

SpiderSuite Scanner

Introduction To SpiderSuite

Product Overview

SpiderSuite is an Advance Multi-feature web Crawler/Spider for Cyber Security professionals. It contains a powerful crawler which can crawl even the most sophisticated web pages and produce a readable output and an intuitive user interface to interact with the results.

SpiderSuite is a tool that can be easily utilized by web application developers, penetration testers, bug bounty hunters and cyber security researchers to map a target website and inspect each individual page and assets.

SpiderSuite contains a suite of tools aimed at easing the recon phase of web penetration testing and gives a detailed overview of the attack surface of a web application.

Requirements

SpiderSuite runs on 64 bit machines only, it does not support x32 system and currently available for Windows and Linux operating systems.

Installation

Installing SpiderSuite involves a series of simple steps which must be completed in the correct sequence for the installation to be successful.

To Download

The installers and portable executable files are available on the SpiderSuite's Github repository release page: <https://github.com/3nock/SpiderSuite/releases>

SpiderSuite currently supports **Windows** and **Linux** operating systems with x64 architecture. On the release page there are two types of download packages for each of the two systems i.e. **installers** and **portable executable**.

Installers will install SpiderSuite and its dependencies in default chosen location on your machine. The installers are ***SpiderSuite_vX.X.X_win64_installer.exe*** for windows and ***SpiderSuite_vX.X.X_linux_installer.run*** for linux.

Portable executables do not need any installation; you simply download SpiderSuite and use it. The portable executables are ***SpiderSuite_vX.X.X_win64.zip*** for Windows and ***SpiderSuite_vX.X.X_linux.ApplImage*** and ***SpiderSuite_vX.X.X_linux.tar.gz*** for Linux

To Install

Please Note: *SpiderSuite is a Graphical User Interface application so all of the steps in this guide refer to the SpiderSuite GUI.*

- **For Windows Portable Executables:**

Download and extract ***SpiderSuite_vX.X.X_win64.zip*** archive and place it at your chosen location, extract the archive then run ***SpiderSuite.exe*** simply by double clicking on the program.

- **For Linux Portable Executable:**

Download and extract ***SpiderSuite_vX.X.X_linux.ApplImage*** archive and place it at your chosen location, then run it simply by double clicking or using command line with the command ***./SpiderSuite_vX.X.X_linux.ApplImage***.

OR

Download and extract ***SpiderSuite_vX.X.X_linux.tar.gz*** archive and place it at your chosen location, extract the archive on desired location (you can use GUI option or command line: ***tar -xzf SpiderSuite_vX.X.X_linux.tar.gz***, then run it simply by double

clicking on the **SpiderSuite/AppRun.Wrapper** or using command line with the command **./ SpiderSuite / AppRun.Wrapper**.

- **For Windows Installer**

Download **SpiderSuite_vX.X.X_win64_installer.exe** then run the installer, then fill in the required information such as installation location and shortcut names precedually until you finish the installation procedure.

- **For Linux Installer:**

Download **SpiderSuite_vX.X.X_linux_installer.run** then run the installer using the command line **./SpiderSuite_vX.X.X_linux_installer.run**, then fill in the required information such as installation location and shortcut names precedually until you finish the installation procedure.

NOTE:

In most windows environment SpiderSuite will run on the first try but case of SipderSuite fails to run on the first try on windows, this could be an indication that your machine does not contain the required MSVC-redistributable package.

Don't worry, SpiderSuite comes packaged with the required MSVC-redistributable package in case of this.

*Simply Install the MSVC-redistributable package which comes with SpiderSuite (**SpiderSuite/vcredist_x64.exe**).*

*Also In case SpiderSuite fails to connect to the internet and shows SSL errors, install the **OpenSSL package** which comes packaged with SpiderSuite (**SpiderSuite/Win64 OpenSSL v1.1.1n Light.msi**).*

