**Image for the blog background (choose one that fit well):**

* <https://pixabay.com/photos/chain-stainless-steel-metal-iron-4049725/>
* <https://pixabay.com/photos/keys-open-locks-security-unlock-525732/>
* <https://pixabay.com/photos/padlock-lock-chain-key-security-597495/>

**Title:**

How to test an OAuth2/OpenID Connect based system?

**Abstract:**

Delegation of the authorization/authentication via the leveraging of OAuth2/OpenID Connect (OIDC) is more and more common in modern systems but how ensure that no weaknesses were left during the implementation and the configuration or such systems? Let's we show you some key points to verify…

**GitHub repository associated that will be moved to public when blog post will be released:**

<https://github.com/ExcelliumSA/OAuth2OIDC-Study>

**SEO rules indicated by Mathilde:**

* Paragraphs with fewer than 300 words.
* Keyword used as much as possible: *oauth, oidc, security*
* Presence of sections.

**Author(s):** Dominique Righetto

**Note:** In the content below, all figure captions refer to the image file that Mathilde must be when she creates the blog post.

# Introduction

This post will present security oriented validation points that should be verified on a system using OAuth/OpenID Connect (called OIDC in the rest of the post). Therefore, it assumes that the reader is familiar with all the concepts related to OAuth/OIDC. All references to OAuth refer to OAuth 2.0

If it is not the case then the reader can refer to this free online course "*Introduction to OAuth 2.0 and OpenID Connect*" [1] kindly created and provided by Dr. Philippe De Ryck [2].

# Context

xxx

# Validation points

xxx

# How to apply control?

xxx

# Conclusion

Xxx

# References

1. <https://courses.pragmaticwebsecurity.com/courses/introduction-to-oauth-2-0-and-openid-connect>
2. <https://pragmaticwebsecurity.com/about.html>