

ZAP Proxy and OWASP Top 10

*Eduard Ramos Flores
Master of Computer Science, Cybersecurity
Dr. Jeff Duffany
Department of Computer Science
Polytechnic University of Puerto Rico*

Abstract — *The Zed Attack Proxy is a well-known and popular assessment tool in the cybersecurity community. The Open Web Application Security Project community offers, develops, and maintains the Zed Attack Proxy. The Open Web Application Security Project community also publishes the top ten security risks faced by web applications. Paired with the Zed Attack Proxy, The Open Web Application Security Project's top 10 security risks publication, serves as a baseline for security professionals assessing the security compliance of web applications. This study aims to evaluate the effectiveness and efficiency of the Open Web Application Security Project's Zed Attack Proxy tool against real-world production web applications and vulnerable by design penetration labs web applications.*

Key Terms — *Open-Source, Penetration Testing, Security Assessment, Web Applications.*

INTRODUCTION

Open Web Application Security Project [1] provides the Zed Attack Proxy tool [2] under the Apache Software Foundation 2.0 licensing model. To assess Zed Attack Proxy's capabilities, various scans of web applications were performed using two scan modes. Active Scanning mode is aggressive and will perform actual attacks on its targets. Therefore, the active scanning mode was used on the private web applications test environment. Passive scanning does not alter the traffic between the Zed Attack Proxy host and its target, and it does not perform attacks in any way. This type of scanning is legal to use when a target is a private web application publicly accessible on the internet. In this study, both types of scanning were used to demonstrate Zed Attack Proxy's functionality and capability assessing the Open Web Application Security

Project's top ten security risks [3]. The current top ten categories observed since 2021 are:

A01:2021-Broken Access Control

A02:2021-Cryptographic Failures

A03:2021-Injection

A04:2021-Insecure Design

A05:2021-Security Misconfiguration

A06:2021-Vulnerable and Outdated Components

A07:2021-Identification and Authentication Failures

A08:2021-Software and Data Integrity Failures

A09:2021-Security Logging and Monitoring Failures

A10:2021-Server-Side Request Forgery

Background and Context

According to a survey published in March 2020 [4], in around 120 countries and territories, 69% of participants consider Burpsuite the most useful software or tool for hacking. Leaving Web Proxies - the category where Zed Attack Proxy belongs- in fourth place with 25% popularity. That is an impressive 64% difference between both tools. The aim of this study is to assess the usability of Zed Attack Proxy with regard to the Open Web Application Security Project's own published top ten security risks. And to find the reason for the 64% difference in popularity for the tools.

PROBLEM STATEMENT

For-profit security assessment tools for web applications aim to provide ease of use, effective functionality, and accurate reporting. If a paid-for tool is mature, it can go as far as to provide the ability to perform advanced assessments using functions like automation for example. Security assessment software, paid or free, still requires knowledge and skillset considered intermediate to advanced. Is Zed Attack Proxy capable of successfully meeting the

needs of a security practitioner? Is there a handicap in Zed Attack Proxy that justifies monetary investment in a proprietary tool?

RELEVANCE AND IMPORTANCE

The main purpose of this study is to determine if the proper use of Zed Attack Proxy can result in an effective tool for security assessments of web applications. Thus, if the findings are correct, the need to pay for a tool like Burpsuite would be unnecessary.

Practical Considerations

According to Zed Attack Proxy documentation [5], the Attack scan mode can produce the best assessment results. The documentation also advises against the use of the Attack scan mode on web applications that are active in a production environment. Without explicit authorization from the owner, only passive scans should be performed in production web applications to avoid causing damage.

Practical Implications

The misuse of security assessment tools could result in damage to production web applications, their processes, and data. Additionally, it could lead to disruption of a critical system and could cause harm. Security assessment tools Zed Attack Proxy or Burp Suite must not be employed without explicit consent of the owners of targeted web application. Any security assessment tool should not be employed on any system without first guaranteeing the safety of human life.

