AICS with IBM Qrador

Assignment-4

Name- Shaunak Tanawade
Reg. No- 21BCE8843
Campus- VIT-AP

Burp Suite

Burp Suite is a comprehensive platform for evaluating the security of web applications. From the initial mapping and analysis of an application's attack surface through the discovery and exploitation of security flaws, its numerous tools work in perfect harmony to assist the whole testing process.

System requirements

CPU cores / memory

- **Minimum: 2x cores, 4GB RAM** - This spec is suitable for basic tasks such as proxying web traffic and simple Intruder attacks. While Burp Suite may run on a machine with a lower specification than this, we do not recommend doing sofor performance reasons.
- **Recommended: 2x cores, 16GB RAM** - This is a good general-purposespec.
- **Advanced: 4x cores, 32GB RAM** - This spec is suitable for more intensivetasks, such as complex Intruder attacks or large automated scans.

Free disk space

- Basic installation: **1GB**
- Per project file: **2GB**

Operating system and architecture

Burp Suite supports the latest versions of the following operating systems:

- Windows (Intel 64-bit)
- Linux (Intel and ARM 64-bit)
- OS X (Intel 64-bit and Apple M1)

Download and install

Step 1: Download

Use the links below to download the latest version of Burp Suite Professional or Community Edition

Professional version: https://portswigger.net/burp/releases/professional/latestCommunity version:

https://portswigger.net/burp/releases/community/latest

Step 2: Install

Run the installer and launch Burp Suite.

When asked to select a project file and configuration, just click **Next** and then **Start Burp** to skip this for now.

Note

If you're using Burp Suite Professional, enter your license key when prompted. If you don't have one already, you can subscribe or request a free trial.

Step 3: Start exploring Burp Suite

Burp Suite tools

Burp Suite contains various tools for performing different testing tasks. The tools operate effectively together, and you can pass interesting requests between tools as your work progresses, to carry out different actions.

- **Target** - This tool contains detailed information about your target applications, and lets you drive the process of testing for vulnerabilities.
- **Burp's browser** - This browser is preconfigured to work with the full functionality of Burp Suite right out of the box.
- **Proxy** - This is an intercepting web proxy that operates as a man-in-the-middle between the end browser and the target web application. It lets you intercept, inspect and modify the raw traffic passing in both directions.
- **Scanner** Professional - This is an advanced web vulnerability scanner, which can automatically crawl content and audit for numerous types of vulnerabilities.

- **Intruder** - This is a powerful tool for carrying out automated customized attacks against web applications. It is highly configurable and can be used to perform a wide range of tasks to make your testing faster and more effective.
- **Repeater** - This is a tool for manually manipulating and resending individual messages, and analyzing the application's responses.
- **Sequencer** - This is a sophisticated tool for analyzing the quality of randomness in an application's session tokens or other important data items that are intended to be unpredictable.
- **Decoder** - This is a useful tool for performing manual or automated decoding and encoding of application data.
- **Comparer** - This is a handy utility for performing a visual "diff" between any two items of data, such as pairs of similar HTTP messages.
- **Logger** - This is a tool for recording and analyzing HTTP traffic that Burp Suite generates.
- **Inspector** - This provides some useful features for analyzing and editing HTTP and WebSockets messages.
- **Collaborator** Professional - This is a manual tool for identifying out-of-band vulnerabilities.
- **DOM Invader** - This is a tool for finding DOM XSS vulnerabilities.
- **Clickbandit** - This is a tool for generating Clickjacking attacks.
- **Message editor** - This is a tool for viewing and editing HTTP requests and responses throughout Burp.
- **Engagement tools** Professional - Configure various engagement-related tasks.
- **Search** - This is a tool for performing searches in Burp Suite.
- **Infiltrator** - This is a tool for detecting whether Burp's input is passed to potentially unsafe APIs.
- **Organizer** - This is a tool for storing and annotating HTTP messages that you want to investigate later.

Features

Manual penetration testing features

Intercept everything your browser sees

Burp Suite's built-in browser works right out of the box - enabling you to modify every HTTP message that passes through it.

Quickly assess your target

Determine the size of your target application. Auto-enumeration of static and dynamic URLs, and URL parameters.

Speed up granular workflows

Modify and reissue individual HTTP and WebSocket messages, and analyze the response - within a single window.

Manage recon data

All target data is aggregated and stored in a target site map - with filtering and annotation functions.

Expose hidden attack surface

Find hidden target functionality with an advanced automatic discovery function for "invisible" content.

Break HTTPS effectively

Proxy even secure HTTPS traffic, using Burp Suite's built-in instrumented browser.

