CSE P 590 / CSE M 590 (Spring 2010)

Computer Security and Privacy

Tadayoshi Kohno

Thanks to Dan Boneh, Dieter Gollmann, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

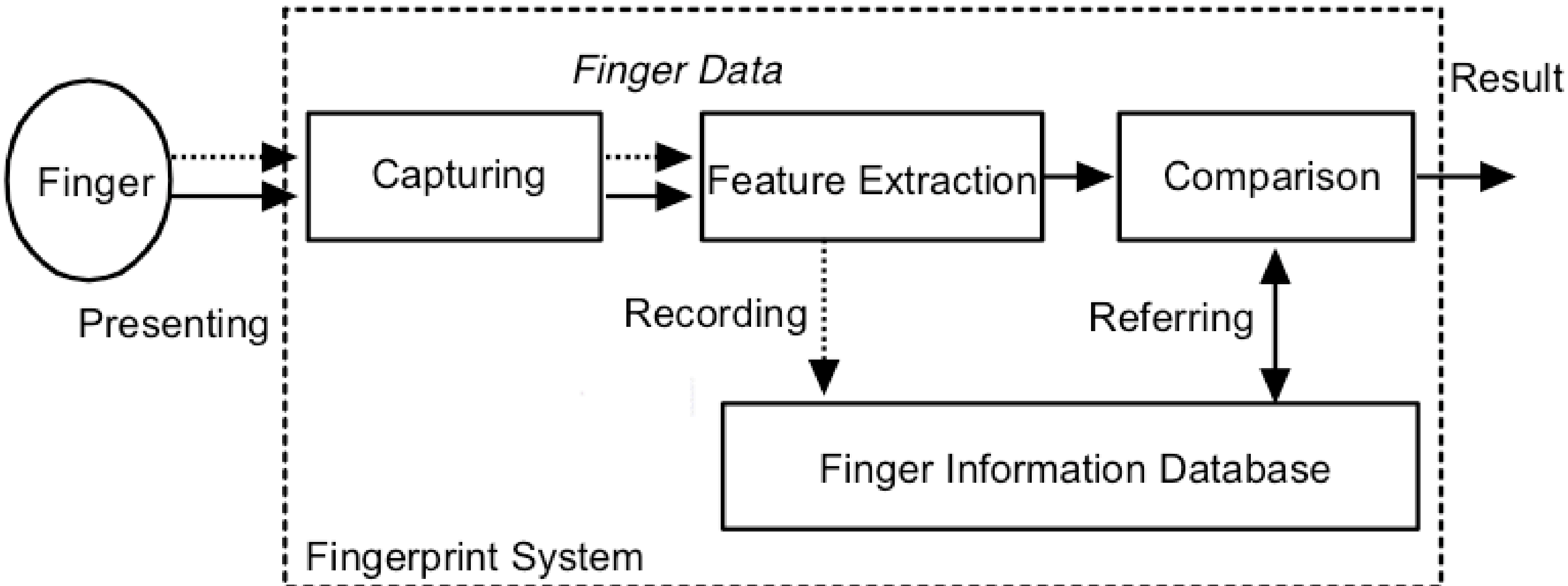
Goals for Today

- User authentication
- Security reviews
- Asymmetric Crypto
- Research reading

What About Biometrics?

- Authentication: What you are
- Unique identifying characteristics to authenticate user or create credentials
  - Biological and physiological: Fingerprints, iris scan
  - Behaviors characteristics - how perform actions: Handwriting, typing, gait
- Advantages:
  - Nothing to remember
  - Passive
  - Can't share (generally)
  - With perfect accuracy, could be fairly unique

Overview [Matsumoto]



Tsutomu Matsumoto's image, from http://web.mit.edu/6.857/OldStuff/Fall03/ref/gummy-slides.pdf

Dashed lines for enrollment; solid for verification or identification

Biometric Error Rates (Non-Adversarial)

- "Fraud rate" vs. "insult rate"
  - Fraud = system incorrectly accepts (false accept)
  - Insult = system rejects valid user (false reject)
- Increasing acceptance threshold increases fraud rate, decreases insult rate
- For biometrics, U.K. banks set target fraud rate of 1%, insult rate of 0.01% [Ross Anderson]

Biometrics

- Face recognition (by a computer algorithm)
  - Error rates up to 20%, given reasonable variations in lighting, viewpoint and expression
- Fingerprints
  - Traditional method for identification
  - 1911: first US conviction on fingerprint evidence
  - U.K. traditionally requires 16-point match
    - Probability of false match is 1 in 10 billion
    - No successful challenges until 2000
  - Fingerprint damage impairs recognition

Other Biometrics

- Iris scanning
  - Irises are very random, but stable through life
    - Different between the two eyes of the same individual
  - 256-byte iris code based on concentric rings between the pupil and the outside of the iris
  - Equal error rate better than 1 in a million
  - Best biometric mechanism currently known
- Hand geometry
  - Used in nuclear premises entry control, INSPASS (discontinued in 2002)

Other Biometrics

- Vein
  - Pattern on back of hand
- Handwriting
- Typing
  - Timings for character sequences
- Gait
- DNA

Issues with Biometrics

- Private, but not secret
  - Maybe encoded on the back of an ID card?
  - Maybe encoded on your glass, door handle, ...
  - Sharing between multiple systems?
- Revocation is difficult (impossible?)
  - Sorry, your iris has been compromised, please create a new one...
- Physically identifying
  - Soda machine to cross-reference fingerprint with DMV?

Issues with Biometrics

- Collection error: Criminal gives an inexperienced policeman fingerprints in the wrong order
  - Record not found; gets off as a first-time offender
- Can be attacked using recordings
  - Ross Anderson: in countries where fingerprints are used to pay pensions, there are persistent tales of “Granny’s finger in the pickle jar” being the most valuable property she bequeathed to her family
- Birthday paradox
  - With false accept rate of 1 in a million, probability of false match is above 50% with only 1609 samples

Risks of Biometrics

BBC NEWS

OPEN **The News in 2 minutes**

News services
Your news when want it

News Front Page

Africa
Americas
Asia-Pacific
Europe
Middle East
South Asia
UK
Business
Health
Science/Nature
Technology
Entertainment

Last Updated: Thursday, 31 March, 2005, 10:37 GMT 11:37 UK

E-mail this to a friend   Printable version

Malaysia car thieves steal finger

By Jonathan Kent
BBC News, Kuala Lumpur

Police in Malaysia are hunting for members of a violent gang who chopped off a car owner's finger to get round the vehicle's hi-tech security system.

The car, a Mercedes S-class, was protected by a fingerprint recognition system.

Accountant K Kumaran's ordeal began when he was run down by four men in a small car as he was about to get into his Mercedes in a Kuala Lumpur suburb.

