Création d’un keylogger

# Initialisation du projet

Un keylogger est un intermédiaire USB que nous allons brancher entre un clavier et un ordinateur. Le but est de récupérer les touches qu’un utilisateur va appuyer sans que cela ne soit détecté par le système.

Avant d’initier le projet, nous devons rappeler que la création et la possession d’un keylogger n’est pas interdit, ce qui est juridiquement illégale, c’est l’utilisation de ce système dans des circonstance autres que sur des appareils que nous possédons. Ce document permet de suivre les étapes que nous avons réalisés pour faire un keylogger, mais en aucun cas ce projet ne doit être détourné à des fin nuisible ou illégale.

Maintenant que nous avons fait un peu de prévention, nous pouvons commencer les explications du projet. Comme nous l’avons mentionné précédemment, l’objectif est de pouvoir construire un intermédiaire entre un clavier et un ordinateur. Pour cela, nous avons commencé nos recherches sur les câblages USB pour pouvoir faire un intermédiaire. Le câblage de l’intermédiaire doit être réalisé de cette manière.

Une image contenant capture d’écran, ligne, diagramme, conception

Description générée automatiquement

Nous avons sur les pins à l’extrémité la masse et l’alimentation en 5 volt. Puis les connecteurs au centre sont utilisés pour transmettre les données. Nous avons les données transmis en polarité positive à le second qui a une polarité négative. Les données sont transmises en série.

# Première version du keylogger

Avec le schéma de la partie précédente, nous avons commencé le montage électrique pour pouvoir réaliser l’intermédiaire. Nous disposions d’un connecteur USB-A mâle et femelle.

Une image contenant intérieur

Description générée automatiquement

Nous avons ensuite soudé des câbles en respectant les couleurs que nous avions définie sur le schéma pour que les connecteurs puissent continuer à transmettre les données. Nous avons utilisé une planche qui permet de connecter des éléments sans soudure pour connecter les deux connecteurs entres eux.

Une image contenant fils électriques, Appareils électroniques, Ingénierie électronique, câble

Description générée automatiquement

A cette étape, nous pouvons vérifier que les connecteurs sons bien branché en connectant un clavier à un ordinateur (un vieil ordinateur pour ne pas causer de problème sur nos machine) pour être sûr que le clavier continue à fonctionner ce qui nous permettras de continuer.

En branchant le clavier à un ordinateur avec cet intermédiaire, nous avons vérifié que le clavier fonctionnait. Ensuite nous avons ajouter une carte Arduino qui récupère les fils D+ et D- sur des broches digitales. Nous avons essayé de comprendre ce qu’il se passais dans la transmission et nous nous sommes rendu compte qu’il aurait fallu implémenter le protocole USB dans l’Arduino pour pouvoir réaliser ce projet. Par manque de temps, nous avons changé notre fusil d’épaule. La seconde partie de ce rapport indique la méthode utiliser pour ne pas à avoir à implémenter le protocole USB.

# Seconde version du keylogger

Pour cette seconde version du keylogger, nous disposons d’une carte Arduino Leonardo, un ESP32 DEVKIT V1. L’objectif est de récupérer les touches appuyer du clavier avec l’ESP32 pour les transmettre à la carte Leonardo, ainsi qu’à distance en wifi. Le schéma de l’installation est représenté ci-dessous.

Une image contenant capture d’écran, conception

Description générée automatiquement

Sur cette représentation, nous avons en rouge le keylogger, et en gris les éléments matériels de la cible. Le choix d’utiliser un Arduino Leonardo est pour le fait que nous pouvons virtualiser un clavier avec celle-ci. L’ESP32 possédé déjà un module Wifi, et le port série peut être utilisé pour recevoir des touches d’un clavier.

## Étape 1 : recevoir et envoyer des touches de l’attaquant à l’ESP32

Pour cette première étape, nous avons fait en sorte que l’ESP32 puisse ouvrir un point d’accès réseaux, pour que l’attaquant puisse s’y connecter. De plus l’ESP32 ouvre un serveur d’écoute sur le port 9999. Donc pour cette première étapes, l’attaquant se connecte à l’ESP32 (en étant en préalable sur le réseau). Puis lorsque l’attaquant envoie un caractère, l’ESP32 transmet le même caractère à l’attaquant.

Une image contenant capture d’écran, texte, Police, diagramme

Description générée automatiquement

Une image contenant Appareils électroniques, Appareil électronique, Appareil de saisie, ordinateur

Description générée automatiquement

Pour commencer, nous avons connecté l’ordinateur au réseau « keylogger-wifi » de l’ESP32. Puis nous avons utilisé netcat pour nous connecter au service de l’ESP32.

Une image contenant capture d’écran, texte, Logiciel multimédia, Logiciel de graphisme

Description générée automatiquement

Réponse de l’ESP32

Message envoyer par l’attaquant

De cette manière, nous pouvons constater que nous pouvons envoyer des caractères à l’ESP32 et que ce dernier peut aussi nous en envoyer.

## Étapes 2 : transmettre le caractère à la carte Leonardo

Maintenant que nous pouvons récupérer les caractères donnés par l’attaquant, nous devons envoyer ces caractères à la carte Leonardo. Pour le moment, la carte Leonardo va donner les caractères reçus par port série.

Pour connecter la carte Leonardo et l’ESP32, nous avons utilisé les pins TX et RX des deux cartes. Nous avons donc une schématisation de l’installation ci-dessous.

Une image contenant capture d’écran, diagramme, conception

Description générée automatiquement

Nous avons donc obtenu le résultat suivant :

Une image contenant ordinateur, intérieur, Ordinateur personnel, Matériel d’ordinateur

Description générée automatiquement

Lorsque nous avons envoyé « bonjour » à partir de l’attaquant, les données sont transmises jusqu’à la carte Leonardo qui les affiches en port série.

