**SAD Lab**

**EXPERIMENT NO. 6**

**Aim**: Use Burp proxy to test web applications.

**Theory**:

1. What is HTTP Protocol?

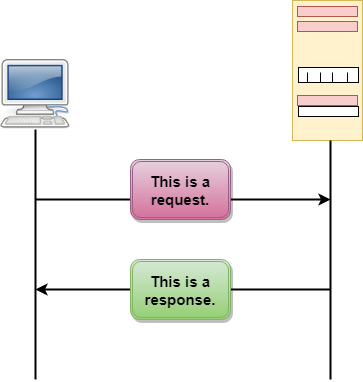
HTTP stands for HyperText Transfer Protocol. It is a protocol used to access the data on the World Wide Web. The HTTP protocol can be used to transfer the data in the form of plain text, hypertext, audio, video, and so on. This protocol is known as HyperText Transfer Protocol because of its efficiency that allows us to use it in a hypertext environment where there are rapid jumps from one document to another document.

HTTP is similar to the FTP as it also transfers the files from one host to another host. But, HTTP is simpler than FTP as HTTP uses only one connection, i.e., no control connection to transfer the files. HTTP is used to carry the data in the form of MIME-like format. HTTP is similar to SMTP as the data is transferred between client and server. The HTTP differs from the SMTP in the way the messages are sent from the client to the server and from server to the client. SMTP messages are stored and forwarded while HTTP messages are delivered immediately.

**Features of HTTP**:

1. **Connectionless protocol**: HTTP is a connectionless protocol. HTTP client initiates a request and waits for a response from the server. When the server receives the request, the server processes the request and sends back the response to the HTTP client after which the client disconnects the connection. The connection between client and server exists only during the current request and response time only.
2. **Media independent**: HTTP protocol is media independent as data can be sent as long as both the client and server know how to handle the data content. It is required for both the client and server to specify the content type in the MIME-type header.
3. **Stateless**: HTTP is a stateless protocol as both the client and server know each other only during the current request. Due to this nature of the protocol, both the client and server do not retain the information between various requests of the web pages.

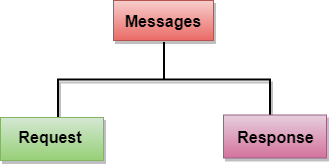
**HTTP Transactions**



The above figure shows the HTTP transaction between client and server. The client initiates a transaction by sending a request message to the server. The server replies to the request message by sending a response message.

**HTTP Messages**

HTTP messages are of two types: request and response. Both the message types follow the same message format.



1. **Request Message**: The request message is sent by the client that consists of a request line, headers, and sometimes a body.
2. **Response Message**: The response message is sent by the server to the client that consists of a status line, headers, and sometimes a body.
3. What are the Owasp Web Security Testing Guidelines?

OWASP has compiled a wealth of cybersecurity knowledge and know-how from both professionals and volunteers. The result: the Web Security Testing Guide (WSTG) an excellent guide for web application testing that anyone can read and consult with freely. The OWASP Web Security Testing Guide (WSTG) is a comprehensive guide for testing the security of web applications. It describes techniques, methods, tools and resources for testing most common web application security issues.

WSTG’s current version is 4.2. It is web-hosted and also has a PDF document version. You can also access version 5.0 that is currently being developed and is continually updated. Since new attack vectors and new methods for attacking web applications are discovered daily, we recommend using this latest online version.

**Topics of WSTG**

* **Information Gathering**: In this section the guide includes tests to obtain information about the web server, the application and the application architecture.
* **Configuration and Deployment Management Testing**: Tests for Network configuration, Application Configuration, HTTP enumeration and recently also topics like Subdomain Takeover and Cloud Storage. This is an example of how complete and updated this guide is!
* **Identity Management Testing**: Tests for role definitions, user registration and account enumeration.
* **Authentication Testing**: For Default Credentials, Weak Lock Out, Weak Password Policy and so on.
* **Authorization Testing**: Directory Traversal File Include, Privilege Escalation and Insecure Direct Object References (IDOR).
* **Session Management Testing**: Tests for Session Fixation, Cross Site Request Forgery, Session Hijacking and JSON Web Tokens.
* **Input Validation Testing**: Covering Cross-Site Scripting (reflected and stored), all types of Injection including Server-Side Template Injection, File Inclusion and Server-Side Request Forgery.
* **Testing for Weak Cryptography**: Covers Weak Transport Layer Security and Weak Encryption.
* **Business Logic Testing**: For the tester to find flaws in business logic. Here is an example: the application needs the user to execute step 1 to continue with step 2. If we find a to execute step 2 without doing step 1, this is a flaw. There is no specific method you can apply to find this kind of vulnerability, but WSTG describes procedures for testing.
* **Client-Side Testing**: Examples are: DOM Cross-Site Scripting, HTML and CSS injection, Cross Origin Resource Sharing (CORS) and Clickjacking.

**Integrating WSTG to Your SDLC**

Here are my recommendations for a successful WSTG implementation:

1. **Security Training**: Developer and testing teams should be trained periodically on the WSTG, and the tools selected for the tests. Although WSTG is testing oriented, the guide includes recommendations and useful documentation for developers.
2. **Prioritize your Tests**: It is impossible to apply all WSTG tests at the beginning of your integration project. First, you should choose the tests that are appropriate for your organization and can answer the bigger risks. Give your team time for mastering the tests. Then, gradually add more tests to your SDLC according to your organization’s resources. Take into consideration that your development team will need to remediate these issues, which will also affect your software development process.

To identify which test are suited for your organization to initiate your WSTG project with, we recommend choosing:

* Tests that are related to your application architecture and the information the organization protects. For example, if your application uses S3 buckets, then it is relevant to include Test Cloud Storage in your scope.
* Tests related to OWASP Top Ten. These will cover the most relevant vulnerabilities in applications.
* Tests in which the testing team has more expertise. As the testing team get trained and has more experience they can gradually add more tests in their scope.
* Implement WSTG with other OWASP projects – The WSTG focuses on application testing, but OWASP has projects for different SDLC phases in addition to testing.
* Assign a WSTG Champion, a member of your team that is the most skilled and knowledgeable in WSTG. This role will help all teams to support your implementation project.
* Periodically monitor your results and report them to management. Indicators like “Number of vulnerabilities found by severity” can show you how the testing team is finding real vulnerabilities and how security practices are improving the security of your web applications.

1. What is VAPT Testing? Tools for VAPT testing.

Vulnerability Assessment and Penetration Testing (VAPT) is a term used to describe security testing that is designed to identify and help address cyber security vulnerabilities. The meaning of VAPT can vary from one geographical region to another, either as a bracket for multiple distinct services, or a single, combined offering. VAPT as a whole could include anything from automated vulnerability assessments to human-led penetration testing and red team operations.

**Need of VAPT**

The evolving tools, tactics and procedures used by cybercriminals to breach networks means that it’s important to regularly test your organization’s cyber security.

VAPT helps to protect your organization by providing visibility of security weaknesses and guidance to address them. VAPT is increasingly important for organizations wanting to achieve compliance with standards including the GDPR, ISO 27001 and PCI DSS.