SEE ALSO:
- Malaysia to act a pirates
16 Mar 05 | As

RELATED INTER
- Malaysian police

The BBC is not r for the content o internet sites

TOP ASIA-PACIF STORIES
- Australians war cuts
- Taiwan campus

http://news.bbc.co.uk/2/hi/asia-pacific/4396831.stm

Biometric Error Rates (Adversarial)

- Want to minimize "fraud" and "insult" rate
  - "Easy" to test probability of accidental misidentification (fraud)
  - But what about adversarial fraud
    - Besides stolen fingers
- An adversary might try to steal the biometric information
  - Malicious fingerprint reader
    - Consider when biometric is used to derive a cryptographic key
  - Residual fingerprint on a glass

Voluntary: Making a Mold

[Matsumoto]

Put the plastic into hot water to soften it.

Press a live finger against it.

The mold

It takes around 10 minutes.

http://web.mit.edu/6.857/OldStuff/Fall03/ref/gummy-slides.pdf

Voluntary: Making a Finger

[Matsumoto]

Pour the liquid into the mold.

Put it into a refrigerator to cool.

It takes around 10 minutes.

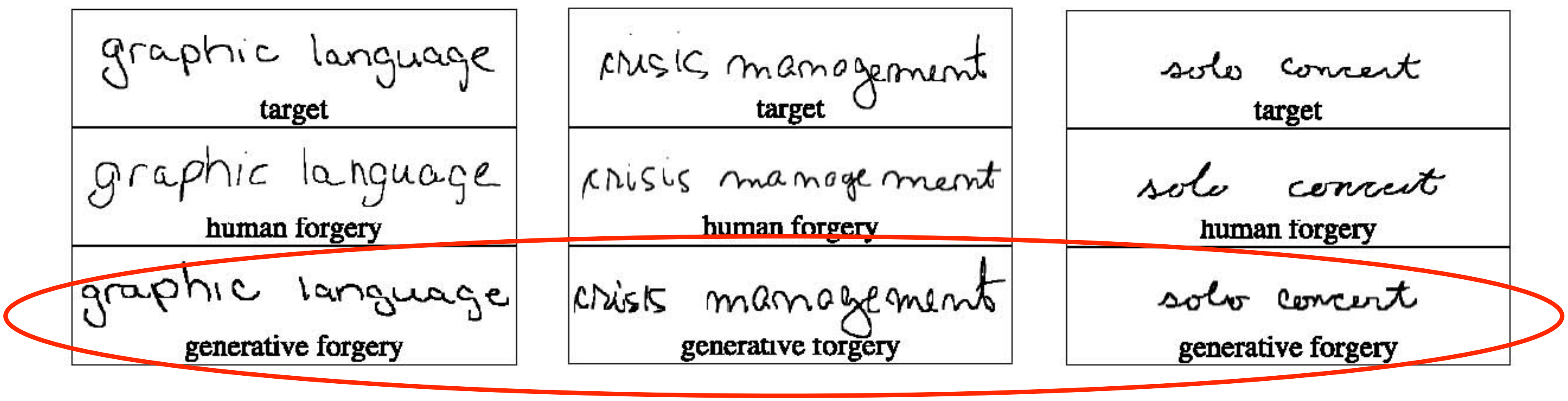
The gummy finger

http://web.mit.edu/6.857/OldStuff/Fall03/ref/gummy-slides.pdf

Authentication by Handwriting

[Ballard, Monrose, Lopresti]

- Maybe a computer could also forge some biometrics

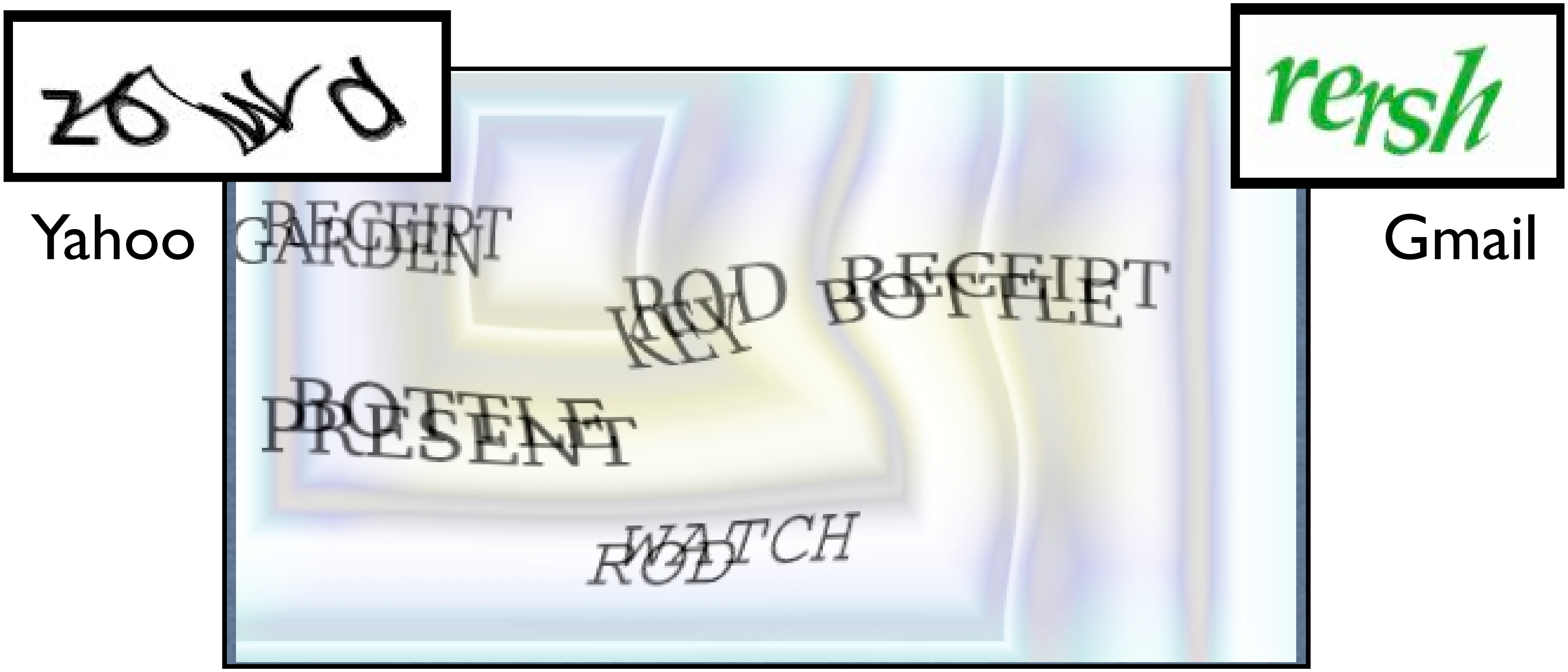


Generated by computer algorithm trained on handwriting samples

Human Verification

- Problem:
  - Want to make it hard for spammers to automatically create many free email accounts
  - Want to make it difficult for computers to automatically crawl some data repository
- Need a method for servers to distinguish between
  - Human users
  - Machine users
- Approach: CAPTCHA
  - Completely Automated Public Turing Test to Tell Computers and Humans Apart

CAPTCHAs



captcha.net

Idea: “easy” for humans to read words in this picture, but “hard” for computers

Caveats

- Usability challenges with visual impairments
- Researchers studying how to break CAPTCHAs
- Some attackers don't break CAPTCHAs; they hire or trick others

The following article describes an attack against the web images (so-called
"CAPTCHAS") that a
applications such
form of "Turing Te
but difficult for
CAPTCHA image to a
CAPTCHA to get the
spammers in creati

"But at least one
Someone designed a
and, when confront
site. Visitors to
they could view mo
answer to complete

Will Solve Captcha for Money?

