الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université de 8 Mai 1945 – Guelma -
Faculté des Mathématiques, d'Informatique et des Sciences de la matière
Département d'Informatique

Mémoire de Fin d'études Master

Filière : Informatique

Option : Sciences et technologie de l information et de communication

Thème :

Conception d'un schéma d'authentification sécurisé pour l'internet des drones

Encadré Par :
Dr. Ferrag Med Amine

Présenté par :
Aouamri aymen

Septembre 2021

Remerciements

Tout d’abord je tiens à remercier Dieu,

Le tout puissant et miséricordieux, qui m’a donné la force,

L’intelligence et la patience d’accomplir ce modeste travail.

Un grand remerciement à Monsieur Ferrag Mohamed amine pour son encadrement.

Il était toujours montré à l’écoute et disponible tout au long de la réalisation de ce mémoire,

Ainsi pour l’inspiration, l’aide et le temps qu’il a bien voulu me consacrer, je te dis merci.

Des remerciements à ma famille et mes amis pour leurs encouragements,

Grâce à eux j'ai pu surmonter tous les obstacles.

Enfin, mes vifs remerciements vont également aux membres de jury

Pour l’intérêt qu’ils ont porté à mon projet en acceptant d’examiner et juger mon travail.

Dédicace

Je tiens en tout premier lieu à remercier Allah.

Je voudrais dédier ce travail :

A ma très chère mère ***seridi farida***

La source de tendresse et l'exemple du dévouement qui n'a pas cessé de m'encourager et de prier pour moi. Ta prière et ta bénédiction m'ont été d'un grand secours pour mener à bien mes études. Aucune dédicace ne saurait être assez éloquente pour exprimer ce que tu mérites pour tous les sacrifices que tu n'as cessé de me donner depuis ma naissance, durant mon enfance et même à l'âge adulte. Tu as fait plus qu'une mère puisse faire pour que ses enfants suivent le bon chemin dans leur vie et leurs études. Je te dédie ce travail en témoignage de mon profond amour. Puisse Dieu, le tout puissant, te préserver

et t'accorder santé, longue vie et bonheur.

A mon Encadreur Ferrag Mohamed amine.

A mes camarades et Mes collègues

Merzougi salah et **cherifi raouf** et **bensalh hazem** Qui m'ont aidé

Dans la réalisation de ce travail

Et à tous ceux que j'aime et à toutes les personnes

Qui m'ont prodigué des encouragements et se sont donné la peine de me soutenir durant Cette formation.

Enfin je le dédie aux enseignants du département et mes collègues d'informatiques.

Résumé

Les drones sont devenus un média important dans notre monde d'aujourd'hui, et sur cette base, l'Internet des aéronefs (IOD) est devenu une approche pour réguler ce nouveau domaine, De plus, les entreprises commerciales s'orientent vers l'adoption des drones comme moyen de transport et s'appuie sur eux dans des domaines civils ou militaires importants, Mais les fondements de la sécurité doivent être fournis pour qu'elle soit fondamentalement adoptée par les autorités officielles,pour assurer à la fois la sécurité et la confidentialité de la transmission des données, une réglementation stricte du mouvement des drones et des décisions concernant la sécurité des drones et la sécurité des missions sont les bases sur lesquelles nous avons proposé un schéma d'authentification IOD basé sur Fog et Blockchain , Ce système est basé sur des étapes fondamentales : initialisation, enregistrement, authentification, consensus et mise à jour du certificat. Ces bases contribuent à maintenir la confidentialité, la vie privée, l'intégrité et l'anonymat. Sur la base du schéma proposé nous avons simulé AVISPA, ce qui a prouvé l'efficacité du schéma par rapport aux autres schémas dans certains aspects.

Mots clés : sécurité, authentification, Fog et cloud, cryptage.

Abstract

Drones have become an important tool in our world today, and it is on that basis that the Internet of drone has become an elaborate approach to organizing this new section, In addition, commercial companies are moving towards adopting drones as a means of transportation and relying on them in important civilian or military fields, But the basic foundations of security must be provided in order for it to be basically adopted by the official authorities, and providing both security and privacy in data transmission, tight regulation of drone movement, and decisions regarding drone security and mission security are the basis for which we have proposed an IOD authentication scheme based on **Fog** and **Blockchain**, This system is based on basic stages: initialization, registration, authentication, consensus and certificate update. These foundations contribute to maintaining confidentiality, privacy, integrity and anonymity, Based on the proposed scheme, we simulated AVISPA, which proved the effectiveness of the scheme compared to other schemes in certain aspects.

Keywords: security, authentication, fog and cloud computing, encryption.

الملخص

إن الطائرات بدون طيار اصبحت وسيلة ذات اهمية في عالمنا اليوم و على هذا الأساس أصبحت أنترنت الطائرات**IOD** منهج محكم لتنظيم هذا الفرع الجديد من التكنولوجيا إضافة لذلك تتجه الشركات التجارية نحو اعتماد الطائرات بدون طيار كوسيلة للنقلوالاعتماد عليها في مجالات مدنية أو عسكرية هامة ، لاكن لابد من توفير اسس الأمن الأساسية حتى تعتمد اساسيا من طرف الجهات الرسمية، إن توفير كلمن الامن و الخصوصية في نقل البيانات و اتخاذ تنظيم محكم في تنقل الطائرات و اتخاذ قرارات تخص أمن الطائرات و أمن المهمات التي تقوم بها الأساس الذي من أجله قمنا باقتراح مخطط مصادقة **IOD** مبني على أساس**Fog**و**Blockchain** , إن هذا النظام مبني على مراحل أساسية :التهيئة ،التسجيل ،المصادقة،الإجماع وتحديث الشهادة هذه الأسس تساهم في الحفاظ على السرية الخصوصية و النزاهة و إخفاء الهوية , ،استناد للمخطط المقترح قمنا بمحاكاة على AVISPAالذي قام بإثبات فعالية المخطط مقارنة مع مخططات الأخرى من جوانب محددة .

الكلمات المفتاحية : الأمن، المصادقة، الحوسبة الضبابية و السحابية، التشفي

Table des matières

Table des figures

Liste des tableaux

Introduction Générale

L'Internet des drones (IoD) a été largement utilisé dans divers domaines et apporte un grand confort à la production et à la vie des gens grâce à leurs types de capteurs. Par exemple elle a été utilisée dans la reconnaissance militaire, le transport logistique et les secours en cas de catastrophe.

Les capteurs intégrés au drone peuvent collecter et analyser les phénomènes physiques (par exemple, l'humidité, la température, la pression atmosphérique, pression atmosphérique), et la caméra et le microphone intégrés peuvent transmettre des vidéos au contrôleur via une technique de communication sans fil (Wi-Fi, Bluetooth, etc.). Ainsi, le contrôleur peut obtenir des informations en temps réel. Les données collectées par les capteurs embarqués dans les drones sont confrontées à de nouveaux défis de sécurité et de respect de la vie privée avec l'évolution de la technologie au fil du temps. Les données collectées peuvent contenir des informations très sensibles par exemple les informations militaires, Plusieurs facteurs doivent être pris en compte lors de la conception d'un schéma, en raison des ressources limitées des drones pour cela on utilise le concept de calcul du Fog qui a été intégré. Il offre la possibilité de collecter, traiter, organiser et stocker des données de trafic en temps réel tout en améliorant l'efficacité de la communication et en minimisant la latence ,Ainsi la technologie du Blockchain a été adaptée comme solution réalisable pour garantir les objectifs de sécurité , Ce travail a donc été réalisé pour concevoir un schéma d'authentification mutuelle basé sur la technologie Blockchain et le principe de calcul du Fog , l'évaluation du programme ont été effectuées à l'aide de l'outil AVISPA .

CHAPITRE I

Internet Des Drones (IoD)

Chapitre I

Internet drones(IoD)

Dans ce chapitre, nous introduisons les réseaux d'internet des drones et de ses différents concepts: technologies, applications, motivations, Ensuite, nous pouvons savoir la classification des véhicules aériens sans pilote.

1.1. Définition

L'Internet des drones (loD) peut être défini dans une architecture pour assurer le contrôle et l'accès entre les drones et les utilisateurs sur Internet. En effet, les drones sont de plus en plus produits de base largement disponibles sur le marché, permettant ainsi son utilisation par tout utilisateur pour effectuer diverses missions en utilisant plusieurs drones dans un espace aérien contrôlé.[1]

1.2. Technologies et applications

3.1.1 Technologies

5G : Avantages et solutions sous-jacentes

Le développement de la cinquième génération d'Internet (5G) a largement contribué au développement de (loD), la (5G) joue un rôle très important dont les drones sont censés faire partie. Par exemple, il existe plusieurs cas d'utilisation de la 5G dans lesquels les drones sont utilisés, tels que les compteurs intelligents, l'agriculture intelligente, fabrication à distance, formation à distance, application industrielle et contrôle, ville intelligente, etc.[2]

La figue 1 montre la révolution de la technologie (5G) à partir de le (1G).

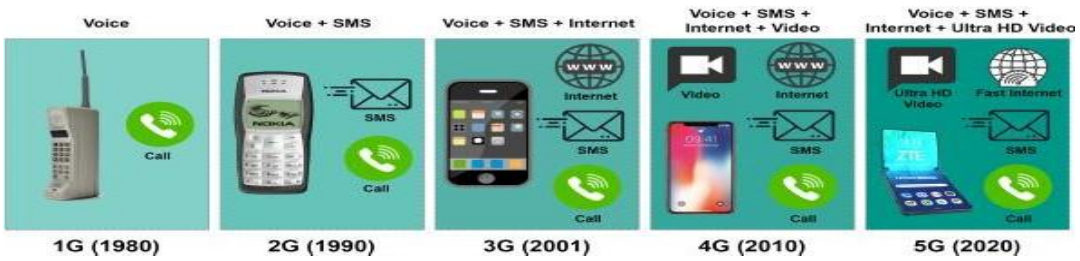

Figure 1 La révolution de la technologie 5G [2]

la (5 G) vient fournir un internet ultra-rapide et un multimédia. Les caractéristiques de la 5G sont les suivantes: la sécurité et la fiabilité du réseau, la formation de faisceaux et des cellules, l'infrastructure en nuage pour améliorer l'efficacité et la facilité de maintenance. L'environnement (loD) basé sur la blockchain est activé par la 5G [2]

Les principaux indicateurs de performance du réseau 5G sont les suivants : [3]

- ***Accessibilité*** : qualité de service élevée en permanence, notamment aux heures de pointe.
- ***La sécurité*** : Protection des données privées des utilisateurs, comme les identifiants et les emplacements des clients.
- ***Disponibilité*** : Assurer une couverture en cas de catastrophe et soutenir une excellente couverture partout.
- ***Mobilité*** : Service d'assistance pour les scénarios de déplacement (où le client est en mouvement) et les scénarios de mobilité au ralenti (où le client est stable).
- ***La pérennité*** : Disponibilité des services aussi longtemps que le client en aura besoin.

Technologie des drones moderne

Nous mentionnons maintenant différents types qui ont pris la tête des drones dans années 2021

Table 1 montre les 10 meilleurs drones dans année 2021 avec sa caractéristique [4]

Nom	Poids	Dimensions	Contrôleur	Duréede vie de la batterie:	Gamme maximale	Vitesse maximale	Autrescaractéristiques
DJI Mavic 2 Zoom	905g	214×91×84 mm	Yes	31 minutes (3850mAh)	8km / 5mi	72kph / 44.7mph	**(+)Très pratique (+)Grandes fonctionnalités du logiciel (-)Coûteuse**
Autel EVO II	1174g	397×397mm	Yes	40 minutes	9km / 5.5mi	72kph / 44mph	**(+)Détecteurs omnidirectionnels (-)La prise de vue 8K est limitée à 25 images par seconde**

DJI Mavic 2 Pro	907g	214×91×84 mm	Yes	31 minutes (3850mAh)	8km / 5mi	72kph / 44.7mph	**(+)Capteur 1 pouce (+)Cellule fiable (+)Grandes fonctionnalités du logiciel (-)Coûteuse**
PowerVision PowerEgg X Wizard	860g	178 x 102 x 102mm	Yes	30 minutes	6 km / 3.7mi	65kph	**(+)Résistance à l'eau (+)synchronisation audio (-)Pas de bouton d'enregistrement en mode caméscope**
PowerVision PowerEgg X Explorer	860g	178 x 102 x 102mm	Yes	30 minutes	6 km / 3.7mi	65kph / 40mph	**(+)synchronisation audio (-)Pas de bouton d'enregistrement en mode caméscope**
DJI Mini 2	249g	38×81×58mm	Yes	31 minutes (2250mAh)	10km / 6.2 miles	57kph / 35.7mph	**(+)Très pratique (+)Facile à piloter (-)Fonctions de suivi limitées**
Parrot Anafi FPV	310g	244×67×65 mm	Yes	25 minutes (2700mAh)	4km / 2.5mi	55kph / 35mph	**(+)Très pratique (+)rotation verticale de 180° etzoom (-)Contrôle à 2 axes seulement**
DJI Mavic Air 2	570g	180×97×84 mm	Yes	34 minutes (3500mAh)	18.5km / 11.4mi	68kph / 42mph	**(+)Portable (+)Évitement d'objets avec correction de trajectoire (-)Pas de capteurs latéraux ou supérieurs**
DJI Phantom 4 Pro V2.0	1375 g	350x350xmm	Yes	25 minutes (5870mAh)	7km / 4.1mi	72kph / 44.7mph	**(+)Design classique (+)Suivi des objets (-)La taille est un peu lourde**
DJI Inspire 2	4000 g	605 diagonal mm	Yes	23-27 minutes (4280mAh dual battery)	7km / 4.1mi	94kph / 58mph	**(+)Système d'objectif interchangeabledisponible (+)Capable de diffuser en direct1080i (-)Coûteuse**

3.1.2 Applications

Les drones sont trouvés dans plusieurs nouvelles utilisations pour améliorer notre mode de vie. Il y a de nombreuses applications pour la technologie des drones la ***figure 2.*** Montre quelques domaines d'application des drones.

Et puisque les domaines des drones sont les mêmes avec (loD), nous montrons les différents domaines d'application des drones

[1. Surveillance des frontières;

2. Recherche et sauvetage;

   a. Accidents d'avions;

   b. Naufrages;

3. Détection des feux de forêt;

4. Relais de communication;

5. Application de la loi;

   a. Contrôle de la foule;

   b. Détection de trafic de drogue;

6. Gestions d'urgences et de désastres;

   a. Tremblement de terre;

   b. Inondations;

   c. Catastrophe nucléaires;

   d. Déversements de pétrole;

7. Recherche scientifique;

   a. Environnementale;

   b. Atmosphérique;

   c. Archéologique;

   d. Pollution;

   e. Recensement de la faune;

   f. Étude des glaces;

8. Application dans l'industrie;

   a. Épandage des récoltes;

Figure 2 les domaines d'application des drones

b. Cinéma;

c. Photo aérienne, cartographie

d. Reportage pour les canaux de nouvelles;

e. Publicité aérienne;

f. Transport de fret;

g. Sécurité;

h. Surveillance des centrales nucléaires, ou des pipelines.] [5]

9. Modéliser les sols et les bâtiments en3D

10. Diffuser le wifi

1.3. Motivations

Le système de livraison par drone actuellement déployé employé pour livrer de la Nourriture, des colis, des médicaments, une aide d'urgence dans les zones inondées, et ainsi de suite. Ils collectent des données telles que (images ou vidéos), la surveillance du trafic, de la foule, de l'environnement, etc. Le système de livraison par drone réduit le temps de livraison des paquets, la consommation de carburant et d'énergie en utilisant la batterie par rapport aux véhicules qui se déplace avec de l'essence.

Les services de messagerie et les fournisseurs de services de livraison se partagent également le commerce du détail en ligne (par exemple, Amazon et DHL ont commencé à livrer des articles à leurs clients). Amazon met en place "Amazon prime Air" pour assurer des livraisons à l'aide de drones connus sous le nom"d'octocoptères".

Ces dernières années, un service de livraison par drone a été mis en place à Londres en raison de la demande de la population, qui peut permettre d'échanger des colis pesant jusqu'à 500 g. En outre, l'entreprise allemande de livraison express, DeutschePost (DHL) utilise également les drones, appelés "parcelcopters" pour la livraison d'urgence (par exemple, des marchandises hautement prioritaires comme médicaments dans des zones reculées).

