Measuring the Insecurity of Mobile Deep Links of Android

Fang Liu, Chun Wang, Andres Pico, Danfeng Yao, Gang Wang

VirginiaTech
Invent the Future
Web Browsing is Going Mobile

• Users spend more time on mobile devices\(^1\)
  – Mobile devices ~ 3.1 hours
  – Laptops/Desktops ~ 2.2 hours

• **Native apps**: the new web interface
  – Shorter loading time

Apps are the future of the web?

---

\(^1\)Mary Meeker, Internet Trends 2017 - Code Conference, KPCB, 2017
Apps vs. Mobile Websites

• Apps cannot replace websites yet
  – Apps sit in a “walled garden”
  – Difficult to navigate across apps
  – Difficult to search and access in-app content globally

• Apps + mobile websites eco-system
  – Complementary to each other
  – Likely to co-exist (for a long time)
Web-App Communication via Deep Links

• Deeper integration of websites and apps
  – Mobile deep links: URIs pointing to pages inside apps

Greatly improve user experience!
Hijacking Risks of Deep Links

• Scheme URL: mobile deep link v1.0
  – Designed for functionality, no security features
  – Apps can register their own scheme to the OS
  – Android and iOS since 2009

• Hijacking URL schemes
  – Phishing
  – Stealing sensitive data in the URL
  – Any app can register other apps’ schemes

\[\text{[Mobisys'11]} \text{[CCS'14]} \text{[CCS'15]}\]
Defense Relying on Users

• Prompt users when multiple apps have the same scheme

• But, user prompting can be skipped
  – If the malicious app installed before the real app
  – If the malicious app tricked users to set preference

• User as the only defense = bad defense
Deep Link v2.0 Prevents Link Hijacking

- **App links**
  - HTTP/HTTPS links only, no custom schemes
  - Requires [app link association](http://facebook.com/)

- **Intent URL**
  - Explicitly specify the target app by package name

```
fb:/// ➔ https://facebook.com/
```

Uniqueness guaranteed by the app market
This Study

• Research questions
  – How are different mobile deep links used in practice?
  – How likely is an app’s scheme hijacked by another app?
  – How effective are the new deep link mechanisms in mitigating the hijacking threats?

• Large-scale empirical measurements
  – Deep links across web and apps
  – Primarily focus on Android (>80% market share)

Let the DATA speak
Outline

• Introduction

• The Adoption of Mobile Deep Links
  – Scheme URL vs. App Link
  – App Links: Vulnerabilities & Misconfigurations

• Characterizing Hijacking Attacks

• Hijacking Threats on the Web
Datasets

• Android apps (25 app categories)
  – 164,322 most popular apps, December 2014
  – 164,963 most popular apps, August 2016
  – 115,399 apps in both snapshots

• Popular websites
  – Alexa top 1 million domain’s index page, October 2016
  – Dynamic crawler to mimic Chrome mobile browser (OpenWPM\textsuperscript{1})
  – Lower bound of mobile deep links on the web

\textsuperscript{1}\textsc{Englehardt}, S., \textsc{And Narayanan}, A. Online tracking: A 1-million-site measurement and analysis. In Proc. of CCS (2016)
Deep Link Usage in Apps

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Total Apps</th>
<th>Apps register Scheme URLs</th>
<th>Apps register App Links</th>
<th>Apps register either Links</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014</td>
<td>164,322</td>
<td>10,565 (6.4%)</td>
<td>4,545 (2.8%)</td>
<td>12,428 (7.6%)</td>
</tr>
</tbody>
</table>

Key observations

• Mobile deep links are getting popular among apps
• The vulnerable scheme URLs are still increasinly used
App Link Verification

• App link association to prevent link hijacking
  – Only HTTP/HTTPS links are allowed
  – Establish the association between the App link and the web domain
  – iOS has a similar mechanism called “universal link”

Unverified links
• Android: still works, but trigger user prompt
• iOS: cannot open the link in the app

Authorized app to open https://facebook.com?

App link: https://facebook.com

OS

Assetlinks.json

facebook.com

Assetlinks.json

Path=/*

Register
App Link Verification in Practice

8,878 apps have adopted App links

415 apps enabled link verification (4.7%)

194 apps configured it correctly (2.2%)

Common Errors (221 apps)
- No associate files (177)
- Under HTTP (11)
- Invalid associate file (10)
- Invalid app manifest (26)

- Rarely do apps verify their App links correctly
  - A lack of incentives: unverified App links can still open apps
- Configuration errors are not identified and mitigated quickly
App Link Vulnerability: Over-

- Allows unverified app links to skip user prompting – “always using this app” preference
  - Preference applies to all the https:// links that an app registered

- Hijack password without user knowledge!
- HTTPS does not help

- Root cause: the preference setting is too excessive
- Reported to Google in Feb 2017, case established in May 2017
Outline

• Introduction

• The Adoption of Mobile Deep Links
  • Scheme URLs are still widely used
  • App links are rarely verified correctly
  • App links introduce a new vulnerability