Posted by CmdrTaco on Wed Sep 06, '06 08:37 AM
from the **i've-done-worse-for-less** dept.

alx_lo writes

"Captchas are a nice idea to protect your blog or guestbook from being spammed by robots. But what good is this protection when you can hire "data entry specialists" to solve captchas for $0.60 per hour for 50 hours a week? Anyone here who can think up a solution that does not include drastically changing the global economy? How about captchas that require cultural background knowledge to solve?"

PC
PCMAG.COM

Four Indicted in CAPTCHA Hacks of Ticket Sites

03.01.10

1 Comment

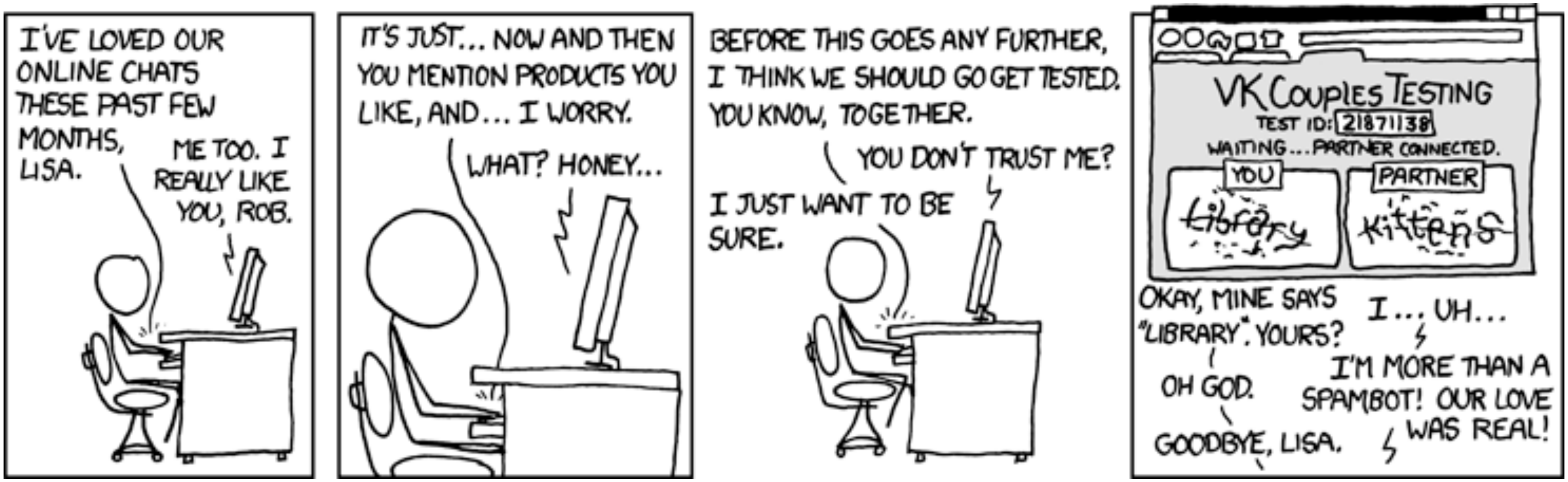
By Chloe Albanesius

Did you miss out on floor seats for Bruce Springsteen's July 2008 concert at Gi

Fo
inc
sn
Tic
ve
Jus

How did they do it? Most online ticket Web sites like Ticketmaster employ CAPTCHA technologies, which requires users to read images that are recognizable to the human eye but confusing to computers, and type them into a box before buying tickets.

The defendants, however, worked with computer programmers in Bulgaria to develop a technology that allowed a network of computers to impersonate individual visitors to online ticket vendors. The ticket vendors did not immediately recognize the purchases as computer-generated, so these "CAPTCHA Bots" let Wiseguy Tickets to flood ticket vendors as soon as tickets went on sale and purchase tickets faster than any human.



I'VE LOVED OUR ONLINE CHATS THESE PAST FEW MONTHS, LISA.
ME TOO. I REALLY LIKE YOU, ROB.
IT'S JUST... NOW AND THEN YOU MENTION PRODUCTS YOU LIKE, AND... I WORRY.
WHAT? HONEY...
BEFORE THIS GOES ANY FURTHER, I THINK WE SHOULD GO GET TESTED. YOU KNOW, TOGETHER.
YOU DON'T TRUST ME?
I JUST WANT TO BE SURE.
VK COUPLES TESTING
TEST ID: 21871138
WAITING... PARTNER CONNECTED.
YOU
Library
PARTNER
Kittens
OKAY, MINE SAYS "LIBRARY". YOURS?
I... UH...
OH GOD.
I'M MORE THAN A SPAMBOT! OUR LOVE WAS REAL!
GOODBYE, LISA.

Phishing

- A form of social engineering
- Some comments here; more with research paper later

Experiments at Indiana University

[Jagatic et al.]

- Reconstructed the social network by crawling sites like Facebook, MySpace, LinkedIn and Friendster
- Sent 921 Indiana University students a spoofed email that appeared to come from their friend
- Email redirected to a spoofed site inviting the user to enter his/her secure university credentials
  - Domain name clearly distinct from indiana.edu
- 72% of students entered their real credentials into the spoofed site

