AYI NEDJIMI Consultants

Reinforcement Learning Appliqué à la Cybersécurité

Catégorie : Intelligence Artificielle Lecture : 22 min Publié le : 15/02/2026 Auteur : Ayi NEDJIMI

RL pour la génération de stratégies d'attaque (CyberBattleSim) et l'optimisation de défenses adaptatives. Thèmes : reinforcement learning.

Cette analyse technique de Reinforcement Learning Appliqué à la Cybersécurité s'appuie sur les retours d'experience d'equipes confrontees quotidiennement aux defis operationnels du domaine. Les methodologies presentees couvrent l'ensemble du cycle de vie, de la conception initiale au deploiement en production, en passant par les phases de test et de validation. Les recommandations sont directement applicables dans les environnements professionnels. RL pour la génération de stratégies d'attaque (CyberBattleSim) et l'optimisation de défenses adaptatives. Thèmes : reinforcement learning. Dans un contexte où l'intelligence artificielle transforme les pratiques de cybersécurité, la maîtrise de ia reinforcement learning cybersecurite devient un avantage stratégique pour les équipes techniques. Nous abordons notamment : table des matières, 1 introduction au reinforcement learning et 2 rl offensif : cyberbattlesim, caldera et génération d'attaques. Les professionnels y trouveront des recommandations actionnables, des commandes prêtes à l'emploi et des stratégies de mise en œuvre adaptées aux environnements d'entreprise.

Table des Matières

1 Introduction au Reinforcement Learning

Le **Reinforcement Learning (RL)**, ou apprentissage par renforcement, constitue le troisième pilier fondamental du machine learning aux côtés de l'apprentissage supervisé et non supervisé. Contrairement à ces deux références qui reposent sur des jeux de données statiques, le RL place un **agent** dans un **environnement** dynamique où il apprend par essais et erreurs à maximiser une récompense cumulative. L'agent observe un état, exécute une action, reçoit une récompense (positive ou négative) et transite vers un nouvel état. Ce cycle fondamental, formalisé par le cadre mathématique des **processus de décision markoviens (MDP)**, confère au RL une capacité unique : apprendre des stratégies optimales dans des environnements complexes, non-stationnaires et adversariaux -- précisément les caractéristiques du domaine de la cybersécurité.

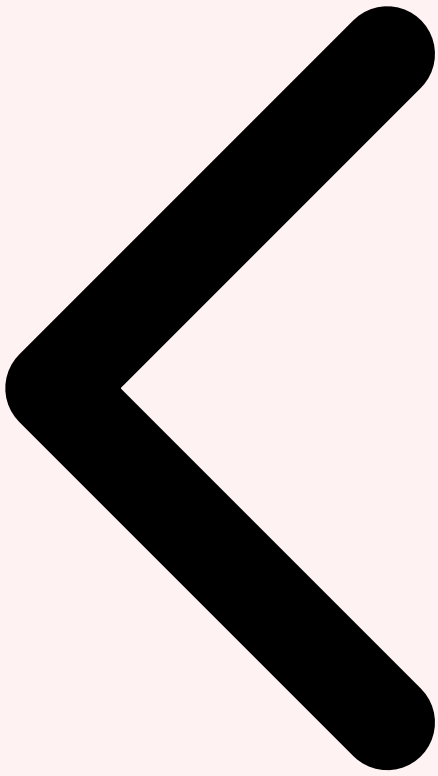
Un MDP est défini par un quintuplet (S, A, P, R, gamma) où S représente l'espace des états, A l'espace des actions, P la fonction de transition (probabilité de passer d'un état à un autre étant donné une action), R la fonction de récompense et gamma le facteur d'actualisation qui pondère l'importance des récompenses futures par rapport aux récompenses immédiates. L'objectif de l'agent est de trouver une **politique optimale** pi* qui maximise l'espérance de la somme des récompenses actualisées. En cybersécurité, l'espace des états peut représenter la topologie réseau et l'état de compromission des machines, l'espace des actions correspond aux techniques d'attaque ou de défense disponibles, et la récompense encode l'objectif opérationnel (par exemple, atteindre un actif critique pour l'attaquant ou minimiser le temps de détection pour le défenseur).

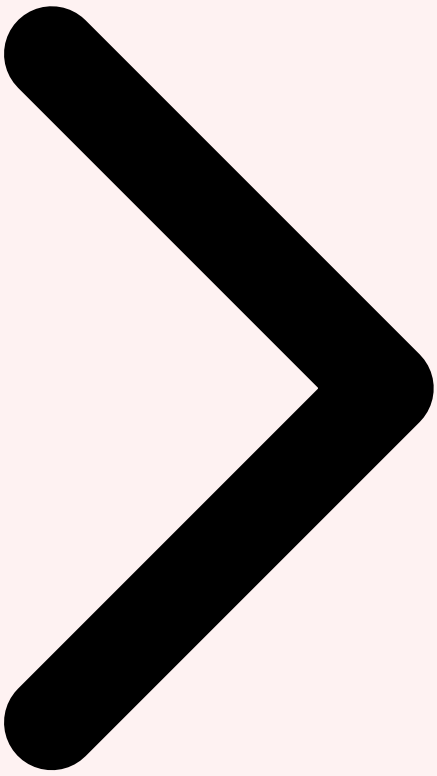
Les algorithmes de RL modernes se divisent en plusieurs familles. Les méthodes **value-based** comme **Deep Q-Network (DQN)** apprennent une fonction de valeur Q(s,a) qui estime la récompense cumulative attendue pour chaque paire état-action. Les méthodes **policy-gradient** comme **REINFORCE** et **Proximal Policy Optimization (PPO)** optimisent directement la politique de l'agent. Les approches **actor-critic** combinent les deux schémas : un acteur qui propose des actions et un critique qui évalue leur qualité. En 2026, PPO reste l'algorithme de référence pour la majorité des applications cyber grâce à sa stabilité d'entraînement et sa capacité à gérer des espaces d'actions discrets et continus.

Vos pipelines de données d'entraînement sont-ils protégés contre l'empoisonnement ?

Concept fondamental : Le Reinforcement Learning appliqué à la cybersécurité permet à des agents autonomes d'apprendre des **stratégies d'attaque et de défense optimales** en interagissant avec des environnements de simulation réseau. L'agent explore l'espace des possibles, découvre des chaînes d'exploitation inédites et développe des politiques de réponse adaptatives impossibles à concevoir manuellement.

L'application du RL à la cybersécurité n'est pas un exercice académique abstrait. En 2026, plusieurs facteurs convergent pour rendre cette approche opérationnellement viable. D'abord, la complexité croissante des infrastructures -- multi-cloud, conteneurs, IoT, OT -- rend les stratégies de défense statiques insuffisantes. Ensuite, les attaquants utilisent déjà des techniques automatisées et adaptatives. Enfin, les environnements de simulation comme CyberBattleSim de Microsoft et CybORG de l'Australian Defence Science and Technology Group fournissent des terrains d'entraînement réalistes où les agents RL peuvent accumuler des millions d'épisodes d'expérience sans risquer de compromettre de véritables systèmes. Cet article explore en profondeur les dimensions offensive, défensive et multi-agent du RL appliqué à la cybersécurité, avec des implémentations concrètes et des retours d'expérience terrain.



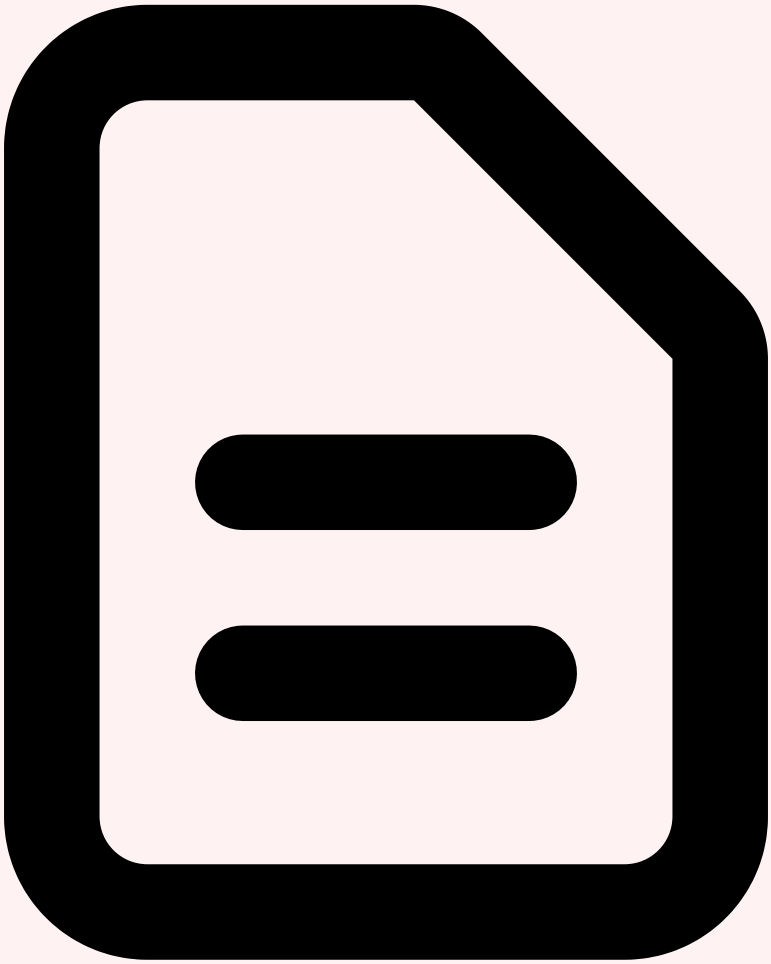


Critere	Description	Niveau de risque
Confidentialite	Protection des donnees d'entrainement et des prompts	Eleve
Integrite	Fiabilite des sorties et detection des hallucinations	Critique
Disponibilite	Resilience du service et gestion de la charge	Moyen
Conformite	Respect du RGPD, AI Act et politiques internes	Eleve

