# LINFO1341: Projet 1

Siberdt Hugo
*Ecole polythechnique de Louvain*
NOMA: 3230200

Schrobiltgen Dorian
*Ecole polythechnique de Louvain*
NOMA: 26462000

***Abstract*—Le présent document présente une analyse de l'activité résau de l'application "Messenger".**

## I. Introduction

L'application Messenger, développée par le groupe Meta, est l'une des applications de messagerie instantanée les plus populaires. Comme toute application de communication, Messenger utilise des protocoles résaux pour permettre l'acheminement des données entre les utilisateurs. Dans le cadre du cours LINFO1341: Réseaux informatiques, nous allons analyser les différents protocoles réseaux utilisés par Messenger à l'aide de l'outil Wireshark. Nous allons tout d'abord nous intéresser au trafic DNS. Ensuite, nous allons examiner comment l'application traverse les NAT ainsi que les protocoles de transport utilisés. Nous étudierons également les mesures de sécurité mises en place pour protéger le trafic réseau et nous finirons par une comparaison de l'activité réseau en fonction de l'utilisation de l'une ou l'autre fonctionnalité proposée par l'application.

## II. Adresses

Avant toutes choses et pour y voir un peu plus clair, nous allons analyser l'origine des différentes adresses IP(v4/v6) utilisées au travers des différentes captures ainsi que les entreprises auxquelles elles appartiennent. Le graphique à la

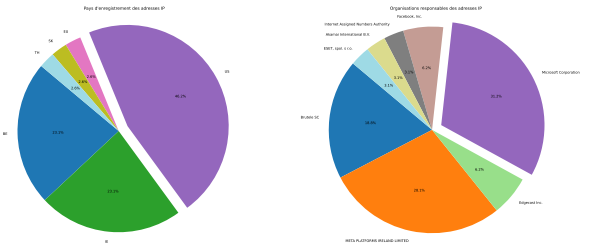


Fig. 1. Pays d'enregistrement des adresses IP(v4/v6) et organisations responsables de ces adresses. La capture a été réalisée lors d'un appel vocal.

Figure 1 nous montre que près de la moitié des adresses IP sont enregistrées aux Etats-Unis. Ce n'est guère surprenant quand on voit que Microsoft Corporation [1], étant basé aux Etats-Unis, en possède presque un tiers. Nous remaquons également qu'une grande partie des adresses (23.1%) sont enregistrées en Irlande. Ceci est dû au fait que META PLATFORMS IRELAND LIMITED, une entreprise en lien avec le groupe Meta Platforms, Inc. [2] soit basée en Irlande. Plusieurs adresses proviennent de Belgique car les fournisseurs d'accès internet (FSI) utilisés lors des captures sont des entreprises belges (Voo et Proximus). Nous noterons toutefois que certaines adresses sont enregistrées en Tchéquie et en Slovaquie. Ceci n'est guère étonnant car l'entreprise ESET, spol. s r.o. [3], basée en Slovaquie, commercialise un antivirus présent sur l'une des machines ayant servi à faire les captures. L'entreprise Amazon Technologies Inc. [4] (qui n'est pas représentée sur le graphique) dispose également de plusieurs adresses IP dans certaines captures. Nous supposons qu'il s'agit là d'une publicité reçue au moment d'effectuer la capture.

## III. Analyse du traffic DNS

Pour commencer, il est important de souligner que les captures de paquets ont été réalisées dans deux zones géographiques relativement "éloignées" l'une de l'autre (une à Louvain-La-Neuve, l'autre à Arlon). Les fournisseurs d'accès internet sont également différents (Proximus d'un côté, Voo de l'autre).
En moyenne, sur les différentes captures effectuées, le traffic DNS ne représente que 0.7% du traffic total. En fonction de la fonctionnalité de l'application utilisée, on constate une certaine variation dans les noms de domaines résolus mais de manière générale, ceux-ci appartiennent en majorité à l'entreprise Meta Platforms, Inc. qui détient, entre autres, l'application Messenger. Les captures analysées ayant été effectuées sous Windows, plusieurs noms de domaines appartiennent également à l'entreprise Microsoft Corporation propriétaire de cet OS.

