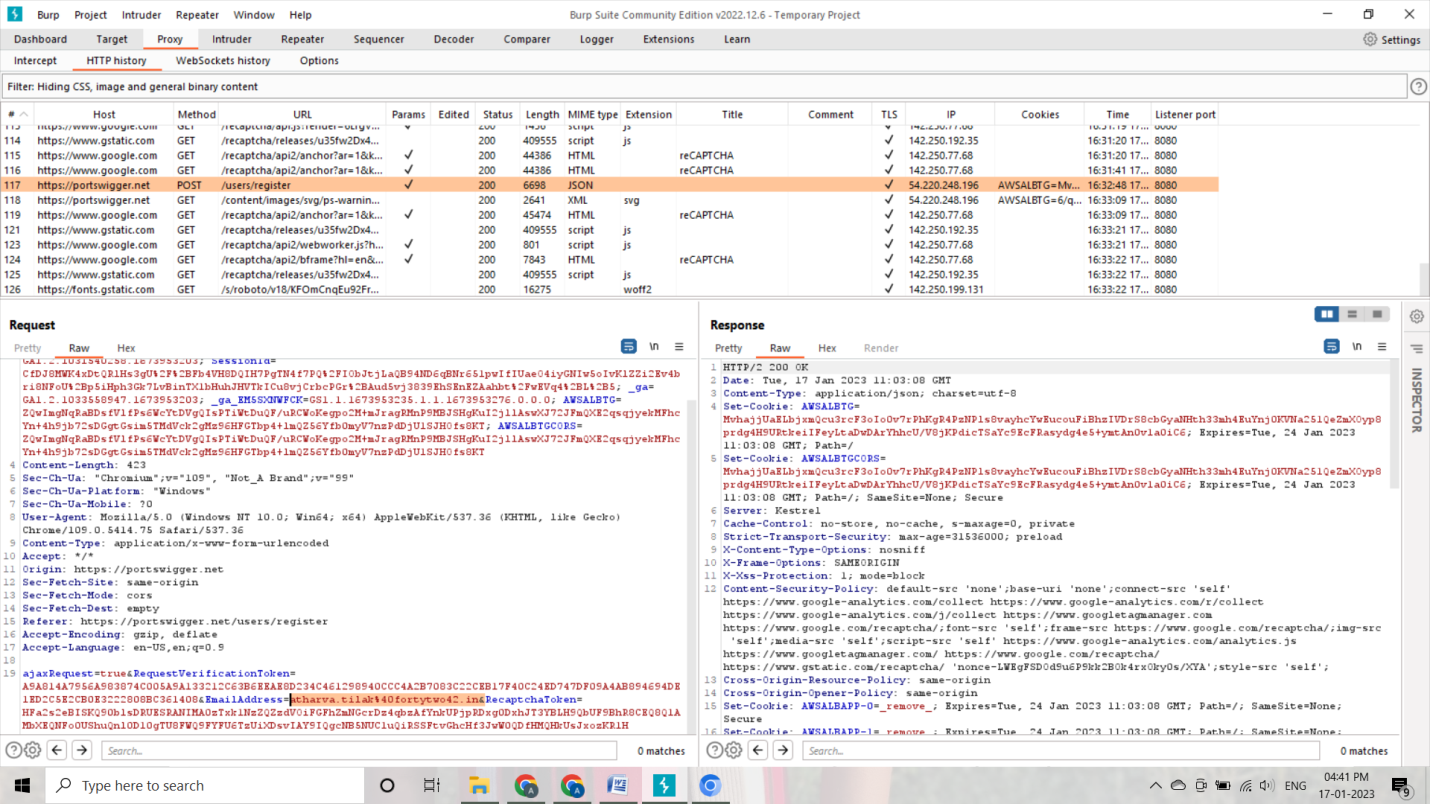
HTTP Rest API Session

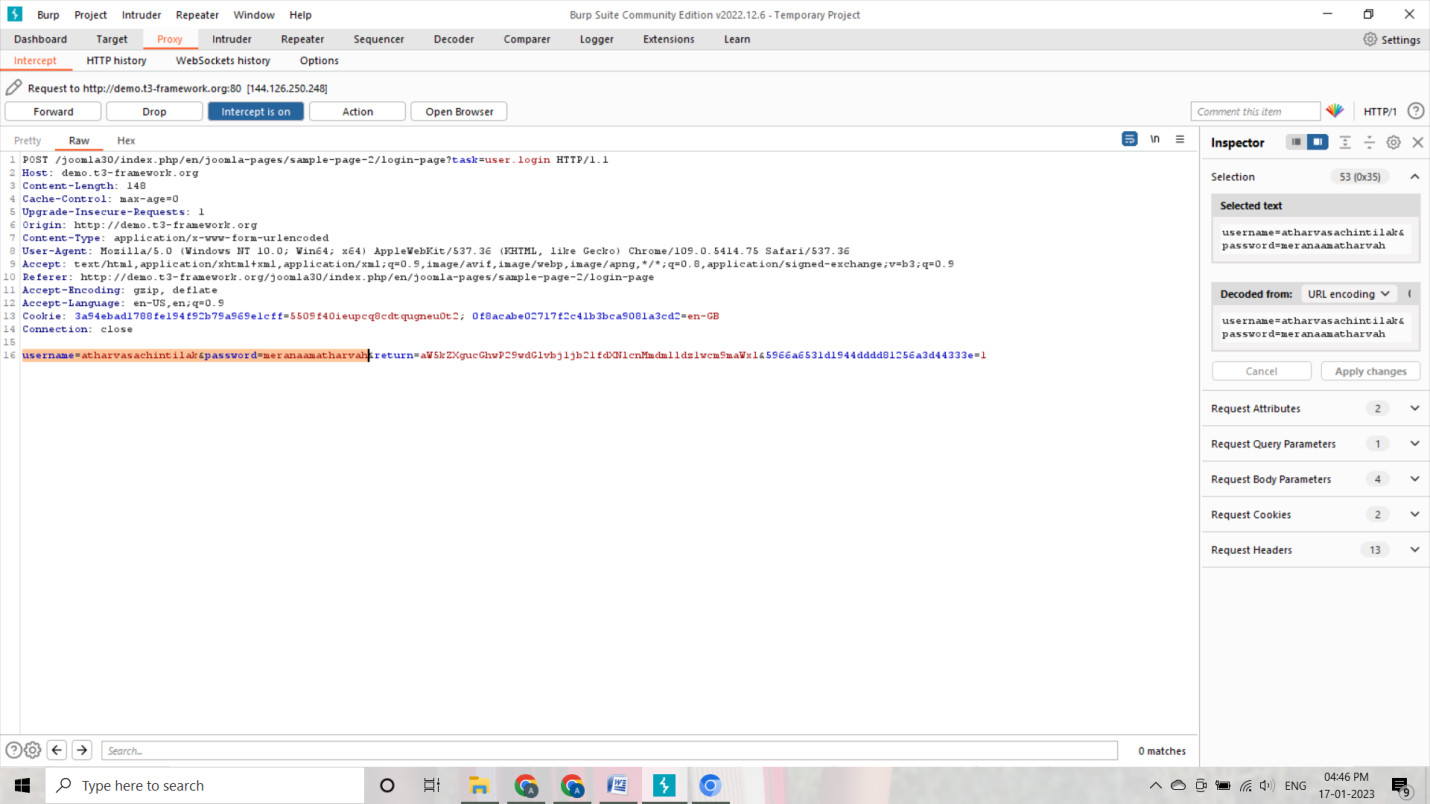
1. HTTP message, methods, urls, headers, body format for request, response.
   1. **HTTP message**: HTTP messages are the way in which the client and the server communicate; there are two types of HTTP messages, request and response. Requests are what the client sends to the server to which the server responds with a response. These messages are actual text data which is encrypted and span over multiple lines because of which it’s impossible to read the actual text data.
   2. **HTTP methods**: There are a lot of HTTP methods but the ones which are used the most are; Get, Post, Put, Delete. This methods are used to read, post, update or delete data on the server. There are other methods like options, patch etc. but they are not used frequently.
   3. **HTTP url**: Web clients and browsers like IE, chrome sends request to the server with the help of the url. Url specifies the location of the server.
   4. **HTTP headers**: HTTP headers are a field of request which contains additional context and metadata about the request and response. Header indicates contexts like; content type, format, media, etc.
   5. **HTTP body format**:
      1. At first a HTTP method like GET, POST, etc. is used to execute the action according to the request.
      2. Then location of the server is fetched with the help of URL or absolute patch of the protocol.
      3. In the URL, if the URL is complete it is called the absolute form of the URL mostly used with GET method
      4. If the URL consists of the domain name and port, then it is called the authority form. It is used when CONECT when setting up with any HTTP tunnel.
      5. The ‘\*’ is used with OPTIONS representing the server as a whole.
   6. **HTTP response**: The first line of the HTTP response is called a status line which contains the following information-
