Server-Side Second Factors

Approaches to Measuring User Authenticity

David Freeman

Head of Anti-Abuse Engineering at LinkedIn

Enigma 2016

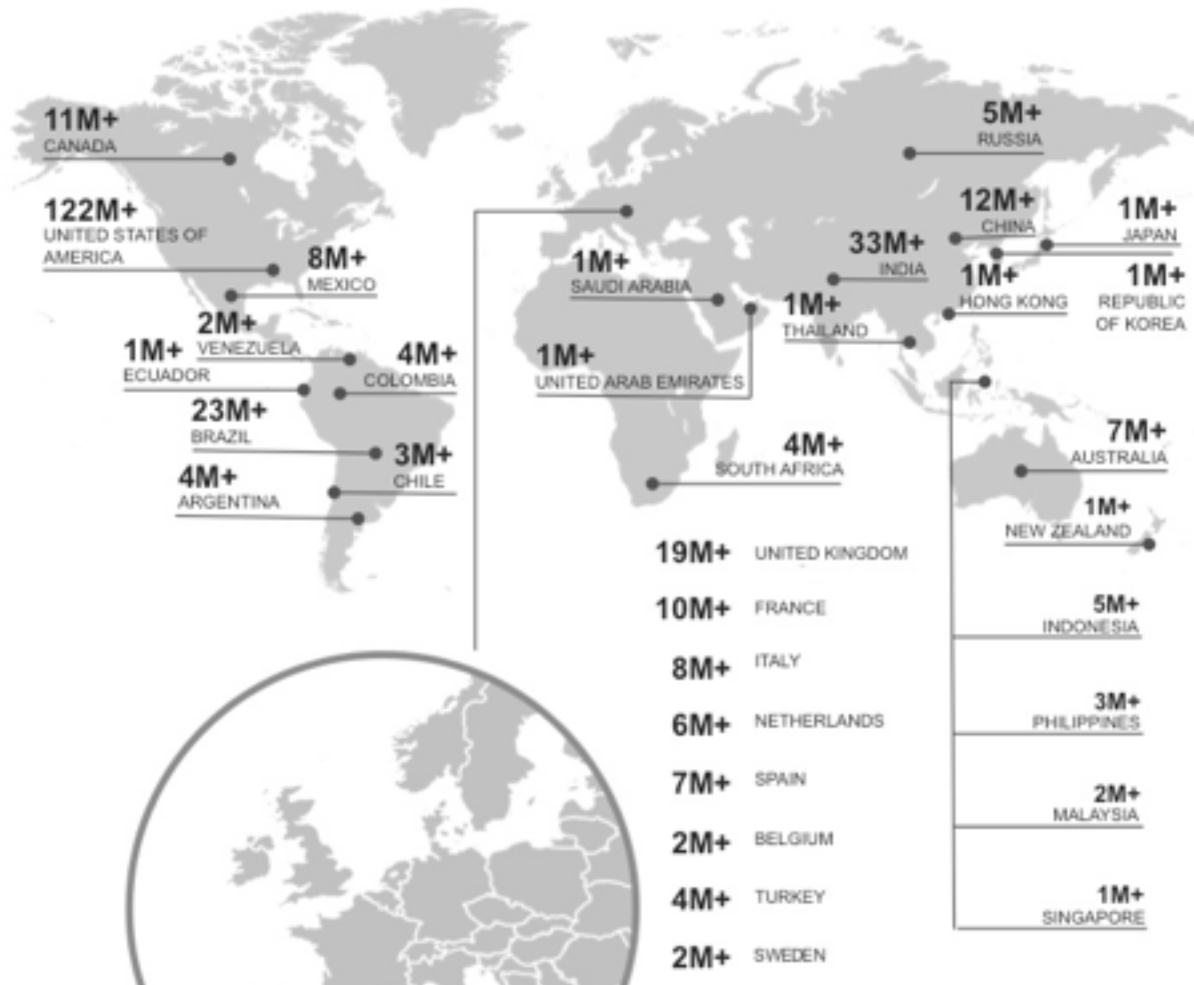
San Francisco, CA

25 Jan 2016

Imagine my job…

400,000,000+
REGISTERED MEMBERS

399,800,000+
ARE NOT SECURITY EXPERTS



An Enigma attendee would never…

use a common password

reuse passwords across sites
(especially sites that get hacked)

