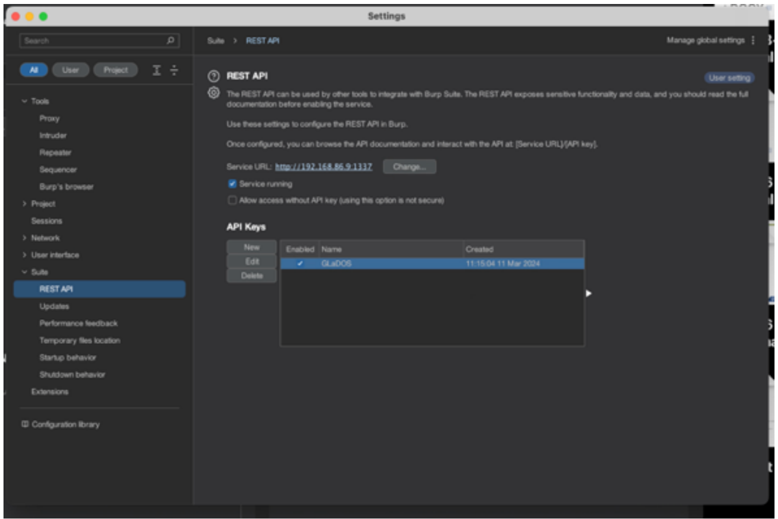
Title: Enhancing Offensive Security: Penetration Testing with AI triggers and Burpsuite REST API

Author: Anson Stahl

Date: March 26,2024

The integration of AI triggers with Burp Suite's REST API through iOS shortcuts
allows cybersecurity professionals to employ AI-driven decision-making to automatically initiate specific actions in Burp Suite based on real-time data analysis and threat detection, all directly from their iOS and apple silicone devices.
By leveraging machine learning or GPT models, the system can proactively identify vulnerabilities and initiate scans or attacks, based on contextual conditions. This shift towards predictive security practices enables a more robust streamlining the penetration testing process but also offer a forward-looking perspective.

- Security Considerations: When dealing with Burp Suite, especially if integrating with its REST API for testing purposes, ensure your API tokens and any sensitive information are securely handled burp will copy the private key for auth to clipboard upon enabling the rest api.
  - API Interaction: (http://youripaddress:port/v0.1/scan/<scan_id>)



REST API

The REST API can be used by other tools to integrate with Burp Suite. The REST API exposes sensitive functionality and data, and you should read the full documentation before enabling the service.

Use these settings to configure the REST API in Burp.

Once configured, you can browse the API documentation and interact with the API at: [Service URL]/[API key].

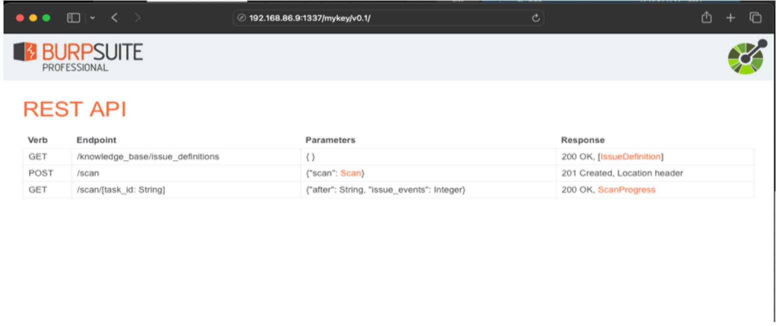
Service URL: http://192.168.86.9:1337

☑ Service running

☐ Allow access without API key (using this option is not secure)

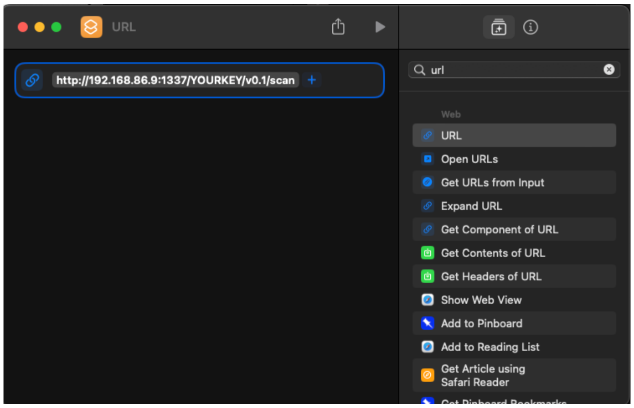
API Keys

Enabled	Name	Created
✓	GLaDOS	11:15:04 11 Mar 2024



REST API

Verb	Endpoint	Parameters	Response
GET	/knowledge_base/issue_definitions	{ }	200 OK, [IssueDefinition]
POST	/scan	{"scan": Scan}	201 Created, Location header
GET	/scan/[task_id: String]	{"after": String, "issue_events": Integer}	200 OK, ScanProgress



http://192.168.86.9:1337/YOURKEY/v0.1/scan

- Create a New Shortcut
- Add Action: Search for “URL” Action
  - • URL Action: Add “Get Contents of URL” Action
  - • Enter the Burp Suite REST API endpoint URL, *Adjust the request method as needed*.
- Handle the Response:
  - Use “Get Dictionary from Input” to parse the JSON response from the API, allowing additional actions with the returned data. This will be used for requests for scan results.
    Add a Response Action
    - *Append a “Speak Text” or “Append to Note” action at the end to tie into Apples TTS engine.*

