Updating the Security BCP

Tim Würtele[1] Pedram Hosseyni[1] Kaixuan Luo[2] Adonis Fung[3]

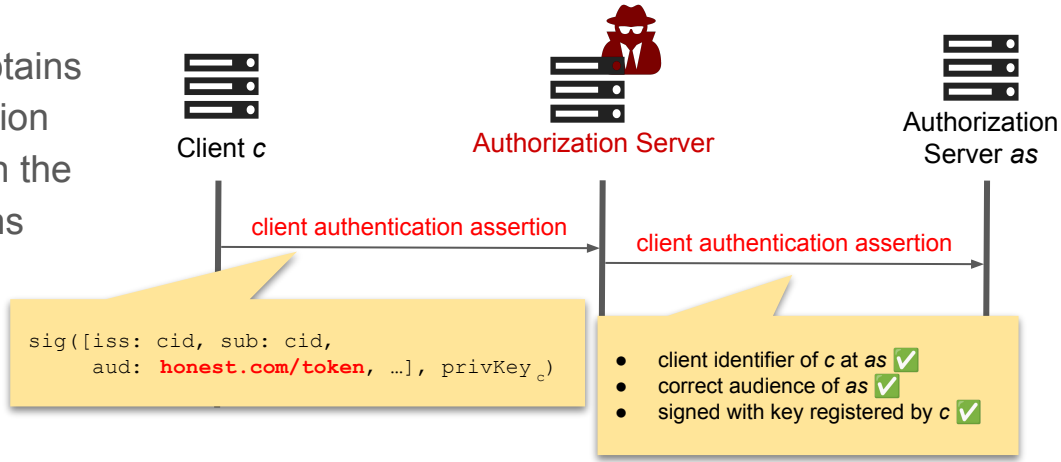
[1]University of Stuttgart, Germany
[2]The Chinese University of Hong Kong, Hong Kong SAR, China
[3]Samsung Research America, USA

Background / History

- Feb 27: New attacks after finalization of RFC9700, discussed at OSW'25 w/ original authors
  - Audience Injection Attacks https://talks.secworkshop.events/osw2025/talk/R8D9BS
    @WG Interim Meeting https://datatracker.ietf.org/meeting/interim-2025-oauth-04/session/oauth
  - Mix-up Attack Variants https://talks.secworkshop.events/osw2025/talk/WG9TEW
- Mar 18: Discussed also at IETF 122
  - Instead of RFC9700 style (everything in one doc), can "collect multiple RFCs under one BCP."
  - "a small focused doc", "go fast, publish, don't delay", "start work on the doc asap"
- Jun 17: Given the "consensus" to start a draft, here's the initial version
  - https://datatracker.ietf.org/doc/draft-wuertele-oauth-security-topics-update

Audience Injection Attacks

- Detailed description of the attack
  - Attack setting and requirements
  - Description of how the attacker obtains a valid client authentication assertion
  - List of example endpoints at which the attacker can obtain such assertions
- Proposed countermeasures
  - Single audience value
  - Issuer identifier
  - Exact URI of the target endpoint



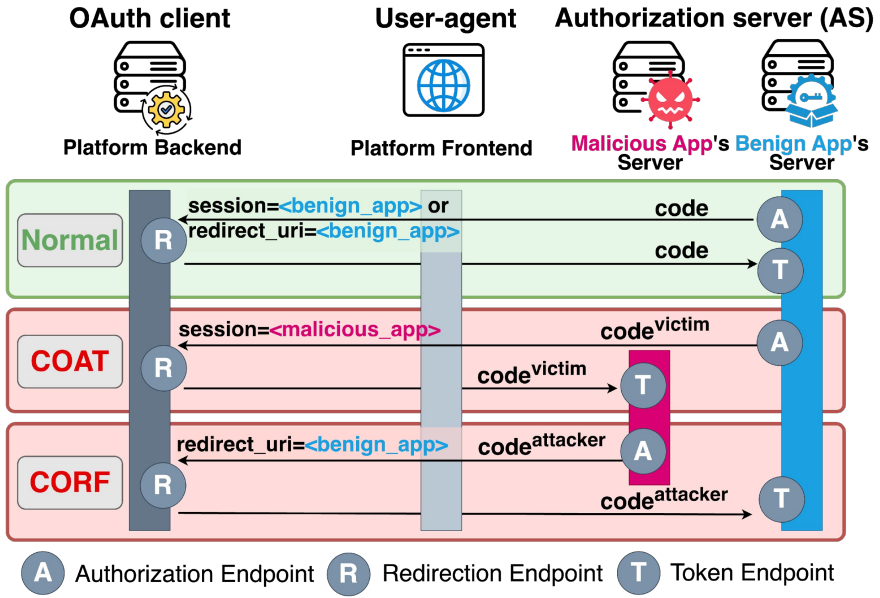
Mix-up Attack Variants: Motivations for a new draft, fast

