Computers/Internet play a vital role in our daily lives

Social Networks and Online Communities
- facebook, flickr, file sharing, etc.

Privacy threats abound (identity fraud, etc.)
- Online activities can be easily tracked

Open (anonymous) communication
- Tunisia, China, etc.

Secure Web Transactions
- Shopping, medical records, human resources, etc.

Multi-disciplinary solutions
- e.g., Forensics covers Ethics, Law, Policy, Technology,...
Computer Security

• Security is often advertised in the abstract
  – “The system is secure”
  – “Our product makes your networks secure”
  – “Your Internet transactions are secure”

• For security professionals, the key questions we should ask are
  – secure from whom?
  – secure from what?

Computer Security

• Understanding how to assess the “security” of a system requires that we understand
  – what assumptions are being made
  – what a particular security technology does (or does not do)
  – what design decisions (conscious or not) were made about attacks it was designed to prevent
  – what security metrics were applied
  – what types of threats it ignores
  – . . .
Security is never black or white; context matters more than technology

- different security technologies play important roles in an overall security solution
- it might be secure against a certain type of adversary (the average criminal vs a national intelligence agency)
- it might be secure as long as certain advances don’t occur, or for a certain period of time

“Secure” is meaningless without context

The unchanging landscape

- Cyberspace isn’t all that different from the physical world
  - people interact with each other, form complex social relationships, have communities, both large and small
  - it is filled with commerce
- Threats in digital world mirror “real” world
  - theft, racketeering, vandalism, exploitation, con games, fraud, etc.
- Attacks will be similar to that in physical world
The unchanging landscape

• Where there is money, there are criminals
• Privacy violations aren’t new either
  – lots of legal paperwork is public record (e.g., real estate transactions, criminal judgments)
  – private investigators use such data routinely to track down individuals; marketers use it to target particular demographics
• Privacy violations can easily lead to fraud
• Some “violations” are difficult to detect

Example: Cookie Tracking
How to control one’s online privacy?

- Opt out, somehow?
  - Regularly check and delete cookies?
  - use “private browsing”
- Use 3rd party add-ons (e.g., TrackerScan)
- Advocate for do-not-track regulation?

Controlling one’s online footprint is more complicated than it needs to be ... primarily because entire new industries for selling users’ online information are springing up.
Measuring “Security”

• Lets look at three cases:
  – password authentication
  – intrusion detection systems
  – cryptography (break the cryptogram!)
E.g., 1: Password Authentication

- Passwords are a widely used user authentication method
- Authenticates ID of user logging and
  - that the user is authorized to access system
  - determines the user’s privileges
  - used in discretionary access control

Password generation advice?

I AM MORDAK, THE PREVENTER OF INFORMATION SERVICES. I BRING NEW GUIDELINES FOR PASSWORDS.

"All passwords must be at least six characters long... Include numbers and letters... Include a mix of upper and lower case..."

"Use different passwords for each system. Change once a month. Do not write anything down!"

"All passwords must be at least six characters long... Include numbers and letters... Include a mix of upper and lower case..."
Password generation advice?

- don’t use words in a dictionary
- composition matters (e.g., digits, special characters)
- choose mnemonic-based passwords which are memorable
- size matters (longer passwords are better)
  - how many 12 char passwords do you have?
- don’t write it down
- don’t share it with anyone
- expire frequently (like Onyen); change it often
- don’t re-use.
  - how many website passwords do you have?
- ....

What makes a good password?

- Password length?
  - 64 bits of randomness is hard to crack
  - 64 bits is ~20 “common” ascii characters
  - but, people can’t remember random strings!
- Pass phrases?
  - English text has roughly 1.3 random bits/char
  - so 50 letters of English text
  - hard to type without making mistakes!
- In practice
  - non-dictionary, mixed case, mixed alphanumeric
Measuring password strength

Password space

- number of $n$-character passwords given $c$ choices per character is $c^n$
  - usually expressed as base-2 logarithm
Password space

- Time required to search: $T = c^n \cdot t \cdot y$
  - $t =$ number of times password mapping is iterated
    - e.g., $t = 25$ on unix systems
  - $y =$ the time per iteration (e.g., $y = 1/125000$ sec)

