**UNIVERSITÉ DU QUÉBEC À MONTRÉAL**

# TRACES DE VIE PRIVÉE CONSÉCUTIVES À L'UTILISATION DE L'ASSISTANT NUMÉRIQUE PERSONNEL CORTANA

MÉMOIRE

PRÉSENTÉ

COMME EXIGENCE PARTIELLE

DE LA MAÎTRISE EN INFORMATIQUE

DE L'UNIVERSITÉ DU QUÉBEC À MONTRÉAL

PAR

SYLVAIN DESHARNAIS

14 DÉCEMBRE 2020

# REMERCIEMENTS

Ce mémoire est le résultat de quatre années d'efforts couronnés de succès. Mais ces efforts ne sont pas que les efforts du rédacteur du présent mémoire. Sans le soutien de mon entourage familial et de mon entourage académique, le projet n'aurait peut-être pas vu le jour et n'aurait certainement pas été aussi agréable.

Merci à Christine, mon épouse : elle rend ma vie (incluant ma vie académique, toutes universités confondues), paisible et sereine. Elle a été l'auditrice attentive de mes projets de session et de recherche, de leurs péripéties et de leurs dénouements. Elle a surtout et toujours rendu ma vie de tous les jours faciles à vivre. Merci à Brigitte, Judith et Gabriel, mes enfants. Ils sont montés sur mes épaules et ont vu les choses de la vie à leurs façons. Ils m'ont souvent vu étudier, depuis toujours, et en ont appris la valeur de la connaissance comme outil d'accomplissement.

Merci aux gens de l'Agence du revenu du Canada : André Lamonde, Louis St-Laurent, Mario Mainville et Johanne Charbonneau, qui m'ont permis de découvrir ma passion pour la recherche, l'innovation et l'enseignement. Merci à Rola Tabaja, Linda Olmstead et Alain Cole pour leur support pendant des années.

Merci à Thomas Maillet et Antoinee Alriquet, élèves-officiers de la Marine française qui m'ont supporté dans mes recherches lors de l'automne 2019. Merci à mes collègues Henri Pineault et Antoine Laurent pour leur support. Merci à Rosin Ngueveu pour sa chaleur, son amitié et son support. Merci à Éric Beaudry qui a été le premier à me donner un « break » en programmation pour me permettre de réussir dans une matière que je trouve difficile. Merci à Marie-Jean Meurs dont l'enthousiasme à l'égard de mes projets me motive et me pousse vers l'avant.

Merci à Sébastien Gambs pour sa direction, sa disponibilité, sa gentillesse et ses commentaires. Il a su mettre en valeur mes forces. Mais surtout, merci pour sa patience à l'égard de mes besoins particuliers. Pendant ces quatre années, j'ai pu continuer à enseigner et mener à bien cette maîtrise. Je formule le vœu qu'il ne change pas!

# DÉDICACE

« *Mais ces pères, anxieux de n'être pas assez aimés et respectés, sont eux aussi condamnés à la douleur, parce qu'ils ignorent que leurs fils doivent grimper sur leurs épaules, pour voir plus loin qu'eux ou mieux qu'eux ou même ailleurs; en tout cas, pour regarder les choses de la vie différemment.* »
Chocana Boukhobza
En postface de
Franz Kafka, *Le verdict*

Cet ouvrage est dédié à tous ceux qui m'ont permis de monter sur leurs épaules et de voir les choses de la vie à ma façon.

# AVANT-PROPOS

De décembre 1999 à avril 2013, j'ai œuvré à titre d'enquêteur régional en informatique, régions Maritimes et Québec (et parfois celle du Nord de l'Ontario). À ce titre, je devais investiguer les cas complexes de fraudes fiscales et j'avais la tâche de faire de la recherche en investigation numérique et de former les enquêteurs informatiques de l'Agence. J'ai aussi travaillé comme investigateur numérique privé de mai 2013 à novembre 2019.

Un des effets inattendus de ces vingt années de travail passionné a été de développer un amour et un intérêt intenses à l'égard du sujet « vie privée ». Nul mieux que moi sait à quel point un ordinateur peut être indiscret, éloquent, loquace et même, disons-le, bavard, logorrhéique et prolixe sur la vie privée de son utilisateur.

Les assistants numériques personnels sont des ordinateurs sur le qui-vive, n'attendant qu'un mot-clé pour s'activer et refléter les intérêts, les questionnements et les états d'esprits des utilisateurs. L'empreinte qu'ils laissent sur l'ordinateur qui les supporte est large et profonde. Bien que ce projet vise à alerter les utilisateurs soucieux de leurs vies privées, les responsables de la sécurité informatique et les agents de l'État y trouveront sans doute des connaissances qui leur permettront de débusquer malandrins et délinquants. Et c'est bien! Elle alertera aussi les agents de l'État chargés de s'immiscer dans la vie de citoyens honnêtes et les pirates informatiques. Et c'est mal!

À tout événement, mieux vaux connaître qu'ignorer.

Le protocole expérimental se divisait en trois vecteurs d'étude : les échanges entre la machine locale (celle exécutant le programme d'assistant numérique personnel) et Internet (dont le serveur de reconnaissance vocale), les événements persistants présents dans la mémoire vive à la fin de la requête vocale, et les fichiers créés sur la machine locale en réponse à la requête. Une liste de requête vocale a été dressée pour étudier l'impact de l'utilisation des assistants numériques personnels. Ces requêtes étaient de l'un ou l'autre des types suivants (que nous expliquerons plus avant dans le chapitre réservé au protocole expérimental) : appel à des utilitaires locaux, renseignements courants sur Internet, recherche de renseignements locaux, test d'intelligence ou de discrimination, requête « administrative ».

Au départ, ce projet devait couvrir tous les types d'assistants numériques. L'ampleur des traces laissées par Cortana, les embûches techniques ainsi que la difficulté d'accéder à ces traces ont forcé la réduction des ambitions de recherche. Notamment, il a fallu gérer les nouvelles particularités de la mémoire vive de Windows 10. Étudier en profondeur Cortana a donc été préféré à survoler tous les assistants numériques personnels.

La contribution de cette recherche au champ de la forensique informatique est donc de dresser un inventaire méticuleux des traces de vie privée consécutives à l'utilisation de l'assistant numérique personnel Cortana de Microsoft installé sur un ordinateur portable alors que la plupart des articles couverts dans la revue de littérature portent sur une partie seulement de ces traces.

# TABLE DES MATIÈRES

# LISTE DES FIGURES

# LISTE DES TABLEAUX

# RÉSUMÉ

Les assistants numériques personnels sont des appareils toujours en éveil, à l'affût de la prononciation du mot-gâchette. Ainsi, Microsoft Windows Cortana attend le mot-gâchette « Hey Cortana ». Les mots de la requête sont enregistrés et la version sonore est envoyée sur un serveur distant qui reconnaîtra les mots qui la compose et concoctera la version texte d'une requête Internet, la soumettra au moteur de recherche Bing et retournera le résultat à l'utilisateur.

Cette requête et ce qui s'ensuit génère de nombreuses traces dans le nuage mais surtout sur le média à partir duquel on exécute Windows et le Cortana de l'utilisateur. Il est possible de récupérer l'audio des huit dernières requêtes soumises à Cortana ainsi que le texte de toutes les requêtes depuis que l'ordinateur est en fonction, ainsi que plusieurs informations quant à la date-heure à laquelle la requête a été soumise. Ces informations sont des traces de la vie privée de l'utilisateur laissées dans l'écosystème numérique du Cortana de l'utilisateur.

Mots-clés : Cortana, assistant numérique personnel, forensique, traces, écosystème numérique, vie privée.

Abstract

Personal digital assistants are devices that are always « awake », expecting for the trigger-word to be pronounced, like « Hey Cortana » for Microsoft Windows Cortana. The words uttered by the user while submitting a request through Cortana are recorded and sent to a distant server that works as a speech-to-text device, creates a Bing request from the recognized text and send back the result to the user.

The request and what follow generate many traces in the cloud and many one on the media running Windows and the user's Cortana. It is easy to recover the eight-last request submitted to Cortana and the text content of all request made through Cortana since the computer is in service. Many timestamps of these request are also recoverable. These data are privacy data traces left in the user's Cortana digital ecosystem.

Keywords: Cortana, personal digital assistant, forensics, traces, digital ecosystem, privacy.

# INTRODUCTION

## 1. Généralités

### 1.1. Définition d'un assistant numérique personnel

Un assistant numérique personnel est un logiciel dont la fonction est de capter les requêtes verbales de l'utilisateur et d'y répondre. Ce logiciel est donc en mesure d'écouter pour capter la requête au moment où celle-ci est énoncée, d'en comprendre le contenu et de fournir une réponse appropriée.

### 1.2. Sujet de la recherche

Dans l'arrêt de la Cour suprême du Canada R. c. Morelli [2010], le juge Fish a écrit « *il est difficile d'imaginer une atteinte plus grave à la vie privée d'une personne que la perquisition de son domicile et la fouille de son ordinateur personnel. En effet, nos ordinateurs contiennent souvent notre correspondance la plus intime… notre situation financière, médicale et personnelle... nos intérêts particuliers, préférences et propensions… tout ce que nous recherchons, lisons, regardons ou écoutons dans l'Internet* ».

Cette recherche vise à démontrer à quel point le juge Fish avait raison d'énoncer une telle généralisation. Car les assistants numériques personnels, comme tout autre ordinateur, contient une nuée de données desquelles on peut tirer ou en inférer des informations de nature privée. Mais cette recherche démontre surtout à quel point nous sommes collectivement inconscients de notre vulnérabilité : Comment un geste aussi banal que de poser une question naïve à une oreille toujours attentive et disponible peut-il laisser des marques aussi nombreuses et profondes dans l'écosystème numérique d'un utilisateur. Quelles sont ces marques? Où se logent-elles?

Afin de retrouver ces traces, nous avons établi une procédure expérimentale comportant le passage de requêtes aux assistants numériques personnels choisis, d'intercepter les flux de communications dirigés vers Internet et de recueillir la mémoire vive de l'ordinateur. À la fin de la journée, nous avons fait une copie forensique bit-à-bit du contenu du disque dur de l'ordinateur.

Cette approche était résolument calquée sur la procédure suivie en investigation numérique (dont nous couvrons les notions de base au chapitre 1 ci-après) pour retrouver les preuves numériques

laissées par ceux sur qui ils ont la charge d'enquêter. Ceci nous a permis de retrouver le plus grand nombre possible de traces de vie privée laissées par nos requêtes.

### 1.3. Motivations

Connaître l'endroit où ces indices se logent, c'est permettre aux personnes voulant protéger leur vie privée de le faire en leur indiquant quoi (leur nature et leur localisation) supprimer de leur appareil. D'un autre côté, c'est aussi permettre aux spécialistes en sécurité informatique et en investigation numérique de repérer ces indices et de les utiliser contre les malfaisants. Un impact indirect de tout ceci serait de sensibiliser les utilisateurs aux dangers auxquels ils exposent leur vie privée lorsqu'ils utilisent leurs assistants numériques personnels.

### 1.4. Motivations secondaires

Accessoirement, cette recherche vise à déterminer le contenu privé se logeant dans la mémoire vive de l'ordinateur à partir duquel on passe les requêtes et les transmissions Internet sortantes et entrantes générées par la requête de l'utilisateur.

### 1.5. Étendue du phénomène « assistant numérique personnel »

Le logiciel « assistant numérique personnel » peut être installé sur un ordinateur standard, sur un appareil dédié à cette fin, sur un cellulaire ou dans le nuage informatique.

Microsoft Cortana ou Apple Siri sont des exemples d'assistants numériques personnels installés par défaut sur un ordinateur standard et mis gratuitement à la disposition du public. Mais d'autres sont disponibles par le biais d'un téléchargement, par exemple Mycroft (Voir *Mycroft [2020]*) qui peut être installé sur un Raspberry Pi ou sur un Linux et bientôt sur un Windows et un Mac OS.

Amazon Alexa et Google Home Mini sont des exemples d'assistants numériques personnels installés sur des appareils dédiés à cette fin. D'autres nouveaux venus tentent aussi de s'inscrire dans ce marché, comme Mycroft qui vend deux modèles pour moins de 200$ US. Sans compter les fabricants d'appareils (généralement un haut-parleur) qui achètent le logiciel d'un concurrent pour l'installer sur leur propre appareil et le personnaliser (Voir le cas de Harman Kardon Invoke avec *Ash [2017]*).

Apple Siri ou Samsung Bixby sont des exemples d'assistants numériques personnels fonctionnant sur des cellulaires. Là aussi Mycroft (encore eux) offrent gratuitement leur assistant numérique personnel version Android. Dans le nuage, Oracle vend les services de son « Digital assistant » (Voir *OracleDA SaaS*) pour 6 cents (canadiens!). Mycroft, de son côté, offre d'harnacher leur assistant numérique personnel sur Docker.

### 1.6. Portée de la recherche

Le projet de ce mémoire porte sur le repérage des traces laissées dans l'écosystème de l'assistant numérique personnel Cortana lorsque l'utilisateur énonce une requête verbale. Les traces en question se rapportent notamment à ce qui est inscrit dans la mémoire vive et le disque dur de l'ordinateur exécutant l'assistant numérique personnel, ce qui est émis vers Internet puis retourné à l'assistant numérique personnel suite à une requête de l'utilisateur et, enfin, ce qui est inscrit dans le nuage de façon plus ou moins permanente.

Afin de circonscrire la portée de la recherche, seuls les éléments retrouvables à l'aide de logiciels de forensique ou de sécurité ont été considérés, excluant donc les phénomènes parallèles à l'utilisation des assistants numériques personnels comme la revente de données personnelles par les fabricants de l'assistant numérique personnel, ou le fabricant du logiciel d'icelui à des tiers.

## 2. Problématique

### 2.1. Introduction

Dans son premier livre blanc sur les assistants vocaux, CNIL [2020] présente les étapes habituelles de fonctionnement des assistants numériques personnels, ce qui permet de dégager leurs principales caractéristiques opérationnelles. La figure Intro-1 reproduit une illustration de ce document de la CNIL montrant le chemin parcouru par les données dans le traitement d'une requête énoncée par l'utilisateur à l'endroit d'un assistant numérique personnel.

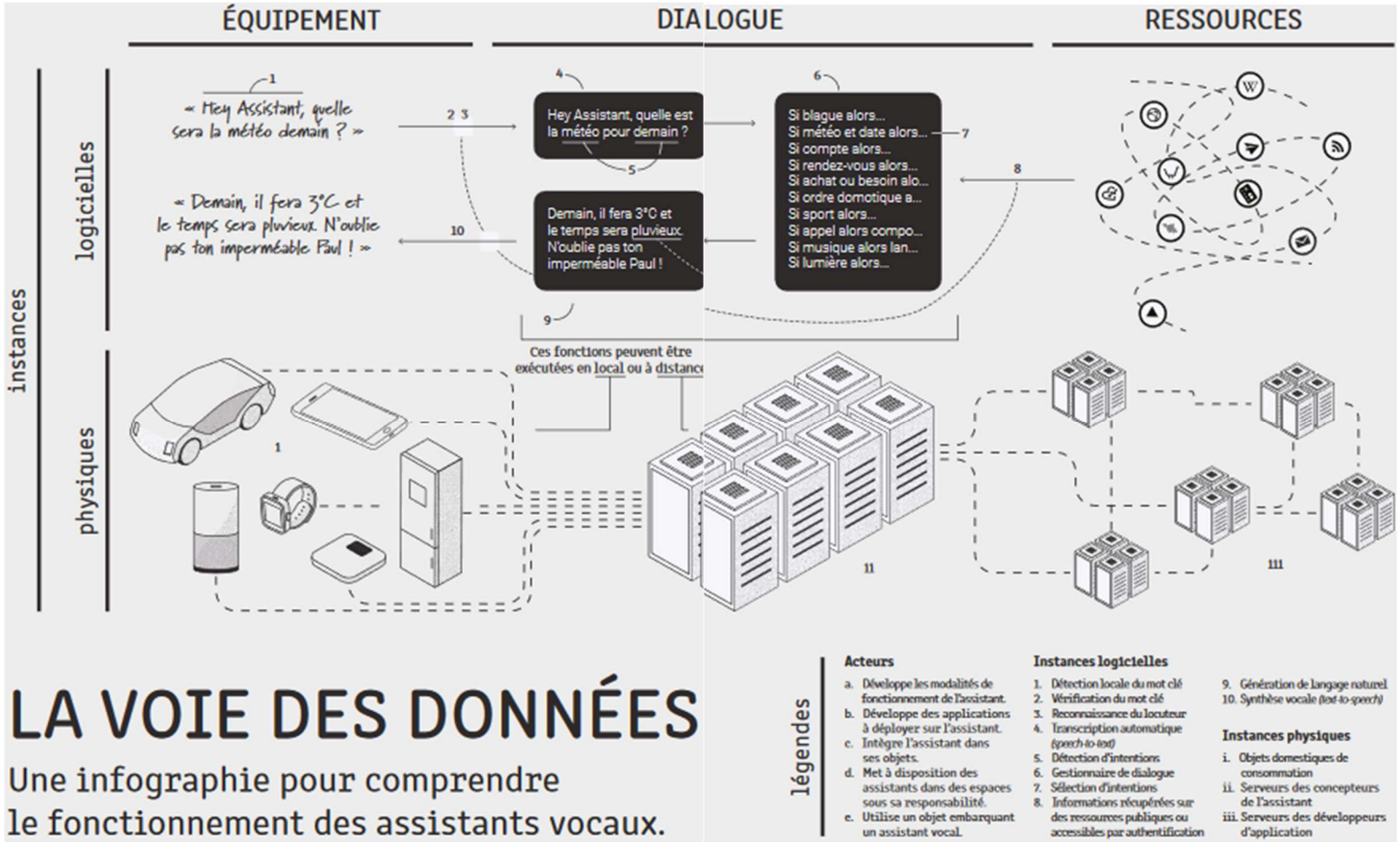



Figure Intro-1 – Chemin suivi par les données lors d'une requête énoncée par un utilisateur auprès d'un assistant numérique personnel (Tiré de CNIL [2020]

Les assistants numériques personnels sont donc constamment à l'écoute et attendent que soit prononcé le mot-gâchette (Hey Cortana, Dis Siri, Alexa, OK Google…). La reconnaissance vocale, de par sa voracité en ressources matérielles (surtout en quantité de mémoire et en vitesse de processeur) contraint les fabricants à minimiser le traitement local de la requête. Pour cette raison cette reconnaissance de la parole se fera sur un site distant.

Les mots reconnus sont alors soumis à des algorithmes afin d'interpréter le sens de la requête et de soumettre une requête textuelle sensée de recherche d'information auprès d'Internet. C'est le résultat de cette requête qui est enfin retournée à l'utilisateur.

Mis à part ces caractéristiques liées au traitement, typiquement les assistants numériques personnels adoptent un aspect discret. Tout au plus une icône sur un écran d'ordinateur ou la forme d'un haut-parleur. Lorsqu'en plus ils sont installés dans des endroits privés (salon, cuisine, chambre à coucher), ils représentent un potentiel de fuite de vie privée plus que certains. Ce sont ces caractéristiques qui rendent les assistants numériques personnels si apte à capter et à générer une mémorisation des aspects de la vie privée présents dans le périmètre qu'ils couvrent.

## 2.2. Mise en contexte

### 2.2.1. Statistiques

Voici quelques statistiques sur la prolifération des assistants numériques personnels incluant Cortana et bien d'autres.

- StatsCan [2019] révèle qu'en 2018 :
  - 91% des canadiens (et 71% des aînés) ont accès à Internet;
  - Le Québec est en queue de peloton (88%) et l'Alberta en tête (94.1%)
- Au Québec, CEFRIO [2020] a publié une étude statistique le 3 juin 2020 où il affirme que :
  - 93% des québécois avaient un accès résidentiel à Internet;
  - 75% des adultes ont un téléphone intelligent;
  - 56% ont une ardoise tactile (une « tablette »)
  - 15% des jeunes de 6 à 17 ans utilisent un assistant vocal;

- 13% des adultes québécois possèdent un appareil connecté destiné au divertissement comme un assistant vocal, un drone ou un robot
- Le nombre de ménages équipés d'un assistant vocal a augmenté de 4% en une année;

- Voicebot [2019] indique :
  - Parmi les utilisateurs d'assistants numériques, 19% utilisent Cortana, 36% Siri, 36% Google et 25% Alexa;
  - 52% des gens interrogés craignent que leurs données ne soient pas en sécurité, 36% ne veulent pas que leurs données personnelles soient utilisées et ça dérange 41% des gens d'être constamment écoutés. Mais seulement 14% ne font pas confiance aux fabricants d'assistants numériques;
- Selon Quoracreative [2020] :
  - Faire l'épicerie par voix constitue plus de 20% du magasinage par voix;
  - Le magasinage par voix dépassera 40 milliards de dollars en 2022;
  - Les trois mots-clés les plus utilisés sont comment (8.6%), quoi (5%) et meilleur (2.6%);
  - Le taux de réussite d'une requête par voix est de 25% pour Cortana et 52% pour Siri;

#### 2.2.2. Anecdotes sur la sécurité et la vie privée liées aux assistants numériques personnels

Dans un article de journal, Johnson [2019] rapporte son évaluation du nouveau (à cette époque) Google Nest Hub. Il teste cet appareil dans le salon, la cuisine, le bureau et la chambre à coucher. Pour chacune des pièces, l'auteur trouve des qualités et des défauts à cet appareil. Il souligne au passage l'absence de caméra de l'appareil en disant « …*une caméra aurait assurément attiré son lot de détracteurs, qui ne sont déjà pas à l'aise avec l'idée de truffer leur maison de microphones* » et finit par conclure que « *C'est finalement dans la chambre à coucher que le Nest Hub a trouvé sa pièce d'adoption* ». On peut s'interroger sur cette apparence de contradiction…

Plusieurs incidents rapportés par la presse mettent en lumière que les assistants numériques personnels peuvent s'activer par eux-mêmes, sans que le mot gâchette n'ait été prononcé. Lynksey [2019] relate l'anecdote d'un ex-employé d'Amazon rentrant chez lui pour entendre son Echo Dot qui, sans avoir été activé par le mot-gâchette, régurgitait les requêtes passées la veille. Latoya [2019] présente la mésaventure d'une mère recevant par UPS deux boîtes contenant 400$ de jouets commandés par sa fille de 6 ans via Alexa qui était couplé avec la carte de crédit familiale. Ogden

[2018] parle du cas d'un couple de l'Oregon dont l'assistant numérique personnel a cru entendre le mot-gâchette, a commencé à enregistrer ce qui se disait dans la pièce, a cru entendre la commande pour envoyer l'enregistrement, a cru entendre le nom d'une collègue de travail et, selon Amazon, a cru entendre la confirmation d'envoyer le message. Fort heureusement, la conversation portait sur les planchers de bois-franc.

Dans un ordre d'idées similaires, on rapporte des affaires judiciaires où des enquêteurs de police, sachant que les assistants numériques personnels sont sujets à des déclenchements d'enregistrement inopinés, ont demandé aux tribunaux la permission d'accéder aux requêtes verbales de certains suspects ou de certaines victimes. Rodriguez [2019] rapporte une affaire policière de Floride où un homme violent a planté une lance dans le corps de sa concubine lors d'une dispute. La police a demandé (et obtenu après une longue saga judiciaire) qu'Amazon leur remette les derniers enregistrements faits par Alexa. Dangerfield [2018] raconte une histoire semblable survenue au New Hampshire où un homme est accusé du meurtre de deux femmes en 2018.

Dans un article rapportant une recherche, Dubois et al [2020] rapportent une expérience où ils ont fait écouter à 11 assistants numériques personnels (de type haut-parleur) 134 heures d'émissions de télé de toutes sortes. Invoke/Cortana a été celui qui a généré le plus d'activations inopinées avec 54 et une moyenne de 33,2 activations par appareil. Ceci représente une activation pour chaque 4 heures d'écoute. Comme nous le verrons à la section 3.4 du chapitre 6 ci-dessous, les enregistrements résultant de ces activations inopinées ont une durée de 5 à 6 secondes mais sont intelligibles.

#### 2.2.3. Pertinence pratique et scientifique

Les anecdotes présentées dans la section précédente se rapportent toutes au déclenchement inopiné d'enregistrement. Le côté un peu inouï de ces événements en font un matériau journalistique attrayant. Du point de vue forensique, ces enregistrements ne sont pas négligeables, certes. Mais ceci n'est que la partie visible de l'iceberg et le domaine de la forensique se doit de s'attarder aussi aux traces de vie privée qui ne sont pas des enregistrements inopinés. La question qui se pose est donc : Hormis les enregistrements inopinés, quelles traces laissons-nous dans notre écosystème numérique lorsque nous utilisons un assistant numérique personnel comme Cortana sur un ordinateur?

Cette question est celle que tout investigateur numérique, tout testeur de pénétration (« pentesters »), tout officier de sécurité et tout pirate se pose. Cette question doit bien sûr être répondue d'une manière irrévocable, c'est-à-dire en se basant sur une approche scientifique reposant sur des procédés ayant été empiriquement prouvés comme fournissant des preuves fiables, authentiques et crédibles à savoir, des procédés forensiques.

Si nous avions une connaissance parfaite de ces traces, les malfaisants seraient à coup sûr découverts et punis. Les brèches de sécurités seraient rapidement colmatées. Et les pirates se régaleraient de leurs victimes... tant et aussi longtemps que les investigateurs numériques ne les attraperaient pas et que les brèches de sécurité ne seraient pas colmatées.

Quelle est la pertinence pratique? Simplement ceci : Savoir, c'est pouvoir. Pouvoir réagir aux intrusions dans la vie privée des utilisateurs. Pouvoir sécuriser l'écosystème numérique de l'utilisateur pour lui procurer une protection solide de son périmètre intime.

L'informatique forensique est une des spécialités de l'informatique au même titre que la programmation, la sécurité ou la gestion d'un parc informatique. Elle requiert des connaissances spécifiques à la spécialité (propriétés des preuves numériques, balances de fichiers, signatures, paradonnées, intradonnées), implique des procédures qui lui sont particulières (aseptisation, copie bit-à-bit, clone) et utilise des outils qui lui sont propres (stratégies de fouille, FTK Imager, Autopsy Sleuth Kit, WinHex). Cette spécialité est fondée sur une approche scientifique lorsqu'il s'agit de développer ses outils, ses techniques et ses procédures. Parce qu'elle a été soumis à l'œil critique et sévère des tribunaux (avocats de la Couronne et de la défense débattant devant un juge impartial de la valeur des preuves), cette spécialité a naturellement adopté une approche empirique en ce qui a trait aux produits des outils, techniques et procédures.

On a de plus en plus d'algorithmes décideurs harnachés opérées par des « pseudo-intelligences artificielles ». On s'inquiète de leurs inférences et déductions parce qu'on ne peut les expliquer avec une simple intelligence humaine. Et on veut retrouver la logique sous-tendant les décisions prises par ces machines. Et on s'inquiète des biais que ces dites IA apprennent des données qu'on leur sert. On s'inquiète de cette boîte noire que sont les algorithmes décideurs. On devrait tout autant s'inquiéter de la boîte noire des assistants numériques personnels et la présente étude s'occupera

de jeter de la lumière dans la boîte noire de l'écosystème numérique des assistants numériques personnels.

### 2.3. Question de recherche

**Quelles informations peut-on recueillir localement au sujet de la vie privée d'un utilisateur lorsque celui-ci utilise l'assistant numérique personnel Cortana et quelles sont les données émises vers Internet et reçues d'Internet lors du traitement consécutif à cette utilisation?**

### 2.4. Définition de « vie privée »

Définir le concept de « vie privée » est important lorsqu'on veut déterminer si l'information extraite des données générées par les requêtes aux assistants numériques personnels se rapporte à la vie privée de l'utilisateur ou non.

Les premiers efforts connus pour définir ce qu'est la vie privée reviennent à Warren et Brandeis [1890] qui, citant le Juge Cooley parlent de « *right to be let alone* », c'est-à-dire le droit de ne pas être importuné. En fait, il s'agit du droit à la vie privée et non de la vie privée comme telle.

Westin [1967] définit ainsi la vie privée : « *Privacy is the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others* ». Mais cette définition soulève des questions. Est-ce qu'un groupe peut se clamer avoir une vie privée? Pensons à une compagnie opérant un magasin ou à un cabinet comptable. Mais pensons aussi à un groupe comme un couple et ses enfants formant une famille. La même question se pose au sujet d'une institution comme l'Agence du revenu du Canada. Ou un hôpital. On pourrait aussi se poser la question au sujet d'un individu. Le fait de commettre un crime enlève-t-il au criminel sa vie privée? Cette définition est plutôt subjective puisqu'on peut inclure ou exclure de chaque catégorie visée (individus, groupes, institutions).

Nissenbaum [2010] affirme que « The central thesis of this book is that a right to *privacy is neither a right to secrecy nor a right to control but a right to appropriate flow of personal information* ». Ici, l'auteure parle donc du droit à la vie privée et ne définit donc qu'indirectement la vie privée. Elle continue en disant « …*some critics have concluded that privacy is at best a culturally relative predilection rather than a universal*

*human value… there is, indeed, great complexity and variability in the privacy constraints people expect to hold over the flow of information, but these expectations are systematically related to characteristics of the background social situation* ». Pour Nissenbaum, on ne peut prononcer les mots « vie privée » sans tenir compte du contexte où se place la donnée.

Waldman [2018] fait appel aux philosophes des Lumières Locke et Kant pour justifier le droit à la vie privée. Il dit que plusieurs théories de la vie privée parlent de séparation, d'isolement et d'exclusion de certains aspects de la vie personnelle face au regard public. Il revient sur les concepts d'autonomie, de choix et de contrôle (on a vu que cette approche était celle de Westin). Après une couverture étendue des théories de la vie privée, il finit par conclure que « … *privacy, at least in the information-sharing context, is not about separating from society, but rather about engaging with it on terms based on trust* ».

Dans son livre « Vie privée et droits fondamentaux », Nadeau [2000] met en équilibre certains éléments pouvant faire partie de la définition de la vie privée :

- « *Bien que ce que l'on considère comme étant privé puisse varier selon les pratiques culturelles, il semble que la quasi-totalité des sociétés humaines recherchent la protection d'une sphère d'intimité pour accomplir certaines activités* »
- « *Ce besoin de réclusion doit aussi être mis en perspective avec le besoin de stimulations sociales* »
- « *…quatre composantes physiques et psychologiques inhérentes à la vie privée : la solitude… l'intimité… l'anonymat… et la réserve* »

Pour Nadeau [2000], vie privée et actions de l'État sont des antonymes. Et pourtant, sans surveillance de l'État, comment pourrait-il y avoir respect des lois et limitation de l'empiètement de quiconque dans la vie privée d'un individu?

L'article 7 de la Charte canadienne des droits et libertés prescrit que « *Chacun a droit à la vie, à la liberté et à la sécurité de sa personne; il ne peut être porté atteinte à ce droit qu'en conformité avec les principes de justice fondamentale* ». Le concept de « vie privée » est inclus dans le concept de « vie » et de « liberté ». C'est la liberté de vivre sa vie comme on l'entend. L'article 35 du Code civil du Québec précise : « *Toute personne a droit au respect de sa réputation et de sa vie privée.* » Beaudoin et Renaud [2015], commentant la cause Thomas c. Publications Photo-police Inc., précisent que « *Le droit au respect de la vie privée est un droit de la personnalité qui découle de la simple existence de l'être humain.* »

Dans la jurisprudence issue des cours de justice au Canada, on parle souvent d'attente (*Voir R. c. Gomboc [2010]*), de droit, d'atteinte, d'expectative, d'empiètement, de protection ou d'immixtion dans la vie privée d'une personne.

Pineau [2014], parlant de la protection de la vie privée, écrit : « *Trois aspects sont visés par cette protection : la personne, les lieux et l'information* ». Dans l'arrêt R. c. Dyment [1988], au paragraphe 19, le juge La Forest décrit ces trois aspects comme étant les sphères personnelle, spatiale (ou territoriale) et informationnelle.

Pour Desharnais [2015c], « *La vie privée d'une personne est constituée des aspects de soi qu'une personne est légitimée de préserver par-devers elle, pour son utilisation exclusive ou dans le but de créer et maintenir son confort moral* ». Une fois qu'on a dit ceci, on n'a pas dit beaucoup. On a simplement mis en place une base de discussion.

Dans R. c. Jarvis [2019], le juge en chef Wagner écrit que « *Le concept de « vie privée », selon le sens qui y est habituellement donné, n'est pas absolu… [et] une intrusion dans la vie privée dépend plutôt d'un ensemble de facteurs…* »

S'il s'avère que le concept de « vie privée » est un concept basé sur un ensemble de facteurs. Le mystère suivant reste donc entier : « Sur quels critères objectifs et immuables peut-on se fonder pour déterminer si un objet, tangible ou intangible, fait partie ou non de la vie privée? »

En droit canadien, on oppose « objectif » à « subjectif » et on comprend ce dernier mot dans le sens que lui donne la juge McLachlin dans R. c. Hundal [1993], à savoir « … *un critère subjectif,* [est] *fondé sur ce qui s'est vraiment passé dans l'esprit de l'accusé* ». Quant à la norme objective, on le comprend sous l'angle donnée par le juge Mainville au paragraphe 35 de sa décision dans la cause Canada c. Buckingham [2011] que « *Le renvoi à une personne raisonnablement prudente indique clairement que le critère est objectif* ». Ici, la façon de voir du droit ne nous apportera pas beaucoup pour résoudre le mystère de ce qu'est la vie privée puisqu'on utilise pour la définir le concept de « personne raisonnablement prudente », qui est lui-même un concept flou.

### 2.5. Importance de la vie privée

Solove [2014] énonce des raisons pourquoi la vie privée est importante. Cet éminent professeur de l'université George Washington dresse en fait une liste des raisons de l'importance du droit à la vie privée. Il cite notamment le respect des individus (dont Emmanuel Kant nous entretient dans ses « Fondements de la Métaphysique des mœurs »), de contrôle de sa propre vie (on en revient Warren et Brandeis), de la possibilité de changer et d'avoir une seconde chance (pensons à une personne commettant un crime et s'amendant par la suite en devenant un citoyen exemplaire) et de ne pas avoir à s'expliquer ou à se justifier des décisions nous concernant. Solove poursuit en disant que la protection contre l'immixtion de l'État et des grandes organisations dans les vies privées est une raison d'être de la vie privée. Il parle aussi d'intimité qui est une façon de maintenir une frontière sociale entre soi et les autres.

Dans son « Cycle de Ténébreuse », Marion Zimmer Bradley (Voir Wikipédia [2020b] et Bradley [1972]) parle d'une planète peuplée de télépathe. Un vaisseau peuplé de colons humain s'y écrase. Certains humains ayant une prédisposition pour la télépathie sont en mesure de communiquer avec d'autres par la pensée. Fort heureusement, Bradley a inventé une série de règles de bienséance où les télépathes ne doivent pas épier les autres et doivent suivre un strict protocole de prise de contact.

Imaginons un monde télépathe sans règle de bienséance. Où vos observations sur l'habillement des professeur.e.s ou des étudiant.e.s, leur peignure ou leurs manies ou tics pendant les cours soient portées instantanément à la connaissance des personnes qui vous entourent, en général, et à la vôtre en particulier. Un monde où l'image que vous avez de votre physique soit transparents à ceux que vous rencontrez. Un monde où il est impossible de mentir. Ni de garder pour soi ses réflexions. Un monde où les antipathies sont étalées au grand jour. Où les pensées lubriques ou violentes sont comme une émission qu'on regarde à la télé. Un monde où tout le monde est tellement présent qu'il en écrase, broie et détruit la personnalité individuelle à coup sûr. Un tel monde tuerait l'autonomie de penser, la créativité, le plaisir d'être avec une personne aimée. Il mettrait à mal le développement de l'autonomie de l'enfant, l'image qu'il se fait de lui-même et de sa valeur puisque c'est la norme sociale qui lui indiquerait quoi faire, quand le faire et comment le faire. Il nivellerait les sentiments jusqu'à ce qu'il n'y en ait plus.

Dans « *L'invention du mensonge* » (voir Gervais et Robinson [2009]), Ricky Gervais personnifie un homme vivant dans un monde où le mensonge n'existe pas. Il tombe amoureux d'une femme qui ne le trouve pas de son goût. Pour se faire valoir, il créé le mensonge, ce qui l'amènera à devenir extrêmement riche… mais toujours pas aimé de la femme élue de son cœur.

Dans ces deux mondes, la vie privée en tant que lubrifiant social n'existe pas. Dans l'un, nos vagabondages intellectuels sont publics et dans l'autre la vérité toute nue est imposée aux autres sans regard à son désir de la connaître ou pas.

Avoir une vie privée, c'est pouvoir se protéger contre les immixtions de tous, pas juste de l'État. En tant que telle, elle sert à établir une frontière sociale afin de ne pas être vulnérable aux attaques des autres, que celles-ci soient le fait d'une personne mal intentionnée ou non. Mais ça nous met aussi à l'abri de procès d'intention, basés sur des pensées auxquelles on n'aurait pas donné suite. La vie privée est un lubrifiant social.

Certes, les mondes dont on a parlé dans les paragraphes ci-haut sont extrêmes. Mais le point auquel les ordinateurs sont bavards est tout aussi extrême. Dans un monde idéal, ces derniers ne retiendraient aucune information de nature privée.

## 3. Sommaire de l'état de la question

Pour rappel, les assistants numériques personnels sont des logiciels pouvant s'exécuter sur plusieurs plateformes (ordinateur, appareil dédié, cellulaire, dans le nuage) qui ont comme propriétés principales d'être toujours en éveil et disponibles au moment où un mot est prononcé (« Hey Cortana ») ou un geste posé. Lorsque l'utilisateur soumet une requête verbale à son assistant numérique personnel, la bande son de la requête est expédiée à un serveur distant qui la traduira en texte, formulera la requête Internet et renverra le résultat à l'émetteur de la requête.

Ce procédé laisse des traces à plusieurs niveaux dans l'écosystème numérique de l'appareil recueillant la requête, localement et sur des appareils distants. Au niveau local, l'assistant numérique personnel utilisera la mémoire vive et la mémoire de débordement, y laissant des traces. Comme tout programme informatique, l'assistant numérique personnel doit utiliser des processus (« process »), des fils (« threads ») et des descripteurs (« handles ») pour pouvoir s'exécuter. Ces

fonctionnalités laissent des traces dans les mémoires locales. Parallèlement à l'assistant numérique personnel, d'autres programmes s'exécutent, notamment les navigateurs web, agendas, courriels.