La sécurité joue un rôle très important dans l'environnement IoD. Cela conduit à concevoir un mécanisme sécurisé de livraison et de collecte de données avec l'aide des drones déployés. Cependant, nous devons maintenir plusieurs exigences de sécurité, telles que

La Confidentialité, l'authentification, contrôle d'accès, non-répudiation, disponibilité, fraîcheur, etc.

En dehors de ces exigences, nous sommes également confrontés à divers défis de sécurité dans un environnement IoD, notamment le "détournement à distance d'un drone", la "confidentialité", le "rejet et l'intrusion humaine", "l'attaque par usurpation d'identité", "l'initié privilégié", "attaque par capture physique du drone", etc. Il est donc essentiel de garder à l'esprit que le protocole de sécurité conçu doit être résistant à de telles attaques

1.4. Classification des véhicules aériens sans pilote (UAV)

Pour mettre en place des plateformes de communication aérienne, la tâche la plus difficile est de choisir le type de drones à déployer. Le type de drone sélectionné doit répondre à diverses exigences, telles que la qualité de service, la capacité énergétique, l'environnement, la sécurité et la fiabilité exigences et les réglementations fédérales. [6]

Différentes caractéristiques sont prises en compte pour classer les drones, telles que l'altitude opérationnelle, le poids au décollage, la propriété, la méthode de lancement, la classe d'espace aérien et le niveau de sécurité, la propriété, la méthode de lancement, la classe d'espace aérien et le niveau d'autonomie du contrôle . Sur la base de ses caractéristiques, les UAV peuvent être classé en trois catégories :[6]

1. les plateformes à basse altitude(LAP).
2. les plateformes à haute altitude(HAP).
3. les satellites.

La figue 3 : montre la catégorisation et les différents cas d'utilisation des drones.

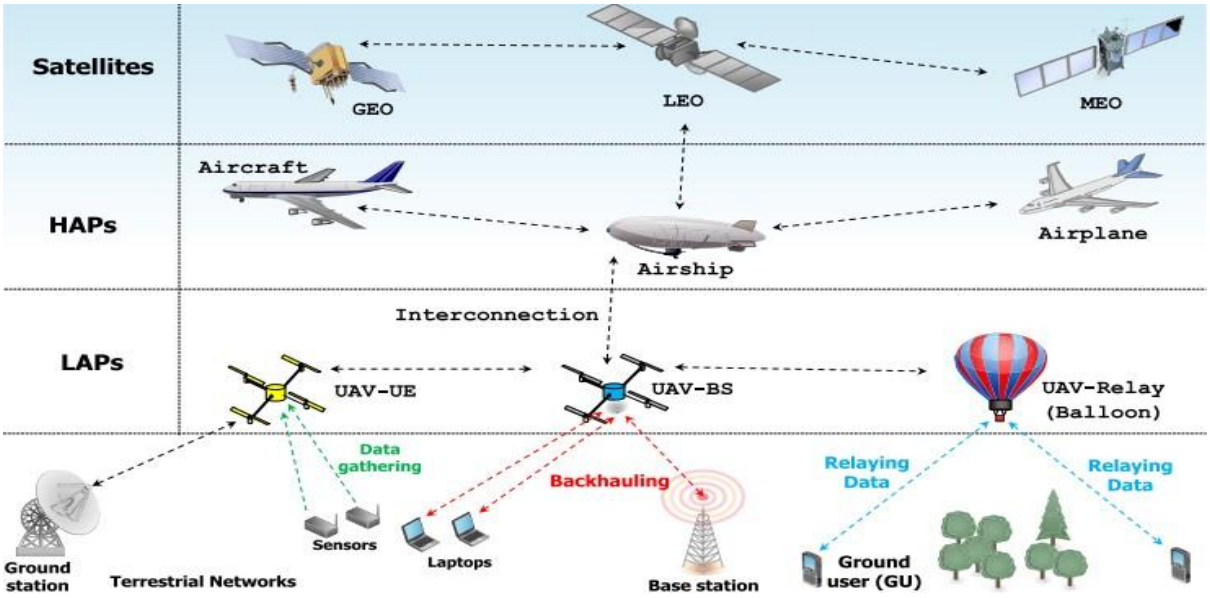

Figure 3Catégorisation et différents cas d'utilisation des drones. [6]

Dans les sections suivantes, nous décrivons chaque catégorie de drones et leurs différents caractéristiques drones en fonction de différents critères, tels que l'altitude, le temps de déploiement, l'endurance, etc.

.**le tableau 2.** présente une étude comparative entre ces types d'UAV en fonction de différents critères.

Table 2 Comparaison entre les plateformes aériennes. [6]

	LAPs	***HAPs***	***Satellites***
Altitude	Up to 10000 m	Up to 23000 m	Up to 36000 m
Ownership	Up to 50 kg	Up to 5000 kg	Undefined
Launch method	Individual users	Company	Government
Airspace class	Class A	Class D	Class E
Leval of control autonomy	Flexible	Less Flexible	Not Flexible
Deployment time	Short-term	Mide –term	Long-term
Endurance	Up to 40 min	Up to 100 days	Up to 15 year
Cost	Cheaper	Expensive	Highly expensive
PayLoad	Up to 7 kg	Up to 1000 kg	Up to 25000 kg
Couvrage	Medium	Large	Huge
Weight	Up to 10 kg	Up to 1000 kg	Up to 450 tons
Line-of- Sight(los)	Low	Medium	High
Functionality	Simple	Medium	Complex
Flight range	Up to 200 km	Up to 20 million km	Undefined
Mobility	Highly mobile	Less flexible mobility	Quasi statution
Regulation	Safty laws	Global laws	Inernationallaws
Energy	Battries	Fual	Fossi
Exemple	Balloons	Airships	Geo

Nous allons faire une définition simple de ces trois classifications :

- **LAP:**

Les LAP sont des plateformes aériennes sans pilote stationnaires/très mobile par exemple (petits drones ou ballons) avec une mobilité très flexible et une altitude inférieure à la stratosphère. Le déploiement de ces dispositifs peut se faire plus rapidement et de la manière la plus simple. La flexibilité des LAP leur permets d'être très utiles pour être combinés avec le concept cellulaire à large bande, tel que la 3G, 4G, 5G ,B5G, et même la 6G. En général, les LAP fournissent des liens de communication à courte portée en visibilité directe, ce qui permet de réduire considérablement leur consommation d'énergie. En outre, la mobilité des LAP est ajustable de manière flexible pour obtenir les emplacements souhaitables qui peuvent offrir une couverture maximale avec une consommation d'énergie. Dans le cas d'une défaillance des LAP[6]

- **HAP:**

Le HAP est une autre plateforme aérienne, qui a également suscité l'intérêt du monde du sans-fil récemment. Les HAP sont considérées comme des plateformes aériennes sans pilote de longue durée, avec une mobilité moins flexible et une altitude supérieure à la stratosphère. Les HAP peuvent offrir diverses caractéristiques, comme une couverture sans fil à grande échelle dans des zones géographiques avec de grandes surfaces, de faibles retards de propagation, des débits élevés, un déploiement progressif, une faible puissance de transmission, moins d'infrastructures au sol et des temps d'atterrissage et de décollage plus longs que les LAP. Leurs déploiements sont là pour fournir une connectivité à long terme aux régions qui posséderaient une connexion partielle. Cependant, l'adoption des HAP pose de nombreux problèmes, tels que le coût non rentable, la complexité, la consommation d'énergie élevée et les interférences intercellulaires importantes.[6]

- **LES SATILITES:**

Les satellites qui sont un type particulier de drones, sont des sondes spatiales non habitées placées en orbite autour de la Terre. Ils peuvent être déployés à diverses fins commerciales ou scientifiques, comme les téléphones par satellite, les systèmes de positionnement globaux (GPS), les télescopes et le suivi météorologique. Ajoutant aussi qu'ilexiste un nombre important de satellites en orbite autour de la Terre. Cependant, du point de vue des utilisateurs terrestres, la mobilité des satellites est considérée comme stationnaire. [6]

CHAPITRE II

Architecture de communication sécurisée d'une flotte de drones

Chapitre II

Architecture de communication sécurisée d'une flotte de drones

2.1. Cadre général flog et cloud dans UAV

Dans cette section pour comprendre le principe de flog et cloud nous traitons un modèle proposé .La représentation du problème est illustrée dans la figure 4 Chaque UAV est responsable de la détermination de sa trajectoire de manière autonome en fonction de la tâche assignée. De plus, chaque UAV capture des images et des informations nécessaires à ses missions. Pendant l'exécution de la tâche, certaines commandes sont de la responsabilité du GS (station au sol), par exemple, la supervision, le contrôle de la position, le stockage des données vidéo et l'affectation des tâches. [7]

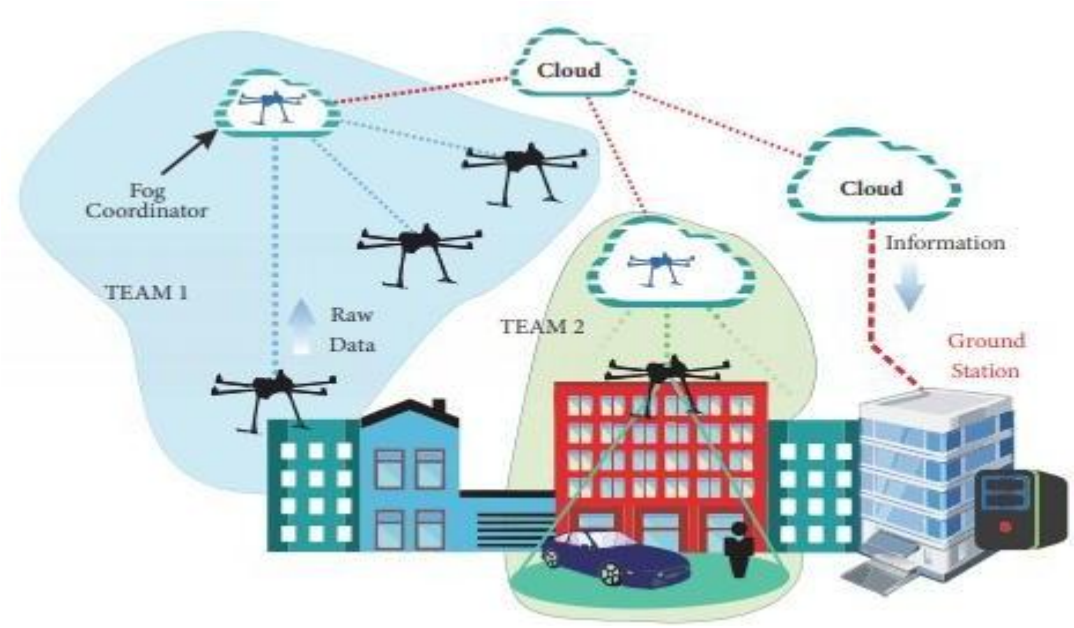

Figure 4 Représentation du problème [7].

Ces définitions de base jettent les bases de la définition du problème architectural et de l'organisation des composants.

2.1.1. Localisation des nœuds du fog :

La localisation du nœud d'informatique en fog est un élément clé de la définition de l'architecture. Deux possibilités sont envisagées ***La première*** est le placement du nœud de fog sur le sol, géographiquement proche du groupe de drones. ***La deuxième*** possibilité est d'intégrer le dispositif du fog dans le coordinateur du drone qui se déplacera avec son équipe pendant l'exécution de la mission.

Le principal avantage de la première option est d'éviter les restrictions d'énergie grâce à la possibilité de connecter le dispositif de fog à des centrales électriques ou à des générateurs.

En contrepartie, cette configuration crée des restrictions de transmission de données dès lors que les objets au sol peuvent interférer avec le signal du réseau.

Le déploiement d'un nœud du fog rend l'architecture des drones plus flexible et plus autonome. Un exemple est le temps de combat du coordinateur principal qui n'affecte pas la mission du fait que tous les aéronefs exécutant les missions partagent les mêmes restrictions de puissance.[7]

2.1.2. Services des nœuds du fog :

Les premières mises en œuvre de l'informatique en Fog n'étaient responsables que du regroupement des informations. Cependant, les applications actuelles intègrent plusieurs types de services préprogrammés capables de traiter les informations entrantes directement au niveau du dispositif du flog. Ce travail utilise la deuxième approche. Le coordinateur principal gère les informations de son groupe pour fournir des services et regrouper les informations dans le cloud. Il est important de connaître les exigences des drones pour déterminer quels services sont nécessaires au niveau du flog dans ce travail, les aéronefs peuvent fonctionner de manière autonome, c'est-à-dire qu'ils peuvent effectuer le contrôle de la lutte et la collecte de données et prendre des décisions concernant leurs tâches. En particulier pour le contexte SAR (Chercher et sauver), les drones doivent se battre le long d'un certain chemin pour capturer des images pour la reconnaissance d'objets. En outre, ils doivent décider si la mission est accomplie ou non en fonction des informations acquises

Le GS (station au sol), assiste l'aviateur lorsque des activités nécessitent un certain niveau de cognition élevé. Habituellement, le système de supervision effectue la planification des tâches, la surveillance et la planification de la trajectoire cependant, la plupart de ces activités ne nécessitent pas une intervention humaine directe et peuvent également être réalisées par un autre système autonome, ce qui est le cas de la plupart des services. Dans l'approche de ce travail, certains services sont déployés sur le « Fog-computing». Chaque fois que cela est possible pour réduire le débit du fog-cloud (par exemple, le stockage des données). D'autres tâches, telles que la définition et la supervision des objectifs de la mission, sont traitées dans lecloud.

La répartition des tâches proposée exige que des informations importantes soient partagées avec le cloud et qu'une grande partie des données restent au niveau du Fog. En ce sens, un algorithme doit classer les données entrantes pour déterminer celles qui doivent être envoyées au cloud. En partant du principe que chaque drone peut prendre des décisions liées à l'exécution de la tâche, il est possible de dire que les données critiques de la mission peuvent également être classées en fonction de leur importance.

Cette explication est représentée dans la ***figure 4***. Un coordinateur situé au niveau du fog gère les informations des nœuds les plus proches. Les données qui ne sont pas traitées et stockées localement dans le coordinateur sont transmises au système de supervision situé dans le cloud pour un traitement ultérieur. [7]

2.1.3. Caractéristiques des nœuds du fog

Les caractéristiques du matériel sont importantes pour mener l'analyse de l'architecture proposée, les méthodes de transmission des données concernant la communication inter- UAV et fog-cloud sont sélectionnées.

Chaque technologie doit offrir des débits de données et des plages de transmission différentes. **Le tableau 3** présente une liste des principales caractéristiques des technologies de communication sans fil typiques. La sélection du matériel approprié pour l'application UAV doit tenir compte du rapport énergie/couverture pour les temps de combat importants. [7]

Au début, nous découvrirons les types les plus importants de technologies de communication sans fil. [7]

1. ***Bluetooth:***

   Une importante technologie de communication à courte portée est bien sûr le Bluetooth, qui est devenue très important dans l'informatique et sur de nombreux marchés de produits de consommation. On s'attend à ce qu'elle joue un rôle clé dans les produits vestimentaires en particulier, en se connectant à l'IdO, mais probablement via un smartphone dans de nombreux cas. Le nouveau Bluetooth à faible énergie (BLE) - ou Bluetooth Smart, comme on l'appelle désormais - est un protocole important pour les applications IoT (internet of things). Il est important de noter que, tout en offrant une portée similaire à celle d Bluetooth, il a été conçu pour offrir une consommation d'énergie considérablement réduite. [8]

2. ***Wifi:***

   La connectivité Wi-Fi est souvent un choix évident pour de nombreux développeurs, notamment en raison de l'omniprésence du Wi-Fi dans l'environnement domestique au sein des réseaux locaux. Elle nécessite peu d'explications supplémentaires, si ce n'est l'évidence qu'il existe une vaste infrastructure existante et qu'elle offre un transfert de données rapide et la capacité de traiter de grandes quantités de données. [8]

3. ***HSPA(High Speed Packet Access):***

   L'accès par paquets à haut débit (HSPA) désigne un ensemble de technologies issues d'améliorations apportées aux systèmes d'accès multiple par répartition en code à large bande (WCDMA). Le HSPA est composé du protocole HSDPA (High-Speed Downlink Paquet Access) et du protocole HSUPA (High-Speed Uplink Paquet Access). Il offre des débits de données de pointe pouvant atteindre 14 Mbps en liaison descendante et 5,7 Mbps en liaison montante. [9]

4. ***Zigbee:***

ZigBee, comme le Bluetooth, dispose d'une large base installée de fonctionnement, bien que peut être traditionnellement plus dans les milieux industriels. ZigBee PRO et ZigBee remonte Control (RF4 CE), parmi d'autres profils ZigBee disponibles, sont basés sur le protocole IEEE802.15.4, qui est une technologie de réseau sans fil standard de l'industrie fonctionnant à 2,4 GHz et ciblant les applications qui nécessitent des échanges de données relativement peu fréquents à de faibles débits sur une zone restreinte et dans un rayon de 100 m, comme dans une maison ou un bâtiment. [8]

Table 3 Caractéristiques des normes sans fil typiques. [7]

	Bluetooth	***Wi-Fi***	***HSPA***	***ZigBee***
Couvrage	***100m***	***0 .1-2Km***	***5Km***	***1.2-14km***
Thoughput	***22Mps LF120-134Khz***	***Up to 300Mbps***	***5 .76-11Mbps***	***0.25-72Mbps***
Frequency	***HF13.56MHz UHF850-960MHz***	***2.4 ; 5GHz***	***TDD1.85-3.8Ghz FDD0.7-2.6GHz***	***0.9 ;1 .2 ;2.4GHz***
Energy efficiency	***High***	***Low***	***Deppends on the signal strength***	***Depends on the model***

2.1.4. Architecture de l'informatique en fog.

Cette approche est détaillée dans l'architecture présentée dans la figure 4
Certaines de ces données sont envoyées aux dispositifs de supervision comme les images et les positions géographiques. Cependant, toutes les données sont marquées en fonction du type de message. Les données provenant des différents aéronefs sont reçues par le coordinateur et classées en fonction de leur importance.

Postérieurement, elles sont envoyées soit au traitement local, soit aux filtrations si elles sont classées comme importantes, les données vont directement au bloc filtration pour être regroupées et envoyées directement au GS (station au sol).

Sinon, les données sont transférées au traitement local, qui peut les traiter ou les stocker. Cela signifie que le coordinateur dispose d'algorithmes similaires à ceux employés sur le GS, c'est- à-dire que le niveau du Fog peut assister les drones pendant

l'exécution de leurs tâches. Les images et autres données peuvent également être stockées à ce niveau pour être récupérées par le GS dans le futur.

2.2. Taxonomie des protocoles de sécurité pour l'environnement IoD

La figure 5 présente une taxinomie des différents protocoles de sécurité liés à l'environnement IoD. Nous allons discuter des mécanismes de sécurité importants suivants qui sont nécessaires pour sécuriser la communication IoD.

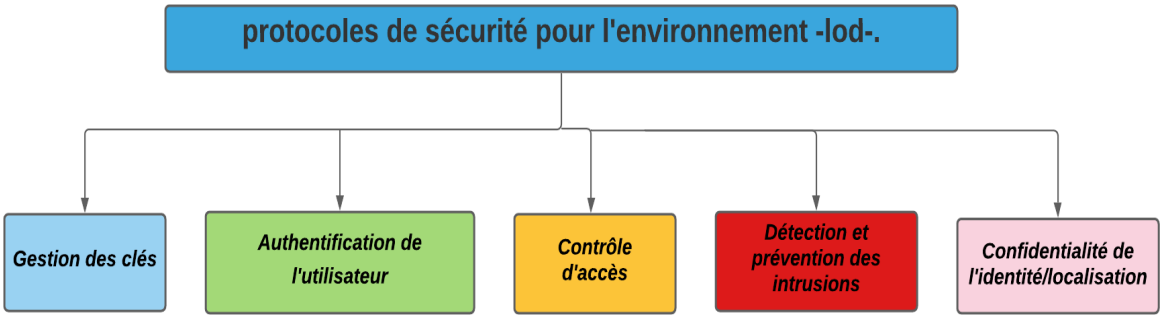

Figure 5 Taxonomie des protocoles de sécurité dans l'environnement IoD [9].

2.2.1. Gestion des clés

La gestion des clés est l'un des principaux services de sécurité utilisés dans l'environnement IoD. Avant le déploiement des drones dans une zone de vol particulière, une autorité de confiance (par exemple, la salle de contrôle) pré charge les informations d'identification secrètes dans leur mémoire, appelée porte-clés. Après le déploiement, deux drones voisins, DRi et DRj, essaient d'établir des clés secrètes par paires en utilisant les informations d'identifications communes présentes dans leurs porte-clés. Sur la base de la probabilité d'établissements d'une clé entre DRi et DRj , un schéma de gestion de clé peut être classé en deux catégories : **(1) probabiliste et (2) déterministe**.

2.2.2. Authentification de l'utilisateur

La plupart des applications impliquées dans l'environnement IoD sont des informations en temps réel. Par conséquent, il est évident que les utilisateurs (parties externes) sont généralement fascinés par l'accès aux données de détection en temps réel des drones volant dans certaines zones particulières. Cela est possible si les utilisateurs sont autorisés à accéder directement aux données en temps réel à partir des drones volants dans l'environnement IoD et non à partir du serveur.

Il devient un autre service de sécurité important pour avoir un protocole d'authentification des utilisateurs efficaces et sûrs dans l'environnement IoD. En fonction des types et le nombre de facteurs utilisés dans le protocole d'authentification de l'utilisateur, nous appelons ce schéma d'authentification utilisateur un facteur unique (par exemple, si seule la carte à puce d'un utilisateur est utilisée), à deux facteurs (par ex. carte à puce et le mot de passe d'un utilisateur), à trois facteurs (par ex. carte à puce, mot de passe et données biométriques personnelles) [9]

2.2.3. Contrôle d'accès

Il est inévitables de déployer dynamiquement de nouveaux nœuds (drones) dans l'environnement IoD, car les drones sont sujets à des attaques de capture physique et peuvent s'éteindre en raison de la consommation d'énergie ou d'une défaillance matérielle. Un drone déployé peut ne pas être toujours un nœud authentique, car les nœuds malveillants peuvent être déployés par un adversaire, et il est difficile de distinguer les nouveaux nœuds malveillants des nœuds authentiques existant dans le réseau. Il est donc nécessaire de concevoir un mécanisme de contrôle d'accès pour le déploiement de nouveaux nœuds afin d'empêcher les nœuds malveillants de pénétrer dans l'environnement IoD. Le mécanisme de contrôle d'accès comprend principalement les deux tâches suivantes, comme dans le cas des réseaux de capteurs sans fil (WSN):

- **Authentification des drones** : les drones nouvellement déployés doivent s'authentifier auprès de leurs voisins pour prouver qu'ils sont des nœuds authentiques et qu'ils peuvent accéder aux services.

- **Création de la clé** : un drone nouvellement déployé doit établir les clés secrètes avec ses drones voisins pour assurer une communication sécurisée tout en transmettant les données seulement après que le processus d'authentification du drone soit terminé.

Un mécanisme de contrôle d'accès peut être conçu sur la base de certificats émis par une autorité de confiance utilisant sa propre clé privée ou sans certificats. Ainsi, un schéma de contrôle d'accès peut être classé en deux catégories : **(1) basé sur des certificats et (2) sans certificat.** [9]

2.2.4. Détection et prévention des intrusions

Une intrusion est un accès non autorisé à un réseau pour accéder à des informations sensibles à des fins malveillantes. L'intrusion peut être de deux types:

- **l'intrusion active :** par exemple (la fabrication et la transmission de paquets malveillants, l'interception et diverses attaques partout)
- **l'intrusion passive :** par exemple (la collecte d'informations en surveillant le réseau et en écoutant aux portes).

Un système de détection d'intrusion (IDS) est un dispositif ou un logiciel d'application qui surveille le réseau cible pour détecter toute violation de la vie privée ou tout accès non autorisé. [9]

2.2.5. Confidentialité de l'identité/localisation

Les informations privées, telles que le lieu d'origine des drones déployés, les entreprises qui travaillent, les propriétaires ou la route principale de déplacement. Ces informations sont liées à un individu et peuvent révéler des informations sur son mode de vie, ce qui peut conduire à un profilage ou à une perte de confidentialité.

En outre, un adversaire a la possibilité d'utiliser la localisation pour déduire l'état de santé d'une personne, ses préférences personnelles ou ses opinions politiques, ainsi que pour mener une attaque physique (par exemple, un vol ou un harcèlement). Étant donné que les algorithmes cryptographiques à clé symétrique [par exemple, la norme de chiffrement des données (DES) et ses variantes, 2 DES, 3 DES et différents modes des DES , et la norme de chiffrement avancé (AES)] sont légers et consomment peu d'énergie, il est souhaitable de déployer de tels algorithmes cryptographiques légers pour traiter la question de la localisation et de la confidentialité dans le cas des systèmes à ressources limitées confidentialité pour les dispositifs à ressources limitées dans l'environnement IoD. [9]

2.3. Travaux connexes :

Nous allons maintenant parler des travaux connexes sur l'authentification pour IoD

Table 4 Travaux connexes sur l'authentification pour IoD

Schème	Le model réseau	Les objectifs	Les méthodescryptographiques	Performance(+) et Limitation(-)
WAZID *et al.*[10] (2019) Cité 63 12/05/2021	Iod qui fournit les données en temps réel entre les drones et utilisateurs et le serveur	améliorer les caractéristiques de sécurité et de fonctionnalité, et lacommunication	- cryptographie à courbe elliptique (ECC) -Fuzzy extractors	**(+)**améliore la securite**(+)**préserver les messages **(+)**Améliorer la communication **(-)**débit de comminution **Comparé avec:** Turkanovic´ et al [11]. Challa*et al.* [12].
TCALAS [13] (2019) Cité 63 13/05/2021	un nouveau mécanisme léger d'authentification anonyme et légère pour l'environnement IoD (TCALAS) qui est un système à trois facteurs utilisant de l'appareil mobile de l'utilisateur, un mot de passe et des données biométriques. TCALAS est rigoureusement testé pour sa partie	Améliorer la sécurité et la communication et le calcul pour les drones ou les dispositifs de détection à ressources limitées dans l'environnement IoD.	Trois facteurs (carte à puce, mot de passe utilisateur et biométrie) ; applique la "fonction de hachage cryptographique à sens unique" et '"extracteur flou pour la vérification biométrique" ; basé sur " les références temporelles".	**(+)**Il est protégé contre diverses attaques connues. **(+)**Il fournit une analyse formelle de la sécurité **(+)**Il est également sécurisé contre les "attaques par déni de service". **(-)**plusieurs facteurs **Comparé avec:** Wazid et al[14] Challa et al[15] Turkanovic et al[16] Tai et al[17]

	sécurité à l'aide d'une analyse de sécurité formelle de sécurité à l'aide du modèle ROR (Real-or Random)			
Zhang et al[18] (2020) Cité 13 14/05/2021	une architecture d'IoD base sur un schéma AKA léger entre les drones et les utilisateurs avec l'aide du serveur.	améliorer la sécurité de l'environnement IoD et résister à diverses attaques et améliorer le coût de communication et de calcul	- cryptographie à courbe elliptique (ECC) -Fuzzy extractors	**(+)**exigences de sécurité **(+)**coût de calcul. **(+)**le coût de la communication. Comparé avec: Wazid et al [19] Singh et al[20]
Kirsal Ever [21] (2020) Cité 8 15/05/2021	un schéma cadre d'authentification est proposé pour les drones considérés comme des puits mobiles pour les WSN - Wireless sensor networks-.	pour améliorer la consommation d'énergie et de ressources des nœuds de capteurs. Les techniques de drones existants ont des privilèges d'authentification restreints pour leur interaction avec les WSN et pour améliorer la sécurité de l'environnement	- symmetricencryptionfunction - cryptographie à courbe elliptique (ECC)	**(+)**Coût de calcul réduit. **(+)**Temps de communication réduit **(+)**exigences de sécurité **(-)**Gamme limitée de réseau WSN **Comparé avec:** Al-Turjman et al [22] Das et al[23] Srinivaset al [24]

Gope et al [25] (2020) Cité 6 16/05/2 021	Iod qui fournit données en temps réel base sur trois élément : un ensemble de drones et un ensemble d'opérateurs d'infrastructure et un fournisseur de services de drones (USP)	Assure une communication sécurisée entre les éléments d'Iod	-Fonction de PUF -Fonction physiquement non clonable	(+)Authentification mutuelle (+)Anonymat (+)Sécurité contre les attaques de falsification (+)Menace d'emplacement (+)Physical Security of the UAV (-)Conditions de la synchronisation des horloges **Comparé avec:** Tian et al. [26] Zhang et al. [27] Srinivas et al. [28]
Gope, Millwoodet al [29] (2021) Cité 1 17/05/2 021	Les drones sont équipés d'étiquettes avec RFID et doivent passer par un point de contrôle où l'étiquette est scannée et envoyées à un serveur sécurisé pourvérification.	Garantir diverses caractéristiques de sécurité importantes telles que l'anonymat, la protection contre les attaques DoS, etc., qui sont impératives pour toute application et toutservice IoT.	- FuzzyExtractor - Physically uncloneablefunction - Pseudo random function	(+)Authentification (+)mutuelle, Intraçabilité, et Evolutivité **Comparéavec:** Moriyama et al[30] Aysu et al [31]

WAZID et al [10]

WAZID et al ont présenté un schéma qui permet de faire une nouvelle authentification entre un utilisateur et un drone accédé avec l'aide du serveur. Une clé de session établie après une authentification mutuelle réussie entre un utilisateur et un drone les aide à communiquer en toute sécurité, de sorte que diverses attaques connues sont empêchées par un adversaire , le schéma proposé est efficace en termes de communication et de calcul, et offre également plus de sécurité et de fonctionnalités par rapport aux autres schémas connexes.

TCALAS [13]

TCALAS un schéma à trois facteurs utilisant le dispositif mobile de l'utilisateur, le mot de passe et la biométrie. TCALAS est testé rigoureusement pour sa partie sécurité à l'aide d'une analyse de sécurité formelle utilisant le modèle ROR, TCALAS est meilleur en termes de sécurité, supporte plus d'attributs de fonctionnalité, et a des coûts de communication et de calcul plus faibles pour les drones.

Zhang et al[18]

Zhang et al ont présenté un schéma AKA léger entre les drones et les utilisateurs avec l'aide du serveur. Notre schéma proposé peut être prouvé sûr sous le modèle de l'oracle aléatoire, et il peut également répondre aux exigences de sécurité de l'environnement IoD et résister à diverses attaques.

Kirsal Ever[21]

le schéma présenté par KirsalEver considères les drones comme des puits mobiles pour les WSN. Les travaux existant sur l'authentification de l'environnement WSN-Drones sont examinés et étendus. Le cadre proposé est évalué par une analyse de sécurité informelle afin de s'assurer de sa résilience face à des attaques potentielles importantes et bien connues.

Gope et al [25]

Gope et al proposait un schéma d'accord de clé authentifiée efficace pour les drones assistés par la périphérie. Le protocole proposé utilise des fonctions cryptographiques telles que les PUF et les opérations de hachage. L'analyse de sécurité informelle montre que le protocole propose la capacité à résister aux principales attaques des écurité.

Gope, Millwoodet al [29]

Gope,*Millwood*et al. [] présente un nouveau protocole d'authentification préservant la confidentialité basé sur les PUF pour un système de drones basé sur le RFID, qui peut garantir plusieurs propriétés de sécurité impératives telles que la résilience contre les attaques de type "man-in-the-middle". la confidentialité contre l'écoute clandestine, etc., qui sont nécessaires pour toute application basée sur le RFID. La sécurité et l'analyse des performances démontrent que le schéma d'authentification est sûr et efficace ; il peut donc être utile pour plusieurs solutions de sécurité RFID réalisables en pratique et utilisant des PUF.

CHAPITRE III

Un schéma d'Authentification Sécurisé Pour L'IOD

Chapitre III

Un schéma d'authentification sécurisé pour l'IoD

Dans ce chapitre, nous introduisons les différentes connaissances de base concernant les méthodes cryptographiques afin de réaliser ce travail. Ensuite, nous présenterons une explication détaillée de notre schéma d'authentification proposé avec ses différentes phases.

3.1. Les éléments fondamentaux

3.1.1. Le crypto-system des courbes elliptiques(ECC)

Définition:

Une courbe elliptique est un l'ensemble de points qui satisfait une équation mathématique spécifique [32]. L'équation d'une courbe elliptique ressemble à ceci:

$$Y^2=x^3+ax+b. \quad (1)$$

Où le discriminant de $x^3 + ax + b$ soit non nul.

$$\Delta == -(4a^3 + 27b^2) \neq 0. \quad (2)$$

Puis en rajout pour cette courbe un point qui tend vers l'infini noté O.

Théorème:

Pour un ECC, nous nous intéressons à une forme restreinte de courbe elliptique qui est définie sur un champ fini.

- Choisir deux entiers non négatifs a et b . qui satisfont:

$$4a^3 + 27\,b^2 \pmod p \neq 0. \quad (3)$$

P est un

nombrepremier ;

a ;b< P

- Puis former le groupe elliptique Ep (a, b) modulo p dont les éléments x, y sont des paires d'entiers non négatifs inférieurs à p satisfaisant:

$$y^2 = x^3 + ax + b \pmod p. \quad (4)$$

Avec le point O qui tend vers l'infini.

- Le problème du logarithme discret de la courbe elliptique pour :

$$Q = xP. \quad (5)$$

Fixer un point P premier

xP représente le point P sur la courbe elliptique ajouté à lui-même x fois.

  - Il est relativement facile de calculer Q étant donné x et P, mais il est très difficile de déterminer x étant donné Q et P

Chiffrement /Déchiffrement ECC:

Nous suivons les étapes suivantes :

- coder le message en clair m à envoyer en tant que point x-y noté P_m.
- P_m qui sera chiffré sous forme de cipher texte et ensuite déchiffré.

- Le système de cryptage/décryptage nécessite comme paramètres un point G et un groupe elliptique Ep (a,b).
- Alice sélectionne une clé privée n_A et génère une clé publique P_A pour crypter et envoyer un message P_m à Bob.

$$P_A = n_A * xG. \quad (6)$$

- Alice choisit un entier positif aléatoire x et produit le cipher texte Cm composé de la paire de points:

$$Cm = \{xG, P_m + xP_B\} \quad (7)$$

- Alice a utilisé la clé publique de Bob P_B pour déchiffrer le cipher texte.
- Donc Bob multiplie le premier point de la paire par sa clé secrète n_B et soustrait le résultat du deuxième point:

$$P_m + xP_B - n_B (xG) = P_m + x (n_B G) - n_B (xG) = P_m \quad (8)$$

- Alice a masqué le message Pm en y ajoutant x P_B. Personne sauf Alice ne connaît la valeur de x, donc même si P_B est une clé publique, personne ne peut supprimer le masque x P_B. Cependant, Alice inclut également un indice qui aide à supprimer le masque si l'on connaît la clé privée n_B. Pour qu'un attaquant récupère le message, il devrait calculer x étant donné G et xG, ce qui est difficile.

3.1.2. Fonction de hachage

Définition:

Une fonction de hachage en général est donc une fonction transformant des données de tous types en données de taille fixe en tant que sortie appelée code de hachage ou simplement hachage du message d'entrée .Ces données peuvent tout à fait avoir plusieurs formes. Cela peut ainsi très bien être par exemple une image ou simplement du texte. [33]

Théorème:

Dans ce théorème on va parler sur l'algorithme de SHA256 qui passe par 7 étapes [34]

étape 1 :*Ajout des bits de remplissage*

Le message est rempli de manière à ce que sa longueur soit congruente à 448, modulo 512. Ce remplissage est un bit unique ajouté à la fin du message, suivi d'autant de zéros que nécessaire pour que la longueur des bits soit égale à 448 modulos 512. [34]

étape 2 : *Ajouter de longueur*

Une représentation de 64 bits de la longueur du message est ajoutée au résultat. Cette étape permet de faire de la longueur du message un multiple exact de 512 bits de longueur [34]

étape 3 : *Analyser le message*

Le message complété est décomposé en N blocs de message de 512 bits, M(1), M(2) , . . ., M(N) en ajoutant un bloc de 64 bits. [34]

étape 4 : *Initialiser la valeur dehachage*

La valeur de hachage initiale, H(0), est définie et se compose de 8 mots de 32 bits, sous une forme hexadécimale. [34]

étape 5 :*préparer le calendrier du message*

SHA256 utilise un calendrier de messages de 64 mots de 32 bits. Les mots du calendrier du message sont étiquetés W_0,

$$W_i = \begin{cases} M_t^{(t)} & 0 \le t \le 15 \\ \sigma_1^{(256)}(W_{i-2}) + W_{i-7} + \sigma_0^{(256)}(W_{i-15}) + W_{i-16} & 16 \le t \le 63 \end{cases}$$

W_1,..., W_{63}

Où:

σ1(256) ($\mathcal{W}i$−2)= (($\mathcal{W}i$−2) ROTR 17) ⊕ (($\mathcal{W}i$−2) ROTR 19) ⊕ (($\mathcal{W}i$−2) SHR 10)

σ0(256) ($\mathcal{W}i$−15)= (($\mathcal{W}i$−15) ROTR 7) ⊕

(($\mathcal{W}i$−15) ROTR 18) ⊕ (($\mathcal{W}i$−15) SHR 3) [34]

étape 6 : *initialiser les 8 variables de travail a, b, c, d, e, f, g et h avec la (i-1)ème valeur de hachag*

For *t*=0 to 63:

{

T1 = h + Σ1(256) (e) + Ch(e,f,g) + K1(256) + Wt

T2 = Σ0(256) (a) + Maj(a,b,c)

H = G

G = F

F = E

E = d + T1

$$D = C$$

$$C = B$$

$$B = A$$

$$A = T1 + T2$$

}

Où:

$$\Sigma 1\ (256)\ (\boldsymbol{e}) = (e\ ROTR\ 6) \oplus (e\ ROTR\ 11) \oplus (e\ ROTR\ 25)$$

$$\Sigma 0\ (256)\ (\boldsymbol{a}) = (e\ ROTR\ 2) \oplus (e\ ROTR\ 13) \oplus (e\ ROTR\ 22)$$

$$Ch\ (e, f, g) = (e \wedge f) \oplus (\sim e \wedge g)$$

$$Maj\ (a, b, c) = (a \wedge b) \oplus (a \wedge c) \oplus (b \wedge c)$$ [34]

étape 7 : *sortie*

Après avoir répété les étapes 1 à 4 au total N fois, la fonction de hachage résultante est: $H0(N) \parallel H1(N) \parallel H2(N) \parallel H3(N) \parallel H4(N) \parallel H5(N) \parallel H6(N) \parallel H7(N)$ [34]

3.1.3. La Blockchain

Définition:

La Blockchain est une technologie moderne de stockage et de transmission d'informations. Elle fonctionne sans organe central de contrôle, mais apporte transparence et sécurité grâce à la validation des transactions par les nœuds du réseau. [35]

La blockchain est assimilable à une base de données contenant l'historique de tous les échanges effectués depuis sa création. Puisqu'elle est partagée entre tous ses utilisateurs et sans intermédiaire, chacun peut vérifier sa validité et confirmer son intégrité.[36]

La Structure de la Blockchain:

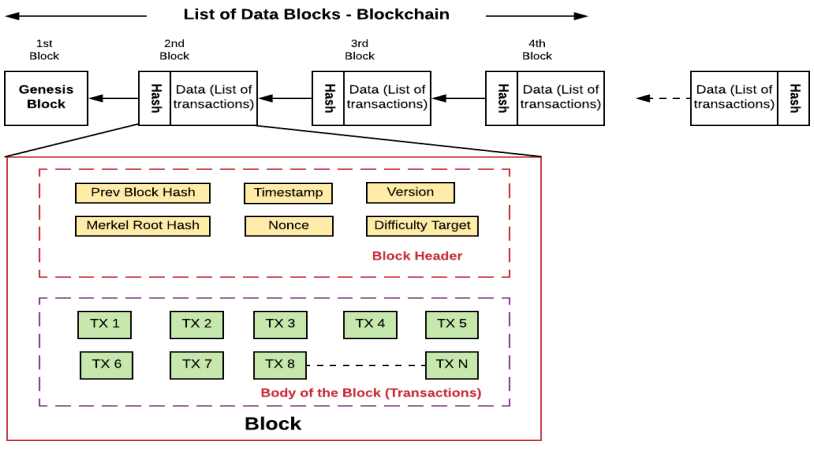

Figure 6 Schéma de Blockchain [37].

- Chaque bloc se compose d'une section d'en-tête et d'une section de corps.
- La section d'en-tête du bloc peut être constituée des informations suivantes : blockchain bitcoin, un bloc est composé de 80bytes.
    - Racine de Merkel (32bytes)
    - Nonce (4bytes)
    - Timestamp (4bytes)
    - Hash de l'en-tête du bloc précédent (32 bytes)
    - Numéro de version (4bytes)
    - Objectif de difficulté (4bytes)
- La section du corps du bloc peut être constituée de la liste des transactions. Ceci est représenté dans le diagramme ci-dessus.
- Le hachage de l'en-tête de bloc (80 bytes) du bloc N donne lieu à 32 bytes (256 bits en utilisant SHA-256) qui sont stockés en tant que "hachage de l'en-tête de bloc précédent" - une partie de l'en-tête de bloc du bloc N+1.[37]

3.1.4. Algorithme de consensus

Définition:

Un algorithme de consensus peut être défini comme le mécanisme par lequel un réseau Blockchain parvient à mettre un accord. Les blocs publics sont construits comme des systèmes distribués et, puisqu'ils ne dépendent pas d'une autorité centrale, les nœuds distribués doivent se mettre d'accord sur la validité des tractions en utilisant un algorithme de consensus [37].

3.2. Schéma d'authentification pour l'IoD:

3.2.1. Le modèle architecture

Un modèle de réseau basé sur la blockchain pour IoD, présenté à la **figure 7** a été adapté à la conception du schéma proposé.

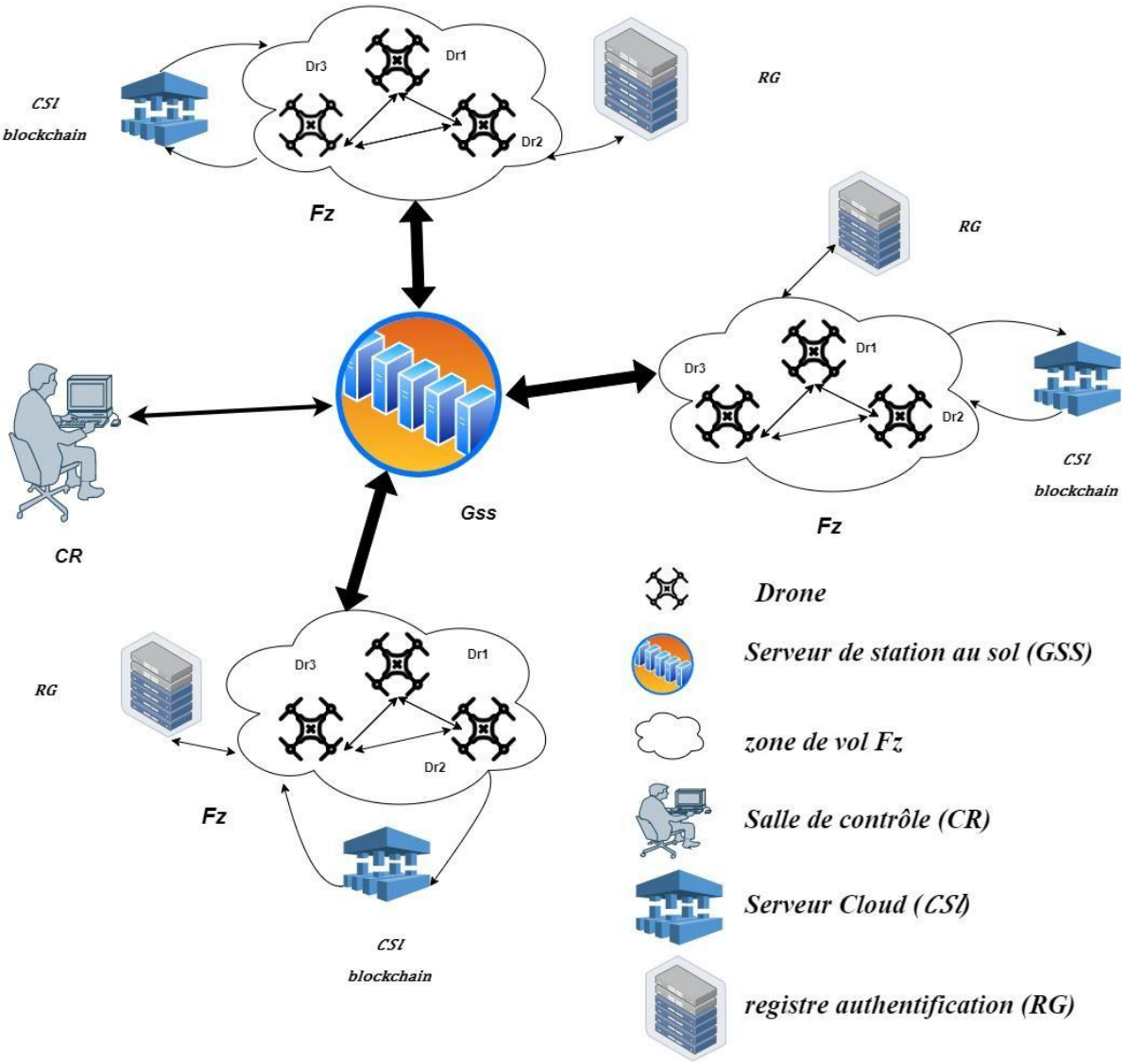

Figure 7 Le modèle architecture basé sur le Fog et la Blockchain.

Les entités suivantes existantes :

- ***Salle de contrôle (CR) :*** Le CR est une autorité de confiance qui est responsable de l'enregistrement de tous les drones déployés DRj et du serveur de la station au sol(GSS).

- ***Serveur de station au sol (GSS) :*** Le GSS est chargé de contrôler les drones dans leurs zones de vol respectives. En outre, le GSS collectera les données en temps réel des drones DRj et former des blocs contenant les transactions. Les blocs construits sont ensuite transmis au serveur cloud (CSl).

- ***Serveur Cloud (CSl) :*** Un serveur en cloud CSl jouera le rôle de leader parmi tous les serveurs en cloud du réseau P2P CS. Ensuite, le leader est responsable de la vérification et de l'ajout des blocs dans la blockchain en utilisant l'algorithme de consensus de lablockchain.

- ***Drone:*** Un drone est un véhicule aérien sans pilote (UAV) , c'est-à-dire un aéronef contrôlé à distance ou par des ordinateurs de bord. Le site drones peut naviguer de manière autonome sans implication du contrôlehumain.

- ***Registre authentification (RG):*** les résultats de l'authentification enregistrés dans le grand registre public **RG** a l'aide de ***CSL***

Table 5 Les notations utilisées dans notre schéma. [38] [39]

Symbole	Notation
E	Courbe elliptique
P	Point de base deE
n	Un grand nombre premier
PKx	Clé publique deX
SKx	Clé secret de X
(•)	Opération multiplicative ECC
SHA265 ( )	Fonction de hachage
IDx	Identifiant de X
TWx	Fenêtre de temps généré par X
ΔT	Délai maximal de transmission associé à un message
To$_i$	Un jeton
h	Un hachage
CPKx(M)	Chiffre le message M avec la clé publique de X
Cert	Un certificat
$\oplus$	Opération XOR
S	Une signature
na	Un nombre aléatoire $\in$Z*p
N_i	Un nombre construit de rni
*AutJ*xS	Un jeton d'authentification généré par X
K$_{ij}$*CD*	Une clé de session
Kf()	Fonction de dérivation de clé
Vote$_{req}$	Un jeton de demande de vote
B_i	Indice du bloc i
Bloc	Contenue du bloc
AM_P	Identifiant de l'AM marqué président
SignK(*B*)	Signé B avec la clé K
Vote$_{res}$	Un jeton de la réponse du vote
*Acc*req	Un jeton de demande d'accès au service Fog

3.2.2. Les phases de modèle system

- Dans ce modèle system les phrases dans lesquelles on utilise dans le schéma d'authentification sont: **Initialisation,Enregistrement**,**Authentification mutuelle et échange de clés**, **Consensus et mise à jour des drones**

.

