Apprentissage par Renforcement et Blockchain : Nouvelle approche pour sécuriser l'IoT

Reinforcement Learning and Blockchain to secure the Internet of Things

Aissam Outchakoucht[1], Hamza Es-Samaali[1], Anas Abou El Kalam[2], Siham Benhadou[3]

[1] Ecole Nationale Supérieure d'Electricité et de Mécanique, Université Hassan II, Maroc, IPI, Paris, France, aissam.outchakoucht@gmail.com, hamza.essamaali@gmail.com
[2] Ecole Nationale des Sciences Appliquées, Marrakech, Université Cadi Ayyad, Maroc a.abouelkalam@uca.ac.ma
[3] Ecole Nationale Supérieure d'Electricité et de Mécanique, Université Hassan II, Maroc, Laboratoire LISER, siham.benhadou@gmail.com

RÉSUMÉ. La sécurité est un des sérieux problèmes qui menace le développement de l'internet des objets. Cependant, cette mission devient plus complexe dans les environnements IoT vu qu'ils ont des exigences intrinsèques supplémentaires telles que l'hétérogénéité, les capacités limitées de stockage et de calcul ainsi que le grand nombre de dispositifs. Pour remédier à ce problème, cet article propose un processus inspiré du concept de l'émergence visant à tirer profit de ce grand nombre d'objets intelligents et à en extraire les caractéristiques significatives que nous ne pouvons pas capter dans les systèmes avec un petit nombre. Le papier propose ensuite un framework de contrôle d'accès dédié aux environnements IoT basé sur trois notions de base : Réseaux de Blockchain, systèmes de réputation et algorithmes d'apprentissage par renforcement.

ABSTRACT. Securing the IoT world in not a luxury task; it is even a matter of urgency given this exponential growth of IoT market. In fact, one can easily imagine the catastrophic damages of an attack in the field of e-Health or in the smart cities and critical infrastructures management. That being said, serious problems derived from these constrained environments block the proposal of pertinent solutions. This paper is a contribution step in this direction. To address these problems, we expose a global framework inspired from the concept of emergence in order to take advantage of this large number of devices and extract the "emergent" characteristics that are nonexistent in smaller systems. The framework is built on top of two pillars: Blockchain as architecture and Reinforcement Learning as processing toolkit.

MOTS-CLÉS. Sécurité, Internet des objets, Contrôle d'accès, Politique dynamique, Émergence, Blockchain, Réputation, Apprentissage automatique, Apprentissage par Renforcement.

KEYWORDS. Security; Internet of Things, Access control, Dynamic policy, Emergence, Blockchain, Reputation, Machine Learning, Reinforcement learning.

1. Introduction[1]

On peut affirmer sans crainte que l'Internet des objets (IdO) perd son statut d'idée futuriste et devient de plus en plus un concept réel et quotidien qui ne cesse de s'étendre sur de nouveaux domaines [1, 1a]. Aujourd'hui, l'IdO se prépare à reconstruire notre vie quotidienne. Microsoft suppose que l'IdO est un élément clé de la transformation numérique mondiale. Certains spécialistes vont jusqu'à le considérer comme l'une des principales révolutions technologiques du XXIe siècle [2a, 2b, 2c].

Cependant, l'Internet des objets, même avec de sérieux efforts de normalisation, et même avec son grand succès en termes de ventes et de publicité, reste (au moins sous les yeux de nombreux chercheurs et experts du domaine) en phase d'exploration et d'expérimentation. En effet, l'IdO jusqu'à présent n'est toujours pas en mesure de surmonter un grand nombre de ses défauts intrinsèques, certaines sont critiques et peuvent même bloquer cette explosion de vente des objets intelligents, à savoir les

[1] Cet article est une extension de notre article intitule "Emergence-Based Access Control: New Approach to Secure the Internet of Things", Proceeding, DTUC'18, 1st International Conference on Digital Tools & Uses Congress, October 2018, ACM, 10.1145/3240117.3240136

contraintes de capacités de calcul, de stockage, le nombre énorme et l'hétérogénéité des objets qui caractérisent les plateformes IdO, sans citer les problèmes d'alimentation, d'identification, de migration vers IPv6, etc. Entre autres, nous pensons que la sécurité du monde de l'IdO doit être une priorité compte tenu de l'impact et de la nature des catastrophes qui peuvent être causées par sa déficience. Il va sans dire que lorsqu'on parle de l'IdO dans le secteur de la santé, par exemple, attaquer ou compromettre un dispositif intelligent se traduit directement par la mise de la vie des gens entre les mains des caprices du pirate.

La situation devient encore plus délicate lorsque nous arrivons à la conclusion décevante affirmant que nous ne pouvons pas appliquer simplement et aveuglément les solutions de sécurité traditionnelles dans le contexte de l'IdO, étant donné les limites des dispositifs d'IdO, notamment en termes de capacités de calcul.

Pour répondre aux attentes de ses utilisateurs, l'IdO doit également être en mesure de vérifier plusieurs exigences conceptuelles et techniques qui doivent couvrir : la grande flexibilité, l'hétérogénéité, l'évolutivité et la coopération. Pourtant, comme le souligne l'IEEE [3], la sécurité et la protection de la vie privée demeurent les exigences essentielles de l'IdO.

Malheureusement, ce n'est pas évident d'assurer ces derniers, puisque pour ce faire l'IdO doit au moins couvrir les besoins de la triade de la CIA (Confidentialité, Intégrité et Disponibilité) ; d'autres exigences sont ajoutées dans des circonstances spécifiques (à savoir fiabilité, traçabilité, non-répudiation, et bien d'autres). Dans le présent document, nous traitons ces besoins au moyen de solutions de contrôle d'accès.

1.1. *IdO et Control d'accès*

Un environnement typique d'IdO se compose généralement de dispositifs hétérogènes avec des capteurs intégrés interconnectés par l'intermédiaire d'un réseau. Les objets IdO sont identifiables de manière unique et se caractérisent principalement par une faible puissance d'alimentation, une faible mémoire et une capacité de traitement limitée. Les passerelles (s'ils existent) sont déployées pour connecter des dispositifs IdO au monde extérieur afin de fournir à distance des données et des services aux utilisateurs [4].

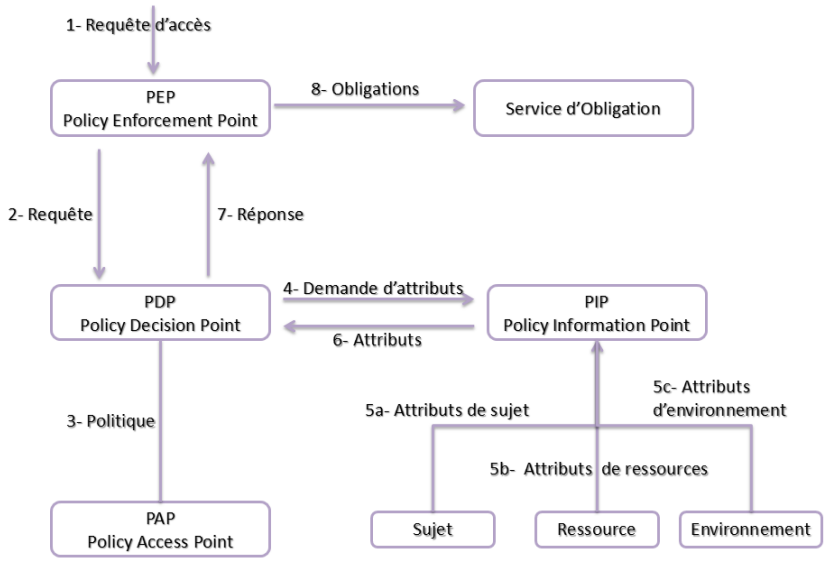

Figure 1. *Blocs principaux du contrôle d'accès*

Par ailleurs, dans le monde de la sécurité, le contrôle d'accès (CA) a été considéré comme un facteur clé étant donné sa mission de protéger les accès aux ressources d'un système (numériques ou

physiques) en définissant et en mettant en œuvre qui a (ou n'a pas ou doit avoir) accès à quoi, quand et dans quelles conditions.

En termes d'architecture, comme le montre la Figure 1, le Point d'administration de la politique (PAP) met d'abord la politique de CA complète à la disposition de l'ADF (Access Decision Facility) ou du PDP (Policy Decision Point). Chaque demande présentée par un sujet est interceptée par un service d'exécution (AEF) ou un point d'application des politiques (PEP), puis transmise au PEP/FDA. Par la suite, le PDP invoque le Point d'information sur les politiques (PIP) pour récupérer les valeurs des attributs liées aux sujets, aux ressources, aux actions et à l'environnement. Le PDP évalue la demande grâce aux règles du PAP et aux différents attributs obtenus. La décision d'autorisation prise par le PDP est transmise au PEP, il peut s'agir d'un "permis" ou d'un "refus", avec les obligations appropriées. Le PEP applique cette décision de façon appropriée en s'acquittant de ses obligations et en autorisant ou en refusant l'accès, conformément à la décision du PDP.