### A. Résolutions de noms de domaines

Le débit des requêtes DNS est assez variable selon la fonctionnalité de l'application analysée. Il est toutefois possible de remarquer certains éléments. Il y a des résolutions de noms de domaines (qui peuvent être différents selon les captures) lorsqu'un "évènement" se produit sur l'application, c'est-à-dire, lors de la réception d'un message, l'envoi/réception d'un fichier, un appel vidéo, ... Il est toutefois possible de noter que l'envoi d'un message ne semble pas provoquer de résolution de noms de domaines.

### B. Adresses IP

Les requêtes DNS ne sont que des requêtes de type A ou AAAA et ce, indépendamment de la fonctionnalité de l'application utilisée. Aucune des deux familles d'adresses IP ne se démarque par rapport à l'autre, le nombre de requêtes pour l'un ou l'autre type d'adresse étant souvent le même.

### C. Réponses DNS

La plupart des noms de domaine résolus appartiennent à l'entreprise Meta. Contrairement aux requêtes, Les réponses DNS ne sont par contre pas toutes de type A ou AAAA. La grande majorité des réponses sont de type CNAME, c'est-à-dire que le nom de domaine à résoudre (donc celui demandé par la requête) dispose d'un alias. C'est l'adresse IP de cet alias qui sera envoyée comme réponse à la requête. Notons également qu'il arrive d'obtenir une réponse de type SOA (Start Of a zone of Authority) pour demander le nom du serveur responsable d'une certaine zone DNS.

### D. Serveurs autoritatifs

Selon le réseau sur lequel ont été effectuées les captures, les paquets DNS ont plus ou moins de serveurs autoritatifs. Nous avons pu remarquer que les captures réalisées à Arlon avec le FSI Voo, le nombre de serveurs autoritatifs était beaucoup plus élevé que pour les captures réalisées à Louvain-La-Neuve avec le FSI Proximus. La majorité de ces serveurs autoritatifs appartient à Microsoft. Certains serveurs autoritatifs appartiennent aussi à l'entreprise Akamai Technologies, Inc. [5], [6]. Cette société met à disposition des serveurs caches pour diverses autres entreprises afin de permettre, entre autres, une économie du débit internet. Les entreprises Microsoft et Meta font partie de leurs clients.

### E. Records DNS additionnels

Le nombre de records DNS additionnels est plus élevé lorsque nous utilisons le FSI Voo. Ils permettent de fournir des informations supplémentaires aux résolveurs DNS. On remarque que ces records servent en majorité à obtenir les adresses IPv4 et IPv6 des serveurs autoritatifs (via des requêtes de type NS).

### F. Requêtes SOA

Comme mentionné plus haut, il existe plusieurs réponses de type SOA pour demander le nom du serveur responsable de cette zone DNS. La plupart du temps, les entreprises responsables des domaines qui font ce type de requêtes sont Microsoft (et plus particulièrement au service en lien avec Azure DNS [7]) et Akamai. Dans certaines captures, les entreprises ESET et Amazon sont responsables d'un ou plusieurs noms de domaine, ce qui n'est pas étonnant au vu de l'analyse sur les adresses IP faite précédemment.

### G. Remarques et observations

Lorsque nous avons réalisé un appel vidéo, nous avons pu remarquer que le domaine 'edge-stun.facebook.com' [8] était résolu au lancement de l'appel. Il s'agit d'un serveur STUN [9] de l'entreprise Meta. Les serveurs STUN sont utiles pour traverser les NAT. Une section sera dédiée à ce sujet dans la suite de ce rapport.

### H. Comportements DNS inattendus

Il est à noter que certains paquets DNS renvoient une erreur de type "No such name". Il s'agit en réalité d'un nom de domaine qui n'existe plus. Par exemple, nous avons pu remarquer que les requêtes pour la résolution du domaine "wpad.voo.be" lançait ce type d'erreur. En tapant ce domaine dans un navigateur, il y a effectivement une erreur du type "DNS_PROBE_FINISHED_NXDOMAIN". WAPD (Web Proxy Auto-Discovery) est un protocole inventé par Microsoft pour permettre à l'organisation de configurer un serveur proxy sur le sytème, c'est-à-dire effectuer automatiquement le paramétrage d'accès à l'internet de son navigateur [10]–[12]. Nous détectons aussi des requêtes pour résoudre le domaine "wpad.home" (FSI Proximus) renvoyant également une erreur du même type que la précédente.