ZED ATTACK PROXY AND BURP SUITE COMPARED

There are many popular vulnerability assessment tools that are free to use but have caveats. Of the many options available, two were selected as examples of the common trend found in the market, tools are free to try, and others offer a free version with limited functionality. For example, Pentest-tools.com [6] offers an online vulnerability scanner that allows for a light and full scan. The light scan

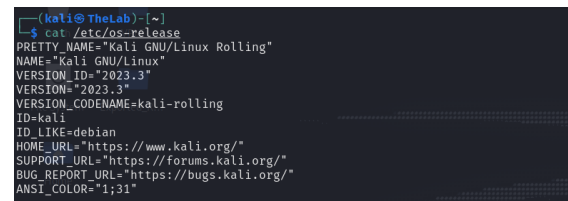
will detect vulnerabilities based on versions and misconfigurations. It will not perform assessments of SQL injections for example, or server-side request forgeries. These features are reserved for the full scan which is part of a monthly subscription system. Another popular vulnerability assessment tool is called Burp Suite, available from the company PortSwigger. The Open Web Application Security Project's community maintains Zed Attack Proxy. It is freely available and open to contributions from many developers around the world. Burp Suite is released in three versions, community, professional, and enterprise. The only version of Burp Suite that is free of charge is the community edition. In this study, a comparison of Zed Attack Proxy with Burp Suite Community edition is made considering that the Community edition of Burp is free of charge like Zed Attack Proxy. The Community edition though, lacks automated scanning and reporting capabilities included in Zed Attack Proxy. For this reason, Zed Attack Proxy is also compared with the Professional edition of Burp which contains those two features. The user interface of Zed Attack Proxy is intuitive, friendly, and extensible. It allows for ease of use by inexperienced users as well as more senior security experts. Zed Attack Proxy offers a Heads-Up Display (HUD) that allows you to perform most functions using a graphical user interface. The Heads-Up Display serves as a visual aid if required. Zed Attack Proxy and Burp Suite both provide proxy functionality that intercepts web traffic. This functionality allows for inspection and manipulation of said traffic. Zed Attack Proxy offers automated scanning of vulnerabilities like Cross Site Scripting (XSS), and Structured Query Language Injection (SQL injection), to name a few. Burp Suite, on the other hand, features automated scanning with more comprehensive detection in the Professional version. Zed Attack Proxy provides reporting capabilities for results obtained from scans. Reports come as High Level, Modern HTML, Risk, and Confidence, among others. All Zed Attack Proxy reports show Common Vulnerabilities and Exploits (CVE) by category. The Risk and Confidence report extends this functionality by adding confidence percentages

of possible discovered vulnerabilities. The report also adds the full uniform resource locator (URL) of the detected vulnerability. Contrary to Zed Attack Proxy, Burp Suite Community does not provide a reporting feature. The reporting feature of Burp Suite is part of the professional and enterprise editions of the tool. Both Zed Attack Proxy and Burp Suite provide the option of extending the application's functionality using plugins or addons. Zed Attack Proxy features a marketplace where addons can be downloaded free of charge. For All the editions of Burp Suite there is the Burp Application Store (BApp Store). The Burp Application Store allows for the installation of extensions, although some require a paid-for-licensed version of the software. An extension to automate vulnerability scans, for example, is only available for the professional and enterprise versions of the tool. In terms of application support, Zed Attack Proxy has a community of developers from around the world dedicated to maintaining the tool and the addons found in its marketplace. The Open Web Application Security Project team allows for third parties to engage in offering support even at an enterprise level if the third party aligns with the teams and open-source licenses terms. Burp Suite's support is provided by PortSwigger, the company that develops and maintains the tool. Zed Attack Proxy allows for manipulation of web requests and responses and the use of breakpoints while code is being processed. Zed Attack Proxy features the Resend Request Editor that allows for repeatedly sending requests to a target. A technique known as Fuzzing, which entails the submission of a great amount of data to a target is also part of Zed Attack Proxy. Similar tools are available in the community edition of Burp Suite. Like Zed Attack Proxy, Burp Suite provides a Hypertext Transfer Protocol and WebSocket proxy and history log. Burp Suite Community also provides a command repeater, a traffic decoder, and a sequencer that allows to scan token randomness, and a comparer to analyze differences in data traveling through by the proxy. The free tool also includes a feature named Burp Intruder, a tool to automate customized attacks

against web applications, but only in a demo version in the Community Edition. Burp Suite offers a feature known as Collaborator which allows applications being evaluated to interact with an external server to discover blind or asynchronous vulnerabilities. One of the drawbacks of Burp Suite Community is that it does not allow projects to be saved. Any assessment performed using Burp Suite Community Edition must be started from scratch if the application is closed. If there is an error while executing Burp Suite, the web application for the assessment must be restarted. Both applications offer documentation to allow users to get started learning how to work with the platform. Burp Suite, on the other hand, offers the Web Security Academy [7], a platform to learn and practice vulnerability assessment and penetration testing, free of charge. Zed Attack Proxy provides many features that allow for an assessment of web applications using methods found in many paid competitors. The tool contains many features that make it a great alternative to similar paid-for software. The ability to perform active and passive scans, use spiders, and the addon marketplace, are assets that make the tool relevant.