get phished

tell someone their password

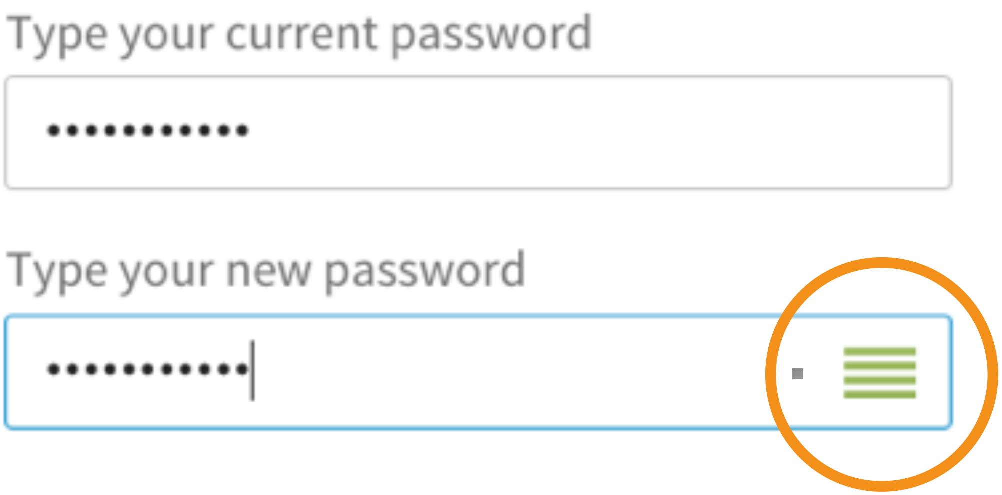
…but some of the other 399.8 million might!

Why take over a LinkedIn account?

– Spam looks more legitimate when it comes from someone you know.

– Accounts can have valuable contacts, messages, and other data.

Can we force better passwords?

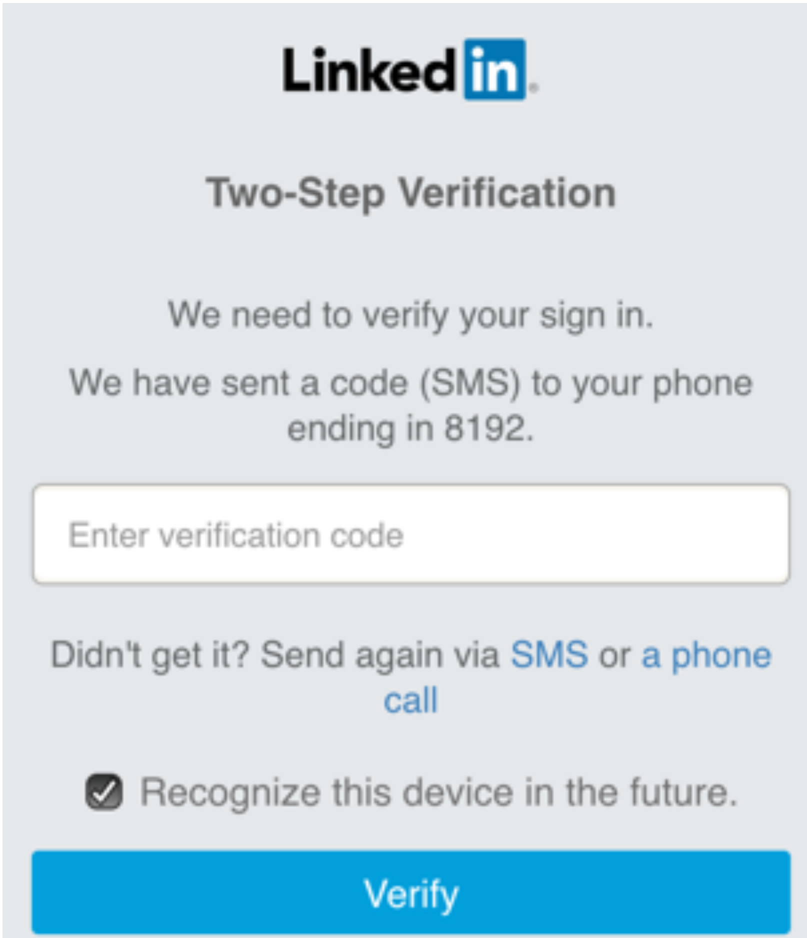


dashlane

LastPass

StickyPassword

Alternatives to better passwords?



