

**Institut Universitaire des Sciences**

**Faculté des Sciences et Technologies**

**Projet dans le cadre du cours de Réseaux 2**

**Simulation d'un Réseau IoT avec Cisco Packet Tracer**

**Préparé par Wendy COLAS**

**A l'attention de Monsieur Ismaël SAINT AMOUR**

**Mai 2025**

## **I. Introduction**

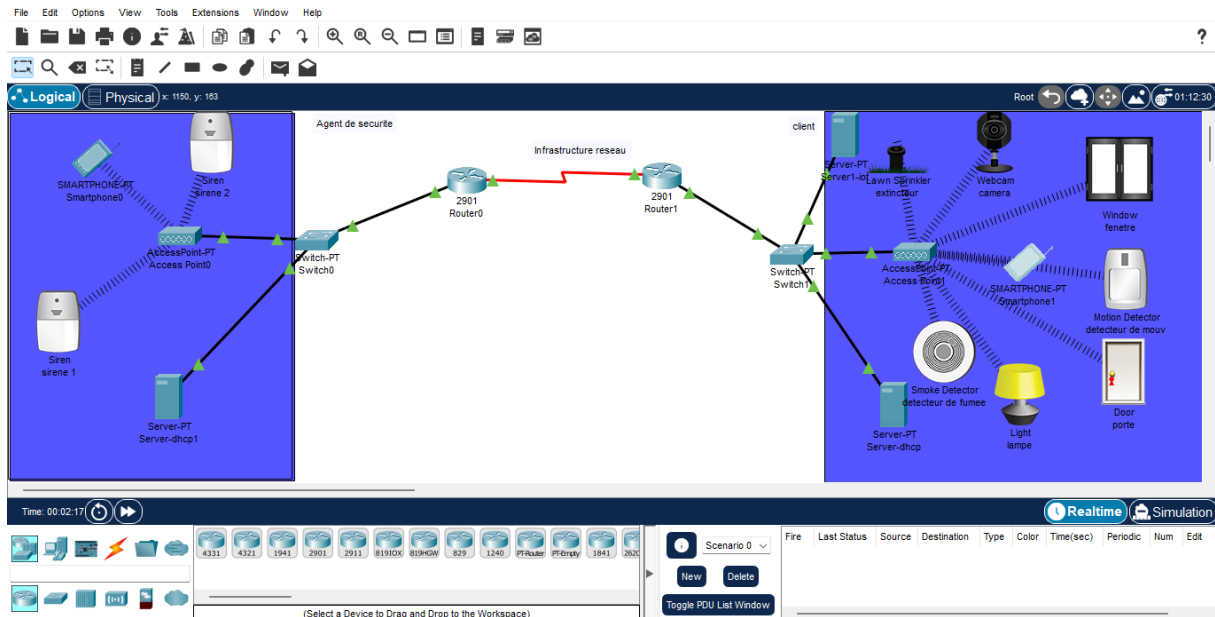
Avec l'évolution rapide des technologies, l'internet des objets (IoT) joue un rôle essentiel dans la connectivité et l'automatisation des systèmes. Les appareils connectés échangent des données en temps réel, facilitant ainsi de nombreuses applications, que ce soit dans la domotique, l'industrie ou la gestion des infrastructures intelligentes.

Ce projet vise à simuler un réseau IoT en utilisant Cisco Packet Tracer, un outil puissant permettant de créer et analyser des architectures réseau. Grâce à cette simulation, nous pouvons observer le fonctionnement des périphériques IoT, étudier la communication entre eux et comprendre les protocoles qui assurent leur interconnexion.

Dans ce rapport, nous allons détailler les étapes de conception du réseau, les dispositifs utilisés ainsi que les résultats obtenus. L'objectif est de mieux comprendre le déploiement d'un réseau IoT et ses implications pratiques.

Dans un contexte de modernisation des infrastructures hôtelières, l'intégration de technologies intelligentes devient un levier essentiel pour garantir sécurité, confort et efficacité. Le projet intitulé « Simulation d'un réseau IoT dans Cisco Packet Tracer » s'inscrit dans cette dynamique en proposant la configuration d'un réseau connecté adapté à un environnement hôtelier. L'architecture retenue repose sur une structure simplifiée composée de trois pôles : une chambre client, une chambre pour l'agent de sécurité et une infrastructure centrale. Grâce à une série de dispositifs IoT (capteurs de fumée, détecteurs de mouvement, caméras, alarmes, etc.), les différentes zones communiquent entre elles. Ainsi, en cas d'incident dans la chambre du client, comme la détection de fumée, une alerte est immédiatement transmise à l'agent de sécurité via le réseau, permettant une intervention rapide. Ce projet illustre comment les technologies IoT peuvent être exploitées pour renforcer la surveillance et automatiser les réponses aux situations critiques dans un cadre hôtelier.

## **I. Conception du réseau**



Ce réseau IoT est structuré de manière à permettre une connectivité fluide entre les dispositifs des deux chambres tout en assurant la sécurité et le contrôle via des équipements dédiés. Chaque section a un rôle bien défini, assurant une gestion efficace du réseau.

## A. La partie physique

### 1) La chambre de l'agent de sécurité

Cette section est dédiée au contrôle et à la surveillance du réseau. Elle contient plusieurs dispositifs essentiels :

- **SMARTPHONE-PT (smartphone0)** : il permet la gestion mobile du réseau et la communication avec les dispositifs IoT.
- **Sirène 1 et 2** : déclenchent une alarme en cas d'intrusion ou d'urgence.
- **AccessPoint-PT (Access Point0)** : Assure la connexion WI-FI aux appareils de cette chambre.
- **Server-PT (Server-dhcp1)** : fournit des adresses IP automatique aux appareils via le protocole DHCP.

### 2) La chambre du client

Cette partie du réseau est axée sur la connectivité domestique et la surveillance. Elle contient :

- **SMARTPHONE-PT (Smartphone1)** : Il permet de surveiller et contrôler les équipements IoT.
- **Camera** : sert à la surveillance vidéo en temps réel.
- **Fenêtre** : capteur qui détecte l'ouverture ou la fermeture de la fenêtre.

- **Détecteur de mouvement** : analyse les mouvements pour identifier une présence suspecte.
- **Porte** : capteur qui surveille l'état d'ouverture ou de fermeture de la porte.
- **Lampe** : permet de gérer l'éclairage intelligent.
- **AccessPoint-PT (Access Point1)** : assure la connexion WI-FI aux appareils du client.
- **Server-PT (Server-iot)** : gère les fonctionnalités IoT et stocke les données liées aux dispositifs connectés.
- **Server-PT (Server-dhcp)** : fournit des adresses IP automatique aux appareils via le protocole DHCP.
- **Détecteur de fumée** : permet de détecter la présence de la fumée donc éventuellement de feu dans la chambre du client.
- **Extincteur de feu** : se déclenche quand la présence de la fumée est détectée.

### 3) L'infrastructure réseau

Cette section joue un rôle de passerelle centrale et assure la communication entre les deux chambres. Elle contient :

- **Switch-PT (Switch0 et Switch1)**: ces deux switches appartiennent à l'infrastructure centrale et relient tous les équipements réseau, assurant le routage des données entre les différentes sections du réseau.
- **2901 (Router0 et Router1)** : les deux routeurs jouent un rôle fondamental dans la gestion du trafic réseau.
- **Router0** : assure la communication entre la chambre de l'agent de sécurité et l'infrastructure réseau, permettant le transfert sécurisé des données IoT.
- **Router1** : relie la chambre du client à l'infrastructure réseau et gère les connexions entre les appareils IoT et les services réseau.

