Chiffrement de bout en bout

Cryptologie moderne – Projet

Thomas Peugnet – 2023

[État de l’art 3](#_Toc146275068)

[Introduction 3](#_Toc146275069)

[L'Ère Pré-E2EE 3](#_Toc146275070)

[Qu'est-ce que l'End-to-End Encryption? 4](#_Toc146275071)

[E2EE dans les Applications de Messagerie 5](#_Toc146275072)

[Futurs Tendances de l'E2EE 5](#_Toc146275073)

[Cas pratique – Utilisation des sockets pour envoyer un message 6](#_Toc146275074)

[Introduction 6](#_Toc146275075)

[Première approche : Sans chiffrement de bout en bout 7](#_Toc146275076)

[Seconde approche : Utilisation du chiffrement de bout en bout 10](#_Toc146275077)

[Conclusion 12](#_Toc146275078)

[Bibliographie 12](#_Toc146275079)

[État de l’art 12](#_Toc146275080)

[Cas pratique 12](#_Toc146275081)

État de l’art

## Introduction

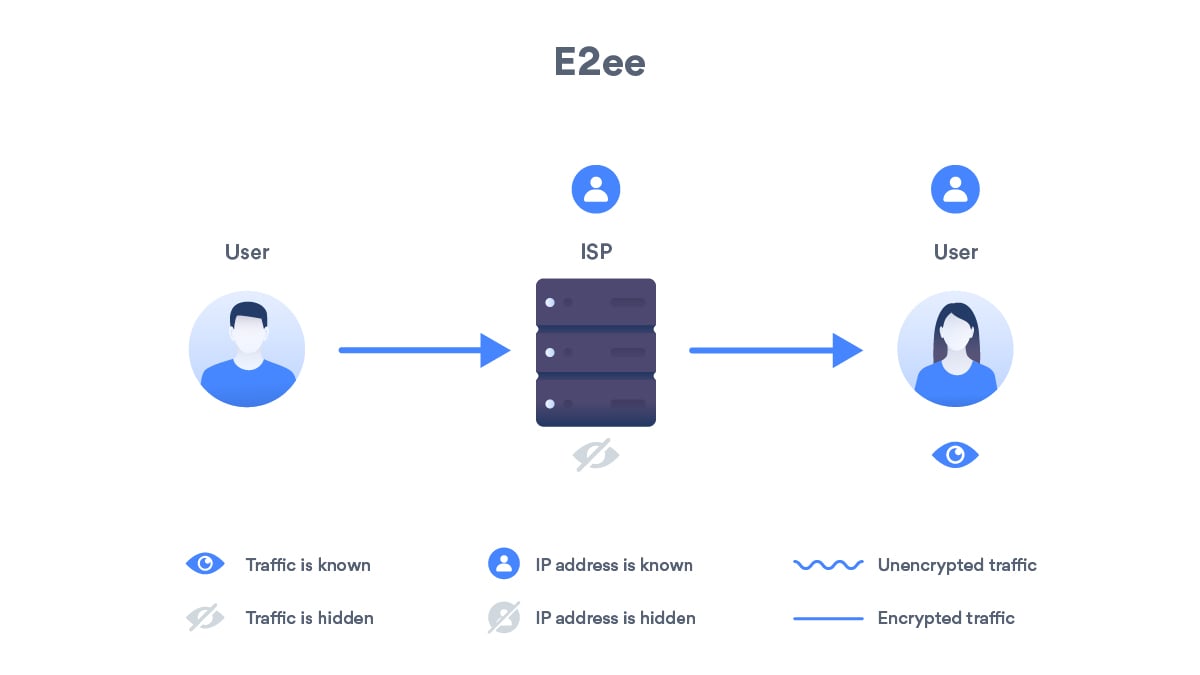
Dans un monde de plus en plus numérique, la sécurité des données et la protection de la vie privée sont devenues des préoccupations majeures. L'encryption de bout en bout (E2EE) est une technologie fondamentale qui a révolutionné la manière dont nous sécurisons nos communications et nos données en ligne. Ce texte a pour objectif de fournir un État de l'art de l'end-to-end encryption, en examinant son rôle dans l'évolution de la sécurité des données et de la confidentialité.

## L'Ère Pré-E2EE

Avant l'adoption généralisée de l'E2EE, la sécurité des données en ligne laissait beaucoup à désirer. Les communications électroniques étaient souvent exposées aux interceptions malveillantes, mettant en danger la confidentialité des informations échangées.

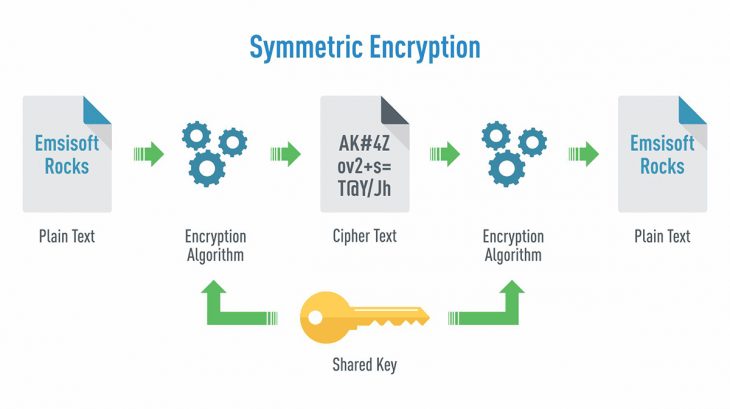
Pendant cette période, les données sensibles pouvaient être compromises lors de leur transmission sur Internet, et les fournisseurs de services de communication avaient un accès sans restriction aux données de leurs utilisateurs. Les pirates informatiques et les gouvernements pouvaient exploiter ces vulnérabilités pour accéder à des informations privées, soulevant ainsi d'énormes préoccupations en matière de sécurité et de confidentialité.

De façon plus schématique, on peut regarder le dessin suivant.