More Details

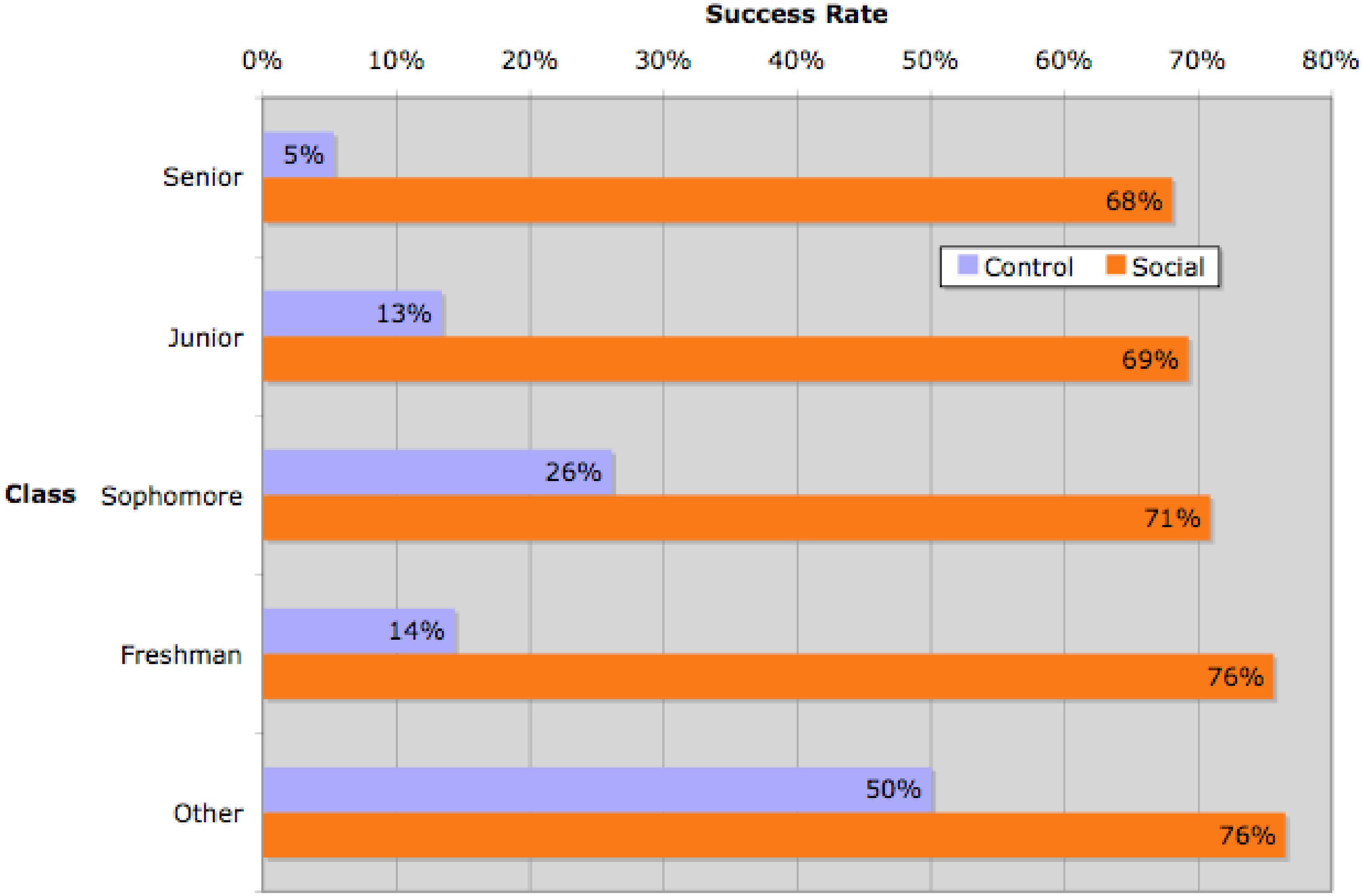
- Control group: 15 of 94 (16%) entered personal information
- Social group: 349 of 487 (72%) entered personal information

- 70% of responses within first 12 hours
- Adversary wins by gaining users' trust

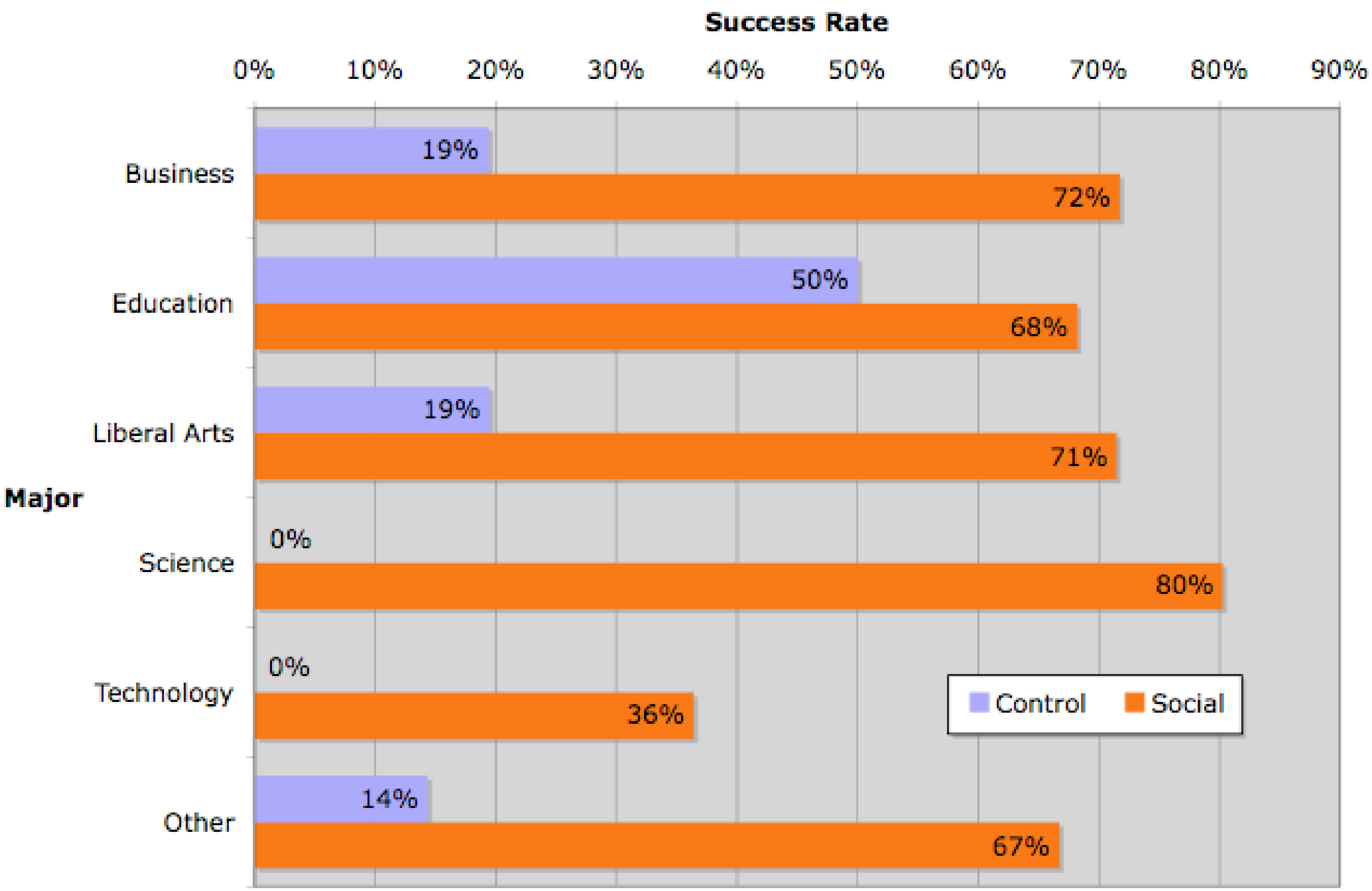
More Details

	To Male	To Female	To Any
From Male	53%	78%	68%
From Female	68%	76%	73%
From Any	65%	77%	72%

