TECHNICAL REPORT — TR-2026-01

Opération « OpenClaw »

Anatomie d'une cyberattaque pilotée par intelligence artificielle contre une entreprise pharmaceutique

De la reconnaissance OSINT à la double extorsion : modélisation d'une kill chain agentique en cinq phases

Auteur : Fabrice Pizzi

Affiliation : Université Paris Sorbonne

Date : Février 2026

Version : 8.0

⚠ AVERTISSEMENT

Ce document présente l'intégralité de l'Opération « OpenClaw », une cyberattaque fictive pilotée par intelligence artificielle ciblant l'entreprise pharmaceutique MediFrance SA. Il couvre cinq phases : reconnaissance OSINT automatisée (WormGPT, Shodan, stylométrie), armement par supply chain IA (skill piégée ClawHub, PromptLock, clone vocal), livraison multi-vecteurs (infostealer, VPN 0-day, installation skill), mouvement latéral par agent IA autonome (LotL, Mimikatz, prompt injection Slack, PoisonGPT, exfiltration R&D), et actions finales (ransomware PromptLock + double extorsion 2 M€). Les techniques, outils et procédures décrits le sont exclusivement à des fins pédagogiques et de sensibilisation académique.

Introduction Générale

L'Opération « OpenClaw » est un scénario de cyberattaque fictif conçu pour illustrer la convergence entre intelligence artificielle offensive et techniques d'intrusion avancées. Il décrit l'attaque complète d'une entreprise pharmaceutique française de taille intermédiaire, MediFrance SA (500 employés, CA 120 M€), par un groupe d'attaquants exploitant cinq capacités IA distinctes à chaque étape de la Cyber Kill Chain de Lockheed Martin [1].

Ce document consolide les cinq phases de l'opération, de la reconnaissance initiale (J−30) jusqu'aux actions finales sur l'objectif (J+6), soit six semaines d'activité offensive (il est à prévoir que cette timeline soit réduite drastiquement si on considère les progrès actuels enregistrés dans le domaine de la recherche). L'analyse s'appuie sur des recherches et incidents documentés de 2024–2026 : le rapport Verizon DBIR 2025 [10], le framework OWASP Top 10 LLM Applications 2025 [25], les vulnérabilités documentées d'OpenClaw (CVE-2026-25253, CVSS 8.8) [7], la preuve de concept PoisonGPT de Mithril Security [123], la vulnérabilité EchoLeak CVE-2025-32711 (Microsoft 365 Copilot, CVSS 9.3) [121], ou encore le modèle de Promptware Kill Chain de C. Schneider (2026) [120].

Marcus Sachs (Center for Internet Security) prédit que dès 2026, les moteurs de mouvement latéral entièrement automatisés ne nécessiteront que peu ou pas d'intervention humaine [3]. John Grady (Omdia) observe que les techniques Living-off-the-Land ne feront que croître avec l'émergence des agents IA offensifs [3]. Control Risks avertit que la question n'est plus de savoir si l'IA va transformer la cybermenace, mais à quelle vitesse les organisations seront confrontées à cette réalité [4]. Le rapport Securin 2025, analysant 7 061 victimes confirmées à travers 117 groupes de ransomware, confirme que l'IA fonctionne désormais comme un accélérateur à chaque étape d'une attaque [147]. L'Opération OpenClaw incarne concrètement ces quatre prédictions.

Synthèse de la Kill Chain OpenClaw

Le tableau ci-dessous résume les cinq phases de l'opération, chacune exploitée en détail dans la suite du document :

Phase	Kill Chain	Technique IA clé	Durée	Impact
1	Reconnaissance	WormGPT OSINT + Shodan fingerprint OpenClaw	J−30 à J−15	Cible + vulnérabilités
2	Armement	Skill piégée ClawHub + PromptLock (Go/Ollama)	J−15 à J−7	Arsenal prêt
3	Livraison + Exploitation	supply chain skill	J−7 à J	4 accès
4	Latéralisation + C2	Prompt injection Slack + PoisonGPT + LotL	J+1 à J+5	Domain Admin + PI
5	Actions sur objectif	PromptLock polymorphe + double extorsion	J+6	≥2,89 M€

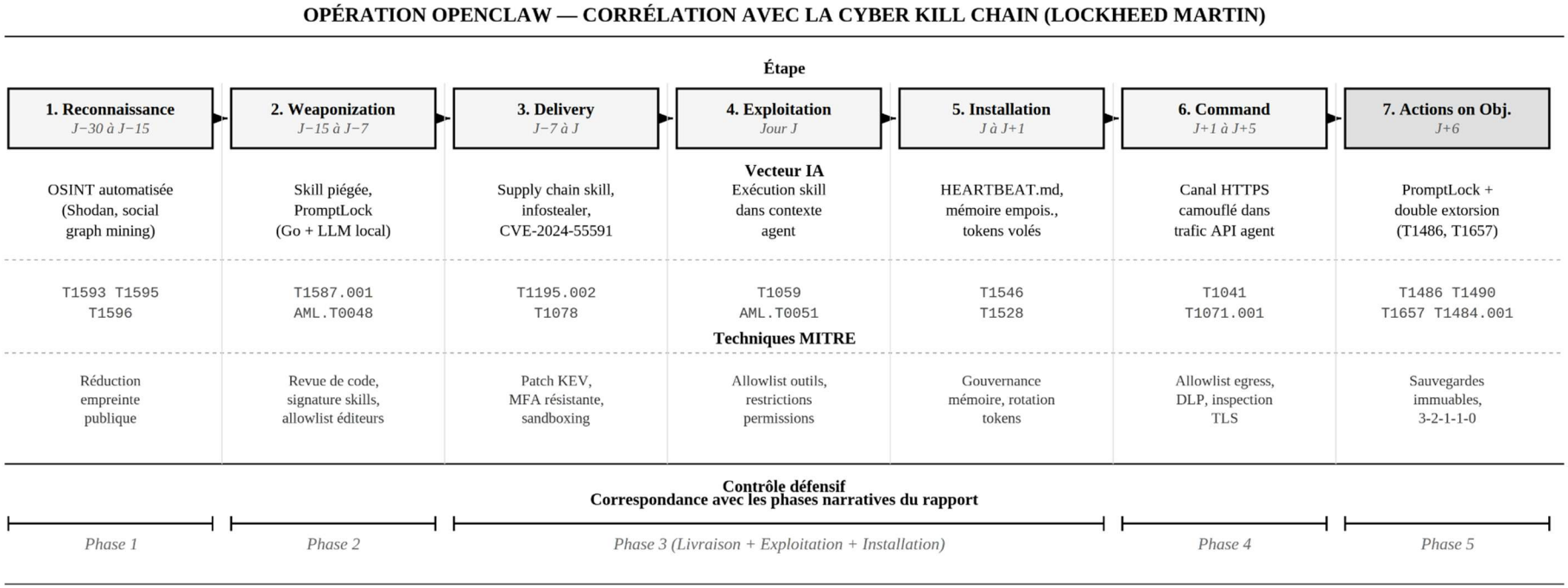


Figure 1. Corrélation de l'Opération OpenClaw avec les sept étapes de la Cyber Kill Chain de Lockheed Martin [1].
Pour chaque étape : le vecteur IA exploité, les techniques MITRE ATT&CK/ATLAS correspondantes, et le contrôle défensif prioritaire.
Les accolades inférieures montrent la correspondance avec les cinq phases narratives du rapport. La Phase 3 couvre trois
étapes de la Kill Chain (Delivery, Exploitation, Installation), ce qui reflète le chevauchement temporel de ces actions dans le scénario.
L'étape 7 (fond grisé) représente l'objectif final de l'opération.

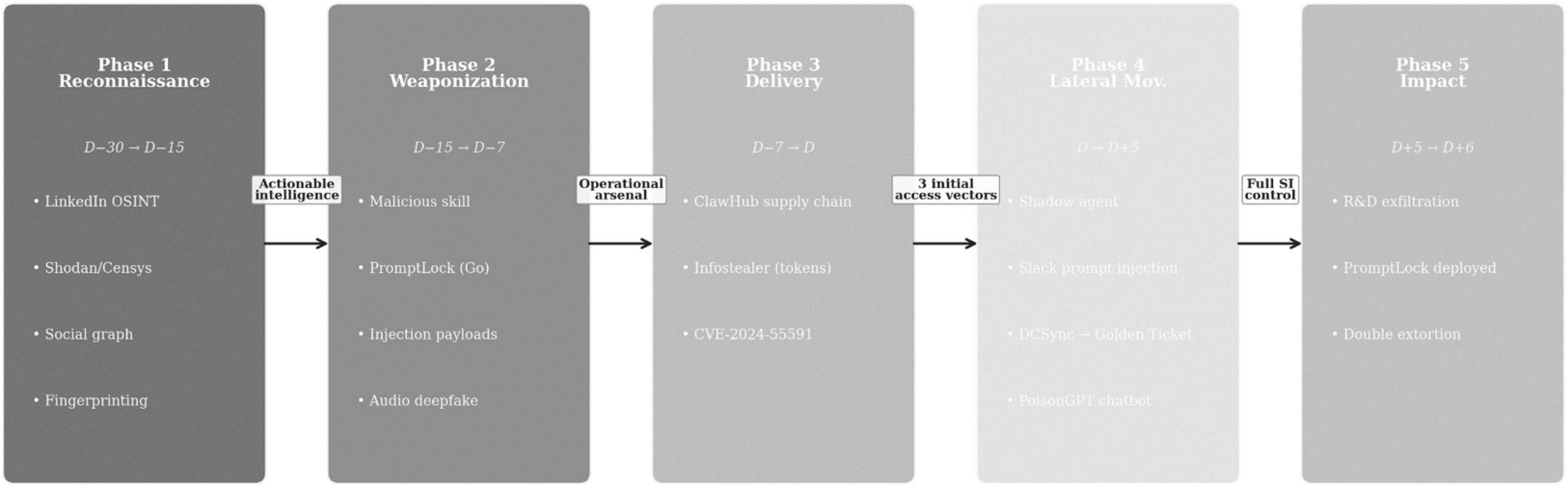
> Le rôle central d'OpenClaw dans l'opération
>
> L'agent IA OpenClaw n'est pas un simple outil auxiliaire : il est le fil rouge de toute l'opération. Phase 1 : son instance exposée sur Shodan révèle la cible et ses vulnérabilités (CVE-2026-25253, gateway 0.0.0.0). Phase 2 : sa marketplace communautaire ClawHub est exploitée pour la supply chain. Phase 3 : sa skill piégée vole credentials, clés API et commence l'exfiltration HTTPS. Phase 4 : son agent est détourné par prompt injection indirecte via Slack, devenant un « insider involontaire » exécutant des commandes réseau via son accès terminal légitime, tandis que son trafic HTTPS légitime sert de canal d'exfiltration invisible au WAF/EDR pendant 5 jours. Phase 5 : les données R&D exfiltrées via OpenClaw alimentent la double extorsion.

