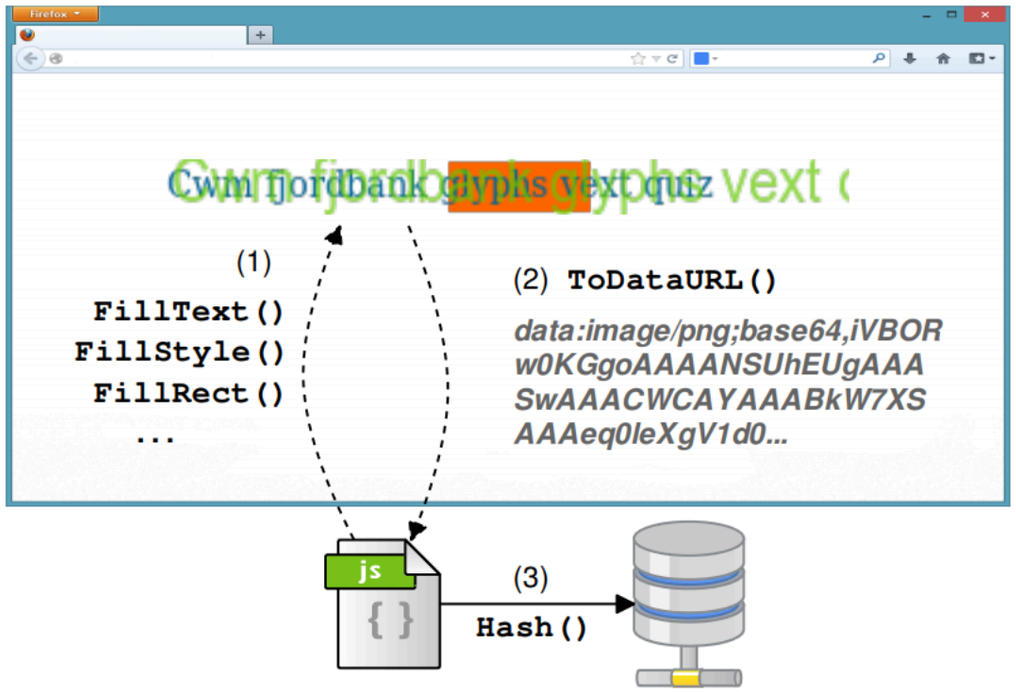


Figure 1: Canvas fingerprinting basic flow of operations

Paper: “The web never forgets”

Scripts in the wild

addthis.com

Fingerprinting script	Number of including sites	Text drawn into the canvas
ct1.addthis.com/static/r07/core130.js	5282	Cwm fjordbank glyphs vext quiz, 😀
i.ligatus.com/script/fingerprint.min.js	115	http://valve.github.io
src.kitcode.net/fp2.js	68	http://valve.github.io
admicro1.vcmedia.vn/fingerprint/figp.js	31	http://admicro.vn/
amazonaws.com/af-bdaz/bquery.js	26	Centillion
*.shorte.st/js/packed/smeadvert-intermediate-ad.js	14	http://valve.github.io
stat.ringier.cz/js/fingerprint.min.js	4	http://valve.github.io
cya2.net/js/STAT/89946.js	3	ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz0123456789+/
images.revtrax.com/RevTrax/js/fp/fp.min.jsp	3	http://valve.github.io
pof.com	2	http://www.plentyoffish.com
*.rackcdn.com/mongoose.fp.js	2	http://api.gonorthleads.com
9 others*	9	*(Various)*
TOTAL	5559 *(5542 unique[1])*	-

More info

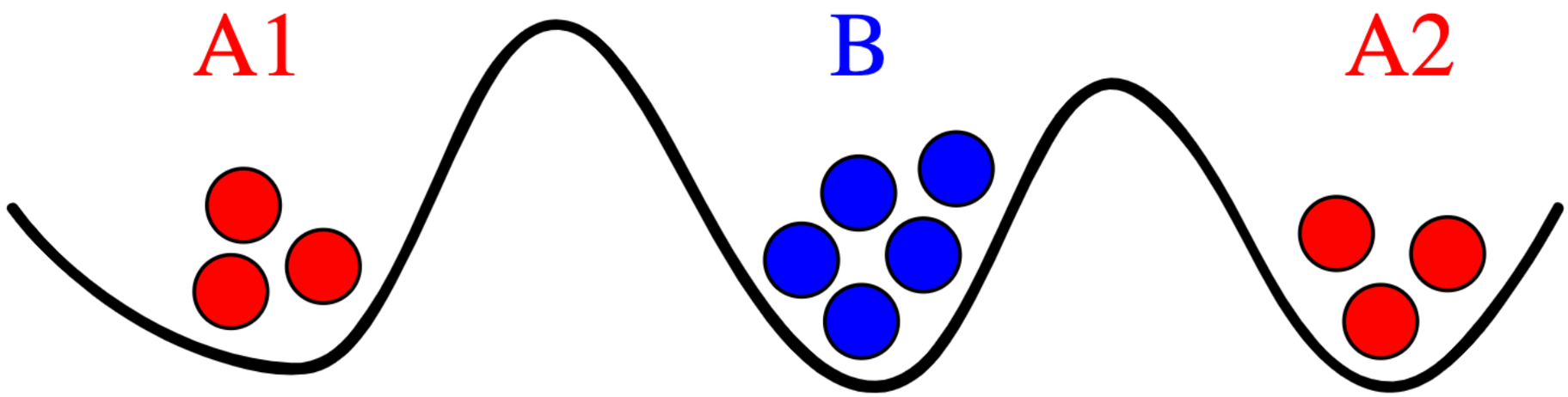
The Web Never Forgets: Persistent Tracking Mechanisms in the Wild

Gunes Acar[1], Christian Eubank[2], Steven Englehardt[2], Marc Juarez[1]
Arvind Narayanan[2], Claudia Diaz[1]

Theory

- The Two Generals Problem
- The Byzantine Generals Problem
- Proved unsolvable

Two Generals



Army “B” intercepts coordination messages between A1 and A2 armies.

Shortened URLs

- Security by obscurity
- URLs to Google Drive and Dropbox documents
- If you have the URL, you are authenticated.

URL Shorteners

- Convenient:
    - Can be pasted where there’s a length limit.
    - Can be incorporated into a text message, in the middle of a sentence.
    - Can detect vulnerable redirects, and act like an “anti-virus”.
- Hide bad urls in good urls.
- Guess URLs (brute force search)
- Collect Analytics (Tracking)
- Dangerous if the issuing service is hacked.
- Deliberate malicious url shortener services.
- http:// supporting shorteners + man-in-the-middle bypass SSL.

Shortening as Obfuscation

- Shortener hides malicious URLs and allows to bypass checks.

- Can be used for SPAM emails.

- Mitigate: checkshorturl.com

URL Shortener Example

End-to-end encryption fails

- Many web services rely on embedding an authorization key in the URL.
- The key is long and hard to guess.
- But, the shortened URL of it is easy to guess.

Trust to Shortener Service

- The shortener service knows the same location info about you that the target site will know.
- IP & geo location, HTTP Referrer, your browser fingerprint.
- Can add its own cookie, to link all your sessions into one profile. (Knows all websites that you visit.)
- Mitigation?
  - Block cookies for known shortener services.
  - Still possible through browser fingerprinting.

Shortener Service Hacked

- The service cli.gs got hacked (2009)
- 2.2 million URLs got redirected to a different destination.
- (according to their own blog report, later.)
- Affects all existing links, that previously were considered clean by users.

Malicious Shortener Services

- Operate a legitimate looking URL Shortener Service (USS)
- Redirect only some clients selective to malicious pages.
  - Clients using old browsers with known vulnerabilities.
    - Advanced users would not notice, because they use the latest browsers.
  - Clients specifically targeted.
  - Sell portion of traffic to botnets.

SSL Bypass

- If the target website is https://
- But, the shortener is http://
- Then a MITM can intercept, using SSLStrip.

SSRF

- USS services usually check the links submitted, to avoid linking to other competing USS.

- They run the check from their own server.

- If a certain website is vulnerable through a URL request, an attacker can get the USS service to issue the request on his behalf, thereby obfuscating his identity.

Mass Surveillance

Observation by @dakotasmith :

“Twitter's URL shortener is a Colombian domain. Facebook's domain is in Montenegro. Bit.ly is Lybian. NSA only targets international traffic.”

Brute Force Shorteners

- What can be gained by trying out random short URLs?
- This paper is from April, 2016.
- Exposed OneDrive and Google Maps short links.

Gone in Six Characters:
Short URLs Considered Harmful for Cloud Services

Martin Georgiev
independent *

Vitaly Shmatikov
Cornell Tech

Scan against official API

- Bit.ly API:
  - 2.6 queries / second, indefinitely.
  - 227 queries / second, results in temporary blocking of IP address.
- Google Maps API (goo.gl/maps):
  - 1M queries per day, for free. Higher quote if paid.

Brute Force Algorithm

- Generate 100 million random shortened URLs.
- Hit the API, find out which of them are valid
- Also, estimate density
- Math: density = # found / 100 M.
- Given the density, estimate total URLs encoded.

OneDrive Results

- Found: 42 million working URLs on bit.ly 6 char space.
  - 2000 of these are to OneDrive, and are live
- Found: 30 million working URLs on bit.ly 7 char space
  - 25,000 of these are to OneDrive, and are live
- Other user's files on OneDrive can be traversed, given one good URL.
- 300 OneDrive documents allow *write* access

Traverse OneDrive

- http://1drv.ms/1xNOWV7 (operated by bit.ly)

- https://onedrive.live.com/?cid=485bef1a805
  39148&id=485BEF1A80539148!115&
  ithint=folder,xlsx&authkey=!AOOp2TqTTSMT5

- Use this parameters and construct the root URL.

- Then, parse the HTML of the root page, to find links to other documents.

OneDrive Traverse Results

- bit.ly 6 char space:
    - Traversing from 2000 live OneDrive urls led to:
        - 227,000 more OneDrive readable documents.
- bit.ly 7 char space:
    - traversing 22k documents, led to:
    - 1M more documents.

Exploit Delivery

- OneDrive editable folders, allow to deliver exploits
- Exploits land on user’s hard drives.
- 2000 OneDrive documents, led through traversal to 150 editable folders.
- Microsoft’s Response: fixed the problem by removing shortened urls altogether, but stated that the report is not a vulnerability.

Google Maps Results

- Found: 24 million working URLs on google maps.
- 10% of them are for directions.
    - Sensitive locations: special clinics (cancer, mental illness), addiction, abortion, gentlemen’s clubs, correctional facilities, payday lenders.
    - Identify the individual using one of the “home” end-points.
- 90% are individual’s location.

Google Maps Results

- Found: 24 million working URLs on google maps.
- 90% are locations.
- 10% of them are for directions.
- Sensitive locations: special clinics (cancer, mental illness), addiction, abortion, gentlemen’s clubs, correctional facilities, payday lenders.
- Identify the individual using one of the “home” end-points.
- Create a map of who has visited whom.
- Find all clients of a particular business. Example: car towing.
- Google API for short URLs reveals *when* the URL was created.

