International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 4 Issue 4, June 2020 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Flaws in Oauth 2.0: Can Oauth be used as a Security Server

Pooja Krushna Paste, Pratik Ramakant Vaidya

Department of MCA, ASM Institute of Management & Computer Studies, Thane, Maharashtra, India

ABSTRACT

OAuth 2.0 is the business standard convention for approval. OAuth 2.0 spotlights on customer engineer straightforwardness while giving explicit approval streams to web applications, work area applications, cell phones, and lounge room gadgets. The scientists analyzed 600 top U.S. also, ChAndroid versatile applications that utilization OAuth 2.0 APIs from Facebook, Google and Sina—which works Weibo in China—and backing SSO for outsider applications. The scientists found that 41.2 percent of the applications they tried were defenseless against their attackinese.

KEYWORDS: OAuth, Proxy Servers, Vpns, Authorization tokens

How to cite this paper: Pooja Krushna Paste | Pratik Ramakant Vaidya "Flaws in Oauth 2.0: Can Oauth be used as a Security Server" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-4 | Issue-4, June 2020, pp.997-999, URL: www.ijtsrd.com/papers/ijtsrd31319.pdf

IJTSRD31319

1. INTRODUCTION

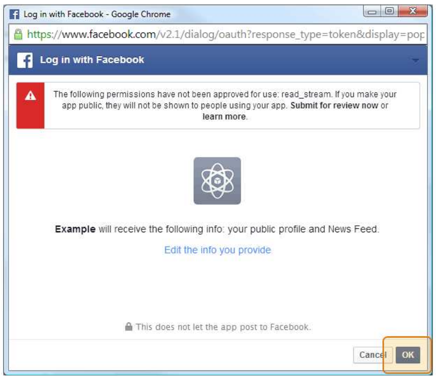
The OAuth 2.0 convention is one of the most broadly sent approval/single sign-on (SSO) conventions and furthermore fills in as the establishment for the new SSO standard OpenID Connect. Notwithstanding the notoriety of OAuth, so far examination endeavors were generally focused at discovering bugs in explicit executions and depended on formal models which conceptual from many webs includes or didn't give a proper treatment by any stretch of the imagination. In this paper, we do the primary broad conventional examination of the OAuth 2.0 standard in an expressive web model. Our investigation targets setting up solid approval, verification, and meeting uprightness ensures, for which we give formal definitions. In our proper investigation, each of the four OAuth Grant Types (approval code award, certain award, asset Owner Password Credentials Grant, and the customer Credentials Grant) are secured. They may even run all the while in the equivalent and distinctive depending gatherings and Identity Provider (IDP), where noxious depending parties, personality suppliers, and programs are considered also. Our demonstrating and examination of the OAuth 2.0 standard expect that security proposals and Best Practices are followed, so as to stay away from clear and known assaults.

2. OAuth Review-

The OAuth and Google Sign-In connecting type includes Google Sign-In top of OAuth based record connecting. This gives consistent voicebased connecting to Google clients while likewise empowering account connecting for clients who enrolled to your administration with a non-Google personality. This connecting type starts with Google Sign-In, which permits you to check if the client's Google profile data exists in your framework. In the event that the client's data isn't found in your framework, a standard OAuth stream starts. The client can likewise decide to make another record with their Google profile data

I. OAuth Used by Facebook-

267 million Facebook users' data has supposedly been leaked. Comparitech and security analyst Bob Diachenko have revealed a database containing in excess of 267 million Facebook users' data that was left uncovered on the web, with not so much as a secret key forestalling unapproved access to it. On the off chance that you've at any point utilized a "Sign in With Facebook" catch, or given an outsider application access to your Twitter account, you've utilized OAuth. It's additionally utilized by Google, Microsoft, and LinkedIn, just as numerous other record suppliers. Basically, OAuth permits you to concede a site access to some data about your record without giving it your genuine record secret key.