Structure du document

Ce document est organisé en cinq chapitres correspondant aux cinq phases de l'Opération OpenClaw, chacun incluant : le contexte opérationnel, l'analyse technique détaillée des techniques et procédures, les correspondances MITRE ATT&CK et ATLAS, les références académiques et industrielles. Un bilan global consolidé et une bibliographie unifiée ([1] à [160]) closent le document.

- **Phase 1 — Reconnaissance (J−30 à J−15) :** cartographie organisationnelle par Social Graph Mining, stylométrie computationnelle, fingerprinting passif et inférence de vulnérabilités ;
- **Phase 2 — Armement (J−15 à J−7) :** création de la skill piégée PharmaResearch Assistant, assemblage de PromptLock (Go/Ollama), et préparation de l'arsenal complet ;
- **Phase 3 — Livraison et Exécution (J−7 à J) :** attaque multi-vecteurs simultanée : , infostealer via supply chain fournisseur, exploitation VPN Fortinet CVE-2024-55591, installation de la skill piégée ;
- **Phase 4 — Mouvement Latéral et Persistance (J+1 à J+5) :** agent IA autonome LotL, compromission AD via Mimikatz (DCSync, Golden Ticket), détournement OpenClaw via prompt injection Slack, empoisonnement chatbot PoisonGPT, exfiltration R&D via HTTPS légitime, destruction des sauvegardes ;
- **Phase 5 — Actions sur l'Objectif (J+6) :** bilan exfiltration R&D, déploiement PromptLock (200 postes + 15 serveurs en 40 min), double extorsion (2 M€ rançon + menace publication PI).

Anatomie d'une cyberattaque pilotée par intelligence artificielle contre une entreprise pharmaceutique



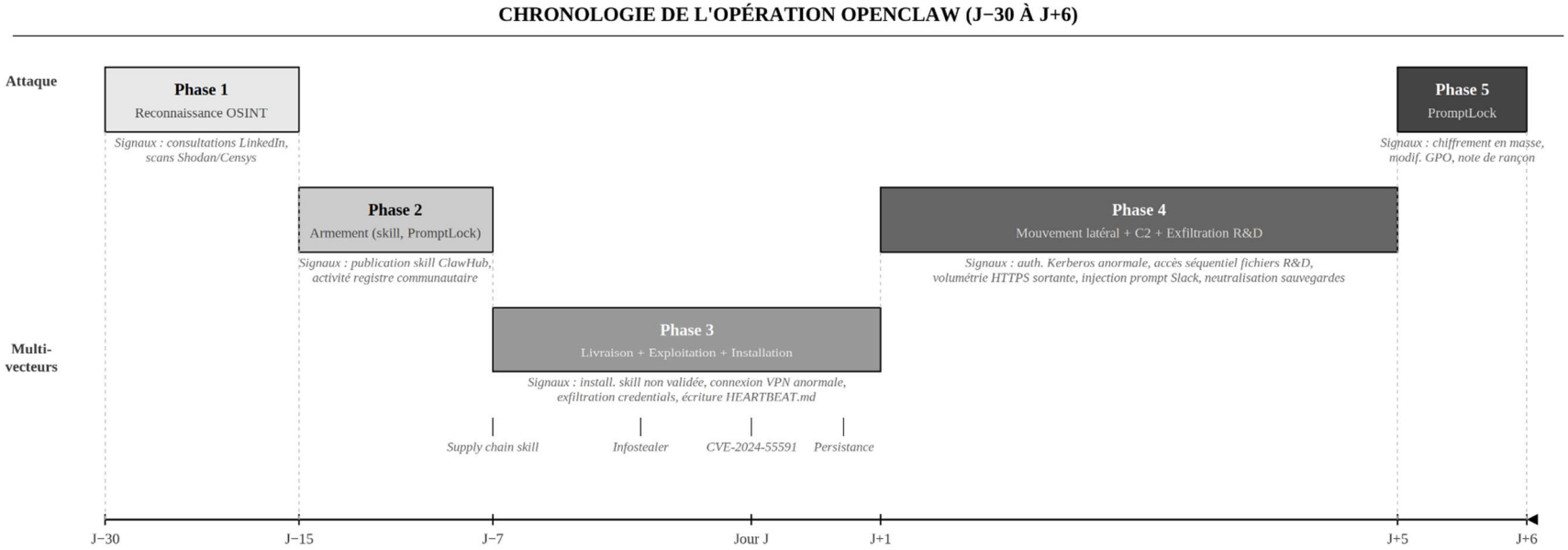


Figure 2. Chronologie de l'Opération OpenClaw. Les barres représentent la durée de chaque phase ; leur intensité croissante reflète la progression du niveau de privilège de l'attaquant. Les signaux de détection identifiés sous chaque phase constituent autant de fenêtres d'intervention pour les équipes défensives. La Phase 3 couvre trois étapes de la Kill Chain (Delivery, Exploitation, Installation) avec des vecteurs parallèles. La temporalité est illustrative (cf. section 2.3).

Résumé

Ce document présente l'analyse académique exhaustive de la phase de reconnaissance (J-30 à J-15) de l'Opération « OpenClaw », un scénario de cyberattaque fictif mais réaliste ciblant une ETI pharmaceutique française via l'exploitation de l'agent IA autonome open-source OpenClaw. La méthodologie de reconnaissance se décompose en trois axes d'acquisition simultanés : (1) la cartographie organisationnelle par fouille de graphes sociaux (Social Graph Mining), (2) le profilage sémantique et stylistique par stylométrie computationnelle, et (3) l'énumération passive de l'infrastructure technique par fingerprinting et corrélation de vulnérabilités (CVE). L'ensemble du processus est orchestré par un LLM non aligné (WormGPT) et s'appuie sur un modèle d'inférence fondé sur les Information Inference Diagrams (I2D). L'analyse intègre les vulnérabilités réelles d'OpenClaw documentées par Cisco, Sophos, SecurityScorecard et BitSight (janvier-février 2026), et les taxonomies OWASP Top 10 LLM 2025, MITRE ATT&CK et MITRE ATLAS.

Mots-clés : reconnaissance automatisée, OSINT, Social Graph Mining, stylométrie computationnelle, fingerprinting passif, inférence de vulnérabilités, WormGPT, OpenClaw, Shodan, CVE, I2D, shadow AI

1. Introduction et cadre d'opération stratégique

L'évolution contemporaine des chaînes de frappe cybernétique (Cyber Kill Chains) démontre une compression significative des délais d'exploitation, couplée à une sophistication structurelle accrue des phases préparatoires [1]. La séquence temporelle s'étendant de J-30 à J-15 représente la fenêtre critique de la reconnaissance initiale. Historiquement, cette phase pré-offensive était caractérisée par des processus manuels chronophages d'agrégation d'informations ou par une automatisation rudimentaire reposant sur des balayages de ports actifs [1, 2].

L'émergence des modèles de langage de grande taille (LLM) non alignés, tels que WormGPT ou FraudGPT, a fondamentalement altéré cette dynamique opérationnelle en permettant une automatisation systémique, intelligente et asymétrique de la collecte de renseignements de sources ouvertes (OSINT) [3, 4]. WormGPT 4, analysé par l'Unit 42 de Palo Alto Networks en novembre 2025, est un chatbot construit sur le modèle GPT-J, entraîné sur des données liées aux malwares, et commercialisé via Telegram et les forums underground [5]. Il génère des scripts fonctionnels de chiffrement, des notes de rançon, et des e-mails de spear-phishing sans aucun garde-fou éthique [5, 6].

Parallèlement, l'adoption fulgurante de l'agent IA autonome OpenClaw (anciennement ClawdBot, puis MoltBot) – 180 000+ étoiles GitHub, 720 000 téléchargements hebdomadaires, plus de 40 000 instances exposées sur Internet [7, 8] – crée une surface d'attaque sans précédent. Cisco qualifie OpenClaw de « cauchemar sécuritaire » [9], Sophos le classe comme PUA [10], et le chercheur Jamieson O'Reilly (Dvuln) a démontré que les instances exposées sont détectables par simple recherche d'empreintes HTML sur Shodan [11].

Le présent document détaille de manière exhaustive les mécanismes algorithmiques et les heuristiques de corrélation sous-jacents à la phase de reconnaissance de l'Opération OpenClaw, un scénario d'attaque ciblant l'entreprise fictive MediFrance SA (ETI pharmaceutique, 500 employés). Trois employés R&D ont installé OpenClaw sans validation de la DSI, créant une situation de Shadow AI exploitée par l'attaquant.

2. Axe 1 : Cartographie organisationnelle par fouille de graphes sociaux

2.1 Découverte de la cible via Shodan et empreinte OpenClaw

La première étape de la reconnaissance exploite la configuration par défaut non sécurisée d'OpenClaw. Les versions initiales du projet liaient le gateway sur 0.0.0.0:18789, écoutant sur toutes les interfaces réseau, y compris Internet [8, 12]. Lorsqu'un reverse proxy Nginx ou Caddy est placé devant l'instance, toute connexion externe apparaît comme provenant de 127.0.0.1, et OpenClaw, qui accorde par défaut sa confiance à localhost, traite ces connexions comme locales et fiables [7, 12].