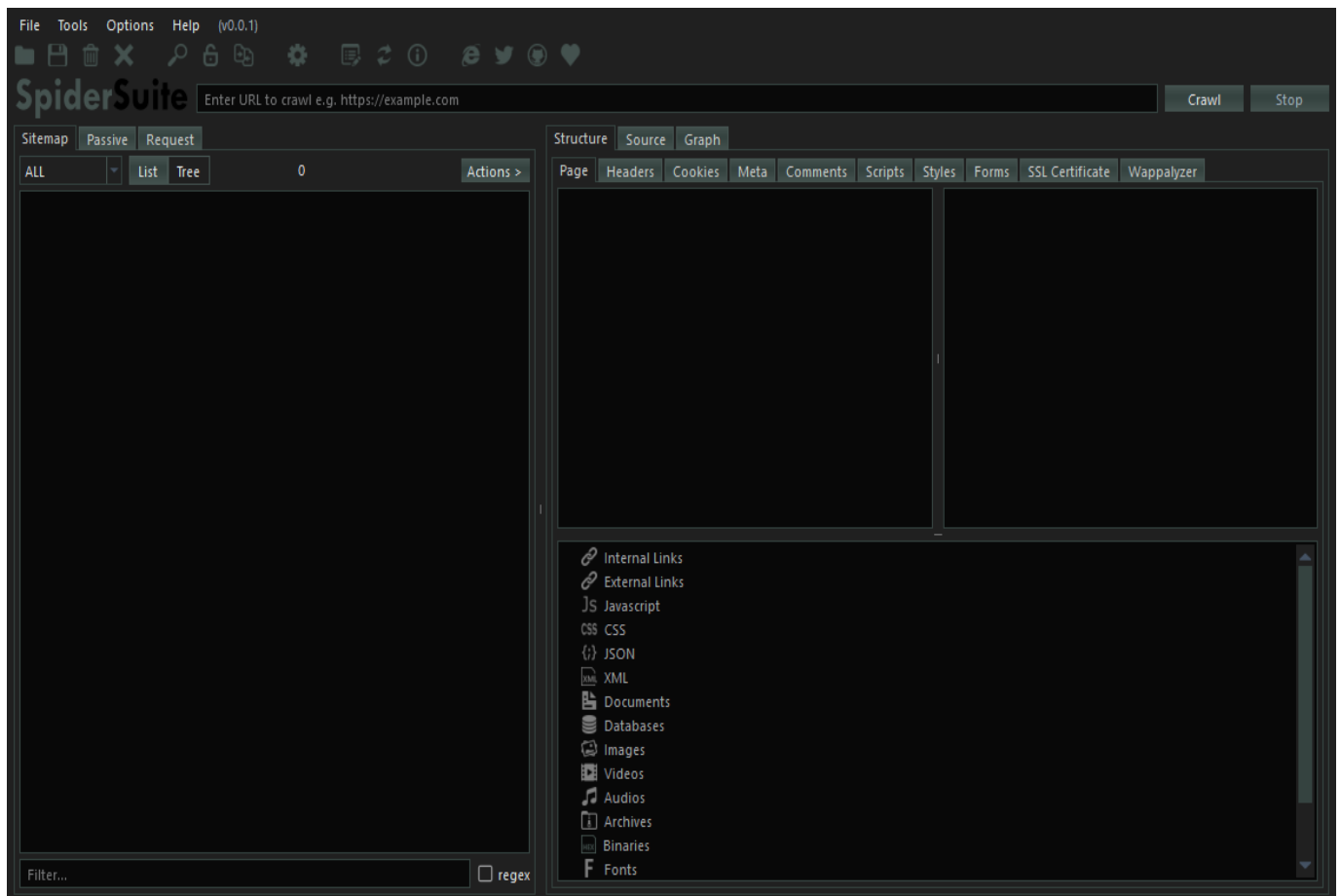
To Uninstall SpiderSuite

To uninstall SpiderSuite for portable SpiderSuite, just delete the SpiderSuite folder and you are done. For installed SpiderSuite run the uninstaller located in the SpiderSuite installation directory (SpiderSuite/maintanance_tool.exe).

SpiderSuite User Interface

Main Window

SpiderSuite's Main window contains the following information.



Menu Bar Items

File

- **Open** - opens a project from the file system.
- **Open Recent** - shows and opens recent SpiderSuite projects you've been working on
- **Import From** - Imports links and their data from other programs and file types such as;
links from Zed Attack Proxy(ZAP)(.xml files), data from Fiddler(.saz files), links from acunetix(.xml files), data exported from BurpSuite(.xml files), data from HTTP Archives(.har files), links from version controls (.git/index files & .svn/entries files) and many more will be added in the coming versions.
- **Save** - Saves the current project you're working on without closing it.

- **Clear** – Clears and deletes all the data of the current loaded project.
- **Close** - closes the current loaded project without deleting any data from that particular project.
- **Exit** - Closes the application.

Tools

- **Search Tool** – Searches the current project's data in the database and returns all the pages that contains that particular search query. The search can take a long or short period depending on the size of the project.
- **Decoder Tool** – Encodes, Decodes or hashes the input data using the choosen encoding, decoding or hashing algorithm. Many other encoding,decoding and hashing algorithms will be added in the coming versions
- **Compare Tool** - compares two different pages or crawls then highlights the differences and similarities of the two pages or crawls.

Options

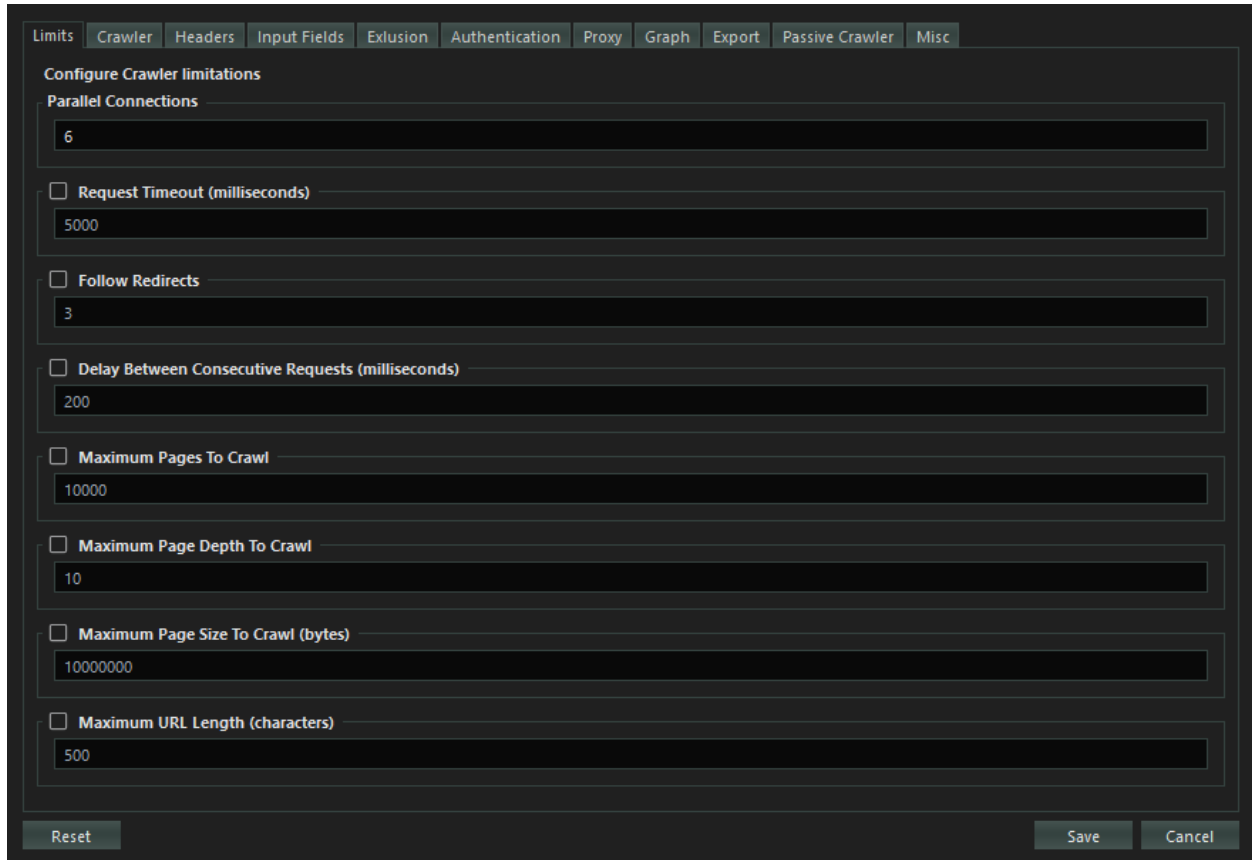
- **Preferences** - All program's settings and scan configurations.