TEST ENVIRONMENT

The test environment includes the VirtualBox Hypervisor from Oracle, and two virtual machines. One being Kali Linux (see Figure 1), the attacking machine hosting Zed Attack Proxy and Burp Suite.



```
(kali@TheLab) [~]
└─$ cat /etc/os-release
PRETTY_NAME="Kali GNU/Linux Rolling"
NAME="Kali GNU/Linux"
VERSION_ID="2023.3"
VERSION="2023.3"
VERSION_CODENAME=kali-rolling
ID=kali
ID_LIKE=debian
HOME_URL="https://www.kali.org/"
SUPPORT_URL="https://forums.kali.org/"
BUG_REPORT_URL="https://bugs.kali.org/"
ANSI_COLOR="1;31"
```

Figure 1
Kali Linux Attack Host

The second virtual machine is an Ubuntu Linux hosting vulnerable web applications and database. The Ubuntu Linux distribution is provided by the Vulnerable Pentesting Lab Environment [8] or VPLE.

The Virtual Pentesting Lab Environment VM hosts a Docker Engine instance (see Figure 2). The

docker containers host the vulnerable web applications including a MySQL database.

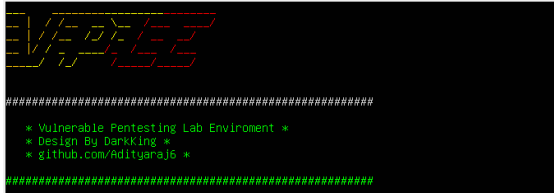


Figure 2
Virtual Pentesting Lab Environment

Among the tested web applications are Buggy Web App, OWASP Mutillidae II, DVWApp, and WebGoat. For comparison against real world scenarios, the following privately owned web applications, Facebook, Amazon, and Microsoft were scanned. Lastly, the public page scanme.nmap.org was scanned to assess the tool's functions against a vulnerable web application not hosted under the control of our private lab environment.

ASSESSMENT METHOD

To evaluate the capabilities of the assessment tools, scans of open-source vulnerable web applications were performed. These vulnerable web applications are purposely designed to present flaws like SQL Injection, Broken Authentication, Cross Site Scripting (XSS) among many others. The following resources were used as reference of the assessments and feature validation of Zed Attack Proxy: Portswigger Web Security Academy, Zed Attack Proxy in Ten, Open Web Application Security Project Web Security Testing Guide 4.2 [9].

The automated scan is immediately present in the main Zed Attack Proxy window, see Figure 3.

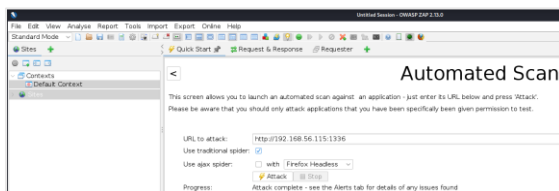


Figure 3
Zed Attack Proxy Automated Scan

Zed Attack Proxy will follow links and traverse folders during scans, as seen in Figure 4. In the test environment the number of links followed and

traversed folders have been limited to conserve the virtual machine's memory and scan time.

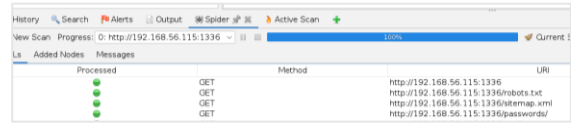


Figure 4
Zed Attack Proxy Spider

The assessment begins with a spider crawling through the target assessing the web application's logical structure. Afterward, the assessment tool performs an active scan on the web application. The process, as seen in Figure 5, will look for the most common web application vulnerabilities listed in the Open Web Application Security Top 10.

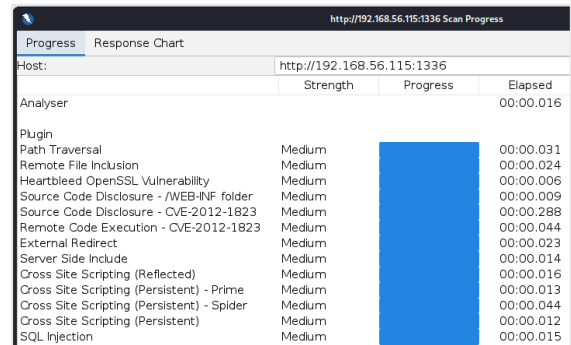


Figure 5
Zed Attack Proxy Scan Progress

The results of the scan will produce a list of vulnerabilities in a tab named Alerts. The alerts will contain details of the identified vulnerabilities found on the targeted web application (see Figure 6).

