Integro: Leveraging Victim Prediction for Robust Fake Account Detection in OSNs

Yazan Boshmaf, Matei Ripeanu, Konstantin Beznosov
University of British Columbia

Dionysios Logothetis, Georgios Siganos
Telefonica Research

Jorge Laria, Jose Lorenzo
University of British Columbia

Presented at NDSS’15, San Diego, Feb 2015
Integro: Leveraging Victim Prediction for Robust Fake Account Detection in OSNs

Why is it important to detect fakes?

Yazan Boshmaf, Matei Ripeanu, Konstantin Beznosov
University of British Columbia

Dionysios Logothetis, Georgios Siganos
Telefonica Research

Jorge Laria, Jose Lorenzo
University of British Columbia

Presented at NDSS’15, San Diego, Feb 2015
Fake accounts are bad for business

“… If advertisers, developers, or investors do not perceive our user metrics to be accurate representations of our user base, or if we discover material inaccuracies in our user metrics, our reputation may be harmed and advertisers and developers may be less willing to allocate their budgets or resources to Facebook, which could negatively affect our business and financial results…”
Fake accounts are bad for users

OSNs are attractive medium for abusive content

Social Infiltration

Connecting with many benign users (friend request spam)
Fake accounts are bad for users

OSNs are attractive medium for abusive content

Social Infiltration

Data collection

Online surveillance, profiling, and data commoditization

Fake accounts are bad for users

OSNs are attractive medium for abusive content

Social Infiltration  Data collection  Misinformation

Influencing users, biasing public opinion, propaganda

Ratkiewicz et al. Detecting and tracking political abuse in social media. Proc. of ICWSM. 2011
Fake accounts are bad for users

OSNs are attractive medium for abusive content

Social Infiltration → Data collection → Misinformation → Malware Infection

Infecting computers and use it for DDoS, spamming, and fraud

Thomas et al. The Koobface botnet and the rise of social malware. Proc. of MALWARE, 2010
Fake accounts are bad for users

OSNs are attractive medium for abusive content

How do OSNs detect fakes today?

Social Infiltration  Data collection  Misinformation  Malware Infection

Infecting computers and use it for DDoS, spamming, and fraud

Thomas et al. The Koobface botnet and the rise of social malware. Proc. of MALWARE, 2010
Feature-based detection

- Interactions
- Pictures
- Friends
- Posts
- Triadic closure
- Ad clicks

Feature-based detection

Fake accounts mimic real accounts

Feature-based detection is ineffective

Only 20% of fakes were detected

All manually flagged by concerned users

Graph-based detection

Assumes social infiltration on a large scale is infeasible

Finds a (provably) sparse cut between the regions by ranking
Graph-based detection

Assumes social infiltration on a large scale is infeasible

Finds a (provably) sparse cut between the regions by ranking
Graph-based detection

Ranks computed from landing probability of a short random walk

Most real accounts rank higher than fakes

Cao et al. Aiding the detection of fake accounts in large scale social online services, In proc. of NSDI, 2012
Graph-based detection is not resilient to social infiltration

50% of fakes had more than 35 attack edges

Boshmaf et al. Graph-based Sybil detection in social and information systems. In proc. of ASONAM, 2013
Graph-based detection is not resilient to social infiltration

Can we do better?

Hint: What if we integrate both?

50% of bots had more than 35 attack edges
Premise: Regions can be tightly connected
Identify potential victims with some probability

Potential victims are real accounts that are likely to be victims
Leverage victim prediction to reduce cut size

Cut size = $1.9 \times 10^{-10}$

Assign lower weight to edges incident to potential victims
Delimit the real region by ranking accounts

Ranks computed from landing probability of a short random walk

Most real accounts are ranked higher than fake accounts
(Bound on ranking quality)

Number of fake accounts that rank equal to or higher than real accounts is $O(\text{vol}(E_A) \log n)$ where $\text{vol}(E_A) \leq |E_A|$

Most real accounts are ranked higher than fake accounts.

Assuming a fast mixing real region and an attacker who establishes attack edges at random
Integro: Victim classification

Identifies potential victims in $O(n \log n)$ time

Pros:
- Proactive protection
- Near real-time responses
- Scales to millions of users
- Hard to circumvent

Cons:
- Doesn’t identify fakes
- May introduce usability issues
- Not provably secure
Victim classification is feasible using low-cost features

Random Forests (RF) achieves up to 52% better than random

No need to train on more than 40K feature vectors on Tuenti
Integro: User account ranking

Integrates victim classification (labels + probabilities) into graph as edge weights

Pros:
- Scales to millions of users
- Hard to circumvent
- Accurate detection
- Provably secure

Cons:
- Reactive protection
- Batch processed


Ranks accounts based on a short random walk in $O(n \log n + m)$ time
Ranking is resilient to infiltration

Integro delivers up to 30% higher AUC, and AUC is always $> 0.92$

Targeted-victim attack

Random-victim attack
Deployment at Tuenti confirms results

Integro delivers up to an order or magnitude better precision

Precision at lower intervals

Precision at higher intervals
Deployment at Tuenti confirms results

Integro delivers up to an order or magnitude better precision

Victim prediction yields robust detection (new security paradigm)
In conclusion, Integro achieves:

☑️ Proactive protection
☑️ Near real-time responses
☑️ Scales to millions of users
☑️ Hard to circumvent
☑️ Accurate detection
☑️ Provably secure
Fork or clone Integro now!