Work with HTTP/2

Burp Suite offers unrivaled support for HTTP/2-based testing - enabling you to work with HTTP/2 requests in ways that other tools cannot.

Work with WebSockets

WebSockets messages get their own specific history - allowing you to view and modify them.

Manually test for out-of-band vulnerabilities

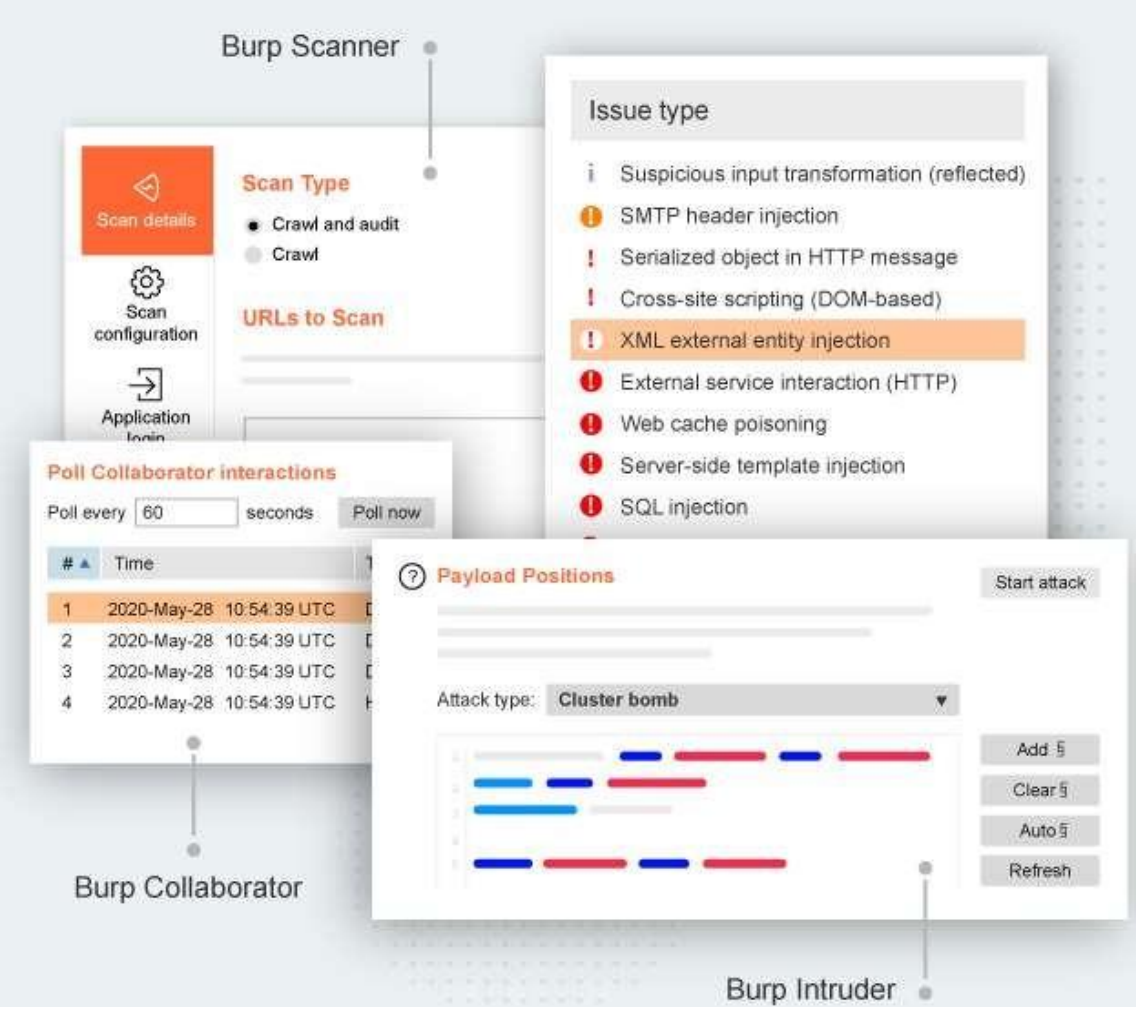
Make use of a dedicated client to incorporate Burp Suite's out-of-band (OAST) capabilities during manual testing.

DOM Invader

Use Burp Suite's built-in browser to test for DOM XSS vulnerabilities more easily - with DOM Invader.

Assess token strength

Easily test the quality of randomness in data items intended to be unpredictable (e.g. tokens).



