front matter

preface

I have been a professional software developer, off and on, for about 20 years now, and I've worked with a wide variety of APIs over those years. My youth was spent hacking together adventure games in BASIC and a little Z80 machine code, with no concern that anyone else would ever use my code, let alone need to interface with it. It wasn't until I joined IBM in 1999 as a pre-university employee (affectionately known as "pooeys") that I first encountered code that was written to be used by others. I remember a summer spent valiantly trying to integrate a C++ networking library into a testing framework with only a terse email from the author to guide me. In those days I was more concerned with deciphering inscrutable compiler error messages than thinking about security.

Over time the notion of API has changed to encompass remotely accessed interfaces where security is no longer so easily dismissed. Running scared from C++, I found myself in a world of Enterprise Java Beans, with their own flavor of remote API calls and enormous weight of interfaces and boilerplate code. I could never quite remember what it was I was building in those days, but whatever it was must be tremendously important to need all this code. Later we added a lot of XML in the form of SOAP and XML-RPC. It didn't help. I remember the arrival of RESTful APIs and then JSON as a breath of fresh air: at last the API was simple enough that you could stop and think about what you were exposing to the world. It was around this time that I became seriously interested in security.

In 2013, I joined ForgeRock, then a startup recently risen from the ashes of Sun Microsystems. They were busy writing modern REST APIs for their identity and access management products, and I dived right in. Along the way, I got a crash course in modern token-based authentication and authorization techniques that have transformed API security in recent years and form a large part of this book. When I was approached by Manning about writing a book, I knew immediately that API security would be the subject.

The outline of the book has changed many times during the course of writing it, but I’ve stayed firm to the principle that details matter in security. You can’t achieve security purely at an architectural level, by adding boxes labelled “authentication” or “access control.” You must understand exactly what you are protecting and the guarantees those boxes can and can’t provide. On the other hand, security is not the place to reinvent everything from scratch. In this book, I hope that I’ve successfully trodden a middle ground: explaining why things are the way they are while also providing lots of pointers to modern, off-the-shelf solutions to common security problems.

A second guiding principle has been to emphasize that security techniques are rarely one-size-fits-all. What works for a web application may be completely inappropriate for use in a microservices architecture. Drawing on my direct experience, I’ve included chapters on securing APIs for web and mobile clients, for microservices in Kubernetes environments, and APIs for the Internet of Things. Each environment brings its own challenges and solutions.

acknowledgments

I knew writing a book would be a lot of hard work, but I didn’t know that starting it would coincide with some of the hardest moments of my life personally, and that I would be ending it in the midst of a global pandemic. I couldn’t have got through it all without the unending support and love of my wife, Johanna. I’d also like to thank our daughter, Eliza (the littlest art director), and all our friends and family.

Next, I’d like to thank everyone at Manning who’ve helped turn this book into a reality. I’d particularly like to thank my development editor, Toni Arritola, who has patiently guided my teaching style, corrected my errors, and reminded me who I am writing for. I’d also like to thank my technical editor, Josh White, for keeping me honest with a lot of great feedback. A big thank you to everybody else at Manning who has helped me along the way. Deirdre Hiam, my project editor; Katie Petito, my copyeditor; Keri Hales, my proofreader; and Ivan Martinovic´ , my review editor. It’s been a pleasure working with you all.

I'd like to thank my colleagues at ForgeRock for their support and encouragement. I'd particularly like to thank Jamie Nelson and Jonathan Scudder for encouraging me to work on the book, and to everyone who reviewed early drafts, in particular Simon Moffatt, Andy Forrest, Craig McDonnell, David Luna, Jaco Jooste, and Robert Wapshott.

Finally, I'd like to thank Jean-Philippe Aumasson, Flavien Binet, and Anthony Vennard at Teserakt for their expert review of chapters 12 and 13, and the anonymous reviewers of the book who provided many detailed comments.

To all the reviewers, Aditya Kaushik, Alexander Danilov, Andres Sacco, Arnaldo Gabriel, Ayala Meyer, Bobby Lin, Daniel Varga, David Pardo, Gilberto Taccari, Harinath Kuntamukkala, John Guthrie, Jorge Ezequiel Bo, Marc Roulleau, Michael Stringham, Ruben Vandeginste, Ryan Pulling, Sanjeev Kumar Jaiswal (Jassi), Satej Sahu, Steve Atchue, Stuart Perks, Teddy Hagos, Ubaldo Pescatore, Vishal Singh, Willhelm Lehman, and Zoheb Ainapore: your suggestions helped make this a better book.

about this book

Who should read this book

API Security in Action is written to guide you through the techniques needed to secure APIs in a variety of environments. It begins by covering basic secure coding techniques and then looks at authentication and authorization techniques in depth. Along the way, you'll see how techniques such as rate-limiting and encryption can be used to harden your APIs against attacks.

This book is written for developers who have some experience in building web APIs and want to improve their knowledge of API security techniques and best practices. You should have some familiarity with building RESTful or other remote APIs and be confident in using a programming language and tools such as an editor or IDE. No prior experience with secure coding or cryptography is assumed. The book will also be useful to technical architects who want to come up to speed with the latest API security approaches.

How this book is organized: A roadmap

This book has five parts that cover 13 chapters.

Part 1 explains the fundamentals of API security and sets the secure foundation for the rest of the book.

