# TYPO3 Core - Feature #73050

## Add a CSPRNG to TYPO3

2016-01-31 21:07 - Christian Futterlieb

| | | | |
|---|---|---|---|
| **Status:** | Closed | **Start date:** | 2016-01-31 |
| **Priority:** | Should have | **Due date:** | |
| **Assignee:** | | **% Done:** | 100% |
| **Category:** | | **Estimated time:** | 0.00 hour |
| **Target version:** | | | |
| **PHP Version:** | | **Complexity:** | |
| **Tags:** | | **Sprint Focus:** | |

**Description**

I'd like to bring some crypto-related code into TYPO3 core. First topic: a CSPRNG

As always in cryptography, using a widely used/adopted/reviewed library should be the way to go. This one seems to do a good job: https://github.com/paragonie/random_compat. It is a PHP 5.x polyfill for PHP 7's random_bytes() and random_int().

In the proposed change, I cover following tasks:

1. remove all the GeneralUtility::generateRandomBytesXYZ methods, because they're covered by random_compat. Leave just the fallback method in place (and slightly improve it)
2. Add a simple API in \TYPO3\CMS\Core\Crypto\Random to produce crypto-save random bytes, integers and hex strings
3. Add a check to \TYPO3\CMS\Install\SystemEnvironment\Check that creates a warning, when no CSPRNG can be generated on the system (and the fallback will be used therefor). From the crypto-view it would be much better to fail instead of just warn.. please share your opinion on this!

Furthermore I'd like to come up with things like a Crypto\Hash class to do proper hashing and verifying, a Crypto\Password class for password-related stuff, a saltedpasswords salt and so on. I'll open new tasks for these ideas when they're ready.

**Related issues:**

| | | |
|---|---|---|
| Related to TYPO3 Core - Task #67268: Introduce RandomUtility and move methods | **Closed** | **2015-06-03** |
| Related to TYPO3 Core - Bug #37780: Possibility to get duplicate sessionId fo... | **Closed** | **2012-06-06** |
| Related to TYPO3 Core - Feature #73164: Add crypto-safe hashing API | **Rejected** | **2016-02-06** |
| Related to TYPO3 Core - Feature #73456: Timing attack vulnerability in Hash c... | **Closed** | **2016-02-15** |
| Related to TYPO3 Core - Task #72292: PHP7 >= only | **Closed** | **2015-12-17** |

### Associated revisions

**Revision 5bc1aed1 - 2016-03-05 17:53 - Christian Futterlieb**

[FEATURE] Add a CSPRNG

- Add a simple API for generation of random bytes, integers and hex-strings: Crypto\Random
- Deprecate the methods GeneralUtility::generateRandomBytes() and GeneralUtility::getRandomHexString()
- Replace occurrences of the deprecated methods

Change-Id: If4d6daa00138eac791954a4fd9a4fc26a79ddf07
Resolves: #73050
Releases: master
Reviewed-on: https://review.typo3.org/46507
Reviewed-by: Christian Kuhn <lolli@schwarzbu.ch>
Tested-by: Christian Kuhn <lolli@schwarzbu.ch>
Reviewed-by: Oliver Hader <oliver.hader@typo3.org>
Tested-by: Oliver Hader <oliver.hader@typo3.org>

### History

**#1 - 2016-01-31 21:23 - Georg Ringer**

Thanks for creating this issue.

As PHP7 will be required for version 8, there would be no need for the polyfill.

#### #2 - 2016-01-31 21:33 - Christian Futterlieb

Great you mentioned that, I was just about pushing the stuff to gerrit ;-)

What about version 7.6? I'd like to have a good CSPRNG in there too.. So, how should I proceed? Because: in PHP7 it's just creating the Crypto\Random class and changing GeneralUtility accordingly. But with PHP<7 there comes in some additional code and the system env check. Should I open a new issue for TYPO3 7.6 though?

