Web Application Security

Tejas Mhaske

Cyber Security Trainer

What is Web application Security

Web application security is the practice of protecting websites, web applications, and APIs from various cyber threats and attacks.

How does web security work

1. **User Interaction**: Users interact with the web application through a web browser.
2. **Request Handling**: The user’s request is sent to the web server.
3. **Firewall Filtering**: The firewall filters the request to block any malicious traffic
4. **Authentication and Authorization**: The application checks the user’s credentials and permissions.
5. **Data Processing**: The application processes the request, interacting with the database if necessary.
6. **Response Handling**: The processed data is sent back to the user through the web server.
7. **Encryption**: Data is encrypted during transmission to ensure security.

Web Application Lifecycle Management

- Application Lifecycle Management (ALM) is the creation and maintenance of a software application until it is no longer used. It involves multiple processes, tools, and people working together to manage every aspect of the life cycle, such as ideas, design and development, testing, production, support, and eventual redundancy.
- ALM is also known as integrated application lifecycle management because various software experts, like developers, analysts, testers, and change managers, work together throughout the application life cycle. Collaboration among teams and the use of various supporting tools ensure that application development meets business goals and that the project succeeds.

Why is ALM important?

1. **Requirement Setting:** ALM helps companies set and meet appropriate requirements for projects.

2. **Process Improvement:** ALM improves the development process by incorporating frequent, thorough testing.

3. **Flexibility:** It assists developers in adjusting development processes and goals throughout the software lifecycle.

4. **Team Collaboration:** ALM ensures effective collaboration among all teams, including development, operations, and security, to produce the best possible software.

5. **Frequent Updates:** Leading software companies deploy updates daily, supported by ALM.

6. **Efficiency and Competitive Edge:** ALM helps businesses achieve high efficiency and gain a competitive edge by accelerating workflows and ensuring the deployment of top-quality products.

Stages of ALM

- Application lifecycle management consists of five stages:

1. Defining requirements
2. Development of the product
3. Testing and quality assurance
4. Deployment
5. Continuous maintenance and improvement of the product

Defining requirements

- **Application requirements gathering**
- In the initial stage, relevant stakeholders define what they require from the application. They analyze how the application will support them to meet their business goals and regulatory compliance requirements. Requirements management typically involves writing user stories that show how different users will interact with the application.
- **Application requirements gathering example**
- A bank is looking at building a web banking application. It defines two users: customers and administrators. The requirement management team identifies two user stories:
- A customer user story that states that customers use the application to submit a request to open a new bank account.
- An administrator user story that states that administrators use the application to approve customer documents.
- Also, the requirement management team identifies that the application's software system should comply with security standards that meet data privacy laws.

Application development

- In the development phase, various teams work together to convert the requirements into a working application. These steps are an example:
- **Project managers** estimate the time and development cost.
- **Developers** identify the design tasks and programming activities.
- **Quality analysts** add review tasks and checkpoints for quality and progress checks.

Application development example

- The bank’s IT team makes a development plan for the web application. The team members identify that they need to complete the customer’s user story first, then test it thoroughly before starting on the administrator’s requirements. However, they know they have to complete both requirements before launching the new product. They code the application and release it to a beta group in two months.

Application testing

- The bank’s quality assurance team verifies the business case of opening accounts for its web banking application.
- They find that a customer can select only a driver's license as ID proof.
- Since the bank also accepts passports as ID proof, they ask the development team to update the application to include this information.

Application deployment

- During deployment, the developers release the application to end users. Release management also includes planning how the team deploys software changes over time.
- **Example:-**
- The bank’s web application team uses a cloud server to host the application code so that administrators can access it from a website.

Application maintenance

- In the maintenance phase, support and development teams work together to resolve remaining bugs, plan new updates, and improve the product further.
- They incorporate user feedback and release new features that are relevant to customers.

Web Application Lifecycle Maintenance

1. **Continuous Monitoring and Performance Optimization**

- **Application Performance Monitoring (APM):** Implement APM tools to track response times, throughput, error rates, and other critical performance metrics in real-time. Ensure the application is optimized for both server and client-side performance.
- **Log Management and Analysis:** Utilize centralized logging solutions (e.g., Splunk) to aggregate and analyze logs from various sources. This helps in the early detection of anomalies and performance bottlenecks.
- **Load Balancing and Auto-scaling:** Use load balancers and auto-scaling groups to manage traffic spikes and maintain application availability.

2. **Security Patch Management**

- **Vulnerability Scanning:** Regularly scan the web application and its underlying infrastructure (servers, databases, etc.) for known vulnerabilities using tools like Nessus, or Qualys.
- **Automated Patch Deployment:** Implement automated patch management solutions to quickly apply security patches and updates to the application, libraries, and server OS.

