

AUDIT : EDEN BOHBOT

Problème de Versionnage :

Il y a un problème de versionnage au le gitignore ; en effet le .env n'y est pas et le vendor est en commentaire

```
/.idea
#> symfony/framework-bundle ###
/.env.local
/.env.local.php
/.env.*.local
/config/secrets/prod/prod.decrypt.private.php
/public/bundles/
/var/
#/vendor/
###< symfony/framework-bundle ###

###> phpunit/phpunit ###
/phpunit.xml
.phpunit.result.cache
###< phpunit/phpunit ###

###> symfony/phpunit-bridge ###
.phpunit.result.cache
/phpunit.xml
###< symfony/phpunit-bridge ###
```

Conséquence :





Si mon projet est stocké sur github, quelqu'un de mal intentionné pourrait récupérer ma base de données

Solution :

Rajouter dans le gitignore le .env et les vendors

Problème au niveau de l'accès à la base de données :

1. Dans le .env on peut voir que l'accès a la base de données n'est pas sécurisé, en effet l'utilisateur est root qui possède tous les privilèges.

| | | | | | | | |
|--------------------------|------|-----------|--------|----------------|-----|--|--|
| <input type="checkbox"/> | root | % | global | ALL PRIVILEGES | Oui |  Éditer les privilèges |  Exporter |
| <input type="checkbox"/> | root | localhost | global | ALL PRIVILEGES | Oui |  Éditer les privilèges |  Exporter |

De plus il ne possède pas de mot de passe

```
mysql://root@localhost/app_pasdebol?serverVersion=mariadb-10.5.8&charset=utf8mb4"
```

Conséquence :

Avec le problème précédant, un hacker qui aurait accès a nos .env verrait l'utilisateur et rentrerait dans notre base de données sans difficulté et donc avec tous les privilèges qu'il aurait pourrait faire ce qu'il veut

Solution :

Il faudrait créer plusieurs utilisateurs avec des mots de passe fort et unique pour accéder à notre base de données et leur donner différents privilèges

2. Problème de la route extract :

N'importe qui pourrait avoir accès à la route extract qui représente la base de données en format json

```

"data": [
  {
    "user_account": {
      "email": "company@mail.dev",
      "password": "$2y$13$2C.VEvR2603Lx3CNKK7AweMUrKyUjIzIkUh9pK2aRFhnrJi",
      "plainPassword": "password"
    },
    "company_data": {
      "name": "Picard SARL",
      "siret": "21828981700023",
      "address": "59, avenue Millet\n51 532 Charrier-la-Forêt"
    },
    "contributions_data": [
      {
        "ctb_data": {
          "year": "2022",
          "amount": 12332,
          "base": 72544
        },
        "payment_data": {
          "card_owner": "Aimée Peron",
          "card_numbers": "4532333625689173",
          "card_expiration": "10/24",
          "card_code": "925"
        }
      },
      {
        "ctb_data": {
          "year": "2023",

```

Conséquence :

N'importe qui pourrait avoir accès à l'information de la base de données, avec les infos sur les utilisateurs ainsi que leur mot de passe qui est en clair

Solution :

Enlever complètement cette partie du code

Problème lors de la création d'un nouveau utilisateur :

1. Lors d'une inscription, l'utilisateur doit remplir un champ mot de passe, cependant les critères de ce mot de passe n'est pas sécurisé il ne doit contenir seulement 6 caractères, ce qui ne respecte pas les recommandations de la CNIL

```

->add( child: 'plainPassword', type: PasswordType::class, [
  // instead of being set onto the object directly,
  // this is read and encoded in the controller
  'mapped' => false,
  'attr' => ['autocomplete' => 'new-password'],
  'constraints' => [
    new NotBlank([
      'message' => 'Please enter a password',
    ]),
    new Length([
      'min' => 6,
      'minMessage' => 'Your password should be at least {{ limit }} characters',
      // max length allowed by Symfony for security reasons
      'max' => 4096,
    ]),
  ],
),
1,
1)

```

Conséquence :