1.2. *Contribution*

Même avant l'apparition du paradigme de l'IdO tel que nous le connaissons aujourd'hui, les responsables de la sécurité des systèmes d'information (RSSI) souffraient toujours plus lorsque le nombre de dispositifs à protéger est élevé et leurs tâches deviennent de plus en plus complexes au fur et à mesure que le nombre de nœuds augmente. En général, les gens faisaient un lien direct et intuitif entre l'augmentation du nombre d'appareils et la génération de problèmes supplémentaires.

Cependant, aujourd'hui cette idée s'évapore, du moins dans plusieurs systèmes comme les réseaux torrents, le Deepweb ou encore les réseaux Blockchain. Dans ces types d'environnements, l'ajout de nœuds augmente clairement la sécurité du système.

Le processus qui consiste à tirer profit du grand nombre d'objets intelligents que l'IdO traite et à en extraire les informations pertinentes que nous ne pourrons pas voir dans les systèmes avec un plus petit nombre d'entités est appelé " Emergence ". Le présent document examinera cette notion de manière approfondie tout en essayant d'en tirer le meilleur. Ceci en se basant sur trois concepts : Blockchain, Systèmes de réputation et l'Apprentissage par renforcement (Reinforcement Learning).

L'objectif de ce papier est de fournir un cadre conceptuel et pratique de contrôle d'accès qui répond aux besoins de décentralisation, d'optimisation et de confiance tout en restant efficace.

1.3. *Organisation*

Cet article document est organisé comme suit : La section 2 expose un aperçu de toutes les notions et concepts clés sur lesquelles reposes le framework proposé, à savoir l'émergence, la blockchain, l'apprentissage par renforcement et les systèmes de réputation. Ensuite, la section 3 présente l'état de l'art des concepts étudiés. La section 4 dévoile plus de détails sur notre framework en présentant ses composantes et son processus de fonctionnement, tandis que la section 5 présente une étude de cas pour illustrer, expliquer et appliquer notre proposition. Enfin, nous présentons quelques futurs travaux et conclusions dans la section

2. Contexte

Pour répondre aux exigences des environnements IdO, nous construirons progressivement un framework de contrôle d'accès qui intègre plusieurs notions telles que Blockchain, les systèmes de réputation et les algorithmes d'apprentissage par renforcement, le tout dans une approche d'"émergence". Dans cette section, nous expliquons les essentielles notions liées à ces concepts.

Tout d'abord, rappelons que, quelle que soit l'application envisagée, le framework de contrôle d'accès dédié de l'IdO devrait avoir, au moins, les caractéristiques suivantes :

– *Politique de CA entièrement distribuée*, en harmonie avec la décentralisation qui caractérise les plates-formes IdO. Pour s'en assurer, la proposition présentée ici s'appuie sur une technologie extrêmement distribuée et largement diffusée, qui a démontré son efficacité dans l'un des domaines les plus sérieux : le marché monétaire. Cette technologie porte le nom de Blockchain.

– *Système coopératif et coordonné*. Pour ce faire, nous avons opté pour un système de réputation basé sur les rétroactions/retours (feedbacks) afin d'améliorer le niveau de confiance dans les réseaux IdO et d'exploiter ainsi le grand nombre de dispositifs IdO dans une approche coopérative au lieu d'un processus de CA déconnectés où chaque objet ou famille d'objets est traitée indépendamment de son entourage.

– *Politique dynamique* qui prend en compte le contexte des objets intelligents, mais aussi qui peut être améliorée dans le temps.

Cette amélioration n'est évidemment pas (et ne pourrait pas être) gérée par un être humain, étant donné la quantité énorme et hétérogène de données que l'IdO génère. Notre proposition s'appuie donc sur la puissance des algorithmes d'apprentissage automatique (Machine Learning), en particulier ceux de l'apprentissage par renforcement (Reinforcement Learning), pour accomplir cette tâche.

Cela dit, dans les sous-sections suivantes, ces trois piliers de notre framework seront présentés de façon plus détaillée.

2.1. *Émergence*

Ce terme apparaît dans plusieurs domaines qui peuvent sembler largement séparés les uns des autres. En effet, l'émergence est une notion partagée entre la philosophie, la science, l'art et plusieurs autres domaines. Il s'agit de phénomènes qui ne surviennent que lorsqu'un grand nombre d'entités commencent à interagir les unes avec les autres, alors que ces phénomènes ne sont pas présents dans chaque entité séparément. C'est l'une des caractéristiques les plus fascinantes et les plus mystérieuses de notre univers.

Le concept de l'émergence pourrait aussi être compris par la règle fréquente qui dit : "Le tout est plus grand que la somme de ses parties". Par exemple, on dit parfois que la conscience est une propriété émergente du cerveau [5], ce dernier n'étant biologiquement qu'une simple collection de neurones.

Prenons l'exemple de l'humidité ; quelque chose de nouveau créée seulement par beaucoup d'interactions individuelles entre les molécules d'eau. Beaucoup de choses interagissent suivant un certain ensemble de règles, créant ainsi quelque chose qui va au-delà d'elles-mêmes. Une colonie de fourmis pourrait aussi être un excellent exemple ; elle peut construire des structures complexes. Cependant, une fourmi est pratiquement "stupide", et pourtant, beaucoup de fourmis ensemble sont intelligentes. Certaines colonies gardent des fermes de champignons, d'autres s'occupent du bétail, elles peuvent faire la guerre ou se défendre.

Contrairement aux politiques et modèles classiques de contrôle d'accès, un contrôle d'accès adapté pour l'IdO devrait tirer profit du nombre de nœuds IdO, extraire l'intelligence collective afin de se reconfigurer, réadapter la politique et prévoir des attaques nouvelles ou potentielles... D'une manière ou d'une autre, les modèles proposés doivent proposer une stratégie pour *enseigner* aux "objets IdO" comment *s'auto-sécuriser*.

En effet, l'IdO a (et aura) des millions, voire des milliards, d'objets dits intelligents. Comment les utiliser en exploitant le concept d'émergence ? C'est ce que nous produisons dans cette proposition en intégrant des systèmes de réputation, des réseaux Blockchain et des algorithmes d'apprentissage par renforcement.

Figure 2. *L'émergence = Nombre × Coopération*

La figure 2 montre que l'émergence devient plus significative lorsque les deux paramètres, nombre et coopération, sont amplifiés.

2.2. *Blockchain*

Le concept de Blockchain a été initialement introduit en 2008 [6] pour la simple raison de soutenir le réseau de crypto-monnaie Bitcoin. Il s'agit d'un grand « registre » (ledger) public et distribué qui contient toutes les transactions précédemment exécutées dans son réseau. Ce registre est protégé cryptographiquement et partagé collectivement par tous les nœuds appartenant au réseau. Le principal avantage de la Blockchain est qu'elle permet aux individus de se faire confiance les uns les autres sans faire appel à un tiers de confiance. Un réseau de nœuds assure cette fonction intermédiaire de manière cryptographiquement vérifiable. En d'autres termes, la Blockchain est une infrastructure de certification collaborative des transactions sur Internet, sans aucune tierce partie de confiance. Pour cette raison, la blockchain se répand même en dehors du champ de la crypto-monnaie ; elle est maintenant présente dans les systèmes de stockage [7], les systèmes de réputation [8], la gestion des identités [9], etc.

Plus formellement, la Blockchain est une base de données distribuée qui s'articule autour de trois éléments fondamentaux :

1) *Identités* : Dans les réseaux basés sur la technologie Blockchain, les utilisateurs ont des identités numériques (*adresses*) qu'ils utilisent pour envoyer et recevoir des transactions. Ces adresses doivent être auto-générées (indépendantes de toute autorité centrale) et, dans la mesure du possible, anonymes (ne rien révéler sur l'identité réelle de leur propriétaire). Par conséquent, chaque utilisateur dispose d'au moins un endroit où il peut stocker ses identifiants, ses adresses et les transactions qui le concernent. Dans la terminologie de Blockchain, ce lieu s'appelle un *portefeuille* (wallet), il détient toutes les clés nécessaires pour enregistrer, identifier ou signer les transactions de son propriétaire.

2) *Transactions* : Une transaction fait référence au transfert d'une valeur (argent, droit ou autre information) entre deux adresses. Les transactions sont créées par leurs expéditeurs puis réparties sur l'ensemble du réseau. Notez que ces transactions sont publiquement vérifiables et irréversibles ; en d'autres termes, une fois qu'une transaction est enregistrée dans la Blockchain, il devient impossible de la modifier ou de la supprimer.

3) *Consensus* : Pour fonctionner à une échelle mondiale, un grand registre (chaine de blocs) partagé nécessite forcément un algorithme de consensus efficace. Dans les réseaux basés sur la Blockchain, chaque nœud suit des règles strictes pour traiter les transactions sans aucune interaction humaine afin de valider de manière autonome l'exécution correcte du protocole, et obtenir les mêmes résultats. Par conséquent, les utilisateurs du réseau partagent exactement le même livre, ce qui permet d'assurer un consensus parfait de toutes les parties prenantes dans la blockchain correspondante.