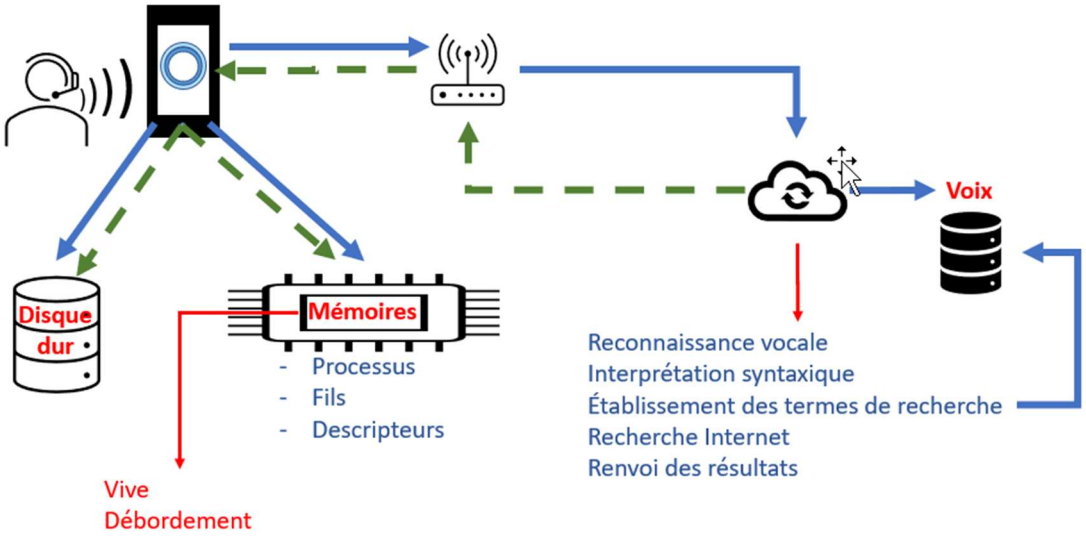



Figure Intro-2 – Écosystème numérique d'un assistant numérique personnel et localisation des traces laissées par son utilisation

L'audio de la requête est recueilli puis mis en paquets pour être envoyés par le biais d'Internet. Il est possible de recueillir ces paquets et les décortiquer. L'utilisateur peut interdire à Cortana de répondre à une requête soumise par une voix autre que la sienne. Ceci est rendu possible par l'identification d'une voix grâce à des facteurs sous-jacents de la voix et qui sont propres à chaque individu sur Terre.

## 4. Objectifs généraux et spécifiques

Afin de répondre à la question de recherche, nous avons d'abord voulu confirmer les traces d'utilisation de Cortana déjà découvertes par d'autres chercheurs et localiser d'autres traces afin de fournir un portrait global des traces laissées par Cortana suite à son utilisation. L'utilisation faite par Cortana de la mémoire vive et de débordement ayant été, à ce jour, fort peu couvert par la littérature. Nous nous sommes occupés de défricher cet aspect ce qui nous a permis de déterminer la séquence de démarrage des processus et des fils utilisés par Cortana lorsque l'utilisateur lui soumet une requête. Enfin, nous avons voulu savoir si un adversaire suffisamment motivé pourrait

récupérer les paquets contenant la voix de l'utilisateur afin de créer ses propres requêtes avec la voix de sa victime.

## 5. Sommaire des méthodes et des protocoles expérimentaux utilisés

Afin de débuter la phase expérimentale de notre recherche avec un environnement exempt de toute interaction utilisateur-machine, nous avons installé le système d'exploitation Windows 10 Pro sur notre ordinateur équipé d'un disque préalablement aseptisé (« wipé ») de manière forensique.

Nous avons soumis à Cortana une série des requêtes déterminées d'avance et dans l'ordre déterminé. Le groupe expérimental a pris en note heure:minute:seconde à laquelle telle requête a été soumise à Cortana afin de nous permettre d'utiliser les tampons horodateurs rattachés à l'utilisation des mémoires vive et de débordement ainsi qu'à la sauvegarde de données sur le disque dur occupé par le système d'exploitation.

Pendant chaque requête, nous avons recueilli les flux de paquets transmis vers et reçu d'Internet. Nous avons utilisé Wireshark pour en faire la capture et l'analyse. Après chaque requête, nous avons recueilli une copie forensique des mémoires vive et de débordement.

À la fin de chaque phase expérimentale, nous avons pris une image forensique du disque dur occupé par le système d'exploitation et nous en avons fait le traitement (indexation, désencastrement, tri, recherche par type et plus) pour pouvoir en examiner le contenu de manière plus efficace.

## 6. Structure du présent document

Le chapitre 1 présente les concepts de base en forensique, suivi au chapitre 2 des logiciels utilisés pendant le projet de recherche. L'état de l'art sur la forensique liée aux assistants numériques personnels se trouve au chapitre 3 et celui sur les systèmes d'exploitation exploitant un assistant numérique personnel est au chapitre 4. Les protocoles expérimentaux exécutés pendant le projet sont divulgués au chapitre 5. Les observations et résultats ont été livrés en deux chapitres. Au chapitre 6, nous faisons état des observations et résultats tirés de l'analyse des mémoires vive et de

débordement. Le chapitre 7 livre nos observations et résultats se rapportant aux autres analyses : celle des copies forensiques du disque dur de l'ordinateur supportant Cortana ainsi que celle des flux transmis et reçus d'Internet. La conclusion livre enfin un sommaire des découvertes, une ouverture sur les perspectives futures avant de conclure.

# CHAPITRE 1 – FORENSIQUE

## 1. Introduction

Comme mentionné au paragraphe 2.2.3 du chapitre introductif, l'informatique forensique est une des spécialités de l'informatique qui requiert des connaissances spécifiques à la spécialité, implique des procédures qui lui sont particulières et utilise des outils qui lui sont propres. Pour bien comprendre comment les résultats ont été obtenus, il est nécessaire de faire un survol des notions propres à l'informatique forensique.

## 2. Définition

Forensique : Desharnais [2015a] définit ainsi ce terme « *Qualité des actions de recherche, de cueillette et de production d'une preuve authentique et fiable lors d'une procédure administrative, civile ou criminelle.* »

C'est ce terme que nous utiliserons dans ce mémoire pour qualifier les techniques et procédures utilisées pour trouver les traces de vie privée laissées par l'utilisation des assistants numériques personnels. Aussi, nous utiliserons indifféremment « informatique forensique » et « investigation numérique » en notant que ce dernier terme correspond à l'utilisation de l'informatique forensique afin de mener une enquête sur un délit civil ou criminel.

Ce terme est déconseillé par l'Office québécois de la langue française (*Voir GDT - Criminalistique*) car, disent-ils, il provient soit de l'anglais ou de l'allemand, que l'utilisation de ce terme pourrait semer la confusion avec les autres termes suggérés par ce noble office qui sont, disent-ils, bien implantés dans le vocabulaire populaire. Or, le terme forensique vient du latin forensis, terme que Gaffiot [1934] définit ainsi : « …*de la place publique, du forum, judiciaire : domesticus, forensis labor…le travail chez soi (du cabinet), le travail du forum [= la plaidoirie]* … ». Quant à l'acceptation et l'implantation dans le milieu, nous n'avons jamais entendu quiconque dire qu'il faisait de la criminalistique informatique.

## 3. Rôle

L'assistant numérique personnel Cortana utilisé lors de la phase expérimentale de ce mémoire était installé sur un ordinateur portable. Comme pour tout contenu d'un ordinateur, les données générées par les requêtes de l'utilisateur de Cortana sont substantielles. Dans R. c. Vu [2013], le juge Cromwell indique « *les ordinateurs sont susceptibles de donner aux policiers accès à de vastes quantités de données sur lesquelles les utilisateurs n'ont aucune maîtrise, dont ils ne connaissent peut-être même pas l'existence ou dont ils peuvent avoir choisi de se départir, et qui d'ailleurs pourraient fort bien ne pas se trouver concrètement dans le lieu fouillé* »

Lors de notre recherche, nous avons donc appliqué des techniques et des procédures visant à recueillir et protéger l'intégrité des traces de vie privée laissées par les assistants numériques personnels. Cette attitude a été maintenue constamment lors des tests, avec les exceptions que nous divulguerons au fur et à mesure. Nous voulions par là retrouver, recueillir et examiner les traces de vie privée dans l'état le plus près possible de l'état où elles étaient au moment de leur création.

Les principales techniques forensiques utilisées dans ce travail de recherche sont la copie forensique bit-à-bit , le clonage, la restauration forensique bit-à-bit et l'aseptisation qui seront couvertes à la section 3.2 du présent chapitre.

Les principales procédures forensiques utilisées sont le traitement des copies-images et du contenu de la mémoire vive, l'analyse des extrants résultant du traitement et l'extraction des preuves relevées pendant le traitement.

## 4. ABC de l'informatique forensique

### 4.1. Fondements de l'informatique forensique – Propriétés des preuves numériques

#### 4.1.1. Principe de contamination

Ce principe est souvent appelé à tort le principe de Locard. Wikipédia [2020] indique qu'Edmond Locard est un criminologue lyonnais ayant œuvré au début du 20ième siècle. Il œuvrait donc dans le domaine physique et non dans celui du numérique. Le principe de Locard veut que la violence

avec laquelle les crimes sont commis font en sorte que tout contact entre l'assaillant et sa victime laisse forcément des traces.

*Le principe de contamination veut que tout média mis en contact avec un autre sera contaminé par ce dernier et contaminera ce dernier.* Ceci a mené cette spécialité de la forensique à développer des outils pour permettre à un système d'exploitation d'accéder au contenu du média sans toutefois y inscrire des données. Les outils utilisés dans cette spécialité peuvent être du matériel ou un logiciel. Par exemple, lorsqu'un investigateur numérique veut examiner une clé USB, il exécutera d'abord, sur l'ordinateur d'examen, un script qui modifiera la base de registre afin d'enlever la permission d'écrire au pilote des ports USB. Ou alors il branchera la clé USB sur un appareil dont la tâche est de bloquer les ordres d'écriture.

#### 4.1.2. Principe de persistance

*Le principe de persistance indique qu'une donnée inscrite sur un média y restera un temps substantiel sauf si des moyens spécifiques sont pris pour les éliminer.* Un système de fichiers est un programme informatique qui gère le stockage de l'information sur un média. Afin de diminuer la latence lors du stockage des données, les systèmes de fichiers les plus populaires (notamment FAT, NTFS, ExtFS, HFS+ et APFS) ne réutilisent les espaces libérés par la suppression d'un fichier que s'il n'y a nulle part ailleurs où stocker le fichier de façon séquentielle. Ceci est la source de l'existence du second fondement de l'informatique forensique, le principe de persistance. Le spécialiste se servira alors d'outils comme FTK Imager ou Autopsy Sleuth kit pour examiner le contenu des fichiers effacés et les extraire en cas de besoin.

#### 4.1.3. Principe de dispersion

*Le principe de dispersion veut que l'ensemble des preuves soit rarement localisé dans un seul répertoire.* Cortana, par exemple, sauvegarde la requête verbale faite par l'utilisateur dans un répertoire par défaut assigné par le système d'exploitation au moment où il est installé, mais une partie de la réponse à cette requête est localisée dans les répertoires « *Temporary Internet Files* » du navigateur Edge et certaines données envoyées ou reçues lors d'une requête et de sa réponse se retrouvent dans la mémoire vive ou dans les mémoires de débordement. Par ailleurs, il y a la tendance très humaine des utilisateurs à sauvegarder leurs données dans de multiples répertoires et en plusieurs versions. Le spécialiste voulant accéder à ces données éparses, utilisera alors un outil comme Autopsy Sleuth

kit. S'il sait où les données se trouve, il utilisera des logiciels programmés pour une seule tâche, par exemple Prefetch Analyzer.

#### 4.1.4. Principe de furtivité des preuves

*Le principe de furtivité des preuves veut que certaines preuves numériques se trouvent dans des endroits peu ou pas accessible à l'utilisateur lambda.* Avec les assistants numériques personnels, on est servis! Chacun de ceux-ci sauvegardent des données un peu partout dans son écosystème numérique, parfois très profondément dans l'arborescence du média à partir duquel s'exécute le système d'exploitation. C'est-à-dire que pour accéder à l'information, soit on utilise des outils logiciels spéciaux, soit on sait comment accéder aux répertoires normalement cachés à l'utilisateur régulier. Ces répertoires furtifs représentent une menace patente à la vie privée des utilisateurs. Beaucoup d'utilisateurs en connaissent l'existence et savent que des données y sont sauvegardées à leur corps défendant. Toutefois, peu c'utilisateurs se soucient de leur existence ou de leur contenu lorsque vient le moment de décider d'utiliser les logiciels générant ces données furtives. Généralement, la localisation de ces fichiers est bien connue du spécialiste expérimenté qui utilisera alors un logiciel comme FTK Imager.

#### 4.1.5. Principe de volatilité des preuves

*Le principe de volatilité des preuves veut que celles-ci ont une existence éphémère et que la durée de cette existence est influencée par plusieurs facteurs*, notamment l'aspect transitoire de la donnée (par exemple un paquet Internet transmis, qui ne peut être poursuivi et récupéré une fois qu'il a dépassé le point où il devait être récupéré) ou le fait d'avoir été supprimé dans l'interface graphique de navigation.

Le principe de volatilité des preuves Lorsqu'un assistant numérique personnel est sollicité, une partie des données envoyées au titre de la requête de l'utilisateur et une partie des données reçues au titre de la réponse à la requête, restent dans la mémoire vive ou dans les mémoires de débordement, même après que la requête soit satisfaite et terminée. Or, le contenu de la mémoire vive ne persiste pas une fois qu'on aura redémarré l'ordinateur. Ce contenu est donc volatile. Le spécialiste avisé fera donc une copie du contenu de la mémoire vive avec FTK Imager et décortiquera le vidage de mémoire avec un logiciel Volatility.

#### 4.1.6. Conséquences des propriétés des preuves numériques

Les fondements énoncés à la présente section 4.1 ne constituent qu'une partie des fondements de l'investigation numérique. Mais ils justifient à eux seuls les procédures utilisées par les investigateurs numériques et ont fait en sorte qu'au fil du temps, ceux-ci ont conçu des outils spécialisés et des techniques et procédures propres à cette spécialité informatique.

### 4.2. Concepts de base

Ci-haut, nous avons mentionné les trois principales techniques utilisées pendant cette recherche, à savoir l'aseptisation, la copie forensique bit-à-bit et la restauration forensique bit-à-bit. De plus, nous avons mentionné comme procédures le traitement des copies-images et du contenu de la mémoire vive, l'analyse résultant des traitements en question et l'extraction des preuves. Décrivons brièvement en quoi elles consistent.

#### 4.2.1. Copie forensique bit-à-bit

Cette technique est aussi désignée sous les vocables équivalents de « image », « image forensique », et « copie bit-à-bit ». Cette procédure demande un logiciel spécialisé comme FTK Imager mais elle peut aussi être réalisée avec le premier copieur forensique bit-à-bit de l'histoire informatique : la commande de terminal issue de Unix « dd ». Faire une copie forensique bit-à-bit, c'est de lire séquentiellement les octets d'un média pour les stocker dans un ou plusieurs fichiers. La séquence adoptée est de lire le contenu des secteurs dans leur ordre de numérotation en commençant au décalage[1] 0 du secteur pour terminer au décalage 511[2]. La taille des fichiers conteneurs est généralement paramétrables, de même que la qualité de la compression[3]. Quant au nombre de fichiers générés, il dépend de la quantité de données sur le média d'origine, de la taille configurée pour les fichiers conteneurs et de la qualité de la compression.

---

[1] Décalage : C'est le nombre d'octets qui séparent le premier octet d'un point de référence fixe de l'endroit qu'on veut désigner.
[2] Dans la plupart des systèmes de fichiers, les secteurs comportent 512 octets numérotés de 0 à 511. La seule exception connue est le cas des CD et DVD où les secteurs ont une taille de 2 048 octets numérotés de 0 à 2 047.
[3] La qualité de la compression est paramétrable parce que plus on compresse et plus la copie forensique bit-à-bit prend du temps. Par ailleurs, moins on compresse et plus on a besoin d'espace sur le média contenant les conteneurs. Le spécialiste doit donc faire la part des choses en ce qui a trait à la compression vs le temps disponible pour l'exécution du mandat vs l'espace dont il dispose sur le média contenant les fichiers conteneurs.

#### 4.2.2. Clone

Un clone est similaire à une copie bit-à-bit. Mais au lieu de sauvegarder les bits dans un fichier, on les envoie directement sur un média de taille égale ou supérieure à la taille du média source. À la fin du processus, le média cible a exactement la même empreinte numérique que le média source.

#### 4.2.3. Restauration forensique bit-à-bit

Cette technique est en fait l'inverse de la copie forensique bit-à-bit. On lit séquentiellement les fichiers conteneurs résultant de la copie forensique bit-à-bit (souvent désignés, globalement, sous le nom d'image) et on en transfère le contenu sur un média de taille égale ou supérieure à la taille du média original. On obtient ainsi une copie authentique absolument identique à l'original, empreinte numérique[4] à l'appui. Pour produire un niveau de confiance similaire dans le monde papier, on devrait faire appel à un notaire affirmant que la copie est conforme à l'original.

#### 4.2.4. Aseptisation

Cette technique permet de nettoyer un média avant de l'utiliser, par exemple avant une expérience ou avant de l'utiliser comme conteneur d'une copie forensique bit-à-bit. Cette technique est en fait un cas particulier de la restauration où l'intrant n'est pas un fichier conteneur d'une copie forensique bit-à-bit, mais plutôt une séquence de nombre hexadécimaux. Ce flux est alors écrit de manière séquentielle sur le média à nettoyer, écrasant ainsi toutes les données qui y sont, de manière que seules les données qu'on y mettra fasse partie de ce média. Dans notre recherche, nous avons utilisé cette technique avant de placer des données sur les disques durs USB contenant les copies forensiques des mémoires vives et les copies forensiques bit-à-bit des disques durs.

---

[4] Empreinte numérique : Plus connue sous le nom de « hash ». L'empreinte numérique est un nombre (plutôt chaîne de caractères) qui constitue le résultat d'un calcul fait selon une formule mathématique convenue. L'intrant est le contenu dont on calcule l'empreinte numérique et c'est généralement le contenu du fichier. Si on calcule l'empreinte numérique d'un fichier et qu'on modifie un seul bit choisi au hasard (ou pas, en fait!) et qu'on recalcule l'empreinte numérique, on obtiendra un résultat totalement différent du premier, sans aucune ressemblance entre eux que le fait qu'ils sont constitués de caractères. Les métadonnées localisées hors du fichiers peuvent être modifiées sans affecter l'empreinte numérique.

## 4.3. Stratégies de fouille

On peut diviser en trois grandes familles (métadonnées, données et exploration de contenu) les méthodes pour retrouver des informations sur un média qu'on examine :

### 4.3.1. Métadonnées

#### 4.3.1.1. Paradonnées – Extensions de fichiers

Les extensions de fichiers sont constituées d'un nombre limité de lettres, de chiffres ou de symboles et leur fonction est d'établir un lien entre un fichier et l'application qui l'a créé. C'est ainsi qu'un simple clic sur mémoire.docx permet d'ouvrir avec Microsoft Word. Qu'arrive-t-il si on change l'extension .docx pour .jpg? MS Word rejette simplement le fichier en disant qu'il est corrompu.

#### 4.3.1.2. Intradonnées – Signatures de fichiers

Une signature de début de fichier est une séquence de 2 à 16 octets localisée dans les 32 premiers octets du fichier. Elle a comme rôle de valider la « filiation » établie par l'extension avec le programme. S'il y a désaccord entre l'extension et la signature, le fichier est déclaré corrompu et le logiciel refuse d'ouvrir le fichier. Il y a aussi les signatures de fin de fichier, constituées de 2 à 16 octets situés dans les 32 derniers octets du fichier. Cette signature a pour but de réduire les ambiguïtés sur la localisation de la fin du fichier.

#### 4.3.1.3. Intra et paradonnées

Ce concept est très utile à l'investigateur numérique et contient des informations utiles comme la localisation géographique de la prise d'une photo, le propriétaire d'un logiciel, l'usager de l'ordinateur.

### 4.3.2. Données

#### 4.3.2.1. Empreintes numériques incluant et excluant

Nous avons couvert dans un paragraphe précédent le concept d'empreinte numérique. Lorsqu'on connaît l'empreinte numérique d'un fichier, il est extrêmement simple et rapide de le retracer sur

un média où il est présent. Les logiciels d'investigation numérique calculent systématiquement ces valeurs et si on leur fournit des valeurs, dans une base de données qu'on appelle « jeu d'empreintes numériques » (« hash set »), il comparera les empreintes numériques du média à celles du jeu d'empreintes numériques. Et selon qu'on aura qualifié le jeu de « jeu excluant » ou de « jeu incluant », le logiciel exclura de l'affichage final le fichier en question ou le placera dans un répertoire particulier de l'affichage final. L'exemple typique de fichiers à exclure est celui des fichiers créés par les fabricants des logiciels ou du système d'exploitation de l'ordinateur dont examine le disque dur. L'exemple typique de fichiers à inclure est celui de fichiers de pornographie infantile ou de documents volés.

#### 4.3.2.2. Mots ou phrases clés

Habituellement utilisée par l'utilisateur lambda, cette stratégie de fouille est à la fois puissante et trompeuse. Puissante parce qu'on aboutit à un résultat rapide. Trompeuse parce qu'exposée aux faux négatifs et positifs. Cette exposition est le résultat de sa conception même : Si on fait une erreur typographique dans l'énoncé de la recherche (on recherche, par exemple, Des Harnais plutôt que Desharnais), on aboutit à des faux négatifs. Si le mot ou la phrase recherchée contient les lettres d'un mot commun sur le média recherché, on aura une avalanche de résultats qui ne sera pas efficacement utilisable (par exemple, si le propriétaire de l'entreprise se prénomme Jean et que son entreprise se nomme « Le Jeans Délavé », on aura le « Jean » de « Jeans » sur chaque facture faites par son entreprise)

#### 4.3.2.3. Expression régulière

Une expression régulière est une formule simili-mathématique décrivant la structure d'une donnée. Par exemple, si on veut décrire la structure d'un numéro de téléphone nord-américain, on écrira : [0-9]{3}-[0-9]{3}-[0-9]{3} ou encore \d{3}-\d{3}-\d{4}
Il s'avère que beaucoup de logiciels d'investigateur numérique disposent d'une fonctionnalité « expression régulière », notamment FTK Imager.

#### 4.3.2.4. Exploration des données

On peut aussi consulter le contenu d'un fichier et y rechercher une information particulière. Par exemple, rechercher les chèques émis à mon nom dans la comptabilité de l'entreprise.

#### 4.3.3. Exploration de contenu

##### 4.3.3.1. Arborescence

On peut regarder le contenu d'un média avec un explorateur de fichiers pour apprendre quels sont les logiciels installés sur l'ordinateur. Ceci peut nous mettre sur la piste du type de données générées par cet ordinateur et l'extension et la signature du fichier dans lequel elles se trouvent.

##### 4.3.3.2. Lieux probables

Certains programmes informatiques, lors de leur installation, configureront le répertoire par défaut où les fichiers qu'ils produisent seront sauvegardés. Par exemple, MS Word sauvegarde ses documents dans \Users\utilisateur\Documents. Si on recherche un fichier .docx, on commence par le rechercher dans ce répertoire avant de passer à autre chose.

##### 4.3.3.3. Données furtives

Naviguer sur Internet implique que des fichiers soient sauvegardés sur l'ordinateur dans des répertoires auxquels l'utilisateur lambda n'a pas accès. Cortana fait de même. Nous verrons à quel point les fichiers qu'il laisse derrière lui une fois la requête terminée constituent un ensemble très loquace.

##### 4.3.3.4. Données volatiles

Enfin, il y a les données comme le contenu de la mémoire vive et les signaux de télécommunication émis et reçus.

### 4.4. Autres concepts utiles

#### 4.4.1. Physique ou logique

On doit noter que l'aseptisation, la copie et la restauration forensique peuvent se faire à deux niveaux. On peut appliquer ces techniques au niveau physique ou au niveau logique. Au niveau physique, la technique s'appliquera à l'ensemble du média, incluant les secteurs de démarrage, les partitions et les balances de partition. Au niveau logique, la technique s'appliquera à la partition choisie seulement.

#### 4.4.2. Entête, corps et remorque de fichier

La plupart des fichiers se divisent en ces trois parties. Le corps contiendra les données proprement dites : ce qu'on tape dans un traitement de texte, l'extrant du programme Python, les pixels de la photo de notre enfant et ainsi de suite. L'entête contiendra généralement des données comme le propriétaire du logiciel utilisé pour produire le fichier, la version de Python utilisée pour créer les données, les coordonnées topographiques où on était lorsqu'on a pris la photo. La remorque du fichier contiendra des données comme les versions précédentes du document ou les données de formatage, la durée du traitement pris par Python pour produire l'extrant, des remarques sur la photo.

#### 4.4.3. Métadonnées, paradonnées et intradonnées

La définition classique d'une métadonnée est qu'elle constitue une donnée au sujet des données. Les métadonnées décrivent les caractéristiques des données. Les métadonnées situées dans le même fichier que les données qu'elles décrivent sont appelées « intradonnées ». Les métadonnées situées à l'extérieur du fichier contenant les données qu'elles décrivent sont appelées « paradonnées ».

Ces deux mots sont des néologismes créés pour des raisons pédagogiques afin d'expliquer pourquoi un logiciel de forensique doit être utilisé pour copier des données au lieu d'utiliser Windows Explorer (ou Linux ou Mac Explorateur de fichiers…). Ces néologismes sont très utiles car il suffit de dire que les explorateurs de fichiers ne copient que les données et les intradonnées, mais pas les paradonnées. Or, ces paradonnées contiennent généralement des informations comme la date de création du fichier sur le média examiné, le nom du propriétaire du compte de l'ordinateur qui a été utilisé pour créer ou modifier le fichier, l'endroit sur le disque où est stocké le fichier. Au point de vue forensique, ces métadonnées ont vraiment beaucoup d'utilité et elles ont servi lors de cette recherche.

#### 4.4.4. Allocation des grappes

Dans les systèmes de fichiers classiques, l'allocation d'un espace de stockage à un fichier ou à un répertoire se fait grappe par grappe et non secteur par secteur. Une grappe est un regroupement

de secteurs. Un fichier ou un répertoire peut occuper plus d'une grappe mais une grappe ne peut pas être allouée à plus d'un fichier ou répertoire.

#### 4.4.5. Balance (de fichier, de partition)

Les fichiers n'ont pas tous la même taille. Lorsqu'un fichier est stocké dans une grappe, le système de fichiers prend note que cette grappe n'est plus disponible pour d'autres fichiers. Pour les fins de l'exemple, nous dirons que cet indicateur est à 1 si la grappe est occupée et 0 si elle est disponible pour recevoir un fichier. Dans l'illustration de la figure 1-1, le fichier intitulé « Hachuré oblique.txt » est sauvegardé sur un média quelconque au temps t=0. Ce fichier a une taille de 4 096 octets et occupe donc la totalité de la grappe (car l'utilisateur a choisi, au moment du formatage, d'utiliser des grappes de 4 192 octets. Quelque temps après, au temps t=1, on efface ce fichier. Dans notre système de fichiers, le seul effet de l'ordre de supprimer est de changer la valeur de l'indicateur de Indicateur = 1 à Indicateur = 0. Ceci étant le seul effet, c'est donc dire que toutes les données de « Hachuré oblique.txt » sont encore disponibles pour récupération. Le système de fichiers a donc pris note que la grappe #1234, assigné à « Hachuré oblique.txt » avant qu'on ne le supprime, est maintenant disponible. L'utilisateur décide peu après, au temps t=2, de sauvegarder le fichier « Damier.txt » qui a une taille de 256 octets. Le système de fichiers repère les emplacements disponibles et assigne à ce fichier la grappe 1234. Il sauvegarde « Damier.txt » dans le premier secteur de la grappe, inscrit des zéros entre la fin de ce fichier et la fin du secteur et change l'indicateur pour un 1 (pour confirmer que cette grappe n'est plus disponible pour tout autre fichier). Et rien d'autres. Dans la grappe #1234, il y a donc le fichier « Damier.txt » (256 octets), 256 octets à zéro et 3 584 octets du fichier « Hachuré oblique.txt », à savoir les 3 584 derniers octets de ce fichier. Ces 3 584 derniers octets constituent la balance de fichier du fichier « Damier.txt » (et non la balance du fichier « Hachuré oblique.txt »).

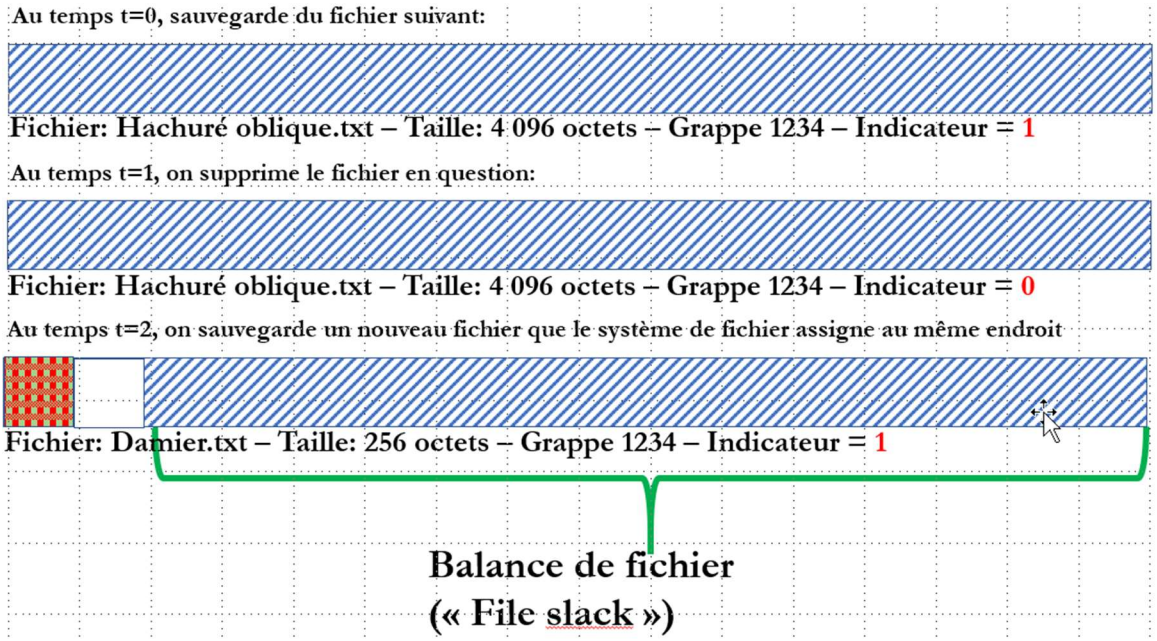



Figure 1-1 – Illustration d'une balance de fichier

#### 4.4.6. Grappe non allouée

*Une grappe non-allouée (« unallocated cluster ») est une grappe, contenant ou non des données, qui est disponible pour en recevoir, mais qui n'est pas encore chargée par le système de fichiers de recevoir des données (qui proviennent généralement de la sauvegarde d'un fichier).*

Une grappe est non-allouée lorsqu'elle n'a jamais contenu de données, lorsqu'elle a contenu des données mais que l'utilisateur a supprimé l'entité (par exemple le fichier) dont les données occupaient cette grappe ou que l'entité a été déplacée dans une autre grappe ou, finalement, lorsque la taille de l'entité a diminué et que la grappe a ainsi été libérée.

#### 4.4.7. Fichier orphelin

*Un fichier orphelin est un fichier effacé dont le contenu ou une partie du contenu est toujours sur le média mais dont le répertoire parent a été effacé.* Ceci fait en sorte qu'on peut, par des techniques forensique, repérer ce qui reste du fichier, récupérer ce reste et parfois même le récupérer en entier et pouvoir le lire dans son logiciel d'origine.

#### 4.4.8. Flux alternatif de données (« alternate data streams)

Manumation [2008] dit que : « …*vous* n'avez *accès en temps normal qu'à un seul flux de données… le "flux normal". Mais NTFS offre la possibilité d'en ajouter un ou plusieurs à un même fichier. Ces flux additionnels sont en quelque sorte des métadonnées…, ils sont complètement invisibles !* ».

Ces flux de données peuvent contenir des informations qu'un logiciel veut cacher à la vue de l'utilisateur mais qui sont utiles pour ce logiciel. Malheureusement, cette fonctionnalité NTFS sert aussi de véhicule d'intrusion pour des pirates informatiques.

#### 4.4.9. $MFT

Il s'agit d'un fichier sauvegardé à la racine du média à partir duquel on exécute Windows (Normalement, le C:). Il constitue le « catalogue » de tout ce qui se trouve dans la partition opérant le Windows en question. Donc, aussitôt qu'un fichier est sauvegardé (par l'utilisateur ou par le système d'exploitation) dans cette partition, une page de 1024 octets est ajoutée au catalogue. Lorsque le fichier est supprimé, l'indicateur « Entrée disponible » remplace l'indicateur « Entrée non-disponible », les données sont laissées telles quelles dans la grappe où elles avaient été sauvegardées et la grappe en question est par la suite considérée comme une grappe non-allouée. Il faut aussi comprendre que lorsque le fichier est très petit (moins de 700 octets approximativement), les données sont sauvegardées à même l'entrée $MFT et non dans une grappe réservée à l'usage exclusif de ce fichier. Ce genre de fichier se nomme « fichier résident ». Lorsqu'on examine ce fichier, on doit s'attendre à retrouver à la fois des données résidentes et des titres de fichiers.

#### 4.4.10. Fichiers désencastrés (« carved files »)

Dans un document comme celui que vous lisez présentement, l'auteur ajoutera parfois des objets : photos, illustrations, graphiques, tableaux. Afin de faciliter l'examen de l'investigateur numérique, les logiciels de forensique comme Autopsy SleuthKit extraient les objets en question et les inscrivent dans la liste des éléments accessibles à l'investigateur numérique comme s'ils étaient des fichiers à part entière. Ceci fait en sorte qu'un média comportant 100 000 fichiers, Autopsy (Comme FTK, Encase et XWays), afficheront 125 000 (par exemple) objets/fichiers à examiner. Ceci pose parfois problème au niveau des preuves déposées en Cour puisque chacun de ces

logiciels découpe les objets à sa façon, si bien qu'au bout du compte le nombre d'objets n'est pas le même. En bout de ligne, la question de la Cour est « Quel objet a été oublié? Quel objet a été désencastré en trop? ».

## 5. Conclusion

Les méthodes appliquées pendant cette recherche sont calquées sur les méthodes d'informatique judiciaire afin d'obtenir des extrants de valeur indiscutable par leur authenticité et leur fiabilité. Les concepts couverts dans ce chapitre sont des concepts techniques qui produisent des résultats dont les investigateurs numériques ont empiriquement démontré le caractère substantiel. Mais ces résultats seraient laborieusement obtenus si ce n'était des logiciels dont nous parlerons au prochain chapitre.

# CHAPITRE 2 – LOGICIELS UTILISÉS

## 1. Introduction

Lors de la présente étude, plusieurs logiciels ont été utilisés. Certains de ceux-ci sont des outils typiquement utilisés par les personnes impliquées dans la sécurité informatique tandis que d'autres, sont connus par les professionnels travaillant en investigation numérique. Afin de bien situer le lecteur lors de sa lecture des chapitres sur le protocole et sur les résultats de cette recherche, nous fournissons ci-dessous une description sommaire de ces logiciels. D'abord les logiciels typiquement utilisés en sécurité informatique, puis ceux d'investigation numérique et enfin, quelques logiciels utilitaires.

## 2. Logiciels de sécurité

### 2.1. Wireshark

Pour permettre aux ordinateurs de communiquer avec d'autres ordinateurs, l'industrie a défini la structure et les standards selon lesquels la communication devait s'établir et se maintenir. Une des caractéristiques des signaux de communication sur Internet est leur regroupement et leur envoi/réception par paquets. Wireshark permet d'intercepter, de stocker et d'analyser en profondeur des données envoyées de et reçues par un ordinateur selon de multiples protocoles de communication.

Selon Wireshark [2020a] « *Wireshark is the world's foremost and widely-used network protocol analyzer. It lets you see what's happening on your network at a microscopic level…* ». Lors de notre recherche, ce logiciel a été utilisé pour recueiller les données transitant par la carte réseau de l'appareil où était installé l'assistant numérique personnel communiquant ou recevant des données. Nous l'avons ensuite utilisé pour analyser le contenu des transmissions.

### 2.2. MITMproxy

Lorsqu'un ordinateur établit une session avec un autre ordinateur à une adresse https, la communication est chiffrée dès le début en vertu d'un certificat numérique émis par un tiers de confiance. mitmproxy est un logiciel permettant de créer un certificat qu'on peut implanter dans l'ordinateur dont on veut écouter les communications. Ce faisant l'ordinateur écouté transmettra tous ses flux de communication à l'ordinateur ayant émis le certificat numérique et ces communications pourront être déchiffrées par ce dernier puisqu'il est l'émetteur du certificat en question. Les communications vers l'ordinateur écouté se feront aussi via l'émetteur du certificat numérique. L'installation de ce faux « tiers de confiance » sur une machine qu'on contrôle permet de connaître la clé de déchiffrement convenue avec la source et de déchiffrer les paquets envoyés vers la destination https finale. Les paquets peuvent ainsi être interceptés en clair par Wireshark (et stockés) puis rechiffrés en vertu d'une session établie avec le https final et expédiés. Lorsque le https envoie la réponse à la requête, mitmproxy réalise le processus inverse. mitmproxy joue donc le rôle d'intercepteur.

mitmproxy [2020a] se déclare le « *swiss-army knife for debugging, testing, privacy measurements, and penetration testing. It can be used to intercept, inspect, modify and replay web traffic*… ». Dans le cadre de notre recherche, nous avons installé ce logiciel sur un ordinateur Linux sur lequel nous avons aussi installé Wireshark. Puis, nous avons implanté le certificat mitmproxy sur l'ordinateur-source (celui exécutant Cortana) et avons établi une communication poste-à-poste entre lui et l'ordinateur jouant le rôle d'intercepteur.

## 3. Logiciels de forensique

### 3.1. FTK Imager

Ce logiciel permet d'examiner un média en lecture seule, d'en créer une copie forensique bit-à-bit (Voir Chapitre 1 – Section 4.2.1) et d'en extraire des artéfacts (Tout objet tangible, ordinateur, disque dur, clé USB, etc., ou intangible, programme, fichier, donnée, etc.).

Selon AccessData [2020a] « *FTK Imager … lets you quickly assess electronic evidence to determine if further analysis … is warranted. FTK Imager can also create perfect copies (forensic images) of computer data without*

*making changes to the original evidence.* ». Pendant notre recherche, FTK Imager a été utilisé pour générer des copies forensique bit-à-bit du disque dur et de la mémoire vive de l'ordinateur sur lequel l'assistant numérique personnel testé s'exécutait. Il a aussi été utilisé pour faire une recherche rapide, sur les copies forensiques, de traces numériques laissées par l'opération de Cortana.

