Chapitre 2. Réseaux personnels sans-fils (WPAN)

c) Le contrôleur de liaisons (Link Controller)

- Cette couche gère la configuration et le contrôle de la liaison physique entre deux appareils ;
- Le rôle du contrôleur de lien est de commander la construction de paquets à la couche inférieure (baseband), un à un, afin d'établir et de maintenir une transmission fiable.

d) Le gestionnaire de liaisons (Link Manager)

- Ce gestionnaire exécute l'établissement de la liaison, l'authentification, la configuration de la liaison et d'autres protocoles ;
- Il gère les liens entre les périphériques maîtres et esclaves (dans les réseaux *Bluetooth*) ;
- Il découvre d'autres LM distants et communique avec eux via la couche LMP (Link Manager Protocol) ;
- Le LM utilise les services du contrôleur de liaison (LC) sous-jacent pour remplir son rôle ;
- Il gère aussi les types de liaisons (synchrones ou asynchrones) ;
- C'est le gestionnaire de liaisons qui implémente les mécanismes de sécurité tels que :
  - L'authentification ;
  - Le pairage ;
  - La création et la modification des clés ;
  - Le cryptage.

e) L'interface de contrôle de l'hôte (HCI)

- La couche HCI fait le lien entre les couches physiques (module) et les couches applicatives (hôte).
- Cette couche fournit une méthode uniforme pour accéder aux couches matérielles ;
- Son rôle de séparation permet un développement indépendant du hardware et du software ;
- Les protocoles de transport suivants sont supportés :
  - USB (Universal Serial Bus) ;
  - PC Card ;
  - RS-232 ;
  - UART.

f) La couche L2CAP

- La couche L2CAP (Logical Link Control & Adaptation Protocol) permet d'utiliser simultanément différents protocoles de niveaux supérieurs ;
- Un mécanisme permet d'identifier le protocole de chaque paquet envoyé pour permettre à l'appareil distant de passer le paquet au bon protocole, une fois celui-ci récupéré ;
- Elle gère également la segmentation (et le réassemblage) des paquets de protocoles de niveaux supérieurs en paquets de liaison de 64 Ko.

e) Les protocoles

- Le protocole SDP (Service Discovery Protocol) permet à un appareil *Bluetooth* de rechercher d'autres appareils et d'identifier les services disponibles ;
- Le protocole OBEX (OBject EXchange) permet le transfert des données grâce au protocole d'échange de fichiers IrDA.
- Le protocole TCS (Telephony Control protocol Specification) est un protocole de signalisation téléphonique qui permet d'échanger des commandes (envoi d'un appel, décroché, raccroché, etc.) ;
- Le protocole RFCOMM, basé sur les spécifications RS-232, permet l'émulation des liaisons séries. Il peut notamment servir à faire passer une connexion IP par *Bluetooth* ;

Les profils Bluetooth

- Chaque puce *Bluetooth* s'accompagne d'une pile de protocoles. Il s'agit d'un paquet logiciel comprenant les services permettant d'utiliser les différents profils *Bluetooth* ;
- Comme pour les pilotes d'un ordinateur, ces profils définissent quels types de données peuvent être transmis entre les appareils et quels services sont ainsi disponibles ;
- Les profils qu'un appareil peut maîtriser peuvent généralement être lus à partir de ses données techniques. Pour permettre l'utilisation de certaines fonctionnalités, tous les appareils participants doivent supporter les mêmes profils ;
- Le tableau suivant contient certains des profils standard les plus utilisés. Ce tableau n'est pas exhaustif, du fait que de nouveaux profils sont constamment ajoutés afin de réagir de façon flexible aux nouvelles exigences des appareils.

Tableau.1. Profils Bluetooth

Sigle du profil	Nom de profil	Fonction	Appareils (exemples)
A2DP	Advanced Audio Distribution Profile	Transmission de données audio en qualité stéréo	Kits mains libres, écouteurs, lecteur MP3
AVRCP	Audio/Video Remote Control Profile	Commande à distance de lecteurs audio et vidéo	Télévision, chaîne hi-fi, notebooks
BIP	Basic Imaging Profile	Transmission de fichiers images	Appareils photo numériques, imprimantes, smartphones
BPP	Basic Printing Profile	Connexion aux imprimantes	Imprimantes, notebooks, smartphones