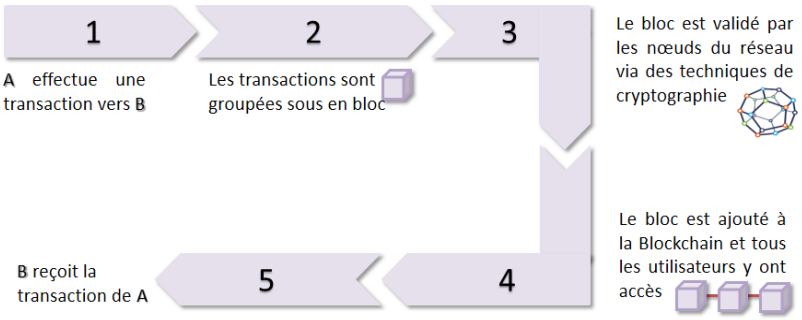

Figure 3. *Fonctionnement global du concept Blockchain*

Il est également important de réaliser qu'un bloc (groupe de transactions) doit être validé avant de rejoindre le registre public de la blockchain ; cette tâche est confiée à certaines entités appelées mineurs (miners) qui sont en charge de résoudre certains problèmes cryptographiques coûteux en ressources appelés preuve du travail (PoW).

La figure 3 décrit comment les transactions sont ajoutées à la blockchain. Elle présente une procédure en cinq étapes expliquant la logique de cette technologie.

2.3. *Apprentissage par Renforcement*

Les bases du paradigme de l'apprentissage automatique (Machine Learning - ML) ont été fondées vers la fin des années 1940 [9] en s'inspirant de plusieurs domaines comme les mathématiques, les statistiques, les neurosciences et l'informatique. Aujourd'hui, l'apprentissage automatique est une branche active de l'intelligence artificielle, ses techniques sont utilisées dans différents domaines et activités dont la génétique, le traitement du langage naturel, la régression, la classification, la détection de la fraude, la détection du spam, les moteurs de recherche et les réseaux publicitaires.

Malgré cette diversité, les algorithmes de ML se classent généralement dans l'une de ces trois catégories principales [11] :

– Apprentissage supervisé (SL) : Dans lequel les algorithmes sont fournis avec des données d'apprentissage étiquetées. Ces dernières sont généralement présentées comme un ensemble de données d'entrée *(x)* et de données de sortie *(y)*, l'objectif de l'algorithme est de comprendre la relation entre ces données (identifier la fonction de liant les deux $y=f(x)$) si bien que lorsqu'on lui donne une nouvelle entrée *(x)*, il peut prévoir la sortie correspondante *(y)*.

– Apprentissage non supervisé (UL) : Les algorithmes ici n'utilisent pas de données d'apprentissage étiquetées. On ne leur fournit que les données d'entrée *(x)* et on les laisse explorer, apprendre puis les classer dans différents groupes (clusters) après avoir révélé les similitudes et/ou les relations entre eux.

– Apprentissage par renforcement (RL) : Cette catégorie, également connue sous le nom d'apprentissage en ligne (Online Learning), décrit le processus d'apprentissage d'un comportement uniquement par le biais des interactions entre un agent (qui représente l'algorithme) et un environnement dynamique (la plate-forme active avec laquelle l'algorithme interagit). En 2017, RL a été choisi comme l'une des 10 technologies de pointe de la revue MIT Technology.

Cela dit, ces trois approches ne sont pas mutuellement exclusives, on peut trouver des algorithmes qui partagent les caractéristiques des méthodes d'apprentissage supervisé et non supervisé (algorithmes hybrides) afin d'augmenter leurs forces et de réduire leurs inconvénients [10].

Dans les grands réseaux hétérogènes comme dans le cas des environnements IoT, nous pensons que l'utilisation des algorithmes RL est le choix approprié pour résoudre les problèmes des politiques de

CA statiques et non contextuelles. En effet, pour plus d'efficacité et d'autonomie, l'algorithme proposé doit détecter pendant le fonctionnement du système (c'est-à-dire en même temps que la politique de sécurité est entrain de s'exécuter) les règles de CA qui ne sont pas optimales ou même qui présentent ou entraînent des problèmes de sécurité. En ce sens, il s'agit d'offrir un apprentissage en ligne.

En tant que preuve de concept, le RL n'est pas une nouvelle notion ; elle a démontré son efficacité dans plusieurs domaines. En fait, un grand nombre de chercheurs dans différents domaines ont adopté le RL ; ils l'ont principalement utilisée comme outil de calcul pour construire des systèmes autonomes et autogérés capables de se développer avec des expériences d'essais et erreurs. Ces applications ont touché plusieurs domaines dont la robotique, l'industrie ainsi que des problèmes de recherche combinatoire tels que les jeux vidéo et bien d'autres. A notre connaissance, ce travail est le premier qui introduit et applique la RL au contrôle d'accès dans des environnements IdO.

2.4. *Systèmes de réputation*

Étant donné que le nombre de personnes, de services et de dispositifs d'IdO qui interagissent en ligne augmentant de manière exponentielle, il est donc fortement recommandé d'avoir recours à la notion de réputation. En fait, en utilisant ce type de systèmes, les appareils du réseau gardent un œil l'un sur l'autre et peuvent donc exclure ceux dont le comportement est suspect.

En général, les systèmes de réputation sont utilisés afin d'assurer la confiance entre les nœuds constituant un environnement donné.

Ils atteignent cet objectif en agissant comme un tout et en veillant les uns sur les autres.

Après chaque expérience entre deux entités, et selon le modèle utilisé par le système de réputation, on peut soit agir en binaire (i.e. noter par bon ou mauvais, 0 ou 1) soit donner des points qui s'accumulent progressivement et forment l'image (réputation) de l'entité en question.

Pratiquement, se référer aux feedbacks donnés par les systèmes de réputation permet aux utilisateurs de décider à qui ils feront confiance dans les transactions à venir et dans quelle mesure. De plus, ces types de systèmes sont socialement correctifs, parce que un objet avec une réputation positive encourage plus de nœuds à interagir avec lui et du coup renforcer plus sa réputation, et réciproquement [12]. Un benchmark de systèmes de réputations est détaillée en [13].

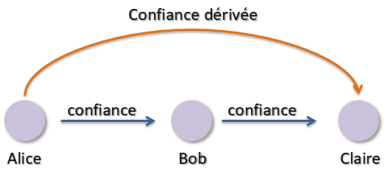

Figure 4. *Transitivité de la confiance*

Comme le montre la figure 4 ci-dessus, la robustesse de la réputation repose également sur sa vérification de la propriété de transitivité. Dans sa forme simplifiée, si *A* fait confiance à *B*, et que *B* fait confiance à *C*, *A* peut donc faire confiance à *C* sans qu'il y est une interaction directe entre eux. Cela renforcera le réseau de confiance d'un objet avec des nœuds avec lesquels il n'a jamais effectué de transactions.

3. Travaux connexes

Dans cette section, et en fonction des besoins et des problèmes exposés précédemment, nous décrivons les travaux existants dans la littérature et essayons de comprendre leurs faiblesses et leurs limites dans le contexte de l'Internet des objets.

3.1. *Contrôle d'accès*

Intuitivement, le contrôle d'accès peut se résumer en une phrase : "qui a accès à quoi". Par ailleurs, il est clair dans la Figure 1 que pour prendre une décision d'autoriser ou de refuser l'accès à une ressource, le moteur de CA doit consulter la politique de CA. Cette dernière, et sur la base de la définition donnée par les Critères communs (CC), peut être représenté comme une ou plusieurs règles, procédures, pratiques ou directives de sécurité imposées par une organisation sur ses opérations[14].

Jusqu'à présent, pour plusieurs organisations ainsi que pour plusieurs modèles de CA, les politiques de CA constituent en général une liste plus ou moins exhaustive des propriétés de sécurité à satisfaire ainsi que des règles à consulter et à appliquer « mécaniquement » pour prendre une décision de CA (permission, obligation, interdiction, recommandation).

De nombreux modèles de contrôle d'accès ont été proposés dans la littérature pour résoudre les problèmes de sécurité. On trouvera ci-après un résumé des plus récents et/ou des plus pertinents, ainsi qu'une analyse de leur pertinence dans le contexte de l'IdO :

L'un des pionniers modèles de CA est le modèle de CA basé sur les rôles (Role Based Access Control - RBAC) [15]. Elle régit l'accès des utilisateurs aux ressources des systèmes, en fonction de la notion de rôles. Cette dernière est définie comme une agrégation de nombreux sujets qui partagent certaines caractéristiques (qui jouent le même rôle).

Plusieurs travaux ont étendu le modèle RBAC au contexte de l'IdO. Dans [16, 17], les auteurs adoptent une approche basée sur les services pour coller à l'Internet des objets ; l'objet physique est donc mappé à un ou plusieurs services. Ensuite, le service est considéré comme la cible de la demande, dont l'autorisation est vérifiée par contrôle d'accès avant d'être effectuée par le service. Le modèle RBAC est étendu à un nouveau modèle appelé contrôle d'accès contextuel par l'introduction du contexte fourni par le service Web. Dans ce modèle, la permission est attribuée au rôle en fonction des caractéristiques et des informations contextuelles recueillies dans l'environnement de l'objet physique [18, 19].

Plus récemment, nous avons proposé un modèle de CA basé sur l'organisation (Organization Based Access Control - OrBAC) [20] avec l'idée de généraliser l'abstraction de toutes les entités de la politique de sécurité (sujets, objets et actions) et de séparer complètement la spécification de la politique de sécurité (uniquement par le biais d'entités abstraites) de la décision d'accès. Nous avons donc proposé de rassembler les sujets pour former des *rôles* (comme le fait RBAC), mais aussi d'abstraire des objets qui partagent les mêmes propriétés en *vue* et de regrouper des actions qui achèvent les mêmes tâches en *activités*. OrBAC fait ainsi une séparation parfaite entre le niveau abstrait (rôles, vues, activités) et le niveau concret (sujet, objet, action).

De plus, OrBAC introduit deux dimensions originales : le concept organisationnel (qui peut être traduit par le propriétaire de l'objet dans le monde de l'IdO), et les informations contextuelles, exceptionnellement indispensables avant de prendre une décision de CA. Toutefois, le gros inconvénient de ces modèles dans le contexte de l'IdO est leur architecture centralisée.

Cela dit, de nombreuses propositions ont été publiées pour étendre l'OrBAC et remédier à cet inconvénient : PolyOrBAC[21] traite ce problème en utilisant le modèle OrBAC comme base pour gérer les politiques internes de chaque organisation, puis ajoute une couche de collaboration entre organisations. Néanmoins, les nœuds limités de l'IdO ne supporte pas systématiquement les technologies utilisées par PolyOrBAC (par exemple, les services Web basés sur SOA). Pour remédier à

cela, SmartOrBAC [22, 23] a essayé de réajuster OrBAC afin de garantir les exigences de l'IdO. Pour ce faire, les auteurs se sont basés sur une architecture hybride et ont introduit une couche intermédiaire moins contrainte capable de gérer les objets par clusters ; sauf qu'en fin de compte, SmartOrBAC ne fournit pas d'outils ou de mécanismes légers pour réduire la complexité de l'OrBAC afin qu'il puisse être maintenu par les dispositifs contraints utilisés dans les environnements IdO.

Récemment, une attention particulière a été accordée au contrôle d'accès basé sur les attributs (Attribute Based Access Control - ABAC) [24] qui permet bien évidemment de répondre aux problèmes de CA en se reposant sur la notion d'attributs. Ces derniers caractérisent chaque sujet et objet et les identifient. La recherche dans ce sens est très prometteuse, mais elle n'a pas encore atteint sa maturité jusqu'à présent.

Par ailleurs, XACML [25] est un langage de contrôle d'accès basé sur XML qui a été normalisé par OASIS (Organisation for the Advancement of Structured Information Standards). Il décrit à la fois un langage de politique de contrôle d'accès (ABAC) et des décisions de contrôle d'accès (demande/réponse). Un autre cadre d'autorisation générique pour l'Internet des objets est proposé dans [26]. Encore une fois, en raison de la complexité de l'évaluation des politiques XACML pour les objets soumis à de fortes contraintes de ressources, une grande partie du processus décisionnel d'autorisation d'accès est externalisée (PDP externalisé). Malgré la fonctionnalité de verbosité de XACML, la solution de Seitz fait face à cet inconvénient en proposant une représentation compacte en JSON pour le format d'affirmation. Néanmoins, la solution est encore trop lourde pour être intégrée côté objets IdO. Par conséquent, une grande partie du processus décisionnel d'octroi d'accès est externalisée (PDP) et l'objet impose simplement l'exécution de l'autorisation pour prendre en compte la décision locale qui justifie le niveau de distribution et la légèreté des solutions basées sur XACML.

Le contrôle d'accès basé sur les capacités/aptitudes (Capability Based Access Control - CapBAC) est également une autre famille de modèles de CA qui a été appliquée à l'IdO. Le concept de capacité a été introduit dans [27] en tant que "*jeton (token), ticket ou clé qui donne à son possesseur la permission d'accéder à une entité ou à un objet dans un système informatique* ". Dans [28], les auteurs présentent un protocole d'authentification et de contrôle d'accès pour l'IdO basé sur l'ECC, appelé modèle IACAC (Identity Authentication and Capability Based Access Control) implémenté dans un environnement Wi-Fi.

Un autre modèle de CA intéressant est le contrôle d'utilisation (UCON) proposé dans [29], il est considéré comme la prochaine génération de modèles de contrôle d'accès pour la raison qu'il présente plusieurs nouveautés non disponibles dans les modèles de CA traditionnels tels que RBAC et ABAC. Il traite les problèmes générés dans la phase d'autorisation, avant l'exécution de l'accès, après l'exécution de l'accès, ou même pendant l'exécution. De plus, il a la capacité de supporter la mutabilité des attributs ; en d'autres termes, si un problème est produit dans la politique de sécurité (pendant l'exécution) suite à une altération de certains attributs d'accès, l'accès autorisé est annulé et l'utilisation devient invalide. De plus amples informations sur le modèle UCON sont détaillées dans [30]. De nombreuses recherches (comme dans [31]) ont également appliqué l'UCON dans des systèmes collaboratifs.

OAuth est un autre framework de contrôle d'accès qui a également été appliqué à l'IdO [32, 35]. Il s'agit d'un framework d'autorisation qui permet aux utilisateurs d'accorder aux applications tierces, agissant en tant que partie de confiance, l'accès aux ressources protégées (hébergées dans un service particulier jouant le rôle d'un fournisseur d'identité (Identity Provider - IdP)) sans révéler leurs identifiants de connexion au tiers. Les principaux inconvénients et défis de l'adoption d'OAuth par l'IdO sont les suivants : 1) Difficulté à obtenir une implémentation sûre : Il y a toujours un compromis à faire entre la sécurité et la facilité d'utilisation. En effet, une analyse empirique des mises en œuvre existantes du protocole a montré qu'il était difficile de parvenir à une implémentation sûre et fiable compte tenu de ses spécifications ouvertes et de la quantité de facteurs de sécurité à prendre en compte. 2) OAuth est

liée à un certain nombre d'hypothèses adoptées lors de sa conception : par conséquent, OAuth ne couvre pas toutes les exigences de sécurité des systèmes qui ne sont pas conformes à ces hypothèses. Malheureusement, OAuth n'a pas été conçu pour répondre aux besoins spécifiques de l'IdO. Il ne couvrira donc pas tous ses besoins en matière de sécurité. Par exemple, OAuth suppose que les services à protéger sont mis en œuvre et gérés par la même entité commerciale. En outre, OAuth suppose que le type de ressources à sécuriser est connu au moment de la conception, ce qui n'est bien évidemment pas toujours le cas dans les scénarios IdO. 3) Divulgation des renseignements personnels : OAuth confie aux fournisseurs de services la responsabilité de définir les autorisations que le consommateur peut demander. Cependant, la plupart des fournisseurs définissent uniquement un ensemble de droits d'accès tels que l'accès complet aux données ou en lecture seule. Toutefois, cela va à l'encontre des objectifs de protection de la vie privée. 4) Mise en œuvre lourde, en particulier du côté des fournisseurs de services (SP) : la mise en œuvre d'OAuth du côté des SPs est une tâche complexe, longue et laborieuse sur le plan informatique. De plus, cela implique l'enregistrement des utilisateurs et des applications clients, ainsi que les permissions que l'utilisateur accorde aux applications grand public. Ce processus entrave considérablement l'implémentation des logiques OAuth sur les périphériques contraints.

Nous concluons cette discussion par l'UMA (User-Managed Access) [36] ; ce dernier présente une architecture et un nouveau protocole de délégation de contrôle d'accès. UMA fournit aux utilisateurs une méthode permettant de contrôler l'accès des applications tierces à leurs ressources protégées, via un gestionnaire d'autorisations centralisé qui prend les décisions d'accès en fonction des directives utilisateur. Le protocole UMA utilise actuellement le protocole OAuth V2.0 pour l'interaction entre les entités de l'architecture proposée [37]. Il ajoute deux éléments majeurs : une API de protection formelle présentée par le serveur d'autorisation, et le concept de "demandeur" distinct du propriétaire de la ressource. Plus d'avantages de l'UMA qui sont pertinents : Règles préétablies : Contrairement à OAuth, qui est conçu pour la permission synchrone, UMA permet la permission asynchrone basée sur des règles préétablies. Orienté utilisateur : UMA apporte une approche novatrice en incluant l'utilisateur au cœur de son modèle. Le modèle repose sur l'utilisateur pour attribuer les droits d'accès aux ressources qui peuvent être hébergées sur divers serveurs de ressources. En effet, l'utilisateur a la responsabilité de définir et de configurer ses propres politiques nécessaires pour prendre des décisions de contrôle d'accès. UMA utilise un serveur d'autorisation centralisé qui facilite le partage des données de manière sélective basée sur les instructions de l'utilisateur. Prise en charge du contrôle d'accès basé sur les demandes afin de permettre à la décision d'accès d'être dynamique et de permettre à l'utilisateur d'imposer des conditions contractuelles qui contrôlent les droits d'accès avant d'accorder l'autorisation. Ceci dit, UMA s'appuie sur OAuth et hérite par conséquent de ses inconvénients en matière d'interopérabilité et de sécurité d’implémentation.

