

# SAVER\_LOGGER - Burp Suite Extension

---

Author: Sachhit Anasane

Date: 03 July 2025

## Introduction

SAVER\_LOGGER is a custom-built professional extension for Burp Suite Pro that logs every HTTP request across all Burp tools, including Proxy, Scanner, Repeater, Intruder, and Collaborator. It tracks host, request method, URL, status codes, request counts, insertion points, and time windows of the scanned endpoints. The extension is highly reliable and designed for enterprise-grade & personal use, including automatic log saving.

## Real-World Use Case & Critical Need

In modern security assessments and incident response scenarios, **proving the legitimacy and accuracy of testing logs** is crucial. Organizations often face:

- **Regulatory audits** requiring timestamped testing logs.
- **Client reporting** obligations with evidence of methodology
- **Post-breach investigations** needing exact attack footprints.
- **Technical failures** (e.g., Burp state file corruption, crashes)

Without dependable logs, credibility and accountability can be compromised.

**SAVER\_LOGGER** eliminates this risk by persistently recording and exporting logs — maintaining verifiable records of request activity, even if Burp is closed accidentally. It ensures **traceability, reproducibility, and legal defensibility** for your testing work.

## Key Features

### ✚ **Precise Request Tracking**

Logs every HTTP request sent from **all Burp tools**, including passive and active scanners, manual tools, and automated scans.

### ✚ **Full Request Metadata Logging**

Captures:

Serial No, Host, Request Method, URL, Status Code, Tool Name, Request Count, Insertion Point Count, Start Time, End Time

### ✚ **Active Scan Integration**

Counts and correlates **insertion points** used by Burp Scanner accurately — crucial for vulnerability reporting.

### ✚ **Auto-Save on Exit**

Automatically exports logs to a styled .csv file on **Desktop** when Burp is closed — even forcefully or accidentally.

### ✚ **Manual Export Capability**

Easy-to-use "**Export Log as CSV**" button from the SAVER\_LOGGER panel to manually save logs at any time.

### ✚ **Authenticity & Proof Block**

Every CSV export includes a metadata block to **prove origin and integrity**, with:

- # Generated By: SAVER\_LOGGER Burp Extension
- # Burp Suite Version: <autofetched>
- # Exported Timestamp: <autofetched>
- # Extension Runtime ID: <unique per session>

### ✚ **Human-Readable CSV Format**

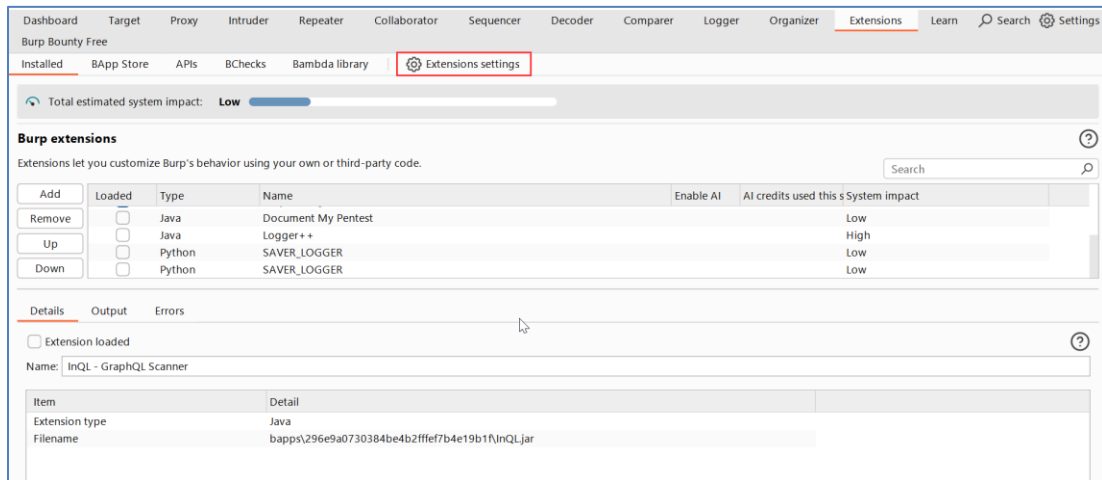
CSVs are saved in **UTF-8** encoding and are fully compatible with spreadsheet software like Excel, LibreOffice, or Google Sheets.

