<https://www.researchgate.net/publication/325732203_An_Empirical_Study_of_OAuth-Based_SSO_System_on_Web>

An Empirical Study of OAuth-Based SSO System on Web

## **Abstract**

More and more websites use OAuth 2.0 protocol to provide SSO services to ease password management for users. Although OAuth 2.0 has been implemented carefully by following many guidelines, still some parts have been ignored. In this paper, we discover a new attack mode for hijacking the account in the OAuth-based SSO system. We conduct an empirical study for the proposed attack on top 500 Chinese websites of Alexa supporting SSO services by 6 IdPs. Our results uncover four vulnerabilities that allow attackers hijack the victim’s account without knowing the user’s username and password. Closer examination reveals that 68.67%, 12.87%, 68.67% and 59.66% of the websites are vulnerable to the four vulnerabilities respectively and 45.49% of the websites can be conducted proposed complete attack. To defend this attack, we provide developers simple practical recommendations to the critical vulnerable nodes.

<https://www.researchgate.net/publication/262158854_The_devil_is_in_the_implementation_details_an_empirical_analysis_of_OAuth_SSO_systems>

# The devil is in the (implementation) details: an empirical analysis of OAuth SSO systems

Millions of web users today employ their Facebook accounts to sign into more than one million relying party (RP) websites. This web-based single sign-on (SSO) scheme is enabled by OAuth 2.0, a web resource authorization protocol that has been adopted by major service providers. The OAuth 2.0 protocol has proven secure by several formal methods, but whether it is indeed secure in practice remains an open question. We examine the implementations of three major OAuth identity providers (IdP) (Facebook, Microsoft, and Google) and 96 popular RP websites that support the use of Facebook accounts for login. Our results uncover several critical vulnerabilities that allow an attacker to gain unauthorized access to the victim user's profile and social graph, and impersonate the victim on the RP website. Closer examination reveals that these vulnerabilities are caused by a set of design decisions that trade security for implementation simplicity. To improve the security of OAuth 2.0 SSO systems in real-world settings, we suggest simple and practical improvements to the design and implementation of IdPs and RPs that can be adopted gradually by individual sites.

<https://www.researchgate.net/publication/289587726_A_Comprehensive_Formal_Security_Analysis_of_OAuth_20>

# A Comprehensive Formal Security Analysis of OAuth 2.0

## **Abstract**

The OAuth 2.0 protocol allows users to grant relying parties access to resources at identity providers. In addition to being used for this kind of authorization, OAuth is also often employed for authentication in single sign-on (SSO) systems. OAuth 2.0 is, in fact, one of the most widely used protocols in the web for these purposes, with companies such as Google, Facebook, or PayPal acting as identity providers and millions of websites connecting to these services as relying parties. OAuth 2.0 is at the heart of Facebook Login and many other implementations, and also serves as the foundation for the upcoming SSO system OpenID Connect. Despite the popularity of OAuth, so far analysis efforts were mostly targeted at finding bugs in specific implementations and were based on formal models which abstract from many web features or did not provide a formal treatment at all. In this paper, we carry out the first extensive formal analysis of the OAuth 2.0 standard in an expressive web model. Our analysis aims at establishing strong authorization and authentication guarantees, for which we provide formal definitions. In our formal analysis, all four OAuth grant types are covered. They may even run simultaneously in the same and different relying parties and identity providers, where malicious relying parties and identity providers are considered as well. While proving security, we found two previously unknown attacks on OAuth, which both break authorization and authentication in OAuth. The underlying vulnerabilities are present also in the new OpenID Connect standard and can be exploited in practice. We propose fixes for the identified vulnerabilities, and then, for the first time, actually prove the security of OAuth in an expressive web model. In particular, we show that the fixed version of OAuth provides the authorization and authentication properties we specify.