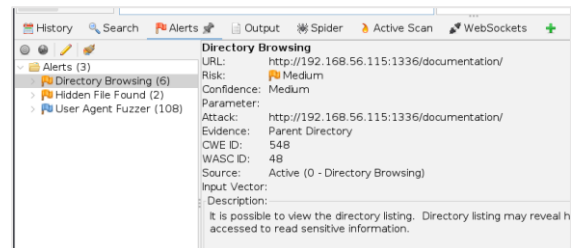


Figure 6
Zed Attack Proxy Alerts

Selecting any of the Alerts will display the location of the vulnerability, the risk, the confidence of the diagnostic, and other useful information. The diagnostic will also show how the vulnerability ties

with a category of the Open Web Application Security top 10 categories (see Figure 7).

Alert Tags:	
Key	Value
OWASP_2021_A01	https://owasp.org/Top10/A01_2021-Broken_Access_Control/
OWASP_2017_A05	https://owasp.org/www-project-top-ten/2017/A5_2017-Broken_Access_Control.html

Figure 7
Zed Attack Proxy Alert Reference

Attacking the Mutillidae II web application reported six instances of broken access control, two instances of security misconfiguration and one hundred and eight user agent fuzzer warnings. Attacking the Buggy Web App reveals the app has a vulnerability of an exposed hidden file. The hidden file matches three vulnerabilities from the Open Web Application Security Project’s top ten. One of the features that the Buggy Web Application provides is a menu to activate several types of vulnerabilities. Using these options, the application was configured to expose the phpinfo.php server settings file. In real-world scenarios, this represents a serious security misconfiguration. The reason is that this file contains sensitive information of the php parameters set in a server and other important server details. It should not be exposed easily, and it should only be visible from within the server itself. Burp Suite Professional offers four scan options: Lightweight, Fast, Balanced, and Deep. The dashboard (see Figure 8) does an excellent job displaying the target, the discovered resources, the vulnerabilities encountered, and the advisories describing the issues found.



Figure 8
Burp Suite Scan Dashboard

Lightweight scans performed showed expected results from the vulnerable web apps targeted in the test laboratory environment. The results provide advisories and vulnerability classification references to the popular common vulnerabilities and exploits database mitre.org (see Figure 9).

The tool also presents references to PortSwigger’s web security academy with detailed explanations of the issues discovered. The advisories

though, do not reference the Open Web Application Security Project’s Top 10 (Figure 10). The reports generated by Burp Suite Professional are very detailed and well laid out. The reports can be configured to include the issue, issue details, remediation, remediation details, and references if desired.

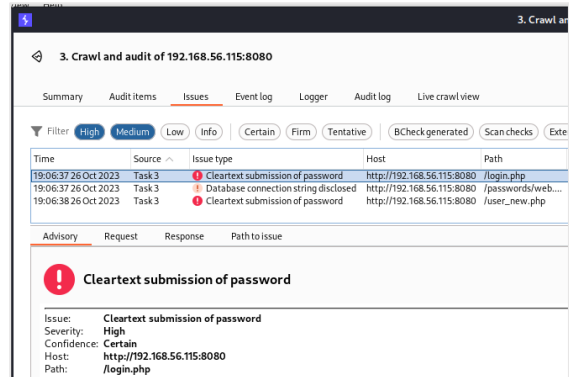


Figure 9
Burp Suite Scan Crawl and audit of target

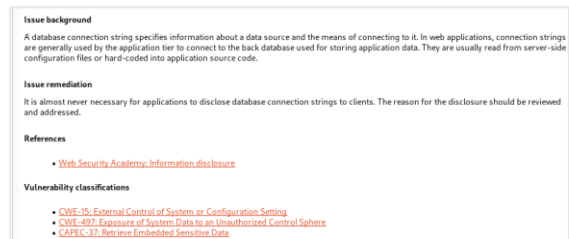


Figure 10
Burp Suite Vulnerability Classification

SCAN FINDINGS

The following are vulnerability lists that consist of vulnerabilities found by both tools, and vulnerabilities discovered by either Zed Attack Proxy or Burp Suite Pro but not both.