Les routeurs optimisent les flux de communications et appliquent les politiques réseau (comme la gestion des adresses IP ou le filtrage du trafic).

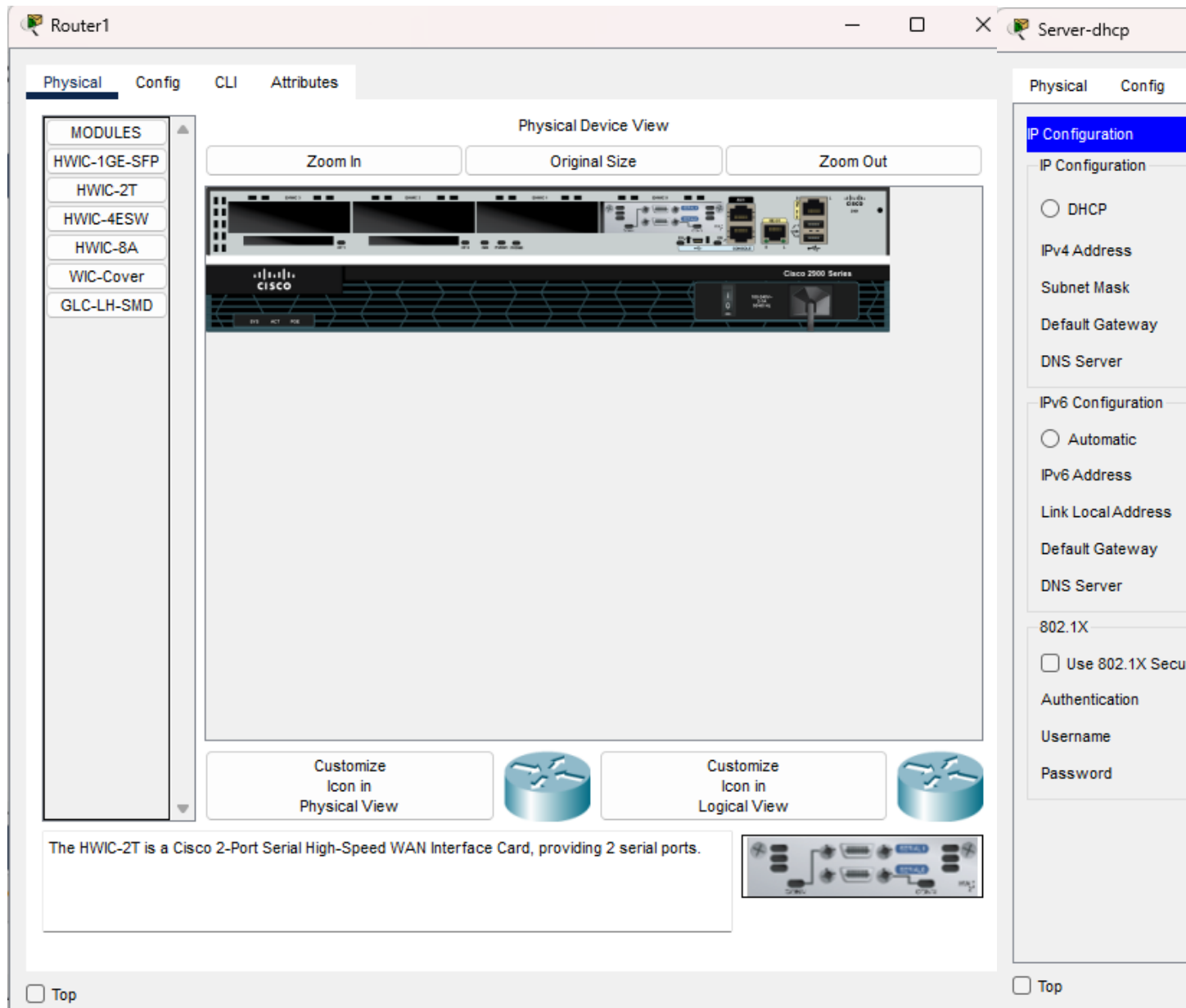
## B. La connectivité

### 1) Câble Ethernet (Copper Straight-Through)

Ces câbles sont utilisés pour connecter les serveurs aux switches, les access points aux switches et les switches aux routeurs. Je les ai utilisés car ils assurent une communication stable et rapide.

### 2) Câble Serial

Ils sont utilisés pour connecter les routeurs entre eux. Je les ai utilisées car ils permettent une connexion plus réaliste des réseaux étendus. Pour cela, j'ai dû ajouter des ports adaptés au câble serial sur chaque routeur.



### 3) Connection WI-FI

La connexion WI-FI est utilisé pour les smartphones et les dispositifs IoT. Tout d'abord, elle permet plus de mobilité aux appareils, surtout aux smartphones qui doivent gérer les équipements à distance. Mais également, les objets IoT sont souvent conçues pour fonctionner en WI-FI.

Les Access Point servent de relais pour assurer une communication fluide entre les appareils et l'infrastructure réseau.

## II. La configuration du réseau

The screenshot shows the 'Server-dhcp' configuration window with the 'Services' tab selected. On the left, a 'SERVICES' list includes HTTP, DHCP (selected), DHCPv6, TFTP, DNS, SYSLOG, AAA, NTP, EMAIL, FTP, IoT, VM Management, and Radius EAP. The main area is titled 'DHCP' and contains the following fields:

- Interface: FastEthernet0
- Service: ☒ On, ☐ Off
- Pool Name: serverPool
- Default Gateway: 192.168.1.1
- DNS Server: 0.0.0.0
- Start IP Address: 192, 168, 1, 14
- Subnet Mask: 255, 255, 255, 0
- Maximum Number of Users: 512
- TFTP Server: 0.0.0.0
- WLC Address: 0.0.0.0

Below these fields are 'Add', 'Save', and 'Remove' buttons. At the bottom, a table lists the configured pool:

Pool Name	Default Gateway	DNS Server	Start IP Address	Subnet Mask	Max User	TFTP Server	WLC Address
serverPool	0.0.0.0	0.0.0.0	192.168...	255.255...	512	0.0.0.0	0.0.0.0

A. Configuration des serveur DHCP

The screenshot shows the 'Server-dhcp1' configuration window with the 'Desktop' tab selected. It contains two main sections: 'IP Configuration' and 'IPv6 Configuration'.