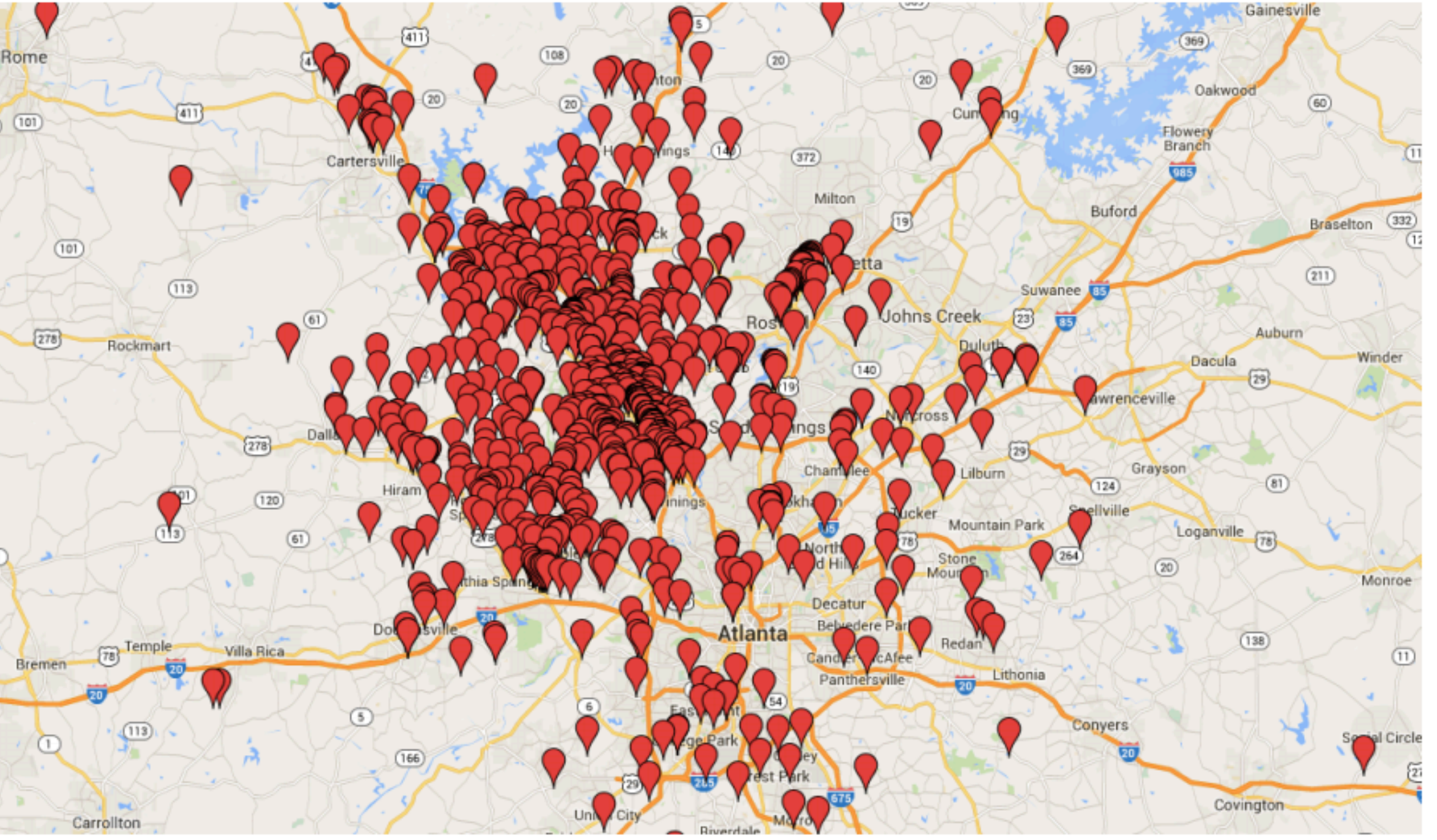


Figure 5: Locations associated with D & D Autows Inc.

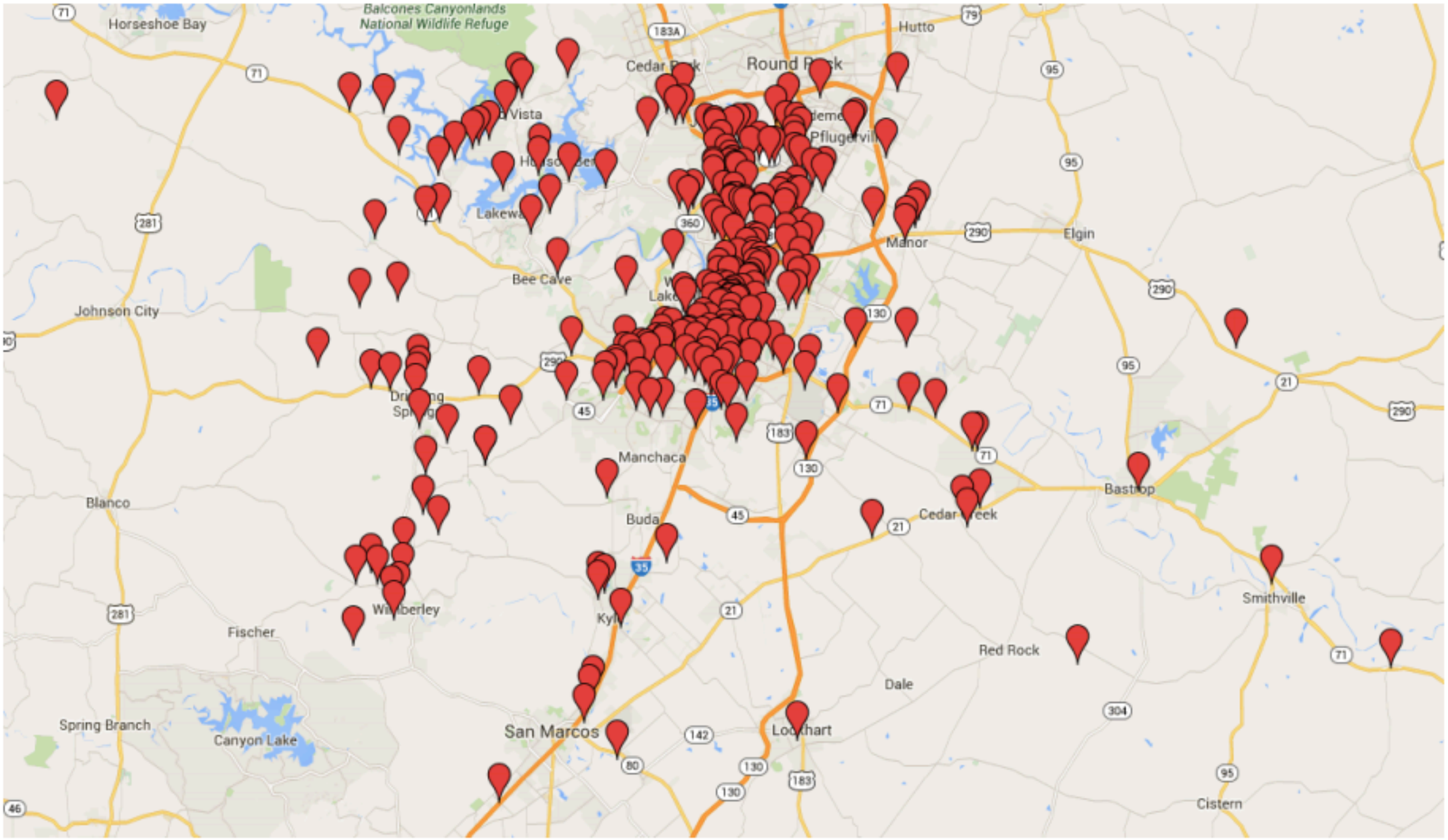


Figure 4: Locations associated with a single user in Austin, TX.

Responses to CVE reports

- Microsoft:
  - Fixed the problem by removing shortened urls altogether,
  - But: stated that the report is not a vulnerability.
- Google:
  - fixed, by increasing URL length to 12 characters.

One-Time-Password

- Hash Chain
- Original idea by Leslie Lamport
- Hash OTP, HMAC-OTP, Time-OTP

Hash Chain

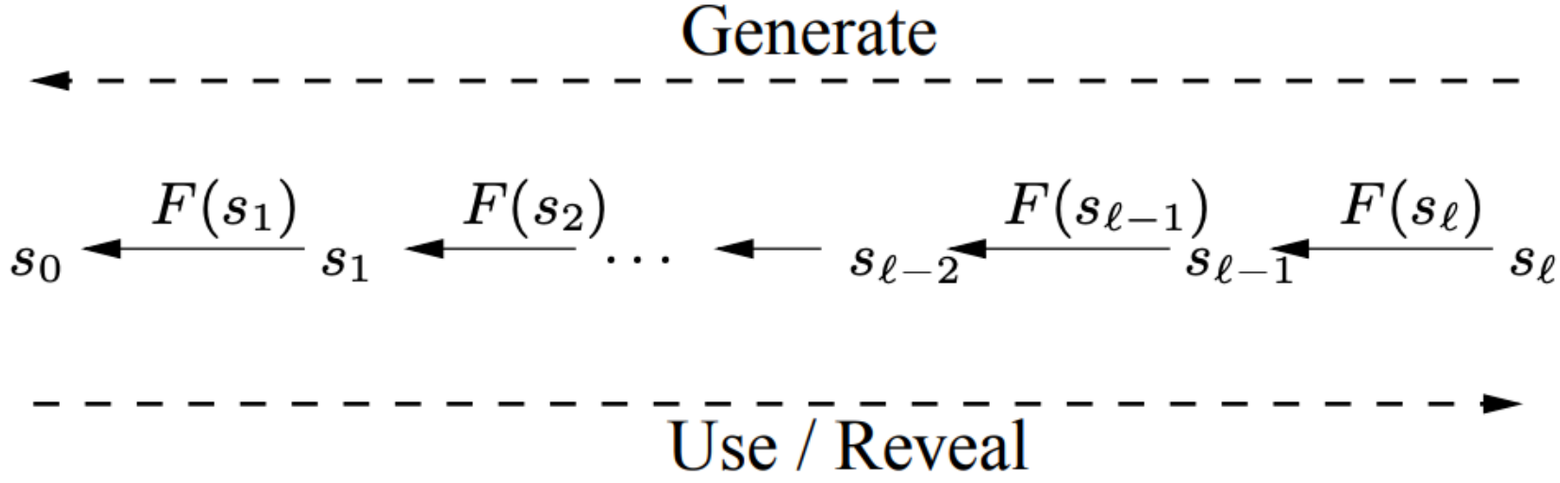


Figure 1: One-way chain example. The sender generates this chain by randomly selecting s_ℓ and repeatedly applying the one-way function F. The sender then reveals the values in the opposite order.

Key Derivation

- Authentication = identification

- Key Derivation = Authentication + end-to-end encryption

Key Stretching

- scrypt
- bcrypt
- PBKDF
- input: short password, output: long enough key
- must be memory, computation and space expensive.

Biometric Authentication

- Touch ID
- Face ID
- Typing speed pattern
- Voice
- Imprecise answers to a list of questions

Biometric Authentication

- Calculate delta between expected input and received input

- If delta is small, user is authenticated.

Biometric Key Derivation

- Idea: Quantize input and add a correction
- Registration:
    - F1 is the input, generate key
    - Also, calculate F2 = quantize(F1), and D = F1-F2
    - Store D on the server.
- Login:
    - F3 is new input. Calculate: quantize(F3)+D.

