Sri Lanka Institute of Information Technology

BUG BOUNTY REPORT 03

(Wickr Web site)

IE2062 – Web Security

W.A.K.S Wijethunga

IT23361768

Table of Contents

1. Introduction to bug bounty program and audit scope

❖ Wickr

Wickr is a secure messaging platform that offers end-to-end encrypted communication for individuals, businesses, and government organizations. It was initially founded in 2012 and later acquired by Amazon Web Services (AWS) in 2021. Wickr provides both Wickr Me (for individuals) and Wickr Pro / Enterprise (for organizations), supporting secure messaging, file sharing, voice/video calls, and collaboration.

In Hackerone bug bounty program, they defined these subdomains (and all inclusive) as valid subdomains for testing.

- www.wickr.com
- **admin.wickr.com**
- **support.wickr.com**

Eligible in-scope subdomains for bug bounty program are mentioned below and they mention that any subdomain under **wickr.com** is in scope,

Asset name	Type	Coverage	Max. severity	Bounty	Last update	Resol Repo
www.wickr.com	Domain	In scope	Critical	Ineligible	Jan 24, 2023	1 (11
admin.wickr.com	Domain	In scope	Critical	Eligible	Jan 24, 2023	3 (33
support.wickr.com	Domain	Out of scope	None	Ineligible	Jan 24, 2023	0 (0%

1-14 of 14

2. Reconnaissance

The goal of this reconnaissance is to gather information about the **wickr.com** website, including its infrastructure, technologies, and potential security posture. This information will help identify potential vulnerabilities and attack vectors.

I. Find Domain using Sublist3r Tool

Sublist3r, a Python-based tool, is designed to discover subdomains associated with a specified target website. Leveraging search engines and online web services, it scours the web for available subdomains linked to the designated target domain. Given the freedom to scrutinize any subdomain under reddit.com, it's prudent to identify additional subdomains for testing purposes.

To install Sublist3r, navigate to its GitHub repository at https://github.com/aboul3la/Sublist3r.git. This repository hosts all the necessary files required for installing the tool. Execute the following command in your shell to download it:

```
***git clone https://github.com/aboul3la/Sublist3r.git***
```

Please note that Sublist3r necessitates either Python 2.7 or Python 3.4 to operate smoothly.

After downloading the files, go inside the 'Sublist3r' directory and install the requirements by entering,

sudo pip install -r requirements.txt

After installing the requirements, enter

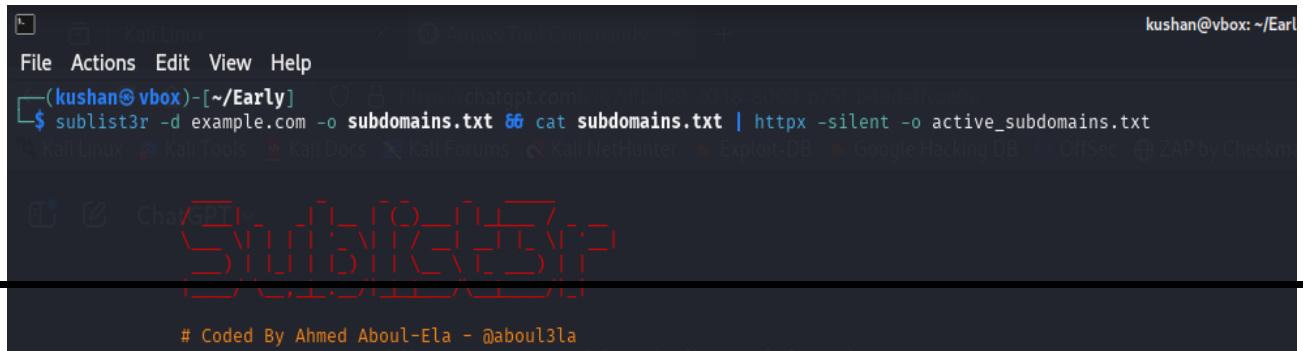
sublist3r -d wickr.com -o subdomains.txt

to find subdomains under the mentioned domain.

Upon examining for accessible subdomains, the next step involves identifying those that are operational. This can be accomplished by employing an additional tool known as '**httpx**'.