## Qu'est-ce que l'End-to-End Encryption?

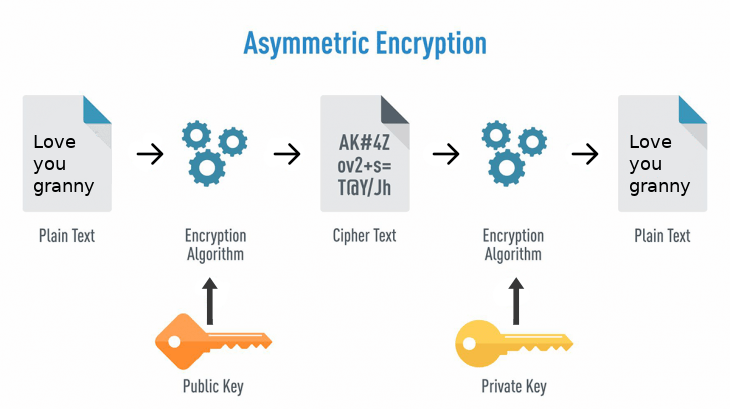
L'end-to-end encryption (E2EE), ou chiffrement de bout en bout, est une technologie de sécurité de pointe qui garantit que seuls l'expéditeur et le destinataire d'un message ou d'une donnée peuvent la lire. Contrairement aux méthodes de chiffrement traditionnelles, où les clés de déchiffrement étaient souvent détenues par un tiers, l'E2EE permet aux utilisateurs de conserver un contrôle total sur leurs données.



Le fonctionnement de l'E2EE repose sur l'utilisation de clés de chiffrement spécifiques à chaque utilisateur.

Lorsqu'un expéditeur envoie un message, il est chiffré avec la clé publique du destinataire, et seule la clé privée du destinataire peut le déchiffrer. Ainsi, même si un pirate informatique intercepte le message en transit, il ne peut pas le lire sans la clé privée du destinataire.

A noter que le schéma ci-dessus indique un chiffrement de bout en bout avec deux clés identiques partagées. Le schéma ci-dessous illustre davantage le fonctionnement avec un système de clés publiques/privées.



## E2EE dans les Applications de Messagerie

L'E2EE a été largement adopté dans les applications de messagerie populaires telles que WhatsApp, Signal et Telegram. Cette adoption généralisée a considérablement amélioré la confidentialité des conversations en ligne. Les utilisateurs peuvent désormais échanger des messages, des photos et des vidéos en toute confiance, sachant que leurs communications sont sécurisées contre les regards indiscrets.

Cependant, l'implémentation de l'E2EE dans les applications de messagerie n'est pas sans défis. Les entreprises doivent résoudre des problèmes tels que la gestion des clés de chiffrement, l'interopérabilité entre différentes applications, la convivialité de l'interface utilisateur et la conformité réglementaire. De plus, bien que l'E2EE protège les messages en transit, elle ne garantit pas nécessairement la sécurité des appareils des utilisateurs, qui peuvent toujours être vulnérables aux attaques.

Enfin, le chiffrement de bout en bout garantit la confidentialité des données durant le transit, mais pas avant ce dernier : si le développeur de l’application de messagerie décide de récupérer les données durant le moment où elles sont tapées sur le clavier, avant que l’utilisateur clique sur Envoyer, ces données ne sont pas nécessairement régies par du chiffrement de bout en bout.

## Futurs Tendances de l'E2EE

L'E2EE continue d'évoluer pour répondre aux besoins croissants de sécurité et de confidentialité des utilisateurs. Les chercheurs et les développeurs travaillent sur des améliorations pour simplifier la gestion des clés de chiffrement, améliorer la résilience contre les attaques, et étendre son utilisation à d'autres domaines tels que la téléphonie et la vidéoconférence.

De plus, la montée en puissance de l'informatique quantique pose de nouveaux défis à la sécurité de l'E2EE. Les chercheurs se penchent sur des technologies de chiffrement post-quantique qui seront capables de résister aux attaques des ordinateurs quantiques puissants. Cette avancée est cruciale pour maintenir la sécurité de l'E2EE à long terme.

# Cas pratique – Utilisation des sockets pour envoyer un message

## Introduction

Pour illustrer l’intérêt du chiffrement de bout en bout, nous allons procéder à une petite expérience entre deux protagonistes, Bob et Alice. Ces derniers souhaitent pouvoir communiquer, certes sur la même machine (127.0.0.1) mais en utilisant la technologie des sockets.

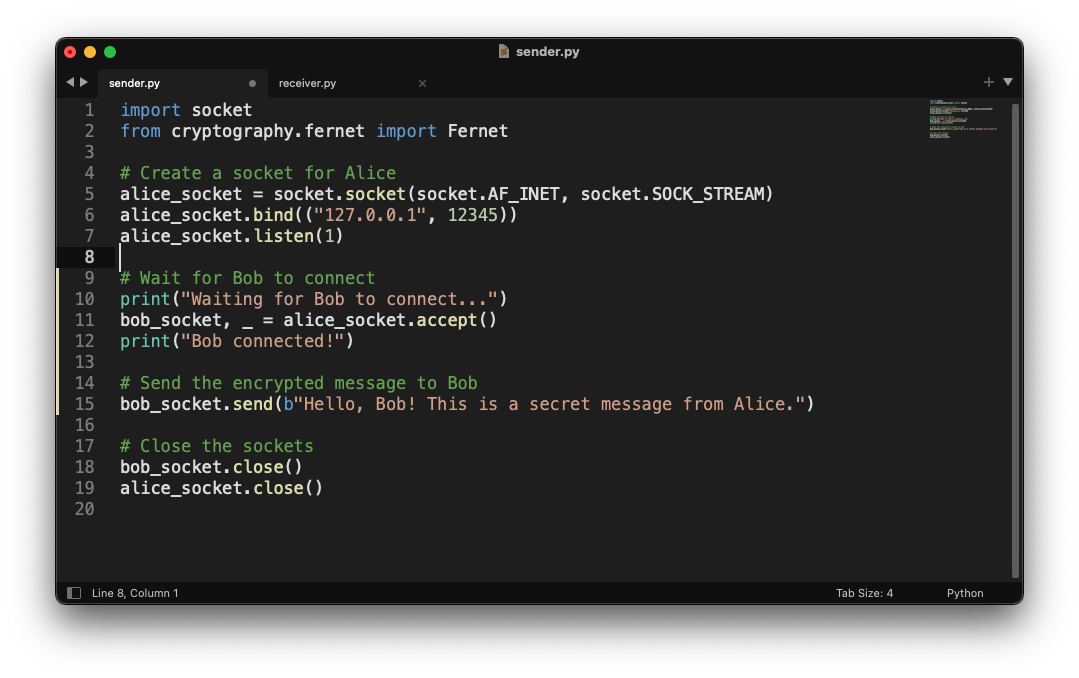
Pour cela, ils vont développer un programme (sender.py et receiver.py) afin de pouvoir discuter. Sender.py fera hôte de serveur, et écoutera les connexions entrantes. Receiver.py se connectera et recevra le message envoyé à la connexion au serveur.

