# Symmetric-Key Cryptography

*CS 161: Computer Security*


**Prof. Vern Paxson**

TAs: Paul Bramsen, Apoorva Dornadula,
David Fifield, Mia Gil Epner, David Hahn, Warren He,
Grant Ho, Frank Li, Nathan Malkin, Mitar Milutinovic,
Rishabh Poddar, Rebecca Portnoff, Nate Wang

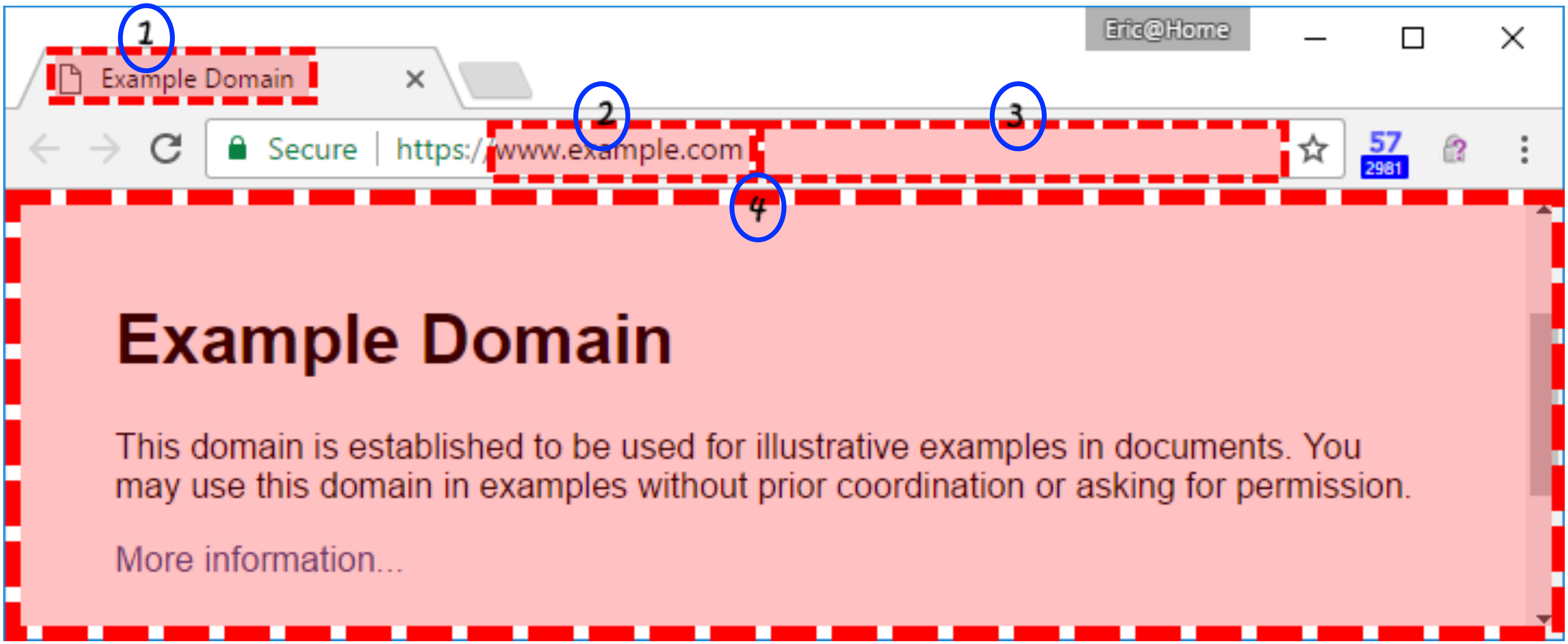

*http://inst.eecs.berkeley.edu/~cs161/*

**February 21, 2017**

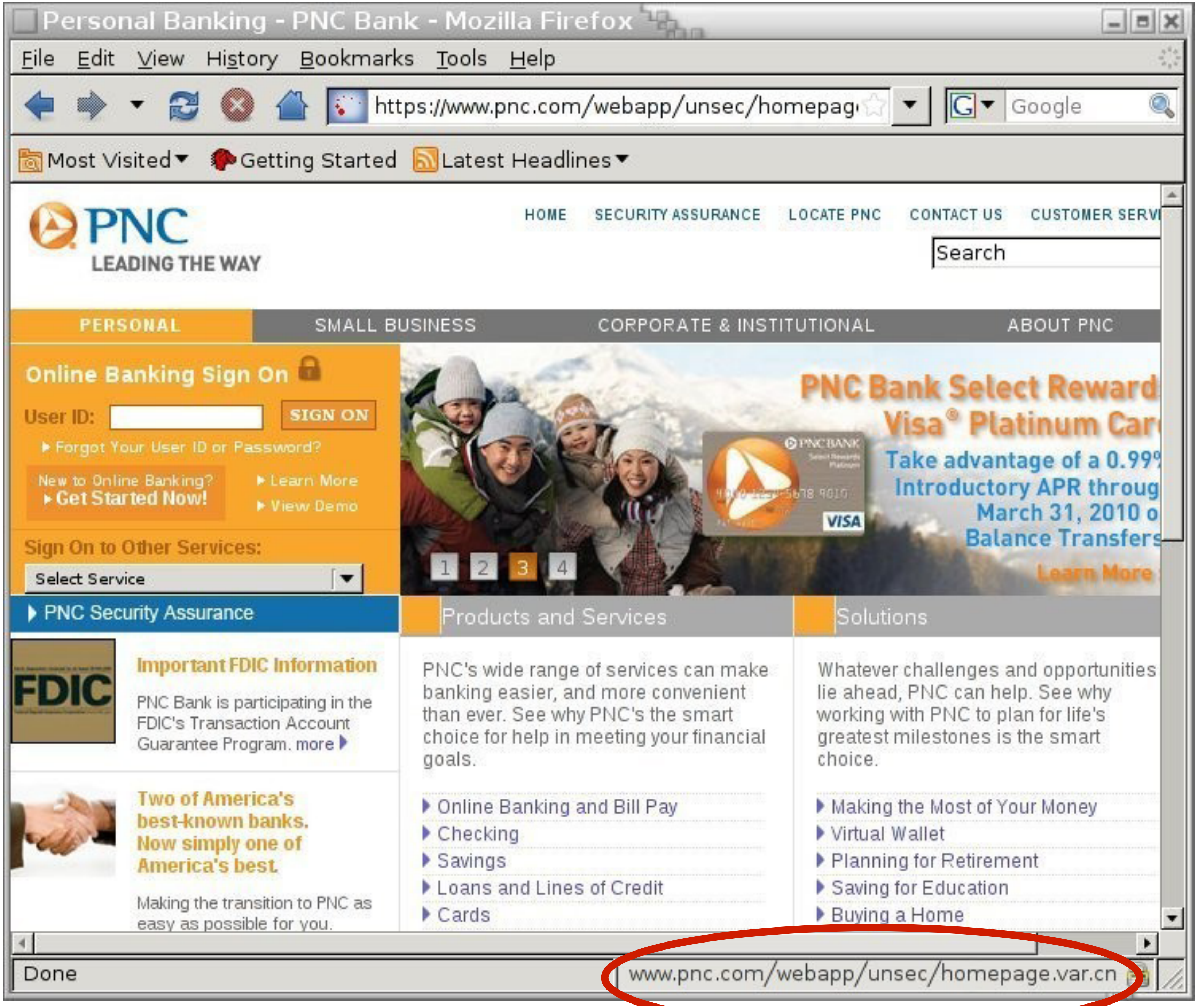
# Demo: Phishing via Browser Tab Manipulation Sneakiness

# The Problem of Phishing

- Arises due to mismatch between reality & user's:
  - Perception of how to assess legitimacy
  - Mental model of what attackers can control
    - Both Email and Web
- Coupled with:
  - Deficiencies in how web sites authenticate
    - In particular, "replayable" authentication that is vulnerable to theft
- Attackers have many angles …




1. Text and left-side pixels fully under attacker control
2. Domain name cannot be altered (but can be misleading!)
3. Path after the domain name fully under attacker control
4. All pixels fully under attacker control



Personal Banking - PNC Bank - Mozilla Firefox
File Edit View History Bookmarks Tools Help
https://www.pnc.com/webapp/unsec/homepag
Google
Most Visited
Getting Started
Latest Headlines
PNC
LEADING THE WAY
HOME