Help

- **Log Viewer** – Displays all scan and program logs.
- **Donate** - Takes you to a page for donating to the SpiderSuite project.
- **Blog** - Takes you to SpiderSuite blog where all you can find all articles and documentation on SpiderSuite.
- **Twitter** - Takes you to SpiderSuite's twitter page.
- **Github** - Takes you to SpiderSuite's Github repository page.
- **Check For Updates** - checks for any available updates on SpiderSuite.
- **About** - Information about SpiderSuite.
- **About Qt** - Information about Qt C++ Framework used to create SpiderSuite.

Configurations

SpiderSuite Configurations are stored in json format in a SpiderSuite.conf configuration file that is located in the installation folder of SpiderSuite.



The screenshot displays the SpiderSuite configuration window with the 'Limits' tab selected. The window has a dark theme and a tabbed interface at the top. The tabs are: Limits, Crawler, Headers, Input Fields, Exclusion, Authentication, Proxy, Graph, Export, Passive Crawler, and Misc. The 'Configure Crawler limitations' section is active, showing several settings with checkboxes and input fields:

- Parallel Connections:** A text input field containing the value '6'.
- Request Timeout (milliseconds):** A checkbox is unchecked, and the text input field contains '5000'.
- Follow Redirects:** A checkbox is unchecked, and the text input field contains '3'.
- Delay Between Consecutive Requests (milliseconds):** A checkbox is unchecked, and the text input field contains '200'.
- Maximum Pages To Crawl:** A checkbox is unchecked, and the text input field contains '10000'.
- Maximum Page Depth To Crawl:** A checkbox is unchecked, and the text input field contains '10'.
- Maximum Page Size To Crawl (bytes):** A checkbox is unchecked, and the text input field contains '10000000'.
- Maximum URL Length (characters):** A checkbox is unchecked, and the text input field contains '500'.

At the bottom of the window, there are three buttons: 'Reset' on the left, and 'Save' and 'Cancel' on the right.

You can easily modify the configuration values and save for the changes to take place. You can also easily reset the configurations to default values that come with SpiderSuite, any changes saved or reset will also reset the SpiderSuite.conf file.

Limits Configuration:

Configures the Crawler's limitations.

Parallel Connections:

Sets the number of parallel connections to connect to the target server with during crawling. According to SpiderSuite's design, 6 connections is the most optimal configuration value as SpiderSuites employs other techniques to increase the efficiency of the spider such as; use of *HTTP pipelining* and *HTTP2 multi-plexing*.

Timeout:

Sets the maximum waiting duration in milliseconds for a request to elicit a response from the target server.

If checked (set to *true*), when maximum timeout is reached and there is no response from target server the request will be aborted and closed.

If unchecked (set to *false*), the request will remain active until it gets a response from server or until the 30 seconds threshold is reached.

Follow Redirects:

Sets the following of the redirect URL in case of a 3XX response status and the maximum number of redirects to follow.

If checked (set to *true*), when a request elicits a 3XX redirect response, the crawler will automatically redirect to the received redirection URL. You can also set the maximum number of redirects to follow.

If unchecked (set to *false*), when a 3XX response is received the crawler will not redirect the request, it will simply save the result and continue to crawl other links

Delay between Consecutive Requests:

Sets the wait time between sending consecutive requests. The delay is configured in milliseconds.

If checked (set to *true*), when spider sends a request to the target server it will wait for XXX milliseconds before sending another request to that target server.

If unchecked (set to *false*), the spider will send multiple requests in a tight loop to the target server.

This should only be checked (set to *true*) in cases of targets that;

- Can't handle too many requests
- Detects and prevents high speed crawling
- Has strict crawling rules

Otherwise it slows down the crawling process, as the spider has to delay for XXX milliseconds before sending another request.

Maximum Pages to Crawl:

Sets the maximum number of successful result pages to be crawled. This only takes into account the successfully crawled pages from the target server.

If checked (set to *true*), when the spider successfully crawls the XXX number of pages from the target server the spider will automatically stop regardless whether there are still crawlable links available

If not checked (set to *false*), the spider will crawl the target site until either you stop the crawling manually or until the spider finishes crawling all the links.

Maximum Crawl Depth:

Sets the maximum page depth to crawl for a particular target. Depth refers to the position of the page from the host domain.

e.g. ***https://www.example_domain.com/depth_1/depth_2/depth_3?name=value***

If checked (set to *true*), the spider will not crawl a page whose depth exceeds the maximum crawl depth. Only pages whose depth are between 0 and max crawl depth will be crawled.

If not checked (set to *false*), the spider will crawl all pages of all depths of the particular target.

Maximum Page Size to Crawl:

Sets the maximum page size in bytes for download.

If checked (set to *true*), pages whose size is above the maximum page download size will not be downloaded.

If not checked (set to *false*), pages of all sizes will be downloaded and crawled.