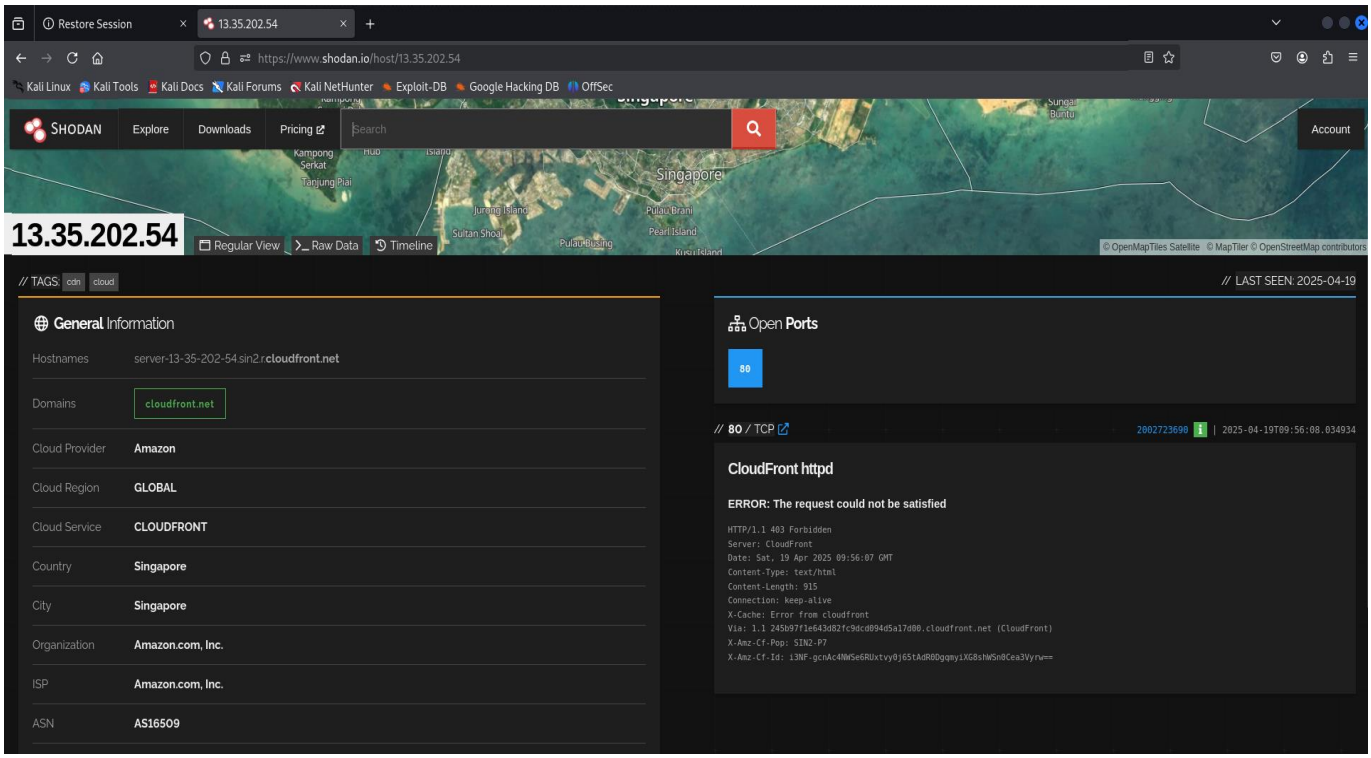
This tool can find domains that are up and running. To find active subdomains under this site, I am using the text file generated before by the sublist3r and writing the active subdomains to another new file.

Following the completion of the scan, the findings reveal that the majority of the subdomains are indeed active.



II. Identify exposed services using Shodan

Shodan is a potent search engine made to look through and index gadgets that are linked to the internet. Shodan concentrates on hardware, such as servers, routers, and Internet of Things devices, as well as services, such as web servers, databases, and remote access tools, in contrast to standard search engines that crawl websites. It is a useful tool for security researchers, penetration testers, and bug bounty hunters since it gathers metadata from these devices, such as banners, open ports, and software versions. Shodan can be used to find exposed services that could be at danger to the organization due to misconfigured or attack-prone settings.



III. Detect technologies using Whatweb

Whatweb is a powerful open-source tool designed to identify the technologies used by websites. It works by analyzing the responses from a web server, such as HTTP headers, HTML content, cookies, and scripts, to detect the underlying technologies.

To detect technologies used by a website, simply run :

whatweb wickr.com

This command will analyze the website and display a summary of the detected technologies**.**

```
┌──(kush㉿Kushan)-[~]
└─$ whatweb wickr.com
http://wickr.com [301 Moved Permanently] CloudFront, Country[UNITED STATES][US], HTTPServer[CloudFront], IP[3.165.75.112], RedirectLocation[https://wickr.com/], Title[301 Moved Permanently], UncommonHeaders[x-amz-cf-pop,alt-svc,x-amz-cf-id], Via-Proxy[1.1 f702fc84c341cf70cce98d6cffe36e54.cloudfront.net (CloudFront)]
https://wickr.com/ [200 OK] Country[UNITED STATES][US], Frame, HTML5, HTTPServer[AmazonS3], IP[13.35.202.112], JQuery, MetaGenerator[Elementor 3.18.3; features: e_font_icon_svg, block_editor_assets_optimize, e_image_loading_optimization; settings: css_print_method-internal, google_font-enabled, font_display-auto,Site Kit by Google 1.118.0], Open-Graph-Protocol[website], Script[application/ld+json], Strict-Transport-Security[max-age=31536000; includeSubDomains], Title[AWS Wickr | Protecting Communications with End-to-End Encryption], UncommonHeaders[x-amz-cf-pop,alt-svc,x-amz-cf-id], Via-Proxy[1.1 322fd7826352d6d295b7196056be4ec2.cloudfront.net (CloudFront)], WordPress

┌──(kush㉿Kushan)-[~]
└─$
```