Shared Findings

The following are findings that were discovered by both tools. The scan of the Buggy Web App showed results for vulnerabilities CWE-319 and CAPEC-117. These vulnerabilities refer to web forms that accept submission of credentials without using a secure medium like SSL or TLS. This result is accurate with the configuration of the Buggy Web App having a login.php that does not enforce secure communication with clients. A cross site scripting

CWE-1021 was found by both tools for the OWASP Mutillidae II, Facebook, and NmapWeb web applications. The flaw would be of concern for the OWASP Mutillidae II website if the application were used in a real production environment. The reason being that the flaw was discovered on the website's index page. The vulnerability, for Facebook and NmapWeb, on the contrary, is of little or no real concern. Cross site scripting vulnerabilities pose a bigger threat to areas where sensitive information is processed. This is not the case with Facebook ("facebook.com/settings/language/language/") and "scanme.nmap.org/," which manage no sensitive information in the areas flagged by the tools. The vulnerability encountered in Amazon, CWE-523 represents a threat depending on the information exchanged between the client and server. CWE-523 manifests when an application does not enforce the use of secure traffic in the form of SSL or TLS between a client and a server. An attacker could be able to access or modify traffic between the two nodes without the client and server being aware, a maneuver known as a man-in-the-middle attack. A list of the results can be seen in Table 1.

Table 1
Vulnerabilities Found By Both
Burp Suite And Zed Attack Proxy

Site	Vulnerability	Category
bWApp	Clear Text Password Submission	CWE-319 CAPEC-117
DVWApp	None	
OWASP Mutillidae II	Cross-Site Scripting	CWE-1021
WebGoat	None	
Amazon	HTTP Strict Transport Security (HSTS)	CWE-523
Facebook	Cross-Site Scripting	CWE-1021
NmapWeb	Missing Anti-clickjacking Header	CWE-693 CWE-1021 CAPEC-103
Microsoft	None	

Zed Attack Proxy Findings

The Zed Attack Proxy scan of Buggy Web App showed vulnerabilities related to the exposure of the file `phpinfo.php`, OWASP_2021_05, WSTG-v42-CONF-05, and OWASP_2017_A06 (see Table 2). The exposed file contains server configuration information that could be used to undertake

unintended privileges and perform undesired activities in the web application. For the OWASP Mutillidae II scan revealed CVE-2012-1823, Directory Browsing and SQL Injection vulnerabilities. CVE-2012-1823 allows execution of commands by placing them in a query string through a php page. A CWE-16 or CAPEC-31 takes place when a cookie has been set without the `HttpOnly` flag, which means that JavaScript can access the cookie. If a malicious script can be run on the page, or on the client side, then the cookie will be accessible and can be transmitted to another site. If the cookie is controlling a session, then hijacking the session may be possible. The NmapWeb issue CWE-200 controls whether a server response header field sent back to clients includes a description of the generic OS-type of the server as well as information about the compiled-in modules. Although the Apache documentation considers "security through obscurity" a "myth", revealing the version of the software is considered a vulnerability. The vulnerable JavaScript library found on Microsoft's website could allow the execution of untrusted code from an untrusted source using jQuery's DOM manipulation methods like `html()`, `append()`, and others. The vulnerability would require the web application to use `html <>option` tags to execute requests to an untrusted source to be exposed to the vulnerability.

Table 2
Vulnerabilities Found By Zed Attack Proxy
But Not Found By Burp Suite

Site	Vulnerability	Category
bWApp	Hidden Sensitive File Found <code>phpinfo.php</code>	OWASP_2021_A05 WSTG-v42-CONF-05 OWASP_2017_A06
DVWApp	None	
OWASP Mutillidae II	Directory Browsing, SQL Injection, Source Code Disclosure	CVE-2012-1823
WebGoat	None	
Amazon	None	
Facebook	Cookie No <code>HttpOnly</code> Flag	CWE-16 CAPEC-31
NmapWeb	Server Leaks Version Information via "Server" HTTP Response Header Field	CWE-200
Microsoft	Vulnerable JS Library	CVE-2020-11023 CVE-2020-11022 CVE-2019-11358

Burp Suite Findings

For the case of Buggy Web App, Burp Suite discovered a Cookie No HttpOnly flag for two web pages inside the application, portal.php and sql_i_1.php. The DVWApp and WebGoat web applications showed CWE-319 and CAPEC-117, a result of having a form that accepts passwords in cleartext, which makes them vulnerable to interception. Burp Suite encountered an XPath vulnerability on the OWASP Mutillidae II web application covered by CWE-94, CWE-116, CWE-159, CWE-643, and CAPEC-83. The injection vulnerability arises when user-controllable data is incorporated into XPath queries in an unsafe manner. An attacker can supply crafted input to break out of the data context in which their input appears and interfere with the structure of the surrounding query. A TLS certificate informational event was issued for Amazon and Microsoft based on CWE-295, CWE-326, and CWE-327. The warning is issued when a security certificate cannot be validated. The issue might also manifest when an inadequate encryption strength or a cryptographic algorithm that is broken is employed by the certificate. Most browsers have a facility to remember user credentials that are entered into HTML forms. This function can be configured by the user and by applications that employ user credentials, such as in the case of Facebook. The vulnerability CWE-200 discovered shows if such a function is enabled. The NmapWeb application's landing page allows users to connect to it over unencrypted connections, as reported by CWE-326, see Table 3 for Burp Suite's scan results.