SECURITY ASSURANCE
LOCATE PNC
CONTACT US
CUSTOMER SERVI
Search
PERSONAL
SMALL BUSINESS
CORPORATE & INSTITUTIONAL
ABOUT PNC
Online Banking Sign On
User ID:
SIGN ON
Forgot Your User ID or Password?
New to Online Banking?
Get Started Now!
Learn More
View Demo
Sign On to Other Services:
Select Service
PNC Security Assurance
PNC Bank Select Reward
Visa® Platinum Car
Take advantage of a 0.99
Introductory APR throug
March 31, 2010 o
Balance Transfers
Learn More
1 2 3 4
FDIC
Important FDIC Information
PNC Bank is participating in the FDIC's Transaction Account Guarantee Program. more
Two of America's best-known banks. Now simply one of America's best.
Making the transition to PNC as easy as possible for you.
Products and Services
PNC's wide range of services can make banking easier, and more convenient than ever. See why PNC's the smart choice for help in meeting your financial goals.
Online Banking and Bill Pay
Checking
Savings
Loans and Lines of Credit
Cards
Solutions
Whatever challenges and opportunities lie ahead, PNC can help. See why working with PNC to plan for life's greatest milestones is the smart choice.
Making the Most of Your Money
Virtual Wallet
Planning for Retirement
Saving for Education
Buying a Home
Done
www.pnc.com/webapp/unsec/homepage.var.cn