To get detailed information about the detection process:

whatweb -v wickr.com

```
┌──(kush㉿Kushan)-[~]
└─$ whatweb -v wickr.com
WhatWeb report for http://wickr.com
Status    : 301 Moved Permanently
Title     : 301 Moved Permanently
IP        : 13.35.202.112
Country   : UNITED STATES, US

Summary   : CloudFront, HTTPServer[CloudFront], RedirectLocation[https://wickr.com/], UncommonHeaders[x-amz-cf-pop,alt-svc,x-amz-cf-id], Via-Proxy[1.1 19d4948b5334a7f8592cc99e40fc9ca2.cloudfront.net (CloudFront)]

Detected Plugins:
[ CloudFront ]
        CloudFront Server


[ HTTPServer ]
        HTTP server header string. This plugin also attempts to
        identify the operating system from the server header.

        String       : CloudFront (from server string)

[ RedirectLocation ]
        HTTP Server string location. used with http-status 301 and
        302

        String       : https://wickr.com/ (from location)

[ UncommonHeaders ]
        Uncommon HTTP server headers. The blacklist includes all
        the standard headers and many non standard but common ones.
        Interesting but fairly common headers should have their own
        plugins, eg. x-powered-by, server and x-aspnet-version.
        Info about headers can be found at www.http-stats.com

        String       : x-amz-cf-pop,alt-svc,x-amz-cf-id (from headers)

[ Via-Proxy ]
        This plugin extracts the proxy server details from the Via
        param of the HTTP header.

        String       : 1.1 19d4948b5334a7f8592cc99e40fc9ca2.cloudfront.net (CloudFront)

HTTP Headers:
        HTTP/1.1 301 Moved Permanently
        Server: CloudFront
        Date: Tue, 22 Apr 2025 08:13:22 GMT
```

```
WhatWeb report for https://wickr.com/
Status    : 200 OK
Title     : AWS Wickr | Protecting Communications with End-to-End Encryption
IP        : 108.156.144.11
Country   : UNITED STATES, US

Summary   : Frame, HTML5, HTTPServer[AmazonS3], JQuery, MetaGenerator[Elementor 3.18.3; features: e_font_icon_svg, block_editor_assets_optimize, e_image_loading_optimization; settings: css_print_method-internal, google_font-enabled, font_
display-auto,Site Kit by Google 1.118.0], Open-Graph-Protocol[website], Script[application/ld+json], Strict-Transport-Security[max-age=31536000; includeSubDomains], UncommonHeaders[x-amz-cf-pop,alt-svc,x-amz-cf-id], Via-Proxy[1.1 14641fda
3bf050f0f26ed5c961893124.cloudfront.net (CloudFront)], WordPress

Detected Plugins:
[ Frame ]
        This plugin detects instances of frame and iframe HTML
        elements.


[ HTML5 ]
        HTML version 5, detected by the doctype declaration


[ HTTPServer ]
        HTTP server header string. This plugin also attempts to
        identify the operating system from the server header.

        String       : AmazonS3 (from server string)

[ JQuery ]
        A fast, concise, JavaScript that simplifies how to traverse
        HTML documents, handle events, perform animations, and add
        AJAX.

        Website     : http://jquery.com/

[ MetaGenerator ]
        This plugin identifies meta generator tags and extracts its
        value.

        String       : Elementor 3.18.3; features: e_font_icon_svg, block_editor_assets_optimize, e_image_loading_optimization; settings: css_print_method-internal, google_font-enabled, font_display-auto,Site Kit by Google 1.118.0

[ Open-Graph-Protocol ]
        The Open Graph protocol enables you to integrate your Web
        pages into the social graph. It is currently designed for
        Web pages representing profiles of real-world things .
        things like movies, sports teams, celebrities, and
        restaurants. Including Open Graph tags on your Web page,
        makes your page equivalent to a Facebook Page.

        Version      : website
```

```
        a web browser from accessing a website without the security
        of the HTTPS protocol.

        String       : max-age=31536000; includeSubDomains

[ UncommonHeaders ]
        Uncommon HTTP server headers. The blacklist includes all
        the standard headers and many non standard but common ones.
        Interesting but fairly common headers should have their own
        plugins, eg. x-powered-by, server and x-aspnet-version.
        Info about headers can be found at www.http-stats.com

        String       : x-amz-cf-pop,alt-svc,x-amz-cf-id (from headers)

[ Via-Proxy ]
        This plugin extracts the proxy server details from the Via
        param of the HTTP header.

        String       : 1.1 14641fda3bf050f0f26ed5c961893124.cloudfront.net (CloudFront)

[ WordPress ]
        WordPress is an opensource blogging system commonly used as
        a CMS.

        Aggressive function available (check plugin file or details).
        Google Dorks: (1)
        Website     : http://www.wordpress.org/

HTTP Headers:
        HTTP/1.1 200 OK
        Content-Type: text/html
        Transfer-Encoding: chunked
        Connection: close
        Date: Wed, 05 Mar 2025 20:55:17 GMT
        Server: AmazonS3
        ETag: W/"6800ff1d78ba40b05a0e018a4354e348"
        Last-Modified: Wed, 05 Mar 2025 20:54:30 GMT
        Cache-Control: public, max-age=0, s-maxage=31536000
        strict-transport-security: max-age=31536000; includeSubDomains
        Content-Encoding: gzip
        Vary: Accept-Encoding
        X-Cache: Hit from cloudfront
        Via: 1.1 14641fda3bf050f0f26ed5c961893124.cloudfront.net (CloudFront)
        X-Amz-Cf-Pop: SIN2-P7
        Alt-Svc: h3=":443"; ma=86400
        X-Amz-Cf-Id: QkV7LCOq2OJkeO7P0-iD0E7BuRbNy95NwxEznBi9Xx9vSVx4TqlCQg==
        Age: 4101489
```

