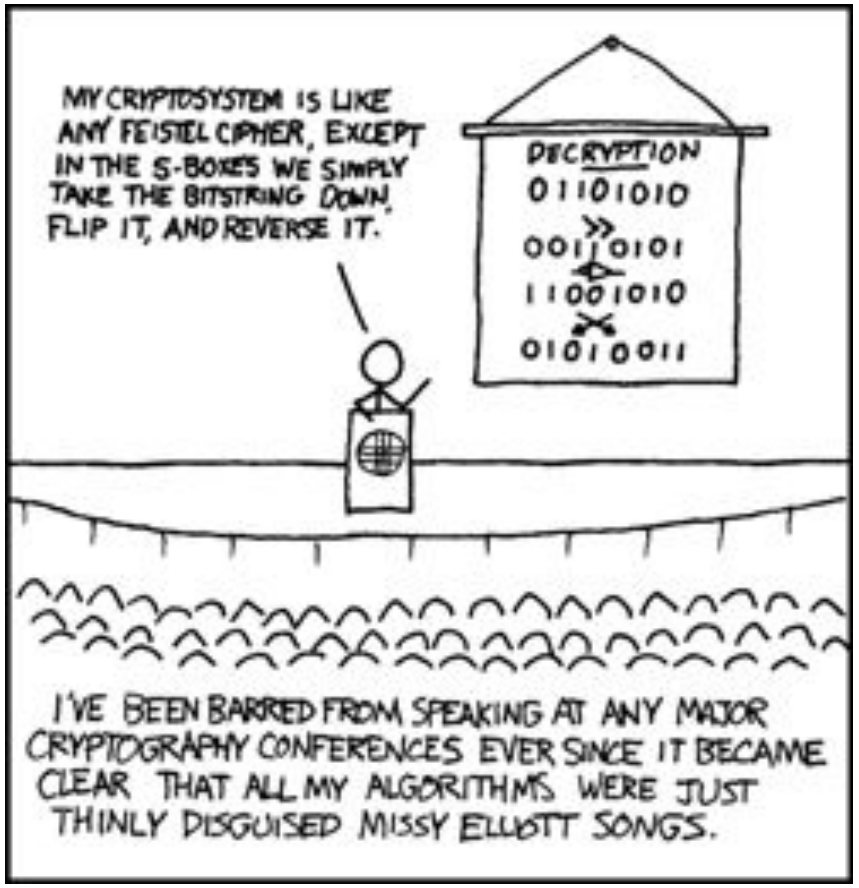
Cryptography Basics

About Me

- John Downey
- Senior Software Developer
- Housing and Food Services
- Microsoft/.NET by day
- Open Source by night
- Crypto Enthusiast
  - Applied Cryptography was on my Christmas list

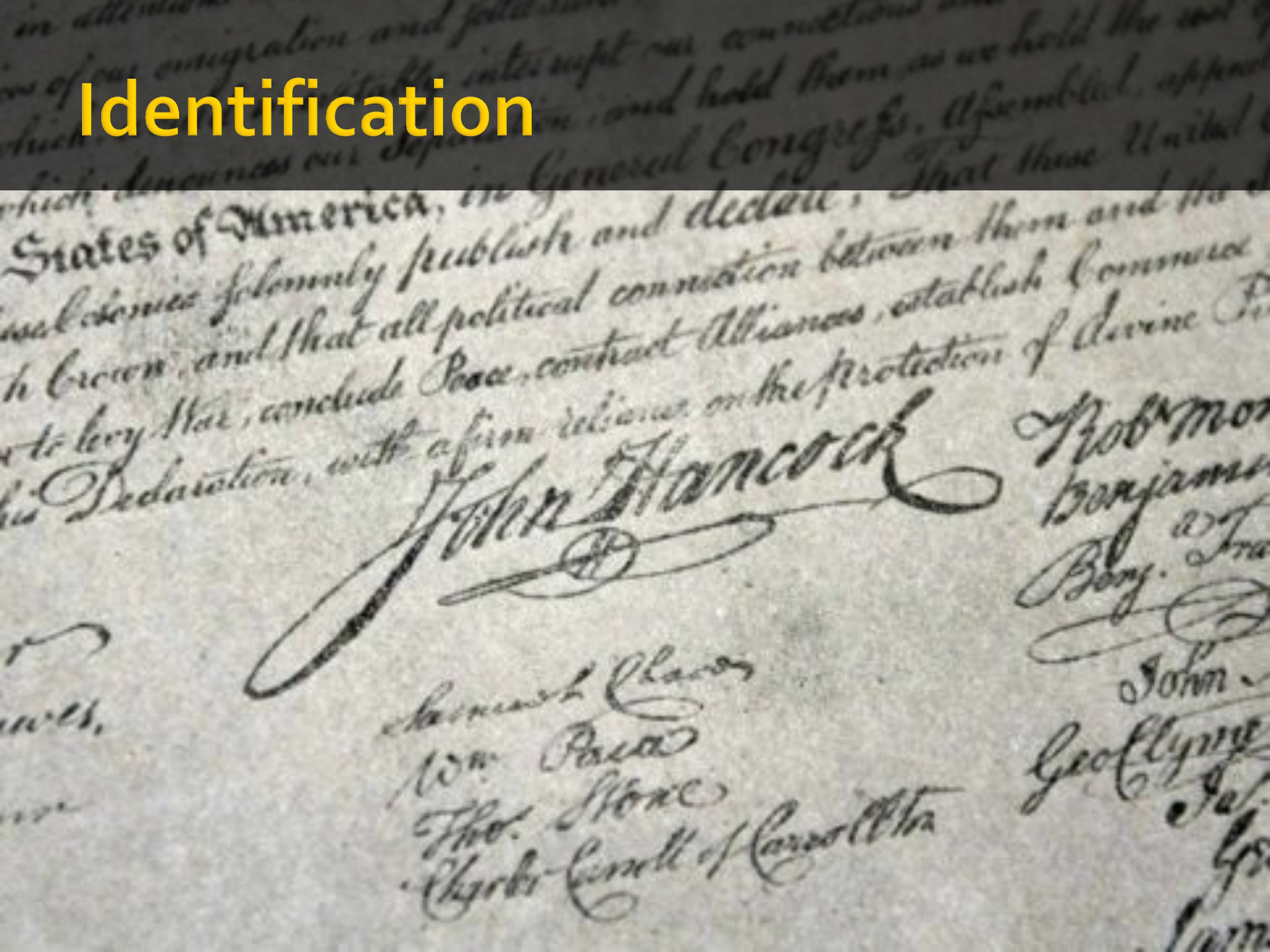


What is cryptography?