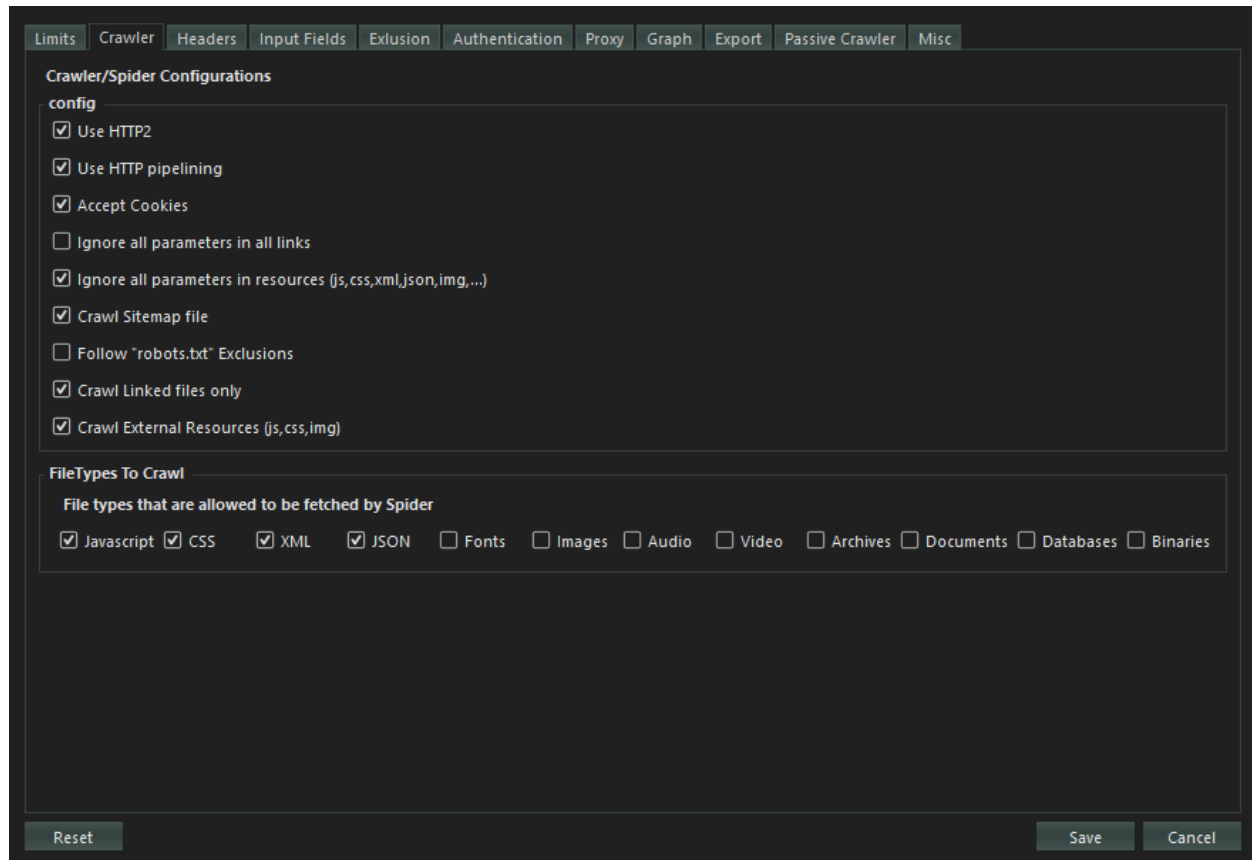
Maximum URL Length:

Sets the maximum allowed length in bytes/characters for a URL link.

If checked (set to *true*), links whose length is above the maximum link length value will not be crawled. If not checked (set to *false*), links of all lengths will be crawled.

Crawler Configuration

Crawler specific configurations.



The screenshot shows a configuration window for a crawler with several tabs: Limits, Crawler, Headers, Input Fields, Exclusion, Authentication, Proxy, Graph, Export, Passive Crawler, and Misc. The 'Crawler' tab is selected. The 'Crawler/Spider Configurations' section contains a 'config' box with the following options:

- ☒ Use HTTP2
- ☒ Use HTTP pipelining
- ☒ Accept Cookies
- ☐ Ignore all parameters in all links
- ☒ Ignore all parameters in resources (js,css,xml,json,img,...)
- ☒ Crawl Sitemap file
- ☐ Follow "robots.txt" Exclusions
- ☒ Crawl Linked files only
- ☒ Crawl External Resources (js,css,img)

The 'FileTypes To Crawl' section contains a box titled 'File types that are allowed to be fetched by Spider' with the following options:

- ☒ Javascript
- ☒ CSS
- ☒ XML
- ☒ JSON
- ☐ Fonts
- ☐ Images
- ☐ Audio
- ☐ Video
- ☐ Archives
- ☐ Documents
- ☐ Databases
- ☐ Binaries

At the bottom of the window are three buttons: 'Reset', 'Save', and 'Cancel'.

Use HTTP2:

Configures the use of HTTP version 2 protocol for request to the target server. If a server turns out to not support HTTP/2 the spider will fall back to using HTTP version 1 by default.

HTTP/2 is the second version of the HTTP protocol aiming to make applications faster, simpler, and more robust by improving many of the drawbacks of the first HTTP version. Has features including;

- **Binary protocols** – Binary protocols consume less bandwidth, are more efficiently parsed and are less error-prone than the textual protocols used by HTTP/1.1.
- **Multiplexing** – HTTP/2 is multiplexed, i.e., it can initiate multiple requests in parallel over a single TCP connection. As a result, web pages containing several elements are delivered over one TCP connection.

- **Header compression** – HTTP/2 uses header compression to reduce the overhead caused by TCP's slow-start mechanism.
- **Server push** – HTTP/2 servers push likely-to-be-used resources into a browser's cache, even before they're requested. This allows browsers to display content without additional request cycles.
- **Increased security** – Web browsers only support HTTP/2 via encrypted connections, increasing user and application security.

HTTP Pipelining:

Configures the spider to use http-pipelining technique to increase speed and efficiency. Works when using HTTP/1.

HTTP pipelining is a feature of HTTP/1.1 which allows multiple HTTP requests to be sent over a single TCP connection without waiting for the corresponding responses. HTTP/1.1 requires servers to respond to pipelined requests correctly, with non-pipelined but valid responses even if server does not support HTTP pipelining. Despite this requirement, many legacy HTTP/1.1 servers do not support pipelining correctly, forcing most HTTP clients to not use HTTP pipelining.