**IP Configuration:**

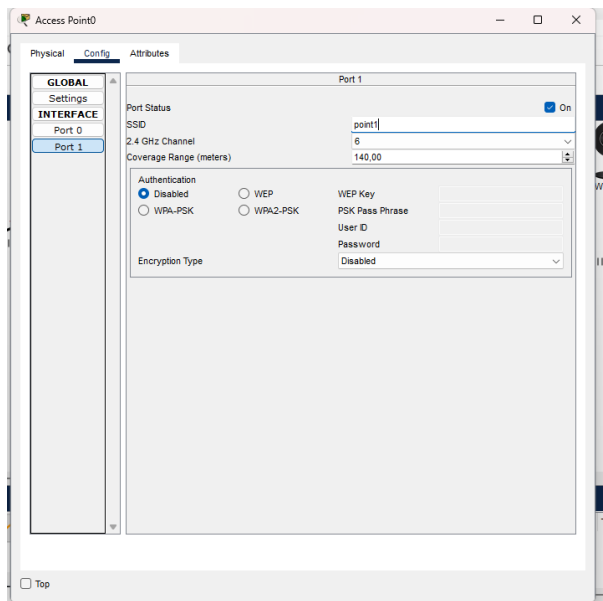
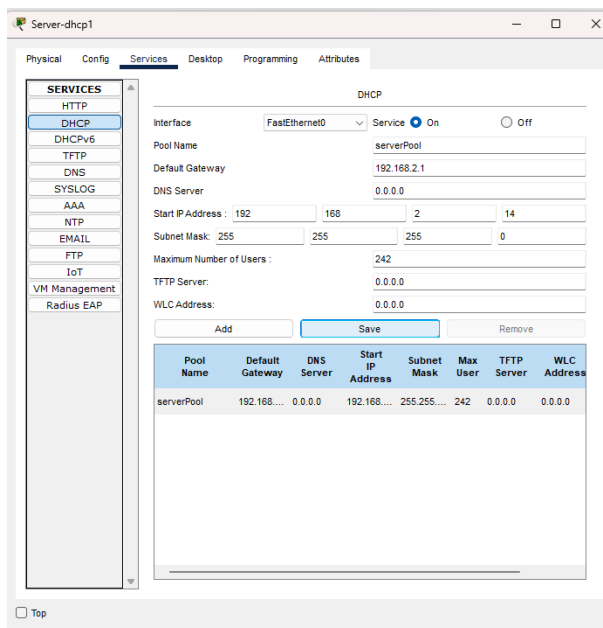
- ☐ DHCP, ☒ Static
- IPv4 Address: 192.168.2.13
- Subnet Mask: 255.255.255.0
- Default Gateway: 192.168.2.1
- DNS Server: 0.0.0.0

**IPv6 Configuration:**

- ☐ Automatic, ☒ Static
- IPv6 Address: [empty]
- Link Local Address: FE80::20C:85FF:FEE2:AB8
- Default Gateway: [empty]
- DNS Server: [empty]

**802.1X:**

- ☐ Use 802.1X Security
- Authentication: MD5
- Username: [empty]
- Password: [empty]



Pour configurer le protocole DHCP, on commence par attribuer une adresse IP statique à chacun des server-dhcp. Puis, on se rend dans l'onglet services, ensuite on choisit DHCP pour activer le protocole. Par la suite, on ajoute l'adresse de la passerelle et les spécifications concernant la plage d'adresse à utiliser et on appuie sur « save » pour enregistrer la configuration. On fait la même chose pour chacun des serveur DHCP.

## B. Configuration des Access Point et des objets IoT

Access Point1

Physical

Config

Attributes

GLOBAL

Settings

INTERFACE

Port 0

Port 1

Port Status

SSID

2.4 GHz Channel

Coverage Range (meters)

Disabled

WPA-PSK

WEP

WPA2-PSK

WEP Key

PSK Pass Phrase

User ID

Password

Encryption Type

Disabled

Port 1

On

point2

6

140.00

Top

IoT2

Specifications

Physical

Config

Attributes

GLOBAL

Settings

Algorithm Settings

Files

INTERFACE

Wireless0

Bluetooth

Port Status

Bandwidth

MAC Address

SSID

Disabled

WPA-PSK

WPA

802.1X

WEP

WPA2-PSK

WPA2

WEP Key

PSK Pass Phrase

User ID

Password

Method

MD5

Encryption Type

Disabled

IP Configuration

DHCP

Static

IPv4 Address

169.254.172.123

Subnet Mask

255.255.0.0

IPv6 Configuration

Automatic

Static

IPv6 Address

/

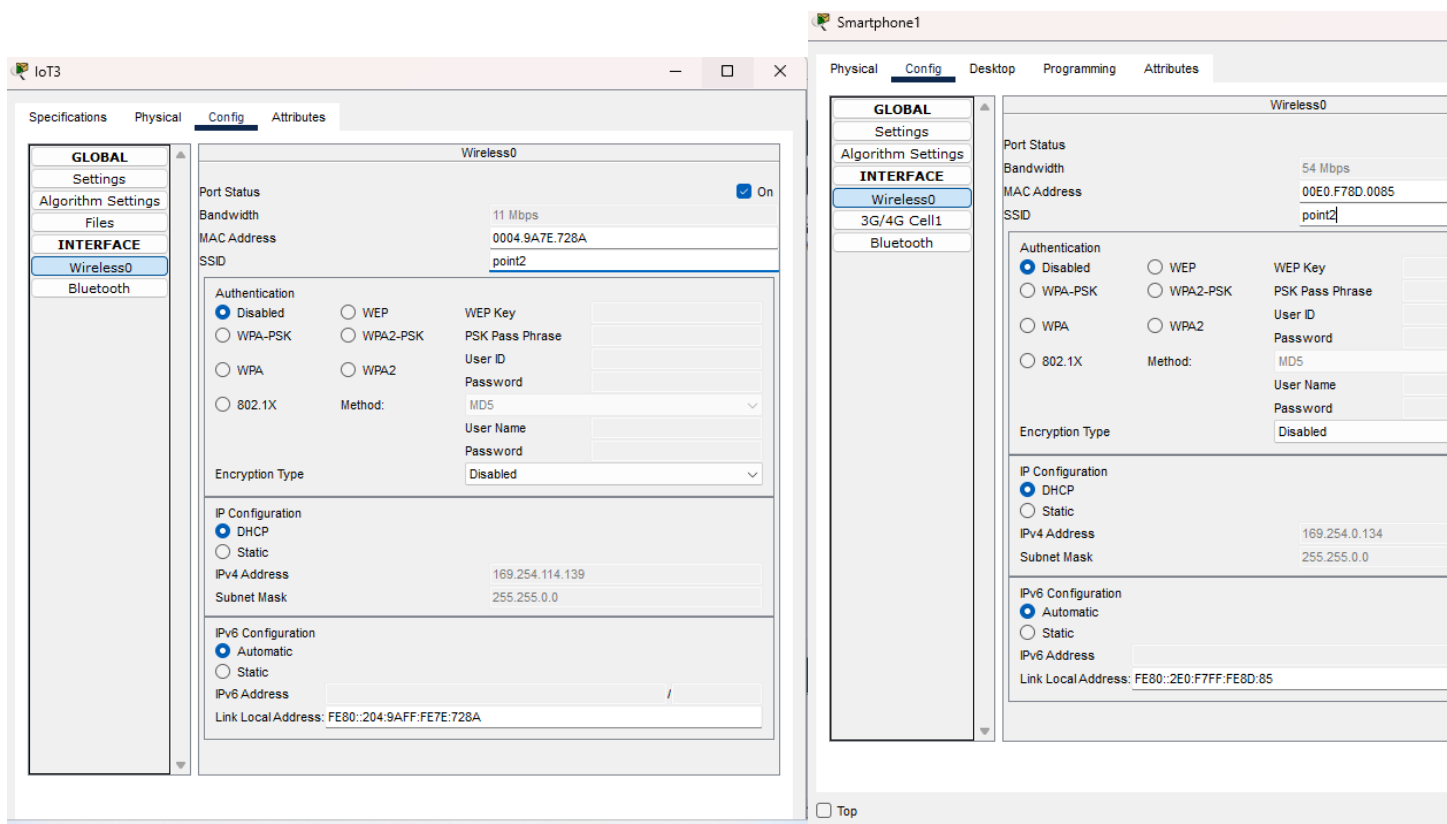
Link Local Address

FE80::210:11FF:FE87:AC7A

Top

Advanced

7



Dans cette partie, il est question de configurer les Access Point en se rendant dans l'onglet config, puis en cliquant sur port 1 afin de modifier le SSID. Par la suite, on se rend sur chaque objet IoT dans l'onglet config, puis on clique sur « Wireless 0 » et on ajoute le SSID de l'Access Point dans lequel on souhaite connecter notre objet. Pour les smartphones, après cette configuration, on se rend dans l'onglet Desktop, puis sur IP configuration pour constater l'attribution automatique de l'adresse IP par le protocole DHCP.

