

## netsparker.auth

Contains form authentication helper methods.

### Methods

`applyOtp(otpopt)`

Tries to find an OTP form and fill the specified password.

Parameters:

Name	Type	Attributes	Description
otp	string	<optional>	The OTP token to fill.

`click(el, delayopt)`

Simulates a click for the specified el.

Parameters:

Name	Type	Attributes	Description
el	string   HTMLElement		The element to click.
delay	number	<optional>	The number of milliseconds (thousandths of a second) that the click should be delayed by. Note that all delays are timed from the beginning of the script execution and does not work in sequence if you have several function calls with delay.

Example

```
// Click element by id  
netsparker.auth.click('LoginButton');  
// Click element by id after 2 seconds  
netsparker.auth.click('LoginButton', 2000);  
// Click element by DOM element reference  
netsparker.auth.click(document.getElementsByTagName('BUTTON')[0]);
```

`clickByQuery(query, delayopt)`

Simulates a click for the element specified by the CSS query.

Parameters:

Name	Type	Attributes	Description
query	string		The CSS query that locates the element.
delay	number	<optional>	The number of milliseconds (thousandths of a second) that the click should be delayed by. Note that all delays are timed from the beginning of the script execution and does not work in sequence if you have several function calls with delay.

Example

```
// Click element by using a complex CSS query
netsparker.auth.clickByQuery('#loginForm > div:nth-child(2) > button');
// Click element by id
netsparker.auth.clickByQuery('#LoginButton');
// Click element by id after 2 seconds
netsparker.auth.clickByQuery('#LoginButton', 2000);
```

`executeInFrame(frameEl, code, delayopt)`

Executes the supplied code in specified frame element.

Parameters:

Name	Type	Attributes	Description
frameEl	string   HTMLElement		The frame element id or DOM reference to execute code.
code	string		The code to execute.
delay	number	<optional>	The number of milliseconds (thousandths of a second) that the code execution should be delayed by. Note that all delays are timed from the beginning of the script execution and does not work in sequence if you have several function calls with delay.

Example

```
// Clicks an element inside a frame  
var frame = document.getElementsByTagName('IFRAME')[1];  
netsparker.auth.executeInFrame(frame, 'netsparker.auth.click("LoginButton");');
```

`log(message)`

Logs a message.

Parameters:

Name	Type	Description
message	string	The message.

Example

```
// Log a simple message  
netsparker.auth.log('Hello world!');
```

`login(usernameopt, passwordopt, otpopt, delayopt)`

Tries to find a login form and fill the specified credentials.

Parameters:

Name	Type	Attributes	Description
username	string	<optional>	The username to fill.
password	string	<optional>	The password to fill.
otp	string	<optional>	The one time password to fill.
delay	number	<optional>	The number of milliseconds (thousandths of a second) that the login should be delayed by. Note that all delays are timed from the beginning of the script execution and does not work in sequence if you have several function calls with delay.

Example

```
// Login using hard-coded values
netsparker.auth.login('john.doe', 'p4ssw0rd', '543326');
// Login using hard-coded values after 2 seconds
netsparker.auth.login('john.doe', 'p4ssw0rd', '543326', 2000);
// Login using implicit credentials (current persona)
netsparker.auth.login();
// Login using implicit credentials (current persona) after 2 seconds
netsparker.auth.login(2000);
```

`request(uriopt, requestObjectopt)`

Sends HTTP requests and returns response details.

There are three usages of this function:

1. If called with a string parameter, sends a GET request to the given URI and returns the response.
2. If called with no parameters, returns a requestObject to be passed to the function itself.
3. If called with requestObject parameter, sends a request which is defined by requestObject.

This function returns a response object which is explained at the latest example.

Parameters:

Name	Type	Attributes	Description
uri	string	<optional>	The target uri to send GET request.
requestObject	object	<optional>	Object containing request details.

Examples

```
var response = netsparker.auth.request("http://example.com");
```

```
var requestObject = netsparker.auth.request();  
requestObject.uri = "http://example.com";  
requestObject.method = "POST";  
requestObject.parameters.add("parameterKey", "parameterValue", 1); // 0 - GET (QueryString) 1 - POST parameter  
requestObject.headers.add("headerName", "headerValue");  
var response = netsparker.auth.request(requestObject);
```

```
var response = netsparker.auth.request("http://example.com");  
response.body; // the body of the response.  
response.headers; // collection of response headers.  
response.responseTime; // response time.  
response.statusCode; // HTTP status code of the response.  
response.statusDescription; // Human readable description of the HTTP response code.  
response.uri; // The uri that HTTP request has sent.
```

`setInputValue(el, value, delayopt)`

Sets the value of specified el.

Parameters:

Name	Type	Attributes	Description
el	string   HTMLElement		The element to click.
value	string		The value to set.
delay	number	<optional>	The number of milliseconds (thousandths of a second) that this set value operation should be delayed by. Note that all delays are timed from the beginning of the script execution and does not work in sequence if you have several function calls with delay.

Example

```
// Set value by id  
netsparker.auth.setInputValue('Username', 'john.doe');  
// Set value after 2 seconds  
netsparker.auth.setInputValue('Username', 'john.doe', 2000);  
// Set value by DOM element reference  
netsparker.auth.setInputValue(document.forms[0].elements[0], 'john.doe');
```

`setValueByQuery(query, value, delayopt)`

Finds input element using the CSS query and sets its value.

Parameters:

Name	Type	Attributes	Description
query	string		The CSS query that locates the element.
value	string		The value to set.
delay	number	<optional>	The number of milliseconds (thousandths of a second) that this set value operation should be delayed by. Note that all delays are timed from the beginning of the script execution and does not work in sequence if you have several function calls with delay.

Example

```
// Query complex paths
netsparker.auth.setValueByQuery('#loginForm > div:nth-child(2) > input', 'john.doe');
// Query by id
netsparker.auth.setValueByQuery('#Username', 'john.doe');
// Set value after 2 seconds
netsparker.auth.setValueByQuery('#Username', 'john.doe', 2000);
```