ActivPass: Your Daily Activity is Your Password

Thomas Finelli, Artian Kica, Evan Gilgenbach
Introduction

- Authentication through passwords
  - Pros:
    - Secure
    - Simple
  - Cons:
    - Secure passwords are hard
    - Password reuse is common and dangerous
    - Easy to share; also security risk

- Create new authentication method
  - Simple and easy to use
  - Resistant to sharing
Vision

- Authenticate based on user’s activity
- Ask user questions about their activity
  - What is the caller’s name from the first call you received today?
  - Which news site did you not visit today? (CNN, CBS, BBC, ABC)
- Question generation
  - Call logs
  - SMS logs
  - Facebook activities
  - Browser history
- Use outlier events
  - Typically easy to remember hard to guess
Vision

● Applications
  ○ Password sharing prevention (Netflix, HBO Go, etc.)
  ○ Replacement for password hints
  ○ In addition to regular password

● User can customize
  ○ Number of questions
  ○ Question format
  ○ Activity permissions

● Perform user studies
Other Forms of Authentication

- **Text Password**
  - Easily sharable
  - Security risks if leaked

- **Physical Biometric Authentication**
  - Face, Fingerprint, Iris, Audio, Gait
  - Security risks if leaked

- **Graphical Password**
  - Touch predetermined areas or sequences of images
Other Forms of Authentication

- **HCI-based Biometric**
  - Input Device Interaction Based
    - Keystroke, Mouse, Haptic
  - Software Interaction Based
    - Email behavior, typing style, game strategy
  - Keystroke pattern based authentication had 50% false acceptance rate

- **One Time Password**
  - Only works for one session
  - Dynamically Changing password which is based on “something you have”
  - Mitigates password sharing by being dynamic
How does ActivPass work?

- **Activity Listener** runs in the background and logs metadata
  - SMS, calls, web pages, etc.

- **When user invokes an application:**
  - Password Generation Module (PGM) creates $n$ password questions based on collected metadata
  - If user can answer $k$ correctly, the application is launched
How does ActivPass work?

- Periodically draws logs in order to classify them with an Activity Categorization Module.
  - Attempts to determine “outliers”
    - e.g. Browser homepage vs infrequently visited webpage
  - Erases any “irrelevant” logs
    - e.g. Unknown Caller
### What sort of questions are asked?

<table>
<thead>
<tr>
<th>Source</th>
<th>Details of data collected</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMS</td>
<td>Time, Receiver/Sender Name</td>
</tr>
<tr>
<td>Call</td>
<td>Time, Type (incoming, outgoing), Name of other person, Duration</td>
</tr>
<tr>
<td>Audio</td>
<td>Title of Music added in this week, Alarm tone, Ring tone</td>
</tr>
<tr>
<td>Web</td>
<td>URL, Time of visit</td>
</tr>
<tr>
<td>Link visited from Facebook</td>
<td>URL, Time of visit</td>
</tr>
<tr>
<td>Facebook Group</td>
<td>Name of Private (secret and closed) groups</td>
</tr>
<tr>
<td>Facebook Pages</td>
<td>Name of pages created by user</td>
</tr>
<tr>
<td>Facebook Profile</td>
<td>Name of Facebook friends of user</td>
</tr>
<tr>
<td>Facebook Message</td>
<td>Time (in milliseconds from epoch), Name of other person, Msg Id, Thread Id</td>
</tr>
</tbody>
</table>
Results came in three stages

- Stage 0 determined feasibility using user-created questions
- Stage 1 determined first-pass numbers from the initial design
- Stage 2 was a re-design that eliminated poorly-performing activities and questions.
Results (cont.)

- Over 50 volunteers given 20 questions:
  - average recall rate: 86.3% ± 9.5
  - average guessability: 14.6% ± 5.7

- Devised Bayesian estimate of challenge given $n$ questions where $k$ are required

- 95% success rate with 5.5% guessability over 15 volunteers

<table>
<thead>
<tr>
<th>n</th>
<th>k</th>
<th>Authentic user</th>
<th>Impostor</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>4</td>
<td>0.554</td>
<td>0.0004</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>0.906</td>
<td>0.011</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>0.989</td>
<td>0.1043</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>0.998</td>
<td>0.468</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0.642</td>
<td>0.0031</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td><strong>0.948</strong></td>
<td><strong>0.0577</strong></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0.996</td>
<td>0.3771</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0.745</td>
<td>0.0213</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0.981</td>
<td>0.2707</td>
</tr>
</tbody>
</table>
Discussion

- Password sharing in a secure threat-model
- Tests against dedicated attackers
- Three Factor Model
- Compares favorably to other non-biometric “something you are” solutions (keystroke)
- Prior Art: Facebook security verification