3. Scanning Vulnerability Identifies

One of the most important steps in finding security flaws in a system, network, or application is vulnerability scanning. It entails identifying known vulnerabilities, configuration errors, and possible attack routes using automated technologies. The objective is to evaluate the target's security posture and offer practical advice to reduce risks. For this, tools **like Nessus, OpenVAS, Nikto**, and **Nmap** are frequently utilized. In order to find vulnerabilities like out-of-date software, shoddy setups, or exposed sensitive data, the procedure involves scanning open ports, services, and applications.

i. Open ports services

Nmap (Network Mapper) is a powerful tool for scanning open ports and identifying running services on a target system. By using the **nmap -sV** command, you can detect the version of services running on open ports, helping assess potential vulnerabilities. The -p- option scans all 65,535 ports, while -A enables OS detection, version detection, script scanning, and traceroute for a comprehensive analysis. The results typically display open ports, their associated services, and potential security risks, making it an essential tool for penetration testers and system administrators.

Scan the most commonly used on **wickr.com,**

```
┌──(kush㉿Kushan)-[~]
└─$ nmap wickr.com
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-22 03:24 CDT
Nmap scan report for wickr.com (13.35.202.112)
Host is up (0.0080s latency).
Other addresses for wickr.com (not scanned): 13.35.202.73 13.35.202.54 13.35.202.50
rDNS record for 13.35.202.112: server-13-35-202-112.sin2.r.cloudfront.net
Not shown: 995 filtered tcp ports (no-response)
PORT     STATE SERVICE
25/tcp   open  smtp
80/tcp   open  http
443/tcp  open  https
2000/tcp open  cisco-sccp
5060/tcp open  sip

Nmap done: 1 IP address (1 host up) scanned in 4.87 seconds
```

Identify services running on open ports,

```
┌──(kush㉿Kushan)-[~]
└─$ nmap -sV wickr.com
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-22 03:24 CDT
Nmap scan report for wickr.com (13.35.202.112)
Host is up (0.011s latency).
Other addresses for wickr.com (not scanned): 13.35.202.50 13.35.202.54 13.35.202.73
rDNS record for 13.35.202.112: server-13-35-202-112.sin2.r.cloudfront.net
Not shown: 995 filtered tcp ports (no-response)
PORT     STATE SERVICE     VERSION
25/tcp   open  smtp?
80/tcp   open  http        Amazon CloudFront httpd
443/tcp  open  ssl/http    Amazon CloudFront httpd
2000/tcp open  cisco-sccp?
5060/tcp open  sip?
3 services unrecognized despite returning data. If you know the service/version, please submit the following fingerprints at https://nmap.org/cgi-bin/submit.cgi?new-service :
==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
SF-Port25-TCP:V=7.95%I=7%D=4/22%Time=68075263%P=x86_64-pc-linux-gnu%r(Hell
SF:o,2A,"552\x20Invalid\x20domain\x20name\x20in\x20EHLO\x20command\.\r\n")
SF:%r(GenericLines,28,"500\x20Syntax\x20error,\x20command\x20unrecognized\
SF:r\n")%r(GetRequest,32A0,"HTTP/1\.1\x20403\x20Forbidden\r\nX-Frame-Optio
SF:ns:\x20SAMEORIGIN\r\nX-XSS-Protection:\x201;\x20mode=block\r\nX-Content
SF:-Type-Options:\x20nosniff\r\nContent-Security-Policy:\x20frame-ancestor
SF:s\x20'self'\r\nContent-Type:\x20text/html;\x20charset=\"utf-8\"\r\nCont
SF:ent-Length:\x2013709\r\nConnection:\x20Close\r\n\r\n<!DOCTYPE\x20html><
SF:html\x20lang=\"en\">\x20<head>\x20<meta\x20charset=\"UTF-8\">\x20<meta\
SF:x20http-equiv=\"X-UA-Compatible\"\x20content=\"IE=8;\x20IE=EDGE\">\x20<
SF:meta\x20name=\"viewport\"\x20content=\"width=device-width,\x20initial-s
SF:cale=1\">\x20<style\x20type=\"text/css\">\x20body\x20{\x20height:\x2010
SF:0%;\x20font-family:\x20Helvetica,\x20Arial,\x20sans-serif;\x20color:\x2
SF:0#6a6a6a;\x20margin:\x200;\x20display:\x20flex;\x20align-items:\x20cent
SF:er;\x20justify-content:\x20center;\x20}\x20input\[type=date\],\x20input
SF:\[type=email\],\x20input\[type=number\],\x20input\[type=password\],\x20
SF:input\[type=search\],\x20input\[type=tel\],\x20input\[type=text\],\x20i
SF:nput\[type=time\],\x20input\[type=url\],\x20select,\x20textarea\x20{\x2
SF:0color:\x20#262626;\x20vertical-align:\x20baseline;\x20margin:\x20\.2em
SF:;\x20border-style:\x20solid;\x20border-width")%r(HTTPOptions,368B,"HTTP
SF:/1\.1\x20403\x20Forbidden\r\nX-Frame-Options:\x20SAMEORIGIN\r\nX-XSS-Pr
SF:otection:\x201;\x20mode=block\r\nX-Content-Type-Options:\x20nosniff\r\n
SF:Content-Security-Policy:\x20frame-ancestors\x20'self'\r\nContent-Type:\
SF:x20text/html;\x20charset=\"utf-8\"\r\nContent-Length:\x2013709\r\nConne
SF:ction:\x20Close\r\n\r\n<!DOCTYPE\x20html><html\x20lang=\"en\">\x20<head
SF:>\x20<meta\x20charset=\"UTF-8\">\x20<meta\x20http-equiv=\"X-UA-Compatib
SF:le\"\x20content=\"IE=8;\x20IE=EDGE\">\x20<meta\x20name=\"viewport\"\x20
SF:content=\"width=device-width,\x20initial-scale=1\">\x20<style\x20type=\
SF:"text/css\">\x20body\x20{\x20height:\x20100%;\x20font-family:\x20Helvet
SF:ica,\x20Arial,\x20sans-serif;\x20color:\x20#6a6a6a;\x20margin:\x200;\x2
SF:0display:\x20flex;\x20align-items:\x20center;\x20justify-content:\x20ce
```

