//CODE

import sys import socket import subprocess import os import time import signal import random import string import threading import re

from urllib.parse import urlsplit # Scan Time Elapser intervals = (

('h', 3600),

('m', 60),

('s', 1),

)

def display\_time(seconds, granularity=3): result = [] seconds = seconds + 1

for name, count in intervals: value = seconds // count if value:

seconds -= value \* count result.append("{}{}".format(value, name)) return ' '.join(result[:granularity]) def url\_maker(url):

if not re.match(r'http(s?)\:', url):

url = 'http://' + url parsed = urlsplit(url) host = parsed.netloc if host.startswith('www.'):

host = host[4:] return host def check\_internet():

os.system('ping -c1 github.com > sa\_net 2>&1') if "0% packet loss" in open('sa\_net').read():

val = 1 else: val = 0

os.system('rm sa\_net > /dev/null 2>&1') return val

# Initializing the color module class class bcolors:

HEADER = '\033[95m'

OKBLUE = '\033[94m'

OKGREEN = '\033[92m'

WARNING = '\033[93m'

BADFAIL = '\033[91m'

ENDC = '\033[0m'

BOLD = '\033[1m'

UNDERLINE = '\033[4m'

BG\_ERR\_TXT = '\033[41m' # For critical errors and crashes BG\_HEAD\_TXT = '\033[100m'

BG\_ENDL\_TXT = '\033[46m'

BG\_CRIT\_TXT = '\033[45m'

BG\_HIGH\_TXT = '\033[41m'

BG\_MED\_TXT = '\033[43m'

BG\_LOW\_TXT = '\033[44m'

BG\_INFO\_TXT = '\033[42m'

# Classifies the Vulnerability's Severity def vul\_info(val): result = '' if val == 'c':

result = bcolors.BG\_CRIT\_TXT+" critical "+bcolors.ENDC elif val == 'h':

result = bcolors.BG\_HIGH\_TXT+" high "+bcolors.ENDC elif val == 'm':

result = bcolors.BG\_MED\_TXT+" medium "+bcolors.ENDC elif val == 'l':

result = bcolors.BG\_LOW\_TXT+" low "+bcolors.ENDC else:

result = bcolors.BG\_INFO\_TXT+" info "+bcolors.ENDC return result

# Legends proc\_high = bcolors.BADFAIL + "●" + bcolors.ENDC proc\_med = bcolors.WARNING + "●" + bcolors.ENDC proc\_low = bcolors.OKGREEN + "●" + bcolors.ENDC

# Links the vulnerability with threat level and remediation database def vul\_remed\_info(v1, v2, v3):

print(bcolors.BOLD+"Vulnerability Threat Level"+bcolors.ENDC) print("\t"+vul\_info(v2)+" "+bcolors.WARNING + \ str(tool\_resp[v1][0])+bcolors.ENDC)

print(bcolors.BOLD+"Vulnerability Definition"+bcolors.ENDC) print("\t"+bcolors.BADFAIL+str(tools\_fix[v3-1][1])+bcolors.ENDC) print(bcolors.BOLD+"Vulnerability Remediation"+bcolors.ENDC) print("\t"+bcolors.OKGREEN+str(tools\_fix[v3-1][2])+bcolors.ENDC)

# Scanner Help Context def helper():

print(bcolors.OKBLUE+"Information:"+bcolors.ENDC) print("------------")

print("\t./scanner.py example.com: Scans the domain example.com") print("\t./scanner.py --update : Updates the scanner to the latest version.") print("\t./scanner.py --help : Displays this help context.") print(bcolors.OKBLUE+"Interactive:"+bcolors.ENDC) print("------------")

print("\tCtrl+C: Skips current test.") print("\tCtrl+Z: Quits Scanner.")

print(bcolors.OKBLUE+"Legends:"+bcolors.ENDC) print("--------") print("\t["+proc\_high + \

"]: Scan process may take longer times (not predictable).") print("\t["+proc\_med+"]: Scan process may take less than 10 minutes.") print("\t["+proc\_low+"]: Scan process may take less than a minute or two.") print(bcolors.OKBLUE+"Vulnerability Information:"+bcolors.ENDC) print("--------------------------") print("\t" + \ vul\_info(

'c')+": Requires immediate attention as it may lead to compromise or service unavailability.") print("\t" + \ vul\_info(

'h')+" : May not lead to an immediate compromise, but there are high chances of probability.") print("\t" + \ vul\_info(

'm')+" : Attacker may correlate multiple vulnerabilities of this type to launch a sophisticated attack.") print("\t" + \ vul\_info(

'l')+" : Not a serious issue, but it is recommended to attend the finding.") print("\t" + \

vul\_info(

'i')+" : Not classified as a vulnerability, simply an useful informational alert to be considered.\n") # Clears Line def clear():

sys.stdout.write("\033[F") sys.stdout.write("\033[K")

# Scanner Logo def logo():

print(bcolors.WARNING)

print("""

O O O O O O O O O O O O O O O O O O O O O O O

O O O O O O O O O O O O

O O O O O O O O O O O O

O O O O O O O O O O O O O O O O

O O O O O O O O O O O O O O O O O

O O O O O O O O O O O O O O

O O O O O O O O O O O O O O O O O O O O O O O O

MINI PROJECT BY VINOD AND TEAM""")

print(bcolors.ENDC)

# Initiliazing the idle loader/spinner class class Spinner: busy = False delay = 0.10 @staticmethod def spinning\_cursor():

while 1:

for cursor in ' ( \* ) ( \* ) ': yield cursor # ←↑↓→

# for cursor in '←↑↓→': yield cursor def \_\_init\_\_(self, delay=None):

self.spinner\_generator = self.spinning\_cursor() if delay and float(delay):

self.delay = delay def spinner\_task(self): try:

while self.busy:

