All Your Face Are Belong to Us: Breaking Facebook’s Social Authentication

Jason Polakis, Marco Lancini, Georgios Kontaxis, Federico Maggi, Sotiris Ioannidis, Angelos Keromytis, Stefano Zanero

stefano.zanero@polimi.it
@raistolo

HackCon 2013
Introduction

- Social Networks
  - Massive user base (Facebook: 1 Billion active users)
  - Appealing targets
- Compromised accounts sold in underground markets
- Majority of spamming accounts compromised, not fake [Gao et al., IMC 2010]
- Recent Facebook phishing attacks
  - Use compromised accounts
  - Steal personal info
  - Social engineering
Social Authentication (SA)

- Two-factor authentication scheme
  - 2nd factor: something user knows
  - Difficult for the attacker to learn

- More user-friendly
  - No need for physical tokens
  - Easy for people to recognize their friends
  - People accustomed to tagging friends (creating the labeled dataset for Facebook)
Social Authentication (SA)

- 7 challenges
- 3 photos per challenge
- 6 possible answers
- User has to correctly answer 5/7 challenges
Motivation

“Can adversaries break SA in an automated manner?”
Triggering Social Authentication

- When log-in considered suspicious
  - From geo-location never seen before
  - From device never seen before

- Requirements
  - Friend list: 50 Friends
    - Gradually increased # of friends in dummy accounts
  - Tagged photos
    - Friends must be tagged in adequate # of photos
SA Photo Selection

“Are photos randomly selected?”

- 2,667 SA photos from real SA tests checked
  - 84% containing faces in manual inspection
  - 80% in automatic inspection by software

- 3,486 random Facebook photos checked
  - 69% contained faces in manual inspection

- Face detection procedures used for selecting photos with faces
SA shortcomings

- Number of friends influences usability
  - Difficult for users with many friends
  - Dunbar’s number

- Content of photos
  - May not contain faces, or even the user tagged
  - Initial user feedback expressed frustration

- Current implementation by Facebook
  - Users can bypass SA by entering date of birth
    - Trivial for attackers to obtain
Threat model

- SA considered safe against adversaries that
  - Have stolen credentials
  - Are *strangers* (not members of the victim’s social circle)

- Not safe against friends or family
- Or any tightly connected network (e.g. University) [Kim et al., FC ‘12]

- We demonstrate SA not safe even against strangers
  - Publicly available data
  - Face recognition software
Attack Scenarios

- Casual Attacker
  - Collects publicly available data

- Determined Attacker
  - Penetrates victim’s social circle
    - Befriends victim’s friends
  - Employs fake accounts
    - Different characteristics appeal to different demographics [Irani, DIMVA ’11]
  - Collects as much private data as possible
1. Crawling Friend List (offline)
   - Crawler retrieves names and UIDs of target’s friends
2. Issuing Friend Requests (offline, optional)
   - Can use dummy accounts
3. Photo Collection/Modeling (offline)
   1. Photo collection
   2. Face extraction and Tag matching
   3. Facial Modeling
   4. Name Lookup
Face recognition

- Custom solution
  - Based on OpenCV library
  + Versatility in parameter tuning
  + Offline
  - Not as accurate

- Cloud Service
  - Face.com (subsequently acquired by Facebook)
  - Exposes API to developers
  + Superior accuracy
  - API rate limiting
Experimental Evaluation

- We collect data as *casual attackers* (publicly available data)
  - We have not compromised or damaged any user accounts (as if I'd ever tell... :-)
- Determined attacker experiment
  - Through simulation
  - Custom face recognition software (flexible)
- Casual attacker experiment
  - Using face.com (accurate)
Dataset

