



# Burpsuite



# Security Testing



@cybertechvt



@cybertechvt



cybertechvt.com

# Disclaimer/Warning

The Kali Guides provided to you by the Cyber Tech Awareness team are meant for educational purposes ONLY.

The tools covered in the Kali Guides can be used for malicious purposes, but should not be used as such.

The CyberTech Awareness team and the Leahy Center for Digital Forensics and Cybersecurity is NOT responsible any malicious activity conducted with aid from these Kali Guides.

Only use this tool on domains, addresses, files and resources that you have permission to.

## Brief Summary

A suite of tools used for **penetration testing web applications** to check for exploitable security vulnerabilities. Its tools are capable of capturing data, converting that data into different formats, scanning for vulnerabilities, and more.

## Helpful Vocabulary

---

### **Penetration Testing:**

- A form of ethical hacking where a hacker tests a system's security by simulating an attack, finding the vulnerabilities and exploits that could harm the system, but without causing any damage and then documenting and reporting the findings.

### **Network Protocol:**

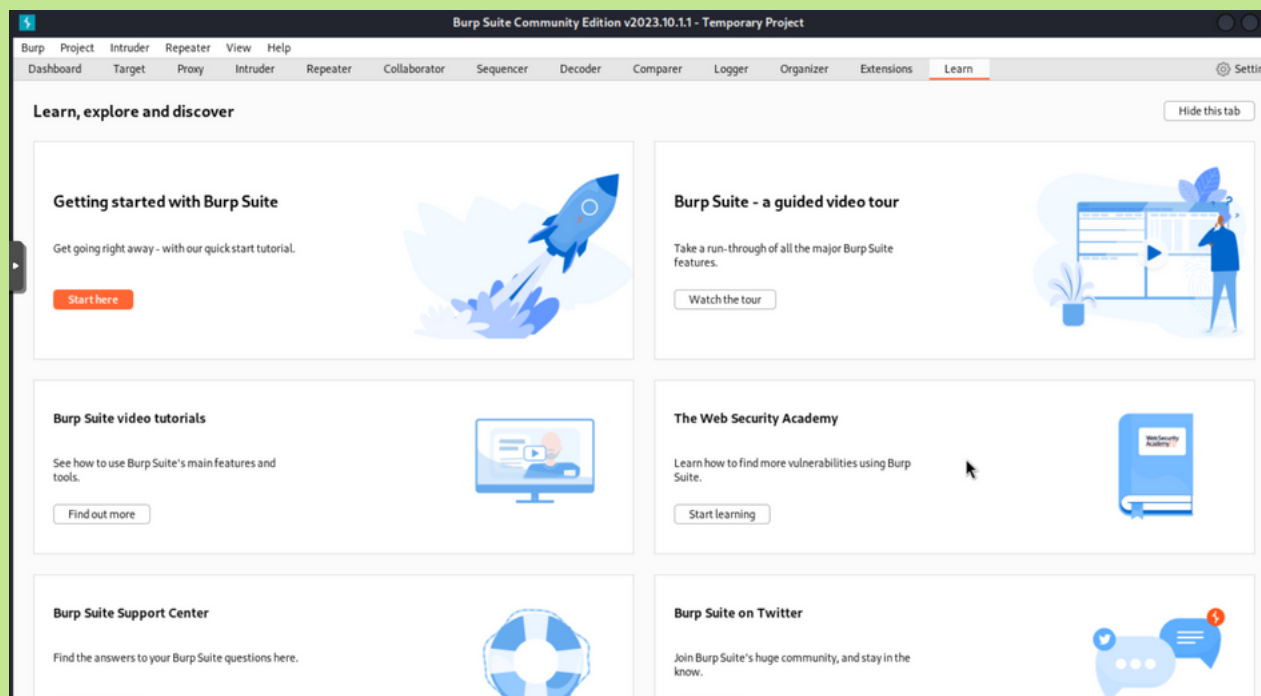
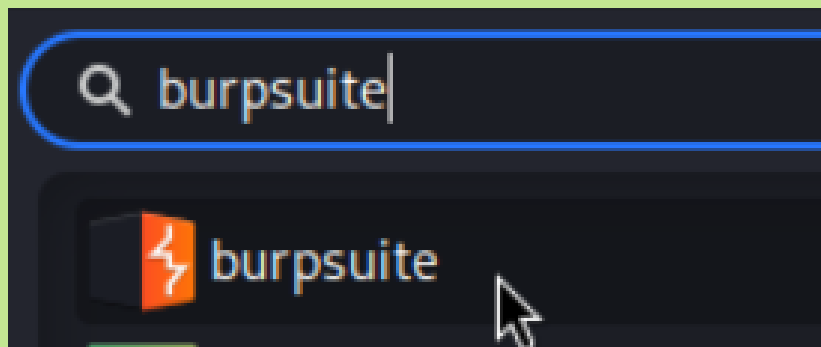
- A universal language for communication between different devices. Each protocol has a set of rules that regulates communications and/or data processing across networks.