Log in with Facebook - Google Chrome

https://www.facebook.com/v2.1/dialog/oauth?response_type=token&display=pop

Log in with Facebook

The following permissions have not been approved for use: read_stream. If you make your app public, they will not be shown to people using your app. **Submit for review now** or **learn more**.

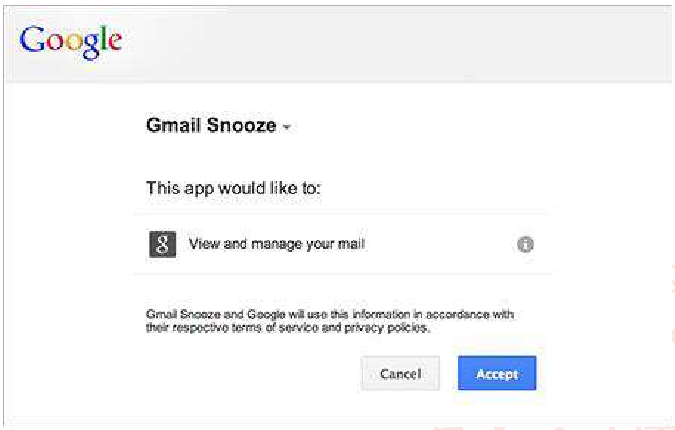
Example will receive the following info: your public profile and News Feed.

Edit the info you provide

This does not let the app post to Facebook.

II. OAuth used by Google-

Google APIs utilize the OAuth 2.0 convention for confirmation and approval. Google bolsters regular OAuth 2.0 situations, for example, those for web server, customer side, introduced, and constrained information gadget applications. To start, acquire OAuth 2.0 customer accreditations from the Google API Console. At that point your customer application demands an entrance token from the Google Authorization Server, extricates a token from the reaction, and sends the token to the Google API that you need to get to. For an intelligent exhibit of utilizing OAuth 2.0 with Google (counting the alternative to utilize your own customer qualifications), try different things with the OAuth 2.0 Playground.



Google

Gmail Snooze ▾

This app would like to:

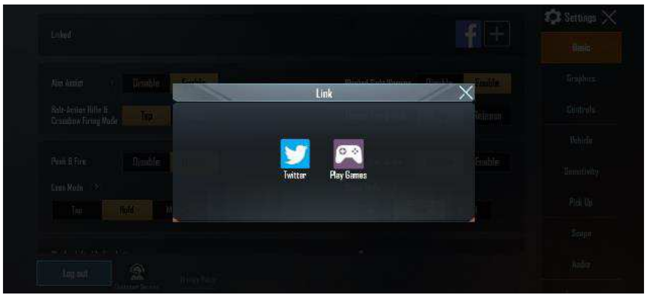
View and manage your mail

Gmail Snooze and Google will use this information in accordance with their respective terms of service and privacy policies.

Cancel Accept

III. Games that use OAuth-

This record clarifies how applications introduced on gadgets like telephones, tablets, and PCs utilize Google's OAuth 2.0 endpoints to approve access to Google APIs. OAuth 2.0 permits clients to impart explicit information to an application while keeping their usernames, passwords, and other data private. For instance, an application can utilize OAuth 2.0 to get consent from clients to store records in their Google Drives. Introduced applications are disseminated to singular gadgets, and it is accepted that these applications can't keep insider facts. They can get to Google APIs while the client is available at the application or when the application is running out of sight. This approval stream is like the one utilized for web server applications. The fundamental contrast is that introduced applications must open the framework program and gracefully a neighborhood divert URI to deal with reactions from Google's approval server.



Linking games with your google play store account has always been secure but what if we start to link another social media app with the application currently using than it may result in data lose or even by using proxy server s can be used to present a user a fake identity. Or an attacker can use another username and can log into the game as another user.

