QuizGame: Administrative API module lets unauthenticated requests through

Actions

Closed, Resolved Public SECURITY

Assigned To

ashley

Authored By

ashley
2022-02-21 09:20:35 (UTC+0)

Tags

Security
Vuln-MissingAuthz (Tracked)
QuizGame (Backlog)
Social-Tools (Backlog)
User-RhinosF1 (Done)
SecTeam-Processed (Completed)

Referenced Files

None

Subscribers

Aklapper
ashley
lcawte
Legoktm
• MZMcBride
RhinosF1

Description

In **88754ee5b7c7f745f1172cf0238f8e65442e1bd0** (27 July 2013 (!)) I converted QuizGame to use an API module instead of the then-deprecated `sajax_*` functions provided by MediaWiki core to make the extension compatible with MediaWiki 1.21.
The old AJAX entry point, `wfQuestionGameAdmin`, which despite the name **wasn't** admin-only nor truly intended to

be such as it handles flagging, did feature a rudimentary "is the request allowed?" check; but I'm not sure if it really provided any *real* security. Either way, when creating the API module, even this check was lost.

So right now all the API module really cares about is that:
1.) The request has a valid anti-CSRF token (`new mw.Api().postWithEditToken( ... )` will take care of that)
2.) The request was POSTed
3.) The `quizaction` and `id` parameters are set and that the former corresponds to one of the actions handled by the module's `switch()` loop

As long as those requirements are met, an attacker can easily abuse the QuizGame administrative API to their nefarious purposes - `quizadmin` rights not needed!

Hilariously, though, the UI in `/extensions/QuizGame/includes/specials/SpecialQuizGameHome.php` **does** perform correct user right checks for the admin panel and whatnot and redirects unauthorized users to the main quiz game landing page.

Quick patch to add block and proper user rights checking to the QuizGame admin API (while allowing all users to flag quizzes):

```
diff --git a/includes/api/ApiQuizGame.php b/includes/api/ApiQuizGame.php
index 3236d7e..56d5a82 100644
--- a/includes/api/ApiQuizGame.php
+++ b/includes/api/ApiQuizGame.php
@@ -34,6 +34,21 @@ class ApiQuizGame extends ApiBase {
                // ApiBase's getDB() supports only slave connections, lame...
                $dbw = wfGetDB( DB_PRIMARY );

+               // Fail early if the user is sitewide blocked.
+               // (This snippet copied from MW core /includes/api/ApiTag.php)
+               $block = $user->getBlock();
+               if ( $block && $block->isSitewide() ) {
+                       $this->dieBlocked( $block );
+               }
+
+               // Allow non-quizadmins to use the flagging feature but require quizadmin
+               // rights for all other stuff
+               if ( $action !== 'flagItem' ) {
+                       if ( !$user->isAllowed( 'quizadmin' ) ) {
+                               $this->dieWithError( 'badaccess-group0' );
+                       }
+               }
+
                switch ( $action ) {
                        case 'unprotectItem':
                                $dbw->update(
```

Details

	Project	Subject
	mediawiki/extensions/QuizGame	[SECURITY] QuizGame admin API module should check for the 'quizadmin'...

Customize query in gerrit

Related Objects

Mentions

Mentioned In
~~T305209: Write and send supplementary release announcement for extensions and skins with security patches (1.35.7/1.37.3/1.38.2)~~
~~T297839: Write and send supplementary release announcement for extensions and skins with security patches (1.35.6/1.36.4/1.37.2)~~

Mentioned Here
rEQGA88754ee5b7c7: QuizGame: made compatible with MW 1.21.1.

ashley created this task. 2022-02-21 09:20:35 (UTC+0)

Restricted Application added a subscriber: **Aklapper**. · View Herald Transcript 2022-02-21 09:20:36 (UTC+0)

ashley claimed this task. 2022-02-21 09:21:19 (UTC+0)

ashley added projects: **Vuln-MissingAuthz**, **QuizGame**, **Social-Tools**.

ashley added a subscriber: **RhinosF1**. 2022-02-21 21:06:57 (UTC+0)

Adding **@RhinosF1** of Miraheze per request since some Miraheze wikis use QuizGame.