**Tools of VAPT**

1. **Astra**: Astra’s VAPT scan analyzes the entire application and its underlying infrastructure, including all network devices, management systems, and other components. It’s a deep analysis that helps you find security weaknesses, so you can fix them before a hacker does.
2. **OWASP Zap**: works by starting a web server and then interacting with the application through a proxy. This allows it to automatically and dynamically discover and scan hidden parameters and cookies, and other content that is not otherwise visible. The resulting data is then displayed in a clear and concise format so that you can see what you should be protecting. ZAP works on any platform and any technology.
3. **Nmap**: Nmap uses raw IP packets in novel ways to determine what hosts are available on the network, what services (application name and version) those hosts are offering, what operating systems (and OS versions) they are running, what type of packet filters/firewalls are in use, and dozens of other characteristics. It was designed to scan large networks but works fine against single hosts rapidly.
4. **Metasploit**: HD Moore created the Metasploit Project to provide the security community with a public resource for information on security vulnerabilities. The project provides information about security vulnerabilities used by penetration testers during security audits and network administrators to ensure the correct configuration of the network’s devices.
5. **Burp Suite**: Burp Suite is an integrated platform for performing security testing of web applications. The software comprises multiple tools which are used to test applications for security vulnerabilities.
6. **Wireshark**: Wireshark is a network traffic analyzer, monitoring software that allows you to see what traffic flows through your system network. It is open-source and is the most popular network analyzer in the world. Network administrators and professionals mainly use it to troubleshoot network and system performance issues and monitor and filter different network protocols.
7. **Nikito**: Nikto is a web server scanner that performs comprehensive tests against web servers for multiple items, including over 3300 potentially dangerous files/CGIs, checks for outdated server versions, and version-specific problems on over 270 server-side applications.
8. What is Burp Suite?

Burp or Burp Suite is a set of tools used for penetration testing of web applications. It is developed by the company named Portswigger, which is also the alias of its founder Dafydd Stuttard. BurpSuite aims to be an all in one set of tools and its capabilities can be enhanced by installing add-ons that are called BApps.

**Features of Burp Suite**

Features of Burp Suite are as follows:

* Easy to use and user friendly
* Numerous extensibility points allow you to tailor Burp to specific needs
* Over 1000 plugins that enable you to find and exploit specific vulnerabilities

**Tools offered by Burp Suite**

1. **Spider**: It is a web spider/crawler that is used to map the target web application. The objective of the mapping is to get a list of endpoints so that their functionality can be observed and potential vulnerabilities can be found. Spidering is done for the simple reason that the more endpoints you gather during your recon process, the more attack surfaces you possess during your actual testing.
2. **Proxy**: BurpSuite contains an intercepting proxy that lets the user see and modify the contents of requests and responses while they are in transit. It also lets the user send the request/response under monitoring to another relevant tool in BurpSuite, removing the burden of copy-paste. The proxy server can be adjusted to run on a specific loop-back ip and a port. The proxy can also be configured to filter out specific types of request-response pairs.
3. **Intruder**: It is a fuzzer. This is used to run a set of values through an input point. The values are run and the output is observed for success/failure and content length. Usually, an anomaly results in a change in response code or content length of the response. BurpSuite allows brute-force, dictionary file and single values for its payload position.
4. **Repeater**: Repeater lets a user send requests repeatedly with manual modifications. It is used for verifying whether the user-supplied values are being verified.
5. **Sequencer**: The sequencer is an entropy checker that checks for the randomness of tokens generated by the webserver. These tokens are generally used for authentication in sensitive operations: cookies and anti-CSRF tokens are examples of such tokens. Ideally, these tokens must be generated in a fully random manner so that the probability of appearance of each possible character at a position is distributed uniformly.
6. **Decoder**: Decoder lists the common encoding methods like URL, HTML, Base64, Hex, etc. This tool comes handy when looking for chunks of data in values of parameters or headers. It is also used for payload construction for various vulnerability classes. It is used to uncover primary cases of IDOR and session hijacking.
7. **Extender**: BurpSuite supports external components to be integrated into the tools suite to enhance its capabilities. These external components are called BApps. These work just like browser extensions. These can be viewed, modified, installed, uninstalled in the Extender window. Some of them are supported on the community version, but some require the paid professional version.
8. **Scanner**: The scanner is not available in the community edition. It scans the website automatically for many common vulnerabilities and lists them with information on confidence over each finding and their complexity of exploitation. It is updated regularly to include new and less known vulnerabilities.

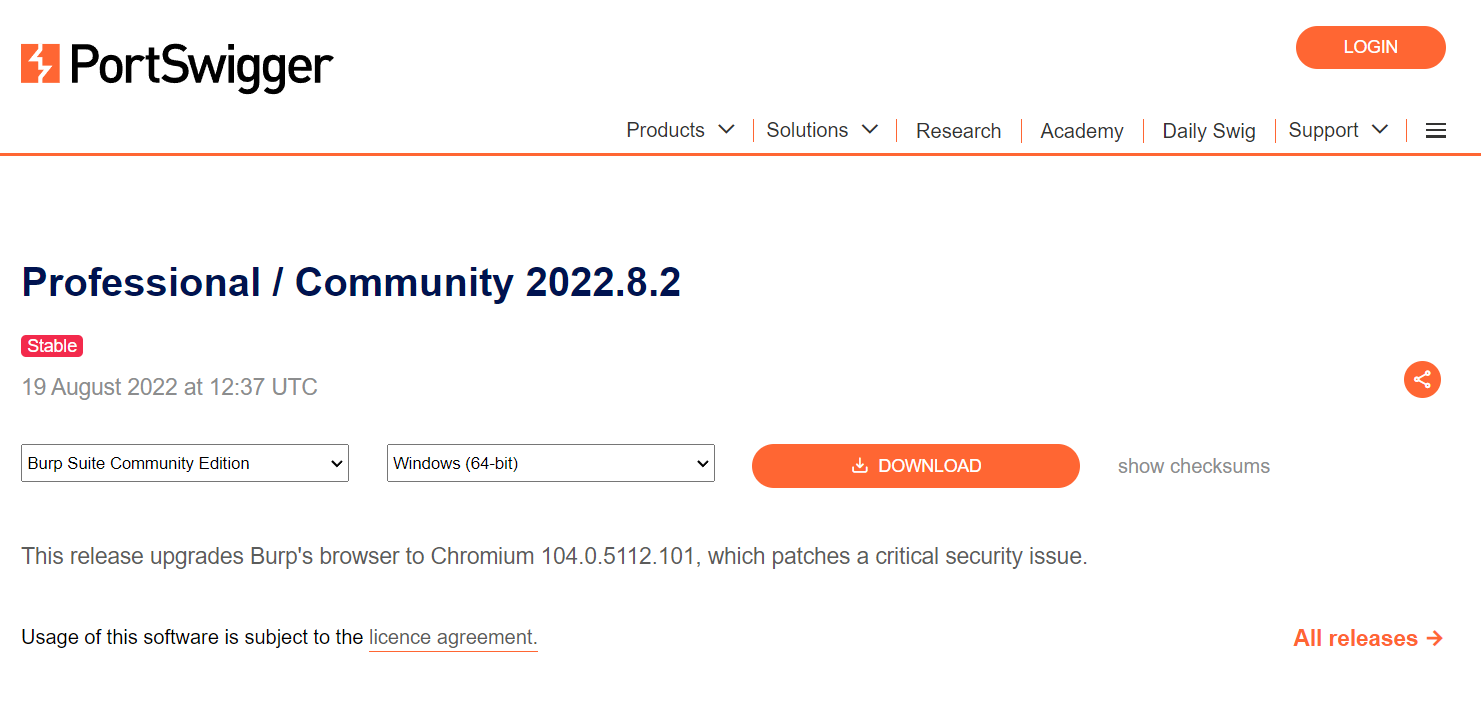
**Get Started with Burp Suite**

Getting started with Burp Suite professional and community edition is achieved in the following five steps:

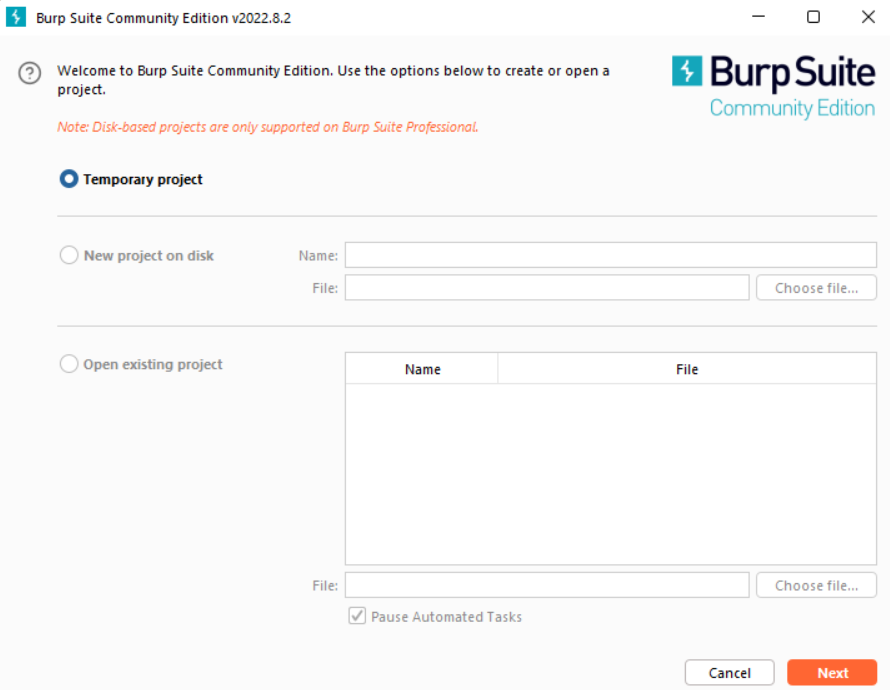
1. Download and install Burp Suite Software
2. Intercept HTTP traffic with Burp Proxy
3. Modify requests in Burp Proxy
4. Reissue requests with Burp Repeater manually

**1. Download and Install**

Step 1: You can download the latest version of Burp Suite professional and community edition using the links provided on the PortSwigger website.

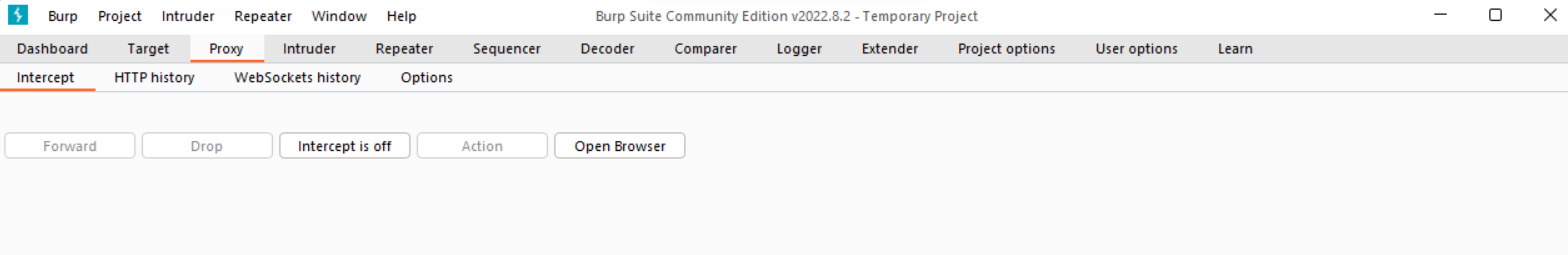


Step 2: Now, run the installer and open the Burp Suite software.

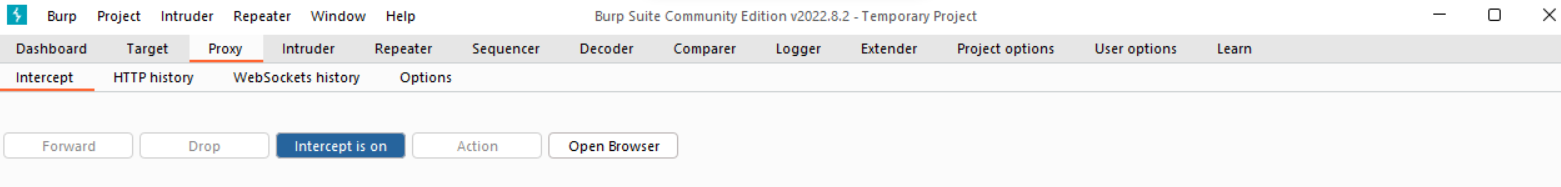


Intercepting HTTP traffic with Burp Proxy.

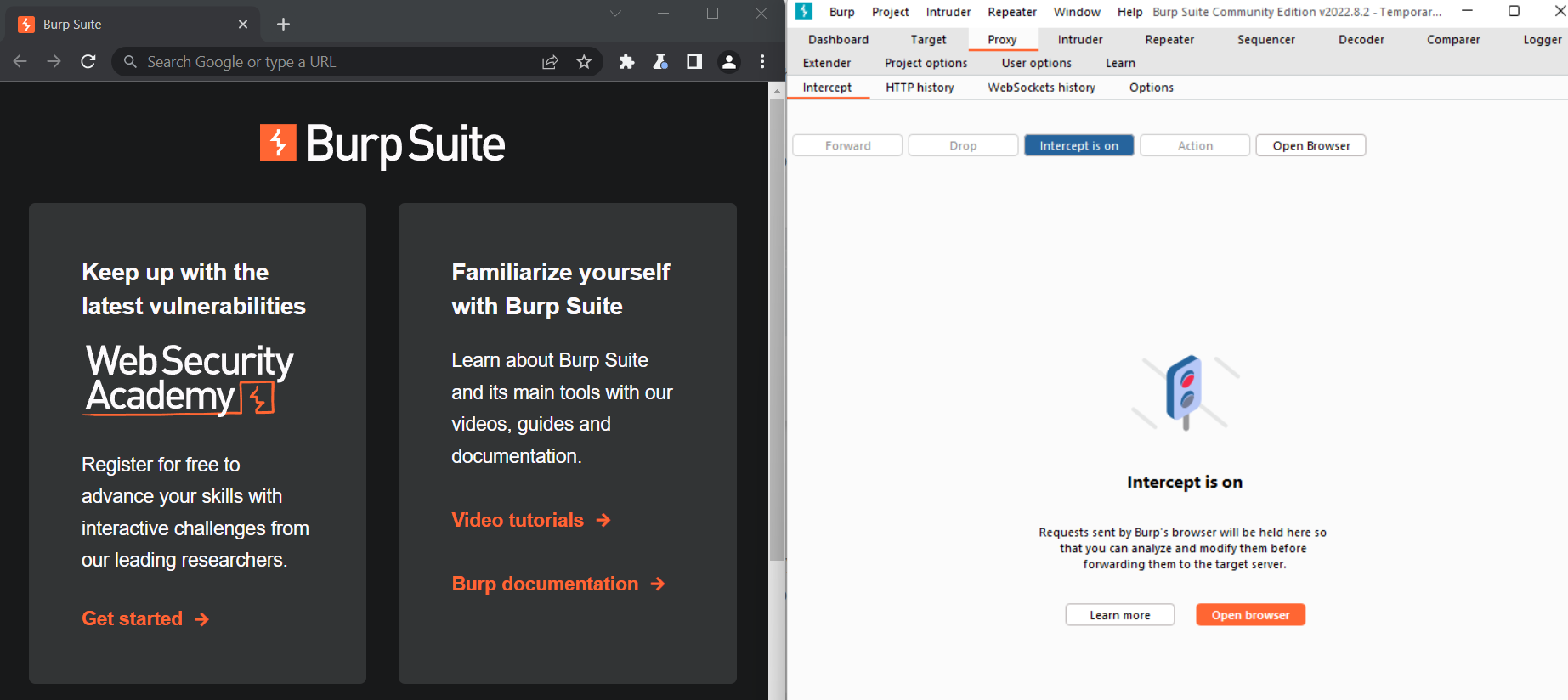
1. Go to the Proxy -> Intercept tab.