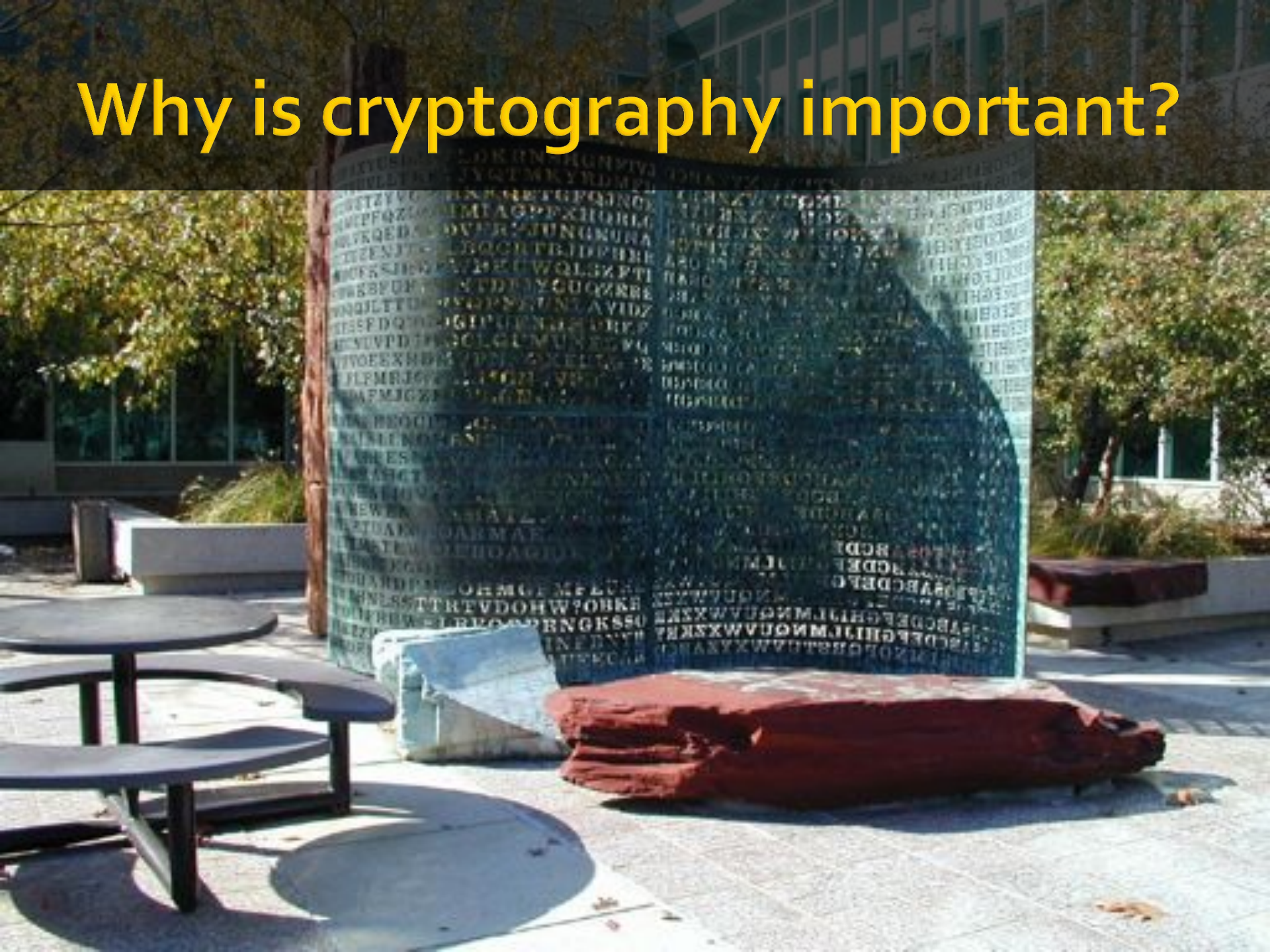
Encryption

Authentication

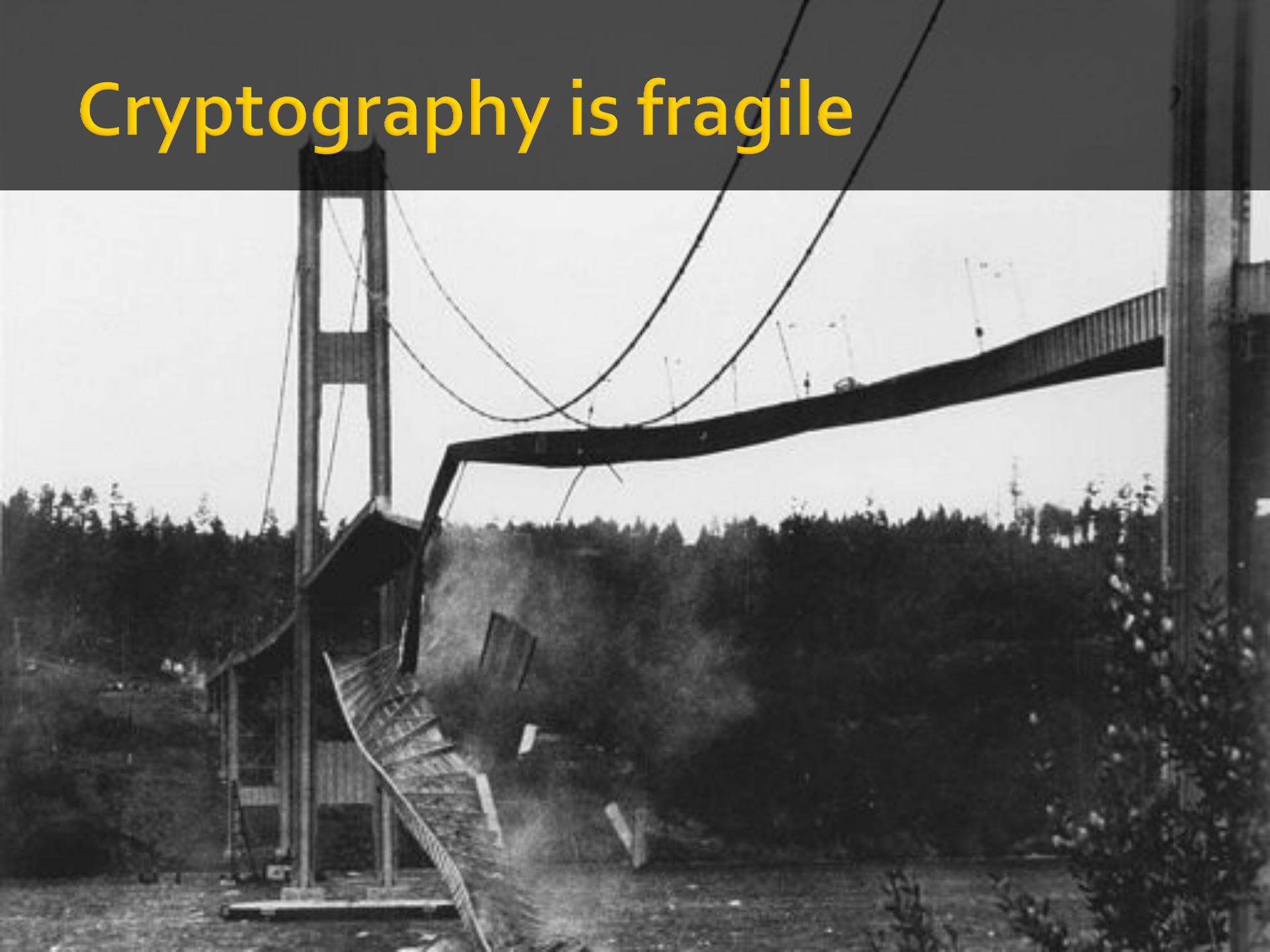
Identification



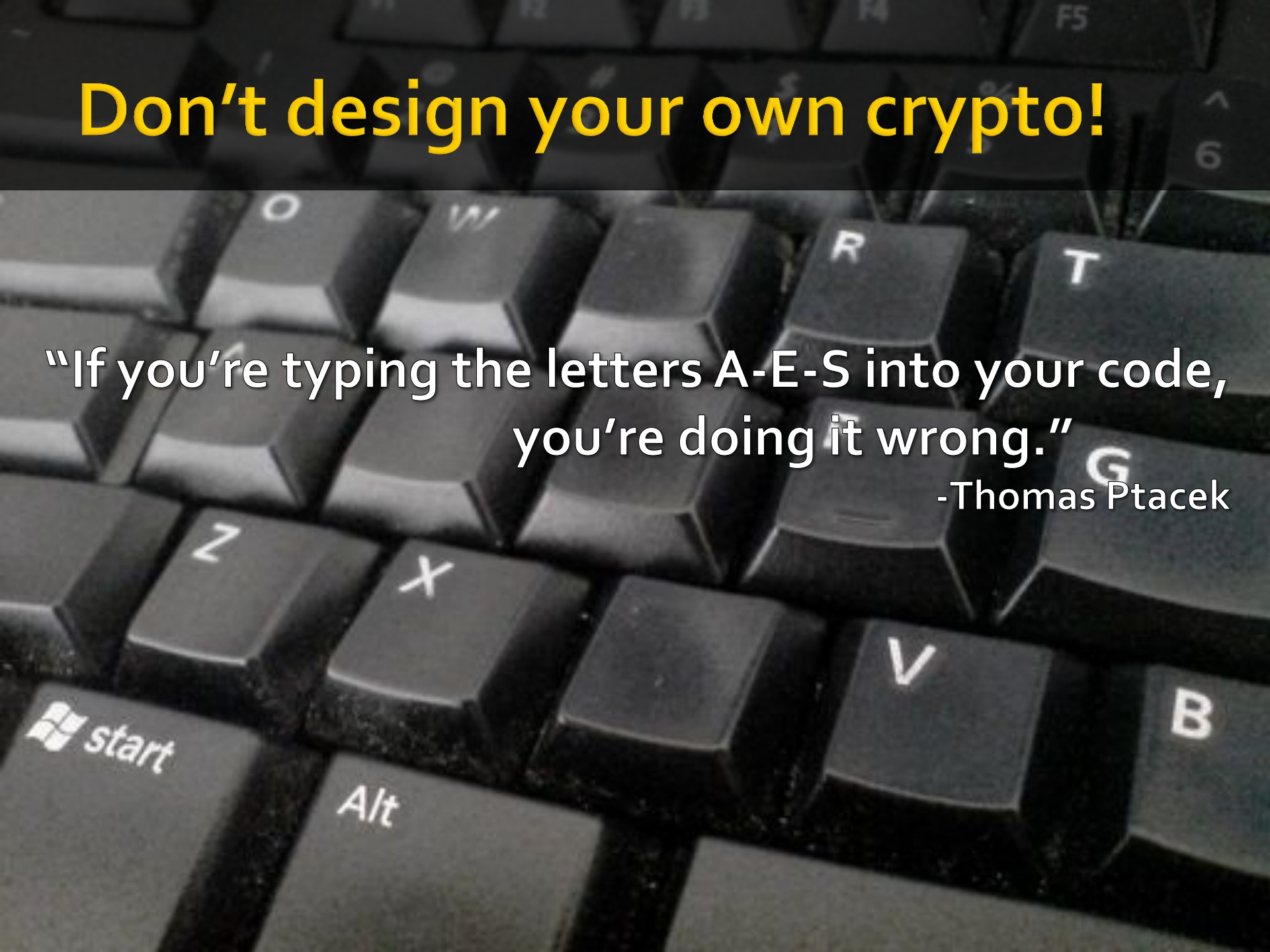
Why is cryptography important?