# Homograph Attacks

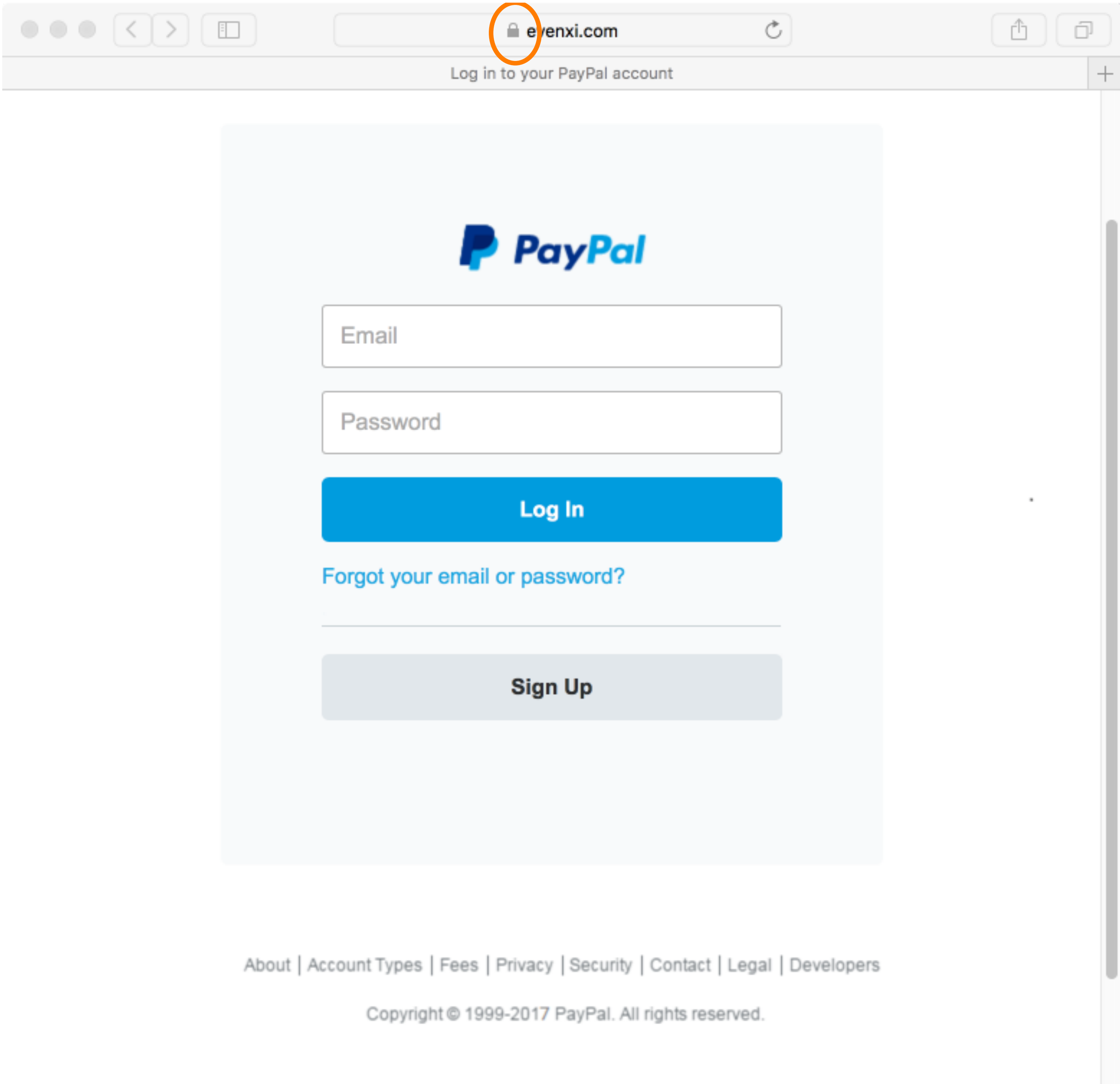
- International domain names can use international character set
  - E.g., Chinese contains characters that look like / . ? =
- **Attack**: Legitimately register var.cn …
- … buy legitimate set of HTTPS certificates for it …
- … and then create a subdomain:

  www.pnc.com⁄webapp⁄unsec⁄homepage.var.cn

This is one subdomain

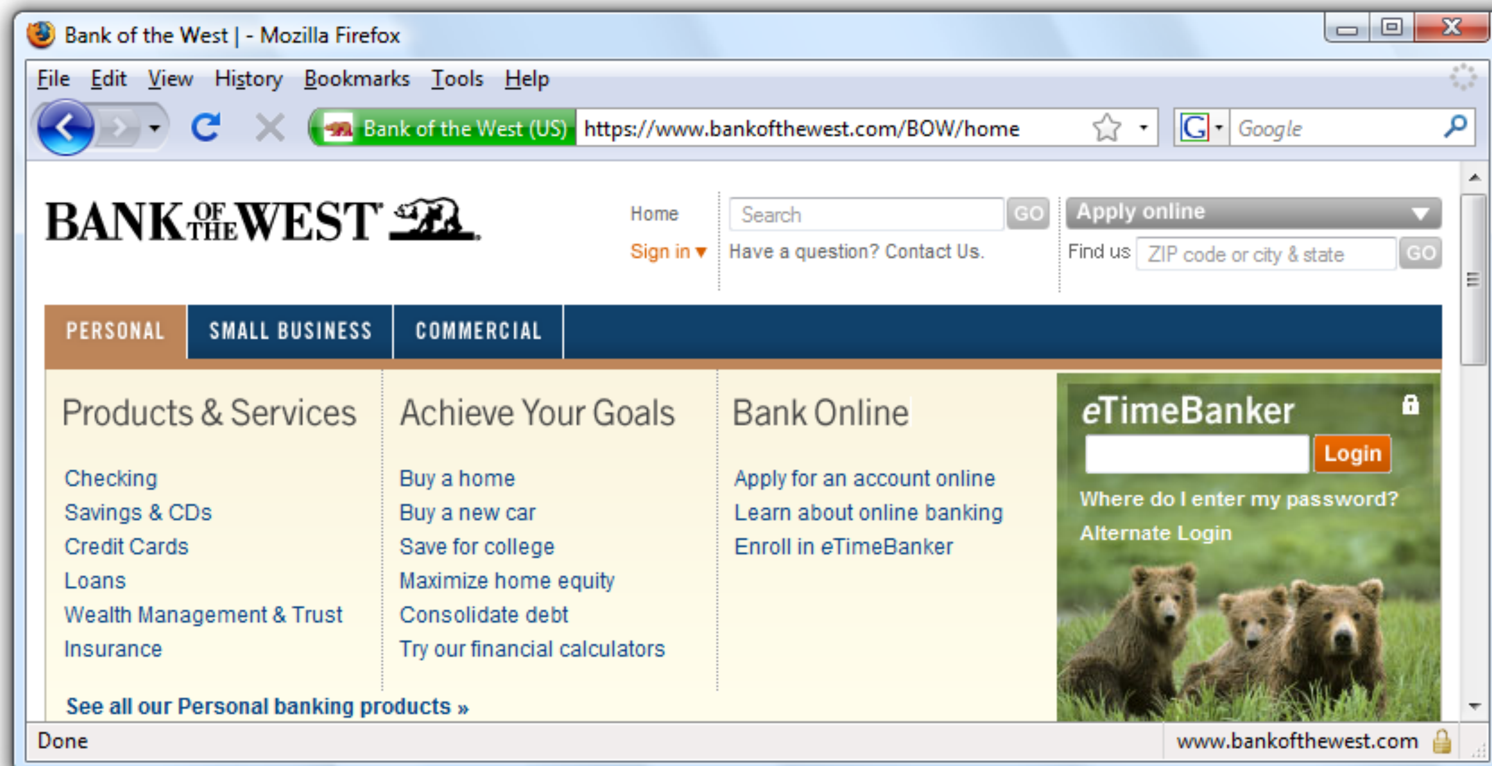
# Check for a padlock?


Wachovia - Personal Finance and Business Financial Services - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://www.wachovia.com/
WACHOVIA
LOGIN
User ID:
Remember my User ID
Password:
(case sensitive)
Service:
Choose a service...
Login
Forgot User ID or Password?
Retirement Plan Participants: Login
Education Loan Customers: Login
PERSONAL FINANCE
Online Services
Online Banking with BillPay
Mobile Banking
Online Brokerage
More...
Retirement Planning
Tools & information for
Lifetime Retirement Planning
Investing
Accounts & Services
IRAs
More...
Banking
Checking
Savings & CDs
Credit Cards
Check Cards
More...
Lending
Mortgage
Home Equity New!
Education Loans
Vehicle Loans
Rates