L'attaquant utilise Shodan pour scanner Internet à la recherche des signatures HTTP caractéristiques du panneau de contrôle OpenClaw. Cette technique a été démontrée publiquement par Jamieson O'Reilly, fondateur de Dvuln, qui a identifié des instances exposées en recherchant les empreintes HTML spécifiques du gateway [11]. BitSight a observé plus de 30 000 instances exposées entre le 27 janvier et le 8 février 2026, avec des pics de croissance en temps réel durant l'enquête [8]. SecurityScorecard a confirmé 28 663 adresses IP uniques avec des panneaux de contrôle exposés répartis dans 76 pays, dont 12 812 vulnérables à l'exécution de code à distance (RCE) via CVE-2026-25253 (CVSS 8.8) [7].

L'attaquant identifie une instance OpenClaw chez MediFrance SA, rendue accessible depuis l'extérieur via un reverse proxy Nginx mal configuré. Via l'accès au gateway exposé, il observe les requêtes DNS vers les modèles IA utilisés et identifie les skills installées. Comme l'a documenté Hudson Rock, les fichiers de configuration OpenClaw (openclaw.json, device.json, soul.md) contiennent les jetons d'authentification, les clés cryptographiques et les principes opérationnels de l'agent [13].

> **Vulnérabilité clé : CVE-2026-25253 (CVSS 8.8)**
>
> Exécution de code à distance en un clic via détournement WebSocket inter-sites. Un utilisateur visitant une page web malveillante voit son jeton d'authentification volé en millisecondes. L'attaquant se connecte alors au gateway de la victime, désactive le sandboxing et obtient une RCE complète [7, 11, 12].

2.2 Automatisation du profilage OSINT par LLM malveillants

Le profilage OSINT (Open Source Intelligence) connaît une mutation profonde avec l'émergence de Large Language Models non alignés, dont WormGPT constitue l'archétype le plus documenté. Ces modèles, catégorisés comme *maligned* selon la taxonomie établie, sont des LLM dont les garde-fous éthiques ont été intentionnellement supprimés pour faciliter des opérations offensives [6]. *Il convient de distinguer trois catégories : les modèles* unaligned *(jamais alignés mais non malveillants),* uncensored *(alignement retiré sans intention malveillante) et* maligned *(intentionnellement malveillants et probablement illégaux). WormGPT appartient à cette dernière catégorie* [5]. WormGPT 4, développé à partir du modèle open-source GPT-J et commercialisé sur des forums clandestins et Telegram depuis 2023, permet l'automatisation de tâches malveillantes : génération de campagnes de phishing/BEC (Business Email Compromise), création de malware polymorphe, et automatisation avancée de la reconnaissance [6][5].

Architecture technique du profilage automatisé

L'agent d'IA déploie une architecture multi-couches pour extraire systématiquement les métadonnées des profils LinkedIn. Le processus s'articule autour de deux composants dont les capacités sont inégalement documentées :

1. Collecte automatisée (Automated Reconnaissance) — *capacité établie et documentée*

Les attaquants utilisent des techniques d'OSINT automatisées via des frameworks comme Maltego, Recon-ng, SpiderFoot et TheHarvester pour extraire massivement des données LinkedIn [165]. LinkedIn représente une « mine d'or » pour l'OSINT, avec 60 % des campagnes d'intrusion débutant par une phase de reconnaissance sur réseaux sociaux [17]. Les métadonnées collectées incluent : nom, titre professionnel, entreprise actuelle, localisation géographique, compétences listées, publications, commentaires, connexions visibles, durée dans les postes, et timestamps d'activité.

2. Traitement par LLM non aligné — *capacités partiellement documentées*

WormGPT intervient pour traiter ces données brutes et générer des inférences contextuelles. Contrairement aux LLM standards (ChatGPT, Claude) qui refuseraient de participer à des opérations offensives, WormGPT n'applique aucun filtre éthique [6]. *Ses capacités documentées dans la littérature incluent* la génération de phishing/BEC et la création de malware polymorphe [5]. *Les capacités suivantes sont techniquement plausibles et facilement réalisables avec tout LLM non censuré, mais ne font pas l'objet de publications académiques spécifiques à WormGPT* :

- Analyser sémantiquement les descriptions de postes pour identifier les responsabilités techniques (accès VPN, gestion d'infrastructures critiques)
- Corréler des informations fragmentées pour reconstruire l'organigramme organisationnel
- Identifier les cibles prioritaires (administrateurs systèmes, responsables sécurité)
- Générer des prétextes d'ingénierie sociale personnalisés

Note : ces opérations ne requièrent aucune capacité technique spécifique à WormGPT — tout LLM général (y compris des modèles open-source non censurés type Mistral ou LLaMA sans garde-fous) en est capable. La barrière est éthique, non technique.

2.3 Fouille de graphes sociaux (Social Graph Mining)

Fondements théoriques du Social Network Alignment

La fouille de graphes sociaux constitue le premier pilier de l'inférence passive, permettant de cartographier l'écosystème relationnel d'une organisation cible. Le Social Network Alignment (SNA) est un domaine de recherche légitime en data science qui vise à associer plusieurs identités numériques appartenant au même individu ou à identifier des relations cachées entre entités [14].

Évolution méthodologique — état de l'art établi

L'approche traditionnelle reposait sur des modèles supervisés nécessitant d'immenses corpus de données étiquetées (labeled datasets) pour entraîner des classificateurs. Ces modèles présentaient plusieurs limitations :

coût prohibitif d'annotation manuelle, faible généralisation hors distribution, et vulnérabilité aux évolutions des plateformes sociales.

Les modèles contemporains déploient des algorithmes d'apprentissage non supervisé et auto-supervisé, notamment [166] :

- **Graph Attention Networks (GAT)** : réseaux de neurones capables de pondérer dynamiquement l'importance des nœuds voisins
- **Contrastive Multi-View Learning** : apprentissage contrastif exploitant plusieurs vues (structurelle, sémantique, temporelle) pour améliorer la représentation des entités
- **Graph Convolutional Networks (GCN)** : convolutions adaptées aux structures graphiques non-euclidiennes
- **Unsupervised Node Embeddings** : techniques type Node2Vec [162], DeepWalk générant des représentations vectorielles des nœuds sans supervision

Ces architectures permettent un alignement de réseaux sociaux capable d'identifier automatiquement des patterns relationnels sans données d'entraînement préalablement étiquetées [14].

Toutefois, les publications scientifiques sur le SNA ne documentent pas leur utilisation spécifique par des LLM malveillants comme WormGPT à des fins offensives. La convergence entre SNA et LLM offensifs constitue un scénario prospectif techniquement réalisable, construit à partir de composants individuellement établis.

2.4 Modélisation mathématique du graphe organisationnel

Formalisation du graphe social — modélisation théorique à partir de composants établis

L'organisation MediFrance est mathématiquement modélisée sous la forme d'un graphe orienté et pondéré $G = (V, E)$, où :

- $V = \{v_1, v_2, ..., v_n\}$ représente l'ensemble des nœuds (les n collaborateurs identifiés)
- $E \subseteq V \times V$ représente l'ensemble des arêtes (les relations déduites)
- $w : E \rightarrow \mathbb{R}^+$ est une fonction de pondération assignant un poids positif à chaque arête

Une arête orientée $e = (u,v) \in E$ indique une relation inférée du collaborateur u vers le collaborateur v (par exemple : u mentionne v, u commente les publications de v, u et v partagent des connexions communes).

Calcul des poids d'arêtes

Le poids $w(u,v)$ de chaque arête est calculé selon une fonction composite intégrant trois dimensions de proximité [14] :

$$w(u,v) = \alpha \cdot S\_sem(u,v) + \beta \cdot S\_spa(u,v) + \gamma \cdot S\_temp(u,v)$$

où α, β, γ sont des hyperparamètres de pondération (avec $\alpha + \beta + \gamma = 1$).

1. Proximité sémantique S_sem(u,v) — métrique établie [14]

Mesure la similarité des contenus textuels associés aux profils. Les embeddings vectoriels sont générés pour chaque collaborateur basés sur : descriptions de poste et compétences listées, contenu des publications et commentaires, vocabulaire technique employé.

La similarité sémantique est calculée via similarité cosinus :

$S\_sem(u,v) = (emb\_u \cdot emb\_v) / (\|emb\_u\| \cdot \|emb\_v\|)$

2. Proximité spatiale S_spa(u,v) — *modélisation composite établie*

Quantifie la distance géographique entre les collaborateurs et la distance topologique dans le graphe social :

$S\_spa(u,v) = \omega_1 \cdot e^{(-d\_geo(u,v)/\sigma)} + \omega_2 \cdot |N(u) \cap N(v)| / |N(u) \cup N(v)|$

où d_geo est la distance géographique, N(u) l'ensemble des voisins de u, σ un paramètre d'échelle, et le second terme est le coefficient de Jaccard sur les voisinages [14].

3. Proximité temporelle S_temp(u,v) — métrique établie

Analyse la synchronisation des activités : publications simultanées, commentaires dans des fenêtres temporelles proches, transitions de carrière corrélées :

$S\_temp(u,v) = corr(TS\_u, TS\_v)$

où TS_u et TS_v représentent les séries temporelles d'activité des utilisateurs u et v.