- Prevalent attacks found everywhere
  - Impacted Google, Microsoft, Amazon, Samsung, Xiaomi, Baidu, Alibaba, etc.
  - Partially attribute to the lack of clear standards/practices in platform settings
  - Joint research of CUHK and Samsung. Details published in USENIX Security ’25 [1]
- Severity
  - Worst case: 1-click account takeover for any apps integrated in a platform (COAT)
  - e.g., CVE-2023-36019 (CVSS 9.6/10) for 1-click takeover of Outlook Emails & Azure Vault secrets
- Urgency: Problems repeating in new Agentic AI ecosystems

[1] https://www.usenix.org/conference/usenixsecurity25/presentation/luo-kaixuan

Overview of Mix-up Attack Variants

- Platform integrated w/ many apps
  (instead of one app w/ many IdPs for SSO)
  - Cross-app OAuth Account Takeover (COAT)
  - Cross-app OAuth Request Forgery (CORF)
- What's lacking in RFC9700 Section 4.4
  - Platform context w/ Open Ecosystem
  - Shared issuer: two apps share the same AS (e.g., 2 Dropbox apps/integrations)



Root Cause and Fix

- Enforce “App that starts” = “App that completes” w/ per-app redirect_uri
  - As well-received and adopted by many companies we help fixed
  - No normative/protocol changes, but best practice tailored for multi-app platforms
  - Does NOT invalidate existing issuer-based defenses as defined in RFC9700 & 9207
  - Introduced as alternative defense (Section 2.3.3)
- Addressing comments from Daniel Fett (original author), Guido, Aaron, et al.
  - Draft written more with the tone of OAuth language
  - Using “*Client Configuration*” to capture the “integrated app” concept (Section 2.3.1)
  - Also added clarifications and other security recommendations over RFC9700 Section 4.4
- Documented in Section 2.2 and 2.3