The technique was superseded by multiplexing via HTTP/2.

Accept Cookies:

Configures the spider to accept and use cookies sent by the target server. The spider will store the cookie and send it back to the same server with later requests.

An **HTTP cookie** (web cookie, browser cookie) is a small piece of data that a server sends to a user's web browser (in this case SpiderSuite). Cookies are mainly used for three purposes: Session management, Personalization & Tracking.

Ignore all Parameters in all links:

Configures the crawler to ignore all **query parameters** in all links it crawls. e.g.

https://example.com?param1=1¶m2=2 → https://example.com

Ignore all Parameters in Resource links:

Configures the crawler to ignore all query parameters in resource links only. Resource links here refers to links which **are not of html filetype** e.g. js, css, xml, json links.

Crawl sitemap File:

Configures the crawler to first fetch sitemap.xml file if the site contains one. If the site does possess the sitemap.xml file then the spider will use the links from the sitemap.xml file as seed links for crawling the target site.

An XML sitemap is a file that lists a website's essential pages, making sure Google can find and crawl them all. It also helps search engines understand your website structure. You want Google to crawl every important page of your website.

Follow “robots.txt” Exclusions:

Configures the crawler to follow the “/robots.txt” exclusions if available. The robots.txt exclusion is highly dependent on the **user-agent** used hence the crawler will follow the exclusions according to the **user-agents** you’ve chosen.

The robots exclusion standard, also known as the robots exclusion protocol or simply robots.txt, is a standard used by websites to communicate with web crawlers and other web robots.

Crawl linked files only:

Configures the crawler to crawl only the links that it extracts from the crawled pages meaning it will not crawl other directories in the link structure.

E.g. For a linked link ***https://example.com/dir1/dir2/dir3***, the crawler will only fetch for page ***https://example.com/dir1/dir2/dir3*** and won't try to crawl other directory pages found in this link such as ***https://example.com/dir1*** and ***https://example.com/dir/dir2***.

Crawl External Resources:

Configures the spider to crawl resources (images, css, js, json, xml & fonts) hosted from a different host. Most webpages uses resources hosted from different hosts. If this feature is selected external resources will be crawled if this feature not selected only resources from the target host will be crawled.

File types to Crawl:

Configures which file types are allowed to be crawled. **HTML files are crawled by default**, but for other file types you must allow (by checking the particular checkbox) for them to be crawled. If not allowed the file type will not be added to the spider seed.

Headers Configuration

Crawler's request headers configuration.

The screenshot shows the 'Headers' tab in a configuration window. The title is 'HTTP Headers and User-Agents to be sent with each Crawler/Spider request'. There are two main sections: 'Request Headers' and 'User Agent Header'.

Request Headers: A list of headers with checkboxes and values. The 'Referer' header is selected. Other headers include 'Accept', 'Accept-Charset', 'Accept-Encoding', and 'Accept-Language'.

Header	Value
<input checked="" type="checkbox"/> Referer	
<input type="checkbox"/> Accept	*/*
<input type="checkbox"/> Accept-Charset	utf-8
<input type="checkbox"/> Accept-Encoding	br
<input type="checkbox"/> Accept-Encoding	gzip
<input type="checkbox"/> Accept-Language	*

Below the list are input fields for 'Enter header name...' and 'Enter header value...', and an 'Add' button.

User Agent Header: A list of user agents with checkboxes. The 'Alcatel-OT-708' user agent is selected. Other user agents include various BlackBerry models.

User Agent
<input checked="" type="checkbox"/> Alcatel-OT-708/1.0 Profile/MIDP-2.0 Configuration/CLDC-1.1 ObigoInternetBrowser/Q03C
<input type="checkbox"/> BlackBerry7100i/4.1.0 Profile/MIDP-2.0 Configuration/CLDC-1.1 VendorID/103
<input type="checkbox"/> BlackBerry7130e/4.1.0 Profile/MIDP-2.0 Configuration/CLDC-1.1 VendorID/104
<input type="checkbox"/> BlackBerry7230/3.7.0
<input type="checkbox"/> BlackBerry7250/4.0.0 Profile/MIDP-2.0 Configuration/CLDC-1.1
<input type="checkbox"/> BlackBerry7520/4.0.0 Profile/MIDP-2.0 Configuration/CLDC-1.1
<input type="checkbox"/> BlackBerry7730/3.7.0
<input type="checkbox"/> BlackBerry8130/4.3.0 Profile/MIDP-2.0 Configuration/CLDC-1.1 VendorID/109
<input type="checkbox"/> BlackBerry8310/4.2.2 Profile/MIDP-2.0 Configuration/CLDC-1.1 VendorID/121
<input type="checkbox"/> BlackBerry8320/4.3.1 Profile/MIDP-2.0 Configuration/CLDC-1.1

Below the list are input fields for 'Enter user_agent...' and an 'Add' button.

At the bottom of the window are 'Reset', 'Save', and 'Cancel' buttons.

Request Headers:

Choose (check the header's check box of) the headers you want to use for crawling the target site. You can add your own custom headers by introducing the header name and value then click add button to add the new header to the headers list.

When checked (in use) the **Referer** header value will be added automatically by the spider during the scan depending on where the link was extracted from.

User Agent Header

The User-Agent request header is a characteristic string that lets servers and network peers identify the application, operating system, vendor, and/or version of the requesting user agent.

Check the (User Agent Header) checkbox to allow the use of user agents in crawling. Then choose the user agent(s) to use. If multiple user agents are chosen, they will all be used randomly for each request sent.

Input Fields Configuration

input name...	input value...
Accept-Language	en
Client-IP	127.0.0.1
DOB	1967/1/1
Origin	http://www.test.com/
Referer	http://www.google.com/search?hl=en&q=testing
Via	1.1 wa.www.test.com
X-Forwarded-For	127.0.0.1
X-Forwarded-Host	localhost
accounts	1001160141
addr	3137 Laguna Street
address	3137 Laguna Street
age	20
alter	20
anno	1967
area	555
awayyear	9999
benutzer	aaabbbccc
beruf	programmer
birth	01/01/1967
birth_year	1967
birthday	01/01/1967

Configures the default values for the listed common input name and types for HTML forms.

