Check-list NIS2 technique pour Symfony

40 points de verification avant un audit officiel

Cette check-list traduit les exigences de la directive NIS2 en verifications concretes sur une application Symfony. Cochez chaque point, axe par axe, pour situer votre niveau de preparation avant de solliciter un auditeur PASSI. Elle constitue une base de travail, pas une attestation de conformite.

1. Durcissement et configuration

- [ ] Mode debug desactive en production (APP_ENV=prod, APP_DEBUG=0).
- [ ] Secrets geres via Symfony Secrets ou un coffre dedie, jamais en clair.
- [ ] APP_SECRET unique par environnement et rotation documentee.
- [ ] Headers de securite poses : CSP, HSTS, X-Content-Type-Options, X-Frame-Options.
- [ ] Cookies de session en Secure, HttpOnly et SameSite.
- [ ] Messages d'erreur generiques cote production, pas de stack trace exposee.
- [ ] Principe de moindre privilege applique aux droits base de donnees et conteneurs.

2. Authentification et controle d'acces

- [ ] Authentification multifacteur active sur les comptes a privileges.
- [ ] MFA resistante au phishing envisagee (passkeys WebAuthn, cles physiques).
- [ ] Politique de mots de passe alignee sur l'etat de l'art.
- [ ] Verrouillage et temporisation apres echecs d'authentification repetes.
- [ ] Roles et permissions revus regulierement, comptes inactifs desactives.
- [ ] Acces d'administration separes des acces utilisateurs standards.
- [ ] Sessions invalidees au changement de mot de passe et a la deconnexion.

3. Journalisation et detection

- [ ] Evenements de securite journalises (connexions, echecs, elevations de privileges).
- [ ] Journaux structures et horodates via Monolog.
- [ ] Donnees personnelles filtrees a la source et duree de conservation bornee.
- [ ] Centralisation des journaux vers une stack d'observabilite.
- [ ] Alertes sur les signaux faibles et les comportements anormaux.
- [ ] Tracabilite suffisante pour reconstituer la chronologie d'un incident.

4. Dependances et vulnerabilites

- [ ] composer audit execute regulierement et en integration continue.
- [ ] Politique de mise a jour priorisee selon la criticite reelle.
- [ ] Dependances abandonnees identifiees et remplacees.
- [ ] Verrouillage des versions via composer.lock et builds reproductibles.
- [ ] Veille active sur les CVE des composants critiques.

- [ ] Analyse statique au niveau maximum supporte (PHPStan).

5. Reponse a incident et continuite

- [ ] Procedure d'escalade definie avant tout incident.
- [ ] Capacite a emettre une alerte precoce sous 24 heures.
- [ ] Notification complete possible sous 72 heures.
- [ ] Sauvegardes testees et restauration eprouvee.
- [ ] Plan de continuite documente et rejouable sous pression.
- [ ] Exercices de crise menes periodiquement.

6. Gouvernance et preuves

- [ ] Configurations versionnees dans le depot.
- [ ] Tests automatises couvrant les correctifs de securite.
- [ ] Pipelines d'integration continue tracant les builds.
- [ ] Dossier de preuves techniques constitue et tenu a jour.
- [ ] Organe de direction informe et implique dans la gestion des risques.
- [ ] Sensibilisation des equipes techniques et metier a la securite.
- [ ] Cartographie des services, flux de donnees et sous-traitants a jour.
- [ ] Recoupement avec DORA et RGAA pour mutualiser les chantiers.
