IETF 95 Buenos Aires

OAuth 2.0 for Apps (Draft BCP)
William Denniss & John Bradley
Embedded User Agent (WebView)
48% of users abandon app sign-in flows when no authentication state is present.
In addition, WebView doesn’t protect the session

The host app can extract the cookies:

```java
String cookies = CookieManager.getInstance().getCookie(url);
```

Or inject javascript:

```javascript
webView.evaluateJavascript(
    "(function() { return document.getElementById('password').value; })();",
    new ValueCallback<String>() {
      @Override
      public void onReceiveValue(String s) {
        Log.d("WebViewField", s);
      }
    });
```
Even trusted developers can create risk
Draft Best Current Practice for Apps


1. Apps should perform Authorization Requests in an External User Agent (i.e. a browser).
2. Authorization Servers should not assume all clients are confidential.
3. Custom URI schemes should be supported for OAuth redirects.
4. RFC7636 (PKCE) should be used.
In-App Browser Tabs

- Secure Context (External User Agent)
- Shared cookie state
- Presented in the app (no app switch)

Android

iOS

Chrome Custom Tabs

SFSafariViewController
AppAuth Demo

Sign in with:

- Google
- PingFederate

auth request

auth response

auth response

Google

Choose an account

Bill Dee
billd1600@gmail.com

Bill D Work
bill.d.work@icog.net

Add account

https://accounts.google.com

https://youtu.be/A5KmUihRIs
Most users & platforms supported through using the browser when in-app browser tabs are not available.
OAuth 2.0 Authorization Server

Authorization Request

Authentication Server

OAuth 2.0 Token Endpoint

Browser Authentication Flow (black box)

User Authentication

password, two-factor, Mutual TLS, FIDO, whatever!

User Authentication Response

Authorization Code + PKCE Verifier

Access Token & Refresh Token

Authorization Response

App

OAuth 2.0 Authorization Server

Authentication Server

OAuth 2.0 Token Endpoint
Advanced Options: Mutual TLS with Certificates
Advanced Options: FIDO U2F over NFC!

YubiKey NEO

https://youtu.be/egt7MiVBnb4
Interop

The following authorization servers already support the requirements of the best practice:

Google

Ping Identity

gluu
Interop in Progress

Django OIDC Provider
https://github.com/juanifioren/django-oidc-provider/

- Open source OAuth + OIDC server
- Supports Custom URI schemes
- Public client & PKCE support in progress
AppAuth: OAuth Client Libraries for Apps

Google made the initial contribution of AppAuth for iOS and Android to the Connect Working Group of the OpenID Foundation.

iOS library: [http://openid.github.io/AppAuth-iOS](http://openid.github.io/AppAuth-iOS)

I-D ChangeLog & Status


- Adoption by the WG following IETF94.
- Editorial refactoring, but core recommendations remain the same.
- Open source client libraries for iOS & Android released.
- Authorization Server interop progressing well.
- Plan to move to WG Last Call before Berlin.
THANK YOU
Android Custom Tabs

Common Android API available to all browsers. See: CustomTabsService. Provided through a support library (API level 4+) http://developer.android.com/tools/support-library/features.html#custom-tabs

Any browser can add Custom Tab support. Example provider implementation: https://github.com/GoogleChrome/custom-tabs-provider

Example client app:
https://github.com/GoogleChrome/custom-tabs-client