Web Application Lifecycle Maintenance

3. **Security Monitoring and Incident Response**

- **Intrusion Detection and Prevention Systems (IDPS):** Deploy IDPS to monitor and block suspicious activities. Ensure these systems are updated with the latest threat signatures.

4. **Regular Penetration Testing and Security Audits**

- **Penetration Testing:** Conduct periodic penetration tests, both internally and externally, to identify and remediate security weaknesses before they can be exploited.
- **Code Reviews and Secure Coding Practices:** Enforce secure coding standards (e.g., OWASP Secure Coding Practices) and perform regular code reviews to catch potential security flaws during development.

5. **Backup and Disaster Recovery**

Automated Backups: Implement automated and encrypted backups for both application data and configurations. Ensure backups are stored in geographically diverse locations.

SSL/TLS

- **SSL (Secure Sockets Layer) and TLS (Transport Layer Security)**
- Cryptographic protocols designed to provide secure communication over a computer network.

How SSL/TLS Secure data?

- It encrypts the data between the client and the server
- Confidentiality
- Integrity
- Authentication

Some of the well-known Certificate Authorities include

- DigiCert
- GoDaddy
- Let's Encrypt
- GlobalSign

Importance of Web Application Security

1. **Protection Against Attacks:**

- **Concept**: Web applications are vulnerable to various types of cyber-attacks, including SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF).

2. **Data Privacy and Compliance:**

- **Concept**: Ensuring that sensitive data, such as personal information and financial details, is protected is crucial for maintaining user privacy and meeting regulatory requirements.
- **Technical Tools:** Data Encryption (using protocols like TLS/SSL for data in transit and AES for data at rest) helps safeguard data from unauthorized access.

Importance of Web Application Security

3. **Maintaining Integrity:**

•**Concept:** Web application security ensures that the data and functionality of the application remain unchanged and reliable, preventing unauthorized modifications.

•**Technical Tools:** Hashing algorithms (such as SHA-256) and digital signatures are used to verify data integrity and authenticity.

4. **Preventing Unauthorized Access:**

•**Concept:** Proper authentication and authorization mechanisms are essential to control access to the application and its resources.

•**Technical Tools:** OAuth and OpenID Connect are widely used for secure authentication and authorization. Multi-Factor Authentication **(**MFA**)** adds an extra layer of security.

Area	Web Security	Network Security
Definition	Web Security refers to the practices, technologies, and measures designed to protect web applications and the data they handle from various security threats	Network Security involves the strategies, tools, and practices used to protect the integrity, confidentiality, and availability of data as it is transmitted across or accessed through a network.
Focus	• Protects web applications from attacks that exploit vulnerabilities in the application itself. • Ensures that the web application functions correctly and securely for users	Protects the entire network infrastructure, including hardware and software components, from threats that could disrupt or damage network operations.

Key Areas	1. **Input Validation**: Ensuring that user input is validated to prevent attacks such as SQL injection or cross-site scripting (XSS). 2. **Authentication and Authorization**: Protecting user credentials and managing permissions. 3. **Session Management:** Securing user sessions to prevent session hijacking. 4. **Secure Communication**: Using HTTPS to encrypt data transmitted between the user and the server. 5. **Error Handling**: Managing error messages to avoid exposing sensitive information.	1. **Firewall**s: Filtering incoming and outgoing traffic based on predetermined security rules. 2. **Intrusion Detection/Prevention Systems (IDS/IPS)**: Monitoring network traffic for suspicious activity and responding to potential threats. 3. **Virtual Private Networks (VPNs):** Creating secure, encrypted connections over a less secure network. 4. **Access Control**: Ensuring only authorized users have access to network resources.

Common Threats	1. SQL Injection 2. Cross-Site Scripting(XSS) 3. Cross-Site Request Forgery (CSRF) 4. Session Hijacking	1. Distributed Denial of Service (DDoS) Attacks 2. Man-in-the-Middle (MitM) Attacks 3. Unauthorized Access

HTTP Request- Response

- **HTTP Request**
- An HTTP request is a message sent by a client to a server, asking for a resource or performing an action. It contains various components that instruct the server on how to respond.
- **Request Line:**

✓ Method: Specifies the action to be performed (e.g., GET, POST, PUT, DELETE).

✓ URL (Uniform Resource Locator): Specifies the resource to be accessed (e.g., /index.html).

✓ HTTP Version: Specifies the version of HTTP being used (e.g., HTTP/1.1).