Inférence des relations hiérarchiques — *techniques établies*

L'analyse du graphe pondéré exploite des algorithmes de centralité établis [161] :

- **Arêtes de forte pondération** : relations collaboratives fréquentes, potentiellement au sein d'une même équipe
- **Analyse de centralité** : identification des hubs (managers, coordinateurs) via métriques de betweenness centrality et eigenvector centrality
- **Détection de communautés** : clustering algorithmique (Louvain, Leiden) pour segmenter les départements/équipes

2.5 Inférence des nœuds de confiance et de la hiérarchie fonctionnelle

Pour filtrer le bruit des connexions aléatoires, des modèles sensibles au degré (Degree-aware models, tels que DegUIL) corrigent les biais de voisinage inhérents aux graphes sociaux à distribution scale-free [168]. Le moteur d'inférence calcule trois métriques de centralité pour chaque nœud du graphe G :

- **Centralité de degré C_D(v)** : évalue le nombre de connexions directes d'un nœud, identifiant les collaborateurs les plus visibles dans le réseau professionnel.
- **Centralité d'intermédiarité C_B(v)** [161] : identifie les nœuds agissant comme des ponts vitaux entre différents clusters du réseau. C'est la métrique clé pour isoler les nœuds de confiance.
- **Centralité de proximité C_C(v)** : mesure la distance moyenne d'un nœud par rapport à tous les autres, identifiant les relais d'information les plus efficaces.

L'analyse isole spécifiquement les nœuds de confiance via la centralité d'intermédiarité. Ces nœuds stratégiques ne détiennent pas nécessairement un pouvoir décisionnel formel de haut niveau, mais possèdent une position névralgique dans les flux d'information : assistants de direction, coordinateurs de projets transversaux, ingénieurs du support IT [14]. Ces individus disposent généralement de privilèges d'accès étendus (délégation de messagerie, accès aux répertoires partagés critiques, droits d'administration) et sont culturellement conditionnés pour exécuter rapidement des requêtes urgentes émanant de la direction [17][6].

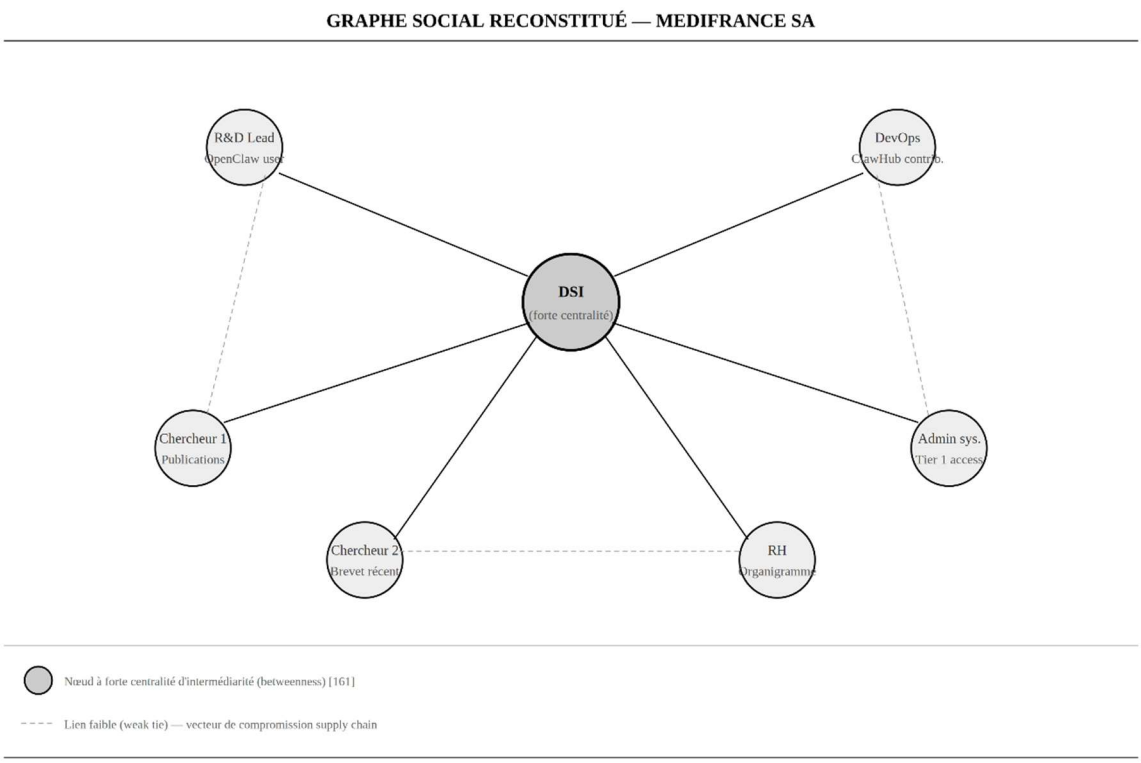


Figure 5. Graphe social reconstitué de MediFrance SA par fouille OSINT (LinkedIn, publications, ClawHub).
Le nœud DSI présente la plus forte centralité d'intermédiarité [161], le désignant comme cible prioritaire.
Les liens en pointillés représentent les weak ties exploités pour la compromission de la supply chain (Phase 2).

Implications opérationnelles — *dans le scénario OpenClaw*

Cette modélisation permet à l'attaquant de :

- Identifier les cibles prioritaires disposant d'accès privilégiés et utilisant OpenClaw
- Planifier la compromission via supply chain (les nœuds de confiance à forte centralité qui mentionnent OpenClaw dans leurs publications LinkedIn deviennent les cibles d'installation de la skill piégée en Phase 2)
- Exploiter le Shadow AI [26] : ces chercheurs R&D installent des skills depuis ClawHub sans validation de la DSI

3. Axe 2 — Stylométrie computationnelle (Writeprints) : attribution et détection d’usurpation

Cet axe vise à exploiter un signal souvent négligé en OSINT offensif : le **style d’écriture** comme empreinte comportementale. La stylométrie (ou “writeprints”) ne cherche pas d’abord le *quoi* (le contenu), mais le *comment* (les habitudes linguistiques), afin d’attribuer un texte à un auteur probable, de regrouper des messages par “main”, ou de détecter une usurpation (BEC, faux ordre de virement, faux message de direction) quand l’identité affichée n’est pas cohérente avec le style observé. Des travaux de forensique email ont montré l’intérêt pratique de la démarche, notamment via le clustering d’emails par style puis l’extraction de “writeprints” propres aux clusters.scienced

3.1 Fondements : pourquoi le style est un identifiant

Le style s'exprime dans des marqueurs relativement stables et peu "conscients" : fréquence des mots-outils (déterminants, prépositions, conjonctions), structures de phrases, ponctuation, préférences de vocabulaire fonctionnel. La littérature sur l'attribution d'auteur souligne que les **function words** sont particulièrement utiles car ils sont plus indépendants du sujet traité que les mots de contenu, et portent donc une information stylistique robuste.ar5iv.

3.2 Caractéristiques (features) : du texte brut au writeprint

Le pipeline stylométrique commence par normaliser et représenter chaque message (email, Slack, ticket) sous forme de vecteurs de caractéristiques. Les familles de signaux suivantes sont généralement efficaces :

- **Lexical** : distributions de mots-outils, ratio mots-outils / mots de contenu ("densité fonctionnelle"), richesse lexicale, répétitions.
- **N-grams** : n-grams de caractères (très utiles sur textes courts), n-grams de mots, patterns de bigrammes fréquents.
- **Structure** : longueurs moyennes de phrases, usage de puces, titres, signatures, formules d'ouverture/fermeture, "templates" récurrents.spectrum.library.
- **Ponctuation/typographie** : espaces, tirets, guillemets, emoji, majuscules, abréviations, fautes récurrentes. (À manier avec prudence, car facilement "corrigé" par relecture ou outils.)

Ces writeprints peuvent être calculés par message, puis agrégés au niveau d'un auteur (profil) ou d'un rôle (profil "CFO/DSI/chef de projet") pour obtenir un modèle de référence.

3.3 Modèles : attribution vs vérification (authorship validation)

Deux tâches sont à distinguer :

1. **Authorship attribution** : "qui, parmi N candidats, a écrit ce texte ?" (multi-classes).
2. **Authorship validation / verification** : "cet expéditeur est-il cohérent avec son style habituel ?" (binaire, très utile en sécurité email).

Des travaux récents proposent justement la "per-sender authorship validation" comme mécanisme défensif temps réel : pour chaque expéditeur, on maintient un profil de style, puis on classe les nouveaux emails comme authentiques vs inauthentiques. Ils évaluent par exemple un modèle naïf (Naive Bayes) et un classifieur plus moderne de type **Char-CNN** (réseau convolutionnel caractère), et discutent l'intégration dans une pile de sécurité email avec un coût opérationnel faible.

3.4 Intégration avec l'Axe 1 (graphe social) : "qui parle comme qui"

En offensif, un LLM non aligné peut tenter d'imiter le style à partir d'un corpus (mails publics, tickets, messages internes exfiltrés), ce qui augmente l'efficacité des prétextes BEC et réduit les signaux d'alerte humains. En défensif, la stylométrie sert de **capteur d'anomalie** pour repérer une rupture de style chez un expéditeur "de confiance", en complément des contrôles techniques (SPF/DKIM/DMARC, réputation, détection URL).