  - Add Actions for File Reading

- Read File Action:
- Add an “Get File” action. This allows you to select a file from your device.
- Ensure the file contains the JSON data model

  - Parse JSON from File reading a JSON file stored on your device, parsing it into a dictionary, and then merging this data with your API call.

Here’s how to integrate this process:

Add Actions for File Reading

- Read File Action:
- Add an “Get File” action. This allows you to select a file from your device.
- Ensure the file contains the JSON data model

Parse JSON from File

• Convert File to Text: Add a “Get Text from Input” action right after the file action to ensure the file’s contents are treated as text.

• Create Dictionary from JSON: Follow with a “Get Dictionary from Input” action. This converts the text, assumed to be JSON format, into a dictionary within Shortcuts.

Perform API Call

• URL and API Request Setup: If you need to incorporate values from the JSON file into an API request:

• Set up the “URL” action with the Burp Suite API endpoint.

• Use “Get Contents of URL” to make the API call, modifying the method, headers, and body as needed. Include JSON values by using variables that reference the earlier created dictionary.

For Burp Suite, ensure you include any necessary authentication tokens in the headers to authenticate your request. We also can return scan results, metrics, and or data relevant to our needs.

As you can see, they all default to null except for the two requirements we have added in our demonstration.

```
{
  "urls": [String],
  "name": String,                          // defaults to: null
  "scope": Scope,                          // defaults to: null
  "application_logins": [ApplicationLogin], // defaults to: []
  "scan_configurations": [Configuration],   // defaults to: []
  "resource_pool": String,                 // defaults to: null
  "scan_callback": Callback,               // defaults to: null
  "protocol_option": ProtocolOption        // defaults to: null
}
```

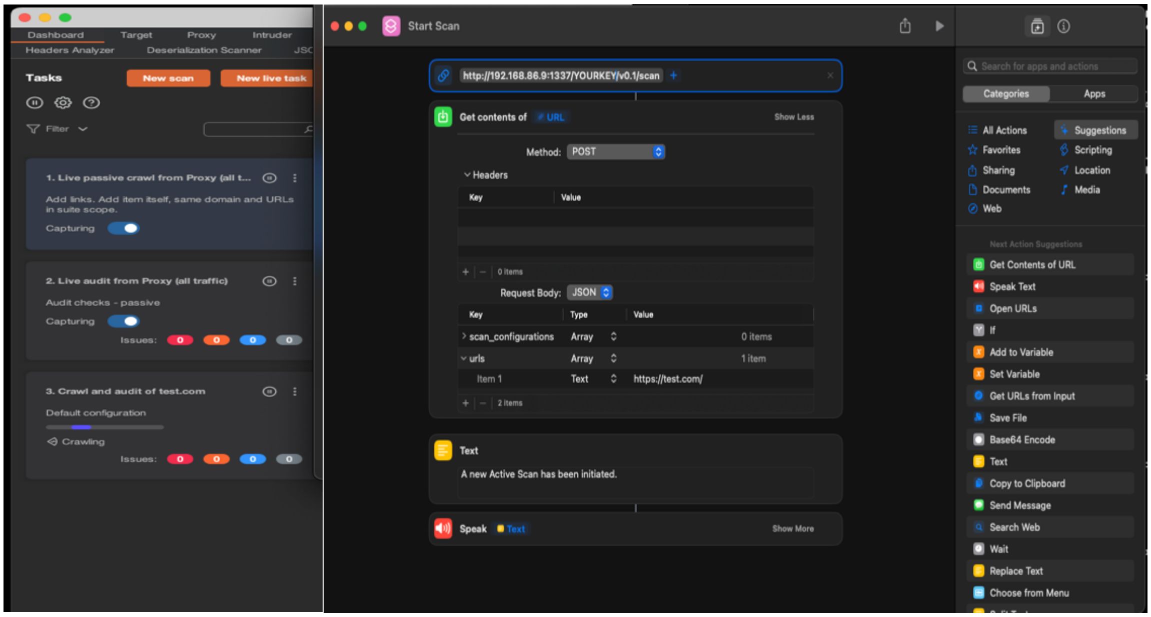
- Use JSON Values

Manipulate Dictionary: Now that you have your JSON data as a dictionary, use “Get Value for Key” actions to access specific values. You can use these values directly in your API calls, display them, or perform other logic.