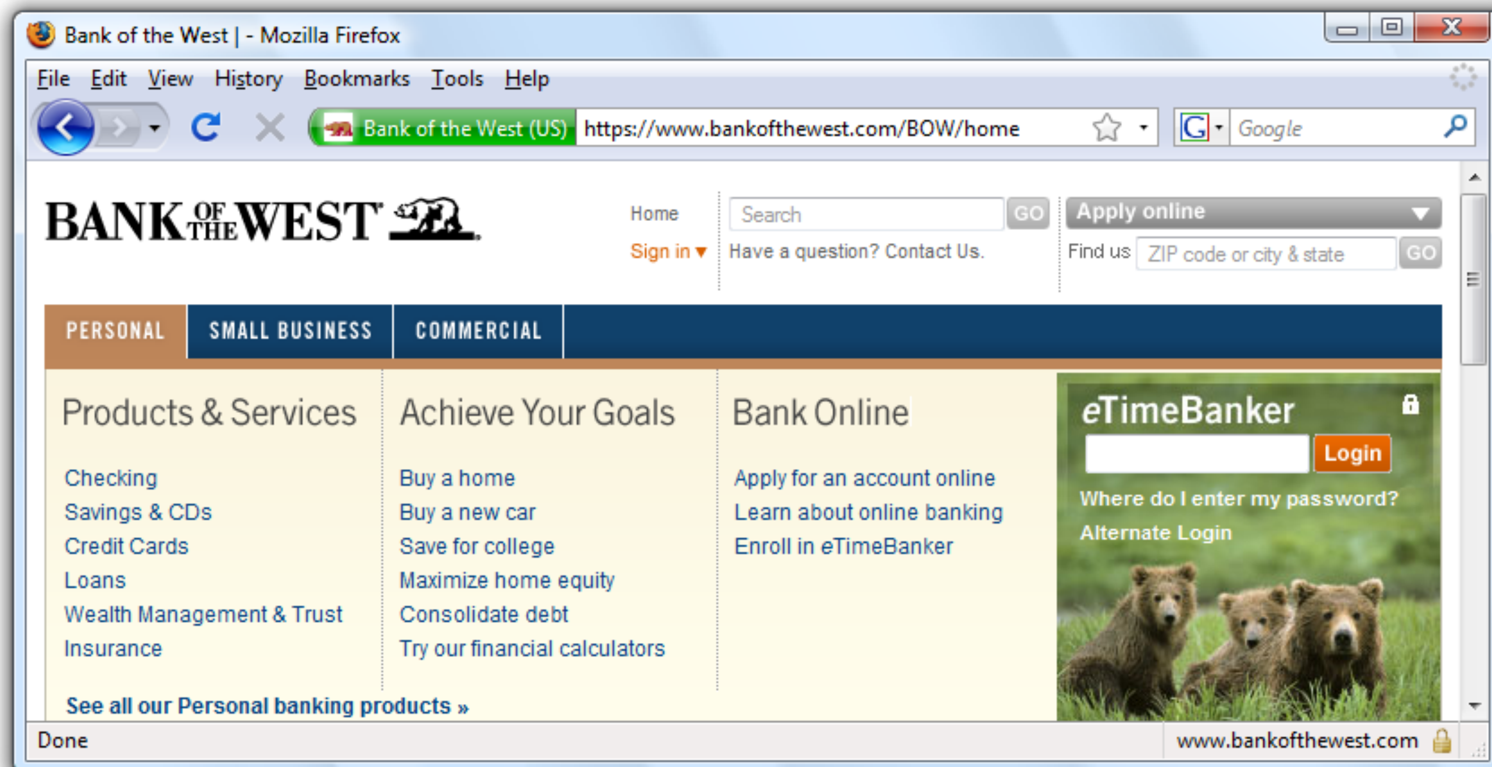


evenxi.com
Log in to your PayPal account
PayPal
Email
Password
Log In
Forgot your email or password?
Sign Up
About | Account Types | Fees | Privacy | Security | Contact | Legal | Developers
Copyright © 1999-2017 PayPal. All rights reserved.

# Check for "green glow" in address bar?



# Check for Everything?



# “Browser in Browser”


*Apparent* browser is just a ***fully interactive image*** generated by Javascript running in real browser!

# Why does phishing work?

- Because users are stupid?

# Why does phishing work?

- User mental model vs. reality
  - Browser security model too hard to understand!
- The easy path is insecure; the secure path takes extra effort
- Risks are rare
- Users tend not to suspect malice; they find benign interpretations and have been *acclimated to failure*

# Questions?

# Cryptography:

## Secure communication over insecure paths

(and/or:
Secure data storage on insecure servers)