2 RL offensif : CyberBattleSim, CALDERA et génération d'attaques

L'application offensive du Reinforcement Learning vise à développer des agents autonomes capables de **découvrir et exploiter des chemins d'attaque** dans un réseau cible. Contrairement aux scanners de vulnérabilités traditionnels qui opèrent de manière linéaire et déterministe, un agent RL explore l'environnement de manière stratégique, apprend à enchaîner des techniques d'exploitation de manière séquentielle et adapte

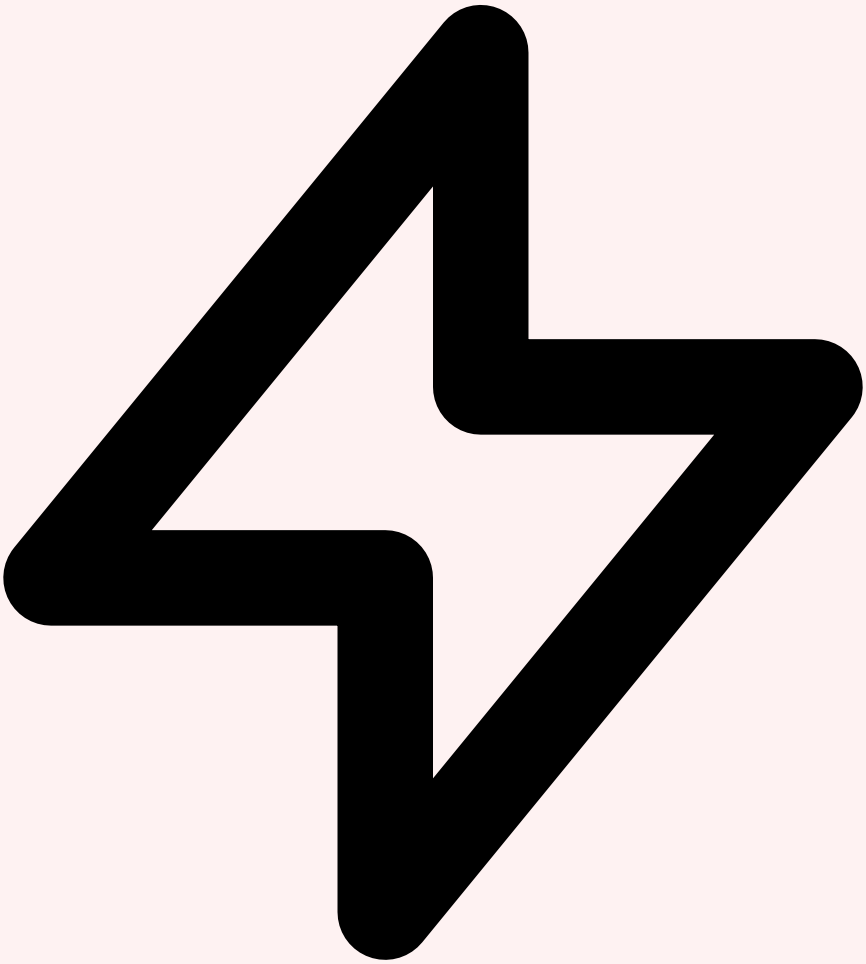
dynamiquement sa stratégie en fonction des réponses du réseau. Cette capacité est fondamentale pour le **red teaming automatisé** et l'évaluation continue de la posture de sécurité.



CyberBattleSim : l'environnement de référence Microsoft

CyberBattleSim, publié par Microsoft Research en 2021 et activement maintenu jusqu'en 2026, est un environnement de simulation de type Gym (OpenAI) conçu spécifiquement pour entraîner des agents RL à effectuer du mouvement latéral post-compromission. L'environnement modélise un réseau comme un graphe où chaque noeud représente une machine avec des propriétés spécifiques : services actifs, vulnérabilités, identifiants stockés, niveau de privilège. L'agent RL démarre avec un accès initial sur un noeud compromis et doit découvrir et exploiter les vulnérabilités des machines adjacentes pour progresser dans le réseau. Les actions disponibles incluent la découverte de services, l'exploitation de vulnérabilités locales et distantes, et l'utilisation d'identifiants volés pour du mouvement latéral.

L'architecture de CyberBattleSim repose sur un espace d'observation riche qui encode la connaissance partielle de l'agent sur le réseau (modèle de type **POMDP** -- Partially Observable Markov Decision Process). L'agent ne voit que les machines qu'il a déjà découvertes et les informations qu'il a collectées. La fonction de récompense favorise la compromission de nouvelles machines et pénalise les actions échouées ou détectées. Les chercheurs de Microsoft ont démontré que des agents entraînés avec DQN et PPO peuvent découvrir des chemins d'attaque complexes impliquant des chaînes de pivoting à travers plusieurs segments réseau, souvent plus rapidement et plus exhaustivement qu'un pentester humain dans les mêmes conditions.



CALDERA et l'intégration RL avec MITRE ATT&CK

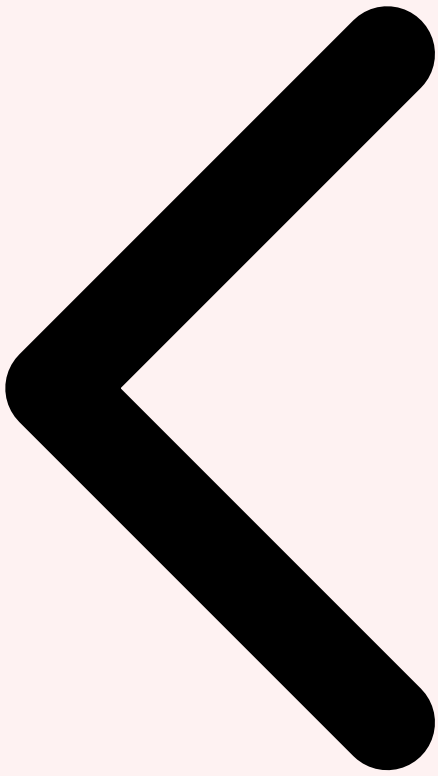
CALDERA, développé par MITRE, est une plateforme de red teaming automatisé qui orchestre des actions offensives selon le framework ATT&CK. L'intégration du RL dans CALDERA représente une avancée significative : plutôt que de suivre des plans d'attaque prédéfinis (les *adversary profiles*), un agent RL peut **sélectionner dynamiquement les techniques ATT&CK** les plus appropriées en fonction de l'état courant de la campagne.

L'espace d'actions de l'agent est directement mappé sur les techniques et sous-techniques du framework ATT&CK, ce qui garantit la pertinence opérationnelle des stratégies découvertes.

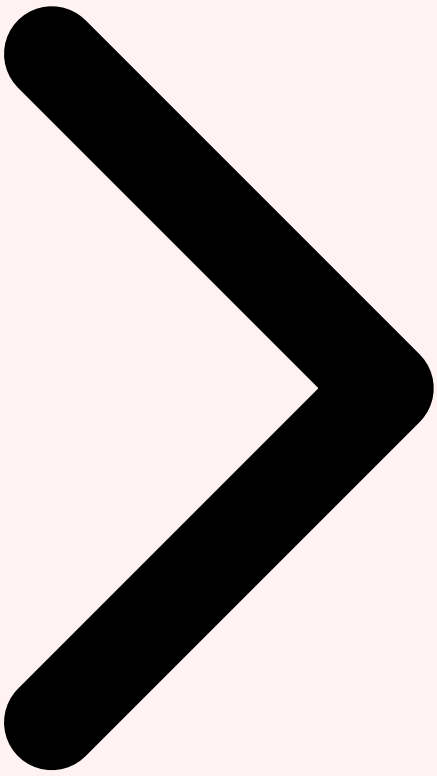
Cas concret
En 2024, des chercheurs de Cornell ont publié une étude démontrant l'empoisonnement de données d'entraînement de modèles de vision par ordinateur avec seulement 0.01% d'images malveillantes, suffisant pour créer des backdoors indétectables par les méthodes de validation standard.

Les travaux récents sur l'intégration RL-CALDERA utilisent des architectures **Hierarchical RL** (HRL) pour gérer la complexité de l'espace d'actions. Un agent de haut niveau sélectionne les tactiques ATT&CK (reconnaissance, exécution, persistence, mouvement latéral, exfiltration), tandis que des agents de bas niveau choisissent les techniques spécifiques au sein de chaque tactique. Cette décomposition hiérarchique améliore considérablement la vitesse de convergence et la qualité des stratégies découvertes. Les résultats expérimentaux montrent que les agents HRL-CALDERA identifient en moyenne 40% plus de chemins d'attaque viables que les profils adversaires statiques, tout en réduisant de 60% le nombre d'actions bruyantes susceptibles de déclencher des alertes. Pour approfondir, consultez Sparse Autoencoders et Interprétabilité Mécanistique.