Cryptography is fragile



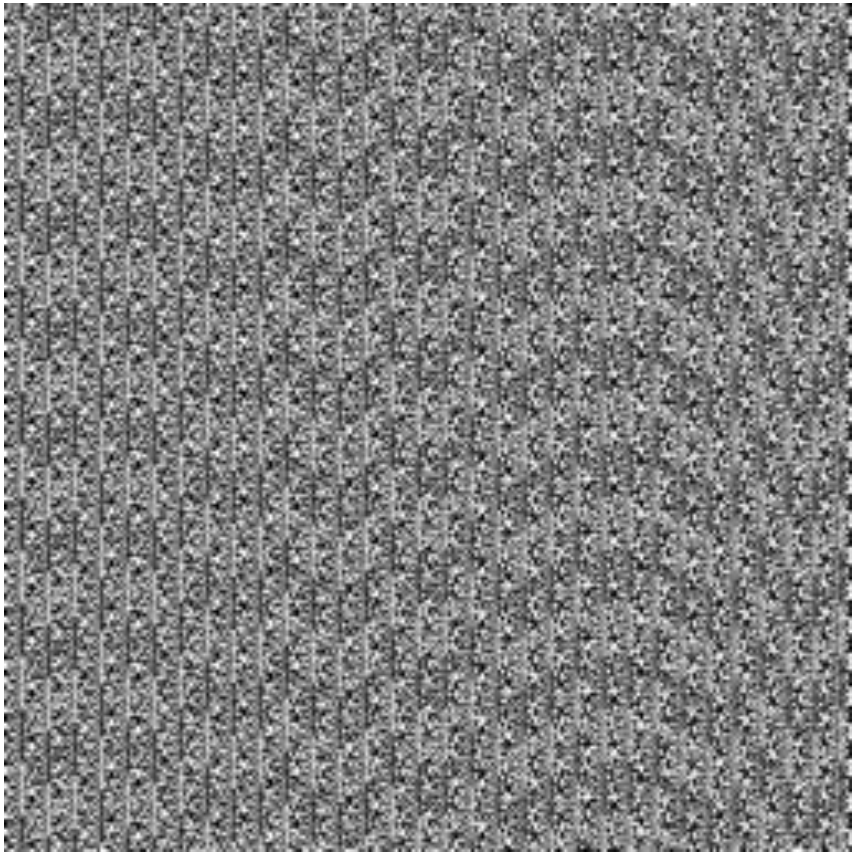
Don’t design your own crypto!

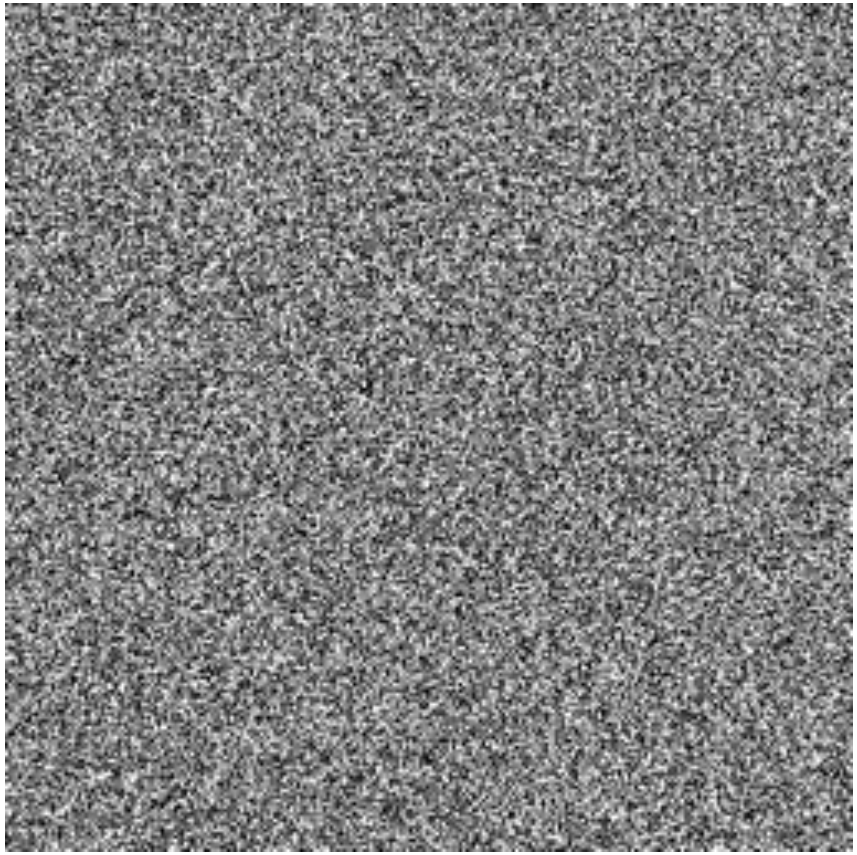


“If you’re typing the letters A-E-S into your code, you’re doing it wrong.”

-Thomas Ptacek

Random Number Generators





CVE-2008-0166

```
MD_Update(&m,buf,j);
```

Hash Function

USCYBERCOM plans, coordinates, integrates, synchronizes and conducts activities to: direct the operations and defense of specified Department of Defense information networks and; prepare to, and when directed, conduct full spectrum military cyberspace operations in order to enable actions in all domains, ensure US/Allied freedom of action in cyberspace and deny the same to our adversaries.

9EC4C12949A4F31474F299058CE2B22A

Hash Function

Hash Function

- Sometimes called a *message digest* or *fingerprint*
- Maps any length input to n-bit output
  - Collision resistant
  - One-way
- *H(x)* might allow you to derive *H(y)*
  - Length-extension attacks (MD0-5, SHA0-2)
  - If you know *H(x)* you can sometimes find *H(x || y)*

Hash Function

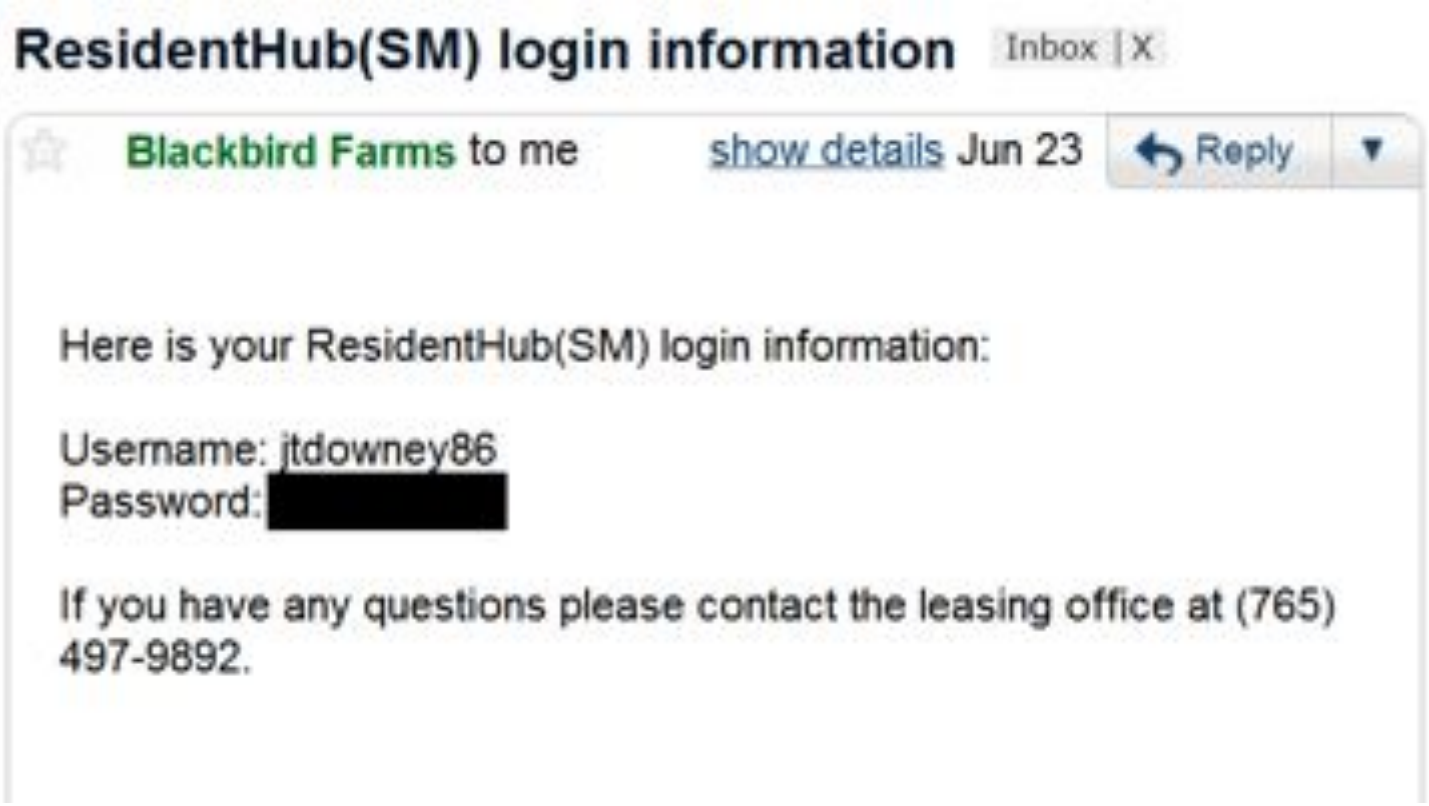
- DO: use SHA-256 (SHA-2)
- DO: switch to SHA-3 in 5-10 years
- AVOID: MD5
- DON'T: use MD2, MD4, SHA-0
- DON'T: use a hash as a signature
  - Still ok to use as a checksum

Hash Function

Name	Size	Date Modified
[parent directory]		
8.0-RELEASE-amd64-bootonly.iso	45.2 MB	11/20/09 11:00:00 PM
8.0-RELEASE-amd64-disc1.iso	658.3 MB	11/20/09 11:00:00 PM
8.0-RELEASE-amd64-dvd1.iso.gz	1.8 GB	11/20/09 11:00:00 PM
8.0-RELEASE-amd64-livefs.iso	321 MB	11/20/09 11:00:00 PM
8.0-RELEASE-amd64-memstick.img	996.3 MB	11/20/09 11:00:00 PM
CHECKSUM.MD5	351 B	11/20/09 11:00:00 PM
CHECKSUM.SHA256	526 B	11/20/09 11:00:00 PM