# Three main goals

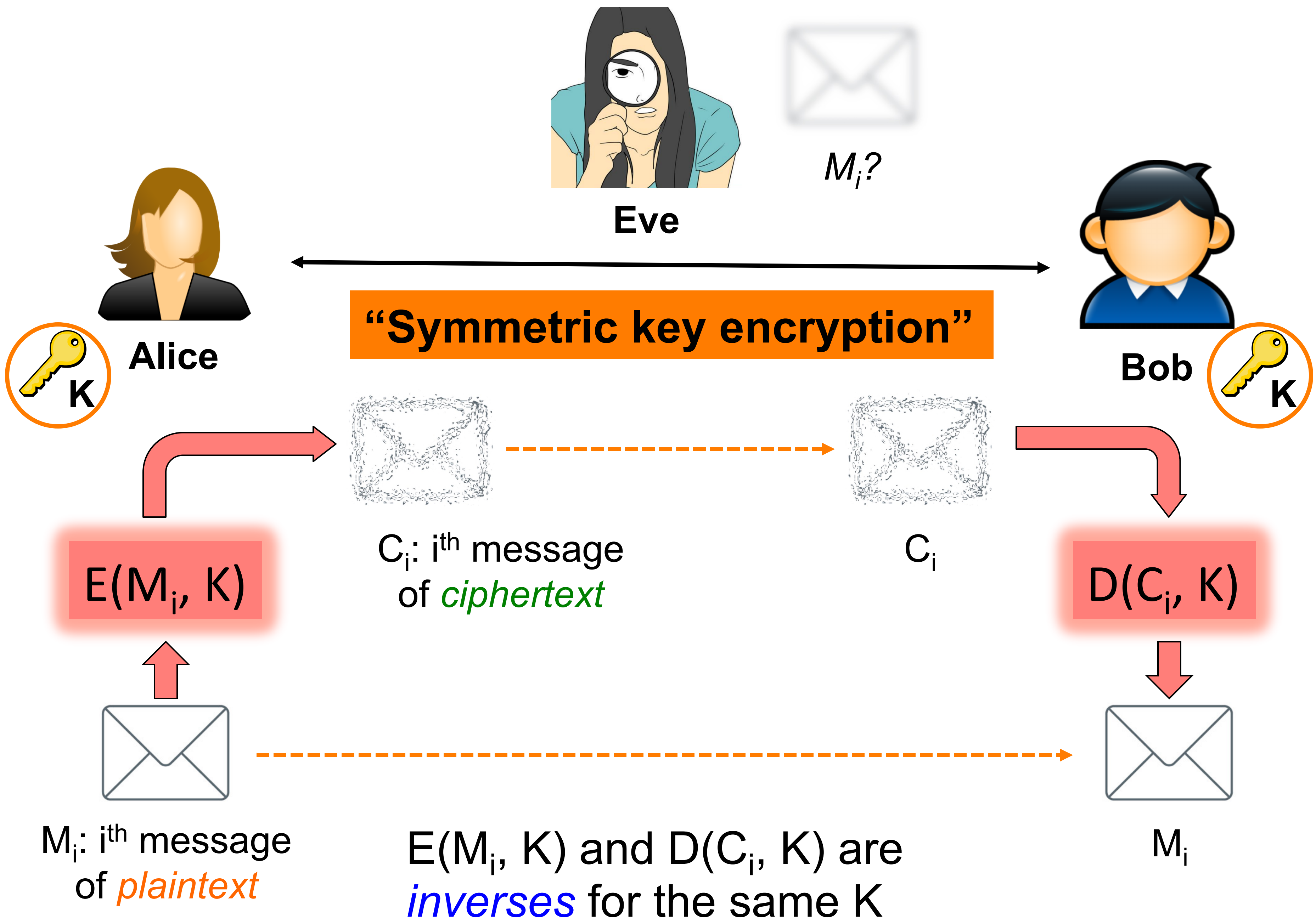
- Confidentiality: preventing adversaries from reading our private data
  - Data = message or document
- Integrity: preventing attackers from altering our data
  - Data itself *might or might not be private*
- Authentication: determining who created a given message or document
  - Generally implies/requires integrity

# Special guests

- Alice  (sender of messages)
- Bob  (receiver of messges)
- The attackers
  - Eve: “eavesdropper”
  - Mallory: “manipulator”

Eve

# Confidentiality


Eve
$M_i$?
Alice
K
"Symmetric key encryption"
Bob
K
$E(M_i, K)$
$C_i$: $i^{th}$ message of *ciphertext*
$C_i$
$D(C_i, K)$
$M_i$: $i^{th}$ message of *plaintext*
$E(M_i, K)$ and $D(C_i, K)$ are *inverses* for the same K
$M_i$

# The Ideal Contest

- **Attacker's goal:** *any* knowledge of $M_i$ beyond an upper bound on its length
  - Slightly better than 50% probability at guessing a single bit: attacker wins!
  - Any notion of how $M_i$ relates to $M_j$: attacker wins!