L'utilisateur pourrait donner un mot de passe vraiment simple comme abcdefg qui pourrait se trouver dans un dictionnaire de mot de passe, et donc facile a trouvé

Solution :

Rajouter dans le RegistrationType d'autre contraintes comme minimum une majuscule et une minuscule, un chiffre et un caractère spécial ainsi qu'un minimum de 12 caractères

2. Lors d'une inscription, le rôle de l'utilisateur n'est pas attribué

```
public function getRoles(): array
{
    $roles = $this->roles;

    return array_unique($roles);
}

public function setRoles(array $roles): self
{
    $this->roles = $roles;

    return $this;
}
```

Conséquence :

Si le rôle n'est pas attribué, les cas d'utilisation de cahier des charges ne pourront pas être mis en pratique

Solution :

Attribué un rôle à chaque nouvelle inscription.

3. Dans la base de données, nous avons deux colonnes pour les mots de passe ; les mots de passe crypté ce qui est une bonne pratique, cependant a cote nous avons aussi une colonne avec les mots de passe non-cryptés

| id | company_id | email | roles (DC2Type=json) | password | plain_password |
|----|------------|----------------|-------------------------|--|----------------|
| 1 | NULL | admin@mail.dev | ["ROLE_ADMIN"] | \$2y\$13\$.n4C543wpXQFNSfGgBS0NunFc/zIJuXJ2i/9Ty5YAj1... | password |

En effet dans le registration controller nous demandons d'enregistre dans notre base de données le 'plainPassword'

```

class RegistrationController extends AbstractController
{
  #[Route('/register', name: 'app_register')]
  public function register(Request $request, UserPasswordHasherInterface $userPasswordHasher, UserAuthenticatorInterface $userAuthenticator)
  {
    $user = new User();
    $form = $this->createForm( type: RegistrationFormType::class, $user);
    $form->handleRequest($request);

    if ($form->isSubmitted() && $form->isValid()) {

      // encode the plain password
      $user->setPassword(
        $userPasswordHasher->hashPassword(
          $user,
          $form->get('plainPassword')->getData()
        )
      );

      $user->setPlainPassword($form->get('plainPassword')->getData());
      $user->setCompany($this->makeCompany($form->all()));

      $entityManager->persist($user);
      $entityManager->flush();
    }
  }
}

```

Conséquence :

Un hacker qui rentre dans notre base de données pourrait rapidement récupérer un compte client et se connecter en tant que telle et s'envoyer de l'argent

Solution :

Supprimer cette partie du code et seulement laisser dans la base de données le mot de passe en crypté

Problème au niveau des accès contrôles :

Dans security.yaml les ACL ne sont pas définis en effet ils sont commentés. Il n'est donc pas défini en tant que quel rôle je peux accéder à quelles routes

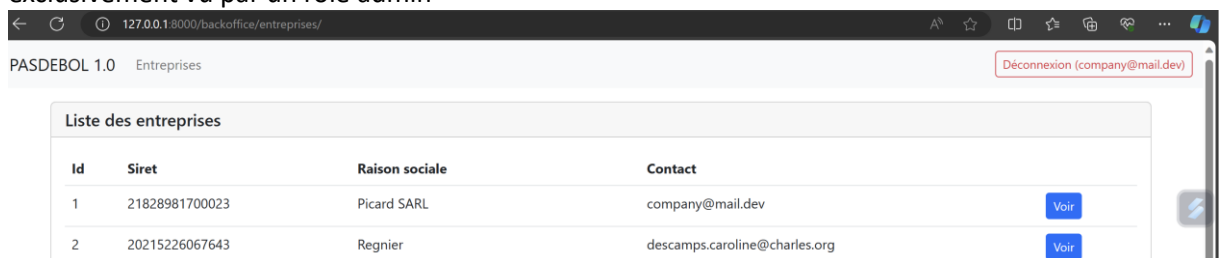
```

access_control:
  # - { path: ^/admin, roles: ROLE_ADMIN }
  # - { path: ^/profile, roles: ROLE_USER }

```

Conséquence :

Une personne qui n'a pas de rôle admin peut accéder au page qui sont censé être exclusivement vu par un rôle admin