# sys.stdout.write(next(self.spinner\_generator)) print(bcolors.BG\_ERR\_TXT + \

next(self.spinner\_generator)+bcolors.ENDC, end=' ') sys.stdout.flush() time.sleep(self.delay) sys.stdout.write('\b') sys.stdout.flush() except (KeyboardInterrupt, SystemExit):

# clear()

print("\n\t" + bcolors.BG\_ERR\_TXT + \

"Scanner received a series of Ctrl+C hits. Quitting..." + bcolors.ENDC) sys.exit(1) def start(self):

self.busy = True

threading.Thread(target=self.spinner\_task).start() def stop(self): try:

self.busy = False time.sleep(self.delay) except (KeyboardInterrupt, SystemExit):

# clear()

print("\n\t" + bcolors.BG\_ERR\_TXT + \

"Scanner received a series of Ctrl+C hits. Quitting..." + bcolors.ENDC) sys.exit(1)

# End ofloader/spinner class # Instantiating the spinner/loader class spinner = Spinner()

# Scanners that will be used and filename rotation (default: enabled (1)) tool\_names = [

["host", "Host - Checks for existence of IPV6 address.", "host", 1],

["wp\_check","WordPress Checker - Checks for WordPress Installation.","wget",1],

["uniscan","Uniscan - Checks for robots.txt & sitemap.xml","uniscan",1],

["wafw00f","Wafw00f - Checks for Application Firewalls.","wafw00f",1],

["nmap","Nmap - Fast Scan [Only Few Port Checks]","nmap",1],

["nmap\_header","Nmap [XSS Filter Check] - Checks if XSS Protection Header is present.","nmap",1],

["nmap\_sloris","Nmap [Slowloris DoS] - Checks for Slowloris Denial of Service Vulnerability.","nmap",1],

["nmap\_hbleed","Nmap [Heartbleed] - Checks only for Heartbleed Vulnerability.","nmap",1],

["nmap\_poodle","Nmap [POODLE] - Checks only for Poodle Vulnerability.","nmap",1],

["nmap\_ccs","Nmap [OpenSSL CCS Injection] - Checks only for CCS Injection.","nmap",1],

["nmap\_freak","Nmap [FREAK] - Checks only for FREAK Vulnerability.","nmap",1],

["nmap\_logjam","Nmap [LOGJAM] - Checks for LOGJAM Vulnerability.","nmap",1],

["nmap\_telnet","Nmap [TELNET] - Checks if TELNET service is running.","nmap",1],

["nmap\_ftp","Nmap [FTP] - Checks if FTP service is running.","nmap",1],

["nmap\_stuxnet","Nmap [STUXNET] - Checks if the host is affected by STUXNET Worm.","nmap",1],

["webdav","WebDAV - Checks if WEBDAV enabled on Home directory.","davtest",1],

["golismero\_finger","Golismero - Does a fingerprint on the Domain.","golismero",1],

["uniscan\_filebrute","Uniscan - Brutes for Filenames on the Domain.","uniscan",1],

["uniscan\_dirbrute", "Uniscan - Brutes Directories on the Domain.","uniscan",1],

["uniscan\_ministresser", "Uniscan - Stress Tests the Domain.","uniscan",1],

["uniscan\_rfi","Uniscan - Checks for LFI, RFI and RCE.","uniscan",1],

["uniscan\_xss","Uniscan - Checks for XSS, SQLi, BSQLi & Other Checks.","uniscan",1],

["nikto\_xss","Nikto - Checks for Apache Expect XSS Header.","nikto",1],

["nikto\_subrute","Nikto - Brutes Subdomains.","nikto",1],

["nikto\_shellshock","Nikto - Checks for Shellshock Bug.","nikto",1],

["nikto\_internalip","Nikto - Checks for Internal IP Leak.","nikto",1],

["nikto\_putdel","Nikto - Checks for HTTP PUT DEL.","nikto",1],

["nikto\_headers","Nikto - Checks the Domain Headers.","nikto",1],

["nikto\_ms01070","Nikto - Checks for MS10-070 Vulnerability.","nikto",1],

["nikto\_servermsgs","Nikto - Checks for Server Issues.","nikto",1],

["nikto\_outdated","Nikto - Checks if Server is Outdated.","nikto",1],

["nikto\_httpoptions","Nikto - Checks for HTTP Options on the Domain.","nikto",1], ["nikto\_cgi","Nikto - Enumerates CGI Directories.","nikto",1],

["nikto\_ssl","Nikto - Performs SSL Checks.","nikto",1],

["nikto\_sitefiles","Nikto - Checks for any interesting files on the Domain.","nikto",1],

["nikto\_paths","Nikto - Checks for Injectable Paths.","nikto",1],

["dnsmap\_brute","DNSMap - Brutes Subdomains.","dnsmap",1],

["nmap\_sqlserver","Nmap - Checks for MS-SQL Server DB","nmap",1],

["nmap\_mysql", "Nmap - Checks for MySQL DB","nmap",1],

["nmap\_oracle", "Nmap - Checks for ORACLE DB","nmap",1],

["nmap\_rdp\_udp","Nmap - Checks for Remote Desktop Service over UDP","nmap",1],

["nmap\_rdp\_tcp","Nmap - Checks for Remote Desktop Service over TCP","nmap",1],

["nmap\_full\_ps\_tcp","Nmap - Performs a Full TCP Port Scan","nmap",1],

["nmap\_full\_ps\_udp","Nmap - Performs a Full UDP Port Scan","nmap",1],

["nmap\_snmp","Nmap - Checks for SNMP Service","nmap",1],

["aspnet\_elmah\_axd","Checks for ASP.net Elmah Logger","wget",1],

["nmap\_tcp\_smb","Checks for SMB Service over TCP","nmap",1],

["nmap\_udp\_smb","Checks for SMB Service over UDP","nmap",1],

["wapiti","Wapiti - Checks for SQLi, RCE, XSS and Other Vulnerabilities","wapiti",1],