To get more detailed information, including **operating system detection**

```
┌──(kush㉿Kushan)-[~]
└─$ nmap -A wickr.com
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-22 03:29 CDT
Stats: 0:03:09 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 90.00% done; ETC: 03:32 (0:00:01 remaining)
Stats: 0:03:17 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 97.50% done; ETC: 03:33 (0:00:00 remaining)
Nmap scan report for wickr.com (13.35.202.50)
Host is up (0.0021s latency).
Other addresses for wickr.com (not scanned): 13.35.202.54 13.35.202.112 13.35.202.73
rDNS record for 13.35.202.50: server-13-35-202-50.sin2.r.cloudfront.net
Not shown: 995 filtered tcp ports (no-response)
PORT     STATE SERVICE     VERSION
25/tcp   open  smtp?
| fingerprint-strings:
|   GenericLines:
|     500 Syntax error, command unrecognized
|   GetRequest, HTTPOptions:
|     HTTP/1.1 403 Forbidden
|     X-Frame-Options: SAMEORIGIN
|     X-XSS-Protection: 1; mode=block
|     X-Content-Type-Options: nosniff
|     Content-Security-Policy: frame-ancestors 'self'
|     Content-Type: text/html; charset="utf-8"
|     Content-Length: 13708
|     Connection: Close
|     <!DOCTYPE html><html lang="en"> <head> <meta charset="UTF-8"> <meta http-equiv="X-UA-Compatible" content="IE=8; IE=EDGE"> <meta name="viewport" content="width=device-width, initial-scale=1"> <style type="text/css"> body { height: 10
0%; font-family: Helvetica, Arial, sans-serif; color: #6a6a6a; margin: 0; display: flex; align-items: center; justify-content: center; } input[type=date], input[type=email], input[type=number], input[type=password], input[type=search], in
put[type=tel], input[type=text], input[type=time], input[type=url], select, textarea { color: #262626; vertical-align: baseline; margin: .2em; border-style: solid; border-width
|   Hello:
|_    552 Invalid domain name in EHLO command.
|_smtp-commands: Couldn't establish connection on port 25
80/tcp   open  http        Amazon CloudFront httpd
|_http-server-header: CloudFront
|_http-title: Did not follow redirect to https://wickr.com/
443/tcp  open  ssl/http    Amazon CloudFront httpd
|_http-generator: Site Kit by Google 1.118.0
| http-server-header:
|   AmazonS3
|_  CloudFront
|_http-title: AWS Wickr | Protecting Communications with End-to-End Encryption
| ssl-cert: Subject: commonName=*.wickr.com
| Subject Alternative Name: DNS:*.wickr.com, DNS:wickr.com
| Not valid before: 2025-04-10T00:00:00
|_Not valid after:  2026-05-09T23:59:59
2000/tcp open  cisco-sccp?
| fingerprint-strings:
```

ii. Web vulnerabilities

Nikto is an open-source web server scanner designed to identify vulnerabilities, outdated software, and security misconfigurations on web servers. It performs comprehensive testing for over 6700 vulnerabilities, including misconfigured files, outdated server software, and security holes.