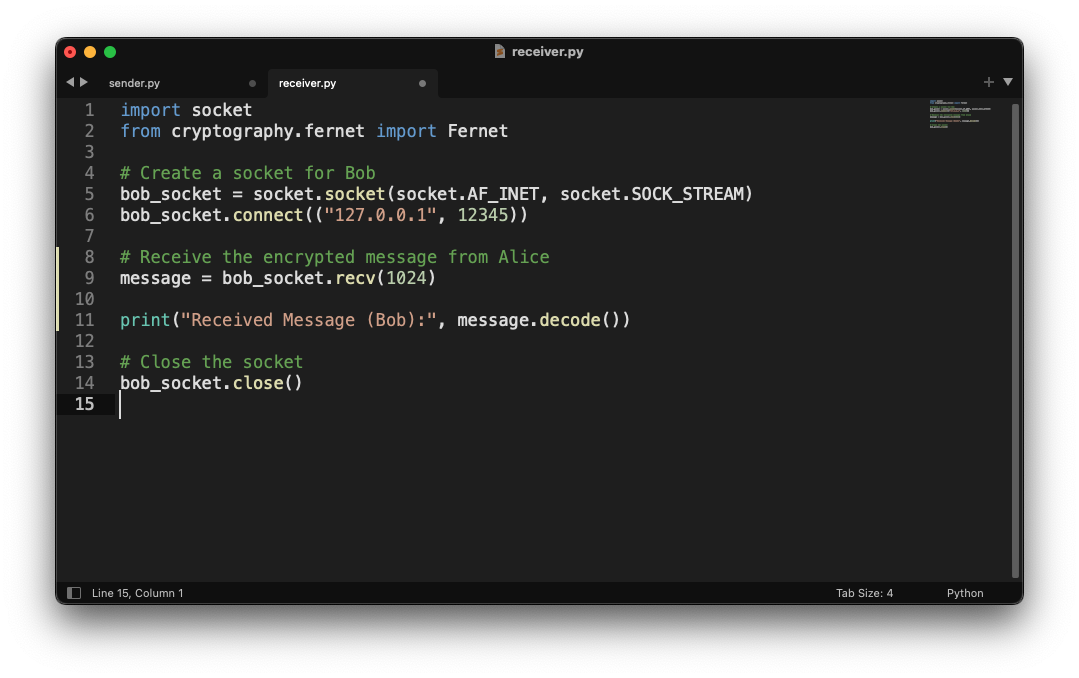
Cependant, un méchant pirate va tenter de d’intercepter leur message (oui, toujours sur la même machine, on fait avec les moyens matériels du bord.. !). Ce pirate se nomme BadBob.

BadBob, par l’intermédiaire du logiciel Wireshark, va tenter de récupérer les paquets transmis entre Bob et Alice.

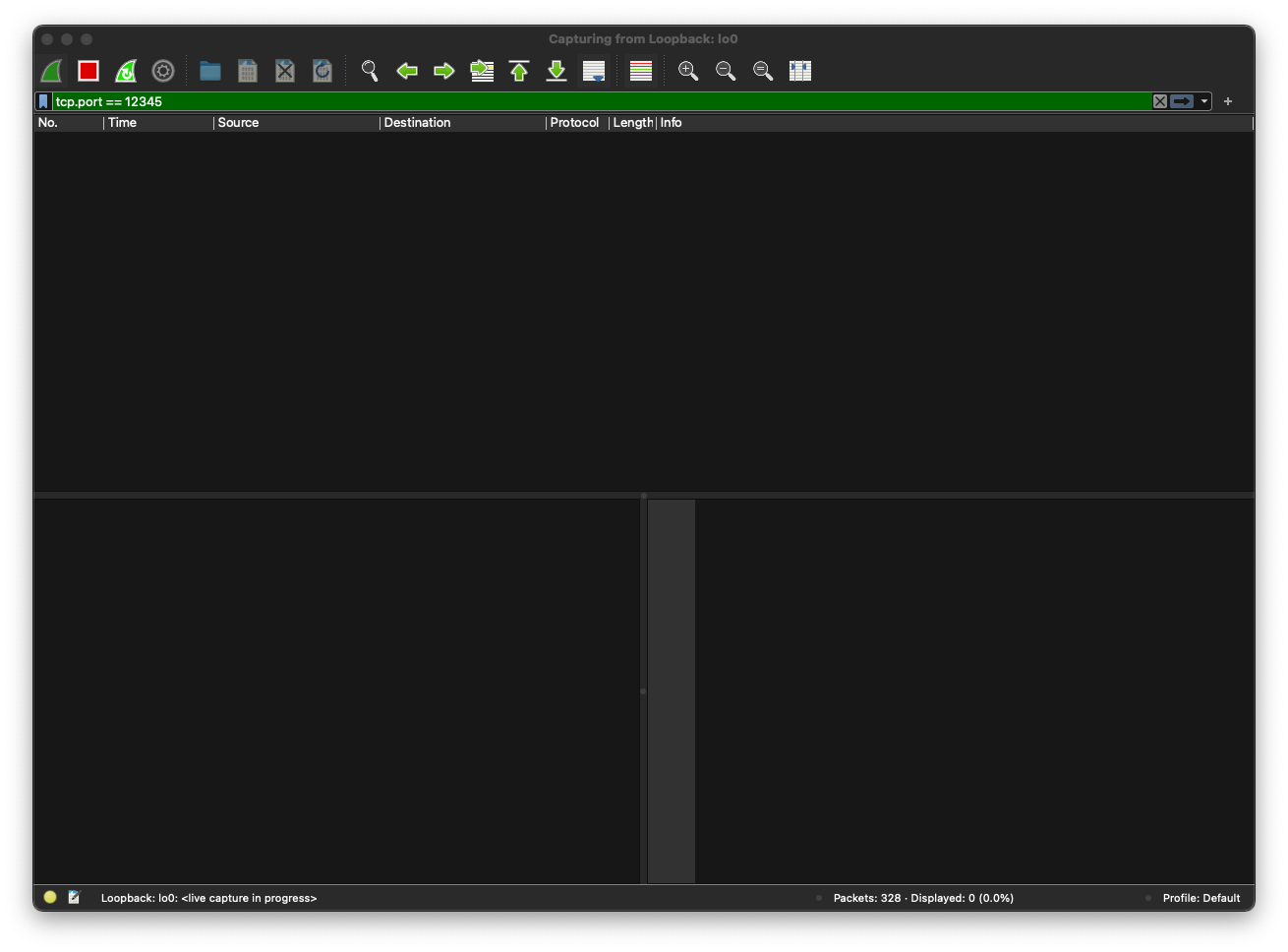
## Première approche : Sans chiffrement de bout en bout