<table>
<thead>
<tr>
<th>$n$</th>
<th>26 (lowercase)</th>
<th>36 (lowercase alphanumeric)</th>
<th>62 (mixed case alphanumeric)</th>
<th>95 (keyboard characters)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.67 hr</td>
<td>3.4 hr</td>
<td>51 hr</td>
<td>430 hr</td>
</tr>
<tr>
<td>6</td>
<td>17 hr</td>
<td>120 hr</td>
<td>130 dy</td>
<td>4.7 yr</td>
</tr>
<tr>
<td>7</td>
<td>19 dy</td>
<td>180 dy</td>
<td>22 yr</td>
<td>440 yr</td>
</tr>
<tr>
<td>8</td>
<td>1.3 yr</td>
<td>18 yr</td>
<td>1400 yr</td>
<td>42000 yr</td>
</tr>
<tr>
<td>9</td>
<td>34 yr</td>
<td>640 yr</td>
<td>86000 yr</td>
<td>$4.0 \times 10^7$ yr</td>
</tr>
<tr>
<td>10</td>
<td>890 yr</td>
<td>23000 yr</td>
<td>$5.3 \times 10^8$ yr</td>
<td>$3.8 \times 10^8$ yr</td>
</tr>
</tbody>
</table>

That’s odd. Why then is password cracking still so successful?

Password space: a closer look

- The choices within the space are not equiprobable as user-selected passwords
- Most users selected passwords from a small subset of the full password space
  - many of which can be uncovered by trying words from a list (so-called dictionary attack)
- Implication $\Rightarrow$ Exhaustive search as a metric for security is misleading here
UNC Onyen – “Only Name You’ll Ever Need”
- Broadly used by UNC faculty, staff, students, and employees of UNC hospitals
- Widely used at UNC for private services such as email, access to payroll management, etc.

### Transform Sets Considered

**Common Belief/Assumption:** Enforcing password expiration is helpful from a security standpoint

<table>
<thead>
<tr>
<th>Transform set</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Distance</td>
<td><em>password</em> → <em>p</em>assword</td>
</tr>
<tr>
<td>Edit Distance with Substring Moves</td>
<td><em>password</em> → <em>word</em>assword</td>
</tr>
<tr>
<td>Location Independent Transforms</td>
<td><em>Hand crafted, 8 subsets, only 534 primitive transforms</em></td>
</tr>
<tr>
<td>Pruned Location Independent Transforms</td>
<td><em>Top 50 transforms of location independent transforms</em></td>
</tr>
</tbody>
</table>
Location Independent Transforms

<table>
<thead>
<tr>
<th>CATEGORY</th>
<th>EXAMPLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capitalization</td>
<td>tarheels#1 → tArheels#1</td>
</tr>
<tr>
<td>Deletion</td>
<td>tarheels#1 → tarheels1</td>
</tr>
<tr>
<td>Duplication</td>
<td>tarheels#1 → tarheels11</td>
</tr>
<tr>
<td>Substitution</td>
<td>tarheels#1 → tarheels#2</td>
</tr>
<tr>
<td>Insertion</td>
<td>tarheels#1 → tarheels#12</td>
</tr>
<tr>
<td>Leet Transform</td>
<td>tarheels#1 → t@rheels#1</td>
</tr>
<tr>
<td>Block Move</td>
<td>tarheels#1 → #tarheels1</td>
</tr>
<tr>
<td>Keyboard Transform</td>
<td>tarheels#1 → tarheels#!</td>
</tr>
</tbody>
</table>

51141 hashes from 10374 defunct Onyen accounts
- 4 to 15 hashes per account in temporal order
- Hashes are provided without plaintext passwords

After 8 months, 31074 hashes (60.8%) were cracked for 7936 Onyen accounts (76.5%)

Learn from history
- History of transform is strong predictor of future use
- Given old password, 40% of future passwords cracked in under 3 secs!
Example 2: Intrusion Detection

Terminology

• **Virus**: code that replicates a possibly evolved copy of itself.

• **Worms**: network viruses, primarily replicated on computer networks.
  - typically executes itself *automatically* on a remote machine with user intervention.
  - (mass mailer worms are an exception)
Terminology

- **Trojan horses**: typically try and interest the user with some useful functionality to entice the user to run a program.

- These malicious software are called **malware**

Propagation strategies

- Hit-list and/or topological scanning
- Social-engineering
- Web-based malware (drive-by downloads)
- Exploiting social-networks (e.g. KoobFace botnet)
- Malicious documents (flash, pdf, etc)
- . .
Propagation

- Slammer worm (2003): doubling time of ~8.5 seconds. Peaked at ~3mins
- >55 million IP scans/sec
- 90% of internet scanned in <10 mins

Worm Detection

- Signature inference: automatically learn the content “signature” for a new outbreak

**Example:**
- monitor network and look for strings common to traffic with “worm-like” behavior
- Build signatures that can then be used for content filtering