CTP	Cordless Telephony Profile	Connexion à des téléphones sans fil	Ordinateurs, notebooks, téléphones sans fil
FAX	Fax Profile	Connexion à des fax	Ordinateurs, fax, notebooks, smartphones
GATT	Generic Attribute Profile	Transmission basse consommation de petites quantités de données avec *Bluetooth* 4.0 Low Energy	Ordinateur, notebooks, smartphones
HDP	Health Device Profile	Connexion sécurisée à des appareils médicaux	Télécommandes, appareils médicaux
HFP	Hands-Free Profile	Connexion à des kits mains libres	Kits mains libres, smartphones
HID	Human Interface Device Profile	Connexion à des appareils de saisie	Ordinateurs, souris, claviers
HSP	Headset Profile	Connexion à des écouteurs	Ordinateurs, kits mains libres, smartphones
ICP	Intercom Profile	Communication vocale directe	Ordinateurs, téléphones sans fil, smartphones
OBEX	Object Exchange Profile	Échange de données générique entre deux appareils	Ordinateur, notebooks, smartphones
PBA	Phonebook Access Profile	Mise à disposition de données de répertoire	Kits mains libres, smartphones
(r)SAP	(remote) SIM Access Profile	Mise à disposition des données d'une carte SIM de téléphone	Kits mains libres, smartphones
VDP	Video Distribution Profile	Transmission de signaux vidéo	Caméras vidéo, ordinateurs, lecteurs vidéo portables

Adressage Bluetooth

- **DBA (ou DB_ADDR)**:
  L'adresse du périphérique *Bluetooth* (ou BD_ADDR) est un identifiant unique de 48 bits attribué à chaque périphérique *Bluetooth* par le fabricant. L'adresse *Bluetooth* est généralement affichée sous forme de 6 octets écrits en hexadécimal et séparés par des « deux-points » (exemple

00:11:22:33:FF:EE). C'est l'équivalent de l'adresse MAC d'une carte réseaux.

- **AMA (ou AM_ADDR)** : « Active Member Address » est l'adresse d'un esclave dans le piconet. Elle est codée sur 3 bits dont l'adresse « 000 » est réservée pour le broadcast. Il peut donc avoir 7 esclaves au maximum dans un piconet ;
- **PMA (ou PM_ADDR)** : « Parked Member Address » est l'adresse d'un esclave lorsqu'il se trouve à l'état parqué. Elle est codée sur 8 bits, donc il ne peut y avoir que 255 esclaves parqués au maximum ;
- **ARA (ou AR_ADDR)** : « Access Request Address » est l'adresse de demande d'accès utilisé par les esclaves parqués. Cette adresse n'est pas nécessairement unique.

Diagramme d'état du contrôleur de liaison

Les différents états d'un terminal *Bluetooth* sont montrés dans la figure 22. Les modules Bluetooth entrent en liaison de la façon suivante :

- Ils sont au départ en mode « Standby » (attente) et cherchent toutes les 1,28 s la présence de transmissions à proximité ;
- Le module Bluetooth qui souhaite communiquer envoie des requêtes en mode Inquiry si les adresses sont inconnues et en mode Page si elles sont connues. ➔ Il envoie sur un canal radio une rafale de paquets ID de courte durée (68 µs) pour inviter les stations à l'écoute à envoyer leurs informations personnelles ;
- Il devient alors le maître du piconet, et son adresse définit la suite des sauts en fréquence suivie par les esclaves ;
- Dans un état de marche normal, l'esclave doit être dans l'état Inquiry Scan ;
- Le maître émet une signalisation pour initialiser la communication. Dans ce cas, si l'esclave reçoit les messages, il passe dans un état Inquiry Response. Cet état va lui permettre d'envoyer un message au maître lui précisant son adresse et l'état de son horloge ;
- Il passe ensuite dans un nouvel état, Page scan, dans lequel il attend de recevoir un paquet contenant son adresse sur l'une des fréquences disponibles ;
- A réception du message, le maître passe dans l'état Page, dans lequel il met à jour ses tables de connexion puis envoie un message vers l'esclave ;
- Lorsque l'esclave détecte ce message, il se place dans l'état Slave Response puis répond au maître en indiquant son code d'accès ;
- Le maître se met alors dans l'état Master Response et envoie un paquet FHS, qui permet à l'esclave de se synchroniser sur l'horloge du maître, puis passe dans l'état connecté (Connected) ;