Passwords

Password Fail



ResidentHub(SM) login information Inbox | X

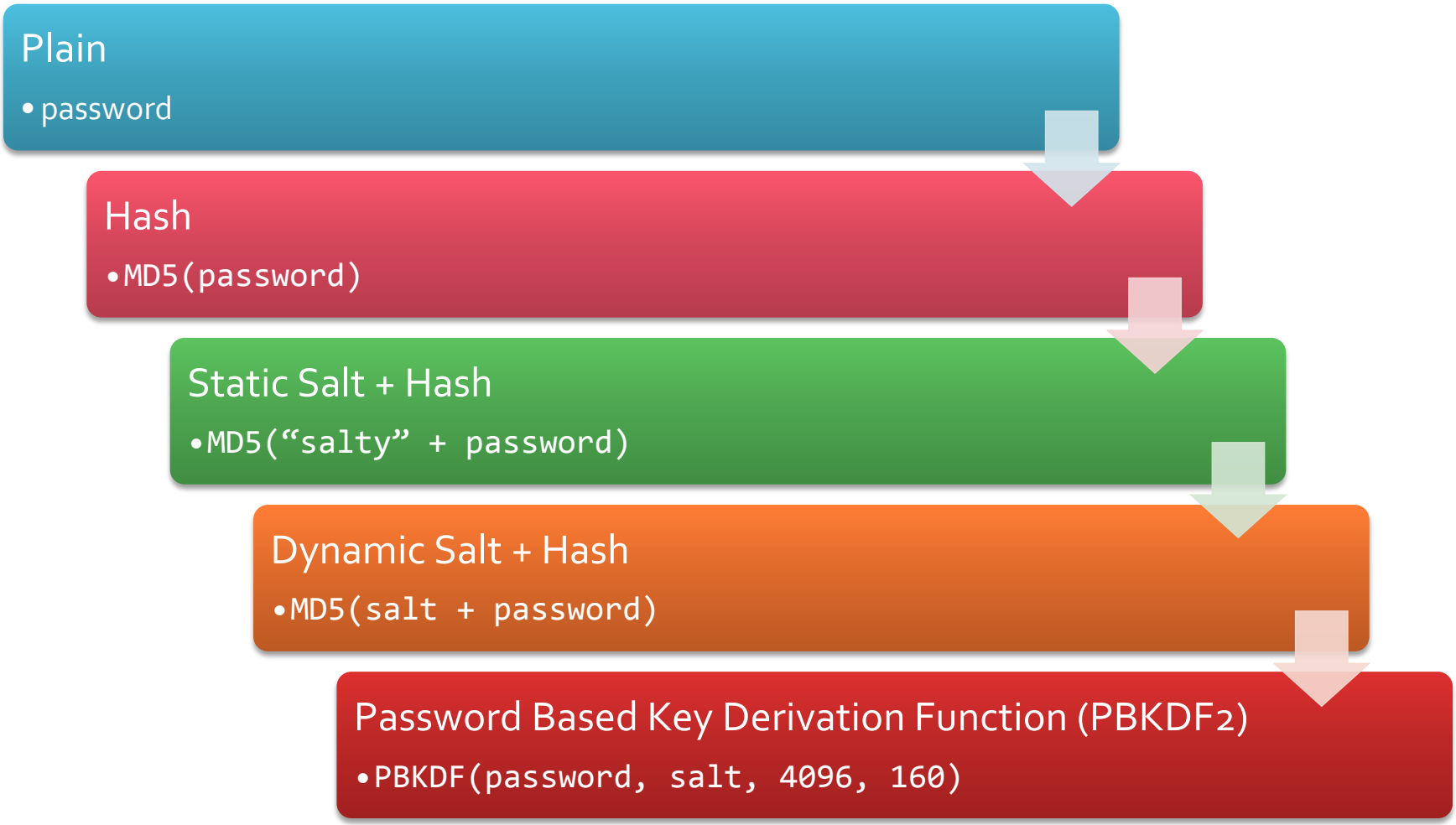
Blackbird Farms to me show details Jun 23 Reply

Here is your ResidentHub(SM) login information:

Username: jtdowney86
Password: [redacted]

If you have any questions please contact the leasing office at (765) 497-9892.

Password Storage on the Web



Key Derivation Function

- Often called key strengthen or stretching
- Performs many iterations
  - Intentionally slow to deter brute forcing
  - Usually over 1000
- Keys can be used for many inputs
  - Symmetric cipher keys
  - Symmetric authentication keys
  - Password storage

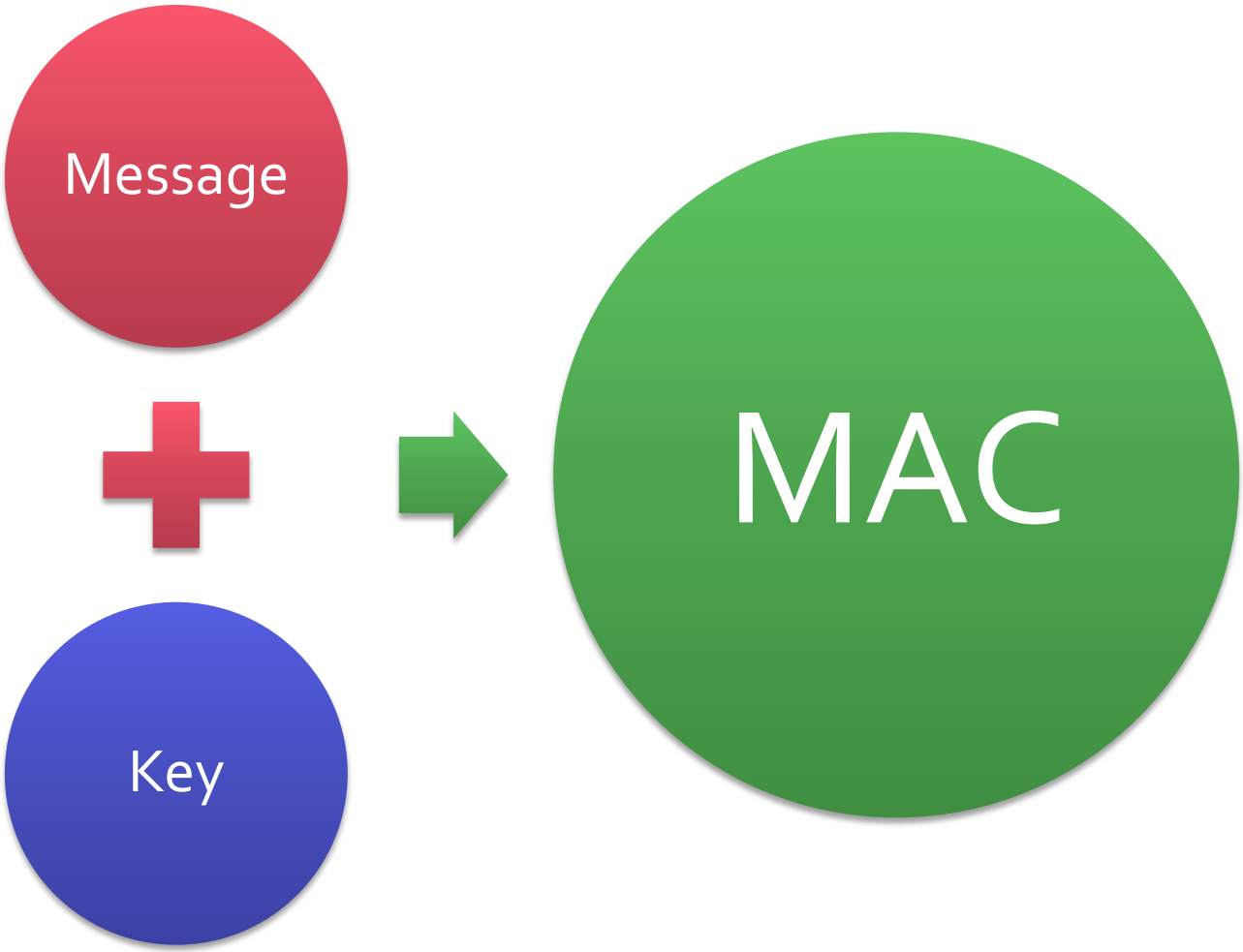
Passwords

- Delegate authentication if possible
  - CAS
  - Kerberos
- Change passwords into a key as soon as possible
  - Using PBDKF2, bcrypt, scrypt, etc
- DON'T: store passwords on your server
  - Even if it is encrypted or encoded
  - Store one-way keys or verifiers