      1. The protocol version
      2. Status code
      3. A status text describing the current status code.
2. Use following methods to keep your API’s secured-
   1. **Scan for API vulnerabilities**: To maintain high level security of API services it is vital to enable API automatic scanning which will keep the API under watch all the time.
   2. **Use HTTPS/TLS for Rest API’s**: Using HTTPS/TLS will provide security to the API’s. It is the method of connecting the web browser with the server in a secured manner. HTTPS also helps to protect the credentials in transit as API’s need to maintain confidentiality, integrity and authentication.
   3. **HTTP methods**: HTTP methods(GET, PUT, POST, DELETE) should be used with a strict allow list to reduce the chances of the data getting hacked while sharing.
   4. **Validation**: API’s should include sufficient validation schemes on the servers as well as the clients side to provide another layer of security making API’s more authenticated.
   5. **API gateway**: If the organization uses a lot of API’s it is better to combine them and form a API gateway. This gateway acts as a centralized location for the API requests. This platform also provides different services like telemetry, rate limiting and user authentication. In short it acts like a gatekeeper to the collection of API’s present inside
3. Using Nessus for web application penetration

I penetrated a website [www.synergy-spark.com](http://www.synergy-spark.com) using nessus, I have attached the required documents to this mail showing the result of the penetration test.

1. Use burp site to intercept traffic and get the credentials of sample HTTP/HTTPS pages.
   1. HTTPS site



* 1. HTTP site



1. Use postman call HTTPS/HTTP API

I have attached the JSON file in this email as a result of testing postman.