## IV. Traversée des NAT

Lorsque IPv4 est utilisé, l'application Messenger utilise le protocole STUN [13] pour traverser les NAT. Nous n'allons pas entrer dans les détails du fonctionnement de ce protocole. Il est particulièrement utilisé dans les applications qui utilisent la voix/vidéo (ce qui est donc le cas de Messenger). Tout d'abord, il est à noter que nous ne retrouvons des paquets STUN que lorsque nous lançons un appel vocal ou un appel vidéo. Le protocole STUN fonctionne en générale sur le port UDP 3478. Cependant, nous observons dans les captures qu'un petit nombre de paquets STUN sont sur le port TCP 3479 [14]. Ce port est semble-il lié à l'entreprise 2Wire, Inc. [15], [16] qui fournit aux entreprises de télécommunications du matériel, des logiciels, des plateformes de services et des systèmes de gestion à distance des CPE [17].

## V. Couche transport

Dans cette section, nous allons nous intéresser aux deux principaux protocoles de transport, à savoir UDP et TCP, ainsi qu'aux fonctionnalités permettant de régir chacun d'eux. Nous allons également aborder le protocole QUIC.

### A. UDP

UDP (User Datagram Protocol) est un protocole de transport qui a la particularité de fonctionner sans connexion. On le retrouve principalement lors d'un appel vidéo et audio. Il est également utilisé lors des requêtes DNS, mais aussi sous le protocole QUIC. Il est intéressant de noter que nous observons également le protocole SSDP qui est utilisé au dessus de UDP. SSDP (Simple Service Discovery Protocol) est un protocole qui décrit comment les services sont découverts dans le réseau. [18]

### B. QUIC

QUIC est un nouveau protocole publié par Google, permettant l'envoi rapide de paquets via le protocole UDP, en réduisant la latence. Avec QUIC, le chiffrement est obligatoire. Une connexion peut permettre plusieurs streams (parallélisme). Ce protocole n'est pas observé lors de l'analyse via Wireshark de l'application bureau Messenger. Cependant,

lorsqu'on analyse les paquets échangés via le site www.messenger.com, on observe bien du traffic QUIC, version 1. A l'aide de Wireshark nous pouvons faire une analyse plus détaillée du traffic QUIC.[1] En analysant le premier paquet QUIC envoyé, nous pouvons identifier plusieurs paramètres spécifiques à QUIC, à l'intérieur d'une extension TLS (il s'agit de l'extension quic_transport_parameters). Voici une liste non-exhaustive des extensions négociées dans le handshake:

- version_information
- initial_max_streams_uni: 16
- max_idle_timeout: 30000 ms
- initial_max_streams_bidi: 16
- initial_max_data: 25165824
- initial_max_stream_data_uni: 1048576

Cependant, étant donné que nous n'avons pas la clé de cryptage, nous ne pouvons pas voir la réponse du serveur. Toutes les données sont cryptés (hormis le "Connection ID" et la longueur du paquet), et nous n'avons accès qu'au "QUIC Short Header". [19]–[22]

### C. TCP

TCP (Transmission Control Protocol) est un protocole fournissant un transport des données fiable et séquentiel (l'ordre des paquets est conservé). De plus, il assure une détection et une correction des erreurs de transmission. On le retrouve principalement lors de l'ouverture de l'application Messenger, mais également lors de l'échange de messages et de fichiers.

*1) Etablissement de la connexion:* L'initialisation de la connexion TCP fonctionne via le processus "Three-Way Handshake". Durant cette phase, plusieurs paramètres clés sont négociés. Tout d'abord, les deux hôtes négocient les numéros de séquence initiaux dans chaque direction de la connexion.[2] Ensuite, ils négocient le Maximum Segment Size (MSS), variant de 1392 à 1440; la taille de la fenêtre, 256 (S=8) et l'option SACK (Selective ACKnowledgement), ici autorisée pour chacune des connexions. Cette dernière permet à l'émetteur de ne retransmettre que les segments manquants et non tous les segments précédents lors d'une erreur de transmision.

