Web Application Asset Addition Workflow

Stepper-1: Create Asset

Asset Type: **dropdown**

Asset Name: **Input Box**

Asset URL: **Input Box with URL validation (only URLs with http://, https:// ws://, wss:// should be allowed)**

Stepper-2: Asset Validation

Is your asset protected with HTTP Basic/Digest Authentication? Yes/No (Enabled/Disable)

If yes, show them the Username & Password input box.

Validate Button – It will work similar to Website Asset

Back Button - It will work similar to Website Asset

Continue Button - It will work similar to Website Asset

Advanced Settings

1. Do you want to run the scan on test server IP Address keeping the above domain?
2. Validate the IP Address
3. IP Address Verification

Stepper-3: Asset Verification (Same as Website Asset)

Stepper-4: Configuration

- Select a suitable Login Method from below.

1. Login with Credentials
    a. URL (text box with the same server validation as mentioned in Asset Validation)
    b. Username (text box with max-length set to 50 from client & server side both)
    c. Password (password box with max-length set to 50 from client & server side both)
2. Login with Session Cookies
    a. Cookie Header Input Box (text box to add value of Cookie Header) – Refer to the below shared flow and screenshot.
3. Login with Auth Headers
    a. Header Input Box (text box to add a complete header) – Refer to the below shared flow and screenshot.
4. Login Sequence Recorder

a. File Upload Box to upload a .side file
   Note: Make sure that the server side validation for the file extension is properly set. Only .side file should be allowed. No files like .php.side or .side.php or .side;.php or .php;.side or .php%00.side or .side%00.php should be allowed. PHP is just for an example, but no such type of double extensions with other technologies should be allowed like .asp, .aspx, .jsp, .html, .phtml, .shtml, .htm, etc.

- After Login URL (example - /dashboard) : input box with URL validations as mentioned above

  Understanding - Attempt to automatically login with the preferred method, and navigate to the the after login URL. Take a screenshot of after login URL and show it to the customer as similar to the Asset Validation)