Secure Remote Password Protocol

Symmetric Authentication

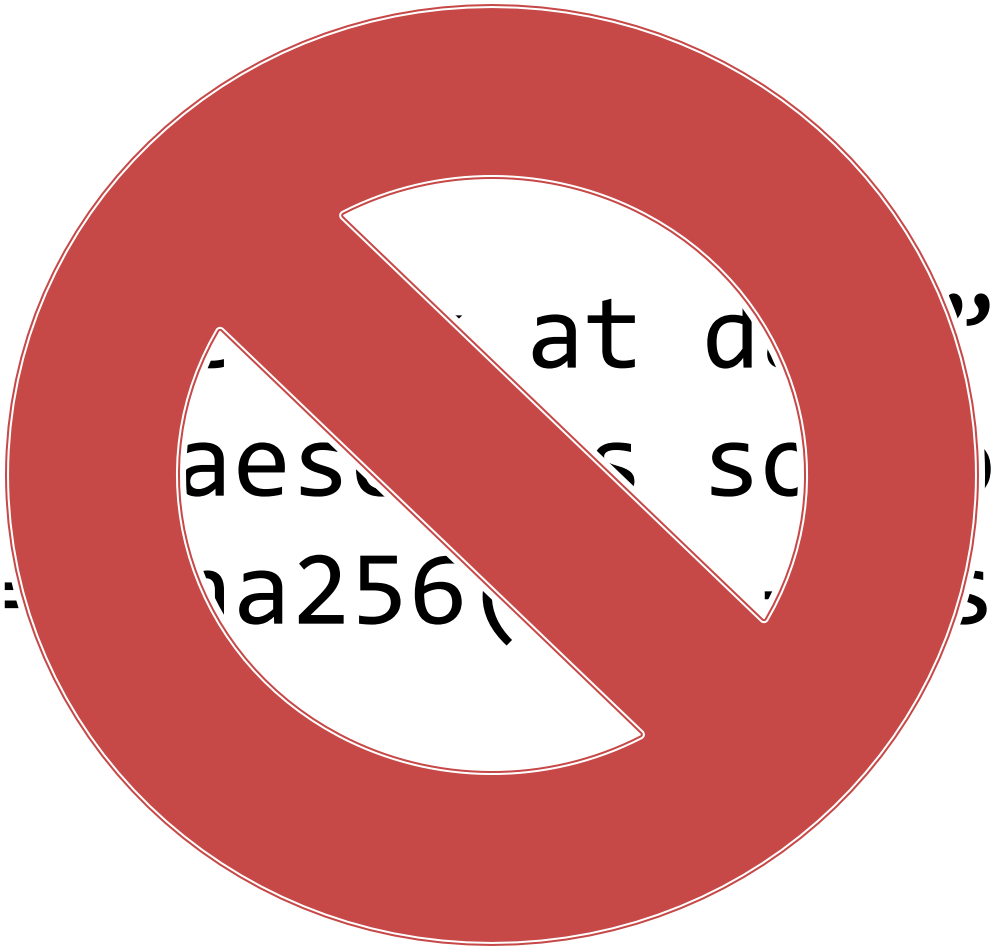
Symmetric Authentication



Symmetric Authentication

```
msg = “Attack at dawn”
key = “Caeser is so cool”
hash = sha256(key + msg)
```

Symmetric Authentication



Symmetric Authentication

- Maps any length input to n-bit output using a key
- $M_k(x)$ doesn't allow you to derive $M_k(y)$
  - No length extension attack
- Flickr API used a hash to authenticate API requests where they should have used a MAC
  - TouchNet also makes this mistake

Symmetric Authentication

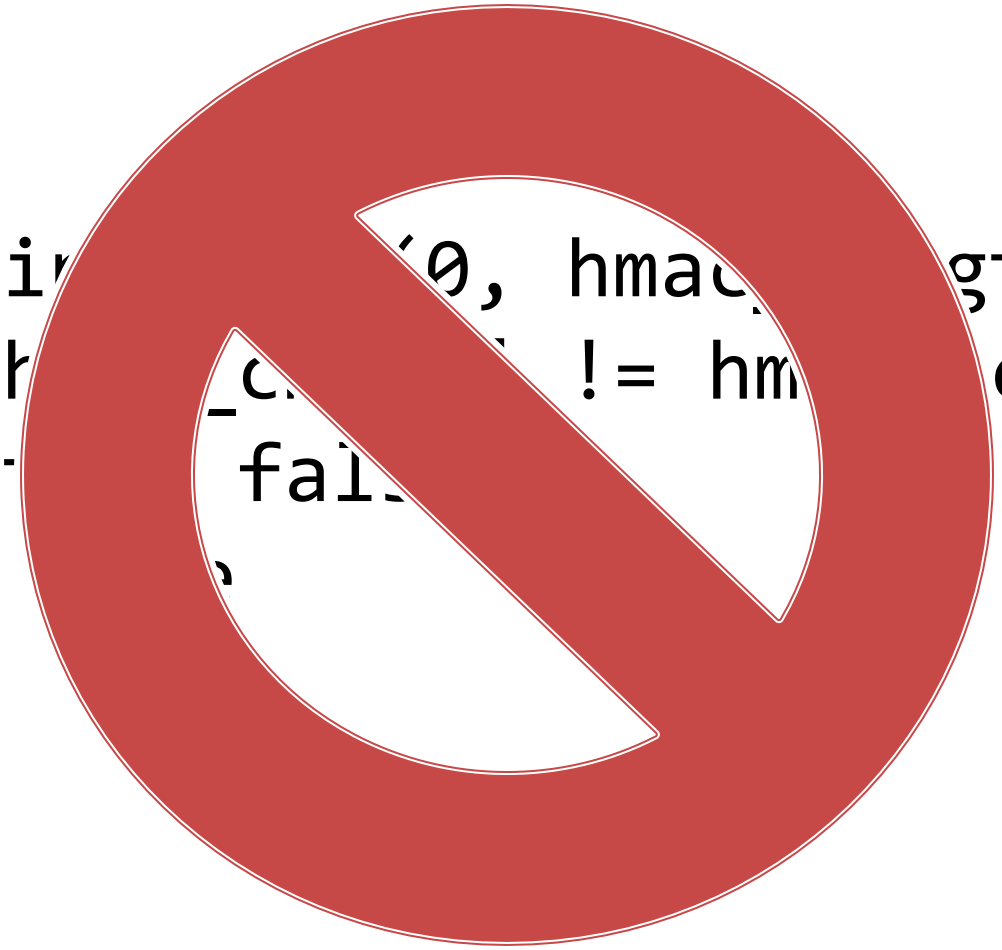
- DO: use HMAC-SHA256
- DO: keep your data structured
  - Amazon got this wrong
    - key1=value1&key2=value2 →key1value1key2value2
- DO: verify the entire signature
  - Nintendo Wii got this wrong
- DON'T: leak information via timing side channels when you verify a signature
  - Many OpenID implementations have this wrong

Symmetric Authentication

```
for i in range(0, hmac_length)
  if (hmac_cmp[i] != hmac_rcv[i])
    return false
return true
```

Symmetric Authentication

```
for i ir      0, hmac      gth)
   if (h     _c      != hm      cv[i])
      re      fal
return      e
```



Symmetric Authentication

```
x = 0
for i in range(0, hmac_length)
  x |= hmac_cmp[i] xor hmac_rcv[i]
return x == 0
```

Stream Cipher

Stream Cipher

- Maps n-bit input to n-bit output using a key
  - Usually works by combining message with an n-bit keystream (using XOR)
- Key re-use is fatal
  - $E_1 = msg_1$ XOR keystream
  - $E_2 = msg_2$ XOR keystream
  - E_1 XOR $E_2 = msg_1$ XOR msg_2
  - What if I know one of the messages?

Malleability

```
8b 93 d3 36 29 bb be 78 13 2e 49 97
3f 9b a0 52 7f 37 3c 09 d4 2b 82 14
e2 08 7a 5b 20 70 a4 7b de 8e bc
```

Malleability

<to>501249</to><amount>150</amount>

Malleability

```
8b 93 d3 36 29 bb be 78 13 2e 49 97
3f 9b a0 52 7f 37 3c 09 d4 2b 82 14
e2 08 7a 5b 20 70 a4 7b de 8e bc
```

Malleability

```
00 00 00 00 07 03 05 01 01 00 00 00
00 00 00 00 00 00 00 00 00 00 00 08
0c 09 00 00 00 00 00 00 00 00 00
```

Malleability

<to>234359</to><amount>999</amount>

Stream Cipher

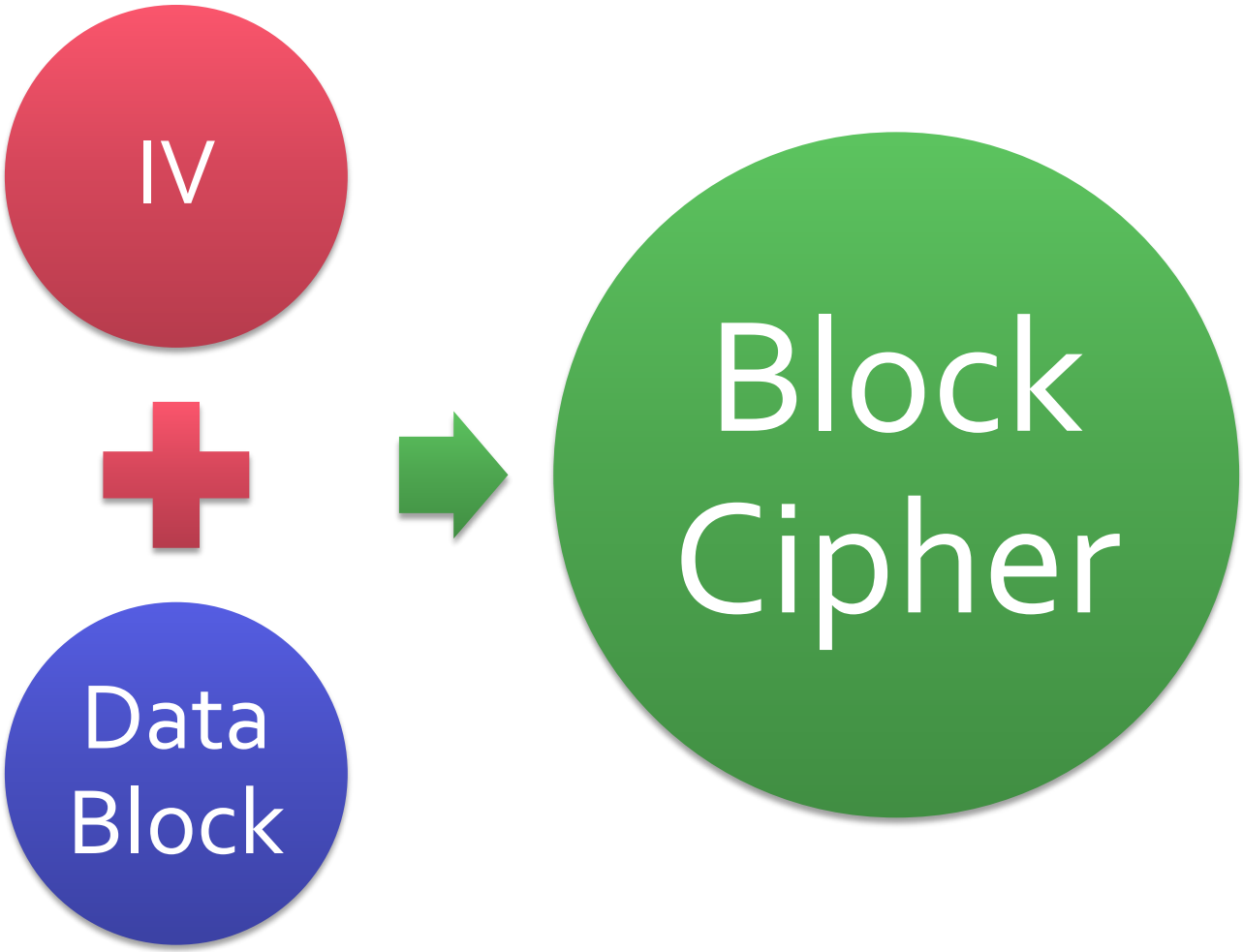
- DO: use ciphers in the eSTREAM portfolio
  - European contest similar to AES process
  - Hardware and Software profiles
- DON'T: use RC4
- DO: use a MAC (i.e., HMAC-SHA256) to authenticate your encrypted data
- DO: verify the authenticity of your encrypted data before you decrypt it