3.2.2.1 Phase d'initialisation du système

Dans cette phase, le CR choisit les paramètres du système en suivant les étapes suivantes [39] :

- ***Étapes 1:***

  génère la courbe elliptique E et fixe ses paramètres publics N (un grand nombre premier) et P (point de base de cette courbe) .

- ***Étapes 2:***

  CR calcule ses clés publiques et secrètes comme (SK_{CR}& PK_{CR}) || génère une clé secrète comme: $SK_{CR} \in Z^*_p$ et une clé public comme: PK_{CR}= SK_{CR}• P.

- ***Étapes 3:***

  - CR définit une fonction unidirectionnelle *SHA265* ( ) fonction de hachage à utiliser pour vérifier l'intégrité lors de la phaseIII
  - les paramètres <*E*, *P*, *n*, *SHA265* ( ), PK_{CR}> sont publiés publiquement.
  - le CR choisit sa propre identité commeID_{CR}.

*3.2.2.2.**Phase d'enregistrement***

- Initialement, une autorité de confiance spécifiquement, la salle de contrôle (CR) inscrit tous les drones (DRs) et leur serveur de station au sol associé (GSS).
- les DR et GSS sont enregistrés à proximité dans le CR. Leurs identités respectives (ID_{DRi}et ID_{CRk}) sont gardées anonymes et ne sont jamais échangées sans être chiffrées

DR $_i$ /Gss$_k$	**CR**
-SélectionneID_{DRi} -Générer une fenêtre de temps: TW_{DRi} -Générer une clé scrète aléatoire: $SK_{0DRi} \in Z_{*p}$ -Calculer sa clé publique: $PK_{0\ DRi} = SK_{0DRi} \bullet P$ -Calculer: $To_0 = \{ID_{DRi} \| TW_{DRi} \| PK_{0DRi}\}$ -Calculer: $h_0 = SHA256\ (To_0)$ -Calculer: $CPK_{CR}\ (To_0)$ <$CPK_{CR}(To_0)$, h_0, TW_{DRi}> ⟶	- Déchiffrer $CPK_{CR}\ (To_0)$ en utilisantSK_{CR} - Calculer: $h_0' = SHA256(To_0)$ - Si $h_0'=h_0$alorsc ontinuersinon demande auDR_i de réessayer l'envoie

- Extraire ID_{DRi}et TW_{DRi}à partir deTo_0
- ValiderTW_{DRi}
-Vérifier la disponibilité de l'$ID_{DR\ i}$dans son Blockchain
- Attribuer un identifiant unique ID_{DRi}pour l'DR_i
- Générer une clé secrète aléatoire: $SK_{DRi} \in Z^*_p$
- Calculer sa clé publique: $PK_{DRi} = SK_{DR} \bullet P$
- Calculer: $To_1 = <PK_{DRi}, SK_{DRi}>$
- Générer une fenêtre de temps:TW_{CR}
- Calculer: $h_1 = SHA256\ (To_1, SHA256\ (SK_{CR} || ID_{CRi}), TW_{CR})$
- stocker: $\boldsymbol{Cert} = SHA256\ (SK_{DRi} || ID_{DRi}) \oplus SHA256\ (SK_{CR} || ID_{DRi})$
- Calculer: $CPK_{DRi}(To_1, SHA256\ (SK_{CR} || ID_{DRi}), TW_{CR})$

$< CPK_{0DRi}\ (To_1, SHA256\ (SK_{CR} || ID_{DRi}), TW_{CR}), h_1, TW_{CR} >$ sur un canal sécurisé

- Déchiffrer $CPK_{0DRi}\ (To_1, SHA256\ (SK_{CR} || ID_{DRi}), TW_{CR})$ en utilisantSK_{0DRi}
- Valider TW_{CR}
- Calculer:

$h_1' = SHA256\ (To_1, SHA256\ (SK_{CR} || ID_{DRi}), TW_{CR})$

- Si $h_1' = h_1$ alors continuer sinon demande au CR de réessayerl'envoie
- Sauvegarder PK_{DRi}, SK_{DRi},et

S = SHA256 (SK_{CR} || ID_{DRi})

Algorithme 3.1. Phase II: Phase d'enregistrement. [38]

❖ Explication d'Algorithme

Étape 1

-Sélectionne l'identificateur au niveau des drones : ID_{DRi}

-Générer une fenêtre de temps (TW_{DRi}) pour connaître l'heure d'envoi et de réception

-Générer une clé secrète aléatoire ($SK_{0DRi} \in Z^*_p$) puis Calculer sa clé publique en utilisant $PK_{0\,DRi} = SK_{0DRi} \bullet P$

Étape 2

-DR crée un jeton temporaire composéde: identificateurs (ID_{DRi}) et fenêtrede temps (TW_{DRi}) et de sa clé publique initiale ($PK_{0\,DR}$).

- hachage jeton temporaire (To) en utilisant SHA256 () la fonction dehachage.

-Le jeton chiffré $CPK_{TA}(To_0)$ ont utilisé la clé publique de TA PK_{CR} pour le crypter.

- $CPK_{CR}(To_0)$ et h_0 et TW_{DRi}sont envoyés auCR.

Étape 3

- la réception du message $<CPK_{CR}(To_0), h_0, TW_{DRi}>$

- le CR le déchiffre à l'aide de sa clé secrèteSK_{CR}

- calcule $h_0' = SHA256(To_0)$ pour le comparer avech_0

- Si $h_0' = h_0$ alors continuer sinon demande au Dr_i de réessayer l'envoie Extraction de ID_{Dri}, TW_{Dri} et PK_{0Dri}.

- ValiderTW_{DRi}

- Vérifier la disponibilité de l'ID_{DRi}dans sonBlockchain

- Attribuer un identifiant unique ID_{DRi}pour l'DR_idans leCR

- Générer une clé secrète : SK_{DRi}et calculer la clé publicPK_{DRi}

- Calculer jeton To_1 qui compose de la clé secrète et la clé public

- Générer une fenêtre de temps pour le CR :TW_{CR}

- Calculer: $h_1 = SHA256$ (To_1, $SHA256$ (SK_{CR}|| ID_{CRi}),TW_{CR})

- LeCRcalculeetstocke $Cert$=S_{SHA256}(SK_{DRi}||ID_{DRi})⊕S_{SHA256}(SK_{CR}||ID_{DRi})dans les registre destockage

- chiffrer (To_1, $SHA256$ (SK_{CR} || ID_{DRi}), TW_{CR}), h_1, TW_{CR} avec la clé publique de DR et envois sur un canalsécurisé

Étape 4

- Déchiffrer CPK_{0DRi} en utilisantSK_{0DRi}

- ValiderTW_{CR}

- Calculer: $h_1' = SHA256$ (To_1, $SHA256$ (SK_{CR} || ID_{DRi}),TW_{CR})

- Si $h_1' = h_1$ alors continuer sinon demande au CR de réessayerl'envoie

- Sauvegarder PK_{DRi}, SK_{DRi}, et $S = SHA256$ (SK_{CR} ||ID_{DRi})

3.2.2.3.Phase d'authentification mutuelle et d'échange de clés

- les drones échangent des clés et s'authentifient mutuellement avec serveur cloud
- le serveur cloud partage les clés avec tous les éléments pour la communication

DR $_i$	CSl_k
-Sélectionne $ID_{DR}i$ - Création d'un nombre aléatoire : na_1 -Générer une clé scrète aléatoire: $SK_{DR}i \in Z^*p$ -Calculer sa clé publique: $PK_{DR}i = SK_{DR}i \bullet P$ - Générer une fenêtre de temps:TW_{DRi} - Calculer: $N1 = na_1 \bullet P \bullet SK_{DRi}$ - Calculer: $To_{DRi} = SHA256\ (SK_{DRi} \| ID_{DRi}) \oplus S$ - Calculer: $AutJ_{DRi} = SHA256\ (N1 \| ID_{DRi} \| TW_{DRi} \| To_{DRi})$ - Calculer: $CPKcsl_k(AutJ_{DRi}, na_1, ID_{DR}i, TW_{DRi})$	
<CPKcsl$_k$($AutJ_{DRi}$, na, $ID_{DR}i$, TWDRi), TWDRi> →	
	- Déchiffrer $CPKcsl_k(AutJ_{DRi}, na_1, ID_{DR}i,$ TWDRi) Avec la clé priver SK_{csl} - Vérifie $\| TW'_{DRi} - TW_{DR}i \| \leq \Delta T$ Accepter sinon reject -Chercher la valeur *Cert* avec ID_{DRi}dans la **Blockchain** - Calculer: $AutJ'_{DRi} = SHA256\ (N1 \| IDDRi \| TWDRi \|$ ***Cert***) -***Vérifie*** $AutJ'_{DRi} = AutJ_{DRi}$ ***si*** $AutJ'_{DRi} = AutJ_{DRi}$alor***authentification avec succès marque dans le DR*** ***sinon échec de authentification « rependre la phase 2»*** - création d'un nombre aléatoire :na_2 - Générer une fenêtre de temps:TW_{CSlk}

- Calculer: $N2 = na \bullet P \bullet SK_{CSlk}$

$AutJ_{CSlk} = SHA256\ (N2 \| TW_{CSlk} \| Cert)$

- Calculer:

$SKik = CDKf(\ ID_{DRi} \| Cert \| TW_{DRi} \| TW_{CSlk})$

- Calculer: $CPK_{DRi}(AutJ_{CSlk}j, na_2, TW_{CSlk})$

< $CPK_{CSlkj}\ (AutJ_{CSlk}, na_2, TW_{CSlk}), TW_{CSlk}$> sur un canal sécurisé

←

Déchiffrer $CPK_{CSlkj}\ (AutJ_{CSlk}, na_2, TW_{CSlk})$, en utilisant SK_{DRi}

- Vérifie $| TW'_{CSlk} - TW_{CSlk}| \leq \Delta T$

Accepter sinon reject

- Calculer:

$AutJ^*_{CSlk} = SHA256\ (na2 \bullet PK_{CSlk} \| TW_{CSlk} \| To_{DRi})$

- Vérifier **si**: $AutJ^*_{CSlk} = AutJ_{CSlk}$

alor *authentification avecsuccès*

marque dans le CSlsinon échec

de authentification

Calculer: $SK_{ik} = CDKf(\ ID_{DRi} \| To_{DRi} \| TW_{DRi} \| TW_{CSlj})$

Algorithme 3.2. Phase III: Phase d'authentification mutuelle et d'échange de clés[38][39]

❖ Explication d'Algorithme

Étape 1

-Sélectionne identificateur au niveau des drones : ID_{DRi}

- création d'un nombre aléatoire $na_1 \in Z^*_p$

-Générer une clé scrète aléatoire ($SK_{0DRi} \in Z^*_p$) puis Calculer sa clé publique ont utilisant $PK_{0\ DRi} = SK_{0DRi} \bullet P$

-Générer une fenêtre de temps (TW_{DRi}) pour connaître l'heure d'envoi et de réception

- Calculer: $N1 = na1 \bullet P \bullet SKDRi$

- Calculer jeton To_{DRi} $SHA256$ (SK_{DRi} || ID_{DRi}) $\oplus$ $\boldsymbol{S}$ sachant bien que $\boldsymbol{s}$ indique dans algorithme 3.2.

- Calculer $AutJ_{DRi}$ puis chiffré ($AutJ_{DRi}$, na1, ID_{DRi}, TW_{DRi}) avec la clé publique de CSl et envoie auCSl

Étape 2

- la réception du message < CPKcsl*k* ($AutJ_{DRi}$, na1, ID_{DRi}, TW_{DRi}), TW_{DRi} >
- le csl le déchiffre à l'aide de sa clé secrète SK_{csl}

-Vérifie | TW'_{DRi} - TW_{DRi}| si inférieure ou égale le délai maximal de transmission associé à un message ΔT si oui accepter sinon rejeter

- le Csl recherchera la valeur *Cert* correspondante à l'aide IDcsl*i* dans la liste des DR enregistrée sur la Blockchain dans la phase 2
- calcule le jeton d'authentification $AutJ'_{DRi}$= $SHA256$ ($N1$ || ID_{DRi} || TW_{DRi} || ***Cert***)
- **validation de véracité** si $AutJ'_{DRi=}$ $AutJ_{DRi}$ si oui alors *authentification avec succès* sinon échec de authentification

Étape 3

- Création d'un nombre aléatoire $na_2 \in Z^*p$ par csl

- Générer une fenêtre de temps: TW_{CSlk}

- Calculer: $N2 = na \bullet P \bullet SK_{CSlk}$

- Calcule le jeton d'authentification $AutJ_{CSlk} = SHA256\ (N2 \parallel TW_{CSlk} \parallel Cert)$

- chiffrer $AutJ_{CSlk}\ j$ et na_2 et TW_{CSlk} avec la clé publique de DR et envois sur un canal sécurisé

- Pour **réauthentification** dans les prochaines sessions de communication entre ces deux noeuds: $SKik = CDKf\ (ID_{DRi} \parallel Cert \parallel TW_{DRi} \parallel TW_{CSlk})$ en utilisant la fonction de dérivation de clé de session

Étape 4

- la réception du message $CPK_{CSlk}\ j\ (AutJ_{CSlk}\ j, na_2, TW_{CSlk}), TW_{CSlk}>$

- le DR le déchiffre à l'aide de sa clé secrète SK_{DR}

-Vérifie $| TW'_{CSl} - TW_{CSl}|$ si inférieure ou égale le délai maximal de transmission associé à un message ΔT si oui accepter sinon rejeter

- Calculer le jeton $AutJ^*_{CSlk} = SHA256\ (na2 \bullet PK_{CSlk} \parallel TW_{CSlk} \parallel To_{DRi})$

- **validation de véracité** Vérifier **si**: $AutJ^*_{CSlk} = AutJ_{CSlk}$ **alor** *authentification avec succès* sinon échec d'authentification

- Pour **réauthentification** dans les prochaines sessions de communication entre ces deux noeuds: $SKik = CDKf\ (ID_{DRi} \parallel To_{DRi} \parallel TW_{DRi} \parallel TW_{CSli})$en utilisant la fonction de dérivation de clé de session

3.2.2.4.Phase deconsensus

Il y a beaucoup d'algorithmes de consensus mais **PBFT** à un fonctionnement efficace dans les systèmes asynchrones (pas de limite supérieure quant au moment où la réponse à la demande sera reçue). Il est optimisé pour un faible temps d'exécution. Son objectif était de résoudre de nombreux problèmes associés aux solutions de tolérance aux pannes déjà disponibles. Les domaines d'application comprennent l'informatique distribuée et la

blockchain.

Le mécanisme de travail dans ce cas et de créer un grand registre avec le consensus de PBFT, On suppose qu'il y a nombre de RG avec la possibilité d'écrire un bloc dans le grand registre dans chacun FZ. [40]

Pour algorithme de consensus il y a 2 rôles le **Speaker** et le**Congressmen**.