✓ Example: GET /index.html HTTP/1.1

HTTP Request- Response

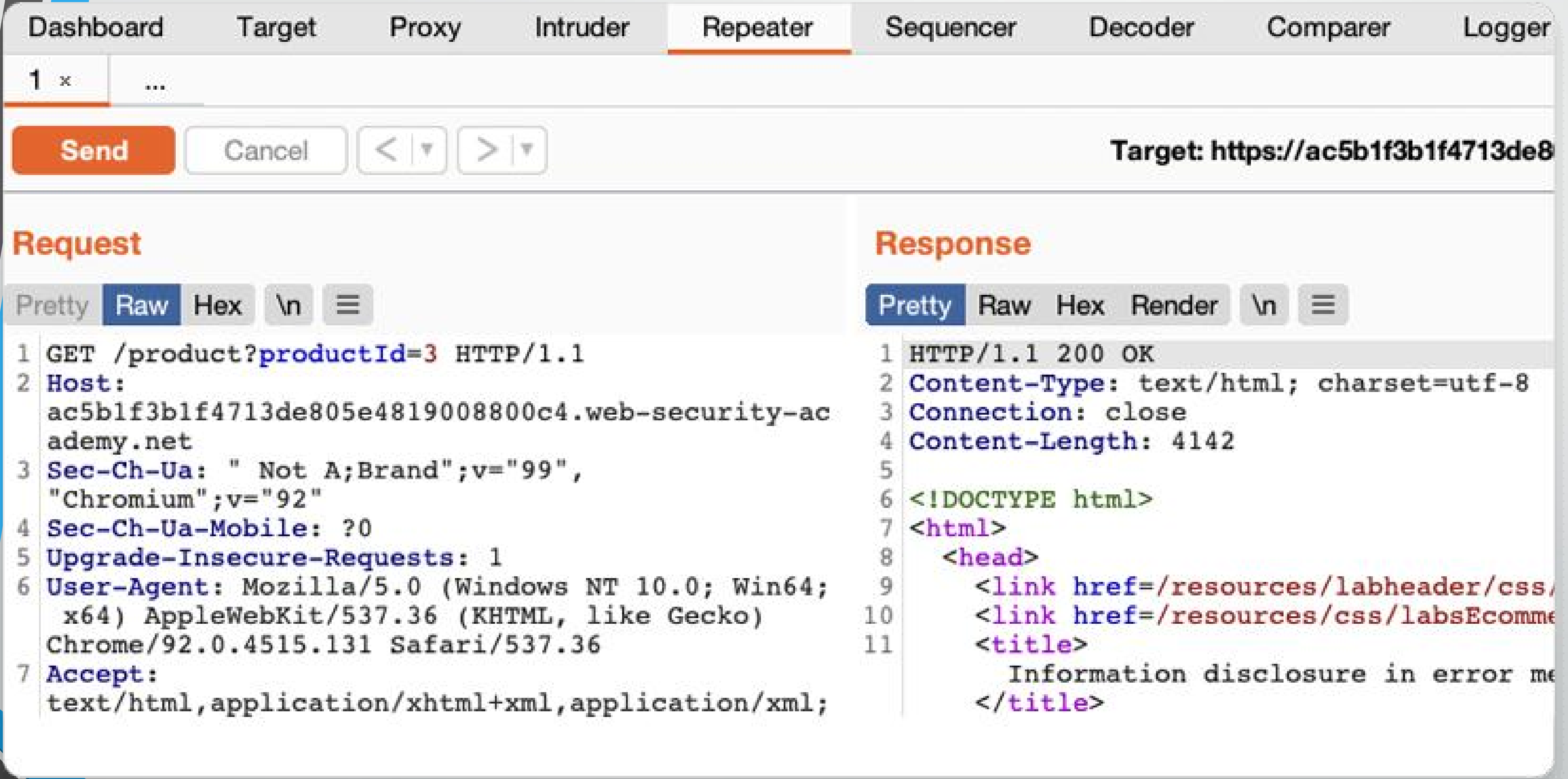
- **HTTP Response**
- An HTTP response is a message sent by a server to a client in reply to an HTTP request. It contains the data requested or information about the status of the request.
- **Status Line:**

✓ HTTP Version: Specifies the version of HTTP being used (e.g., HTTP/1.1).

✓ Status Code: Indicates the result of the request (e.g., 200, 404, 500).

✓ Status Message: A textual description of the status code.

✓ Example: HTTP/1.1 200 OK



```
GET /product?productId=3 HTTP/1.1
Host: ac5b1f3b1f4713de805e4819008800c4.web-security-academy.net
Sec-Ch-Ua: " Not A;Brand";v="99", "Chromium";v="92"
Sec-Ch-Ua-Mobile: ?0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.131 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;
```