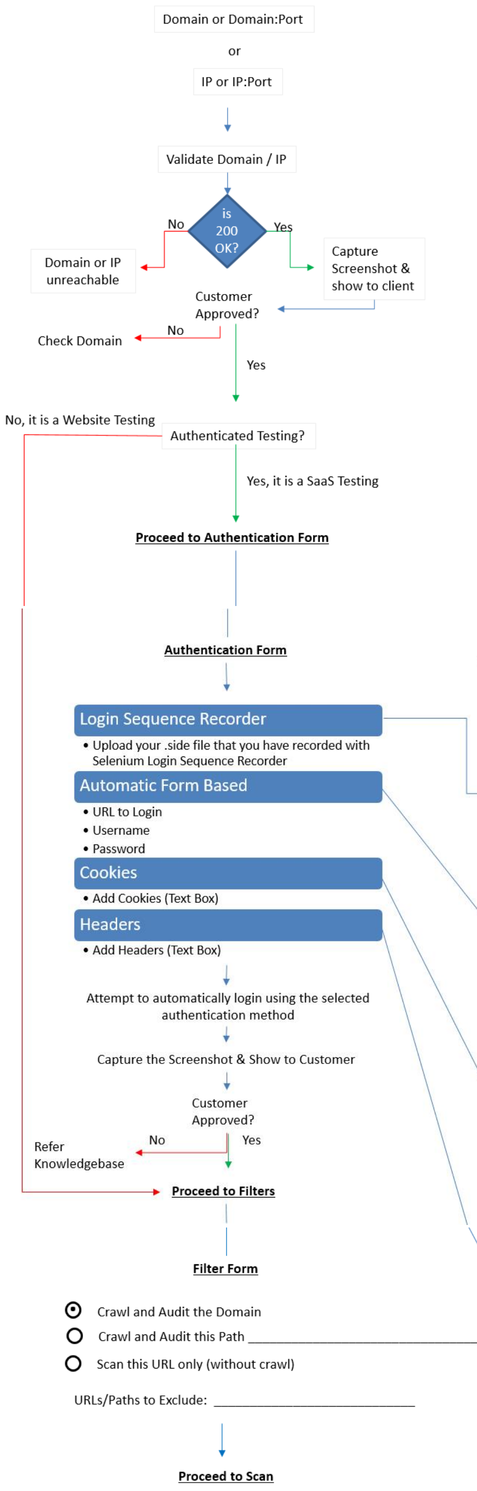
Stepper-5: Scan Profile

- ○ **Crawl & Audit** (radio button)
- ○ **Scan this URL only** (radio button)

- **Enter the URLs to exclude** (text box with URL Validations)
- **Logout URL*** (text box with URL Validations)
- **Your API Endpoint**
- **Your WebSocket Endpoint**
- **Your CDN, Cloud Storage & Other Endpoints**

Stepper-6: IP Whitelist

List of IP Addresses to whitelist inside the Box (Refer BeagleSecurity for the same)



Domain or Domain:Port

or

IP or IP:Port

Validate Domain / IP

is 200 OK?

No

Domain or IP unreachable

Yes

Capture Screenshot & show to client

Customer Approved?

No

Check Domain

Yes

No, it is a Website Testing

Authenticated Testing?

Yes, it is a SaaS Testing

Proceed to Authentication Form

Authentication Form

Login Sequence Recorder

- Upload your .side file that you have recorded with Selenium Login Sequence Recorder

Automatic Form Based

- URL to Login
- Username
- Password

Cookies

- Add Cookies (Text Box)

Headers

- Add Headers (Text Box)

Attempt to automatically login using the selected authentication method

Capture the Screenshot & Show to Customer

Customer Approved?

No

Refer Knowledgebase

Yes

Proceed to Filters

Filter Form

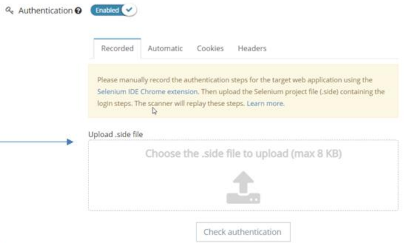
● Crawl and Audit the Domain

○ Crawl and Audit this Path ____

○ Scan this URL only (without crawl)

URLs/Paths to Exclude: ________________________________

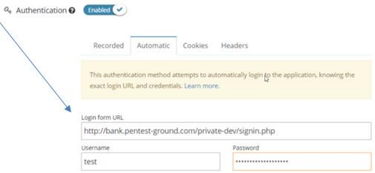
Proceed to Scan



Please manually record the authentication steps for the target web application using the Selenium IDE Chrome extension. Then upload the Selenium project file (.side) containing the login steps. The scanner will replay these steps. Learn more.

Upload .side file

Choose the .side file to upload (max 8 KB)

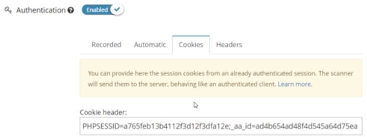


This authentication method attempts to automatically login to the application, knowing the exact login URL and credentials. Learn more.

Login form URL: http://bank.pentest-ground.com/private-dev/signin.php

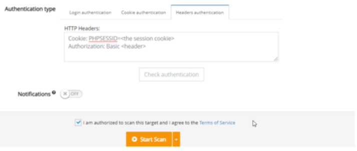
Username: test

Password: ••••••••••••••••••



You can provide here the session cookies from an already authenticated session. The scanner will send them to the server, behaving like an authenticated client. Learn more.

Cookie header: PHPSESSID=a765feb13b4112f3d12f3dfa12e;_aa_id=ad4b654ad48f4d545a64d75ea



HTTP Headers:

```
Cookie: PHPSESSID=<the session cookie>
Authorization: Basic <header>
```

☑ I am authorized to scan this target and I agree to the Terms of Service