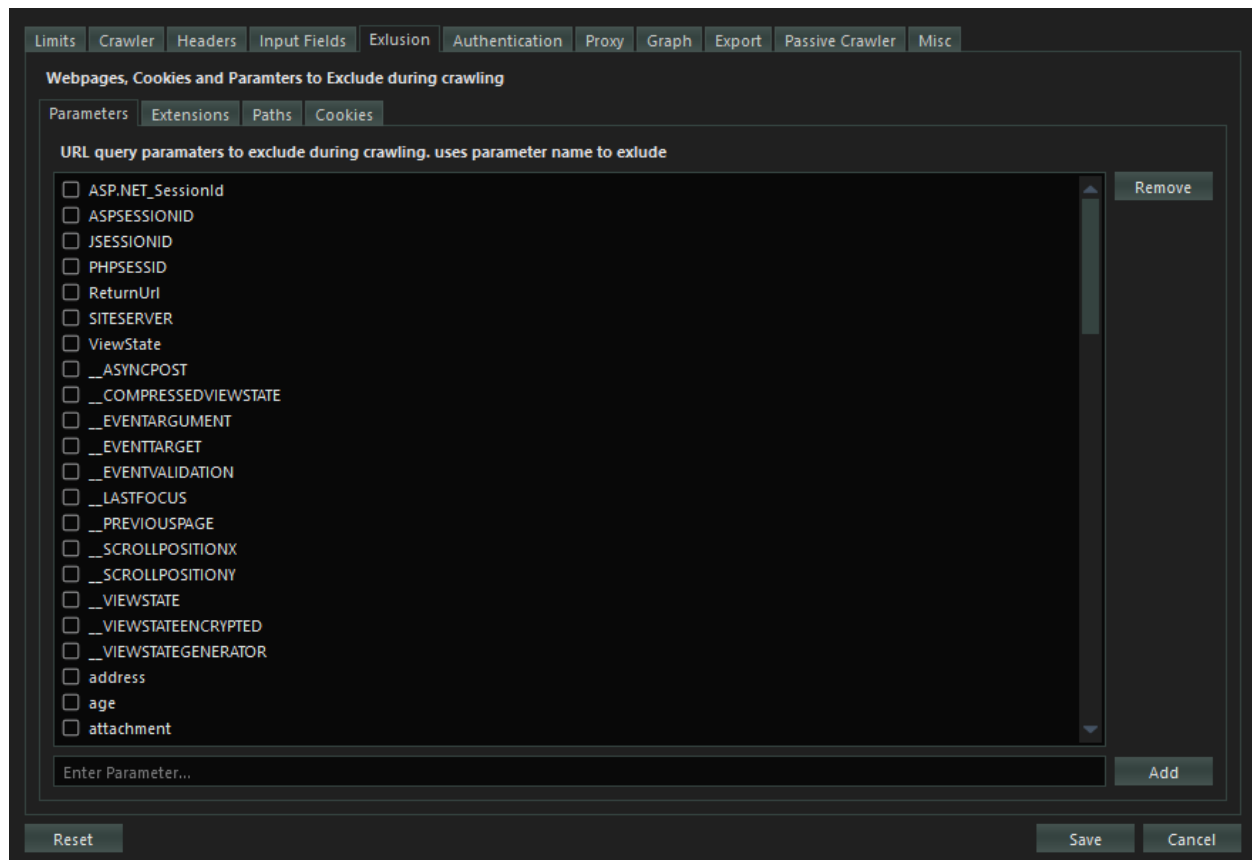
When crawling the HTML forms, values from all empty input values will be filled by the values from this list.

You can also add your own custom input name/type and value to be used in HTML form's inputs.

You can also parse and extract input fields from a URL automatically by fetching it then adding values to the input fields manually (***This is more efficient and precise***).

Exclusion Configuration

Configures what link's query parameters, paths, extensions and cookies should be excluded from crawling.



- **Exclude Parameters**

Configures which query parameters should be excluded from a network request if present. If exclude all parameters is chosen then query parameters from all links will be removed.

- **Exclude File Extensions**

Configures the Crawler to exclude crawling all links with file types of the specified extensions (pdf, exe, docx). You can add a file type extension to be excluded

- **Exclude Paths**

Configures the Crawler to exclude crawling the paths which match to the exclusion patterns. You can add custom excluded paths patterns (***regular expression patterns***) for each scan.

- **Exclude Cookies**

Configures the spider to exclude the chosen cookie patterns from the cookie jar. You can add custom cookie patterns (regular expression patterns) to exclude from the scan.