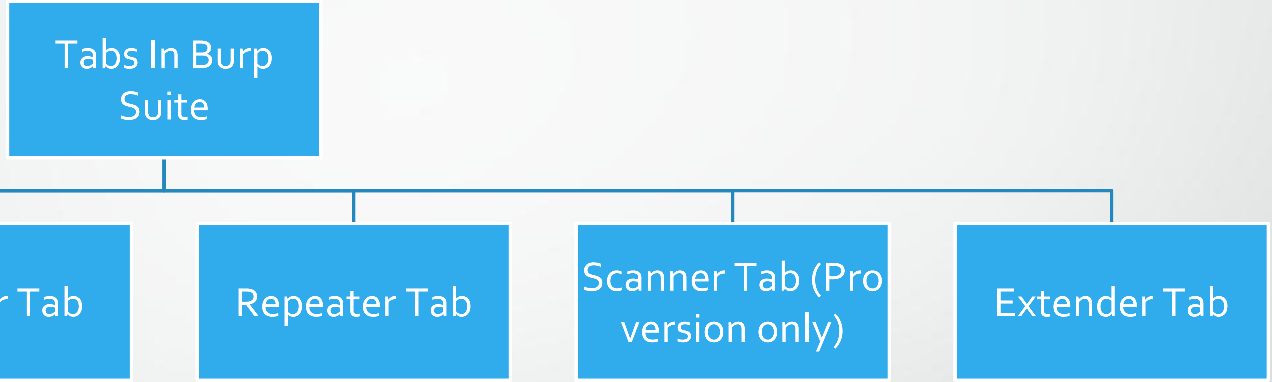
```
HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Connection: close
Content-Length: 4142

<!DOCTYPE html>
<html>
  <head>
    <link href=/resources/labheader/css/
    <link href=/resources/css/labsEcomme
    <title>
      Information disclosure in error me
    </title>
```

Status Codes:

- 200 OK: The request was successful, and the server is returning the requested resource.
- 301 Moved Permanently: The requested resource has been moved to a new URL.
- 400 Bad Request: The server could not understand the request due to invalid syntax.
- 401 Unauthorized: Authentication is required or has failed.
- 404 Not Found: The requested resource could not be found.
- 500 Internal Server Error: The server encountered an error and could not complete the request.

Burp Suite

- **Burp Suite** is a popular web vulnerability scanner and penetration testing tool developed by PortSwigger.
- It is widely used by cybersecurity professionals to test the security of web applications.
- Burp Suite provides a suite of tools for intercepting, analyzing, and modifying HTTP/HTTPS traffic between a browser and web servers.



Tabs In Burp Suite

Target Tab

Proxy Tab

Intruder Tab

Repeater Tab

Scanner Tab (Pro version only)

Extender Tab

1. Target Tab

- **Functionality:** The Target tab provides an overview of the entire target web application structure. It displays a sitemap, which lists all the directories and files discovered during a scan or manual browsing session.

2. Proxy Tab

- **Functionality:** This tab is central to Burp Suite. The Proxy intercepts the traffic between your browser and the web server, allowing you to view and modify requests and responses in real-time before they reach their destination.

3. Intruder Tab

- **Functionality:** The Intruder tab is used for automating customized attacks on web applications. You can set payloads (values) to test various inputs, such as brute-forcing login credentials or identifying vulnerabilities like SQL injection.

4. Repeater Tab

- **Functionality:** The Repeater allows you to manually modify and re-send individual HTTP requests to the server, enabling detailed analysis and exploitation of specific vulnerabilities.