More Details



More Details



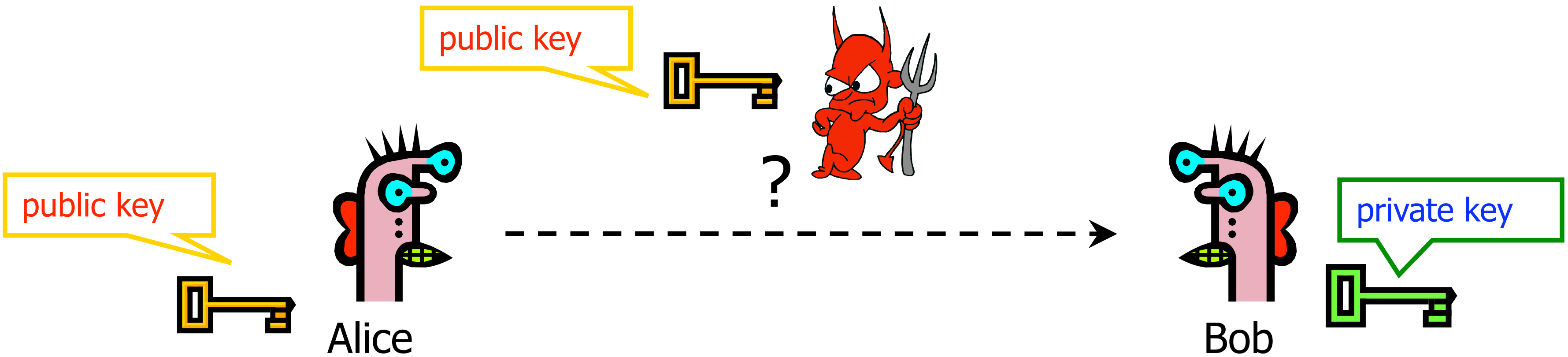
Security Reviews

Security Reviews

- Summary of system
- Assets
- Adversaries and threats
- Potential weaknesses (possibly speculative)
- Potential defenses
- Risks
- Conclusions

Public Key Cryptography

Basic Problem



Given: Everybody knows Bob's public key
Only Bob knows the corresponding private key

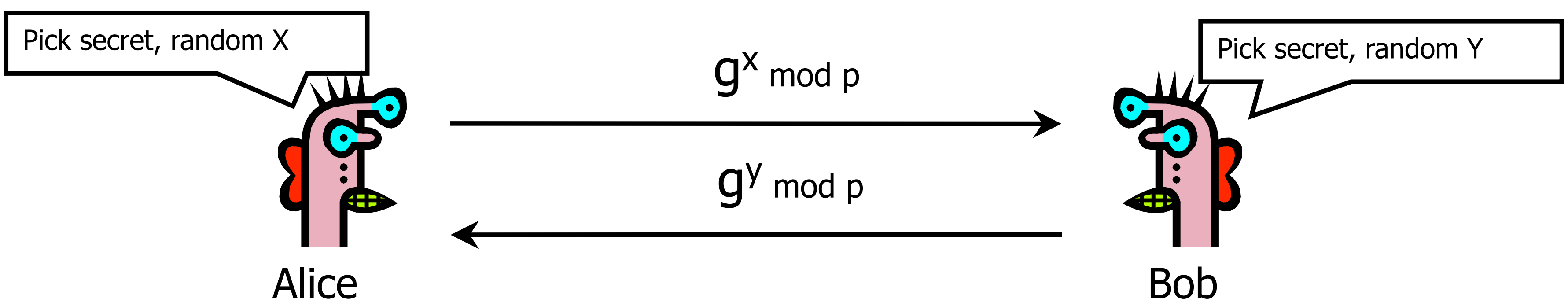
Goals: 1. Alice wants to send a secret message to Bob
2. Bob wants to authenticate himself

Applications of Public-Key Crypto

- Encryption for confidentiality
  - Anyone can encrypt a message
    - With symmetric crypto, must know secret key to encrypt
  - Only someone who knows private key can decrypt
  - Key management is simpler (maybe)
    - Secret is stored only at one site: good for open environments
- Digital signatures for authentication
  - Can "sign" a message with your private key
- Session key establishment
  - Exchange messages to create a secret session key
  - Then switch to symmetric cryptography (why?)

Diffie-Hellman Protocol (1976)

- Alice and Bob never met and share no secrets
- Public info: p and g
  - p is a large prime number, g is a generator of Z_p^*
    - $Z_p^*=\{1, 2 \dots p-1\}$; $\forall a \in Z_p^*\ \exists i$ such that $a=g^i \bmod p$
    - Modular arithmetic: numbers "wrap around" after they reach p



Compute $k=(g^y)^x=g^{xy} \bmod p$

Compute $k=(g^x)^y=g^{xy} \bmod p$

Why Is Diffie-Hellman Secure?

- Discrete Logarithm (DL) problem:

  given $g^x \bmod p$, it's hard to extract x
  - There is no known efficient algorithm for doing this
  - This is not enough for Diffie-Hellman to be secure!
- Computational Diffie-Hellman (CDH) problem:

  given g^x and g^y, it's hard to compute $g^{xy} \bmod p$
  - … unless you know x or y, in which case it's easy
- Decisional Diffie-Hellman (DDH) problem:

  given g^x and g^y, it's hard to tell the difference between $g^{xy} \bmod p$ and $g^r \bmod p$ where r is random

Properties of Diffie-Hellman

- Assuming DDH problem is hard, Diffie-Hellman protocol is a secure key establishment protocol against <u>passive</u> attackers
  - Eavesdropper can't tell the difference between established key and a random value
  - Can use new key for symmetric cryptography
    - Approx. 1000 times faster than modular exponentiation
- Diffie-Hellman protocol (by itself) does not provide authentication

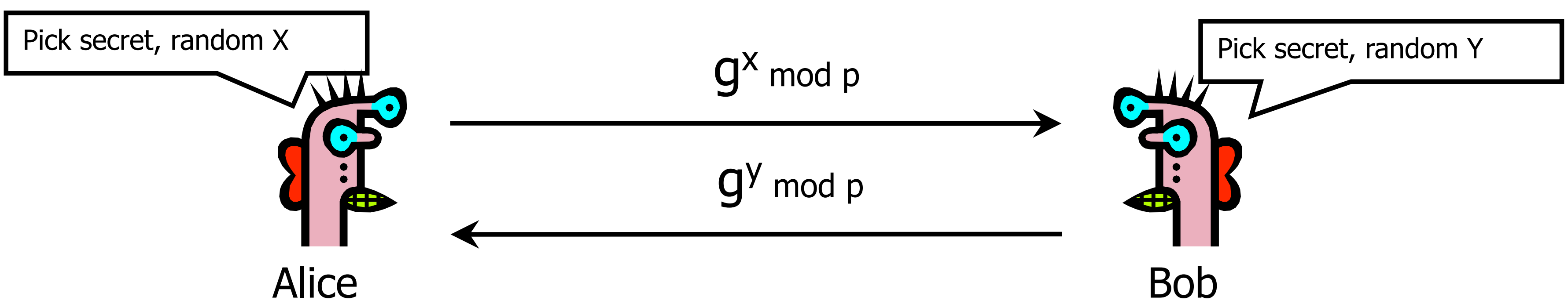
Properties of Diffie-Hellman

- DDH: not true for integers mod p, but true for other groups
- DL problem in p can be broken down into DL problems for subgroups, if factorization of p-1 is known.
- Common recommendation:
  - Choose $p = 2q+1$ where q is also a large prime.
  - Pick a g that generates a subgroup of order q in Z_p^*
  - (OK to not know all the details of why for this course.)