3.2. *Systèmes de réputation*

En général, les systèmes de réputation sont divisés en deux grandes catégories : Systèmes de réputation implicites et explicites [12]. La première catégorie concerne les systèmes qui n'ont pas de structure explicite pour exploiter les aspects de la réputation. Aujourd'hui, nous sommes entourés de plusieurs exemples de ce type de systèmes de réputation dans les réseaux sociaux (RSs) tels que Facebook ou LinkedIn. Les utilisateurs de ces RSs peuvent obtenir beaucoup d'informations sur d'autres utilisateurs et ainsi former un certain degré de confiance basé sur les informations recueillies par les amis de leurs amis.

D'autre part, les systèmes de réputation explicites sont ceux qui ont été mis en œuvre intentionnellement pour évaluer et faciliter l'estimation de la confiance entre les membres d'un environnement. Un système de réputation explicite est généralement utilisé dans un environnement qui repose sur une interaction fréquente avec un ensemble suffisamment important et diversifié de membres.

Un exemple de ces systèmes peut être construit à partir de modèles de rétroaction (feedbacks). Une taxonomie plus détaillée est présentée dans [12].

Le tableau 1 résume les principales caractéristiques des systèmes de réputation implicites et explicites.

Systèmes de réputation implicites	Systèmes de réputation explicites
- Ne sont pas volontairement conçus pour mesurer la confiance - S'appuie sur l'analyse du comportement - Exemples : LinkedIn, Facebook, Google, ...	- Volontairement conçus pour être des systèmes de réputation - S'appuient sur les notions de note, classement, récompense, … - Exemples : eBay, Amazon, ...

Tableau 1. *Systèmes de réputation Explicite vs. Implicite*

Notez que le contrôle d'accès basé sur la confiance a été appliqué à l'IdO en utilisant la logique floue (Fuzzy logic) avec la notion de niveaux de confiance pour la gestion des identités [38]. Pour le calcul de la note de confiance, les valeurs linguistiques de l'expérience, des connaissances et des recommandations sont utilisées. L'approche floue présentée pour les calculs de confiance traite de l'information linguistique des dispositifs de contrôle d'accès dans l'IdO. Le résultat de la simulation montre que l'approche floue du contrôle d'accès basé sur la confiance garantit l'évolutivité et l'efficacité énergétique. Il utilise la valeur calculée de la confiance liée aux facteurs comme l'expérience (EX), les connaissances (KN) et la recommandation (RC) en saisissant leurs valeurs vagues. Cependant, tant qu'il n'est pas intégré dans une approche globale auto-optimisée, il reste difficile à gérer, en particulier avec l'augmentation exponentielle du nombre de dispositifs IoT.

4. Notre framework

Dans cette section, nous présentons notre framework et répondons aux problèmes d'approches de CA déconnectés, statiques et centralisés. Cette proposition vise à réunir les paradigmes Blockchain et RL (apprentissage par renforcement) autour du concept d'émergence tout en ajoutant une couche coopérative basée sur la notion de réputation explicite.

4.1. *Énoncé du problème et pistes de recherche*

Aujourd'hui, la plupart des solutions CA sont basées sur le concept d'autorités centralisées auxquelles il faut faire confiance (gouvernements, fabricants, fournisseurs de services, ...) [39]. Cependant, ces entités sont souvent à l'origine de problèmes d'éthique et de protection de la vie privée tels que les accès non autorisés, la collecte et l'analyse des données des utilisateurs, etc. C'est pour cela qu'aujourd'hui, l'utilisation des systèmes décentralisés augmente d'une manière continue [40, 41].

Pratiquement, aucun agent, aussi sophistiqué soit-il, n'est omniscient de son environnement, et encore moins n'a des ressources infinies pour stocker et traiter instantanément tous les données auxquelles il fait face. Parfois, même des liens de communication fiables entre l'autorité centrale et les autres nœuds sont difficiles à établir. La communication peut même s'avérer impossible en raison de l'impossibilité d'un canal/langage technique universellement accepté par les parties prenantes. Celles-ci sont quelques raisons pour lesquelles les approches centralisées peuvent ne pas réussir dans le cas de systèmes à grande échelle. L'exclusion de la faisabilité d'un planificateur central ne laisse que l'option de de laisser les agents finaux se débrouiller eux-mêmes pour s'auto sécuriser [42].

Dans une architecture distribuée, le processus de contrôle d'accès est géré par le composant final lui-même. Cela signifie que chaque dispositif doit être qualifié pour gérer les processus d'autorisation et

disposer de ressources suffisantes pour le faire. Dans notre raisonnement, nous pensons que les caractéristiques des approches distribuées sont primordiales en IdO pour garantir à la fois pour son expansion, sa performance et sa sécurité. Surtout en prenant en compte que les objets augmentent toujours leur capacité de calcul, et du coup qu'il y a de plus en plus d'occasions d'apporter de l'intelligence sur les appareils eux-mêmes. De plus, cette approche présente les avantages suivants [43] :

1) Les objets finaux agissent intelligemment et sont autonomes ;

2) C'est moins cher que de louer une instance Cloud pour chaque objet intelligent ; surtout ceux qui auront besoin d'une connectivité pour une longue durée ;

3) La confiance pourrait être maintenue avec une approche décentralisée plutôt que centralisée puisque les politiques peuvent être définies sans avoir recours à aucune autorité centrale ;

4) Permet à l'information contextuelle de contribuer à la décision d'autorisation.

Cependant, l'extension des composants aussi contraignants d'IdO via l'introduction de la logique et des outils de contrôle d'accès rend la mise en œuvre de cette approche lourde et non pertinente. En effet, pour ce faire, ils doivent pouvoir garantir davantage de capacités de calcul et de stockage ; et c'est pourquoi nous avons opté pour la technologie Blockchain totalement distribuée qui peut répondre efficacement à ce dilemme [44] étant donné qu'elle assure l'aspect décentralisé tout en externalisant la partie calcul des objets IdO vers le réseau.

Ceci dit, certes, les stratégies centralisées sont parfois d'une utilité remarquable. Des fois c'est bon de mettre quelqu'un ou quelque chose en charge. Le problème est que les gens auparavant se sont basés presque entièrement sur les approches centralisées. L'idée de décentralisation était ignorée et/ou négligée. Les solutions centralisées s'était vues comme 'La solution'.

Un autre problème auquel est confronté le CA dans le contexte de l'IdO réside dans la difficulté de gérer 'intelligemment' la politique de sécurité, surtout quand on prend en considération le nombre important de dispositifs censés être contrôlés dans les situations d'IdO. Cela conduit, dans les modèles traditionnels, à adopter des politiques statiques. Le principal inconvénient de cette approche est que la politique ne détecte jamais si elle contient des règles qui génèrent ou contribuent à générer des problèmes de sécurité, qui produisent des conflits ou qui ne sont pas optimales. Cette approche ne prend jamais en considération les conséquences de ses décisions antérieures et ne tire jamais des leçons de ses points forts ni de ses erreurs.

Ce nouveau framework surmonte ces deux contraintes frustrantes en offrant 1) Une infrastructure : une plate-forme Blockchain pour garantir le contrôle d'accès sans faire confiance à aucune entité centrale externe (décentralisation) ; et 2) un CA dynamique, coopératif et auto optimisé basé sur des algorithmes RL et systèmes de réputation afin de gérer intelligemment le nombre énorme de dispositifs IdO et les politiques de contrôle d'accès.

En outre, les trois piliers du framework proposé (Blockchain, réputation et RL) deviennent plus utiles, plus fiables et plus efficaces à mesure que le système qui les englobe s'élargit et engage davantage de nœuds, ce qui en fait les meilleurs candidats dans les environnements IdO.

4.2. *FairAccess*

FairAccess [45] fait référence à un framework de contrôle d'accès distribué préservant la vie privée conçu pour l'IdO, il répond aux besoins de contrôle d'accès en utilisant les mécanismes Blockchain.

La Figure 5 montre le processus de FairAccess : Lorsqu'un sujet *B* veut effectuer une action sur un objet *A*, il soumet sa demande au point de gestion des autorisations AMP (présenté par le wallet) qui agit comme un point d'application de la politique (PEP).

Ce dernier soumet une telle demande sous la forme d'une transaction *GetAcess* et la diffuse sur le réseau Blockchain et attend les mineurs, ces derniers agissent en tant que Point de décision de politique (PDP) distribué, et évalue la transaction. Le PDP vérifie la demande avec la politique définie, en exécutant un SmartContract déjà déployé dans la Blockchain par une transaction précédente appelée *GrantAccess*. L'exécution du SmartContract détermine si la demande doit être autorisée ou refusée. Enfin, si la demande est approuvée, le SmartContract envoie un jeton d'accès à l'adresse du demandeur via une transaction *AllowAccess*.