5. Scanner Tab (Pro version only)

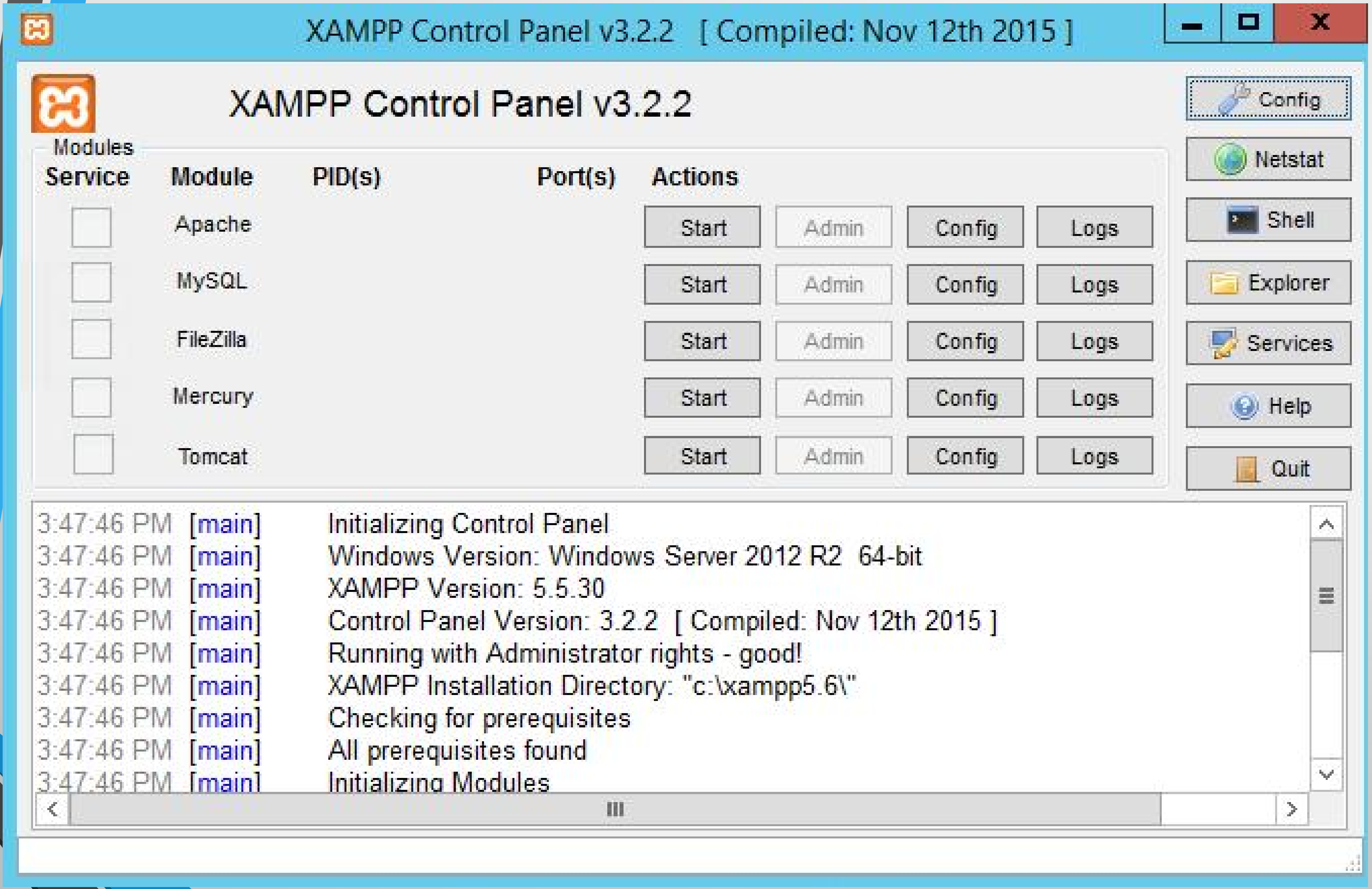
- **Functionality:** This tab provides automated scanning capabilities. It scans the target application for vulnerabilities like SQL injection, XSS, and other security flaws.

6. Extender Tab

- **Functionality:** The Extender allows you to enhance Burp Suite's functionality by adding custom extensions or integrating third-party plugins. It uses the Burp Extender API for this purpose.