  - Hash output of DH key exchange to get the key

Diffie-Hellman Protocol (1976)

- Alice and Bob never met and share no secrets
- Public info: p and g
  - p, q are large prime numbers, $p=2q+1$, g a generator for the subgroup of order q
    - Modular arithmetic: numbers "wrap around" after they reach p



Compute $k=H((g^y)^x)=H(g^{xy}) \bmod p$

Compute $k=H((g^x)^y)=H(g^{xy}) \bmod p$

Requirements for Public-Key Encryption

- **Key generation:** computationally easy to generate a pair (public key PK, private key SK)
  - Computationally infeasible to determine private key SK given only public key PK
- **Encryption:** given plaintext M and public key PK, easy to compute ciphertext $C=E_{PK}(M)$
- **Decryption:** given ciphertext $C=E_{PK}(M)$ and private key SK, easy to compute plaintext M
  - Infeasible to compute M from C without SK
  - Even infeasible to learn partial information about M
  - Trapdoor function: Decrypt(SK,Encrypt(PK,M))=M

Some Number Theory Facts

- Euler totient function $\varphi(n)$ where $n \geq 1$ is the number of integers in the [1,n] interval that are relatively prime to n
  - Two numbers are relatively prime if their greatest common divisor (gcd) is 1
- Euler's theorem:

  if $a \in Z_n^*$, then $a^{\varphi(n)} = 1 \bmod n$
- Special case: <u>Fermat's Little Theorem</u>

  if p is prime and $\gcd(a,p)=1$, then $a^{p-1} = 1 \bmod p$

RSA Cryptosystem [Rivest, Shamir, Adleman 1977]

- Key generation:
  - Generate large primes p, q
    - Say, 1024 bits each (need primality testing, too)
  - Compute $n=pq$ and $\varphi(n)=(p-1)(q-1)$
  - Choose small e, relatively prime to $\varphi(n)$
    - Typically, $e=3$ or $e=2^{16}+1=65537$ (why?)
  - Compute unique d such that $ed = 1 \bmod \varphi(n)$
  - Public key = (e,n); private key = (d,n)
- Encryption of m: $c = m^e \bmod n$
  - Modular exponentiation by repeated squaring
- Decryption of c: $c^d \bmod n = (m^e)^d \bmod n = m$

Why RSA Decryption Works

- $e \cdot d = 1 \bmod \varphi(n)$
- Thus $e \cdot d = 1 + k \cdot \varphi(n) = 1 + k(p-1)(q-1)$ for some k

- Let m be any integer in Z_n
- If $\gcd(m,p)=1$, then $m^{ed} = m \bmod p$
  - By Fermat's Little Theorem, $m^{p-1} = 1 \bmod p$
  - Raise both sides to the power k(q-1) and multiply by m
  - $m^{1+k(p-1)(q-1)} = m \bmod p$, thus $m^{ed} = m \bmod p$
  - By the same argument, $m^{ed} = m \bmod q$
- Since p and q are distinct primes and $p \cdot q = n$,

  $m^{ed} = m \bmod n$

On RSA

- Encrypted message needs to be in interpreted as an integer less than n
  - Reason: Otherwise can't decrypt.
  - Message is very often a symmetric encryption key.

Why Is RSA Secure?

- RSA problem: given n=pq, e such that gcd(e,(p-1)(q-1))=1 and c, find m such that $m^e = c \bmod n$
  - i.e., recover m from ciphertext c and public key (n,e) by taking e^{th} root of c
  - There is no known efficient algorithm for doing this
- Factoring problem: given positive integer n, find primes $p_1, \ldots, p_k$ such that $n = p_1^{e_1} p_2^{e_2} \ldots p_k^{e_k}$
- If factoring is easy, then RSA problem is easy, but there is no known reduction from factoring to RSA
  - It may be possible to break RSA without factoring n

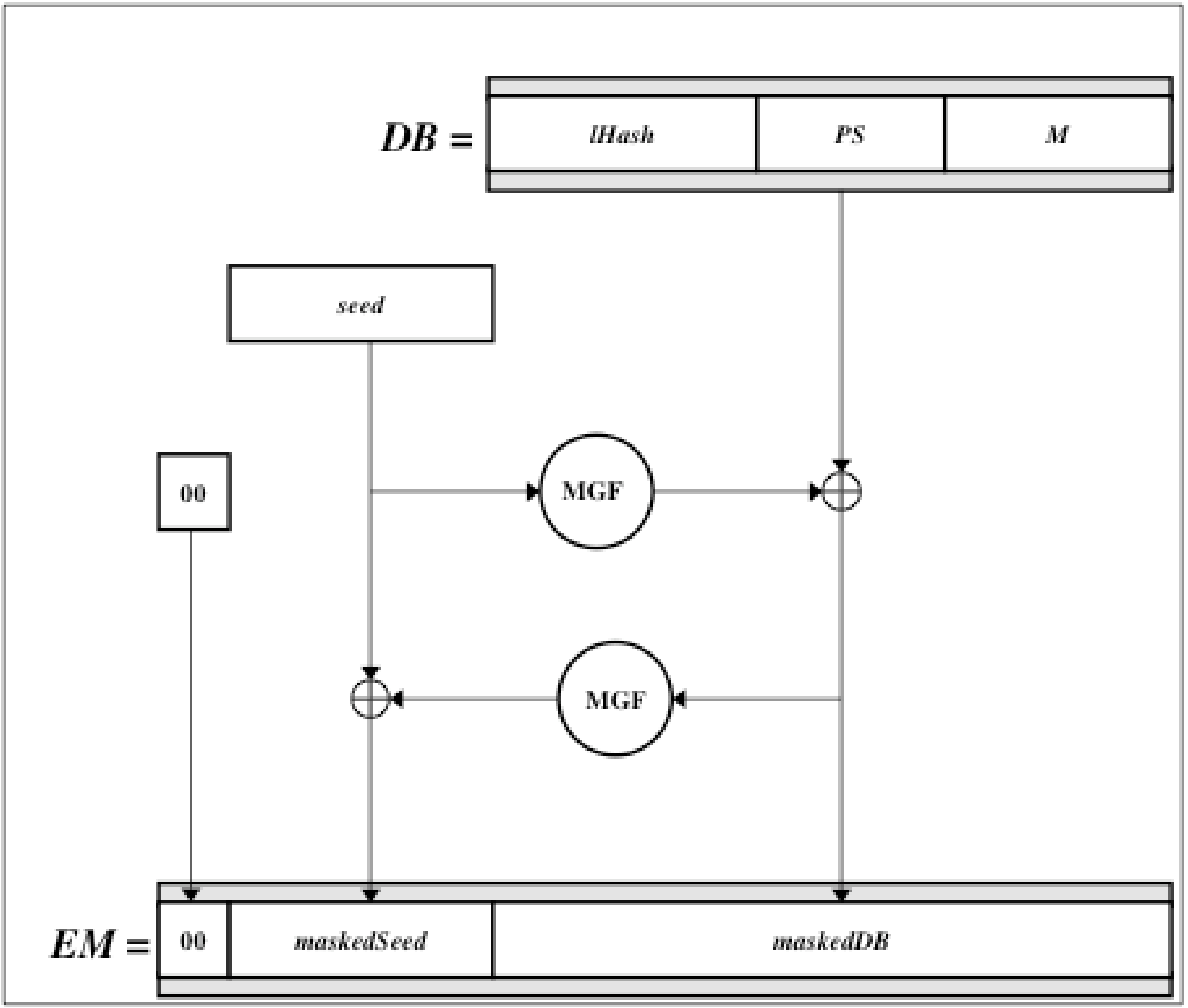
Caveats

- $e = 3$ is a common exponent
  - If $m < n^{1/3}$, then $c = m^3 < n$ and can just take the cube root of c to recover m
    - Even problems if "pad" m in some ways [Hastad]
  - Let $c_i = m^3 \bmod n_i$ - same message is encrypted to three people
    - Adversary can compute $m^3 \bmod n_1 n_2 n_3$ (using CRT)
    - Then take ordinary cube root to recover m
- Don't use RSA **directly** for privacy!

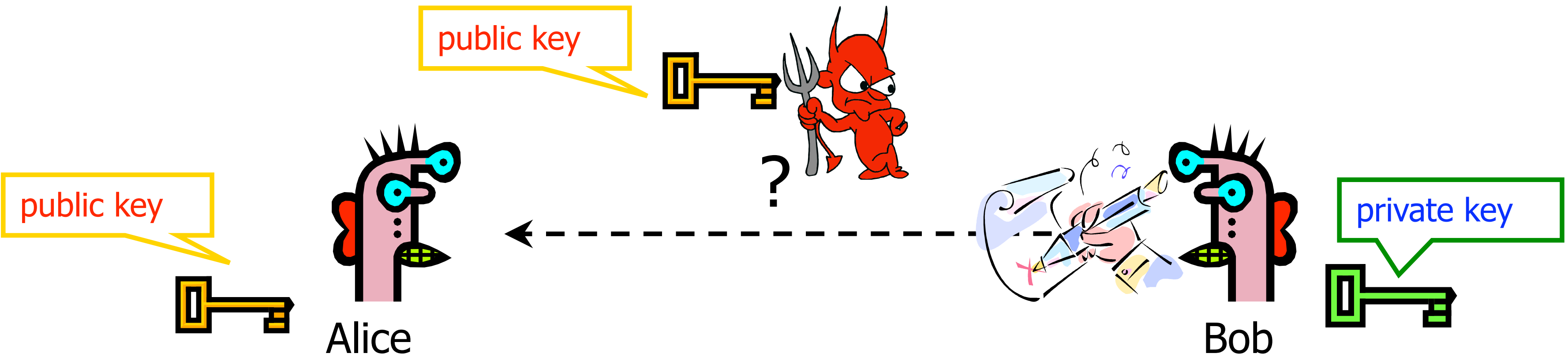
Integrity in RSA Encryption

- Plain RSA does not provide integrity
  - Given encryptions of m_1 and m_2, attacker can create encryption of $m_1 \cdot m_2$
    - $(m_1^e) \cdot (m_2^e) \bmod n = (m_1 \cdot m_2)^e \bmod n$
  - Attacker can convert m into m^k without decrypting
    - $(m_1^e)^k \bmod n = (m^k)^e \bmod n$
- In practice, OAEP is used: instead of encrypting M, encrypt $M \oplus G(r)\ ;\ r \oplus H(M \oplus G(r))$
  - r is random and fresh, G and H are hash functions
  - Resulting encryption is plaintext-aware: infeasible to compute a valid encryption without knowing plaintext
    - … if hash functions are “good” and RSA problem is hard

OAEP (image from PKCS #1 v2.1)



Digital Signatures: Basic Idea



<u>Given</u>: Everybody knows Bob's public key

Only Bob knows the corresponding private key

<u>Goal</u>: Bob sends a "digitally signed" message

1. To compute a signature, must know the private key
2. To verify a signature, enough to know the public key

RSA Signatures

- Public key is (n,e), private key is d
- To sign message m: $s = m^d \bmod n$
  - Signing and decryption are the same **underlying** operation in RSA
  - It's infeasible to compute s on m if you don't know d
- To verify signature s on message m:

  $s^e \bmod n = (m^d)^e \bmod n = m$
  - Just like encryption
  - Anyone who knows n and e (public key) can verify signatures produced with d (private key)
- In practice, also need padding & hashing
  - Standard padding/hashing schemes exist for RSA signatures