L'une des contributions majeures du RL offensif est la capacité à **modéliser l'évasion**. Un agent RL peut apprendre à minimiser son empreinte de détection tout en progressant vers son objectif. En encodant dans la fonction de récompense une pénalité proportionnelle à la probabilité de détection de chaque action (estimée à partir des règles SIEM et des signatures IDS du réseau cible), l'agent développe des stratégies furtives poussées. Il peut par exemple apprendre à préférer les techniques living-off-the-land (LOLBins) aux outils offensifs connus, ou à temporiser ses actions pour éviter de déclencher des corrélations temporelles dans le SIEM.



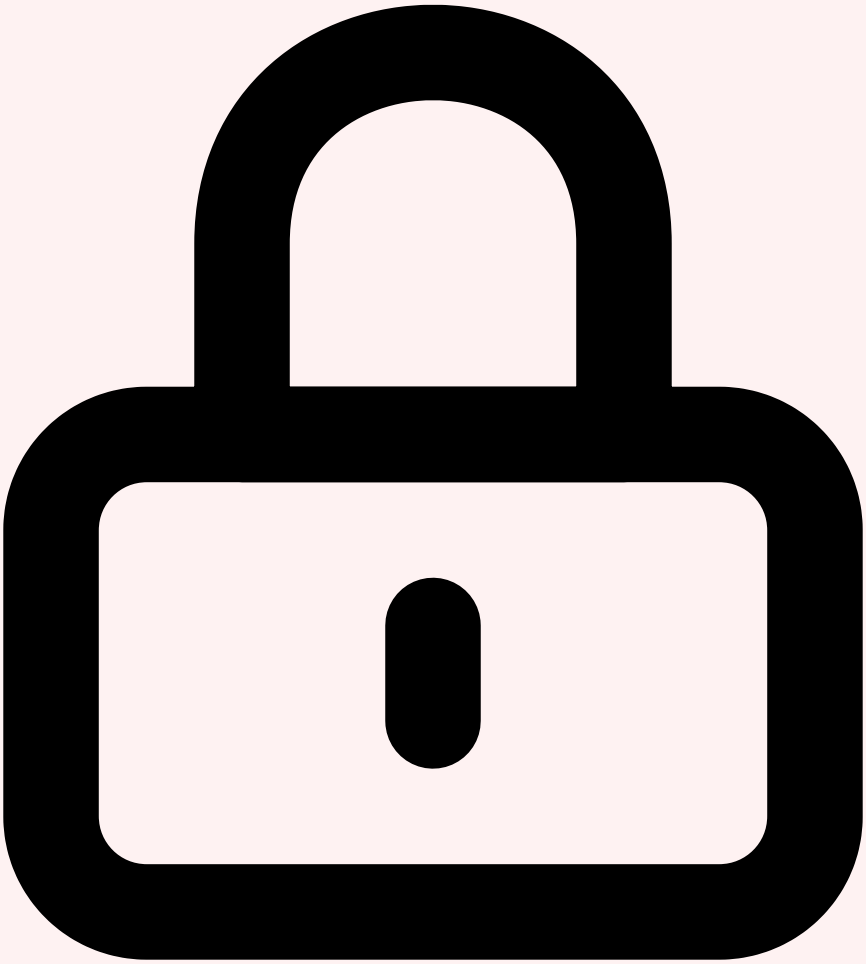
Introduction au RL RL Offensif RL Défensif



Votre organisation est-elle prête à faire face aux attaques basées sur l'IA ?

3 RL défensif : firewalls adaptatifs et IDS intelligents

Si le RL offensif automatise l'attaque, le **RL défensif** transforme fondamentalement la manière dont les systèmes de sécurité répondent aux menaces. Les défenses traditionnelles -- firewalls à règles statiques, IDS à signatures, SIEM à corrélation déterministe -- partagent une faiblesse commune : elles sont réactives et prévisibles. Un attaquant suffisamment déterminé peut étudier et contourner ces mécanismes. Le RL défensif introduit une dimension **adaptative et non-déterministe** dans les défenses, rendant le comportement du système de protection lui-même difficile à prédire et à contourner.

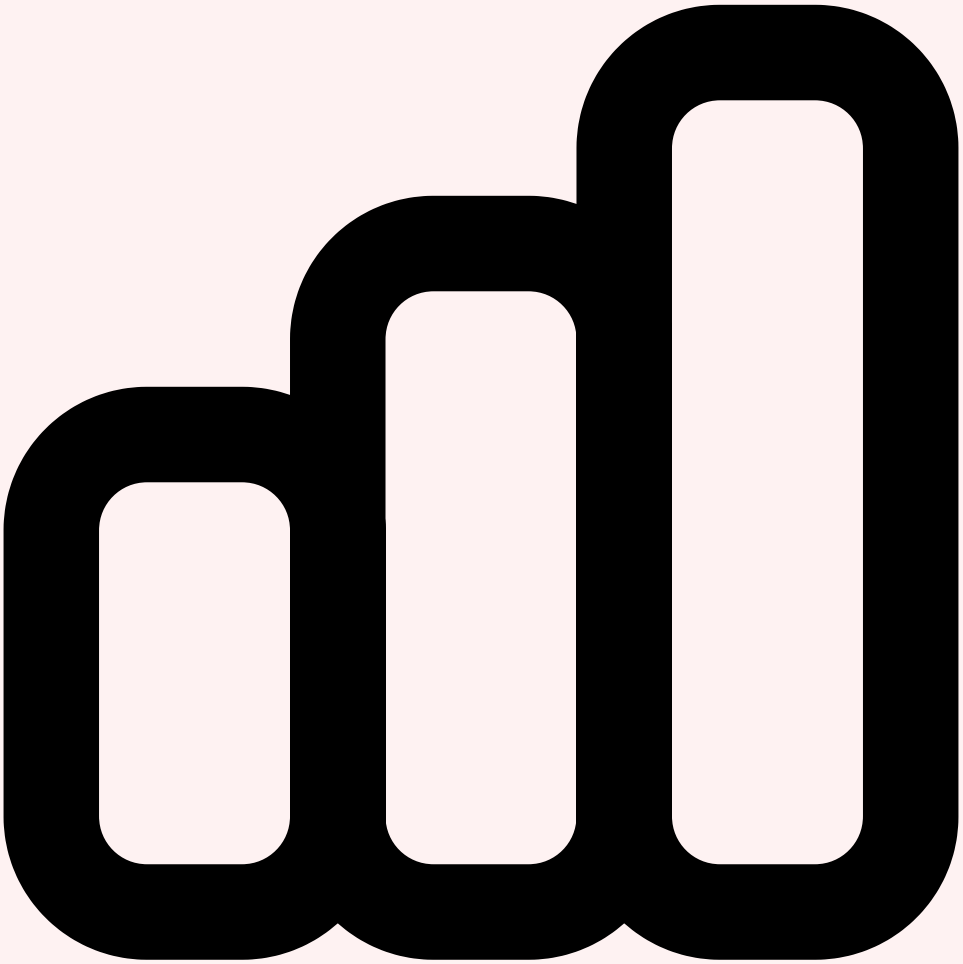


Firewalls adaptatifs par RL

Un **firewall adaptatif piloté par RL** remplace les règles statiques de filtrage par une politique apprise qui ajuste dynamiquement les décisions d'autorisation et de blocage en fonction du contexte temps réel. L'état observé par l'agent inclut les métriques de trafic réseau (débit, distribution des ports, entropie des adresses source), les indicateurs de compromission (IoC) actifs, l'historique récent des alertes et la charge des serveurs protégés. Les actions disponibles couvrent un spectre allant du simple blocage d'IP au rate limiting sélectif, en passant par la redirection vers des honeypots, l'activation de règles de DPI (Deep Packet Inspection) ciblées ou le déclenchement de micro-segmentation dynamique.

La fonction de récompense d'un firewall RL encode un compromis fondamental en cybersécurité : **maximiser la détection des menaces tout en minimisant les faux positifs**. Un blocage correct d'une tentative d'intrusion génère une récompense positive élevée. Un faux positif (blocage d'un utilisateur légitime) produit une pénalité proportionnelle à l'impact métier. Un faux négatif (laisser passer une attaque) entraîne une forte pénalité différée lorsque la compromission est détectée ultérieurement. Ce cadre de

récompense multi-objectif pousse l'agent à développer des politiques nuancées qui adaptent leur agressivité en fonction du niveau de menace perçu -- une capacité impossible à obtenir avec des règles déterministes.



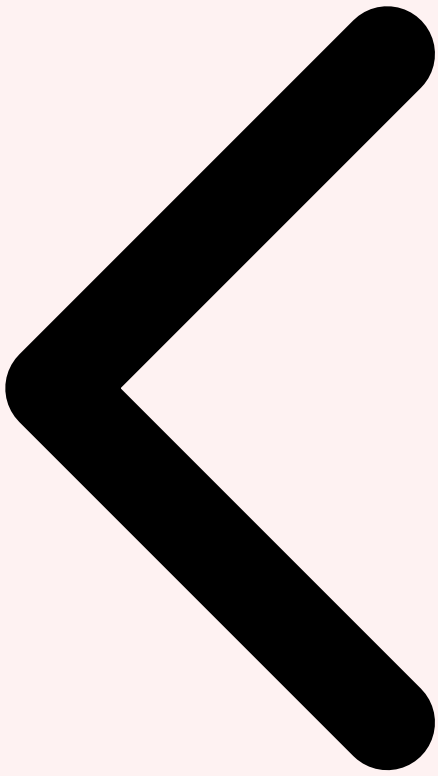
IDS adaptatifs et détection d'anomalies

Les **systèmes de détection d'intrusion (IDS) pilotés par RL** dépassent les limites des approches classiques à signatures et à seuils statistiques. Un agent RL-IDS apprend à ajuster dynamiquement ses seuils de détection, à sélectionner les features les plus discriminantes en fonction du contexte et à orchestrer des réponses graduées. Par exemple, face à un scan de ports lent (slow scan), un IDS traditionnel peut ne pas déclencher d'alerte car chaque sonde individuelle reste sous le seuil. Un agent RL entraîné apprend à corréler ces micro-événements sur des fenêtres temporelles variables et à ajuster dynamiquement sa sensibilité.