#### #3 - 2016-02-05 19:04 - Gerrit Code Review

*- Status changed from New to Under Review*

Patch set 1 for branch **master** of project **Packages/TYPO3.CMS** has been pushed to the review server.
It is available at https://review.typo3.org/46507

#### #4 - 2016-02-05 19:14 - Christian Futterlieb

Hi
I just pushed the change for master. To get the CSPRNG into 7.6 LTS, the changes would be very different to these in the proposed one:

- No deprecation
- Leaving in a (unsafe) fallback
- According tests
- Adding system check (warn when the CSPRNG is not available on the system)
- Adding the polyfill

Can these things be covered by 'backporting' the change to 7.6? Or do I have to add a separate change? Any statements from anybody, please?

#### #5 - 2016-02-10 18:21 - Gerrit Code Review

Patch set 2 for branch **master** of project **Packages/TYPO3.CMS** has been pushed to the review server.
It is available at https://review.typo3.org/46507

#### #6 - 2016-02-11 18:46 - Gerrit Code Review

Patch set 3 for branch **master** of project **Packages/TYPO3.CMS** has been pushed to the review server.
It is available at https://review.typo3.org/46507

#### #7 - 2016-02-15 15:36 - Helmut Hummel

Christian Futterlieb wrote:

> What about version 7.6? I'd like to have a good CSPRNG in there too..

Seriously: What exactly is wrong with the current Random methods in 7.6?

#### #8 - 2016-02-15 17:40 - Christian Futterlieb

Helmut Hummel wrote:

> Seriously: What exactly is wrong with the current Random methods in 7.6?

Glad you asked that, nobody else did.. The key part is that it is not **crypto-save**, say: unusable for cryptographic applications (like a login system). At the moment there are some problems in GeneralUtility::generateRandomBytes(), imo:

1. preferrably uses openssl_random_pseudo_bytes() (which can fail to produce secure data):
   - without checking 2nd parameter
   - without checking for PHP versions, that contain this bug: https://bugs.php.net/bug.php?id=70014
2. uses mcrypt_create_iv() (which can fail to produce secure data), see https://bugs.php.net/bug.php?id=52523, https://bugs.php.net/bug.php?id=55169
3. uses a silent fallback without further noticing anybody

As a consequence of this, TYPO3 cannot be treated as save (from a cryptographic view) by default. Sure, one is able to configure his system to make TYPO3 produce secure data, but this needs knowledge of core-internals.

Therefor, my proposal is to implement:

1. a 'real' CSPRNG (with random_bytes() for PHP>=7 and random_compat polyfill for the rest). Like this we have the advantage of code written by cryptographers and an api for everybody inside TYPO3
2. the secure hashing API, see #73164

#### #9 - 2016-02-15 18:01 - Christian Futterlieb

And: just to take into consideration: these are the security-related components in TYPO3:

- core (authentication, managing the sessionIds, formProtection, generating sessionTokens). Wide parts of the core depend on the security of these.
- saltedpasswords (every Salt class besides Blowfish)
- felogin (generating 'reset-password' hashes)
- install (generating the TYPO3 encryptionKey [!])

TL/DR: Many core parts of TYPO3 rely on the unpredictability of the RNG. Imo it is very important to truly fulfil this responsibility.

#### #10 - 2016-02-25 19:15 - Gerrit Code Review

Patch set 4 for branch **master** of project **Packages/TYPO3.CMS** has been pushed to the review server.
It is available at https://review.typo3.org/46507

#### #11 - 2016-02-25 20:42 - Gerrit Code Review

Patch set 5 for branch **master** of project **Packages/TYPO3.CMS** has been pushed to the review server.
It is available at https://review.typo3.org/46507

#### #12 - 2016-03-05 17:49 - Gerrit Code Review

Patch set 6 for branch **master** of project **Packages/TYPO3.CMS** has been pushed to the review server.
It is available at https://review.typo3.org/46507

#### #13 - 2016-03-05 17:57 - Christian Futterlieb

*- Status changed from Under Review to Resolved*

*- % Done changed from 0 to 100*

Applied in changeset 5bc1aed17b811a3f58bb50f37dbb1cf903593d4f.

#### #14 - 2018-10-02 11:20 - Benni Mack

*- Status changed from Resolved to Closed*