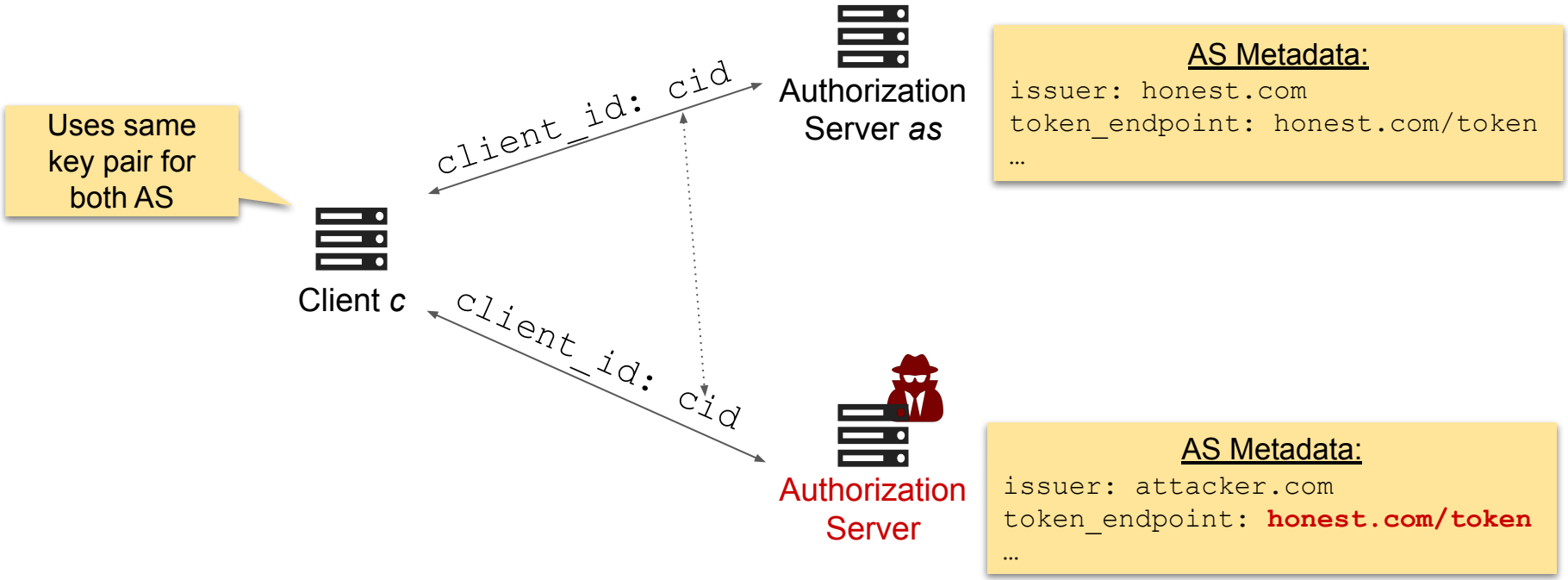
Towards Working Group Draft

- What more does it take to make it a working group draft?
- Expected timeline? Still “go fast, publish, don't delay”?
- Title?
  - New Security BCP, Updates to Security BCP, etc. ?
- Others?

- Thanks! and surely, further reviews on the content are most welcomed
  https://datatracker.ietf.org/doc/draft-wuertele-oauth-security-topics-update

Backup Slides

Audience Injection Attacks (Section 2.1)



Updates to Mix-Up Attacks (Section 2.2)

- Mix-up Attacks with Per-AS Redirect URIs
  - Basic Mix-up in RFC9700 assumes shared redirect_uri as precondition
  - Under distinct redirect_uri setup: RFC 9700 only covers “Cross Social-Network Request Forgery”
  - Two more mix-up variants under this category
    - [NEW] Advanced Mix-up Attack
      - Each AS has distinct redirect_uri
      - Client fails to compare redirect_uri with session
    - [NEW] Naïve RP Session Integrity Attack
  - Studied by Daniel Fett et al. and us
  - Prevalent in the wild

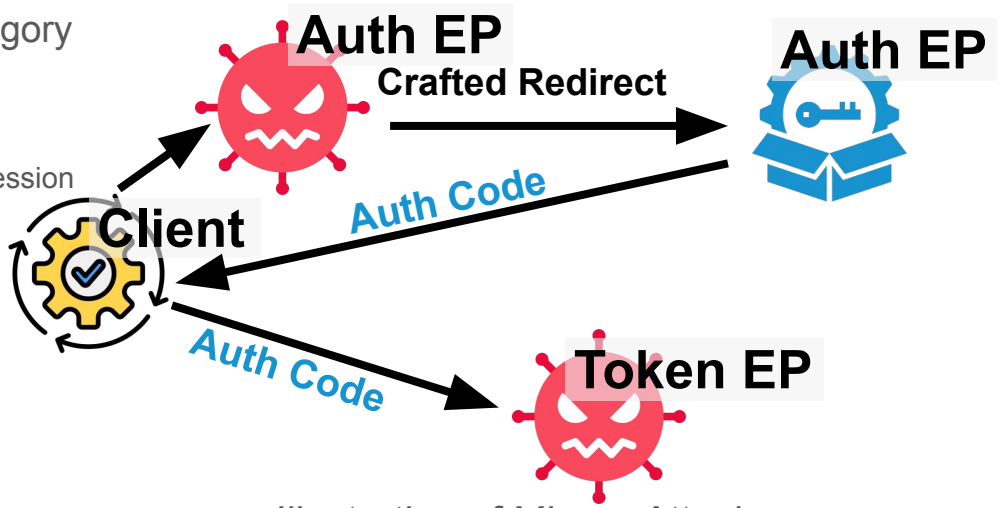
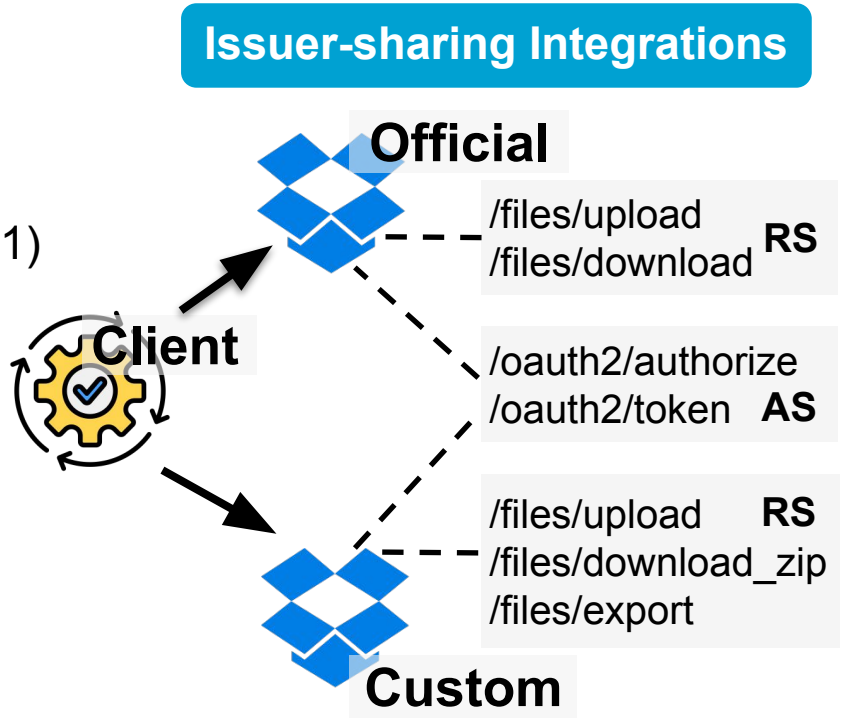