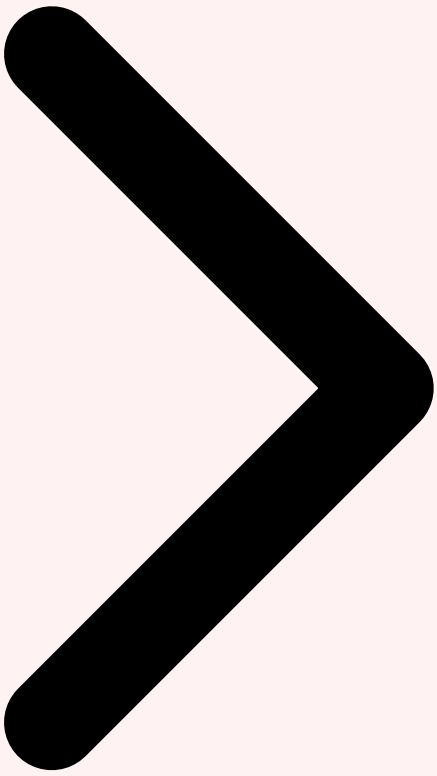
Les architectures les plus avancées combinent des **réseaux récurrents (LSTM, GRU)** pour l'encodage de l'état temporel du trafic avec des algorithmes policy-gradient comme PPO ou SAC (Soft Actor-Critic). L'agent observe des séquences de features réseau et apprend à reconnaître des patterns d'attaque complexes et multi-étapes. Les recherches de

2025-2026 montrent que les IDS-RL surpassent les IDS à machine learning supervisé classique sur les attaques zero-day et les variantes d'attaques connues, avec un gain moyen de 15 à 25 points de F1-score sur les datasets NSL-KDD et CICIDS2017 pour les catégories d'attaques les moins représentées.

Un aspect particulièrement prometteur est l'**apprentissage de la réponse à incident automatisée**. Au-delà de la simple détection, un agent RL défensif peut apprendre à orchestrer des actions de réponse : isoler un segment réseau, révoquer des credentials compromises, déclencher une capture réseau forensique, escalader vers un analyste SOC. L'agent apprend la politique de réponse optimale qui minimise le temps moyen de contention (MTTC) tout en limitant l'impact opérationnel des actions de réponse. Cette approche transforme le modèle du SOC en passant d'une réponse manuelle guidée par des playbooks à une réponse adaptative autonome supervisée par des humains.



RL Offensif RL Défensif Multi-Agent RL



4 Multi-Agent Reinforcement Learning (MARL)

La cybersécurité est fondamentalement un jeu **adversarial** entre attaquants et défenseurs. Le **Multi-Agent Reinforcement Learning (MARL)** formalise cette dynamique en entraînant simultanément des agents offensifs et défensifs qui co-évoluent. Cette approche, inspirée de la théorie des jeux et du concept d'équilibre de Nash, produit des agents significativement plus robustes que ceux entraînés dans des environnements statiques. L'agent attaquant apprend à contourner les défenses adaptatives, tandis que l'agent défenseur apprend à anticiper et neutraliser des stratégies d'attaque avancées -- une course aux armements virtuelle qui renforce les deux parties.

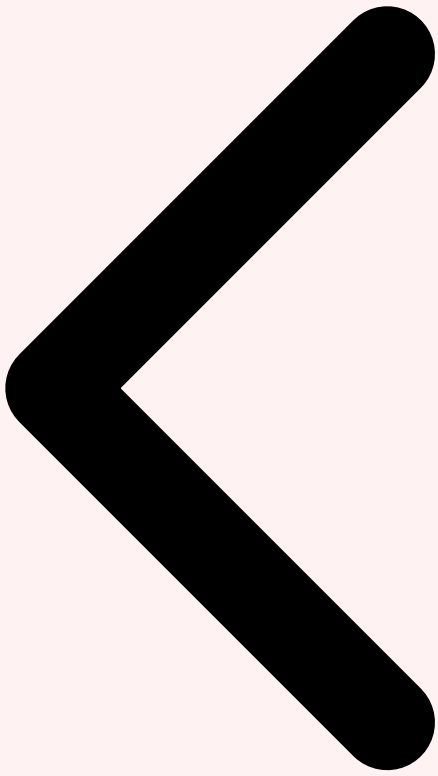
Le cadre MARL le plus utilisé en cybersécurité est le **jeu de Markov à somme nulle** (zero-sum Markov game), où le gain de l'attaquant est la perte du défenseur et vice versa. Formellement, cela se traduit par deux agents partageant le même espace d'états S mais disposant d'espaces d'actions distincts (A_atk pour l'attaquant, A_def pour le défenseur) et de fonctions de récompense opposées (R_atk = -R_def). La dynamique de transition P(s'|s, a_atk, a_def) dépend des actions simultanées des deux agents, créant une

interdépendance stratégique complexe. Les algorithmes de résolution comme **MADDPG** (Multi-Agent Deep Deterministic Policy Gradient), **MAPPO** (Multi-Agent PPO) et **QMIX** étendent les algorithmes single-agent pour gérer cette dynamique multi-agent.

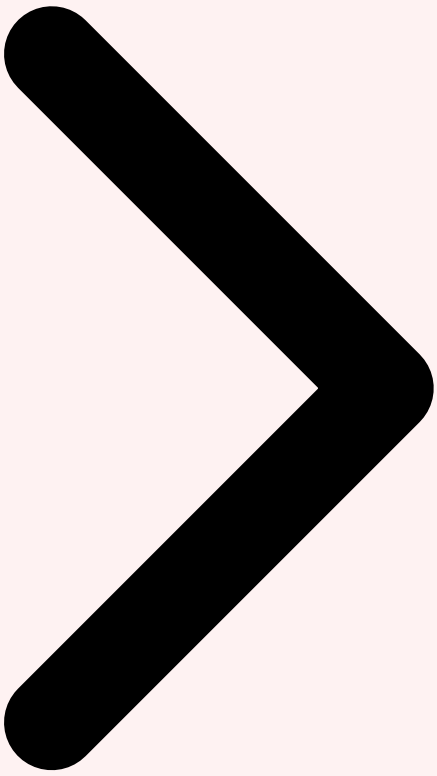
L'un des résultats les plus significatifs de la recherche MARL en cybersécurité concerne le phénomène d'**émergence de stratégies**. Lorsque deux agents co-évoluent sur un nombre suffisant d'épisodes (typiquement des millions), ils développent spontanément des comportements élaborés non programmés. L'agent attaquant apprend des techniques de diversion (créer du bruit sur un segment réseau pour masquer une progression latérale sur un autre), des stratégies de timing (exploiter les fenêtres temporelles de rotation des logs ou de mise à jour des signatures), et des techniques d'anti-forensics (nettoyer ses traces après compromission). L'agent défenseur développe des contre-mesures comme le déploiement dynamique de honeypots, la variation aléatoire des configurations réseau (moving target defense) et les stratégies de deception. Pour approfondir, consultez MCP (Model Context Protocol) : Connecter les LLM à vos.

Point clé MARL : L'entraînement adversarial multi-agent produit des politiques de défense **robustes par construction**. Un défenseur entraîné contre un attaquant RL qui adapte ses stratégies est intrinsèquement plus résilient qu'un défenseur entraîné contre des attaques scriptées. Le MARL constitue donc le approche le plus prometteur pour la conception de défenses adaptatives de nouvelle génération.

Le **self-play**, technique popularisée par AlphaGo et AlphaZero, constitue une variante puissante du MARL où un agent joue contre des versions antérieures de lui-même. En cybersécurité, un agent de red team entraîné par self-play explore progressivement des stratégies d'attaque de plus en plus complexes à mesure que sa version défensive s'améliore. Cette approche évite le problème du *curriculum collapse* où un agent se spécialise excessivement contre un adversaire fixe et échoue face à des stratégies inédites. Les implémentations récentes utilisent le **Population-Based Training (PBT)** qui maintient une population diversifiée d'agents et sélectionne les meilleurs pour la reproduction, créant une pression évolutive qui favorise la généralisation des stratégies apprises.