Authentication Configuration

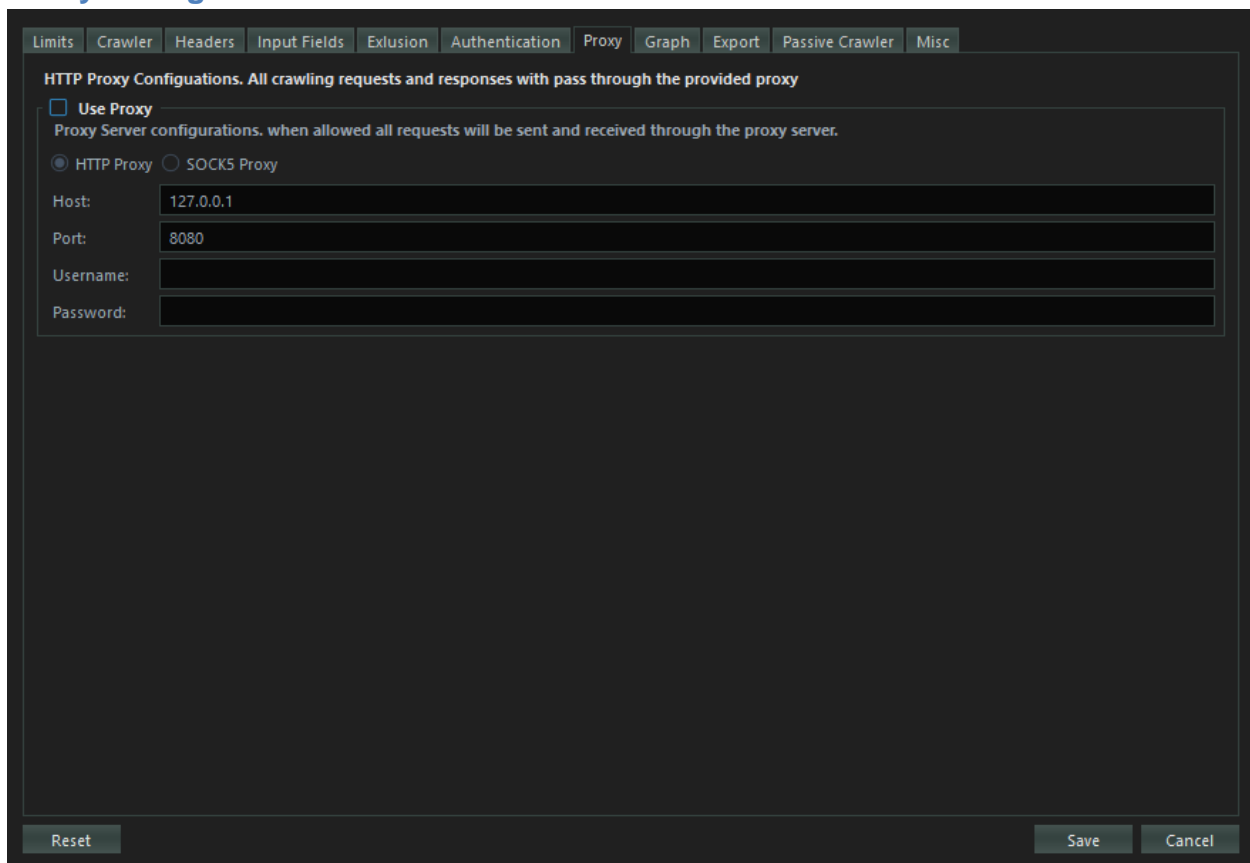
Authentication Configuration window showing the 'Standard Based' tab. The window displays a table for adding authentication credentials (Username, Password, Host, Path) and includes buttons for 'Reset', 'Save', and 'Cancel'.

Configures the credentials for authenticated crawl. For current version 1.0.0 of SpiderSuite, it only supports standard based authentications which include **Basic**, **NTLM version 2**, **Digest-MD5** and **SPNEGO/Negotiate**.

Simply add the credential values; username, password, host site and the path that requires the authentication. After that you can check the credentials for it to be usable when the crawler detects authentication is required.

Form based authentication will be introduced in the coming versions of SpiderSuite.

Proxy Configuration



The image shows a 'Proxy Configuration' dialog box with a dark theme. At the top, there is a horizontal tab bar with the following tabs: Limits, Crawler, Headers, Input Fields, Exclusion, Authentication, Proxy (which is selected), Graph, Export, Passive Crawler, and Misc. Below the tabs, the main content area has a title 'HTTP Proxy Configurations. All crawling requests and responses will pass through the provided proxy'. Under this title, there is a checkbox labeled 'Use Proxy'. Below the checkbox is a text label: 'Proxy Server configurations, when allowed all requests will be sent and received through the proxy server.' There are two radio buttons: 'HTTP Proxy' (which is selected) and 'SOCKS Proxy'. Below these are four input fields: 'Host:' with the value '127.0.0.1', 'Port:' with the value '8080', 'Username:', and 'Password:'. At the bottom of the dialog, there are three buttons: 'Reset', 'Save', and 'Cancel'.

Limits Crawler Headers Input Fields Exclusion Authentication Proxy Graph Export Passive Crawler Misc

HTTP Proxy Configurations. All crawling requests and responses will pass through the provided proxy

☐ Use Proxy
Proxy Server configurations, when allowed all requests will be sent and received through the proxy server.

☒ HTTP Proxy ☐ SOCKS Proxy

Host: 127.0.0.1

Port: 8080

Username:

Password:

Reset Save Cancel

Configures http and socks5 proxy connection for the crawler. In case of anonymity or integration with other tools such as Burp Suite & Zed Attack Proxy (ZAP) and bypassing various IP related crawling drawbacks.

All crawler request and responses will pass through the configured proxy.

Graph Configuration

LimitsCrawlerHeadersInput FieldsExclusionAuthenticationProxyGraphExportPassive CrawlerMisc

Configures graph characteristics/Attributes

Graph Attributes

Graph Type:

Dot

▼

Graph Direction:

Left To Right

▼

Node Shape:

Rectangle

▼

Color By:

Status Code

▼

Colors

HTTP Status	Color
Informational	yellow
Succesful	green
Redirection	orange
Error	red
Notknown	gray

Reset

Save

Cancel

Configures Graph visualizing the Crawled sitemap. You can modify the graph appearance on your own liking for better data presentation.

Configure Exports

Configures what data/information about a page should be exported on various export types.

The screenshot shows a dark-themed window titled 'Configure Exports'. At the top is a horizontal tab bar with the following tabs: Limits, Crawler, Headers, Input Fields, Exclusion, Authentication, Proxy, Graph, Export (selected), Passive Crawler, and Misc. Below the tabs, the text 'Configures what extra information to export for each export type' is displayed. The main area is divided into sections for different export formats: CSV, XML, JSON, and HTML. Each section contains four checkboxes: 'Status Code', 'Size', 'Title', and 'Content Type'. All these checkboxes are checked. Below these sections is a 'Sitemap' section. It contains three checkboxes: 'lastmod', 'changefreq', and 'priority'. Each checkbox is followed by a text input field. The 'lastmod' field has a placeholder 'e.g 2004-10-01T18:23:17+00:00'. The 'changefreq' field has a placeholder 'e.g monthly'. The 'priority' field has a placeholder 'e.g 0.7'. At the bottom of the window are three buttons: 'Reset', 'Save', and 'Cancel'.