- Chapter 1 introduces the topic of API security and how to define what makes an API secure. You'll learn the basic mechanisms involved in securing an API and how to think about threats and vulnerabilities.
- Chapter 2 describes the basic principles involved in secure development and how they apply to API security. You'll learn how to avoid many common software security flaws using standard coding practices. This chapter also introduces the example application, called Natter, whose API forms the basis of code samples throughout the book.
- Chapter 3 is a whirlwind tour of all the basic security mechanisms developed in the rest of the book. You'll see how to add basic authentication, rate-limiting, audit logging, and access control mechanisms to the Natter API.

Part 2 looks at authentication mechanism for RESTful APIs in more detail. Authentication is the bedrock upon which all other security controls build, so we spend some time ensuring this foundation is firmly established.

- Chapter 4 covers traditional session cookie authentication and updates it for modern web API usage, showing how to adapt techniques from traditional web applications. You'll also cover new developments such as SameSite cookies.
- Chapter 5 looks at alternative approaches to token-based authentication, covering bearer tokens and the standard Authorization header. It also covers using local storage to store tokens in a web browser and hardening database token storage in the backend.
- Chapter 6 discusses self-contained token formats such as JSON Web Tokens and alternatives.

Part 3 looks at approaches to authorization and deciding who can do what.

- Chapter 7 describes OAuth2, which is both a standard approach to token-based authentication and an approach to delegated authorization.
- Chapter 8 looks in depth at identity-based access control techniques in which the identity of the user is used to determine what they are allowed to do. It covers access control lists, role-based access control, and attribute-based access control.
- Chapter 9 then looks at capability-based access control, which is an alternative to identity-based approaches based on fine-grained keys. It also covers macaroons, which are an interesting new token format that enables exciting new approaches to access control.

Part 4 is a deep dive into securing microservice APIs running in a Kubernetes environment.

- Chapter 10 is a detailed introduction to deploying APIs in Kubernetes and best practices for security from a developer's point of view.
- Chapter 11 discusses approaches to authentication in service-to-service API calls and how to securely store service account credentials and other secrets.

Part 5 looks at APIs in the Internet of Things (IoT). These APIs can be particularly challenging to secure due to the limited capabilities of the devices and the variety of threats they may encounter.

- Chapter 12 describes how to secure communications between clients and services in an IoT environment. You'll learn how to ensure end-to-end security when API requests must travel over multiple transport protocols.
- Chapter 13 details approaches to authorizing API requests in IoT environments. It also discusses offline authentication and access control when devices are disconnected from online services.

About the code

This book contains many examples of source code both in numbered listings and in line with normal text. In both cases, source code is formatted in a `fixed-width` `font` `like` `this` to separate it from ordinary text. Sometimes code is also `in` `bold` to highlight code that has changed from previous steps in the chapter, such as when a new feature adds to an existing line of code.

In many cases, the original source code has been reformatted; we've added line breaks and reworked indentation to accommodate the available page space in the book. In rare cases, even this was not enough, and listings include line-continuation markers (➥). Additionally, comments in the source code have often been removed from the listings when the code is described in the text. Code annotations accompany many of the listings, highlighting important concepts.

Source code is provided for all chapters apart from chapter 1 and can be downloaded from the GitHub repository accompanying the book at https://github.com/NeilMadden/apisecurityinaction or from Manning. The code is written in Java but has been written to be as neutral as possible in coding style and idioms. The examples should translate readily to other programming languages and frameworks. Full details of the required software and how to set up Java are provided in appendix A.

liveBook discussion forum

Purchase of API Security in Action includes free access to a private web forum run by Manning Publications where you can make comments about the book, ask technical questions, and receive help from the author and from other users. To access the forum, go to https://livebook.manning.com/#!/book/api-security-in-action/discussion. You can also learn more about Manning's forums and the rules of conduct at https://livebook.manning.com/#!/discussion.

Manning's commitment to our readers is to provide a venue where a meaningful dialogue between individual readers and between readers and the author can take place. It is not a commitment to any specific amount of participation on the part of the author, whose contribution to the forum remains voluntary (and unpaid). We suggest you try asking the author some challenging questions lest his interest stray! The forum and the archives of previous discussions will be accessible from the publisher's website as long as the book is in print.

Other online resources

Need additional help?

- The Open Web Application Security Project (OWASP) provides numerous resources for building secure web applications and APIs. I particularly like the cheat sheets on security topics at **https://cheatsheetseries.owasp.org**.
- **https://oauth.net** provides a central directory of all things OAuth2. It's a great place to find out about all the latest developments.

about the author

Neil Madden is Security Director at ForgeRock and has an in-depth knowledge of applied cryptography, application security, and current API security technologies. He has worked as a programmer for 20 years and holds a PhD in Computer Science.

about the cover illustration

The figure on the cover of API Security in Action is captioned "Arabe du désert," or Arab man in the desert. The illustration is taken from a collection of dress costumes from various countries by Jacques Grasset de Saint-Sauveur (1757-1810), titled Costumes de Différents Pays, published in France in 1788. Each illustration is finely drawn and colored by hand. The rich variety of Grasset de Saint-Sauveur's collection reminds us vividly of how culturally apart the world's towns and regions were just 200 years ago. Isolated from each other, people spoke different dialects and languages. In the streets or in the countryside, it was easy to identify where they lived and what their trade or station in life was just by their dress. The way we dress has changed since then and the diversity by region, so rich at the time, has faded away. It is now hard to tell apart the inhabitants of different continents, let alone different towns, regions, or countries. Perhaps we have traded cultural diversity for a more varied personal life--certainly for a more varied and fast-paced technological life. At a time when it is hard to tell one computer book from another, Manning celebrates the inventiveness and initiative of the computer business with book covers based on the rich diversity of regional life of two centuries ago, brought back to life by Grasset de Saint-Sauveur's pictures.