### **HTTP request:**

- A network protocol made by a client that is delivered to a server to access a resource on the server. HTTP requests are deployed when accessing content (text, images, video) on the web!

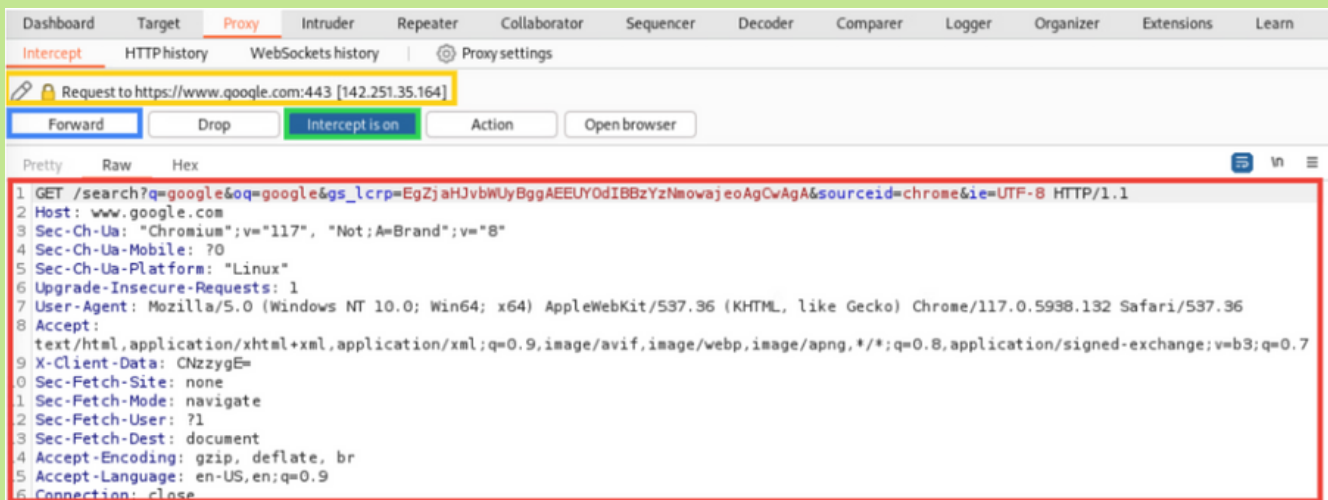
# How to Access Burpsuite

Burpsuite is an application on your desktop, so you can type in “burpsuite” in your terminal to access it, or you can click on the application itself.



This is the burpsuite window that should pop up!

# Example 1: Proxy HTTP Request Intercept and Alteration



## What is this command doing?

The user is intercepting an HTTP request within a browser and forwarding the information to be analyzed later. While this intercept is occurring, the client on the browser will not be able to interact with browser elements until the intercept is stopped.