XAMPP

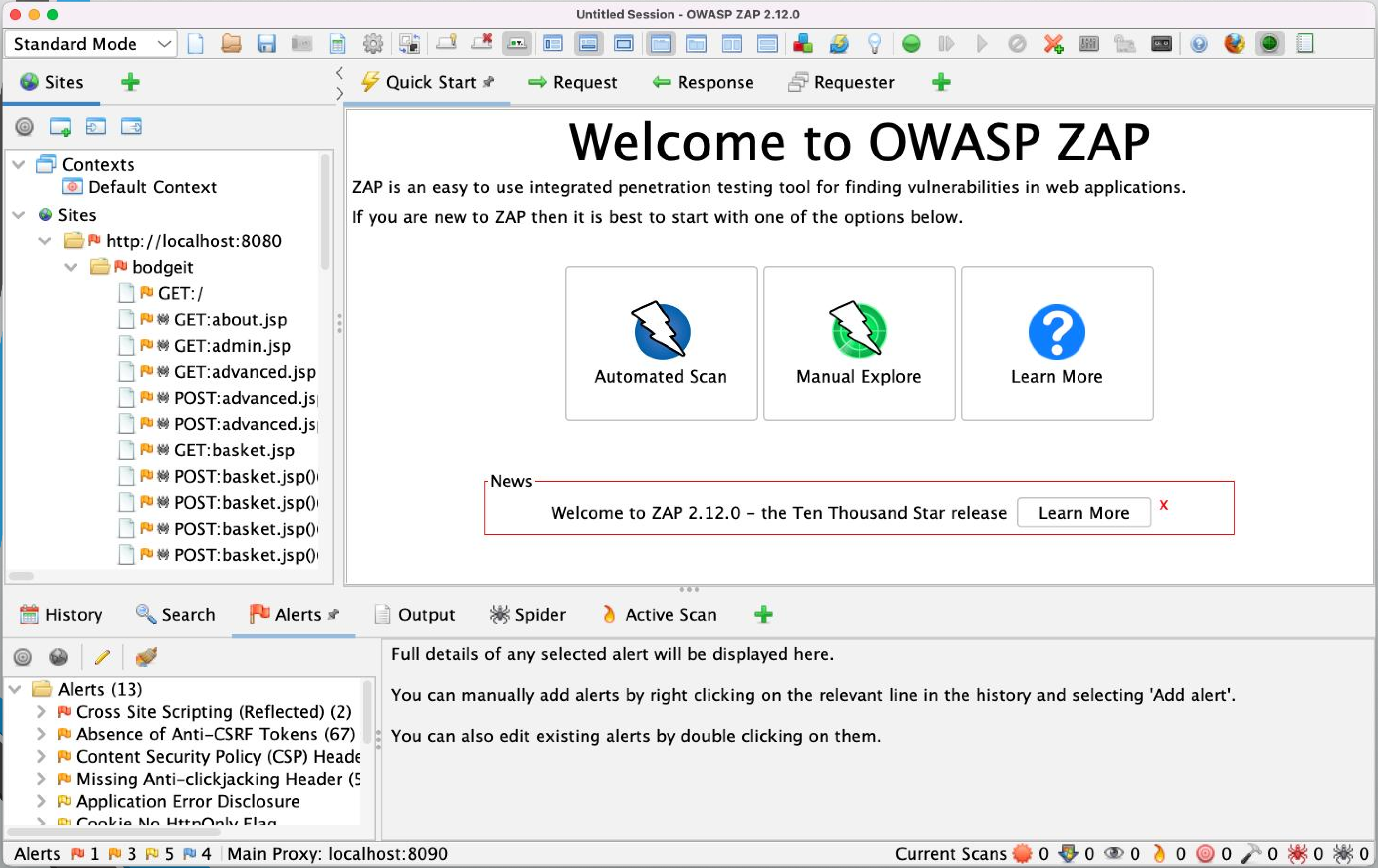
- XAMPP is an open-source, cross-platform web server solution package developed by Apache Friends.
- It stands for X (cross-platform), Apache, MySQL (now MariaDB), PHP, and Perl.
- XAMPP provides a simple and lightweight solution for developers to create and test web applications on their local machines without needing an internet connection or a dedicated server.



```
3:47:46 PM  [main]    Initializing Control Panel
3:47:46 PM  [main]    Windows Version: Windows Server 2012 R2  64-bit
3:47:46 PM  [main]    XAMPP Version: 5.5.30
3:47:46 PM  [main]    Control Panel Version: 3.2.2  [ Compiled: Nov 12th 2015 ]
3:47:46 PM  [main]    Running with Administrator rights - good!
3:47:46 PM  [main]    XAMPP Installation Directory: "c:\xampp5.6\"
3:47:46 PM  [main]    Checking for prerequisites
3:47:46 PM  [main]    All prerequisites found
3:47:46 PM  [main]    Initializing Modules
```

ZAP-Proxy

- ZAP-Proxy, or OWASP ZAP (Zed Attack Proxy), is an open-source web application security scanner developed by the **Open Web Application Security Project (OWASP)**.
- It is designed to help security professionals and developers identify and exploit vulnerabilities in web applications.
- ZAP-Proxy is one of the most popular tools for performing penetration testing and vulnerability assessments on web applications.



Welcome to OWASP ZAP

ZAP is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications.

If you are new to ZAP then it is best to start with one of the options below.

News

Welcome to ZAP 2.12.0 – the Ten Thousand Star release Learn More x

Full details of any selected alert will be displayed here.

You can manually add alerts by right clicking on the relevant line in the history and selecting 'Add alert'.

You can also edit existing alerts by double clicking on them.

