UDC 004.056

INVESTIGATION OF SERVER VULNERABILITIES ON WEBAPP USING CROSS-PLATFORM NODE.JS

Nazarii Dzhaliuk

Lviv Polytechnic National University

nazarii.dzhaliuk.kb.2019@lpnu.ua

Node.js applications are increasing in number and they are no different from other frameworks and programming languages. Node.js applications are prone to all kinds of web application vulnerabilities. Node.js itself is a secure platform – but many third-party open source packages in the ecosystem may not be. In the Node.js threat model, there are trusted elements such as the underlying operating system. Currently, any package can access powerful system resources such as network access.

Packages are used through Node Package Manager (npm) and are susceptible to many different types of vulnerabilities and other Node.js licensing and security risks, opening up the housed applications to a lot of risks. All code running into a node process has the ability to load and run additional arbitrary code by using eval() (or its equivalents). All code with file system write access may achieve the same thing by writing to new or existing files which are loaded.

Node.js has an experimental policy mechanism to declare the loaded resource as untrusted or trusted. However, this policy is not enabled by default yet. Policies are a security feature intended to allow guarantees about what code Node.js is able to load. The use of policies assumes safe practices for the policy files such as ensuring that policy files cannot be overwritten by the Node.js application by using file permissions.

1. *Node.js* v20 Policy, https://nodejs.org/api/permissions.html#policies
2. *Node.js* treat model, https://github.com/nodejs/node/blob/main/SECURITY.md#the-nodejs-threat-model
3. *OWASP* Node.js security, https://cheatsheetseries.owasp.org/cheatsheets/Nodejs_Security_Cheat_Sheet.html

ДОСЛІДЖЕННЯ УРАЗЛИВОСТЕЙ ВЕБ-ДОДАТКІВ НА СТОРОНІ СЕРВЕРА ПРИ ЗАСТОСУВАННІ КРОС-ПЛАТФОРМОВОГО СЕРЕДОВИЩА NODE.JS

Кросплатформове середовище Node.js спроектовано так, що воно має повний доступ до системних ресурсів, що несе загрозу при використанні сторонніх залежностей, наприклад з репозиторію прт. Щоб вирішити цю проблему, було запропоновано експериментальну можливість видачі дозволів процесу Node.js для обмеження доступу до системних ресурсів.

http://iapmm.lviv.ua/mpmm2023/materials/mm11_07.pdf