CS 356 – Lecture 4
User Authentication

Fall 2013
Review

• Chapter 1: Basic Concepts and Terminology
  – Integrity, Confidentiality, Availability, Authentication, and Accountability
  – Types of threats: active vs. passive, insider/outsider
  – Lots of terminology and general concepts

• Chapter 2: Basic Cryptographic Tools
  – Symmetric key encryption and secure hashing
  – Public key cryptography
  – Random Numbers

• Chapter 3 – User Authentication
It’s Time to Play....

CyberSecurity Edition!
Cyber-Security in the News!

Car hacking code released at Defcon

Car computer hacking hit the gas on the first morning of Defcon 21, as hackers revealed how they took over two of the most popular cars in America -- a Chevrolet, and a Toyota Prius -- within a minute of hitting the event.

Hackers demonstrated from the audience how they could control a car's computer systems through a USB drive, which exploited flaws in the car's software.

Some of the car hacking tricks include:

- **Blast the horn, continuing even after car turned off**
- **Prevent car from powering down, draining battery**
- **Change speedometer and gas gauge at will**
- **Abruptly tighten driver's and passenger's seat belt**
- **Cause engine to accelerate (can be overridden with the brake)**
- **Turn headlights on or off when lights left on auto**
- **Disable power steering or jerk the wheel**
- **Slam on brakes at any speed**
Chapter 3

User Authentication
"On the Internet, nobody knows you're a dog."
RFC 2828 defines user authentication as:

“The process of verifying an identity claimed by or for a system entity.”
Authentication Process

- fundamental building block and primary line of defense
- basis for access control and user accountability

- identification step
  - presenting an identifier to the security system

- verification step
  - presenting or generating authentication information that corroborates the binding between the entity and the identifier
### User Authentication

<table>
<thead>
<tr>
<th>something the individual knows</th>
<th>something the individual possesses (token)</th>
<th>something the individual is (static biometrics)</th>
<th>something the individual does (dynamic biometrics)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• password, PIN, answers to prearranged questions</td>
<td>• smartcard, electronic keycard, physical key</td>
<td>• fingerprint, retina, face</td>
<td>• voice pattern, handwriting, typing rhythm</td>
</tr>
</tbody>
</table>
Password Authentication

- widely used line of defense against intruders
  - user provides name/login and password
  - system compares password with the one stored for that specified login

- the user ID:
  - determines that the user is authorized to access the system
  - determines the user’s privileges
  - is used in discretionary access control
Password Vulnerabilities

- offline dictionary attack
- specific account attack
- popular password attack
- password guessing against single

- electronic monitoring
- exploiting multiple password use
- exploiting user mistakes
- workstation hijacking
Countermeasures
Countermeasures

• controls to prevent unauthorized access to password file
Countermeasures

- controls to prevent unauthorized access to password file
- intrusion detection measures
Countermeasures

- controls to prevent unauthorized access to password file
- intrusion detection measures
- rapid re-issuance of compromised passwords
Countermeasures

- controls to prevent unauthorized access to password file
- intrusion detection measures
- rapid re-issuance of compromised passwords
- account lockout mechanisms
Countermeasures

- controls to prevent unauthorized access to password file
- intrusion detection measures
- rapid re-issuance of compromised passwords
- account lockout mechanisms
- policies to inhibit users from selecting common passwords
Countermeasures

- controls to prevent unauthorized access to password file
- intrusion detection measures
- rapid re-issuance of compromised passwords
- account lockout mechanisms
- policies to inhibit users from selecting common passwords
- training in and enforcement of password policies
Countermeasures

- controls to prevent unauthorized access to password file
- intrusion detection measures
- rapid re-issuance of compromised passwords
- account lockout mechanisms
- policies to inhibit users from selecting common passwords
- training in and enforcement of password policies
- automatic workstation logout
Countermeasures

- controls to prevent unauthorized access to password file
- intrusion detection measures
- rapid re-issuance of compromised passwords
- account lockout mechanisms
- policies to inhibit users from selecting common passwords
- training in and enforcement of password policies
- automatic workstation logout
- policies against similar passwords on network devices
Use of Hashed Passwords