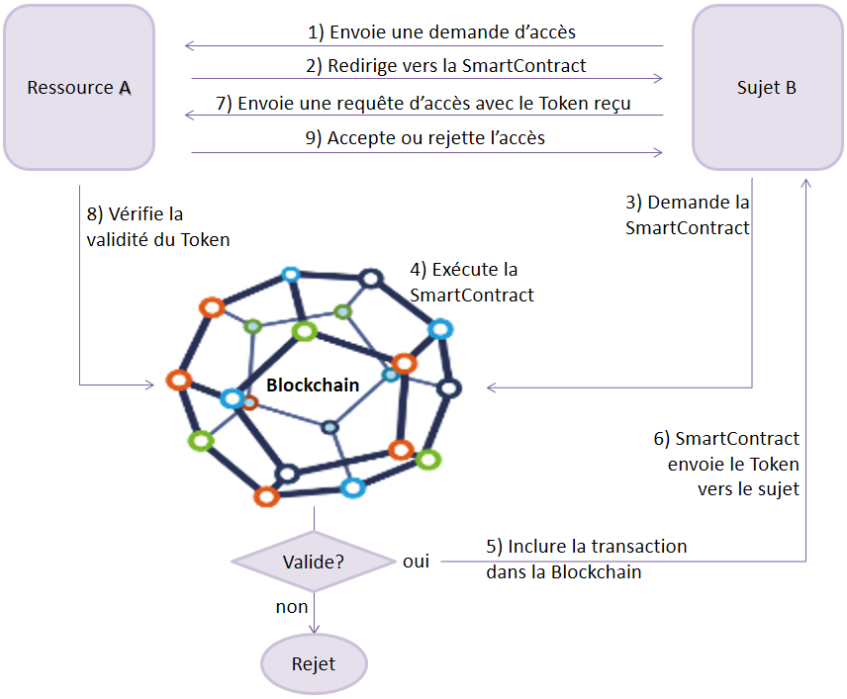

Figure 5. *le processus du rechargement des politiques de contrôle d'accès dans*

FairAccess

Le jeton apparaîtra alors dans la base de données TKN disponible auprès du demandeur.

4.3. *Améliorer/mettre à jour la politique de sécurité avec les algorithmes RL*

Dans notre proposition, le SmartContract représente de la politique du CA ; il est défini par un propriétaire de ressource (Resource Owner - RO) pour un but de gérer les accès à une de ses ressources. Concrètement, SmartContract est un script stocké sur la Blockchain. Puisqu'il vit dans la chaîne, il a automatiquement une adresse unique. Ce SmartContract est déclenché en lui adressant un type de transaction *RequestAccess*, puis il est exécuté indépendamment d'une manière prescrite sur chaque nœud du réseau, selon les données qui ont été incluses dans la transaction de déclenchement. Si les données satisfont les politiques de contrôle d'accès, le SmartContract sera correctement exécuté, puis génère et affecte un jeton d'autorisation (AT) à l'expéditeur de la transaction *RequestAccess*.

Dans les modèles RL standards, un agent est connecté avec son environnement. À chaque étape d'interaction entre les deux, l'agent reçoit en entrée –*i*- et une indication de l'état actuel de l'environnement –*s*- ; l'agent choisit alors une action –*a*- à générer en sortie. L'action change l'état de l'environnement, et la valeur de cette transition d'état est communiquée à l'agent par un signal de renforcement scalaire –*r*-. L'agent doit choisir des actions qui ont tendance à augmenter *la somme à*

long terme des valeurs du signal de renforcement. Il peut apprendre à le faire au fil du temps par essais et erreurs systématiques, guidés par une grande variété d'algorithmes.

L'apprentissage par renforcement est différent du problème plus communément étudié de l'apprentissage supervisé. La différence la plus significative est qu'il n'y a pas de paires entrées/sorties dans le RL ; Au lieu de cela, après avoir choisi une action, l'agent est informé de la récompense immédiate et de l'état suivant, mais il n'est pas informé de l'action qui aurait été dans son meilleur intérêt à long terme. Il est essentiel pour l'agent d'acquérir une expérience utile sur les états possibles/probables du système, les actions, les transitions et les récompenses pour agir de manière optimale. Une autre grande différence par rapport à l'apprentissage supervisé est que la performance en ligne est essentielle en RL étant donné que l'évaluation du système passe simultanément au moment de l'apprentissage.

De plus, ce framework s'appuie sur l'infrastructure entièrement distribuée expliquée dans la section (4.2.) ; d'une part, par l'utilisation du concept de SmartContract qui définit, contient et distribue la politique du CA partout dans le réseau BlockChain et, d'autre part, en sous-traitant la phase de vérification au-delà des nœuds sous contrainte. En outre, tout en exécutant ses taches fonctionnelles d'une manière normale, l'environnement IdO enverra des informations en retour (feedback) après le succès ou l'échec d'une transaction à la Blockchain et mettra ensuite à jour sa politique de sécurité.

Le retour d'information, provenant de chaque transaction, consiste en un 'vecteur d'évaluation' qui évalue les parties prenantes (réputation), leurs actions, le contexte dans lequel la transaction a été réalisée, ainsi que d'autres composantes potentielles. Il peut s'étendre ou se réduire selon les besoins et la criticité du système.

4.4. *Système d'inférence et algorithme*

Pour présenter le système d'inférence de notre framework, nous commençons d'abord par définir ses composantes :

A : Le demandeur qui veut accéder à la ressource (sujet)

B : La ressource ou l'objet auquel le sujet veut accéder/exploiter

Req(A, B) : le sujet *A* envoie une demande d'accès à la cible *B*

SmartContract(B, A, S) : La cible *B* redirige le sujet *A* vers SmartContract *S*

GrantAccess(A, B, T) : Effectue les étapes restantes pour obtenir l'accès (obtenir, envoyer et vérifier la validité du jeton), puis permet à *A* d'accéder à *B*. Il est également responsable de la création de la transaction *T*

AddBC(T) : Ajout de la transaction *T* à la blockchain

Feed(A, B, T) : *A* et *B* envoient des informations en retour sur la transaction *T* ainsi que sur l'évaluation qui servira de note de réputation à la Blockchain. (Ils évaluent la transaction et s'évaluent mutuellement).

Update (B) : Mettre à jour les connaissances de *B* (SmartContract, niveau de crédibilité, réputation, confiance, intégrité,...)

Rejeter (A, B, R) : rejette la demande désormais dénommée *R*, refuse l'accès du sujet *A* à la ressource *B*

Tout d'abord, le RO crée et publie le SmartContract de sa ressource. Le processus se déclenche ensuite par l'envoi d'une demande de *A* à *B*. Cette étape donne naissance à la fonction *SmartContract* englobant *A*, *B* et l'adresse de la politique de CA : *S*.

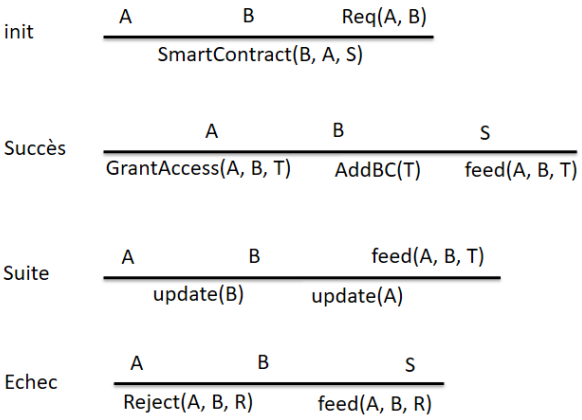

Figure 6. *Système d'inférence de notre framework*

Si l'accès est refusé, un *reject(A, B, R)* accompagné du feedback sera généré ; sinon les autres étapes pour avoir l'accès se suivront, permettant à l'utilisateur *A* d'accéder à *B* puis de générer une transaction *T* qui sera alors ajoutée dans la Blockchain ; sans oublier de transmettre le feedback.

Dans tous les cas, après l'envoi du feedback, les connaissances de *A* et *B* sont mises à jour.

5. Etude de cas

5.1. *Présentation de l'environnement*

Prenons l'exemple d'un parc intelligent. Ce parc se compose d'une multitude de caméras de vidéosurveillance que nous allons représenter par (C_i / $i \in \{1, ..., n\}$). Ces caméras envoient leurs captures en temps réel à un poste de contrôle Org_0 et à un centre de police Org_1.

Les C_i d'une même zone (Z_i) sont reliées entre elles pour former une vue 3D de leur zone et envoyer une vue complète à Org_0 et Org_1.

Le parc est également équipé de centaines de capteurs (capteurs de fumée pour détecter les incendies, capteurs de température, capteurs de mouvement...) et d'actionneurs (alarmes, verrous, injecteurs d'eau en cas d'incendie, ...). Tous ces objets (O_i / i $\in$ {1, ..., m}) sont bien sûr connectés, émettent des données, reçoivent des informations et interagissent avec leur environnement. Toutes ces données sont envoyées à Org_0 et Org_1 (pompiers).

Certains capteurs et actionneurs peuvent être utilisés par des visiteurs spécifiques selon un système de privilège intégré dans leur carte à puce.

Certains capteurs situés dans des zones critiques sont capables de communiquer avec les caméras de leurs zones pour fournir une vue plus complète.