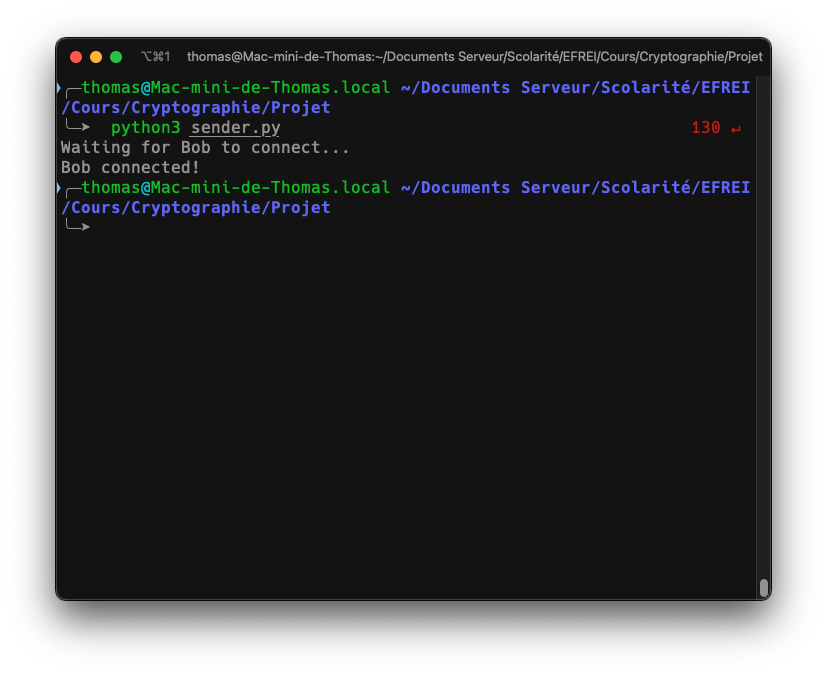
Nous avons donc rédigé deux programmes, sender.py et receiver.py, possédant le code ci-dessous :

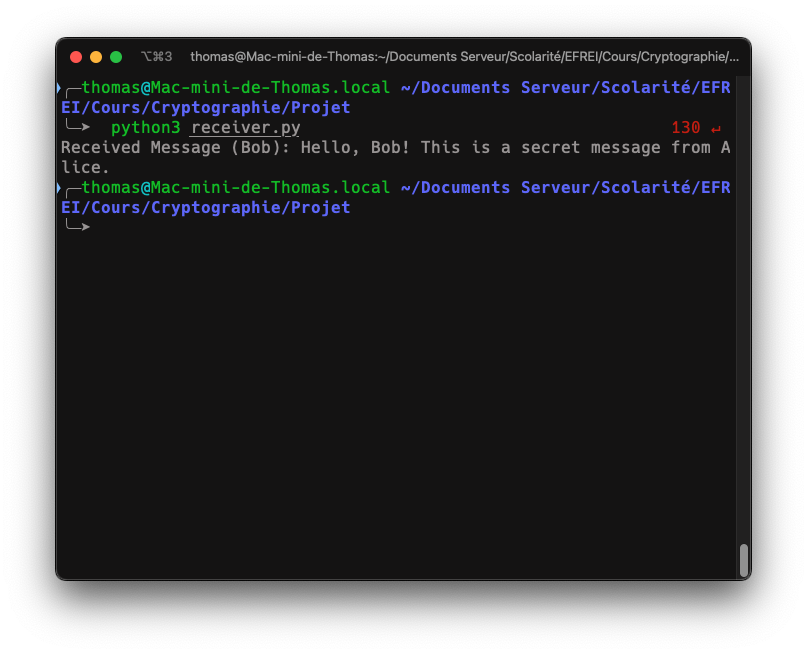
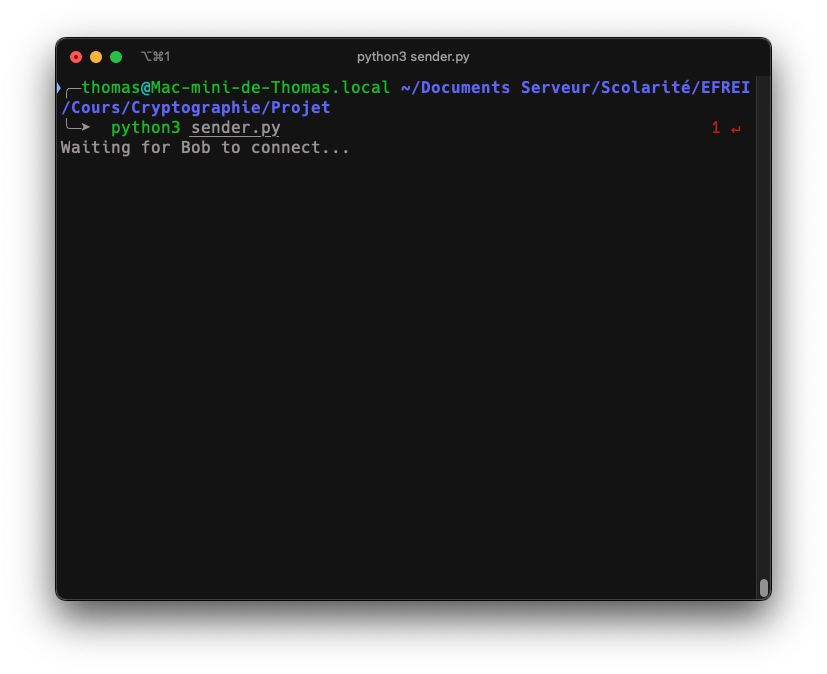