["nmap\_iis","Nmap - Checks for IIS WebDAV","nmap",1],

["whatweb","WhatWeb - Checks for X-XSS Protection Header","whatweb",1]]

# Command that is used to initiate the tool (with parameters and extra params) tool\_cmd = [

["host ",""],

["wget -O temp\_wp\_check --tries=1 ","/wp-admin"],

["uniscan -e -u ",""],

["wafw00f ",""],

["nmap -F --open -Pn ",""],

["nmap -p80 --script http-security-headers -Pn ",""],

["nmap -p80,443 --script http-slowloris --max-parallelism 500 -Pn ",""],

["nmap -p443 --script ssl-heartbleed -Pn ",""],

["nmap -p443 --script ssl-poodle -Pn ",""],

["nmap -p443 --script ssl-ccs-injection -Pn ",""],

["nmap -p443 --script ssl-enum-ciphers -Pn ",""],

["nmap -p443 --script ssl-dh-params -Pn ",""],

["nmap -p23 --open -Pn ",""],

["nmap -p21 --open -Pn ",""],

["nmap --script stuxnet-detect -p445 -Pn ",""],

["davtest -url http://",""],

["golismero -e fingerprint\_web scan ",""],

["uniscan -w -u ",""],

["uniscan -q -u ",""],

["uniscan -r -u ",""],

["uniscan -s -u ",""],

["uniscan -d -u ",""],

["nikto -Plugins 'apache\_expect\_xss' -host ",""],

["nikto -Plugins 'subdomain' -host ",""],

["nikto -Plugins 'shellshock' -host ",""],

["nikto -Plugins 'cookies' -host ",""],

["nikto -Plugins 'put\_del\_test' -host ",""],

["nikto -Plugins 'headers' -host ",""],

["nikto -Plugins 'ms10-070' -host ",""],

["nikto -Plugins 'msgs' -host ",""],

["nikto -Plugins 'outdated' -host ",""],

["nikto -Plugins 'httpoptions' -host ",""],

["nikto -Plugins 'cgi' -host ",""],

["nikto -Plugins 'ssl' -host ",""],

["nikto -Plugins 'sitefiles' -host ",""],

["nikto -Plugins 'paths' -host ",""],

["dnsmap ",""],

["nmap -p1433 --open -Pn ",""],

["nmap -p3306 --open -Pn ",""],

["nmap -p1521 --open -Pn ",""],

["nmap -p3389 --open -sU -Pn ",""],

["nmap -p3389 --open -sT -Pn ",""],

["nmap -p1-65535 --open -Pn ",""],

["nmap -p1-65535 -sU --open -Pn ",""],

["nmap -p161 -sU --open -Pn ",""],

["wget -O temp\_aspnet\_elmah\_axd --tries=1 ","/elmah.axd"],

["nmap -p445,137-139 --open -Pn ",""],

["nmap -p137,138 --open -Pn ",""], ["wapiti "," -f txt -o temp\_wapiti"],

["nmap -p80 --script=http-iis-webdav-vuln -Pn ",""],

["whatweb "," -a 1"]

]

# Tool Responses (Begins) [Responses + Severity (c - critical | h - high | m - medium | l - low | i - informational) + Reference for Vuln Definition and Remediation] tool\_resp = [

["Does not have an IPv6 Address. It is good to have one.", "i", 1],

["WordPress Installation Found. Check for vulnerabilities corresponds to that version.","i",2],

["robots.txt/sitemap.xml found. Check those files for any information.","i",3],

["No Web Application Firewall Detected","m",4],

["Some ports are open. Perform a full-scan manually.","l",5],

["XSS Protection Filter is Disabled.","m",6],

["Vulnerable to Slowloris Denial of Service.","c",7],

["HEARTBLEED Vulnerability Found with Nmap.","h",8],

["POODLE Vulnerability Detected.","h",9],

["OpenSSL CCS Injection Detected.","h",10],

["FREAK Vulnerability Detected.","h",11],

["LOGJAM Vulnerability Detected.","h",12],

["Telnet Service Detected.","h",13],

["FTP Service Detected.","c",14],

["Vulnerable to STUXNET.","c",15],

["WebDAV Enabled.","m",16],

["Found some information through Fingerprinting.","l",17],

["Open Files Found with Uniscan.","m",18],

["Open Directories Found with Uniscan.","m",19],

["Vulnerable to Stress Tests.","h",20],

["Uniscan detected possible LFI, RFI or RCE.","h",21],

["Uniscan detected possible XSS, SQLi, BSQLi.","h",22],

["Apache Expect XSS Header not present.","m",6],

["Found Subdomains with Nikto.","m",23],

["Webserver vulnerable to Shellshock Bug.","c",24], #40

["Webserver leaks Internal IP.","l",25], #41

["HTTP PUT DEL Methods Enabled.","m",26], #43 ["Some vulnerable headers exposed.","m",27], #43

["Webserver vulnerable to MS10-070.","h",28], #44

["Some issues found on the Webserver.","m",36], #

["Webserver is Outdated.","h",29], #45

["Some issues found with HTTP Options.","l",26],

["CGI Directories Enumerated.","l",19], #26

["Vulnerabilities reported in SSL Scans.","m",37], #26

["Interesting Files Detected.","m",18], #25

["Injectable Paths Detected.","l",30], #46

["Found Subdomains with DNSMap.","m",23],

["MS-SQL DB Service Detected.","l",31], #47

["MySQL DB Service Detected.","l",31], #47

["ORACLE DB Service Detected.","l",31], #47

["RDP Server Detected over UDP.","h",32], #48

["RDP Server Detected over TCP.","h",32], #48

["TCP Ports are Open","l",5], #8

["UDP Ports are Open","l",5], #8

["SNMP Service Detected.","m",33], #49

["Elmah is Configured.","m",34], #50

["SMB Ports are Open over TCP","m",35], #51

["SMB Ports are Open over UDP","m",35], #51

["Wapiti discovered a range of vulnerabilities","h",36],#

["IIS WebDAV is Enabled","m",10], #16

["X-XSS Protection is not Present","m",6] #12

]

