Financé par l'Union européenne
NextGenerationEU

AGENCE DU NUMÉRIQUE EN SANTÉ
La transformation commence ici

Guide d’utilisation

Formulaire du test d’intrusion

Statut : Terminé | *Classification : Restreinte* | *Version : v1.1*

Agence du Numérique en Santé
2 - 10 Rue d'Oradour-sur-Glane, 75015 Paris
T. 01 58 45 32 50
esante.gouv.fr

HISTORIQUE ET IDENTIFICATION DU DOCUMENT

Identification du document	
Référence ANS	*ANS_Guide d'utilisation_TI_PSC_EDC_Editeur Logiciel Proxy.docx*
Classification	Restreinte
Nombre de pages	23

Historique de mises à jour du guide d'utilisation			
Version	***Date***	***Auteur***	***Commentaires / modifications***
V1.1	Juillet 2024	ANS	Rédaction de la première version du guide d'utilisation

SOMMAIRE

1. INTRODUCTION

1.1 Le contexte de l'Espace de Confiance Pro Santé Connect

Pro Santé Connect (PSC) est le fédérateur d'identités des professionnels des secteurs sanitaire, médico-social et social enregistrés au Répertoire Partagé des Professionnels de Santé (RPPS). Ce service socle est proposé par l'Agence du Numérique en Santé (ANS).

Il leur offre une manière **simple, sécurisée et unifiée** de se connecter à tous leurs **services numériques en santé**, en pouvant passer de l'un à l'autre de manière particulièrement fluide.

L'Espace de Confiance (EDC) de Pro Santé Connect est un environnement dédié aux services numériques en santé entrant dans le cadre du CI-SIS Volet Transport.

L'EDC vise à instaurer une relation de confiance entre ses différents composants afin, notamment, de garantir les niveaux de sécurité attendus. Pour assurer cette confiance, il est demandé au fournisseur de logiciel de santé de mettre en place une solution de **Proxy e-santé**, désignant un équipement jouant le rôle d'intermédiaire sécurisé entre la solution mise en place par l'éditeur et le reste de l'Espace de Confiance Pro Santé Connect dont notamment les services numériques en Santé.

1.2 Architecture de l'Espace de Confiance

L'Espace de Confiance Pro Santé Connect s'articule autour du Proxy e-santé, que les schémas suivants permettent de resituer vis-à-vis des différents acteurs de l'Espace de Confiance :

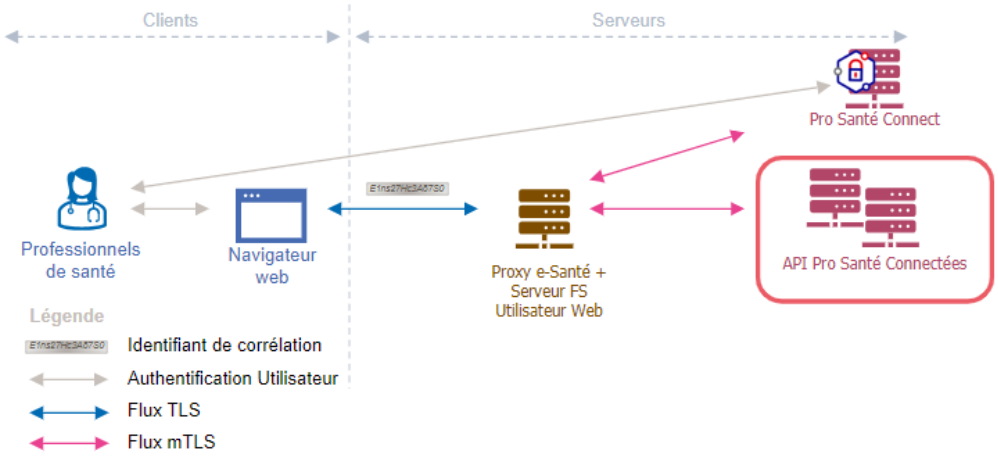


Figure 1 - Principaux acteurs de l'espace de confiance (cas d'une application web)