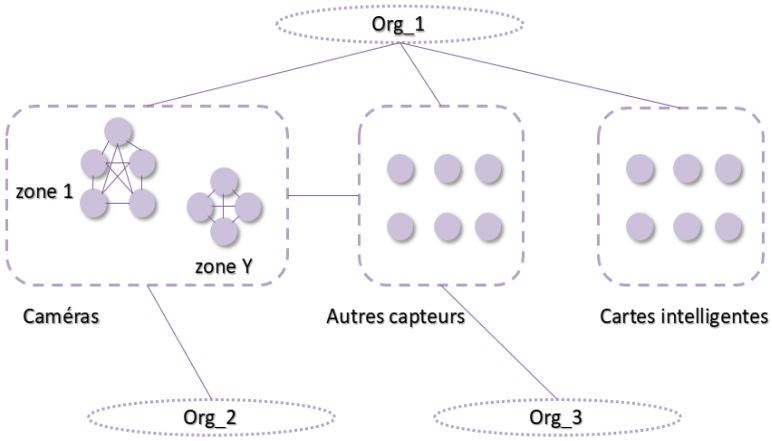

Figure 7. *Présentation générale de l'étude de cas*

Pour accéder au smart parc, celui-ci dispose d'un système de billetterie électronique en ligne que les visiteurs consultent pour acheter un billet électronique puis le téléchargent sur leur smartphone afin de l'exposer à une machine de vérification installée à l'entrée du parc. La machine leur fournit en retour une carte à puce avec une période de validité et un système de géolocalisation embarqué. La figure 7 au-dessus résume les liens entre les différentes entités de cet environnement typique proposé dans cette étude de cas.

5.2. *Fonctionnement*

Le fonctionnement ainsi que la procédure de prise de décision sur laquelle nous avons bâti notre framework sont détaillés dans l'organigramme de la Figure 8.

En projetant ceci sur notre étude de cas, la procédure sera comme suit : Au tout début, Org_i, en tant que propriétaire d'objets, doit d'abord créer les SmartContracts et les publier dans le réseau Blockchain.

Les visiteurs jouent le rôle des sujets et sont présentés par (S_i/ $i \in \{1, ..., p\}$). Supposons maintenant que les visiteurs S_1, ..., S_m aient accès aux objets O_1, ..., O_m selon des règles définies dans les SmartContracts.

En utilisant les algorithmes RL, la politique de CA ainsi que les dispositifs intelligents qui sont entrainés à prendre des décisions adéquates en suivant cette procédure : le SmartContract présentant la politique de contrôle d'accès de l'objet intelligent est exposé à un environnement où il s'entraîne continuellement par essais et erreurs. Par conséquent, ce SmartContract tire des leçons des expériences passées et tente de capter les meilleures connaissances possibles pour prendre des décisions plus pertinentes.

Suivant le même principe, les informations de retour (qui peuvent être représentées par un vecteur avec *k* nombre de composants) contiennent également les informations servant à évaluer les parties prenantes impliquées dans l'opération en question. Le feedback peut être binaire (zéro ou un) ou avec un niveau/poids de satisfaction (confiance, intégrité...) entre zéro et un.

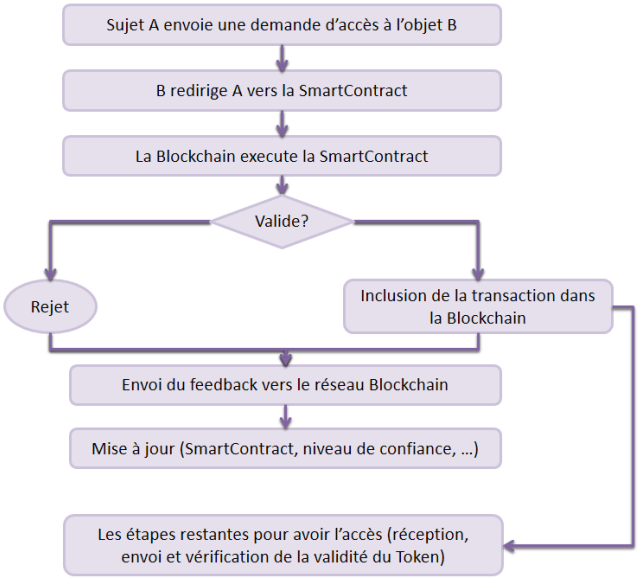

Figure 8. *Organigramme du framework proposé*

5.3. *Evaluation*

Supposons qu'après chaque utilisation d'un objet o_i ($i \in \{1, ..., 50\}$) par le visiteur s_i (un sujet légitime ($i \in \{1, ..., 10\}$), une dégradation de l'objet o_i est remarquée, le retour provenant de o_i sera soit nul soit faible sur l'échelle de crédibilité donnée à s_i. L'algorithme déterminera, surtout si cette situation se répète, que s_i doit être détaché des utilisateurs légitimes autorisés à accéder à o_i. Ainsi, la réputation de s_i va diminuer et la politique de sécurité va se mettre à jour.

Dans un autre scénario, un objet o_i se heurte de façon répétée à des problèmes lorsqu'il est utilisé dans un contexte donnécx_i. Les faibles évaluations provenant de plusieurs transactions qui n'ont rien en commun sauf cx_i conduisent à condamner ce dernier ; et de ce fait à mettre à jour le SmartContract en refusant l'utilisation de l'objet o_i dans le contextecx_i. Là encore, les retours d'expérience recueillis diminueront le niveau de confiance de ce contexte, et si d'autres objets rencontrent des difficultés dans le même contexte, avec le temps, les acteurs du système n'auront plus confiance en celui-ci.

Notez que notre framework n'est pas limité, ni à un modèle de contrôle d'accès spécifique, ni à un système de réputation spécifique. La seule recommandation est d'utiliser un système de réputation explicite étant donné l'utilisation des feedbacks.

Pour cette raison, nous n'avons pas imposé plus d'exigences sur le vecteur *feedback*, il peut inclure le contexte comme nous l'avons vu (par exemple OrBAC, SmartOrBAC), mais aussi les attributs (ABAC), le niveau de crédibilité (I-OrBAC [46]), ainsi que les autres systèmes de notation.

Suivant le même raisonnement, nous n'avons pas spécifié un algorithme d'apprentissage automatique (ML) précis mais une catégorie d'algorithmes ML, à savoir les algorithmes RL, étant donné notre objectif d'introduire un apprentissage en ligne, auto-ajusté et auto-optimisé. Cela dit, chacun peut choisir les algorithmes à implémenter en fonction de ses besoins et de ses intentions.

6. Conclusions & Travaux à venir

Répondre aux besoins de sécurité dans les environnements IdO est un sujet de préoccupation pour la communauté scientifique. Malheureusement, les solutions de sécurité standards utilisées dans les situations IT traditionnelles sont, dans de nombreux cas, obsolètes. Elles ne peuvent pas être appliquées machinalement à l'IdO étant donné ses contraintes intrinsèques : hétérogénéité, capacités

limitées de stockage et de calcul ainsi que le grand nombre de dispositifs à gérer dans les contextes de l'IdO.

Depuis longtemps, les responsables de la sécurité ont vu dans le grand nombre de dispositifs que l'IdO doit orchestrer un obstacle et ont essayé de s'en débarrasser par différentes méthodes. Toutefois, dans le présent document, nous avions traité le problème sous un angle différent. En effet, le but principal de notre proposition est d'utiliser le pouvoir du grand nombre d'entités qui coopèrent entre elles.

Ce travail s'est concentré sur le contrôle d'accès dans les environnements d'Internet des objets. En gros, nous avons proposé est un framework remédiant à deux problèmes frustrants : Les architectures centralisées, qui ne sont absolument pas adaptées à l'IdO, et l'immense nombre de politiques de contrôle d'accès que les responsables sécurité doivent gérer. Le nouveau framework apporte des réponses pertinentes à ces problèmes. Il donne aux propriétaires le contrôle total de leurs appareils IdO sans les obliger à faire confiance à une entité externe ; de plus, il leur fournit une politique de sécurité dynamique et auto-améliorée. Notre proposition est basée sur : 1) Le concept du Blockchain pour garantir le contrôle d'accès sans faire confiance aux entités centrales externes, 2) Renforcer les outils d'apprentissage afin d'offrir une politique de sécurité dynamique, optimisée et auto-ajustée, et 3) des systèmes de réputation explicites basés sur les feedbacks. Par ailleurs, la robustesse de ces trois piliers s'acquiert du fait qu'ils sont utilisés par un grand nombre d'entités, en coordination et en coopération les unes avec les autres. C'est pourquoi nous les avons regroupées le tout sous le concept d'émergence.

La proposition formulée dans le présent papier fait face encore à certaines limites auxquelles nous remédierons dans le cadre d'une extension future. En effet, nous devons concrétiser très prochainement notre framework en mettant en place une étude de cas pratique, afin de nous permettre de tester son efficacité. En outre, la technologie Blockchain présente également certains inconvénients intrinsèques en termes de respect de la vie privée (les SmartContracts sont exposés publiquement). Le système de rétroaction que nous utilisons pour bâtir notre réputation et notre RL doit également être testé dans le monde réel de l'IdO afin d'analyser la consommation d'énergie, d'énergie et de stockage.

References

[1] J. Gubbi et al. 2013. “Internet of Things (IoT): A vision, architectural elements, and future directions”, *Future Generation Computer Systems*, Volume 29, Issue 7, P. 1645-1660.