Nikto -h wickr.com using this command will scan zellepay.force.com for vulnerabilities, misconfigurations, and security issues.

```
┌──(kush㉿Kushan)-[~]
└─$ nikto -h wickr.com
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Multiple IPs found: 13.35.202.112, 13.35.202.54, 13.35.202.50, 13.35.202.73
+ Target IP:          13.35.202.112
+ Target Hostname:    wickr.com
+ Target Port:        80
+ Start Time:         2025-04-22 04:01:36 (GMT-5)
---------------------------------------------------------------------------
+ Server: CloudFront
+ /: Retrieved via header: 1.1 a2c2ae15e1030d0678971e68a40255b6.cloudfront.net (CloudFront).
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: An alt-svc header was found which is advertising HTTP/3. The endpoint is: ':443'. Nikto cannot test HTTP/3 over QUIC. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/alt-svc
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/miss
g-content-type-header/
+ Root page / redirects to: https://wickr.com/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ ERROR: Error limit (20) reached for host, giving up. Last error: opening stream: getaddrinfo problems (Temporary failure in name resolution): Resource temporarily unavailable
+ Scan terminated: 19 error(s) and 4 item(s) reported on remote host
+ End Time:           2025-04-22 04:09:47 (GMT-5) (491 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Scans both HTTP and HTTPS,

```
┌──(kush㉿vbox)-[~]
└─$ nikto -h wickr.com -p 80,443
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Multiple IPs found: 13.35.202.54, 13.35.202.50, 13.35.202.73, 13.35.202.112
+ Multiple IPs found: 13.35.202.54, 13.35.202.50, 13.35.202.73, 13.35.202.112
+ Target IP:          13.35.202.54
+ Target Hostname:    wickr.com
+ Target Port:        80
+ Start Time:         2025-04-23 00:50:33 (GMT-4)
---------------------------------------------------------------------------
+ Server: CloudFront
+ /: Retrieved via header: 1.1 14641fda3bf050f0f26ed5c961893124.cloudfront.net (CloudFront).
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: An alt-svc header was found which is advertising HTTP/3. The endpoint is: ':443'. Nikto cannot test HTTP/3 over QUIC. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/alt-svc
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missin
g-content-type-header/
+ Root page / redirects to: https://wickr.com/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ ERROR: Error limit (20) reached for host, giving up. Last error: error reading HTTP response
+ Scan terminated: 20 error(s) and 4 item(s) reported on remote host
+ End Time:           2025-04-23 01:00:20 (GMT-4) (587 seconds)
---------------------------------------------------------------------------
+ Target IP:          13.35.202.54
+ Target Hostname:    wickr.com
+ Target Port:        443
+ Start Time:         2025-04-23 01:00:20 (GMT-4)
---------------------------------------------------------------------------
+ Server: CloudFront
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missin
g-content-type-header/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ ERROR: Error limit (20) reached for host, giving up. Last error: error reading HTTP response
+ Scan terminated: 19 error(s) and 2 item(s) reported on remote host
+ End Time:           2025-04-23 01:14:35 (GMT-4) (855 seconds)
---------------------------------------------------------------------------
+ 2 host(s) tested

┌──(kush㉿vbox)-[~]
```

nikto -h https://wickr.com -ssl using this command runs a **Nikto** scan on https://zellepay.force.com while explicitly forcing SSL/TLS encryption.

```
┌──(kush㉿vbox)-[~]
└─$ nikto -h wickr.com -ssl
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Multiple IPs found: 13.35.202.112, 13.35.202.73, 13.35.202.50, 13.35.202.54
+ Target IP:          13.35.202.112
+ Target Hostname:    wickr.com
+ Target Port:        443
---------------------------------------------------------------------------
+ SSL Info:        Subject:  /CN=*.wickr.com
                   Ciphers:  TLS_AES_128_GCM_SHA256
                   Issuer:   /C=US/O=Amazon/CN=Amazon RSA 2048 M03
+ Start Time:         2025-04-23 01:23:45 (GMT-4)
---------------------------------------------------------------------------
+ Server: AmazonS3
+ /: Retrieved via header: 1.1 c09ac2ca4c9ff108eb1cd78177168cde.cloudfront.net (CloudFront).
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: An alt-svc header was found which is advertising HTTP/3. The endpoint is: ':443'. Nikto cannot test HTTP/3 over QUIC. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/alt-svc
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missin
g-content-type-header/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ ; Server banner changed from 'AmazonS3' to 'CloudFront'.
+ /: The Content-Encoding header is set to "deflate" which may mean that the server is vulnerable to the BREACH attack. See: http://breachattack.com/
+ ERROR: Error limit (20) reached for host, giving up. Last error: opening stream: can't connect: SSL negotiation failed: error:0A000410:SSL routines::ssl/tls alert handshake failure at /var/lib/nikto/plugins/LW2.pm line 5254.
 at /var/lib/nikto/plugins/LW2.pm line 5254.
;  at /var/lib/nikto/plugins/LW2.pm line 5254.
+ Scan terminated: 20 error(s) and 6 item(s) reported on remote host
+ End Time:           2025-04-23 01:34:07 (GMT-4) (622 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Automated Testing

For automated testing, I've selected OWASP ZAP widely used tool within the industry.