**Signature:** A payload content string specific to some malware
Example: Content Sifting

- Assume there exists some relatively unique invariant bitstring \( W \) across all instances of a particular worm
- Two consequences:
  - **Content Prevalence**: \( W \) will be more common in traffic than other bitstrings of same length
  - **Address Dispersion**: the set of packets containing \( W \) will address a disproportionate number of sources and destinations
- **Content Sifting**: find Ws with high content prevalence and high address dispersion ⇒ drop traffic

---

Content-based Blocking

**Signature for CodeRed II**

- Can be used by intrusion detection systems
Evaluation

- Standard measures:
  - Detection rate: ratio between the number of correctly detected attacks and the total number of attacks
  - False alarm (false positive) rate: ratio between the number of normal connections that are incorrectly misclassified as attacks and the total number of normal connections
  - ...
Detection Rate vs False Alarm Rate

• Suppose 1% of traffic is attack traffic; Detector accuracy is 90%
  – i.e., classifies a valid connection as attack with prob. 10%
• What is the probability that a connection flagged by the detector as an attack is actually valid?

\[
\Pr(\text{valid} | \text{alarm}) = \frac{\Pr(\text{alarm} | \text{valid}) \cdot \Pr(\text{valid})}{\Pr(\text{alarm})} = \frac{\Pr(\text{alarm} | \text{valid}) \cdot \Pr(\text{valid})}{\Pr(\text{alarm} | \text{valid}) \cdot \Pr(\text{valid}) + \Pr(\text{alarm} | \text{Attack}) \cdot \Pr(\text{Attack})}
\]

\[
= \frac{0.10 \cdot 0.99}{0.10 \cdot 0.99 + 0.90 \cdot 0.01} = 92\% \text{ chance that raised alarm is false positive!}
\]

Example 3: Cryptography

• The study of secret (crypto-) writing (-graphy)

• Concerned with developing algorithms:
  – that conceal the context of some message from all except the intended parties (privacy or secrecy)
  – that verify the correctness of a message to the recipient (authentication)
Classical cryptography

- Ancient ciphers
  - have a history of at least 4000 years
  - ancient Egyptians enciphered some of their hieroglyphic writings on monuments
  - ancient Hebrews enciphered certain words in scriptures
  - over 2000 years ago Julius Caesar purportedly used a simple substitution cipher
  - English Philosopher Roger Bacon described several methods in 1200s

Basic concepts

- Plaintext
  - the original intelligible message
- Ciphertext
  - the transformed message
- Cipher
  - an algorithm for transforming an intelligible message into unintelligible by transposition and/or substitution
- Key
  - critical information used by the cipher, known only to the sender and receiver
Basic concepts

- **Encryption** (encipher)
  - the process of converting plaintext to ciphertext

- **Decryption** (decipher)
  - the process of converting ciphertext back to plaintext

- **Cryptanalysis**
  - the study of principles and methods of transforming an unintelligible message back into an intelligible message *without* knowledge of the key
• If both keys are the same, we have a **symmetric** cryptosystem
  – e.g., Data Encryption Standard
• If one key is inverse of the other, we have an **asymmetric** cryptosystem
  – e.g., public-key cryptography

How do we analyze crypto systems?
How do we analyze crypto systems?

- High-level: if the adversary intercepts the ciphertext, s/he cannot recover plaintext
- Issues in making this precise
  - What might your enemy know?
    - The kind of encryption function you are using?
    - Old plaintext-ciphertext pairs?
    - Information about how you chose keys?
- What does “cannot recover plaintext” mean?

On recovering plaintext

Natural language is highly redundant:

Aocdrcnig to rscheearch at Cmabrigde Uinervtisy, it deosn't mtaer in waht oredr the ltteers in a wrod are, the olny iprmoetnt tihng is taht the frist and lsat ltteer be at the rghit pclae. The rset can be a toatl mses and you can sitll raed it wouthit a porbelm. Tihs is bcuseae the huamn mnid deos not raed ervey lteter by istlef, but the wrod as a wlohe.
Example: Classical techniques

• Two basic components
  – *substitutions*: letters replaced by other letters
  – *transposition*: letters rearranged in different order

• These ciphers may be:
  – *monoalphabetic*: only one substitution / transposition
  – *polyalphabetic*: several substitutions / transpositions used

• Product cipher
  – several ciphers concatenated together

Simple Substitution Cipher

• Let $E(k,m)$ be a permutation of the alphabet

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
<th>h</th>
<th>i</th>
<th>j</th>
<th>k</th>
<th>l</th>
<th>m</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>21</td>
<td>12</td>
<td>25</td>
<td>17</td>
<td>24</td>
<td>23</td>
<td>19</td>
<td>15</td>
<td>22</td>
<td>13</td>
<td>18</td>
<td>3</td>
<td>9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>n</th>
<th>o</th>
<th>p</th>
<th>q</th>
<th>r</th>
<th>s</th>
<th>t</th>
<th>u</th>
<th>v</th>
<th>w</th>
<th>x</th>
<th>y</th>
<th>z</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>2</td>
<td>8</td>
<td>16</td>
<td>11</td>
<td>14</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>20</td>
<td>0</td>
<td>6</td>
</tr>
</tbody>
</table>