Digital Signatures

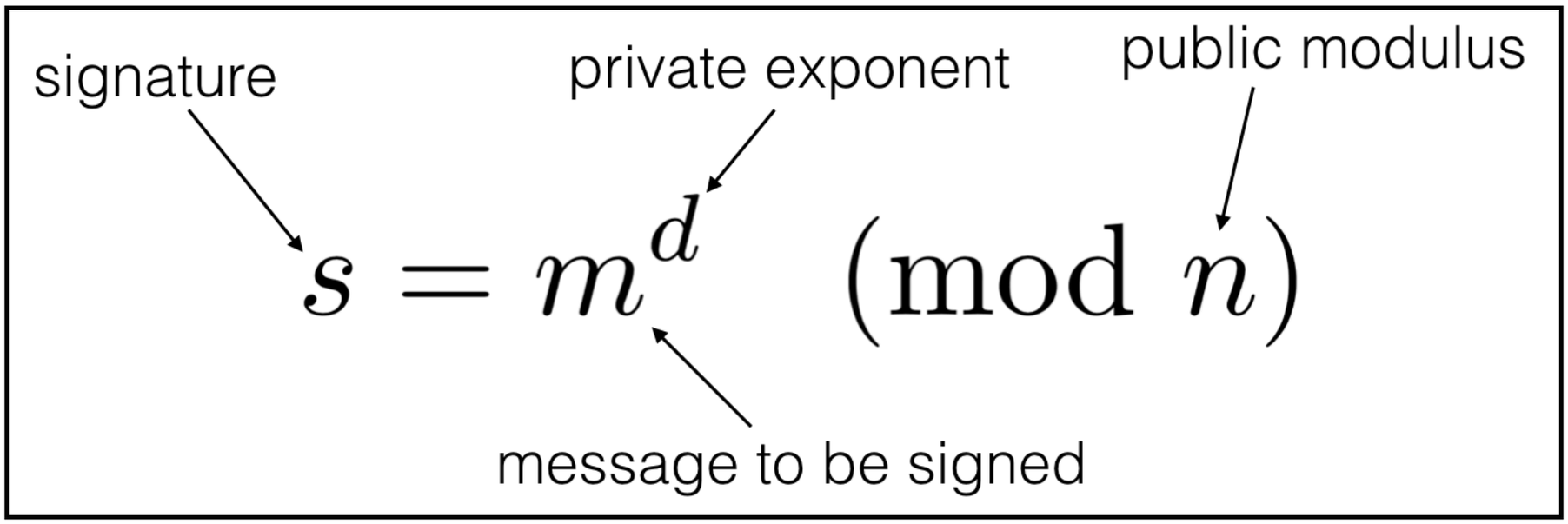
- RSA method

- Elliptic Curve method

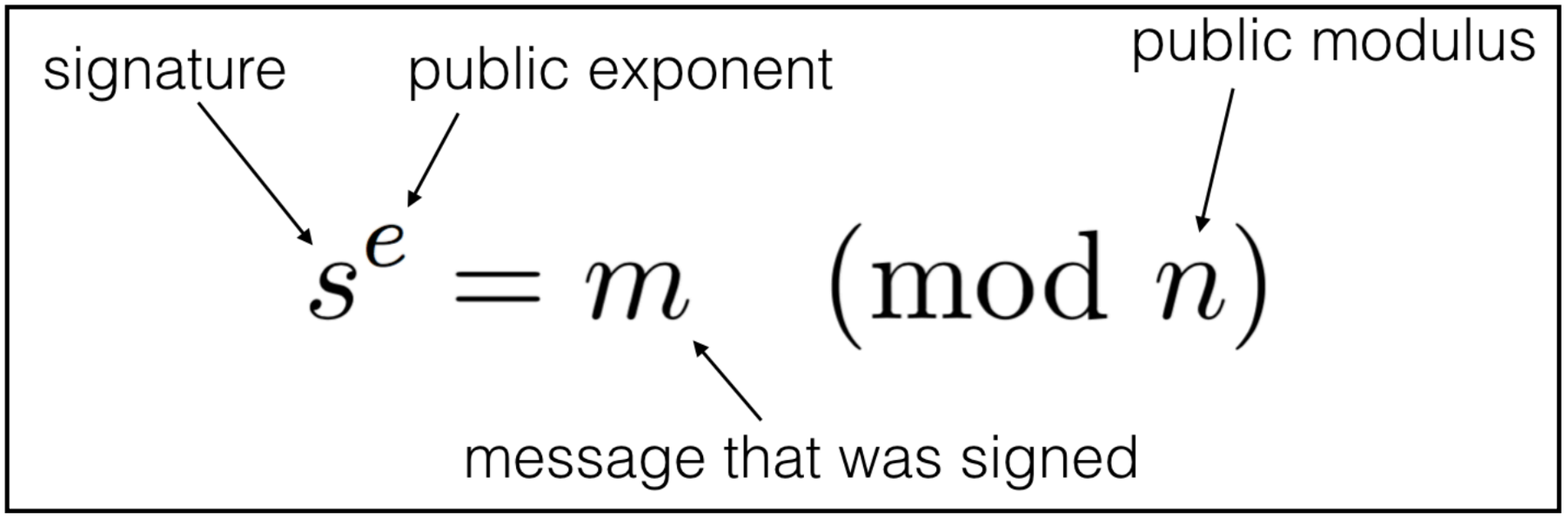
RSA signatures

- Sender encrypts “foo” with sender’s *private* key
- Recipient decrypts signature with sender’s public key.
- Signature is correct, if recipient sees “foo”.

Sign


signature
private exponent
public modulus
$s = m^d \pmod{n}$
message to be signed

Verify


signature
public exponent
public modulus
$s^e = m \pmod{n}$
message that was signed

RSA problem

- But: Recipient doesn’t know senders public key.

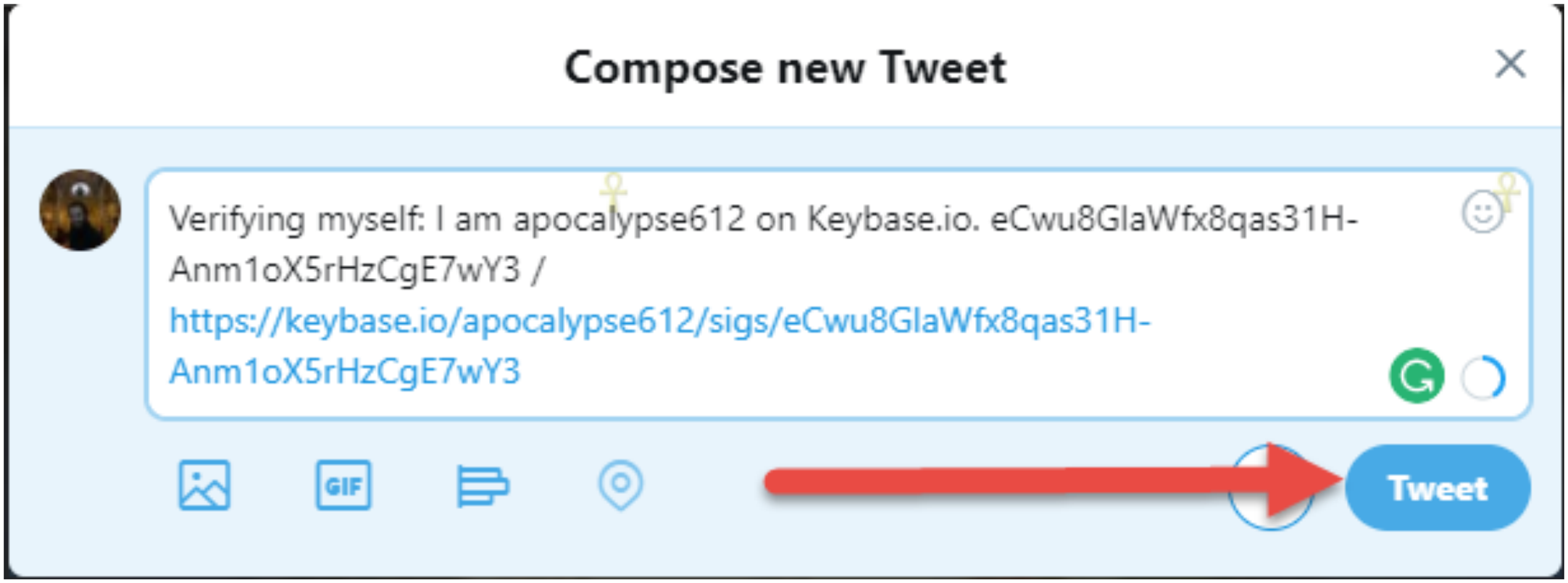
Solutions

- keybase.io
- PGP servers
- GPG web of trust

keybase.io

- Verify user's identity based on his membership on social networking websites.

- Twitter, Facebook, etc.



Bitcoin Transaction Verification

- Uses an Elliptic Curve group secp256k1
- Group of points (x1,y1), (x2,y2), …
- Points satisfy curve: $y^2 = x^3+7 \pmod p$.
- Points P and Q can be combined to produce new point P(+)Q in the set, that also satisfies the curve equation.
- P + P + P is written as 3•P.
- All points are of the form n•P.

Zero Knowledge Proof

- Equation: $h \bullet Q + R = s \bullet B$
- $h = hash(m,R)$, Q is the signature, B is base point
- Signature (m, R, s) proves that:
  - you know private scalar t that:
    - $Q = t \bullet B$
    - set $R = t \bullet B$
    - factor LHS into $ht \bullet (Q + B)$

RFID Authentication

- The ID on the card
- Attack: replicate the same ID on another card
- Card cloning
- Magic card: allows to configure any ID
- Master card: opens all doors in a hotel.

$2 per card

CHENGKA MF1 S50 Clone Copy Back door Rewritable Blank RFID Card Chinese Magic Card 13.56MHZ Changeable MF S50 1K NFC Card

5.0 (2 votes) | 8 orders

Anniversary Sale **US $ 1.66 - 2.07**

For big savings, collect coupons now.

Starts in 5 days

Price: ~~US $2.40 - 3 / Set~~

Discount Price: **US $1.68 - 2.10** / Set -30%

Get our app to see exclusive prices

Blank RFID card

Proxmark3 card

NETWORKING / WIRELESS NETWORKING

Proxmark3 RDV4 Kit

From: $300

Proxmark3 RDV4 Accessory Pack

Add for $100

Mifare Android App

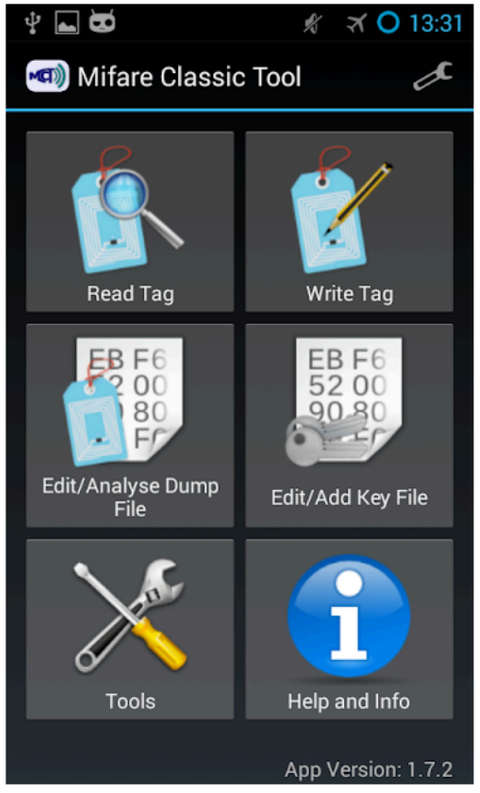
MIFARE Classic Tool - MCT

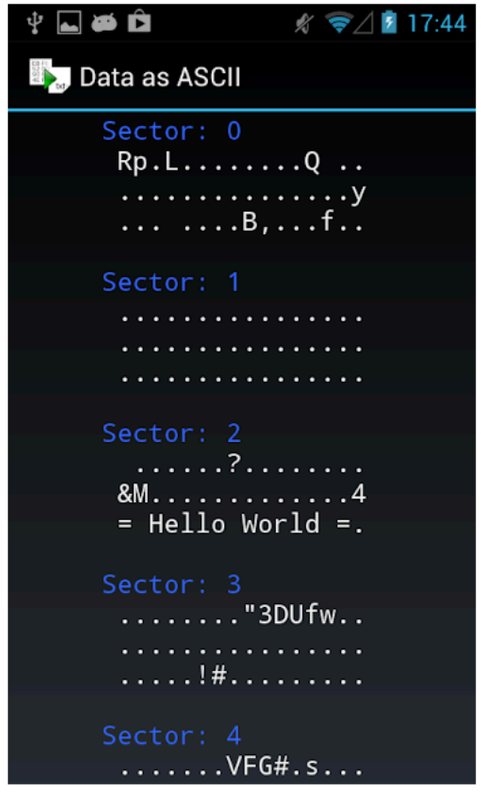
IKARUS Projects Tools 913

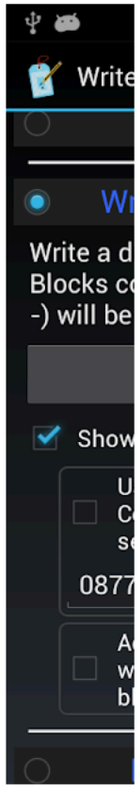
Everyone

You don't have any devices.