Dans la photo je suis connecté en tant que company@mail.dev et je peux avoir accès à la route des admin. De plus je peux aussi voir les entreprises et leurs déclarations

PASDEBOL 1.0 Entreprises

Deconnexion (company@mail.dev)

Picard SARL

| | | | |
|-----------------------|--|------------------|-----------------------------|
| Siret | 21828981700023 | Contact | company@mail.dev |
| Raison Sociale | Picard SARL | Téléphone | +33 4 82 79 79 26 |
| Adresse | 59, avenue Millet 51 532 Charrier-la-Forêt | Accès | company@mail.dev / password |

Suivi des paiements

| Année de contribution | Base de calcul | Montant | Statut | Moyen de paiement |
|-----------------------|----------------|---------|--------|-------------------------|
| 2022 | 72544 € | 12332 € | Payée | Détails |

Solution :

Mettre en place des ACL afin de bien laisser aux personnes autorisées les pages qu'elles peuvent consulter

Problème de cloisonnement des espaces :

Lorsque je suis connecté en tant qu'un certain utilisateur je peux visualiser les contributions d'une autre personne que moi ainsi que les modifier et les payer

```
#[Route('/contribution/{id}', name: 'app_company_contribution_show', requirements: ['id' => '\d+'], methods: ['GET'])]  
public function show(Contribution $contribution): Response  
{  
    return $this->render(view: 'company/contribution/show.html.twig', [  
        'contribution' => $contribution,  
    ]);  
}
```

Conséquence :

1.0

Deconnexion (descamps.caroline@charles.org)

claration pour l'année 2022

| | | | |
|--------------------------------|------------|--------------------------|------------------|
| Date de déclaration | 21/12/2023 | Date de paiement | 21/12/23 20:57 |
| Base de calcul déclarée | 37757 € | Nom | Laurence Rocher |
| Montant dû | 6418 € | Numéro carte | 2561580769752781 |
| | | Date d'expiration | 07/25 |
| | | Code | 554 |

Je suis connecté en tant que madame descamps, j'ai accès à ma contribution si je change dans l'URL l'id de contribution j'aurai accès à la contribution d'une autre personne

1.0

Deconnexion (descamps.caroline@charles.org)

our l'année 2022

| | | | |
|--------------------------------|------------|--------------------------|------------------|
| Date de déclaration | 21/12/2023 | Date de paiement | 21/12/23 20:57 |
| Base de calcul déclarée | 73275 € | Nom | Simone Gomes |
| Montant dû | 12456 € | Numéro carte | 4485801919543238 |
| | | Date d'expiration | 10/26 |
| | | Code | 919 |

Ici j'ai accès à la contribution de monsieur Gomes

Solution :

Dans les méthodes ajouter des `$this->getUser()`

Afin de cloisonner les espaces pour que seule les personnes connectées peuvent avoir accès à leur données

Problème lors d'un paiement :

1. Je peux payer avec n'importe qu'elle carte, même une fausse, une carte qui aurait expiré par exemple

| id | created_at (DC2Type:datetime_immutable) | card_owner | card_numbers | card_expiration_date | card_code | card_type |
|----|--|------------|--------------|----------------------|-----------|-----------|
| 12 | 2023-12-22 10:05:27 | ezra | 57684568 | 11/01/2022 | 567 | VISA |

Conséquence :

Une personne pourrait rentrer de fausses coordonnées bancaires

Solution :

```
public function __construct()
{
    $this->createdAt = new \DateTimeImmutable( 'now', new \DateTimeZone( 'Europe/Paris' ));
}
```

Rajouter dans le constructeur de l'entité paiement une condition que la date d'expiration de la carte ne doit pas être inférieure à la date actuelle