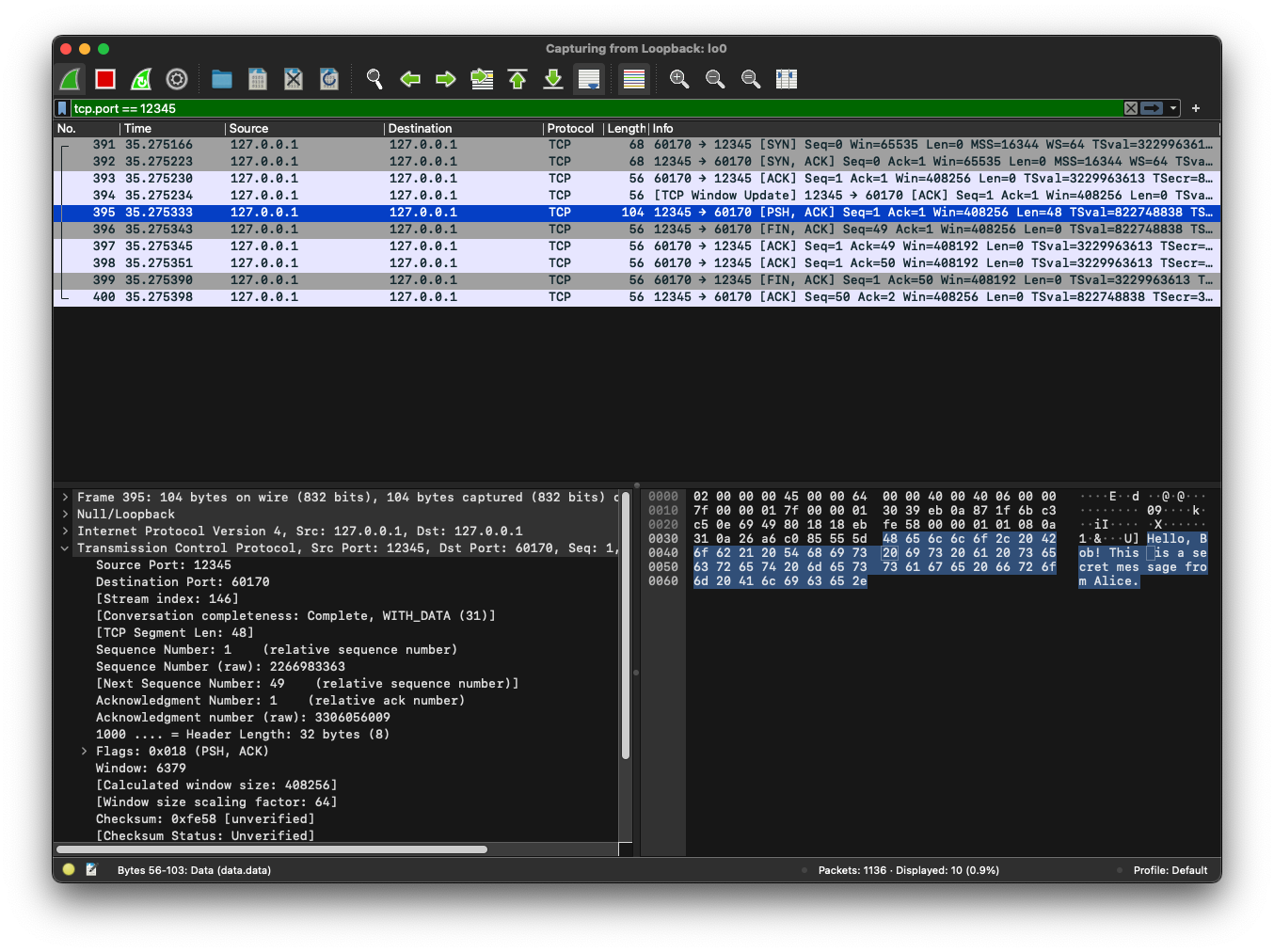
Puis, BadBob a démarré la capture wireshark. Par chance, il a réussi à savoir que le port d’écoute entre Bob et Alice était 12345.



Enfin, Bob et Alice on démarré leur échange. Alice a donc créé le serveur, et a attendu que Bob s’y connecte. Par la suite, Bob s’est connecté et a reçu le message envoyé par Alice.