### Request to \_

- Specifies the HTTP request being intercepted

### Forward

- Each time this button is pressed, information is saved about the intercepted request that will then appear under the “HTTP history” tab to be viewed

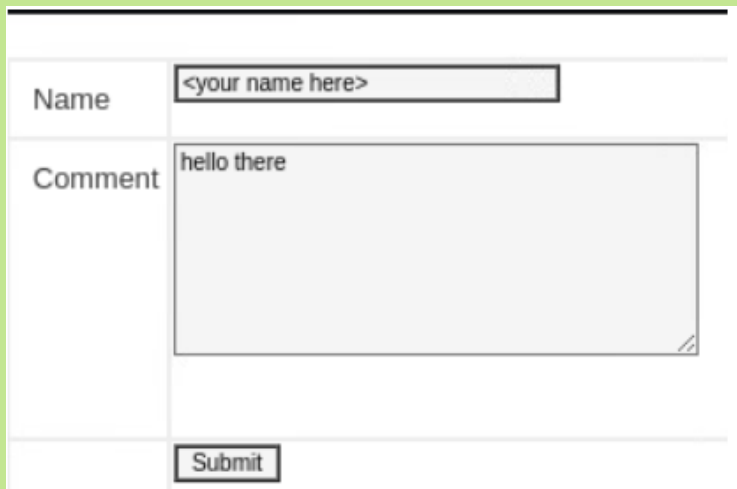
### Intercept is on

- Pauses the request on the client’s side until turned off

### Raw data

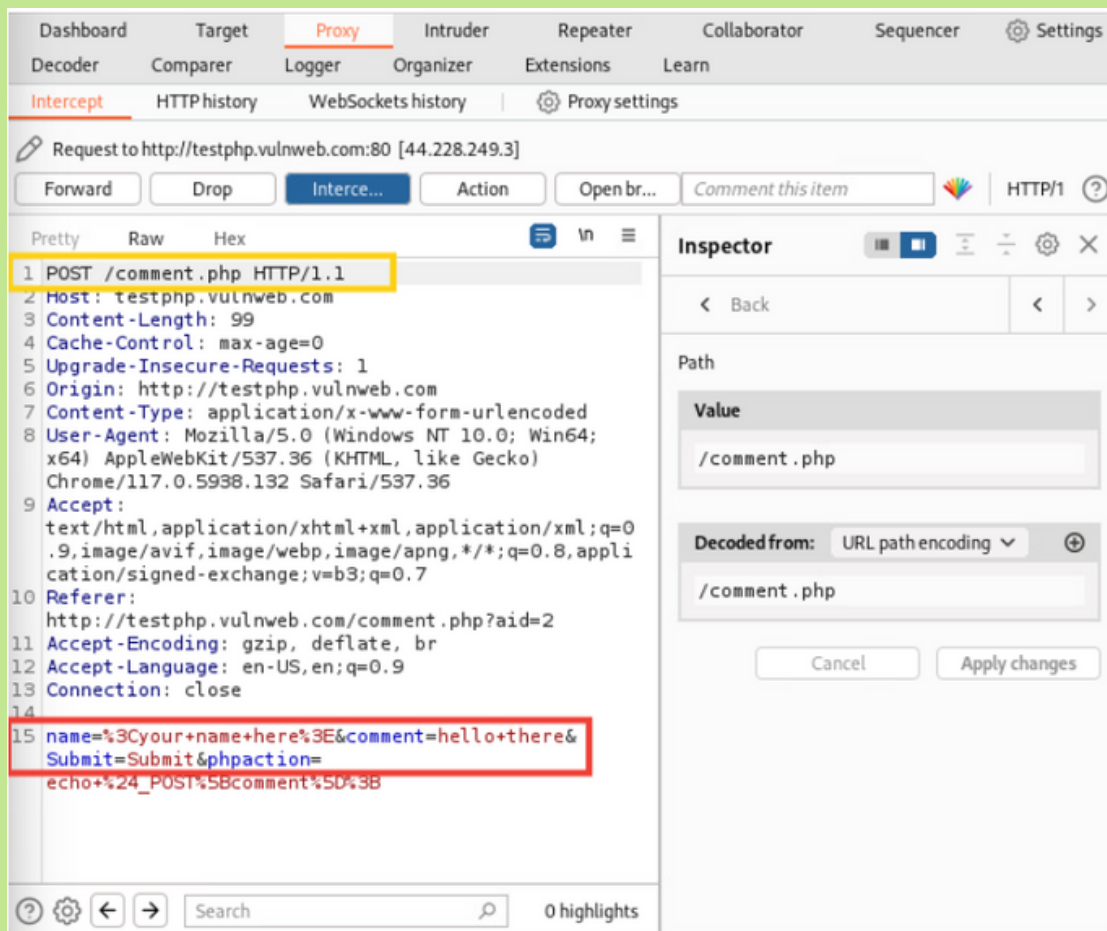
- Displays many details about the intercepted request: host, user-agent, etc