Dans étape initiale ont défini par défaut l'un des RG speaker les autres Congressmen qui participent dans le vote lancé par le Speaker sachant que le speaker ne participe pas dans le vote [41] ; la figure 8 explique le mécanisme de vote

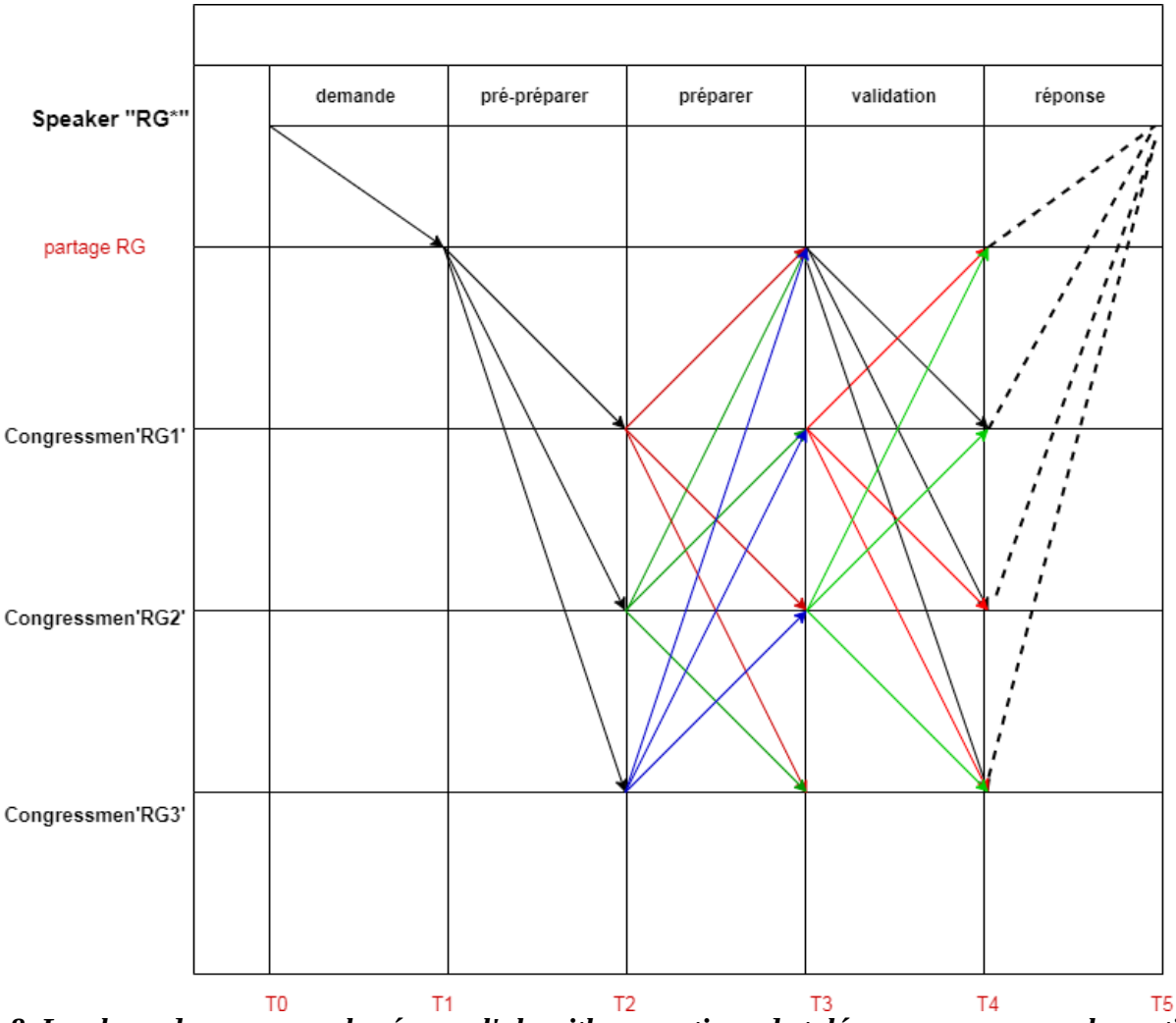

Figure 8 La phase de consensus basée sur l'algorithme pratique de tolérance aux pannes byzantine (PBFT) [40]

- ***Étape 1:*** Sélectionner le président par équation suivante : $P = (B_i \bmod N) + 1$

  - B_i le bloc actuel.
  - N le nombre de RG.

- ***Étape 2:*** lorsque authentification effectuer avec succès entre $\mathbf{DR_i}$ et CSl_k le *k* de CSl partage les résultats avec les $\mathbf{RG_k}$

- ***Étape 3:*** les RG stockent les résultats et transfèrent dans le grand registre public.

- ***Étape 4:*** Le processus de vote commence après étape 3 par le président et envoie la requête $< Vote_{req}, B_i, RG_k, bloc, Sign_{RGk}(block) >$ sachant que :

  - $Vote_{req}$: demande de vote aux membres de congrès
  - B_i : le bloc actuel.
  - RG_p : président
  - Bloc : le bloc actuel
  - $Sign_{RGk}$: désigne

  la signature du bloc avec la clé public RGk La requête envoie par RG partager à tous les membres de congrès

- ***Étape 5 :*** Les membres du congrès lorsque reçu 2f messages *Préparer* des autres membres et le *Pré-Préparer* associé. Par conséquent, ils envoient à tous les autres membres un message *Validation*.

- ***Étape 6 :*** Après avoir reçu $2f + 1$ *Validation* associées, le $N_{ème}$ RG partage son vote en utilisant $< Vote_{res}, B_i, RG_N, bloc, Sign_{RGN}(bloc) >$

- ***Étape7 :*** Après avoir terminé le vote, le bloc contenant les résultats d'authentification est ajouté au grand registre immédiatement.

3.2.2.5.<u>Phase de mise à jour des drones</u>

Les drones se déplacent d'une zone (***fz)*** à l'autre, il est donc obligatoire réauthentifie de manier rapide et sécuriser

Le DR envoie une demande d'accès au CSl $Acc_{req} =$ < Mescrep, TW_{DRi}, *h*> sachant que :

- TW_{DRi} une fenêtre de temps pour connaître l'heure d'envoi et de réception
- *h* le hachage de *de* ID_{dri} et de TW_{DRi}
- Mescrep : CPK_{CSlj} < ID_{DRi}, TW_{DRi}>

Une fois le CSl réceptionne le message il déchiffre le Mescrep avec la clé priver de CSl et extrait le ID_{DRi}, TW_{DRi}

- Calculer h* de CSl
- Compare h eth*

Si h égale h* alors cherche *Cert* par ID_{dri} si le *Cert* exista l'authentification a été effectuée dans le passé et demande au DR de ré-authentifier si la valeur de *Cert* n'existe que dans le registre local puis dans les grandes registre les DR déclarée illégale et registrer ID_{DRI} comme illégale dans les registres et informes tous les CSl de ne fournissent pas les services au DR avec cette ID. [38] [39]

CHAPITRE IV

Implémentation

ChapitreIV

Implémentation

Dans ce chapitre, nous présentons les détails de l'implémentation du schéma d'authentification et les résultats fournis,Nous utilisons un outil pour analyser la sécurité et l'efficacité du schéma proposé. Cet outil est connu sous le nom d’avispa.

4.1. L'outil AVISPA

4.1.1. Definition

AVISPA (Validation automatisée des protocoles et applications de sécurité Internet) est un outil open source pour décrire les protocoles de sécurité et leurs propriétés de sécurité. Il fournit un langage formel modulaire et expressif pour spécifier les protocoles, ainsi qu'un ensemble d'outils pour les formulaires de validation, et intègre différents back-ends qui mettent en œuvre une variété de techniques d'analyse automatique.

4.1.2. Architecture de l'outil AVISPA:

La structure de l'outil AVISPA est présentée à la **Fig 9.** Une spécification HLPSL est traduite en format intermédiaire (IF), en utilisant un traducteur appelé **hlpsl2if.** IF est un langage de plus bas niveau inférieur à HLPSL et est lus directement par les back-ends de l'outil AVISPA , Le HLPSL définit les protocoles entre les différentes fonctions telles que les rôles de base qui représentent le rôle de chaque participant, des transitions qui représentent le comportement de chaque participant en fonction de son état, des rôles de composition qui représentent des scénarios des rôles de base, et un rôle de niveau supérieur nommé environnement pour initier les variables d'environnement et instancier des sessions pour les rôles. Chaque rôle est indépendant des autres, obtenant des informations initiales par paramètres, communiquant avec les autres rôles par canaux.

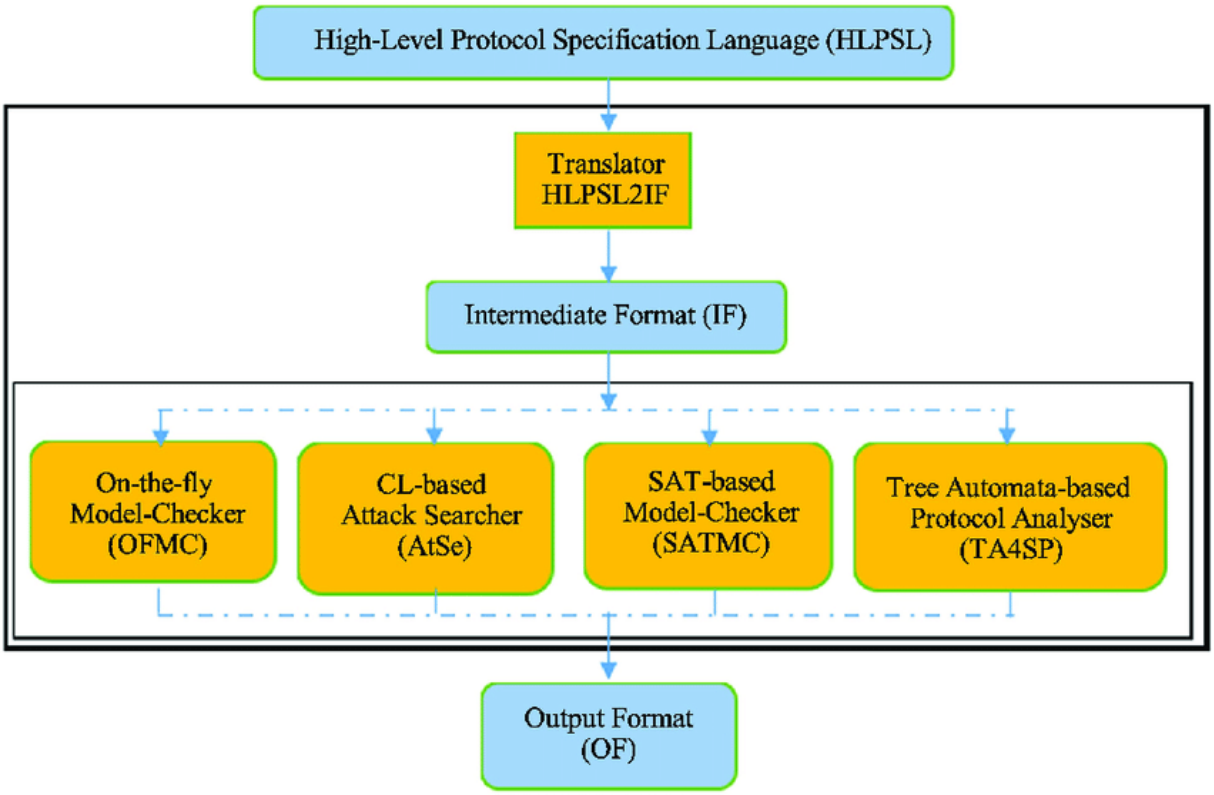

Figure 9 Architecture de l'outil AVISPA

AVISPA s'appuie sur le support de quatre back-end différents pour valider tout protocole de sécurité ***(OFMC, AtSe, SATMC ,TA4SP)***

- Vérificateur de modèle On-the-Fly **(OFMC)**: utilisé pour la détection rapide des attaques et pour prouver l'exactitude du protocole.
- Chercheur d'attaque basée sur CL **(CL-AtSe)**: trouver les attaques sur les protocoles.
- Vérificateur de modèle basé sur SAT **(SATMC)**: utilisé pour découvrir les attaques sur les protocoles et prouver que le protocole satisfait ses exigences de sécurité.
- Analyseur de protocole basé sur l'arborescence des automates **(TA4SP)**: utilisé pour conclure que les propriétés de confidentialité.

4.2. Implémentation

4.2.1. Code

le schéma compose deux agents principaux DR_i et Csl_k, aux niveaux du code les agents connu comme un rôle (le rôle DR et le rôle Csl) avec le rôle de composition « session » et le rôle de « environnement »,Explication de contenu du code en détail nous montrerons ci-dessous.

- ***Le Rôle DR$_i$***

```
%%%%%%%%%%%% Le Noeud DRi %%%%%%%%%%%%
role role_DR(DR:agent,CSL:agent,Cc2:agent,
             P:nat,
             PKdr:public_key,PKcls:public_key,PKcc2:public_key,
             CDKf,SHA256,Mult:hash_func,
             SND,RCV,CHECKcc2,VALIDcc2:channel(dy))
played_by DR
def=
        local
                State:nat,TWdr:text,Nn1:text,N1:message,To:message,Authdr:message,
                TWcls:text,Nn2:text,Authcls:message,
                SKij:message,CERTcls:message
        init
                State := 0
        transition
                1. State=0 /\ RCV(start) =|>
                   State':=2
                   /\ TWdr':=new()
                   /\ Nn1':=new()
                   /\ N1' := Mult(Nn1'.Mult(P.inv(PKdr)))
                   /\ To' := SHA256(inv(PKdr).DR)
                   /\ Authdr':=SHA256(TWdr'.N1'.DR.To')
                   /\ secret({Authdr'.Nn1'},sec_1,{DR,CSL})

                   /\ SND({Authdr'.DR.Nn1'.TWdr'}_PKcls.TWdr')
                2. State=2 /\ RCV({Authcls'.Nn2'.TWcls'}_PKdr.TWcls') =|>
                   State':=4
                   /\ CERTcls':=Authcls
                   /\ secret({CERTcls'.To},sec_6,{Cc2,DR})

                   /\ CHECKcc2({CERTcls'.Nn2'.TWcls'.To}_PKcc2)
                3. State=4 /\ VALIDcc2({CERTcls}_PKdr) =|>
                   State':=6
                   /\ request(DR,Cc2,auth_2,CERTcls)
                   /\ SKij':=CDKf(DR.To.TWdr.TWcls)
end role
```

Figure 10 Rôle DR.

