
# Burp Request Testing Checklist

- Shouvik Dutta

## Comprehensive Assessment of Parameters, Headers, & Methods

This checklist is a comprehensive guide designed to help security professionals and developers perform targeted security assessments on web applications using Burp Repeater within each and every request you send to server.

## Parameters

- Try entering null values, empty values, and special characters.
- Try injecting SQL code, XSS code, and command injection code.
- Try entering values that are too long or too short.
- Try entering values that are out of range.
- Try entering values that are not in the correct format.
- Try adding extra parameters.
- Try modifying the values of existing parameters.
- Try deleting parameters.
- Try entering parameters that are not used by the web application.
- Try modifying the values of parameters that are not used by the web application.
- Try deleting parameters that are not used by the web application.

### Query strings

- Try adding query strings that are not used by the web application.
- Try modifying the values of query strings that are not used by the web application.
- Try deleting query strings.
- Try adding query strings that are used by the web application.
- Try modifying the values of query strings that are used by the web application.
- Try deleting query strings that are used by the web application.

### HTTP Headers

### DELETE

- Try sending a DELETE request to a URL that does not exist. Verify that the web application returns a 404 Not Found error message.
- Try sending a DELETE request to a URL that exists. Verify that the web application deletes the resource and returns a 200 OK status code.
- Try sending a DELETE request with invalid data. Verify that the web application returns an error message.

### GET

- Try sending a GET request to a URL that does not exist. Verify that the web application returns a 404 Not Found error message.
- Try sending a GET request to a URL that exists. Verify that the web application returns the requested resource.

### POST

- Try sending a POST request to a URL that does not exist. Verify that the web application returns a 404 Not Found error message.
- Try sending a POST request to a URL that exists. Verify that the web application processes the request and returns a 200 OK status code.
- Try sending a POST request with invalid data. Verify that the web application returns an error message.
- Try sending a POST request with valid data. Verify that the web application creates the resource and returns a 201 Created status code.

**PUT**

- Try sending a PUT request to a URL that does not exist. Verify that the web application returns a 404 Not Found error message.
- Try sending a PUT request to a URL that exists. Verify that the web application updates the resource and returns a 200 OK status code.
- Try sending a PUT request with invalid data. Verify that the web application returns an error message.
- Try sending a PUT request with valid data. Verify that the web application updates the resource and returns a 200 OK status code.

**HEAD**

- Try sending a HEAD request to a URL that does not exist. Verify that the web application returns a 404 Not Found error message.
- Try sending a HEAD request to a URL that exists. Verify that the web application returns the requested resource, but without the body.

**OPTIONS**

- Try sending an OPTIONS request to a URL that does not exist. Verify that the web application returns a 404 Not Found error message.
- Try sending an OPTIONS request to a URL that exists. Verify that the web application returns a list of the HTTP methods that it supports.

**TRACE**

- Try sending a TRACE request to a URL that does not exist. Verify that the web application returns a 404 Not Found error message.
- Try sending a TRACE request to a URL that exists. Verify that the web application returns the request, including the headers and body.

**Cookies**

- Try setting cookies with invalid names.
- Try setting cookies with invalid values.
- Try setting cookies with invalid domains.
- Try modifying the values of existing cookies.
- Try deleting cookies.
- Try setting cookies with valid names and values.
- Try setting cookies with valid domains.
- Try modifying the values of existing cookies with valid names and values.
- Try deleting cookies with valid names and values.

**Hidden fields**

- Try adding hidden fields that are not used by the web application.
- Try modifying the values of hidden fields that are not used by the web application.
- Try deleting hidden fields.
- Try adding hidden fields that are used by the web application.
- Try modifying the values of hidden fields that are used by the web application.
- Try deleting hidden fields that are used by the web application.

**Host Header**

- Try setting the host header to a different hostname.
- Try setting the host header to a hostname that does not exist.
- Try setting the host header to the same hostname as the web application.
- Try setting the host header to a different domain name than the web application.

**User-Agent Header**

- Try setting the user agent to a different value.
- Try setting the user agent to a value that is not used by a real browser.
- Try setting the user agent to the same user agent as the web application.
- Try setting the user agent to a different user agent than the web application.

**Referer Header**

- Try setting the referer header to a different URL.
- Try setting the referer header to a URL that does not exist.
- Try setting the referer header to the same URL as the web application.
- Try setting the referer header to a different URL than the web application.

**Accept Header**

- Try setting the accept header to a different value.
- Try setting the accept header to a value that is not supported by the web application.
- Try setting the accept header to the same accept header as the web application.
- Try setting the accept header to a different accept header than the web application.

**Content-Type Header**

- Set the Content-Type header to a different content type.
- Verify that the web application handles the content correctly.

**Content-Length Header**

- **Different content length**
  - Set the Content-Length header to a different content length.
  - Verify that the web application handles the content correctly.
- **Zero content length**
  - Set the Content-Length header to zero.
  - Verify that the web application returns an error message.
- **Same content length as the web application**
  - Set the Content-Length header to the same content length as the web application.
  - Verify that the web application processes the request correctly.