Conscript Your Friends into Larger Anonymity Sets with JavaScript

Henry Corrigan-Gibbs
Stanford

Bryan Ford
Yale

ACM Workshop on Privacy in the Electronic Society
4 November 2013
New Anonymity Systems Have a “Chicken-and-Egg” Problem

- Few users
- Small anonymity sets
Overthrow the regime!!

Start the revolution!!

Emacs rulz!!

Adversary could just arrest all three participants.
Overthrow the regime!!

Start the revolution!!

Emacs rulz!!
Idea

• “Conscript” casual Internet users into an anonymity system using JavaScript
  – **Casual users** submit null messages
  – **Savvy users** use a browser plug-in to swap out the null messages with real ones

• Compatible with a number of existing anonymity systems
Outline

• Motivation
• Architecture
• Attacks and Defenses
• Evaluation
Using a *randomized* encryption scheme
GET /index.html

<html><script>...

E_1(E_2(E_3(000))))

m

Plugin

E_1(E_2(E_3(m))))
The Adversary Sees
The Adversary Sees
The Adversary Sees

Start the revolution!

00000000
Security Property

IF Casual users’ messages indistinguishable from savvy users’ messages
THEN Conscripting increases the size of the savvy users’ anonymity set
Compatible Anonymity Systems

1. Monotonic anonymity set size
2. Possible to simulate traffic streams
3. Easy to identify malformed messages

Yes: Timed mix cascade, verifiable shuffles, remailers (maybe), verifiable DC-nets

No: Tor, batching mix net
The ConScript Script

E.g., for a mix-net

• The JavaScript application sends
  – RSA encryption routines,
  – server public keys, and
  – code to POST ciphertext to mix-server.

• Mix servers uses
  Access-Control-Allow-Origin header
Outline

• Motivation
• Architecture
• **Attacks and Defenses**
• Evaluation
Threats

- Web server can serve malicious JavaScript!
- User can submit incorrect messages!
- Vulnerabilities of the underlying anonymity system!
JavaScript Attack

Plugin only swaps out msg if scripts match exactly
More Attacks

• Side-channel attack
• Selective DoS attack ("trickle attack")
• Distribution point monitoring
  – Who downloads the plug-in?
• User-counting attack
• […]

Even if adversary can distinguish:
Anonymity provided ≥ | Savvy users |
Outline

- Motivation
- Architecture
- Attacks and Defenses
- Evaluation
Proof-of-Concept Evaluation

Time (ms) to generate a dummy message on different devices. OpenPGP.js for RSA encryption, SJCL for ECC.

<table>
<thead>
<tr>
<th>Device</th>
<th>Mix-net</th>
<th>Verifiable DC-net</th>
</tr>
</thead>
<tbody>
<tr>
<td>Workstation</td>
<td>81</td>
<td>156</td>
</tr>
<tr>
<td>Laptop</td>
<td>133</td>
<td>231</td>
</tr>
<tr>
<td>iPhone 4</td>
<td>9 009</td>
<td>62 973</td>
</tr>
<tr>
<td>Milestone</td>
<td>–</td>
<td>63 504</td>
</tr>
</tbody>
</table>
Related Work

• **AdLeaks** [Roth et al., FC‘13]
  – Similar idea: JS for dummy messages
  – Works with one particular anonymity system
  – Vulnerable to active attacks by browsers

• **FlashProxy** [Fifield et al., PETS‘12]
  – Use JavaScript to “conscript” browsers into acting as Tor bridges

• **Bauer** [WPES ‘03]
  – Covert channel *between* mix servers
Conclusion

• Conscripted anonymity is one possible way to address the chicken-and-egg problem in online anonymity

• Ongoing work on in-browser crypto could have benefits for anonymity systems too – e.g., W3C Crypto API standard
Questions?

Henry Corrigan-Gibbs
henrycg@stanford.edu

Thanks to David Fifield and David Wolinsky for their comments.