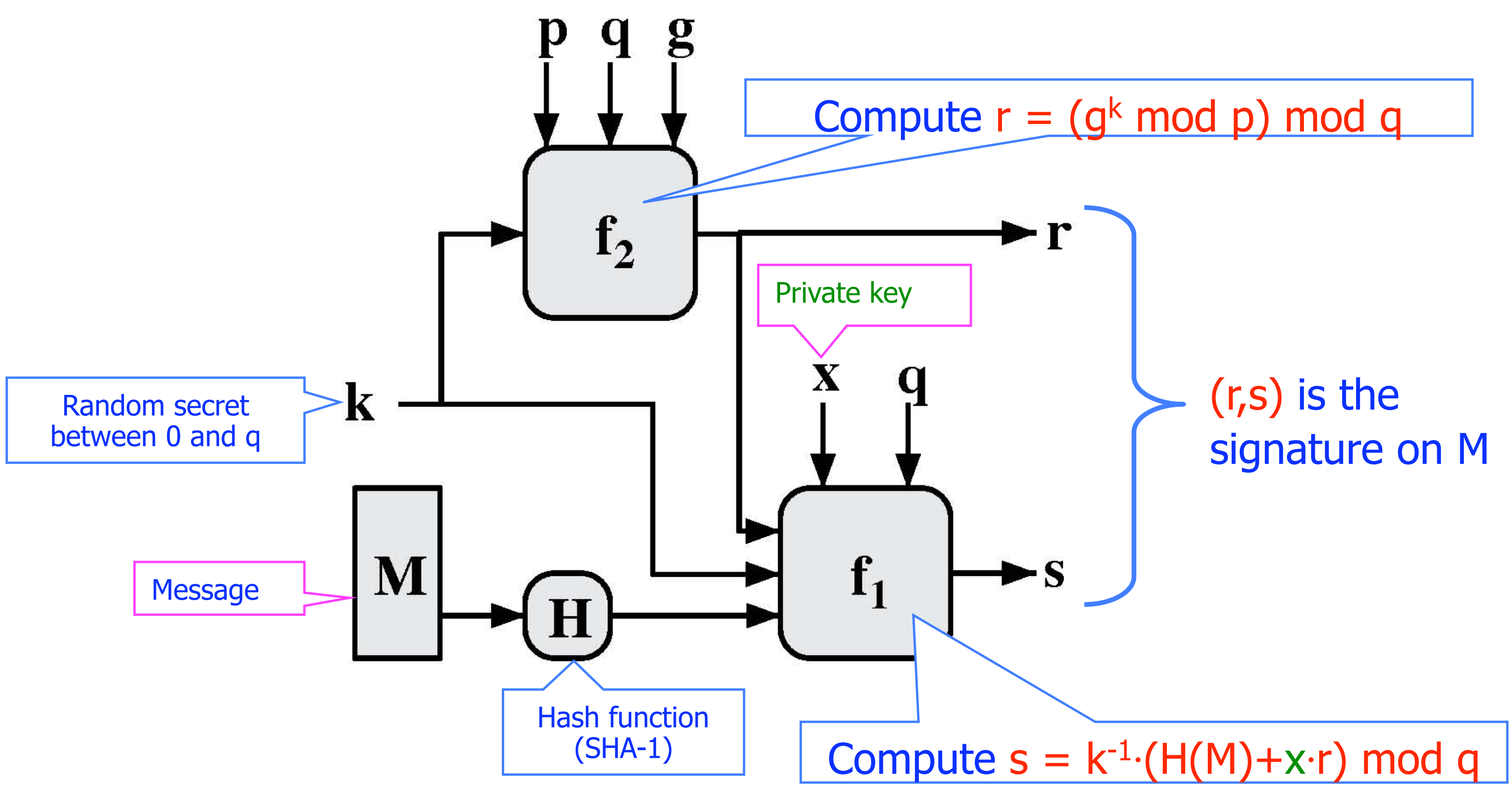
Encryption and Signatures

- Often people think: Encryption and decryption are inverses.
- That's a common view
  - True for the RSA **primitive (underlying component)**
- But not one we'll take
  - To really use RSA, we need padding
  - And there are many other decryption methods

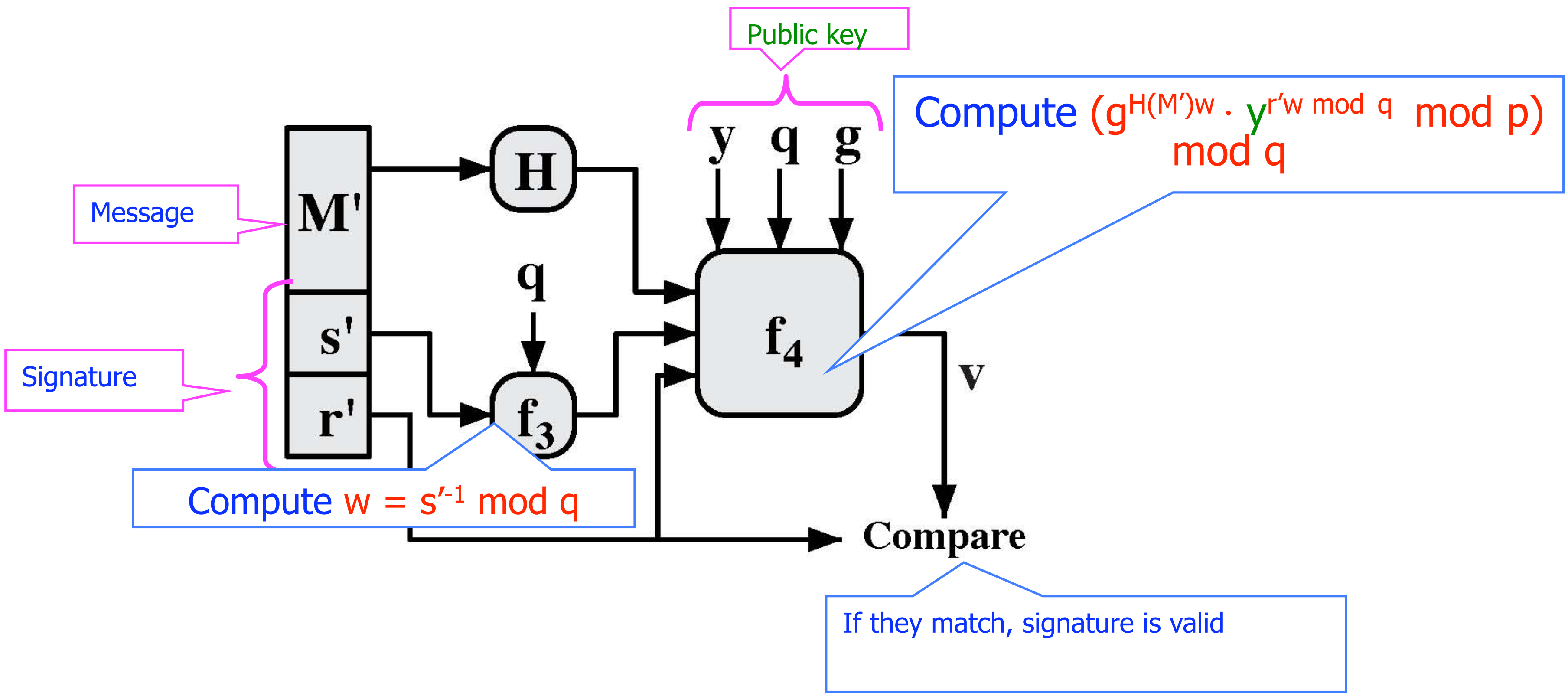
Digital Signature Standard (DSS)

- U.S. government standard (1991-94)
  - Modification of the ElGamal signature scheme (1985)
- Key generation:
  - Generate large primes p, q such that q divides p-1
    - $2^{159} < q < 2^{160}$, $2^{511+64t} < p < 2^{512+64t}$ where $0 \le t \le 8$
  - Select $h \in Z_p^*$ and compute $g = h^{(p-1)/q} \bmod p$
  - Select random x such $1 \le x \le q-1$, compute $y = g^x \bmod p$
- Public key: $(p, q, g, y=g^x \bmod p)$, private key: x
- Security of DSS requires hardness of discrete log
  - If could solve discrete logarithm problem, would extract x (private key) from $g^x \bmod p$ (public key)

DSS: Signing a Message (Skim)



DSS: Verifying a Signature (Skim)



Why DSS Verification Works (Skim)

- If (r,s) is a legitimate signature, then
  $r = (g^k \bmod p) \bmod q$; $s = k^{-1}\cdot(H(M)+x\cdot r) \bmod q$
- Thus $H(M) = -x\cdot r+k\cdot s \bmod q$
  - Multiply both sides by $w=s^{-1} \bmod q$
- $H(M)\cdot w + x\cdot r\cdot w = k \bmod q$
  - Exponentiate g to both sides
- $(g^{H(M)\cdot w + x\cdot r\cdot w} = g^k) \bmod p \bmod q$
  - In a valid signature, $g^k \bmod p \bmod q = r$, $g^x \bmod p = y$
- Verify $g^{H(M)\cdot w}\cdot y^{r\cdot w} = r \bmod p \bmod q$

Security of DSS

- Can’t create a valid signature without private key
- Given a signature, hard to recover private key
- Can’t change or tamper with signed message
- If the same message is signed twice, signatures are different
  - Each signature is based in part on random secret k
- Secret k must be different for each signature!
  - If k is leaked or if two messages re-use the same k, attacker can recover secret key x and forge any signature from then on
  - Example problem scenario: rebooted VMs; restarted embedded machines