### C. Configuration des routeurs



Router1

Physical Config **CLI** Attributes

IOS Command Line Interface

```
Press RETURN to get started!

Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname R2
R2(config)#interface GigabitEthernet0/0
R2(config-if)#ip address 192.168.1.1 255.255.255.0
R2(config-if)#no shutdown

R2(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

R2(config-if)#exit
R2(config)#interface serial0/0/0
R2(config-if)#ip address 10.10.10.4
% Incomplete command.
R2(config-if)#ip address 10.10.10.4 255.255.255.0
R2(config-if)#no shutdown

R2(config-if)#
%LINK-5-CHANGED: Interface Serial0/0/0, changed state to up

R2(config-if)#end
R2#
%SYS-5-CONFIG_I: Configured from console by console

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up

R2#wr
Building configuration...
[OK]
R2#
```

☐ Top

Router0

Physical Config **CLI** Attributes

IOS Command Line Interface

```
R1(config-if)#no shutdown

R1(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
exit
R1(config)#interface serial0/0/0
R1(config-if)#ip address 10.10.10.3 255.255.255.0
R1(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/0/0, changed state to down
R1(config-if)#end
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#wr
Building configuration...
[OK]
R1#
%LINK-5-CHANGED: Interface Serial0/0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up

R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#router ospf 1
R1(config-router)#network 192.168.0.0 0.0.255.255 area 0
R1(config-router)#network 10.10.10.0 0.0.0.255 area 0
R1(config-router)#end
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#wr
Building configuration...
[OK]
R1#
```

☐ Top

```
Router1
Physical Config CLI Attributes
IOS Command Line Interface

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
R2(config-if)#exit
R2(config)#interface serial0/0/0
R2(config-if)#ip address 10.10.10.4
% Incomplete command.
R2(config-if)#ip address 10.10.10.4 255.255.255.0
R2(config-if)#no shutdown

R2(config-if)#
%LINK-5-CHANGED: Interface Serial0/0/0, changed state to up
R2(config-if)#end
R2#
%SYS-5-CONFIG_I: Configured from console by console

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up
R2#wr
Building configuration...
[OK]
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#router ospf 1
R2(config-router)#network 192.168.0.0 0.0.255.255 area 0
R2(config-router)#network 10.10.10.0 0.0.0.255 area 0
R2(config-router)#
00:31:01: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.2.1 on Serial0/0/0 from LOADING to FULL, Loading Done
end
R2#
%SYS-5-CONFIG_I: Configured from console by console

R2#wr
Building configuration...
[OK]
R2#
```

```
Router1
Physical Config CLI Attributes
IOS Command Line Interface

Building configuration...
[OK]
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#router ospf 1
R2(config-router)#network 192.168.0.0 0.0.255.255 area 0
R2(config-router)#network 10.10.10.0 0.0.0.255 area 0
R2(config-router)#
00:31:01: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.2.1 on Serial0/0/0 from LOADING to FULL, Loading Done
end
R2#
%SYS-5-CONFIG_I: Configured from console by console

R2#wr
Building configuration...
[OK]
R2#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

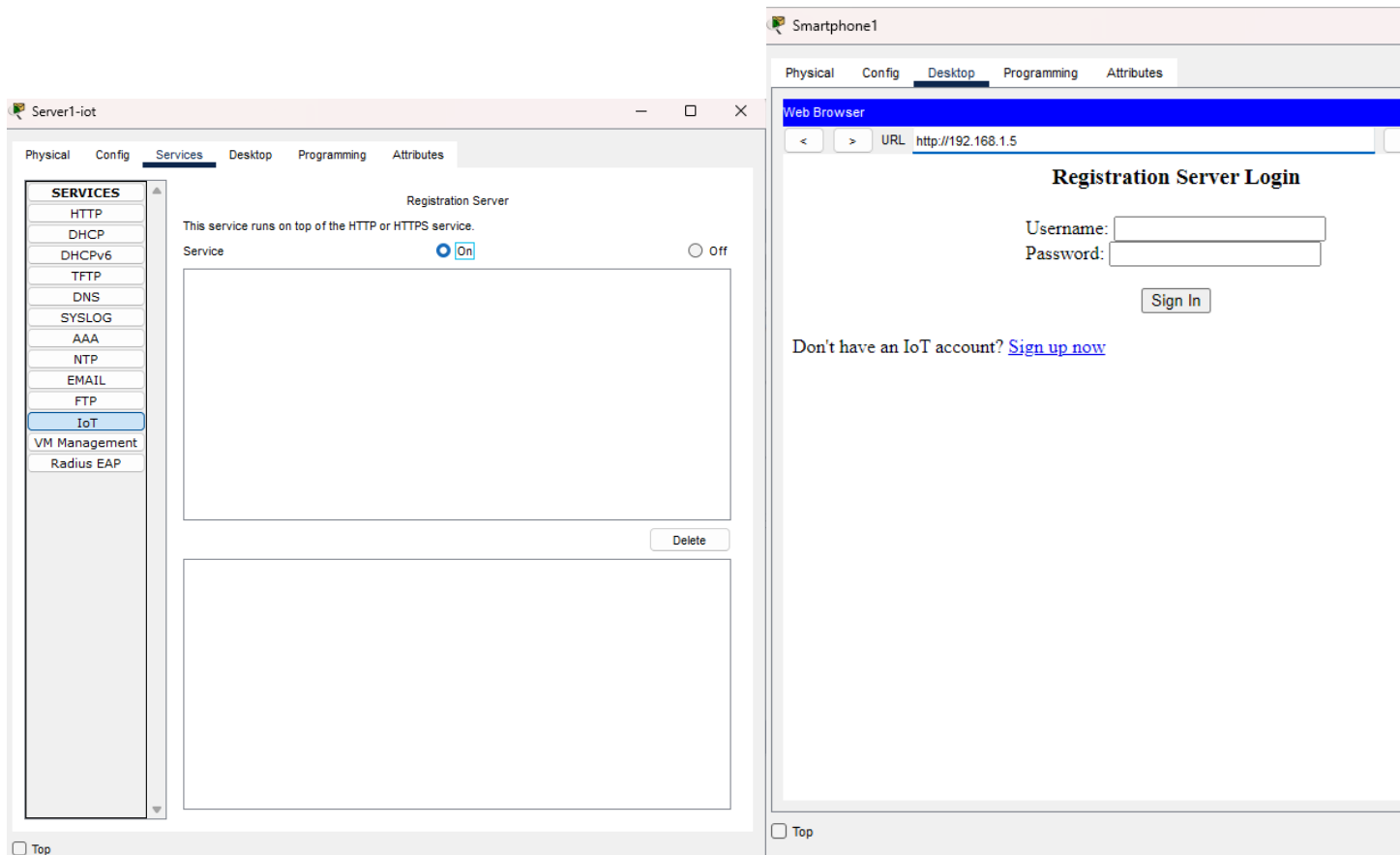
Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C 10.10.10.0/24 is directly connected, Serial0/0/0
L 10.10.10.4/32 is directly connected, Serial0/0/0
192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C 192.168.1.0/24 is directly connected, GigabitEthernet0/0
L 192.168.1.1/32 is directly connected, GigabitEthernet0/0
O 192.168.2.0/24 [110/65] via 10.10.10.3, 00:00:32, Serial0/0/0

R2#
```

Dans cette partie, j'ai procédé à la configuration de base des routeurs en configurant les interfaces GigabitEthernet0/0 et serial0/0/0 sur chacun des routeurs. Puis par la suite, j'ai activé un routage dynamique à l'aide de la commande « router ospf 1 » sur les deux routeurs. Avec la commande « show ip route » je me suis assuré que le routage fonctionne sur chacun des routeurs. Et pour finir, sur le smarphone0 qui se situe dans la chambre de l'agent de sécurité, j'ai fait « ping 192.168.1.1 », qui est l'adresse de passerelle pour la chambre du client, pour voir si les deux chambres communiquent entre elles et ce fut un succès.

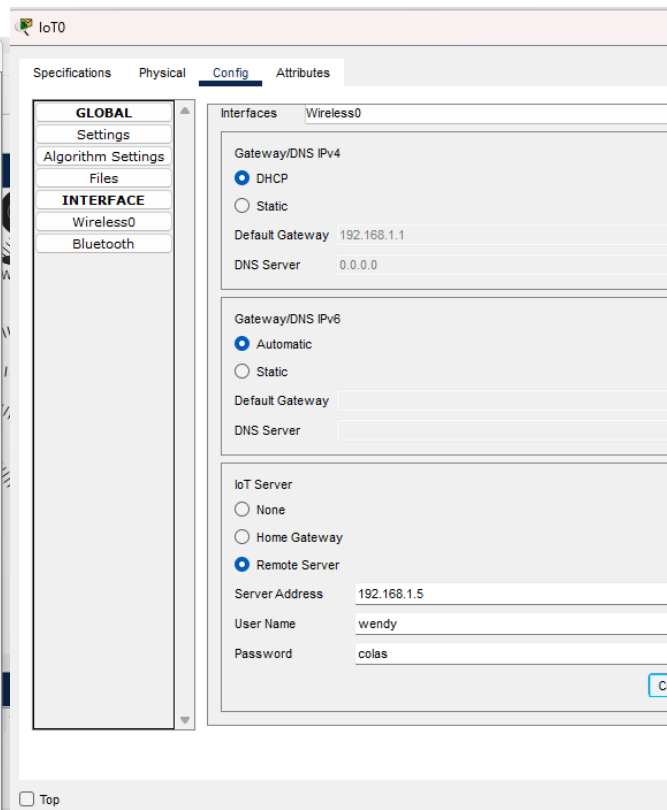
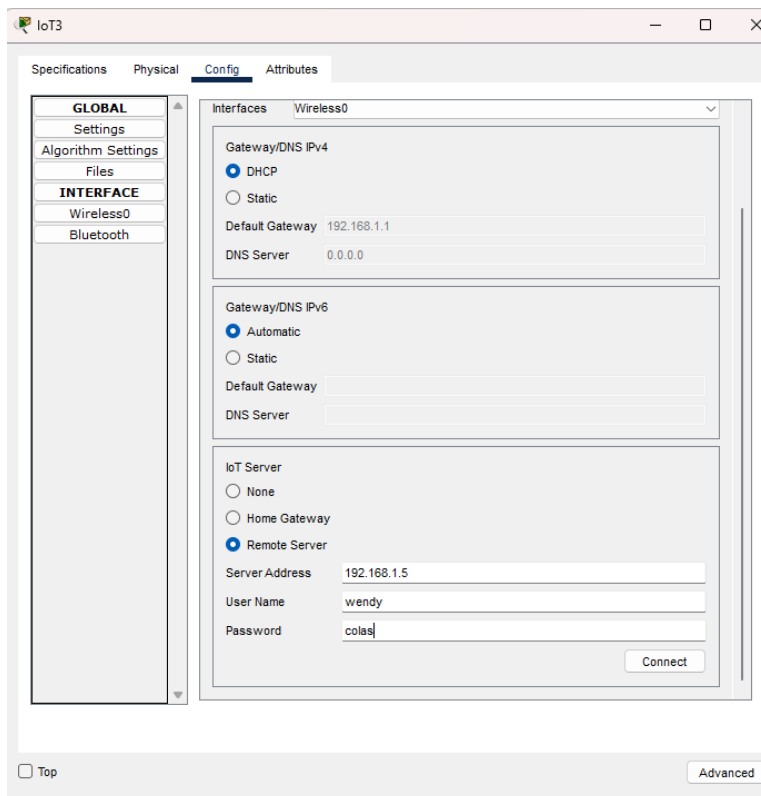
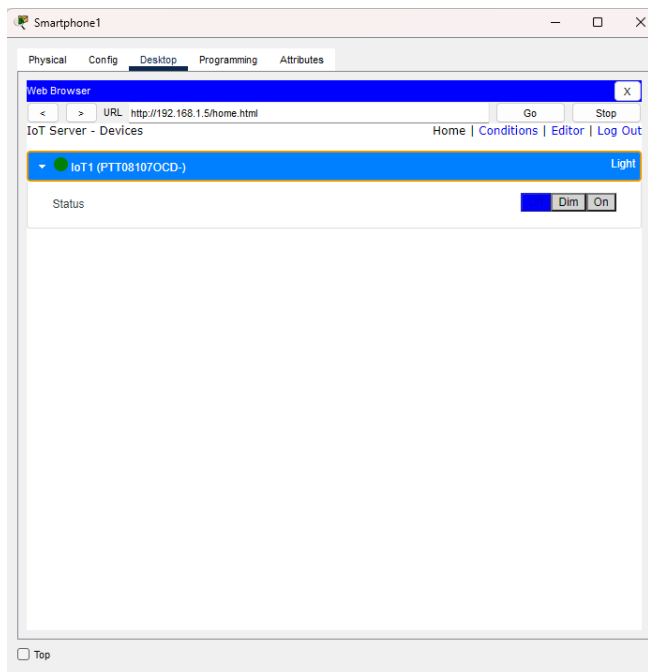
## D. Configuration du protocole IoT

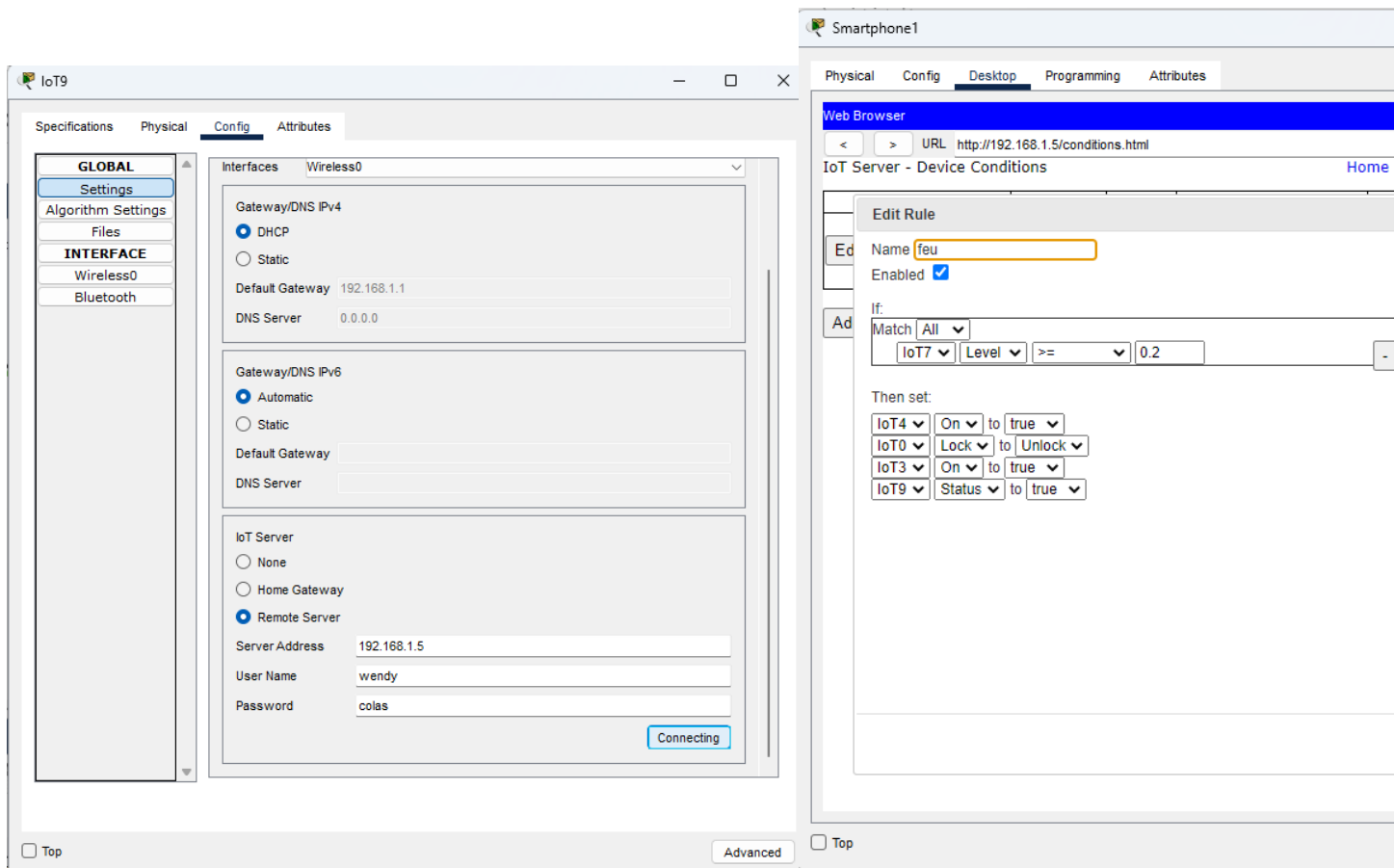


Pour configurer le protocole IoT, il faut tout d'abord attribuer une adresse IP statique au serveur IoT, ensuite on se rend sur l'onglet services, puis on clique sur IoT pour l'activer. Après cela, on se rend sur chacun des smartphones dans l'onglet desktop, puis dans Web browser et on met l'adresse du serveur IoT soit 192.168.1.5 dans notre cas. Par la suite, on crée des comptes pour l'agent de sécurité (Username : colas, password : wendy) et pour le client (Username : wendy, password : colas).

En nous connectant avec nos comptes respectifs, on se rend compte qu'il n'y a aucun objet connecté pour l'instant.

## E. Connexion des différents objets IoT





Pour cette partie, j'ai commencé par les objets qui seront connectés au smartphone du client comme : la fenêtre, la lampe, la porte, la sirène 1, le détecteur de fumée et l'extincteur de feu et ceux qui seront connectés au smartphone de l'agent de sécurité comme : la camera, le détecteur de mouvement et la sirène 2.

Après avoir fait cette séparation, on procède à la connexion de chaque objet. Pour les objets qui seront connectés au smartphone de l'agent de sécurité (smartphone0), on se rend dans l'onglet « config » de l'objet en question, puis on clique sur « settings » et dans la section IoT server on coche « remote server » et on remplit les lignes avec 192.168.1.5 qui est l'adresse du server-iot, puis colas et wendy qui sont respectivement le username et le password du compte sur le server-iot pour l'agent de sécurité. On procède de la même manière pour le client tout en utilisant ses identifiants à lui.

Ensuite on se connecte sur chaque smartphone pour voir si les objets sont bel et bien connectés et s'ils fonctionnent également.

Par la suite, j'ai ajouté plusieurs conditions qui donnent une certaine automatisation aux différents objets IoT. La première condition, ajoutée sur le smartphone de l'agent de sécurité, est une condition de détection : if détecteur de mouvement on is true then set camera on to true and sirène 2 on to true. Avec cette condition, si le détecteur de mouvement détecte une présence, automatiquement la camera s'allumera et la sirène 2 se trouvant dans la chambre de l'agent de sécurité sera allumée aussi.

Il y a une condition de non détection également : if détecteur de mouvement on is false then set camera on to false and sirène 2 on to false.

Des conditions sont aussi ajoutées sur le smartphone se trouvant dans la chambre du client dont le premier s'appelle feu : if détecteur de fumée level  $\geq 0,2$  then set sirène 1 on to true, porte on to true, fenêtre on to true, extincteur de feu status to true. Cette condition stipule que si le détecteur détecte une fumée supérieure ou égale à 0,2 une sirène dans la chambre de l'agent de sécurité se mettra à sonner, la porte et la fenêtre s'ouvriront et l'extincteur de feu s'ouvrira aussi.

Une autre du nom de non feu est aussi posée qui est l'inverse de celle intitulée feu.

Pour vérifier si les conditions fonctionnent, on utilise la touche ALT et le curseur.

### III. Comparaison entre les protocoles IoT

Dans le domaine de l'internet des objets (IoT), le choix du protocole de communication est crucial pour garantir une transmission efficace des données entre les appareils connectés. Trois protocoles majeurs sont largement utilisés : MQTT, CoAP et http. Chacun d'eux présente des caractéristiques uniques, en termes de performance et de complexité, et leur fonctionnement varie selon les besoins du système IoT.

#### • MQTT (Message Queuing Telemetry Transport)

Le MQTT est un protocole de messagerie léger conçu pour les réseaux avec peu de bande passante et les appareils à faible puissance. Il est souvent utilisé dans les systèmes IoT pour transmettre efficacement des données entre appareils.

#### Performance :

- Utilise un modèle de publication-abonnement.
- Les appareils ne communiquent pas directement entre eux car ils envoient et reçoivent des messages via un courtier (serveur central).
- Très rapide et efficace.

#### Complexité :

- Moyenne
- Nécessite un courtier pour gérer les échanges de messages.
- Facile à intégrer mais demande une configuration précise.

#### • CoAP (Constrained Application Protocol)

Le CoAP est un protocole spécialement conçu pour les appareils IoT très limités en énergie et en ressources. Il permet une communication rapide et légère en utilisant un modèle proche de http.

#### Performance :

- Fonctionne avec un modèle client-serveur.