<table>
<thead>
<tr>
<th></th>
<th>Total</th>
<th>Public</th>
<th>Private</th>
</tr>
</thead>
<tbody>
<tr>
<td>UIDs</td>
<td>236,752</td>
<td>167,359</td>
<td>69,393</td>
</tr>
<tr>
<td>Not tagged</td>
<td>116,164</td>
<td>73,003</td>
<td>43,161</td>
</tr>
<tr>
<td>Tagged</td>
<td>120,588</td>
<td>94,356</td>
<td>26,232</td>
</tr>
<tr>
<td>Mean tags per UID:</td>
<td>19.39</td>
<td></td>
<td>10.58</td>
</tr>
<tr>
<td>Tags(^9)</td>
<td>2,107,032</td>
<td>1,829,485</td>
<td>277,547</td>
</tr>
<tr>
<td>Photos</td>
<td>16,141,426</td>
<td>16,141,426</td>
<td>(not collected)</td>
</tr>
<tr>
<td>Albums</td>
<td>805,930</td>
<td>805,930</td>
<td>(not collected)</td>
</tr>
</tbody>
</table>
Breaking SA: determined attacker

- Attacker has access to “all the photos”
- Selected users with enough photos as friends
- Extract faces from photos

- Train our system with $K = 10, 20, \ldots, 120$ faces per friend
- Simulated SA tests from public photos
- Generate 30 simulated SA tests from photos not used for training
Breaking SA: determined attacker

Successfully passed pages as a function of the training set.

Time required to lookup photos as a function of solved pages.
Breaking SA: casual attacker

- Use our dummy accounts as “victims”
- Automated SA triggering through ToR
- Collect snapshot of 127 real SA tests
  - Manually answered the CAPTCHA
- Use face.com to break the tests (challenging conditions)
- ~44 seconds to solve a complete test
Breaking SA: casual attacker

- Manual verification
  - 22% solved
  - 56% need 1-2 guesses

- Failed photos
  - 25% no face in photo
  - 50% unrecogn. face
  - 25% no model available
Attack Surface Estimation

Comromised accounts
- Public friend list
  - Friend list reachable
    - Friend list reachability
      - Public friend list
        - Accept friend request
          - Private friend list
            - 47% (casual attacker)
              - 84% (determined attacker)
        - Refuse friend request
          - Private friend list
            - 79% (casual attacker)
              - 29% (determined attacker)
      - Public photos
        - Photos reachable
          - Photos reachability
            - Public photos
              - Accept friend request
                - Private photos
                  - 70% (casual attacker)
                    - 77% (determined attacker)
            - Refuse friend request
              - Private photos
                - 70% (casual attacker)
                  - 77% (determined attacker)
      - Photos reachability
        - Tags reachable
          - Tags reachability
            - Tags reachable
              - Reached by a casual attacker
                - 42% Tags reachable
                  - Tags found on public photos of friends
                    - 100% Tags reachable
                      - Tags found on public photos of friends
                        - 100% Tags reachable
                          - Refuse friend request
                            - Tags of "private" UIDs
                              - Access only to publicly-available information or private tags that appears on public photos.
                                - Access publicly-available information and issues friend requests to (friends of) victims.

- Dead branch
- Casual attacker
- Determined attacker
Remediation Measures

- Facebook features (opt-in)
  - Login Approval (SMS based) – traditional 2 factor auth.

- Slowing down the attacker
  - Remove suggestions
  - Reduce time window

- Revisit SA
  - Select photos that contain faces software can’t identify
Facebook’s Response

- Acknowledged our results
- “Deployed SA to raise the bar in large-scale phishing attacks”
- “Not designed for small-scale or targeted attacks”
- “Users can enable Login Approval”
  - How many have actually done so?
Discussion

- Eurograbber malware [1]
  - Targets EU banks
  - Infects user’s computer
  - Tricks user into installing smartphone malware via bogus messages and social engineering
  - Intercepts 2nd factor token sent to user’s device

- What are the implications of using the same device as the 2nd factor, and for browsing?

- SA security compared to traditional two-factor with smartphones?

Conclusions

- Designed and implemented an automated SA breaking system
- Demonstrated the weaknesses of SA
- Publicly-available data sufficient for attackers
- Cloud services can be utilized effectively

- Facebook should reconsider its threat model
- Need to revisit the SA approach
Thank you!

Most of this work was funded by the EC under FP7 project SysSec