### 3.2. EnCase Forensic Imager

EnCase Forensic Imager est similaire à FTK Imager, mais il permet d'effectuer la restauration d'une copie forensique bit-à-bit vers un média. Ce procédé permet de récupérer les bits stockés dans les fichiers d'une copie forensique et de les écrire sur un média de taille conséquente. Au final, le média cible aura exactement le même contenu que le média à la source de la copie forensique bit-à-bit.

Selon Guidance [2020a], « *EnCase Forensic Imager: Enables acquisition of local drives, Is free to download and use … Can be deployed via USB stick and used to perform acquisition of a live device* ». Pendant notre recherche, EnCase Forensic Imager a été utilisé pour restaurer les copies forensiques bit-à-bit des disques durs contenant les systèmes d'exploitation sur lesquels les assistants numériques personnels étaient installés.

### 3.3. Volatility et la fourche de Fireeye

Volatility analyse et extrait d'un vidage de mémoire (*« memory dump »*), en les séparant, les éléments qui y sont stockés. Ces éléments peuvent être encore en utilisation ou non. Ces éléments peuvent être reliés à plusieurs types d'artéfact, notamment des processus, des pilotes, des fichiers utilisateurs. Volatility est gratuit et vient avec plus de 125 modules, chacun pouvant extraire un type précis d'artéfact.

En septembre 2013, Microsoft a commencé à utiliser la compression pour certains artéfacts acheminés vers la mémoire vive (*Voir FireEye [2019b]*). On était alors à l'époque de Windows 8.1. FireEye est une compagnie américaine œuvrant dans le domaine de la cybersécurité (*Voir Wikipédia [2020a]*). Le 18 décembre 2018, FireEye ont rendu disponible le code (*Voir FireEye [2019a]*) pour permettre à Volatility d'interpréter la mémoire vive des ordinateurs fonctionnant sous Windows 10. Cette fourche permet d'utiliser tous les modules associés à Volatility.

Comme il est spécifié par Volatility [2020a], ce logiciel à code ouvert et libre de droits est détenu par un organisme sans but lucratif. Pendant notre recherche, nous avons utilisé Volatility pour analyser la mémoire vive de l'ordinateur portatif (opéré par un système d'exploitation Windows 10) sur lequel Cortana a été exécuté.

### 3.4. Autopsy SleuthKit et la fourche de BlackBag Technology

Ce logiciel permet d'analyser en profondeur les images forensiques[5] créées en format e01 ou rawdd. Il indexe le contenu des images forensiques, sépare les artéfacts selon leur nature, décortique certains fichiers pour mettre en évidence certains éléments constitutifs (par exemple les images jpg encastrés dans les fichiers pdf), ségrégue les disparités signature-extension, les appariements d'empreintes numériques et de recherche par mot-clé ou par expression régulière et beaucoup d'autres choses. Il construit aussi la frise chronologique (« timeline ») des artéfacts trouvés sur le média traité.

Autopsy Sleuth Kit a présenté une difficulté car le répertoire résultant est difficilement transférable sur un média autre que celui où les résultats du traitement sont sauvegardés. En effet, ce logiciel créée une arborescence si profonde qu'un simple copier-coller aboutit à un résultat désastreux. Pour pallier ce problème, nous avons créé une copie forensique bit-à-bit du média contenant la sauvegarde du traitement et l'avons restaurée sur un autre média de taille plus pratique.

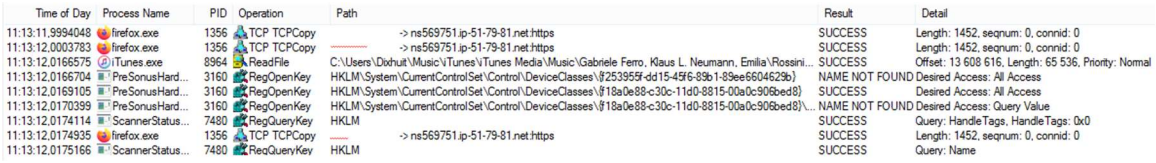
Autopsy [2020] indique « *Autopsy is a digital forensics platform and graphical interface to The Sleuth Kit and other digital forensics tools* ». Nous avons utilisé Autopsy pour traiter le contenu de l'image forensique du disque dur utilisé par l'ordinateur d'où nous avons énoncé les requêtes pour Cortana.

### 3.5. Process Hacker

Process Hacker est un logiciel permettant d'observer en direct le démarrage, la phase active et l'arrêt des processus, des services, des activités réseau et des activités disque. D'un double-clic on peut accéder au détail d'un item de la liste. Le logiciel permet d'effectuer une sauvegarde des items listés.

[5] Copies bit-à-bit

ProcessHacker [2020] se décrit comme « *A free, powerful, multi-purpose tool that helps you monitor system resources, debug software and detect malware* ». Nous avons utilisé Process Hacker pour déterminer quels processus et services étaient utilisés par Cortana, quelles modifications étaient faites sur le disque dur au moment de la requête et quelles activités réseau survenaient au moment de la requête. Grâce à ce logiciel, nous avons pu confirmer et préciser une partie des informations obtenues de Volatility, notamment la séquence de démarrage des processus lors d'une requête soumise à Cortana par un utilisateur.



| Time of Day | Process Name | PID | Operation | Path | Result | Detail |
|---|---|---|---|---|---|---|
| 11:13:11.9994048 | firefox.exe | 1356 | TCP TCPCopy | -> ns569751.ip-51-79-81.net:https | SUCCESS | Length: 1452, seqnum: 0, connid: 0 |
| 11:13:12.0003783 | firefox.exe | 1356 | TCP TCPCopy | -> ns569751.ip-51-79-81.net:https | SUCCESS | Length: 1452, seqnum: 0, connid: 0 |
| 11:13:12.0166575 | iTunes.exe | 8964 | ReadFile | C:\Users\Dixhuit\Music\iTunes\iTunes Media\Music\Gabriele Ferro, Klaus L. Neumann, Emilia\Rossini... | SUCCESS | Offset: 13 608 616, Length: 65 536, Priority: Normal |
| 11:13:12.0166704 | PreSonusHard... | 3160 | RegOpenKey | HKLM\System\CurrentControlSet\Control\DeviceClasses\{f253955f-dd15-45f6-89b1-89ee6604629b} | NAME NOT FOUND | Desired Access: All Access |
| 11:13:12.0169105 | PreSonusHard... | 3160 | RegOpenKey | HKLM\System\CurrentControlSet\Control\DeviceClasses\{f18a0e88-c30c-11d0-8815-00a0c906bed8} | SUCCESS | Desired Access: All Access |
| 11:13:12.0170399 | PreSonusHard... | 3160 | RegOpenKey | HKLM\System\CurrentControlSet\Control\DeviceClasses\{f18a0e88-c30c-11d0-8815-00a0c906bed8}\... | NAME NOT FOUND | Desired Access: Query Value |
| 11:13:12.0174114 | ScannerStatus... | 7480 | RegQueryKey | HKLM | SUCCESS | Query: HandleTags, HandleTags: 0x0 |
| 11:13:12.0174935 | firefox.exe | 1356 | TCP TCPCopy | -> ns569751.ip-51-79-81.net:https | SUCCESS | Length: 1452, seqnum: 0, connid: 0 |
| 11:13:12.0175166 | ScannerStatus... | 7480 | RegQueryKey | HKLM | SUCCESS | Query: Name |

Figure 2-1 – Affichage fourni par Sysinternals Process Monitor

### 3.6. Sysinternals Process Monitor/Explorer/Dump

Ces trois outils de Microsoft Sysinternals sont conçus pour extraire de l'information détaillée sur les processus et éléments affiliés lancés à partir d'un Windows. Ces applications existent en version 32 et 64 bits. Ce sont des applications portables (ne nécessitant pas d'installation). D'une certaine façon, ces trois logiciels sont semblables à ce que fait Process Hacker. Process Monitor affiche la liste des actions prises par le système d'exploitation (Voir Figure 2-1) à cinq égards : lectures/écritures dans la base de registre, activités système, activités réseau, processus et sous-processus, événements. D'un double-clic sur une ligne, on peut afficher le fin détail de la ligne, notamment les dll et les exécutables reliés à l'événement. L'affichage se fait par défaut par ordre temporel et l'affichage nous livre cet aspect jusqu'à la centaine de nanoseconde près[6]. Une autre fonctionnalité intéressante de ProcMon est qu'on peut sauvegarder l'historique des événements en format natif (PML), CSV ou XML, ce qui fait qu'on peut récupérer ces données et les traiter à l'aide d'autres logiciels (notamment avec Excel).

Process Explorer affiche l'arborescence des processus. Il permet à l'observateur de connaître les liens entre les différents processus, leurs dépendances. Il est parfois utilisé en sécurité pour repérer

[6] La raison étant que Microsoft utilise pour ses systèmes de fichiers un format de date nommé… Microsoft qui est un nombre de 64 bits indiquant le nombre de centaines de nanosecondes écoulées depuis le 1er janvier 1601.

les processus anormaux pouvant représenter des applications malveillantes. Process Dump permet de faire une sauvegarde d'un ou plusieurs processus ciblés. Il s'agit d'un logiciel en mode console.

Le principal inconvénient de ces logiciels est qu'ils sont très voraces en ressources (mémoire, processeur, disque). Cette voracité a fait en sorte que lors des deux premières séances expérimentales (que nous désignerons sous le vocable de « rondes ») nous n'ayons pas pu les utiliser de façon concomitante avec Cortana et FTK Imager. Ceci a créé la nécessité de tenir une troisième ronde expérimentale.

Sysinternals [2019] indique « *Whether you're an IT Pro or a developer, you'll find Sysinternals utilities to help you manage, troubleshoot and diagnose your Windows systems and applications* ». Sysinternals, c'est 78 applications gratuites. Lors de notre recherche, nous avons utilisé ProcMon et ProcExp lors de la ronde 4 sur l'ordinateur où se déroulait notre expérience pour récupérer les informations relatives à l'utilisation des processus et sous-processus.

### 3.7. Mandiant Redline

Redline réalise les mêmes fonctions que Volatility en interface graphique. De plus, il est en mesure de repérer les menaces persistantes avancées (« APT », « advanced persistent threat »). Il prend en intrant les copies de la mémoire vive et en fait le traitement.

FireEye [2019c] affirme « *When hosts are suspected of being compromised or infected Redline acts like cyber security adrenaline, rapidly accelerating the triage process while simultaneously supporting in-depth, real-time memory analysis* ». Pendant notre recherche, nous avons utilisé FireEye pour confirmer certains résultats de Volatility, notamment la structure des processus utilisés par Windows lorsque Cortana est utilisé.

## 4. Utilitaires

### 4.1. Long Path Tool

Ce logiciel permet de copier une arborescence profonde présente sur un média lu par Windows. En effet, Windows est incapable de lire un chemin s'il dépasse la longueur limite de 255 caractères. Long Path Tool contourne ce problème et peut lire, copier et coller les éléments dont la longueur du chemin dépasse 255 caractères. Long Path [2007] affirme que « *File name too long? Long Path Tool deletes, copies and renames long path files/folders* ».

### 4.2. Microsoft Excel combiné à Kutools

Nous avons utilisé Excel pour concaténer et trier certains résultats issus de Volatility. Grâce à ce logiciel, nous avons pu donner un sens aux giga-octets d'informations tirées la mémoire vive de l'ordinateur. Kutools est un module d'extension à Excel qui épargne du temps sur les manutentions de données à faire sur les feuilles Excel. Par exemple d'effectuer une rotation de 90 degrés afin que les lignes deviennent des colonnes et les colonnes des lignes.

## 5. Conclusion

Ces logiciels nous ont permis de retrouver efficacement des informations se rapportant à notre sujet de recherche. Mais bien avant de débuter cette recherche, certains faits avaient déjà été relevés par d'autres chercheurs. Les deux prochains chapitres couvriront des faits pertinents se rapportant aux assistants numériques personnels Cortana et au système d'exploitation qui les supporte.

# CHAPITRE 3 – ÉTAT DE L'ART – ASSISTANTS NUMÉRIQUES PERSONNELS

## 1. Introduction

### 1.1. Anecdote de mise en contexte

Roquette [2020] de Radio-Canada Alberta relate une expérience édifiante réalisée par l'université d'Alberta. Quatre assistants numériques personnels (Google Home, Alexa d'Amazon, Cortana de Microsoft et Apple Siri) ont été soumis à un test de compréhension et de réponse à des 123 questions relatives à des premiers soins. Les questions étaient basées sur le guide de la Croix-Rouge. Elle nous fait part d'une situation :

> « *Dis, Siri, j'ai mal à la poitrine.* »
> *Réponse de Siri : « Je n'ai pas de corps. »*
> *« Ce type d'échange illustre à quel point les assistants vocaux sont encore loin de pouvoir aider à prodiguer des premiers secours en cas d'urgence.* »

Google a compris 98% des questions et a donné 56% de bonnes réponses. Alexa a donné 19% de réponses correctes et les deux autres ont eu une performance trop médiocre pour que les réponses puissent être analysées. Ceci illustre que nous ayons dû, lors de notre expérience, renoncer à poser certaines questions.

### 1.2. Rappel – Types d'assistants numériques personnels

Les assistants numériques personnels peuvent se diviser en catégories selon leur nature. Il y a les assistants numériques personnels exécutés sur un ordinateur standard (notamment Cortana de Microsoft), ceux installés dans un appareil conçu à cette fin (par exemple Alexa d'Amazon), ceux exécutés sur un téléphone cellulaire (comme le Bixby de Samsung) et ceux s'exécutant dans le nuage (comme le « Digital Assistant » d'Oracle). La présente recherche ne porte que sur le Cortana de Microsoft.

## 2. Généralités au sujet de Cortana

Skulkin & de Courcier [2017] mentionnent que Cortana est toujours en éveil, même lorsque l'ordinateur qui le supporte est en mode veille et qu'il peut déclencher une réaction en réponse à une occurrence prédéterminée. Les auteurs révèlent que le fichier CortanaCoreDb.dat[7] contient des informations personnelles sur l'utilisateur, notamment à l'égard du lieu où se trouvait un suspect à un moment donné[8]. Ce fichier SQLite contient aussi des données sur les rappels programmés dans Cortana notamment la liste des lieux-gâchette (la présence de l'utilisateur dans un lieu-gâchette déclenchera un rappel effectué par Cortana) qui ont été configurés par l'utilisateur.

Careless [2019] indique que Cortana fonctionne en arrière-plan, même si, au moment de l'installation, on décline l'offre d'utiliser la reconnaissance vocale de Cortana et que Cortana fonctionne même lorsqu'on fait une recherche locale (par exemple une recherche de fichiers sur le disque dur de l'ordinateur). Ces recherches locales et Internet sont enregistrées dans le fichier WebCacheV01.dat. Careless indique que Bing est le moteur de recherche utilisé par Cortana. Il mentionne aussi que les suggestions faites par Cortana lorsqu'on tape une requête, sont enregistrées dans un fichier .json localisées dans le répertoire INetCache[9].

L'auteur nous informe qu'on peut retrouver des informations dans le dossier « Recent » dans un fichier lnk[10] et dans son sous-répertoire AutomaticDestinations avec un titre de fichier dont le format est semblable à ccba5a5986c77e43.automatidDestination-ms dans le sous-répertoire …\Recent\AutomaticDestinations. Il souligne aussi que plusieurs artefacts Cortana sont localisés dans le répertoire « Packages » (à l'endroit décrit ci-haut par Skulkin & de Courcier) mais que d'autres endroits sont aussi utilisés. Compte tenu de cette multiplicité de lieux, nous pouvons affirmer qu'il est plus que probable que des informations personnelles survivront à un éventuel nettoyage du contenu du disque de l'ordinateur.

---

[7] Skulkin & de Courcier localisent ce fichier \Users\[Utilisateur]\AppData\Local\Packages\Microsoft.Windows.Cortana_xxxx\LocalState\ESEDatabase_CortanaCoreInstance\CortanaCoreDb.dat

[8] Ceci n'étant vrai, bien sûr, que si on prouve que la personne au clavier est bien le suspect. Ce dernier pourrait, par exemple, avoir fixé rendez-vous à Monsieur Untel à 15h00 puis avoir décommandé sans en avoir notifié Cortana.

[9] À l'aide de FTK Imager, ce fichier a été retrouvé dans \Users\[Utilisateur]\AppData\Local\Packages\Microsoft.Windows.Cortana_xxxx\AC\

[10] \Users\[Utilisateur]\AppData\Roaming\Microsoft\Windows\Recent

**Table 1 – Cortana artifacts and their location.**

| Artifact name | Path | Description |
|---|---|---|
| IndexedDB.edb | %Userprofile%\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\AppData\Indexed DB | Indexed data |
| CortanaCoreDb.dat | %Userprofile%\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\LocalState\ESEDatabase_CortanaCoreInstance | Cortana core data |
| WebCacheV01.dat | %Userprofile%\AppData\Local\Microsoft\Windows\WebCache | Edge browser history |
| JSON files | %Userprofile%\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\AC\INetCache\<folder #> | Web search keywords |
| Wav files | %Userprofile%\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\LocalState\LocalRecorder\Speech | Audio files |
| Jump List | %UserProfile%\AppData\Roaming\Microsoft\Windows\Recent\AutomaticDestinations\9d1f905ce5044aee.automaticDestinations-ms | Edge browser Jump List |
| Contacts.json | %Userprofile%\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\LocalState\Cortana\Upload\Contacts | Contacts details |
| Prefetch file | %SystemDrive%\Windows\Prefetch\SEARCHUI.EXE-14F7ADB7.pf | Search and Cortana application |
| Amcache.hve | %SystemDrive%\Windows\appcompat\Programs\Amcache.hve | Registry hive file |
| Cache files | %Userprofile%\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\AC\AppCache\<#> | Cortana main cache |
| RecentDocs Registry | NTUSER.DAT\SOFTWARE\Microsoft\CurrentVersion\Explorer\RecentDocs\.&input= | |
| .com/search Registry | NTUSER.DAT\SOFTWARE\Microsoft\CurrentVersion\Explorer\FileExts/.com | |
| Link files | %UserProfile%\AppData\Roaming\Microsoft\Windows\Recent\https<bing-search> | Web search URL |

Figure 3-1 – Extrait de Singh & Singh [2017] – Artéfacts Cortana et leurs localisations

Singh & Singh [2017] indiquent que Edge est le seul navigateur Internet pouvant travailler avec Cortana lorsque des recherches Internet sont demandées, directement ou indirectement, par requête d'utilisateur, avec tout ce que ça implique de traces locales. Les deux fichiers retenant le plus de données lors de l'opération de Cortana sont IndexedDB.edb[11], et CortanaCoreDb.dat[12]. Plusieurs autres fichiers sont impliqués au niveau de l'opération de Cortana. La figure 3-1 présente un aperçu d'écran tiré de l'article donnant la liste des artéfacts Cortana, liste à laquelle ils ajoutent la description de l'organisation interne de plusieurs de ces fichiers.

Jester [2018] donne une liste des artefacts d'intérêt (pour un investigateur numérique) présents sur Android lorsqu'on utilise Cortana. Les fichiers portent des noms parfois différents, mais les fonctions de ceux-ci sont les mêmes que pour Windows et Mac OS X : localisation géographique, rappels, historique des requêtes, etc.

[11] Qu'on retrouve dans trois sous-répertoires de \Users\[Utilisateur]\AppData\Local\, soit :

- Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AppData\User\Default\Indexed DB\IndexedDB.edb
- Microsoft\Internet Explorer\Indexed DB\EDGE\IndexedDB.edb
- Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\AppData\Indexed DB\IndexedDB.edb

[12] \\Users\[Utilisateur]\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\LocalState\ESEDatabase_CortanaCoreInstance\CortanaCoreDb.dat

## 3. Reconnaissance de la voix : processus local ou processus distant

Selon Bhat & al [2017], le processus de reconnaissance de la voix est un processus en 5 étapes. Premièrement, il y a traitement du signal (en local), du microphone à la carte audio, puis mise en paquet et expédition du signal sonore de la requête de l'utilisateur. Deuxièmement, la reconnaissance de la voix se fait sur un site distant. La voix est alors décortiquée en mots. Puis, il y a interprétation sémantique sur le site distant. Lors de cette étape, on assigne aux séquences de mots une signification contextuelle. Quatrièmement, des algorithmes de gestion du dialogue (sur le site distant) effectuent la correction des erreurs d'interprétation et l'interprétation finale de la requête. Finalement, il y a production de la réponse.

Domingues & Frade [2016] indiquent « *Note that cw5n1h2txyewy is a hash that represents the PublisherID of Microsoft Windows, that is, CN=Microsoft Windows, O= Microsojt Corporation, L=Redmond, S=Washington, C=US*. » et « *The search box functionality of Cortana is provided by one of the executable that exists in the Microsoft.Windows.Cortana_cw5n1h2txyewy directory: SearchUI.exe.* ». Ils indiquent aussi que Cortana fonctionne de façon persistante car même si on arrête le processus SearchUI, celui-ci redémarrera de lui-même peu après.

Microsoft [2019] indiquent que les flux vocaux des utilisateurs de leurs algorithmes de reconnaissance de la voix sont utilisés de manière agrégée pour améliorer la reconnaissance de ce service en ligne. Ils disent que l'utilisateur peut choisir de ne pas utiliser le service en ligne pour la reconnaissance vocale et de s'appuyer uniquement sur les capacités locales de son ordinateur. Enfin, ils révèlent que, si l'utilisateur autorise Cortana à le faire, les données du calendrier et de la liste des contacts sera utilisée pour améliorer le service.

## 4. Hypertrucage et attaques par voix

Nguyen et al [2020] décrit les hypertrucages (« deep fake ») comme « *technique that can superimpose face images of atarget person to a video of a source person to create a video of the target person doing or saying things thesource person does* ». Pour créer un hypertrucage on utilise des algorithmes d'apprentissage profond pour extraire de vidéos existantes les structures sous-jacentes de la figure en mouvement de la cible et de son élocution, de sa façon de parler. Pour pouvoir créer un hypertrucage, on doit donc

disposer d'une quantité minimale de vidéos où la cible parle et agit de façon naturelle. Cet article mentionne cinq applications disponibles sur Github. Donc, accessibles à monsieur Tout-l'monde.

Jafar et al [2020] quant à eux comptent une douzaine d'application. Ils rapportent une expérience utilisant deux jeux de données : Deepfake Forensics et Deepfake Vid-TIMIT dans laquelle ils comparent des algorithmes d'apprentissage profond pour débusquer les hypertrucages.

Guera et Delp [2018] affirment que « …*these realistic fake videos are used to create political distress, blackmail someone or fake terrorism events* ». Petalbert [2019] rapporte que le Parti socialiste Anders de Belgique a utilisé un hypertrucage du président américain Trump où le faux-président dénigrait les belges pour leur décision de continuer à se conformer à l'Accord de Paris sur le climat, créant ainsi un malaise entre la Belgique et les États-Unis.

Carlini et Wagner [2018] parlent d'une expérience réussie où des sons inaudibles sont combinés à une phrase quelconque « A » et soumise à une application de reconnaissance vocale qui l'interprétera comme étant une phrase « B » totalement différente de la phrase « A ».

## 5. Enjeux relatifs à la vie privée

Nous partons de l'affirmation que les assistants numériques personnels présentent une menace à la vie privée de leurs utilisateurs. Certains chercheurs ont fait des recherches de nature sociale et d'autres de nature technique.

Abdi et al [2019] font état de l'ignorance des utilisateurs quant au lieu, en local sur l'appareil ou dans le nuage ou sur un site web appartenant au fabricant de l'appareil, où sont stockées les données confiées à leur assistant numérique personnel. Leur recherche mentionne aussi que les utilisateurs manquaient d'informations sur les bonnes pratiques permettant de sécuriser leurs appareils et leurs données. Ce manque de connaissances fait en sorte que les utilisateurs mettent en place des stratégies de sécurité amenant une utilisation non optimale des capacités de l'appareil. Notamment en désactivant les commandes passées vocalement ou en utilisant un ordinateur pour compléter un achat passé vocalement à leur haut-parleur futé.

Pour Ramokapane et al. [2019], une partie du problème se rapporte aux fonctionnalités activées par défaut par les constructeurs d'appareils cellulaires. Selon leur étude, 37% des utilisateurs ne

connaissent pas les fonctionnalités de protection de la vie privée disponibles sur les appareils qu'ils utilisent. 41% connaissent ces fonctionnalités mais ne se sentent pas concernées par ces mesures de protection.

Xuejing et al [2018] ont démontré qu'il était possible d'insérer des commandes vocales dans des chanson. Ces commandes vocales étaient inaudibles pour l'humain et faisaient en sorte d'activer l'assistant numérique personnel pour qu'il exécute certaines tâches. Notamment d'appeler un certain numéro de téléphone ou d'effectuer un paiement sur une carte de crédit.

Mhaidli et al. [2020] ont testé une méthode faisant en sorte que l'assistant numérique personnel (un haut-parleur futé) ne réponde que si on regardait ou parlait dans sa direction. Un capteur Kinect a été relié au haut-parleur et a servi pour la capture du son et de l'image. Les auteurs rapportent que :

> « *In 57% of the measurements, participants managedto activate the device within one attempt; and withinthree attempts for 88% of the measurements.* »

Conclusion : Certes, les fabricants d'assistants numériques personnels pourraient en améliorer la sécurité, mais les utilisateurs pourraient avoir un impact sur la protection de leurs propres données privées en se donnant la peine de fouiller les manuels des appareils pour y découvrir ce qui peut être fait pour leur propre sécurité.

# CHAPITRE 4 – RAPPEL DE CERTAINES NOTIONS – SYSTÈMES D'EXPLOITATION

## 1. Système d'exploitation

Un système d'exploitation est un logiciel chargé de gérer les interactions entre le niveau matériel de l'ordinateur et les logiciels qui y sont installés. Il gérera aussi les interactions entre les différents dispositifs (« devices ») matériels et entre les différents logiciels. Dans ce but, le système d'exploitation acceptera les demandes des un et des autres pour les rediriger au bon endroit.

Lorsqu'un logiciel (Cortana, par exemple) démarre, il doit réserver sa place dans le grand orchestre que constitue un ordinateur. Cette réservation se fait par la création de processus, de fils et de descripteurs. Pour chacun, le système d'exploitation créera un espace dans la mémoire vive (RAM) ou dans la mémoire de débordement (« pagefile.sys » dans le cas de Windows), espace qui sera libéré lorsque la tâche à accomplir sera terminée.

## 2. Mémoire vive

LA référence en analyse forensique de la mémoire vive est « The Art of Memory Forensics » de Ligh et al [2014], référence ci-après désignée sous le nom de « Ligh ». La référence en matière de système Windows est Yosifovich et al [2017], référence ci-après désignée sous le nom de « Yosifovich ».

Ligh définit la mémoire vive, dite « mémoire RAM » (RAM pour « Random Access Memory ») comme un espace de stockage du code et des données utilisées activement par le processeur. Lorsque l'exécution d'un programme ou d'un processus requiert que des données soient stockées temporairement, celle-ci sont placées en mémoire vive et récupérées au besoin. Ligh indique que ce qui est placé en mémoire vive y est maintenu tant et aussi longtemps que l'espace en question est alimenté en électricité. Ajoutons que ce contenu peut être effacé, mis-à-jour ou modifié.

Afin de mettre un peu d'ordre dans la pléthore de données transmises à la mémoire vive, la plupart des systèmes d'exploitation utilisent la technique de pagination (Rouse [2013]).

Ligh indique que l'accès au contenu de la mémoire vive se fait par le biais d'espaces d'adressage. Deux types d'adresses sont utilisées. D'abord les adresses linéaires ou virtuelles, qui sont des espaces auxquels accède un programme qui s'exécute. Enfin, les adresses physiques qui sont des espaces auxquels accède le processeur. Le passage d'une adresse physique à une adresse virtuelle se fait par le biais d'un calcul basé sur les bits et non les octets (autrement dit : on morcelle les octets composant l'adresse physique pour obtenir l'adresse virtuelle). Bien que ce soit intéressant si on programme un logiciel comme Volatility, c'est peu pertinent au niveau de la présente étude qui implique l'utilisation de logiciels déjà écrits comme Volatility et ProcMon64. Par contre, cette notion aide à comprendre le fonctionnement des mémoires d'ordinateur.

Ligh avance aussi la notion de pagination. Il écrit « *Paging provides the ability to virtualize the linear address space* ». Interprétation : les données envoyées en mémoire vive sont regroupées en grappes qui, une fois transmises à la mémoire vive, sont insérées dans un espace déterminé qu'on appelle « page ». Par expérience, nous pouvons dire que pour les dispositifs (« devices ») fondés sur une infrastructure Intel 32 ou 64-bits, la taille des pages est de 4 kilooctets (4096 octets).

Yosifovich précise qu'au démarrage de l'exécution d'un programme, la fonctionnalité « gestionnaire d'adresse » octroie une adresse de mémoire virtuelle au processus puis lui fait correspondre une adresse de mémoire physique.

> Fireeye [2019] révèle que « *Until August 2013… a complete Windows memory analysis only required forensic tools to parse physical memory and fill in any missing gaps from the pagefile. In Windows 8.1 Microsoft upended this paradigm with the introduction of memory compression and a new virtual store designed to contain compressed memory* »

## 3. Processus, « handles » et « threads »

### 1.1. Processus

Selon Ligh « *A process is an instance of a program executing in memory. The operating system is responsible for managing process creation, suspension, and termination* ». Pour le reste du présent document, le mot anglais « process » sera traduit en français par « processus ».

Yosifovich dit « …*a process is a container for a set of resources used when executing the instance of the program* »

### 1.2. Threads

Le mot « fil », suggéré par l'Office québécois de la langue française, sera adopté dans ce mémoire pour traduire le mot anglais « thread ».

Pour Ligh, le fil est l'unité fondamentale de l'exécution du code d'un programme. En matière de forensique informatique, les fils peuvent fournir des informations comme l'horodatage du début de l'exécution d'un processus et son adresse de départ en mémoire. Pour Yosifovich : « *A thread is an entity within a process that Windows schedules for execution* ».

### 1.3. Handle

Le mot « descripteur » ou la locution « descripteur de processus », suggéré par l'Office québécois de la langue française, seront utilisés dans ce mémoire.

Pour Ligh : « *A handle provides the process with a unique identifier for accessing and manipulating system resources. It is also used to enforce access control to those resources and track their usage* ». Pour Yosifovich : « *These map to various system resources such as semaphores, synchronization objects, and files that are accessible to all threads in the process* ». Signalons qu'il y a toujours au moins un descripteur par processus initié.

### 1.4. Fibre

Yosifovich décrit les fibres comme des « fils poids-légers », invisible au noyau (« kernel »).

### 1.5. Relation entre processus, fils et descripteurs

Voici une illustration tirée de Ligh (Figure 4-1) :

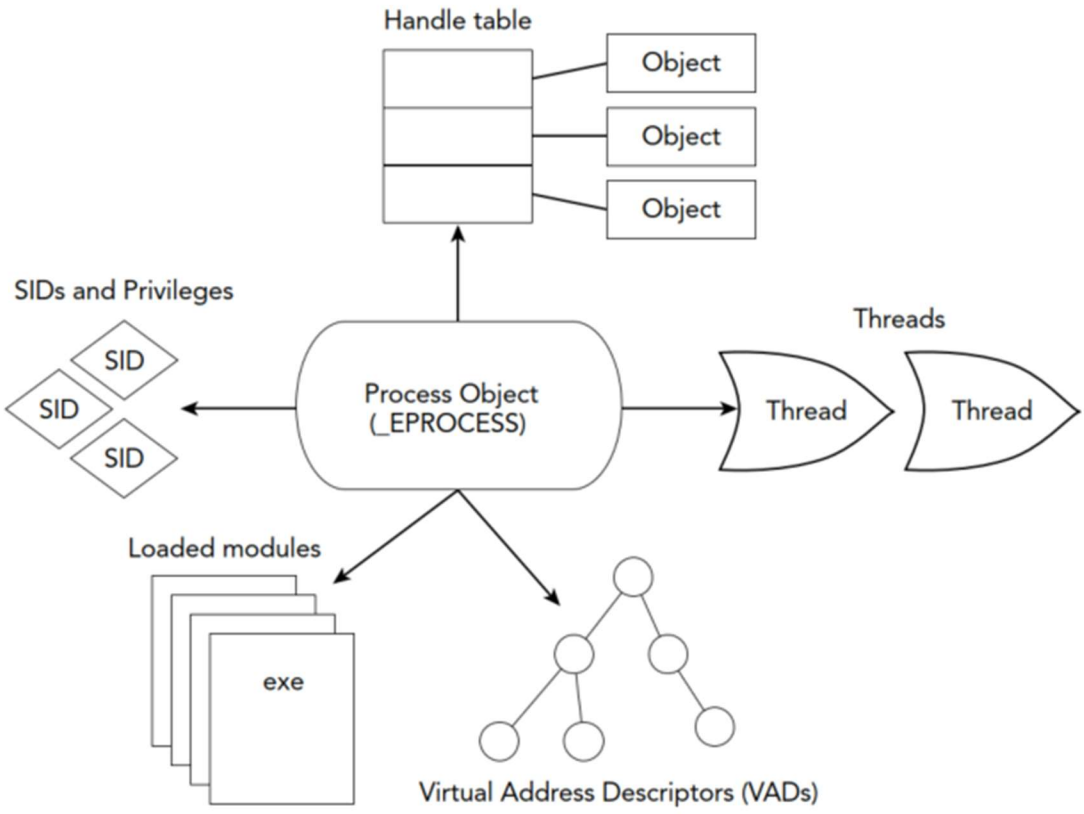



**Figure 6-1:** A high-level diagram showing basic process resources

Figure 4-1 – Extrait de Ligh – Diagramme des ressources liées aux processus

## 4. Mémoire de débordement

Typiquement, Windows 10 se sert du fichier pagefile.sys comme espace de stockage sur le disque dur de l'appareil d'où ce système d'exploitation s'exécute. Cet espace de stockage reçoit principalement les éléments de la mémoire vive qui sont accédés le moins souvent lorsque la mémoire vive est pleine (Hoffman [2018]). Contrairement à la mémoire vive, le contenu de ce fichier n'est effacé que s'il atteint en taille la limite fixée pour ce système d'exploitation et l'effacement n'est que partiel. Curieusement, en forensique, le contenu de ce fichier est mal et peu

exploité. Il en a été de même dans le cadre de ce mémoire malgré la conviction intime qu'on pourrait y trouver des trésors d'informations.

## 5. Les sons

### 5.1. Flux audio

Bhattacharjee [2019] explique sommairement le fonctionnement local du processus de reconnaissance vocale. Il rappelle que la transformation d'un signal analogue en signal digital se fait sur base d'échantillonnage et que ce dernier doit respecter le théorème de Nyquist. Ce sont ces prélèvements discrets qui sont encodés selon différents formats (les codecs). Le signal digital est alors séparé en segments et ceux-ci sont prétraités selon différentes métriques (Amplitude, fréquence, MFCC etc). De telles métriques permettent d'associer un signal sonore à une lettre ou à une diphtongue.

Amilcare [2020] explique le théorème de Nyquist-Shannon en disant que la fréquence de l'échantillonnage doit être d'au moins deux fois la fréquence du signal enregistré si on veut en restituer toutes les qualités.

Villedieu [1988] indique que l'oreille humaine entend les sons entre 15 Hz et 20 000 Hz, que les voix humaines produisent des sons entre 40 Hz (voix grave) et 1500 Hz (voix aigüe) et que les instruments de musique produisent des sons entre 16 et 16 000 Hz.

On peut donc en déduire que la fréquence d'échantillonnage minimale d'une voix humaine doit donc être de 3 kHz. Or, lorsqu'on s'enregistre sur un ordinateur, c'est plutôt une fréquence de 44,1 kHz qui est utilisée. Pourquoi? On peut faire l'hypothèse qu'afin de se simplifier la vie, les manufacturiers du domaine de l'électronique (matériel comme logiciel) s'alignent sur les 20 kHz de sons audibles dont parle Villedieu [1988] : 20 kHz * 2 =… 44,1 kHz (pourquoi pas!). Ce qui est la valeur utilisée pour enregistrer des pièces musicales sur CD/DVD.

Sergère [2015] explique en quoi consiste la définition sonore, qui s'exprime en nombre de bits. Supposons que pour chaque prélèvement sonore on veuille quantifier la fréquence et l'amplitude du son à ce moment donné. Si ces deux quantités sont exprimées à l'aide d'un seul bit chacun, on aura le choix entre fréquence haute ou fréquence basse et entre amplitude haute et amplitude faible.

Si on ajoute un deuxième bit, on pourra qualifier la fréquence de haute, mi-haute, mi-basse et basse. Si la fréquence s'exprime par le biais d'un octet, on pourra lui assigner jusqu'à 256 valeurs. Les standards utilisés par défaut en électronique est de 16, 24 ou 32 bits. Selon Sergère, une fréquence de 96 kHz est difficilement distinguable d'une fréquence de 44,1 kHz. Mais une définition de 24 bits peut être distinguée, par l'oreille humaine, d'une définition de 16 bits.

Ces facteurs (fréquence d'échantillonnage et définition) sont généralement fournis dans les intradonnées des fichiers audio.

Xiph [2005] donne la structure des données dans un fichier ogg (type de fichier utilisé par Cortana pour transmettre la voix de l'utilisateur soumettant une requête. On voit dans le tableau 4-1 ci-dessous que plusieurs facteurs sont à considérer pour construire un fichier audio.

| **Décalages** | | | |
|---|---|---|---|
| **Début** | **Fin** | **Donnée** | **Taille** |
| 0 | 3 | Signature « OggS » | 4 |
| 4 | 4 | Version de la structure de flux | 1 |
| 5 | 5 | Indicateur de type d'entête | 1 |
| 6 | 13 | Position granulaire absolue | 4 |
| 14 | 17 | Numéro de série du flux | 4 |
| 18 | 21 | Numéro de séquence des pages | 4 |
| 22 | 25 | CRC32 de la page (algorithme direct, valeurs initiale et finale du Xor de 0, polynôme générateur 0x04C11DB7) | 4 |
| 26 | 26 | Numéro de segment de page | 1 |
| 27 | FinFichier | Segment (données) | 26 |

Tableau 4-1 – Structure des données dans un fichier ogg

### 5.2. Propriétés du son

Les propriétés les plus connues de la voix sont la fréquence et l'amplitude. Mais bien d'autres facteurs sous-jacents sont présents dans la voix. Selon Castro & Depardieu « … *la voix ne crée pas qu'une seule fréquence, sinon elle pourrait produire des sons purs. Elle produit également des fréquences d'harmoniques… Le nombre et l'amplitude des fréquences d'harmoniques caractérisent le timbre… Chaque individu a un timbre qui lui est propre* ».