Tableau — Données → features → modèle → sortie → usage offensif/défensif

Données (entrée)	Features (writeprints)	Modèle / méthode	Sortie	Usage offensif (attaquant)	Usage défensif (blue team)
Emails internes (historiques par expéditeur), corpus "Enron-like"	Lexical/syntaxique/structurel (ponctuation, gabarits, longueurs), n-grams caractères/mots	Clustering stylométrique + extraction de writeprints par cluster	Groupes de styles (clusters) + empreintes par groupe	Regrouper les communications pour inférer "même auteur / même opérateur", choisir les personnes dont le style est facilement imitable	Forensique : regrouper des emails anonymes par style et isoler des campagnes / auteurs probables
Emails signés d'un cadre (CFO/DSI) + nouveaux emails "sensibles"	Mots-outils + n-grams (fort sur messages courts), formules d'ouverture/fermeture, signatures	Authorship **validation** "per-sender" (binaire : authentique vs inauthentique) ; baseline Naive Bayes + Char-CNN	Score de cohérence stylistique + alerte si rupture	Générer des emails BEC "dans le style" pour réduire les soupçons humains	Détection d'usurpation d'identité (BEC) en complément de SPF/DKIM/DMARC et règles métier
Messages Slack/Teams (courts) + métadonnées (heure, canal, destinataires)	N-grams caractères, ponctuation/emoji, longueur, patterns de templates	Classifieur léger + corrélation (style + contexte)	Alerte "style atypique" contextualisée	"Template injection" : produire des messages brefs conformes aux habitudes (ton, format, abréviations)	Signal faible mais utile pour prioriser des investigations quand combiné à d'autres signaux (heure inhabituelle, demande urgente)
Tickets IT / Jira / Confluence (textes moyens)	Mélange lexical + structure (sections, code blocks), n-grams	Attribution multi-classes (N auteurs) ou vérification (1 auteur)	Auteur probable / probabilité	Repérer les rédacteurs "procéduriers" (bons vecteurs d'instructions), copier leur	Détecter un compte compromis qui "écrit différemment", aider au triage SOC

				style pour des demandes “legit-looking”	
Notes de rançon / communiqués d’extorsion	Style global + patterns récurrents + n-grams	Similarité stylométrique / clustering inter-notes	Liaison entre incidents (même “main”)	Réutiliser un style “marque” pour crédibilité/pression	Threat intel : relier des notes à des familles / opérateurs, enrichir la qualification d’incident

3.5 Limites et garde-fous

La stylométrie est moins fiable sur des textes très courts et très bruités (messages Slack de 1–2 phrases), et elle dépend d'un corpus historique suffisant par expéditeur. Les résultats peuvent aussi être perturbés par des changements légitimes (nouveau rôle, délégation d'assistant, traduction, utilisation d'outils de correction), et par des adversaires qui utilisent des LLM pour "lisser" ou "transférer" un style. C'est pourquoi l'approche est la plus robuste quand elle est utilisée comme **signal de risque** (score) corrélé à d'autres signaux (horaire, device, géolocalisation, destinataires inhabituels, demande de paiement, urgence).

3. Axe 3 : Énumération passive et fingerprinting de l'infrastructure technique

3.1 Récupération passive de bannières (Passive Banner Grabbing)

Le troisième axe repose sur une observation non interactive vis-à-vis de la cible (*no direct probing*), s'appuyant sur des observations de télémétrie collectées par des tiers [167]. L'agent exploite exclusivement les API de bases de données de télémétrie Internet massives (Shodan, Censys, Shadowserver) pour récupérer l'historique des réponses réseau sans générer de trafic direct vers l'infrastructure cible [1][23].

Précision terminologique : le terme *passive banner grabbing* est employé ici au sens strict de l'exploitation de données déjà collectées par des moteurs de recherche d'actifs Internet, par opposition au banner grabbing actif qui impliquerait une connexion directe aux services cibles. La couverture, la fraîcheur et les biais d'échantillonnage de ces plateformes tierces constituent des limitations inhérentes à cette approche.

L'analyse porte sur les métadonnées encapsulées dans les couches applicatives et cryptographiques : le balisage des en-têtes HTTP/HTTPS (bannière « Server », structure des ETags), l'analyse des champs des certificats TLS (Subject Alternative Names, émetteurs), et l'extraction des chaînes SNI (Server Name Indication) lorsque disponibles dans les jeux de données [23].

Dans le scénario OpenClaw, l'agent identifie simultanément deux cibles au sein de l'infrastructure MediFrance :

- **L'instance OpenClaw exposée** : détectée via l'empreinte HTML caractéristique du gateway, selon la méthodologie démontrée par Jamieson O'Reilly (Dvuln) [11]. BitSight a observé plus de 30 000 instances exposées entre janvier et février 2026 [8], et SecurityScorecard a confirmé 28 663 adresses IP uniques avec panneaux de contrôle exposés répartis dans 76 pays, dont 12 812 vulnérables à l'exécution de code à distance via CVE-2026-25253 (CVSS 8.8) [7].
- **Le concentrateur VPN Fortinet** : l'identification repose sur un recoupement d'artefacts passifs — cookies de session caractéristiques (SVPNCOOKIE), pages de redirection par défaut d'authentification SSL-VPN (/remote/login), et réponses d'erreurs HTTP liées à FortiOS [77]. *Ces indicateurs présentent un niveau de confiance variable : la présence de SVPNCOOKIE est fortement discriminante, tandis que les pages de redirection peuvent être altérées par des reverse proxies, WAF, ou personnalisations d'interface.*

Incertitudes et limitations du fingerprinting passif

L'inférence à partir de données collectées par des tiers est soumise à plusieurs sources d'erreur qui doivent être explicitées dans tout cadre académique :

- **Bannières masquées ou modifiées** : les équipes de sécurité peuvent supprimer ou falsifier les bannières HTTP, invalidant le fingerprint.
- **Reverse proxies et CDN** : un intermédiaire (Cloudflare, Akamai, Nginx) peut masquer l'identité réelle du service backend.
- **Mutualisation d'adresses IP** : plusieurs services distincts peuvent partager une même IP (hébergement mutualisé, NAT).
- **Données obsolètes** : le décalage temporel entre le scan Shodan/Censys et l'analyse peut être significatif ; un correctif appliqué entre-temps invalide l'inférence.
- **Faux positifs de versioning** : la corrélation version → CVE est conditionnelle à l'identification non ambiguë de la version exacte du firmware, ce qui n'est pas toujours possible à partir des seules métadonnées passives.

3.2 Identification passive de vulnérabilité critique : CVE-2024-55591

Une fois l'empreinte numérique isolée, le système croise le numéro de version avec les registres publics CVE et les flux de Cyber Threat Intelligence [77]. L'inférence de vulnérabilité est formulée comme une décision probabiliste conditionnée par un score de confiance du fingerprint — et non comme une déduction formelle au sens mathématique strict.

Dans le scénario « Opération OpenClaw », l'analyse automatisée de la surface d'attaque révèle l'exposition du portail d'administration FortiOS. L'agent détecte, à partir de la lecture des bannières HTTP et de l'analyse des scripts JavaScript exposés, une version de firmware antérieure aux correctifs de janvier 2025. Cette version est spécifiquement affectée par la **CVE-2024-55591** — *une vulnérabilité réelle, critique, exploitée in the wild, et documentée par plusieurs sources indépendantes* :

Mécanisme technique de la CVE-2024-55591 (CVSS 9.6)

Il s'agit d'une faille logique de type *Authentication Bypass* (CWE-288 — Authentication Bypass Using an Alternate Channel) résidant dans le module WebSocket (jsconsole) de l'interface de gestion FortiOS [77] :

- **Contournement d'authentification** : en manipulant les requêtes WebSocket adressées au module *jsconsole*, un attaquant distant non authentifié force l'API de gestion à délivrer un contexte de session « Super Admin » valide, sans jamais fournir de mot de passe ni de jeton MFA.
- **Furtivité** : l'attaque ne nécessite aucune corruption de mémoire (*pas de buffer overflow*) et laisse peu de traces dans les logs systèmes classiques, car la session générée apparaît comme légitime aux yeux du système.
- **Accès total** : l'accès obtenu est de niveau System/Root, permettant la création immédiate d'utilisateurs persistants ou la désactivation des logs avant toute action offensive majeure.

Cette CVE a été activement exploitée in the wild début 2025, comme documenté par Fortinet PSIRT (FG-IR-24-535), watchTowr Labs, Tenable Research, et l'ANSSI (CERTFR-2025-ALE-002).

Tableau — Éléments observables, règles d'inférence et niveaux de confiance

Élément observable	Règle d'inférence	Confiance	Principales sources d'erreur

Bannière HTTP FortiOS + scripts JS exposés → version firmware < 7.0.17	CVE-2024-55591 : Auth Bypass WebSocket/jsconsole (CVSS 9.6) [77]	Élevée si version identifiée sans ambiguïté	Bannière masquée, version patchée sans mise à jour de bannière, reverse proxy
Cookie SVPNCOOKIE + page /remote/login	Présence confirmée d'un portail SSL-VPN FortiOS	Élevée	Customisation d'interface, honeypot, serveur tiers mimant FortiOS
Empreinte HTML gateway OpenClaw [11]	Instance OpenClaw exposée → CVE-2026-25253 potentielle (CVSS 8.8) [7]	Moyenne	Version patchée, instance sandbox/test, empreinte altérée
Champs SAN/Issuer du certificat TLS	Corrélation organisationnelle (nom entreprise, domaines internes)	Faible à moyenne	Certificats wildcard, CDN, hébergement mutualisé

Références bibliographiques spécifiques à la CVE-2024-55591 :

- Fortinet PSIRT (2025), « FG-IR-24-535: Authentication bypass in Node.js websocket module (CVE-2024-55591) ». FortiGuard Labs Advisory.
- watchTowr Labs (2025), « Pot-Pourri: Fortinet FortiOS Authentication Bypass (CVE-2024-55591) Technical Analysis ».
- Tenable Research (2025), « CVE-2024-55591: Fortinet Authentication Bypass Zero-Day Exploited in the Wild ».
- ANSSI (2025), « CERTFR-2025-ALE-002 : Vulnérabilité critique dans les produits Fortinet ».

4. Modélisation formelle : des Data Flow Diagrams aux Information Inference Diagrams

Les axes précédents produisent des items d'information hétérogènes (topologie de confiance du graphe social, empreintes techniques, inférences de vulnérabilité) et des hypothèses pondérées par des scores de confiance. La corrélation et la dérivation de connaissances nouvelles à partir de ces items constituent le mécanisme central de la reconnaissance assistée par IA — or ce mécanisme n'est pas représenté nativement dans les modèles de menace traditionnels. D'où le besoin d'un formalisme d'inférence.