OWASP ZAP

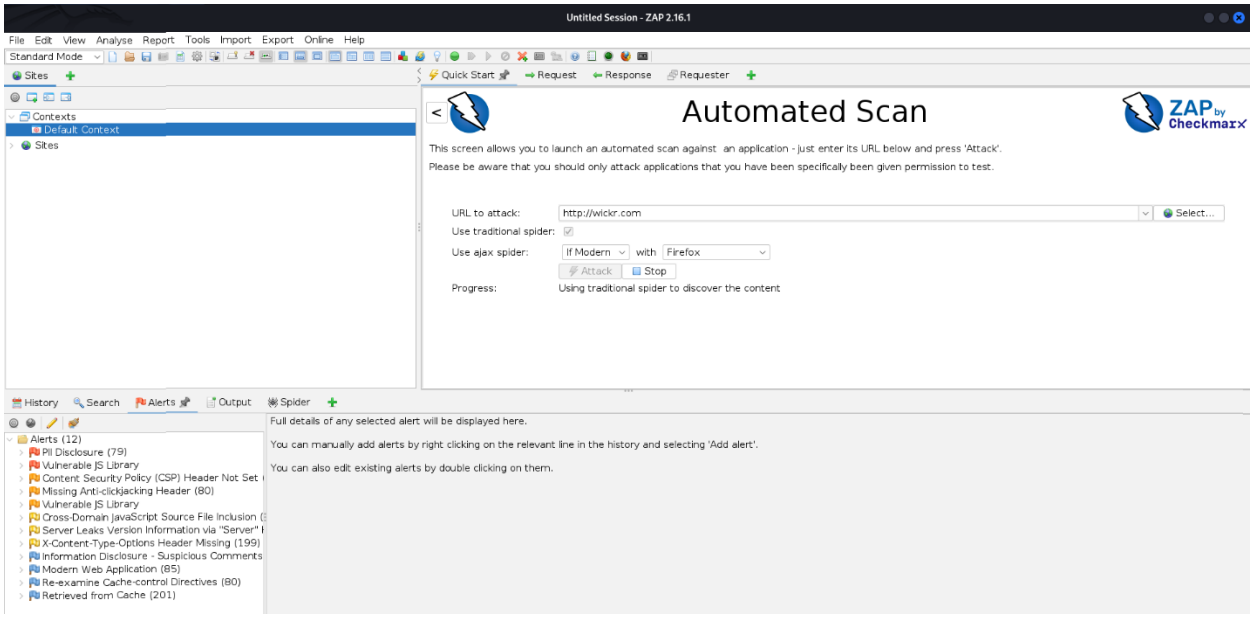
The Open Web Application Security Project Zed Attack Proxy (OWASP ZAP) is an open-source vulnerability scanner renowned for its capability to function as a Man-in-the-Middle (MITM) proxy. It assesses various vulnerabilities by scrutinizing responses from the web application or server. Notably convenient to utilize, OWASP ZAP offers customization options through the installation of modules, enabling efficient management of results.

Within this proxy, there are primarily two scan types available:

1. Automated Scan: Users input the target URL and initiate the attack. The behavior can be tailored by selecting the ZAP mode. This triggers all scripts against the target to detect vulnerabilities and generates reports accordingly.

2. Manual Explore: Users can navigate to the target web application and commence exploration. During manual exploration, ZAP HUD (Heads Up Display) captures each page, while the ZAP proxy records responses.

For this assessment, I am running ZAP on automated mode.



After specifying the target URL in the designated textbox, simply select "Attack" to initiate the scanning process. Upon completion, a comprehensive report of the findings can be generated by selecting "Report." Below are screenshots showcasing the results obtained after scanning several domains.

iii. Web server misconfigurations

Detailed Analysis of Missing Security Headers

1. **Missing X-Frame-Options Header**

Risk: The absence of the X-Frame-Options header makes the website potentially vulnerable to **clickjacking attacks**.

Impact:

- An attacker can embed the website inside an invisible or disguised <**iframe**> on a malicious page.
- Users may unknowingly interact with hidden UI elements (ex:-clicking buttons that perform unintended actions like fund transfers or password changes).
- This could lead to unauthorized transactions, account takeovers, or phishing scams if sensitive actions are exposed.

2. **Missing X-Content-Type-Options Header**

Risk: Without this header, browsers may perform **MIME sniffing**, which can lead to:

- **Cross-Site Scripting (XSS)**: If a file (ex:- an uploaded image) is misinterpreted as executable code.
- **Content Spoofing**: Attackers could disguise malicious scripts as harmless files (ex:- .jpg executing as JavaScript).

Impact:

- Exploitable in file upload features or improperly served static content.
- Could allow attackers to bypass security filters and execute malicious scripts in the context of the website.

`

4. **Exploitation & Validation**

Clickjacking Attack Analysis

Clickjacking is a UI redressing attack where an attacker embeds a legitimate website inside an invisible or disguised iframe on a malicious page. This tricks users into clicking elements they don't intend to, leading to fraudulent actions, session hijacking, or sensitive data exposure.

The absence of the **X-Frame-Options** or **Content-Security-Policy (CSP) frame-ancestors** headers makes the application vulnerable to Clickjacking.

