Agenda

• Why are we here?
• Where are we?
• Where do we go next?
Why are we here?
Anatomy of an attack

Social Engineering

Open attachment

Click a URL

Browse to a website

Exploitation & Installation

Command & Control

Brute force account or use stolen account credentials

User account is compromised

Attacker attempts lateral movement

Privileged account compromised

Domain compromised

Attacker collects reconnaissance & configuration data

Attacker exploits protocol technical design or implementation issues

Attacker accesses sensitive data

Exfiltrate data
Mind the Gap – Where Attackers (often) Enter

Social Engineering
- Open attachment
- Click a URL

- Brute force account or use stolen account credentials
- Attacker exploits protocol technical design or implementation issues

- Attacker collects reconnaissance & configuration data

User account is compromised
- Attacker attempts lateral movement
- Privileged account compromised
- Domain compromised

Exfiltrate data
- Attacker accesses sensitive data
Cross-Device Flow Social Engineering Exploit

1. Get a Code
2. Change Context
3. Scan or enter a Code, click on link
4. Authenticate/Authorize
5. Retrieve Tokens

Attack Pattern Summary: Exploit the Unauthenticated Channel
1. Initiate the session, retrieve code (QR code, user code)
2. Use social engineering to change context and persuade user to authorize session (illicit consent grant)
3. Bypasses multi-factor authentication (don’t need to harvest credentials)
Where are we?

draft-ietf-oauth-cross-device-security-04

Abstract

This document describes threats against cross-device flows along with near term mitigations, protocol selection guidance, and the analytical tools needed to evaluate the effectiveness of these mitigations. It serves as a security guide to system designers, architects, product managers, security specialists, fraud analysts and engineers implementing cross-device flows.
What’s New: Cross-Device Session Transfer Pattern

Feedback at OSW

- Two of the examples did not cleanly map to the general patterns described.
  - Example A5/B5 and A7/B7
  - User starts flow on authorization device, not consumption device.
  - QR code is scanned to transfer a session not request authorization
  - Cross-Device Session Phishing
  - Example: OpenID4VCI pre-auth code

- Thanks to Marco Pernpruner and Giada Sciarretta
What’s New: 2 new mitigations, 1 Rename

New mitigations
• User Education
• Request Binding with Out-of-Band Data

New name
• Authenticate-then-Initiate
What’s New: 2 new exploits observed in the wild

Fake Helpdesk

3.3.5. Example B4.2: Fake Helpdesk (Backchannel-Transferred Session Pattern)

An attacker obtains the contact information for a user and contacts them, pretending to be a representative of the user’s financial institution. The attacker informs the user that there were a number of fraudulent transactions against their account and asks them to review these transactions by approving or rejecting them. The attacker then triggers a sequence of transactions. The user receives an authorization request for each transaction and declines them as they do not recognize them. The attacker then informs the user that they need to close the users account and transfer all the funds to a new account to prevent further fraudulent transactions. The user receives another authorization request which they approve, or provide additional authorization information to the attacker which enables the attacker to complete their attack and defraud the user.

Consent Request Overload

3.3.10. Example B9: Illicit Access to Administration Capabilities Through Consent Request Overload (Backchannel-Transferred Session Pattern)

An attacker attempts to access an administration portal repeatedly, generating a stream of authorization requests to the network administrator. The attempts are timed to occur while the administrator is asleep. The administrator is woken by the incoming requests on their phone, and, in an attempt to stop the notifications, they accidentally approve access and the attacker gains access to the portal.
What’s New: SHOULD, RECOMMENDED and MAY

• Discussed at IETF 117

• Several “should, may, recommended”, no “SHOULD, MAY or RECOMMENDED”
  • Applies to the Authorization Server, Resource Server or Client

• Why change
  • Provide clear guidance to implementors
  • Emphasise importance of mitigations
  • Make conformance\adoption meaningful
What’s New: Editorial updates

• Editorial scrub
• Adopted the OpenID Foundation terminology from CIBA
• Acknowledgements
  • Marco Pernpruner
  • Giada Sciarretta
  • Maryam Mehrnezhad
Where do we go Next?
<table>
<thead>
<tr>
<th>Issue</th>
<th>Description</th>
<th>Status</th>
<th>Created</th>
<th>Assigned</th>
<th>Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consider fine-tuning the pattern descriptions and diagrams</td>
<td>#104 opened on Sep 19 by danielfett</td>
<td>4 tasks</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reference ISO mdl</td>
<td>#100 opened on Sep 8 by danielfett</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rewrite formal analysis section</td>
<td>#97 opened on Sep 8 by danielfett</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Next Steps

• Update Formal Analysis section (December)
• WG Last Call before IETF 119?
Questions?