How to Troubleshoot WordPress Ajax Errors

Author: Svetoslav Marinov (Slavi) | All Rights Reserved.
https://wpsandbox.net/wordpress/troubleshoot-wordpress-ajax-errors-p1316

From time to time you'll inevitably encounter a frustrating WordPress ajax error where the user clicks on a button or tries to submit a form and nothing happens. It is very likely a JavaScript issue, but to find out what's the actual error, you need to do some debugging. Just a reminder that an ajax request can send and retrieve data from a server asynchronously (in the background). That way users can see the results without a full page reload, which is quicker than reloading the whole page and is a better user experience.

Common WordPress ajax issues:

- ajaxurl js variable is not defined; that's usually the case on the public side.
- The current theme or a (custom) plugin is loading a different version of jQuery instead of using the one that ships with WordPress. How to Check if your WordPress Site is Loading Multiple jQuery Scripts
- admin-ajax.php returns 0 => that means that there's no (active) callback to handle the request defined by the action parameter.
- if you're using Cloudflare, you shouldn't block requests to /wp-admin/admin-ajax.php because these requests need to run on your server.
- the hook that handles the ajax call(s) tries to call a php function or a method that doesn't exist (not loaded e.g. inactive plugin) or is not loaded yet. Such requests usually return a 500 Internal Server Error.
- theme or plugin conflicts
- a security plugin, anti-spam WordPress plugin or a web firewall blocks access to it

Understanding the WordPress Ajax Endpoint (/wp-admin/admin-ajax.php)

WordPress uses the admin-ajax.php file located in /wp-admin/ for handling Ajax requests both on the public side and in the admin area. There can be many reasons why an Ajax request fails, and identifying the error is the first step toward solving it.

The very first thing is to find out what's the error to narrow down the possible causes.

a plugin/theme or a PHP code snippet handling WordPress Ajax requests need to register 2 (php) hooks per ajax action. One is for logged in users and the other one is for guests (see nopriv).

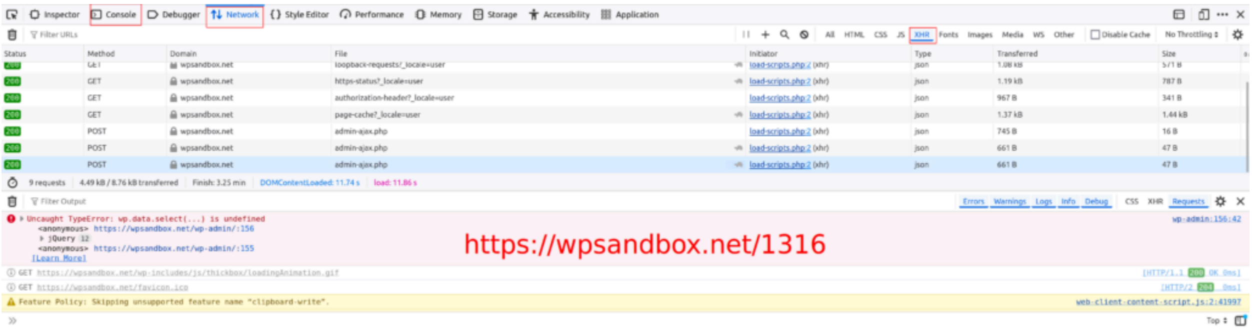
```
// Hook the function to handle logged in users
add_action('wp_ajax_my_action', 'my_company_plugin_name_ajax');

// handle guests (non-logged in users
add_action('wp_ajax_nopriv_my_action',
'my_company_plugin_name_ajax');
```

How to find the Ajax error on the Frontend?

use Developer Tools to find out what's happening.

Open Developer tools both in Chrome/Firefox/Brave F12 or CTRL+SHIFT+I



Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	wpsandbox.net	loopback-requests?_locale=user	load-scripts.php:2 (xhr)	json	1.08 kB	571 B
200	GET	wpsandbox.net	https-status?_locale=user	load-scripts.php:2 (xhr)	json	1.19 kB	787 B
200	GET	wpsandbox.net	authorization-header?_locale=user	load-scripts.php:2 (xhr)	json	967 B	341 B
200	GET	wpsandbox.net	page-cache?_locale=user	load-scripts.php:2 (xhr)	json	1.37 kB	1.44 kB
200	POST	wpsandbox.net	admin-ajax.php	load-scripts.php:2 (xhr)	json	745 B	16 B
200	POST	wpsandbox.net	admin-ajax.php	load-scripts.php:2 (xhr)	json	661 B	47 B
200	POST	wpsandbox.net	admin-ajax.php	load-scripts.php:2 (xhr)	json	661 B	47 B