Julien [2013] écrit « *Un des paramètres développés pour caractériser l'aspect phonétique d'un signal est les MFCC [Mermelstein, 1976]. Le MFC (Mel-Frequency Cepstrum) utilise l'échelle de Mel, qui est une échelle de fréquences basée sur la perception humaine* ». Lors du cours « Apprentissage automatique », Desharnais et Juigné (document non publié), ont utilisé ce coefficient (MFCC) pour exploiter des données voix d'une trentaine de personnes. Ces personnes, hommes et femmes, d'accent québécois ou français, énonçaient pendant 3 minutes les chiffres 1, 2, 3. Le but de l'apprentissage automatique était que l'algorithme distingue le son prononcé (1, 2 ou 3), le sexe du locuteur ou de la locutrice et son accent. À cet effet, nous avons adapté l'algorithme « Japanese vowel », disponible chez Matlab, et nous lui avons soumis 1218 enregistrements. Pour le pan « distinction de 1-2-3 », la précision était de 98%, pour la « distinction de sexe », 94%, et pour la « distinction de l'accent, 89%.

### 5.3. Biométrie vocale

JdM [2019] nous révèle que « *Les banques, comme plusieurs autres établissements, ont aussi recours à un système d'analyse vocale pour contrer la fraude. Comme une empreinte digitale, la voix — même enrhumée! — permet d'authentifier l'interlocuteur et d'éviter le vol d'identité* ». Mais cette idée de traiter le signal sonore produit par l'humain pour communiquer n'est pas nouveau. Cazade [1999] mentionne qu'IBM faisait déjà des recherches sur la reconnaissance vocale (comprendre : traduire les sons des paroles en textes) dès 1984. La préoccupation principale à ce moment n'était donc pas l'identification d'une personne en se basant sur sa voix.

CRIM [2018] nous donne une définition de la biométrie vocale : « … *domaine scientifique et technologique qui vise à développer des applications permettant de vérifier l'identité d'une personne seulement grâce à sa voix* ». La reconnaissance du locuteur (la personne qui parle) est quelque chose que nous (les humains) réalisons sans difficulté ni effort conscient. Dans son cours sur les communications sans fils, le Professeur Ajib Wessam de l'UQAM explique que cette reconnaissance est plus difficile lorsqu'on parle à une personne pour la première fois via un appareil spécifique. Il explique que la captation des sons émis par le locuteur se fait par échantillonnage, que lors de la transmission certains octets du signal sonore sont sujet à erreur et que la transformation du signal digital en signal analogique par le récepteur sont trois distorsions affectant le son de la voix du locuteur. Il n'est pas illogique de penser que l'enregistrement de la voix par Cortana et son envoi par Internet à un serveur de traitement sont victimes des mêmes distorsions.

CRIM [2018] explique qu'afin d'identifier un individu par sa voix, on comparera un l'échantillon de sa voix recueilli au temps t=0 (échantillon d'inscription) à un échantillon recueilli ultérieurement au temps t=1 (échantillon d'authentification). On calculera alors un score de similarité en tenant compte d'un niveau de concordance et en le comparant au seuil accepté on décidera si l'échantillon d'authentification provient de la même personne que l'échantillon d'inscription. De plus, cet auteur explique aussi que l'identification se fait par le biais de facteurs sous-jacents à la voix, facteurs qui font l'objet de mesures.

# CHAPITRE 5 – PROTOCOLES EXPÉRIMENTAUX

## 1. Introduction

Ce chapitre présente l'infrastructure matérielle de cette recherche ainsi que les actions qui ont été posées afin d'obtenir les informations qui seront présentées dans les derniers chapitres.

## 2. Appareils utilisés

Deux ordinateurs ont été utilisés pour réaliser la partie expérimentale, ordinateurs que nous désignerons par les alias suivants : GMO et Lenovo. Ces ordinateurs avaient les propriétés indiquées dans le tableau 5-1 ci-dessous.

| | **GMO** | | **Dokapoutinn** |
|---|---|---|---|
| **Marque** | Toshiba | | Lenovo |
| **Modèle** | Satellite Pro P300 | | IdeaPad U310 |
| **Processeur** | Intel DuoCore T9550 @ 2.66GHz | | Intel Core i3-3217U CPU @ 1.80GHz |
| **Mémoire vive** | DDR2 SO-DIMM PC2-6400 8 Go | | SODIMM DDR3 1600 MHz 4Gio |
| **Moniteur** | 17 po | | 13 po |
| **Disque dur A** | SSD 240 Go | | SSD 120 Go |
| **Disque dur B** | SSD 1 000 Go | | N/A |
| **Système d'exploitation** | Windows 10 Pro 64 bits Build 17134 | | Linux Ubuntu 18.04.3 LTS Bionic Beaver |
| **Carte Wi-Fi** | Intel WIFI Link 5100AGN | | Intel Centrino Wireless-N 2200 |
| **Carte filaire** | Marvell Yukon 88E8072 PCI-E Gigabit Ethernet | | Realtek RTL810xE PCI express fast Ethernet |

Tableau 5-1 – Caractéristiques des appareils utilisés

## 3. Généralités

### 3.1. Approche générale

L'objectif final est de déterminer l'impact d'une requête soumises à l'assistant numérique personnel Cortana via l'examen du contenu de ses communications Internet, de son disque dur et de ses mémoires vive et de débordement. Cette détermination visait la réponse de Cortana aux requêtes

soumises par son utilisateur et si cette réponse contenait ou créait des données pouvant être assimilées à des renseignements relatifs à la vie privée de l'utilisateur. Ces données peuvaient être de trois natures : celles émises/reçues par l'ordinateur exécutant l'assistant numérique personnel, celles écrites/lues sur son disque dur et celles écrites/lues dans sa mémoire vive.

Nous pressentions que le traitement du langage naturel contenu dans la requête n'était pas fait en mode local. De plus, dans une phase de tests préparatoires, nous avons constaté que certaines réponses des assistants numériques personnels étaient simplement des pages Internet. Nous en avons donc conclu que l'ordinateur supportant Cortana échangeait des données par le biais de sa carte réseau. Pendant la phase préparatoire, nous avons déterminé que plusieurs artéfacts découlant de la requête et de sa réponse étaient inscrits sur le disque dur, généralement dans des répertoires furtifs. Utiliser un ordinateur, c'est utiliser sa mémoire vive. La question n'est donc pas de savoir si, mais plutôt de savoir quelles données relatives à la requête ou à sa réponse restent dans la mémoire vive.

### 3.2. Objectifs généraux

La présente recherche vise trois objectifs généraux, tous relatifs à la soumission d'une requête à Cortana. D'abord, nous voulons recueillir les flux de communication réseau de l'ordinateur offrant les services d'assistant numérique personnel. Ensuite, nous voulons recueillir les écritures et les lectures faites sur le disque dur de l'ordinateur où Cortana s'exécute. Enfin, nous voulons recueillir les écritures et les lectures faites dans les mémoires vive et de débordement de l'ordinateur où Cortana s'exécute.

### 3.3. Limitations liées aux outils utilisés pour réaliser les objectifs généraux

L'ordinateur utilisé pour expérimenter Cortana avait des ressources limitées empêchant qu'on recueille en direct les flux d'écriture et de lecture vers le disque dur et la mémoire vive. Le temps d'analyse pour un disque dur de la taille de GMO est d'environ trois jours pour chaque copie forensique bit-à-bit. Le temps d'analyse pour une copie forensique de mémoire vive de la taille de celle de GMO est d'environ une semaine pour chaque copie forensique. Nous avons recueilli une copie forensique de la mémoire vive après chaque requête, ainsi qu'une copie forensique bit-à-bit du disque dur de l'ordinateur de requêtes après la fin d'une ronde expérimentale.

### 3.4. Prise de notes manuscrites

Antoine Alriquet, Thomas Maillet et moi-même avons pris en note nos observations et commentaires dans un carnet. Lors des rondes expérimentales, nous avons plutôt opté pour des feuilles volantes sur lesquelles étaient inscrites les requêtes telles qu'elles devaient être dites. Nous y avons reporté l'heure de passage de la requête à la minute près. Ces deux procédures sont à la fois forensiques et scientifiques et nous ont bien servi.

## 4. Requêtes pour les assistants numériques personnels

### 4.1. Langue des requêtes

Parce que le système d'exploitation installé sur GMO est en version anglaise, les requêtes ont été rédigées en anglais. Elles sont livrées ci-dessous avec leur numéro de repère et l'heure à laquelle les requêtes ont été soumises le 23 septembre 2019. Comme nous le verrons, il y a eu une ronde préliminaire le 9 septembre 2019, mais l'analyse s'est surtout appuyée sur l'expérience du 23 septembre car les résultats recueillis étaient complets pour les motifs exposés plus loin dans ce chapitre.

### 4.2. Types de requête

Lors de la phase d'analyse, nous avons voulu savoir s'il est possible de déterminer le type de requête passée par l'utilisateur, simplement à l'aide des résultats obtenus de Volatility (analyse de la mémoire vive). Si tel était le cas, on aurait la possibilité d'espionner un utilisateur à l'aide du contenu de sa mémoire vive ou de sa mémoire de débordement [13]. Les requêtes ci-dessous ont ensuite été séparées en quatre types. Les indicateurs « Type » de la deuxième colonne reflètent les types. Nous avons aussi ajouté un indicateur pour indiquer qu'il ne s'agit pas d'une requête mais d'une instruction pour aider à l'exécution de l'expérience. Quatre (4) types de requêtes ont été créées. Le premier type est constitué des appels à des utilitaires locaux (UL). Nous voulions déterminer si Cortana était en mesure de démarrer l'application visée par la requête et d'interpréter correctement la directive de l'utilisateur. Le deuxième type de requête concerne la demande de renseignements

[13] En Windows, les mémoires de débordement sont des mémoires utilisées lorsque la mémoire vive est pleine ou désactivée. Exemples : pagefile.sys, hiberfil.sys…

courants sur Internet (RC), notamment en ce qui se rapporte à l'actualité ou à la météo. Le troisième type est la recherche de renseignements locaux (RL). Ce type inclut l'appel à l'utilitaire local pertinent. En supposant que l'assistant numérique personnel interprète correctement la requête de l'expérimentateur, est-il en mesure d'ouvrir l'application pertinente, d'en extraire l'information demandée et de retourner un résultat pertinent et cohérent? Enfin, il y a les tests d'intelligence ou de discrimination (I/D). Est-ce que l'assistant numérique personnel est en mesure de répondre à une question inattendue dont la réponse nécessite réflexion? Si on demande à l'assistant numérique personnel de distinguer une personne (et une seule), est-elle en mesure de s'exécuter correctement? Cortana offre la fonctionnalité « Ma voix seulement ». Cette fonctionnalité permet à un utilisateur d'interdire à Cortana de répondre à une requête passée par une voix autre que la sienne.

### 4.3. Requêtes

Dans le tableau 5-2 ci-dessous se trouvent les requêtes dans l'ordre où elles ont été soumises à Cortana (en anglais puisque le Windows utilisé était installé en anglais). Elles sont numérotées de 1 à 24 avec sous-questions (en lettres minuscules) le cas échéant. Dans ce mémoire, les requêtes qui ont été traitées et qui ont eu un effet sur l'écosystème numérique sont renumérotées en ajoutant 100 au numéro de la requête. La requête 1a devient à ce moment la requête 101a. Ce changement de notation permet de distinguer la requête tirée de liste du tableau 5-2 (numérotée entre 1 et 23) des éléments résultant du traitement des artéfacts recueillis lors des expériences.

| #req. | Type | Soumis à | Requête énoncée |
|---|---|---|---|
| **1** | **UL** | | **Alarm, timer, alarm-clock:** |
| a | UL | 10h11:05 | Set the timer in 1 minute |
| b | UL | 10h15:50/ 10h20:50 | Set an alarm in 5 minutes |
| c | UL | 10h23:50/ 10h24:50 | Set the alarm-clock at [give an hour time in 1 minutes] |
| d | UL | 10h27:25 | Set the timer in 1 minute [wait 15 seconds and] cancel timer |
| e | UL | 10h29:40 | Set an alarm in 5 minutes [wait 15 seconds and] cancel timer |
| f | UL | 10h32:15 | Set the alarm-clock at [give an hour time in 4 minutes] [wait 15 seconds and] cancel timer |
| **2** | **UL** | 10h40:55 | **Countdown from 9 to 0** |
| **3** | **RC** | 10h55:05 | **What is the foundation date of Canada?** |
| **4** | **RL** | | **Contacts:** |
| a | RA | 11h03:10 | Create a new contact – Name Dupont, First name Jean, Birthdate 25 march 2000, Phone number 514-555-1234, Address 5678, Amherst Street, City Montreal, email desharnais@cfij.org |
| b | RL | 11h13:25 | What is the birth date of Jo Bleau? |
| c | RL | Échec | What is the birth date of Jo Binne? |

| | | | |
|---|---|---|---|
| d | RL | Échec | What is the phone number of Jo Binne? |
| e | RL | Échec | What is the phone number of Jo Bleau? |
| f | RL | Échec | What is the address of Jean Dupont? |
| g | RL | Échec | What is the address of Djjîîînne Doupontt? |
| **5** | **UL/RL** | | **Emails:** |
| a | UL | 11h18:40 | Open email application |
| b | RL | 11h21:55 | Create an email for Djjîîînne Doupontt |
| c | UL | Échec | [Content of the email:] Dear Djjîîînne Test one two three. Regards. Alfred |
| d | UL | Échec | Send email [Check if email is received by Jean. If so, reply to it. Then:] |
| e | UL | Échec | Check received email |
| f | UL | Échec | Delete email |
| g | UL | 11h23:50 | Close email application |
| **6** | **UL** | | **Text document:** |
| a | UL | 11h27:15 | Create a text document |
| b | UL | Échec | This is a transmission from Iceland |
| c | UL | Échec | Save document in My Document folder as transmission.txt |
| d | UL | Échec | Delete transmission.txt |
| **7** | **UL/RL** | | **Agenda:** |
| a | UL | 11h33:40 | Open my agenda |
| b | RL | 11h35:50 | Create an appointment, next Tuesday with dentist at 13h00 |
| c | RL | 11h39:00 | Create an appointment, next Friday with doctor at 12 o'clock |
| d | RL | 11h41:40 | When is my appointment with the dentist? |
| e | RL | 11h44:50/ 11h48:00 | Cancel my next appointment with the doctor |
| **8** | **UL** | 11h55:50 | **Open calculator** |
| **9** | **UL** | 11h59:25 | **Open Paint** |
| **10** | **UL/RL** | | **Find content in Gallery:** |
| a | UL | 12h02:10/ 12h04:20 | Open my gallery |
| b | RL | Échec | Find a photo of a flower in my gallery |
| c | RL | Échec | Find a photo of a cow in my gallery |
| d | UL/RL | Échec | Delete the photo of a cow |
| e | UL | 13h38:25 | Open my web cam |
| f | UL | 13h40:05 | Take a picture of myself |
| g | UL | 13h43:35 | Close my gallery |
| **11** | **RC** | 13h45:35 | **Display Donald Trump's last tweet** |
| **12** | **RC** | 13h50:20 | **How high and how deep is lake Titicaca?** |
| **13** | **RC** | | **Jokes:** |
| a | RC | 13h52:05 | Tell me a joke for kids |
| b | RC | 13h54:05 | Tell me a joke for kids in French |
| c | RC | 13h57:25 | Raconte-moi une histoire |
| d | RC | Échec | Raconte-moi une histoire pour enfants |
| **14** | **RC** | 13h59:35 | **Donald Trump, est-ce qu'il décida d'aller à Boston?** |
| **15** | **RC** | | **How many murders there was in Montreal in:** |
| a | RC | S/O | 2019 |
| b | RC | S/O | 2018 |
| c | RC | 14h01:55 | 2017 |
| **16** | **RC** | 14h03:55 | **Who was Jean Brillant?** |
| **17** | **RC** | Échec | **What is the birthdate of de Maisonneuve? [Pronounce : Maille-Zônn-Nouvv]** |
| **18** | **RC** | 14h06:30 | **What is the birthdate of Paul Chomedey de Maisonneuve? [Pronounce : Pol Tchô-Mé-Dayy deu Maille-Zônn-Nouvv]** |
| **19** | **RC** | | **Inhabitants:** |
| a | RC | 14h09:00 | How do you call a person living in Montréal? [Pronounce Montréal à-la-Québécoise] |
| b | RC | 14h11:30 | How do you call a person living in Montreal? [Pronounce it: Ma-onttt-ri-al] |

| | | | |
|---|---|---|---|
| c | RC | | How do you call a person living in Trois-Rivières? [Pronounce Trois-Rivières à-la-Québécoise] |
| d | RC | | How do you call a person living in Three-Rivers? |
| **20** | **RC** | 14h13:30 | **How high over the level of St-Lawrence River is the cliff in Quebec City?** |
| **21** | **RC** | | **Meteo:** |
| a | RC | 14h16:00 | What is the forecast for today? |
| b | RC | S/O | What is the forecast in Longueuil for today? |
| c | RC | 14h18:30 | What is the forecast in Paris for today? |
| **22** | I/D | Échec | **If I cut a piece of butter in two, I will have two pieces of butter. If I cut a table in two, what will I have? No more table, one table or two tables?** |
| **23** | | | **After having activated the "My Voice Only" function, make the following request to Cortana: How many petals a daisy has?** |
| a | I/D | 14h29:40 | Sylvain makes the query |
| b | I/D | 14h32:10 | Thomas makes the query |
| c | I/D | 14h34:40 | Antoine makes the query |
| d | I/D | S/O | Christine makes the query |
| | | 14h22:15 | Partial queries: Hey… Hey Cor… Hey Cortttt… Hey Corta… Cortana<br>Followed each time by the query "Who is Mickey Mouse?" |

Tableau 5-2 – Liste des requêtes soumises à Cortana

## 5. Ronde 1 : Requêtes sur GMO/Cortana

### 5.1. Objectifs

Afin d'atteindre les objectifs généraux, nous soumettrons à Cortana les requêtes prédéterminées en utilisant l'ordinateur GMO. Nous pourrons alors recueillir les flux de communication envoyés vers Internet contenant les requêtes vocales de l'expérimentateur et commencer à les interpréter ainsi que les flux de communication reçus d'Internet contenant les réponses aux requêtes et commencer à les interpréter. Nous recueillerons aussi une copie forensique du contenu des mémoires vive et de débordement de GMO après chaque requête. À la fin de la ronde de requête, nous recueillerons une copie forensique bit-à-bit du disque dur de GMO.

### 5.2. Date

9 septembre 2019 de 9h00 à 21h00 au laboratoire de CFIJ.

### 5.3. Protocole

Deux configurations ont été testées successivement. La première configuration a impliqué de connecter GMO par sans-fil sur une borne équipée d'une clé cellulaire 3G+, avec Dokapoutinn écoutant et recueillant les signaux Wi-Fi provenant et à destination de GMO. Cette configuration

a été abandonnée car la maintenir impliquait un investissement financier déraisonnable compte tenu de la disponibilité de solutions gratuites.

La deuxième configuration consistait à brancher Dokapoutinn sur réseau filaire, le qualifier comme borne Internet et connecter GMO à cette borne, Dokapoutinn recueillant les signaux passant par sa carte Wi-Fi.

Certaines actions préalables à l'expérience ont été posées. D'abord, nous avons aseptisé le disque dur devant supporter le Windows utilisé. Puis nous avons restauré l'image forensique GMO0 afin de retourner à la configuration d'octobre 2018 (date à laquelle un pré-test a été effectué dans le cadre d'un cours sur les données massives). Nous avons ensuite installé les mises à jour Microsoft et configuré les comptes de Microsoft de GMO, notamment le compte de courriel. Nous avons aussi ajouté les adresses de Jos Binne et Jos Bleau et importé des photos de vaches et de fleurs. Nous avons enfin créé une image forensique bit-à-bit de GMO00 afin que, plus tard, nous puissions retourner dans le même état qu'à ce moment-là, effaçant toute traces laissées par des opérations ultérieures. **Ce groupe de procédures a fait en sorte qu'au début de chaque ronde expérimentale, nous avions un environnement physique identique et exempt des données provenant de la ronde précédente**.

Le jour de l'expérience, nous avons connecté physiquement Dokapoutinn sur un commutateur réseau connecté directement sur modem Internet, démarré GMO et Dokapoutinn et avons connecté le premier sur le second. Nous avons ensuite procédé aux cycles « Démarrage de l'interception, requête, fin d'interception, cueillette de mémoire vive GMO » suivis d'une copie forensique après la requête 23d).

### 5.4. Erreurs détectées et correctifs en vue de la ronde 2

Suite à l'analyse des éléments recueillis, nous avons constaté que les signaux réseau étaient chiffrés parce que l'adresse avec laquelle Cortana échange des données est une adresse https et ces échanges sont chiffrés selon le protocole TLS. La solution à ce problème était d'utiliser mitmproxy. Nous avons aussi constaté qu'une bonne partie de la mémoire vive de GMO (Windows 10 pro) était compressée et que l'application de base Volatility était incapable d'en faire l'analyse. L'utilisation de la fourche FireEye (*Voir FireEye [2019a]*) a constitué la solution à ce problème. Lorsque Autopsy fait le traitement de la copie bit-à-bit, il Autopsy créé une arborescence trop profonde pour qu'on

puisse en assurer la pérennité. La solution a été de placer les images forensiques sur des médias réservés à cet effet. Ceci a permis une ronde expérimentale #2 fructueuse.

## 6. Ronde 2 : Objectifs, Cortana, mitmproxy, Wireshark

Cette phase de la recherche s'est déroulée le 23 septembre 2019 de 8h10 à 15h00. **Nous verrons dans l'analyse que l'horodatage tient une place majeure dans notre recherche**. Les objectifs de cette phase expérimentale sont identiques à ceux de la ronde 1. Les procédés sont les mêmes sauf que dans cette phase, Dokapoutinn a servi de borne Wi-Fi et d'autorité émettrice de certificat de sécurité, permettant alors le déchiffrement des paquets IP avec Wireshark.

Deux intuitions ont vu le jour pendant les premiers moments d'analyse des artéfacts recueillis pendant la seconde ronde expérimentale l'une se rapportant à la reconstitution des flux audio et l'autre à la présence présumée de processus révélateurs de l'activité de l'utilisateur de Cortana.

Pour démontrer la possibilité de reconstituer les flux audio de la requête vocale soumise par Cortana et transmise à Microsoft, nous avons été en mesure d'extraire la charge utile des paquets réseau montants (ceux allant vers le site Internet de Cortana). Nous nous sommes posé la question suivante : Est-il possible de reconstituer les flux audios et de les placer dans un fichier qu'on pourrait faire jouer par un lecteur audio, par exemple un lecteur mpeg? Si tel avait été le cas, aurions-nous pu faire rejouer ces flux audio avec une qualité suffisante pour tromper un Cortana dont la fonctionnalité « Ma voix seulement » aurait été activée?

Pour ce qui est des processus révélateurs, nous nous demandions si Cortana utilisait des processus qui lui sont propres, des processus partagés avec d'autres application ou un savant mélange des deux? Le premier cas échéant, la question était : est-il possible de déterminer quel type de requête a été soumise simplement en examinant les processus et fils activés au moment de la requête? Par « type de requête », on entend bien sûr ce qui a été défini ci-haut, à la section 4.2 du présent chapitre. Cet objectif éclaire le soin apporté par l'équipe expérimentale à noter le moment exact auquel la question a été posée.

## 7. Ronde 3 : ProcessHacker, ProcMon, ProcExp, ProcDump

Cette phase expérimentale s'est déroulée entre janvier et octobre 2020. La ronde 3 visait à répondre à certaines questions non répondues après la ronde 2. Notamment, quels sont les éléments de la base de registre qui sont appelés par Cortana ? Dans quels fichiers du disque dur Cortana écrit-il lorsqu'il écoute la requête, traite la requête et y répond? Quelle est la structure des processus, « handles » et « threads » lorsqu'une requête est énoncée par l'utilisateur?

Process Monitor 64 (ProcMon64) a été choisi parmi quatre logiciels (ProcessHacker, ProcMon, ProcExp, ProcDump) parce qu'il livrait le plus d'informations pertinentes. Une seule requête a été utilisée pour cette phase, la requête #3 (« What is the foundation date of Canada? »). ProcMon64 nous a permis de déterminer les écritures et les lectures faites, notamment par les processus et les fils, au moment où la requête a été passée, d'élaguer des éléments recueillis les écritures et lectures qui ne sont pas causées par l'utilisation de Cortana et d'examiner les fichiers où les écritures sont faites suite à l'utilisation de Cortana. Nous avons peu tenu compte des lectures car elles ne laissent pas de traces en local. Cette nouvelle ronde nous a aussi permis de révéler la séquence de déclenchement des processus et fils causés par l'utilisation de Cortana et de préciser les résultats obtenus suite à l'analyse des éléments avec Volatility.

# CHAPITRE 6 – RÉSULTATS ET OBSERVATIONS – MÉMOIRE VIVE, PROCESSUS ET FILS

## 1. Introduction

Dans le présent chapitre et dans le suivant, nous abordons les observations faites et les résultats obtenus lors des phases expérimentales. Le chapitre 6 se rapporte à l'analyse de la mémoire vive tandis que le prochain chapitre traitera de l'analyse de la copie forensique du disque dur de l'ordinateur supportant Cortana et des flux émis et reçus d'Internet suite à l'utilisation de Cortana.

Dans le présent chapitre, nous énoncerons d'abord des observations d'ordre général puis nous couvrirons les observations et résultats relatifs à la mémoire vive avec Volatility puis avec Process Monitor.

## 2. Observations de portée générale

Initialement, le navigateur par défaut sur l'ordinateur expérimental était Firefox et ceci nous a empêché de faire fonctionner Cortana. Nous avons aussi pu déterminer que les suggestions faites par Cortana lors de l'énonciation d'une requête verbale ne sont plus localisées dans des fichiers json mais sont listées dans un fichier html.

## 3. Analyse de la mémoire RAM – Volatility

### 3.1. Introduction : déroulement et statistiques

La première partie de l'analyse de la mémoire vive porte sur les cueillettes effectuées immédiatement après chaque requête lors de la ronde du 9 septembre 2019. Cette cueillette a généré 104 fichiers totalisant 565 Go de données. Le traitement de ces données a généré 63502 fichiers totalisant 77 Go de données. Le 11 septembre 2019, le constat que Volatility version originale ne peut pas lire la mémoire vive d'un Windows 10 (parce que compressée à certains égards) est fait.

Le 14, la solution Fireeye est installée et le traitement avec Volatility-Fireeye commence pour se terminer le 10 octobre. Les résultats du traitement sont stockés dans 43 répertoires.

La cueillette étant une cueillette statique effectuée après la fin de la requête, la première analyse qui suit fait bien sûr état de la situation existant une fois que la requête est terminée. Bien sûr, une cueillette dynamique aurait été préférable car on aurait alors eu une meilleure idée de ce qui se passe pendant l'énoncé de la requête.

### 3.2. Hiérarchie des processus

La requête #103, « What is the foundation date of Canada? », a été utilisée pour dresser un « organigramme » des processus de la mémoire vive telle qu'elle se présente après la fin de la requête et la fin de la réception de la réponse à la requête. Cet organigramme se trouve à la figure 6-1. L'utilité de cette illustration est surtout de clarifier l'organisation des processus dans un Windows 10 Pro. Un tel organigramme constitue un point de départ solide à une personne chargée de traquer des maliciels car il indique clairement quel processus relève d'un autre.

Parmi ces processus, on note le « MemCompression » (en haut à gauche) présidant à la compression des éléments pertinents de la mémoire vive, « SearchUI » qui est activé lorsque Cortana reçoit une requête (Ligh), « SpeechRuntime » qu'on peut présumer être à l'origine de la reconnaissance vocale et les « MicrosoftEdge » à gauche.

### 3.3. Première analyse de mémoire vive

Après chaque requête soumise à Cortana, FTK Imager a été utilisé pour recueillir la mémoire vive et la mémoire de débordement de GMO. Bien qu'il ait été constamment en exécution (ce logiciel restait « ouvert » entre chaque requête), le fait de déclencher une nouvelle cueillette des mémoires vive et de débordement faisait démarrer un nouveau processus FTK Imager. Puisque ce processus a toujours été démarré quelques secondes après la fin de la soumission de la requête, le moment approximatif d'initialisation du processus de cueillette a servi **à valider les tampons horodateurs générés par Windows et recueillis et interprétés par Volatility et de cibler de façon plus précise les processus impliqués dans les activités de Cortana**.

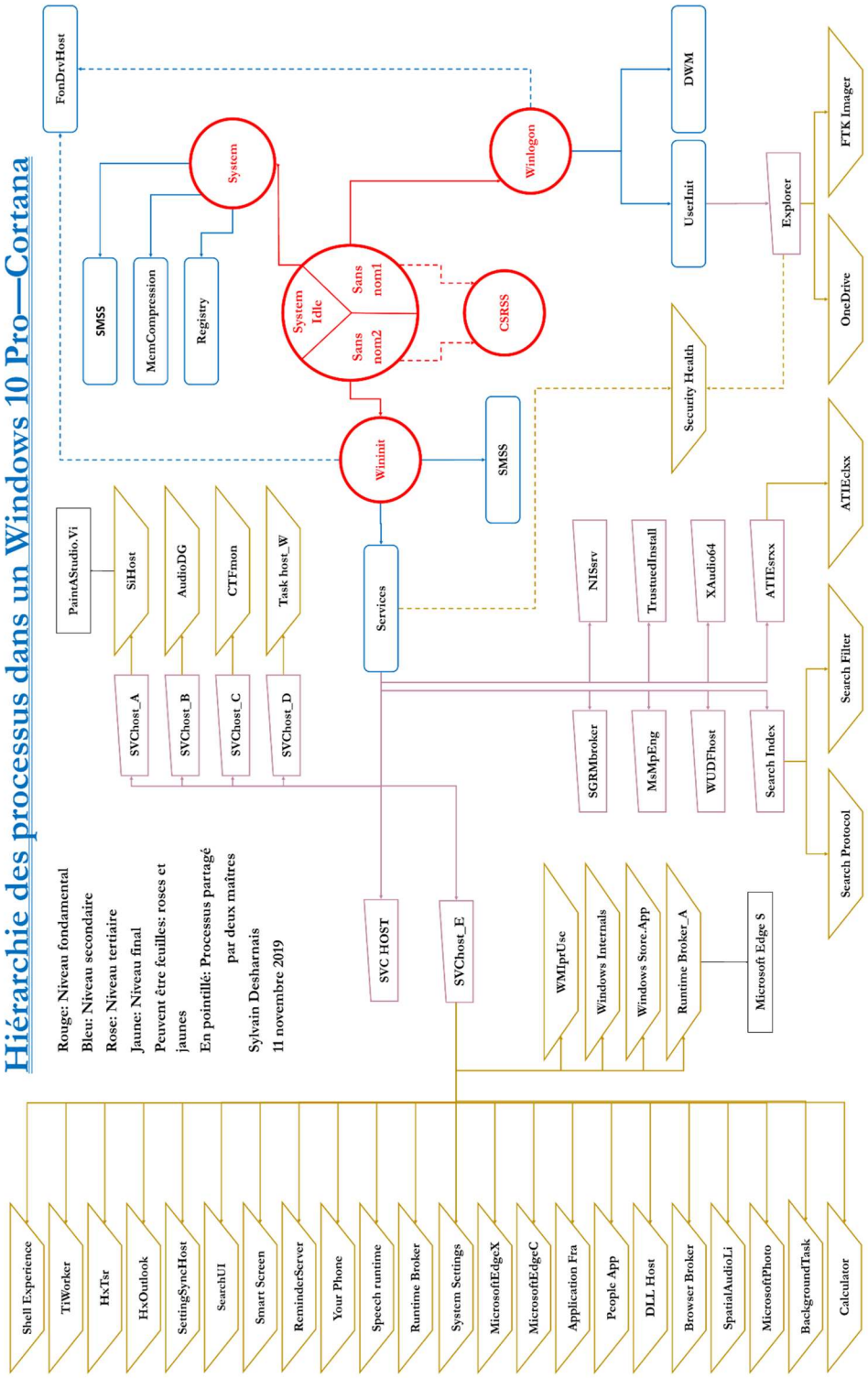



Figure 6-1 – Structure hiérarchique des processus Windows lorsque Cortana est utilisé

L'analyse a porté sur 18 requêtes (choisies en fonction de leur possibilité de fournir des résultats intéressants) contenant au total 1 333 occurrences du processus de cueillette FTK Imager relevées par Volatility. Chaque requête est présentée avec le statut qui lui est associé : « En exécution » (« Running »), « En attente d'écrire la requête de l'utilisateur » (« Waiting:WrUserRequest »), « Terminé » (« Terminated »), « En attente de la requête de l'utilisateur » (« Waiting:UserRequest ») et « En attente dans la queue » (« Waiting:WrQueue »). **Ces statuts permettent de raffiner l'analyse et de cibler, pour examen, les processus les plus probablement utilisés par Cortana.** Process Monitor (ProcMon) présente des statuts identiques dans ses résultats. À la différence de **Volatility qui fait de l'analyse statique, ProcMon**, comme nous le verrons dans la prochaine section, **fait de l'analyse dynamique**. L'analyse statique est basée sur un prélèvement de la mémoire vive une fois la requête complétée, une sorte de « photo finish » qui ne livre que l'état de la situation au moment du prélèvement. L'analyse statique garde une trace de tous les événements survenus avant, pendant et après la requête, pour tout le temps pendant lequel on enregistre l'avènement de ces événements.

La première occurrence d'utilisation par Windows de FTK Imager relevée par Volatility dans la mémoire vive recueillie suite à la requête 102 est estampillée 14h09:28, donc bien avant que la requête #2 ne soit soumise à Cortana. Lorsqu'on analyse l'horodatage de la requête #102 (102 car tirée du traitement de la requête #2) du tableau 5-2 du chapitre 5, on en déduit que la mémoire vive de la requête 102 a « hérité », selon Volatility, de ce qui a été fait par FTK Imager lors des requêtes #101a à 101f. À 14h33:57, il y a un processus FTK Imager débutant et qui se termine à 14h35:06. Puis, à 14h41:43 (soit 48 secondes après le début de la requête 102), Volatility indique que FTK Imager présente un statut « Running ».

Dans l'illustration à la figure 6-2, on voit ce qui s'est passé dans les mémoires vive et de débordement lors du passage des requêtes 101f (« Set the alarm-clock at [give an hour time in 4 minutes] [wait 15 seconds and] cancel timer ») et 102 (« Countdown from 9 to 0 »).

Bien sûr, nous aurions pu enchaîner avec la requête 103, à droite de la 102. Puis la 108… jusqu'à la dernière requête (la 123b). Au lieu de quoi, nous avons créé le tableau 6-1, qui présente les données relatives aux dernières occurrences présentant un statut « Terminated » pour chaque requête examinée. Les heures sont toutes exprimées dans leur valeur GMT afin de faciliter la comparaison.


Figure 6-2 – Séquence requête-cueillette pour les événements 101f et 102 détectée par Volatility en mémoire vive

| Numéro | Expérimentateur | FTKi Début | Écart(1) | FTKi Fin | Écart(2) | Début Req. Suivante | Écart(3) | Statut |
|---|---|---|---|---|---|---|---|---|
| 101f | 14:32:15 | 14:33:57 | 00:01:42 | 14:35:06 | 00:01:10 | 14:40:55 | 00:05:49 | Pos |
| 102 | 14:40:55 | 14:41:43 | 00:00:48 | 14:42:59 | 00:01:16 | 15:48:00 | 01:05:01 | Repas |
| 108 | 15:48:00 | 15:49:16 | 00:01:16 | 15:50:31 | 00:01:15 | 15:55:50 | 00:05:19 | Pos |
| 109 | 15:55:50 | 15:56:46 | 00:00:56 | 15:57:53 | 00:01:07 | 17:43:35 | 01:45:42 | Repas |
| 111 | 17:43:35 | 17:44:27 | 00:00:52 | 17:45:20 | 00:00:53 | 17:45:35 | 00:00:15 | Pos |
| 112 | 17:45:35 | 17:46:25 | 00:00:50 | 17:47:30 | 00:01:05 | 17:50:20 | 00:02:50 | Pos |
| 113a | 17:50:20 | 17:51:00 | 00:00:40 | 17:51:57 | 00:00:57 | 17:57:25 | 00:05:28 | Pos |
| 114 | 17:57:25 | 17:58:16 | 00:00:51 | 17:59:24 | 00:01:08 | 18:01:55 | 00:02:31 | Pos |
| 116 | 18:01:55 | 18:02:34 | 00:00:39 | 18:03:48 | 00:01:14 | 18:03:55 | 00:00:07 | Pos |
| 118 | 18:03:55 | 18:05:04 | 00:01:09 | 18:06:20 | 00:01:16 | 18:09:00 | 00:02:40 | Pos |
| 119b | 18:09:00 | 18:09:38 | 00:00:38 | 18:10:52 | 00:01:14 | 18:11:30 | 00:00:38 | Pos |
| 120 | 18:11:30 | 18:12:12 | 00:00:42 | 18:13:25 | 00:01:13 | 18:13:30 | 00:00:05 | Pos |
| 121a | 18:13:30 | 18:16:46 | 00:03:16 | 18:17:46 | 00:01:00 | 18:16:00 | 00:01:46 | Nég |
| 121c | 18:16:00 | 18:19:28 | 00:03:28 | 18:20:16 | 00:00:48 | 18:18:30 | 00:01:46 | Nég |
| 124 | 18:18:30 | 18:19:32 | 00:01:02 | 18:21:03 | 00:01:31 | 18:22:15 | 00:01:12 | Pos |
| 123a | 18:22:15 | 18:28:19 | 00:06:04 | 18:29:23 | 00:01:04 | 18:29:40 | 00:00:17 | Pos |
| 123b | 18:29:40 | 18:30:32 | 00:00:52 | 18:31:42 | 00:01:10 | 00:00:00 | 00:00:00 | Zéro |

Tableau 6-1 – Extrants Volatility pour le processus FTK Imager pour des requêtes choisies

**Les leçons à tirer de cette observation se présentent en trois points**. **Premièrement**, le contenu de la mémoire vive à un moment donné suivant un événement spécifique peut contenir des éléments provenant des requêtes précédentes ou d'événements autres survenus à des instants précédents. **Deuxièmement**, on doit en conclure que les valeurs des tampons horodateurs de FTK Imager, comparés aux dates-heures notées par les expérimentateurs, confirment que Volatility interprète correctement les valeurs dates-heures sauvegardées par Windows dans la mémoire vive de l'ordinateur GMO utilisé pour l'expérience. **Enfin**, dans chaque analyse que nous ferons par la suite des extrants Volatility pour une requête particulière, il faudra ne pas tenir compte des fils terminés avant le passage de ladite requête. La raison de cet élagage est que si le processus s'est terminé avant qu'on ne passe ladite requête, c'est qu'il n'a pas été déclenché par ladite requête mais par quelque chose qui la précède. Bien que ce raisonnement soit récursif, il doit être formellement énoncé afin de faire un choix judicieux des occurrences à analyser parmi les extrants Volatility pour une requête. Notamment parce qu'il faudra ne pas tenir compte des fils démarrés avant la requête sous analyse car eux aussi n'ont rien à voir, en principe, avec la requête sous analyse. Mais nous survolerons, par acquis de conscience, les processus démarrés entre le début de la requête précédant la requête sous analyse et le début de cette dernière.