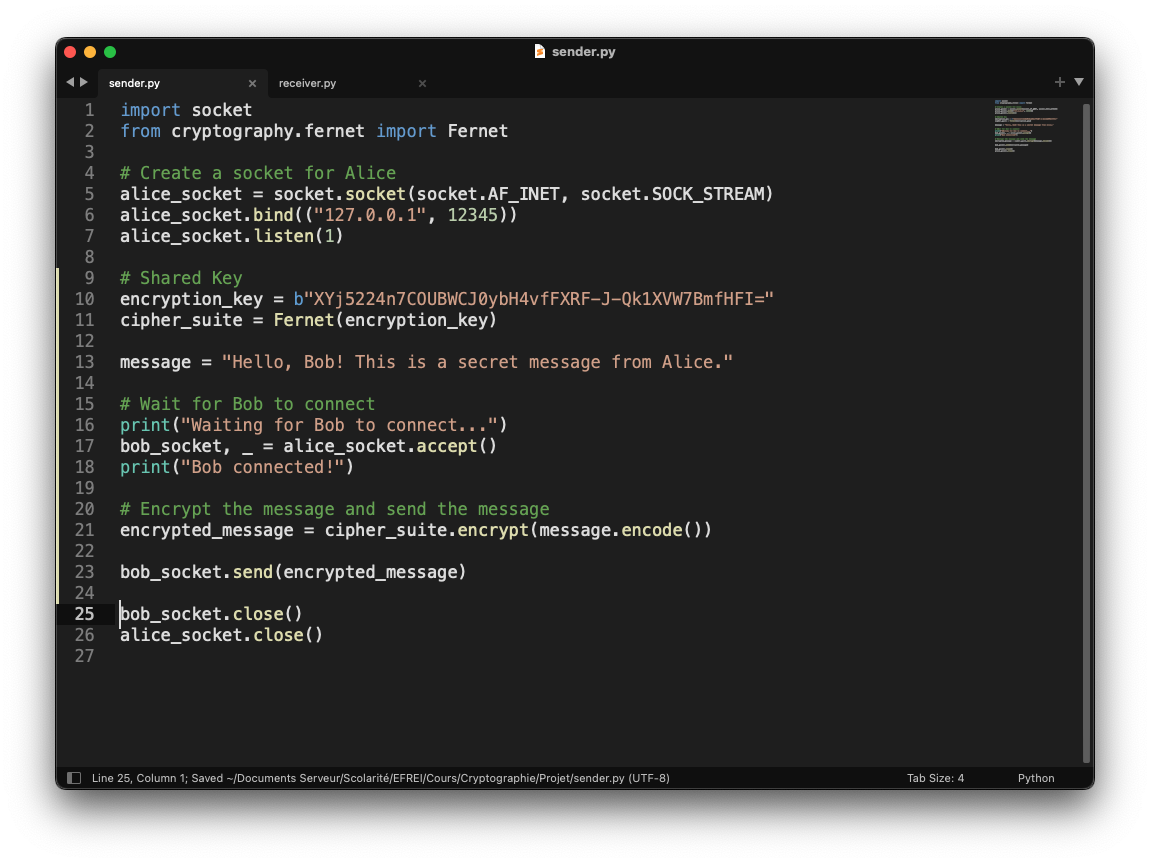
BadBob a pu capturer les paquets, et en voici la trace sur Wireshark :