# Tool Responses (Ends)

# Tool Status (Response Data + Response Code (if status check fails and you still got to push it +

Legends + Approx Time + Tool Identification + Bad Responses) tool\_status = [

["has IPv6", 1, proc\_low, " < 15s", "ipv6", ["not found", "has IPv6"]], ["wp-login",0,proc\_low," < 30s","wpcheck",["unable to resolve host address","Connection timed out"]],

["[+]",0,proc\_low," < 40s","robotscheck",["Use of uninitialized value in unpack at"]],

["No WAF",0,proc\_low," < 45s","wafcheck",["appears to be down"]],

["tcp open",0,proc\_med," < 2m","nmapopen",["Failed to resolve"]],

["XSS filter is disabled",0,proc\_low," < 20s","nmapxssh",["Failed to resolve"]],

["VULNERABLE",0,proc\_high," < 45m","nmapdos",["Failed to resolve"]],

["VULNERABLE",0,proc\_low," < 30s","nmap1",["Failed to resolve"]],

["VULNERABLE",0,proc\_low," < 35s","nmap2",["Failed to resolve"]],

["VULNERABLE",0,proc\_low," < 35s","nmap3",["Failed to resolve"]],

["VULNERABLE",0,proc\_low," < 30s","nmap4",["Failed to resolve"]],

["VULNERABLE",0,proc\_low," < 35s","nmap5",["Failed to resolve"]],

["open",0,proc\_low," < 15s","nmaptelnet",["Failed to resolve"]],

["open",0,proc\_low," < 15s","nmapftp",["Failed to resolve"]],

["open",0,proc\_low," < 20s","nmapstux",["Failed to resolve"]],

["SUCCEED",0,proc\_low," < 30s","webdav",["is not DAV enabled or not accessible."]],

["No vulnerabilities found",1,proc\_low," < 15s","golism10",["Cannot resolve domain name","No vulnerabilities found"]],

["[+]",0,proc\_med," < 2m","uniscan2",["Use of uninitialized value in unpack at"]],

["[+]",0,proc\_med," < 5m","uniscan3",["Use of uninitialized value in unpack at"]],

["[+]",0,proc\_med," < 9m","uniscan4",["Use of uninitialized value in unpack at"]],

["[+]",0,proc\_med," < 8m","uniscan5",["Use of uninitialized value in unpack at"]],

["[+]",0,proc\_med," < 9m","uniscan6",["Use of uninitialized value in unpack at"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto1",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto2",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto3",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto4",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto5",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto6",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto7",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto8",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto9",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto10",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto11",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto12",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto13",["ERROR: Cannot resolve hostname","0 item(s) reported","No web server found","0 host(s) tested"]],

["0 item(s) reported",1,proc\_low," < 35s","nikto14","ERROR: Cannot resolve hostname , 0 item(s) reported"],

["#1",0,proc\_high," < 30m","dnsmap\_brute",["[+] 0 (sub)domains and 0 IP address(es) found"]],

["open",0,proc\_low," < 15s","nmapmssql",["Failed to resolve"]],

["open",0,proc\_low," < 15s","nmapmysql",["Failed to resolve"]],

["open",0,proc\_low," < 15s","nmaporacle",["Failed to resolve"]],

["open",0,proc\_low," < 15s","nmapudprdp",["Failed to resolve"]],

["open",0,proc\_low," < 15s","nmaptcprdp",["Failed to resolve"]],

["open",0,proc\_high," > 50m","nmapfulltcp",["Failed to resolve"]],

["open",0,proc\_high," > 75m","nmapfulludp",["Failed to resolve"]],

["open",0,proc\_low," < 30s","nmapsnmp",["Failed to resolve"]],

["Microsoft SQL Server Error Log",0,proc\_low," < 30s","elmahxd",["unable to resolve host address","Connection timed out"]],

["open",0,proc\_low," < 20s","nmaptcpsmb",["Failed to resolve"]],

["open",0,proc\_low," < 20s","nmapudpsmb",["Failed to resolve"]],

["Host:",0,proc\_med," < 5m","wapiti",["none"]],

["WebDAV is ENABLED",0,proc\_low," < 40s","nmapwebdaviis",["Failed to resolve"]],

["X-XSS-Protection[1",1,proc\_med," < 3m","whatweb",["Timed out","Socket error","X-XSSProtection[1"]]

]

# Vulnerabilities and Remediation tools\_fix = [

[1, "Not a vulnerability, just an informational alert. The host does not have IPv6 support. IPv6 provides more security as IPSec (responsible for CIA - Confidentiality, Integrity and Availablity) is incorporated into this model. So it is good to have IPv6 Support.",

"It is recommended to implement IPv6. More information on how to implement IPv6 can be found from this resource. https://www.cisco.com/c/en/us/solutions/collateral/enterprise/cisco-on-cisco/IPv6Implementation\_CS.html"],

[2, "It is not bad to have a CMS in WordPress. There are chances that the version may contain vulnerabilities or any third party scripts associated with it may possess vulnerabilities","It is recommended to conceal the version of WordPress. This resource contains more information on how to secure your WordPress Blog. https://codex.wordpress.org/Hardening\_WordPress"],

[3, "Sometimes robots.txt or sitemap.xml may contain rules such that certain links that are not supposed to be accessed/indexed by crawlers and search engines. Search engines may skip those links but attackers will be able to access it directly.", "It is a good practice not to include sensitive links in the robots or sitemap files."],