We must assume the worst

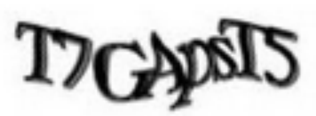
- The member’s password is weak or known.
- The account is not opted in to two-factor authentication.
- The attacker could be a bot or human.
- Members are not going to change their behavior.

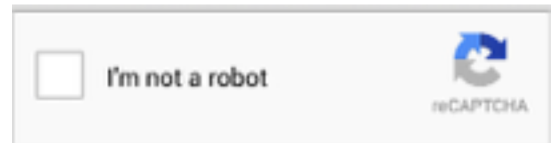
Scoring login attempts

- Assess level of suspiciousness.
- Require second factor above some threshold.
- Cover all entry points.

What second factors could we require?

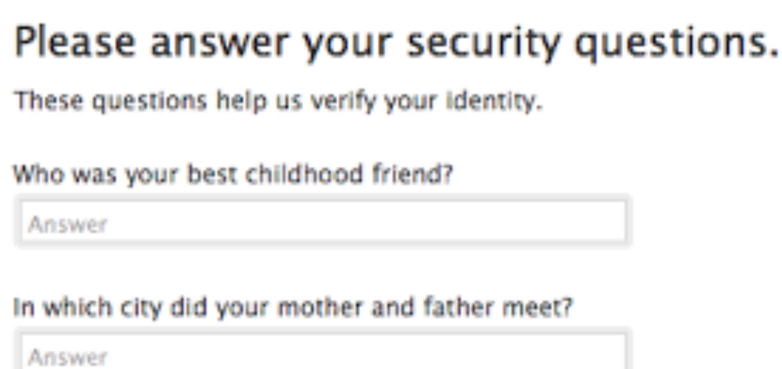
- Prove you’re a human





- Establish contact through another channel

- Repeat back information you gave us earlier



Tradeoffs

vs.

Second factor needs to be easy for good users, hard for bad guys.

- Biggest gap: SMS verification
- Smallest gap: “First name challenge”

What data do we have to score logins?

- Request data:
  - IP address (and derived country, ISP, etc.)
  - Browser's useragent (and OS, version, etc.)
  - Timestamp
  - Cookies
  - and more…
- Reputation scores for all of the above
- Global counters on all of the above
- History of member's previous (successful) logins

Which logins are suspicious?

Heuristics can get you pretty far:

Effectiveness of heuristics

Good at stopping large-scale/indiscriminate attacks.

- Bot attack from Jan 2015:
  99% blocked on country mismatch
- Bot attack from Nov 2015:
  98% matched country
  100% blocked by rate-limiting

Not so good at stopping targeted attacks.

- 96% of legitimate logins match country
- 93% of compromises match country

Formulating the problem statistically

Ultimately, model needs to decide whether

$$\frac{\Pr[\text{attack}|u, X]}{\Pr[\text{legitimate}|u, X]} > 1.$$

[Notation:

X = user data (timestamp, IP address, browser, etc.)

u = user identity]

Formulating the problem statistically

Ultimately, model needs to decide whether

$$\frac{\Pr[\text{attack}|u, X]}{\Pr[\text{legitimate}|u, X]} > 1.$$

But it's hard to estimate this ratio directly from the data!

- Most members are never attacked (numerator is 0)
- Only a few samples per member.
- Members come from previously unseen values of *X* (IP addresses, browsers, etc.)