- Utilise UDP au lieu de TCP, ce qui le rend plus rapide et adapté aux objets connectés.
- Permet aux petits appareils de faire des requêtes REST (GET, POST, PUT, DELETE) sur les serveurs.

#### **Complexité :**

- Moyenne.
- Basé sur UDP.
- Peut être plus difficile à gérer lorsqu'on doit assurer la fiabilité des messages.
- **HTTP (Hypertext Transfert Protocol)**

HTTP est le protocole de communication le plus utilisé sur internet. Il est conçu pour échanger des données entre les navigateurs et les serveurs Web.

#### **Performance :**

- Fonctionne avec un modèle serveur-client.
- Utilise TCP pour assurer une transmission fiable des données.

#### **Complexité :**

- Elevée.
- Trop lourd pour les objets connectés avec peu de ressources.

### **IV. Importance de l'IoT**

L'Internet des Objets joue un rôle fondamental dans la collecte de données et l'automatisation des processus grâce à la connexion intelligente entre appareils. Il permet :

#### **1. La collecte efficace des données**

- Les capteurs IoT peuvent surveiller en temps réel divers paramètres (température, humidité, mouvement, ...).
- La transmission automatique des données permet d'éviter les erreurs humaines et améliorer la précision des analyses.
- Ces informations permettent d'anticiper les problèmes, comme la maintenance préventive des machines avant qu'elles ne tombent en panne.

#### **2. L'automatisation des processus**

L'IoT permet aux systèmes connectés d'agir sans intervention humaine. Par exemple :

- Les usines intelligentes ajustent leur production en fonction des besoins détectés.
- Les bâtiments intelligents optimisent l'éclairage et le chauffage pour économiser de l'énergie.

#### **3. L'amélioration de l'efficacité et la réduction des coûts**

- Moins d'intervention humaine = moins d'erreurs et plus de rapidité dans l'exécution des tâches.
- Optimisation de l'utilisation des ressources.
- Amélioration de la sécurité.

## V. Création du projet sur Jira

Dans le cadre du projet intitulé Simulation d'un Réseau IoT avec Cisco Packet Tracer , une démarche de gestion agile a été mise en œuvre à travers l'utilisation de la méthode Scrum . L'objectif de ce travail était de simuler un environnement de développement itératif en structurant les différentes phases du projet sous forme de user stories , regroupées dans des Epics , et planifiées dans un Sprint de deux semaines à l'aide de l'outil Jira. Ce rapport présente les étapes de mise en œuvre de cette démarche, de la création du backlog à la simulation du sprint, en passant par l'organisation des tâches, la planification et l'évaluation.

### 1. Création du projet Jira et structuration en Epics

La première étape a consisté à créer un nouveau projet Scrum dans Jira, en lien avec le thème principal du projet IoT. Le travail a été découpé en 5 grands Epics , chacun représentant une composante majeure du système :

- Infrastructure réseau
- Intégration des objets IoT
- Automatisation des objets
- Expérience utilisateur via smartphone
- Choix et analyse des protocoles IoT

Ces Epics ont permis de mieux organiser les tâches selon les différentes dimensions du projet technique et fonctionnel.

Un backlog a été élaboré à partir des besoins fonctionnels du système. Dix user stories ont été formulées en suivant le format classique ("En tant que... je veux... afin de..."). Chaque user story a été associée à un Epic, facilitant leur hiérarchisation et planification. Cette étape a permis d'identifier les fonctionnalités clés à développer : configuration réseau, connexion des capteurs, automatisation des alertes, interface utilisateur, etc.

Un premier Sprint de deux semaines a été organisé. Son objectif principal était de mettre en place l'infrastructure de base, connecter les premiers objets IoT, configurer les smartphones, et amorcer les règles d'automatisation.

Des tâches concrètes ont été créées pour chaque user story sélectionnée pour ce sprint. Un effort particulier a été mis sur les tâches prioritaires liées à la connectivité réseau, à la configuration du DHCP et à la communication inter-chambre.

Dans Jira, un tableau Scrum a été configuré avec les colonnes "À faire", "En cours", "En revue" et "Terminé". Les tâches ont ensuite été déplacées au fil de la progression, repro-



duisant ainsi le workflow agile . Cette représentation visuelle a permis de suivre en temps réel l'évolution du sprint et d'identifier rapidement les éventuels blocages.

À la fin du sprint, une démonstration des réalisations a été simulée : les objets IoT ont bien été connectés, la configuration des serveurs DHCP validée, et les premières automatisations testées avec succès.

La rétrospective a permis d'identifier les points forts (bonne structuration initiale, cohérence des Epics) et les axes d'amélioration (gestion des dépendances entre tâches, planification plus réaliste de certaines fonctionnalités avancées).

The image is a composite of three screenshots from a Jira project. The top left screenshot shows a 'Smartphone1' window with a 'Web Browser' tab displaying 'IoT Server - Device Conditions' and an 'Edit Rule' dialog box. The top right screenshot shows a Jira project page for 'Simulation d'un Réseau IoT avec Cisco Packet Tracer' with fields for 'Nom', 'Clé', and 'Autorisations de projet'. The bottom screenshot shows the Jira backlog for the same project, with a 'Créer' button and a 'Quickstart' button. The bottom right screenshot shows a Jira sidebar with navigation options like 'Projets', 'Tableaux de bord', and 'Équipes'.

colaswendy10-1744211164491.atlassian.net/jira/software/projects/SDRIACPT/boards/34/backlog?epics=visible

Jira

Rechercher

+ Créer Passer à l'offre supérieure

Projets

Simulation d'un Réseau IoT avec Cisco Packet Tracer

Résumé Chronologie Backlog Tableau Calendrier

Rechercher... Version Epic

Epic

Aucun(e) epic

Mise en place de l'infrastructure réseau

Intégration des objets IoT

+ Créer epic

Automatisation du compor...

Les champs obligatoires sont marqués d'un astérisque \*

Projet \*

Simulation d'un Réseau IoT avec Cisco Packet Tr

Type de ticket \*

Epic

État

TO DO

Résumé \*

Automatisation du comportement des objets

Description

Tt ... A ... : ... +

Saisissez @ pour mentionner un collègue et lui

Créer une autre

Créer

Quickstart

29°C Ensoleillé

Rechercher

FRA US

09:38 28/05/2025

colaswendy10-1744211164491.atlassian.net/jira/software/projects/SDRIACPT/boards/34/backlog?epics=visible

Jira

Rechercher

+ Créer Passer à l'offre supérieure

Projets

Simulation d'un Réseau IoT avec Cisco Packet Tracer

Résumé Chronologie Backlog Tableau Calendrier Liste Formulaires Objectifs More 6 +

Rechercher... Version Epic Type

Epic

Aucun(e) epic

Mise en place de l'infrastructure réseau

Intégration des objets IoT

Automatisation du comportement des objets

SDRIACPT-6 En tant qu'administrateur réseau, je veux config...

À FAIRE

SDRIACPT-7 En tant qu'administrateur réseau, je veux configurer ...

À FAIRE

SDRIACPT-8 En tant qu'administrateur réseau, je veux vérifier la ...

À FAIRE

SDRIACPT-9 En tant que client, je veux connecter mes objets IoT ...

À FAIRE

SDRIACPT-10 En tant qu'agent de sécurité, je veux recevoir une ...

À FAIRE

SDRIACPT-11 En tant qu'utilisateur, je veux que des règles d'auto...