[4, "Without a Web Application Firewall, An attacker may try to inject various attack patterns either manually or using automated scanners. An automated scanner may send hordes of attack vectors and patterns to validate an attack, there are also chances for the application to get DoS`ed (Denial of Service)", "Web Application Firewalls offer great protection against common web attacks like XSS, SQLi, etc. They also provide an additional line of defense to your security infrastructure. This resource contains information on web application firewalls that could suit your application. https://www.gartner.com/reviews/market/web-application-firewall"],

[5, "Open Ports give attackers a hint to exploit the services. Attackers try to retrieve banner information through the ports and understand what type of service the host is running","It is recommended to close the ports of unused services and use a firewall to filter the ports wherever necessary. This resource may give more insights. https://security.stackexchange.com/a/145781/6137"],

[6, "As the target is lacking this header, older browsers will be prone to Reflected XSS attacks.", "Modern browsers does not face any issues with this vulnerability (missing headers). However, older browsers are strongly recommended to be upgraded."],

[7, "This attack works by opening multiple simultaneous connections to the web server and it keeps them alive as long as possible by continously sending partial HTTP requests, which never gets completed. They easily slip through IDS by sending partial requests.", "If you are using Apache Module, `mod\_antiloris` would help. For other setup you can find more detailed remediation on this resource. https://www.acunetix.com/blog/articles/slow-http-dos-attacks-mitigate-apache-http-server/"],

[8, "This vulnerability seriously leaks private information of your host. An attacker can keep the TLS connection alive and can retrieve a maximum of 64K of data per heartbeat.", "PFS (Perfect Forward Secrecy) can be implemented to make decryption difficult. Complete remediation and resource information is available here. http://heartbleed.com/"],

[9, "By exploiting this vulnerability, an attacker will be able gain access to sensitive data in a n encrypted session such as session ids, cookies and with those data obtained, will be able to impersonate that particular user.", "This is a flaw in the SSL 3.0 Protocol. A better remediation would be to disable using the SSL 3.0 protocol. For more information, check this resource. https://www.uscert.gov/ncas/alerts/TA14-290A"],

[10, "This attacks takes place in the SSL Negotiation (Handshake) which makes the client unaware of the attack. By successfully altering the handshake, the attacker will be able to pry on all the information that is sent from the client to server and vice-versa","Upgrading OpenSSL to latest versions will mitigate this issue. This resource gives more information about the vulnerability and the associated remediation. http://ccsinjection.lepidum.co.jp/"],

[11, "With this vulnerability the attacker will be able to perform a MiTM attack and thus compromising the confidentiality factor.", "Upgrading OpenSSL to latest version will mitigate this issue.

Versions prior to 1.1.0 is prone to this vulnerability. More information can be found in this resource. https://bobcares.com/blog/how-to-fix-sweet32-birthday-attacks-vulnerability-cve-2016-2183/"],

[12, "With the LogJam attack, the attacker will be able to downgrade the TLS connection which allows the attacker to read and modify any data passed over the connection.", "Make sure any TLS libraries you use are up-to-date, that servers you maintain use 2048-bit or larger primes, and that clients you maintain reject Diffie-Hellman primes smaller than 1024-bit. More information can be found in this resource. https://weakdh.org/"],

[13, "Through this deprecated protocol, an attacker may be able to perform MiTM and other complicated attacks.", "It is highly recommended to stop using this service and it is far outdated. SSH can be used to replace TELNET. For more information, check this resource https://www.ssh.com/ssh/telnet"],

[14, "This protocol does not support secure communication and there are likely high chances for the attacker to eavesdrop the communication. Also, many FTP programs have exploits available in the web such that an attacker can directly crash the application or either get a SHELL access to that target.", "Proper suggested fix is use an SSH protocol instead of FTP. It supports secure communication and chances for MiTM attacks are quite rare."],

[15, "The StuxNet is level-3 worm that exposes critical information of the target organization. It was a cyber weapon that was designed to thwart the nuclear intelligence of Iran. Seriously wonder how it got here? Hope this isn't a false positive Nmap ;)", "It is highly recommended to perform a complete rootkit scan on the host. For more information refer to this resource. https://www.symantec.com/security\_response/writeup.jsp?docid=2010-071400-3123-99&tabid=3"],

[16, "WebDAV is supposed to contain multiple vulnerabilities. In some case, an attacker may hide a malicious DLL file in the WebDAV share however, and upon convincing the user to open a perfectly harmless and legitimate file, execute code under the context of that user","It is recommended to disable WebDAV. Some critical resource regarding disbling WebDAV can be found on this URL. https://www.networkworld.com/article/2202909/network-security/-webdav-is-bad---says-securityresearcher.html"],

[17, "Attackers always do a fingerprint of any server before they launch an attack. Fingerprinting gives them information about the server type, content- they are serving, last modification times etc, this gives an attacker to learn more information about the target","A good practice is to obfuscate the information to outside world. Doing so, the attackers will have tough time understanding the server's tech stack and therefore leverage an attack."],

[18, "Attackers may find considerable amount of information from these files. There are even chances attackers may get access to critical information from these files.", "It is recommended to block or restrict access to these files unless necessary."],

[19, "Attackers may find considerable amount of information from these directories. There are even chances attackers may get access to critical information from these directories.", "It is recommended to block or restrict access to these directories unless necessary."],

[20, "Attackers mostly try to render web applications or service useless by flooding the target, such that blocking access to legitimate users. This may affect the business of a company or organization as well as the reputation","By ensuring proper load balancers in place, configuring rate limits and multiple connection restrictions, such attacks can be drastically mitigated."],

[21, "Intruders will be able to remotely include shell files and will be able to access the core file system or they will be able to read all the files as well. There are even higher chances for the attacker to remote execute code on the file system.", "Secure code practices will mostly prevent LFI, RFI and RCE attacks. The following resource gives a detailed insight on secure coding practices. https://wiki.sei.cmu.edu/confluence/display/seccode/Top+10+Secure+Coding+Practices"],

[22, "Hackers will be able to steal data from the backend and also they can authenticate themselves to the website and can impersonate as any user since they have total control over the backend. They can even wipe out the entire database. Attackers can also steal cookie information of an authenticated user and they can even redirect the target to any malicious address or totally deface the application.", "Proper input validation has to be done prior to directly querying the database information. A developer should remember not to trust an end-user's input. By following a secure coding methodology attacks like SQLi, XSS and BSQLi. The following resource guides on how to implement secure coding methodology on application development. https://wiki.sei.cmu.edu/confluence/display/seccode/Top+10+Secure+Coding+Practices"],

[23, "Attackers may gather more information from subdomains relating to the parent domain.