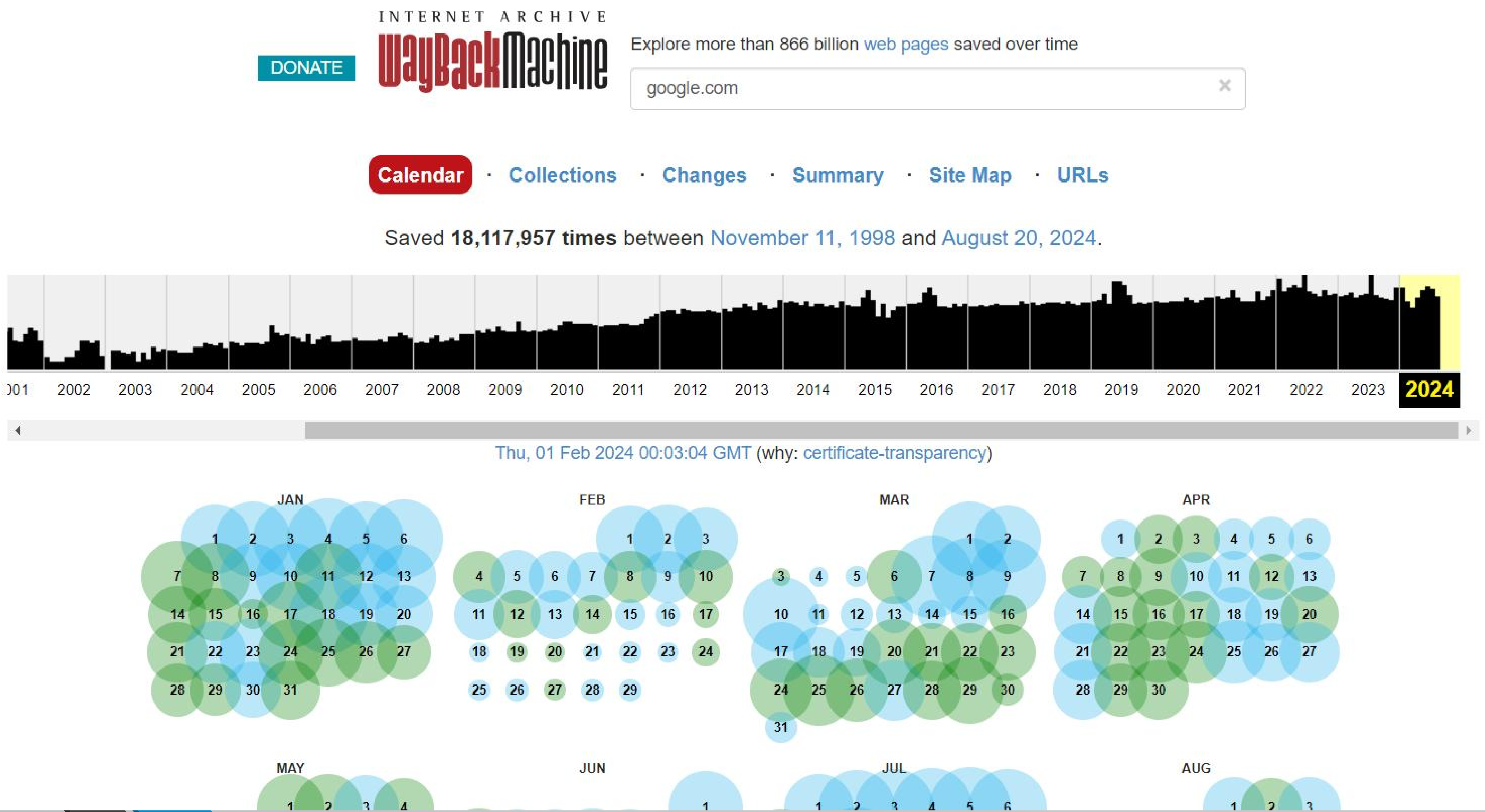
Nuclei

- Nuclei is an open-source vulnerability scanner developed by ProjectDiscovery that enables security researchers and penetration testers to perform automated security testing.
- It uses YAML-based templates to define and execute security checks against web applications, APIs, cloud infrastructures, and other network services.
- Nuclei is known for its flexibility, speed, and ability to scale, making it a popular tool in the cybersecurity community.

Use of GitHub

Wayback Machine

- The Wayback Machine is a digital archive of the internet, managed by the Internet Archive, a non-profit organization.
- Launched in 2001, it allows users to browse archived versions of web pages dating back to 1996.
- The Wayback Machine captures and stores snapshots of websites over time, preserving historical content even after it has been modified or deleted from the live web.



Explore more than 866 billion web pages saved over time

google.com ×

Calendar · Collections · Changes · Summary · Site Map · URLs

Saved **18,117,957 times** between November 11, 1998 and August 20, 2024.

Thu, 01 Feb 2024 00:03:04 GMT (why: certificate-transparency)

