9

Reduced purmations on encryption

Share:

TIMELINE

realguyman0 submitted a report to Nextcloud. Apr 18th (3 years ago)

`OC\Security\SecureRandom::generate`

Reduced Permutations

`OC\Security\SecureRandom::generate` will by default use `a-Z0-9+/` (64 bytes) character set. The numbers are not predictable, due to the use of `random_int`.

Most notably the `OC\Security\Crypto::encrypt` method uses an IV with a length of 16 bytes. It is chosen randomly via `OC\Security\SecureRandom::generate` with the default character set. There are 256 possible bytes, but in this case it is *actually* 64 bytes. The permutations is 64^16 (instead of 256^16), which equates to a 12-byte, or 96-bit IV (instead of the expected 16-byte, or 128-bit IV). **Use raw bytes when doing cryptographic operations, via `random_bytes`.**

Do not use `OC\Security\Crypto::generate` for cryptographic keys.

Cache Timing Attacks

It is *potentially* vulnerable to cache timing attacks because the secret number is used as an index to look up a byte value in string. Read more about cache-timing attacks here.

Impact

1. Reduced permutations increase the chances of IV re-use (which can destroy confidentially), and bring encryption key strength down (chances are still too low with a 256-bit encryption key).

2. If the complex cache timing attack vector exists, and is abused: it is possible to determine secret values generated with `OC\Security\SecureRandom::generate`.

BOT: posted a comment. Apr 18th (3 years ago)

Thanks a lot for reporting this potential issue back to us!

Our security team will take a look at this issue as soon as possible. We will reply to your report within 72 hours, usually much faster. For obvious reasons we'd like to ask you to not disclose this issue to any other party.

realguyman0 posted a comment. Apr 21st (3 years ago)

If there is anything wrong with the report, please let me know. Have a good day.

nickvergessen Nextcloud staff posted a comment. Apr 22nd (3 years ago)

We are currently still checking and investigating the details of your report.

rullzer posted a comment. May 11th (3 years ago)

Hi @lynn-stephenson ,

I'm tackling this now. I'll get back once I have a PR.

Cheers,
--Roeland

rullzer posted a comment. May 11th (3 years ago)

Hi @lynn-stephenson,

https://github.com/nextcloud/server/pull/20915 Should fix this I believe.
Mind to have a quick look?

Cheers,
--Roeland

rullzer changed the status to **Triaged**. May 11th (3 years ago)

Nextcloud rewarded realguyman0 with a **$150** bounty. May 13th (3 years ago)

Congratulations! We have determined this to be eligible for a reward of $150.

Thanks a lot for making the internet a safer place and keep hacking. Please keep in mind that we didn't release a patch for the vulnerability yet, so please do not share this information with any third-parties.

nickvergessen Nextcloud staff closed the report and changed the status to **Resolved**. May 13th (3 years ago)

Thanks a lot for your report again. This has been resolved in our upcoming maintenance releases and we're working on the advisories at the moment.

Please let us know how you'd like to be credited in our official advisory. We require the following information:

- Name / Pseudonym
- Email address (optional)
- Website (optional)
- Company (optional)

realguyman0 posted a comment. May 23rd (3 years ago) nickvergessen Nextcloud staff posted a comment. Jun 8th (3 years ago)

Advisory will be published around 4th of july on https://nextcloud.com/security/advisory/?id=NC-SA-2020-023

nickvergessen Nextcloud staff changed the report title from **Reduced Purmations and Potential Cache Timing Attacks** to **Reduced purmations on encryption**. Jun 8th (3 years ago)

nickvergessen Nextcloud staff requested to disclose this report. Sep 28th (2 years ago)

This report has been disclosed. Oct 28th (2 years ago)