Attackers may even find other services from the subdomains and try to learn the architecture of the target. There are even chances for the attacker to find vulnerabilities as the attack surface gets larger with more subdomains discovered.", "It is sometimes wise to block sub domains like development, staging to the outside world, as it gives more information to the attacker about the tech stack. Complex naming practices also help in reducing the attack surface as attackers find hard to perform subdomain bruteforcing through dictionaries and wordlists."],

[24, "Attackers exploit the vulnerability in BASH to perform remote code execution on the target. An experienced attacker can easily take over the target system and access the internal sources of the machine","This vulnerability can be mitigated by patching the version of BASH. The following resource gives an indepth analysis of the vulnerability and how to mitigate it.

https://www.symantec.com/connect/blogs/shellshock-all-you-need-know-about-bash-bug-vulnerability https://www.digitalocean.com/community/tutorials/how-to-protect-your-server-against-the-shellshockbash-vulnerability"],

[25, "Gives attacker an idea on how the address scheming is done internally on the organizational network. Discovering the private addresses used within an organization can help attackers in carrying out network-layer attacks aiming to penetrate the organization's internal infrastructure.", "Restrict the banner information to the outside world from the disclosing service. More information on mitigating this vulnerability can be found here. https://portswigger.net/kb/issues/00600300\_private-ip-addressesdisclosed"],

[26, "There are chances for an attacker to manipulate files on the webserver.", "It is recommended to disable the HTTP PUT and DEL methods incase if you don't use any REST API Services. Following resources helps you how to disable these methods. http://www.techstacks.com/howto/disable-http-methods-in-tomcat.html https://docs.oracle.com/cd/E19857-01/820-5627/gghwc/index.html

https://developer.ibm.com/answers/questions/321629/how-to-disable-http-methods-head-put-deleteoption/"],

[27, "Attackers try to learn more about the target from the amount of information exposed in the headers. An attacker may know what type of tech stack a web application is emphasizing and many other information.", "Banner Grabbing should be restricted and access to the services from outside would should be made minimum."],

[28, "An attacker who successfully exploited this vulnerability could read data, such as the view state, which was encrypted by the server. This vulnerability can also be used for data tampering, which, if successfully exploited, could be used to decrypt and tamper with the data encrypted by the server.", "Microsoft has released a set of patches on their website to mitigate this issue. The information required to fix this vulnerability can be inferred from this resource. https://docs.microsoft.com/en-us/securityupdates/securitybulletins/2010/ms10-070"],

[29, "Any outdated web server may contain multiple vulnerabilities as their support would've been ended. An attacker may make use of such an opportunity to leverage attacks.", "It is highly recommended to upgrade the web server to the available latest version."],

[30, "Hackers will be able to manipulate the URLs easily through a GET/POST request. They will be able to inject multiple attack vectors in the URL with ease and able to monitor the response as well","By ensuring proper sanitization techniques and employing secure coding practices it will be impossible for the attacker to penetrate through. The following resource gives a detailed insight on secure coding practices. https://wiki.sei.cmu.edu/confluence/display/seccode/Top+10+Secure+Coding+Practices"],

[31, "Since the attacker has knowledge about the particular type of backend the target is running, they will be able to launch a targetted exploit for the particular version. They may also try to authenticate with default credentials to get themselves through.", "Timely security patches for the backend has to be installed. Default credentials has to be changed. If possible, the banner information can be changed to mislead the attacker. The following resource gives more information on how to secure your backend. http://kb.bodhost.com/secure-database-server/"],

[32, "Attackers may launch remote exploits to either crash the service or tools like ncrack to try brute-forcing the password on the target.", "It is recommended to block the service to outside world and made the service accessible only through the a set of allowed IPs only really neccessary. The following resource provides insights on the risks and as well as the steps to block the service. https://www.perspectiverisk.com/remote-desktop-service-vulnerabilities/"],

[33, "Hackers will be able to read community strings through the service and enumerate quite an information from the target. Also, there are multiple Remote Code Execution and Denial of Service vulnerabilities related to SNMP services.", "Use a firewall to block the ports from the outside world. The following article gives wide insight on locking down SNMP service. https://www.techrepublic.com/article/lock-it-down-dont-allow-snmp-to-compromise-network-security/"],

[34, "Attackers will be able to find the logs and error information generated by the application. They will also be able to see the status codes that was generated on the application. By combining all these information, the attacker will be able to leverage an attack.", "By restricting access to the logger application from the outside world will be more than enough to mitigate this weakness."],

[35, "Cyber Criminals mainly target this service as it is very easier for them to perform a remote attack by running exploits. WannaCry Ransomware is one such example.", "Exposing SMB Service to the outside world is a bad idea, it is recommended to install latest patches for the service in order not to get compromised. The following resource provides a detailed information on SMB Hardening concepts. https://kb.iweb.com/hc/en-us/articles/115000274491-Securing-Windows-SMB-and-NetBios-NetBTServices"],

[36, "Particular Scanner found multiple vulnerabilities that an attacker may try to exploit the target.", "Refer to RS-Vulnerability-Report to view the complete information of the vulnerability, once the scan gets completed."],

[37, "SSL related vulnerabilities breaks the confidentiality factor. An attacker may perform a MiTM attack, intrepret and eavesdrop the communication.", "Proper implementation and upgraded version of SSL and TLS libraries are very critical when it comes to blocking SSL related vulnerabilities."],