Block Cipher

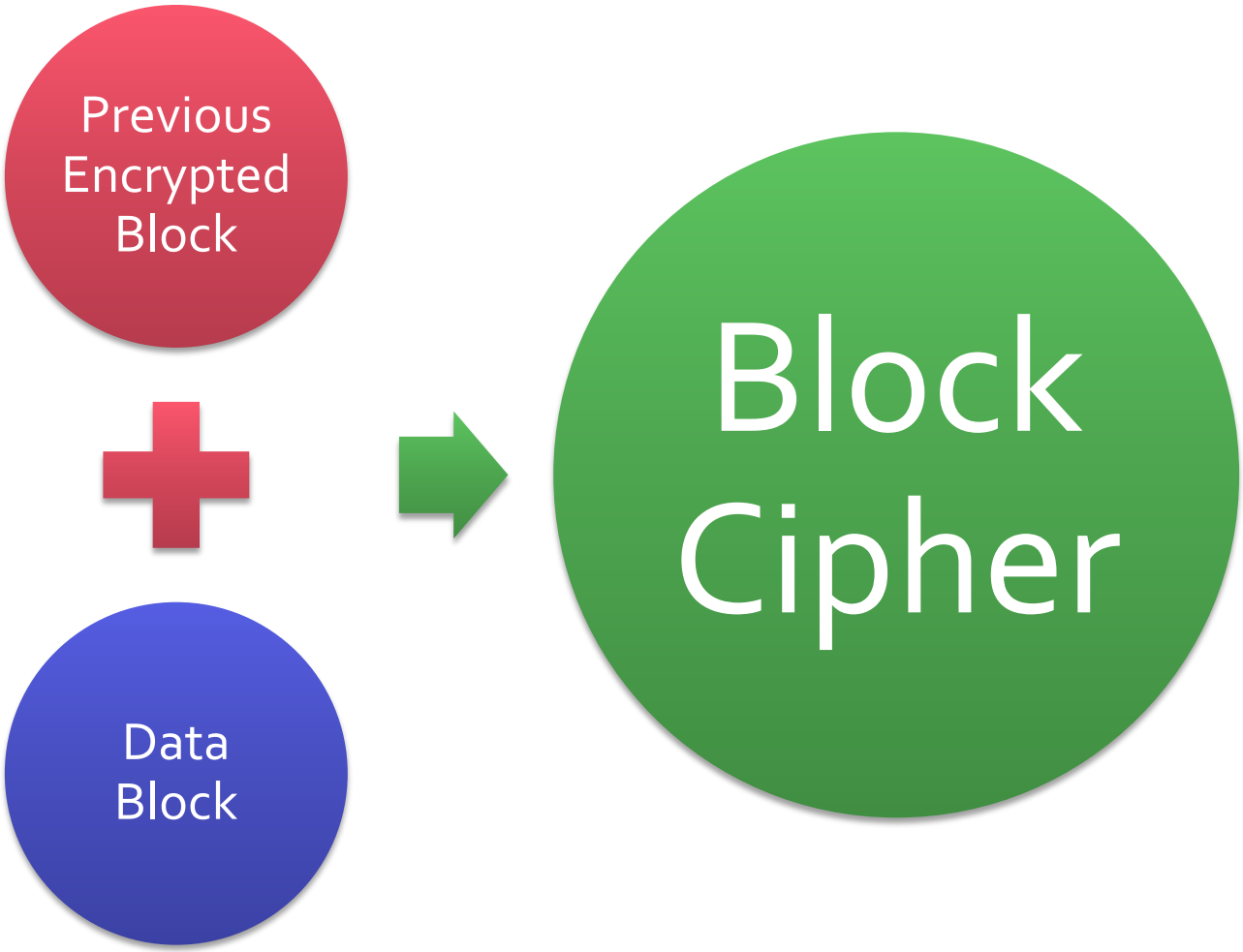
Block Cipher

- Maps n-bit input to n-bit output using a key
  - Works in block steps (AES block = 128-bit)
- Longer data needs to use a cipher mode
  - Electronic Code Book (ECB)
  - Cipher Block Chaining (CBC)
  - Counter Mode (CTR)
  - Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (CCMP)