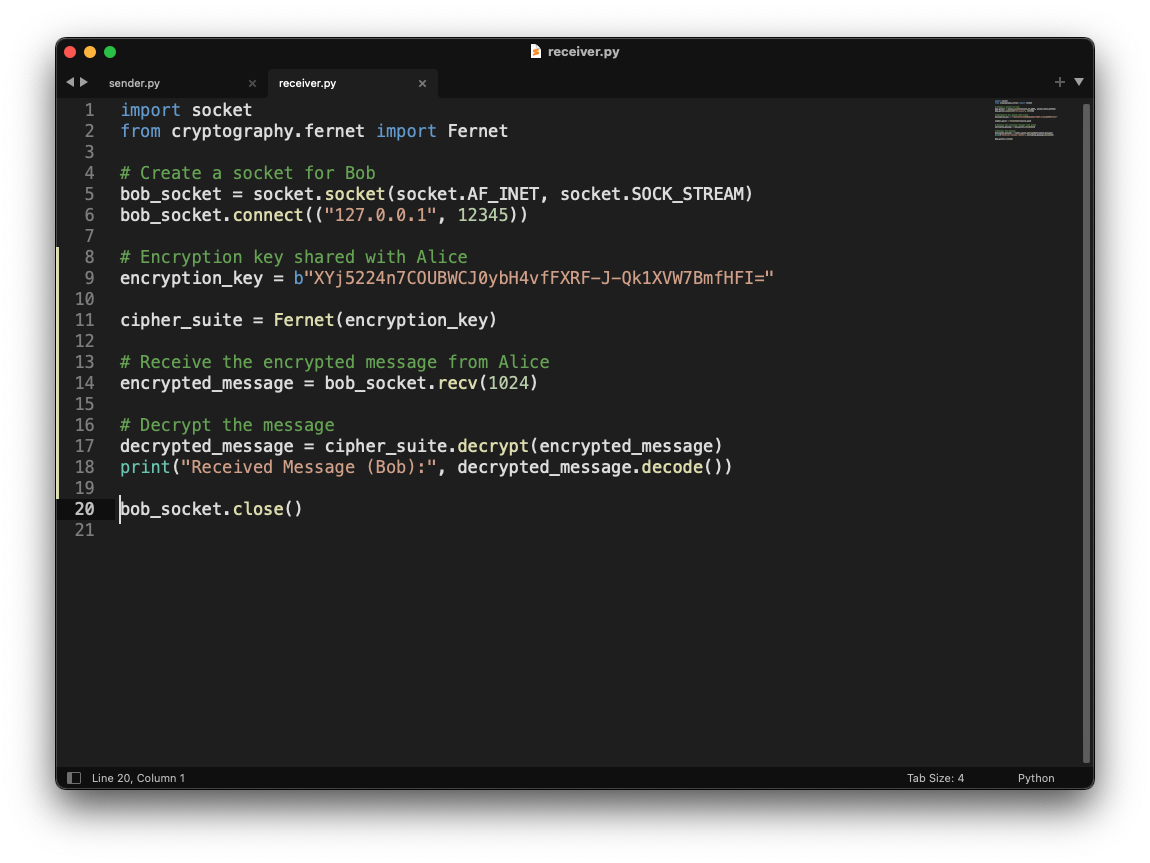


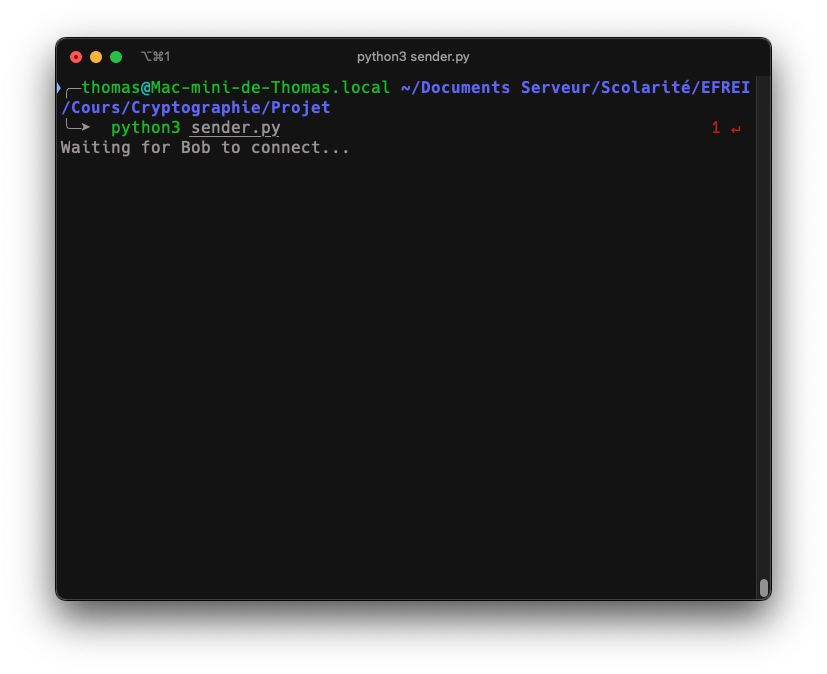
Nous pouvons remarquer que, sur la capture ci-dessus, le paquet sélectionné contient le message envoyé par Alice. BadBob a donc réussi à intercepter le message.

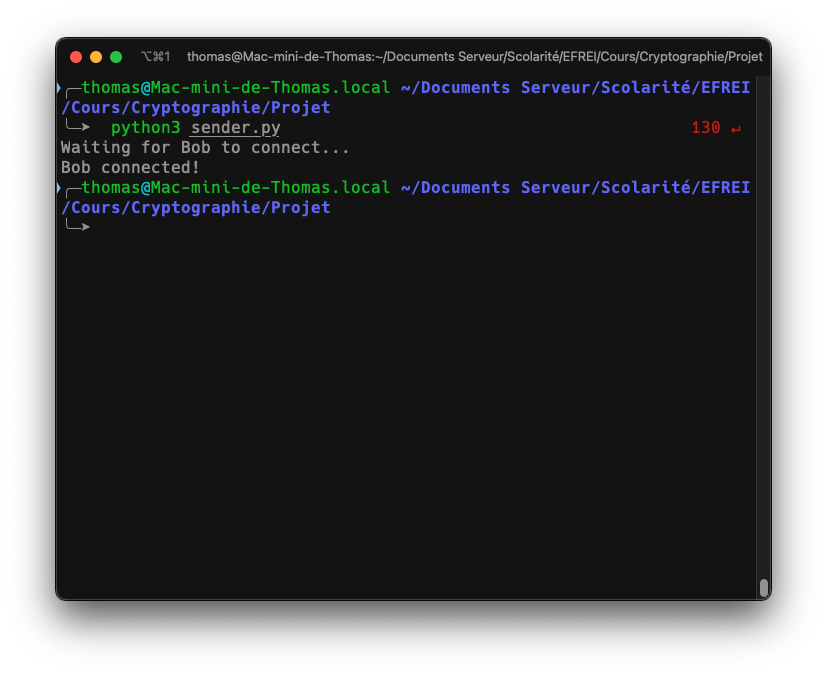
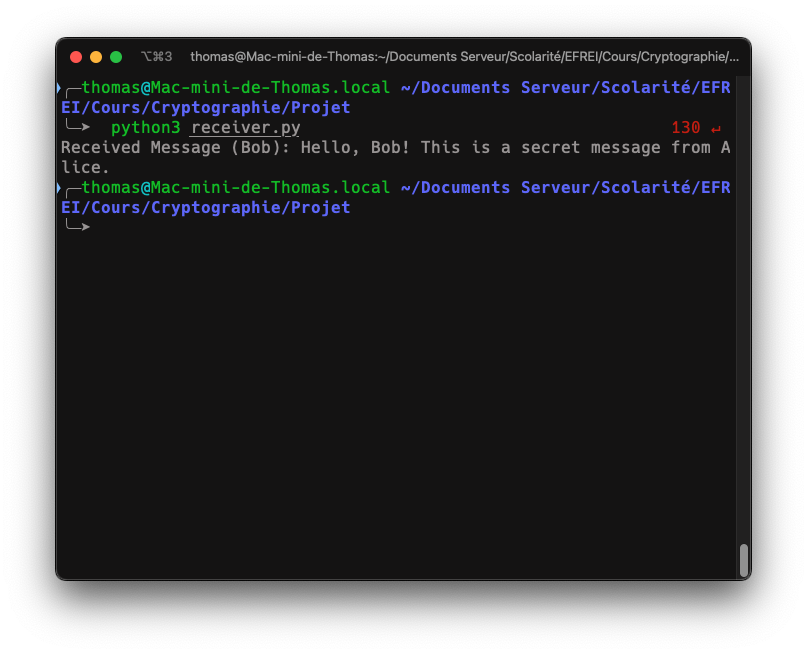
## Seconde approche : Utilisation du chiffrement de bout en bout

Nous nous retrouvons dans un scénario identique, voici les fichiers sender.py et receiver.py, modifiés pour intégrer une solution de chiffrement de bout en bout. Nous utiliserons une clé partagée définie en statique dans le code.