- **Defender's goal:** ensure attacker has no reason to think any $M' \in \{0,1\}^n$ is more likely than any other
  - (for $M_i$ of length n)

# Eve's Capabilities/Foreknowledge

- **No knowledge of K**
  - We assume K is selected by a truly random process
  - For b-bit key, any $K \in \{0,1\}^b$ is equally likely
- **Recognition of success:** Eve can *generally* tell if she has **correctly** and **fully** recovered $M_i$
  - But: Eve *cannot* recognize anything about *partial* solutions, such as whether she has correctly identified a particular bit in $M_i$
  - Does not apply to scenarios where Eve *exhaustively* examines every possible $M_i' \in \{0,1\}^n$

# Eve's Available Information

1. **Ciphertext-only attack:**
   – Eve gets to see *every* instance of $C_i$
   – Variant: Eve may also have partial information about $M_i$
      • "It's probably English text"
      • Bob is Alice's stockbroker, so it's either "Buy!" or "Sell"
2. **Known plaintext:**
   – Eve knows part of $M_i$ and/or entire other $M_j$'s
   – How could this happen?
      • E.g. encrypted HTTP request: starts with "`GET`"
      • E.g. Eve sees earlier message she knows Alice will send to Bob
      • E.g. Alice transmits in the clear and then resends encrypted

# Eve's Available Information, con't

3. **Chosen plaintext**
   - Eve gets Alice to send $M_j$'s of Eve's choosing
   - Example: Eve sends Alice an email spoofed from Alice's boss saying "Please securely forward this to Bob"
4. **Chosen ciphertext:**
   - Eve tricks Bob into decrypting some $C_j$' of her choice and he reveals something about the result
   - How could this happen?
     - E.g. repeatedly send ciphertext to a web server that will send back different-sized messages depending on whether ciphertext decrypts into something well-formatted
       - Or: measure *how long* it takes Bob to decrypt & validate

# Eve's Available Information, con't

**5. Combinations of the above**

- Ideally, we'd like to defend against this last, the most powerful attacker

- And: we can!, so we'll mainly focus on this attacker when discussing different considerations

# Designing Ciphers

- Clearly, the whole trick is in the design of E(M,K) and D(C,K)
- One very simple approach:

  $$E(M,K) = ROT_K(M);\ D(C,K) = ROT_{-K}(C)$$

  i.e., take each letter in M and "rotate" it K positions (with wrap-around) through the alphabet
- E.g., $M_i$ = "DOG", K = 3
  $C_i = E(M_i,K) = ROT_3$("DOG") = "GRJ"
  $D(C_i,K) = ROT_{-3}$("GRJ") = "DOG"
- "Caesar cipher"

# Attacks on Caesar Ciphers?

- **Brute force**: try *every possible value* of K
  - Work involved?
  - At most 26 “steps”

# Attacks on Caesar Ciphers?

- **Brute force**: try *every possible value* of K
  - Work involved?
  - At most 26 "steps"
- **Deduction**:
  - Analyze letter frequencies ("ETAOIN SHRDLU")
  - Known plaintext / guess possible words & confirm
    - E.g. "JCKN ECGUCT" =?

# Attacks on Caesar Ciphers?

- **Brute force**: try *every possible value* of K
  - Work involved?
  - At most 26 “steps”
- **Deduction**:
  - Analyze letter frequencies (“ETAOIN SHRDLU”)
  - Known plaintext / guess possible words & confirm
    - E.g. “JCKN ECGUCT” =? “HAIL CAESAR”

# Attacks on Caesar Ciphers?

- **Brute force**: try *every possible value* of K
  - Work involved?
  - At most 26 "steps"
- **Deduction**:
  - Analyze letter frequencies ("ETAOIN SHRDLU")
  - Known plaintext / guess possible words & confirm
    - E.g. "JCKN ECGUCT" =? "HAIL CAESAR" ⇒ K=2
  - Chosen plaintext
    - E.g. get a general to send "ALL QUIET", observe "YJJ OSGCR" ⇒ K=24

# 5 Minute Break

Questions Before We Proceed?

# Kerckhoffs' Principle

- Cryptosystems should remain secure even when attacker knows all internal details
  - Don't rely on security-by-obscurity
- Key should be only thing that must stay secret
- It should be easy to change keys

# Better Versions of Rot-K ?

- Consider $E(M,K) = \text{Rot-}\{K_1, K_2, \ldots, K_n\}(M)$
  - i.e., rotate first character by $K_1$, second character by $K_2$, up through $n^{th}$ character. Then start over with $K_1$, ...
  - $K = \{ K_1, K_2, \ldots, K_n \}$
- How well do previous attacks work now?
  - Brute force: key space is factor of $26^{(n-1)}$ larger
    - E.g., n = 7 ⇒ 300 million times as much work
  - Letter frequencies: need more ciphertext to reason about
  - Known/chosen plaintext: works just fine
- Can go further with "chaining", e.g., 2nd rotation depends on $K_2$ *and* first character of ciphertext
  - We just described 2,000 years of cryptography