### ✚ **User Interface Panel**

Clean, informative panel under "**SAVER\_LOGGER**" tab in Burp Suite with feature info and export control.

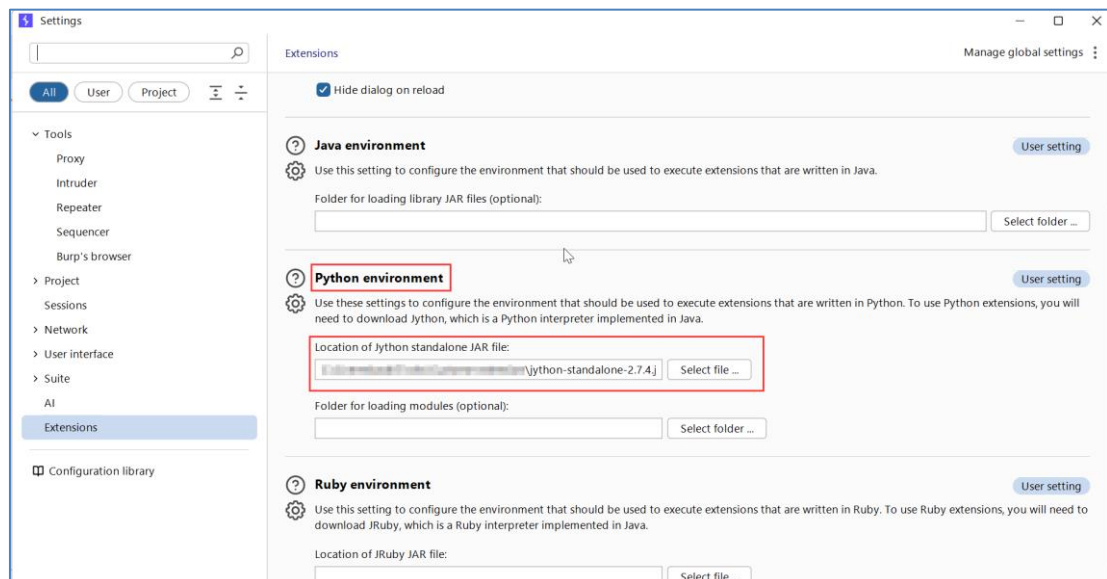
## Usage Instructions

1. Go to Extensions tab in burp suite, click on extension settings.



2. In Python Environment section, add a location for downloaded Jython Standalone JAR File.

(Download Link:- <https://repo1.maven.org/maven2/org/python/jython-standalone/2.7.4/jython-standalone-2.7.4.jar>)



3. Once Jython Jar file is loaded, go to Installed section and load the extension into Burp Suite.

DashboardTargetProxyIntruderRepeaterCollaboratorSequencerDecoderComparerLoggerOrganizerExtensionsLearnSearchSettings

Burp Bounty FreeSAVER\_LOGGER

InstalledBApp StoreAPIsBChecksBambda libraryExtensions settings

Total estimated system impact: Low

Burp extensions

Extensions let you customize Burp's behavior using your own or third-party code.

AddRemoveUpDown

LoadedType

Name

Enable AI

AI credits used this \$

System impact

	<input type="checkbox"/>	Java	Param Miner			Low
	<input checked="" type="checkbox"/>	Java	Burp Bounty, Scan Check Builder			Low
	<input type="checkbox"/>	Java	Document My Pentest			Low
	<input checked="" type="checkbox"/>	Python	SAVER_LOGGER			Low

DetailsOutputErrors

☒ Extension loaded

Name: SAVER\_LOGGER

Item	Detail
Extension type	Python
Filename	C:\Commback\Tools\Custom extension\Final Extension\SAVER_LOGGER.py
Method	registerExtenderCallbacks
Extension state listeners	1
HTTP handlers	1
Audit issue handlers	1

Event log (19)All issues

Memory: 255.0MB

Load Burp extension

Please enter the details of the extension, and how you would like to handle standard output and error.