WHY DIFFERENT RESULTS?

Although both tools have identified issues in unison for some web applications as shown in Table 1, Tables 2 and 3 set them apart. Thus, raising the question of why the tools would find different issues in their assessments. Zed Attack Proxy's default settings perform a passive scan with unlimited duration in minutes using a spider that crawls unlimited child nodes (or subdirectories and pages).

TABLE 3
VULNERABILITIES FOUND BY BURP SUITE BUT NOT FOUND BY ZED ATTACK PROXY

Site	Vulnerability	Category
bWApp	Cookie No HttpOnly Flag	CWE-16
		CAPEC-31
DVWApp	Clear Text Password Submission	CWE-319
		CAPEC-117
OWASP Mutillidae II	XPath injection	CWE-94
		CWE-116
		CWE-159
		CWE-643
		CAPEC-83
WebGoat	Clear Text Password Submission	CWE-319
		CAPEC-117
Amazon	TLS certificate	CWE-295
		CWE-326
		CWE-327
Facebook	Password field with autocomplete enabled	CWE-200
NmapWeb	Unencrypted communications	CWE-326
Microsoft	TLS certificate	CWE-295
		CWE-326
		CWE-327

This kind of scan allows for more breadth of coverage but increases the number of false positives encountered. Burp Suite Pro default settings offer four scans under the audit and crawl category. The scans aim to provide feedback under 15 minutes for the lightweight, 60 for fast, and 60+ minutes for balanced. There is no time specification for the deep scan category. Each tool sources its own database for vulnerability assessments. Zed Attack Proxy references the Open Web Application Security Project classifications and the Common Weakness Enumeration of mitre.org. Burp Suite uses the Common Attack Pattern Enumerations and Classifications, and, like Zed Attack Proxy, the Common Weakness Enumeration database offered mitre.org to reference issues encountered by its assessments. The Zed Attack Proxy passive scan is similar in aggressiveness to Burp Suite's default scans. Zed Attack Proxy, in Attack Scan mode, is aggressive in its assessments, something that, though configurable by users, is avoided by default in Burp Suite. The results discovered by both tools may differ too depending on the promptness that their assessment tools definitions are updated with the newer vulnerabilities discovered by security researchers. In their default configuration, the tools

will perform scans that are not as intrusive, but when configured in Attack mode, Zed Attack Proxy becomes exponentially more aggressive than Burp Suite. And Burp Suite must be manually configured or scripted to reach this level of aggressiveness.

Which One is Better?

Zed Attack Proxy is free of charge, which makes it affordable for anyone needing to use the tool for learning or professional purposes. The tool provides a wealth of customizable features needed to complete a security assessment from scan to report delivery. This is reason enough to place Zed Attack Proxy in a strong contender position when considering a web application security assessment tool. Both tools have a steep learning curve and require at least foundational cybersecurity knowledge from users. There is also the topic of Automation of tasks using containers, something that can be achieved with Zed Attack Proxy but would require an Enterprise Edition license for Burp Suite. The case of Zed Attack Proxy having more false positives because of its more breadth scans methods could be raised. But it is the task of security researchers to analyze the results obtained from assessments and check their validity. One example of this requirement is the case of Burp Suite and TLS certificates issues. Burp Suite uses a Java trust store to determine whether certificates should be trusted. The issue details in scan results warn that the Java trust store does not include every Root Certification Authority normally found in web browser trust stores. This could lead Burp Suite to report an issue with a TLS certificate because it does not have the means to evaluate its legitimacy. These kinds of results require that security researchers focus not only on what are the expectations of the tool being used but also on how precise the results of assessments are.

Should Both Tools Be Used?

Many security systems, like unified threat management firewalls for example, use multiple antivirus, antispam, or intrusion detection engines from various vendors for real-time detection and protection. The case should not be different for pentesting tools used for web application security assessments. Security researchers, the people responsible for detecting and addressing cyber threats work for public or private organizations.