- De même, lorsque l'esclave reçoit ce message, il passe dans l'état connecté ; Le maître n'a plus alors qu'à effectuer une interrogation (polling) vers l'esclave pour vérifier qu'il y a bien eu connexion ;
- L'état « parqué » (Park) indique que le terminal ne peut ni recevoir ni émettre. Il peut seulement se réveiller de temps en temps pour consulter les messages émis par le maître. Dans cet état, il utilise un minimum d'énergie, il n'est pas comptabilisé dans un piconet et peut être remplacé par un autre terminal dans les 7 connexions que peut recevoir un maître ;
- L'état suspendu (Hold) indique que le terminal ne peut que recevoir des communications synchrones de type SCO. De ce fait, le terminal se met en veille entre les instants synchrones de réception de paquet ;
- L'état de repos actif (Sniff) permet au terminal de décider des slots pendant lesquels il travaille et de ceux pendant lesquels il se met à l'état de repos ;
- Après l'échange de données, le module actif peut retourner en mode d'attente ou adopter l'un des trois états suivants permettant de réduire la consommation :
  - Mode de maintien (Hold) dans lequel l'appareil reste actif dans le « piconet » ;
  - Mode « renifleur » (Sniff), l'esclave est programmé pour se mettre périodiquement à l'écoute sur le « piconet » afin de déterminer si ce dernier désire lui envoyer des données ;
  - Mode parqué (Park) : l'esclave se retire temporairement du « piconet » et se resynchronise périodiquement sur le maître.

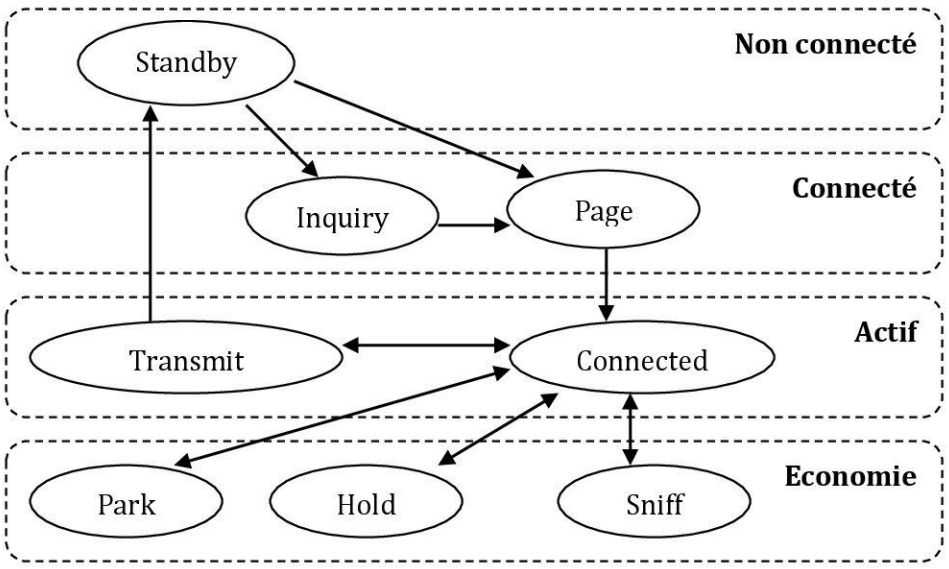


Figure.1. Différents états d'un terminal Bluetooth

Structure des paquets Bluetooth

La figure (23) montre le format générique de tous les paquets *Bluetooth*. Dans cette section, nous expliquerons brièvement la structure du code d'accès et de l'en-tête. Nous présenterons également les paquets spéciaux de séquence de sauts de fréquence (FHS) qui sont utilisés lors de la découverte de périphériques et de l'établissement de la connexion.