4.1 Limites du modèle DFD pour la reconnaissance assistée par IA

Les Data Flow Diagrams (DFD) sont largement utilisés comme support de *threat modeling*, notamment dans la méthodologie STRIDE telle que décrite par Microsoft dans le cadre du Security Development Lifecycle [164]. Ils offrent une représentation simple des processus, magasins de données, flux, entités externes et limites de confiance, ce qui en fait un bon point d'entrée pour des ateliers multi-parties prenantes et un outil de communication efficace entre équipes de développement et équipes de sécurité [164].

Toutefois, plusieurs travaux ont mis en évidence des limites expressives lorsque l'analyse nécessite de raisonner de manière plus systématique sur les concepts de sécurité et sur la signification des données. Sion et al. (2020) montrent notamment que, dans les DFD, les données sont fréquemment modélisées de façon ad hoc — souvent réduites à des libellés de flux —, ce qui entraîne des hypothèses hors-modèle et affaiblit la reproductibilité de l'analyse. *Leur critique est multifactorielle* : au-delà du manque de sémantique formelle sur l'information, ils identifient des limites liées à la représentation inadéquate de certains concepts de sécurité, aux niveaux d'abstraction disponibles, et aux informations de déploiement — concluant que les DFD « are not enough » pour une modélisation systématique et reproductible.

Les DFD ne sont pas « inadéquats » au sens absolu — ils restent utiles comme artefact de démarrage et comme support de communication —, mais leur sémantique est insuffisante pour représenter explicitement les items d'information et les chaînes d'inférence qui en découlent.

Dans le contexte des axes précédents, la difficulté est structurelle : un DFD modélise principalement le déplacement des données entre composants, mais ne fournit pas de formalisme natif pour représenter comment des sources disparates (métadonnées OSINT, télémétrie Shodan/Censys, signaux humains du graphe social) se combinent pour produire une connaissance nouvelle — mécanisme central des chaînes d'inférence assistées par IA. C'est précisément ce *gap* que des cadres complémentaires comme les Information Inference Diagrams (I2D) visent à combler, en modélisant la propagation et l'inférence d'information tout en restant compatibles avec des analyses DFD existantes [163].

4.2 Les Information Inference Diagrams (I2D) : complémentarité avec les DFD

Les Information Inference Diagrams (I2D) ont été proposés pour compléter les DFD en modélisant explicitement la propagation et l'inférence d'information [163]. Contrairement aux DFD, les I2D définissent des items d'information, des entités, et des relations de partage et d'inférence, ce qui permet d'analyser comment des informations peuvent être déduites par corrélation de sources hétérogènes [163]. *Les I2D restent compatibles avec les DFD et peuvent être traduits depuis ceux-ci, constituant ainsi une extension et non un remplacement.*

Le formalisme I2D est particulièrement adapté à la modélisation de la phase de reconnaissance car il permet de représenter :

- Les **items d'information** collectés par chaque axe (métadonnées LinkedIn, empreintes Shodan, bannières HTTP, certificats TLS)
- Les **relations d'inférence** entre items (ex. : profil LinkedIn mentionne OpenClaw + empreinte Shodan révèle gateway OpenClaw → confirmation que l'organisation utilise OpenClaw en production)
- La **propagation de confiance** : chaque item et chaque inférence sont associés à un score de confiance qui se propage dans le graphe d'inférences

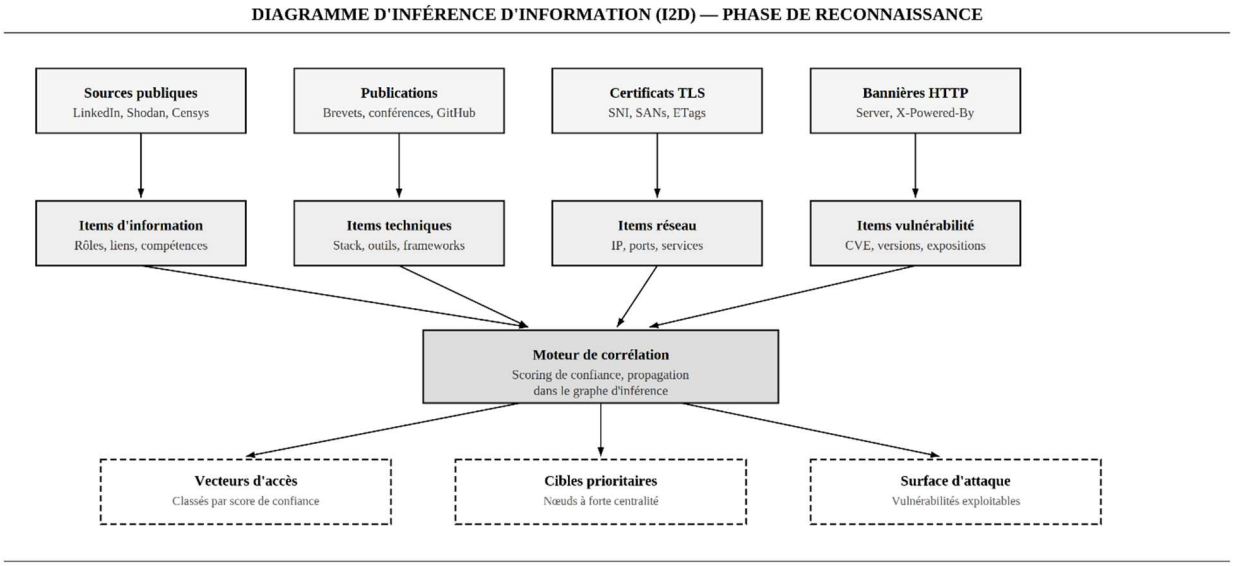


Figure 4. Diagramme d'inférence d'information (I2D) appliqué à la phase de reconnaissance. Les sources publiques alimentent des items d'information typés, corrélés par un moteur de scoring pour produire l'intelligence actionnable : vecteurs d'accès, cibles prioritaires et surface d'attaque. Extension du modèle DFD pour la modélisation OSINT [163].

Précision importante : le fait de combiner les I2D avec des méthodes d'automatisation (LLM, scoring heuristique) est une surcouche applicative possible, mais n'est pas inhérent au formalisme I2D tel que défini dans la littérature [163]. La section suivante décrit cette combinaison comme un scénario prospectif.

4.3 Application au scénario OpenClaw — *modélisation prospective*

Dans le cadre du scénario OpenClaw, le moteur de reconnaissance combine les sorties des deux axes en exploitant la structure I2D. Le processus suit une réduction progressive de l'incertitude :

(1) Ingestion et normalisation — *opérations établies*

Les données OSINT brutes sont collectées, nettoyées et assignées à des classes sémantiques. Les frameworks existants (Maltego, SpiderFoot) effectuent déjà cette normalisation [165].

(2) Traitement par axe — combinaison de techniques établies

- **Axe 1 (Social Graph Mining)** : génère la topologie de confiance du graphe organisationnel G = (V, E), avec identification des nœuds à forte centralité d'intermédiarité [161]
- **Axe 2 (Fingerprinting passif)** : croise les empreintes techniques avec les bases CVE, produisant des inférences de vulnérabilité conditionnées par un score de confiance (cf. section 4.2)

(3) Corrélation I2D — scénario prospectif, techniquement réalisable

C'est à cette étape que le formalisme I2D démontre sa valeur ajoutée par rapport aux DFD. Un exemple formalisé de règle d'inférence I2D :

Règle R1 : SI [Axe 2 : portail FortiOS vulnérable à CVE-2024-55591, confiance ≥ 0.8] ET [Axe 1 : nœud $v_k \in V$ avec $C_B(v_k) > seuil_T$ et v_k mentionne « OpenClaw » dans ses publications] ALORS [Inférence : v_k est un vecteur d'installation prioritaire pour la skill piégée, score = $\min(confiance_{CVE}, C_B(v_k))$]

Règle R2 : SI [Axe 2 : instance OpenClaw exposée sur port non standard, confiance ≥ 0.7] ET [Axe 1 : ≥ 3 nœuds du cluster R&D ont installé des skills depuis ClawHub] ALORS [Inférence : supply chain via ClawHub est le vecteur d'accès initial optimal, score = moyenne(confiance_OpenClaw, densité_cluster)]

Ces règles illustrent le passage d'un raisonnement narratif à un raisonnement formalisable et falsifiable : les variables d'entrée, les seuils et les scores de sortie sont explicités. La gestion des contradictions (données obsolètes, scans datés, faux positifs) s'opère par propagation des scores de confiance — un item source de faible confiance réduit mécaniquement le score des inférences en aval.

Tableau — Comparaison DFD traditionnel vs I2D

Critère	DFD traditionnel	Modèle I2D
Objet principal	Mouvement de données entre composants (processus, magasins, flux) [164]	Propagation d'information, relations de partage et chaînes d'inférence entre items [163]
Sémantique formelle	Limitée : données modélisées comme étiquettes de flux, hypothèses hors-modèle fréquentes	Explicite : items d'information typés, relations d'inférence formalisées, scores de confiance
Application à la reconnaissance (J−30 / J−15)	Identification des ports ouverts et points d'entrée réseau [164]	Raisonnement sur chaînes d'inférence multi-sources reliant signaux humains (graphe social) et techniques (fingerprinting)
Gestion de l'incertitude	Faible : exige des flux déterministes documentés	Gestion explicite via scores de confiance sur items et relations, propagation dans le graphe d'inférences
Compatibilité	Standard largement adopté (STRIDE, outils Microsoft SDL)	Compatible et traduisible depuis les DFD ; extension, non remplacement [163]

5. Synthèse : l'intelligence actionnable à J−15

5.1 Bilan de la reconnaissance

L'extrant final du moteur de corrélation constitue une **Intelligence Actionnable** au seuil de la phase J−15, définie ici comme un ensemble structuré d'hypothèses associées à des scores de confiance, identifiant les vecteurs d'accès initial les plus probables et les cibles humaines prioritaires pour la phase suivante [1][163].