• Hijacking Threats on the Web
Identifying Potential Hijacking Apps

- **Link collision**: multiple apps that registered the same Link
  - 18,839 unique schemes (e.g., fb://)
  - 18,561 unique App link hosts (e.g., facebook.com)
Classifying Link Collisions
Not all link collisions are malicious

<table>
<thead>
<tr>
<th>Scheme URL</th>
<th>App Link</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Functional scheme</strong></td>
<td><strong>Functional web host</strong></td>
</tr>
<tr>
<td>Represents a common functionality e.g. geo:///, tel:///, file://</td>
<td>N/A</td>
</tr>
<tr>
<td><strong>Third-party scheme</strong></td>
<td><strong>Third-party web host</strong></td>
</tr>
<tr>
<td>Used by 3rd-party library and APIs e.g., x-oauthflow-twitter:///</td>
<td>e.g., zxing.appspot.com</td>
</tr>
<tr>
<td><strong>Per-app scheme</strong></td>
<td><strong>Per-app web host</strong></td>
</tr>
<tr>
<td>Represents individual apps e.g., fb://, twitter://</td>
<td>e.g., facebook.com, twitter.com</td>
</tr>
</tbody>
</table>

Potentially Malicious Hijacking
Classifying Per-App Hijacking

- Manual examination by 3 judges
- Automated classifiers for per-app hijacking
  - 54% accuracy overall, 84% accuracy for link collisions of 4+ apps

### Link Collisions

<table>
<thead>
<tr>
<th>Schemes (7,432 apps)</th>
<th>3,272 web hosts (2,868 apps)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Functional</strong></td>
<td></td>
</tr>
<tr>
<td>30 (2,135)</td>
<td></td>
</tr>
<tr>
<td><strong>3rd-party</strong></td>
<td></td>
</tr>
<tr>
<td>197 (3,972)</td>
<td></td>
</tr>
<tr>
<td><strong>Per-app</strong></td>
<td></td>
</tr>
<tr>
<td>149 (893)</td>
<td></td>
</tr>
<tr>
<td>2,314 (1,593)</td>
<td></td>
</tr>
</tbody>
</table>

Search for online tutorials for the 3rd-party libs and APIs Not from the same developer
Hijacking Case Studies

• Traffic hijacking
  – google.com registered by 480 apps (305 non-Google developers)
  – google.navigation:// registered by 79 apps (32 developers)
  – Other popular targets

• Competing Apps
  – Careem (5M downloads)
    widely integrated with hotel websites/apps
  – QatarTaxi (10K downloads)
    also registered careem://*
Case Studies (Cont.)

• Redirection apps and MITM
  – Resolve deep links and redirect users to target apps
  – Hard-coded mapping, without permission of the target app
  – Log URL and parameters to files

• Example: URLLander
  – Registered payments.ebay.com while the official eBay app did not
  – Registered www.paypal.com (SESSIONID parameter)
Outline

• Introduction

• The Adoption of Mobile Deep Links

• Characterizing Hijacking Attacks

• Hijacking Threats on the Web
  – Usage of Intent URL
  – Hijacked App Links vs. Scheme URLs
Deep Links on Alexa Top 1M Websites

- Extracting deep links from web pages
  - Regular expression matching with scheme URLs/App links
  - Intent URL has clear patterns:

<table>
<thead>
<tr>
<th>Number of Deep Links</th>
<th>Scheme URL</th>
<th>App Link</th>
<th>Intent URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>431K links</td>
<td>10,000,000</td>
<td></td>
<td>(19.7% websites)</td>
</tr>
<tr>
<td>3.2M links</td>
<td>10,000,000</td>
<td></td>
<td>(48.5% websites)</td>
</tr>
<tr>
<td>1203 links</td>
<td>1,000</td>
<td></td>
<td>(0.05% websites)</td>
</tr>
</tbody>
</table>

Intent URLs are rarely used!
“Hijacked” Deep Links on the Web

- Deep links on the web that may take users to the wrong app
  - Deep links registered by multiple apps vs. links on the web pages

![Graph showing the comparison of functional, third-party, and per-app hijacked deep links]

- **Scheme URLs**
  - App links introduce more hijacked links than Scheme URLs to the web
- **App links**
  - Massive HTTP/HTTPS links already exist on the web → Hijackable
Discussion

• **Scheme URLs** are still widely used by apps and websites

• The new **App link** not only fails to improve security, but significantly increases hijacking risks
  – App links are rarely verified (2.2% apps did it correctly)
  
  **iOS App links**: 1,925 out of 12,570 (15%) apps have misconfigured the verification

• **Intent URLs** are rarely used on the web
Countermeasures

- Disable per-app scheme
  - Whitelist functional schemes
- Enforce App link verification
- Fix App link over-permission
  - Set it to the link/domain level
- Break legacy links on the web
- Apps will need a web front

1^https://developer.android.com/about/dashboards/index.html
Thank You