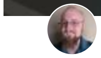

Ram Gall

April 28, 2020

High-Severity Vulnerabilities Patched in LearnPress

On March 16, 2020, LearnPress – WordPress LMS Plugin, a WordPress plugin with over 80,000 installations, patched a high-severity vulnerability that allowed subscriber-level users to elevate their permissions to those of an "LP Instructor", a custom role with capabilities similar to the WordPress "author" role, including the ability to upload files and create posts containing unfiltered HTML, both of which could be used as part of an exploit chain allowing site takeover.

Our Threat Intelligence team analyzed the vulnerability in order to create a firewall rule to protect Wordfence customers. In the process, we discovered two additional vulnerabilities. One of these vulnerabilities was almost identical in consequences to the original vulnerability in that it allowed an attacker to elevate the permissions of any user to "LP Instructor". The other allowed a logged-in user with minimal permissions, such as a subscriber, to create new pages on the site with arbitrary titles and to change the status of any existing post or page.

We privately disclosed these vulnerabilities to the plugin's author the next day, on March 17, 2020, and quickly received a response. Unfortunately, however, no patch was released for more than a month. We followed up with the plugin's author on April 16, 2020, and after receiving no response, contacted the WordPress plugins team. A few hours later, the plugin developer got back in touch and let us know that a patch was in the works. A sufficiently patched version was finally released on April 22, 2020.

We highly recommend updating to version 3.2.6.9 immediately as these security issues are fully patched in that version.

Wordfence Premium users received a new firewall rule on March 16, 2020 to protect against exploits targeting both the original vulnerability and the newly discovered flaws. Free Wordfence users received this rule on April 15, 2020.

Description: Privilege Escalation
Affected Plugin: LearnPress
Plugin Slug: learnpress
Affected Versions: < 3.2.6.9
CVE ID: CVE-2020-11511
CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:L/A:L/E:P/RL:O/RC:C
CVSS Score: 8.6(High)
Patched Version: 3.2.6.9

LearnPress is a WordPress plugin that allows site owners to create an online learning portal, including the ability to assign users as "LP Instructors" capable of adding their own course material to the site. One functionality of the plugin sends an email to the administrator whenever a user requests to become an instructor, allowing that administrator to approve the request by clicking a link. The function that handles this request runs automatically as soon as plugins are loaded and as such is always 'listening' for specific parameters:

```
function learn_press_accept_become_a_teacher() {
   $action  = ! empty( $_REQUEST['action'] ) ? $_REQUEST['action'] : '';
   $user_id = ! empty( $_REQUEST['user_id'] ) ? $_REQUEST['user_id'] : '';
   if ( ! $action || ! $user_id || ( $action != 'accept-to-be-teacher' ) ) {
       return;
   }

   if ( ! learn_press_user_maybe_is_a_teacher( $user_id ) ) {
       $be_teacher = new WP_User( $user_id );
       $be_teacher->set_role( LP_TEACHER_ROLE );
       delete_transient( 'learn_press_become_teacher_sent_' . $user_id );
       do_action( 'learn_press_user_become_a_teacher', $user_id );
       $redirect = add_query_arg( 'become-a-teacher-accepted', 'yes' );
       $redirect = remove_query_arg( 'action', $redirect );
       wp_redirect( $redirect );
   }
}

add_action( 'plugins_loaded', 'learn_press_accept_become_a_teacher' );
```

Due to the way this function was added, it was possible for an attacker to send a request to any valid location within wp-admin with the `action` parameter set to `accept-to-be-teacher` and the `user_id` parameter set to the ID of the user to be granted instructor privileges. This meant that even an unauthenticated attacker could send a request to `wp-admin/admin-post.php` containing these parameters and elevate the permissions of a user of their choice, though they would need their own user ID to take full advantage of the vulnerability.

Once a user was granted the `LP Instructor` role, they had access to create new posts, courses, lessons, and quizzes. Additionally, `LP Instructor` users were granted a capability typically reserved only for editors and administrators: the `unfiltered_html` capability, which would allow them to insert custom code into any of the pages they created. With this capability, an attacker could easily insert malicious JavaScript into any posts they created, which could then be used to redirect visitors to malvertising sites or even be used for site takeover if a logged-in administrator viewed one of these posts.

Description: Authenticated Page Creation and Status Modification
Affected Plugin: LearnPress
Plugin Slug: learnpress
Affected Versions: < 3.2.6.9
CVE ID: CVE-2020-11510
CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:L/A:H/E:F/RL:U/RC:C
CVSS Score: 7.1(High)
Patched Version: 3.2.6.9