A noté que nous avons brancher les broches VIN et GND de l’ESP32 à la carte Leonardo pour que l’ESP32 soit alimenté par la carte. Les schémas de branchements seront détaillé par la suite.

## Étapes 3 : Virtualiser un clavier avec la carte Leonardo

Maintenant que nous pouvons récupérer les caractères dans la carte Leonardo, nous pouvons maintenant virtualiser un clavier avec cette dernière pour que le PC écrive les caractères reçus.

Le schéma de branchement est le même que précédemment.

Une image contenant intérieur, ordinateur, Ordinateur personnel, ordinateur portable

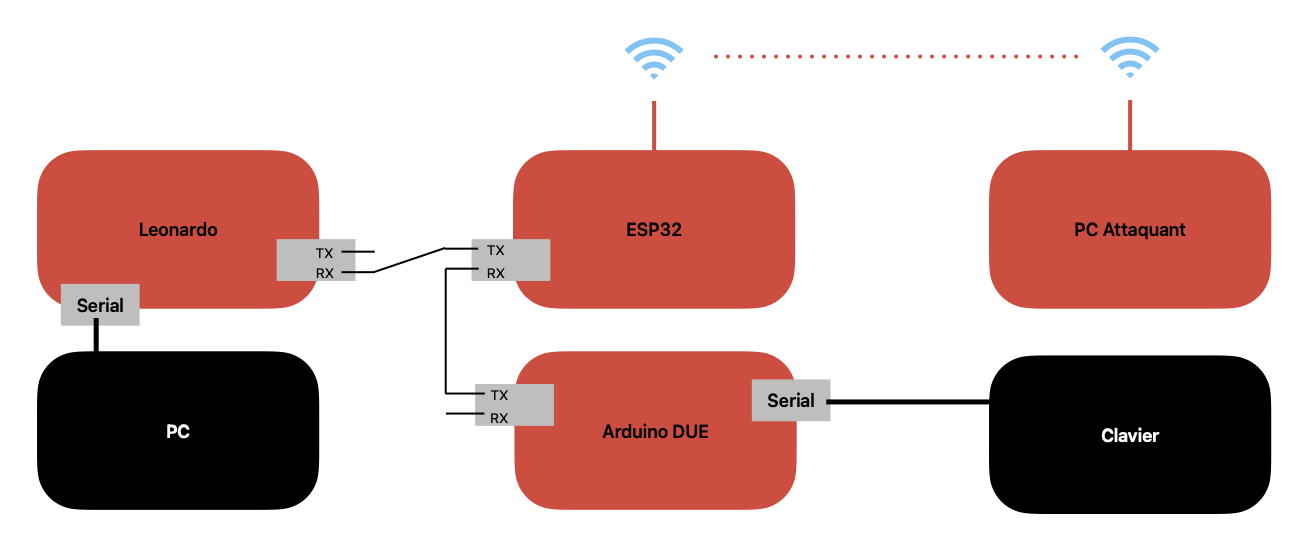
Description générée automatiquement

Dans cette image, nous avons envoyé un message depuis le PC de l’attaquant qui les à transmis jusqu’au PC de la cible. Le texte à étais écrit dans un bloc note. Nous pouvons donc voir que la carte Leonardo et bien détecté comme un clavier ce qui nous permet de transmettre des touches.

## Étapes 4 : Récupérer les touches d’un clavier

Pour réaliser cette étape, initialement, nous voulions connecter un clavier à l’ESP32 sur le port série avec un câble OTG. Nous avons rencontré un problème, le câbles que nous voulions réaliser ne permet pas d’alimenter le clavier. Donc pour palier à ce problème, nous avons ajouté une nouvelle carte Arduino DUE qui permet de lire le clavier.

Le câble est actuellement disposé par le schéma ci-dessous.



Avec ce câblage, nous avons deux méthode de fonctionnement, Le premier qui permet d’envoyer des caractères depuis le PC de l’attaquant.

Une image contenant capture d’écran, conception

Description générée automatiquement

Le second fonctionnement permet d’envoyer une touche du clavier au PC, tout en récupérant la touche au PC de l’attaquant.

Une image contenant capture d’écran, diagramme, conception

Description générée automatiquement

Nous pouvons maintenant utiliser le système comme keylogger.

<IMG du système en fonctionnement>

## Étapes 5 : Créer un script pour l’attaquant

## Schéma détaillé de l’assemblage des cartes

Dans cette partie nous allons visualiser le câblage réalisé lors de ce projet de manière détaillé.

Dans un Premier temps nous allons visualiser les broches de chaque carte.

Une image contenant texte, capture d’écran

Description générée automatiquement

Sur cette carte, les éléments que nous avons besoins sont les broches TX et RX (broche 16 et 17) et les broches d’alimentation VIN et GND.

La carte Arduino DUE :

Une image contenant Appareils électroniques, texte, circuit, Ingénierie électronique

Description générée automatiquement

Sur cette carte, nous avons besoins des broches TX (broche D18) du port série qui permet de d’alimenter un clavier et de pouvoir interagir avec. Nous avons aussi besoins des broches VIN et GND pour alimenter cette carte.

Pour la carte Leonardo :

Une image contenant texte, Appareils électroniques, circuit, Composant électronique

Description générée automatiquement

Sur cette carte, les broches que nous avons besoins sont les broches 5V et GND, mais aussi le broche RX (broche PD2). Le port série sera aussi utilisé pour être branché à l’ordinateur, de cette manière la carte sera alimentée, et pourra virtualiser un clavier.

La câblage final de ces trois cartes est définie dans le schéma ci-dessous :

Une image contenant Appareils électroniques, Ingénierie électronique, Composant électronique, circuit

Description générée automatiquement