If it reports **"X-Frame-Options header is missing"**, the site might be vulnerable

```
┌──(kush㉿Kushan)-[~/XSStrike]
└─$ nmap --script http-headers -p 80,443 zellepay.force.com

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-27 09:01 CDT
Nmap scan report for zellepay.force.com (136.146.40.218)
Host is up (0.30s latency).
Other addresses for zellepay.force.com (not scanned): 136.146.45.218 136.146.43.218
rDNS record for 136.146.40.218: dcl9-ncg1-c8-iad5.na240-ia7.force.com

PORT    STATE SERVICE
80/tcp  open  http
| http-headers:
|   Date: Thu, 27 Mar 2025 14:01:50 GMT
|   Set-Cookie: CookieConsentPolicy=0:1; path=/; expires=Fri, 27-Mar-2026 14:01:50 GMT; Max-Age=31536000; secure
|   Set-Cookie: LSKey-c$CookieConsentPolicy=0:1; path=/; expires=Fri, 27-Mar-2026 14:01:50 GMT; Max-Age=31536000; secure
|   Content-Security-Policy: upgrade-insecure-requests
|   Cache-Control: no-cache,must-revalidate,max-age=0,no-store,private
|   Expires: Thu, 01 Jan 1970 00:00:00 GMT
|   Location: https://zelleservice.my.site.com/
|   Content-Length: 0
|   Connection: close
|
|_  (Request type: GET)
443/tcp open  https
| http-headers:
|   Date: Thu, 27 Mar 2025 14:01:33 GMT
|   Set-Cookie: CookieConsentPolicy=0:1; path=/; expires=Fri, 27-Mar-2026 14:01:33 GMT; Max-Age=31536000; secure
|   Set-Cookie: LSKey-c$CookieConsentPolicy=0:1; path=/; expires=Fri, 27-Mar-2026 14:01:33 GMT; Max-Age=31536000; secure
|   Content-Security-Policy: upgrade-insecure-requests
|   Strict-Transport-Security: max-age=63072000; includeSubDomains
|   Cache-Control: no-cache,must-revalidate,max-age=0,no-store,private
|   Expires: Thu, 01 Jan 1970 00:00:00 GMT
|   Location: https://zelleservice.my.site.com/
|   Content-Length: 0
|   Connection: close
|
|_  (Request type: GET)

Nmap done: 1 IP address (1 host up) scanned in 19.64 seconds
```

5. **Report Writing**

Title: Use of Vulnerable JavaScript Library: DOMPurify v3.0.3

Summary: A vulnerable version of the DOMPurify JavaScript library (v3.0.3) was detected at the following URL:

https://wickr.com/wp-content/plugins/elementor-pro/assets/js/preloaded-elements-handlers.min.js

This version has multiple known vulnerabilities that could be exploited to bypass sanitization and potentially lead to Cross-Site Scripting (XSS) attacks. The issues are related to improper handling of SVG elements and URI-based payloads.

Affected Component:

- DOMPurify JavaScript Library v3.0.3
- File: preloaded-elements-handlers.min.js
- Plugin: elementor-pro on WordPress

Impact Assessment (Risk: High | Confidence: Medium):
The vulnerability may allow attackers to bypass sanitization logic, inject malicious scripts, and execute arbitrary code in the context of the user's session. This could lead to:

- Session hijacking
- Credential theft
- Defacement
- Browser exploitation
- ZAP Reference: org/zaproxy/zap/extension/pscanrules/AntiClickjackingScanRule.java

Steps to Reproduce

Passive Detection via Retire.js/ZAP:

1. Scan https://wickr.com with OWASP ZAP or Retire.js.

2. Alert triggered:
   Passive (10003 - Vulnerable JS Library)
3. Identified vulnerable JS:
   DOMPurify v3.0.3
4. Library path:
   https://wickr.com/wp-content/plugins/elementor-pro/assets/js/preloaded-elements-handlers.min.js

Evidence:

- Tool Used: OWASP ZAP Passive Scan
- Source: Passive Detection (Retire.js Plugin)
- Alert ID: 10003 – Vulnerable JS Library
- CWE ID: CWE-1395: Dependency on Vulnerable Third-Party Component
  origin.

Known Vulnerabilities in DOMPurify v3.0.3

- **CVE-2024-47875**
- **CVE-2025-26791**
- **CVE-2024-45801**

Security Advisories & Fixes:

- **https://github.com/advisories/GHSA-gx9m-whjm-85jf**
- **https://github.com/cure53/DOMPurify/releases/tag/3.2.4**
- **https://github.com/cure53/DOMPurify/security/advisories/GHSA-mmhx-hmjr-r674**
- **Patches:**
  - **https://github.com/cure53/DOMPurify/commit/0ef5e537a514f904b6aa1d7ad9e749e365d7185f**

- **https://github.com/cure53/DOMPurify/commit/6ea80cd8b47640c20f2f230c7920b1f4ce4fdf7a**

Recommended Mitigation

- Upgrade DOMPurify to the Latest Version: Update the DOMPurify library to **v3.2.4 or newer**, which contains patches for the known vulnerabilities.
- Perform Dependency Audits: Regularly monitor third-party JS libraries using tools like;
  - OWASP Dependency-Check
  - Retire.js
  - Snyk

- Security Headers & CSP**:** Implement strict Content-Security-Policy headers to reduce XSS impact if any script injection occurs.

References

- DOMPurify Release Notes
- OWASP A06:2021 – Vulnerable and Outdated Components
- DOMPurify Advisory - GHSA-gx9m-whjm-85jf

Conclusion

The presence of DOMPurify v3.0.3 in a production asset of Wickr.com introduces a high-risk vulnerability due to multiple CVEs. This issue can allow malicious input to bypass client-side sanitization mechanisms,

leading to XSS and other attacks. An immediate upgrade of the affected library is strongly recommended.