# Example 1: Proxy HTTP Request Intercept and Alteration



A screenshot of a web form. The form has two main sections: 'Name' and 'Comment'. The 'Name' field contains the placeholder text '<your name here>'. The 'Comment' field is a text area containing the text 'hello there'. Below the text area is a 'Submit' button.

This image shows the request that is being intercepted- to post a comment that reads "hello there".

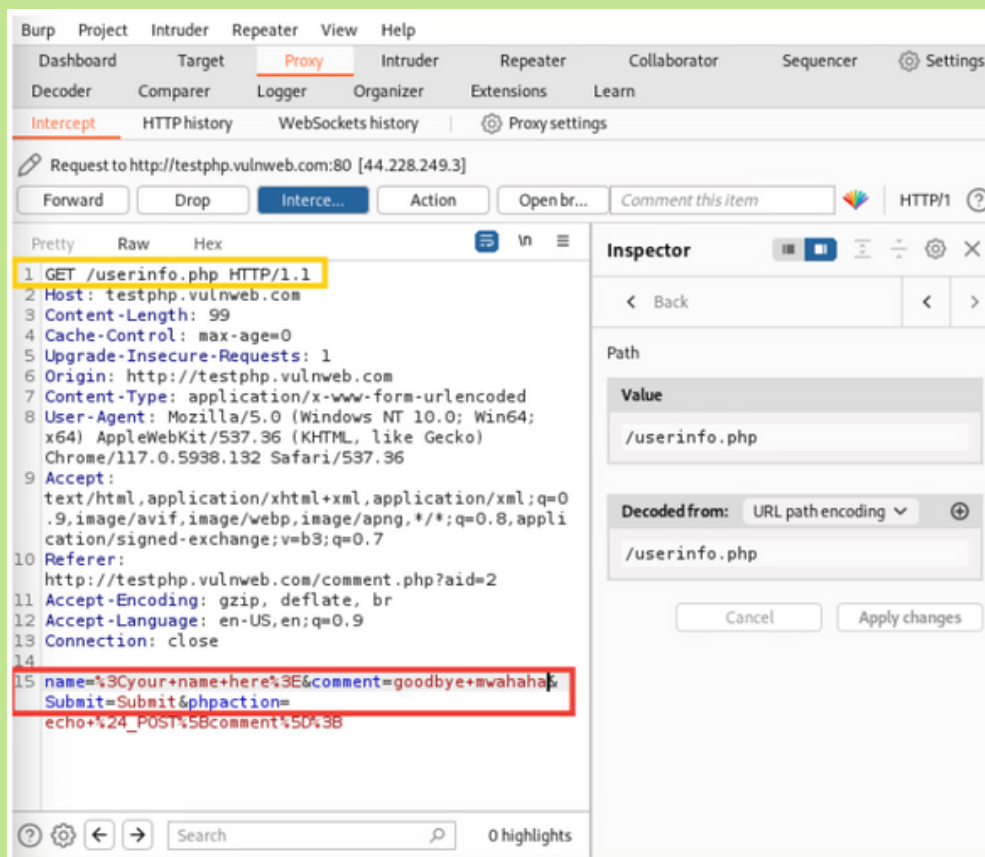


A screenshot of the Burp Suite interface showing an intercepted HTTP request. The 'Proxy' tab is active, and the 'Intercept' button is highlighted. The request is for the URL `http://testphp.vulnweb.com:80 [44.228.249.3]`. The request details are shown in the 'Raw' view, and the 'Inspector' panel is open on the right, showing the path `/comment.php`.

```
1 POST /comment.php HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Length: 99
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://testphp.vulnweb.com
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.5938.132 Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
10 Referer: http://testphp.vulnweb.com/comment.php?aid=2
11 Accept-Encoding: gzip, deflate, br
12 Accept-Language: en-US,en;q=0.9
13 Connection: close
14
15 name=%3Cyour+name+here%3E&comment=hello+there&Submit=Submit&phfaction=echo+%24_POST%5Bcomment%5D%3B
```

The second image is the intercepted request, where the user can not only see the intercepted request but can type directly into the box to change the request as they wish.

# Example 1: Proxy HTTP Request Intercept and Alteration



This third image is the alterations that the user made to the request to make the comment read “goodbye mwahaha” and to redirect the user to a page that will not actually post the comment.

## What is this command doing?

Whereas the first example showed a user intercepting an HTTP request, this example shows the user altering the request after intercepting it.

When the intercept is turned off, the page will load on the web page as the altered version.