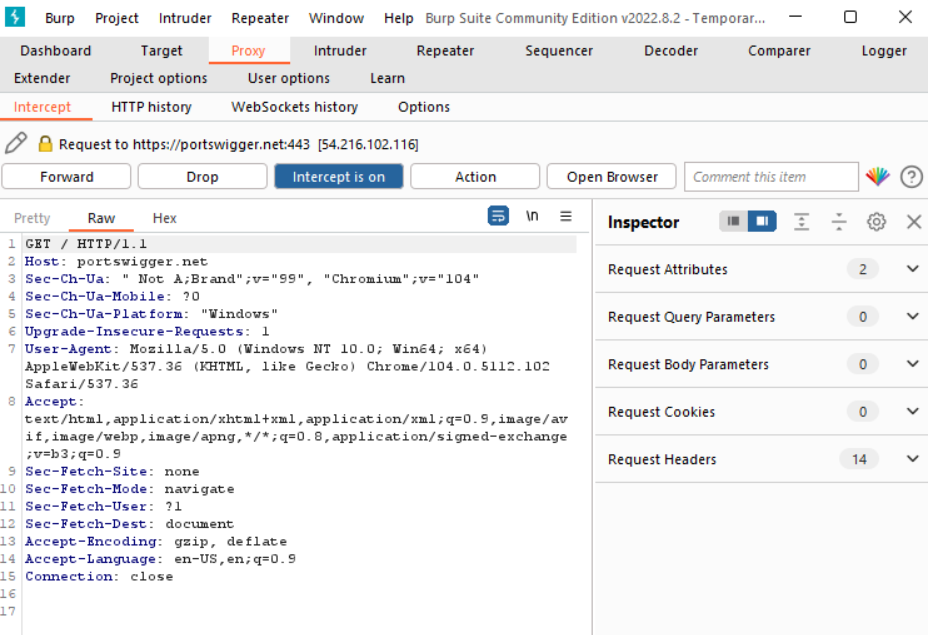
1. Click the Intercept is off button, so it toggles to Intercept is on.



1. Click Open Browser. This launches Burp's browser, which is preconfigured to work with Burp right out of the box. Position the windows so that you can see both Burp and Burp's browser.

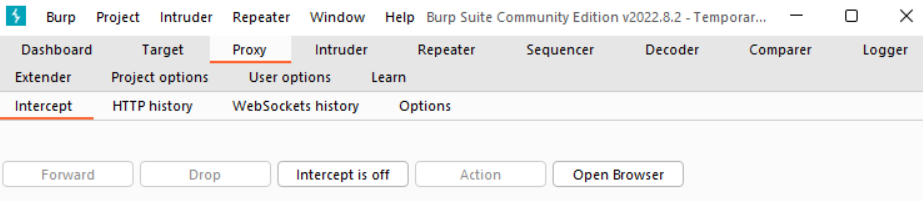


1. Using Burp's browser, try to visit [*https://portswigger.net*](https://portswigger.net) and observe that the site doesn't load. Burp Proxy has intercepted the HTTP request that was issued by the browser before it could reach the server. You can see this intercepted request on the Proxy -> Intercept tab.



The request is held here so that you can study it, and even modify it, before forwarding it to the target server.

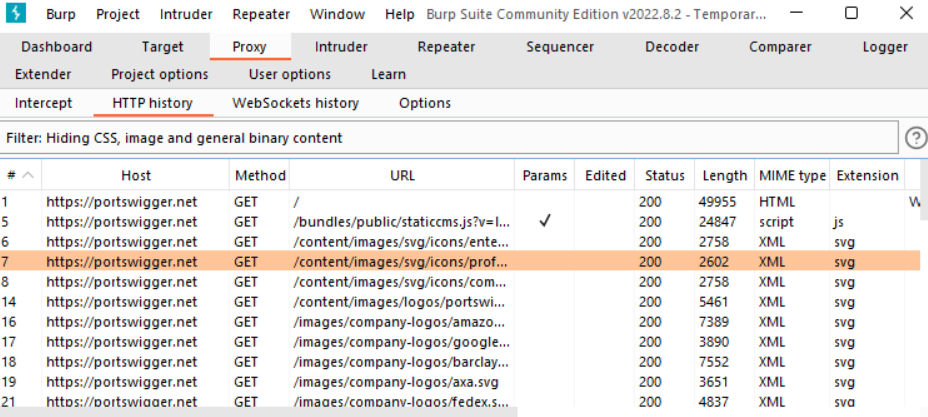
1. Click the Forward button several times to send the intercepted request, and any subsequent ones, until the page loads in Burp's browser.
2. Due to the number of requests browsers typically send, you often won't want to intercept every single one of them. Click the Intercept is on button so that it now says Intercept is off.

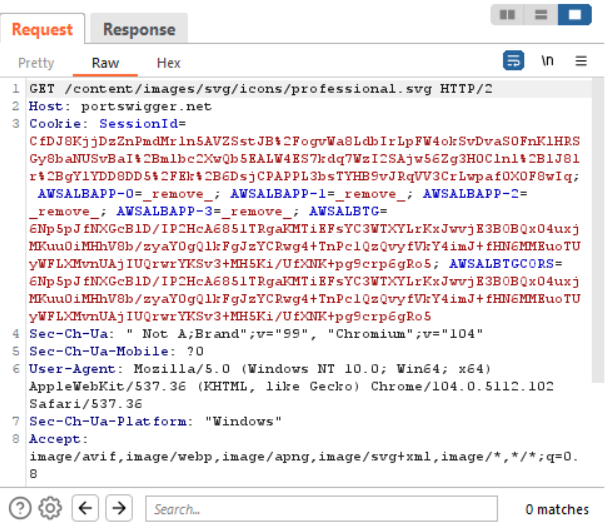


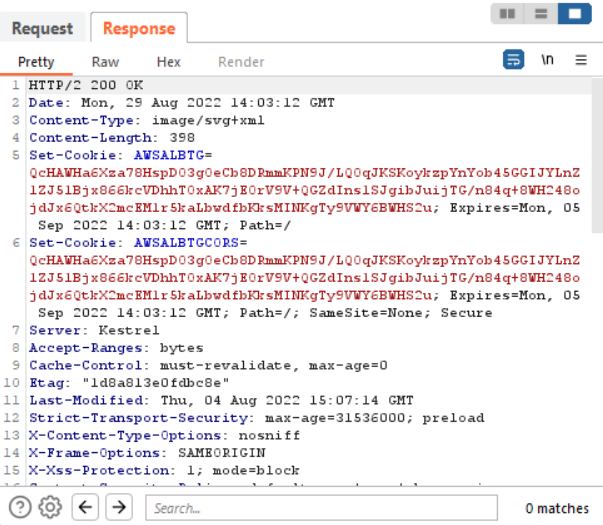
Go back to the browser and confirm that you can now interact with the site as normal.

1. In Burp, go to the Proxy > HTTP history tab. Here, you can see the history of all HTTP traffic that has passed through Burp Proxy, even while interception was switched off.

Click on any entry in the history to view the raw HTTP request, along with the corresponding response from the server.







This lets you explore the website as normal and study the interactions between Burp's browser and the server afterward, which is more convenient in many cases.

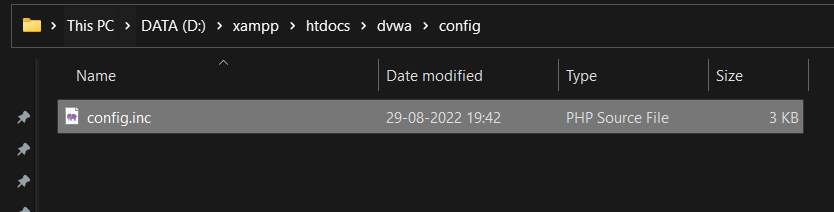
Step 3: Now Download DVWA (Damn Vulnerable Web Application) from Github for Pen Testing purpose. [*https://github.com/digininja/DVWA*](https://github.com/digininja/DVWA)



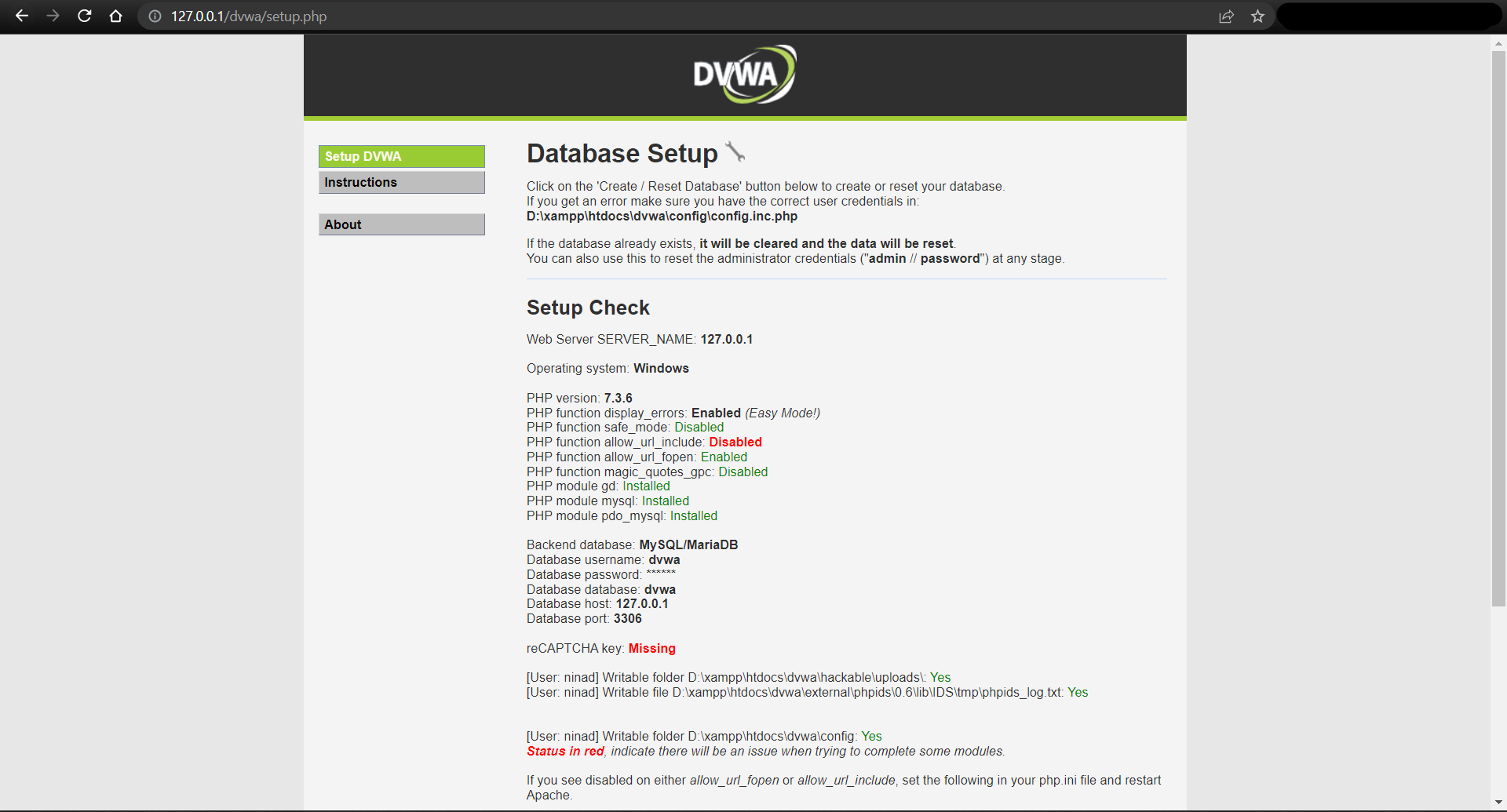
Step 4: Download Xampp and install. Reference for installation of Burp Suite, Xampp and DVWA. [*https://www.youtube.com/watch?v=cak2lQvBRAo*](https://www.youtube.com/watch?v=cak2lQvBRAo)

Step 5: Move the DVWA folder to htdocs folder under Xampp:

* Goto config -> Change the **config.inc.php.dist** file name to **config.inc.php**



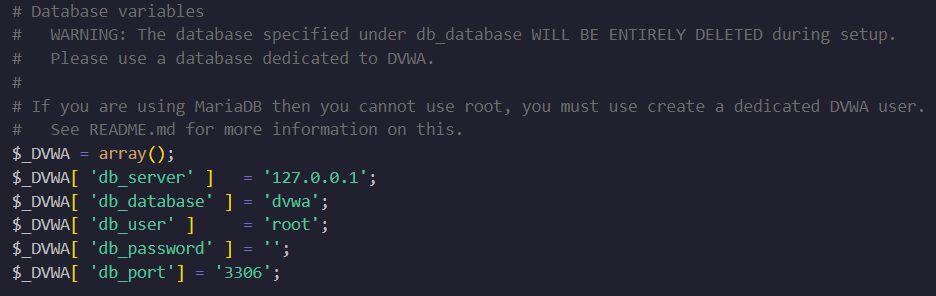
* Change the port number (8080) if required in Apache (httpd.conf) file
* Browse to [*http://127.0.0.1/DVWA-master*](http://127.0.0.1/DVWA-master)

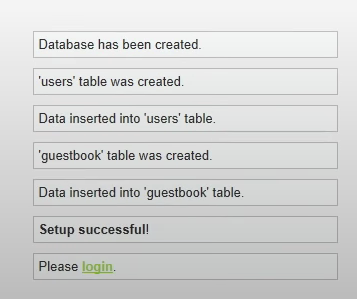


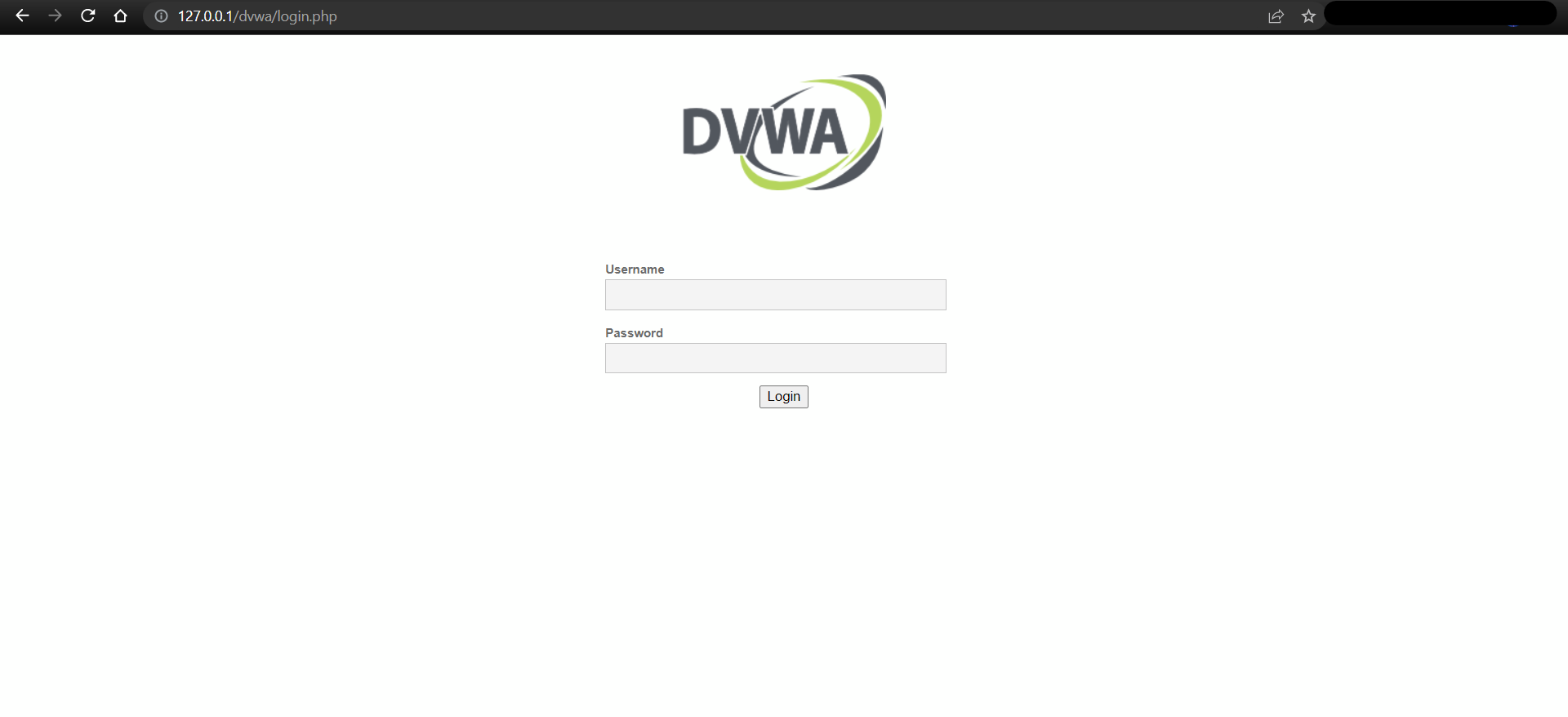
* Try to open <http://127.0.0.1:8012/DVWA-master/DVWA-master/setup.php>
* If it gives Sql Error , Open Config.inc.php file and change the

*$\_DVWA[ 'db\_user' ] = 'root';*

*$\_DVWA[ 'db\_password' ] = '';*

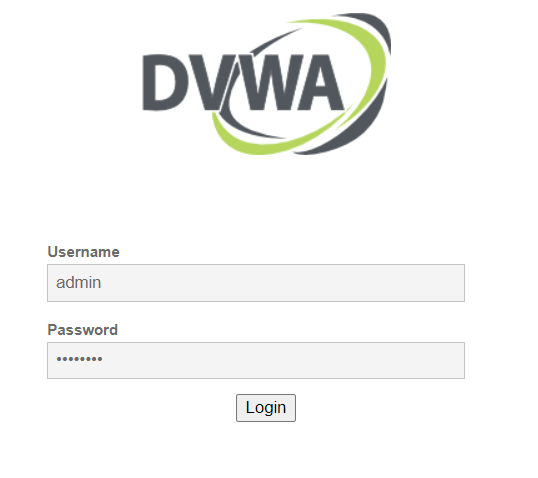


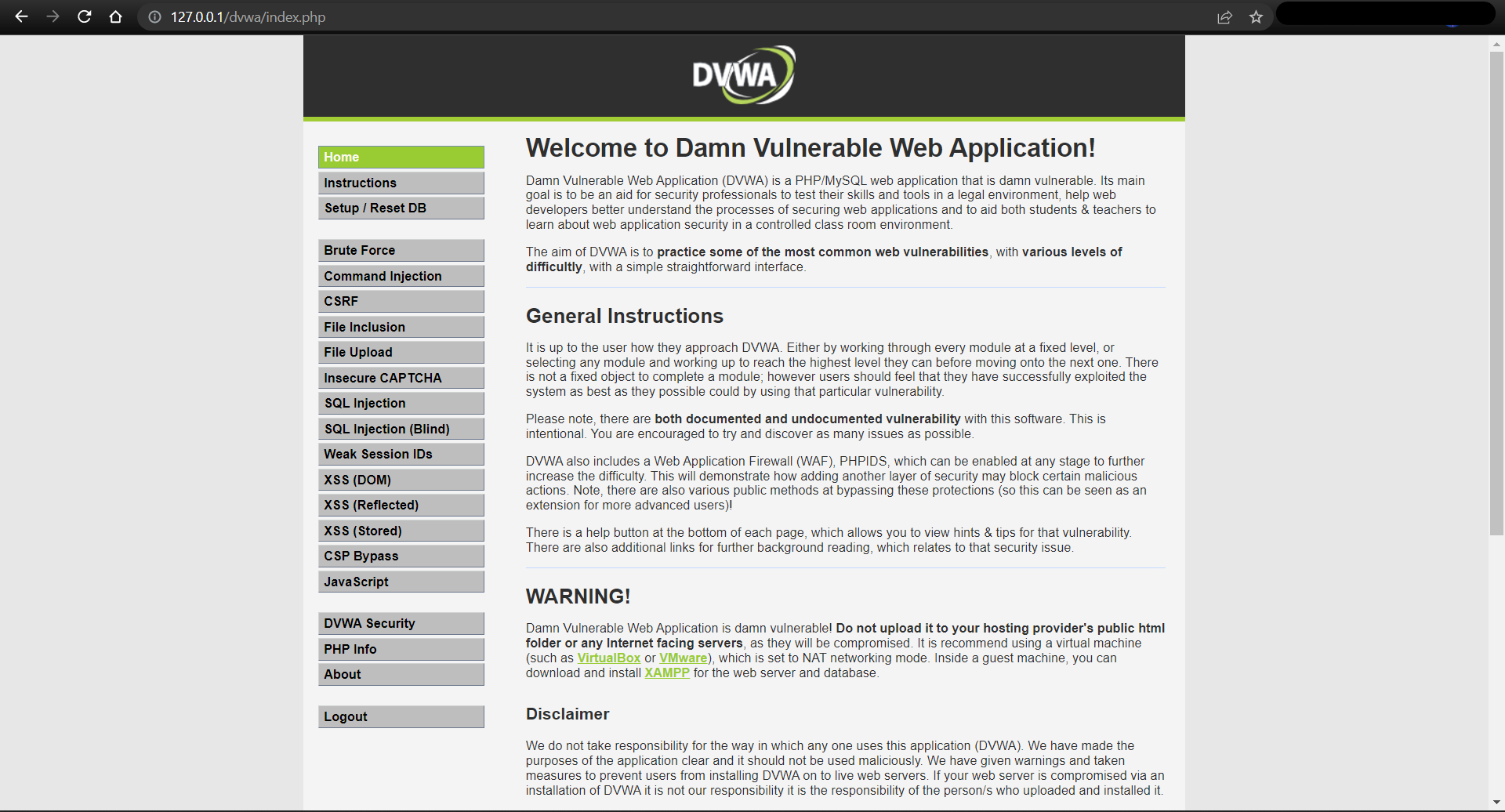




**2. Default Credentials**

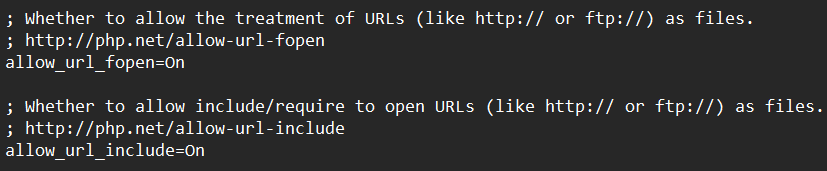
* Default username = admin
* Default password = password

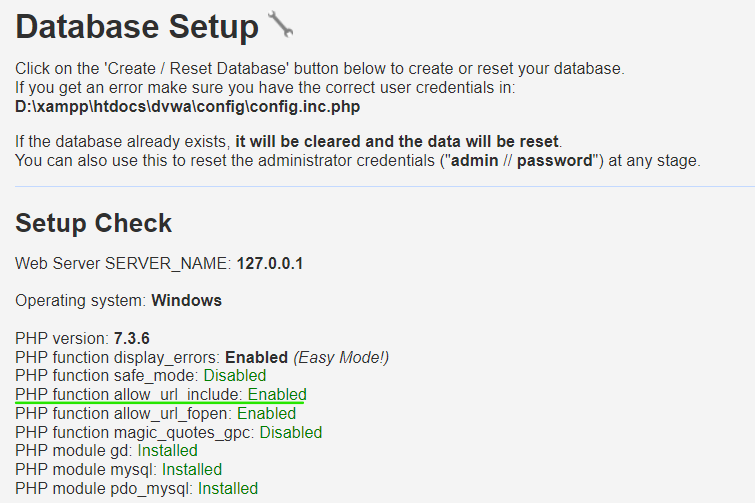




*allow\_url\_fopen = on*

This allows for Remote File Inclusions (RFI) [allow\_url\_fopen]. Make this change to a php.ini file in Xampp. Now try to login to the web application



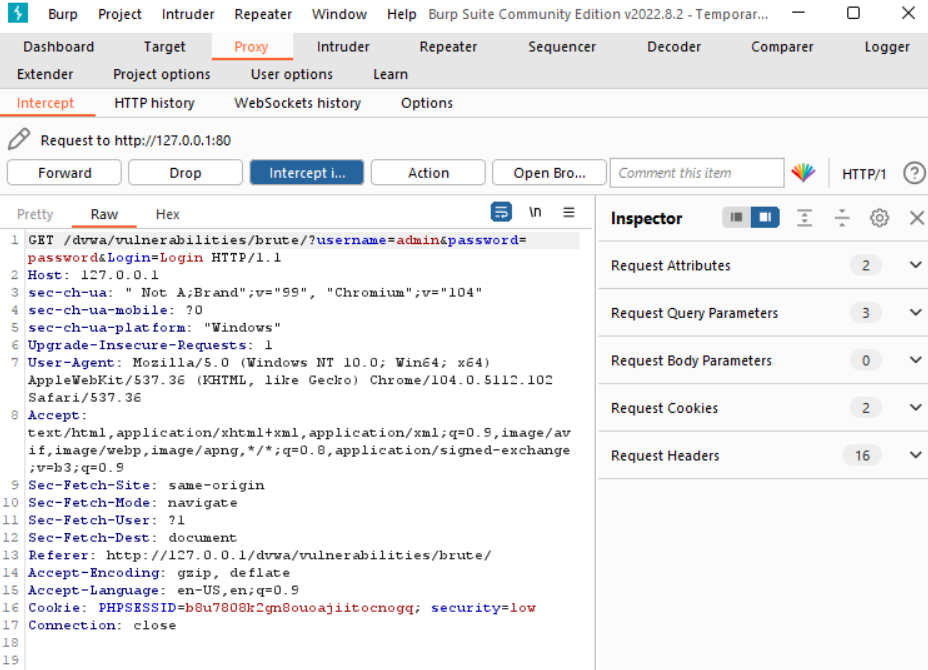


**3. Modify HTTP requests with Burp Proxy**

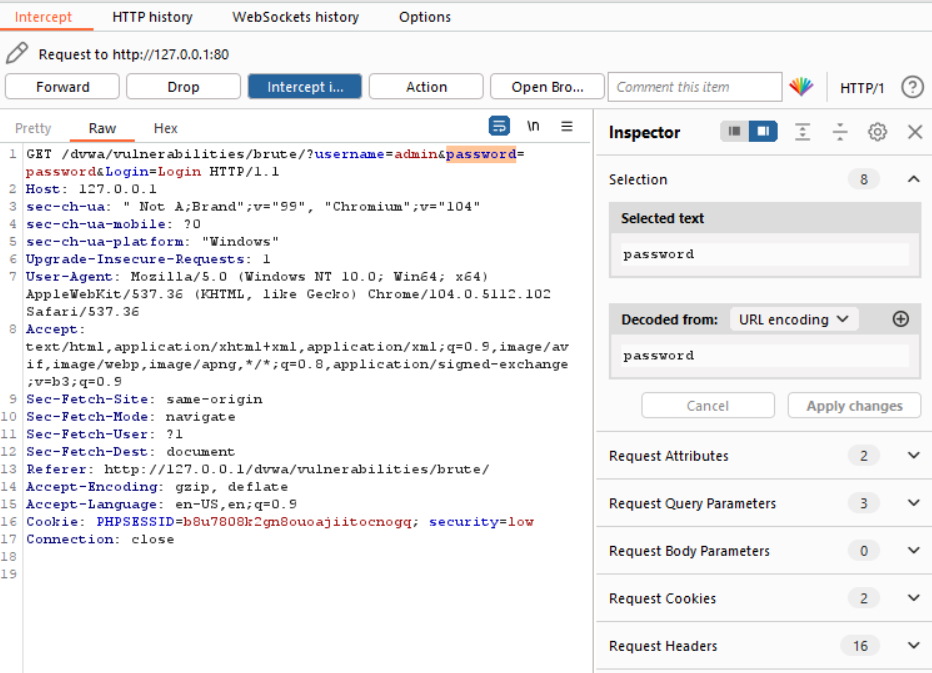
Change the security settings to low to turn off security features. Move to the Brute force tab and try to login using the default credentials. Intercept should be on.



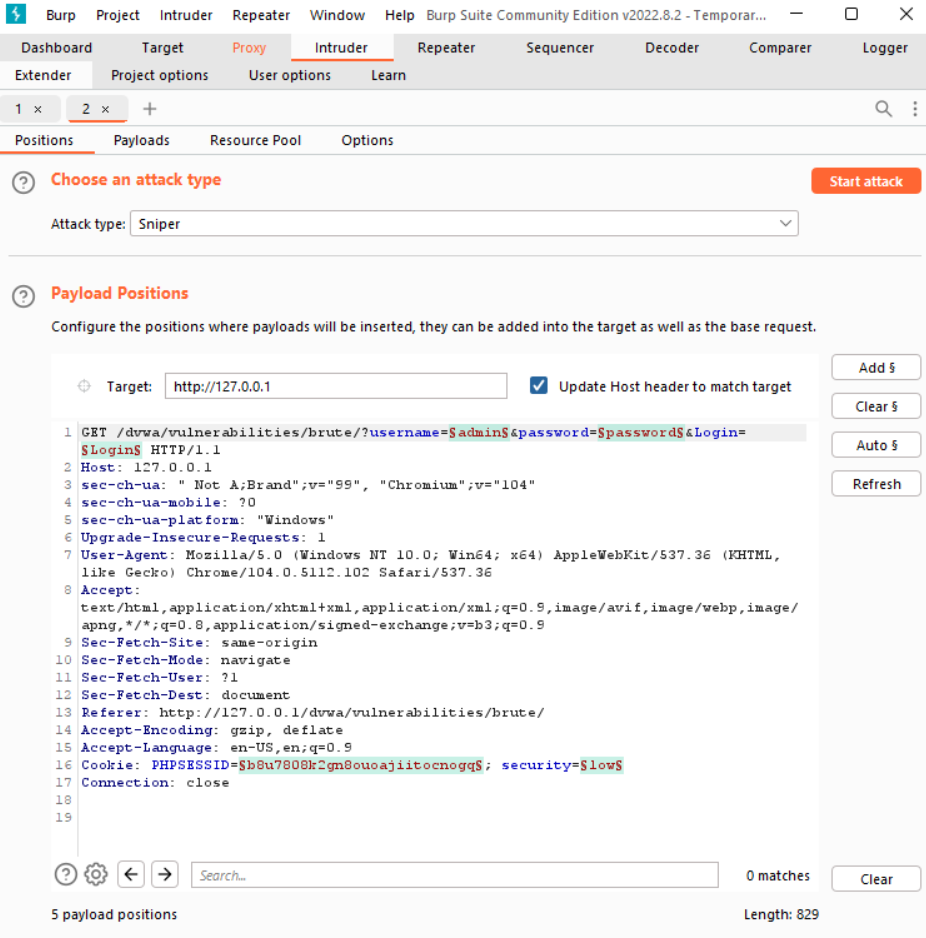
Now you can see the request in your intercept work area.



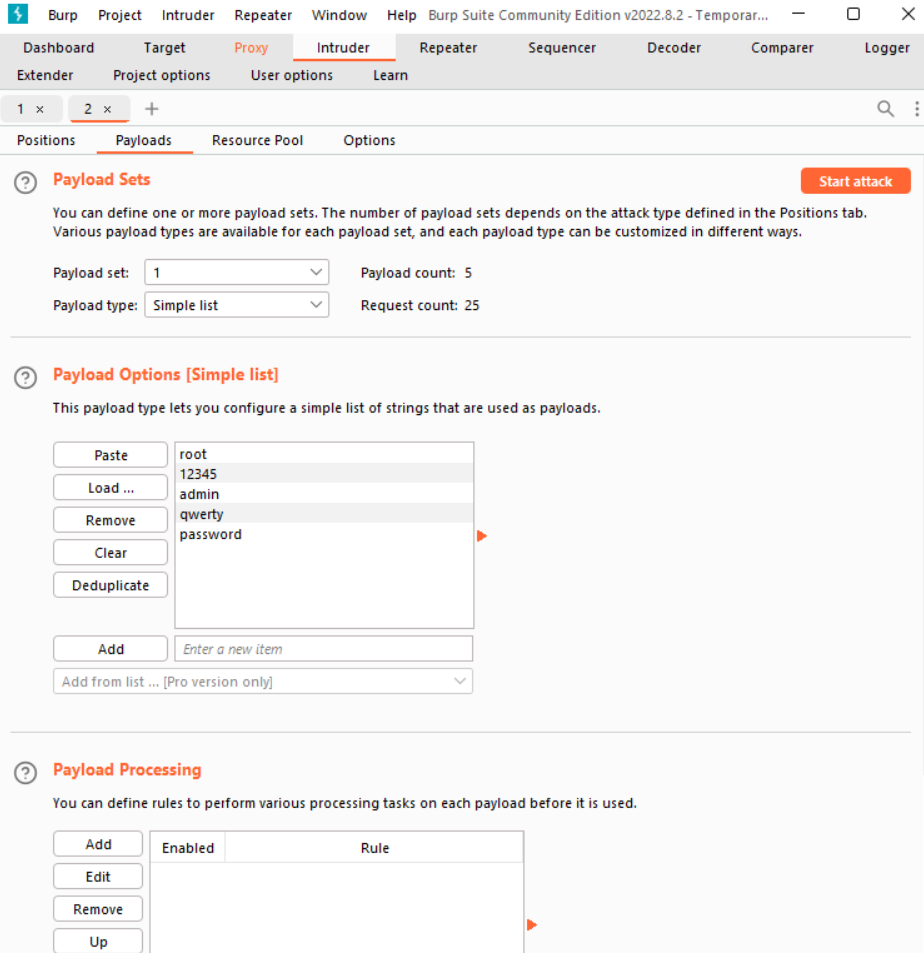
Click on the Action Tab and Send this request to the Intruder.



Modifying the Request. Now, you can make changes to the page.



Intruder will now be able to change the required field in the request and start an attack to find the correct password.



**Conclusion**:

Thus we have studied Owasp Web Security Testing Guidelines, VAPT Testing and the tools used in VAPT. We also have used Burp proxy to test web applications.