The page URLs will all be exported by default but other data about the webpage such as Status Code, Size, title and Content Type will only be exported when you choose them to be exported.

This feature gives control to the user on what data to be included in the export file.

Passive Crawler Configuration

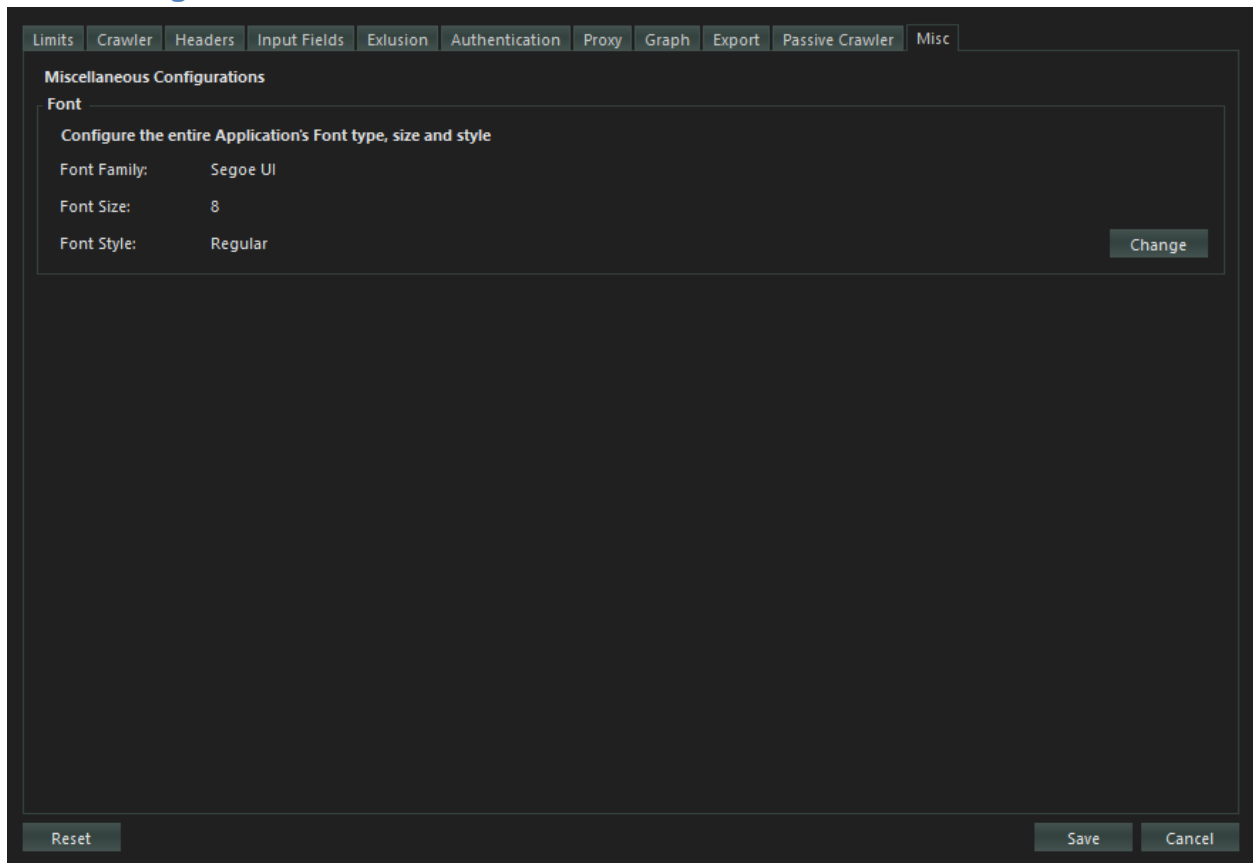
Configures the SpiderSuite passive crawler.

The screenshot shows the 'Passive Crawler' configuration window. At the top, there is a horizontal tab bar with the following tabs: Limits, Crawler, Headers, Input Fields, Exclusion, Authentication, Proxy, Graph, Export, Passive Crawler (which is currently selected), and Misc. Below the tabs, the main content area is titled 'Configurations for the passive crawler'. Under this title, there is a section labeled 'Allowed Pages'. Inside this section, a subtitle reads 'File types that are allowed to be added from the passive sitemap'. Below the subtitle is a row of checkboxes for various file types: HTML (checked), Javascript (checked), CSS (checked), Json (checked), XML (checked), Font (checked), Image (unchecked), Audio (unchecked), Video (unchecked), Document (unchecked), Database (unchecked), Archive (unchecked), Binary (unchecked), and Misc (unchecked). At the bottom of the window, there are three buttons: 'Reset' on the left, and 'Save' and 'Cancel' on the right.

- **Allowed Pages**

Configures which pages (page type e.g. html, js, css) are allowed to be added to the sitemap after the crawling is finished or when you add links.

Misc Configuration



The screenshot shows the 'Misc' tab in the SpiderSuite configuration interface. The 'Miscellaneous Configurations' section is active, and the 'Font' sub-section is expanded. It contains a heading 'Configure the entire Application's Font type, size and style' and three settings: 'Font Family' set to 'Segoe UI', 'Font Size' set to '8', and 'Font Style' set to 'Regular'. A 'Change' button is located to the right of the 'Font Style' setting. At the bottom of the window, there are 'Reset', 'Save', and 'Cancel' buttons.

Miscellaneous Configurations	
Font	
Configure the entire Application's Font type, size and style	
Font Family:	Segoe UI
Font Size:	8
Font Style:	Regular

Buttons: Reset, Save, Cancel

Miscellaneous configurations

- **Font**

Configures the SpiderSuite's entire application font type, size and style, simply change the application's font by using the font dialog.

The changes will take effect after restarting SpiderSuite.

Crawling