Le rôle DR_i se compose de :

Les paramètres d'entrer d'environnement :

- les agents « les rôles DR ,Csl,Cc2 »
- les fonctions : **SHA256** pour calculer les hachages ;

  Mult pour l'opération multiplicative ECC (•) ;

  CDKf pour la dérivation des clés de session ;
- les canaux : **SND ,RCV** pour la communication avec le rôle Csl;

  CHEKcc2 ,VALIDcc2 pour communication avec le rôle Contrôleur cc2 ;

Les variables locales : States les étapes

TWdr,TWcsl les fenêtres de temps

Authcsl, **Authdr** ,**CERTdr** les jetons d'authentification

To signature

les transitions:

1. le calcul du jeton d'authentification DR qui va être ensuite transmis au CSL
2. le DR réceptionnait le jeton d'authentification de CSL
3. le jeton réceptionnait de DR transmet au CC2 pour la validité du jeton et DR procède au calcul de la clé de session SKij

- ***Le Rôle CSL_j***

```
%%%%%%%%%%%%% Le Noeud CSL %%%%%%%%%%%%%
role role_CSL(DR:agent,CSL:agent,Cc1:agent,
              P:nat,
              PKcls:public_key,PKdr:public_key,PKcc1:public_key,
              CDKf,SHA256,Mult:hash_func,
              SND,RCV,CHECK,VALID:channel(dy))
played_by CSL
def=
        local
                State:nat,TWdr:text,Nn1:text,Authdr:message,
                CERTdr:message,IDdr:message,Cert:message,
                TWcls:text,Nn2:text,N2:message,Authcls:message,SKij:message
        init
                State := 1
        transition
                1. State=1 /\ RCV({Authdr'.DR.Nn1'.TWdr'}_PKcls.TWdr') =|>

                   State':=3

                   /\ IDdr':=DR
                   /\ CERTdr':= Authdr
                   /\ secret({CERTdr'.IDdr'.TWdr.Nn1},sec_3,{Cc1,CSL})
                   /\ CHECK({CERTdr'.IDdr'.TWdr.Nn1}_PKcc1)
                2. State=3 /\ VALID({CERTdr.Cert'}_PKcls) =|>
                   State':=5
                   /\ request(CSL,Cc1,auth_1,CERTdr)
                   /\ TWcls':=new()
                   /\ Nn2':=new()
                   /\ N2' := Mult(Nn2'.Mult(P.inv(PKcls)))
                   /\ Authcls':=SHA256(N2'.TWcls'.Cert)
                   /\ secret({Authcls'.Nn2'},sec_2,{DR,CSL})

                   /\ SND({Authcls'.Nn2'.TWcls'}_PKdr.TWcls')
                   /\ SKij':=CDKf(IDdr.Cert.TWdr.TWcls')
end role
```

Le rôle CSL_j se compose de :

Les paramètres d'entrer d'environnement :

- les agents « les rôles DR ,Csl,Cc1 »
- les fonctions : **SHA256** pour calculer les hachages ;

  Mult pour l'opération multiplicative ECC (□) ;

  CDKf pour la dérivation des clés de session ;
- les canaux : **SND ,RCV** pour la communication avec le rôle Csl;

  CHEK ,VALID pour communication avec le rôle Contrôleur cc1 ;

Les variables locales : States les étapes

TWdr,TWcsl les fenêtres de temps

Authcsl, **Authdr** ,**CERTdr** les jetons d'authentification

To signature

Cert (signature du DR obtenue du cc1)

Les transitions:

1. après la réception du jeton d'authentification de DRi, le Cslj le transmet au CC1 pour le valider

2. après la validation du jeton d'authentification du DRi le Cslj calcule son jeton d'authentification et les transmis au DRi ensuite il procède au calcul de la clé de session SKij

- ***Le Rôle CC1 ,CC2***

```
role role_Cc1(CSL:agent,Cc1:agent,
              P:nat,
              PKcls:public_key,PKdr:public_key,PKcc1:public_key,
              SHA256,Mult:hash_func,
              CHECK,VALID:channel(dy))
played_by Cc1
def=
        local
                State:nat,TWdr:text,Nn1:text,IDdr:message,Cert:message,CERTdr:message
        init
                State := 0
        transition
                1. State=0 /\ CHECK({CERTdr'.IDdr'.TWdr'.Nn1'}_PKcc1) =|>
                   State':=1
                   /\ Cert':= SHA256(inv(PKdr).IDdr)
                   /\ CERTdr':=SHA256(TWdr'.Mult(Nn1'.Mult(P.inv(PKdr))).IDdr.Cert')
                   /\ secret({CERTdr'.Cert'},sec_4,{Cc1,CSL})
                   /\ VALID({CERTdr'.Cert'}_PKcls)
                   /\ witness(Cc1,CSL,auth_1,CERTdr')
end role
```

Figure 11 Rôle Contrôleur composé1 .

```
role role_Cc2(DR:agent,Cc2:agent,
              P:nat,
              PKcls:public_key,PKdr:public_key,PKcc2:public_key,
              SHA256,Mult:hash_func,
              CHECKcc2,VALIDcc2:channel(dy))
played_by Cc2
def=
        local
                State:nat,TWcls:text,Nn2:text,IDdr:message,To:message,CERTcls:message
        init
                State := 0
        transition
                1. State=0 /\ CHECKcc2({CERTcls'.Nn2'.TWcls'.To'}_PKcc2) =|>
                   State':=1
                   /\ CERTcls':=SHA256(Mult(Nn2'.Mult(P.inv(PKcls))).TWcls.To)
                   /\ secret({CERTcls'},sec_5,{Cc2,DR})
                   /\ VALIDcc2({CERTcls'}_PKdr)
                   /\ witness(Cc2,DR,auth_2,CERTcls')
end role
```

Figure 12 Rôle Contrôleur composé 2 .

Le contrôleur composé 1 et 2 à des rôles supplémentaires à cause de la limite des opérations dans l'outil avispa « les calculs et les comparaisons»

Dans ce cas :

- le Rôle de CC1 calculer le jeton *Auth** DR*i*, ensuite l'envoie au CLS pour qu'il puisse déterminer la validité de *Auth*DR*i* reçu
- le Rôle de CC2 calculer le jeton *Auth** CSl*j*, ensuite l'envoie au DR pour qu'il puisse déterminer la validité de *Auth*CSl*j* reçu

Le Rôle Session

La composition de la session si tous les paramètres qui participe au niveau de la session

- **Les paramètres d'entrer d'environnement :**
  - les agents
  - les clés
  - les fonctions
  - les canaux
- **les canaux**
- **la composition :** dans cette partie la session fait appel aux autres rôles avec ces paramètres pour le déroulement de processus

```
role session(DR:agent,CSL:agent,Cc1:agent,Cc2:agent,
             P:nat,
             PKcls:public_key,PKdr:public_key,PKcc1:public_key,PKcc2:public_key,
             CDKf,SHA256,Mult:hash_func)
def=
        local
                CHECK4,VALID4,CHECK3,VALID3,CHECK2,VALID2,CHECK1,VALID1,SND2,RCV2,SND1,RCV1:channel(dy)

        composition
                role_Cc1(CSL,Cc1,P,PKcls,PKdr,PKcc1,SHA256,Mult,CHECK2,VALID2)
             /\ role_CSL(DR,CSL,Cc1,P,PKcls,PKdr,PKcc1,CDKf,SHA256,Mult,SND2,RCV2,CHECK1,VALID1)
             /\ role_DR(DR,CSL,Cc2,P,PKdr,PKcls,PKcc2,CDKf,SHA256,Mult,SND1,RCV1,CHECK3,VALID3)
             /\ role_Cc2(DR,Cc2,P,PKcls,PKdr,PKcc2,SHA256,Mult,CHECK4,VALID4)

end role
```

Figure 13 Rôle Session.

Le Rôle environnement

- **les variables globales :** les clés, les agents, les identificateurs, les fonctions
- **la spécification des connaissances**
- **la composition :** dans cette partie les variables et les spécifications passent comme des paramètres au rôle session

```
%%%%%%%%%%%% L'environment essentielle pour l'execution du processus %%%%%%%%%%%%
role environment()
def=
        const

                p:nat,
                pkcc1,pki,pkcls,pkdr,pkcc2:public_key,
                dr,cls,cc1,cc2:agent,
                hash_0,cdfk,sha256,mult:hash_func,
                sec_1,sec_2,sec_3,sec_4,sec_5,sec_6,auth_1,auth_2:protocol_id
        intruder_knowledge = {dr,cls,pkdr,pkcls,pki,inv(pki),cdfk,sha256,mult}
        composition

        session(dr,cls,cc1,cc2,p,pkcls,pkdr,pkcc1,pkcc2,cdfk,sha256,mult)
end role

%%%%%%%%%%%% Les exigences %%%%%%%%%%%%
goal
        secrecy_of sec_1
        secrecy_of sec_2
        secrecy_of sec_3
        secrecy_of sec_4
        secrecy_of sec_5
        secrecy_of sec_6
        authentication_on auth_1
        authentication_on auth_2
end goal
environment()
```

Figure 14 Rôle environnement.

4.2.2. Exécution :

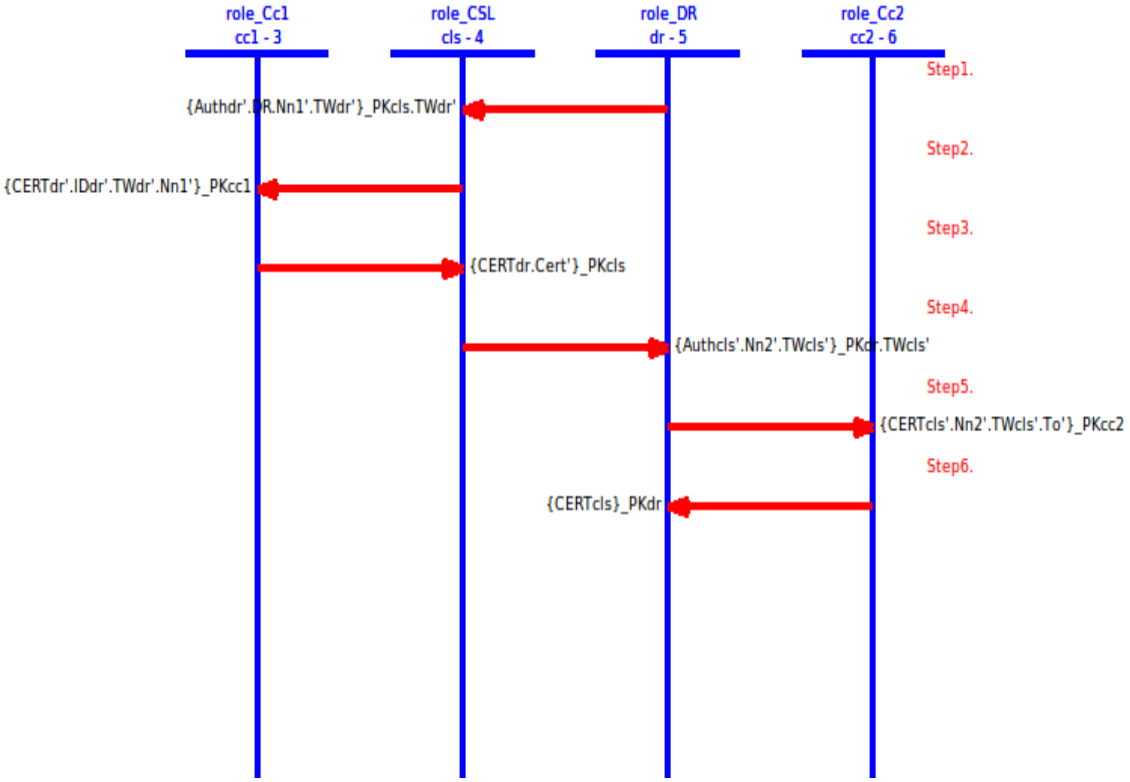

Figure 15 protocole Simulation d'authentification sur AVISPA.

Dans cette partie on parle sur Les étapes d'exécution

- Etape 1 :
  - Le DR calculer le jeton **Authdr** avec ses propres variables et envoie au CSL
  - Le DR assure la confidentialité de message entre lui et le CSL
- Etape 2 :
  - Le CSL reçoit le message et change son état
  - Valider le TW_{DR}
  - Enregistre IDdr et Authdr dans CERTdr
  - Recalculer le CERTdr et la comparer avec l'ancienne valeur
  - la condition d'authenticité ajoutée sur le jeton CERTdr
  - CERTdr reçue doit être égale à CERTdr envoyée
  - ajoute une condition de confidentialité sur le message pour qu'il soit secret entre CSL et CC1
- Etape 3:
  - Lorsque CC1 reçoit le message
  - Il récupère la signature **Cert** de DR en se basant sur son **IDdr**
  - reconstruit le jeton **CERTdr** il les renvoie à CSL
  - ajouter la condition de confidentialité et une condition d'authenticité la valeur qu'il envoyée soit égale à la valeur qu'il a reçue.
- Etape 4:
  - le CSL parvient à recevoir le message envoyé par CC1 cela signifie que **CERTdr** est valide donc **DR** est marqué authentique.
  - CSL Génère ses propres variables et calcule son authentification **AuthTcsl** et il les envoie à DR
  - Ajouter une condition de confidentialité enter CSL et DR
  - Le CSL génère une clé de session **SKij** pour les communications futures avec le DR.
- Etape 5:
  - Le DR reçoit le message et change son état
  - Valider le TW_{CSL}

- Enregistre IDcsl et Authcsl dans CERTcsl
- Recalculer le CERTcsl et le comparer avec l'ancienne valeur
- la condition d'authenticité ajoutée sur le jeton CERTcsl
- CERTcsl reçue doit être égale à CERTcsl envoyée
- ajoute une condition de confidentialité sur le message pour qu'il soit secret entre DR et CC2

- Etape 6:
  - Lorsque CC2 reçoit le message
  - reconstruit le jeton **CERTcsl** il les renvoie à DR
  - ajouter la condition de confidentialité et une condition d'authenticité la valeur qu'il envoyée soit égale à la valeur qu'il a reçue.
- Etape 7:
  - le DR recevoir le message envoyé par CC2 cela signifie que **CERTcsl** est valide donc le **CSL** est marqué authentique.
  - Le DR génère une clé de session **SKij** pour les communications futures avec le CSL.

4.3. Résultats:

Les deux supports de back-end pour vérifier la confidentialité et l'intégrité des messages transmirent entre les nœuds DRi et CSLj

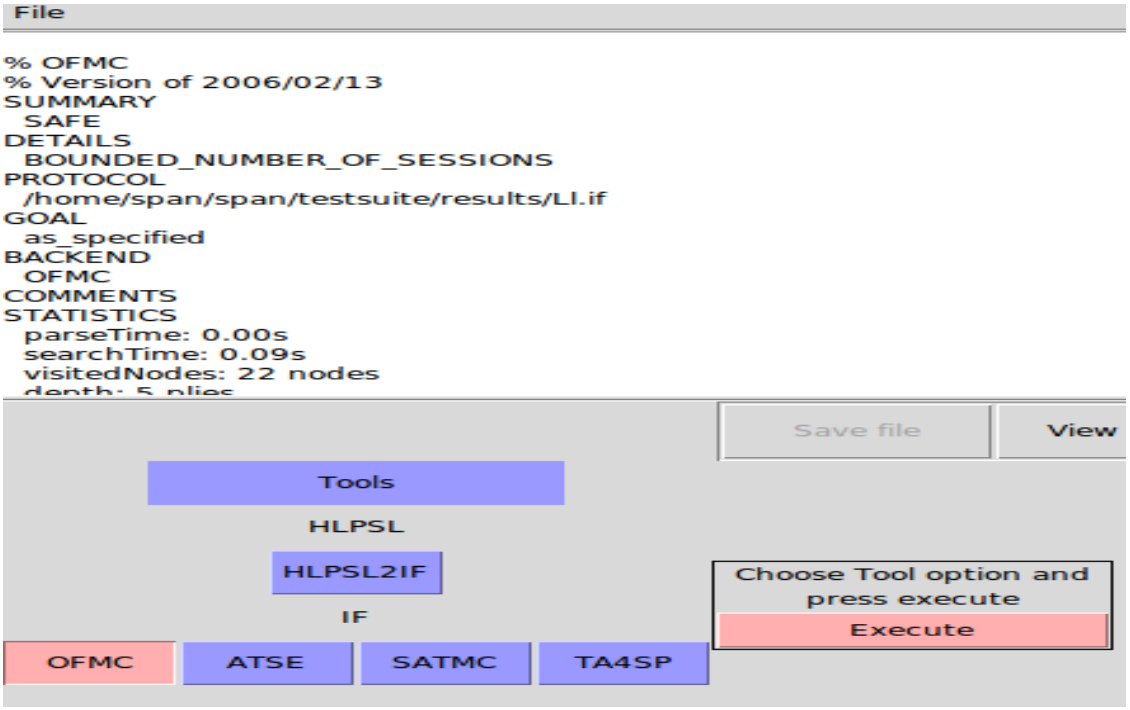

Figure 16 Vérificateur de modèle On-the-Fly (OFMC).

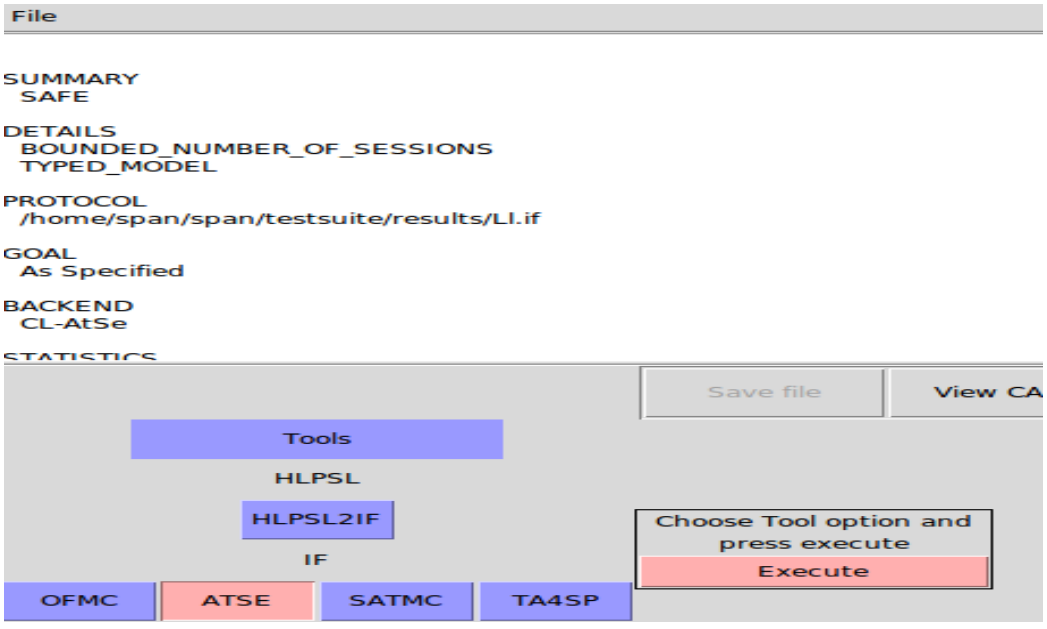

Figure 17 Chercheur d'attaque basée sur CL (CL-AtSe).