Add to Wishlist Install







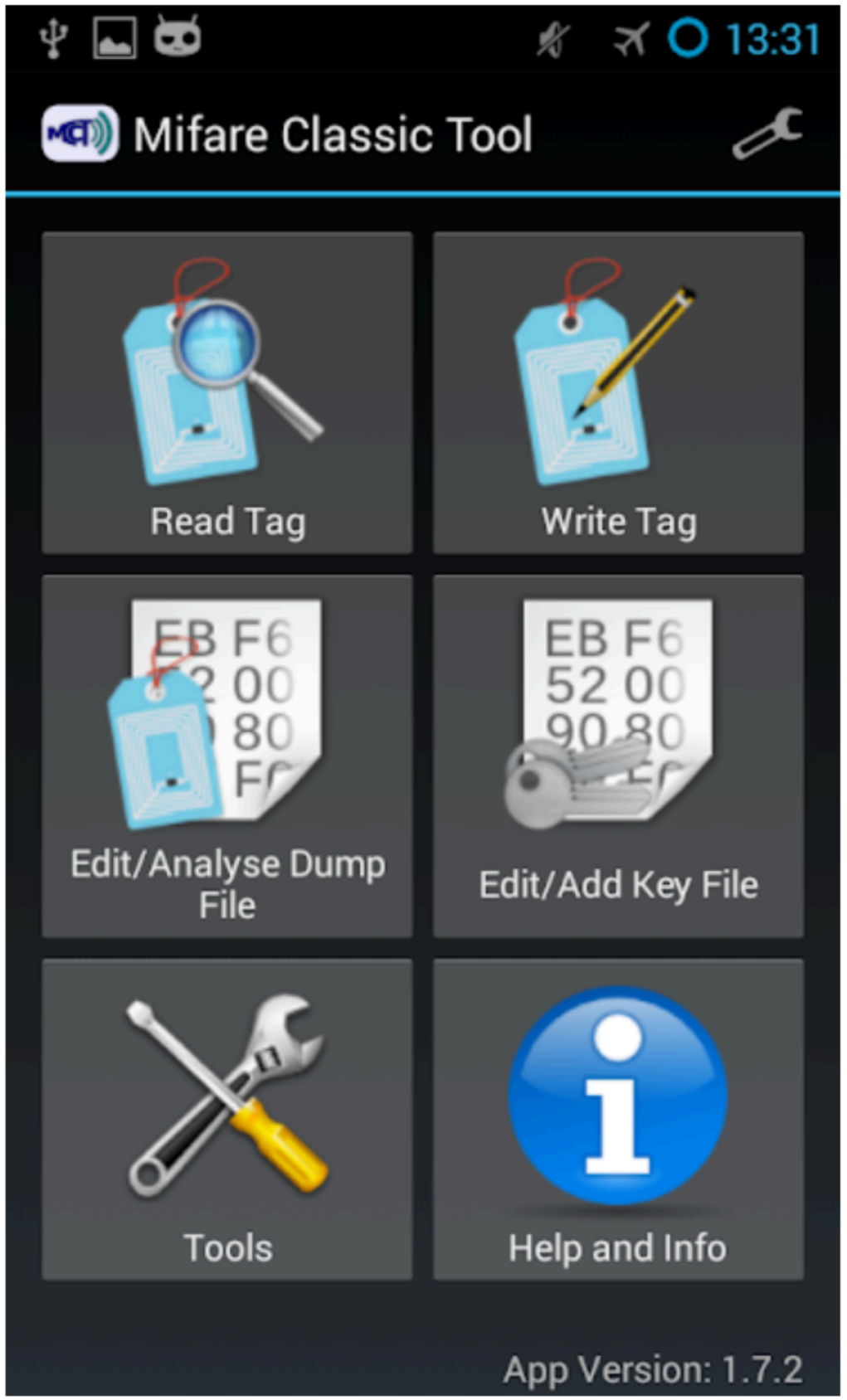
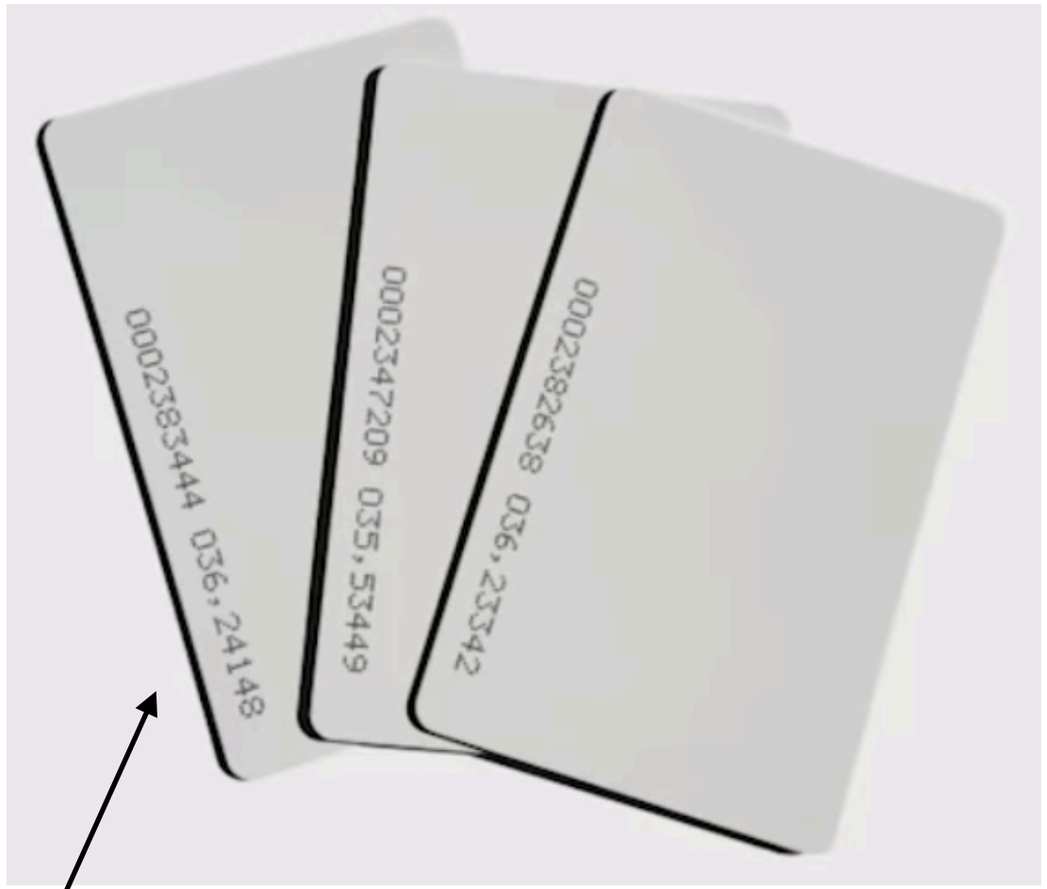
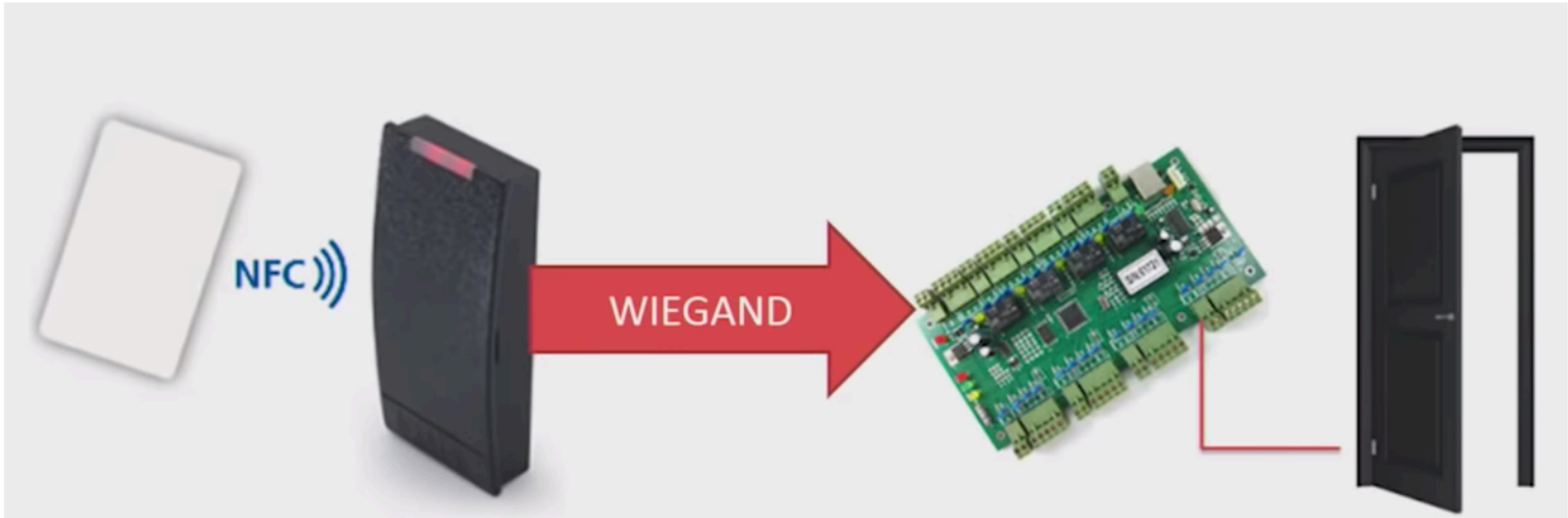

13:31
Mifare Classic Tool
Read Tag
Write Tag
EB F6
52 00
90 80
Edit/Analyse Dump File
Edit/Add Key File
Tools
Help and Info
App Version: 1.7.2