Computing the likelihood of attack

After a few assumptions and a lot of Bayes' rule, we get:

Global likelihood of seeing data *X*

Asset Reputation Score (interpreted as a probability)

Value of account to attacker

$$\frac{\Pr[\text{attack}|u,X]}{\Pr[\text{legitimate}|u,X]} = \Pr[\text{attack}|X] \cdot \frac{\Pr[X]}{\Pr[X|u]} \cdot \frac{\Pr[u|\text{attack}]}{\Pr[u]}$$

Appearance of data *X* in *u*'s (legitimate) login history

Likelihood of member *u* logging in

No per-member attack data required!

Computing the likelihood of attack

After a few assumptions and a lot of Bayes' rule, we get:

Asset Reputation Score
(interpreted as a probability)

Global likelihood of
seeing data *X*

Value of account
to attacker

$$\frac{\Pr[\text{attack}|u, X]}{\Pr[\text{legitimate}|u, X]} = \Pr[\text{attack}|X]^{\alpha} \cdot \frac{\Pr[X]^{\beta}}{\Pr[X|u]^{\gamma}} \cdot \frac{\Pr[u|\text{attack}]^{\delta}}{\Pr[u]^{\epsilon}}$$

Appearance of data *X*
in *u*'s (legitimate) login history

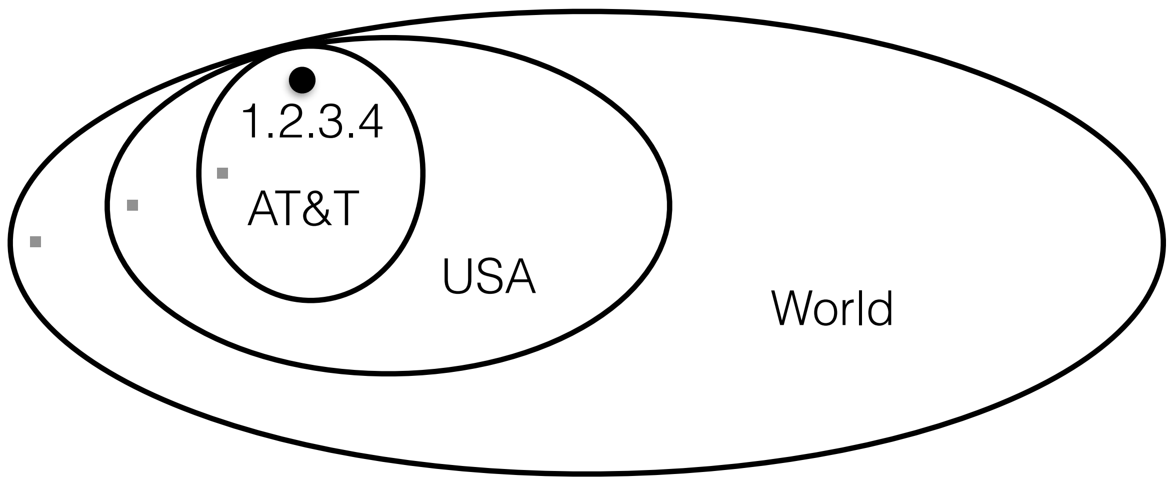
Likelihood of member *u*
logging in

If you have labeled attack data, use it to learn feature weights.

Smoothing

Q: How do we estimate $\Pr[X|u]$ when *X* is an IP address that *u* has never logged in from?

A: We have auxiliary information about unseen IPs:



- Use ISP- or country-level data to estimate probabilities.
- Give higher weight to **unseen** events from a **known** ISP.

Experimental Results

Prototype model using two features:

IP hierarchy & useragent hierarchy

(useragent, browser version, browser, OS)

Test data:

(a) 6 months of compromised accounts

(b) botnet observed in Jan 2015

Results	Country Match	Model Result	Model+Heuristics
Botnet	99%	95%	99%
Compromised accounts	7%	77%	81%
False positives	4%	10%	3%

Take-aways

- Protect *all* users.
- Minimize friction.
- Use both heuristics and machine learning.
- Use statistical models even if you don't have good labeled data.

Questions?

dfreeman@linkedin.com

Login scoring model is joint work with Sakshi Jain (LinkedIn),
Markus Dürmuth (Ruhr Universität Bochum), and
Battista Biggio and Giorgio Giacinto (Università di Cagliari), to appear at NDSS '16.