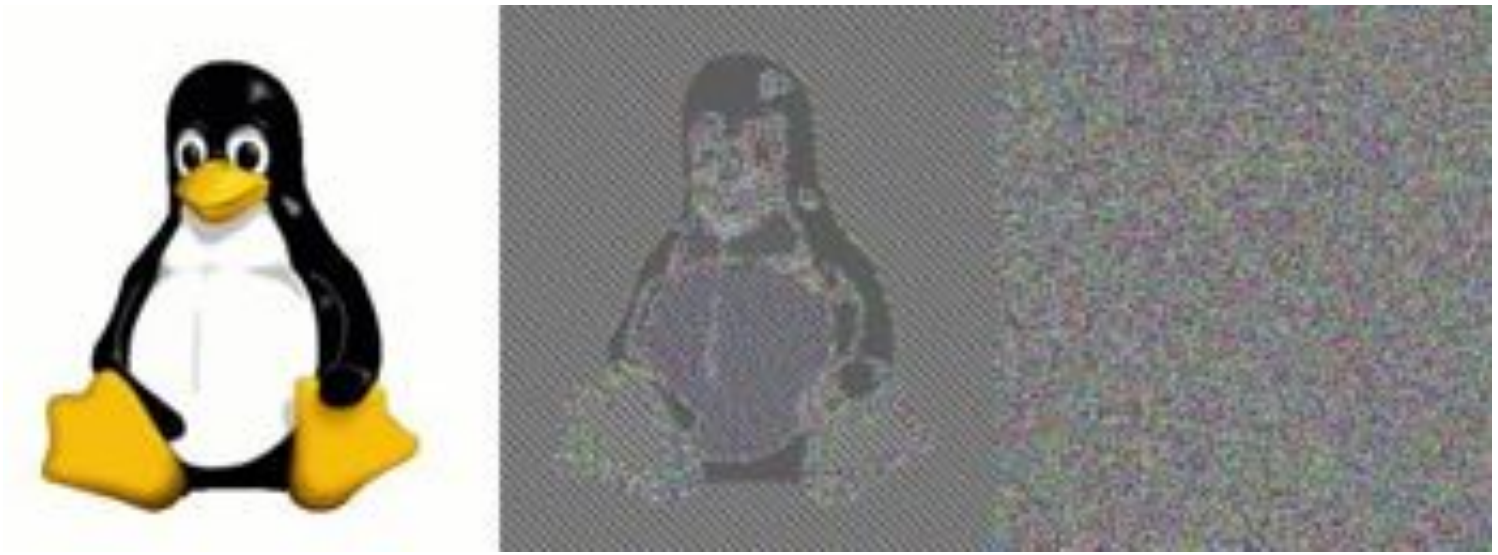
Cipher Block Chaining



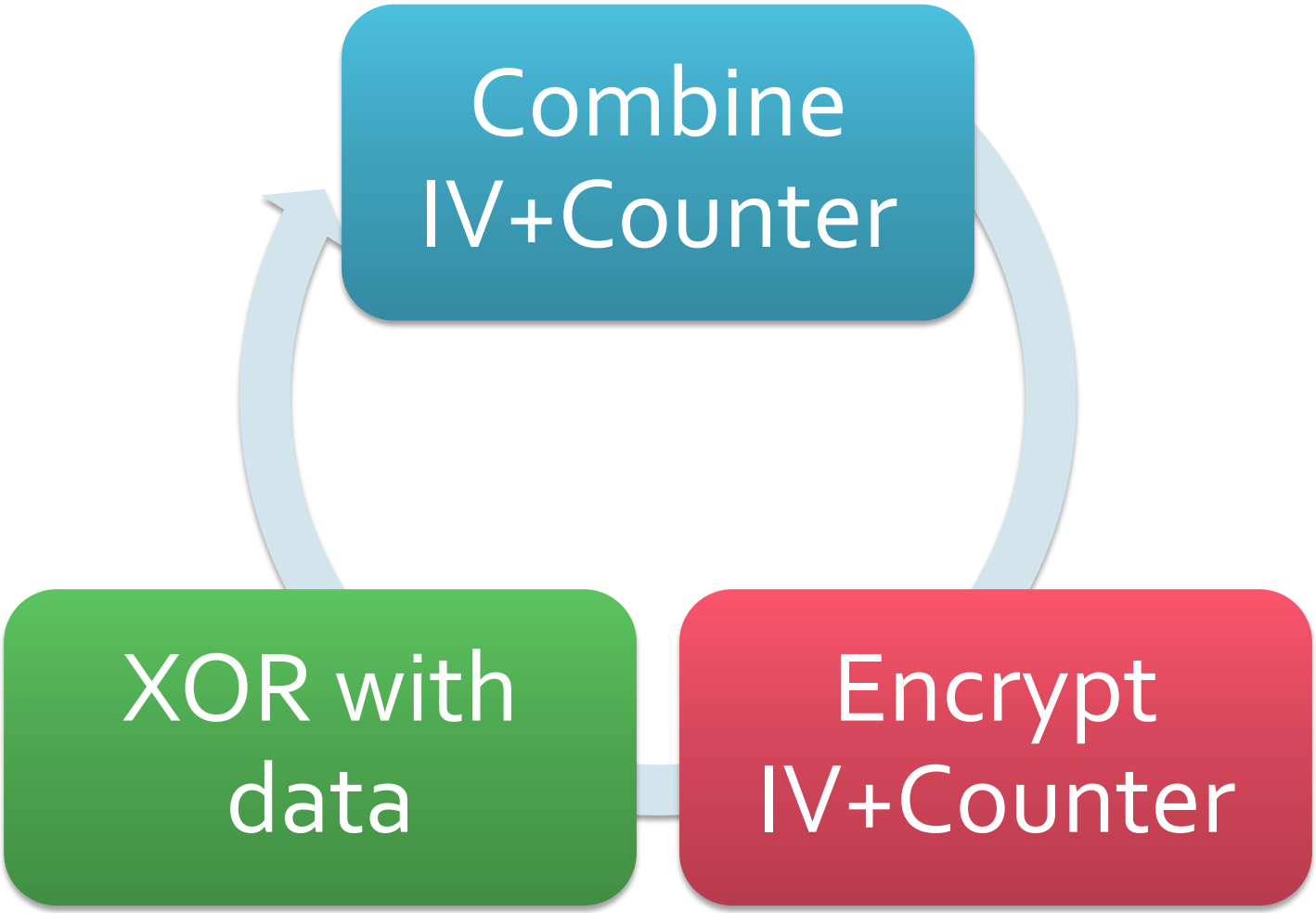
Cipher Block Chaining



ECB vs CBC mode



Counter Mode



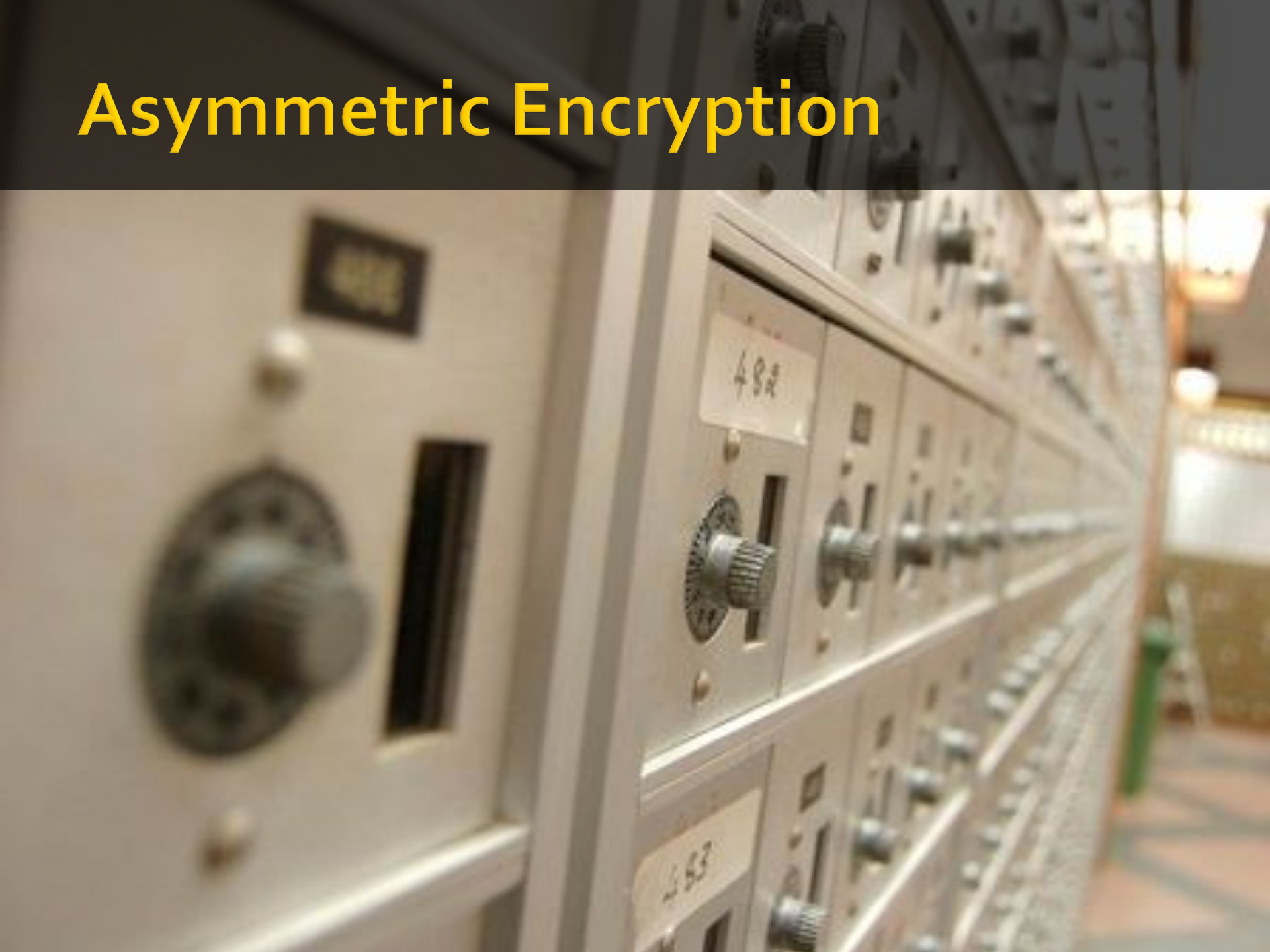
Block Cipher

- DO: use AES-256
- AVOID: using Blowfish and TripleDES
- STOP: using DES
- DO: use a MAC to authenticate your encrypted data
- DO: verify the authenticity of your encrypted data before you decrypt it
- DON'T: use a block cipher without a cipher mode

Block Cipher Modes

- DO: use CTR mode
  - Keep in mind stream cipher caveats
  - Unless the key-size is <= 64 bit
- DO: use CBC mode
  - Prefer CTR mode though
- DON'T: use ECB mode
- DON'T EVER: reuse initialization values (IVs)

Asymmetric Encryption



Asymmetric Encryption

- An *encrypting key* (public key) transforms plaintext into cipher text
- A *decryption key* (private key) transforms cipher text into plaintext
- Typically a shared key is encrypted
  - Later used with a block or stream cipher

