# **L3: Discussion 1 - Mapping a Web Application**

No unread replies. No replies.

#### **Purpose**

You have learned how to use 3 toolkits (Zap, Burp Suite, and WebScarab). This discussion will enhance your learning of how to map a web application by exchanging your understanding and experience with your classmates.

#### **Tasks**

Please submit an original post focusing on the following questions and guidelines.

* Which toolkit (Paros or ZAP, Burp, or WebScarab) is your favorite?
* Describe one specific scenario or example you found that demonstrates how using *manual crawling* can overcome some limitations of *automated spidering*.

Personally, my favorite toolkit I’ve used so far is OWASP ZAP. I found it to be the most intuitive. I was able to discover the functionality with relative ease without having to read too much documentation. Burp Suite is similar, but there are a few more steps needed in order to achieve the same basic functionality (like turning intercept off). This is relatively minor, but it does make a difference if you are switching tools frequently to test different things. The other piece of ZAP that sets it apart is the “Alerts” section. It automatically analyzes requests and flags potential vulnerabilities. It also describes the vulnerability, tells you the potential method of attack that would be used to take advantage of it, and provides a solution.

One specific scenario where manual crawling will overcome the limitation of automated spidering is when submitting a form that has some form of input validation. For example, on the bWAPP webpage that comes pre-loaded on our linux VM, there is a form for account creation. You must enter a username, password, email, password confirmation, and a secret. The automated spider will attempt to submit something for this form. It does this by defaulting all of the fillable fields with “ZAP”. However, the email validation fails because it is not in an email format ([test@site.com](mailto:test@site.com)). The automated crawler is not smart enough to detect this and retry with the correct format. It also won’t always be able to tell what format the input needs to be. By manually crawling the site, you can enter data in the correct format the first time around (or after getting an error message). This means you can continue to explore the application structure that is locked behind the successful account creation.