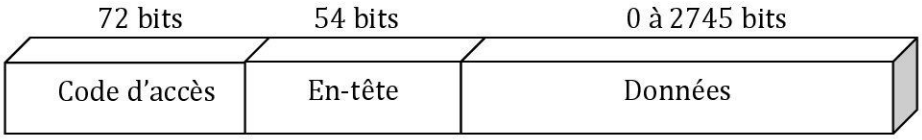


Figure.2. Structure du paquet Bluetooth

Code d'accès

- La structure du code d'accès est illustrée dans la figure (24) ;
- Ce champ de 72 bits permet de transporter le code d'accès tout en effectuant une synchronisation entre les composants *Bluetooth* ;
- Comme le montre la figure (24), cette zone se compose de 4 bits de préambule 0101 ou 1010, permettant de détecter le début de la trame, puis de 64 ou 68 bits pour le code et enfin de 4 bits de terminaison, lorsque le corps fait 64, permettant de détecter la fin de la synchronisation en utilisant les séries 0101 ou 1010 ;

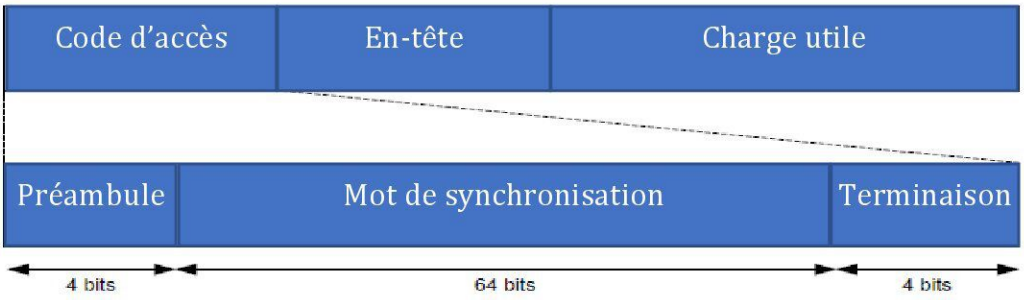


Figure.3. Structure du code d'accès

- Un récepteur *Bluetooth* écoute sur le fuseau hertzien le **préambule** (Preambule) et décode le mot de Synchronisation (Synchronization Word) qui suit ce **préambule**.
- Le **mot de synchronisation** est le même pour tous les paquets envoyés dans un piconet ;
- Le **mot de synchronisation** est construit sur la base de l'adresse BD_ADDR du maître du piconet ;
- Si le maître est le récepteur d'un paquet, il le saura (il peut le connaitre) à partir du mot de synchronisation ;
- Si la destination est un esclave, tous les esclaves du piconet doivent continuer à lire l'en-tête pour décoder l'adresse AM_ADDR ;
- Le code d'accès comprend également 4 bits de fin utilisés lors de la synchronisation, mais cela n'est inclus que si une charge utile est incluse

dans le paquet. Ceci explique pourquoi le code d'accès peut être de 68 ou 72 bits.

Entête

- L'en-tête du paquet *Bluetooth* est illustré sur la figure (25) ;
- L'en-tête contient l'adresse AM_ADDR qui indique l'AM_ADDR de l'esclave, selon qu'il s'agisse d'un paquet esclave-maître ou maître-esclave ;

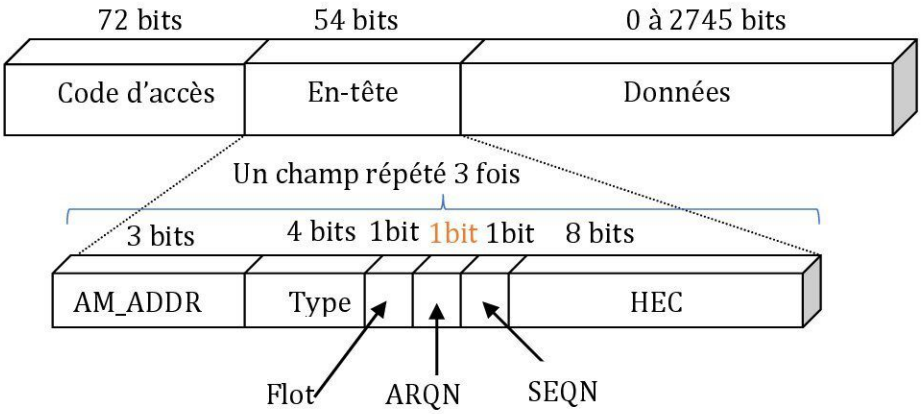
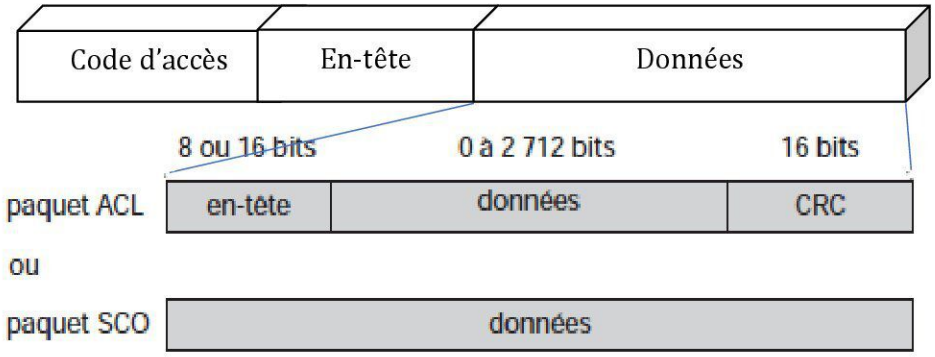