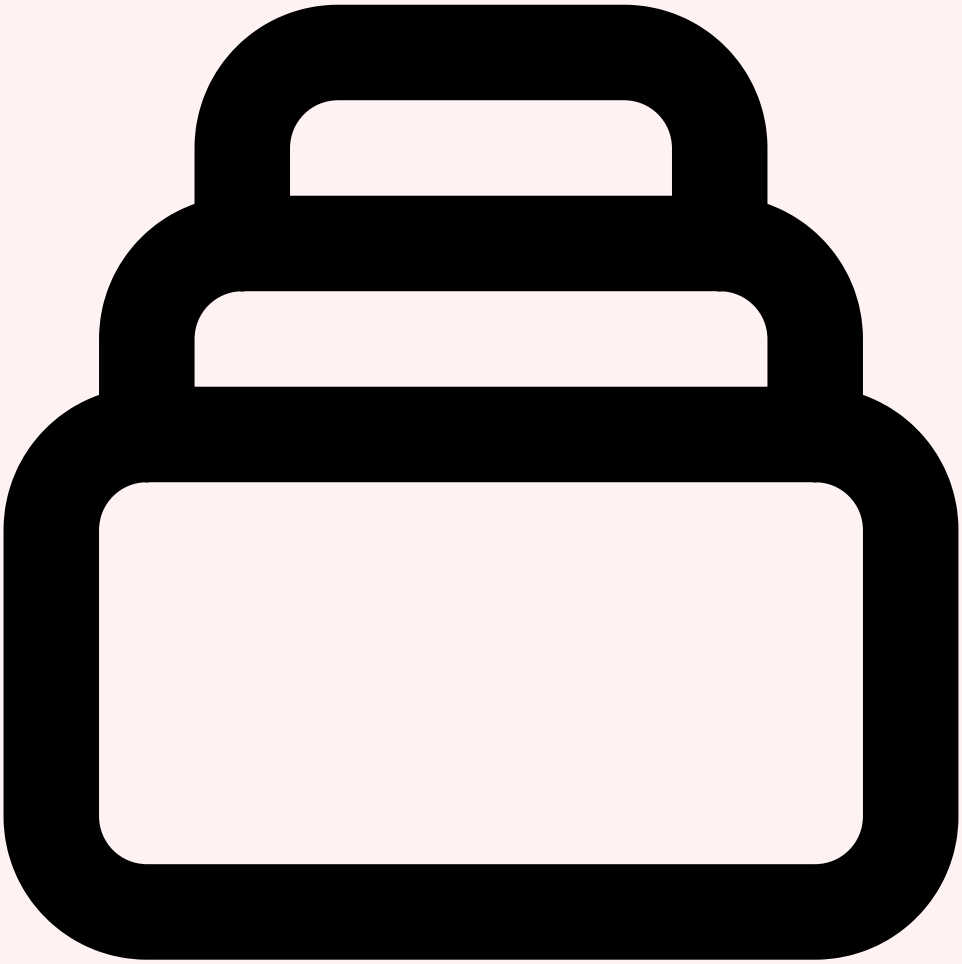


RL Défensif Multi-Agent RL Environnements



5 Environnements de simulation cyber

La qualité et le réalisme des environnements de simulation conditionnent directement l'applicabilité des agents RL entraînés sur des systèmes réels. En 2026, l'écosystème des simulateurs cyber a considérablement mûri, offrant un spectre allant de l'abstraction pure à la simulation haute fidélité avec émulation réseau complète. Le choix du simulateur dépend du compromis fondamental entre **vitesse d'entraînement** (nombre d'épisodes par seconde) et **fidélité de la simulation** (proximité avec un environnement réel).



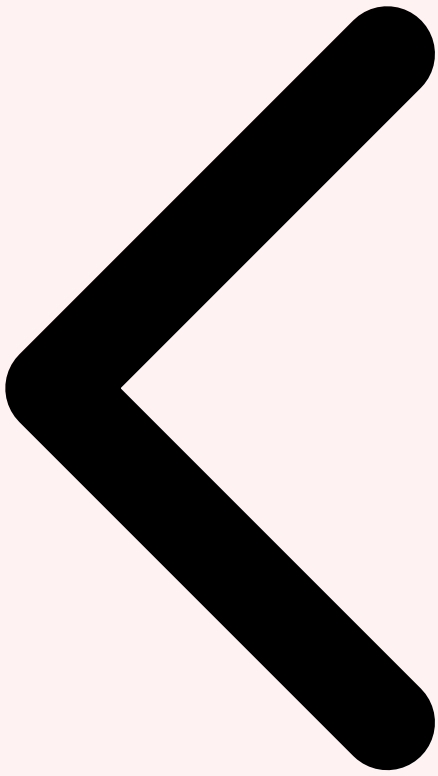
Panorama des simulateurs

CyberBattleSim (Microsoft) opère à un niveau d'abstraction élevé : le réseau est un graphe, les vulnérabilités sont des propriétés de noeuds, et les actions sont des opérations symboliques. Cette abstraction permet des vitesses d'entraînement de l'ordre de 10 000 épisodes par seconde sur un GPU unique, idéal pour la recherche exploratoire et le prototypage rapide d'algorithmes. En contrepartie, les politiques apprises nécessitent une adaptation significative pour fonctionner sur des réseaux réels.

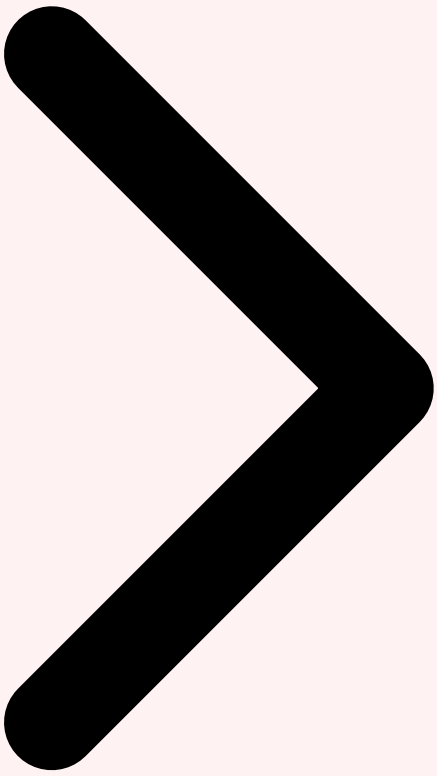
CybORG (Australian Defence Science and Technology Group) propose un niveau intermédiaire avec un modèle réseau plus détaillé incluant les sous-réseaux, les systèmes d'exploitation, les services et les processus. CybORG intègre nativement un scénario MARL où un agent Red Team affronte un agent Blue Team sur une infrastructure inspirée d'un réseau militaire. La version 3 de CybORG (2025) a introduit le support des scénarios multi-étapes avec persistance, la modélisation du chiffrement et la possibilité de déployer des honeypots comme action défensive. La vitesse d'entraînement est de l'ordre de 500 à 1000 épisodes par seconde.

FARLAND (Federated Autonomous Reinforcement Learning Across Network Domains) est un framework développé par le DARPA qui pousse la fidélité de simulation au maximum en émulant des machines virtuelles complètes via une intégration avec des hyperviseurs comme KVM ou des conteneurs Docker. Chaque action de l'agent RL se traduit par l'exécution réelle d'un outil ou d'une commande dans la VM cible. Cette approche haute fidélité produit des politiques directement transférables mais au prix d'une vitesse d'entraînement drastiquement réduite (1 à 10 épisodes par seconde), nécessitant des clusters de calcul dédiés et des techniques d'entraînement efficaces en données.

Network Attack Simulator (NASim), développé par l'université de Melbourne, offre un cadre formellement défini pour la recherche en RL cyber. NASim modélise les réseaux comme des ensembles de sous-réseaux interconnectés et utilise un formalisme mathématique rigoureux pour les transitions d'état. Son avantage principal est la capacité à générer automatiquement des scénarios de difficulté croissante, permettant un entraînement par curriculum learning. **Yawning Titan**, développé par le DSTL britannique (Defence Science and Technology Laboratory), adopte une approche similaire mais avec une interface graphique permettant aux analystes non experts en RL de concevoir des scénarios et de visualiser le comportement des agents.

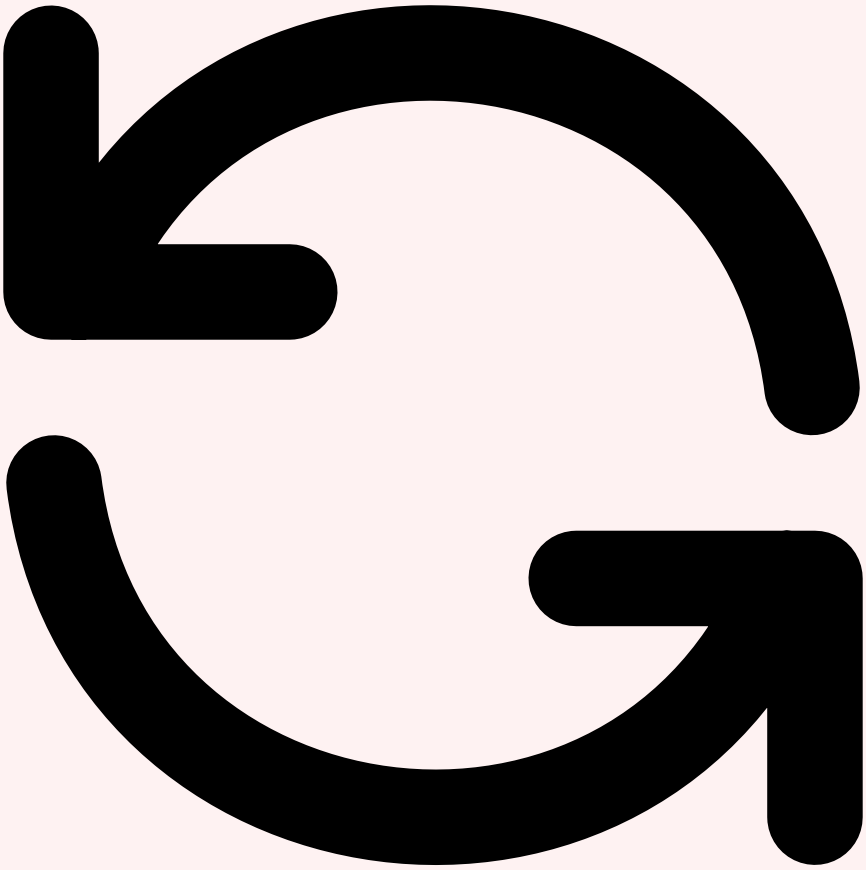


Multi-Agent RL Environnements de Simulation Transfer Learning



6 Transfer learning en RL cyber

Le **transfer learning** constitue l'un des défis les plus critiques et les plus actifs de la recherche en RL appliqué à la cybersécurité. Le problème fondamental est le suivant : un agent RL entraîné sur un environnement de simulation spécifique (avec une topologie réseau, des vulnérabilités et des configurations données) ne généralise pas naturellement à des environnements différents. Or, chaque réseau réel est unique. Le **sim-to-real gap** -- l'écart entre les performances de l'agent en simulation et en conditions réelles -- peut être considérable si des techniques de transfert appropriées ne sont pas mises en oeuvre.