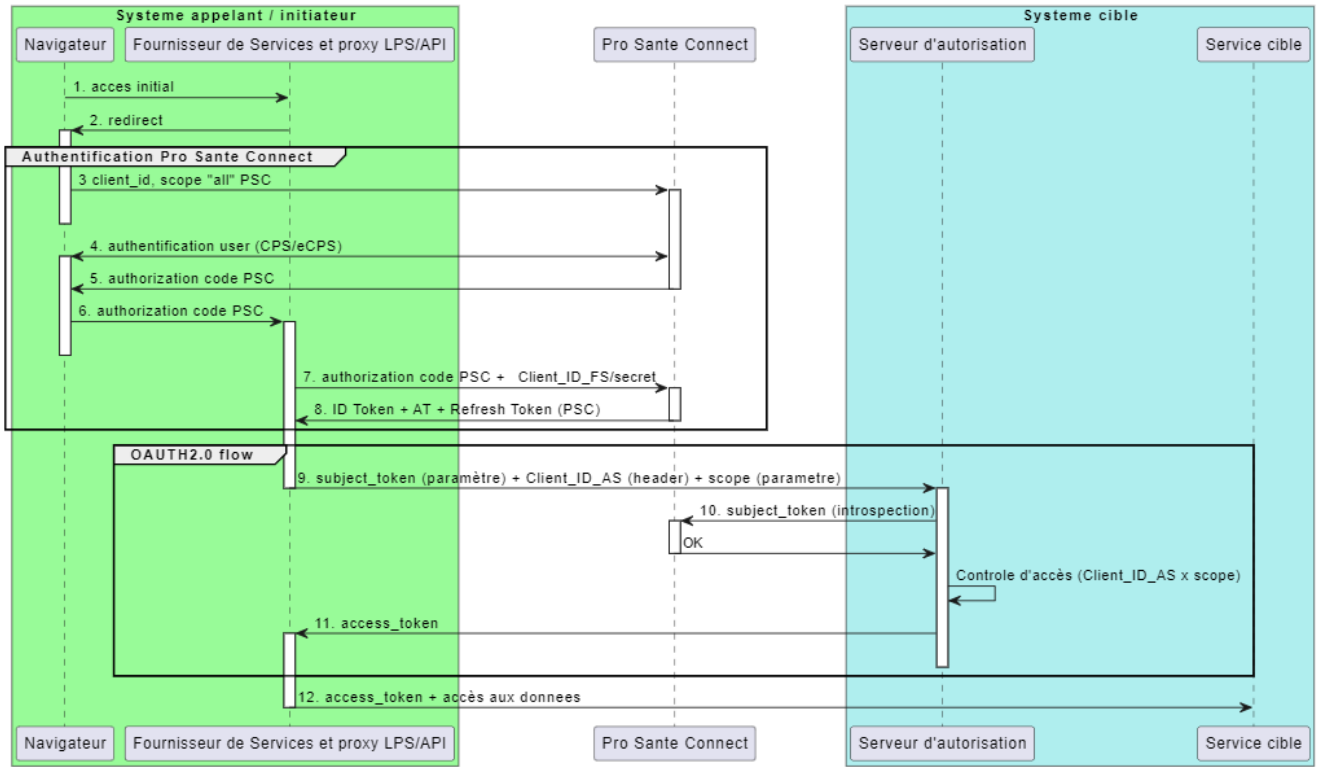


Figure 2 - Diagramme de séquence montrant les principales étapes d'un appel à une API Pro Santé Connectée

1.3 Objet du document

Les éditeurs souhaitant candidater pour le référencement d'une solution qu'ils produisent et qu'ils souhaitent voir entrer dans l'Espace de Confiance doivent donc passer par **un processus d'évaluation de la conformité de cette solution à des exigences portant sur la sécurité des systèmes d'information.** La **réalisation d'un test d'intrusion** spécifique au Proxy, et différant de celui portant sur le logiciel candidat est notamment exigé. Celui-ci donne lieu au remplissage d'un formulaire par l'auditeur qui permet de fixer le périmètre du test et atteste des résultats obtenus. Il constitue une preuve requise pour l'habilitation à l'EDC.

Le présent document vise à décrire le mode opérationnel à suivre pour réaliser ce test d'intrusion et transmettre le formulaire complété dans ce cadre.

2. CONDITIONS & PERIMETRE DE REALISATION DU TEST D'INTRUSION

2.1 Prérequis

Le test d'intrusion doit être réalisé par un prestataire d'audit, à la demande de l'éditeur. Afin de garantir les compétences du prestataire d'audit sélectionné et ainsi l'équité du processus, **il est demandé de faire appel à un prestataire d'audit de la sécurité des systèmes d'information qualifié** (PASSI : https://cyber.gouv.fr/produits-services-qualifies?sort_bef_combine=nom_du_fournisseur_ASC&field_type_service_value%5Bpassi%5D=passi&categorie_psq=).

Cette prestation est à la charge de l'éditeur candidat au référencement de sa solution.

La réalisation d'un audit PASSI (conditions de réalisation spécifiques et auditeur certifié PASSI) n'est pas requise. L'unique prérequis est de faire réaliser le test d'intrusion par un organisme qualifié PASSI.

2.2 Déroulement du test d'intrusion

Le test d'intrusion se déroule généralement en trois phases représentées sur le schéma ci-dessous :

2.2.1 Phase de cadrage

L'audit technique doit être précédé d'une phase de cadrage organisée par l'éditeur pour planifier avec le prestataire sélectionné l'audit à réaliser. Ce cadrage doit permettre d'établir les modalités de réalisation de l'audit et de valider notamment :

- Les dates de réalisation de l'audit, l'environnement mis à disposition, les personnes à contacter ;
- Les éléments décrivant le fonctionnement nominal de la solution et toute information utile à l'auditeur ;
- Les prérequis nécessaires pour l'auditeur à la réalisation de l'audit (cf. Chapitre 3. Rôle de l'éditeur).

Recommandation
Lors de la phase de cadrage, il est recommandé de prévoir **une clause de revoyure** avec l'auditeur pour les exigences non validées lors du test d'intrusion. En effet, il sera demandé à l'auditeur **d'intervenir une seconde fois (réalisation d'un contre audit) afin de vérifier si des mesures de sécurité ont été implémentées** sur les exigences précédemment non validées.

2.2.2 Phase de réalisation des tests

Durant cette phase, l'auditeur adopte le rôle d'un attaquant potentiel cherchant à identifier les vulnérabilités de l'application. En vue de vérifier des points de contrôle spécifiques, il disposera d'un accès à des éléments de l'application afin d'approfondir son analyse. La durée de la phase de test fluctue en accord avec la complexité de la solution, cependant, en moyenne, elle s'étend sur environ une semaine.

La majorité des points à contrôler sont basés sur le référentiel de l'**Open Web Application Security Project - OWASP** (cf. Annexe 1 : Définition et concepts généraux & cf. annexe 5).

Ils sont listés dans le formulaire à compléter par l'auditeur, et **sont au nombre de 18**, indépendamment du type de logiciel que le Proxy e-santé protège.

Les points de contrôles au niveau du formulaire du test d'intrusion sont tous de **gravité haute : La non-conformité au point de contrôle attendu est éliminatoire**.

Si un ou plusieurs manquements aux règles de sécurité sont présents, ces derniers doivent être corrigées **avant la fin du processus de référencement** afin de garantir la conformité aux seuils de validation définis ci-dessus.

En cas de vulnérabilité majeure découverte sur la solution ayant un score CVSS v3 (Common Vulnerability Scoring System) supérieur ou égal à 8 (cf. Annexe 1 : Définition et concepts généraux), **l'auditeur doit en informer l'éditeur qui transmettra l'information au CERT santé**.

Par ailleurs, si l'auditeur juge que **l'exploitation d'une vulnérabilité est complexe** en raison du contexte applicatif de la solution ou des mesures de sécurité mises en place pour la protéger, il peut confirmer les règles de sécurité en incluant ces détails dans un commentaire explicatif.

2.2.3 Phase de rapports

Une fois les tests finalisés, l'auditeur constituera le rapport du test d'intrusion en **remplissant le formulaire**. Ce document regroupera l'ensemble des **résultats** du test d'intrusion ainsi que le **référencement** de l'application.

De plus, cette phase aura pour finalité d'échanger sur le rapport du formulaire du test d'intrusion. En effet, si un ou plusieurs manquements aux règles de sécurité sont relevés, l'objectif de l'auditeur sera de transmettre **les résultats du test d'intrusion** à l'éditeur en lui expliquant :

- Les détails techniques des vulnérabilités identifiées ;
- L'impact potentiel des manquements aux règles de sécurité et un niveau de risque associé ;
- Les solutions concrètes permettant de corriger les potentielles failles ;
- La définition des prochaines étapes (définition d'une date permettant d'évaluer de nouveau les vulnérabilités recensées lors d'un audit de contrôle).

2.2.4 Phase d'audit(s) de contrôle éventuel

Dans le cas où l'éditeur doit apporter **des corrections aux vulnérabilités de sa solution pour se conformer** au seuil de validation des points de contrôle de gravité haute du test d'intrusion, **il doit informer l'auditeur des vulnérabilités corrigées**.

Ensuite, l'auditeur **teste de nouveau ces points pour vérifier les corrections** et **met à jour le rapport du test d'intrusion** avec les résultats obtenus. Une version actualisée du formulaire, incluant tous les résultats du test ainsi que le référencement de l'application, est ensuite fournie.

3. ROLE DE L'EDITEUR

L'éditeur intervient essentiellement en amont de l'audit afin d'assurer sa préparation. Cette phase est essentielle afin que l'auditeur dispose de l'ensemble des éléments nécessaires et soit ainsi en mesure de réaliser l'ensemble de la prestation dans les conditions attendues et dans le respect des délais prévus.

3.1 Mise à disposition de l'environnement

Concernant l'environnement mis à disposition pour la réalisation de l'audit, il est notamment indispensable de prévoir les éléments suivants :

- Le périmètre du test d'intrusion doit être parfaitement cadré pour limiter les impacts sur l'application ou le système d'information. Ainsi, la prestation doit être réalisée de préférence sur un **environnement « iso-prod »** (tel qu'un environnement de développement, de test, etc.) plutôt que sur un environnement de production ;
- L'éditeur doit mettre à disposition de l'auditeur **un logiciel de santé de test compatible avec le proxy e-santé étudié,** et représentatif des différents cas d'usage du proxy.
- L'éditeur doit mettre à disposition de l'auditeur une simulation de service numérique en santé, ou autre service exposant une **API Pro Santé Connectée compatible avec le proxy e-santé étudié**. Ce service doit être hébergé sous le contrôle de l'éditeur, et il est dans tous les cas **interdit** de réaliser un test d'intrusion sur un environnement communiquant avec un « bac à sable » officiel d'un fournisseur d'API Pro Santé Connectée tiers **sans son accord explicite préalable**. Bien qu'il soit recommandé autant que possible d'utiliser un environnement pleinement fonctionnel, il est possible d'utiliser la simulation fournie par l'ANS pour assurer a minima les fonctionnalités de serveur d'autorisation OIDC.
- Le **périmètre du test d'intrusion doit se limiter au service proxy e-santé**, et ne doit en aucun cas inclure le logiciel de santé de test qui doit être fourni à l'auditeur, ou une éventuelle simulation d'API Pro Santé connectée. Le logiciel de santé de test doit uniquement permettre d'identifier les principales fonctionnalités du proxy e-santé et faciliter le test d'intrusion.
- L'application testée doit correspondre à la **même version majeure que celle en production** (données similaires, niveau de sécurité suffisant, paramétrage identique, etc.) afin d'obtenir une vision réaliste de la sécurité ;
- Il est essentiel de donner les moyens à l'auditeur de tester le produit tel qu'il est commercialisé que ce soit en tant qu'application web installée localement sur l'appareil d'un utilisateur ou déployée à distance via une plateforme de virtualisation (ex : Citrix, etc.). Une désactivation de l'ensemble des dispositifs de sécurité **(WAF, sondes, passerelles, etc.)** est nécessaire, **s'ils ne font pas partie de la solution commercialisée**.
- Le formulaire du test d'intrusion se concentre sur **l'évaluation des vulnérabilités et des failles de sécurité au niveau logiciel**. Ainsi, **les parties matérielles des solutions** (borne physique, poste d'un usager, etc.) sont exclues du test d'intrusion.

3.2 Préparation du formulaire pour l'auditeur

Pour assurer la complétude du formulaire, l'éditeur **doit renseigner l'ensemble des informations propres à son application** dans le formulaire du test d'intrusion (fichier Excel, onglet « 1 – Résultat Formulaire »).

3.3 Eléments à communiquer à l'auditeur

Pour assurer la réalisation de la prestation, **l'éditeur doit communiquer à l'auditeur** :

- Les **informations nécessaires pour pouvoir communiquer avec le service proxy**, incluant par exemple une URL ou une IP, les mécanismes de filtrage et mise en liste blanche à prendre en compte etc… ;
- **Un accès à un logiciel de santé compatible avec le service proxy audité**, sous toute forme pertinente (exécutable, identifiants permettant l'accès à un compte crée pour l'occasion…) et avec un niveau de permissions permettant de réaliser *a minima* des opérations basiques représentatives des différents cas d'usage du Proxy. Il est conseillé à l'éditeur de fournir le même logiciel que celui utilisé lors des phases de développement et recette.
- **Une liste des services et comptes génériques essentiels** présents sur le serveur hébergeant le service proxy, afin de vérifier l'exposition de ces éléments ;
- **La procédure de traitement des alertes** mise en place par l'éditeur en cas de détection de requête considérée non-conforme ainsi que **la liste des cas considérés comme « mésusage ou usage anormal »** menant à une interruption de requêtes ;
- **Une extraction des journaux techniques** sur les tests réalisés lors du **premier jour opérationnel** afin de suivre les tentatives d'authentification, la présence de données sensibles (toute donnée à caractère personnel, qu'elle soit ou non de santé, ou participant à la sécurité du système d'information constitué par le système seul ou le système d'information auquel il participe) et le format des événements. Ces journaux doivent être transmis le plus rapidement possible à l'auditeur ;
- L'éditeur doit informer l'auditeur de toute **spécificité de l'environnement de test** qui n'est pas ou ne sera pas applicable à son environnement de production. Cela inclut par exemple : les différences d'architecture et d'exposition, les différences de paramétrage en particulier sur les durées de session.
- **L'éditeur doit informer l'auditeur de toutes les vulnérabilités connues,** ainsi que des mesures de sécurité mises en œuvre pour les contourner.

Une fois le formulaire rempli, celui-ci sera **généré sous forme de PDF puis signé électroniquement** par l'auditeur et sera retourné à l'éditeur qui le déposera sur l'outil Convergence. Une fiche de synthèse des éléments ci-dessus est mise à disposition en Annexe 2 : Synthèse de rappel pour les éditeurs.

Outre ces prérequis, dans l'optique de permettre un audit approfondi et efficace, l'éditeur pourra mettre à disposition de l'auditeur tous les détails et informations qu'il juge pertinents concernant la solution à référencer (ex : Dossier d'Architecture Technique, documentation du projet, schémas techniques, etc.).

4. ROLE DE L'AUDITEUR

L'auditeur intervient sur la phase de la réalisation de l'audit. Cette phase est essentielle afin que l'auditeur réalise l'ensemble des tests et puisse permettre de vérifier si la solution de l'éditeur est sécurisée.

4.1 Prérequis

Lors de la phase de cadrage, l'auditeur doit s'assurer que l'ensemble des éléments lui permettant d'assurer la réalisation de la prestation lui ont été transmis (ou le seront au début de la phase de test dans le cas des journaux).

Les éléments cités en section Eléments à communiquer à l'auditeur doivent notamment lui être communiqués par l'éditeur.

En complément, l'auditeur pourra demander à l'éditeur lors de la phase de cadrage toute information jugée utile à la réalisation du test d'intrusion. Cette phase est essentielle afin de bien cadrer avec l'éditeur les éléments entrants à produire pour plus d'efficacité lors de la phase d'audit.

4.2 Remplissage du formulaire

Il est attendu de la part de l'auditeur de tester l'ensemble des points de contrôle listés dans le formulaire de test d'intrusion et d'évaluer la conformité de l'application à ces différents points. Il peut alors renseigner le formulaire de test d'intrusion. Plusieurs périmètres sont couverts dans le formulaire afin de donner une vision complète du niveau de sécurité de l'application. **Les consignes suivantes doivent impérativement être respectées** :

Gestion des onglets

Uniquement un onglet doit être obligatoirement rempli par les auditeurs: il s'agit de l'onglet "2 - Proxy".

L'utilisation de l'onglet "3 – Génération PDF et Signature" est expliquée au paragraphe suivant.

Notation des règles

Seule la notation "Oui, Non ou N/A" est autorisée afin de savoir si l'ensemble des mesures de sécurité sont mises en oeuvre. **Pour chacune des règles de sécurité, une notation doit être attribuée.**

Réponse N/A

La notation N/A est utilisée uniquement lorsque :

- la règle de sécurité **ne depend pas de l'éditeur mais de la structure utilisatrice** (ex : gestion des serveurs physique dans le cadre d'une solution SaaS)
- la règle de sécurité n'est **pas applicable au système,** ou n'est pas observable dans les conditions de réalisation de l'audit décrites dans ce document (point de contrôle P17).

Dans tous les cas, **une justification en commentaire est attendue**

Réponse Non

La notation "Non" doit être **limitée et nécessite une justification en commentaire**.

Elle ne peut pas être justifiée par un manquement lié à la documentation incluse dans les éléments à fournir à l'auditeur (voir section 3.3)

4.3 Génération PDF & Signature électronique

Une fois le formulaire rempli, **un fichier PDF reprenant les éléments qu'il contient doit être généré à partir de la macro du formulaire du test d'intrusion puis signé électroniquement, avec l'approbation d'un TSP** (Trust Service Provider). Pour apposer cette signature électronique, deux options sont possibles (détails des étapes disponibles dans l'annexe 4 : processus autour de la signature électronique) :

- Via une application bureautique telle qu'Adobe avec un certificat de signature délivré par un TSP ;
- Via une plateforme de signature électronique reconnue telle que DocuSign, Docaposte, YouSign, etc. ;

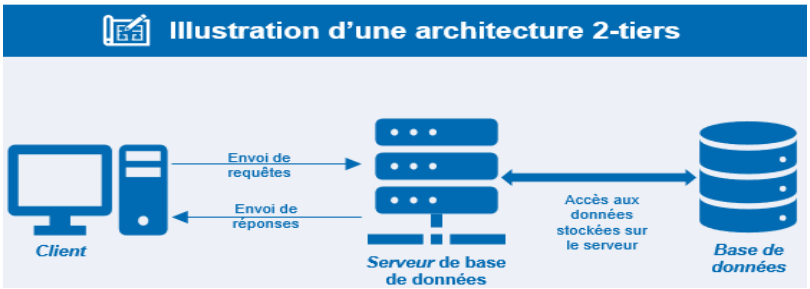
Ce fichier signé électroniquement, avec l'approbation du TSP via un certificat de signature ou via une plateforme de signature électronique, constitue une preuve essentielle que l'éditeur devra soumettre dans le cadre de sa candidature pour l'habilitation à l'EDC.

En plus du rapport, dans le cas où certains points de contrôles auraient été jugés non conformes, l'auditeur pourra proposer à l'éditeur des mesures correctives à mettre en œuvre. Une fiche de synthèse avec les principaux éléments qui concernent l'auditeur est mise à disposition en Annexe 3 : Synthèse de rappel pour les auditeurs.

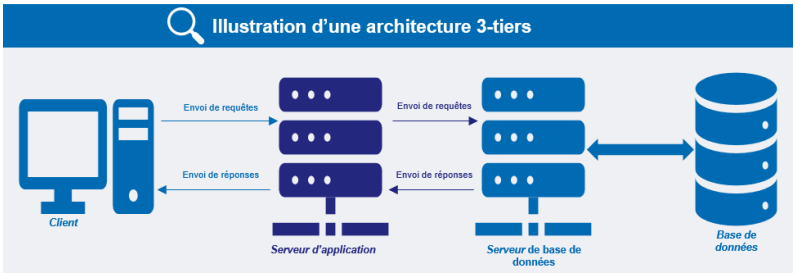
4.4 Annexe 1 : Définition et concepts généraux

Ce guide d'utilisation ainsi que le formulaire du test d'intrusion font référence à des concepts précis de la sécurité des systèmes d'information. Ainsi, certaines précisions et définitions sont proposées ci-dessous afin d'assurer une compréhension commune pour faciliter leur compréhension.

Architecture 2-tiers : c'est une architecture qui se compose de deux principales couches : la couche client et la couche serveur. Dans cette architecture, le client réalise directement les requêtes sur le serveur de base de données.



Architecture 3-tiers : c'est une architecture qui se compose de trois principales couches : la couche client, la couche applicative et la couche données. Dans cette architecture, le client réalise les requêtes sur la couche applicative qui requête elle-même le serveur de base de données.



Client – serveur : architecture désignant la séparation des tâches d'une application entre deux entités distinctes et cloisonnées. Le côté client désigne l'ensemble des composants logiciels manipulés par les utilisateurs. Le côté serveur désigne l'ensemble des composants logiciels responsables des traitements, c'est-à-dire l'implémentation de la logique métier, et de l'accès aux données.

Exemple générique d'application client-serveur : Système où les clients (utilisateurs) se connectent à un serveur pour demander des services ou des ressources. Le serveur fournit ces services ou ressources en réponse aux requêtes des clients. Cette architecture est utilisée dans une variété d'applications, telles que la messagerie instantanée, le partage de fichiers, les services de bases de données, etc. Les clients envoient des requêtes au serveur, qui les traite et renvoie des réponses appropriées. Les clients interagissent avec le serveur pour obtenir des informations ou effectuer des actions, ce qui permet une distribution efficace des services et des ressources.

Compte de service : désigne un compte utilisé par des process ou un quelconque actif informatique pour accéder à ou exploiter une ressource (par exemple une application, un système de fichiers partagé…) sans intervention humaine.

Compte générique : désigne un compte individuel dont l'usage est partagé par plusieurs utilisateurs (humains ou machines) pour un même besoin métier spécifique. L'utilisation de ces derniers doit être évitée autant que possible, et contrôlée avec une attention particulière par un renforcement du contrôle d'accès et de la traçabilité là où il n'est pas possible d'éviter leur utilisation.

Donnée sensible (est considérée comme telle) :

- Toute donnée à caractère personnel, qu'elle soit ou non de santé ;
- Toute information participant à la sécurité du système d'information constitué par le système seul ou le système d'information auquel il participe.

Journaux métiers : Journaux (logs) contenant des informations liées aux activités, aux opérations spécifiques aux métiers, et peuvent englober des données relatives aux transactions, aux utilisateurs, aux applications, etc. qui sont essentielles dans le contexte des activités d'une entreprise.

Journaux techniques : Journaux (logs) contenant des informations relatives au fonctionnement des SI (ex : logs des serveurs, des applications, des composants matériels, etc.) Ils comprennent généralement des détails sur les erreurs, les défaillances, les performances, les données de débogage, l'activité du réseau, etc.

Opération sensible, dans l'utilisation du système (est considérée comme telle) :

- Tout déclenchement d'action susceptible d'induire directement ou indirectement l'enregistrement, l'affichage, la transmission, la modification ou l'effacement d'information sensible, sauf à ce que l'analyse de risque du système ait déterminé que l'action déclenchée sur les informations considérées n'était pas porteuse d'enjeu particulier ;
- Tout déclenchement d'action susceptible d'induire des effets négatifs sur la santé de patients ;
- Tout autre déclenchement d'action identifiée par l'analyse de risque du système comme devant faire l'objet de mesures de protection et/ou de contrôle renforcées.

L'Open Web Application Security Project (OWASP) est une fondation à but non lucratif qui œuvre à l'amélioration de la sécurité des logiciels. La fondation OWASP est une source d'information mondialement reconnue par les spécialistes et suivie par les développeurs pour sécuriser les applications de type web.

La réalisation du test d'intrusion dont il est question dans le présent document et les thématiques abordées dans le formulaire associé se basent essentiellement sur le Top 10 de l'OWASP. Il s'agit de classements de référence des principales attaques rencontrées. Il existe un classement concernant les applications web dont les risques référencés en 2021 incluent :

A01:2021 – Broken Access Control	A06:2021 – Vulnerable and Outdated Components
A02:2021 – Cryptographic Failures	A07:2021 – Identification and Authentication Failures
A03:2021 – Injection	A08:2021 – Software and Data Integrity Failures
A04:2021 – Insecure Design	A09:2021 – Security Logging and Monitoring Failures
A05:2021 – Security Misconfiguration	A10:2021 – Server-Side Request Forgery (SSRF)

Il existe également un classement concernant les API dont les risques référencés en 2023 incluent :

API1:2023 Broken Object Level Authorization	API6:2023 Unrestricted Access to Sensitive Business Flows
API2:2023 Broken Authentication	API7:2023 Server Side Request Forgery

API3:2023 Broken Object Property Level Authorization	API8:2023 Security Misconfiguration
API4:2023 Unrestricted Resource Consumption	API9:2023 Improper Inventory Management
API5:2023 Broken Function Level Authorization	API10:2023 Unsafe Consumption of APIs

Score CVSS v3 (Common Vulnerability Scoring System): système standardisé de notation des vulnérabilités établi par le FIRST (Forum of Incident Response and Security Teams). Il attribue à une vulnérabilité un score, qui est entre 0 et 10, en fonction de plusieurs critères, tels que la complexité de l'exploitation, l'impact sur la confidentialité, l'intégrité et la disponibilité des données, et d'autres facteurs. Plus le score est élevé, plus la vulnérabilité est critique (niveau faible, moyen, élevé ou critique).

Secret : donnée sensible et protéiforme caractérisée par son usage dans les processus d'authentification des utilisateurs et de contrôle d'accès à des ressources et/ou fonctionnalités de composants système ou applicatif. Elle peut désigner des identifiants de comptes, identifiants de session, des mots de passe, clés de chiffrement (privées), clés d'API, jetons d'authentification, etc.

TSP (Trust Service Provider) : Un TSP est un fournisseur de services de confiance qui fournit des certificats numériques utilisés pour authentifier et sécuriser les signatures électroniques. Ces certificats sont associés à une identité spécifique (par exemple, une personne ou une organisation) et contiennent des informations telles que la clé publique de cette identité, des données d'authentification, et sont signés numériquement par l'autorité de certification. Le TSP joue un rôle essentiel en vérifiant l'identité du titulaire du certificat avant de lui délivrer un certificat de signature. Cela assure la confiance dans les signatures électroniques, car elles sont associées à des identités vérifiées.

4.5 Annexe 2 : Synthèse de rappel pour les éditeurs

- Le test d'intrusion doit être effectué par un prestataire d'audit de la sécurité des systèmes d'information qualifié (PASSI) ;
- Lors de la phase de cadrage, il est recommandé de prévoir une « clause de revoyure » avec l'auditeur afin de valider les corrections effectuées dans l'hypothèse où certaines exigences ne seraient pas validées lors du test d'intrusion ;
- La prestation est à réaliser de préférence dans un environnement iso-prod (tel qu'un environnement de développement, de test, etc.) ;
- L'application testée doit correspondre à la même version majeure que celle en production ;
- L'auditeur ne doit pas être limité par les dispositifs de sécurité (WAF, sondes, etc.), ceux-ci doivent impérativement être désactivés s'ils ne font pas partie de la solution commercialisée ;
- L'éditeur doit compléter dans le formulaire du test d'intrusion l'ensemble des informations propres à son application dans l'onglet « 1 – Résultat Formulaire » ;
- Le formulaire atteste de la réalisation du test d'intrusion. Il doit être complété puis signé électroniquement par l'auditeur dans un format PDF qu'il aura généré en suivant les indications du formulaire. Ce document constitue une preuve à fournir dans le cadre de la candidature pour l'habilitation à l'EDC;
- L'éditeur doit communiquer à l'auditeur l'ensemble des informations recensées dans la liste suivante :
    - Les informations nécessaires pour pouvoir communiquer avec le service proxy audité ;
    - Un accès à un logiciel de santé compatible avec le service proxy ;
    - La liste des services et comptes génériques essentiels ;

- La procédure de traitement des alertes et la liste des cas considérés comme « mésusage ou usage anormal »
- Une extraction des journaux techniques sur les tests réalisés lors du premier jour opérationnel ;
- Toutes les différences avec l'environnement de production ;
- Toutes les vulnérabilités connues et les mesures de sécurité mises en œuvre pour les contourner.

L'ensemble de ces informations (hors journaux du premier jour) doivent être transmises dès la phase de cadrage afin de faciliter le travail de l'auditeur et limiter les échanges lors de la phase de tests qui pourraient occasionner un dépassement des délais prévus pour la prestation.

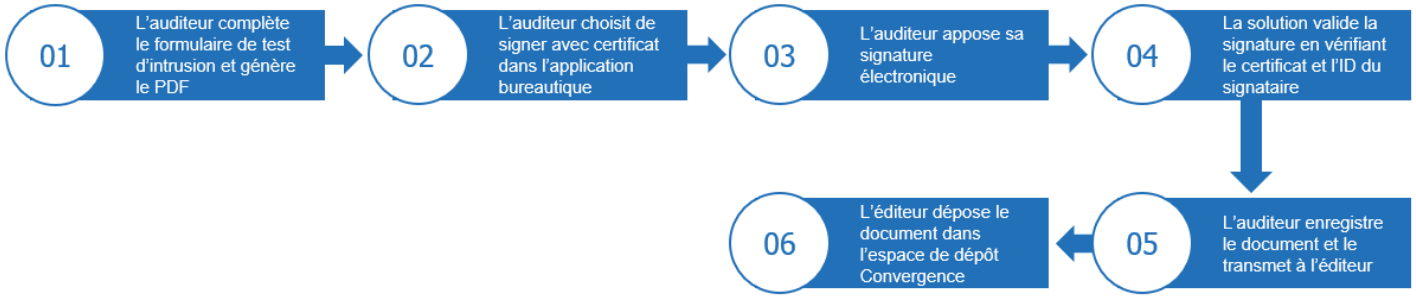
4.6 Annexe 3 : Synthèse de rappel pour les auditeurs

- Le test d'intrusion doit être effectué par un prestataire d'audit de la sécurité des systèmes d'information qualifié (PASSI). Cependant il ne s'agit pas d'un audit PASSI (la certification PASSI n'est pas requise pour l'auditeur et les conditions de réalisation du test d'intrusion ne sont celles d'un audit PASSI) ;
- Lors de la phase de cadrage, il est recommandé de prévoir une « clause de revoyure » avec l'auditeur afin de valider les corrections effectuées dans l'hypothèse où certaines exigences ne seraient pas validées lors du test d'intrusion ;
- Le formulaire atteste de la réalisation du test d'intrusion. Il doit être complété puis signé électroniquement par l'auditeur dans un format PDF qui peut être généré en suivant la macro du formulaire. La signature doit être approuvée par TSP (Trust Service Provider) via un certificat de signature ou via une plateforme de signature électronique. Ce document à transmettre à l'éditeur constitue la preuve que celui-ci devra fournir dans le cadre de la candidature à l'habilitation EDC;
- Uniquement deux onglets du formulaire doivent être obligatoirement remplis par les auditeurs : l'onglet « Base commune » ainsi que l'onglet correspondant au type de l'application auditée ;
- Un formulaire doit être réalisé par type d'application ou type d'environnement ;
- Pour chaque point de contrôle, la conformité de l'application doit être évaluée et consignée dans le formulaire ;
- Les notes « N/A » et « Non » à l'évaluation de conformité de l'application nécessite impérativement une justification dans la colonne commentaire ;
- Il est essentiel de s'assurer que les informations suivantes ont bien été transmises par l'éditeur en amont de la phase de tests :
    - Les informations nécessaires pour pouvoir communiquer avec le service proxy audité ;
    - Un accès à un logiciel de santé compatible avec le service proxy ;
    - La liste des services et comptes génériques essentiels ;
    - La procédure de traitement des alertes et la liste des cas considérés comme « mésusage ou usage anormal »
    - Une extraction des journaux techniques sur les tests réalisés lors du premier jour opérationnel ;
    - Toutes les différences avec l'environnement de production ;
    - Toutes les vulnérabilités connues et les mesures de sécurité mises en œuvre pour les contourner.

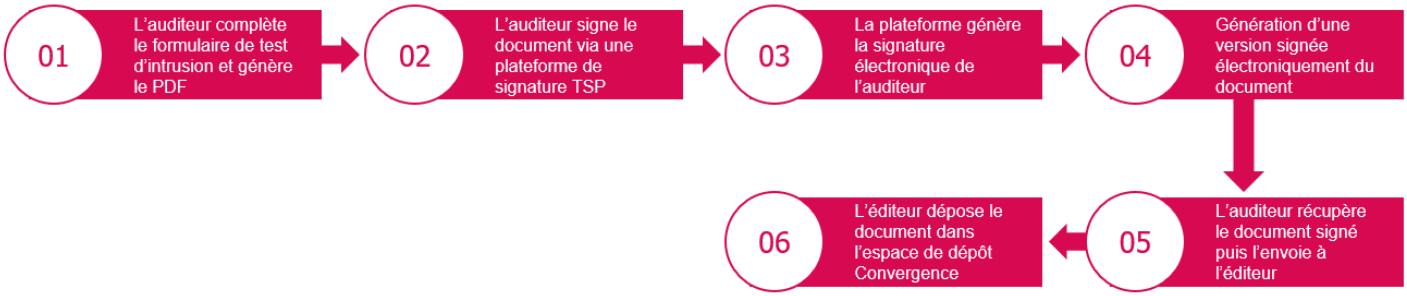
L'ensemble de ces informations doivent être transmises dès la phase de cadrage afin de faciliter le travail de l'auditeur et limiter les échanges lors de la phase de tests qui pourraient occasionner un dépassement des délais prévus pour la prestation.

