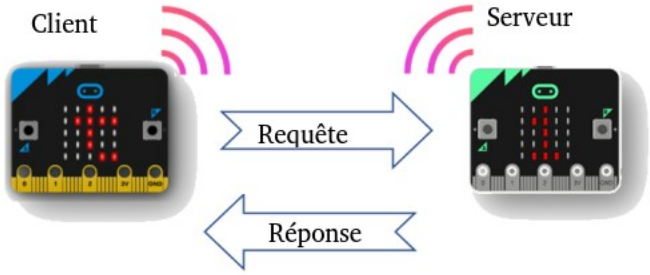
MINIPROJET 4 Simulation d'un protocole d'échange sécurisé TLS

Pour simuler l'établissement d'une communication sécurisée par le protocole TLS utilisé en *HTTPS*, nous allons programmer 2 cartes micro:bit, l'une représentant le **client**, l'autre représentant le **serveur**.



TRAVAIL A REALISER

1. Créer un protocole de communication sécurisé entre 2 cartes en utilisant le chiffrement RSA.

Attention : toutes les cartes communiqueront à l'aide du *channel 23*.

Phase du *handshake* :

- Générer des clés RSA et diffuser les clés publiques
- Envoyer un certificat valide (à partir d'une liste de certificats valides connue de tous)
- Définir une clé de session pour le chiffrement des échanges

Phase de communication :

- Effectuer une requête client sécurisée, représentée par le message « www.client.fr » qui devra s'afficher (scroll) sur la carte serveur.
- Envoyer une réponse serveur sécurisée, représentée par le message « index.html» qui devra s'afficher (scroll) sur la carte client.

2. Présentation orale des travaux et démonstrations.

- Justification des choix effectués.
- Préciser les difficultés rencontrées et les améliorations possibles.

On utilisera les ressources :

- https://microbit-micropython.readthedocs.io/fr/latest/

Et (au choix) les plateformes de simulation en ligne :

- https://fr.vittascience.com/microbit/?mode=code&console=bottom&toolbox=vittascience
- https://python.microbit.org/v/3/reference

TIMELINE

< 28/02	28/02	01/03	07/03	08/03
Préparation des algos Analyse fonctionnelle Travail en ligne	Sprint 1	Sprint 2	Debug	Présentation

EVALUATION :

Fonctionnalités demandées :

• *Mettre en place deux phases distinctes : handshake et communication sécurisée*	*/2*	• *Générer des clés RSA et diffuser les clé publiques*	*/2*
• *Envoyer un certificat et le vérifier*	*/2*	• *Chiffre et déchiffrer un message par la méthode RSA*	*/3*
• *Définir et utiliser une clé de session*	*/3*		

Fonctionnalités BONUS : (+1pt par fct)

• *...*		• ...	

Code :

• *Lisibilité du code & variables explicites*	**/2**
• *Utilisation des classes*	**/2**
• *Pas de répétitions, utilisation de boucles et de fonctions/méthodes*	**/2**
• *Commentaires & docstrings pertinents*	**/2**

REA	• Mettre en œuvre une solution, par la traduction d'un algorithme ou d'une structure de données dans un langage de programmation.			
	J'écris les grandes étapes du code.	J'écris un code qui répond au problème.	J'écris un code rigoureux qui répond au problème. Je documente et justifie mes choix de langage.	... J'explique clairement le cadre et les limites de la solution. Je propose des améliorations et alternatives possibles.
	• Imaginer et concevoir une solution, décomposer en blocs, se ramener à des sous-problèmes simples et indépendants, adopter une stratégie appropriée			
	J'écris quelques fonctions.	J'écris et utilise des fonctions que je documente.	J'écris et utilise des fonctions, des classes et des modules adaptés que je documente.	J'écris et utilise les fonctions, classes et modules les plus adaptés au problème. Je documente et explique mes choix
COM				
	Je réponds aux questions posées en quelques phrases courtes.	Je réponds de façon claire et cohérente, en utilisant un vocabulaire adapté.	Je réponds de façon claire et cohérente, en utilisant un vocabulaire adapté. J'utilise un argumentaire construit.	Je réponds de façon claire et cohérente, en utilisant un vocabulaire adapté et précis. J'utilise un argumentaire construit pertinent, qui soutien efficacement le discours.