Reprenons la requête 102 et voyons ce que l'application des critères donne en termes de quantité. Initialement, Volatility a extrait 2 420 fils dans la mémoire vive recueillie après la requête 102. 190 de ces fils se sont terminés avant 14:40:55 (heure à laquelle la requête 102 a été soumise à Cortana). 1983 de ces fils ont été créés avant 14:32:15 (heure à laquelle la requête 101f, requête qui précédait immédiatement dans le temps la requête 102, a été soumise à Cortana). Il reste donc 247 fils à analyser dont le démarrage va de 14:32:15 jusqu'à 14:42:59.

Parmi les 247 fils restants de la requête 102 suite aux élagages du paragraphe précédent, certains ne sont pas impliqués dans le travail fait par Cortana, notamment parce que ce sont des logiciels démarrés par l'expérimentateur pour surveiller l'activité de l'ordinateur GMO comme FTK Imager et ProcMon64.

Les extrants Volatility montrent que FTK Imager est exécuté par un processus présentant 10 fils. Ces 10 fils ne sont sûrement pas dépendant de Cortana puisque si FTK Imager n'est pas activé sur Windows, Cortana continue de fonctionner (c'est-à-dire que Cortana n'exige pas que FTK Imager soit démarré pour fonctionner. De plus, c'est un logiciel qui a été démarré par l'expérimentateur

de manière délibérée. ProcMon64 utilise 12 fils qui ne sont sûrement pas dépendant de Cortana puisque lui aussi a été démarré par l'expérimentateur de manière délibérée.

Parmi les 225 fils restants de la requête 102 suite aux élagages des paragraphes précédents, certains sont peut-être impliqués dans le travail fait par Cortana, sans être spécifiques à Cortana. Ici, c'est une question de jugement professionnel. Mais chaque élément éliminé se justifie par le fait que tous les processus faisant partie du noyau rouge (voir « Hiérarchie de la mémoire vive à la figure 6-1 ci-haut »), à savoir les processus « System » (44 lignes), « Wininit » (aucune ligne pour la requête 102), « Winlogon » (aucune ligne pour la requête 102) et « CSRSS » (aucune ligne pour la requête 102). Bien que ces processus interviennent sûrement dans une activité Cortana, ils ne sont pas **spécifiques** à Cortana et n'amènent rien à l'objectif de notre recherche.

Il reste donc 182 fils dont seulement 66 ont été créés après 14:40:55, heure à laquelle la requête 102 a été soumise à Cortana. Ces 66 fils sont liés à 19 numéros de processus distincts. Il y a répétition de certains fils : même numéro de processus, même numéro de fil, mêmes dates-heures de début et de fin. Si nous éliminons les trente (30) de ces doublons, il reste 36 fils. Posons la présomption que les fils qui sont créés suite à la soumission de la requête sont créés dans les 30 secondes de la soumission de la requête et que tous ceux créés par après ne sont pas le fait de la soumission de la requête.

| **Première conclusion majeure** : Cinq processus principaux se sont activés lorsque l'utilisateur a énoncé la requête 102 : backgroundTask, dllhost, explorer, RuntimeBroker et SearchUI. |
|---|

### 3.4. Analyses suivantes

L'analyse faite avec la requête 102 a été répétée avec 17 autres requêtes[14]. De cette analyse nous avons tiré un tableau de 1 076 lignes et 7 colonnes[15]. 29 processus ont été initiés suite à l'une ou l'autre des 18 requêtes analysées sur les 60 secondes (malgré la présomption du paragraphe précédent qui était de 30 secondes) suivant l'énoncé de ces requêtes. Le nombre moyen

[14] 101a, 102, 103, 108, 109, 111, 112, 113a, 114, 116, 118, 119b, 120, 121a, 121c, 123a, 123b, 124.
[15] Numéros de requête, de fil, de processus, nom du processus, date de début de fil et de fin s'il en est.

d'activations de ces 29 processus sur ces 60 secondes est de 21.7 initialisations[16]. Afin de ne tenir compte que des processus intervenant le plus souvent lors de requêtes soumises à Cortana, seuls les processus activés plus de 22 fois (supérieurs à la moyenne) ont été considérés. Le graphique de la figure 6-3 a été créé à l'aide de ce tableau de 1076 lignes, tableau similaire au tableau 6-1.

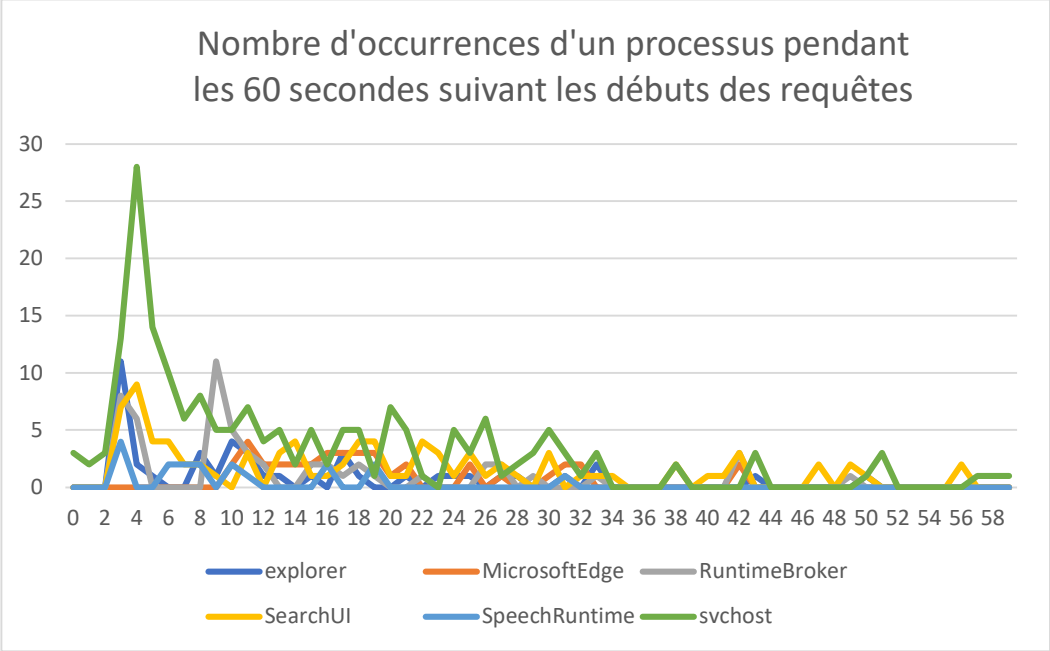



Figure 6-3 – Nombre d'occurrence des processus les plus actifs lors des 60 secondes suivant le début de l'énoncé des requêtes Cortana

L'objectif était de déterminer si certains processus étaient **systématiquement** initialisés suite à l'énoncé d'une requête et à quel moment. Il s'est avéré que très peu de processus ont été initialisés (8 initialisations sur les 432 relevés) pendant les trois premières secondes (de 0,0 à 2,9) du début de la requête. La zérotième seconde étant le moment noté par les expérimentateurs comme marquant le début de la soumission de la requête. Un tel délai (de 2,9 secondes) peut sembler surprenant à prime abord, mais il est toutefois cohérent lorsqu'on pense au temps de réaction entre

[16] Parce qu'il y a partage des ressources, notamment les processus, par les composants matériels et logiciels d'un ordinateur, un processus déclenché n'exécutera pas la totalité de la tâche qui lui est demandée. Il lira le code permettant l'activation du processus, réalisera une partie de la tâche, puis lira à nouveau le code du processus pour exécuter une autre partie de la tâche. Ainsi de suite jusqu'à finalisation de la tâche. Dans notre expérimentation, il y a eu **en moyenne 21,7 itérations** autour des requêtes faites à Cortana.

le signal du chronométreur et le début de l'énonciation de la requête par l'utilisateur ainsi qu'à la présence d'un délai entre la locution-gâchette « Hey Cortana » énoncé par l'expérimentateur et l'affichage dans Windows du signal comme quoi Cortana écoutait. La durée de l'énoncé des requêtes se situe entre 5 et 7 secondes.

Cette absence d'initialisation de processus est marquée et contraste avec ce qui se passe entre 3,0 et 3,99 secondes du début de l'énoncé. En effet, pendant ce laps de temps, 43 processus démarrent et à la quatrième seconde, 45 démarrages ont lieu.

> Comme on peut le voir sur le graphique illustré à la figure 6-3, les processus les plus actifs de 3,0 à 10,0 secondes des débuts des énoncés, sont, dans l'ordre svchost, explorer, SearchUI et RuntimeBroker. **Cette observation appuie la conclusion livrée à la fin du paragraphe 3.3 ci-haut.**

### 3.5. Conclusion sur l'analyse de la mémoire vive à l'aide de Volatility

L'analyse post-hoc de la mémoire vive recueillie suite à un événement présente une faille majeure. Comme l'analyse d'une photo prise à la suite d'un accident, on analyse la situation d'une manière statique, ce qui peut nous donner des indices sur les causes et le déroulement de l'accident mais ça ne constitue pas une frise temporelle de ce qui se passe lors de la requête car Volatility n'extrait que ce qu'il voit dans le fichier de sauvegarde, en l'occurrence ce qui a été capté 1 à 3 minutes après la fin de la requête.

Par contre, cette « photo statique » peut nous mettre sur la piste d'éléments qui auraient été négligés sinon : la contribution de la « progéniture » des processus svchost et explorer, la présence systématique dans les premières secondes de la requête des processus SearchUI et RuntimeBroker. C'est pourquoi, afin de préciser ces découvertes, de nouvelles requêtes ont été soumise au Cortana de GMO et ProcMon a été utilisé pour recueillir des données dynamiques.

## 4. Analyse de la mémoire RAM – ProcMon Sysinternals

Rappelons-nous que Sysinternals, dirigé par Mark Russinovitch, a créé ce gratuiciel qui permet d'enregistrer toutes les opérations impliquant les processus, la base de registre, les DLL, les fichiers et le système d'exploitation.

Afin de préciser ce qui a été dit au paragraphe précédent, la requête 103 a été soumise à Cortana dans une session différente de celle du 23 septembre 2019, soit le 5 mars 2020 à 16h56:00,00 HE. Le premier élément enregistré par ProcMon est tamponné 16h56:01,74. Le délai de réaction est ici moins grand que lors de l'expérience du 23 septembre 2019 (où il était de 3 secondes) dû au fait que l'expérimentateur se donnait lui-même le signal de début d'enregistrement. Le temps relatif est donc à 0,00 lorsqu'il est 16h56:01,74 (ces 1.74 secondes constituant ce que nous désignerons plus loin sous le vocable de délai de système). **Pour l'analyse qui suit, le temps relatif sera utilisé**.

Ce genre de cueillette génère une énorme quantité d'informations. L'enregistrement du 5 mars 2020 a duré 24,65 secondes. ProcMon a enregistré 290 418 entrées réparties en **5 catégories** : base de registre, système de fichiers, réseau, processus et fils et événements de profilage. Chaque entrée comporte trois catégories de détails montrant plusieurs lignes de détail : description de l'événement, processus auquel cet événement est lié (qui livrent individuellement d'autres détails), la pile (« stack ») sur laquelle l'événement se trouve (livrant individuellement d'autres détails).

Beaucoup de données, donc. **D'où la nécessité d'une méthode pour en faire une interprétation efficace**. Dans un premier temps, un traitement processus par processus (svchost, SearchUI, explorer, RuntimeBroker et SpeechRuntime) a été réalisé en utilisant les filtres de catégories « Système de fichier » et « Processus et fils ». Énoncer une requête prend environ 3 à 7 secondes, excluant le délai pour attendre la réaction de Cortana au mot-gâchette. Recevoir la réponse à cette requête prend environ 4 à 5 secondes. Donc 1.5 secondes de délai de réaction de Cortana au mot-gâchette, plus 7 plus 5 plus une zone de confort d'une valeur arbitraire de 10%, moins un délai système de 1.74 secondes font 13.07 secondes. **Ce sont donc seulement les données générées avant la 13ième secondes qui ont donc été analysées.**

## 4.1. SvcHost

### 4.1.1. Introduction

L'icône de surfiltre « File System » a été sélectionnée et le filtre présenté à la figure 6-4 a été configuré. Ceci a filtré 5113 entrées parmi les 290 418 entrées de la requête 103 du 5 mars 2020. Comme il s'agissait de la première analyse avec ProcMon, elle a été très détaillée afin de pouvoir préciser ce qui peut être trouvé grâce à cette méthode. La première ligne indique que cette première partie de l'analyse porte sur le processus SVChost.

| Column | Relation | Value | Action |
|---|---|---|---|
| Process Name | is | svchost.exe | Include |
| Process Name | is | Procmon.exe | Exclude |
| Process Name | is | Procexp.exe | Exclude |
| Process Name | is | Autoruns.exe | Exclude |
| Process Name | is | Procmon64.exe | Exclude |
| Process Name | is | Procexp64.exe | Exclude |
| Process Name | is | System | Exclude |
| Operation | begins with | IRP_MJ_ | Exclude |
| Operation | begins with | FASTIO_ | Exclude |
| Result | begins with | FAST IO | Exclude |
| Path | ends with | pagefile.sys | Exclude |
| Path | ends with | $Mft | Exclude |
| Path | ends with | $MftMirr | Exclude |
| Path | ends with | $LogFile | Exclude |
| Path | ends with | $Volume | Exclude |
| Path | ends with | $AttrDef | Exclude |
| Path | ends with | $Root | Exclude |
| Path | ends with | $Bitmap | Exclude |
| Path | ends with | $Boot | Exclude |
| Path | ends with | $BadClus | Exclude |
| Path | ends with | $Secure | Exclude |
| Path | ends with | $UpCase | Exclude |
| Path | contains | $Extend | Exclude |
| Path | begins with | HK | Exclude |
| Event Class | is | Profiling | Exclude |
| Event Class | is | Network | Exclude |
| Event Class | is | Process | Exclude |
| Event Class | is | Registry | Exclude |

Figure 6-4 – Filtre appliqué sur la requête 103 du 5 mars 2020 pour l'analyse du processus svcHost

Le tableau 6-2 montre les détails relatifs au déclenchement des processus suite à la soumission de la requête à Cortana. Les **deux colonnes importantes** sont le temps relatif « T_relatif » et le nom du fichier consulté parce que c'est de ce tableau que nous avons tiré la figure 6-5.

| Processus | T_relatif | Répertoire | Fichier | Action |
|---|---|---|---|---|
| 2976 | 1,74 | \Windows\SystemApps\Microsoft.Windows.Cortana_cw5n1h2txyewy\ | SearchUI.exe | Lecture |
| 2700 | 1,99 | \Users\inter\AppData\Local\ConnectedDevicesPlatform\bb717611e150420e\ | ActivitiesCache.db-wal | Écriture |
| 2700 | 1,99 | \Users\inter\AppData\Local\ConnectedDevicesPlatform\bb717611e150420e\ | ActivitiesCache.db-shm | Lecture |
| 2700 | 2,02 | \Windows\Registration\ | R00000000000d.clb | Écriture |
| 2700 | 2,02 | \Windows\Registration\ | R000000000001.clb | Écriture |
| 1680 | 2,08 | \ProgramData\Microsoft\Windows\AppRepository\ | StateRepository-Machine.srd-shm | Lecture |
| 2976 | 2,05 | \Users\inter\AppData\Local\Packages\Microsoft.Windows.CloudExperienceHost_cw5n1h2txyewy\AC | | Lecture |
| 808 | 3,21 | \ProgramData\Microsoft\Windows\AppRepository\Packages\Microsoft.AAD.BrokerPlugin_1000.18362.449.0_neutral_neutral_cw5n1h2txyewy\ | S-1-5-21-4281360298-3334865483-766225500-1001.pckgdep | Lecture |
| 808 | 3,21 | \Windows\apppatch\ | sysmain.sdb | Lecture |
| 2976 | 3,21 | \Windows\System32\Speech_OneCore\common\ | SpeechRuntime.exe | Lecture |
| 852 | 3,29 | \Windows\Registration\ | R00000000000d.clb | Lecture |
| 852 | 3,29 | \Windows\Registration\ | R000000000001.clb | Lecture |
| 808 | 3,69 | \Windows\Temp | | Lecture |
| 808 | 3,69 | \Users\inter\AppData\Local\Temp | | Lecture |
| 808 | 3,69 | \Windows\SystemApps\Microsoft.AAD.BrokerPlugin_cw5n1h2txyewy | | Lecture |
| 808 | 3,69 | \Windows\apppatch\ | sysmain.sdb | Lecture |
| 808 | 3,69 | \Windows\SystemApps\Microsoft.AAD.BrokerPlugin_cw5n1h2txyewy | | Lecture |
| 808 | 3,69 | \Windows\System32\ | backgroundTaskHost.exe | Lecture |
| 1128 | 4,15 | \Windows\System32\ | LocationNotificationWindows.exe | Lecture |
| 1300 | 4,28 | \Windows\System32\ | npmproxy.dll | Lecture |
| 1300 | 4,28 | \Windows\System32\ | netprofm.dll | Lecture |
| 2976 | 4,28 | \Windows\System32\ | TokenBroker.dll | Lecture |
| 2976 | 4,28 | \Windows\System32\ | backgroundTaskHost.exe | Lecture |
| 1300 | 5,49 | \Windows\System32\ | aepic.dll | Lecture |
| 1300 | 5,49 | \Windows\appcompat\Programs\ | Amcache.hve | Lecture |
| 2976 | 5,87 | \Windows\WinSxS\amd64_microsoft-windows-services-svchost_31bf3856ad364e35_10.0.19041.1_none_6bac6724a4ab4460 | svchost.exe | Lecture |
| 1876 | 5,89 | \Windows\Prefetch\ | BACKGROUNDTASKHOST.EXE-901440A8.pf | Lecture |
| 3584 | 6,19 | \Users\inter\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\AC\Microsoft\CryptnetUrlCache\MetaData\ | 77EC63BDA74BD0D0E0426DC8F8008506 et autres de titres similaires | Lecture |
| 808 | 7,87 | \ProgramData\Microsoft\Windows\AppRepository\Packages\Microsoft.Windows.Cortana_1.13.0.18362_neutral_neutral_cw5n1h2txyewy\ | ActivationStore.dat | Lecture |
| 852 | 10,62 | \Windows\AppRepository\Packages\Microsoft.MicrosoftEdge_44.18362.449.0_neutral__8wekyb3d8bbwe\ | ActivationStore.dat | Lecture |
| 8888 | 11,34 | \Windows\Prefetch\ | SVCHOST.EXE-98090C0A.pf | Lecture |
| 8888 | 11,34 | \Windows\System32\ | Plusieurs dll | Lecture |
| 808 | 12,27 | \Windows\System32\ | MicrosoftEdgeCP.exe | Lecture |

Tableau 6-2 : Premières occurrences de sollicitation des participants à la résolution d'une requête soumise à Cortana

#### 4.1.2. Analyse du déroulement des événements de catégorie « File System »

Le tableau 6-5 contient les informations relevées lors de l'analyse. La colonne de gauche contient le numéro du processus, la colonne suivante le temps relatif marquant le début de l'événement et la troisième le fichier ou le répertoire impliqué dans l'événement. La plupart des processus sont sollicités à plusieurs reprises. Pour bien comprendre ce phénomène de bouclage, il faut savoir que tout espace est lu ou écrit par lot de page mémoire (avec Windows, ces pages sont normalement de 4 096 octets) parce que le système d'exploitation lit les données pour ensuite les placer en mémoire. S'il s'agit d'écriture, le système d'exploitation prend un page mémoire et la stocke au complet dans le fichier. Nous n'avons colligé que la première occurrence afin d'éviter la redondance et parce que la première occurrence est plus significative que les suivantes en ce qui a trait à la réaction du système d'exploitation lorsqu'il reçoit la requête.

### 4.2. Observations sur les processus et les fils

La requête 103 a généré 112 opérations d'écriture. Cette requête a guidé nos recherches lors de l'analyse forensique des traces locales. Les tableaux A-1 et A-2 de l'annexe A, présentent la liste des fichiers dans lesquels ces écritures ont été faites, à quel temps relatif (en secondes) est survenue la première occurrence de cette écriture, le nombre d'écritures faites, une description sommaire de la nature du fichier où l'écriture s'est faite et la référence bibliographique où cette description sommaire a été trouvée. Il y a deux types d'écritures : l'écriture de données comme telle et l'écriture de métadonnées. Ici, ProcMon ne précise pas s'il s'agit de paradonnées ou d'intradonnées.

Si on schématise la procédure déclenchée par le « Hey Cortana! », en négligeant les multiples actions récursives, on obtient la frise chronologique de la figure 6-5 ci-dessous. Attention toutefois : La hauteur des colonnes représente simplement le temps relatif calculé par ProcMon. Le temps relatif est le décalage temporel entre le démarrage de l'enregistrement et l'avènement de l'événement. **La figure 6-5 fournit donc de plus amples détails à la figure 6-3.**

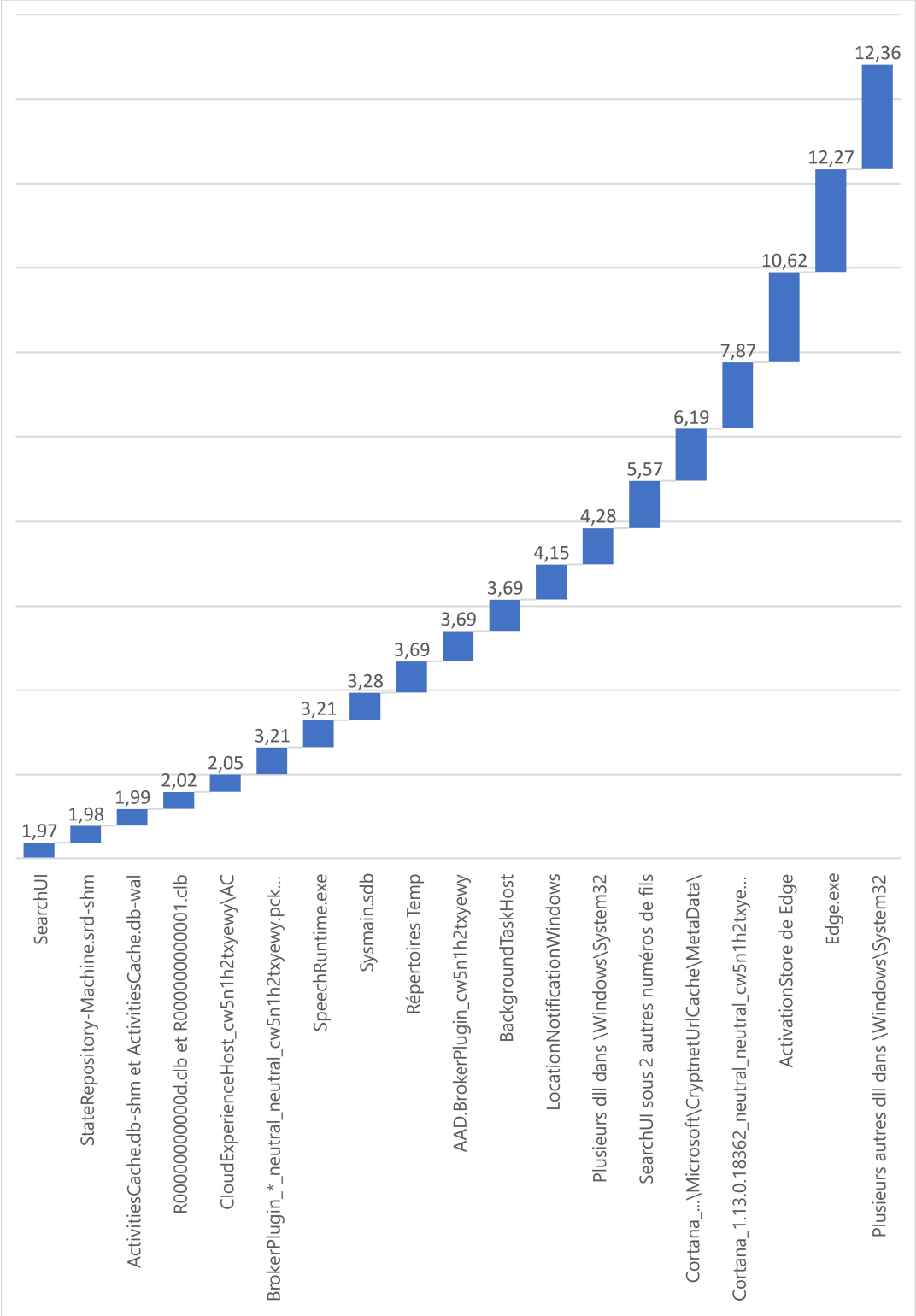



Figure 6-5 – Frise chronologique - Démarrage des fils Requête #103 - Mars 2020

## 4.3. Base de registre

### 4.3.1. Introduction

Il y a 374 entrées se rapportant aux opérations de base de registre, dont 64 sont des écritures de données ou de métadonnées. Les autres entrées sont des lectures ou ne sont pas étiquetées. Seulement trois processus et dix fils ont utilisé la base de registre pendant la requête 103 de mars 2020.

### 4.3.2. Analyse

*Registry\A :* ProcMon désigne sous ce vocable les ruches d'application. Msuhanov [2019] explique qu'une ruche d'application « …*has no visible mount poinr* [. A] *program that is going to access such a hive should load it and use a handle to a root key returned*…*There is no way to describe a path to a key found in an application hive using typical conventions*…[It is a] *mount point which can't be enumerated* ». Ce sont donc des ruches temporaires, créées pour les besoins de la cause. Elles n'existent qu'en mémoire et sont évacuées aussitôt que le système n'en a plus de besoin, raison pour laquelle il a été impossible de les analyser.

*Registry\WC\silo### :* La nature de cette ruche est pour le moins nébuleuse. Mangan [2017] dit « *This is a mount event of the virtual registry file of the package, done as a single event. Further virtual registry entries in the trace show the \\Registry\WC\Silo_… path being used to access individual entries… to turn Win32 apps into Windows Store Apps, utilized these containers to package things up* ». Depuis 2017, silence radio mis à part une non-réponse sur un forum de Microsoft (voir Chen [2019]). Quoi qu'il en soit, dans la base de registre de l'ordinateur GMO, le CLSID mentionné dans cette entrée porte le nom de ShellServiceHostBrokerProvider (SvcHost veut dire Service Host).

*Comroot :* Là aussi, peu d'information.

*CLSID* : Les 17 CLSID mentionnés dans les « REGISTRY\A » n'ont pas de correspondance dans la base de registre de GMO. Donc, ces CLSID sont créés de toutes pièces dans une base de registre virtuelle détruite lors de la fermeture des descripteurs de processus. Le 23 septembre 2020, l'image du disque dur opéré actuellement par GMO a été monté dans FTK Imager. Il y a dans cette image forensique des traces d'occurrences de la clé de registre WC sans que ça n'apporte d'information

supplémentaire. Un CLSID comptant pour 95 occurrences dans ProcMon y a aussi été recherché, sans succès.

#### 4.3.3. Conclusion :

La base de registre est une piste pauvre en renseignements. Ceci est dû en grande partie à ce que Windows utilise abondamment des clés virtuelles qui disparaissent une fois le descripteur de processus fermé.

### 4.4. Analyse de SearchUI

Ce processus a produit 3 508 entrées, dont la première survient à 1.87 secondes de temps relatif. Sur les 2 182 entrées « File System » et « Process and Handles », 262 sont de catégorie « Écriture », 27 « Écriture de métadonnées », 121 « Lectures », 455 « Lecture de métadonnées » et 1 317 de catégorie indéfinie.

#### 4.4.1. Séquence

La première entrée, à 1,87 de temps relatif se rapporte à SearchUI.exe comme tel. À 1.881, Windows consulte pour la première fois le répertoire \Windows\Speech_OneCore\Engines. Nous avons peu d'informations au sujet de ce répertoire, mais nous avons examiné le contenu de certains fichiers avec WinHex. Notamment, le fichier c4105.fe contenant entre 400 et 500 entrées du style « 16 kHz 16 bit Mono PCM Metadata Stream » et au sujet duquel on peut formuler l'hypothèse qu'il s'agisse des configurations possibles des voix disponibles pour Cortana. D'autres fichiers dont les titres contiennent des prénoms comme Eva, Zira, Mark, et David contiennent probablement les paramètres pour différentes voix de Cortana. Windows lira à 646 reprises dans ce répertoire et ses sous-répertoires mais n'y écrira jamais.

À 1.882 de temps relatif, une écriture est faite dans le fichier \Users\inter\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\TempState\Traces\CortanaTrace1.etl pour la première fois. En moins de 11 secondes, on y écrira 44 autres fois et il y aura une dernière écriture dans ce fichier à 21.98. Ce fichier est un journal des opérations de Cortana qui devra être étudié dans la section sur les traces locales.

À 2.74, le fichier \Users\inter\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\AC\TokenBroker\Cache\fbaf94e759052658216786bfbabcdced1b67a5c2.tbres est ouvert et lu 38 fois en 3 secondes. Filext précise, au sujet des fichier .tbres « *The TBRES file type is primarily associated with Program packages. TBRES file extension accompanies files which are meant for certain applications and Microsoft Office programs. The TBRES files are usually program packages and application packages which are used for the purpose of interface customization of the application or just the software* ».

À 2.795, Windows procédera à la création d'une floppée de fichiers virtuels dans \Users\inter\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\LocalState\ mais n'écrira que dans un seul fichier réellement « SpeechAudioFile_4.wav », fichier qui contient la version audio de la requête 103 (celle présentement analysée). C'est donc dire que Cortana stocke un échantillon de la voix de l'utilisateur. À 2.91, le fichier sera créé. Entre 6.32 et 9.21, 23 lots de 8192 octets et un lot de 5654 octets seront inscrits dans ce fichier de 189 kilooctets. Le désir de confirmer la taille de ce fichier en utilisant FTK Imager sur l'ordinateur GMO a créé une surprise : le fichier original contenant la requête 103 de mars 2020 avait été écrasé par un nouveau contenu. Il faut savoir que Cortana préserve sur le disque dur l'audio des 8 dernières requêtes. L'écoute de ces 8 fichiers a été, là aussi, une surprise : tous les fichiers contenaient des bouts de phrases prononcées dans l'environnement de GMO en mai, juin et juillet 2020. Ces phrases étaient faciles à identifier puisqu'elles avaient fait l'objet d'un enregistrement dans le cadre de la dispensation d'un cours en ligne pour les fins de l'université Laval. Bien que le contenu soit anodin, cet « incident » souligne bien que Cortana se déclenche de lui-même (sans que les mots-gâchette « Hey Cortana » n'aient été prononcés). La présence de 8 échantillons de la voix de l'utilisateur et le déclenchement spontané de Cortana constituent une menace certaine à la sécurité et à la vie privée. Les autres fichiers consultés par Cortana pendant les secondes qui suivent sont des fichiers avec une extension cfg. Ces fichiers contiennent une multitude de mots et d'expressions séparés par le nombre hexadécimal 0x00 00 00.

Entre 3.04 et 9.16, Cortana créé à 14 reprises le fichier \Users\inter\AppData\Local\Packages\microsoft.windows.cortana_cw5n1h2txyewy\AC\GEH\POF.dat, fichier qui n'a pourtant pas été retrouvé sur le disque dur de GMO. À 3.27 puis à 11.62, Cortana consulte des fichiers avec une extension .json, fichiers aux noms évocateurs du style « speaking », « listening » et « calm » qui n'ont pas été retrouvés dans l'image forensique. À 3.275, Windows lit des fichiers aux mêmes noms évocateurs dans le répertoire

\Windows\SystemApps\Microsoft.Windows.Cortana_cw5n1h2txyewy\Assets\Persona\. Ces fichiers n'ont pas été retrouvés à cet endroit ni ailleurs.

De 4.34 à 11.23, Windows écrit/lit à 104 reprises dans \Users\inter\AppData\Local\Packages \Microsoft.Windows.Cortana_cw5n1h2txyewy\AC\Microsoft\Internet Explorer\DOMStore\ 1KXLS3UJ le fichier www.bing[1].xml. Bing est le moteur de recherche exigé par Windows pour que Cortana puisse fonctionner. À compter de 8.47, ProcMon révèle des lectures et des écritures vers le répertoire \Users\inter\AppData\Local\Packages\Microsoft.Windows.Cortana_ cw5n1h2txyewy\AC\INetCache\ où sont stockés les résultats des requêtes. Enfin, à 8.51, Windows lit plusieurs fichiers dans \Windows\System32\WinMetadata\.

### 4.4.2. Sommaire des étapes

La figure 6-6 présente la séquence de démarrage des processus telle que tirée de l'analyse du processus SearchUI livrée aux paragraphes précédents.

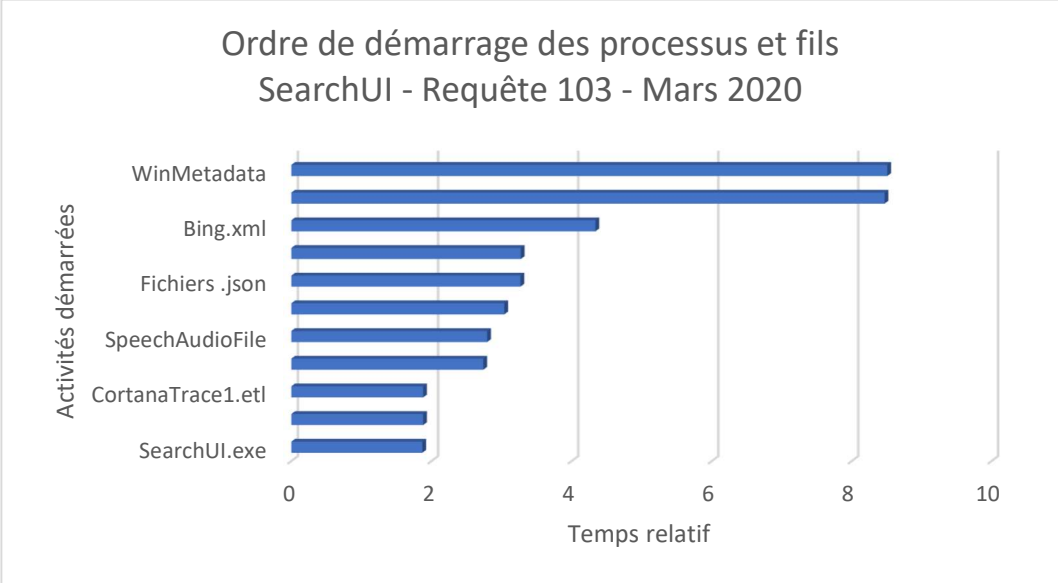



Figure 6-6 – Ordre de démarrage des processus et fils SearchUI – Requête 103 – Mars 2020

#### 4.4.3. Conclusion – SearchUI

Cortana démarre spontanément sans avoir été sollicité, ce qui constitue une menace certaine d'immixtion dans la vie privée des personnes mises en présence de l'appareil exploitant Cortana. Les éléments importants de la séquence de démarrage sont le démarrage de SearchUI et des moteurs de reconnaissance vocale de Cortana, l'utilisation d'un journal des opérations de Cortana, l'utilisation du moteur de recherche Bing et de la cache de Cortana INetCache.

## 4.5. Analyse d'Explorer.exe

#### 4.5.1. Introduction

Il y a 5 463 événements liés à ce processus. De ceux-ci, il y a 786 lectures, 1 292 lectures de métadonnées, 308 écritures, 10 écritures de métadonnées et 3 067 de la catégorie indéfinie.

#### 4.5.2. Analyse

De 2.57 à 14.5, Windows 322 fois l'un ou l'autre des fichiers .pri dans \Windows\SystemApps\Microsoft.Windows.Cortana_cw5n1h2txyewy. Microsoft [2018] révèle que « *The build system will be able to create, version, and dump (as XML) package resource index (PRI) files to whatever level of complexity your UWP app needs* ».

À compter de 2.69, Windows lit des fichiers thumbcache dans le répertoire \Users\inter\AppData\Local\Microsoft\Windows\Explorer\. Le site de ThumbcacheViewer nous dit que ces fichiers « *are databases that are native to Windows… systems. They contain thumbnail images of various content on your system* ». Par expérience, le contenu varié en question peut aussi bien être des images (exemple jpg, png) que des aperçus d'écran créés dans le répertoire du thumbcache qu'un fichier de bureautique comme un docx.

De 7.17 à 7.19, Explorer lit 46 fois \Windows\System32\LocationNotificationWindows.exe. À compter de 10.74, Explorer lit aussi, dans divers répertoires : Windows.Shell.ServiceHost Builder.dll, Firefox.exe, LauncWinApp.exe, iexplore.exe, desktop.ini, ieproxy.dll, Windows. System.Launcher.dll, ApplicationFrameHost.exe et CapabilityAccessManagerClient.dll.