RhinosF1 added a project: **User-RhinosF1**. 2022-02-21 21:08:19 (UTC+0)

RhinosF1 added a comment. 2022-02-21 22:11:00 (UTC+0)

```
From 26cab2b8ca3fba27849c7453f48b7f9189e89fc9 Mon Sep 17 00:00:00 2001
From: www-data <www-data@miraheze.org>
Date: Mon, 21 Feb 2022 22:06:16 +0000
Subject: [SECURITY] QuizGame: Administrative API module lets
 unauthenticated requests through

---
 includes/api/ApiQuizGame.php | 15 +++++++++++++++
 1 file changed, 15 insertions(+)

diff --git a/includes/api/ApiQuizGame.php b/includes/api/ApiQuizGame.php
index 3236d7e..081e64e 100644
--- a/includes/api/ApiQuizGame.php
+++ b/includes/api/ApiQuizGame.php
@@ -34,6 +34,21 @@ class ApiQuizGame extends ApiBase {
                // ApiBase's getDB() supports only slave connections, lame...
                $dbw = wfGetDB( DB_PRIMARY );

+               // Fail early if the user is sitewide blocked.
+               // (This snippet copied from MW core /includes/api/ApiTag.php)
+               $block = $user->getBlock();
+               if ( $block && $block->isSitewide() ) {
+                       $this->dieBlocked( $block );
+               }
+
```

```
+		// Allow non-quizadmins to use the flagging feature but require quizadmin
+		// rights for all other stuff
+		if ( $action !== 'flagItem' ) {
+			if ( !$user->isAllowed( 'quizadmin' ) ) {
+				$this->dieWithError( 'badaccess-group0' );
+			}
+		}
+
		switch ( $action ) {
			case 'unprotectItem':
				$dbw->update(
--
2.30.2
```

didn't apply for me, ended up working with sudo -u www-data git apply --ignore-space-change --ignore-whitespace ~/quizgame.patch

That's the new patch file

sbassett edited projects, added **SecTeam-Processed**; removed **Security-Team**. 2022-02-22 16:33:37 (UTC+0)

sbassett mentioned this in ~~**T297839: Write and send supplementary release announcement for extensions and skins with security patches (1.35.6/1.36.4/1.37.2)**~~.

Legoktm added a subscriber: **Legoktm**. 2022-02-24 02:06:53 (UTC+0)

The patch looks correct to me, +2. One nit:

```
+		if ( $action !== 'flagItem' ) {
+			if ( !$user->isAllowed( 'quizadmin' ) ) {
+				$this->dieWithError( 'badaccess-group0' );
+			}
+		}
```

These two if blocks can be merged: `if ( $action !== 'flagItem' && !$user->isAllowed( 'quizadmin' ) )`.

(Aso in the future it's better to post patches as attachments generated with `git format-patch HEAD~1`, so that any weird whitespace, etc. are preserved.)

Also, also, I noticed that SpecialQuizGameHome is checking for *any* block, not a sitewide block. That can be fixed publicly, I just looked because you mentioned it implemented the checks correctly.

Reedy changed the visibility from "**Custom Policy**" to "Public (No Login Required)". 2022-03-13 18:38:53 (UTC+0)

Reedy changed the edit policy from "**Custom Policy**" to "All Users".

• **MZMcBride** added a subscriber: • **MZMcBride**. 2022-03-14 04:46:40 (UTC+0)

ashley closed this task as *Resolved*. 2022-03-28 17:13:12 (UTC+0)

Finally closing this as the patch was merged about a month ago (oops). @Legoktm 's note regarding Special:QuizGameHome is worth investigating, but that's a separate matter altogether. (☂ Social-Tools basically have no knowledge and thus no support for the concept of partial blocks; you're either blocked or not, social tools don't currently care about how partial the block is.)

Maintenance_bot moved this task from **Radar** to **Done** on the **User-RhinosF1** board. 2022-03-28 17:15:31 (UTC+0)

sbassett mentioned this in ~~**T305209: Write and send supplementary release announcement for extensions and skins with security patches (1.35.7/1.37.3/1.38.2)**~~. 2022-04-18 16:12:01 (UTC+0)