Photo of a card



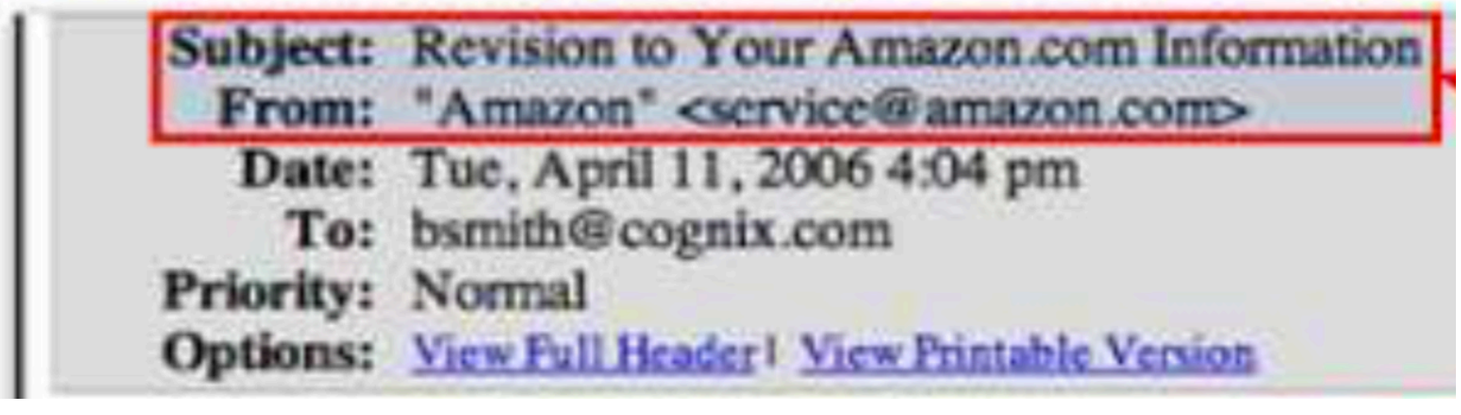
ID is decimal

Capture Reader’s output



Email Fishing

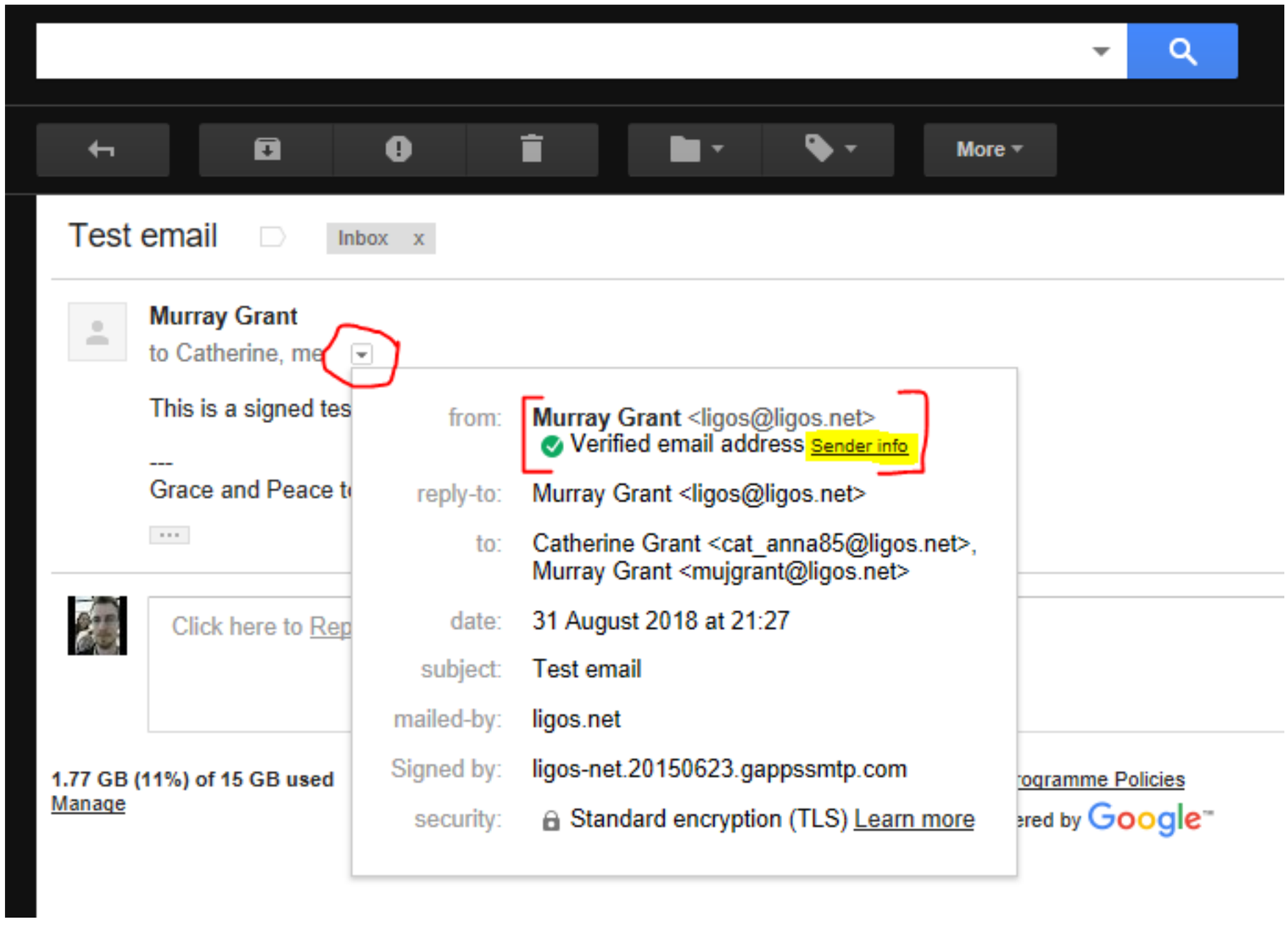
- Impersonate name of a sender
- Impersonate photo of a sender



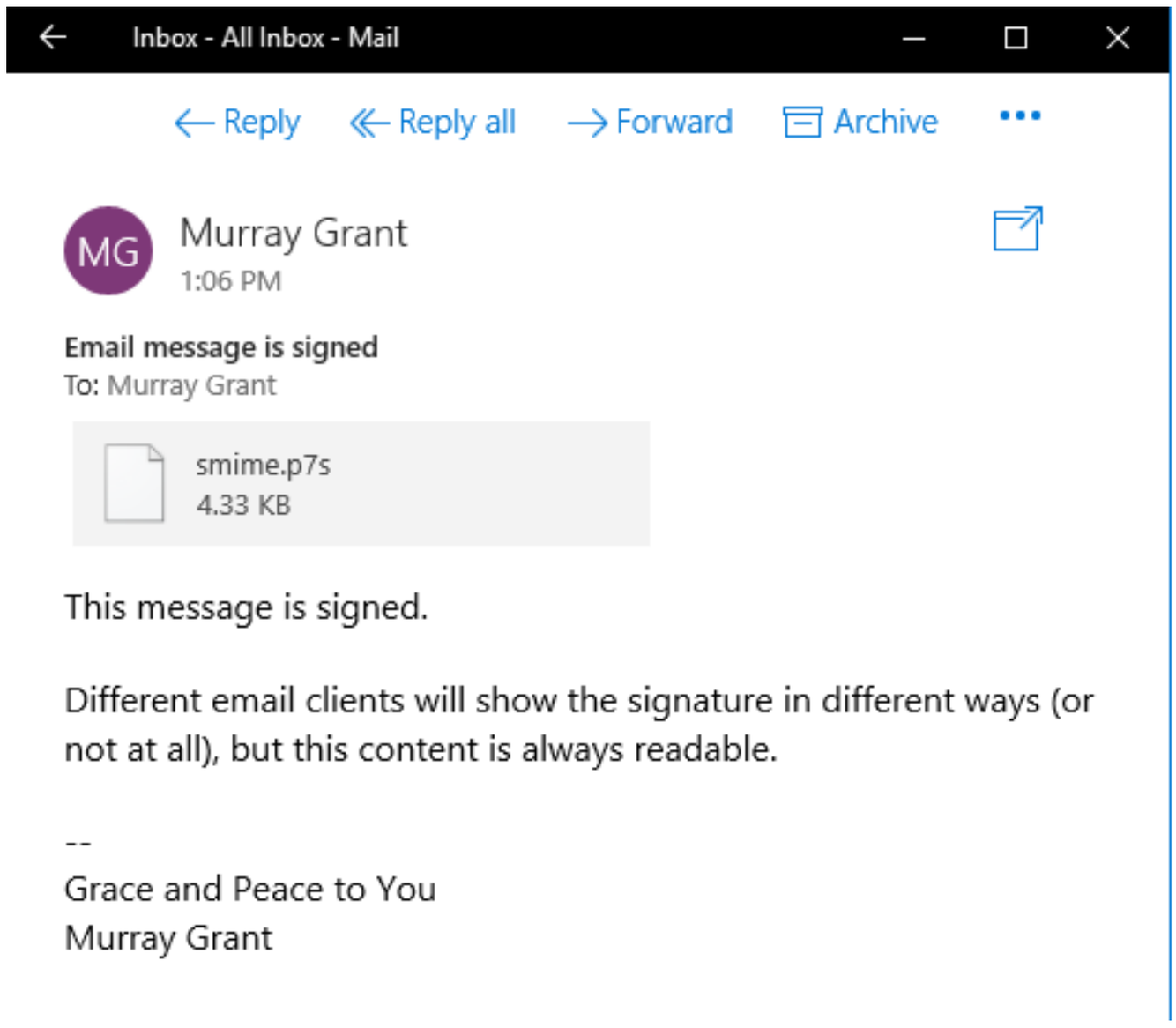
S/Mime

- Certificate is associated with the sender.
- Recipient must trust Certificate Authority (CA) that signed the certificate.
- To send, you must have the certificate of recipient.
- However, this works well for signing.
- But, it attaches a weird attachment file.

Gmail S/MIME signature



Gmail S/MIME signature



The Problem

- Who is “Murray Grant” ? Any one who can prove this name, can get a certificate like that.

- Chinese Names: there are many people with the same last name such as “Cho”.

PGP

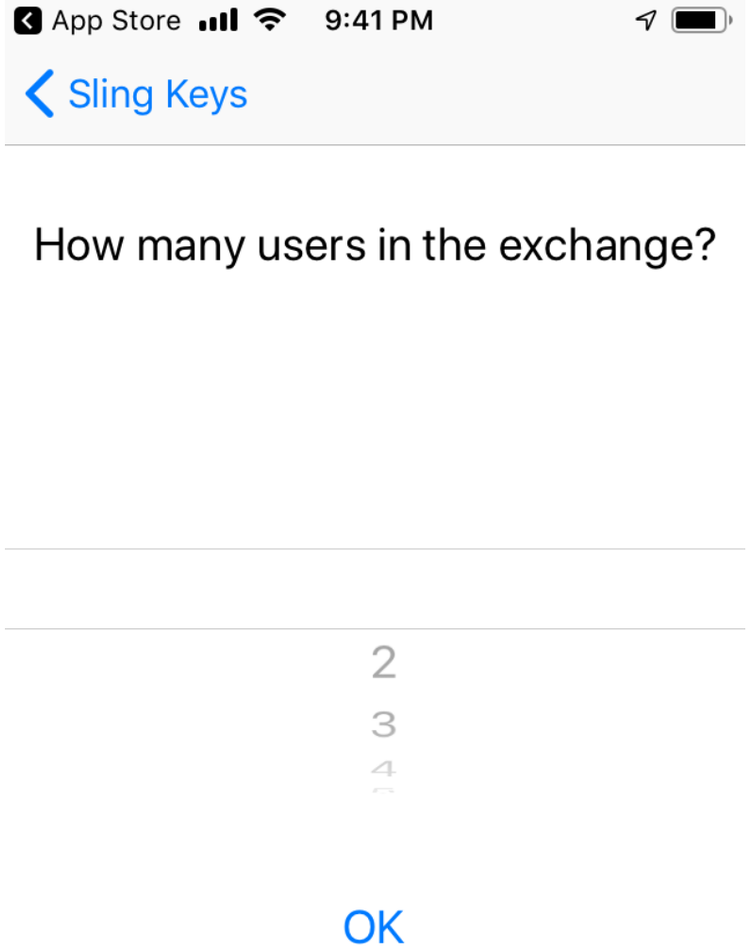
- PGP is RSA + standard to embed things in the body of email.
- Requires distribution of public keys. How?
- GPG, and PGP servers
- PGP signing parties.
- Problem: public keys get lost, email becomes unreadable.