]

# Tool Set tools\_precheck = [ ["wapiti"], ["whatweb"], ["nmap"], ["golismero"], ["host"], ["wget"], ["uniscan"], ["wafw00f"], ["dirb"], ["davtest"], ["theharvester"], ["xsser"], [

"dnsrecon"], ["fierce"], ["dnswalk"], ["whois"], ["sslyze"], ["lbd"], ["golismero"], ["dnsenum"], ["dmitry"], ["davtest"], ["nikto"], ["dnsmap"]

]

# Shuffling Scan Order (starts)

scan\_shuffle = list(zip(tool\_names, tool\_cmd, tool\_resp, tool\_status)) random.shuffle(scan\_shuffle)

tool\_names, tool\_cmd, tool\_resp, tool\_status = list(zip(\*scan\_shuffle)) # Cross verification incase, breaks.

tool\_checks = (len(tool\_names) + len(tool\_resp) + len(tool\_status)) / 3

# Shuffling Scan Order (ends)

# Tool Head Pointer: (can be increased but certain tools will be skipped) tool = 0 # Run Test runTest = 1

# For accessing list/dictionary elements arg1 = 0,arg2 = 1,arg3 = 2,arg4 = 3,arg5 = 4,arg6 = 5 # Detected Vulnerabilities [will be dynamically populated] sa\_vul\_list = list() sa\_vul\_num = 0 sa\_vul = 0

# Total Time Elapsed sa\_total\_elapsed = 0 # Tool Pre Checker sa\_avail\_tools = 0 # Checks Skipped sa\_skipped\_checks = 0 if len(sys.argv) == 1:

logo() helper() else:

target = sys.argv[1].lower() if target == '--update' or target == '-u' or target == '--u': logo()

print("Scanner is updating....Please wait.\n") spinner.start()

# Checking internet connectivity first... sa\_internet\_availability = check\_internet() if sa\_internet\_availability == 0:

print("\t" + bcolors.BG\_ERR\_TXT + \

"There seems to be some problem connecting to the internet. Please try again or later." + bcolors.ENDC)

spinner.stop() sys.exit(1)

cmd = 'sha1sum scanner.py | grep .... | cut -c 1-40' oldversion\_hash = subprocess.check\_output(cmd, shell=True) oldversion\_hash = oldversion\_hash.strip() os.system(

'wget -N https://github.com/Malwareman007/Scanner-and-

Patcher/tree/main/Full%20Scanner/Web\_scan.py -O Web\_scan.py > /dev/null 2>&1') newversion\_hash = subprocess.check\_output(cmd, shell=True) newversion\_hash = newversion\_hash.strip() if oldversion\_hash == newversion\_hash: clear()

print("\t" + bcolors.OKBLUE + \

"You already have the latest version of Scanner." + bcolors.ENDC) else:

clear()

print("\t" + bcolors.OKGREEN + \

"Scanner successfully updated to the latest version." + bcolors.ENDC) spinner.stop() sys.exit(1) elif target == '--help' or target == '-h' or target == '--h':

logo() helper() sys.exit(1) else:

target = url\_maker(target)

os.system('rm te\* > /dev/null 2>&1') # Clearing previous scan files os.system('clear')

os.system('setterm -cursor off') logo()

print(bcolors.BG\_HEAD\_TXT + \

"[ Checking Available Security Scanning Tools Phase... Initiated. ]"+bcolors.ENDC) unavail\_tools = 0 unavail\_tools\_names = list() while (sa\_avail\_tools < len(tools\_precheck)):

precmd = str(tools\_precheck[sa\_avail\_tools][arg1]) try:

p = subprocess.Popen([precmd], stdin=subprocess.PIPE,

stdout=subprocess.PIPE, stderr=subprocess.PIPE, shell=True) output, err = p.communicate() val = output + err except:

print("\t"+bcolors.BG\_ERR\_TXT+"Scanner was terminated abruptly..."+bcolors.ENDC) sys.exit(1) if "not found" in str(val):

print("\t"+bcolors.OKBLUE + \

tools\_precheck[sa\_avail\_tools][arg1]+bcolors.ENDC + \ bcolors.BADFAIL+"...unavailable."+bcolors.ENDC) for scanner\_index, scanner\_val in enumerate(tool\_names): if scanner\_val[2] == tools\_precheck[sa\_avail\_tools][arg1]:

# disabling scanner as it's not available.

scanner\_val[3] = 0 unavail\_tools\_names.append( tools\_precheck[sa\_avail\_tools][arg1]) unavail\_tools = unavail\_tools + 1 else:

print("\t"+bcolors.OKBLUE + \

tools\_precheck[sa\_avail\_tools][arg1]+bcolors.ENDC + \ bcolors.OKGREEN+"...available."+bcolors.ENDC) sa\_avail\_tools = sa\_avail\_tools + 1 clear()

unavail\_tools\_names = list(set(unavail\_tools\_names)) if unavail\_tools == 0:

print("\t"+bcolors.OKGREEN + \

"All Scanning Tools are available. All vulnerability checks will be performed by Scanner."+bcolors.ENDC) else:

print("\t"+bcolors.WARNING+"Some of these tools "+bcolors.BADFAIL + \ str(unavail\_tools\_names)+bcolors.ENDC+bcolors.WARNING + \

" are unavailable. Scanner can still perform tests by excluding these tools from the tests. Please install these tools to fully utilize the functionality of Scanner."+bcolors.ENDC) print(bcolors.BG\_ENDL\_TXT + \

