Reset password token not set to null after reset password

Low lchrusciel published GHSA-mf3v-f2qq-pf9g on Mar 14

Package

php sylius/sylius (Composer)

Affected versions	Patched versions
>=1.10 <1.10.11 || >=1.11 < 1.11.2	1.10.11, 1.11.2

Description

Impact

The reset password token was not set to null after the password was changed. This is causing behaviour in which the same token can be used several times, so it can result in a leak of the existing token and an unauthorised password change.

Patches

The issue is fixed in versions: 1.10.11, 1.11.2 and above

Workarounds

You have to overwrite your `Sylius\Bundle\ApiBundle\CommandHandler\ResetPasswordHandler` class using this code:

```php
<?php
declare(strict_types=1);

namespace App\CommandHandler\Account;

use Sylius\Bundle\ApiBundle\Command\Account\ResetPassword;
use Sylius\Component\Core\Model\ShopUserInterface;
use Sylius\Component\Resource\Metadata\MetadataInterface;
use Sylius\Component\User\Repository\UserRepositoryInterface;
use Sylius\Component\User\Security\PasswordUpdaterInterface;
```

```php
use Symfony\Component\Messenger\Handler\MessageHandlerInterface;
use Webmozart\Assert\Assert;

final class ResetPasswordHandler implements MessageHandlerInterface
{
    private UserRepositoryInterface $userRepository;
    private MetadataInterface $metadata;
    private PasswordUpdaterInterface $passwordUpdater;

    public function __construct(
        UserRepositoryInterface $userRepository,
        MetadataInterface $metadata,
        PasswordUpdaterInterface $passwordUpdater
    ) {
        $this->userRepository = $userRepository;
        $this->metadata = $metadata;
        $this->passwordUpdater = $passwordUpdater;
    }

    public function __invoke(ResetPassword $command): void
    {
        /** @var ShopUserInterface|null $user */
        $user = $this->userRepository->findOneBy(['passwordResetToken' => $command->resetPasswor

        Assert::notNull($user, 'No user found with reset token: ' . $command->resetPasswordToken

        $resetting = $this->metadata->getParameter('resetting');
        $lifetime = new \DateInterval($resetting['token']['ttl']);

        if (!$user->isPasswordRequestNonExpired($lifetime)) {
            throw new \InvalidArgumentException('Password reset token has expired');
        }

        if ($command->resetPasswordToken !== $user->getPasswordResetToken()) {
            throw new \InvalidArgumentException('Password reset token does not match.');
        }

        $user->setPlainPassword($command->newPassword);

        $this->passwordUpdater->updatePassword($user);
        $user->setPasswordResetToken(null);
    }
}
```

And register it in container:

```yaml
App\CommandHandler\Account\ResetPasswordHandler:
    arguments:
        - '@sylius.repository.shop_user'
        - !service
            class: Sylius\Component\Resource\Metadata\MetadataInterface
            factory: [ '@sylius.resource_registry', 'get' ]
            arguments:
```

```
                - 'sylius.shop_user'
        - '@sylius.security.password_updater'
    tags:
        - { name: messenger.message_handler, bus: sylius.command_bus }
        - { name: messenger.message_handler, bus: sylius_default.bus }

Sylius\Bundle\ApiBundle\CommandHandler\ResetPasswordHandler:
    class: App\CommandHandler\Account\ResetPasswordHandler
    arguments:
        - '@sylius.repository.shop_user'
        - !service
            class: Sylius\Component\Resource\Metadata\MetadataInterface
            factory: [ '@sylius.resource_registry', 'get' ]
            arguments:
                - 'sylius.shop_user'
        - '@sylius.security.password_updater'
    tags:
        - { name: messenger.message_handler, bus: sylius.command_bus }
        - { name: messenger.message_handler, bus: sylius_default.bus }
```

For more information

If you have any questions or comments about this advisory:

- Open an issue in Sylius issues
- Email us at security@sylius.com

Severity

Low

CVE ID

CVE-2022-24743

Weaknesses

No CWEs