• plaintext:  proceed meeting as agreed
• ciphertext:  cqkzyyr  jyyowft  vl  vtqyyr
Simple Substitution Cipher

- D(k,c) is given by reversing table

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
<th>h</th>
<th>i</th>
<th>j</th>
<th>k</th>
<th>l</th>
<th>m</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>24</td>
<td>21</td>
<td>15</td>
<td>11</td>
<td>22</td>
<td>13</td>
<td>25</td>
<td>20</td>
<td>16</td>
<td>12</td>
<td>14</td>
<td>18</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>n</th>
<th>o</th>
<th>p</th>
<th>q</th>
<th>r</th>
<th>s</th>
<th>t</th>
<th>u</th>
<th>v</th>
<th>w</th>
<th>x</th>
<th>y</th>
<th>z</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
</tr>
<tr>
<td>9</td>
<td>19</td>
<td>7</td>
<td>17</td>
<td>3</td>
<td>10</td>
<td>6</td>
<td>23</td>
<td>0</td>
<td>8</td>
<td>5</td>
<td>4</td>
<td>2</td>
</tr>
</tbody>
</table>

- ciphertext: cqkzyyr jyyowft vl vtqyyr
- plaintext: proceed meeting as agreed

Simple Substitution Cipher

- Here, a plaintext or ciphertext message is a single character
- Message space is size $26! > 4 \times 10^{26}$
  - But this cipher is very weak. Why?
• In English (and most languages) certain letters are used more often than others
• It would be a good guess that the letters that occur most often in the ciphertext are actually the most common English letters

---

Frequency Analysis

<table>
<thead>
<tr>
<th>Letter</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>E</td>
<td>11.1</td>
</tr>
<tr>
<td>S</td>
<td>5.7</td>
</tr>
<tr>
<td>H</td>
<td>3.0</td>
</tr>
<tr>
<td>V</td>
<td>1.0</td>
</tr>
<tr>
<td>A</td>
<td>8.5</td>
</tr>
<tr>
<td>L</td>
<td>5.5</td>
</tr>
<tr>
<td>G</td>
<td>2.5</td>
</tr>
<tr>
<td>X</td>
<td>0.3</td>
</tr>
<tr>
<td>R</td>
<td>7.5</td>
</tr>
<tr>
<td>C</td>
<td>4.5</td>
</tr>
<tr>
<td>B</td>
<td>2.0</td>
</tr>
<tr>
<td>Z</td>
<td>0.3</td>
</tr>
<tr>
<td>I</td>
<td>7.5</td>
</tr>
<tr>
<td>U</td>
<td>3.6</td>
</tr>
<tr>
<td>F</td>
<td>1.8</td>
</tr>
<tr>
<td>J</td>
<td>0.2</td>
</tr>
<tr>
<td>O</td>
<td>7.1</td>
</tr>
<tr>
<td>D</td>
<td>3.3</td>
</tr>
<tr>
<td>Y</td>
<td>1.8</td>
</tr>
<tr>
<td>Q</td>
<td>0.2</td>
</tr>
<tr>
<td>T</td>
<td>7.0</td>
</tr>
<tr>
<td>P</td>
<td>3.2</td>
</tr>
<tr>
<td>W</td>
<td>1.3</td>
</tr>
<tr>
<td>N</td>
<td>6.7</td>
</tr>
<tr>
<td>M</td>
<td>3.0</td>
</tr>
<tr>
<td>K</td>
<td>1.1</td>
</tr>
</tbody>
</table>
Ciphertext only attack!

TCR KWR WGLC CPC VGQBC
VQVR BCT IXZ CFWGIUWCZ WYC
WCFW QN IJKUCR DCRRIOCR

Answer

Yes, it’s true. Eve broke Bob’s key and extracted the text of Alice’s messages.
• Several courses offered to help you better understand these issues:
  – Computers & Society, Computer Networking, Operating Systems, Software Engineering, etc.
  – Intro to Computer Security, Network Security, Cryptography, Digital Forensics
• Stop by (3rd floor, this building)
  – We are always looking for undergrads (finance, math, linguistics, SILS, etc)
  – send email ({fabian, reiter}@cs.unc.edu)