CSE 484 / CSE M 584
Computer Security: Cryptography

TA: Adrian Sham
adrsham@cs
Original slides by Franzi

[Examples/Images thanks to Wikipedia.]
Lab 1 Deadline Reminders

- Lab 1 Final due next week (5/1, 5pm).
- Upcoming office hours:
  - Tomorrow (Friday) 9:30 am – Michael & Adrian
  - Monday 9:30 am – Franzi
  - Wednesday 3:30 pm – Adrian & Peter
  - Thursday 12:30 pm – Peter & Michael
Illustration of DH as paint mixing

common paint

+ secret color A

assume that mixture separation is expensive

+ secret color A

public transport

common secret

+ secret color B

+ secret color B

Wikipedia image
**DH Summary**

- Public info: $p$ (large prime) and $g$ (generator of $Z_p^*$)
  
  $Z_p^* = \{1, 2, \ldots, p-1\}; \forall a \in Z_p^* \exists i$ such that $a = g^i \mod p$
RSA Summary

• 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 \( \varphi(n) \)
  – Compute unique \( d \) such that \( ed = 1 \mod \varphi(n) \)
  – Public key = \( (e,n) \); private key = \( (d,n) \)

• Encryption of \( m \): \( c = m^e \mod n \)
  – Modular exponentiation by repeated squaring

• Decryption of \( c \): \( c^d \mod n = (m^e)^d \mod n = m \)
Sample RSA Decryption

- 26 2 15 13 7 14 13 13 1 28 14 15 13 14 20 9 6 31 25 26 14 16 23 15 26 2 6 13 1
- p=3, q=11, n=33, e=7, d=3
Sample RSA Decryption

• How to compute \( d \)?
  – Recall: \( ed = 1 \mod \varphi(n) \) (where \( \varphi(n) = (p-1)(q-1) \))
  – So \( d \) is inverse of \( e \) mod \( \varphi(n) \).
  – How to compute modular inverse?
    • Use extended Euclidean algorithm
    • … or Wolfram Alpha 😊
    • Note that this is hard if you don’t know \( \varphi(n) \) (i.e., can’t factor \( n \)).
Public Key Crypto Summary

• **Diffie-Hellman:** *Why is it secure?*
  – Discrete log; computational DH problem; decisional DH problem are hard.

• **RSA:** *Why is it secure?*
  – Taking $e^{th}$ root is hard; Factoring is hard.
Cryptography Summary

• **Goal: Privacy**
  – One-time pad
  – Block ciphers w/ symmetric keys (e.g., DES, AES)
    • Modes: EBC, CBC, CTR
  – Public key crypto (e.g., Diffie-Hellman, RSA)

• **Goal: Integrity**
  – MACs, often using hash functions (e.g., MD5, SHA-256)

• **Goal: Privacy and Integrity**
  – Encrypt-then-MAC (why?)

• **Goal: Authenticity (and Integrity)**
  – Digital signatures (e.g., RSA, DSS)
Block Cipher Mode: ECB

Electronic Codebook (ECB) mode encryption
Block Cipher Mode: ECB

Electronic Codebook (ECB) mode decryption
ECB Pros and cons

- Encryption and decryption parallelizable
- Does not hide data patterns well, not recommended
Block Cipher Mode: CBC

Cipher Block Chaining (CBC) mode encryption
Block Cipher Mode: CBC

Cipher Block Chaining (CBC) mode decryption
CBC Pros and cons

• Encryption not parallelizable
• Decryption is parallelizable
Block Cipher Mode: CTR

Counter (CTR) mode encryption
Block Cipher Mode: CTR

Counter (CTR) mode decryption
Pros and cons

• Encryption and decryption parallelizable
• CBC and CTR usage recommended by Yoshi, Niels and Bruce Schneier! (Cryptography Engineering, 2010)
CBC-MAC question

Given a message $M$ with tag $T$ (aka $\text{CBC-MAC}(M)=T$), can you construct a message $M'$ (not necessarily the same length as $M$) for which the tag is *also* $T$, aka $\text{CBC-MAC}(M')=T$?
Password Salting

• Servers shouldn’t store passwords, but password hashes. (*Why?*)

• Threat: *rainbow tables* (pre-computed password hashes)

• Solution: *salt*
  
  – Each password is hashed/stored with a random value. Now a pre-computed table is useless.

  – Other benefits?
Real world example, by xkcd

Hackers recently leaked 153 million Adobe user emails, encrypted passwords, and password hints.

Adobe encrypted the passwords improperly, misusing block-mode 3DES. The result is something wonderful:

<table>
<thead>
<tr>
<th>USER PASSWORD</th>
<th>HINT</th>
</tr>
</thead>
<tbody>
<tr>
<td>4e18acc1ab2626</td>
<td>WEATHER VANE SWORD</td>
</tr>
<tr>
<td>4e18acc1ab2626</td>
<td>NAME_1</td>
</tr>
<tr>
<td>3abb667e06b6d1</td>
<td>DUH</td>
</tr>
<tr>
<td>8abb667e06b6d1</td>
<td>57</td>
</tr>
<tr>
<td>8abb667e06b6d1</td>
<td>FAVORITE OF 12 APOSTLES WITH YOUR OWN HAND YOU HAVE DONE ALL THIS</td>
</tr>
<tr>
<td>4e18acc1ab2626</td>
<td>SUGARLAND</td>
</tr>
<tr>
<td>1eb2a8666666666666660</td>
<td>NAME + JERSEY #</td>
</tr>
<tr>
<td>877a78999396261</td>
<td>ALPHA</td>
</tr>
<tr>
<td>877a78999396261</td>
<td>OBVIOUS</td>
</tr>
<tr>
<td>877a78999396261</td>
<td>MICHAEL JACKSON</td>
</tr>
<tr>
<td>38a7c27f7c6e4bd4491bc0d75d34c6d5</td>
<td>HE DID THE MASH, HE DID THE PURLOINED</td>
</tr>
<tr>
<td>38a7c27f7c6e4bd4491bc0d75d34c6d5</td>
<td>FALL WINTER-3 POKEMON</td>
</tr>
</tbody>
</table>

The greatest crossword puzzle in the history of the world

HTTP://XKCD.COM/1286/
Additional Resources

• Stanford online crypto class: https://class.coursera.org/crypto-preview/class

• Books:
  – “The Codebreakers” by David Kahn