## Example 2: Burpsuite Intruder

The screenshot displays the Burp Suite Intruder interface. On the left, the 'Payload sets' section is active, showing a 'Simple list' type with a 'Payload count' of 8 and a 'Request count' of 8. A list of usernames (admin, test, julia, sarah, tom, ben, jerry, selene) is visible in the 'Payload settings [Simple list]' section. On the right, the 'Attack' tab shows a table of results for 8 requests, all with a status code of 302 and a length of 258. Below the table, a 'Request' tab shows the raw HTTP request details for the first request (index 1), including headers like 'Host: testphp.vulnweb.com' and 'User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.5998.132 Safari/537.36'.

Request	Payload	Status code	Error	Timeout	Length	Comment
0		302	<input type="checkbox"/>	<input type="checkbox"/>	258	
1	admin	302	<input type="checkbox"/>	<input type="checkbox"/>	258	
2	test	302	<input type="checkbox"/>	<input type="checkbox"/>	258	
3	julia	302	<input type="checkbox"/>	<input type="checkbox"/>	258	
4	sarah	302	<input type="checkbox"/>	<input type="checkbox"/>	258	
5	tom	302	<input type="checkbox"/>	<input type="checkbox"/>	258	
6	ben	302	<input type="checkbox"/>	<input type="checkbox"/>	258	
7	jerry	302	<input type="checkbox"/>	<input type="checkbox"/>	258	
8	selene	302	<input type="checkbox"/>	<input type="checkbox"/>	258	

### What is this command doing?

Using Burp Intruder, a command is being run to test if a list of usernames are valid within a specific website.

### Payload settings

- Includes the list of usernames being tested. Can also be set up to scan for passwords.

### Results

- What a hacker may be looking at is the Length column which shows the response time from a server. If there is an abnormal response time, it likely means that the server processed it differently than the other inputs

### Request

- A further inspection of a specific request after a specific username input

## Additional Resources:

### **Kali Linux Burpsuite Official Webpage:**

<https://www.kali.org/tools/burpsuite/>

### **Burp Suite documentation:**

<https://portswigger.net/burp/documentation>

### **Getting started with Burp Suite:**

<https://portswigger.net/burp/documentation/desktop/getting-started>

### **Burp Suite Training:**

<https://portswigger.net/training>

### **Burp Suite Guide:**

<https://burpsuite.guide/>