SyPy and Integro are publicly released

http://boshmaf.github.io/sypy

https://grafos.ml
Fork or clone Integro now!

SyPy and Integro are publicly released

SyPy

Graph-based Sybil detection.

- Download ZIP
- Download TAR
- View On GitHub

Backup

This project is maintained by boshmaf

http://boshmaf.github.io/sypy

https://grafos.ml
Integro in a nutshell

Uses distributed machine learning and graph processing infrastructure

Runs in $O(n \log n + m)$ time end-to-end
Datasets

• Labeled feature vectors
  – 8.8K public Facebook profiles (32% victims)
  – 60K full Tuenti profiles (50% victims)

• Graph samples
  – Time stamped infiltration targeting 2.9K real accounts, with 65 fakes and 748 attack edges
  – 6.1K real accounts
# Feature engineering

<table>
<thead>
<tr>
<th>Feature</th>
<th>Brief description</th>
<th>Type</th>
<th>RI Score (%)</th>
<th>Facebook</th>
<th>Tuenti</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>User activity:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Friends</td>
<td>Number of friends the user had</td>
<td>Numeric</td>
<td>100.0</td>
<td>84.5</td>
<td></td>
</tr>
<tr>
<td>Photos</td>
<td>Number of photos the user shared</td>
<td>Numeric</td>
<td>93.7</td>
<td>57.4</td>
<td></td>
</tr>
<tr>
<td>Feed</td>
<td>Number of news feed items the user had</td>
<td>Numeric</td>
<td>70.6</td>
<td>60.8</td>
<td></td>
</tr>
<tr>
<td>Groups</td>
<td>Number of groups the user was member of</td>
<td>Numeric</td>
<td>41.8</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Likes</td>
<td>Number of likes the users made</td>
<td>Numeric</td>
<td>30.6</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Games</td>
<td>Number of games the user played</td>
<td>Numeric</td>
<td>20.1</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Movies</td>
<td>Number of movies the user watched</td>
<td>Numeric</td>
<td>16.2</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Music</td>
<td>Number of albums or songs the user listened to</td>
<td>Numeric</td>
<td>15.5</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>TV</td>
<td>Number of TV shows the user watched</td>
<td>Numeric</td>
<td>14.2</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Books</td>
<td>Number of books the user read</td>
<td>Numeric</td>
<td>7.5</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td><strong>Personal messaging:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sent</td>
<td>Number of messages sent by the user</td>
<td>Numeric</td>
<td>N/A</td>
<td>53.3</td>
<td></td>
</tr>
<tr>
<td>Inbox</td>
<td>Number of messages in the user’s inbox</td>
<td>Numeric</td>
<td>N/A</td>
<td>52.9</td>
<td></td>
</tr>
<tr>
<td>Privacy</td>
<td>Privacy level for receiving messages</td>
<td>5-Categorical</td>
<td>N/A</td>
<td>9.6</td>
<td></td>
</tr>
<tr>
<td><strong>Blocking actions:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Users</td>
<td>Number of users blocked by the user</td>
<td>Numeric</td>
<td>N/A</td>
<td>23.9</td>
<td></td>
</tr>
<tr>
<td>Graphics</td>
<td>Number of graphics (photos) blocked by the user</td>
<td>Numeric</td>
<td>N/A</td>
<td>19.7</td>
<td></td>
</tr>
<tr>
<td><strong>Account information:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Last updated</td>
<td>Number of days since the user updated the profile</td>
<td>Numeric</td>
<td>90.77</td>
<td>32.5</td>
<td></td>
</tr>
<tr>
<td>Highlights</td>
<td>Number of years highlighted in the user’s time line</td>
<td>Numeric</td>
<td>36.3</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Membership</td>
<td>Number of days since the user joined the OSN</td>
<td>Numeric</td>
<td>31.7</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>Gender</td>
<td>User is male or female</td>
<td>2-Categorical</td>
<td>15.8</td>
<td>7.9</td>
<td></td>
</tr>
<tr>
<td>Cover picture</td>
<td>User has a cover picture</td>
<td>2-Categorical</td>
<td>10.5</td>
<td>&lt; 0.1</td>
<td></td>
</tr>
<tr>
<td>Profile picture</td>
<td>User has a profile picture</td>
<td>2-Categorical</td>
<td>4.3</td>
<td>&lt; 0.1</td>
<td></td>
</tr>
<tr>
<td>Pre-highlights</td>
<td>Number of years highlighted before 2004</td>
<td>Numeric</td>
<td>3.9</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Platform</td>
<td>User disabled third-party API integration</td>
<td>2-Categorical</td>
<td>1.6</td>
<td>&lt; 0.1</td>
<td></td>
</tr>
</tbody>
</table>

18 features (Facebook), 14 features (Tuenti)
Both systems are sensitive to seed-targeting attack, follow seed selection strategy.
Scalability

Near linear scalability with number of accounts

RF is “embarrassingly parallel”

Ranking is “PageRank scalable”