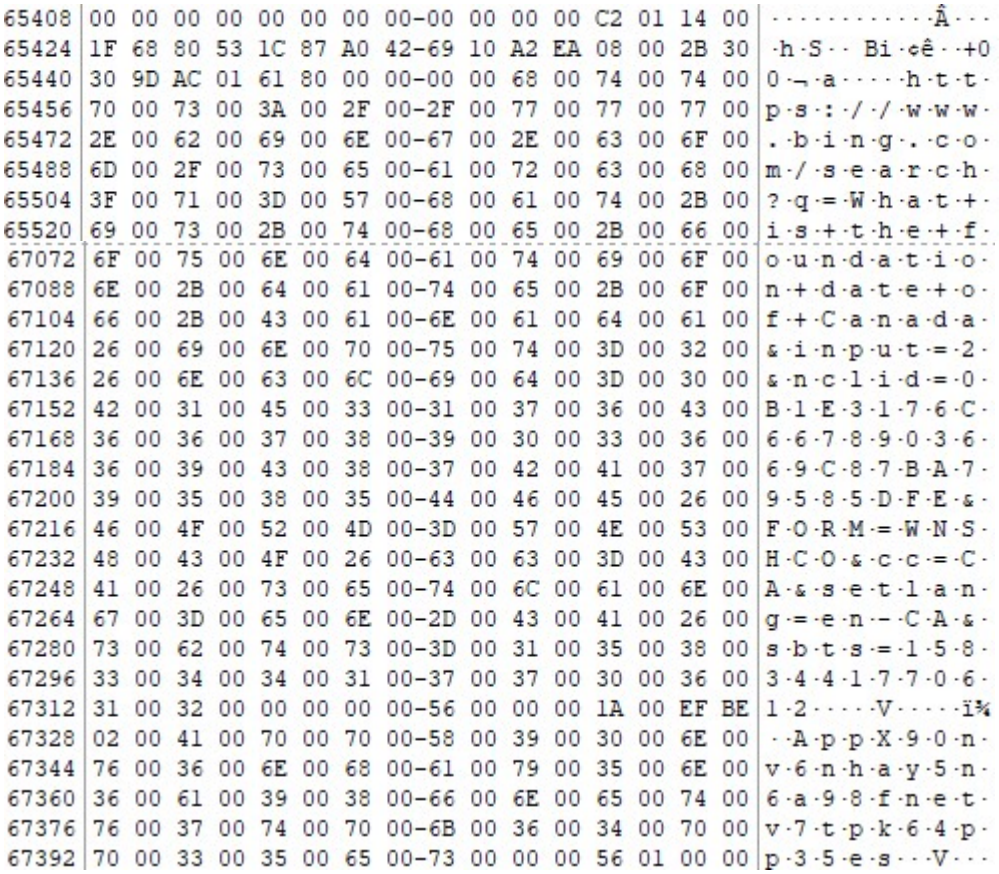


Figure 6-7 – Aperçu d'écran – Requête sauvegardée dans un fichier automaticDestinations-ms

De 11.28 à 11.34, Windows commence écrira 258 fois dans le fichier \Users\inter\AppData\Roaming\Microsoft\Windows\Recent\AutomaticDestinations\9d1f905ce5044aee.automaticDestinations-ms. 261 des 308 occurrences d'écriture faites par Explorer sont faites dans ce fichier. Lorsqu'on regarde au décalage (« offset ») indiqué par ProcMon, on s'aperçoit que ce fichier agit à titre de mémoire-tampon, en partie comme une mémoire vive et en partie comme un espace de sauvegarde. En effet, les inscriptions dans ce fichier sont désordonnées : la première occurrence dans le temps n'est pas forcément en tête de fichier, les suivantes étant placées à la queue-leu-leu, ce qui est un comportement typique de la mémoire vive et du fichier « pagefile ». À preuve les deux écritures consécutives à 11.283 faites aux décalages 0d67814 et 0d512. Mais contrairement aux mémoires vives et de débordement, les sauvegardes ne se font pas par page de 4 096 octets. À preuve les deux écritures consécutives faites à 11.275 : sauvegarde de 114 octets au décalage 0d65422 puis 336 octets au décalage 0d67072. Lorsqu'on regarde les éléments sauvegardés à ces

endroits, on observe une autre brèche potentielle dans la vie privée de l'utilisateur puisqu'on y divulgue la teneur exacte de la requête. Voir la figure 6-7. Mais cette intrusion va bien au-delà des requêtes verbales faites à Cortana puisque des activités faites sur cet ordinateur il y a de cela une année ont été retracées!

À 13.55, Explorer créé le fichier \Users\inter\AppData\Roaming\Microsoft\Windows\Recent\https--www.bing.com-searchq=What+is+the+foundation+date+of+Canada&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1583441770612.lnk. Un fichier lnk est créé lorsque Windows ouvre un fichier. Ce fichier contient notamment le tampon horodateur de l'ouverture.

À 14.77, Explorer écrira dans le fichier \Users\inter\AppData\Roaming\Microsoft\Windows\Recent\The Internet (2).lnk. En fait, le titre du fichier n'est pas « The Internet (2).lnk » mais plutôt « https--www.bing.com-searchq=xyz » où xyz est, littéralement, ce qui est montré dans l'aperçu d'écran de la figure 6-7 ci-haut. Le dossier « Recent » contient 164 fichiers dont les premiers datent du 23 septembre 2019, date du second laboratoire (Voir la figure 6-8). Presque tous les fichiers sont le reflet des requêtes passées auprès de Cortana et sont horodatés. Il y a donc moyen de reconstituer les activités de l'utilisateur sur une année. De 1.77 à 21.07, Windows lit 2 486 fois des fichiers .clb dans le dossier \Windows\Registration\.

| |
|---|
| L'analyse livre donc une **autre conclusion importante** : le navigateur Edge utilisé par Cortana laisse des traces importantes et précises sur les activités de l'utilisateur de Cortana. |

### 4.6. Analyse de RuntimeBroker

Ce processus fait l'objet de 3 459 entrées, dont 2 352 lectures dans le répertoire Windows\Registration notamment pour des fichiers clb. À 1.89, RuntimeBroker lit le fichier SearchUI.exe pour la première fois puis procède à la lecture de 52 fichiers dll. Malgré cette activité somme toute normale, ce groupe de processus présente peu d'intérêt.

### 4.7. Analyse de SpeechRuntime

Ce processus génère uniquement 1 193 entrées dans « File System » et « Process and threads » sans toutefois générer une seule écriture ni écriture de métadonnées. SpeechRuntime a lu 1 036 fois dans le répertoire \Windows\Registration ainsi que 130 fois des dll. Une fois ces deux derniers mis de côté, il ne reste que 27 lignes à analyser : SpeechRuntime.exe et un fichier tbres qui font l'objet de lectures seulement. Donc, SpeechRuntime ne laisse pas de trace permanente au niveau local.

| | | | |
|---|---|---|---|
| Add+contact&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569251027269.lnk | 1. | R... | 2019-09-23 3:03:49 PM |
| Close+calculator&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569254175528.lnk | 1. | R... | 2019-09-23 3:56:16 PM |
| Close+my+email+application&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569252203893.lnk | 1. | R... | 2019-09-23 3:23:25 PM |
| Close+my+email+application&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569252239651.lnk | 1. | R... | 2019-09-23 3:24:00 PM |
| Close+my+gallery&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569260628405.lnk | 1. | R... | 2019-09-23 5:43:49 PM |
| Close+paint&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569254399215.lnk | 1. | R... | 2019-09-23 4:00:00 PM |
| Close+people&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569251863899.lnk | 1. | R... | 2019-09-23 3:17:45 PM |
| Create+Alpha+dot+TXT&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569252477454.lnk | 1. | R... | 2019-09-23 3:27:58 PM |
| Create+an+appointment+next+Tuesday+with+the+dentist+at+13+holler&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=... | 1. | R... | 2019-09-23 3:36:07 PM |
| Dismiss+paint&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569254411523.lnk | 1. | R... | 2019-09-23 4:00:12 PM |
| Display+Donald+trump%27s+last+tweet&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=15692... | 1. | R... | 2019-09-23 5:45:47 PM |
| Display+mail&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569251949439.lnk | 1. | R... | 2019-09-23 3:19:10 PM |
| Donald+Trump+asking+this+done&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569261586... | 1. | R... | 2019-09-23 5:59:48 PM |
| How+do+you+call+a+person+living+in+Montreal&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&s... | 1. | R... | 2019-09-23 6:11:42 PM |
| How+do+you+call+a+person+living+in+mountain&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&... | 1. | R... | 2019-09-23 6:09:12 PM |
| How+high+over+the+level+of+St+Lawrence+River+is+the+Cliff+in+Quebec+City&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNS... | 1. | R... | 2019-09-23 6:13:44 PM |
| How+many+murders+was+there+in+Montreal+in+2017&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-... | 1. | R... | 2019-09-23 6:02:10 PM |
| How+many+petals+a+Daisy+has&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=15692633894... | 1. | R... | 2019-09-23 6:29:51 PM |
| Key+a+lameo+world+baseball&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1585688191083.l... | 1. | R... | 2020-03-31 8:56:32 PM |
| Look+in+people&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569251813253.lnk | 1. | R... | 2019-09-23 3:16:54 PM |
| Look+in+people+for+the+phone+number+of+job+in&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-C... | 1. | R... | 2019-09-23 3:17:17 PM |
| Open+notepad+and+create+Alpha+dot+TXT&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1... | 1. | R... | 2019-09-23 3:27:29 PM |
| Play+classic+music&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1585688701193.lnk | 1. | R... | 2020-03-31 9:05:02 PM |
| Read+out+loud+my+last+note&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569252675713.... | 1. | R... | 2019-09-23 3:31:16 PM |
| Tell+me+a+joke+in+French&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569261253529.lnk | 1. | R... | 2019-09-23 5:54:15 PM |
| What+is+a+tree+of+the+power+of+4&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1583382... | 1. | R... | 2020-03-05 4:25:59 AM |
| What+is+the+birth+date+of+joblo&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569251429... | 1. | R... | 2019-09-23 3:10:32 PM |
| What+is+the+birth+date+of+Joe+blow&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=156925... | 1. | R... | 2019-09-23 3:13:53 PM |
| What+is+the+foundation+date+of+Canada&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=15... | 1. | R... | 2019-09-23 2:53:30 PM |
| What+is+the+foundation+date+of+Canada&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=15... | 1. | R... | 2019-09-23 2:55:21 PM |
| What+is+the+foundation+date+of+Canada&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=15... | 1. | R... | 2020-02-27 10:25:16 PM |
| What+is+the+foundation+date+of+Canada&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=15... | 1. | R... | 2020-02-27 10:28:13 PM |
| What+is+the+foundation+date+of+Canada&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=15... | 1. | R... | 2020-02-27 10:31:14 PM |
| What+is+the+foundation+date+of+Canada&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=15... | 1. | R... | 2020-02-27 10:34:16 PM |
| What+is+the+foundation+date+of+Canada&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=15... | 1. | R... | 2020-02-27 10:37:14 PM |
| What+is+the+foundation+date+of+Canada&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=15... | 1. | R... | 2020-03-02 8:58:14 PM |
| What+is+the+foundation+date+of+Canada&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=15... | 1. | R... | 2020-03-05 8:56:15 PM |
| What+is+the+next+TV+program&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=158338240770... | 1. | R... | 2020-03-05 4:26:48 AM |
| What+is+the+next+TV+program+at+radio+Canada+Montreal&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlan... | 1. | R... | 2020-03-05 4:27:41 AM |

Figure 6-8 – Aperçu du contenu du répertoire
\Users\inter\AppData\Roaming\Microsoft\Windows\Recent

### 4.8. Conclusion de l'analyse Procmon

Process Monitor est un logiciel exceptionnel pour faire une analyse en profondeur du comportement d'une application. Même si les filtres choisis ont éliminé 88% des lignes, l'analyse nous a permis de relever les noms des acteurs les plus actifs au niveau de l'écriture de données et

de métadonnées. Ceci nous permettra d'être plus efficace dans l'analyse des traces locales générées par l'utilisation de Cortana.

Cette analyse nous a aussi permis de déterminer l'**ordre de démarrage de certains processus déclenchés par les processus principaux que sont svcHost et SearchUI. Quant à Explorer, RuntimeBroker et SpeechRuntime ils sont peu impliqués dans le démarrage de processus.** Toutefois, une réserve peut être inscrite quant à cette conclusion car l'étude n'a porté que sur une seule requête. Même si l'étude de cette requête semble confirmer ce qui a été découvert avec Volatility, il faudrait cet ordre de démarrage à l'aide de plusieurs requêtes. Cette analyse de multiples requêtes et types de requêtes permettrait aussi de déterminer ce qui se passe lorsque Cortana fait appel à des applications autres que Microsoft Edge, par exemple avec les requêtes 108 (« Calculator ») et 109 (« Paint »).

L'analyse de la requête 103 avec Process Monitor semble avoir donné des résultats plus précis que celle de toutes les requêtes avec Volatility. Une analyse encore plus en profondeur aurais pu être faites, par exemple, notamment en décomptant le nombre de fois qu'un processus lit un fichier (par exemple, SearchUI est lu 21 fois en 4 phases, grosso modo) et pourquoi c'est lu et relu de cette façon et pas d'une autre et est-ce que la façon de lire dépend du type de requête? On pourrait aussi chercher à savoir ce qui est lu et pourquoi seulement cet item est lu, car l'analyse de SearchUI à 1.87 indique que seuls 16 384 octets (soit exactement 4 pages mémoire standard pour Windows) sont lus au décalage 8 534 016 de SearchUI dont la taille est de 11 280 712 octets? On pourrait aussi examiner les dll utilisés et en quoi le type de requête influence les dll utilisés. Enfin, on pourrait compléter cet aspect en indexant le contenu des extrants Volatility et ProcMon à l'aide d'Autopsy SleuthKit pour couvrir ce qui aurait pu être oublié dans l'analyse de ces extrants.

# CHAPITRE 7 – RÉSULTATS ET OBSERVATIONS – IMAGE FORENSIQUE, SON ET COMMUNICATION INTERNET

## 1. Introduction

Le présent chapitre livre l'analyse des éléments autres que ceux de la mémoire vive. La section 2 divulguera les observations et résultats tirés de l'analyse des images bit-à-bit avec FTK Imager et Autopsy Sleuth Kit. La section 3 abordera l'analyse des fichiers de son. La section 4 énoncera nos découvertes quant à l'historique des données présumément stockées chez Microsoft et la section 5 fera une brève analyse des transmissions réseau.

## 2. Analyse des copies forensiques bit-à-bit

### 2.1. Avec FTK Imager

#### 2.1.1. Introduction

Afin d'examiner les éléments d'intérêt révélés par l'analyse de la mémoire vive avec Volatility, l'image forensique GMO03 du 23 septembre 2019 a été montée dans FTK Imager.

#### 2.1.2. ActivitiesCache.db-wal

Ce fichier, localisé dans \Users\inter\AppData\Local\ConnectedDevicesPlatform\ bb717611e150420e\, est un tampon pour stocker des données relatives aux recherches faites par le navigateur Edge. Kacos2000 [2020], dans un document très bien fouillé sur les artéfacts du répertoire « ConnectedDevicesPlatform », indique que ce répertoire sert à Windows 10 pour établir la frise chronologique des activités survenues sur son système d'exploitation. On retrouve dans ce fichier l'identifiant ECB32AF3-1440-4086-94E3-5311F97F89C4 que Kacos2000 [2020] relie à Microsoft Edge.

Certaines « pages » (pour adopter la terminologie des mémoires volatiles) ont une structure similaire à des bases de données SQLite. Dans la section « Observations sur les processus et les fils » de l'analyse du processus SvcHost avec Volatility, il est rapporté que Mikhailov [2019] affirme les fichiers ActivitiesCache.db sont effectivement des bases de données SQLite version 3.

Pour ce fichier, l'analyse ProcMon de la requête 103 de mars 2020 révèle qu'il adopte une structure d'écriture 24 puis 4096 octets. Au décalage 0d37 200 commence une zone de 4 120 octets (24 + 4 096) contenant le texte des réponses à une quinzaine de requêtes soumises à Cortana ce jour-là. Ce genre de zone se répète, à quelques octets près, plusieurs fois dans ce même fichier. Voici un exemple d'une de ces entrées : https://www.bing.com/search?q=How+high+over+the+level+of+St+Lawrence+River+is+the+Cliff+in+Quebec+City&input=2&**nclid=0B1E3176C6678903669C87BA79585DFE**&FORM=WNSHCO&cc=CA&setlang=en-CA

La partie soulignée et en gras se répète pour toutes les réponses de requête et **semble être un identifiant particulier à Cortana** puis que lorsqu'une requête est passée manuellement à Bing, on obtient plutôt quelque chose comme « cvid=nombre hexadécimal ». Peu d'information sur ce nclid.

On note aussi que l'ordinateur GMO a été configuré pour utiliser l'anglais canadien (&setlang=en-CA), ce qui peut donner des indices sur la nationalité ou la localisation de l'utilisateur.

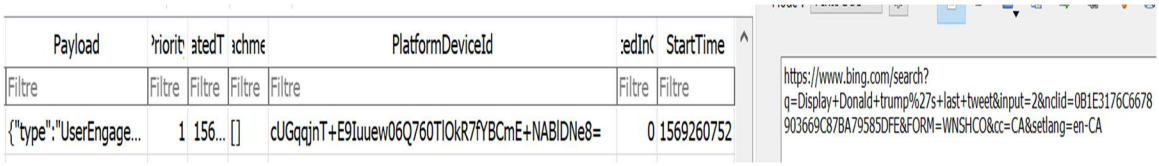



Figure 7-1 – Aperçu d'écran dans DB SQLiteBrowser de la requête #111 (alias #11, alias opération 346) soumise à Cortana à 17h45:35 GMT le 23 septembre 2020

En ouvrant ActivitiesCache.db avec DB SQLiteBrowser on a accès à beaucoup plus d'informations et elles sont en clair. La figure 7-1 présente les données de l'opération #346 correspondant à la requête « Display Donald Trump's last tweet » soumise à Cortana le 23 septembre 2019 à 13h45:35 heure avancée de l'Est. En convertissant le « StartTime » (1569260752, une date de format Unix) en date-heure intelligible aux humains, on obtient 23 septembre 2019 à 17h45:52 GMT, soit 17 secondes après le début de la requête noté par les expérimentateurs. Ce décalage existe pour toutes les entrées que nous avons examinées et comparées au chrono des expérimentateurs, mais la valeur du décalage pour ces items varie de 17 à 31. On peut faire l'hypothèse que les écritures ne sont faites qu'une fois l'opération complétée.

D'autres rubriques non présentes dans l'illustration à la Figure 7-1 donnent des informations sur le logiciel employé (« AppId » (identifiant d'application) = Microsoft Edge _8wekyb3d8bbwe!) et sur le fuseau horaire utilisé (« Payload » (charge utile) = Fuseau horaire Toronto) par l'ordinateur au moment de la requête. Nous pouvons aussi obtenir l'heure à laquelle GMO s'est connecté la première fois (23 septembre 2019 à 02h19:50 GMT) et sur quelle adresse (http://10.42.0.1:8080/), donnée confirmée par les notes prises par les expérimentateurs. Nous pouvons aussi déterminer, à l'aide de ce fichier, que GMO s'est connecté sur http://mitm.it/ à 13h20:12 le 23 septembre 2019, ce qui est confirmé de la même façon.

D'autres entrées indiquent divers logiciels activés pendant l'expérimentation : MS People, MS Paint, Notepad… **Les fichiers ActivitiesCache sont donc un historique fidèle des utilisations faites du système d'exploitation par l'utilisateur et constituent donc une mine d'informations privées sur les activités de l'utilisateur.**

2.1.3. Fichiers de prélecture

BackgroundTask, SvcHost et Microsoft Edge. Ces fichiers fournissent des dates-heures de démarrages exécutés avant l'exécution de l'image forensique ainsi que les DLL utilisés. Peu d'information historique spécifique à Cortana. Par ailleurs, les programmes en question desservent beaucoup d'application Windows et sont donc susceptibles de démarrer, mais pas juste pour Cortana.

2.1.4. InetCache

Dans le répetoire \Users\inter\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\AC\INetCache\, on trouve des sous-répertoires contenant les fichiers « temporaires » de navigation internet (Les « TIF » : « Temporary Internet Files »), fichiers très révélateurs de l'activité de l'utilisateur de Cortana.

\DSDDTUNG\th(3).jpg a finalisé sa sauvegarde à 18h04:17 GMT suite à la requête #116 (Who was Jean Brillant?). L'expérimentateur a noté que l'énoncé de la requête s'est produite à 14h03:55 HE. L'image du jpg recueilli est celle de l'acteur Jon Brion et non celle du lieutenant Jean Brillant (Voir la figure 7-2).

| |
|---|
| Le malentendu illustre **deux points importants au point de vue forensique**. **Premièrement**, Cortana n'est pas en mesure d'interpréter correctement certains éléments d'une requête lorsque l'énoncé de la requête est affecté par un accent autre que la langue installée sur le système d'exploitation d'où s'exécute Cortana. **Deuxièmement**, un investigateur numérique recueillant cette preuve ne devrait pas l'interpréter comme étant hors de tout doute raisonnable sous peine d'incriminer une innocente personne. |

| | |
|---|---|
|  |  |
| Jon Brion – Acteur et producteur de films | Jean Brillant – Lieutenant – Croix de Victoria – Héros canadien de la bataille de la Somme |

Figure 7-2 – Malentendu entre Cortana et l'utilisateur

\DSDDTUNG\th(6).jpg présente une vue de la tour Eiffel. Le fichier a terminé sa sauvegarde à 18h18:42 GMT et a été téléchargé suite à la requête #121c « What is the forecast in Paris for today? » soumise à Cortana à 14h18:30 HE. Cette information peut révéler l'intérêt de l'utilisateur pour Paris, par exemple l'intention de s'y rendre ou la présence d'une personne chère en ce lieu ou un simple intérêt bien banal.

\DSDDTUNG\th(9).jpg présente une photo de Mickey Mouse dont la sauvegarde s'est terminée à 18h23:04 GMT et correspondant à la requête #124 « Who is Mickey Mouse? » qui a été soumise

à Cortana à 14h22:15. Cette information pourrait révéler la présence d'un enfant sur les lieux où est opéré l'ordinateur, par exemple.

\DSDDTUNG\speech_render[1].htm est un fichier contenant le résultat retourné par Edge suite à la requête #123a « How many petals a daisy has? ». La requête a été soumise à 14h29:40 HE et le retour est marqué à 18h29:49 GMT

On remarque qu'il y a un délai entre ce qui est révélé par FTK Imager et la réalité notée par l'expérimentateur. Ce sont des délais qui vont de 9 à 49 secondes. Même si ces délais sont inégaux en durée, l'investigateur numérique ferait bien de ne pas conclure trop rapidement. La question sur Jean Brillant a nécessité que l'utilisateur de Cortana répète sa question 3 fois avant que Cortana n'interprète la requête (Incorrectement, soit! Mais interprétée tout de même).

\GVM9U1TF\th(1).jpg et th(7).jpg sont des images du lac Titicaca (requête #112) et à Paul Chomedey de Maisonneuve (requête #118) avec 12 et 32 secondes de délai respectivement. Ce dernier délai a été plus long car l'utilisateur a dû répéter 2 fois la question pour que Cortana fasse la recherche. Encore une fois, l'investigateur d'un tel phénomène doit se méfier car Cortana n'a réagi qu'à une prononciation du style « Pââl Tchômeudeille dey Maizônn-Nouvvv ». L'utilisateur se bute, encore une fois, sur la langue configurée du Cortana utilisé.

> Non seulement l'anglais doit être utilisé, mais l'accent doit être correct pour que Cortana comprenne bien. **L'équivoque est donc possible**, ce qui pourrait introduire un **doute raisonnable** quant aux intentions ou aux actions de l'utilisateur si ces fichiers constituaient des preuves d'un crime.

## 2.2. Avec Autopsy SleuthKit

### 2.2.1. Introduction

Difficile de se rendre compte de la quantité de données stockées sur un disque dur. Le disque dur à analyser est un 240 gigaoctet rempli à seulement 20%. Si on avait l'équivalent en format papier, imprimé recto-verso, on aurait besoin d'un camion de 50 mètres cubes pour y loger tout ce papier. Il contient 496 885 artéfacts de divers types. Heureusement, 90% environ de ces artéfacts sont des fichiers exécutables, des pilotes ou autres fichiers destinés à l'opération du système d'exploitation.

Malheureusement, le 10% qui reste fait quand même plus de 49 000 artéfacts. À raison d'un artéfact par 60 secondes d'examen par artéfact, l'examen durerait 820 heures.

Afin de limiter le champ de recherche, 40 mots-clés ont été recherchés dans les images forensiques générées le soir du 23 septembre 2019. Cette méthode n'est efficace, bien sûr, que si les éléments recherchés sont des éléments texte. Pour les éléments graphiques, audio et vidéo, on doit adopter une approche fondée sur la nature des fichiers (en se basant sur les extensions des fichiers ou sur leur signature).

Le choix d'un mot-clé efficace est difficile. Par exemple, le mot-clé « foundation » de la question #103 (« What is the foundation date of Canada? ») a été choisi et a résulté en 9 762 occurrences. En effet, Microsoft (le fabricant de Windows) utilise des expressions comme « Microsoft-Windows-Foundation-Default-Security » pour désigner les fondements de la sécurité sur leur système d'exploitation.

Les mots-clés qui ont bien fonctionné sont ceux mentionnés au tableau 6-4 ci-dessous.

| | | | |
|---|---|---|---|
| Chomedey (1) | Daisy (60) | Titicaca (10) | Canada (1221) |
| Maisonneuve (1) | Dentist (89) | Trump (125) | Mickey (39) |
| Montreal (123) | Doctor (169) | Tweet (79) | Paintbrush (90) |
| Quebec City (16) | Joke (109) | Brillant (44) | St-Lawrence River(2) |

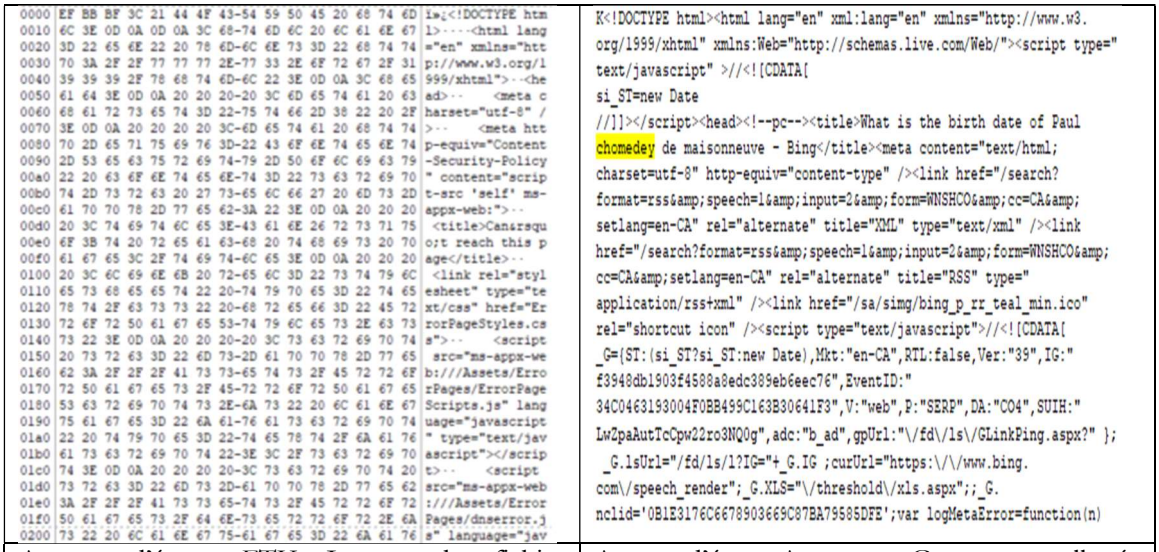
Tableau 7-1 – Tableau des mots-clés (nombre d'occurrences) efficaces soumis à Autopsy SleuthKit pour traiter le contenu de l'image forensique du 23 septembre 2019

### 2.2.2. Grappes non-allouées

À ce point de la lecture, il est pertinent de revoir certains concepts de forensique énoncés au chapitre 1. Notamment le concept de grappes non-allouées. Des mots-clés ont été détectés dans certains secteurs non-alloués et ceux-ci sont entourés d'une phrase complète. Par exemple, Autopsy a trouvé les requêtes #108 (Chomedey et Maisonneuve) et #123a (Daisy) au complet et une seule fois. Par contre, d'autres requêtes ont été retrouvées, au complet et à plusieurs reprises. Notamment les requêtes #115c (« Montreal » sans accent et à l'anglaise), #107b (« Dentist »), #113b (« Joke »), #103 (« Canada »), #111 (« Tweet ») et #124 (« Mickey »). Ensuite, d'autres requêtes ont été retrouvées à plusieurs reprises, au complet et dans plusieurs grappes non-allouées, comme la requête #120 (« Quebec City » et « St-Lawrence River »).

La requête #114 (« Donald Trump, est-ce qu'il décida d'aller à Boston? ») constitue un cas à part. Elle a été créée pour semer la confusion et voir à quel point Cortana pouvait se débrouiller avec des questions qui n'étaient pas dans la langue programmée. Nous avons retrouvé ce que Cortana en a compris, c'est-à-dire peu de chose.

La structure des données où reposaient les requêtes mentionnées au paragraphe précédent variaient. Premièrement, les indices pour la requête #108 (Chomedey et Maisonneuve) sont présentes dans une structure typique d'un fichier html. La figure 7-3 compare un fichier html quelconque et le contenu de la grappe Unalloc_904084_19254681600_21337923584. Les deux aperçus d'écrans montrent les caractères <!DOCTYPE html> qui, sans être une signature de ce type de fichier (pour un html, la signature est 0xEFBBBF) en est tout de même un élément typique.






| Aperçu d'écran FTK Imager du fichier \Windows\WinSxS\amd64…\dnserror.html | Aperçu d'écran Autopsy – Grappe non allouée Unalloc_904084_19254681600_21337923584 |
|---|---|

Figure 7-3 – Aperçus d'écran html tirés de l'image forensique GMO 23 septembre 2019

Deuxièmement, les requêtes #115c (Montreal) et #120 (Quebec City) dont la structure est celle apparaissant dans la barre d'adresse internet lorsque le navigateur Bing répond à ces requêtes. Voir la figure 7-4.

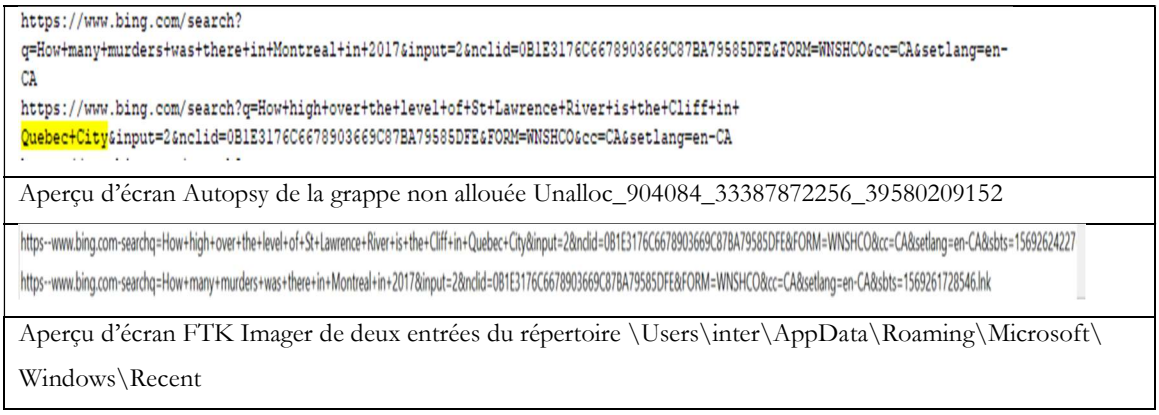



Figure 7-4 – Aperçus d'écran « Recent » tirés de l'image forensique GMO 23 septembre 2019

Enfin, la requête #113b (« Joke ») à la figure 7-5 présente une structure de données différente des deux précédentes mais dont la nature nous est inconnue. **On voit donc, ici aussi, que Cortana laisse plusieurs traces des requêtes de son utilisateur**.


Figure 7-5 – Aperçus d'écran « non-identifiés » tirés de l'image forensique GMO 23 septembre 2019

### 2.2.3. Balances de fichiers

Dans la balance du fichier orphelin prnmngr.vbs-slack, on trouve la requête 119b au complet. Dans la balance du fichier orphelin Package_702_for_KB4516058~31bf3856ad364e35~amd64~~10.0.1.5.mum-slack, on retrouve la requête #115c (« Montreal ») au

complet. Dans les balances des fichiers SRUtmp.log-slack et ActivitiesCache.db-wal-slack on trouve à 14 reprises (chacun) la requête 120 (« Quebec City ») au complet. Dans les fichiers ntuser.dat.LOG1-slack et Package_702_for_KB4516058~31bf3856ad364e35~amd64~~10.1.1.5.mum-slack, on retrouve la requête 113b (« Joke ») au complet à 2 reprises dans chaque fichier. Dans les fichiers ntuser.dat.LOG1-slack, DataStore.edb-slack et HXACCOUNTS.EXE-30765DC.pf.slack, on retrouve la requête 103 (« Canada ») au complet, 1 à 4 occurrences dans chaque fichier.

#### 2.2.4. Fichiers orphelins

Dans les fichiers orphelins stddtype.gdl, Package_1583_for_KB4516058~31bf3856ad364e35~amd64~~10.0.1.5.mum, Package_702_for_KB4516058~31bf3856ad364e35~amd64~~10.0.1.5.cat et prnmngr.vbs on retrouve la requête 119b (« Montreal ») au complet plusieurs fois.

#### 2.2.5. Fichiers du système de fichiers

Le système d'exploitation de GMO était un Windows 10 installé dans une partition formatée en NTFS avec des grappes de 4 096 octets. Autopsy a détecté les mots dans un certain nombre de fichiers constituant de l'infrastructure NTFS.

*\$Extend \$UsnJrnl:$J* : Ce fichier est un flux alternatif de données (« Alternate Data Stream) qui est localisé à la racine du répertoire $Extend pour Autopsy, mais dans le répertoire $RmMetada pour FTK Imager, ce qui semble confirmer ce que dit Manumation [2008] au sujet des flux alternatifs de données. On trouve dans ce fichier les requêtes 113b, 114, 115c, 116 et 119b. Ces requêtes sont en fait le nom des lnk dont le titre correspond à ces requêtes (un peu comme si on avait inscrit dans ce fichier une partie de la liste du répertoire « Recent » (\Users\inter\AppData\Roaming\Microsoft\Windows\Recent\). Ce fichier a une taille de 731 Mo. Curieusement, ce fichier présente la particularité que les dates de création, de dernier accès, de dernière modification et de dernier changement sont toutes égales au 24 mai 2018 à 19h11:46 HE! Curieux en effet puisqu'on y voit des modifications et des accès qui ne peuvent provenir que du 23 septembre 2019. Entre chacune de ces occurrences, on trouve le nom de fonctions ou de processus. Certains de ceux-ci ont déjà été analysés (ActivitiesCache, thumbcache et d'autres), d'autres sont peu intéressants (notamment des fichiers pf dont on sait qu'ils sont appelés par la

fonctionnalité Cortana), d'autres enfin suscitent un intérêt moyen (notamment les fichiers gif, png et jpg). Pour ce qui reste, il y a des fichiers .dat, et .log couverts dans l'une des sections ci-dessous.

*\$LogFile* : Ce fichier est localisé à la racine du média à partir duquel on exécute Windows (Normalement, le C:). Carrier [2005] affirme que ce fichier « *Contains the journal that records the metadata transactions* ». Ce fichier présente la particularité que tous les tampons horodateurs indiquent 24 mai 2018 à 22h58:04 HE. On y trouve uniquement la requête 115c sous la forme du titre d'un fichier lnk : « https--www.bing.com-searchq=How+many+murders+was+there+in+Montreal+in+2017&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569261728546.lnk ».

*\$MFT* : Ce fichier est à la racine du média à partir duquel on exécute Windows (normalement, le C:). Revoir la théorie à ce sujet au chapitre 1. Plusieurs fichiers lnk ont été retracés. Ce sont, dans cet ordre, ceux liés aux requêtes 111, 113b, 114, 115c, 116, 119b, 120, 123a. On a aussi au-delà de 500 de noms de fichiers, dont certains font partie des fichiers déjà vus précédemment dans ce mémoire et d'autres qui pourraient s'avérer intéressants à examiner.

### 2.2.6. Fichiers désencastrés

Réponse à une requête : Certains fichiers désencastrés présentent une ou plusieurs réponses à une requête soumise à Cortana. Par exemple : https://www.bing.com/search?q=How+high+over+the+level+of+St+Lawrence+River+is+the+Cliff+in+Quebec+City&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA (l'auteur souligne)

*Texte en clair* : D'autres fichiers désencastrés contiendront le texte de la requête, sans les ajouts insérés par le navigateur. Par exemple : « How many murders was there in Montreal in 2017 ».

### 2.2.7. Extensions variées

Autopsy assigne à chaque fichier désencastré une extension qui semble, basé sur mes observations, basée sur la signature du fichier. On a, par exemple, le mot-clé « joke » (soit de la requête #113b

ou de la #113a) dans un fichier exe (signature 0x4D5A) dans un fichier .java (sans signature), dans un fichier .doc (signature 0xD0DV11E0) et  dans un fichier .reg (signature 0x72656766).

On a aussi des fichiers qui sont… quelque chose d'autre qui ressemble à du xml mais qui est qualifié de txt parce qu'il n'y a pas de signature identifiée par Autopsy. Voir figure 7-6 ci-dessous.


```
{
    "Label": "Faits",
    "Description": "Personnes, lieux, choses",
    "Id": "941433CA-F09A-4F88-B628-373D85D2E8AA",
    "Tips": [
        {
            "Label": "Quelle est la hauteur du Chrysler Building ?"
        },
        {
            "Label": "Quelle est la hauteur du Mont Blanc ?"
        },
        {
            "Label": "Lac Titicaca altitude"
        },
        {
            "Label": "Date de fondation de Médecins du monde"
        }
    ]
```


Figure 7-6 – Aperçu d'écran du contenu du fichier désencastré f0585616.txt

*Ligne de code* : On a aussi les fichiers désencastrés qui semblent être des lignes de code d'un programme dont le langage n'a pas pu être identifié. Ce fichier est qualifié de png à cause de sa signature 0x89504E47, mais il ne contient que des lignes de code du style présenté à la figure 7-7.

### 2.2.8. Fichiers .dat

*NTuser.dat* : Ce fichier fait partie de la base de registre Windows et il est localisé à la base du répertoire de l'utilisateur en session. Lorsqu'on présente une preuve en Cour, on doit démontrer qui était le propriétaire de la preuve avant qu'on ne s'en saisisse. On appelle cela « mettre la preuve entre les mains de… » et généralement, il s'agit de la mettre entre les mains du suspect. Le fichier NTuser.dat est amendé selon les actions de l'utilisateur en session (« logged in user »). Bien sûr, en Cour, on doit démontrer que notre suspect était au clavier de cet ordinateur et utilisait le compte de cet utilisateur au moment de l'infraction, et ce selon le niveau de preuve requis. Les requêtes 109 (Ouvrir puis fermer le logiciel Microsoft Paintbrush) ont été soumises à Cortana à partir de 11h59:25 le 23 septembre 2019. Quelque temps après 10h56:24 (on ne peut pas être plus précis) Windows a écrit dans NTuser.dat au décalage 0d1 843 684

« search?q=Close+paint&input=2&nclid=0B1E3176C6678903669C87BA79585DFE&FORM=WNSHCO&cc=CA&setlang=en-CA&sbts=1569254399215 ».