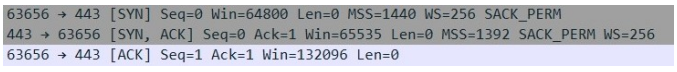


Fig. 2. Etablissement de la connexion

On remarque également la présence de l'option No-Operation (NOP). Celle-ci est utilisée entre les options, afin d'aligner le début d'une option suivante sur une frontière de 32 bits. [23].

[1]Nous analysons pour cette sous-section des captures réalisées lors d'échanges de messages sur le site www.messenger.com, et non plus sur l'application bureau Messenger comme précédemment.

[2]A noter que ceux-ci sont aléatoires, établis sur base d'une horloge de chaque hôte ainsi que sur la valeur hashée de plusieurs paramètres, dont un paramètre secret et non dévoilé au public, et ce pour des raisons de sécurité.

*2) Transfert des données:* Lors du tranfert de données, nous observons plusieurs segments TCP marqués par Wireshark avec "TCP segment of a reassembled PDU" (voir Figure 3). Il s'agit de segments correspondant à une partie du message initial, les autres parties se trouvant dans des segments ultérieurs. Cela se produit lorsque le message initial est trop long, ici lors de l'envoi d'un fichier (taille supérieure à MSS ayant été négotiée lors de l'ouverture de connexion) et doit être découpé sur plusieurs segments.




Fig. 3. Tranfert de données d'un message initial trop long

*3) Terminaison de la connexion:* Le client et le serveur utilisent généralement le flag FIN pour fermer la connexion. Cependant, nous remarquons que dans certains cas, le client renvoie un paquet avec le flag RST après que la connexion aie été fermée avec le flag FIN (voir Figure 4). Cela peut se justifier par le fait que le client ne veut pas attendre 2 fois le MSL (Maximum Segment Lifetime) et devoir retransmettre des ACK si un segment est reçu de la part de l'autre destinataire de la connexion. [24]

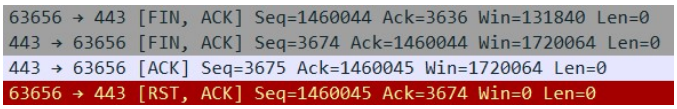


Fig. 4. Fermeture de connexion

## VI. Chiffrement et Sécurité

### A. Sécurité du DNS

Les requêtes DNS ne semblent pas directement sécurisées par DNSSEC. Cependant, Azur DNS fournit une offre d'hébergement (et de sécurité) pour les domaines DNS [25] et l'entreprise Akamai fournit également un service nomé Edge DNS qui sécurise le traffic DNS [26]. Voir ces deux entreprises apparaitre dans la section sur le trafic DNS n'est donc pas étonnant.

### B. Sécurité de UDP

A première vue, le trafic UDP ne semble pas chiffré. Nous ne détectons pas non plus de quelconques moyens de sécurisation. Cependant, l'entreprise Akamai Technologies, Inc. fournit un service nommé "Prolexic" qui empêche les attaques par déni de service (DDoS) contre le traffic UDP [27]. Microsoft et Meta étant client de cette entreprise, nous pouvons supposer (sans réelle certitude) que ce service s'applique à l'application Messenger.

### C. Versions de TLS

Le protocole de transport sécurisé par TLS est en majorité le protocole TCP. Remarquons que les en-têtes pour le protocole TLS annoncent dans la majorité des cas TLSv1.3 ou TLSv1.2. Cependant, les versions utilisées dans les records TLS sont les versions 1.0 (minoritairement, uniquement dans les records "Client Hello") et 1.2.