# One-Time Pad

- Idea #1: use a different key for each message M
  - Different = completely independent
  - So: known plaintext, chosen plaintext, etc., *don't help attacker*
- Idea #2: make the key as long as M
- $E(M,K) = M \oplus K$ ($\oplus$ = XOR)

| $\oplus$ | **0** | **1** |
|---|---|---|
| **0** | 0 | 1 |
| **1** | 1 | 0 |

$X \oplus 0 = X$

$X \oplus X = 0$

$X \oplus Y = Y \oplus X$

$X \oplus (Y \oplus Z) = (X \oplus Y) \oplus Z$

# One-Time Pad

- Idea #1: use a different key for each message M
  - Different = completely independent
  - So: known plaintext, chosen plaintext, etc., *don't help attacker*
- Idea #2: make the key as long as M
- $E(M,K) = M \oplus K$   ($\oplus$ = XOR)
  $D(C,K) = C \oplus K$
  $= M \oplus K \oplus K = M \oplus 0 = M$

| $\oplus$ | **0** | **1** |
|---|---|---|
| **0** | 0 | 1 |
| **1** | 1 | 0 |

$X \oplus 0 = X$
$X \oplus X = 0$
$X \oplus Y = Y \oplus X$
$X \oplus (Y \oplus Z) = (X \oplus Y) \oplus Z$

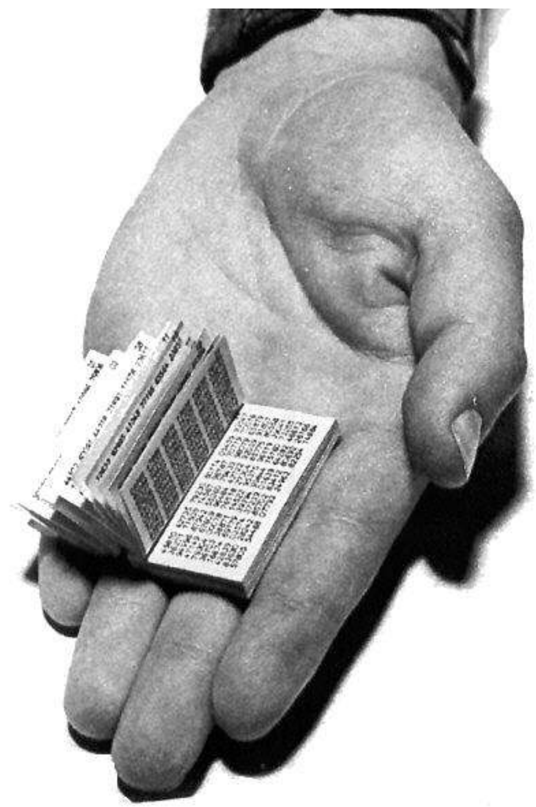
# One-Time Pad: Provably Secure!

- Let's assume Eve has partial information about M
- We want to show: from C, she does not gain any further information
- Formalization: supposed Alice sends either M' or M''
  - Eve doesn't know which; tries to guess based on C
- Proof:
  - For random, independent K, all possible bit-patterns for C are equally likely
  - This holds regardless of whether Alice chose M' or M''
  - Thus, observing a given C does not help Eve narrow down the possibilities in any way

# One-Time Pad: Provably Impractical!

- Problem #1: key generation
  - Need truly random, independent keys

- Problem #2: key distribution
  - Need to share keys as long as all possible communication
  - If we have a secure way to establish such keys, just use that for communication in the first place!



# Two-Time Pad?

- What if we reuse a key K *jeeeest* once?
- Alice sends $C = E(M, K)$ and $C' = E(M', K)$
- Eve observes $M \oplus K$ and $M' \oplus K$
  - Can she learn anything about M and/or M' ?
- Eve computes $C \oplus C' = (M \oplus K) \oplus (M' \oplus K)$

# Two-Time Pad?

- What if we reuse a key K *jeeeest* once?
- Alice sends C = E(M, K) and C' = E(M', K)
- Eve observes $M \oplus K$ and $M' \oplus K$
  - Can she learn anything about M and/or M' ?
- Eve computes $C \oplus C' = (M \oplus K) \oplus (M' \oplus K)$
  $= (M \oplus M') \oplus (K \oplus K)$
  $= (M \oplus M') \oplus 0$
  $= M \oplus M'$
- Now she knows which bits in M match bits in M'
- And if Eve already knew M, now she knows M' !

# Modern Symmetric-Key Encryption:

## *Block Ciphers*

# Block cipher

A function $E : \{0, 1\}^b \times \{0, 1\}^k \rightarrow \{0, 1\}^b$. Once we fix the key K (of size k bits), we get:

$E_K : \{0,1\}^b \rightarrow \{0,1\}^b$ denoted by $E_K(M) = E(M,K)$.

(and also D(C,K), E(M,K)'s inverse)

- Three properties:
  - Correctness:
    - $E_K(M)$ is a permutation (bijective function) on b-bit strings
    - Bijective ⇒ invertible
  - Efficiency: computable in $\mu$sec's
  - Security:
    - For unknown K, "behaves" like a random permutation
- Provides a *building block* for more extensive encryption