Cyprus University of Technology

Department of Electrical Engineering and Computer Engineering and Informatics

Master Thesis

CSP Compliance Through Time

Charalampos Michailidis

Limassol, May 2026

CYPRUS UNIVERSITY OF TECHNOLOGY

Faculty of Engineering and Technology

Department of Electrical Engineering, Computer Engineering, and Informatics

Master Thesis

CSP Compliance Through Time

Charalampos Michailidis

Advisor:

Dr Panagiotis Ilia

Limassol, May 2026

Copyrights

Copyright © 2026 Charalampos Michailidis

All rights reserved.

The approval of the dissertation by the Department of Electrical Engineering, Computer Engineering, and Informaticsdoes not necessarily imply the approval by the Department of the views of the writer.

Acknowledgements

I would like to express my sincere gratitude to my professors, Dr. Panagiotis Ilia and Dr. Costas, for their valuable guidance throughout my entire thesis. They have provided critical technical advice on the implementation of the puppeteer CSP auditing system, that helped me to build a robust and reliable tool. I would also like to thank my friends and family for their support and understanding of the demanding work i had throughout the year.

ABSTRACT

Web applications have become a central part of everyday life as they are supporting a lot of services like online banking, communication, healthcare and a lot of other utilities. As these systems continue to evolve, they rely more heavily on dynamic content like external services and client-side scripting making security increasingly difficult to manage.Among that the most common threats facing modern websites are Cross-Site Scripting (XSS) attacks, where malicious code is executed inside a user's browser through a trusted application. To reduce the likelihood of those attacks to happen the browsers can use Content Security Policy (CSP) which is a mechanism that allows websites to define which resources are trusted and allowed to execute in the browser. Although CSP is widely recommended as a second layer of defense the implementation of it in practice is often inconsistent. a lot of web applications adjust their policies over time to support new features, third-party libraries or even changing the development requirements that they need over time. In a lot of cases those adjustments result in policies that become highly permissive with result to reduce the level of protection of Content Security Policy. This thesis intent to investigates how CSP policy is changing over time and how these changes can affect the security state of the current web applications. To study this behavior we have developed an automated framework to collect and analyze CSP policies and browser resource activity from around 2,000 popular websites collected from the Tranco ranking list. This analysis was performed over multiple time periods and included homepage, login, and sign up pages to be able to observe differences across different application context. This collected data were used to analyze the policy modifications, CSP violations and resources to find the overall strength of those policies. We have give a lot of attention to insecure practices such as the use of wildcard sources and unsafe directives, which were commonly adopted to simplify development but significantly weaken the protection of CSP given to the browser. The results we have find show us that while CSP adoption is relatively common among popular websites but most of them using it with unsecure configurations. As many websites rely on permissive configurations that prioritize functionality and flexibility in development over secure configurations that will make the development more complex and challenging. CSP directives like unsafe-inline, unsafe-eval and wildcard allowances were observed in frequent base often reducing the effectiveness of CSP against XSS attacks. Also this thesis shows that weak configurations have remained stable over time rather than improving suggesting us that insecure practices are often maintained long term to avoid complex development with secure practices. In conclusion this work show us that the CSP policy is not used properly as it indent to be used for the security purposes and highlights how important is to monitor that policy along with the resources in order to find insecure practices

Keywords: Puppeter automation, web security auditing, Content Security Policy, CSP analysis.