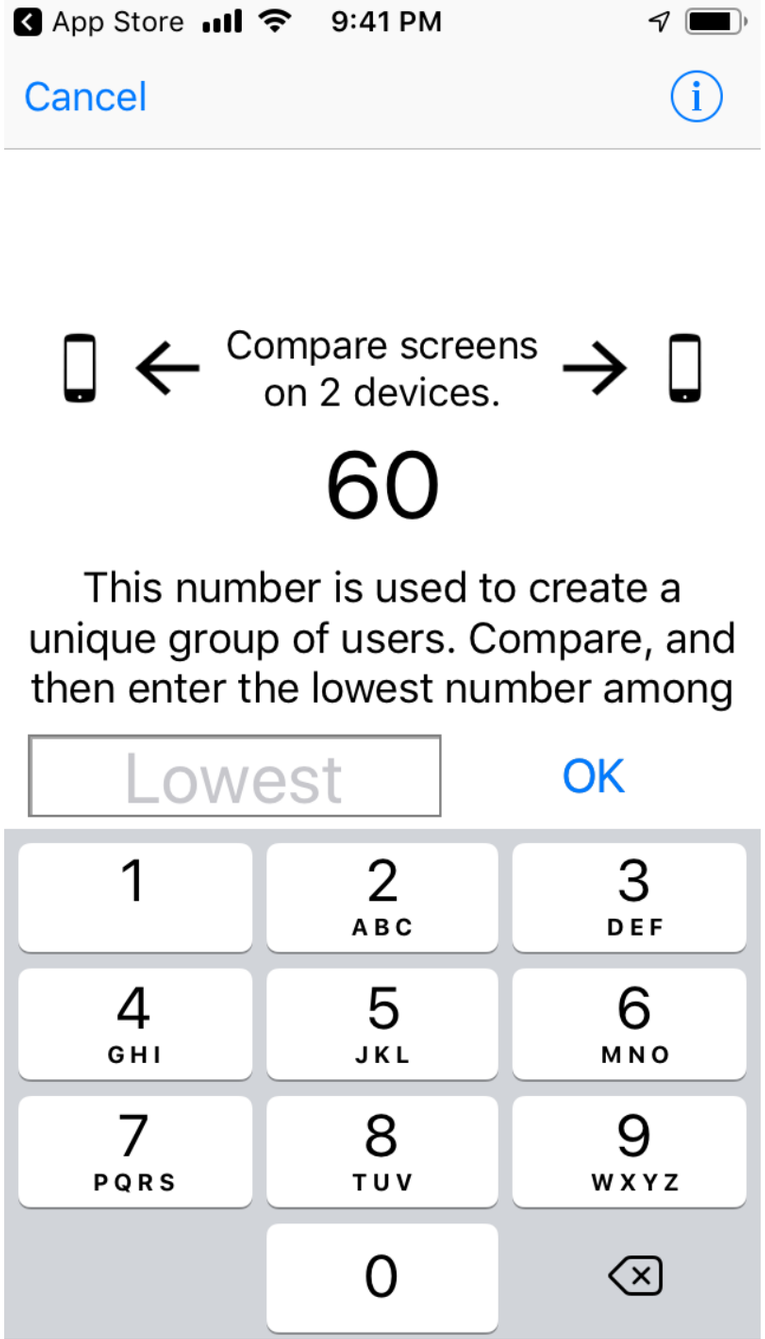
SafeSlinger

- A mobile app to share contacts, in-person. A modern version of a PGP signing party.

- By Carnegie-Mellon university.

SafeSlinger





DomainKeys

- A mail server signs outgoing mail.

- Email header “DKIM”.

DomainKeys

```
DKIM-Signature: v=1; a=rsa-sha1; c=relaxed/relaxed; d=meetup.com;
        h=from:sender:to:subject:mime-version:content-type:content-transfer-encoding;
        s=s1; bh=7rVlVxx51rnBRZGnOKcysrxyY0A=; b=PCaPuc/qf5XLgGblT8ZuYRJ
        hWChagq6JRiz7EfUrlpejC0yvcenqlKUpl7H1I3XA2RzcNSDZRVn5Ig890ZC0tYZ
        kkCliVFVGUr7h6OsSGR+mEQ6pHIU8BTBSfdIbgqtlmPcL1Pgnk9+5QS7ZFkgo29V
        hTuxhdyAqTHR+06M5vJY=
```

```
$ dig TXT s1._domainkey.meetup.com

s1._domainkey.meetup.com. 300   IN      CNAME   s1.domainkey.u3863915.wl166.sendgrid.
et.

s1.domainkey.u3863915.wl166.sendgrid.net. 1800 IN TXT "k=rsa; t=s; p=MIGfMA0GCSqGSIb3
QEBAQUAA4GNADCBiQKBgQChK0RIGEj4ahkPXIhENbmXC6Cu2Q1eVNDM6nZdrJGR2p4jWYNVGQ/EYQRC35Qu+r
cvNvayv8igvCou1A9Y6xso1ls6MCMpT3LjatFo+U+qfMI9Uh6P0sQ+NNS7NAGc0GGl8bAxi+mbG0AHgbgrB6D
JwAz7uGd0IzjPtPdn5EuQIDAQAB"
```

Gmail DKIM

body hash

```
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed;
        d=gmail.com; s=20161025;
        h=mime-version:references:in-reply-to:from:date:message-id:subject:to;
        bh=0SfhqDZkUTtM2Aqg0EJJAngQdx1dHhjX4xG5mGEhy/Y=;
        b=qINF14m9coo6mVBl8gukDDcycH3kKD8nEwQ+xfFYU1amXDjV922gPDnIISz/rQsT86
         gbjyC1jRuwsffz+POEnMcnTz7hqDUA/6BddfInantTi9uZj3gEna4TBjEgE6XHuChqkt
signature 2ME59WIVduUATG39M6OP+PVdoLZL8McLADsJZbeDTlDHOQXsLx/XY+XU42cEXOmtGbYE
        lcvoqHGYnd8CAWL1y/9MwfEZ/un8CmozF9mDaUmwZvxxHWqPgzzmKBL/Wez27jdD4YSH
        F9HMU3aafDodWv6tGNVJfsjqM0lod7ihC682WD5xgcl7NR7XGv1mq2E/NlvuoztxfIqc
        Mibq==
```

selector

```
$ dig TXT 20161025._domainkey.gmail.com

20161025._domainkey.gmail.com. 300 IN   TXT     "k=rsa; p=MIIBIjANBgkqhkiG9w0BAQEFAAOC
AQ8AMIIBCgKCAQEAviPGBk4ZB64UfSqWyAicdR7lodhytae+EYRQVtKDhM+1mXjEqRtP/pDT3sBhazkmA48n2k
5NJUyMEoO8nc2r6sUA+/Dom5jRBZp6qDKJOwjJ5R/OpHamlRG+YRJQqR" "tqEgSiJWG7h7efGYWmh4URhFM9k
9+rmG/CwCgwx7Et+c8OMlngaLl04/bPmfpjdEyLWyNimk761CX6KymzYiRDNz1MOJOJ7OzFaS4PFbVLn0m5mf0
HVNtBpPwWuCNvaFVflUYxEyblbB6h/oWOPGbzoSgtRA47SHV53SwZjIsVpbq4LxUW9IxAEwYzGcSgZ4n5Q8X8T
ndowsDUzoccPFGhdwIDAQAB"
```

domain key

But:

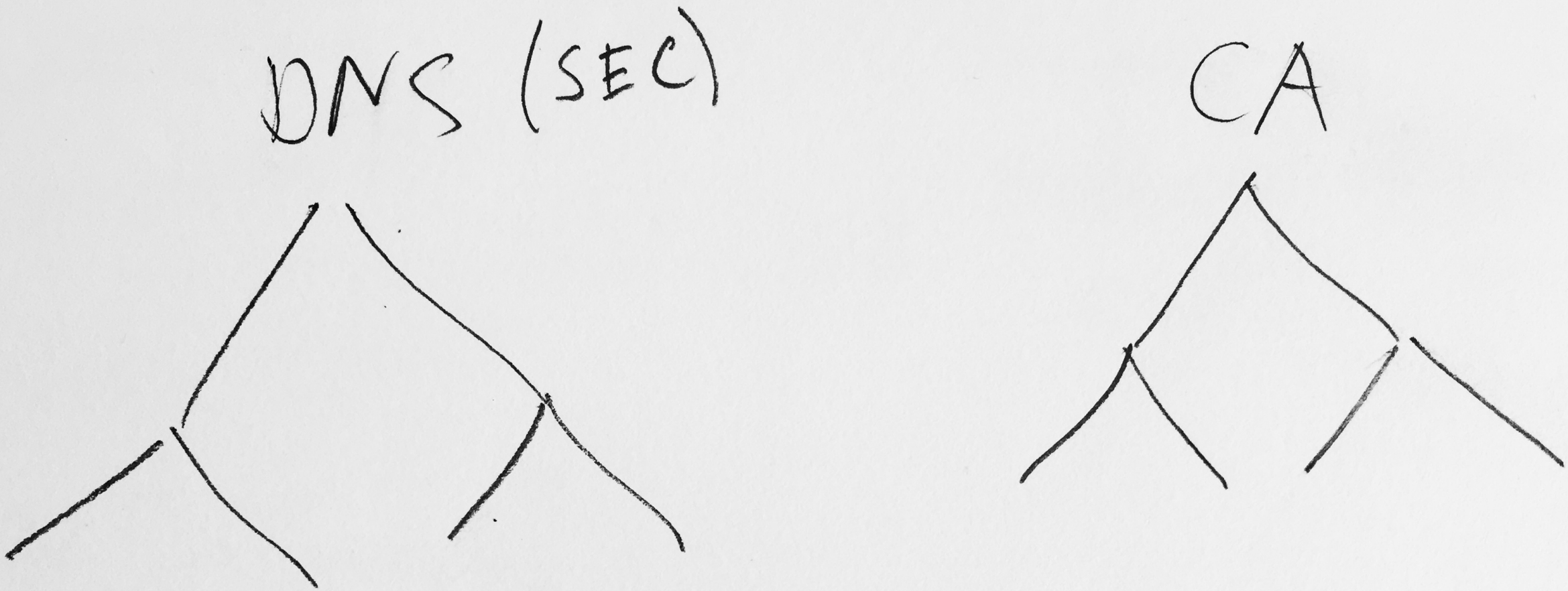
- Relies on the domain’s DNS record to provide the public key.

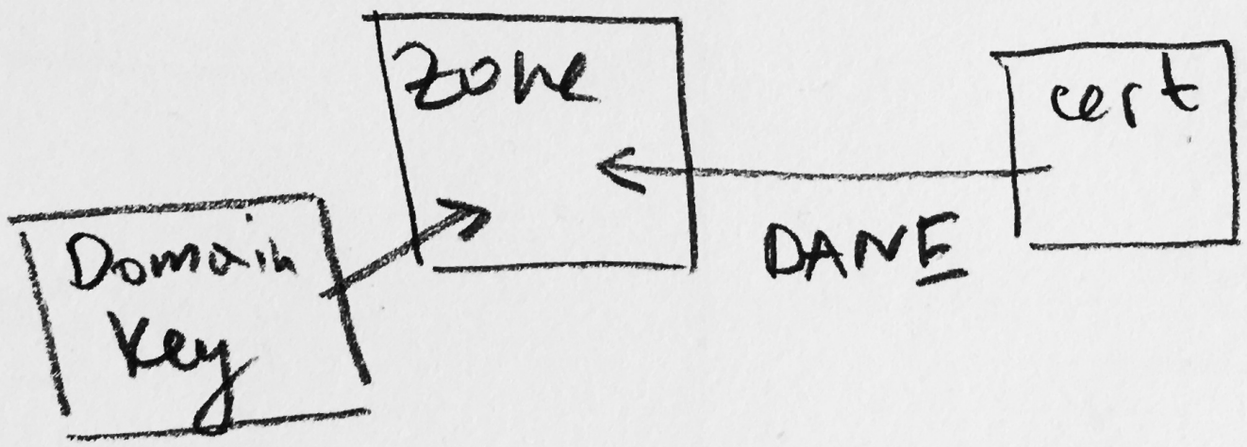
Server Authentication

- User authenticates to server?

- Or: server authenticates to user.