À FAIRE

SDRIACPT-12 En tant qu'utilisateur, je veux que la porte, la fenêtr...

À FAIRE

SDRIACPT-13 En tant que client, je veux gérer les objets de ma c...

À FAIRE

SDRIACPT-14 En tant qu'agent de sécurité, je veux contrôler à di...

À FAIRE

SDRIACPT-15 En tant qu'ingénieur système, je veux comparer M...

À FAIRE

29°C Ensoleillé

Rechercher

FRA US

09:53 28/05/2025

29°C Ensoleillé

colaswendy10-1744211164491.atlassian.net/jira/software/projects/SDRIACPT/boards/34/backlog?epics=visible

### Démarrer le sprint

10 tickets ne sont pas inclus à ce sprint.

Les champs obligatoires sont marqués d'un astérisque \*

Nom du sprint \*

Tableau Sprint 1

Durée \*

2 semaines

Date de début \*

28/05/2025 10:08

Date de fin \*

11/06/2025 10:08

Objectif du sprint

Annuler Démarrer

Passer à l'offre supérieure

Pour vous

Récents

Favoris

Apps

Plans

Projets

Récents

Simulation d'un Résea...

Calculatrice

Afficher tous les projets

Filtres

Tableaux de bord

Équipes

Objectifs

Personnaliser la barre la...

Envoyer du feedback s...

29°C Ensoleillé

Rechercher

FRA US

10:08 28/05/2025

colaswendy10-1744211164491.atlassian.net/jira/software/projects/SDRIACPT/boards/34?sprintStarted=true

### Simulation d'un Réseau IoT avec Cisco Packet Tracer

Résumé Chronologie Backlog **Tableau** Calendrier Liste Formulaires Objectifs More 6 +

Rechercher Epic Type Terminer le sprint Regrouper

A FAIRE 6	EN COURS 2	TERMINE 2
<p>En tant qu'agent de sécurité , je veux recevoir une alerte lorsqu'un mouvement est détecté, pour pouvoir réagir rapidement.</p> <p>INTÉGRATION DES OBJETS IOT</p> <p>SDRIACPT-10</p>	<p>En tant qu'administrateur réseau , je veux configurer le protocole DHCP sur les serveurs, pour automatiser l'attribution des adresses IP.</p> <p>MISE EN PLACE DE L'INFRASTRUCTUR...</p> <p>SDRIACPT-7</p>	<p>En tant qu'administrateur réseau , je veux vérifier la connectivité entre les deux chambres, afin de valider le bon routage du trafic.</p> <p>MISE EN PLACE DE L'INFRASTRUCTUR...</p> <p>SDRIACPT-8</p>
<p>En tant qu'utilisateur , je veux que des règles d'automatisation déclenchent la caméra et la sirène en cas de détection de mouvement.</p> <p>AUTOMATISATION DU COMPORTEME...</p> <p>SDRIACPT-11</p>	<p>En tant que client , je veux connecter mes objets IoT au serveur IoT, afin de pouvoir les contrôler depuis mon smartphone.</p> <p>INTÉGRATION DES OBJETS IOT</p> <p>SDRIACPT-9</p>	<p>En tant qu'administrateur réseau , je veux configurer les routeurs et les switches, afin d'assurer une interconnexion efficace des appareils.</p> <p>MISE EN PLACE DE L'INFRASTRUCTUR...</p> <p>SDRIACPT-6</p>

Passer à l'offre supérieure

Pour vous

Récents

Favoris

Apps

Plans

Projets

Récents

Simulation d'un Résea...

Calculatrice

Afficher tous les projets

Filtres

Tableaux de bord

Équipes

Objectifs

Personnaliser la barre la...

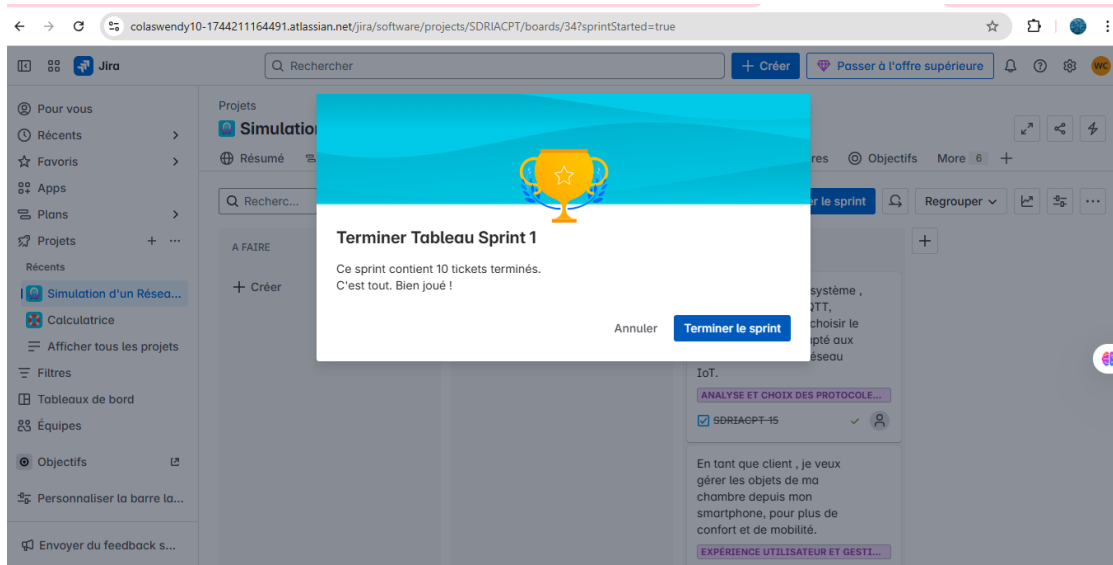
Envoyer du feedback s...

29°C Ensoleillé

Rechercher

FRA US

10:10 28/05/2025



## VI. Conclusion

Ce document explore la conception et la configuration d'un réseau IoT dans Cisco Packet Tracer, en mettant en place une infrastructure connectée pour l'automatisation et la sécurité. Il détaille les différents appareils IoT, leur connectivité, et les protocoles réseau utilisés, notamment DHCP, Wi-Fi et OSPF.

Grâce à cette simulation, nous avons pu observer l'échange de données en temps réel, la gestion des équipements IoT, et l'intégration de règles automatisées pour optimiser le fonctionnement du réseau. En comparant les protocoles de communication comme MQTT, CoAP et HTTP, le document montre comment choisir le protocole le plus adapté selon les besoins du système.

Cette étude met en avant l'importance de l'IoT dans la collecte de données et l'automatisation des processus, soulignant ses bénéfices en matière de sécurité, efficacité et réduction des coûts. Elle démontre également que, grâce à des outils comme Cisco Packet Tracer, il est possible de concevoir des réseaux intelligents et performants, ouvrant la voie à des applications pratiques dans divers domaines.

Ce travail a démontré l'intérêt d'appliquer la méthode Scrum à un projet technique tel que la simulation d'un réseau IoT. L'utilisation de Jira facilite l'organisation, la planification et le suivi des tâches. Cette approche itérative a permis de mieux visualiser la progression du projet et de s'adapter aux besoins au fur et à mesure. Au-delà de l'aspect technique, ce travail a renforcé la maîtrise des outils de gestion de projet agile, essentiels dans le monde professionnel actuel.