The LearnPress plugin also handled several tasks via AJAX actions that lacked nonce checks and capability checks. It registers AJAX actions in a loop, though many of these functions did at least use capability checks:

```
$ajaxEvents = array(
    'create_page'             => false,
```

```
    'load_chart'              => false,
    'search_course_category'  => false,
    /////////////
    //'be_teacher'              => false,
    'custom_stats'            => false,
    'ignore_setting_up'       => false,
    'get_page_permalink'      => false,
    'dummy_image'             => false,
    'update_add_on_status'    => false,
    //'plugin_install'          => false,
    'bundle_activate_add_ons' => false,
    'install_sample_data'     => false,

    // Remove Notice
    'remove_notice_popup'     => false,
    // Update order status
    'update_order_status'     => false,
);
foreach ( $ajaxEvents as $ajaxEvent => $nopriv ) {
    add_action( 'wp_ajax_learnpress_' . $ajaxEvent, array( __CLASS__, $ajaxEvent ) );
```

One action, `update_order_status`, is intended to allow administrators to mark LearnPress orders as paid or refunded. Unfortunately, the function accepted any post ID and any status, even nonexistent ones. As such, it was possible for an attacker to send a request to `wp-admin/admin-ajax.php` with the `action` parameter set to `learnpress_update_order_status`, the `order_id` set to the Post ID to modify, and the `value` parameter set to the desired post status. This would allow the attacker to publish or trash any existing post or page, or even set it to a nonexistent status, at which point it would no longer appear on the site or be accessible from wp-admin, and could only be recovered by modifying its status in the database.

```
public static function update_order_status() {

    $order_id = learn_press_get_request( 'order_id' );
    $value    = learn_press_get_request( 'value' );

    $order = array(
        'ID'          => $order_id,
        'post_status' => $value,
    );

    wp_update_post( $order ) ? $response['success'] = true : $response['success'] = false;

    learn_press_send_json( $response );

    die();
}
```

The other vulnerable action calls a function, `create_page`, which is intended to be used during the setup wizard, in order to create the default pages LearnPress needs to function. This means that an attacker could send a request to `wp-admin/admin-ajax.php` with the `action` parameter set to `learnpress_create_page` and the `page_name` parameter set to a value of their choice.

```
public static function create_page() {
    $page_name = ! empty( $_REQUEST['page_name'] ) ? $_REQUEST['page_name'] : '';
    $response  = array();
    if ( $page_name ) {

        if ( $page_id = LP_Helper::create_page( $page_name ) ) {
            $response['page'] = get_post( $page_id );
            $html               = learn_press_pages_dropdown( '', '', array( 'echo' => false ) );
            preg_match_all( '!value=\"([0-9]+)\"!', $html, $matches );
            $response['positions'] = $matches[1];
            $response['html']      = '<a href="' . get_edit_post_link( $page_id ) . '" target="_blank">' . __( 'Edit
            $response['html']      .= '<a href="' . get_permalink( $page_id ) . '" target="_blank">' . __( 'View Page
        } else {
            $response['error'] = __( 'Error! Page creation failed. Please try again.', 'learnpress' );
        }
    } else {
        $response['error'] = __( 'Empty page name!', 'learnpress' );
    }
    learn_press_send_json( $response );
}
```

Although less severe, this vulnerability would still allow an attacker to publish pages with spam links in the titles, which could be used as part of a malicious SEO campaign.

Disclosure Timeline

March 16, 2020 – Wordfence Threat Intelligence discovers unpatched vulnerabilities in the LearnPress plugin while analyzing recently patched vulnerabilities. Firewall rule released for Wordfence Premium users. Initial outreach to the plugin developer.
March 17, 2020 – Plugin developer confirms appropriate inbox for handling discussion. Full disclosure of vulnerabilities is sent.
April 15, 2020 – Firewall rule becomes available to Wordfence free users.
April 16, 2020 – Followup with plugin developer as issues not yet patched.
April 20, 2020 – We reach out to the WordPress plugins team about the issue and receive a response from the plugin developer shortly afterwards.
April 22, 2020 – Sufficiently patched version released.

Conclusion

In this post, we detailed two vulnerabilities in the LearnPress plugin, including a privilege escalation vulnerability and a post creation and modification vulnerability. These flaws have been fully patched in version 3.2.6.9, and we urge users to update to the latest available version as soon as possible. Sites running Wordfence Premium have been protected against these vulnerabilities since March 16, 2020, while sites still on the free version of Wordfence have been protected since April 15, 2020. If you are currently using a site running LearnPress as a a student, please forward this advisory to the administrator of the site.

Did you enjoy this post? Share it!

Comments

3 Comments

FVobbe *
April 28, 2020
9:04 am

For those of us who are not programmers, or have any skills in modifying pages, can you explain what a user is supposed to do for protection, or what action is needed?

Ram Gall *
April 30, 2020
7:11 am

Hi FVobbe!

The best thing you can do is update the LearnPress plugin. This advice actually applies to almost all of the vulnerabilities we write about - if there's an update available, then you should update right away and you'll be protected. If no update is available, you should disable and remove the vulnerable plugin and you'll be safe. If you're using Wordfence Premium your site is also protected as soon as we create a firewall rule. If you're using the free version of Wordfence your site will receive the rule after 30 days, but you should still be safe from any future attacks against a particular vulnerability if you update or remove the vulnerable plugin immediately.

Gracious Store *

May 2, 2020
10:55 am