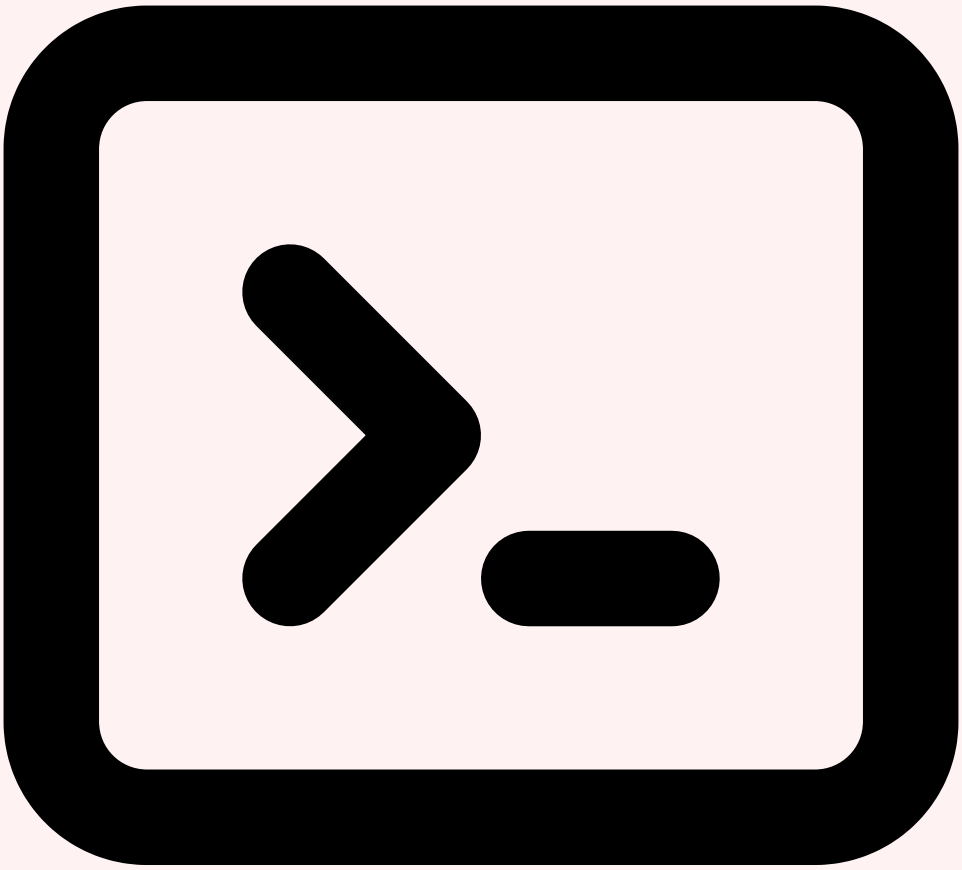


Domain Randomization et Curriculum Learning

La **domain randomization** est la technique de transfer la plus directe : pendant l'entraînement, les paramètres de l'environnement de simulation sont randomisés à chaque épisode. La topologie réseau varie (nombre de machines, connectivité, segmentation), les vulnérabilités sont redistribuées aléatoirement, les configurations de services changent, les latences réseau fluctuent. L'agent apprend ainsi une politique robuste qui ne se surajuste pas à une configuration spécifique mais capture les principes généraux de progression dans un réseau. Les travaux de Andrew et al. (2025) montrent que la domain randomization réduit le sim-to-real gap de 35% en moyenne sur des scénarios de mouvement latéral.

Le **curriculum learning** complète la domain randomization en structurant l'entraînement de manière progressive. L'agent commence par des environnements simples (réseaux de 5 à 10 machines, peu de services, vulnérabilités évidentes) et progresse vers des environnements de complexité croissante. Le curriculum peut être statique (défini manuellement) ou automatique (piloté par les performances de l'agent). La variante **Automatic Domain Randomization (ADR)**, inspirée des travaux d'OpenAI sur la manipulation robotique, ajuste dynamiquement la distribution de randomization pour

maintenir l'agent dans une zone de difficulté optimale -- ni trop facile (pas d'apprentissage), ni trop difficile (signaux de récompense trop rares). Pour approfondir, consultez Comet Browser : Architecture.

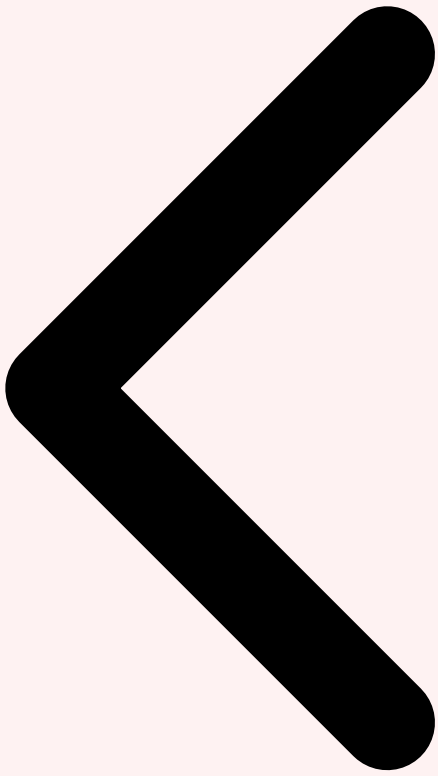


Représentations invariantes et Graph Neural Networks

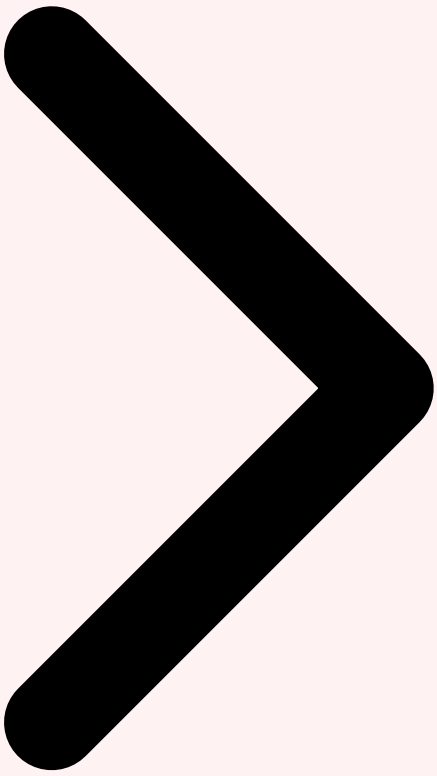
Une approche plus aboutie consiste à entraîner l'agent sur des **représentations invariantes** de l'environnement réseau plutôt que sur des observations brutes. Les **Graph Neural Networks (GNN)** sont particulièrement adaptées à cette tâche : elles encodent la topologie réseau sous forme d'un graphe où les noeuds représentent les machines et les arêtes les connexions, puis extraient des embeddings qui capturent les propriétés structurelles du réseau indépendamment de sa taille ou de sa configuration spécifique. Un agent RL dont l'encodeur d'état est un GNN peut traiter des réseaux de tailles arbitraires -- une capacité fondamentale pour le déploiement en conditions réelles où la topologie est inconnue a priori.

Les architectures **attention-based**, combinant GNN et mécanismes de self-attention inspirés des Transformers, représentent l'état de l'art en 2026 pour le transfer learning en RL cyber. L'agent apprend à focaliser son attention sur les machines et les chemins réseau

les plus pertinents pour sa progression, indépendamment de la taille du réseau. Le modèle **CyberGPT-RL** proposé par des chercheurs de Carnegie Mellon utilise un Transformer pour encoder les séquences d'observations et d'actions passées, créant un agent avec une mémoire contextuelle qui améliore la qualité des décisions dans les environnements partiellement observables. Les résultats expérimentaux montrent un transfer réussi entre des réseaux allant de 20 à 500 machines avec une dégradation de performance inférieure à 12%.