Asymmetric Authentication

Asymmetric Authentication

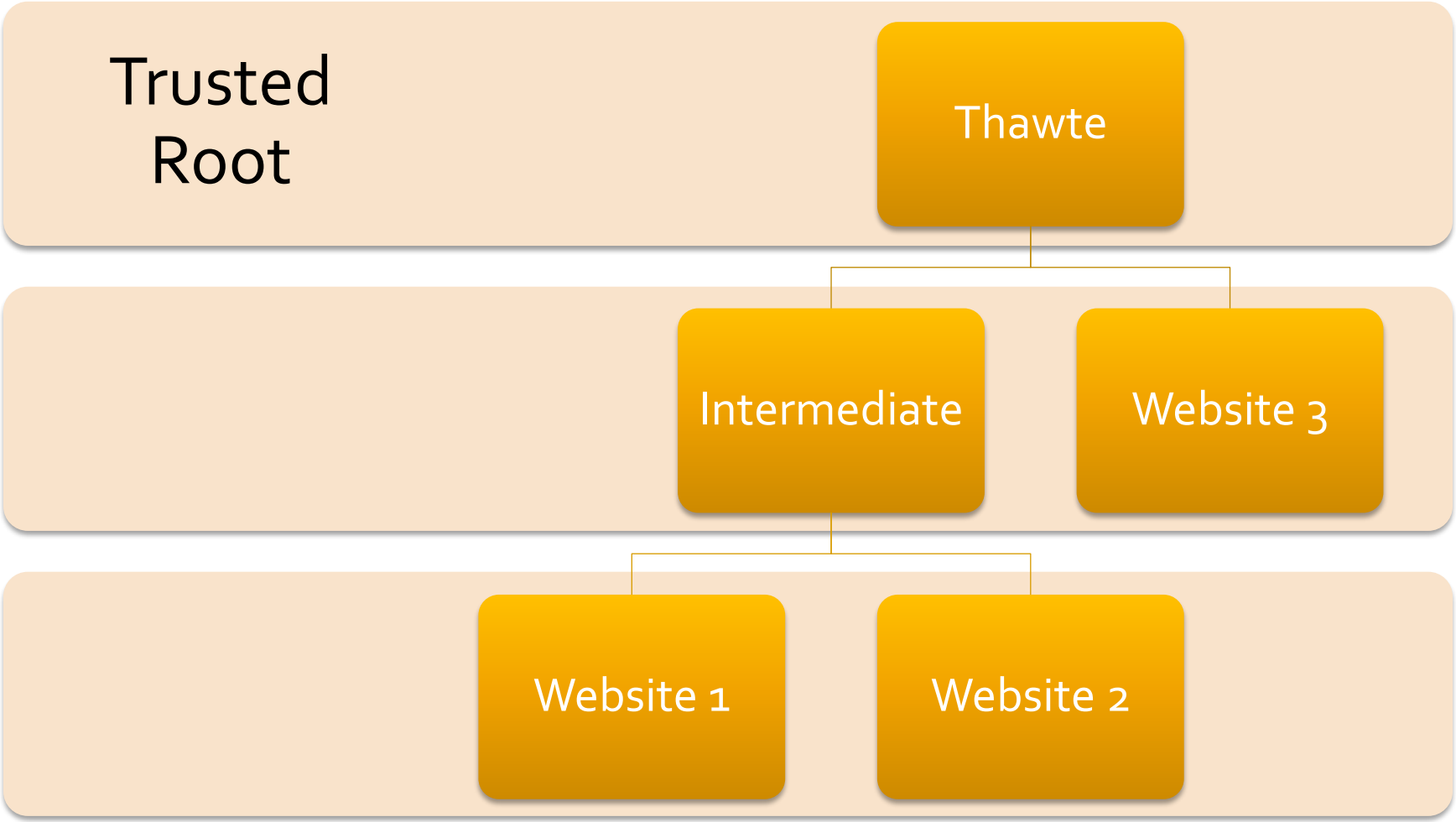
- A *signing key* (private key) transforms plaintext into ciphertext
- A *verification key* (public key) transforms ciphertext into plaintext
- Typically a message digest is signed
  - Hopefully not MD5

Asymmetric Cryptography

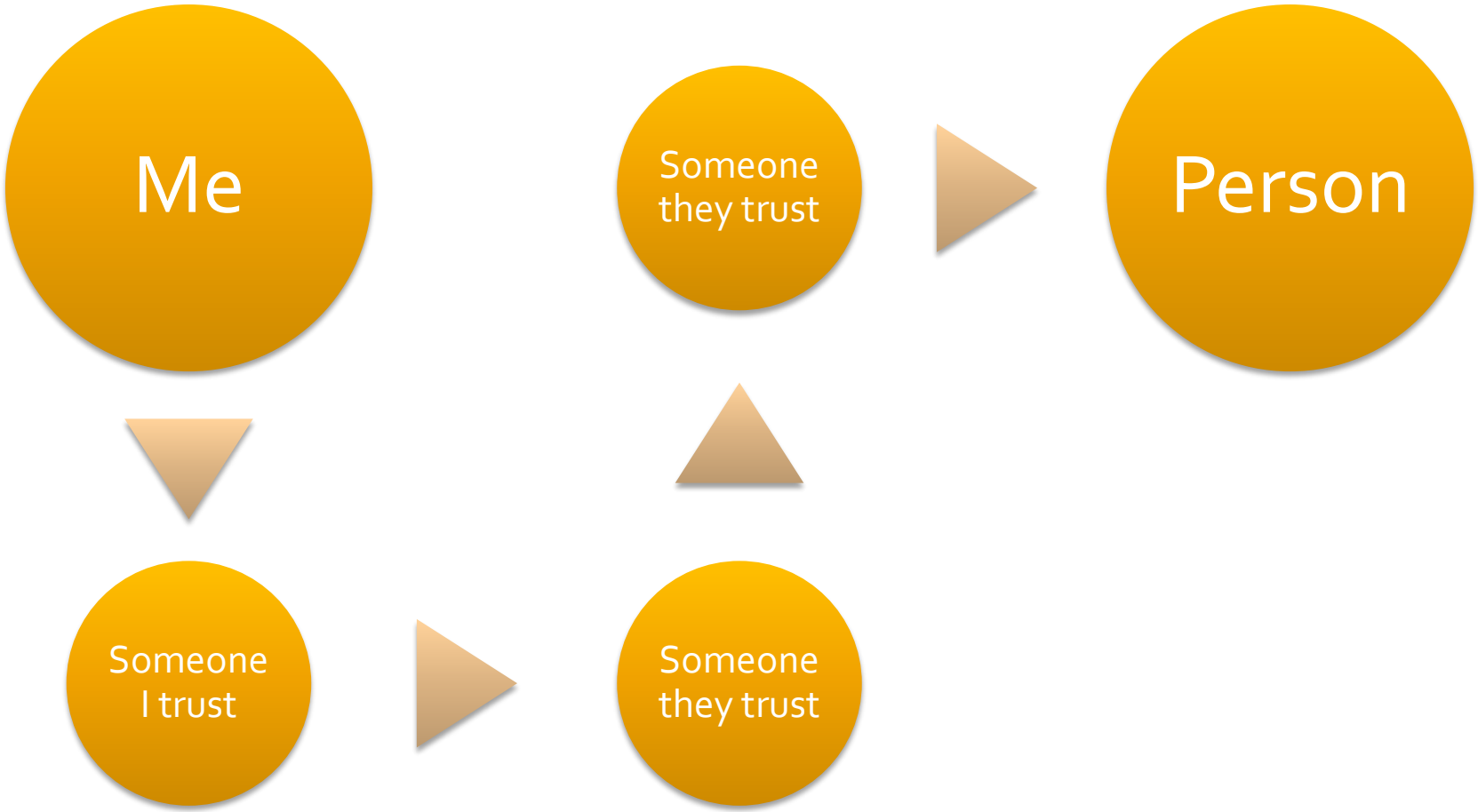
- DO: use a 2048-bit RSA key
- DON'T: use RSA without message padding
- DO: use RSASSA-PSS for signing
- DO: use RSAES-OAEP for encrypting
- AVOID: using PKCS v1.5 padding
- AVOID: using the same RSA key for both authentication and encryption

Public Key Infrastructure

X.509


Trusted Root
Thawte
Intermediate
Website 3
Website 1
Website 2

Web of Trust


Me
Someone they trust
Person
Someone I trust
Someone they trust

Transport Layer Security

TLS / SSL

- X.509 implementation
- Complex system
  - Had its fair share of problems
- Education has focused on the padlock icon
- DO: use TLS to secure your web server, email, etc
- DO: think very carefully which Certificate Authorities you trust

Trusted Root Authorities

- AOL Time Warner Inc.
- AS Sertifitseerimiskeskus
- AddTrust
- Baltimore
- beTRUSTed
- Buypass
- CNNIC
- COMODO CA Limited
- Certplus
- certSIGN
- Chambersign
- Chunghwa Telecom Co., Ltd.
- ComSign
- Comodo CA Limited
- Cybertrust, Inc
- Deutsche Telekom AG
- Deutscher Sparkassen Verlag GmbH
- Dhimyotis
- DigiCert Inc
- DigiNotar
- Digital Signature Trust Co.
- Disig a.s.
- EBG Bilişim Teknolojileri ve Hizmetleri A.Ş.
- EDICOM
- Entrust, Inc.
- Equifax
- GTE Corporation
- GeoTrust Inc.
- GlobalSign nv-sa
- Hongkong Post
- Japan Certification Services, Inc.
- Japanese Government
- Microsec Ltd.
- NetLock Halozatbiztonsagi Kft.
- Network Solutions L.L.C.
- PM/SGDN
- QuoVadis Limited
- RSA Security Inc
- SECOM Trust Systems CO.,LTD.
- SecureTrust Corporation
- Sociedad Cameral de Certificación Digital
- Sonera
- Staat der Nederlanden
- Starfield Technologies, Inc.
- StartCom Ltd.
- SwissSign AG
- Swisscom
- TC TrustCenter GmbH
- TDC
- Taiwan Government
- Thawte
- The Go Daddy Group, Inc.
- The USERTRUST Network
- TÜBİTAK
- TÜRKTRUST
- Unizeto Sp. z o.o.
- VISA
- ValiCert, Inc.
- VeriSign, Inc.
- WISeKey
- Wells Fargo
- XRamp Security Services Inc

When cryptography is outlawed

Questions

Resources

- Videos
  - Theory and Practice of Cryptography series
    - http://www.youtube.com/watch?v=IzVCrSrZIX8
    - http://www.youtube.com/watch?v=KDvt_ocafPw
    - http://www.youtube.com/watch?v=YcgqBEzcD_I
    - http://www.youtube.com/watch?v=ZDnShu5V99s
  - Crypto Strikes Back!
    - http://www.youtube.com/watch?v=ySQloNhW1Jo
- Presentations
  - http://www.bsdcan.org/2010/schedule/attachments/135_crypto1hr.pdf
  - http://www.eff.org/files/DefconSSLiverse.pdf
- Books
  - Applied Cryptography by Niels Ferguson and Bruce Schneier
  - Practical Cryptography by Bruce Schneier
- Blogs
  - http://rdist.root.org/
  - http://www.schneier.com/