DNS, DNSSEC, Certificate Authorities



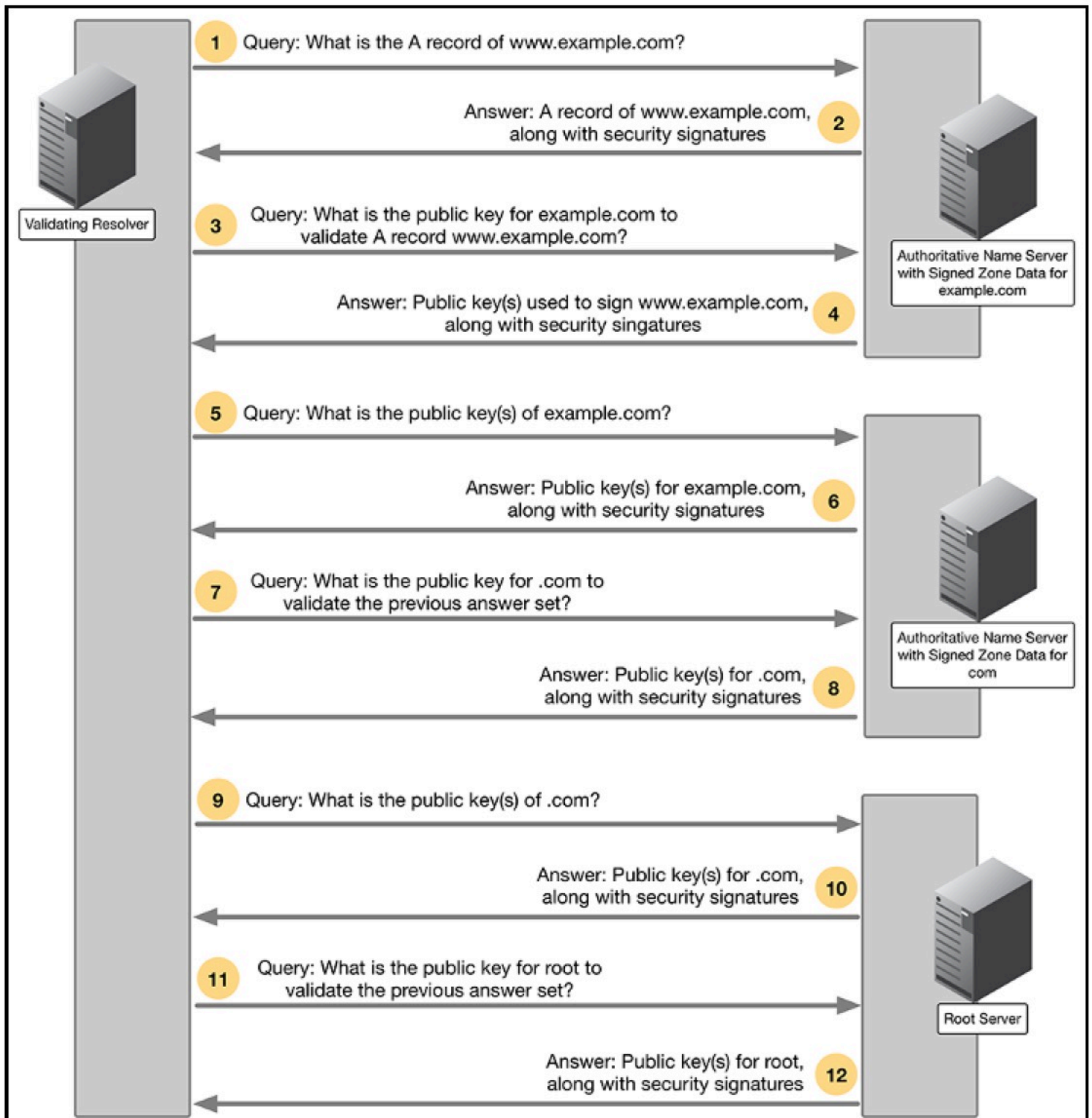


DNSSEC

- Each zone content is signed by parent DNS server.

- Lookup IP for example.co.il

- Verify against signature in DNS for .co.il


Validating Resolver
1
Query: What is the A record of www.example.com?
Answer: A record of www.example.com, along with security signatures
2
Authoritative Name Server with Signed Zone Data for example.com
3
Query: What is the public key for example.com to validate A record www.example.com?
Answer: Public key(s) used to sign www.example.com, along with security singatures
4
5
Query: What is the public key(s) of example.com?
Answer: Public key(s) for example.com, along with security signatures
6
Authoritative Name Server with Signed Zone Data for com
7
Query: What is the public key for .com to validate the previous answer set?
Answer: Public key(s) for .com, along with security signatures
8
9
Query: What is the public key(s) of .com?
Answer: Public key(s) for .com, along with security signatures
10
Root Server
11
Query: What is the public key for root to validate the previous answer set?
Answer: Public key(s) for root, along with security signatures
12

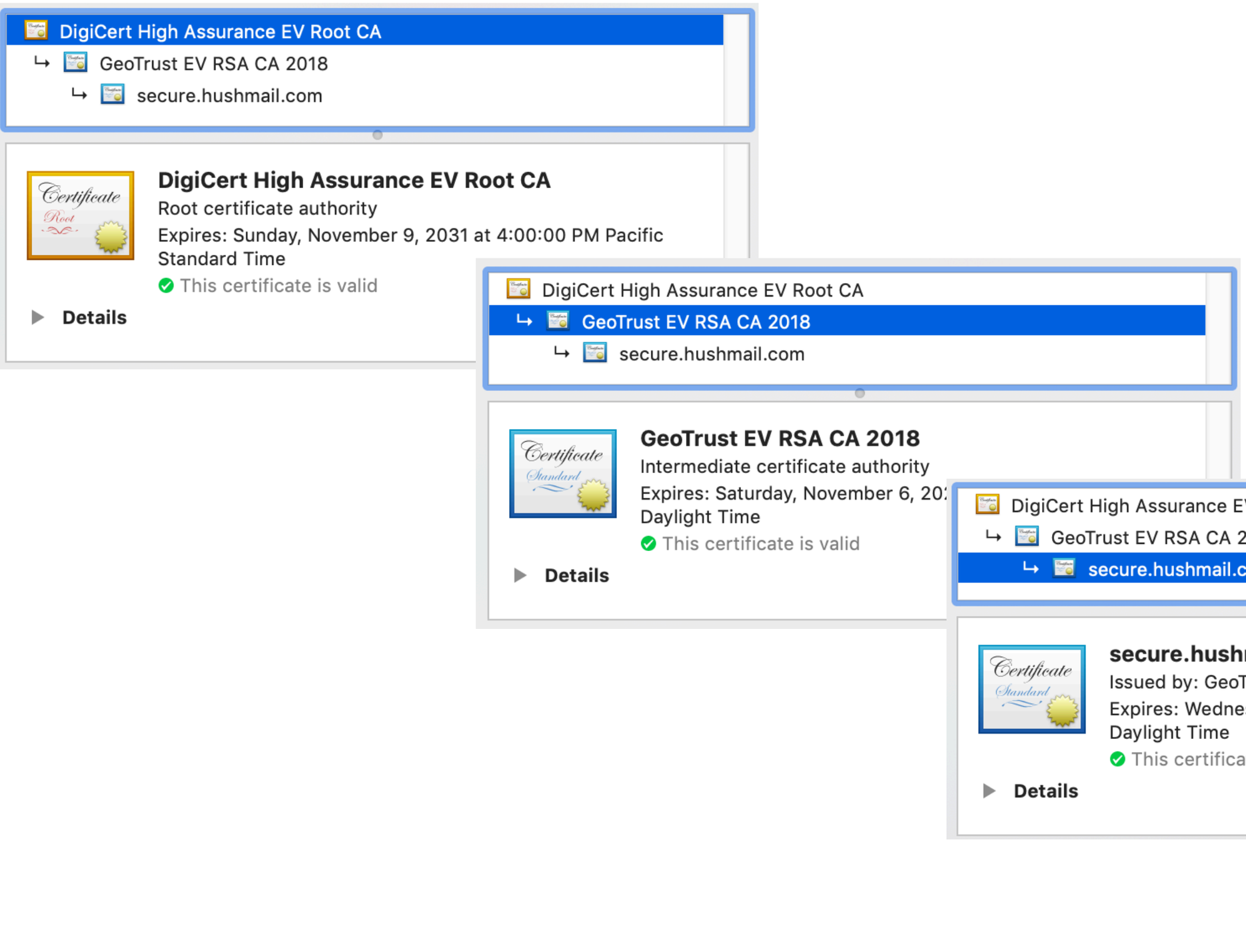
Certificate Authorities

- Certificates signed by a tree of certificate authorities.

- Example: VeriSign, Comodo

Chain of certificates

Trust is placed at root signer



List of available trusted root certificates in iOS 12, macOS 10.14, watchOS 5, and tvOS 12

Trusted certificates

Certificate name	Issued by	Type	Key size	Sig alg	Serial number	Expires	EV policy	Fingerprint (SHA-256)
AAA Certificate Services	AAA Certificate Services	RSA	2048 bits	SHA-1	01	23:59:59 Dec 31, 2028	Not EV	D7 A7 A0 FB 5D 7E 27 31 D7 71
StartCom Certification Authority	StartCom Certification Authority	RSA	4096 bits	SHA-1	01	19:46:36 Sep 17, 2036	1.3.6.1.4.1.23223.2 1.3.6.1.4.1.23223.1.1.1	C7 66 A9 BE F2 D4 07 1C 86 3A 31 AA
Izenpe.com	Izenpe.com	RSA	4096 bits	SHA-1	06 E8 46 27 2F 1F	08:27:25 Dec 13, 2037	1.3.6.1.4.1.14777.6.1.1 1.3.6.1.4.1.14777.6.1.2	23 80 42 03 CA 45 D8 CD E7
Government Root Certification Authority	Government Root Certification Authority	RSA	4096 bits	SHA-256	00 B6 4B 88 07 E2 23 EE C8 5C 12 AD A6 0E	15:59:59 Dec 31, 2037	Not EV	70 B9 22 BF DA 0 3F 4A 3 2E 4E E 2D 57 9 E5 98 D 71 CC 5

Trust govt of other countries

- CA, AT, AU, BE, BG, BM, BR, CA, CH, CL, CN, CO, CZ, DE, DK, EE, ES, EU, FI, FR, GB, HK, HUU, IE, IL, IN, IS, IT, JP, KR, LT, LV, MK, MO, MX, MY, NL, NO, PL PT, RO, RU, SE, SG, SI, SK, TR, TW, UK, US, UY, WW, ZA

- US. Department of Homeland Security

- US Defence Contractors

Scan CAs

The EFF SSL Observatory

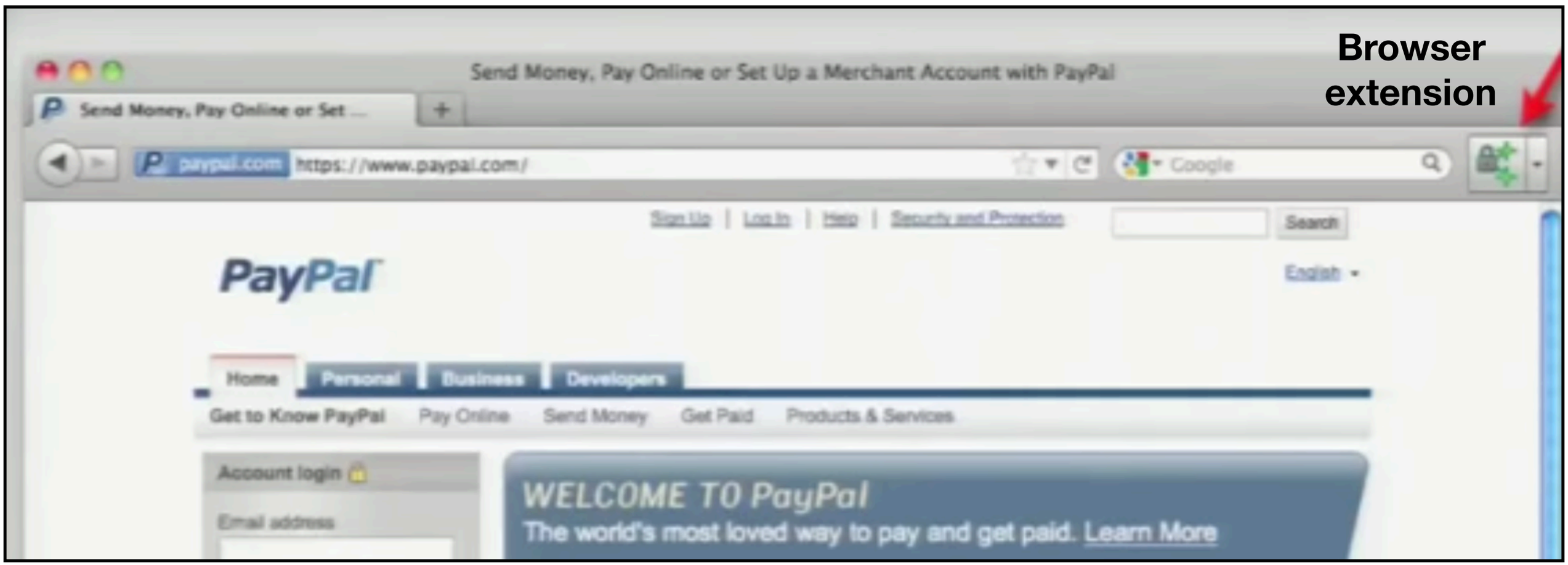
Download all publicly served SSL certificates from websites.