2. Lorsque j'effectue un paiement, au niveau des informations a remplir sur nos carte bancaire je peux mettre des information de type chaine de caractère alors qu'on devrait s'attendre à des nombres. Il y a un donc un manque d'assert

| id | created_at (DC2Type:datetime_immutable) | card_owner | card_numbers | card_expiration_date | card_code | card_type |
|----|--|------------|--------------|----------------------|-----------|-------------|
| 13 | 2023-12-22 10:17:37 | eden | 96435466960 | 24/09/2026 | salut | MASTER CARD |

| id | created_at (DC2Type:datetime_immutable) | card_owner | card_numbers | card_expiration_date | card_code | card_type |
|----|--|------------|--------------|----------------------|-----------|-----------|
| 14 | 2023-12-22 10:30:19 | susanne | 98354678 | coucou | 567 | VISA |

Conséquence :

Une personne pourrait rentrer de fausses coordonnées bancaires

Solution :

Rajouter des asserts dans les entités afin de ne recevoir que des typages attendus dans les champs prévus.

Problème de RGPD :

1. Un admin a le droit peut voir les informations du compte d'une entreprise cependant il a aussi accès à leur email et mot de passe

Bailly Simon SA

| | | | |
|-----------------------|--|------------------|-----------------------------------|
| Siret | 51337532900671 | Contact | jerome.gilbert@sfr.fr |
| Raison Sociale | Bailly Simon SA | Téléphone | 0976013577 |
| Adresse | 8, avenue Pénélope Collet 34699 Moulin | Accès | jerome.gilbert@sfr.fr / password7 |

Suivi des paiements

| Année de contribution | Base de calcul | Montant | Statut | Moyen de paiement |
|-----------------------|----------------|---------|------------|-------------------|
| 2022 | 52762 € | 8969 € | En attente | |

[Retour](#)

Conséquence :

Il y a un manque de confidentialité ;

Solution :

```

<div class="card-body">
  <dl class="row">
    <dt class="col-sm-4">Siret</dt>
    <dd class="col-sm-8">{{ company.siret }}</dd>
    <dt class="col-sm-4">Raison Sociale</dt>
    <dd class="col-sm-8">{{ company.name }}</dd>
    <dt class="col-sm-4">Adresse</dt>
    <dd class="col-sm-8">{{ company.address }}</dd>
  </dl>
</div>
</div>
<div class="col">
  <div class="card h-100">
    <div class="card-body">
      <dl class="row">
        <dt class="col-sm-4">Contact</dt>
        <dd class="col-sm-8">{{ company.user.email }}</dd>
        <dt class="col-sm-4">Téléphone</dt>
        <dd class="col-sm-8">{{ company.phone }}</dd>
        <dt class="col-sm-4">Accès</dt>
        <dd class="col-sm-8">{{ company.user.email }} / {{ company.user.plainPassword }}</dd>
      </dl>
    </div>
  </div>
</div>

```

Retirer dans le twig cette ligne

- Lorsque le paiement est en statut signé, l'admin a également accès au coordonné bancaire de la carte du client

Détails moyen de paiement



| | |
|--------------------------|-------------|
| Détenteur | eden |
| Type | MASTER CARD |
| Numéros | 96435466960 |
| Date d'expiration | 24/09/2026 |
| Code | salut |

Conséquence :

Il y a un manque de confidentialité ; et une personne malintentionnée pourrait récupérer les informations de la carte et l'utiliser pour des achats personnelles

Solution :

```
<dt class="col-sm-4 d-flex text-align-start">Détenteur</dt>
<dd class="col-sm-8 d-flex text-align-start">{{ contribution.payment.cardOwner }}</dd>
<dt class="col-sm-4 d-flex text-align-start">Type</dt>
<dd class="col-sm-8 d-flex text-align-start">{{ contribution.payment.cardType }}</dd>
<dt class="col-sm-4 d-flex text-align-start">Numéros</dt>
<dd class="col-sm-8 d-flex text-align-start">{{ contribution.payment.cardNumbers }}</dd>
<dt class="col-sm-4 d-flex text-align-start">Date d'expiration</dt>
<dd class="col-sm-8 d-flex text-align-start">{{ contribution.payment.cardExpirationDate }}</dd>
<dt class="col-sm-4 d-flex text-align-start">Code</dt>
<dd class="col-sm-8 d-flex text-align-start">{{ contribution.payment.cardCode }}</dd>
```

Dans le twig, ne laisser que le détenteur et le type