Advanced / custom automated attacks

Faster brute-forcing and fuzzing

Deploy custom sequences of HTTP requests containing multiple payload sets. Radically reduce time spent on many tasks.

Query automated attack results

Capture automated results in customized tables, then filter and annotate to find interesting entries / improve subsequent attacks.

Construct CSRF exploits

Easily generate CSRF proof-of-concept attacks. Select any suitable request to generate exploit HTML.

Facilitate deeper manual testing

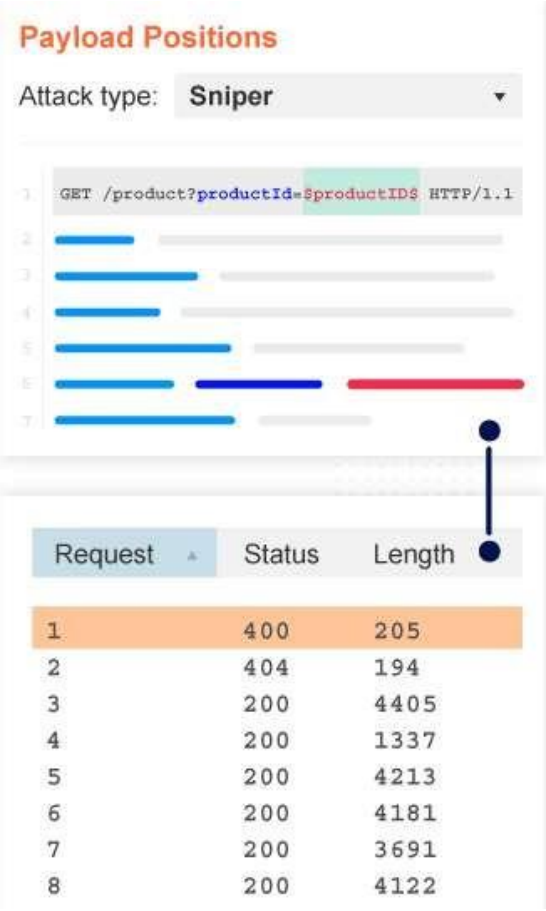
See reflected / stored inputs even when a bug is not confirmed. Facilitates testing for issues like XSS.

Scan as you browse

The option to passively scan every request you make, or to perform active scans on specific URLs.

Automatically modify HTTP messages

Settings to automatically modify responses. Match and replace rules for both responses and requests.



Automated scanning for vulnerabilities

Browser powered scanning

Burp Scanner uses its embedded browser to render its target - enabling it to navigate even complex single-page applications (SPAs).

Harness pioneering OAST technology

High signal: low noise. Scan with pioneering, friction-free, out-of-band-application security testing (OAST).

Remediate bugs effectively

Custom descriptions and step-by-step remediation advice for every bug,from PortSwigger Research and the Web Security Academy.

Fuel vulnerability coverage with research

Cutting-edge scan logic from PortSwigger Research combines with coverage of over 100 generic bugs.

BChecks

Create custom scan checks for Burp Scanner, written in a simple text-based language.

API scanning

Discover more potential attack surface. Burp Scanner parses JSON or YAML API definitions - scanning any API endpoints it finds.

Authenticated scanning

Scan privileged areas of target applications, even if they use complex login mechanisms like single sign-on (SSO).

Conquer client-side attack surfaces

A built-in JavaScript analysis engine help to find holes in client-side attack surfaces.

Configure scan behavior

Customize what you audit, and how. Skip specific checks, fine-tune insertion points, and much more. Or use preset scan modes to get an overview.

Productivity tools

Deep-dive message analysis

Show follow-up, analysis, reference, discovery, and remediation in a feature-rich HTTP editor.

Utilize both built-in and custom configurations

Access predefined configurations for common tasks, or save and reuse custom configurations.

Project files

Auto-save everything you do while on an engagement, as well as the configuration settings you used.

Burp Logger

See every HTTP message that passes through Burp Suite's tools - all in one place - with Burp Logger.

Speed up data transformation

Decode or encode data, with multiple built-in operations (e.g. Hex, Octal, Base64).

Burp Organizer

Store and annotate interesting messages you find while testing, so you can come back to them later.

Make code more readable

Automatically pretty-print code formats including JSON, JavaScript, CSS, HTML, and XML.

Easily remediate scan results

See source, discovery, contents, and remediation, for every bug, with aggregated application data.

Search function

Search everywhere in Burp Suite Professional at once, with its powerful search function.

Simplify scan reporting

Customize with HTML / XML formats. Report all evidence identified, including issue details.

BApp extensions

Create custom extensions

The Montoya API ensures universal adaptability. Code custom extensions to make Burp work for you.

