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

polakis@ics.forth.gr
Outline

- Introduction
- Social Authentication
- Breaking Social Authentication
- Experimental Evaluation
- Remediation Measures
- Discussion
- Conclusions
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
  - Identify your friends
  - Secure profiles against attackers with stolen credentials
Social Authentication (SA)

- Two-factor authentication scheme
  - 2\textsuperscript{nd} 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 challenges
“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 the actual 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
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
Breaking Social Authentication

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
We collect data as *casual attackers* (publicly available data)

We have not compromised or damaged any user accounts

Determined attacker experiment
  - Through simulation
  - Custom face recognition software (flexible)

Casual attacker experiment
  - Using *face.com* (accurate)
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
Successfully passed pages as a function of the training set.

Time required to lookup photos as a function of solved pages.
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
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