Figure 3.1 UNIX Password Scheme
UNIX Implementation

original scheme
• up to eight printable characters in length
• 12-bit salt used to modify DES encryption into a one-way hash function
• zero value repeatedly encrypted 25 times
• output translated to 11 character sequence

now regarded as inadequate
• still often required for compatibility with existing account management software or multivendor environments
Improved Implementations

- much stronger hash/salt schemes available for Unix
- recommended hash function is based on MD5 salt of up to 48-bits
  - password length is unlimited
  - produces 128-bit hash
  - uses an inner loop with 1000 iterations to achieve slowdown
- OpenBSD uses Blowfish block cipher based hash algorithm called Bcrypt
  - most secure version of Unix hash/salt scheme
  - uses 128-bit salt to create 192-bit hash value
Password Cracking

• **dictionary attacks**
  – develop a large dictionary of possible passwords and try each against the password file
  – each password must be hashed using each salt value and then compared to stored hash values

• **rainbow table attacks**
  – pre-compute tables of hash values for all salts
  – a mammoth table of hash values
  – can be countered by using a sufficiently large salt value and a sufficiently large hash length
# Observed Password Lengths

<table>
<thead>
<tr>
<th>Length</th>
<th>Number</th>
<th>Fraction of Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>55</td>
<td>.004</td>
</tr>
<tr>
<td>2</td>
<td>87</td>
<td>.006</td>
</tr>
<tr>
<td>3</td>
<td>212</td>
<td>.02</td>
</tr>
<tr>
<td>4</td>
<td>449</td>
<td>.03</td>
</tr>
<tr>
<td>5</td>
<td>1260</td>
<td>.09</td>
</tr>
<tr>
<td>6</td>
<td>3035</td>
<td>.22</td>
</tr>
<tr>
<td>7</td>
<td>2917</td>
<td>.21</td>
</tr>
<tr>
<td>8</td>
<td>5772</td>
<td>.42</td>
</tr>
<tr>
<td>Total</td>
<td>13787</td>
<td>1.0</td>
</tr>
</tbody>
</table>
## Passwords Cracked from a Sample Set of 13,797 Accounts

