tenable

Multiple Vulnerabilities in Icegram Email Subscribers & Newsletters Plugin for WordPress

Medium

← View More Research Advisories

Synopsis

CVE-2020-5767: Cross-site Request Forgery in send_test_email()

A cross-site request forgery vulnerability exists in the send_test_email() function in class-es-mailer.php due to a lack of CSRF protection mechanisms. An unauthenticated, remote attacker can exploit this issue by convincing a user to click a specially crafted URL, to send emails from the affected user's WordPress email account.

Attached you'll find a Proof of Concept (PoC) called csrf_poc.html which demonstrates how an attacker could exploit this issue to send an email from the admin email account of an affected webpage.

The following is an example of how to use the PoC:

1. Open csrf_poc.html with a text editor and set the following variables
    - Set targetUrl to the webpage containing the affected plugin
    - Set email to the address you wish to send an email
    - Set the subject and content for the email
2. Host csrf_poc.html on a web server
3. Login as an admin to the wordpress webpage from a browser
4. Ensure the mail settings are configured so the admin can send emails
5. While maintaining a logged in session, open another browser tab and visit the csrf_poc.html webpage
6. Verify if an email was sent to the supplied email address

CVE-2020-5768: Authenticated SQL injection in es_newsletters_settings_callback()

An SQL injection vulnerability exists in the es_newsletters_settings_callback() function in class-es-newsletters.php. The value of $broadcast_id which is assigned from a user-controlled variable is not properly sanitized and validated prior to being included in an SQL query. This allows a remote, authenticated attacker with administrative privileges, to exploit this issue via a specially crafted request to disclose potentially sensitive information from the WordPress database.

Attached you'll find a proof of concept script called sqli_info_disclosure_poc.py which sends multiple specially crafted requests to the server to disclose the hash of the admin's password from the WordPress database.

The following is an example of how to use the PoC:

```
python3 sqli_info_disclosure_poc.py http://192.168.1.190/wordpress admin password
```

Solution

Upgrade to version 4.5.1.

Additional References

https://plugins.trac.wordpress.org/browser/email-subscribers/tags/4.5.1/readme.txt
https://wpvulndb.com/vulnerabilities/10321
https://wpvulndb.com/vulnerabilities/10322

Disclosure Timeline

06/22/2020 - Asked for Icegram's security contact via their webform.
06/23/2020 - Icegram customer care responds. We can send the email to hello@icegram.com.
06/23/2020 - Tenable sends the vulnerability report to Icegram. 90-day is Sept 21, 2020.
06/30/2020 - Tenable follows up to ensure report was received.
07/07/2020 - Tenable follows up via Icegram's webform to verify if the report was received.
07/10/2020 - Icegram will fix in the next release.
07/10/2020 - Tenable asks when the next release might be.
07/13/2020 - Icegram will release July 15.
07/13/2020 - Tenable thanks Icegram for the update. Notifies them of intent to publish research advisory and CVEs.

Risk Information

CVSSv2 Base / Temporal Score: 4.3
CVSSv2 Vector: AV:N/AC:M/Au:N/C:N/I:P/A:N
Affected Products: 4.4.8
Risk Factor: Medium

Advisory Timeline

07/16/2020 - Advisory published.
07/20/2020 - Added links to wpvulndb

tenable