Donc notre schéma assure la confidentialité et l'intégrité des messages transmit.

4.4. Analyse de Sécurité:

4.4.1. Les objectifs de sécurité:

- **Confidentialité:**

1. il est difficile pour les attaquants de calculer la clé privée à partir de la clé publique selon le problème du logarithme discret de la courbe elliptique, donc SK_{CSLj} et SK_{DRi} sont difficiles à calculer.
2. Les informations privées transmises sont toujours cryptées dans la phase d'enregistrement entre DR et CR et dans la phase d'authentification entre DR et CSL.
3. Seuls les nœuds autorisés par la loi peuvent accéder au registre public.

Par conséquent, le schéma proposé garantit la confidentialité des informations clés tout au long de ses phases.

- **Intégrité:**

1. il faut s'assurer qu'aucun adversaire ne peut altérer les messages transmis et que les messages altérés puissent être découverts.
2. il est difficile pour les attaquants de calculer la clé secrète SKx.
3. chaque échange de message, les nœuds effectuent un test d'intégrité basé sur la fonction d'hachage SHA256.
4. Dans la phase d'enregistrement les identités réelles des DR et des GSS sont cryptées par la clé publique PK_{CR} de CR et accompagnées par un hash *h*. Un adversaire ne possédant pas la clé privée SK_{CR} pour décrypter les messages envoyés au CR et ne pouvant pas inversé le résultat de *h*, ne pourra ni lire ni falsifier les messages échangés.

- **Authenticité:**

Selon la phase d'authentification il est garanti que tous les messages sont générés par des utilisateurs légitimes

1. CSL identifie DR via le jeton *AuthTok**DR construit via le certificat *Cert* correspondant à l'identité ID_{DR} du DR .

2. DR identifie CSL via le jeton *AuthTok**csl Donc l'authentification mutuelle est assurée.

- **Vie privée et Anonymat:**

L'anonymat garantit qu'aucun adversaire ne puisse extraire de véritables identités lorsque notre système est déployé dans nos system

1. Aucune véritable identité ne peut être obtenue par un adversaire car elle est toujours cachée dans les messages cryptés.
2. Le DR au lieu d'envoyer sa véritable identité, il crée un jeton To_{DRi} basé sur SHA256 ( ) et l'utilise comme sa signature.

Et pour préserver la vie privée

1. Si le DR veut changer sa signature, il doit juste être authentifié de nouveau.
2. les vraies identités sont inscrites dans le registre public, mais seules les entités autorisées par la loi peuvent y accéder.

Donc Vie privée et Anonymat est assurée

- **Traçabilité et Non-répudiation:**

Les CR responsable de la traçabilité des drones, lorsque quelqu'un découvre qu'un drone se comporte mal, il le signale au CR et enregistrer dans comme des drones malveillants En conséquence, l'objectif de non-répudiation est atteint.

- **Non-interactivité:**

Chaque fois le DR accède au service du Fog, il n'envoie qu'un seul jeton qui est la demande d'authentification à un CSL dans le cas de phase d'authentification ou la demande d'accès au service et la demande de mise à jour et n'a pas besoin de transmettre des messages supplémentaires par conséquence il n'y a aucun jeton qui dépend d'un autre. Donc, notre schéma n'est pas interactif.

4.4.2. Comparaison avec d'autre schéma:

Dans cette patrie on va faire une comparaison avec les autre schéma basson sur des critères, la comparaison relative avec deux systèmes existants le tableau 5 Après

avoir comparé notre schéma avec les schémas Zhang *et al.* [42] et Singh [43]

Caractéristiques	[42]	[43]	Notre schéma
Confidentialité	✓	✗	✓
Intégrité	✓	✗	✓
Authentification mutuel	✓	✓	✓
Vie privé et anonymat	✓	✓	✓
Traçabilité et Non-répudiation	✗	✗	✓
Non-interactivité	✗	✓	✓
Transmission sécurisé	✓	✗	✓
Échange de clés	✓	✗	✓

Table 6 Comparaison des caractéristiques de sécurité fournies par notre schéma avec le schéma [42] et [43].

Travaux futurs

dans ce travail nous proposons un schéma d'authentification sécurisé d'IOD pour atteindre l'objectif de sécurité basé sur la Blockchain et le Fog, le schéma proposé peut préserver la confidentialité, l'intégrité, la vie privée, l'anonymat, et la non-répudiation, le futur travail permet de faire évoluer le travail contre d'autres attaques réaliser dans le monde réel, cette conception permet de faire évoluer le system de sécurité des drones de tout fournisseur de services compromis pouvantmettre en situation de danger en divulguant des informations privées , cette recherche est vraiment importante car elle augmente la sécurité et les performances de ces systèmes, Ce domaine de recherche est prometteur et permet d'intervenir et de développer les fondements du domaine des drones, notamment dans le domaine de la sécurité.

Conclusion générale

Les applications de l'architecture IoD ont été largement utilisées dans divers domaines et ont apporté un grand confort d'utilisation dans le domaine militaire et civil. Ces dernières années, plusieurs schémas d'authentification pour l'IoD ont été proposés. Cependant. Nous concevons un schéma léger basé sur la technique Blockchain et ECC et pour un réseau basé sur Cloud et Fog, L'utilisation des méthodes cryptographiques ECC ont efficacement augmenté le niveau de sécurité des informations échangées lors de l'exécution des différentes phases du système analysé de sécurité de nos schémas testés avec l'outil AVISPA qui montrent que le schéma vérifie les objectives de sécurité.

Bibliographie

[1] Koubâa, A., Qureshi, B., Sriti, M. F., Javed, Y., & Tovar, E. (2017, April). A service-oriented Cloud-based management system for the Internet-of-Drones. In 2017 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC) (pp. 329-335). IEEE.

[2] Bera, B., Saha, S., Das, A. K., Kumar, N., Lorenz, P., & Alazab, M. (2020). Blockchain-envisioned secure data delivery and collection scheme for 5g-based iot-enabled internet of drones environment. IEEE Transactions on Vehicular Technology, 69(8), 9097-9111.

[3] Gupta, R., Kumari, A., & Tanwar, S. (2021). Fusion of blockchain and artificial intelligence for secure drone networking underlying 5G communications. Transactions on Emerging Telecommunications Technologies, 32(1), e4176.

[4] [site web] The 10 best camera drones in 2021, https://www.digitalcameraworld.com/buying-guides/the- 10-best-camera-drones; consulter le02/05/2021

[5] [site web] Mario LANDRY COMMANDE DE VOL NON-LINÉAIRE EN TEMPS RÉEL D'UN DRONE À VOILURE FIXE disponible en ligne:https://espace.etsmtl.ca/id/eprint/1010/1/LANDRY_Mario.pdf;(pp. 8)

[6] Oubbati, O. S., Atiquzzaman, M., Ahanger, T. A., & Ibrahim, A. (2020). Softwarization of UAV networks: A survey of applications and future trends. IEEE Access, 8, 98073-98125.

[7] Pinto, M. F., Marcato, A. L., Melo, A. G., Honório, L. M., & Urdiales, C. (2019). A framework for analyzing fog-cloud computing cooperation applied to information processing of UAVs. Wireless Communications and Mobile Computing, 2019. ; disponible en ligne: https://www.hindawi.com/journals/wcmc/2019/7497924/; (pp.4-6)

[8] [site web] Internet of Things (IoT) Protocols You Need to Know About disponible en ligne: https://www.rs-online.com/designspark/eleven-internet-of-things-iot-protocols-you-need-to-know-about; consulter le06/05/2021

[9] [site web] High-Speed Packet Access (HSPA) disponible en ligne: https://www.techopedia.com/definition/772/high-speed-packet-access-hspa; consulter le 06/05/2021

[10] Das, A. K., Wazid, M., Kumar, N., Vasilakos, A. V., & Rodrigues, J. J. (2018). Biometrics-based privacy-preserving user authentication scheme for cloud-based industrial Internet of Things deployment. IEEE Internet of Things Journal, 5(6), 4900-4913.

[11] Turkanović, M., Brumen, B., & Hölbl, M. (2014). A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion. Ad Hoc Networks, 20, 96-112.

[12] Challa, S., Wazid, M., Das, A. K., Kumar, N., Reddy, A. G., Yoon, E. J., & Yoo, K. Y. (2017). Secure signature-based authenticated key establishment scheme for future IoT applications. Ieee Access, 5, 3028-3043.

[13] Srinivas, J., Das, A. K., Kumar, N., & Rodrigues, J. J. (2019). TCALAS: Temporal credential-based anonymous lightweight authentication scheme for Internet of drones environment. IEEE Transactions on Vehicular Technology, 68(7), 6903-6916.;disponible en ligne: https://ieeexplore.ieee.org/abstract/document/8693567;

[14] Wazid, M., Das, A. K., Kumar, N., Vasilakos, A. V., & Rodrigues, J. J. (2018). Design and analysis of secure lightweight remote user authentication and key agreement scheme in Internet of drones deployment. IEEE Internet of Things Journal, 6(2), 3572-3584.

[15] Challa, S., Wazid, M., Das, A. K., Kumar, N., Reddy, A. G., Yoon, E. J., & Yoo, K. Y. (2017). Secure signature-based authenticated key establishment scheme for future IoT applications. Ieee Access, 5, 3028-3043.

[16] Turkanović, M., Brumen, B., & Hölbl, M. (2014). A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion. Ad Hoc Networks, 20, 96-112.

[17] Jiang, Q., Zeadally, S., Ma, J., & He, D. (2017). Lightweight three-factor authentication and key agreement protocol for internet-integrated wireless sensor networks. IEEE Access, 5, 3376-3392.

[18] Zhang, Y., He, D., Li, L., & Chen, B. (2020). A lightweight authentication and key agreement scheme for internet of drones. Computer Communications, 154, 455-464. disponible en ligne: https://www.sciencedirect.com/science/article/abs/pii/S0140366419319358;

[19] Wazid, M., Das, A. K., Kumar, N., Vasilakos, A. V., & Rodrigues, J. J. (2018). Design and analysis of secure lightweight remote user authentication and key agreement scheme in Internet of drones deployment. IEEE Internet of Things Journal, 6(2), 3572-3584. http://dx.doi.org/10.1109/JIOT.2018.2888821.

[20] Singh, J., Gimekar, A., & Venkatesan, S. (2019). An efficient lightweight authentication scheme for human- centered industrial Internet of Things. International Journal of Communication Systems, e4189.,http://dx.doi.org/10.1002/dac.4189.

[21] Ever, Y. K. (2020). A secure authentication scheme framework for mobile-sinks used in the internet of drones applications. Computer Communications, 155, 143-149.disponible en ligne: https://www.sciencedirect.com/science/article/abs/pii/S014036641930790X

[22] Al-Turjman, F., Ever, Y. K., Ever, E., Nguyen, H. X., & David, D. B. (2017). Seamless key agreement framework for mobile-sink in IoT based cloud-centric secured public safety sensor networks. IEEE Access, 5, 24617-24631.

[23] Das, A. K., Sutrala, A. K., Kumari, S., Odelu, V., Wazid, M., & Li, X. (2016). An efficient multi- gateway- based three- factor user authentication and key agreement scheme in hierarchical wireless sensor networks. Security and Communication Networks, 9(13), 2070-2092.

[24] Srinivas, J., Mishra, D., & Mukhopadhyay, S. (2017). A mutual authentication framework for wireless medical sensor networks. Journal of medical systems, 41(5), 80.

[25] Gope, P., & Sikdar, B. (2020). An efficient privacy-preserving authenticated key agreement scheme for edge-assisted internet of drones. IEEE Transactions on Vehicular Technology, 69(11), 13621-13630. disponible en ligne: https://ieeexplore.ieee.org/abstract/document/9174893/

[26] Tian, Y., Yuan, J., & Song, H. (2019). Efficient privacy-preserving authentication framework for edge-assisted Internet of Drones. Journal of Information Security and Applications, 48, 102354.

[27] Zhang, Y., He, D., Li, L., & Chen, B. (2020). A lightweight authentication and key agreement scheme for internet of drones. Computer Communications, 154, 455-464.

[28] Srinivas, J., Das, A. K., Kumar, N., & Rodrigues, J. J. (2019). TCALAS: Temporal credential-based anonymous lightweight authentication scheme for Internet of drones environment. IEEE Transactions on Vehicular Technology, 68(7), 6903-6916.

[29] Gope, P., Millwood, O., & Saxena, N. (2021). A provably secure authentication scheme for RFID-enabled UAV applications. Computer Communications, 166, 19-25. disponible en ligne:https://www.sciencedirect.com/science/article/abs/pii/S0140366420319897

[30] Moriyama, D., Matsuo, S. I., & Yung, M. (2013). PUF-based RFID authentication secure and private under memory leakage. IACR Cryptol. ePrint Arch, 3, 61-83. http://eprint.iacr.org/2013/712.

[31] Aysu, A., Gulcan, E., Moriyama, D., Schaumont, P., & Yung, M. (2015, September). End-to-end design of a PUF-based privacy preserving authentication protocol. In International Workshop on Cryptographic Hardware and Embedded Systems (pp. 556-576). Springer, Berlin, Heidelberg.

[32] Lopez, J., & Dahab, R. (2000). An overview of elliptic curve cryptography.

[33] [site web] SHA256 – Algorithme de hachage SHA256.https://cryptostrategie.com/sha256-algorithme-bitcoin/consulter :13/06/2021.

[34] Rachmawati, D., Tarigan, J. T., & Ginting, A. B. C. (2018, March). A comparative study of Message Digest 5 (MD5) and SHA256 algorithm. In Journal of Physics: Conference Series (Vol. 978, No. 1, p. 012116). IOP Publishing. disponible en ligne: https://iopscience.iop.org/article/10.1088/1742-6596/978/1/012116/pdf

[35] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. Decentralized Business Review, 21260.

[36] [site web] Blockchain https://www.futura-sciences.com/tech/definitions/informatique-blockchain-18277/ consulter : 14/06/2021.

[37] Yao, Y., Chang, X., Mišić, J., Mišić, V. B., & Li, L. (2019). BLA: Blockchain-assisted lightweight anonymous authentication for distributed vehicular fog services. IEEE Internet of

Things Journal, 6(2), 3775-3784.

[38] MERZOUGUI, S. E. (2020). Un schéma d'authentification sécurisé pour l'internet des véhicules.

[39] Bera, B., Chattaraj, D., & Das, A. K. (2020). Designing secure blockchain-based access control scheme in IoT-enabled Internet of Drones deployment. Computer Communications, 153, 229-249.

[40] Kolasa, Y., Bastogne, T., Georges, J. P., & Kubler, S. (2020, July). Quality-by-design-engineered pBFT consensus configuration for medical device development. In 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC) (pp. 5709-5713). IEEE.

[41] Coelho, I. M., Coelho, V. N., Araujo, R. P., Yong Qiang, W., & Rhodes, B. D. (2020). Challenges of PBFT-Inspired Consensus for Blockchain and Enhancements over Neo dBFT. Future Internet, 12(8), 129.

[42] Zhang, Y., He, D., Li, L., & Chen, B. (2020). A lightweight authentication and key agreement scheme for internet of drones. Computer Communications, 154, 455-464.

[43] Singh, J., Gimekar, A., & Venkatesan, S. (2019). An efficient lightweight authentication scheme for human- centered industrial Internet of Things. International Journal of Communication Systems, e4189. http://dx.doi.org/10.1002/dac.4189.