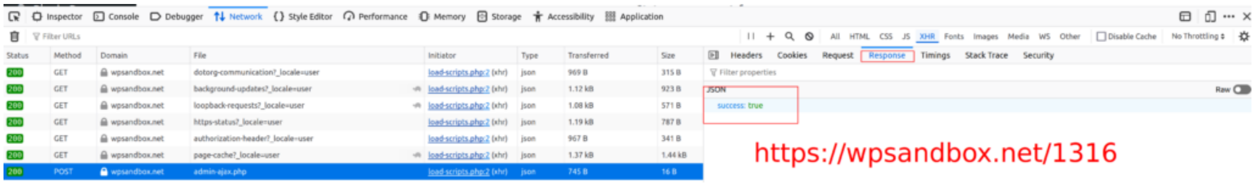
9 requests | 4.49 kB / 8.76 kB transferred | Finish: 3.25 min | DOMContentLoaded: 11.74 s | load: 11.86 s

```
Uncaught TypeError: wp.data.select(...) is undefined                wp-admin:156:42
    <anonymous> https://wpsandbox.net/wp-admin/:156
  > jQuery 12
    <anonymous> https://wpsandbox.net/wp-admin/:155
  [Learn More]
GET https://wpsandbox.net/wp-includes/js/thickbox/loadingAnimation.gif   [HTTP/1.1 200 OK 0ms]
GET https://wpsandbox.net/favicon.ico                                    [HTTP/2 204 0ms]
Feature Policy: Skipping unsupported feature name "clipboard-write".   web-client-content-script.js:2:41997
```

https://wpsandbox.net/1316

- Click on the Networking tab

- Clear the logs (by clicking the recycle bin icon shown just above the green status codes)
- Reload the page (by pressing the reload icon or Windows/Linux CTRL+SHIFT+R and for Mac CMD + SHIFT + R)
- Check for requests shown in red usually with 403, 404, 500, 503 HTTP (error) codes
- You can also a request filter on the right. There's a tab called XHR.
- Check the response tab (right)



Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	wpsandbox.net	dotorg-communication?_locale=user	load-scripts.php:2 (xhr)	json	969 B	315 B
200	GET	wpsandbox.net	background-updates?_locale=user	load-scripts.php:2 (xhr)	json	1.12 kB	923 B
200	GET	wpsandbox.net	loopback-requests?_locale=user	load-scripts.php:2 (xhr)	json	1.08 kB	571 B
200	GET	wpsandbox.net	https-status?_locale=user	load-scripts.php:2 (xhr)	json	1.19 kB	787 B
200	GET	wpsandbox.net	authorization-header?_locale=user	load-scripts.php:2 (xhr)	json	967 B	341 B
200	GET	wpsandbox.net	page-cache?_locale=user	load-scripts.php:2 (xhr)	json	1.37 kB	1.44 kB
200	POST	wpsandbox.net	admin-ajax.php	load-scripts.php:2 (xhr)	json	745 B	16 B

```
JSON
success: true
```

https://wpsandbox.net/1316

Check the backend for errors

It's good to also check the backend errors as there would be more information about what's causing the issue.

Start your favorite FTP program or use your hosting's file manager and navigate to wp-content folder which is in your WordPress folder (usually in public_html, www or htdocs).

Do you see wp-content/debug.log file?

If you don't you may need to enable WordPress debugging. The logging must be enabled as well. You need to turn off error displaying for ajax calls or it will break the data because php errors/warning are showing first before any data.

After you enable the WordPress debugging you need to retry the request that initially produced the error. That way you should see something in the WordPress debug.log file.

How to check if it's an ajax request in php?

if the code is running (after the init hook) then you may be able to use **wp_doing_ajax**(); function.

If that function is not available you can use the following php code to check if it's an ajax request. It checks for the presence of **X-Requested-With** request header and checks if it's equal to XMLHttpRequest. We're using strcasecmp to do a case insensitive comparison just in case.

```
$is_ajax = empty($_SERVER['HTTP_X_REQUESTED_WITH'])
    ? false
    : strcasecmp($_SERVER['HTTP_X_REQUESTED_WITH'], 'xmlhttprequest'
) == 0;
```

Common HTTP Errors

In the **Network** tab of Developer Tools, filter the requests by the **XHR** type to isolate the Ajax requests. Red-highlighted entries usually indicate issues. Common HTTP status codes to look out for include:

- **403 Forbidden:** Check for permission issues on the server.
- **404 Not Found:** Verify the Ajax URL is correct.
- **500 Internal Server Error:** Check the server logs for more details.
- **503 Service Unavailable:** The server might be temporarily down or overloaded.

Conclusion

Troubleshooting WordPress Ajax requests can be challenging sometimes due to the multiple layers involved, from JavaScript to PHP to different server settings. By systematically checking Developer Tools, ensuring proper JavaScript hooks, and examining debug logs, you can narrow down the potential causes and resolve the issue effectively. To be on the safe side we recommend that you use a staging WordPress site which can be hosted on WPSandbox or on your server. That way you won't risk breaking your live site and interrupting your visitors and clients' visits.