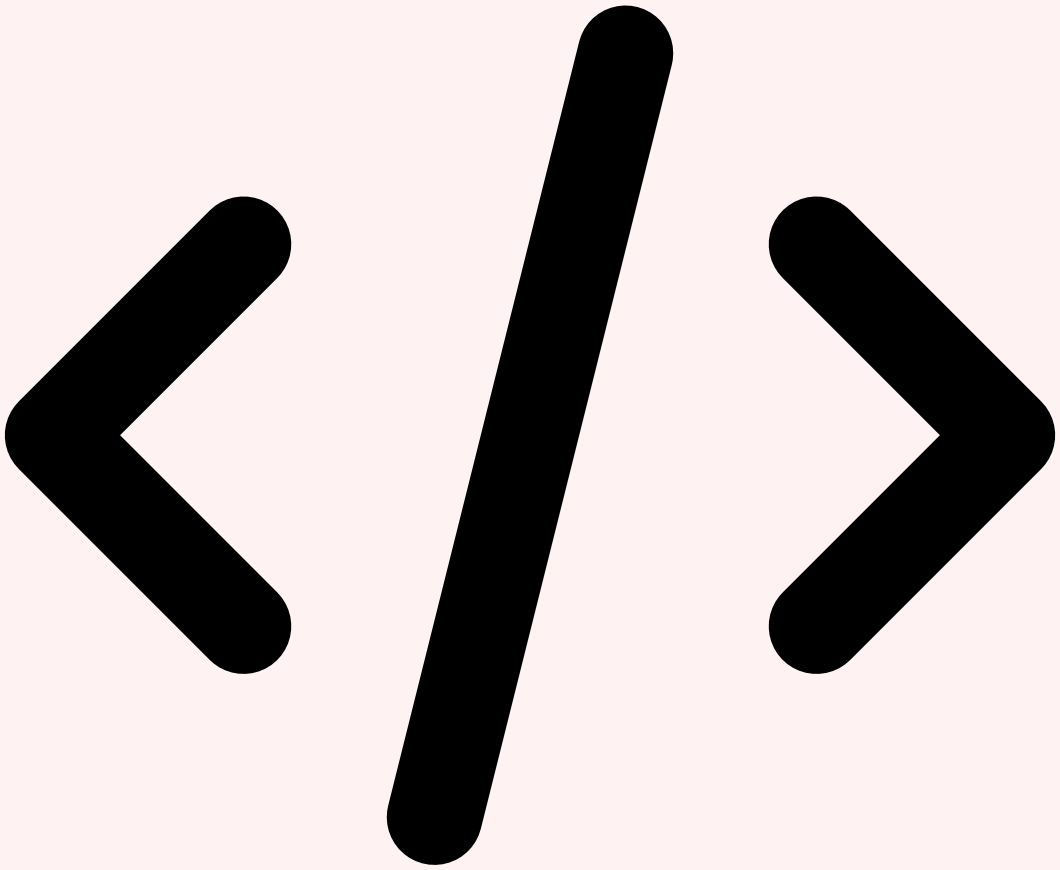


Environnements Transfer Learning Cas Pratiques



7 Cas pratiques et implémentations

Au-delà de la théorie, le déploiement opérationnel du RL en cybersécurité requiert des implémentations concrètes et des retours d'expérience terrain. Cette section présente trois cas pratiques représentatifs des applications les plus matures en 2026 : un agent de pentest automatisé, un système de réponse adaptative pour SOC et un agent de configuration de défense pour environnements OT/ICS.



Cas 1 : Agent de pentest automatisé avec CyberBattleSim + PPO

Ce premier cas pratique implémente un agent de mouvement latéral entraîné avec PPO sur CyberBattleSim. L'architecture réseau cible comprend 30 machines réparties en 4 sous-réseaux (DMZ, bureautique, serveurs applicatifs, base de données). L'agent démarre avec un accès sur une machine de la DMZ et doit atteindre le serveur de base de données en minimisant le nombre d'actions et en évitant la détection. L'espace d'observation est encodé sous forme d'un vecteur de 256 dimensions comprenant la matrice de connectivité connue, les vulnérabilités découvertes, les identifiants collectés et l'historique des 10 dernières actions. L'espace d'actions comprend 15 types d'actions paramétrées (scan, exploit, credential_reuse, lateral_move, etc.) applicables à chaque machine connue.

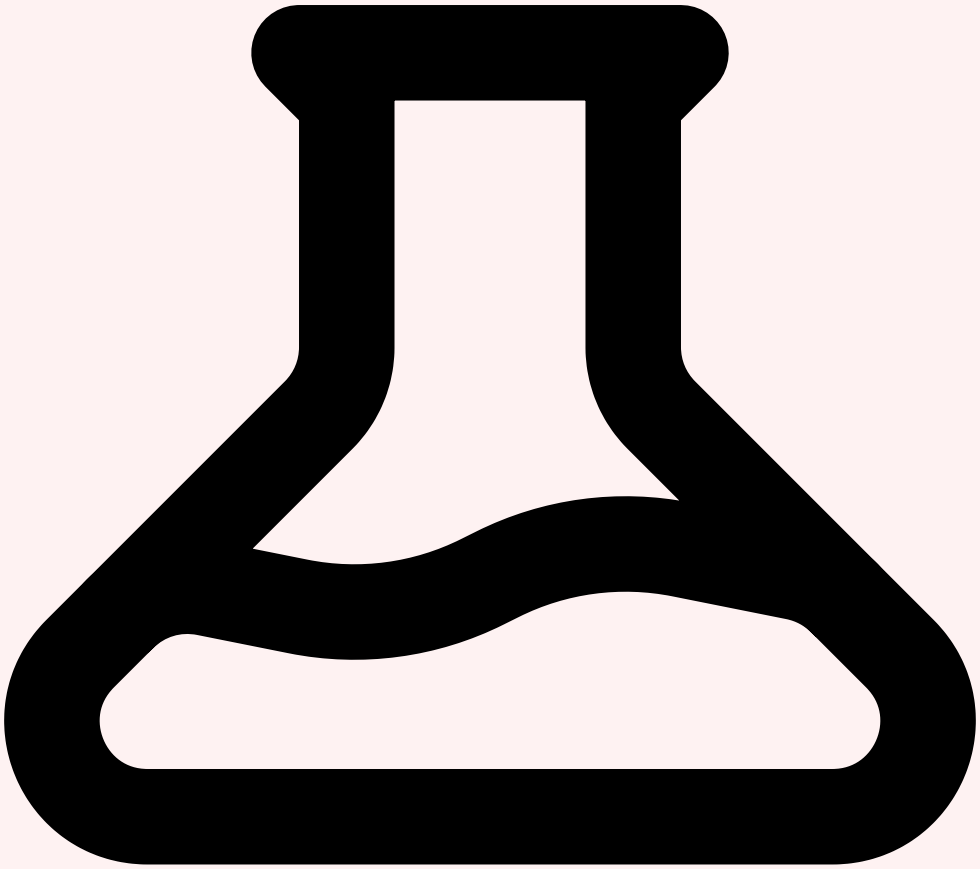
Après 500 000 épisodes d'entraînement (environ 8 heures sur un GPU A100), l'agent atteint un taux de succès de 94% avec un nombre moyen de 23 actions par épisode, contre 45 actions pour un agent DQN et plus de 60 pour une exploration aléatoire guidée. L'analyse des trajectoires révèle que l'agent a appris plusieurs stratégies poussées : il priorise l'exploitation de vulnérabilités permettant la collecte d'identifiants (credential harvesting) avant de tenter des exploits directs, il cible les machines ayant le plus de connexions sortantes pour maximiser sa surface de découverte, et il apprend à éviter les machines

équipées d'agents EDR (encoded comme une propriété du noeud) en privilégiant des chemins alternatifs. Ces comportements émergents n'ont pas été programmés mais résultent de l'optimisation de la récompense.

Cas 2 : Système de réponse SOC adaptative

Ce second cas porte sur un agent RL déployé dans un SOC pour l'orchestration automatisée de la réponse à incident. L'agent observe les alertes SIEM (Splunk/Elastic) en temps réel et décide des actions de réponse parmi un catalogue de 20 playbooks atomiques : blocage d'IP au niveau firewall, désactivation de compte Active Directory, isolement réseau d'un endpoint, déclenchement de scan IOC, collecte de mémoire volatile, escalade vers un analyste L2/L3. L'état comprend les métriques agrégées des dernières 24 heures (volume d'alertes par catégorie, distribution des sévérités, indicateurs de charge SOC), les propriétés de l'alerte courante (source, cible, technique ATT&CK mappée, score de confiance) et le contexte d'asset (criticité métier de la machine, dernier patch, appartenance à un VIP scope).

La fonction de récompense encode un trilemme : vitesse de réponse (bonus décroissant avec le temps de réaction), impact métier (pénalité proportionnelle au temps d'indisponibilité causé par les actions de réponse) et précision (bonus si la réponse est validée par un analyste, pénalité si elle est annulée). Après un entraînement de 3 mois sur des données historiques du SOC (replay d'incidents), l'agent réduit le MTTR (Mean Time To Respond) de 47% tout en diminuant les faux positifs opérationnels de 31%. Le point critique de cette implémentation est le **human-in-the-loop** : l'agent propose des actions de réponse mais un analyste humain conserve un droit de veto sur les actions à fort impact (isolement réseau, désactivation de compte). Ce mode de déploiement progressif, appelé *supervised autonomy*, est essentiel pour construire la confiance organisationnelle dans le système.