Extension details

Extension type: Python

Extension file (py): jn\Final Extension\SAVER\_LOGGER.py

Select file ..

Standard output

☐ Output to system console

☐ Save to file:

Select file ..

☒ Show in UI

Standard error

☐ Output to system console

☐ Save to file:

Select file ..

☒ Show in UI

CancelNext

Load Burp extension

The extension loaded successfully. Any output or error messages generated are shown below.

OutputErrors

1

Clear

Search

0 highlights

Close

DashboardTargetProxyIntruderRepeaterCollaboratorSequencerDecoderComparerLoggerOrganizerExtensionsLearnSearchSettings

Burp Bounty FreeSAVER\_LOGGER

InstalledBApp StoreAPIsBChecksBambda libraryExtensions settings

Total estimated system impact: Low

Burp extensions

Extensions let you customize Burp's behavior using your own or third-party code.

AddRemoveUpDown

LoadedType

Name

Enable AI

AI credits used this \$

System impact

	<input type="checkbox"/>	Java	Param Miner			Low
	<input checked="" type="checkbox"/>	Java	Burp Bounty, Scan Check Builder			Low
	<input type="checkbox"/>	Java	Document My Pentest			Low
	<input checked="" type="checkbox"/>	Python	SAVER_LOGGER			Low

DetailsOutputErrors

☒ Extension loaded

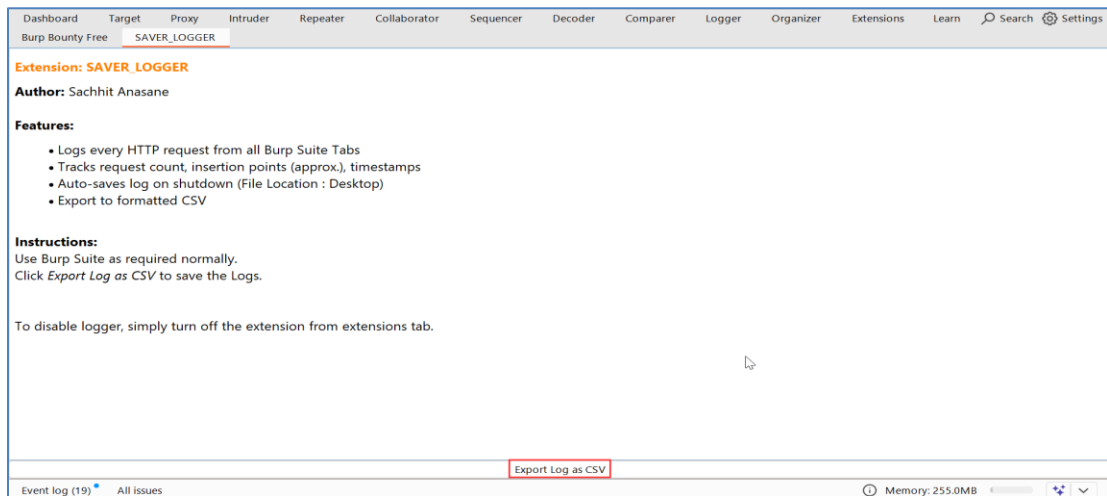
Name: SAVER\_LOGGER

Item	Detail
Extension type	Python
Filename	C:\Commback\Tools\Custom extension\Final Extension\SAVER_LOGGER.py
Method	registerExtenderCallbacks
Extension state listeners	1
HTTP handlers	1
Audit issue handlers	1