IV. Less Risks with short time tokens-

Divert URLs are a basic piece of the OAuth stream. After a client effectively approves an application, the approval server will redirect the client back to the application with either an approval code or access token in the URL. Because of approval sidestep in redirect Uri parameter in OAUTH stream; it's conceivable to divert confirmed clients to subjective spaces with their OAuth accreditations from which it's conceivable to take over their record. On the off chance that an approval code is utilized more than once, the approval server MUST deny the solicitation. OAuth Providers (servers) that carefully follow rfc6749 are defenseless against open divert. Approval repudiated doesn't send an alarm.

V. What Goes in a token Granted-

- request scope: contacts.
- response type: code (implicit or others).
- callback URL.
- client Id

3. Literature review

Redirect URLs are a fundamental bit of the OAuth stream. After a customer viably supports an application, the endorsement server will redirect the customer back to the application with either an endorsement code or access token in the URL. As a result of endorsement avoid in redirect Uri parameter in OAUTH stream, it's possible to occupy affirmed customers to emotional spaces with their OAuth accreditations from which it's possible to take over their record. In case an endorsement code is used more than once, the endorsement server MUST deny the requesting. OAuth Providers (servers) that cautiously follow rfc6749 are vulnerable against open occupy. Endorsement renounced doesn't send an alert.

4. Research Methodology-

OAuth is being used widely for authorization but it is said that It is not so far god for authentication. And thus, now they use a (sso) that is a single sign on. The sso resides as a thin layer above the OAuth. But now let's take a close look at the problems.

5. Problem Statement-

OAuth only takes care about the authorization as its mechanism works with granting and providing tokens to the thirdparty applications on the basis of the permissions granted, however let's take a scenario where the user log's in with its real account the OAuth verifies by sending the alert the token is generated. And as per the working when the token is sent back it carries the details which are vulnerable to attacks. If the user by using a proxy server or a better Vpns gets this sent back token from the OAuth, Now the attacker can easily change the username in the token as OAuth only sends and uses username, thus if the username is changed in the url than the user will login successfully by another username and thus this creates a problem that the OAuth does not provide the user with any security once the token are granted. Thus, the entered username is of another user and thus the attacker can easily login with its name. and the real user of that name is unaware that his Id is being used by someone.

6. Conclusion-

Leaf certificate. By sticking against your leaf testament, you are ensuring with near 100% sureness this is your declaration and along these lines the chain is substantial. Leaf endorsements will in general have a short expiry time and if, for example. On the off chance that the solicitation flops because of a missing, invalid, or jumbling redirection URI, or if the customer identifier is absent or invalid the approval server SHOULD educate the asset proprietor regarding the error and MUST NOT consequently divert the client specialist to the invalid redirection URI. Tokens ought not to make some long memories expiry date. For android gadgets the information ought not to be put away on neighbourhood stockpiling. Transient expiry tokens are helpful.

7. Future Enhancements-

OAuth can use security alerts to the user after the tokens are generated. Thus, this will create an alert for the user that his account has been used, which will result either removing the data of game from the google account which is not being linked by the user.

8. Reference-

[1] Security Flows in OAuth 2.0 Framework: A Case Study: {https://www.researchgate.net/publication/319453579}

[2] OAuth 2.0 Hack Exposes 1 Billion Mobile Apps to Account Hijacking: (https://threatpost.com/OAuth-2-0-hackexposes-1-billion-mobile-apps-to-accounthijacking/121889/).

[3] Top 10 OAuth 2 Implementation Vulnerabilities: (http://blog.intothesymmetry.com/2015/12/top-10-OAuth-2-implementation.html).

[4] OAuth authentication fails in a proxy scenario between Exchange Server 2013 hybrid on-premises and Office 365: (https://support.microsoft.com/enin/help/3137585/OAuth-authenticationfails-in-a-proxy-scenario-betweenexchange-server).

[5] Four Attacks on OAuth - How to Secure Your OAuth Implementation: https://www.sans.org/readingroom/whitepapers/application/attacks-OAuth-secure-OAuth-implementation-33644).