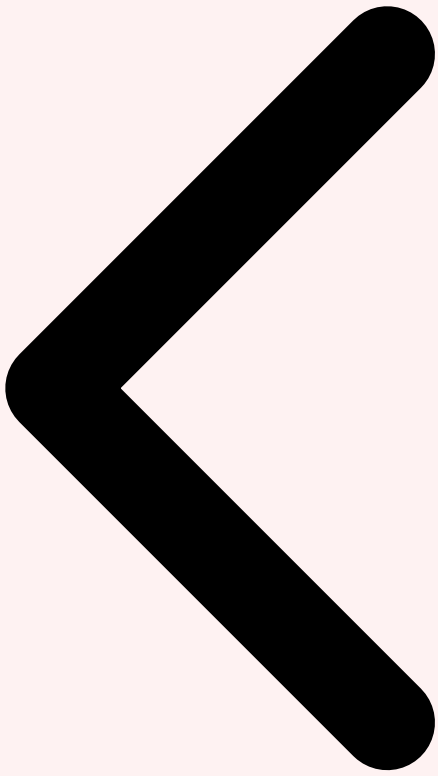


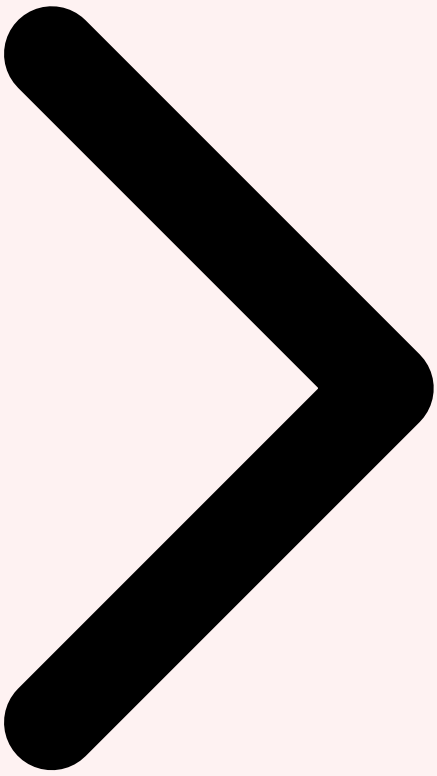
Cas 3 : Agent de défense OT/ICS

Les environnements OT (Operational Technology) et ICS (Industrial Control Systems) présentent des contraintes spécifiques qui rendent le RL défensif particulièrement pertinent mais aussi particulièrement délicat. La disponibilité prime sur la confidentialité : un faux positif qui bloque une commande légitime vers un automate industriel peut causer des dommages physiques. Les protocoles industriels (Modbus, OPC-UA, DNP3, IEC 61850)

ont des patterns de trafic très réguliers et prévisibles, ce qui facilite la détection d'anomalies mais augmente le risque de faux positifs en cas de maintenance légitime. L'agent RL-OT est entraîné sur un simulateur qui modélise un réseau SCADA avec des automates, des stations d'ingénierie, un historien et un serveur OPC-UA. Les actions de défense sont volontairement restreintes et graduées : alerte simple, alerte avec demande de confirmation opérateur, blocage temporaire avec basculement sur un mode dégradé prédéfini, et isolation d'urgence (cette dernière uniquement en confirmation humaine).

L'agent apprend à distinguer les variations légitimes du trafic OT (changement de consigne, procédure de maintenance planifiée, redémarrage d'automate) des tentatives d'intrusion (injection de commandes Modbus malveillantes, modification de firmware d'automate, exfiltration de données de process). La fonction de récompense incorpore un terme de **safety constraint** inspiré du Constrained RL (CRL) qui impose un taux de faux positifs maximum de 0,1% sur les commandes OT critiques. Les résultats sur le testbed montrent que l'agent RL-OT détecte 97% des scénarios d'attaque du dataset ICS-CERT tout en maintenant un taux de faux positifs de 0,07% -- une performance impossible à atteindre avec des règles de détection statiques qui oscillent typiquement entre 85% de détection à 0,5% de faux positifs ou 95% de détection à 2% de faux positifs.





8 Conclusion et perspectives

Le Reinforcement Learning appliqué à la cybersécurité a franchi un seuil de maturité décisif en 2025-2026. Les environnements de simulation sont désormais suffisamment réalistes pour produire des agents transférables. Les algorithmes (PPO, SAC, MAPPO) sont suffisamment stables pour un entraînement fiable. Et les cas d'usage -- du red teaming automatisé à la réponse à incident adaptative en passant par la défense OT -- ont démontré une valeur opérationnelle tangible. Cet article a parcouru les dimensions essentielles de ce domaine : les fondements théoriques du RL, les applications offensives avec CyberBattleSim et CALDERA, les défenses adaptatives par firewalls et IDS pilotés par RL, la dynamique multi-agent adversariale, l'écosystème des simulateurs et les techniques de transfer learning permettant le passage de la simulation au terrain. Pour approfondir, consultez IA et Zero Trust : Micro-Segmentation Dynamique Pilotée par.

Plusieurs défis restent néanmoins ouverts pour les années à venir. Le problème de la **sample efficiency** demeure critique : les agents RL nécessitent des millions d'épisodes d'entraînement, ce qui limite leur déploiement dans des environnements haute fidélité où

chaque épisode est coûteux en calcul. Les approches **model-based RL** (MuZero, Dreamer) qui construisent un modèle interne de l'environnement pour planifier sans interaction directe représentent une piste prometteuse. Le challenge de l'**explainability** est également fondamental : un agent RL qui prend des décisions de sécurité critiques (bloquer un flux, isoler une machine) doit être capable de justifier ses choix auprès des analystes SOC et des régulateurs. Les techniques de RL interprétable (attention visualization, policy distillation en arbres de décision, counterfactual explanations) progressent mais restent insuffisantes pour un déploiement sans supervision humaine dans des environnements critiques.

La convergence du RL avec les **Large Language Models (LLM)** ouvre des perspectives particulièrement intéressantes. Des agents comme ReAct et AutoGPT montrent qu'un LLM peut servir de politique de haut niveau dans un cadre RL, utilisant le raisonnement en langage naturel pour planifier des séquences d'actions complexes. Appliqué à la cybersécurité, un agent LLM-RL pourrait lire la documentation d'un réseau, interpréter des logs en langage naturel, rédiger des rapports d'incident et interagir avec les analystes en langage courant -- tout en bénéficiant de la capacité du RL à optimiser ses décisions par l'expérience. Les premiers prototypes de **pentesting agents LLM-guided** montrent des résultats prometteurs sur des CTF et des environnements de lab, avec des capacités de raisonnement et d'adaptation qui surpassent les agents RL purs sur les scénarios nécessitant une compréhension sémantique de l'infrastructure.

Recommandations pour les praticiens :

- 1.**Commencer par CyberBattleSim ou NASim** pour le prototypage et la validation d'algorithmes, avant de migrer vers des simulateurs haute fidélité comme FARLAND pour le transfer vers le réel
- 2.**Utiliser PPO comme algorithme de base** pour sa stabilité et sa polyvalence, puis explorer SAC pour les environnements à actions continues et MAPPO pour les scénarios multi-agent
- 3.**Implémenter la domain randomization** dès les premières itérations d'entraînement pour garantir la généralisation des politiques apprises à des topologies réseau inédites
- 4.**Encoder les contraintes de sûreté** dans la fonction de récompense via le Constrained RL, en particulier pour les environnements OT où un faux positif peut avoir des conséquences physiques
- 5.**Déployer en mode supervised autonomy** avec un analyste humain dans la boucle pour les décisions à fort impact, puis élargir progressivement le périmètre d'autonomie de l'agent à mesure que la confiance organisationnelle se construit
- 6.**Investir dans l'entraînement MARL adversarial** pour produire des politiques de défense robustes par construction, en utilisant le self-play et le Population-Based Training pour maximiser la diversité des stratégies
- 7.**Explorer les architectures GNN + Transformer** pour l'encodage d'état réseau, permettant le transfert des agents entre réseaux de tailles et de topologies différentes sans réentraînement

Besoin d'un accompagnement expert ?

Nos consultants en cybersécurité et IA vous accompagnent dans vos projets de Reinforcement Learning appliqué à la cybersécurité : red teaming automatisé, IDS adaptatifs et défenses multi-agent. Devis personnalisé sous 24h.

Références et ressources externes

- OWASP LLM Top 10 — Les 10 risques majeurs pour les applications LLM
- MITRE ATLAS — Framework de menaces pour les systèmes d'intelligence artificielle
- NIST AI RMF — AI Risk Management Framework du NIST
- arXiv — Archive ouverte de publications scientifiques en IA
- HuggingFace Docs — Documentation de référence pour les modèles de ML

Pour approfondir ce sujet, consultez notre outil open-source llm-security-scanner qui facilite l'audit de sécurité des modèles de langage.

Sources et références : ArXiv IA · Hugging Face Papers

FAQ

Qu'est-ce que Reinforcement Learning Appliqué à la Cybersécurité ?

Le concept de Reinforcement Learning Appliqué à la Cybersécurité est détaillé dans les premières sections de cet article, qui couvrent les fondamentaux, les enjeux et le contexte opérationnel. Pour un accompagnement sur ce sujet, contactez nos experts.

Pourquoi Reinforcement Learning Appliqué à la Cybersécurité est-il important en cybersécurité ?

La compréhension de Reinforcement Learning Appliqué à la Cybersécurité permet aux équipes de sécurité d'améliorer leur posture défensive. Les sections « Table des Matières » et « 1 Introduction au Reinforcement Learning » détaillent les raisons de cette importance. Pour un accompagnement sur ce sujet, contactez nos experts.

Comment mettre en œuvre les recommandations de cet article ?

Les recommandations pratiques sont détaillées tout au long de l'article, avec des commandes, des outils et des méthodologies éprouvées. La section « Conclusion » fournit une synthèse actionnable. Pour un accompagnement sur ce sujet, contactez nos experts.

Conclusion

Cet article a couvert les aspects essentiels de Table des Matières, 1 Introduction au Reinforcement Learning, 2 RL offensif : CyberBattleSim, CALDERA et génération d'attaques. La mise en pratique de ces recommandations permet de renforcer significativement la posture de securite de votre organisation.

Ayi NEDJIMI Consultants — Expert cybersécurité offensive & intelligence artificielle

ayinedjimi-consultants.fr · ayi@ayinedjimi-consultants.fr