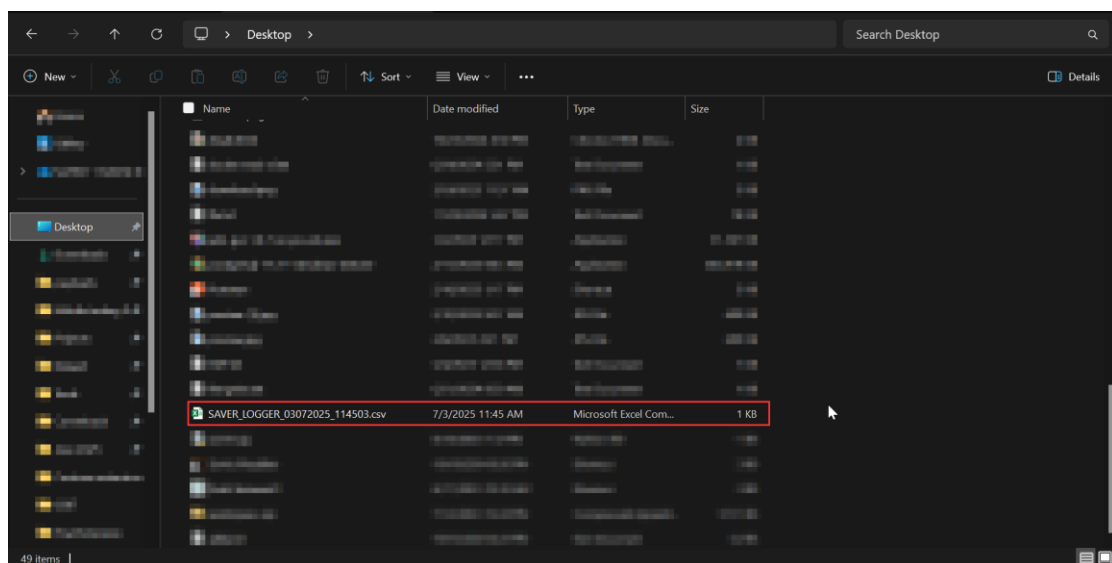
Event log (19)All issues

Memory: 255.0MB

4. Use Burp tools as normal usage (e.g., Proxy traffic, launch scans, Intruder, etc).
5. Go to the 'SAVER\_LOGGER' tab to view extension info.
6. Click 'Export Log as CSV' to save manually.



7. If Burp is closed unexpectedly, the logs are automatically saved to your Desktop.



The screenshot shows a Microsoft Excel spreadsheet with the following data:

Serial No	Host	Request Method	Status	Tool Name	Request Count	Insertion Point	Start Time	End Time
1	XYZ.ABC.COM	GET	https	200 Repeater	1	0	7/3/2025 11:52	7/3/2025 11:52
2	XYZ.ABC.COM	GET	https	200 Repeater	2	0	7/3/2025 11:52	7/3/2025 11:52
3	XYZ.ABC.COM	GET	https	200 Repeater	3	0	7/3/2025 11:52	7/3/2025 11:52
4	XYZ.ABC.COM	GET	https	200 Repeater	4	0	7/3/2025 11:52	7/3/2025 11:52
5	XYZ.ABC.COM	GET	https	200 Repeater	5	0	7/3/2025 11:52	7/3/2025 11:52

Below the table, the following text is displayed:

# Generated By: SAVER\_LOGGER Burp Extension  
# Burp Suite Version: Burp Suite Professional 2025.3.3  
# Exported Timestamp: 2025-07-03 11:52:37  
# Extension Runtime ID: 4bdcf7e-e02d-4188-bf16-1dd63666f920

### CSV Export Format

The CSV includes the following structured columns:

Column	Description
Serial No	Incremental number of requests
Host	Target host
Request Method	HTTP method (GET, POST, etc.)
URL	Complete request URL
Status Code	Response status code (e.g., 200, 404)
Tool Name	Source tool in Burp Suite
Request Count	Number of requests sent to the URL
Insertion Point Count	Parameters/insertion points identified
Start Time	Timestamp of the first request to that URL
End Time	Timestamp of the last request to that URL