This, unfortunately, creates a time gap in the awareness of cyber threats. Unless this condition changes, there will always be vulnerabilities discovered by researchers that will require updating and patching of assessment tools. Zed Attack Proxy will depend on the open-source community and the Open Web Application Security Project to maintain the tool accurate and up to date. On the other hand, Burp Suite will depend on PortSwigger's capacity as a business to keep the tool in a current, relevant state. Both tools should be used for cases where there are few human resources or there is a lack of high-level skillset and cybersecurity knowledge. Researchers could use either tool with the condition that they can guarantee the results obtained using their knowledge and experience. Going back to the TLS certificate issue identified by Burp Suite. Although the tool explains in the issue details the possibility of the issue being a false positive, it is the task of the researcher to validate its truthfulness.

SEVERITY OF ZED ATTACK PROXY MISSED ISSUES

The CWE-16, Cookie No HttpOnly Flag found on the Buggy Web App is a serious threat for web applications that store sensitive information in client cookies. The vulnerability exposes the data in cookies to be accessible through JavaScript and exposes it to be acquired by third parties running code on web browsers. CWE-319 and CAPEC-117 pose the same risk for sensitive data accepted as input in fields that accept passwords in clear text, as it was the case with DVWApp and the WebGoat web applications. The XPath injection missed for the OWASP Mutillidae II site is highly severe. By sending intentionally malformed information into the targeted web site, an attacker could query an XML data structure, or access data not intended to be visible by unprivileged entities. An attacker may even be able to elevate their privileges on a web site if the XML data is being used for authentication for example. As of the time of writing, there has been no evidence that Zed Attack Proxy performs any analysis on the validity of TLS certificates in use by the web applications being assessed. The documentation related to TLS and HTTPS is no longer available on Zed Attack Proxy's website or

the tool's help documents. This is an important feature that is available in Burp Suite Professional. And even though Burp Suite warns of the possibility that a TLS certificate warning could be a false positive (which could be the case of the results for Amazon and Microsoft), tests against the website badssl.com, specifically the expired.badssl.com resulted in an accurate certificate diagnosis by Burp Suite. Facebook's undetected CWE-200 vulnerability exposes a field containing sensitive information. Because of this, developers of web browsers created features to prevent the issue from becoming a real threat. These password fields management developments were also followed by the adoption of password management features by modern web browsers. Discussions from around a decade ago suggest that Zed Attack Proxy might have turned these kinds of detections off based on the consideration of it being a false positive. As mentioned, Zed Attack Proxy does not analyze secure traffic, therefore it will not warn of sites that do not use secure traffic unless it detects specific scenarios like the case of CWE-523.

SEVERITY OF BURP SUITE MISSED ISSUES

Burp Suite did not report encountering the phpinfo.php file exposed by Buggy Web App. This file contains valuable information that is useful for researchers during the reconnaissance phase of an assessment. For the OWASP Mutillidae II, CVE-2012-1823 was not detected. The CVE includes risks that could result in directory browsing, SQL injections, and source code disclosure. The vulnerability has a considerable attack surface and cannot be treated lightly. For Facebook, Burp Suite did not show an issue with the Cookie No HTTPOnly Flag CWE-16 and CAPEC-31. Scanning the NmapWeb missed the identification of CWE-200, an exposure of web server software version. The JavaScript Library vulnerability CVE-2020-11023, CVE-2020-11022, and CVE-2019-11358 found on Microsoft's website were also not identified by Burp Suite during the assessment.

CONCLUSION

The analysis shows that Zed Attack Proxy is very well suited as an everyday tool for novice to advanced security practitioners. The tool bundles all the functionality and reporting needed out of the box allowing for a thorough assessment of a web application's security scorecard. This makes the tool a perfect asset for cybersecurity students and professionals starting out their career as well. Another reason that would make Zed Attack Proxy an excellent choice is that it comes at no cost for the user. The support from the open-source developers community provides some guarantee that the tool will be continuously updated. This also extends to the Zed Attack Proxy's marketplace resulting in continued development of new and innovative addons. Although not at a beginner's level, Zed Attack Proxy's user interface is very intuitive, at first hand more than Burp Suite's. Zed Attack Proxy's interface offers immediate access for specifying a target to begin performing attacks and work can be saved if there is a need to suspend it. The results from the attacks are all available in a single view pane. The Heads-Up Display is a feature that is beneficial to those in need of a visual aid, like students. And the Automation Framework brings powerful scripting capabilities to Zed Attack Proxy. Researchers should be aware of the limitations of Zed Attack Proxy, like not having tools to diagnose TLS certificates and lacking direct support under a binding licensing contract. Burp Suite is a complex, feature rich tool that offers robust functionality. Its Community Edition, although robust, lacks automated scanning capabilities and does not allow for reporting. Burp Suite may enjoy more popularity among seasoned cybersecurity professionals for its advanced tools and granularity, but it comes at a cost. Two of the most useful tools required by every security practitioner, an automated scanner and reporting capabilities, are only available on Burp Suite Professional and Enterprise, which require an annual paid subscription from the user. Burp's user interface requires understanding of the tool's view of how a security assessment of a web application