[1a] I. Saleh. “Internet des Objets (IdO) : Concepts, Enjeux, Défis et Perspectives”. Revue Internet des objets. 2. 10.21494/ISTE.OP.2018.0229.

[2a] J. Lopez, R. Rios, F. Bao and G. Wang. 2017. “Evolving Privacy: From Sensors to the Internet of Things”, p. 1.

[2b] I. Saleh. “Les enjeux et les défis de l’Internet des Objets (IdO)”. Revue Internet des objets. 17. 10.21494/ISTE.OP.2017.0133.

[2c] N. Bouhaï, I. Saleh. « Internet des objets, Evolutions et innovations » ISBN 9781784052713, Volume 1, ISTE, 2017, 238 pages.

[3] Kim Rowe. February 2016. “Internet of things requirements and protocols”. *IEEE standards university magazine*.

[4] M. A. Khan, K. Salah, 2018. “IoT security: Review, blockchain solutions, and open challenges”. Future Generation Computer Systems 82. pp. 395–411.

[5] E. N. Zalta, ed. 2012. "Emergent Properties". The Stanford Encyclopedia of Philosophy, Spring Edition.

[6] S. Nakamoto. 2008. “Bitcoin : A Peer-to-Peer Electronic Cash System,” pp. 1–9.

[7] S. Wilkinson, J. Lowry, and T. Boshevski. 2014. “Metadisk a Blockchain-based decentralized file storage application”.

[8] A. Schaub, R. Bazin, O. Hasan, and L. Brunie. 2016. “A trustless privacy-preserving reputation system,” IFIP Int. Inf.

[9] C. Fromknecht, D. Velicanu, and S. Yakoubov. 2014. “A Decentralized Public Key Infrastructure with Identity Retention” IACR Cryptol. ePrint.

[10] T. O. Ayodele. 2010. “Introduction to Machine Learning”, in New Advances in Machine Learning. Rijeka, Croatia: InTech.

[11] Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin. 2012. “Learning From Data”, AMLBook.

[12] F. Hendrikx et al. 2015. “Reputation system: A survey and taxonomy”, J. Parallel Diist. Compt. 75, pp. 184-197.

[13] A. J. Bidgoly, B. T. Ladani, 2016. “Benchmarking reputation systems: A quantitative verification approach”. Computers in Human Behavior 57. 274-291

[14] “Part 1: Introduction and general model,” in Common Criteria for Information Technology Security Evaluation Version 2.1, p. 11.

[15] R. S. Sandhu. 1998. “Role-based Access Control,” Adv. Comput., vol. 46, pp. 237–286.

[16] Z. Guoping and T. Jiazheng. 2010. “An extended role based access control model for the Internet of Things”. *Information Networking and Automation (ICINA), International Conference on IEEE*, p. V1-319-V1-323.

[17] P. Spiess, S. Kamouskos, et al. 2009. "SOA-based Integration of the Internet of Things in EnterpriseServices". *IEEE International Conference on Web Services*, pp. 968-975.

[18] L. Moreira Sa de Souza, et al.,"SOCRADES:A web Service Based Shop Floor Integration Infrastructure," C.Floerkemeieretal. (Eds.): IOT200S, LNCS4952, pp.50—67.

[19] J. Jia, X. Qiu, C. Cheng. 2012. “Access control method for web of things based on role and sns”, Computer and Information Technology (CIT), *IEEE 12th International Conference on IEEE*, p. 316-321.

[20] A. A. E. Kalam el al. 2003. “Organization based access control,” in Proceedings POLICY 2003. *IEEE 4th International Workshop on Policies for Distributed Systems and Networks*, pp. 120–131.

[21] A. Abou El Kalam, Y. Deswarte, A. Baïna, and M. Kaâniche. 2009. “PolyOrBAC: A security framework for Critical Infrastructures,” Int. J. Crit. Infrastruct. Prot., vol. 2, no. 4, pp. 154–169.

[22] A. Ouaddah, I. Bouij-Pasquier, A. Abou Elkalam, and A. Ait Ouahman. 2015. “Security analysis and proposal of new access control model in the Internet of Thing”. *International Conference on Electrical and Information Technologies (ICEIT)*, pp. 30–35.

[23] I. Bouij-Pasquier, A. A. El Kalam, A. A. Ouahman, and M. De Montfort. 2015. “A Security Framework for Internet of Things,” Springer International Publishing, pp. 19–31.

[24] E. Yuan and J. Tong. 2005. “Attributed based access control (ABAC) for Web services,” in *IEEE International Conference on Web Services (ICWS’05).*

[25] Webfarmr.eu. 2011. “XACML 3.0 enhancements,” Nanoscale Res. Lett., vol. 6, no. 1, p. 297.

[26] L. Seitz, G. Selander, and C. Gehrmann. 2013. “Authorization framework for the Internet-of-Things”. IEEE 14th Int. Symp. a World Wireless, Mob. Multimed. Networks, WoWMoM.

[27] J. Dennis and E. Van Horn. (1966). Programming semantics for multiprogrammed computations, Commun. ACM 9(3), pp. 143–155.

[28] MAHALLE, Parikshit N., ANGGOROJATI, Bayu, PRASAD, Neeli R., et al. 2013. Identity authentication and capability based access control (iacac) for the internet of things. Journal of Cyber Security and Mobility, vol. 1, no 4, p. 309-348.

[29] J. Park and R. Sandhu. 2002. “Towards usage control models: beyond traditional access control,” in Proceedings of the seventh ACM symposium on Access control models and technologies - SACMAT ’02, p. 57.

[30] A. Lazouski, F. Martinelli, and P. Mori. 2010. “Usage control in computer security: A survey,” Comput. Sci. Rev., vol. 4, no. 2, pp. 81–99.

[31] X. Zhang, M. Nakae, M. J. Covington, and R. Sandhu. Feb. 2008. “Toward a Usage-Based Security Framework for Collaborative Computing Systems,” ACM Trans. Inf. Syst. Secur., vol. 11, no. 1, pp. 1–36.

[32] D. H. (ed). October 2012. “The OAuth 2.0 Authorization Framework,” IETF, RFC6749. Available at http://www.rfc-editor.org/rfc/rfc6749.txt.

[33] CIRANI, Simone, PICONE, Marco, GONIZZI, Pietro, et al. 2015. IoT-OAS: An OAuth-Based Authorization Service Architecture for Secure Services in IoT Scenarios. Sensors Journal, IEEE, vol. 15, no 2, p. 1224-1234.

[34] Hannes Tschofenig. 2015. “The OAuth 2.0 Bearer Token Usage over the Constrained Application Protocol (CoAP)” IETF Internet Draft, draft-tschofenig-ace-oauth-bt-01.txt

[35] H. Tschofenig. 2014. “The OAuth 2.0 Internet of Things (IoT) Client Credentials Grant” IETF Internet Draft, draft-tschofenig-ace-oauth-iot-00.txt.

[36] UMA Core Protocol Version 1.0, https://kantarainitiative.org/confluence/display/uma/UMA+1.0+Core+Protocol.

[37] Hardjono, T., Maler, E., Machulak, M., and D. Catalano. February 2015. "User-Managed Access (UMA) Profile of OAuth 2.0", draft-hardjono-oauth-umacore-12 (work in progress).

[38] P. N. Mahalle, P. A. Thakre, N. R. Prasad, and R. Prasad. 2013. “A fuzzy approach to trust based access control in internet of things,” in Wireless VITAE 2013, pp. 1–5.

[39] A. Ouaddah, H. Mousannif, A. A. Elkalam, A. Ait Ouahman. 2017. “Access control in the Internet of Things: Big challenges and new opportunities”, Computer Networks 112, pp. 237–262

[40] Lukas Esterie, “Centralized, Decentralized and Self-Orgainsed Coverage Maximisation in Smart Camera Networks,” in 11th IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO), September 2017, pp.28-34.

[41] C. Dukkipati, Y. Zhang, L.C. Cheng. 2018. “Decentralized, BlockChain Based Access Control Framework for the Heterogeneous Internet of Things” ABAC'18.

[42] A. Masoud, 2007. “Decentralized Self-Organizing Potential Field-Based Control for Individually Motivated Mobile Agents in a Cluttered Environment: A Vector-Harmonic Potential Field Approach”. IEEE Transactions on Systems, Man, and Cybernetics - Part A. pp. 372 – 390

[43] A. Ouaddah. 2017. “FairAccess: A privacy-preserving access control framework based on the Blockchain technology to secure the Internet of Things”, thesis.

[44] A. Ouaddah, H. Mousannif, et al. 29 September – 1 October 2016. Access Control in IoT: Survey & State of the Art *In the Proceeding of the 5th International Conference on Multimedia Computing and Systems (ICMCS'16).* Marrakech, Morocco.

[45] A. Ouaddah, A. Abou Elkalam and A. Ait Ouahman. 2017. “FairAccess: a new Blockchain-based access control framework for the Internet of Things”, Security and Communication Networks, pp. 1-22.

[46] A. Ameziane El Hassani et al. 2014. “Integrity-OrBAC: a new model to preserve Critical Infrastructures integrity”, Int. J. Inf. Secur, Springer-Verlag Berlin