Merge with API Call Data

- Merge Dictionaries: If you’re combining data from the JSON file with data from an API call, you might need to get variables to merge data.

- Create a New Shortcut
- Add Action: Search for “URL” Action
    - URL Action: Add “Get Contents of URL” Action
    - Enter the Burp Suite REST API endpoint URL.
    - Method POST
    - Request Body:JSON
        - Add key scan_configurations Array
        - Add key urls Array
            - Add a text value for urls https://test.com/



1. Live passive crawl from Proxy (all t...

Add links. Add item itself, same domain and URLs in suite scope.

Capturing

2. Live audit from Proxy (all traffic)

Audit checks - passive

Capturing

Issues: 0 0 0 0

3. Crawl and audit of test.com

Default configuration

Crawling

Issues: 0 0 0 0

Start Scan

http://192.168.86.9:1337/YOURKEY/v0.1/scan

Get contents of URL

Method: POST

Headers

Key	Value

0 items

Request Body: JSON

Key	Type	Value
scan_configurations	Array	0 items
urls	Array	1 item
Item 1	Text	https://test.com/

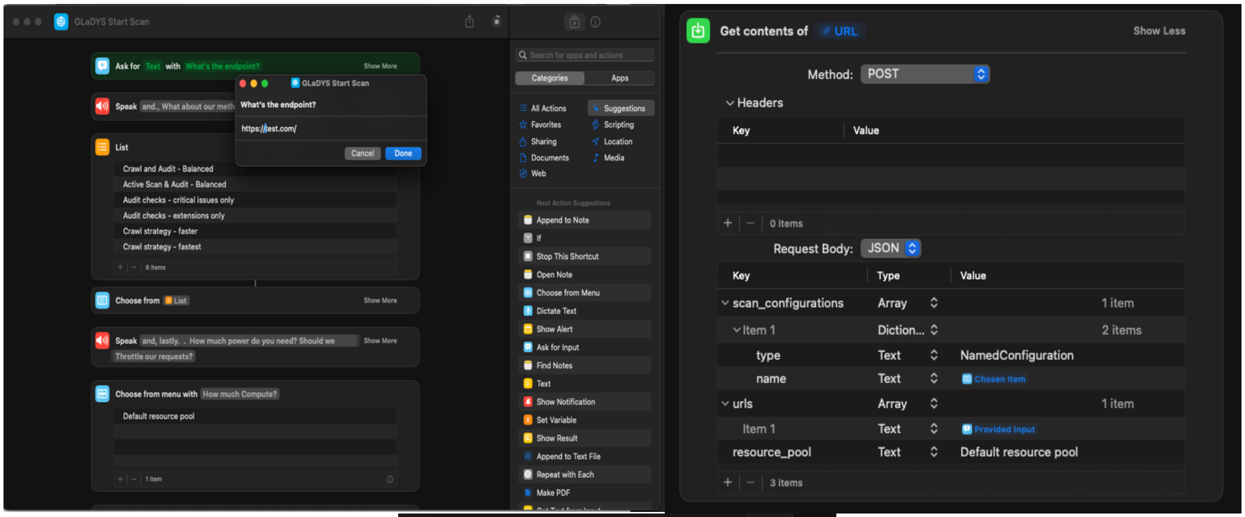
2 items

Text

A new Active Scan has been initiated.

Speak Text

- Additionally, we can insert the variable for a chosen input from menu as shown.



GLaDYS Start Scan

Ask for Text with What's the endpoint?

Speak and., What about our meth

List

- Crawl and Audit - Balanced
- Active Scan & Audit - Balanced
- Audit checks - critical issues only
- Audit checks - extensions only
- Crawl strategy - faster
- Crawl strategy - fastest

6 items

Choose from List

Speak and, lastly. . How much power do you need? Should we Throttle our requests?

Choose from menu with How much Compute?

- Default resource pool

1 item

GLaDYS Start Scan

What's the endpoint?

https://test.com/

Get contents of URL

Method: POST

Headers

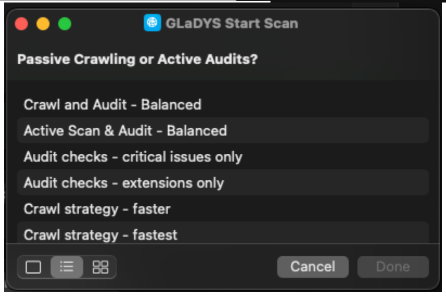
Key	Value

0 items

Request Body: JSON

Key	Type	Value
scan_configurations	Array	1 item
Item 1	Diction...	2 items
type	Text	NamedConfiguration
name	Text	Chosen Item
urls	Array	1 item
Item 1	Text	Provided Input
resource_pool	Text	Default resource pool

3 items



GLaDYS Start Scan

Passive Crawling or Active Audits?