JAN

1 2 3 4 5 6
7 8 9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31

FEB

1 2 3
4 5 6 7 8 9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29

MAR

1 2
3 4 5 6 7 8 9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30
31

APR

1 2 3 4 5 6
7 8 9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30

MAY

1 2 3 4

JUN

1

JUL

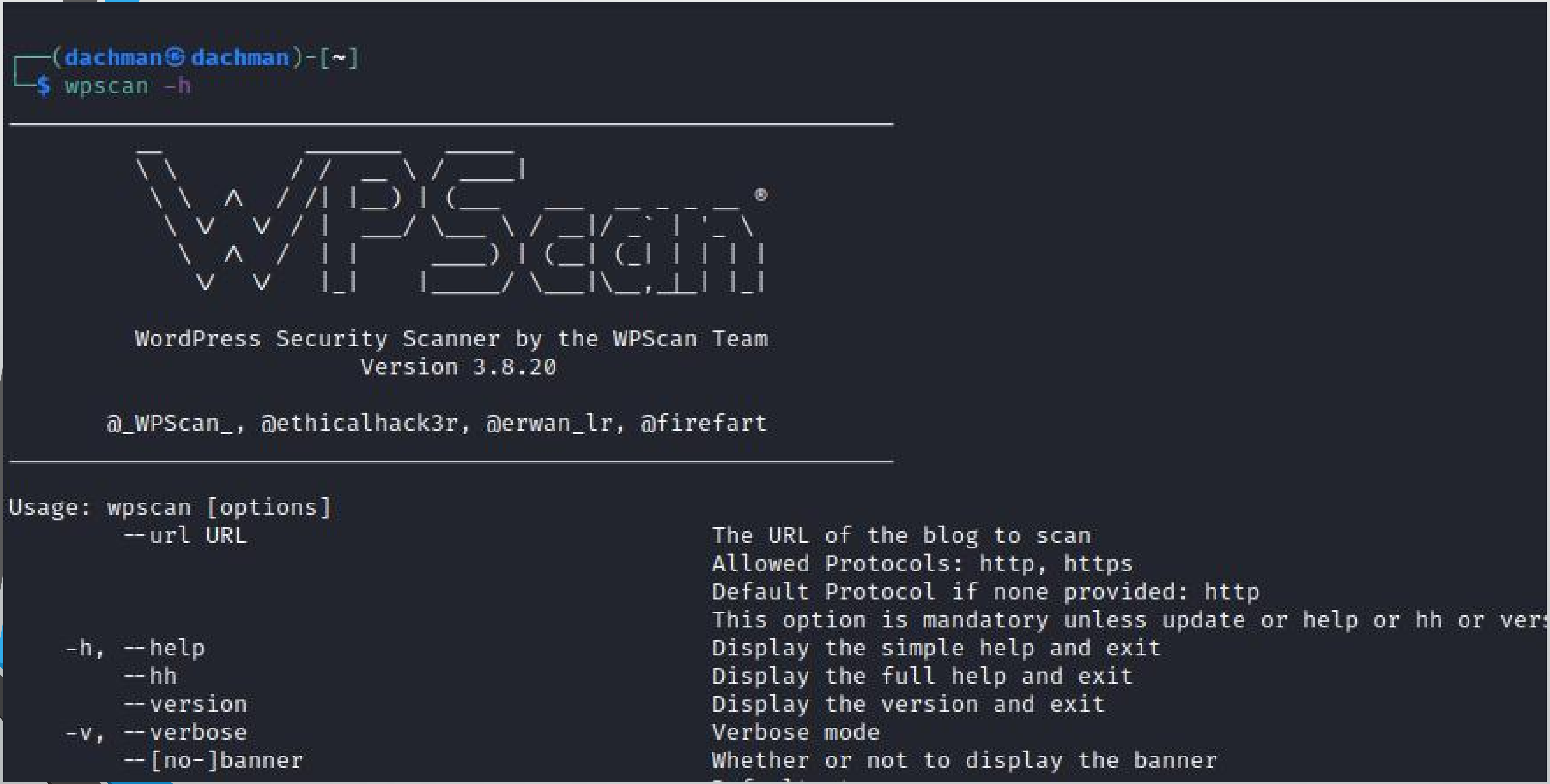
1 2 3 4 5 6

AUG

1 2 3

WPScan

- WPScan is an open-source WordPress vulnerability scanner designed specifically for identifying security issues within WordPress installations.
- Developed by the WPScan Team, it helps security professionals and site administrators detect vulnerabilities, misconfigurations, and security weaknesses in WordPress websites and their associated plugins and themes.



```
┌──(dachman㉿dachman)-[~]
└─$ wpscan -h
_______________________________________________________________
         __          _______   _____
         \ \        / /  __ \ / ____|
          \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
           \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
            \  /\  /  | |     ____) | (__| (_| | | | |
             \/  \/   |_|    |_____/ \___|\__,_|_| |_|

         WordPress Security Scanner by the WPScan Team
                         Version 3.8.20

       @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________

Usage: wpscan [options]
        --url URL                                 The URL of the blog to scan
                                                  Allowed Protocols: http, https
                                                  Default Protocol if none provided: http
                                                  This option is mandatory unless update or help or hh or vers
    -h, --help                                    Display the simple help and exit
        --hh                                      Display the full help and exit
        --version                                 Display the version and exit
    -v, --verbose                                 Verbose mode
        --[no-]banner                             Whether or not to display the banner
```

Wappalyzer

- Wappalyzer is a web application technology profiler that identifies and analyzes the technologies used on websites.
- It detects a wide range of technologies, including content management systems (CMS), e-commerce platforms, web frameworks, server software, and more.
- Wappalyzer is available as a browser extension, command-line tool, and online service.

NS-Lookup

- NS-Lookup (Name Server Lookup) is a network administration command-line tool used to query Domain Name System (DNS) servers for information about domain names.
- It allows users to obtain details such as IP addresses, mail servers, and other DNS records associated with a domain.
- NS-Lookup is a valuable tool for network troubleshooting, domain verification, and security assessments.

धन्यवाद