***Dans le scénario OpenClaw*, l'attaquant dispose à J−15 de :**

(a) Un accès potentiel à l'instance OpenClaw exposée de MediFrance, avec ses intégrations Outlook, Slack et terminal — identifié via des bases de données d'actifs Internet tierces (Shodan), sans sondage direct de la cible [11][8] ;

(b) Une cartographie de l'organigramme fonctionnel, des nœuds de confiance à forte centralité d'intermédiarité, et des employés R&D utilisant OpenClaw et installant des skills depuis ClawHub sans validation DSI [14][26] ;

(c) L'inférence, par corrélation de versioning, d'une exposition potentielle du VPN Fortinet à la CVE-2024-55591 (CVSS 9.6) — *corrélation conditionnelle au score de confiance du fingerprint, soumise aux limitations identifiées en section 4.1* [77].

Précision : l'ensemble de la phase de reconnaissance repose sur l'interrogation de bases de données tierces (Shodan, Censys, LinkedIn) et ne génère pas de trafic direct vers l'infrastructure cible, *réduisant significativement la probabilité de détection périmétrique sans toutefois la garantir à zéro*. Des signaux indirects restent possibles : consultations LinkedIn visibles par la cible, corrélation d'IP sur des plateformes tierces, ou alertes sur les scans effectués en amont par les moteurs d'indexation.

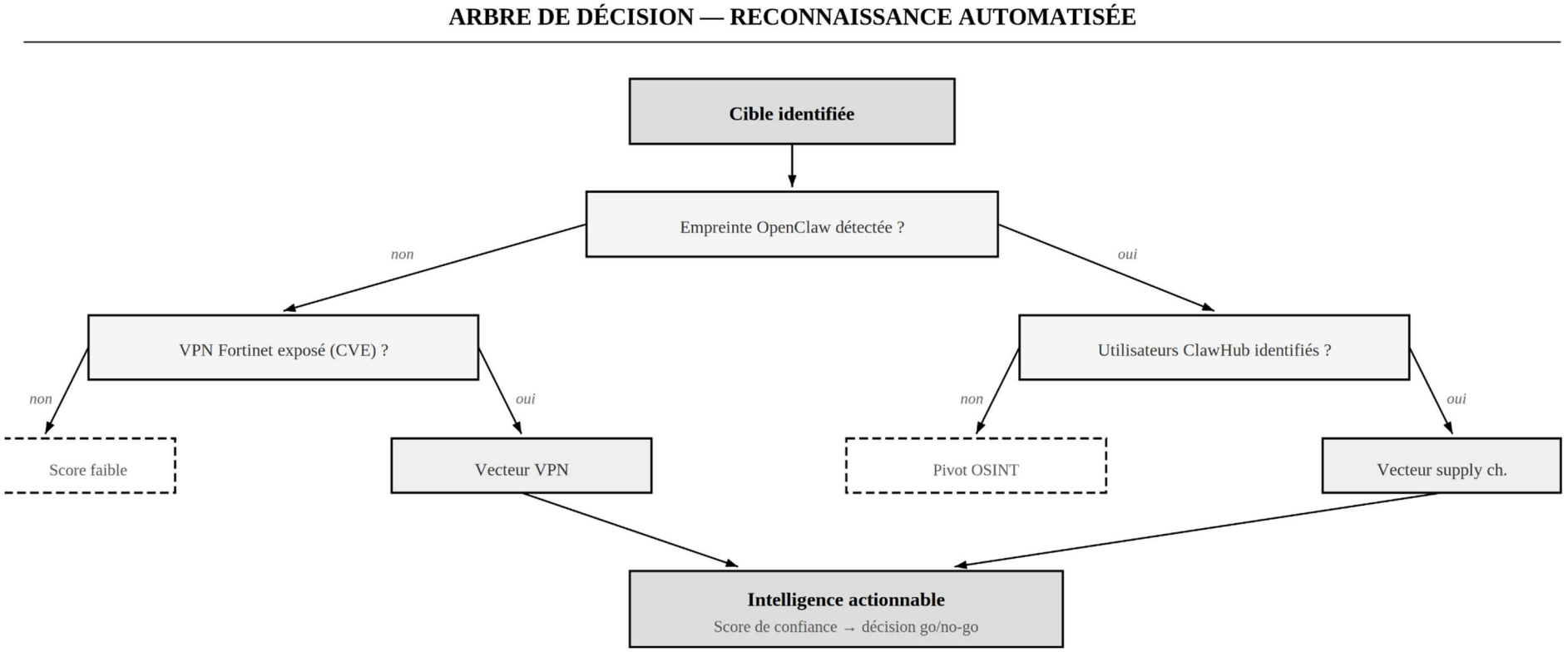


Figure 6. Arbre de décision de la reconnaissance automatisée. L'agent évalue séquentiellement la présence d'empreintes OpenClaw, de vulnérabilités VPN et d'utilisateurs ClawHub pour déterminer les vecteurs d'accès et produire un score de confiance agrégé orientant la décision d'engagement (go/no-go).

5.2 Incertitudes résiduelles et limitations

À J−15, les inférences produites sont soumises à plusieurs sources d'incertitude qui doivent être explicitées :

- **Obsolescence des données** : le décalage temporel entre les scans Shodan/Censys et l'analyse peut invalider les inférences de versioning (correctif appliqué entre-temps).
- **Faux positifs du graphe social** : les connexions LinkedIn ne reflètent pas nécessairement des relations professionnelles actives ; les métriques de centralité peuvent surestimer l'importance de certains nœuds.
- **Ambiguïté du fingerprint** : la corrélation version → CVE est conditionnelle à l'identification non ambiguë de la version firmware, ce qui n'est pas toujours possible à partir des seules métadonnées passives (cf. section 4.1).
- **Score de confiance global** : l'intelligence actionnable doit être interprétée comme un ensemble d'hypothèses pondérées, non comme des certitudes validées. La propagation des scores de confiance dans le graphe I2D (cf. section 5.3) permet de borner ces incertitudes.

5.3 Cartographie MITRE ATT&CK — Phase de reconnaissance

Axe	Technique	Résultat	Taxonomie MITRE	Confiance
Découverte OpenClaw	Interrogation de bases d'actifs Internet (Shodan/Censys) + corrélation empreinte HTML gateway [11]	Instance exposée identifiée, configuration OpenClaw inférée	**T1596** — Search Open Technical Databases	Moyenne à élevée
Axe 1 : Social Graph Mining	OSINT LinkedIn + analyse de centralité (betweenness) [14][161]	Organigramme fonctionnel, nœuds de confiance identifiés, utilisateurs OpenClaw repérés	**T1589** — Gather Victim Identity Information	Moyenne
Axe 2 : Fingerprinting passif	Lecture bannières HTTP/JS + corrélation certificats TLS via données Shodan/Censys [77]	Exposition potentielle VPN Fortinet à CVE-2024-55591 inférée par corrélation de version	**T1592** — Gather Victim Host Information	Variable (cf. tableau section 4.2)

Note sur le mapping MITRE :

- **T1596 (Search Open Technical Databases)** remplace T1595 (Active Scanning) pour la découverte OpenClaw. T1595 implique que l'adversaire sonde directement l'infrastructure cible via du trafic réseau. Dans le scénario, l'attaquant interroge des bases tierces (Shodan/Censys) qui ont préalablement scanné Internet — l'attaquant ne génère pas lui-même le trafic vers la cible [23].
- **T1589** reste cohérent pour l'Axe 1 : la collecte d'informations d'identité (noms, rôles, emails, relations) via des sources ouvertes est précisément le périmètre de cette technique.
- **T1592** reste cohérent pour l'Axe 2 (fingerprinting) : MITRE spécifie explicitement que les informations d'hôtes peuvent provenir de « online or other accessible data sets ».

5.4 Implications stratégiques

L'intégration d'outils d'IA dans des chaînes OSINT déplace l'asymétrie attaquant-défenseur en réduisant le coût de corrélation et de personnalisation, et en permettant une reconnaissance initiale sans sondage direct de la cible, notamment via l'interrogation de bases techniques ouvertes [3][4].

Cette approche ne garantit pas l'absence de détection, mais elle diminue l'exposition aux contrôles périmétriques centrés sur l'inspection du trafic entrant (IDS/IPS/WAF), en déplaçant la collecte vers des sources tierces et des signaux publics [23]. *Des canaux de détection alternatifs restent opérants : monitoring des consultations LinkedIn anormales, alertes d'abus d'API sur les plateformes tierces, corrélation de signaux faibles par les équipes SOC/UEBA, ou flux de Cyber Threat Intelligence signalant des campagnes de fingerprinting massif sur Shodan/Censys.*

La convergence multi-vecteurs (signaux humains issus du graphe social + signaux techniques issus du fingerprinting passif) illustre l'accélération du cycle de reconnaissance et la réduction du coût marginal de

personnalisation des attaques [3][27]. La décision clôturant cette phase — le passage à l'accès initial — est formulée comme une inférence sous incertitude : les scores de confiance propagés dans le graphe I2D (cf. section 5.3) fournissent des bornes d'erreur explicites, et les données possiblement obsolètes sont identifiées comme telles. *Il ne s'agit pas d'une validation déterministe, mais d'une estimation pondérée orientant la stratégie de la phase suivante.*

Ces constats appellent une refonte des postures de cyberdéfense orientée vers le contrôle strict de l'exposition des métadonnées publiques : politique de divulgation minimale sur les réseaux sociaux professionnels, gouvernance des instances d'agents IA exposées sur Internet, et monitoring proactif de l'empreinte numérique de l'organisation sur les bases d'actifs (Shodan, Censys) [9][26].

5.5 Transition vers la Phase 2 (J−15 à J−7)

L'intelligence actionnable produite à J−15 oriente directement la phase d'armement. Les hypothèses à plus forte confiance identifient deux vecteurs d'accès initial complémentaires :

- **Supply chain IA** : la connaissance de l'écosystème OpenClaw de MediFrance (version, marketplace ClawHub, profils d'utilisateurs R&D) indique qu'un artefact malveillant distribué via la marketplace applicative pourrait être installé par des utilisateurs à forte centralité d'intermédiarité, en situation de Shadow AI [26][25].
- **Exploitation de vulnérabilité périmétrique** : l'exposition potentielle du VPN Fortinet à la CVE-2024-55591 (confiance conditionnelle au fingerprint, cf. section 4.2) constitue un vecteur d'accès initial complémentaire exploitable en Phase 3 [77].