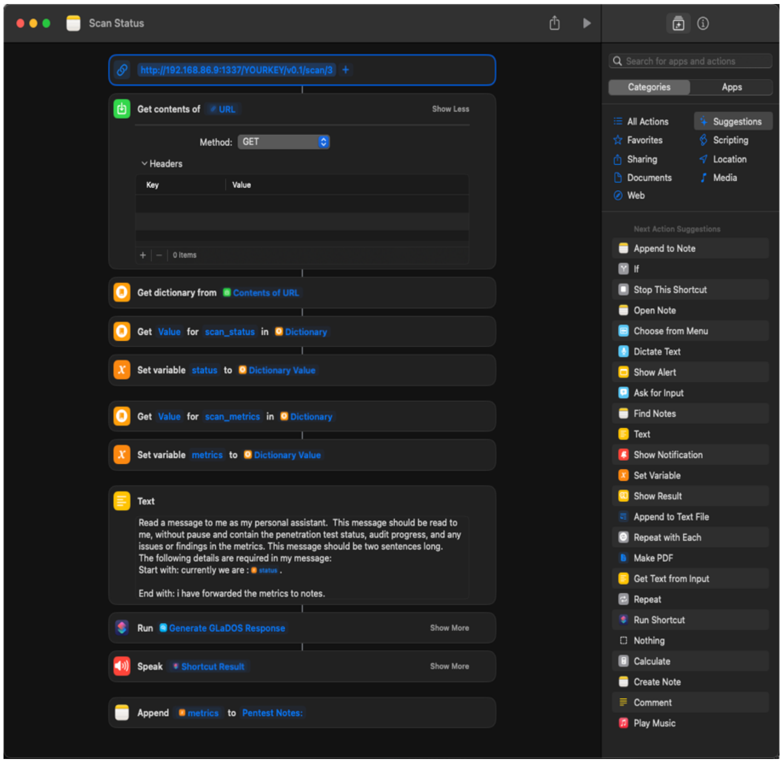
- Crawl and Audit - Balanced
- Active Scan & Audit - Balanced
- Audit checks - critical issues only
- Audit checks - extensions only
- Crawl strategy - faster
- Crawl strategy - fastest

Next, a basic call to GET scan results.

The dictionary contains all issues you specified in either the scanner configuration or the view? Tag on the url

- Create a New Shortcut
- Add Action: Search for “URL” Action
    - • URL Action: Add “Get Contents of URL” Action
    - • Enter the Burp Suite REST API endpoint URL, *Adjust the request method as needed*.
    - Use “Get Dictionary from Input” to parse the JSON response from the API URL, allowing additional actions with the returned data.
      Add a Response Action
        - *Append a “Speak Text” or “Append to Note” action at the end or “Run Shortcut” as shown (running AI models).*

    - Add Actions for File Reading
    - Add an “Get File” action Ensure the file contains the JSON data model
    - Parse JSON from File reading a JSON file stored on your device, parsing it into a dictionary, and then merging this data with your API call.



http://192.168.86.9:1337/YOURKEY/v0.1/scan/3 +

Get contents of URL — Show Less

Method: GET

Headers

Key	Value

+ | − | 0 items

Get dictionary from Contents of URL

Get Value for scan_status in Dictionary

Set variable status to Dictionary Value

Get Value for scan_metrics in Dictionary

Set variable metrics to Dictionary Value

Text

Read a message to me as my personal assistant. This message should be read to me, without pause and contain the penetration test status, audit progress, and any issues or findings in the metrics. This message should be two sentences long.
The following details are required in my message:
Start with: currently we are : status .

End with: i have forwarded the metrics to notes.

Run Generate GLaDOS Response — Show More

Speak Shortcut Result — Show More

Append metrics to Pentest Notes:

- http://youripaddress:port/v0.1/scan/<scan_id>

Notes:
0xffsprung   3/26/24, 6:06 PM :
{“audit_network_errors”:0,”crawl_unique_locations_visited”:0,”crawl_and_audit_caption”:”Paused task “,”audit_queue_items_completed”:0,”crawl_requests_made”:1,”crawl_network_errors”:1,”audit_requests_made”:0,”crawl_requests_queued”:0,”issue_events”:0,”audit_queue_items_waiting”:0,”current_url”:””,”crawl_and_audit_progress”:-1,”total_elapsed_time”:30}

As you can see this metrics view is easier for our model to understand and does not disclose current engagement details this was passed to notes, while AI model Utilized TTS.

If you want full logs it can be achieved by setting the variable to “Contents of URL” for a full dictionary that will return a comprehensive log of the task.

we can now interact with burpsuite rest via macOS, iOS shortcuts along with all the list of actions for scripting easily added to home screen, dock, or watchface for execution.