Figure.25. En-tête du paquet Bluetooth

- Les 54 bits de ce champ consistent en 3 fois une même séquence de 6 champs de longueur 18 bits (3, 4, 1, 1, 1 et 8 bits) ;
- Ces champs servent à indiquer l'adresse d'un membre actif du piconet, ainsi qu'un numéro de code, un contrôle de flux piconet, une demande d'acquittement et un contrôle d'erreur des transmissions ;
- Le champ de 18 bits est répété 3 fois de suite pour être sûr de sa réception correcte au récepteur. Il contient les bits :
  - AM_ADDR (3 bits) : L'adresse de l'esclave actif. 0 pour le broadcast et 1 à 6 pour le périphérique ;
  - Type (4 bits) : SCO, ACL, NULL, POLL / type de FEC / durée du payload ;
  - Flow (1 bit) : Contrôle de flow pour signaler que la mémoire tampon est pleine ;
  - ARQN (1 bit) : Indication de l'acquittement (ACK) ;
  - SEQN (1 bit) : Numéro de séquence (Les paquets sont numérotés sur un bit grâce à ce champ. En cas de répétition d'un même paquet, le champ SEQN n'est pas modifié) ;
  - HEC (8 bits) : « Header Error Control » : code détecteur d'erreurs.

Zone de données

La zone de données s'étende 0 à 2745 bits. Elle contient une zone de détection d'erreur sur 1 ou 2 octets.



Différents types de paquets

⇨ Trois grands types de paquets sont définis dans *Bluetooth* ;

Les paquets de contrôle

Ces paquets permettent de gérer la connexion des terminaux ;

Les paquets SCO (communications synchrones)

- paquets DV (Data Voice) qui portent à la fois des données et de la parole ;
- paquets HVy (High quality Voice), sans correction, la valeur y indique le type de contrôle d'erreur dans le paquet.

Les paquets ACL (transfert de données asynchrones)

- paquets DMx (Data Medium) avec un encodage permettant la correction des erreurs. La valeur x, qui vaut 1, 3 ou 5, indique la longueur du paquet en nombre de slots ;
- paquets DHx (Data High), sans correction, permettant ainsi un débit effectif plus élevé.

Le paquet FHS

⇨ Le paquet FHS est un paquet spécial qui est utilisé pour échanger des informations de synchronisation entre les appareils *Bluetooth* ;

⇨ Ce paquet est illustré dans la figure (26).

LSB MSB

Parity bits	LAP	Unde fined	SR	SP	UAP	NAP	Class of device	AM_ ADDR	CLK_{27-2}	Page scan mode
34	24	2	2	2	8	16	24	3	26	a

Figure.4. Le paquet FHS

Tableau.2. Différents types de paquets Bluetooth

Types	Lien	Nom	Signification	Slot	Caractéristiques
Contrôle		ID	Identify	1/2	Utilisé pour l' inquiry et le paging Code d'accès uniquement
		NULL	Null	1	Code d'accès du canal et un en-tête, sans acquittement, pour envoyer de l'information sur le lien
		POLL	Poll	1	Idem NULL, mais avec acquittement, envoyé par le maître à ses esclaves pour tester les liens
		FHS	Frequency Hopping Synchronisation	1	Adresse du maître (donc séquence de sauts) et horloge du maître, pour la synchronisation du piconet, avec FEC et CRC
Voix	SCO	HV	High Quality Voice	1	Voix avec correction d'erreur 1/3 ou 2/3 FEC sans CRC
Données/voix	SCO	DV	Data Voice	1	Voix, données avec correction 2/3 FEC et CRC
Données	ACL	DM	Data Medium	1, 3, 5	Données avec CRC et correction 2/3 FEC
		DH	Data High	1, 3, 5	Idem DM, sans correction d'erreur
		AUX	Auxiliaire	1	Idem DH, sans CRC