La Phase 2 décrira les mécanismes d'armement : conception de l'artefact malveillant pour la supply chain IA et assemblage du ransomware polymorphe PromptLock.

Références

[1] Application of OSINT Methods in Ensuring Cybersecurity, IPSIT Transactions on Internet Research, juillet 2025. https://ipsitransactions.org/journals/papers/tir/2025jul/p5.pdf

[2] IBM Redbooks, Network Intrusion Prevention Design Guide. https://www.redbooks.ibm.com/redbooks/pdfs/sg247979.pdf

[3] Rapid7, « How LLMs Like WormGPT Are Reshaping Cybercrime in 2025 ». https://www.rapid7.com/blog/post/ai-goes-on-offense-how-llms-are-redefining-the-cybercrime-landscape/

[4] A Comprehensive Review of Large Language Models and AI in Cybersecurity: Applications in Threat Detection, Defense, and Software Security, Preprints.org, juillet 2025. https://www.preprints.org/manuscript/202507.1159

[5] Palo Alto Networks Unit 42, « The Dual-Use Dilemma of AI: Malicious LLMs » (WormGPT 4, KawaiiGPT), novembre 2025. https://unit42.paloaltonetworks.com/dilemma-of-ai-malicious-llms/

[6] CATO Networks, « WormGPT returns: New malicious AI variants built on Grok and Mixtral », juin 2025. https://www.csoonline.com/article/4008912

[7] Techzine, « Over 40,000 OpenClaw agents vulnerable » (SecurityScorecard), février 2026. https://www.techzine.eu/news/security/138633/

[8] BitSight, « OpenClaw Security: Risks of Exposed AI Agents Explained », février 2026. https://www.bitsight.com/blog/openclaw-ai-security-risks-exposed-instances

[9] Cisco AI Threat & Security Research, « Personal AI Agents like OpenClaw Are a Security Nightmare », janvier 2026. https://blogs.cisco.com/ai/personal-ai-agents-like-openclaw-are-a-security-nightmare

[10] Sophos, « The OpenClaw experiment is a warning shot for enterprise AI security », février 2026. https://www.sophos.com/en-us/blog/the-openclaw-experiment-is-a-warning-shot-for-enterprise-ai-security

[11] SecurityWeek, « Vulnerability Allows Hackers to Hijack OpenClaw AI Assistant » (CVE-2026-25253, J. O'Reilly/Dvuln), février 2026. https://www.securityweek.com/vulnerability-allows-hackers-to-hijack-openclaw-ai-assistant/

[12] Barrack.ai, « OpenClaw is a Security Nightmare — Here's the Safe Way to Run It », février 2026. https://blog.barrack.ai/openclaw-security-vulnerabilities-2026/

[13] Hudson Rock, « Infostealer Steals OpenClaw AI Agent Configuration Files and Gateway Tokens », via The Hacker News, février 2026.

[14] ResearchGate, « Linking Users Across Domains with Location Data: Theory and Validation », 2017. https://www.researchgate.net/publication/312638398

[15] ResearchGate, « Interactive Graph Learning for Multilevel Network Alignment », 2024. https://www.researchgate.net/publication/394474660

[16] ResearchGate, « CoLink: An Unsupervised Framework for User Identity Linkage », 2022. https://www.researchgate.net/publication/361545118

[17] Datopian, « AI-driven Metadata Enrichment in Open Data Portals: A Deep Dive ». https://www.datopian.com/blog/ai-driven-metadata-enrichment-in-open-data-portals-a-deep-dive

[18] ResearchGate, « Mapping Users across Networks by Manifold Alignment on Hypergraph », 2015. https://www.researchgate.net/publication/286154028

[19] Virginia Tech Crowd Intelligence Lab, « Co-designing AI-Augmented Collaborative OSINT Investigations for Vulnerability Assessment », CHI 2025. https://crowd.cs.vt.edu/wp-content/uploads/2025/03/chi25b-sub3884-cam-i16.pdf

[20] ResearchGate, « Defending Against Social Engineering Attacks in the Age of LLMs », 2024. https://www.researchgate.net/publication/386187439

[21] A Comprehensive Survey on AI in Counter-Terrorism and Cybersecurity, IEEE Access, 2025. https://ieeexplore.ieee.org/iel8/6287639/10820123/11008653.pdf

[22] Journal of Language and Education, « Detecting LLM-Generated Text with Trigram–Cosine Stylometric Delta ». https://jle.hse.ru/article/view/22211

[23] arXiv, « Stylometry recognizes human and LLM-generated texts in short samples », 2025. https://arxiv.org/abs/2507.00838

[24] StrongestLayer, « AI-Generated Phishing: The Top Enterprise Threat of 2026 » (réf. IBM Research, Harvard). https://www.strongestlayer.com/blog/ai-generated-phishing-enterprise-threat

[25] MITRE ATT&CK, « Active Scanning: Vulnerability Scanning », Sub-technique T1595.002. https://attack.mitre.org/techniques/T1595/002/

[26] Recorded Future, « What is Banner Grabbing? Tools and Techniques Explained ». https://www.recordedfuture.com/threat-intelligence-101/tools-and-techniques/banner-grabbing

[27] The Shadowserver Foundation, « CRITICAL: Vulnerable HTTP Report ». https://www.shadowserver.org/what-we-do/network-reporting/vulnerable-http-report/

[28] Fortinet, « FortiOS and SSL Vulnerabilities ». https://www.fortinet.com/blog/psirt-blogs/fortios-ssl-vulnerability

[29] Rapid7, « Fortinet Firewalls Hit with New Zero-Day Attack, Older Data Leak », janvier 2025. https://www.rapid7.com/blog/post/2025/01/16/etr-fortinet-firewalls-hit-with-new-zero-day-attack/

[30] Darktrace, « From Exploit to Escalation: Tracking and Containing a Real-World Fortinet SSL-VPN Attack ». https://www.darktrace.com/blog/from-exploit-to-escalation-tracking-and-containing-a-real-world-fortinet-ssl-vpn-attack

[31] arXiv, « Information Inference Diagrams: Complementing Privacy and Security Analyses Beyond Data Flows », 2024. https://arxiv.org/html/2405.08356v2

[32] ThreatModeler, « Process Flow Diagrams (PFDs) vs. Data Flow Diagrams (DFDs) in the Modern Threat Modeling Arena ». https://threatmodeler.com/resource/white-papers/process-flow-diagrams-vs-data-flow-diagrams/

[33] ReconSphere: Real-Time AI-Powered OSINT & Facial Recognition Tool, Lingaya's Vidyapeeth, IJISIE. https://www.lingayasvidyapeeth.edu.in/IJISIE/papers/vol1_1/2.pdf

[34] VentureBeat, « OpenClaw proves agentic AI works. It also proves your security model doesn't », février 2026. https://venturebeat.com/security/openclaw-agentic-ai-security-risk-ciso-guide

[35] OWASP, « LLM01:2025 Prompt Injection », Top 10 for LLM Applications 2025. https://genai.owasp.org/llmrisk/llm01-prompt-injection/

[36] CrowdStrike, « Indirect Prompt Injection Attacks: Hidden AI Risks », décembre 2025. https://www.crowdstrike.com/en-us/blog/indirect-prompt-injection-attacks-hidden-ai-risks/

[37] MDPI, « Prompt Injection Attacks in Large Language Models and AI Agent Systems: A Comprehensive Review », Information 17(1):54, janvier 2026. https://www.mdpi.com/2078-2489/17/1/54

[38] MDPI, « The Erosion of Cybersecurity Zero-Trust Principles Through Generative AI: A Survey », 2024. https://www.mdpi.com/2624-800X/5/4/87

[39] Menlo Security, « Predictions for 2026: Why AI Agents Are the New Insider Threat », janvier 2026. https://www.menlosecurity.com/blog/predictions-for-2026

[40] IEEE Xplore, « Cyber Attack Prediction: From Traditional Machine Learning to Generative Artificial Intelligence », 2025. https://ieeexplore.ieee.org/iel8/6287639/6514899/10909100.pdf

Note : *ces références ci-après sont définies dans la bibliographie d'une autre phase du document. Elles sont reproduites ici pour permettre une lecture autonome de chaque phase.*

[77] MITRE ATT&CK, « Groups — APT Techniques for Initial Access and Persistence », v15. https://attack.mitre.org/groups/

→ *Définie en Phase 3*

[120] C. Schneider (2026), Promptware Kill Chain. OWASP Top 10 for Agentic Applications 2026, ASI01 Agent Goal Hijack. https://christian-schneider.net/blog/prompt-injection-agentic-amplification/

→ *Définie en Phase 4*

[121] InstaTunnel, « Prompt-to-Insider Threat: When AI Agents Become Double Agents ». CVE-2025-32711 EchoLeak (M365 Copilot, CVSS 9.3), février 2026. https://instatunnel.my/blog/prompt-to-insider-threat/

→ *Définie en Phase 4*

[123] Mithril Security, « PoisonGPT: How to poison LLM supply chain on Hugging Face » (ROME, GPT-J-6B, Δ 0,1 %). https://blog.mithrilsecurity.io/poisongpt/

→ *Définie en Phase 4*

[147] Securin, « 2025 Ransomware Report » (7 061 victimes, 117 groupes, IA = accélérateur, chatbots extorsion). 17 février 2026. https://www.prnewswire.com/news-releases/securin-2025-ransomware-report-302688125.html

→ *Définie en Phase 5*

[160] OWASP, Top 10 for Agentic Applications 2026. Sandboxing agentique, moindre privilège, audit trafic sortant.

→ *Définie en Phase 5*