### D. Certificats

Le tableau suivant reprend les diverses informations récoltées au travers des captures concernant les certificats.

| Sujet | Emetteur | Validité |
|---|---|---|
| F-Scecure Corporation | DigiCert Inc. | ± 1 an |
| DigiCert Inc | DigiCert Inc. | ± 10 ans |
| WithSecure Corporation | Amazon | ± 1 an |
| Amazon | Amazon | ± 8 ans |
| Amazon | Starfield Technologies, Inc. | ± 22 ans |
| Starfield Technologies, Inc. | Starfield Technologies, Inc. | ± 25 ans |
| Microsoft Corporation | Microsoft Corporation | ± 1 an |
| Microsoft Corporation | DigiCert Inc. | ± 4 ans |
| Microsoft Corporation | Microsoft Corporation | ± 1 an |
| Microsoft Corporation | Microsoft Corporation | ± 15 ans |
| Microsoft Corporation | Cyber Trust | ± 4 ans |

Remrquons pour commencer qu'il n'y a aucun certificat en lien avec l'entreprise Meta, propriétaire de l'application Messenger. Parmi les entreprises présentes dans le tableau, on retrouve diverses entreprises spécialisées dans la cybersécurité telles que DigiCert Inc. [28], Cyber Trust [29] ou encore Starfield Technologies, inc. [30]. Microsoft Corporation est présent dans divers cas et semble pouvoir émettre ses propres certificats. Notons également la présence de l'entreprise Amazon. Remarquons pour finir une chose étonnante, la présence de l'entreprise WithSecure Corporation [31] anciennenement F-Secure Corporation [32], une entreprise finlandaise. Sur le navigateur internet de l'une des machines utilisées pour faire les captures de paquets, se trouvait une extension pour protéger la navigation chiffrée avec les produits de sécurité F-Secure. La seule application ouverte durant la capture étant Messenger, ceci peut paraître étrange. En réalité, le certificat se trouve dans une capture effectuée lors d'un appel vidéo. Or, il se trouve que les appels vidéo ouvre une fenêtre du navigateur internet présent sur la machine (Google Chrome) afin de pouvoir afficher la vidéo... Ce n'est donc pas surpenant de voir apparaître ces entreprises dans les informations sur les certificats.

### E. Chiffrement

Le protocole utilisé pour l'échange de clés de cryptage pré-partagées (PSK) est le protocole (EC)DHE (Elliptic curve Diffie–Hellman) [33].
Lors des connexions TCP, divers algorithmes de cryptage sont utilisés. Citons (par ordre décroissant d'utilisation) :

- TLS_AES_256_GCM_SHA384
- TLS_CHACHA20_POLY1305_SHA256
- TLS_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256

Le premier algorithme ainsi que le troisième sont des algorithmes de chiffrement acceptables pour TLSv1.3 et les versions antérieures. Les autres algorithmes étant globalement acceptables pour TLSv1.2 et les versions antérieures [34].

## VII. Couche application

### A. Application Data Protocol

Nous observons que différents protocoles sont utilisés. Nous retrouvons principalement HTTPS, mais également HTTP2 et MQTT présent en faible nombre. MQTT est un protocole léger permettant une transmission efficace et rapide des données. Facebook utilise pour son application Messenger le protocole MQTT car ce dernier permet aux messages d'être livré efficacement en quelques millisecondes, malgré des connexions Internet incohérentes à travers le monde. Nous le retrouvons cependant dans seulement quelques paquets. [35], [36]

### B. Conversation VS Appels

La différence principale observée entre une conversation comparée à un appel se situe dans le protocole de transport utilisé. UDP est principalement utilisé lors d'un appel (tant audio que vidéo), tandis que TCP est principalement utilisé lors d'une conversation Messenger.
La différence entre appel audio et vidéo se situe dans le débit de données échangées. Comme nous pouvons le voir à la Figure 5, le débit de paquets échangés pendant un appel vidéo est significativement plus grand que celui lors d'un appel audio.[3]

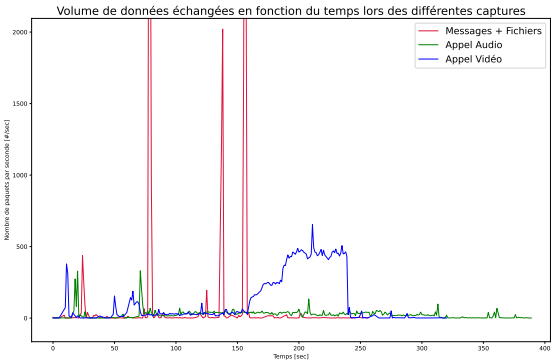


Fig. 5. Volume de données échangées par seconde pour les différentes fonctionnalités testées