Plusieurs autres requêtes apparaissent ensuite, mais l'ordre de succession dans ce fichier ne correspond pas à l'ordre dans lequel les requêtes ont été soumise. Par exemple, la requête 107b suit la requête 109 et est suivi de la requête 115c. Ceci remet en question la date relevée peu avant d'atteindre la requête 109. Bien sûr, on retrouve des entrées similaires dans les fichiers NTuser.dat.log1 et NTuser.dat.log2. Ces deux derniers sont créés lorsque NTuser.dat a besoin d'espace. Windows créé alors des entrées dans ces fichiers. On ne retrouve pas les mêmes requêtes dans « log1 » que dans « log2 » et NTuser.dat.

```
  /** When we find an element, maybe there's one that's just a little bit better... */
  evenBetterElement(node, origRect) {
    let el = node.parentNode;
    const ELEMENT_NODE = document.ELEMENT_NODE;
    while (el && el.nodeType == ELEMENT_NODE) {
      if (!el.getAttribute) {
        return null;
      }
      const role = el.getAttribute("role");
      if (role === "article" || (el.className && typeof el.className === "string" && el.className.search("
tweet ") !== -1)) {
        const rect = Selection.getBoundingClientRect(el);
        if (!rect) {
          return null;
        }
        if (rect.width <= MAX_DETECT_WIDTH && rect.height <= MAX_DETECT_HEIGHT) {
          return el;
        }
        return null;
      }
      el = el.parentNode;
    }
    return null;
  },
```

Figure 7-7 – Lignes de code dans un fichier identifié comme png de par sa signature

*WebCacheV01.dat* : Ce fichier se situe aussi dans l'espace de l'utilisateur Windows \Users\inter\AppData\Local\Microsoft\Windows\WebCache. C'est un fichier d'une taille de 40 Mo qui reflète les activités Internet de l'utilisateur. À la différence d'autres fichiers vus précédemment, les informations sont horodatées, en heure GMT, non pas du début de la requête par l'utilisateur, mais du moment où Cortana répond à la requête.

{8BFA40A1-CEDE-4E6E-8194-953BE206A92C}.dat

```
18d0 00 00 00 00 83 AD AC AB-28 00 00 00 4B 00 69 00 ·····-¬«(···K·i·
18e0 6C 00 6C 00 65 00 72 00-20 00 70 00 65 00 72 00 l·l·e·r· ·p·e·r·
18f0 66 00 6F 00 72 00 6D 00-61 00 6E 00 63 00 65 00 f·o·r·m·a·n·c·e·
1900 20 00 64 00 69 00 61 00-6C 00 20 00 64 00 65 00  ·d·i·a·l· ·d·e·
1910 6C 00 65 00 75 00 7A 00-69 00 61 00 6E 00 20 00 l·e·u·z·i·a·n· ·
1920 49 00 6E 00 64 00 69 00-61 00 6E 00 00 00 00 00 I·n·d·i·a·n·····
```

Figure 7-8 – Interprétation d'une requête en français par Cortana

{8BFA40A1-CEDE-4E6E-8194-953BE206A92C}.dat

```
2100 00 00 02 94 00 00 FF D8-FF E0 00 10 4A 46 49 46 ·····ÿØÿà··JFIF
2110 00 01 01 01 00 00 00 00-00 00 FF DB 00 43 00 06 ··········ÿÛ·C··
2120 04 05 06 05 04 06 06 05-06 07 07 06 08 0A 10 0A ··············
2130 0A 09 09 0A 14 0E 0F 0C-10 17 14 18 18 17 14 16 ··············
2140 16 1A 1D 25 1F 1A 1B 23-1C 16 16 20 2C 20 23 26 ···%···#··· , #&
2150 27 29 2A 29 19 1F 2D 30-2D 28 30 25 28 29 28 FF ')*)··-0-(0%()(ÿ
2160 DB 00 43 01 07 07 07 0A-08 0A 13 0A 0A 13 28 1A Û·C···········(·
2170 16 1A 28 28 28 28 28 28-28 28 28 28 28 28 28 28 ··((((((((((((((
2180 28 28 28 28 28 28 28 28-28 28 28 28 28 28 28 28 ((((((((((((((((
2190 28 28 28 28 28 28 28 28-28 28 28 28 28 28 28 28 ((((((((((((((((
```

Figure 7-9 – Présence d'un fichier jpeg encastré dans un fichier .dat


Figure 7-10 Jpeg du .dat une fois désencastré[17]

[17] L'image est de piètre qualité mais est présentée telle qu'elle est au moment de son désencastrement et il n'est pas possible de la rendre plus claire. Nous posons l'hypothèse que Microsoft a choisi de sauvegarder en basse qualité afin d'épargner de l'espace sur le média opérant Cortana.

Les requêtes ne sont pas sauvegardées dans l'ordre dans lequel elles sont soumises à Cortana. En plus des dates-heures, on a d'autres renseignements comme le nombre de visites faites aux sites web indiqués.

Au décalage 0x1807F0, il y a une entrée indiquant que l'utilisateur a configuré un mandataire (« proxy ») à 06h12:26 HE le 23 septembre 2019. À la suite de cette entrée, on voit toute la préparation technique effectuée (incluant notamment une entrée « ms-cortana : settings/GetApp ») jusqu'à ce que l'utilisateur commence à soumettre des requêtes à Cortana. Au décalage 0x182970, on trouve la requête 103 horodatée 10h10:30 HE, ce qui est erroné puisque cette requête a été soumise à Cortana à 10h55:05 HE. Au décalage 0x183000, on a la requête 104a, horodatée de 11h03:49 HE, ce qui est correct. La dernière entrée de ce bloc dans WebCacheV01.dat est au décalage 0x187B2F et concerne la requête 124a. Il n'y a aucune trace des requêtes 124b, 124c et 124d. Peu avant, à 0x187806, on a une entrée qui dit Visited : inter@ms-cortana://settings/CustomVoiceWizard. Lors de l'expérience, avant de soumettre la requête 124a, Cortana a été configuré pour ne répondre qu'à ma voix. Malheureusement, cette activité n'est pas horodatée.

{88FA40A1-CEDE-4E6E-8194-953BE206A92C}.dat : La présence de ce fichier a été détectée dans une expérience préliminaire faite au mois de décembre 2018. La requête soumise à Cortana, en français, était « Quelle est la profondeur de l'océan Indien? ». Cet énoncé a été interprété par Cortana anglais comme « Killer performance dial deleuzian Indian » (Figure 7-8) et retourné sous sa forme textuelle. Si l'erreur est humaine, elle est autant plus numérique! Lors de l'examen de ce fichier, une signature 0xFFD8, signature de début de fichier jpeg, a été détectée ainsi que la signature de fin de fichier jpeg 0xFFD9. Voir Figure 7-9. Cette portion du fichier, une fois désencastré, nous fournit l'image de la Figure 7-10. Ceci nous donne donc deux leçons. D'abord, pour les organismes d'application des lois, ceci nous montre que les logiciels commettent des erreurs et qu'on doit se garder de conclure trop rapidement, qu'un meurtrier est recherché. Enfin, pour les utilisateurs, que Cortana peut les mettre dans le pétrin!

#### 2.2.9. Fichiers de journalisation (.log)

Les événements survenant sur un ordinateur sont enregistrés dans un fichier, notamment afin de permettre aux utilisateurs ou aux administrateurs de gérer les pannes. V100027.log et autres .log

numérotés : Ces fichiers se trouvent dans le répertoire WebCache. On y retrouve les requêtes 105a horodatée 11h19:10 HE et la requête 104d de 11h17:19 HE (ouverture de Microsoft People pour rechercher le numéro de téléphone de Jo Binne) et de 11h17:45 HE (fermeture de Microsoft People). On retrouve aussi l'ouverture de la calculatrice, requête 108, dans V100028.log à 11h56:16 HE. Dans 100029.log, on voit les requêtes 110a et 110b (Microsoft Gallery) vers 12h43:49, ce qui constitue un horodatage erroné puisque cette activité s'est tenue de 12h02 à 12h05. On retrouve aussi, dans ce dernier fichier la requête 111.

Comme on peut le voir, ces fichiers sont à peu près la seule trace laissée par l'ouverture de logiciel par le biais de Cortana, bien que ces fichiers ne soient pas exclusivement réservés à ce type d'activité.

#### 2.2.10. Cas spéciaux

*Stddtype.gdl* : Sur Internet, il y a peu d'informations sur ce fichier. Mais au tout début de ce fichier, on trouve le texte suivant : « *stddtype.gdl - this file contains templates that define all MS standard datatypes *% that appear in GPD and GDL files* ». Ce fichier comme tel ne contient rien se rapportant à Cortana mais sa balance de fichier est très prolixe car on y retrouve intégralement (et dans cet ordre) les requêtes 116, 119a, 119b, 120, 104c et 103. Le plus étonnant est que ce gdl est un fichier système et que donc, en principe, une fois l'installation du système d'exploitation terminée, il n'est jamais modifié (d'ailleurs, quand on consulte la liste des fichiers extraite à l'aide de FTK Imager de l'image forensique de GMO, la taille de ce fichier ne varie jamais). Comme il ne varie pas en taille, sa balance de fichier aurait dû être constitué uniquement de zéros (puisque le caractère 0x00 a été utilisé pour aseptiser le média avant d'y installer Windows). Nous supposons qu'il a été déplacé ou que Windows a eu besoin d'un stddtype.gdl à un moment spécifique et qu'il l'a créé à un endroit où il y avait précédemment un fichier contenant ces requêtes.

*AutomaticDestinations* : Ces fichiers (au nombre de 7) sont dans un sous-répertoire \Users\inter\Roaming\Microsoft\Windows\Recent\. Ces fichiers ont tous un titre du format « hexadécimal_8_octets.automaticDestinations-ms » sans extension. La signature numérique de ce fichier est 0xD0CF11E0A1B11AE1. Pour cette signature, le site FileSig répertorie 21 types de fichiers, dont 14 se rapportent à des types de fichiers créés par Microsoft (notamment les .doc, .xls et les .msi, les fichiers d'installations Windows). Le plus gros de ces fichiers fait 42 ko et le plus

petit 2 ko. Ils datent tous du 23 septembre 2019 (date de dernière modification). Le premier a été sauvegardé à 7h05 HE et le dernier 14h29 HE (9d1f905ce5044aee.automaticDestinations-ms). Dans le plus gros fichier, à partir du décalage 0x0EEA, on a des entrées dont le texte correspond à des requêtes soumises à Cortana. Par exemple, au décalage 0x3680, on a une entrée horodatée 23 septembre 2019 13h29:51 de 368 octets se rapportant à la requête 123a soumise à 14h29:40. Encore ici, on voit un écart entre le début réel de la soumission de la requête et son inscription comme trace dans les fichiers pertinents.
Dans le fichier 5f7b5f1e01b83767.automaticDestinations-ms (7 ko, dernière modification à 14h25:52) on retrouve une entrée disant « ms-cortana://CapabilitiesPrompt/?RequestedCapabilities=SpeechLanguage,InputPersonalization,Microphone&QuerySource=Settings ». Or, peu avant ce moment, l'utilisateur a configuré Cortana pour qu'il n'obéisse qu'à la voix de l'utilisateur. Le reste du fichier tourne alentour de ce concept. On a donc un fichier sauvegardé à 14h29 contenant des données sur des événements qui se sont déroulés à 13h30 et un autre sauvegardé à 14h25 contenant des données sur des événements qui se sont déroulés vers 14h20. Mon hypothèse est que plusieurs de ces fichiers sont exploités de façon concomitante.

*Fichiers .json* : Le fichier \Users\inter\AppData\Local\Packages\Microsoft.Windows.Cortana.cw5n1h2txyewy\LocalState\Cortana\Upload\Contacts\contacts.json a été créé à 15h30:45 GMT et contient les noms des deux personnes présentes dans le carnet d'adresses de GMO : Jo Binne et Jo Bleau. La requête 104b, demandant la date de naissance de Jo Bleau, a été soumise à 15h13:25 GMT. Cette requête a été soumise sans succès et, en conséquence, les requêtes 104c à 104g, n'ont pas été soumises. Ceci démontre que Cortana, réagissant avec un délai de 17 minutes, a importé dans son périmètre d'opération le carnet d'adresse.

| |
|---|
| Tous ces fichiers contiennent des traces de l'activité de l'utilisateur. En faisant des recoupements pertinents au niveau des données (par exemple le contenu du fichier comme tel : mots de la requête), des intradonnées (par exemple les types de fichiers formant la réponse) et des paradonnées (par exemple les dates ou les périodes d'activité dans la journée), on peut dresser un portrait assez précis et intime de l'utilisateur. |

### 2.3. Les répertoires dédiés

Il s'agit des sous-répertoires non couverts ci-haut qui se trouvent dans le chemin \Users\inter\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy. Dans \Users\...\LocalState\Signals\collection\UserPresence, on a le fichier {50842d11-ab93-457d-aa88-b94ba19be840} avec le contenu présenté au tableau 6-6.

{"Type":"UserPresence","Time":"2019-09-23 15:54:33.785","Reading":{"Present":false}},
{"Type":"UserPresence","Time":"2019-09-23 15:55:38.651","Reading":{"Present":true}},
{"Type":"UserPresence","Time":"2019-09-23 16:10:41.299","Reading":{"Present":false}},
{"Type":"UserPresence","Time":"2019-09-23 17:38:44.658","Reading":{"Present":true}},
{"Type":"UserPresence","Time":"2019-09-23 18:44:08.512","Reading":{"Present":false}}

Tableau 7-2– Contenu du fichier {50842d11-ab93-457d-aa88-b94ba19be840}

Sachant que la première requête après le redémarrage forcé de 10h52 a été soumise à 10h55:05, la première entrée horodatée à 15h54:34 (à l'évidence, les heures sont en GMT, donc 11h54:34 HE) est un peu intrigante : Windows a donc attendu une heure avant d'horodater ce fichier alors que l'activité se déroulant sur cet ordinateur était soutenue. Si on considère la troisième occurrence, l'heure indiquée survient peu après le début de la pause-dîner débutant à 12h05 HE. La quatrième occurrence survient 20 secondes après la première requête après la pause-dîner (requête 110e à 13H28:25) et la dernière moins de 10 minutes après la dernière requête soumise à Cortana le 23 septembre 2019 à 14h34:40.

Deux autres points à souligner : 1) Le fichier ne contient aucune donnée horodatée d'avant le redémarrage. 2) En date du 8 octobre 2020 à 13h29 HE, le sous-répertoire « Signals » et ses sous-répertoires ont été examinés pour constater la disparition complète du fichier {50842d11-ab93-457d-aa88-b94ba19be840} et l'apparition de trois autres fichiers (dont 2 ont été effacés le 25 septembre 2020 et le 2 octobre 2020).

Posons l'hypothèse que ces fichiers sont peut-être effacés de façon sécuritaire à chaque redémarrage. Ou que cette suppression est la conséquence de l'utilisation d'un disque dur SSD comme disque dur de GMO. Mais pourquoi alors y a-t-il trois (et non pas un) nouveaux fichiers à cet endroit.

| Structure du 23 septembre 2019 | Structure du 8 octobre 2020 |
|---|---|

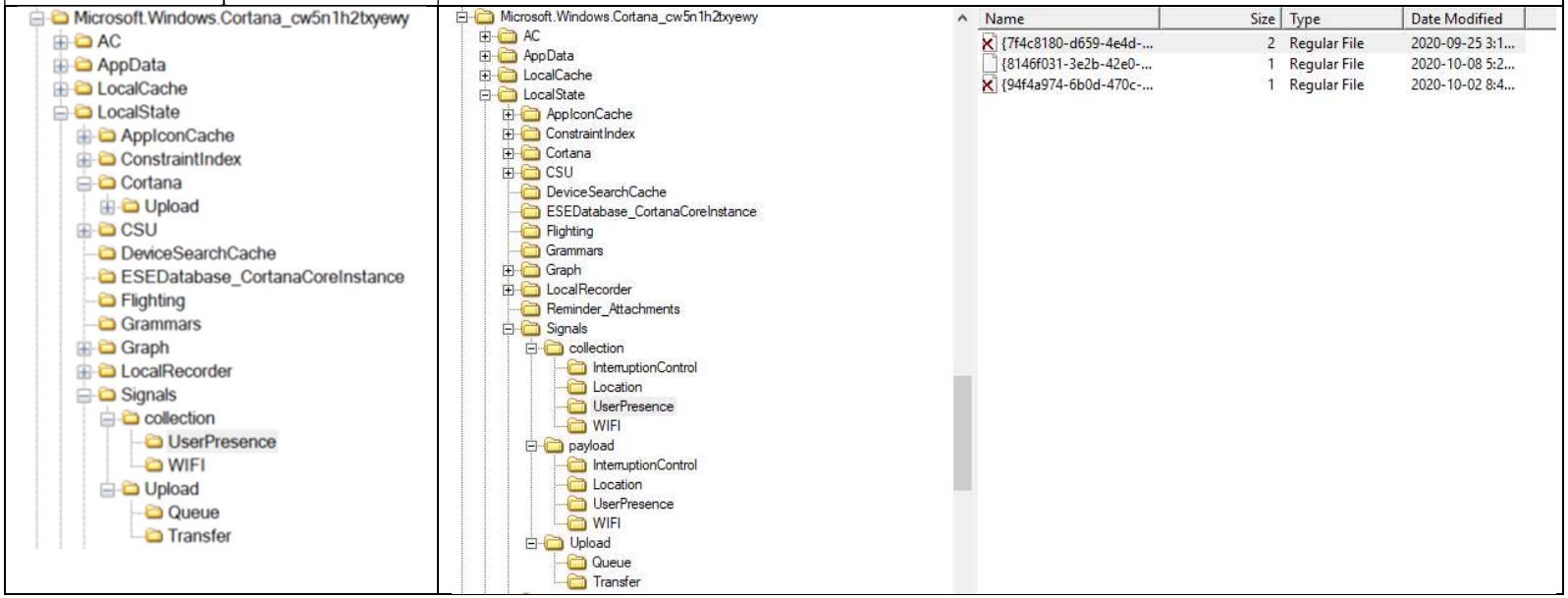



Figure 7-11 – Aperçus d'écrans des sous-répertoires de \Users\...\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy

Enfin, cet examen du 8 octobre 2020 a aussi révélé que la structure retrouvée dans l'image forensique du 23 septembre 2020 s'est complexifiée. Voir la figure 7-11 ci-dessous. Dans le sous-répertoire …\collection\Location\ se trouve le fichier sans extension {5b620631-eeab-4c7c-9784-9a179d5c4639} contenant des coordonnées topographiques horodatées à la demi-heure pour ce 8 octobre 2020 : 45°29'52.44" Nord sur 73°48'34.92" Ouest avec une précision dite de 985 et une « Position Source » de 3.

Après vérification à l'aide de Google Earth Street View, cette coordonnée correspond à la cour arrière du concessionnaire Spinelli Toyota de Pointe-Claire sur le boulevard des Sources. Or, ce lieu se situe à plus de 28 km du lieu où l'ordinateur GMO se situait en tout temps à cette date et il a été impossible de faire le lien entre ces deux endroits par le biais de la précision indiquée (car ce n'est ni 985 mètres, ni 9,85 km, ni 0,985 degrés d'écart).

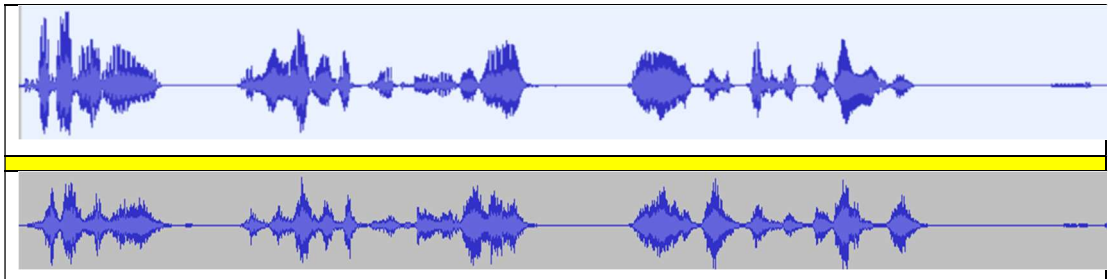
Dans \Users\...\LocalRecorder\Speech on retrouve les fichiers audio des huit (8) dernières requêtes réussies soumises à Cortana (lorsque Cortana doit sauvegarder le neuvième fichier, il recommence à « SpeechAudioFile_0.wav »). Pourquoi préciser « réussies »? Immédiatement avant les requêtes 123, la fonctionnalité « My Voice Only » de Cortana a été configurée. L'expérimentateur a lu les phrases demandées qui ont fait en sorte que Cortana ne réagisse qu'au son de sa voix. La requête 123a a ensuite été soumise par l'expérimentateur et Cortana a répondu à la question (How many petals a daisy has? ». Trois autres personnes ont ensuite prononcé les mots-gâchette « Hey Cortana », sans succès. Aucune preuve de ces essais n'a été relevée. Le seul fichier audio relatif à cette requête est celui de l'expérimentateur.

### 3. <u>Les sons</u>

L'audition des fichiers « SpeechAudioFile_#.wav » a abouti au constat que leur contenu avait été créé par Cortana alors que Cortana s'était déclenché sans l'intervention de l'expérimentateur à au moins huit (8) reprises. Le contenu de six de ces huit enregistrements a été reconnu comme provenant d'un cours en ligne donné par l'utilisateur. Comme ce cours est enregistré, le contenu du cours a été récupéré et a permis une comparaison entre l'audio du cours et l'enregistrement « subreptice ». Ce dernier a été fait par GMO à l'aide du micro intégré, micro de piètre qualité est situé à 2.5 m. et alors que l'expérimentateur faisait dos à GMO. La version enregistrée du cours est faite à l'aide d'un micro de qualité et d'un amplificateur de qualité situé à 5 cm de la bouche du

locuteur. Il est donc difficile de comparer les deux audios parce que la version subreptice présente une faible amplitude de son et un bruit de fond conséquent. Le son a donc dû être amplifié à l'aide du logiciel Audacity (de 20.636 dB à 5.0 dB) et deux bruits de fond (bourdonnement de GMO et un sifflement causé par l'enregistrement à distance) ont dû être supprimés, ce qui rend la comparaison plus facile mais encore boiteuse. À la figure 7-12, l'illustration du haut est un aperçu d'écran de la courbe de son dB en fonction du temps et l'illustration du bas de même mais pour le fichier subreptice.

On voit que les deux tracés ont un air de famille sans être absolument identiques.



Forme des ondes sonores Audacity de l'enregistrement du cours (en haut) et de l'enregistrement subreptice traité

Figure 7-12 – Comparaison de l'allure des courbes du son dB en fonction du temps pour l'enregistrement du cours (en haut) et pour l'enregistrement subreptice

Les spectres et les cepstres des deux bandes ont été comparés. Pour l'illustration sur les spectres (figure 7-13), le diagramme de droite (enregistrement subreptice) a été déplacé vers le bas pour enligner les échelles de gauche (enregistrement du cours) et de droite (enregistrement subreptice) au niveau du -32 dB. Pour l'illustration sur les cepstres (figure 7-14), la ligne rouge attire l'attention sur le fait que la valeur zéro (0) sur l'échelle de gauche n'est pas au même endroit que sur l'échelle de droite. Ceci est dû à la présence de valeurs plus étendues dans l'enregistrement extrait du cours par rapport aux valeurs relevées dans l'enregistrement subreptice.

Les valeurs des spectres de l'enregistrement du cours (haute-fidélité) et de l'enregistrement subreptice (basse qualité) sont très peu semblables. Par contre, les deux cepstres ont un air de ressemblance si on ne considère pas le premier millième de seconde. En effet, pour les événements après 0,001 seconde, les valeurs sont en hausse jusqu'à dépasser 0,00 mel pour ensuite chuter vers

0,0025 seconde et remonter pour stagner autour de 0,00 mel à partir 0,0075. On ne peut pas dire qu'ils sont identiques, ni même similaires, mais les deux cepstres ont un « air de famille ».

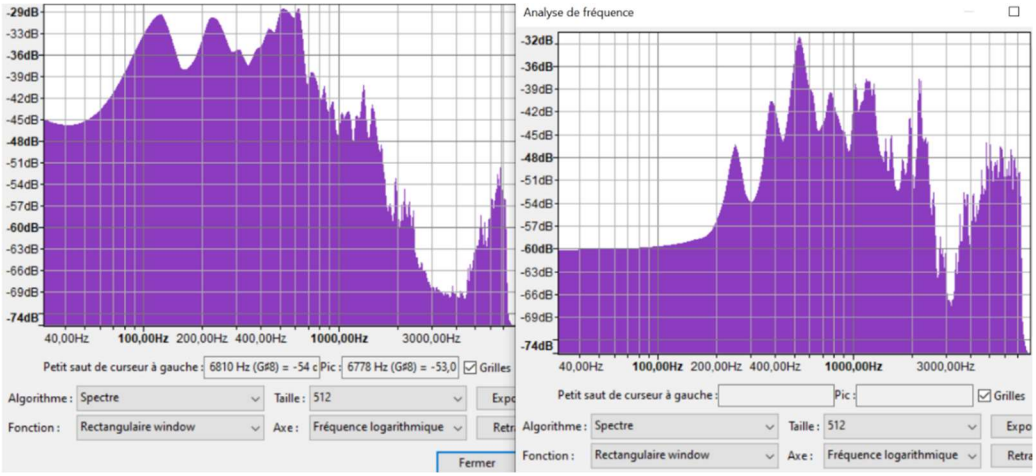



Figure 7-13 – Comparaison des spectres Audacity de l'enregistrement du cours (en haut) et de l'enregistrement subreptice traité

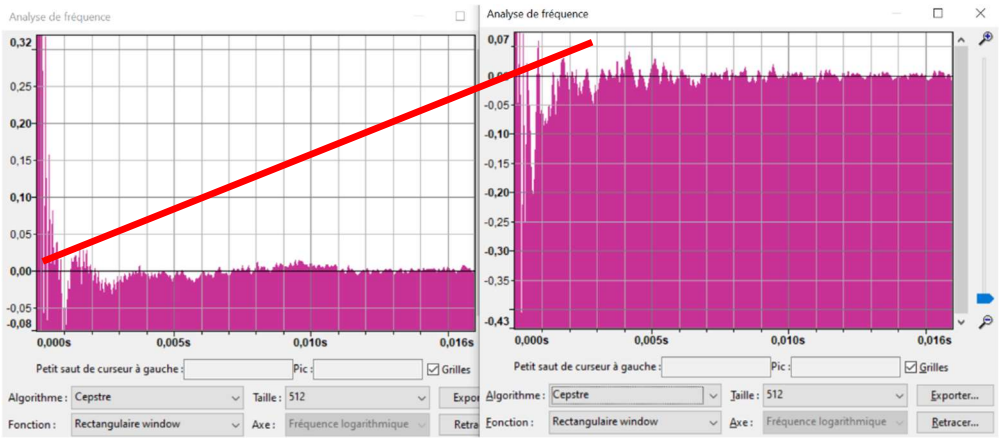



Figure 7-14 – Comparaison des cepstres Audacity de l'enregistrement du cours (en haut) et de l'enregistrement subreptice traité

Afin de déterminer ce qui a pu déclencher l'enregistrement subreptice lors du cours, l'analyse précédente a été refaite sur les 7,203 secondes précédent la portion enregistrée subrepticement. Cette portion de 7,203 secondes a été extraite de la bande audio du cours et convertie du format

flv au format mp3, sans autre traitement. La figure 7-15 montre la forme de l'onde sonore amplitude en fonction du temps pour les 7,203 secondes précédant le début de l'enregistrement subreptice (partie supérieure) et la forme de l'onde des mots-gâchettes « Hey Cortana » récupéré du laptop GMO. L'énoncé « Hey Cortana! » ne dure que 0,8 seconde.

La partie supérieure de ce graphique présente trois groupes de sons. Ce sont les trois moments où les paroles suivantes sont prononcées : « Ramassez mes 25 articles » (groupe d'onde de gauche », « pi euh ça va vous donner d'la lecture euh » (groupe d'onde du milieu) et « pi vous allez comprend' » (groupe d'onde de droite). Pour fins de comparaison, la partie inférieure de l'illustration présente la forme d'onde de l'énonciation de « Hey Cortana! ». Comme on peut le voir, rien ne ressemble, dans les mots prononcés en français québécois, au mot-gâchette prononcé avec accent anglais « Hey Cortana! » devant être prononcés pour utiliser Cortana sur GMO.

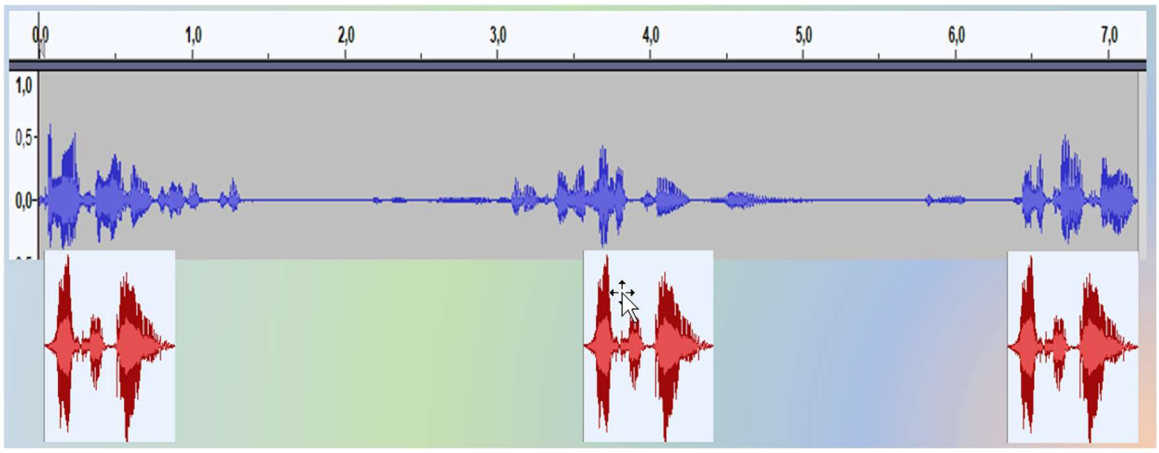



Figure 7-15 – Forme de l'onde sonore 7 secondes précédant l'enregistrement subreptice (en haut) et pour les mots-gâchette « Hey Cortana! » (en bas)

On doit aussi noter que les mots-gâchette « Hey Cortana! » sont énoncés en 0,803 seconde. L'énoncé du groupe du milieu (« pi euh ça va vous donner d'la lecture euh ») se termine 3 secondes avant que l'enregistrement subreptice ne commence. Il est donc fort peu probable que ce soit cet événement qui ait participé au déclenchement de l'enregistrement. L'énoncé de droite est, à cet égard, plus probable. La figure 7-16 présente la comparaison des spectres de « pi vous allez comprend' » et « Hey Cortana! ». On voit que pour 40 Hz, la valeur est d'environ -40 dB pour les deux, avec deux monticules situés à environ 125 et 225 Hz pour les deux énoncés. On constate

aussi qu'aux environs de 2 000 Hz, les valeurs sont aux environs de -42 à -45 dB pour les deux énoncés. Encore ici, les spectres ne sont pas identiques ni similaires mais présentent un certain « air de famille ».

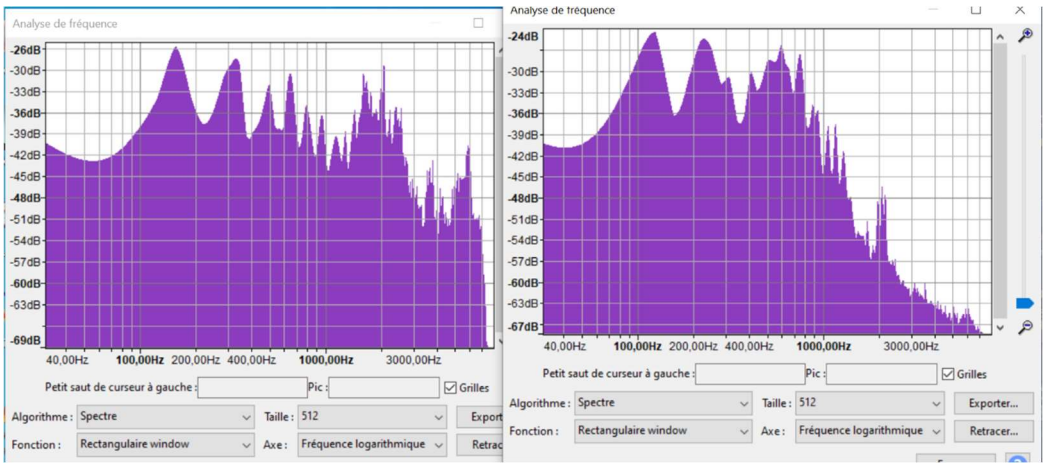



Figure 7-16 – Comparaison des spectres Audacity des deux énoncés « Hey Cortana! » (à gauche) et « pi vous allez comprend' » (à droite)

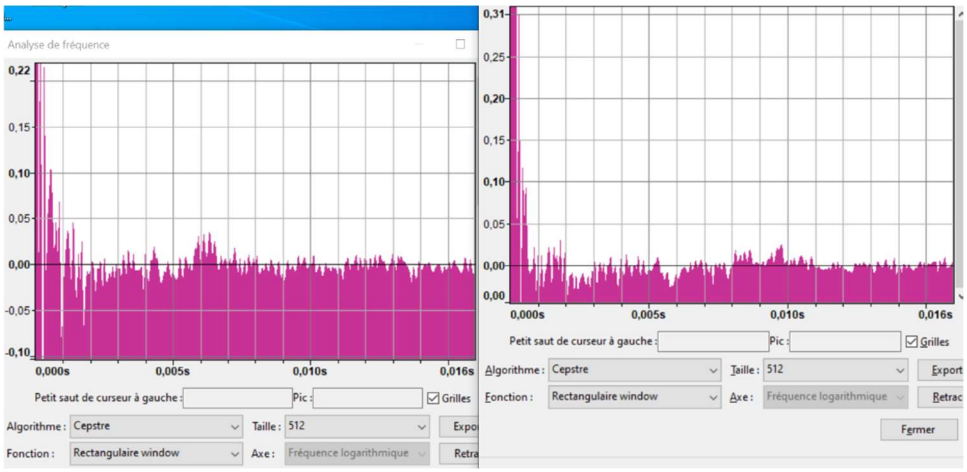



Figure 7-17 – Comparaison des cepstres Audacity des deux énoncés « Hey Cortana! » (à gauche) et « pi vous allez comprend' » (à droite)

La figure 7-17 présente la comparaison des cepstres des deux énoncés « Hey Cortana! » et « pi vous allez comprend' ». Ici aussi, si on ne considère que les valeurs après 0,001 seconde, la forme

générale du cepstre se situe autour de 0 mel. Encore ici, les cepstres ne sont pas identiques ni similaires mais présentent un certain « air de famille ».

Dans une expérimentation extensive du sujet, Dubois et al [2020] rapportent que tous les assistants numériques personnels présentent le défaut de se déclencher sans que le mot-gâchette particulier n'ait été prononcé. Afin de comprendre pourquoi, il a fait écouter à ces assistants 134 heures de programmes de télévision (en anglais britannique et en anglais américain seulement) contenant une multitude de locuteurs et 1 057 000 mots. Pendant ces heures, Cortana s'est activé erronément à 54 reprises pour une durée entre 0 et 3 secondes. Dans leur analyse de la source de l'activation erronée, Dubois et al mentionnent des mots contenant le son « K » suivi de peu par les sons « R » ou « T ».

> Ce qu'il y a de particulier dans le cas mentionné ci-haut est que 6 des 8 activations erronées sont survenues pendant le cours donné à distance, en français québécois, et que le Cortana erronément activé est configuré sur un système d'exploitation Windows anglais et que toutes nos tentatives délibérées de communiquer en français avec ce Cortana (requêtes 113b, 113c, 113d, 114 et, accessoirement les mots francophones des requêtes 116, 117, 118, 119a et 119c) se sont soldées par un échec. On ne peut qu'en conclure que Cortana anglais ne comprend pas le français. Mais il se déclenche erronément sur des mots français québécois. **Ceci constitue donc une faille majeure quant à la préservation de la vie privée de l'utilisateur (involontaire) de Cortana.**

## 4. Historique des données chez Microsoft

Le 20 septembre 2019, les procédures énoncées par Victor [2019] ont été appliquées afin d'obtenir les données Windows Cortana de GMO. Ces données se résumaient à une dizaine de lignes tout au plus.

Le 21 septembre 2019, deux jours avant la tenue de la ronde 2, le disque dur de GMO a été aseptisé en écrivant l'hexadécimale 0x00 pour chaque octet disponible sur l'ensemble du disque dur (et non

seulement pour la partition principale). Ceci a eu pour effet de faire disparaître toute donnée présente sur ce disque dur.

Le 22 septembre 2019, à la veille de la ronde 2, l'image forensique GMO0 (créée en octobre 2018) a été restaurée sur le disque dur aseptisé au préalable. Ici, on écrasait les 0x00 avec des données datant d'une année auparavant.

> Malgré toutes ces précautions (vidange des données chez Microsoft, aseptisation du disque dur et restauration de l'image d'octobre 2018), des données du provenant de la ronde 1 du 9 septembre 2019 sont apparues lors des activités de configuration tenues le matin du 23 septembre 2019, avant le début de la ronde 2. Ces données consistaient en une liste des dernières 20 requêtes soumises à Cortana le 9 septembre 2019. Ces données ne pouvant pas provenir du disque dur local, force est d'en déduire qu'elles provenaient d'un espace de stockage en ligne. **Cette conclusion est majeure** car elle contredire ce qui est déclaré par Microsoft au niveau de la vidange des données de Cortana, données se rapportant à la vie privée de l'utilisateur.

## 5. Analyse des transmissions réseau

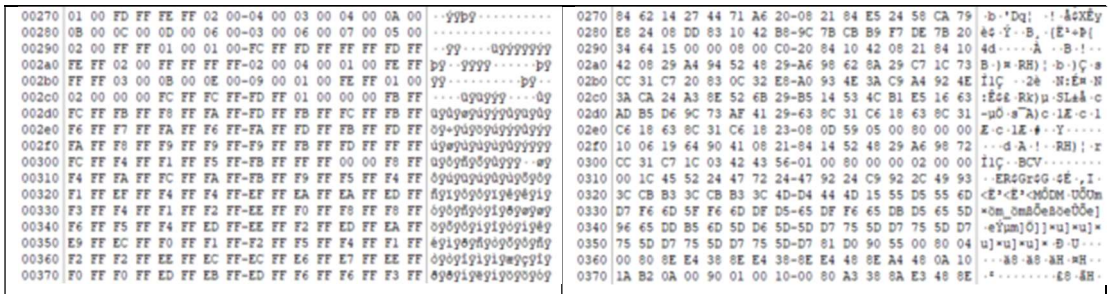
Cette partie de l'analyse des prélèvements a connu plus d'un rebondissement.

Nous avons récupéré la charge utile (« payload ») des paquets transmis vers l'adresse IP 204.79.197.200 (officiant à la reconnaissance vocale pour Cortana) dans le but de les transformer en fichier audio, dans l'espoir de pouvoir les utiliser sur un site comme Lyrebird.ai afin de démontrer un risque à la sécurité.

Sur le site Lyrebird, un utilisateur peut soumettre des échantillons d'une voix, ce qui permet ensuite à un algorithme texte-voix de prendre un texte soumis par cet utilisateur et d'en faire une version orale adoptant la voix soumise en échantillon.