"[ Checking Available Security Scanning Tools Phase... Completed. ]"+bcolors.ENDC) print("\n")

print(bcolors.BG\_HEAD\_TXT+"[ Preliminary Scan Phase Initiated... Loaded "+str( tool\_checks)+" vulnerability checks. ]"+bcolors.ENDC) # while (tool < 1): while(tool < len(tool\_names)):

print("["+tool\_status[tool][arg3]+tool\_status[tool][arg4]+"] Deploying "+str(

tool+1)+"/"+str(tool\_checks)+" | "+bcolors.OKBLUE+tool\_names[tool][arg2]+bcolors.ENDC, end=' ') if tool\_names[tool][arg4] == 0:

print(bcolors.WARNING+"...Scanning Tool Unavailable. Auto-Skipping Test..."+bcolors.ENDC)

sa\_skipped\_checks = sa\_skipped\_checks + 1 tool = tool + 1 continue spinner.start() scan\_start = time.time()

temp\_file = "temp\_"+tool\_names[tool][arg1] cmd = tool\_cmd[tool][arg1]+target + \ tool\_cmd[tool][arg2]+" > "+temp\_file+" 2>&1" try:

subprocess.check\_output(cmd, shell=True) except KeyboardInterrupt:

runTest = 0 except:

runTest = 1 if runTest == 1: spinner.stop() scan\_stop = time.time() elapsed = scan\_stop - scan\_start sa\_total\_elapsed = sa\_total\_elapsed + elapsed print(bcolors.OKBLUE+"\b...Completed in " + \ display\_time(int(elapsed))+bcolors.ENDC+"\n") clear()

sa\_tool\_output\_file = open(temp\_file).read() if tool\_status[tool][arg2] == 0: if tool\_status[tool][arg1].lower() in sa\_tool\_output\_file.lower():

#print "\t"+ vul\_info(tool\_resp[tool][arg2]) + bcolors.BADFAIL +" "+ tool\_resp[tool][arg1] + bcolors.ENDC

vul\_remed\_info(

tool, tool\_resp[tool][arg2], tool\_resp[tool][arg3]) sa\_vul\_list.append(

tool\_names[tool][arg1]+"\*"+tool\_names[tool][arg2]) else:

if any(i in sa\_tool\_output\_file for i in tool\_status[tool][arg6]):

m = 1 # This does nothing. else:

#print "\t"+ vul\_info(tool\_resp[tool][arg2]) + bcolors.BADFAIL +" "+ tool\_resp[tool][arg1] + bcolors.ENDC

vul\_remed\_info(

tool, tool\_resp[tool][arg2], tool\_resp[tool][arg3]) sa\_vul\_list.append(

tool\_names[tool][arg1]+"\*"+tool\_names[tool][arg2]) else:

runTest = 1

spinner.stop() scan\_stop = time.time() elapsed = scan\_stop - scan\_start

sa\_total\_elapsed = sa\_total\_elapsed + elapsed print(bcolors.OKBLUE+"\b\b\b\b...Interrupted in " + \ display\_time(int(elapsed))+bcolors.ENDC+"\n") clear()

print("\t"+bcolors.WARNING + \

"Test Skipped. Performing Next. Press Ctrl+Z to Quit Scanner." + bcolors.ENDC) sa\_skipped\_checks = sa\_skipped\_checks + 1 tool = tool+1

print(bcolors.BG\_ENDL\_TXT + \

"[ Preliminary Scan Phase Completed. ]"+bcolors.ENDC) print("\n")

#################### Report & Documentation Phase ########################### print(bcolors.BG\_HEAD\_TXT + \

"[ Report Generation Phase Initiated. ]"+bcolors.ENDC) if len(sa\_vul\_list) == 0:

print("\t"+bcolors.OKGREEN+"No Vulnerabilities Detected."+bcolors.ENDC) else: with open("SA-Vulnerability-Report", "a") as report: while(sa\_vul < len(sa\_vul\_list)): vuln\_info = sa\_vul\_list[sa\_vul].split('\*') report.write(vuln\_info[arg2])

report.write("\n------------------------\n\n") temp\_report\_name = "temp\_"+vuln\_info[arg1] with open(temp\_report\_name, 'r') as temp\_report:

data = temp\_report.read() report.write(data) report.write("\n\n") temp\_report.close() sa\_vul = sa\_vul + 1

print("\tComplete Vulnerability Report for "+bcolors.OKBLUE+target+bcolors.ENDC+" named "+bcolors.OKGREEN + \

"`SA-Vulnerability-Report`"+bcolors.ENDC + \ " is available under the same directory Scanner resides.") report.close()

# Writing all scan files output into SA-Debug-ScanLog for debugging purposes. for file\_index, file\_name in enumerate(tool\_names): with open("SA-Debug-ScanLog", "a") as report: try: with open("temp\_"+file\_name[arg1], 'r') as temp\_report:

data = temp\_report.read() report.write(file\_name[arg2])

report.write("\n------------------------\n\n") report.write(data) report.write("\n\n") temp\_report.close() except: break report.close()

print("\tTotal Number of Vulnerability Checks : " + \

bcolors.BOLD+bcolors.OKGREEN+str(len(tool\_names))+bcolors.ENDC) print("\tTotal Number of Vulnerability Checks Skipped: " + bcolors.BOLD+bcolors.WARNING+str(sa\_skipped\_checks)+bcolors.ENDC)

print("\tTotal Number of Vulnerabilities Detected : " + \

bcolors.BOLD+bcolors.BADFAIL+str(len(sa\_vul\_list))+bcolors.ENDC) print("\tTotal Time Elapsed for the Scan : "+bcolors.BOLD + \ bcolors.OKBLUE+display\_time(int(sa\_total\_elapsed))+bcolors.ENDC) print("\n")

print("\tFor Debugging Purposes, You can view the complete output generated by all the tools named " + \

bcolors.OKBLUE+"`SA-Debug-ScanLog`"+bcolors.ENDC+" under the same directory.") print(bcolors.BG\_ENDL\_TXT + \

"[ Report Generation Phase Completed. ]"+bcolors.ENDC) os.system('setterm -cursor on')

os.system('rm te\* > /dev/null 2>&1') # Clearing previous scan files