should be performed. The assessment methodology is split into modules. These modules require understanding of how each one works, and how they interact with each other. Burp requires the knowledge of moving a session, data, or results between modules to execute functions. Although this would be beneficial for advanced users, it increases the learning curve required to use the application. Performing an attack with the community edition of Burp Suite requires a user to know exactly what needs to be configured in the application beforehand. Unlike Zed Attack Proxy, there is no automated option to assess a web application for vulnerabilities unless the professional or enterprise applications are purchased. The Professional version of Burp Suite allows access to fast scans, reporting and more advanced tools that provide an advantage to users. Burp Suite is a particularly useful assessment tool for seasoned security professionals looking for a feature rich suite. Additionally, a notable feature needed from an assessment tool lies in its automation capabilities, on how quickly and efficiently it can produce tangible results, especially in present days where continuous delivery and integration are components in software tools. This is an advantage that Zed Attack Proxy provides over the community and professional editions of Burp Suite. Despite its drawbacks (No TLS certificate validation, duration of scans using defaults) Zed Attack Proxy proved to be an effective, intuitive, and easy to use tool. The reporting options of Zed Attack proxy can produce results tailored to technical or management tiers. At the time of writing, two distinctive features missing from Burp Suite, a full featured free version, and the Heads-Up Display (HUD). There are also the Automation features Zed Attack Proxy offers as well that make it stand out. Automation is available in Burp Suite, but it is advertised as a feature for enterprise versions of the tool. Something of significant importance is the support of the application, which is available in Zed Attack Proxy and the Community Edition of Burp Suite in the form of online user forums. This kind of support does not offer a guarantee to users on the expertise of the help received, if received at all. If support for

Zed Attack Proxy is required, it can be obtained from third parties. Support for Burp Suite is available in the professional and enterprise versions directly from Portswigger. As an alternative to a subscription-based tool Zed Attack Proxy is a great option. It could also be considered as a temporary solution for students or newcomer web security practitioners looking to develop skills without spending money on tools that might not generate income. Zed Attack Proxy could also be used as a backup tool by security researchers or when automation is required, and an enterprise license is not available. Security researchers could use Zed Attack Proxy and Burp Suite Professional in combination to perform assessments with more thorough coverage and validate results.

REFERENCES

- [1] OWASP, "OWASP foundation, the open source foundation for application security," owasp.org, Dec. 01, 2001. <https://owasp.org/>
- [2] OWASP, "OWASP ZAP," [Zaproxy.org](https://www.zaproxy.org/), 2020. <https://www.zaproxy.org/>
- [3] Open Web Application Security Project Foundation, "OWASP Top Ten Web Application Security Risks | OWASP," owasp.org, Dec. 01, 2001. <https://owasp.org/www-project-top-ten>
- [4] L. Sujay Vailshery, "Best Tools for Hacking 2020," *Statista*, Mar. 23, 2020. <https://www.statista.com/statistics/800916/worldwide-useful-software-hacking> (accessed May 20, 2023).
- [5] Open Web Application Security Project Foundation, "OWASP ZAP – Getting Started," www.zaproxy.org. <https://www.zaproxy.org/getting-started/>
- [6] Pentest-Tools.com. (2013). Pentest-Tools.com | Powerful Pentesting Tools, Easy to Use. Pentest-Tools.com. <https://pentest-tools.com>
- [7] PortSwigger, "Web Security Academy: Free Online Training from PortSwigger," portswigger.net. <https://portswigger.net/web-security>
- [8] Vulnerable Hub, "Vulnerable Pentesting Lab Environment: 1," *Vulnerable Hub*, Aug. 19, 2021. <https://www.vulnhub.com/entry/vulnerable-pentesting-lab-environment-1,737/>
- [9] Open Web Application Security Project Foundation, "WSTG - v4.2 | OWASP," owasp.org, Dec. 03, 2020. <https://owasp.org/www-project-web-security-testing-guide/v42/>