Illustration of Mix-up Attack

Attacks in Open Ecosystems (Section 2.3)

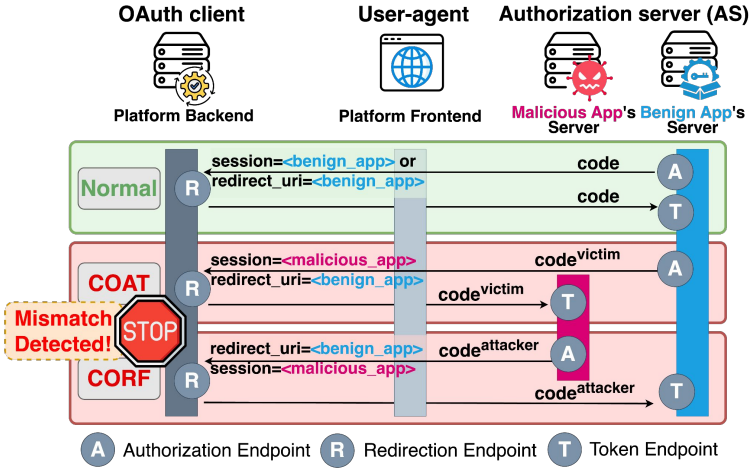
- Open Ecosystems (Sec 2.3.1)
  - Integration Platforms
  - Model Context Protocol (MCP)
- New Concept: Client Configuration (Sec 2.3.1)
  - AS config (manual or RFC8414)
  - Client registration (manual or RFC7591)
  - RS config (manual or RFC9728)
- Possibility of shared AS issuers (Sec 2.3.2)

  e.g.,
  - Same AS
  - Different RS
  - Built by different developers



Attacks in Open Ecosystems (Section 2.3)

- Mix-Up Attacks Reloaded (Section 2.3.3)
  - Defense: Differentiate each *AS Issuer* -> each *Client Configuration* w/ distinct redirect_uri, enforce matching at redirection EP
  - Introduced as alternative defense (“MAY”)
  - Does NOT invalidate existing issuer-based defenses in RFC9700 & 9207



AS Issuer:
`iss` claim in RFC8414 OAuth AS Metadata or an abstract identifier for the AS' <authorization endpoint, token endpoint>

Integrated App / Integration (aka Client Configuration):
- AS <authorization endpoint, token endpoint>
- Client Registration
- RS

Attacks in Open Ecosystems (Section 2.3)

- New Attack: Client Configuration Confusion Attack (Section 2.3.4)
  - Also based on the shared-issuer setting
  - Malicious Client Configuration may reuse a registered client
  - Attack: Send auth code from an honest AS (w/ a registered client) to an attacker's RS
  - Defense: Distinct redirect_uri for each app (client configuration)
  - Complementary to *general defenses* for token misuses
    - e.g., sender-constrained & audience-restricted access tokens
    - Related discussion: OAuth 2.1 issue #215