## VIII. Conclusion

En conclusion, cette analyse des protocoles utilisés par l'application Messenger nous a permis de mettre en lumière des éléments intéressants sur son fonctionnement. L'étude DNS nous a nottament montré l'influence du fournisseur d'accès internet sur le nombre de serveurs autoritatifs, tandis que l'analyse TCP a révélé la présence du flag RST dans certaines connexions, ce qui est assez surprenant. Nous avons également constaté l'absence de sécurisation directe des protocoles DNS et UDP, probablement gérée par une entreprise externe. Pour finir, nous avons pu observer un lien fort entre l'OS utilisé et l'activité résau de l'application.

[3]On remarque 3 pics importants pour la capture "Messages + Fichiers", correspondant à l'envoi d'un fichier, d'une photo, et de ce même fichier.

## REFERENCES


[1] https://fr.wikipedia.org/wiki/Microsoft
[2] https://en.wikipedia.org/wiki/Meta_Platforms
[3] https://www.eset.com/be-fr/
[4] https://www.dnb.com/business-directory/company-profiles.amazon_technologies_inc.355f208edfe05525a9c3cee651bd82e4.html
[5] https://fr.wikipedia.org/wiki/Akamai_Technologies
[6] https://www.akamai.com/fr
[7] https://learn.microsoft.com/fr-fr/azure/dns/dns-overview
[8] https://domain.glass/edge-stun.facebook.com
[9] https://dev.to/aprogrammer22/list-of-free-stun-and-turn-servers-open-relay-project-3a70
[10] https://answers.microsoft.com/en-us/windows/forum/all/what-is-wpadhome-repeating-messages/c62bebfb-d992-41fc-a8ab-2e8ad91ac63c
[11] https://irp.nain-t.net/doku.php/220squid:050_wpad
[12] https://www.howtogeek.com/298460/disable-wpad-in-windows-to-stay-safe-on-public-wi-fi-networks/
[13] https://www.frameip.com/entete-stun/
[14] https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml?search=3&page=27
[15] https://pitchbook.com/profiles/company/13211-74#overview
[16] https://www.adminsub.net/tcp-udp-port-finder/3479
[17] https://en.wikipedia.org/wiki/2Wire
[18] https://archipel.uqam.ca/1499/1/M10572.pdf
[19] https://www.ionos.fr/digitalguide/hebergement/aspects-techniques/quic/
[20] https://dial.uclouvain.be/pr/boreal/object/boreal%3A242063/datastream/PDF_01/view
[21] https://www.bortzmeyer.org/quic-demo.html
[22] https://www.youtube.com/watch?v=jQ1GCkhwGTg
[23] https://www.rfc-editor.org/rfc/rfc791
[24] https://www.rfc-editor.org/rfc/rfc793
[25] https://learn.microsoft.com/fr-fr/azure/dns/dns-overview
[26] https://www.akamai.com/fr/products/edge-dns
[27] https://www.akamai.com/fr/glossary/what-is-udp-flood-ddos-attack
[28] https://fr.wikipedia.org/wiki/DigiCert
[29] https://cyber-trust.eu/
[30] https://www.starfieldtech.com/
[31] https://www.withsecure.com/fr/home
[32] https://fr.wikipedia.org/wiki/F-Secure
[33] https://fr.wikipedia.org/wiki/%C3%89change_de_cl%C3%A9s_Diffie-Hellman_bas%C3%A9_sur_les_courbes_elliptiques
[34] https://www.ssl.com/fr/guide/tls-conformit%C3%A9-aux-normes/
[35] https://fr.linkedin.com/pulse/mqtt-expliqu%C3%A9-en-fran%C3%A7ais-rapha%C3%ABl-delstanche
[36] https://fr.wikipedia.org/wiki/MQTT
[37] Codes utilisés lors des analyses https://github.com/Hughlindien/LINFO1341_P1_V2