Contrôle d'erreur

Codage FEC

- Les données peuvent être protéger par le code correcteur d'erreur FEC (Forward Error Correction) ;
- C'est un mécanisme de détection et correction d'erreurs utilisé dans le codage canal. Il consiste en l'introduction de bits de redondance à la suite de bits de données.
- Dans ce type de protection, nous avons :
  - Le code FEC 2/3 nécessite 3 bits pour en protéger 2 bits. (Exemple : 240 bits utiles pour 360 bits) ;
  - Le code FEC 1/3 nécessite 3 bits pour en protéger 1 bit. (Exemple : 120 bits utiles pour 360 bits).
- Cette protection réduit donc le débit utile mais, en contrepartie, elle permet la correction des paquets erronés sur la liaison.

Codage CRC

- Tous les paquets Bluetooth contiennent un champ de contrôle de redondance cyclique (CRC) qui apparaît à ou près de la fin du paquet ;

- C'est un mécanisme qui est couramment utilisé pour détecter les cas où les données transmises ont été involontairement modifiées en raison de problèmes tels que les collisions ;
- Lorsqu'un nouveau paquet est formulé par la couche liaison (côté Emetteur), une valeur CRC est calculée en appliquant l'algorithme CRC. La valeur résultante est ensuite ajoutée au paquet. Lors de la réception de ce paquet, la couche liaison, dans le dispositif de réception, recalcule le CRC et compare le résultat avec la valeur CRC incluse dans le paquet reçu. Si les deux valeurs ne sont pas les mêmes, on en conclut qu'un ou plusieurs bits dans le paquet transmis ont été modifiés et le paquet est rejeté.

Mécanisme ARQ

- Bluetooth permet de combiner la correction d'erreurs par redondance FEC et la correction par un mécanisme protocolaire par répétition ARQ ;
- Si le récepteur détecte des erreurs, le paquet est jeté et une retransmission est demandée à l'émetteur ;
- Dans ce cas, les paquets sont retransmis jusqu'à ce qu'un acquittement soit reçu, ou que le quantum de temps soit dépassé ;
- Les paquets sont numérotés sur un bit grâce au champ SEQN ;
- En cas de répétition d'un même paquet, le champ SEQN n'est pas modifié ;
- En cas d'émission d'un nouveau paquet, la valeur de SEQN est modifiée ;
- Dans le sens de transmission opposé, le champ ARQN indique un acquittement positif ou négatif. Un paquet de retour perdu est considéré comme un acquittement négatif.

Contrôle de Flux

- Le protocole de bande de base recommande l'emploi de files d'attente de type FIFO dans les liens ACL et SCO pour la transmission et la réception ;
- Le gestionnaire de lien remplit ces files d'attente et le contrôleur de lien les vident automatiquement ;
- Pour éviter que la file d'attente de réception soit pleine, ce qui provoquerait des pertes de paquet et de la congestion, on utilise un contrôle de flux. Une indication d'arrêt est transmise lorsque la queue est pleine. Elle est insérée par le contrôleur de lien du récepteur dans l'en-tête du paquet de retour. Lorsque l'émetteur reçoit l'indication d'arrêt, il bloque ses files d'attentes.
- Lorsque le récepteur est à nouveau prêt il envoie un paquet pour continuer la transmission.

Modes de sécurité

- Bluetooth définit trois modes de sécurité optionnels pour les stations :
  1. **Pas de sécurité.** Une station, dans ce mode, n'initie pas de procédure de sécurité ; elle peut supporter ou non l'authentification. C'est le mode de sécurité par défaut ;
  2. **Sécurité au niveau application**. Une fois la connexion L2CAP établie, la station décide des mécanismes de sécurité à utiliser ;

3. **Sécurité au niveau de la connexion** réalisée par le gestionnaire de liens par échange de messages LMP.

- ⇨ Il y a deux niveaux de sécurité pour les appareils : fiable ou non fiable, et trois niveaux pour les services : autorisé et authentifié, authentifié, accès libre.
- ⇨ Bluetooth permet l'authentification, l'autorisation et le chiffrement des données. Il ne fournit pas de vérification de l'intégrité des données.