CLAIMS

1. A method for preventing a server from being attacked, the method comprising:

dynamically and randomly allocating a page script corresponding to a page request from a plurality of page scripts corresponding to the page request, when receiving the page request sent by a browser (101);

sending the dynamically and randomly allocated page script to the browser, so that the browser executes the page script to obtain a script execution parameter (102);

determining whether a page verification request is expired, when receiving the page verification request sent by the browser (103); and

in response to determining that the page verification request is expired, outputting error prompt information indicating page expiration (104a); or

in response to determining that the page verification request is not expired, verifying whether the script execution parameter comprised in the page verification request is valid (104b); and

in response to determining that the script execution parameter comprised in the page verification request is not valid, rejecting the page request (105b).

2. The method according to claim 1, wherein dynamically and randomly allocating the page script corresponding to the page request from the plurality of page scripts corresponding to the page request comprises:

retrieving a page URL in the page request (201); and

randomly extracting a page script from a plurality of page scripts that exist in a predetermined script library and are corresponding to the page URL, wherein different script execution parameter acquisition logic is used in the plurality of page scripts (202).

3. The method according to claim 2, wherein the script execution parameter acquisition logic comprises retrieving a token value in an encrypted cookie information.

4. The method according to claim 2, wherein before randomly extracting a page script from a plurality of page scripts that exist in a predetermined script library and are corresponding to the

page URL, the method further comprises:

configuring page scripts corresponding to each page URL in the predetermined script library and identifier information corresponding to the page scripts.

5. The method according to claim 1, wherein determining whether the page verification request is expired comprises:

determining whether a request time of the page verification request is later than a sum of a predetermined time period and a time for the browser to execute the page script; and

in response to determining that the request time of the page verification request is later than the sum, determining that the page verification request is expired; or

in response to determining that the request time of the page verification request is not later than the sum, determining that the page verification request is not expired.

6. The method according to any one of claims 1 to 5, wherein the page request further comprises identifier information of the page script executed by the browser, and the verifying whether the script execution parameter comprised in the page verification request is valid comprises:

searching the predetermined script library for a page script corresponding to the identifier information, wherein the predetermined script library further stores identifier information corresponding to each page script; and

verifying, based on the page script corresponding to the identifier information, whether the script execution parameter comprised in the page verification request is valid.

7. The method according to claim 6, wherein verifying, based on the page script corresponding to the identifier information, whether the script execution parameter comprised in the page verification request is valid comprises:

retrieving a local script parameter based on the page script corresponding to the identifier information;

determining whether the local script parameter is the same as the script execution parameter comprised in the page verification request; and

in response to determining that the local script parameter is the same as the script execution parameter, determining that the script execution parameter is valid; or

in response to determining that the local script parameter is not the same as the script execution parameter, determining that the script execution parameter is invalid.

8. The method according to any one of claims 1 to 7, wherein rejecting the page request can be based on a user identity information included in the page verification request.

9. The method according to claim 8, wherein the user identity information comprises a cookie information generated by the server.

10. A device for preventing a server from being attacked, the device comprising a plurality of modules configured to perform the method of any one of claims 1 to 9.