ProTech Professional Technical Services, Inc.

protechtraining.com

Course Outline

Node.js Security

Course Summary

Description

Node.js is a fast-growing platform for building server applications using JavaScript. Now that it is being more widely used in production settings, Node applications will start to be specifically targeted for security vulnerabilities. Protecting your users will require an understanding of attack vectors unique to Node, as well as shared with other web applications.

To secure Node.js applications, we'll start by helping you delve into the building blocks that make up typical Node applications. By understanding all the layers that you are building on top of, you can write code defensively and securely. In doing so, you will be able to protect your user's data and your infrastructure, while still using the rock-star technology behind Node.js.

Teaching you how to secure your Node applications by learning about each of the layers you will be building on top of; starting with JavaScript itself, then the Node platform, and finally the npm module ecosystem. By starting with JavaScript, you will learn what to avoid and what to embrace. Next, we will explain the Node platform, including its unique architecture and core modules, so you know how things work under the hood. Finally, we will introduce the rich ecosystem of npm modules, including modules to help you solve the common security problems you might face. Through hands-on tutorials, you will be able to write secure Node.js applications, ones that will remain online under pressure and be able to weather the most common attacks that face web applications today.

- Examine security features and vulnerabilities within JavaScript
- Explore the Node platform, including the event-loop and core modules
- Solve common security problems with available npm modules

Objective

- Master the origins of the Node.js and npm projects
- Understand the architecture, including the event-loop and asynchronous I/O
- Delve into the key aspects of avoiding some common pitfalls of JavaScript development
- Incorporate ES5's security improvements, including strict-mode
- Add static code analysis and the code-quality it promotes
- Explore the basics of proper error-handling within Node applications
- Understand the architecture of Express and Connect
- Adapt common authentication and authorization schemes

Topics

- Introduction to Node.js
- General Considerations
- Application Considerations
- Request Layer Considerations
- Response Layer Vulnerabilities

Duration

Five Days

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically.

Node.js Security

Course Outline

I. ***Introduction to Node.js***

A. History of Node.js
B. How Node.js differs?
C. Securing Node.js applications

II. ***General Considerations***

A. JavaScript security
B. ES5 features
C. Static program analysis
D. Considerations for Node.js
E. npm modules (third-party code)

III. ***Application Considerations***

A. Introduction to Express
B. Authentication
C. Authorization
D. Security logging
E. Error handling

IV. ***Request Layer Considerations***

A. Limiting the request size
B. Monitoring the event loop's responsiveness
C. Cross-site Request Forgery
D. Input validation

V. ***Response Layer Vulnerabilities***

A. Cross-site Scripting (XSS)
B. Denial of Service
C. Security-related HTTP headers

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically.