

Veille technologique : rapports du CERT-FR

Sommaire

Cahier des charges – Expression des besoins.....	3
Descriptif de l'existant	3
Besoin(s).....	3
Contrainte(s)	3
Ressources	3
Ressources mises à disposition.....	3
Ressources nécessaires à la mise en place	4
Gestion des ressources	4
Analyse	4
Descriptifs des solutions	4
Comparaison des solutions (tableau)	5
Choix d'une solution - Argumentation.....	6
Plan d'adressage - Schéma - Tables de routage.....	6
Étude de l'impact sur le SI existant.....	7
Phasage de l'intervention	8
Prévision des tests de validation.....	9
Mise en place.....	9
Bilan	9
Conclusion.....	9
Auto-évaluation	9

Cahier des charges – Expression des besoins

Descriptif de l'existant

Un serveur personnel Proxmox VE est déjà hébergé chez OVH et dispose de sa propre adresse IP publique. Un reverse-proxy sous Nginx est également déjà en place sur une VM dédiée, avec une adresse IP publique supplémentaire distincte de celle du Proxmox. Ce reverse-proxy assure le routage, le NAT et la passerelle par défaut des VM internes qui ne possèdent qu'une adresse IP privée sur le réseau du Proxmox. Un salon sur un Discord privé, un espace Notion et un accès à l'IA Claude (Anthropic) sont également déjà utilisés à titre personnel.

Besoin(s)

Dans le cadre du BTS SIO, le référentiel demande la réalisation d'une veille technologique sur un sujet choisi librement. J'ai choisi comme thème les rapports du CERT-FR (Centre gouvernemental de veille, d'alerte et de réponse aux attaques informatiques) afin de suivre les alertes, avis et bulletins d'actualité en cybersécurité. Voici comment est organisé ma veille : un serveur hébergeant le service N8N récupère automatiquement les flux RSS du CERT-FR et envoie les nouvelles publications (titre + liens) dans un salon Discord privé via un webhook. Je copie les liens reçus sur Discord et les envoie dans un Projet sur l'IA Claude configuré avec des instructions précises. Claude analyse chaque lien, génère un objet JSON structuré et l'envoie directement au webhook N8N via une requête HTTP. N8N récupère ces JSON et les intègre automatiquement dans une base de données Notion, accessible publiquement sous forme de calendrier.

Contrainte(s)

La solution N8N doit être accessible publiquement en HTTPS afin que l'interface soit administrable à distance. Le workflow N8N doit récupérer les flux RSS du CERT-FR à intervalle régulier et envoyer automatiquement les nouvelles publications (titre + liens) dans le salon Discord privé via webhook. Le projet sur Claude doit être configuré avec des instructions standardisées garantissant que chaque synthèse est générée dans un format JSON identique, afin que N8N puisse insérer systématiquement le bon contenu dans Notion. L'envoi du JSON au webhook N8N par Claude et l'insertion dans Notion doivent s'exécuter sans intervention supplémentaire une fois que Claude a reçu les liens.

Ressources

Ressources mises à disposition

J'ai à ma disposition un serveur Proxmox VE déjà hébergé chez OVH, un reverse-proxy Nginx opérationnel avec sa propre adresse IP publique, un nom de domaine « mathys-demon.fr » déjà enregistré, un salon Discord privé, un espace Notion, un compte Anthropic donnant accès à l'IA Claude, ainsi qu'un accès Internet.

Ressources nécessaires à la mise en place

Pour réaliser ce projet, j'ai besoin d'une VM supplémentaire sous Debian 13 dédiée à N8N et à sa base de données SQLite et d'un vhost en plus sur le reverse-proxy.

Pour la VM N8N : Node.js en version comprise entre v20.19 et v24.x (version imposée par N8N), npm, N8N en version 2.17.7 (ou une version ultérieure) installé via npm, SQLite et systemd pour exécuter N8N en tant que service en arrière-plan au démarrage de la VM.

Pour la VM reverse-proxy : ajout d'un vhost Nginx dédié à n8n.mathys-demon.fr et génération d'un certificat SSL Let's Encrypt via Certbot.

Gestion des ressources

Je dispose déjà de l'hyperviseur Proxmox (loué chez OVH à 20 € TTC/mois avant la mise en place de la veille pour d'autres services personnels ainsi que pour mon portfolio), de l'adresse IP publique du reverse-proxy (1,50 € TTC/mois, déjà payée avant la mise en place de la veille) et du nom de domaine « mathys-demon.fr » (1€ TTC/an puis 11€ TTC/an chez IONOS, déjà enregistré pour le portfolio). Aucun coût supplémentaire n'est engagé pour la mise en place de la veille. Je dispose également de l'ISO de Debian 13 et d'un accès internet via mon reverse-proxy pour télécharger les paquets requis.

Analyse

Descriptifs des solutions

Deux approches permettent de mettre en place une veille technologique automatisée sur les publications du CERT-FR. Je compare N8N en auto-hébergement avec Inoreader en mode SaaS.

N8N auto-hébergé : N8N est une plateforme d'automatisation de workflows open-source (licence fair-code). L'utilisateur installe N8N sur un serveur qu'il administre lui-même, construit ses workflows dans une interface visuelle et orchestre des actions entre services (flux RSS, webhooks, API, bases de données). L'hébergement, la sécurité, les mises à jour et les sauvegardes sont à la charge de l'administrateur. Cette approche offre un contrôle total sur l'infrastructure et sur les données manipulées.

Inoreader : Inoreader est un agrégateur de flux RSS hébergé par l'éditeur. L'utilisateur crée un compte sur <https://inoreader.com> et s'abonne aux flux de son choix. L'offre gratuite est limitée à 150 flux RSS. L'offre Pro, facturée 6,67 € TTC/mois en facturation annuelle ou 8,99 € TTC/mois en facturation mensuelle, ajoute les règles automatiques, les digests, l'accès à l'API et d'autres fonctionnalités avancées. La maintenance est entièrement gérée par Inoreader.

Comparaison des solutions (tableau)

Critères	N8N auto-hébergé	Inoreader
Coûts	20 € TTC/mois (serveur Proxmox déjà loué avant la mise en place de la veille) + 1,50 € TTC/mois (IP publique du reverse-proxy, déjà payée avant la mise en place de la veille) + 11 € TTC/an pour le nom de domaine (déjà enregistré pour le portfolio).	Gratuit (limité à 150 flux RSS) ou 6,67 € TTC/mois en facturation annuelle (80,04 € TTC/an) ou 8,99 € TTC/mois en facturation mensuelle pour l'offre Pro.
Hébergement	Auto-hébergé sur mon serveur Proxmox chez OVH (datacenter de Gravelines en France).	Hébergé par Inoreader, localisation du datacenter non choisie par l'utilisateur.
Personnalisation / automatisation	Totale : workflows construits librement, enchaînement de nœuds (RSS, webhooks, API, code JavaScript) sans limite sur le nombre de règles.	Limitée aux règles et filtres proposés par la plateforme. Fonctionnalités avancées (règles, digests, API) réservées à l'offre Pro.
Intégration Discord & Notion	Nœuds Discord et Notion natifs dans N8N, appels de webhooks et d'API supportés en standard dans les workflows. Webhook N8N exposé pour réceptionner les JSON envoyés directement par Claude.	Pas d'intégration native Discord ni Notion. Passage obligé par l'API Inoreader (offre Pro) et une solution d'automatisation externe.
Contrôle des données	Total : workflows, identifiants et base SQLite stockés sur ma VM, aucune donnée transmise à un tiers en dehors des appels API explicites (Discord, Claude et Notion).	Abonnements, règles et contenus lus stockés sur les serveurs d'Inoreader. Dépendance totale à la politique de confidentialité de l'éditeur.
Facilité d'installation et de configuration	Nécessite la création d'une VM, l'installation de Node.js et de N8N, la configuration du service systemd et l'ajout d'un vhost sur le reverse-proxy.	Immédiate : création du compte et ajout des flux via l'interface web, aucune installation à effectuer.
Maintenance	Entièrement à ma charge : mises à jour de N8N et de Node.js, sauvegardes, renouvellement du certificat SSL.	Gérée par la plateforme.
Montée en compétences	Forte : administration système, gestion d'un service systemd, automatisation de workflows, manipulation de webhooks, d'API et de JSON.	Faible : utilisation d'une interface web grand public.

Choix d'une solution - Argumentation

Je choisis N8N auto-hébergé. Tout d'abord, l'infrastructure est déjà en place (serveur Proxmox et nom de domaine déjà loués pour le portfolio et d'autres projets personnels), aucun coût supplémentaire n'est donc engagé. Ensuite, N8N permet d'orchestrer entièrement les workflows :

- Flux RSS → N8N → Discord
- JSON généré par Claude → webhook N8N → Notion

Ce qu'Inoreader ne permet pas de faire de manière équivalente même avec son offre Pro. La fonctionnalité « Projets » de Claude assure par ailleurs la standardisation du format JSON produit, garantissant la cohérence des entrées dans Notion, et automatise l'envoi du JSON directement au webhook N8N. Enfin, l'auto-hébergement constitue une montée en compétences concrète en administration système et réseaux, en cohérence avec mon BTS SIO option SISR.

Plan d'adressage - Schéma - Tables de routage

- Tableau d'adressage :

Nom du réseau	@ réseau	Masque	1er @ IP	Dernière @ IP	@ broadcast	Passerelle	Nombre d'hôtes
Public Proxmox	149.202.85.0	255.255.255.0	149.202.85.1	149.202.85.254	149.202.85.255	149.202.85.254	254
Public Reverse-proxy	51.68.2.72	255.255.255.255	51.68.2.72	51.68.2.72	51.68.2.72	149.202.85.154	1
Privé	10.0.0.0	255.255.255.0	10.0.0.1	10.0.0.254	10.0.0.255	10.0.0.1	254

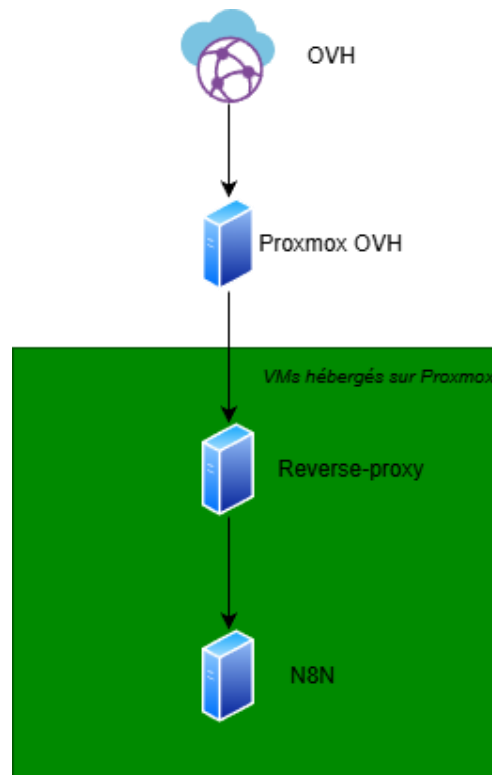
- Tableau des IP :

Nom du réseau	Proxmox OVH	Reverse-proxy	N8N
Public Proxmox	149.202.85.154		
Public Reverse-proxy		51.68.2.72	
Privé	10.0.0.7	10.0.0.1	10.0.0.4

- Table de routage du reverse-proxy :

Adresse réseau	Masque réseau	Adresse passerelle	Interface
0.0.0.0	0.0.0.0	149.202.85.154	ens18
10.0.0.0	255.255.255.0	10.0.0.1	ens19

- Schéma du réseau :



Étude de l'impact sur le SI existant

- **Impact technique :**

Le serveur Proxmox héberge déjà les deux VM concernées (reverse-proxy et N8N). Le seul ajout au niveau réseau est la VM N8N (10.0.0.4), qui obtient une IP privée sur le réseau interne déjà en place et accède à Internet via le reverse-proxy qui fait office de passerelle. Aucun port supplémentaire n'est à ouvrir sur le pare-feu externe : le reverse-proxy expose uniquement les ports 80 (http) et 443 (HTTPS) vers l'extérieur et redirige le trafic en interne vers le port 5678 de la VM N8N. L'ajout d'un vhost Nginx supplémentaire sur le reverse-proxy ne génère pas de charge significative, ce dernier servant déjà d'autres services. N8N utilise SQLite comme base de données locale, ce qui évite toute dépendance ou impact sur le serveur MariaDB existant (non mentionné plutôt). Le renouvellement automatique du certificat SSL via Certbot introduit une dépendance à surveiller : un échec de renouvellement rendrait l'interface N8N inaccessible, ce risque étant toutefois faible et rapidement détecté.

- **Impact organisationnel :**

La maintenance de la veille est entièrement à ma charge : mises à jour de N8N et de Node.js, vérification du renouvellement du certificat SSL, vérification du bon fonctionnement du service systemd et sauvegardes de la VM N8N. Les sauvegardes automatiques toutes les 2 heures déjà en place sur le serveur Proxmox couvrent également la nouvelle VM N8N, ce qui garantit une reprise rapide en cas d'incident. La sauvegarde du serveur Proxmox vers le datacenter OVH de Roubaix via l'agent Backup Agent s'applique également à cette VM, assurant une protection contre les incidents matériels.

- **Impact stratégique :**

La veille technologique sur les publications du CERT-FR constitue un outil de suivi de l'actualité cybersécurité directement en lien avec le référentiel du BTS SIO option SISR. Elle démontre une capacité à concevoir et exploiter un système d'automatisation complet reposant sur des services SaaS et une infrastructure auto-hébergée. L'infrastructure mise en place peut être réutilisée pour d'autres projets de veille ou d'automatisation à l'avenir, sans coût supplémentaire.

- **Impact humain :**

La mise en place de la veille impose d'adopter une nouvelle routine de travail : à chaque notification Discord signalant de nouvelles publications du CERT-FR, il est nécessaire de consulter les liens, de les soumettre au projet Claude et de vérifier que les entrées ont bien été insérées dans Notion. Cette organisation demande une certaine rigueur pour ne pas laisser s'accumuler les publications non traitées. Le système automatisé réduit significativement le temps consacré à la surveillance manuelle des sources, mais il ne dispense pas d'un regard critique sur les résultats produits par l'IA, dont les synthèses doivent être vérifiées pour s'assurer qu'elles reflètent fidèlement le contenu original.

- **Impact juridique :**

La veille ne collecte aucune donnée personnelle : elle traite uniquement des publications publiques du CERT-FR et ne met en œuvre aucun formulaire, aucune authentification utilisateur ni aucun cookie de tracking. Les données transitant dans le pipeline (titres, liens, synthèses) proviennent de sources gouvernementales accessibles publiquement et sont stockées dans un espace Notion personnel. L'hébergement de la VM N8N sur le serveur Proxmox OVH en France (datacenter de Gravelines) garantit que l'infrastructure reste soumise au droit français et européen. OVHcloud est certifié par de multiples normes et est conforme au RGPD.

Phasage de l'intervention

Je commencerai par configurer le reverse-proxy Nginx en créant le vhost dédié à n8n.mathys-demon.fr avec les directives de proxy et de support WebSocket, puis en générant le certificat SSL Let's Encrypt via Certbot.

J'installerai ensuite Node.js sur la VM N8N (IP : 10.0.0.4, OS : Debian 13) en ajoutant le dépôt NodeSource, puis je vérifierai que la version installée est bien comprise entre v20.19 et v24.x, plage imposée par N8N.

Je procéderai ensuite à l'installation de N8N via npm, puis je créerai le fichier de service systemd avec les variables d'environnement nécessaires avant d'activer et de démarrer le service à chaque démarrage de la VM.

Je configurerai ensuite le workflow N8N depuis l'interface web : création des 8 nœuds RSS surveillant chacun un flux CERT-FR distinct, création du nœud Aggregate regroupant les champs « titre » et « lien » des flux RSS, création des deux nœuds Discord pour l'envoi séparé des titres et des liens, création du nœud Webhook exposé en GET pour que Claude envoie le JSON sur ce Webhook, et création du nœud Notion chargé d'insérer les champs du fichier JSON dans la base de données de Notion.

Enfin, je créerai le « Projet » sur <https://claude.ai/projects> et j'y saisisrai les instructions standardisées définissant le format JSON de sortie attendu, l'URL complète du webhook

N8N et les règles d'envoi à respecter selon que je transmets un ou plusieurs liens CERT-FR.

Prévision des tests de validation

Je vérifierai que l'interface de N8N est accessible en saisissant l'URL <https://n8n.mathys-demon.fr/> dans un navigateur. Je contrôlerai que la connexion est bien établie en HTTPS avec un certificat SSL valide.

Je testerai la redirection HTTP vers HTTPS en accédant au site via <http://n8n.mathys-demon.fr/> et en vérifiant que le navigateur bascule automatiquement en HTTPS. Je testerai également le redémarrage automatique du service : après un redémarrage de la VM N8N, je vérifierai que le service géré par systemd redémarre seul et que l'interface est à nouveau accessible.

Enfin, je déclencherai manuellement le workflow pour vérifier la chaîne complète : récupération d'un item du flux RSS du CERT-FR, envoi du titre et du lien sur le webhook Discord, copie manuelle du lien dans le projet Claude, génération du JSON par Claude, envoi automatique de ce JSON au webhook N8N via requête HTTP, puis insertion correcte de la synthèse dans la base de données Notion.

Mise en place

La partie Mise en place a été rédigée sur Notion vous la trouverez via le lien suivant : <https://mathys-demon.notion.site/Mise-en-place-de-la-veille-technologique-CERT-FR-34d8102ecdc981ec9ae2e85c932f6f36>.

Bilan

Conclusion

La veille technologique sur les publications du CERT-FR est opérationnelle. Le workflow construit dans N8N récupère automatiquement les flux RSS et envoie les nouvelles publications (titre et liens) dans le salon Discord dédié via un webhook. Après un envoi manuel des liens dans le projet sur Claude, l'IA génère un fichier JSON structuré pour chaque publication et le transmet directement au webhook N8N via requête HTTP, sans intervention manuelle supplémentaire. N8N archive ensuite chaque entrée dans la base de données Notion, qui présente les synthèses sous forme d'un calendrier consultable publiquement en ligne. Le projet permet un suivi régulier de l'actualité cybersécurité, tout en conservant la maîtrise complète de l'infrastructure et des données. La solution repose intégralement sur des ressources déjà en place (serveur Proxmox, reverse-proxy, nom de domaine), sans coût supplémentaire engagé pour la mise en place de la veille.

Auto-évaluation

La principale difficulté rencontrée a été la construction d'un pipeline fiable entre plusieurs services externes (flux RSS, Discord, API Claude, webhook N8N, Notion). L'appel à l'IA et son retour au format JSON demandent une gestion précise du prompt et de la structure de sortie pour que la base Notion reçoive systématiquement le même format. L'utilisation de la fonctionnalité « Projets » de Claude a permis de standardiser ce format en conservant la même consigne pour chaque synthèse, et d'automatiser l'envoi du JSON directement au webhook N8N sans action manuelle intermédiaire. Ce projet m'a permis de consolider mes compétences en automatisation avec N8N, en manipulation de formats structurés (JSON), en rédaction de prompt pour des IA ainsi qu'en intégration de services SaaS via des API. Si je devais refaire ce projet, je mettrais en place une authentification sur la requête HTTP envoyée par Claude au webhook N8N qui alimente la base Notion. Dans la version actuelle, ce webhook est exposé publiquement en HTTPS mais sans mécanisme d'authentification, ce qui fait que toute personne connaissant l'URL pourrait théoriquement envoyer un JSON et déclencher une insertion dans la base de données sur Notion. L'ajout d'une authentification (par exemple un token secret transmis dans un en-tête HTTP personnalisé, à vérifier côté N8N) permettrait de garantir que seules les requêtes authentifiées soient acceptées, et renforcerait ainsi la sécurité.