<table>
<thead>
<tr>
<th>Type of Password</th>
<th>Search Size</th>
<th>Number of Matches</th>
<th>Percentage of Passwords Matched</th>
<th>Cost/Benefit Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>User/account name</td>
<td>130</td>
<td>368</td>
<td>2.7%</td>
<td>2.830</td>
</tr>
<tr>
<td>Character sequences</td>
<td>866</td>
<td>22</td>
<td>0.2%</td>
<td>0.025</td>
</tr>
<tr>
<td>Numbers</td>
<td>427</td>
<td>9</td>
<td>0.1%</td>
<td>0.021</td>
</tr>
<tr>
<td>Chinese</td>
<td>392</td>
<td>56</td>
<td>0.4%</td>
<td>0.143</td>
</tr>
<tr>
<td>Place names</td>
<td>628</td>
<td>82</td>
<td>0.6%</td>
<td>0.131</td>
</tr>
<tr>
<td>Common names</td>
<td>2239</td>
<td>548</td>
<td>4.0%</td>
<td>0.245</td>
</tr>
<tr>
<td>Female names</td>
<td>4280</td>
<td>161</td>
<td>1.2%</td>
<td>0.038</td>
</tr>
<tr>
<td>Male names</td>
<td>2866</td>
<td>140</td>
<td>1.0%</td>
<td>0.049</td>
</tr>
<tr>
<td>Uncommon names</td>
<td>4955</td>
<td>130</td>
<td>0.9%</td>
<td>0.026</td>
</tr>
<tr>
<td>Myths and legends</td>
<td>1246</td>
<td>66</td>
<td>0.5%</td>
<td>0.053</td>
</tr>
<tr>
<td>Shakespearean</td>
<td>473</td>
<td>11</td>
<td>0.1%</td>
<td>0.023</td>
</tr>
<tr>
<td>Sports terms</td>
<td>238</td>
<td>32</td>
<td>0.2%</td>
<td>0.134</td>
</tr>
<tr>
<td>Science fiction</td>
<td>691</td>
<td>59</td>
<td>0.4%</td>
<td>0.085</td>
</tr>
<tr>
<td>Movies and actors</td>
<td>99</td>
<td>12</td>
<td>0.1%</td>
<td>0.121</td>
</tr>
<tr>
<td>Cartoons</td>
<td>92</td>
<td>9</td>
<td>0.1%</td>
<td>0.098</td>
</tr>
<tr>
<td>Famous people</td>
<td>290</td>
<td>55</td>
<td>0.4%</td>
<td>0.190</td>
</tr>
<tr>
<td>Phrases and patterns</td>
<td>933</td>
<td>253</td>
<td>1.8%</td>
<td>0.271</td>
</tr>
<tr>
<td>Surnames</td>
<td>33</td>
<td>9</td>
<td>0.1%</td>
<td>0.273</td>
</tr>
<tr>
<td>Biology</td>
<td>58</td>
<td>1</td>
<td>0.0%</td>
<td>0.017</td>
</tr>
<tr>
<td>System dictionary</td>
<td>19683</td>
<td>1027</td>
<td>7.4%</td>
<td>0.052</td>
</tr>
<tr>
<td>Machine names</td>
<td>9018</td>
<td>132</td>
<td>1.0%</td>
<td>0.015</td>
</tr>
<tr>
<td>Mnemonics</td>
<td>14</td>
<td>2</td>
<td>0.0%</td>
<td>0.143</td>
</tr>
<tr>
<td>King James bible</td>
<td>7525</td>
<td>83</td>
<td>0.6%</td>
<td>0.011</td>
</tr>
<tr>
<td>Miscellaneous words</td>
<td>3212</td>
<td>54</td>
<td>0.4%</td>
<td>0.017</td>
</tr>
<tr>
<td>Yiddish words</td>
<td>56</td>
<td>0</td>
<td>0.0%</td>
<td>0.000</td>
</tr>
<tr>
<td>Asteroids</td>
<td>2407</td>
<td>19</td>
<td>0.1%</td>
<td>0.007</td>
</tr>
<tr>
<td>TOTAL</td>
<td>62727</td>
<td>3340</td>
<td>24.2%</td>
<td>0.053</td>
</tr>
</tbody>
</table>

*Computed as the number of matches divided by the search size. The more words that need to be tested for a match, the lower the cost/benefit ratio.*
Password File Access Control

- can block offline guessing attacks by denying access to encrypted passwords
  - make available only to privileged users
  - shadow password file
    - a separate file from the user IDs where the hashed passwords are kept

vulnerabilities

- weakness in the OS that allows access to the file
- accident with permissions making it readable
- users with same password on other systems
- access from backup media
- sniff passwords in network traffic
**Password Selection Techniques**

**user education**
- users can be told the importance of using hard to guess passwords and can be provided with guidelines for selecting strong passwords

**computer generated passwords**
- users have trouble remembering them

**reactive password checking**
- system periodically runs its own password cracker to find guessable passwords

**proactive password checking**
- user is allowed to select their own password, however the system checks to see if the password is allowable, and if not, rejects it
- goal is to eliminate guessable passwords while allowing the user to select a password that is memorable
Proactive Password Checking

rule enforcement
• specific rules that passwords must adhere to

password cracker
• compile a large dictionary of passwords not to use

Bloom filter
• used to build a table based on dictionary using hashes
• check desired password against this table
Figure 3.2  Performance of Bloom Filter
What’s Next

• Read Chapter 1, 2, and 3
  – Chap 1: Focus on big picture and recurring concepts
  – Chap 2: Identify cryptographic tools and properties
  – Chap 3: How can you authenticate a user?

• Homework  Posted on Course Website
  – Due Tuesday

• Next Lecture Topics from Chapter 3
  – Biometrics, Smart Cards, Tokens

Thursday, September 5, 13