Hackvertor

Convert between various encodings with Hackvertor. Use multiple nested tags to perform layered encoding. Even execute your own code with custom tags - and more.

Autorize

When testing for authorization vulnerabilities, save time and perform repeat requests with Autorize.

Turbo Intruder

Configured in Python, with a custom HTTP stack, Turbo Intruder can unleash thousands of requests per second.

J2EE Scan

Expand your Java-specific vulnerability catalogue and hunt the most niche bugs, with J2EEScan.

Access the extension library

The BApp Store customizes and extends capabilities. Over 250 extensions, written and tested by Burp users.

Upload Scanner

Adapt Burp Scanner's attacks by uploading and testing multiple file-type payloads, with Upload Scanner.

HTTP Request Smuggler

Scan for request smuggling vulnerabilities - and exploit them more easily by having HTTP Request Smuggler tweak offsets automatically for you.

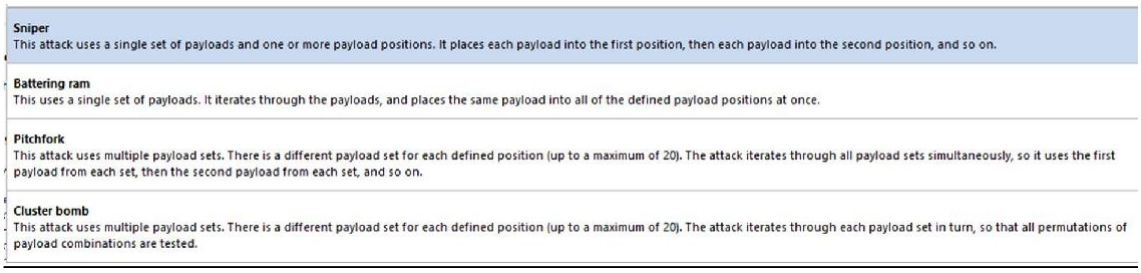
Param Miner

Quickly find unkeyed inputs with Param Miner - can guess up to 65,000 parameter names per second.

Backslash Powered Scanner

Find research-grade bugs, and bridge human intuition and automation, with Backslash Powered Scanner.

Intruder Attack Type



To determine the way in which payloads are assigned to payload positions, you can specify an attack type. Attack types enable you to configure whether:

- Payloads are taken from a single set, or multiple sets (up to 20).
- Payloads are assigned to payload positions in turn, or simultaneously.

To select an attack type, go to **Intruder > Positions**, and click on the drop-down listunder **Choose an attack type**.

Sniper

This attack places each payload into each payload position in turn. It uses a single payload set.

The total number of requests generated in the attack is the product of the number of positions and the number of payloads in the payload set.

The Sniper attack is useful for fuzzing a number of request parameters individually for common vulnerabilities.

Battering ram

This attack places the same payload into all of the defined payload positions simultaneously. It uses a single payload set.

The total number of requests generated in the attack is the number of payloads in the payload set.

The Battering ram attack is useful where an attack requires the same input to be inserted in multiple places within the request. For example, a username within a cookie and a body parameter.

Pitchfork

This attack iterates through a different payload set for each defined position. Payloads are placed into each position simultaneously. For example, the first three requests would be:

- Request one:
    - Position 1 = First payload from Set 1.
    - Position 2 = First payload from Set 2.
- Request two:
    - Position 1 = Second payload from Set 1.
    - Position 2 = Second payload from Set 2.
- Request three:

    - Position 1 = Third payload from Set 1.
    - Position 2 = Third payload from Set 2.

The total number of requests generated in the attack is the number of payloads in the smallest payload set.

The Pitchfork attack is useful where an attack requires different but related input to be inserted in multiple places within the request. For example, to place a username in one parameter, and a known ID number corresponding to that username in another parameter.

Cluster bomb

This attack iterates through a different payload set for each defined position. Payloads are placed from each set in turn, so that all payload combinations are tested. For example, the first three requests would be:

- Request one:
    - Position 1 = First payload from Set 1.
    - Position 2 = First payload from Set 2.
- Request two:
    - Position 1 = First payload from Set 1.
    - Position 2 = Second payload from Set 2.
- Request three:
    - Position 1 = First payload from Set 1.
    - Position 2 = Third payload from Set 2.

The total number of requests generated in the attack is the product of the number of payloads in all defined payload sets - this may be extremely large.

The Cluster bomb attack is useful where an attack requires unrelated or unknown input to be inserted in multiple places within the request. For example, when guessing both a username and password.

Burpsuite screenshot