CA backdoors

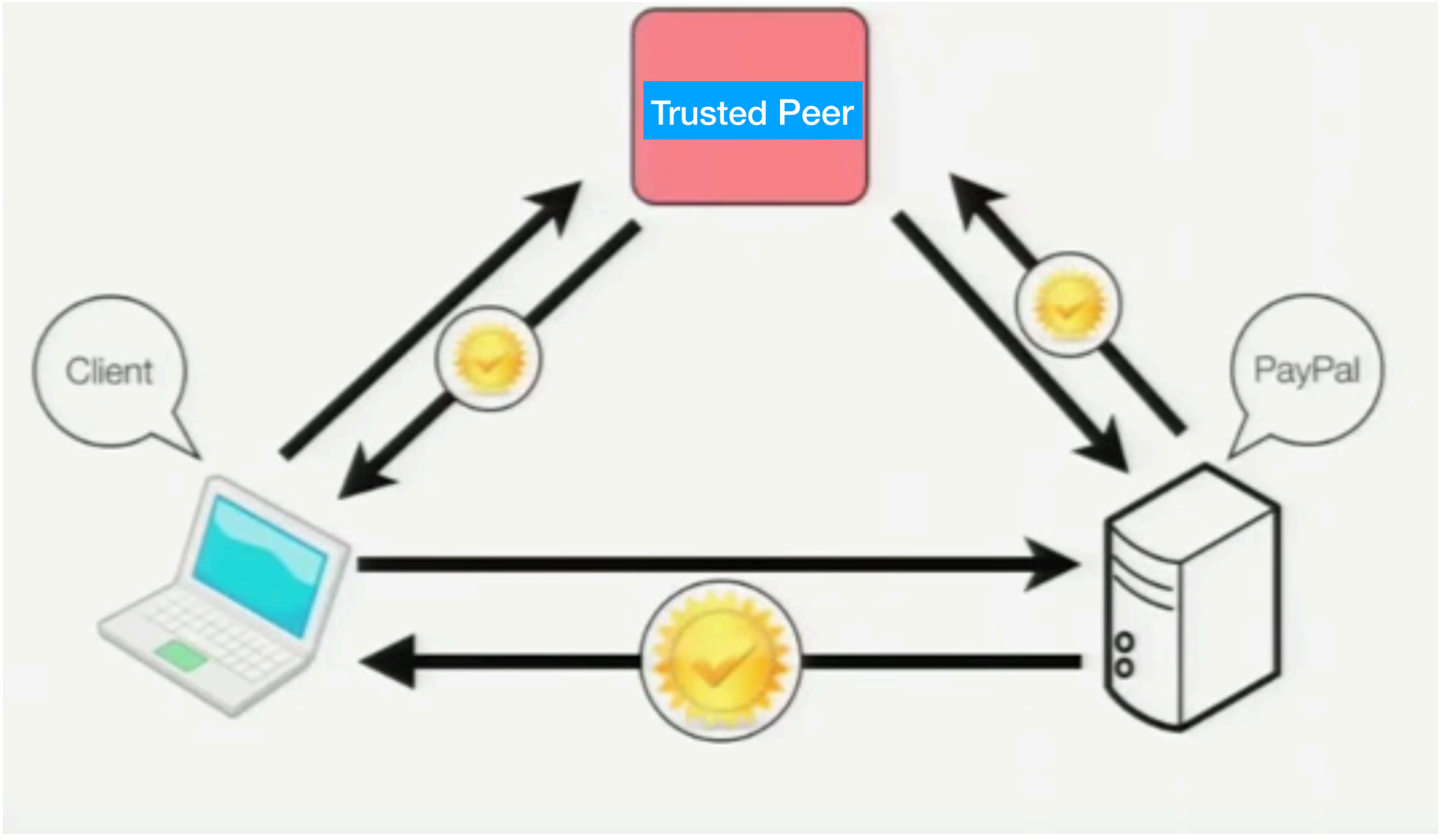
- 2009: 3 vulnerabilities due to CA mistakes
- 2010: evidence of govt compelling CAs
- 2012: Trustwave selling backdoor to private company.

Moxie's Solution

From 2013



Peer verifications



Authenticated Key Exchange

- PAKE: password authenticated key exchange
- SRP: Secure Remote Password protocol
- OPAQUE
- For symmetric keys.
- Can use for login as special case.

SRP

- Implemented in OpenSSL library

- Used by Apple for iCloud Key Vault.

SRP signup

hash function

private key

salt

password

$$x = H(s, p)$$

$$v = g^x$$

obfuscated key verifier

Server stores {s, v} per username

SRP Login

User generates secret “a”
Host generates secret “b”

```
User -> Host:  I, A = g^a
Host -> User:  s, B = kv + g^b
```

obfuscated secrets

public constant

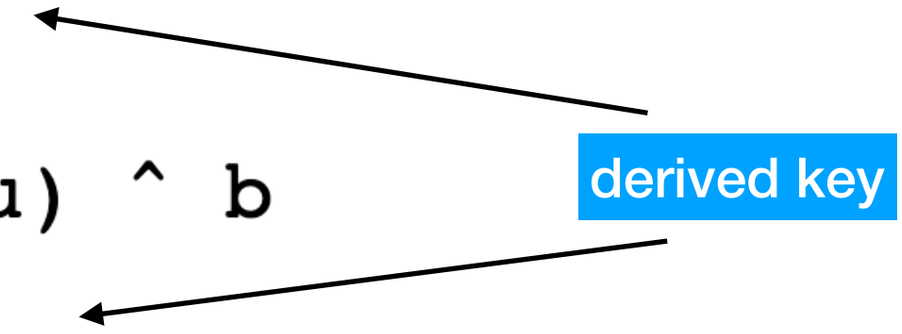
verifier

SRP Login math

```
Both:   u = H(A, B)

User:   x = H(s, p)
User:   S = (B - kg^x) ^ (a + ux)
User:   K = H(S)

Host:   S = (Av^u) ^ b
Host:   K = H(S)
```



Problem with SRP

- No security proof
- Is based on a ring, not a group.
- RSA ok
- not ECC
- already in version 4 after some vulnerabilities found.

OPAQUE

- Newer protocol, a replacement for SRP
- Can work in Curve25519 group.
- Hash to curve function.
- Elligator

OPAQUE in JavaScript

Protecting Passwords from Eavesdroppers

Boris Reitman

Jan 22 · 14 min read ★

This articles gets into technical security discussion of a new key-exchange protocol called OPAQUE.

OPAQUE protocol

RFC Draft

Crypto Forum Research Group — H. Krawczyk
Internet-Draft — IBM Research
Intended status: Informational — October 01, 2018
Expires: April 4, 2019

The OPAQUE Asymmetric PAKE Protocol
draft-krawczyk-cfrg-opaque-00

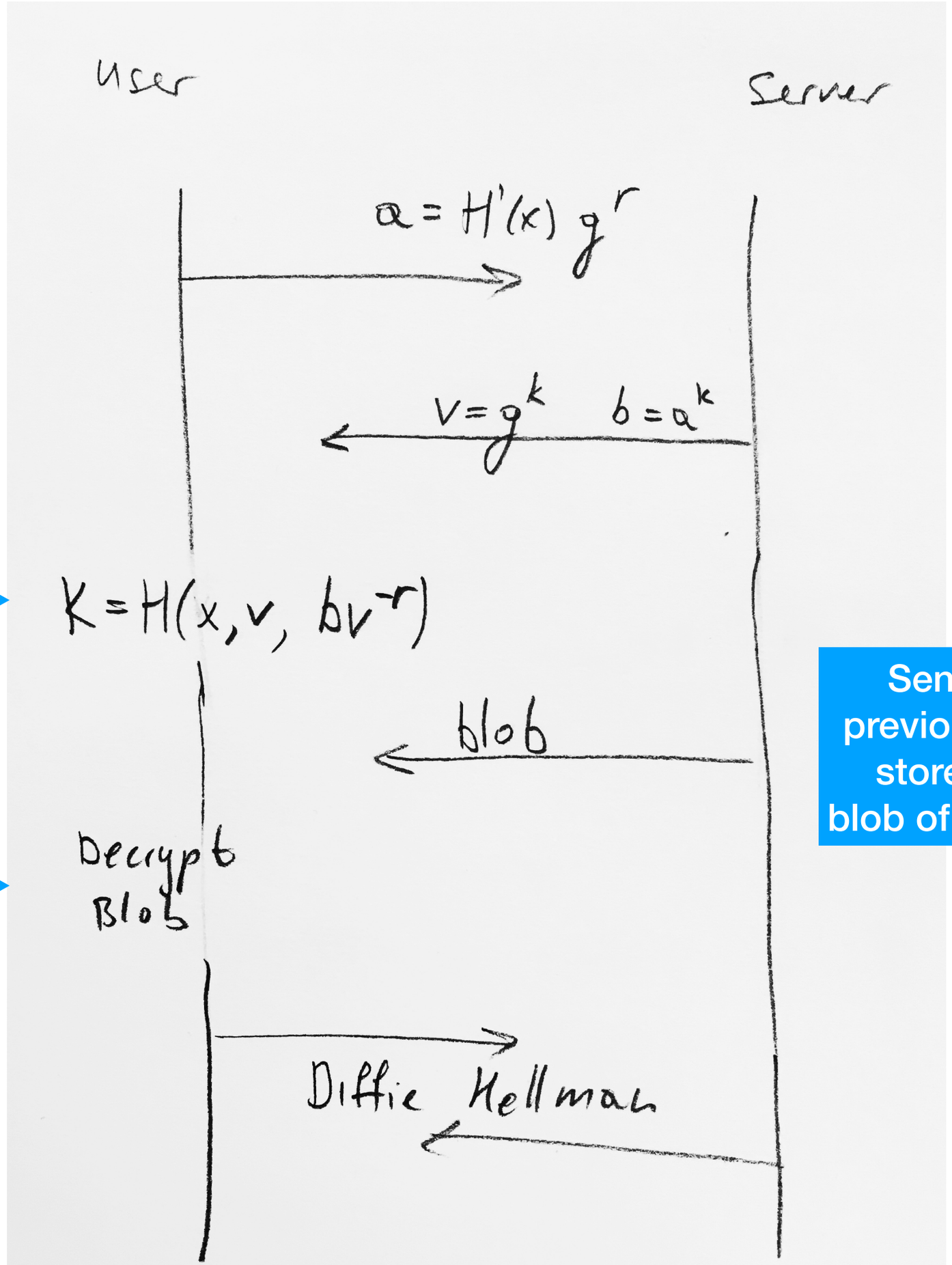
OPAQUE Registration

- User picks weak password “x”
- User generates salt “k”, stores it on the server.
- Generates symmetric key K = f(x,k)
- Blob = Encrypt(K; { server pub key, user pub key })
- Stores the blob on the server.

OPAQUE Login

- User generates a random value “r”
- Sends x^r to the server. (x is the password)
- Server responds with obfuscated “k” value.
- User constructs from this key K.
- Server also sends the encrypted blob.
- User decrypts it with K, and can authenticate the server.
- Also can authenticate himself to the server.

OPAQUE Key Exchange



Key Computed

Stuff user stored about legitimate server

Send previously stored blob of data

Conclusion

- borisreitman.com
- medium.com/@borisreitman
- Other philosophical meetups