Chapter 19. Part II Summary

Today's web applications are host to a wide number of vulnerabilities. Some of these vulnerabilities are easily classified, like the vulnerabilities we evaluated and tested in Part II. Other vulnerabilities are more of a niche—unique to a single application if that application has an uncommon security model or possesses features with unique architecture not found elsewhere.

Ultimately, thoroughly testing a web application will require knowledge of common vulnerability archetypes, critical thinking skills, and domain knowledge so that deep logic vulnerabilities outside of the most common archetypes can be found. The foundational skills presented in Parts I and II should be sufficient to get you up and running on any web application security pen-testing project you take part in in the future.

From this point forward, pay attention to the business model in any application you test. All applications are at risk of vulnerabilities like XSS, CSRF, or XXE, but only by gaining a deep understanding of the underlying business model and business logic in an application can you identify more advanced and specific vulnerabilities.

If the vulnerabilities presented in Part II feel difficult to apply in a real-world scenario, consider why that is the case. It is possible that whatever application you are testing is thoroughly hardened, but it's more likely that while you have developed the knowledge to develop and deploy these attacks, you may need to further improve or apply your recon skills in order to find weaknesses in the application where these attacks can be deployed successfully.

The skills learned in Part II of the book build directly on top of the skills from Part I. Additionally, they will serve you well as you move on to the

final part of this book regarding web application security: defensive mechanisms to protect against attacks.

Keep in mind both the recon techniques and offensive hacking techniques developed so far as you progress through the last part of this book. As you work through the defense examples, continually think to yourself how a hacker would find and exploit an application with and without proper defenses.

You will learn that web application defenses are often broken, which is why they are frequently referred to as "mitigations" rather than "fixes." With the knowledge from Parts I and II, you may be able to determine methods of bypassing or softening specific defenses in Part III. The defenses presented in Part III are mostly considered best practices in the industry, but many are not bulletproof, and multiple defenses should often be combined rather than relying on one at a time.

On a final note, the techniques presented in Part II are indeed dangerous. These are real attacks used by real attackers on a regular basis. You are welcome to test them against your own web applications, but please do not test them against web applications owned by others without explicit written permission from the web application's owner.

The techniques from the prior chapters can be used for both good and evil. As a result, the application and usage of these techniques must be considered thoroughly and not deployed on a whim.

Several of the techniques can also result in the compromise of servers or client machines, even when granted permissions from an application's owner. Keep in mind the impact of each individual attack, and make sure the application owner understands the risks involved with live testing prior to beginning.