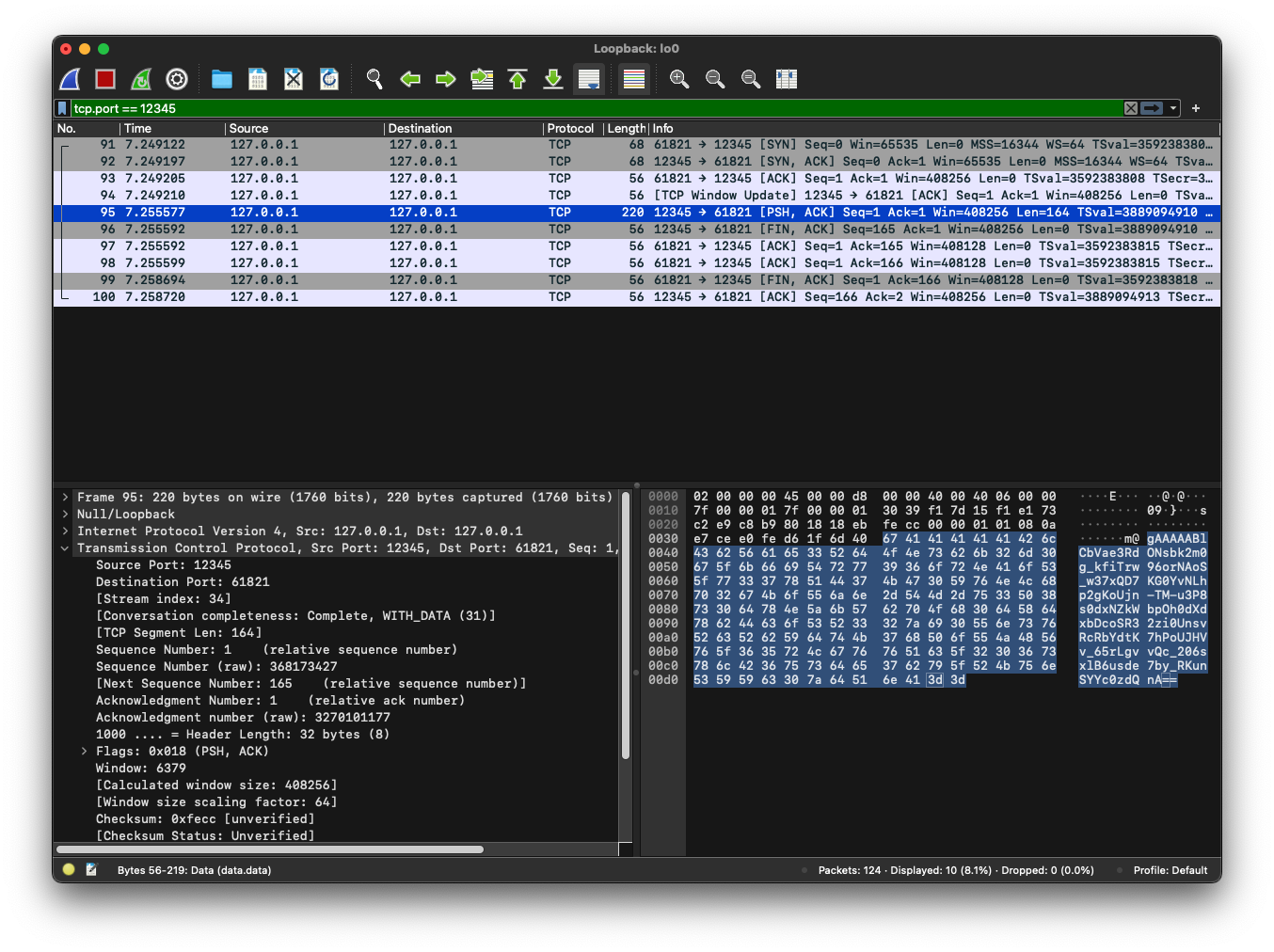




Par la suite, Alice et Bob ont relancé leurs programmes pour s’envoyer et recevoir un message.



Cette fois-ci à nouveau, BadBob a réussi à intercepter le message transmis. En voici la capture Wireshark.



Nous pouvons, cette fois-ci, noter que le paquet contenant le message n’est plus lisible en clair, le message est donc devenu inconnu de BadBob.

# Conclusion

L'end-to-end encryption (E2EE) a marqué une avancée significative dans la protection de la vie privée et de la sécurité des données dans un monde numérique en constante évolution. Elle a radicalement transformé la manière dont nous sécurisons nos communications en ligne et a renforcé la confiance des utilisateurs dans la confidentialité de leurs échanges.

Cependant, les défis techniques et les débats en cours sur l'équilibre entre la sécurité et les besoins des forces de l'ordre continueront à façonner l'avenir de l'E2EE. Il est essentiel de rester attentif aux développements futurs dans ce domaine en constante évolution pour garantir une sécurité et une confidentialité continues dans nos communications en ligne.

Enfin, par l’intermédiaire d’un simple exemple, nous avons pu comprendre les implications non-négligeables de l’utilisation de ce type de chiffrement.

# Bibliographie

## État de l’art

Wikipédia - <https://fr.wikipedia.org/wiki/Chiffrement_de_bout_en_bout>

Google Messages - <https://support.google.com/messages/answer/10252671?hl=fr>

Kaspersky - <https://www.kaspersky.fr/blog/what-is-end-to-end-encryption/15668/>

IBM - <https://www.ibm.com/fr-fr/topics/end-to-end-encryption>

BlogDuModérateur - <https://www.blogdumoderateur.com/chiffrement-bout-en-bout-e2ee/>

Section.io - <https://www.section.io/engineering-education/end-to-end-encryption/>

NordVPN - <https://nordvpn.com/fr/blog/what-is-end-to-end-encryption/>

## Cas pratique

Wikipédia - <https://en.wikipedia.org/wiki/Network_socket>

GeeksForGeeks - <https://www.geeksforgeeks.org/socket-in-computer-network/>

Oracle - <https://docs.oracle.com/javase/tutorial/networking/sockets/definition.html>

Python - <https://docs.python.org/3/library/socket.html>

Python Cryptography - <https://pypi.org/project/cryptography/>