4.7 Annexe 4 : Processus autour de la signature électronique

Signature électronique avec certificat délivré par un TSP



Signature électronique via une plateforme de signature



4.8 Annexe 5 : Vulnérabilités de l'OWASP et mapping des règles de sécurité du formulaire

Les points à contrôler sont basés à la fois sur le référentiel du Top 10 de l'**Open Web Application Security Project - OWASP** (cf. Annexe 1 : Définition et concepts généraux) et sur les exigences du référentiel Communauté PSC Extension Espace de Confiance. Dans le tableau ci-dessous se trouvent les contrôles se basant uniquement sur le Top 10 accompagnés d'une description des vulnérabilités de l'OWASP :

ID Contrôle	Risques de sécurité des Top Ten associés	Descriptions des vulnérabilités
P1	API2:2023 Broken Authentication	Vulnérabilité concernant les mécanismes d'authentification Les attaquants peuvent prendre le contrôle complet des comptes d'autres utilisateurs du système, lire leurs données personnelles et effectuer des actions sensibles en leur nom. **Exemple de vulnérabilités :** *CWE-204: Observable Response Discrepancy* *CWE-307: Improper Restriction of Excessive Authentication Attempts*
P2	A07:2021 Identification and Authentication Failures	Erreurs d'authentification en raison de problématiques de conception ou d'implémentation lors de la confirmation de l'identité, de l'authentification et de la session de l'utilisateur Les failles les plus courantes sont la présence de mot de passe faibles ou stockés en clair / faiblement chiffrés, des sessions d'utilisateurs qui ne sont pas invalidées correctement ou des identifiants de session exposés, etc. **Exemple de vulnérabilités :** *CWE-287 Improper Authentication, ex. CVE-2023-32243* *CWE-521 Weak Password Requirements, ex. CVE-2023-22451* *CWE-613 Insufficient Session Expiration, ex. CVE-2023-28003*
P4	API2:2023 Broken Authentication	Vulnérabilité concernant les mécanismes d'authentification Les attaquants peuvent prendre le contrôle complet des comptes d'autres utilisateurs du système, lire leurs données personnelles et effectuer des actions sensibles en leur nom. **Exemple de vulnérabilités :** *CWE-204: Observable Response Discrepancy* *CWE-307: Improper Restriction of Excessive Authentication Attempts*
P5	API2:2023 Broken Authentication	Vulnérabilité concernant les mécanismes d'authentification Les attaquants peuvent prendre le contrôle complet des comptes d'autres utilisateurs du système, lire leurs données personnelles et effectuer des actions sensibles en leur nom. **Exemple de vulnérabilités :** *CWE-204: Observable Response Discrepancy* *CWE-307: Improper Restriction of Excessive Authentication Attempts*

P6	API2:2023 Broken Authentication	Vulnérabilité concernant les mécanismes d'authentification Les attaquants peuvent prendre le contrôle complet des comptes d'autres utilisateurs du système, lire leurs données personnelles et effectuer des actions sensibles en leur nom. **Exemple de vulnérabilités :** *CWE-204: Observable Response Discrepancy* *CWE-307: Improper Restriction of Excessive Authentication Attempts*
P9	API9:2023 Improper Inventory Management	Une gestion inadéquate des inventaires peut mener à l'exposition involontaire de points d'accès non sécurisés ou obsolètes Les anciennes versions d'API ou les services non corrigés peuvent devenir des cibles faciles pour les attaquants qui peuvent exploiter ces failles pour accéder à des données sensibles ou prendre le contrôle de systèmes critiques. **Exemple de vulnérabilité :** *CWE-1059: Incomplete Documentation*
P14	A02:2021 Cryptographic Failures	Vulnérabilités liées au chiffrement (flux réseaux, base de données, informations sensibles, etc.) et à ses erreurs. Elles résultent d'algorithmes / de protocoles désuets ou faibles, d'une mauvaise gestion des clés de chiffrements, de l'absence de fonction de hash ou de mauvaise validation des certificats. **Exemple de vulnérabilités :** *CWE-326 Inadequate Encryption Strength, ex. CVE-2023-21443* *CWE-331 Insufficient Entropy, ex. CVE-2022-33756* *CWE-523 Unprotected Transport of Credentials, ex. CVE-2022-31805*
P16	A03:2021 Injection API10:2023 Unsafe Consumption of APIs	Vulnérabilités liées à l'injection de données non fiables dans l'application pouvant mener à l'exécution de commandes. Les attaquants introduisent des données malveillantes et peuvent en conséquence, voir, modifier et supprimer des données métiers ou nécessaires au bon fonctionnement d'une application et même obtenir le contrôle du serveur. **Exemple de vulnérabilités :** *CWE-20 Improper Input Validation CVE-2023-31047* *CWE-79 Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting'), ex. CVE-2023-31807* *CWE-89 Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection'), ex. CVE-2023-31038* Risques liés à la confiance excessive des développeurs envers les données provenant d'API externes L'exploitation réussie peut entraîner une exposition d'informations sensibles à des acteurs non autorisés, de nombreux types d'injections, ou un déni de service.

		Exemple de vulnérabilités : *CWE-20: Improper Input Validation* *CWE-200: Exposure of Sensitive Information to an Unauthorized Actor* *CWE-319: Cleartext Transmission of Sensitive Information*
P18	API1:2023 Broken Object Level Authorization	Vulnérabilités où les attaquants peuvent accéder à des objets non autorisés en manipulant les identifiants d'objet dans les requêtes API L'accès non autorisé aux objets d'autres utilisateurs peut entraîner la divulgation de données à des parties non autorisées, la perte de données ou la manipulation de données. **Exemple de vulnérabilités :** *CWE-285: Improper Authorization* *CWE-639: Authorization Bypass Through User-Controlled Key*