En théorie faisable, ceci permettrait à une personne mal intentionnée de faire dire à la personne imitée des propos qui ne sont pas les siens ou d'utiliser la voix échantillonnée pour passer des requêtes qui ne sont pas de son fait.

Bien que Cortana enregistre les requêtes en format .wav, ce n'est pas le format utilisé pour transmettre le flux audio. Ce qu'il faut comprendre, c'est que les données sauvegardées dans un fichier d'un format quelconque ont un genre « d'air de famille » si on les compare aux données sauvegardées dans un fichier de même extension. Par exemple, dans l'illustration à la figure 6-22 ci-dessous, on compare les données d'un fichier .wav (à gauche) avec celles d'un fichier .ogg, les deux fichiers contenant exactement les mêmes paroles.



| Fichier .wav | Fichier .ogg |
|---|---|

Figure 7-18 – Comparaison de la structure des données de deux types de fichier audio

Pour ce qui est des fichiers, on peut bien sûr distinguer les .wav des .ogg de par leurs signatures, .wav ayant une signature RIFF8ʃ (0x5249464638A7) tandis que les fichiers .ogg on une signature OggS (0x4F676753). Mais lorsque le flux audio est transmis vers les serveurs de Cortana, la signature n'est pas transmise. Microsoft a dû considérer que cette donnée était inutile dans la mesure où le Cortana local et le Cortana distant ont dû convenir d'un format de transmission du flux voix.

Un doute subsiste tout de même sur le format de flux par voix utilisé par Cortana à cause du manque de référence quant au flux ni relativement au codec utilisé par Cortana. Plusieurs tests d'adaptation de la charge utile de la requête numéro 3 « What is the foundation date of Canada? » ont été réalisés, notamment avec les codecs Vorbis et Opus, mais sans succès. Qu'en conclure sinon que Microsoft utilise un codec propre à Cortana?

Compte tenu de la moindre importance de cet aspect, cette partie de la recherche a été délaissée pour revenir au sujet principal. Toutefois, si cette recherche devait se continuer, il serait pertinent d'examiner les flux voix descendants, c'est-à-dire ceux que Cortana fait jouer sur GMO en réponse à une requête. Par exemple la requête 13a « Tell me a joke for kids » ou 21a « What is the forecast for today? » afin de déterminer le codec utilisé. Ou utiliser un algorithme d'apprentissage automatique comparant les flux audio sortants et les charges utiles des paquets contenant ce flux audio.

Cet aspect de la recherche a abouti au constat qu'on peut camoufler des données texte dans un fichier .ogg sans impact sur la qualité audio du fichier. Ceci n'est possible que si on octroie à ces données le bon numéro de flux, le bon numéro de page et le bon CRC32. Ceci pourrait permettre à des malandrins de se communiquer des informations leur permettant de commettre leur forfait (une attaque terroriste par exemple) sous le couvert d'un fichier audio de nature innocente (une chanson de Gilles Vigneault par exemple).

# CONCLUSION

## 1. Sommaire des découvertes

Une multitude de processus sont utilisés par Windows en tout temps et de façon concomitante. Certains de ces processus (par exemple SvcHost) sont partagés entre différentes applications, dont Cortana. Toutefois, Cortana utilise des processus qui lui sont propres (SearchUI). Lorsque le mot-gâchette « Hey Cortana » est prononcé en vue de soumettre une requête à Cortana, il s'écoule un délai d'environ 1.7 secondes avant qu'une pléthore de processus et de fils ne se déclenchent pour enregistrer la version orale de la requête et l'expédier à un serveur qui l'interprète et la retourne. La version orale des huit dernières requêtes est enregistrée par Cortana qui écrase la requête 0 avec la requête 9 lorsque celle-ci est soumise.

SvcHost, Explorer et SearchUI sont les trois services les plus sollicités entre la 1.7ième seconde et la 6ième seconde. Les requêtes utilisées durant entre 3 et 7 secondes, on peut penser que tout le long des requêtes, ces services s'agitent.

Au-delà de 325 000 lectures ou écritures sont faites dans les 30 premières secondes d'une requête soumise à Cortana. Les premiers processus démarrés suite à la soumission d'une requête sont SearchUI, SpeechOneCore, StateRepository, ActivitiesCache, CloudExperience, SpeechRuntime. Des écritures surviennent très tôt (à partir de la 2ième seconde) après le début de la requête, dans ActivitiesCache, la base de registre, des fichiers .clb et .tbres. Au final, certains artéfacts font l'objet d'écritures détaillées de tout ce qui a été demandé à Cortana. C'est le cas, notamment, du fichier AutomaticDestinations-ms et du répertoire \Users\[Utilisateur]\AppData\Roaming\Microsoft\Windows\Recent où le mot-à-mot interprété des requêtes est inscrit pour toute la période d'utilisation de l'ordinateur depuis sa première utilisation.

Beaucoup d'artéfacts contiennent des données se rapportant aux requêtes soumises à Cortana : Les fichiers de prélecture .pf, les sous-répertoires de InetCache, les grappes non-allouées, les balances de fichiers, les fichiers orphelins, les fichiers participant au système de fichiers (journaux, $MFT), la base de registre, certains fichiers désencastrés par les logiciels de forensique, des fichiers .json, des fichiers .dat et Stddtype.gdl.

Cortana se déclenche parfois sans que le mot-gâchette n'ait été prononcé (enregistrement subreptice). Après analyse des MFCC des enregistrements audios précédent les enregistrements subreptices, et après avoir lu l'article de Carlini et Wagner [2018], de telles occurrences semblent compréhensibles.

## 2. Perspectives

Pendant la phase expérimentale, les paquets Internet contenant la version vocale des requêtes ont été recueillis. Pendant la phase d'analyse, la charge utile d'un ces paquets a été extraite. Il n'a pas été possible de reconstituer un fichier audio à l'aide de cette charge utile. Une reprise de cet aspect de la recherche est envisageable dans le futur.

La recherche a commencé avec, en tête, l'idée de faire plusieurs assistants numériques personnels. Mais la quantité de données à traiter et la quantité d'informations que nous en avons tiré nous a surpris et nous a contraint à revoir les limites de la présente recherche. Les autres assistants numériques personnels restent donc à étudier. Pour ceux installés sur un ordinateur, la voie à suivre pourra être calquée sur notre approche. Pour les haut-parleurs intelligents, l'accès aux dispositifs de stockage de ces appareils posera problème et donnera sans doute lieu à des prouesses techniques.

Lors d'études futures, l'utilisation d'une application enregistrant dynamiquement ce qui se passe dans l'appareil (comme ProcMon) devra être préféré à l'utilisation d'une application analysant ceci de façon statique (comme Volatility).

## 3. Conclusion

Quatre années d'études. Trois mois d'expérimentation. Douze mois d'analyse. Ça peut faire peur, mais c'est tellement passionnant et gratifiant!

# Glossaire

Application portable : Application qui fonctionne sans qu'on ait à l'installer.

Artéfact : Tout objet tangible (ordinateur, disque dur, clé USB, etc.) ou intangible (programme, fichier, donnée, etc.);

Décalage : C'est le nombre d'octets qui séparent le premier octet d'un point de référence fixe de l'endroit qu'on veut désigner.

Empreinte numérique : Plus connue sous le nom de « hash ». L'empreinte numérique est un nombre qui constitue le résultat d'un calcul fait selon une formule mathématique convenue. L'intrant est le contenu dont on calcule l'empreinte numérique et c'est généralement le contenu du fichier. Si on calcule l'empreinte numérique d'un fichier et qu'on modifie un seul bit choisi au hasard (ou pas, en fait!) et qu'on recalcule l'empreinte numérique, on obtiendra un résultat totalement différent du premier, sans aucune ressemblance entre eux que le fait qu'ils sont constitués de caractères. Les métadonnées localisées hors du fichiers peuvent être modifiées sans affecter l'empreinte numérique.

Ordinateur : L'article 342.1(2) du Code criminel du Canada (*Voir Ccr [2020]*) définit ainsi un ordinateur : « *Dispositif[s qui] … : a) contiennent des programmes d'ordinateur ou d'autres données informatiques; b) conformément à des programmes d'ordinateur : (i) exécutent des fonctions logiques et de commande, (ii) peuvent exécuter toute autre fonction.* »

Vidage de mémoire : Selon le GDT-OQLF, c'est une « *Opération qui consiste à copier sur un autre support ou à imprimer la liste des données rangées dans la totalité ou dans une partie d'une mémoire sans modifier le contenu de la mémoire.* »

# ANNEXE A – LISTE DES FICHIERS AFFECTÉS PAR LES OPÉRATIONS D'ÉCRITURE LORS DE LA REQUÊTE #3

| Nom et chemin | Occ1 | Qté | Description (Les passages en italique sont des citations) | Référence |
|---|---|---|---|---|
| C:\Users\inter\AppData\Local\ConnectedDevicesPlatform\bb717611e150420e\ActivitiesCache.db-wal | 2 | 52 | *User activity information in Windows 10 Timeline is saved to the file ActivitiesCache.db ActivitiesCache.dbActivitiesCache.db is an SQLite database (version 3). Like any SQLite database, it has two auxiliary files: ActivitiesCache.db-shm and ActivitiesCache.db-wal.* | Mikhailov [2019] |
| C:\Windows\appcompat\Programs\Amcache.hve | 5,491 | 3 | *The Amcache.hve is a registry hive file that is created by Microsoft Windows to store the information related to execution of programs* | Singh & Singh [2016] |
| Fichiers de prélecture (première occurrence est backgroundtask) | 5,95 | 8 | Un fichier pf prefetch prélecture est un fichier créé lors du premier démarrage d'un exécutable dans lors d'une session. Pas tous les exe qui ont un pf mais tous les pf sont liés à un exe | Sylvain Desharnais |
| C:\Users\inter\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\#!001\Microsoft\CryptnetUrlCache\MetaData\ (77EC63 BDA74BD0D0E0426DC8F8008506, 57C8EDB95DF3F0AD4EE2DC2B8CFD4157, FB0D848F74F70BB2EAA93746D24 D9749) | 16,6 | 3 | *CryptnetUrlCache is a folder associated with the storage of information or files that are automatically acquired (often without your knowledge) from the Internet. Basically, while navigating through various sites on the Internet, your computer automatically scraps off certain information from these sites, usually to improve your browsing experience. These sets of information are in varying degrees and types and are thus stored in various folders. One of such folders is CryptnetUrlCache.* | Jenic [2019] |
| C:\ProgramData\Microsoft\Network\Downloader\: qmgr.db, edb.chk, edb.log, qmgr.jfm | 22,31 | 21 | *Edb.log is a transaction log. ESE is a form of transactional database which means any changes made to objects in Active Directory are first saved to a transaction log to provide fault tolerance. During non-peak times in CPU activity, the database engine commits the transactions into the main Ntds.dit database. There is one checkpoint file named as Edb.chk, which is used by the transaction logging system to mark the point at which updates are transferred from the log files to Ntds.dit. As transactions are committed, the checkpoint moves forward in the Edb.chk file.* | Prabhu [2015] |
| C:\Windows\System32\winevt\Logs\Microsoft-Windows-SettingSync%4Debug.evtx | 16,11 | 1 | *Windows Event Log captures the details of both system and application events.* | NXlog [2020] |
| | | **88** | | |

Tableau A-1 – Fichiers cibles d'une écriture de données par svchost.exe suite à la requête #103 auprès de Cortana en mars 2020

| Nom et chemin | Occ1 | Qté | Description (Les passages en italique sont des citations) | Référence |
|---|---|---|---|---|
| C:\ProgramData\Microsoft\Windows\AppRepository\Packages\Microsoft.AAD.BrokerPlugin_1000.18362.449.0_neutral_neutral_cw5n1h2txyewy\ActivationStore.dat (et dat.LOG1 et 2) | 3,55 | 10 | Pas de certitude nulle part. Mais logiquement, ça contient les clés de fonctionnement de tout logiciel installé auquel on a fourni une clé d'activation particulière (par exemple WinHex) ou groupée (par exemple MS Office 365) ou par défaut (par exemple Cortana) | Sylvain Desharnais |
| C:\Windows\appcompat\Programs\Amcache.hve | 5,49 | 3 | *The Amcache.hve is a registry hive file that is created by Microsoft Windows to store the information related to execution of programs* | Singh & Singh [2016] |
| C:\ProgramData\Microsoft\Windows\AppRepository\Packages\Microsoft.Windows.Cortana_1.13.0.18362_neutral_neutral_cw5n1h2txyewy\ActivationStore.dat | 8,06 | 1 | Pas de certitude nulle part. Mais logiquement, ça contient les clés de fonctionnement de tout logiciel installé auquel on a fourni une clé d'activation particulière (par exemple WinHex) ou groupée (par exemple MS Office 365) ou par défaut (par exemple Cortana). Ici, c'est pour Cortana. | Sylvain Desharnais |
| C:\ProgramData\Microsoft\Windows\AppRepository\Packages\Microsoft.MicrosoftEdge_44.18362.449.0_neutral__8wekyb3d8bbwe\ActivationStore.dat | 10,62 | 2 | Pas de certitude nulle part. Mais logiquement, ça contient les clés de fonctionnement de tout logiciel installé auquel on a fourni une clé d'activation particulière (par exemple WinHex) ou groupée (par exemple MS Office 365) ou par défaut (par exemple Cortana). Ici, c'est pour le navigateur Edge. | Sylvain Desharnais |
| C:\ProgramData\Microsoft\Windows\AppRepository\Packages\Microsoft.SkypeApp_14.56.102.0_x64__kzf8qxf38zg5c\ActivationStore.dat | 15,31 | 2 | Pas de certitude nulle part. Mais logiquement, ça contient les clés de fonctionnement de tout logiciel installé auquel on a fourni une clé d'activation particulière (par exemple WinHex) ou groupée (par exemple MS Office 365) ou par défaut (par exemple Cortana). Ici, c'est pour Skype. | Sylvain Desharnais |
| C:\Users\inter\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\#!001\Microsoft\CryptnetUrlCache\MetaData\ (77EC63BDA74BD0D0 E0426DC8F8008506, 57C8EDB95DF3F0AD4EE2DC2B8CFD4157,FB0D848F74 F70BB2EAA93746D24D9749) | 16,60 | 6 | *CryptnetUrlCache is a folder associated with the storage of information or files that are automatically acquired (often without your knowledge) from the Internet. Basically, while navigating through various sites on the Internet, your computer automatically scraps off certain information from these sites, usually to improve your browsing experience. These sets of information are in varying degrees and types and are thus stored in various folders. One of such folders is CryptnetUrlCache.* | Jenic [2019] |
| | | **24** | | |

Tableau A-2 – Fichiers cibles d'une écriture de métadonnées par svchost.exe suite à la requête #103 auprès de Cortana en mars 2020

# MÉDIAGRAPHIE

## Introduction


Allen Sabernick III, B. (2016). Development of an Autopsy Forensics Module for Cortana Artifacts Analysis. International Journal of Computer Science and Information Security, 14(7), 111-121.

Anonyme. (2020, 1er avril). La Romance de Ténébreuse. Dans Wikipédia. Récupéré le 25 mai 2020 de https://fr.wikipedia.org/w/index.php?title=La_Romance_de_T%C3%A9n%C3%A9breuse&oldid=169058856

Ash, M. (2017, 8 mai). Harman Kardon Invoke featuring Cortana: Captivating sound meets personal digital assistant [Corporate website]. Dans Windows Experience Blog. Récupéré de https://blogs.windows.com/windowsexperience/2017/05/08/harman-kardon-invoke-featuring-cortana-captivating-sound-meets-personal-digital-assistant/

Commission nationale informatique et libertés. (s. d.). À votre écoute - Livre blanc no 1 - Exploration des enjeux éthiques, techniques et juridiques des assistants vocaux. Récupéré de https://www.cnil.fr/sites/default/files/atoms/files/cnil_livre-blanc-assistants-vocaux.pdf

Cour d'appel fédérale. (2011, 21 avril). Canada c. Buckingham. RCF A-224-10; A-225-10 Cour d'appel fédérale 86. Récupéré de http://canlii.ca/t/fm929

Cour suprême du Canada. (1988, 8 décembre). R. c. Dyment. RCS 19786 Cour suprême du Canada 417. Récupéré de http://canlii.ca/t/1ftc5

Cour suprême du Canada. (1993, 11 mars). R. c. Hundal. RCS 22358 Cour suprême du Canada 867. Récupéré de https://scc-csc.lexum.com/scc-csc/scc-csc/fr/item/977/index.do

Cour suprême du Canada. (2010, 24 novembre). R. c. Gomboc. RCS 33332 Cour suprême du Canada 211. Récupéré de http://canlii.ca/t/2dhll

Cour suprême du Canada. (2010, 19 mars). R. c. Morelli. RCS 32741 Cour suprême du Canada 253. Récupéré de http://canlii.ca/t/28mrf

Cour suprême du Canada. (2019, 14 février). R. c. Jarvis 37833. Récupéré de http://canlii.ca/t/hxj08

Dangerfield, K. (2018, 14 novembre). Alexa, who killed these women? U.S. judge orders Amazon to provide Echo's audio files. Dans Global News. Récupéré de https://globalnews.ca/news/4661963/amazon-alexa-homicide-new-hampshire/

Desharnais, S. (2015). Chronique - Balises de vie privée. La Référence. doi: EYB2015REP1780

Dubois, D. J., Kolcun, R., Mandalari, A. M., Paracha, M. T., Choffnes, D. et Haddadi, H. (2020). When Speakers Are All Ears: Characterizing Misactivations of IoT Smart Speakers. Proceedings on Privacy Enhancing Technologies, 2020(4), 255-276. doi: 10.2478/popets-2020-0072

Gaffiot, F. (1934). Dictionnaire Latin-Français Gaffiot [Dictionnaire]. Récupéré de http://micmap.org/dicfro/search/gaffiot/forensis

Gauthier, B. (dir.). (2009). Recherche sociale: de la problématique à la collecte des données (5th ed). Québec : Presses de l'Université du Québec.

Gayle, L. (2019, 18 décembre). Children secretly order $400 worth of toys. Dans Mail Online. Récupéré de https://www.dailymail.co.uk/femail/article-7805019/Mother-receives-400-worth-toys-children-order-online-without-knowledge.html

Gervais, R. et Robinson, M. (2009, 2 octobre). L'invention du mensonge (The Invention of Lying) [Comedy, Fantasy, Romance]. Warner Bros., Radar Pictures, Media Rights Capital (MRC). Récupéré de https://www.imdb.com/title/tt1058017/?ref_=fn_al_tt_1

Johson, M. (2019, 10 juillet). Essai du Google Nest Hub (dans 4 pièces). Montréal Métro (Montréal). Récupéré de https://journalmetro.com/techno/2346723/le-google-nest-hub/

Justice, M. de la. (2015, 30 juillet). Lois codifiées Règlements codifiés. Dans Lois codifiées du Canada. Récupéré de https://laws-lois.justice.gc.ca/fra/Const/page-15.html

Lynskey, D. (2019, 9 octobre). « Alexa, are you invading my privacy? » – the dark side of our voice assistants. The Guardian, section Technology. Récupéré de https://www.theguardian.com/technology/2019/oct/09/alexa-are-you-invading-my-privacy-the-dark-side-of-our-voice-assistants

Mu, H. (2018, 19 décembre). 10+ Top Open source Voice Assistants Projects for developers (Linux, Raspberry Pi, Windows & Mac OS X) [Informations informatiques]. Dans Medevel. Récupéré de https://medevel.com/10-open-source-voice-assistants/

Nadeau, A.-R. (2000). Vie privée et droits fondamentaux : étude de la protection de la vie privée en droit constitutionnel canadien et américain et en droit international. [Scarborough, Ont.] : Carswell.

Nissenbaum, H. F. (2010). Privacy in context: technology, policy, and the integrity of social life. Stanford, California : Stanford Law Books, an imprit of Standford University Press.

Office québécois de la langue française. (2020, 12 février). criminalistique [Dictionnaire]. Dans Grand dictionnaire terminologique. Récupéré de http://gdt.oqlf.gouv.qc.ca/ficheOqlf.aspx?Id_Fiche=8868924

Ogden, G. (2018, 25 mai). An Amazon Echo recorded someone's conversation and sent it to their work colleague. Dans Shortlist. Récupéré de https://www.shortlist.com/news/amazon-echo-listening-alexa

Oracle. (2020, février). Oracle SaaS [Corporatif]. Dans Data Science Cloud. Récupéré de https://www.oracle.com/cloud/data-science-cloud/

Pineau, A. (2014, 19 septembre). Droit à la vie privée: la jurisprudence de la Cour suprême. Ligue des droits et libertés. Récupéré de https://liguedesdroits.ca/droit-a-la-vie-privee-la-jurisprudence-de-la-cour-supreme/

Québbec (Province), Baudouin, J.-L., Renaud, Y., Québec (Province) et Québec (Province). (2015). Code civil du Québec annoté. (s. l.) : Wilson & Lafleur. 2 vol.

Rodriguez, J. (2019, 2 novembre). Florida police get data from Amazon Alexa which may have recorded murder. Dans CTVNews. Récupéré de https://www.ctvnews.ca/sci-tech/florida-police-get-data-from-amazon-alexa-which-may-have-recorded-murder-1.4667664

Schweppe, D. (s. d.). Mycroft – Open Source Voice Assistant. Dans Mycroft. Récupéré de https://mycroft.ai/

Solove, D. J. (2014, 20 janvier). 10 Reasons Why Privacy Matters [Formation]. Dans TeachPrivacy. Récupéré de https://teachprivacy.com/10-reasons-privacy-matters/

Waldman, A. E. (2018). Privacy as Trust: Information Privacy for an Information Age (1re éd.). Cambridge University Press. doi: 10.1017/9781316888667

Warren, S. D. et Brandeis, L. D. (1890). The Right to Privacy. Harvard Law Review, 4(5), 193-220.

Westin, A. F. (2015). Privacy and freedom. New York : IG Publishing. Récupéré de shorturl.at/gkAX6

(2019, 17 décembre). The Ethical Algorithm | Michael Kearns & Aaron Roth | Talks at Google [Youtube]. Récupéré de https://www.youtube.com/watch?v=tmC9JdKc3sA

## Chapitre 1

Bradley, M. Z. (1972). La Planète aux vents de folie ( A. Vincent, trad.). Paris : Presses pocket.

CEFRIO. (s. d.). NETendances 2019 - La mobilité et les nouvelles tendances en contexte de pandémie. Dans CEFRIO. Récupéré de https://cefrio.qc.ca/fr/nouvelles/communique-netendances2019-mobilite-nouvelles-tendances-contexte-pandemie/

Cour suprême du Canada. (2013, 7 novembre). R. c. Vu. RCS 34687 Cour suprême du Canada 657. Récupéré de http://canlii.ca/t/g1r8q

Desharnais, S. (2015a). Chronique - Balises de vie privée. La Référence. doi: EYB2015REP1780

Desharnais, S. (2015b). Chronique – Propriétés des données numériques (Partie 2). La Référence. doi: EYB2015REP1668

Government of Canada, S. C. (2019, 29 octobre). Le Quotidien — Enquête canadienne sur l'utilisation de l'Internet. Récupéré de https://www150.statcan.gc.ca/n1/daily-quotidien/191029/dq191029a-fra.htm

Manumation. (2008). Présentation des Alternate Data Stream (ADS) [Site de connaissances]. Dans Developpez.com. Récupéré de http://manumation.developpez.com/

Quoracreative. (s. d.). Voice Search Statistics And Trends For 2020. Dans quoracreative.com. Récupéré de https://quoracreative.com/article/voice-search-statistics-trends

Shead, S. (2018, 29 août). Comment les assistants vocaux changeront-ils le monde du travail? | Regus [Presse]. Work Canada. Récupéré de https://www.regus.ca/work-canada/fr-ca/will-voice-assistants-change-workplace/

Voicebot. (2019, 29 avril). Microsoft Releases Voice Assistant Usage Report, Finds Apple Siri And Google Assistant Tied at 36%, and 41% of Respondents Have Privacy Concerns. Dans Voicebot.ai. Récupéré de https://voicebot.ai/2019/04/28/microsoft-releases-voice-assistant-usage-report-finds-apple-siri-and-google-assistant-tied-at-36-and-41-of-respondents-have-privacy-concerns/

Wikipédia. (2020, 3 septembre). Edmond Locard. Dans Wikipédia. Récupéré le 12 novembre 2020 de https://fr.wikipedia.org/w/index.php?title=Edmond_Locard&oldid=174406226

(2014, 19 septembre). Droit à la vie privée: la jurisprudence de la Cour suprême. Ligue des droits et libertés. Récupéré de https://liguedesdroits.ca/droit-a-la-vie-privee-la-jurisprudence-de-la-cour-supreme/

## Chapitre 2

Access Data. (2020, 28 janvier). FTK Imager [Corporatif]. Dans AccessData - Product Download - FTK Imager. Récupéré de https://ad-pdf.s3.amazonaws.com/Imager/4_3_0/FTKImager_UG.pdf

Basis Technology. (2020, février). Autopsy [Corporate website]. Dans Autopsy Sleuth Kit. Récupéré de https://www.sleuthkit.org/autopsy/

Blackbag PR. (2018, 21 décembre). BlackBag Releases APFS Source Code To The Sleuth Kit® Framework [Corporate website]. Dans BlackBag. Récupéré de https://www.blackbagtech.com/press-releases/blackbag-releases-apfs-source-code-to-the-sleuth-kit-framework/

ExtendOffice. (2020). Kutools - Advanced Functions and Tools for Excel [Corporatif]. Dans Kutools - Combines More Than 300 Advanced Functions and Tools for Microsoft Excel. Récupéré de https://www.extendoffice.com/product/kutools-for-excel.html

FireEye. (2019a, 25 juillet). Finding Evil in Windows 10 Compressed Memory, Part One: Volatility and Rekall Tools [Corporate website]. Dans FireEye. Récupéré de https://www.fireeye.com/blog/threat-research/2019/07/finding-evil-in-windows-ten-compressed-memory-part-one.html

FireEye. (2019b, 26 septembre). Fourche FireEye de Volatility. Dans GitHub. Récupéré de https://github.com/fireeye/win10_volatility

FireEye. (2019c, 18 octobre). FireEye Free Security Software. Dans FireEye. Récupéré de https://www.fireeye.com/services/freeware.html

Gouvernement du Canada. (2020, 5 février). Code criminel du Canada. Code criminel Fédéral. , Code criminel. Récupéré de https://laws-lois.justice.gc.ca/PDF/C-46.pdf

Guidance Software. (2020, février). EnCase Forensic Imager. Dans EnCase Forensic Imager. Récupéré de /document/product-brief/encase-forensic-imager

Long Path Tool. (2007). Long Path Tool 5: advanced long path files manager [Corporate website]. Dans Cannot delete files? Cannot copy files? Long Path Tool Can. Récupéré de https://longpathtool.com/

mitmproxy. (2020, 19 janvier). mitmproxy - an interactive HTTPS proxy. Dans Développeur. Récupéré de https://mitmproxy.org/

Paragon. (2018, février). APFS for Windows. Paragon Software Group. Récupéré de https://www.paragon-software.com/fr/home/apfs-windows/

ProcessHacker. (2020). Overview - Process Hacker [Organization website]. Dans Overview - Process Hacker. Récupéré de https://processhacker.sourceforge.io/

Russinovich, M. (2019, 18 décembre). Sysinternals Suite [Corporate website]. Dans Microsoft Sysinternals. Récupéré de https://docs.microsoft.com/en-us/sysinternals/

Telerik. (2020, 20 janvier). Fiddler [Corporate website]. Dans Telerik.com. Récupéré de https://www.telerik.com/fiddler

The Volatility Foundation - Open Source Memory Forensics. (2020, février). Volatility [Corporate website]. Dans Volatilityfoundation. Récupéré de https://www.volatilityfoundation.org

Velocidex. (2019, 17 mai). WinPmem memory imager [Organisationnel]. Dans The Pmem Suite. Récupéré de https://winpmem.velocidex.com/

Wikipédia. (2019, 24 décembre). FireEye. Dans Wikipédia. Récupéré le 16 février 2020 de https://fr.wikipedia.org/w/index.php?title=FireEye&oldid=165675278

Wireshark Organisation. (s. d.). Wireshark · Go Deep. Récupéré de https://www.wireshark.org/

## Chapitre 3

Abdi, N., Ramokapane, K. M. et Such, J. M. (2019). More than Smart Speakers: Security and Privacy Perceptions of Smart HomePersonal Assistants. Communication présentée au Fifteenth Symposium on Usable Privacy and Security, Santa Clara, Ca : Usenix. Récupéré de https://www.usenix.org/system/files/soups2019-abdi.pdf

Amilcare. (2020, 18 juin). Théorème de Nyquist-Shannon – ElettroAmici [Vulgarisation]. Elettroamici. Récupéré de https://www.elettroamici.org/fr/teorema-di-nyquist-shannon/

Aspose Pty. (s. d.). What is a WAV file? [Corporatif]. Dans Fileformat. Récupéré de https://docs.fileformat.com/audio/wav/

Bhat, H. R., Lone, T. A. et Paul, Z. M. (2017). Cortana - Intelligent Personal Digital assistant : A review. International Journal of Advanced Research in Computer Science, 8(7), 55-57. doi: 10.26483/ijarcs.v8i7.4225

Bhattacharjee, S. (2019, 12 janvier). How AI's Speech Recognition Models works in Siri, Google, Alexa, Cortana and Bixby? Tech Blog. Récupéré de https://www.viainsider.com/ai-voice-recognition-model/

Careless, T. (2019, 10 janvier). Edge/Cortana Forensics. Récupéré de https://www.youtube.com/watch?v=9L_BSAAmJsA

Carlini, N. et Wagner, D. (2018). Audio Adversarial Examples: Targeted Attacks on Speech-to-Text. arXiv:1801.01944 [cs]. doi: arXiv:1801.01944v2

Delaney, J. (2018, 3 août). The Forensics of Cortana on Android [Blog d'information]. The Forensics of Cortana on Android. Récupéré de http://delyjester.blogspot.com/2018/08/the-forensics-of-cortana-on-android.html

Domingues, P. et Frade, M. (2016). Digital Forensic Artifacts of the Cortana Device Search Cache on Windows 10 Desktop. Dans 2016 11th International Conference on Availability, Reliability and Security (ARES) (p. 338-344). Salzburg, Austria : IEEE. doi: 10.1109/ARES.2016.44

Guera, D. et Delp, E. J. (2018). Deepfake Video Detection Using Recurrent Neural Networks. Dans 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS) (p. 1-6). Auckland, New Zealand : IEEE. doi: 10.1109/AVSS.2018.8639163

Iqbal, K. (2019, 13 décembre). What is a WAV file? Récupéré de https://docs.fileformat.com/audio/wav/

Jafar, M. T., Ababneh, M., Al-Zoube, M. et Elhassan, A. (2020). Forensics and Analysis of Deepfake Videos. Dans 2020 11th International Conference on Information and Communication Systems (ICICS) (p. 053-058). Irbid, Jordan : IEEE. doi: 10.1109/ICICS49469.2020.239493

LCDI. (2016, 2 février). Windows 10 Forensics - Part 2. Champlain College. Récupéré de https://www.champlain.edu/Documents/LCDI/archive/Windows__10_Forensics_Part_2.pdf

Mare, S., Roesner, F. et Kohno, T. (2020). Smart Devices in Airbnbs: Considering Privacyand Security for both Guests and Hosts. Dans Proceedings on Privacy Enhancing Technologie. PETS. doi: 10.2478/popets-2020-0035

Mhaidli, A., Kandadai, M., Zou, Y. et Schraub, F. (2020). Listen Only When Spoken To:Interpersonal Communication Cues as SmartSpeaker Privacy Controls. Dans Proceedings on Privacy Enhancing Technologie. PETS. doi: 10.2478/popets-2020-0026

Microsoft. (s. d.). Speech, voice activation, inking, typing, and privacy. Récupéré de https://support.microsoft.com/en-us/help/4468250/windows-10-speech-voice-activation-inking-typing-privacy

Molkenthin, B. (2020, 25 mai). Online CRC Calculator Javascript [Page personnelle]. Dans Sunshine's Homepage. Récupéré de http://www.sunshine2k.de/coding/javascript/crc/crc_js.html

Nguyen, T. T., Nguyen, C. M., Nguyen, D. T., Nguyen, D. T. et Nahavandi, S. (2020). Deep Learning for Deepfakes Creation and Detection: A Survey. arXiv:1909.11573 [cs, eess]. doi: arXiv:1909.11573v1

Nickel, M., Murphy, K., Tresp, V. et Gabrilovich, E. (2016). A Review of Relational Machine Learning for Knowledge Graphs. Proceedings of the IEEE, 104(1), 11-33. doi: 10.1109/JPROC.2015.2483592

Petalbert, A. (2019, 30 décembre). Deepfake Examples That Prove How Scary and Amusing This Technology Is. Dans Tech Times. Récupéré de https://www.techtimes.com/articles/246694/20191230/deepfake-examples-that-prove-how-scary-and-amusing-this-technology-is.htm

Ramokapane, K. M., Mazeli, A. C. et Rashid, A. (2019). Skip, Skip, Skip, Accept!!!: A Study on the Usability of Smartphone Manufacturer Provided Default Features and User Privacy. Communication présentée au Privacy Enhancing Technologies. doi: 10.2478/popets-2019-0027

Roquette, T. (2020, 29 janvier). Les assistants vocaux sont mauvais pour recommander des premiers soins. Dans Radio-Canada.ca. Radio-Canada.ca. Récupéré de https://ici.radio-canada.ca/nouvelle/1496906/technologie-medecine-sante-alexa-siri-google-cortana

Sergère, V. (2015, 25 octobre). A-t-on réellement besoin de l'audio HD (24 bits / 192 KHz) sur les smartphones ? [Connaissances populaires]. Dans Frandroid. Récupéré de https://www.frandroid.com/editoid/318732_a-t-on-reellement-besoin-de-laudio-hd-sur-les-smartphones

Singh, B. et Singh, U. (2017). A forensic insight into Windows 10 Cortana search. Computers & Security, 66, 142-154. doi: 10.1016/j.cose.2017.01.007

Skulkin, O. et de Courcier, S. (2017). Windows Forensics Cookbook. Birmingham : Packt Publishing Ltd.

Villedieu, M. (1988). La voix humaine [Connaissances populaires]. Récupéré de http://www.musimem.com/voix-humaine.htm

Xiph Org. (2005). Ogg Documentation [Organisationnel]. Dans Ogg Logical bitstream framing. Récupéré de https://xiph.org/ogg/doc/framing.html

Xuejing, Y., Yuxuan, C., Zhao, Y., Long, Y., Liu, X., Chen, K., … Gunter, C. A. (2018). CommanderSong: A Systematic Approach for Practical Adversarial Voice Recognition. Dans 27th Usenix Security Symposium (p. 49-64). Baltimore, Maryland, USA : USENIX Association. Récupéré de https://www.usenix.org/system/files/conference/usenixsecurity18/sec18-yuan.pdf

## Chapitre 4

Castro, É. et Depardieu, C. (s. d.). Le timbre et les harmoniques [Site de connaissances]. Dans Casser un verre en cristal ave la voix: Mythe ou réalité? Récupéré de https://tpeeffetsvoixverrecristal.weebly.com/timbre-et-harmoniques.html

Cazade, A. (1999). La reconnaissance vocale : quelques éléments théoriques, pratiques et expérimentaux à l'usage des enseignants de langues. Cahiers de l'APLIUT, 19(2), 88-107. doi: 10.3406/apliu.1999.2942

Centre de recherche informatique de Montréal. (2018). Biométrie vocale : vers une identification incontournable [Site d'un organisme]. Dans Réalisations du CRIM. Récupéré de https://www.crim.ca/fr/realisations/Biometrie-vocale-vers-une-identification-incontournable

Computer Hope. (2017, 10 novembre). What is a Jump List? [Blog de vulgarisation]. Dans Free computer help since 1998. Récupéré de https://www.computerhope.com/jargon/j/jumplist.htm

Hoffman, C. (2017, 28 juillet). Why You Shouldn't Turn Off Virtual Memory on Your Mac [Blog d'information]. Dans How-To Geek. Récupéré de https://www.howtogeek.com/319151/why-you-shouldnt-turn-off-virtual-memory-on-your-mac/

Hoffman, C. (2018, 18 septembre). What Is the Windows Page File, and Should You Disable It? Dans How-To Geek. Récupéré de https://www.howtogeek.com/126430/htg-explains-what-is-the-windows-page-file-and-should-you-disable-it/

Journal de Montréal (Le). (2019, 20 novembre). La voix de la raison : la commande vocale stimule le commerce. Dans Le Journal de Montréal. Récupéré de https://www.journaldemontreal.com/La-voix-de-la-raison-la-commande-vocale-stimule-le-commerce-hx19nat11

Julien, É. (2013). Alignement du chant par rapport à une référence audio en temps réel (Mémoire de maîtrise). Sherbrooke. Récupéré de https://savoirs.usherbrooke.ca/bitstream/handle/11143/6184/MR93337.pdf;sequence=1

Ligh, M. H., Case, A., Levy, J. et Walters, A. (2014). The art of memory forensics: detecting malware and threats in windows, linux, and mac memory. Indianapolis, IN : John Wiley and Sons.

Rouse, M. (2013, novembre). What is memory paging? - Definition from WhatIs.com [Vulgarisation]. Dans SearchServerVirtualization. Récupéré de https://searchservervirtualization.techtarget.com/definition/memory-paging

Russinovich, M. E., Solomon, D. A., Ionescu, A. et Yosifovich, P. (2017). Windows Internals - Part 1 - System architecture, processes, threads, memory management, and more (Seventh edition, vol. 1). Redmond : Microsoft.

Stancill, B., Vogl, S. et Sardar, O. (2019, 25 juillet). Finding Evil in Windows 10 Compressed Memory, Part One: Volatility and Rekall Tools [Blog d'information]. Dans FireEye. Récupéré de https://www.fireeye.com/blog/threat-research/2019/07/finding-evil-in-windows-ten-compressed-memory-part-one.html

## Chapitre 5

Aaronaught. (2009, 31 décembre). What is private bytes, virtual bytes, working set? [Forum]. Dans Stack Overflow. Récupéré de https://stackoverflow.com/questions/1984186/what-is-private-bytes-virtual-bytes-working-set

Krishna, V. (2018, 3 novembre). How to Disable Cortana on Lock Screen in Windows 10. Dans Make Tech Easier. Récupéré de https://www.maketecheasier.com/disable-cortana-windows-lock-screen/

PAL Acoustics Tecchnologies Ltd. (2016, 7 octobre). PAL Acoustics Technology [Corporate website]. Récupéré de http://www.pal-acoustics.com/index.php?a=services&id=162

Verdy, P. (2018, 1er décembre). Registry permissions after 1703 update - Unknown Account [Forum]. Dans Microsoft TechNet. Récupéré de https://social.technet.microsoft.com/Forums/en-US/e6926f16-f36a-49d2-acc6-c4746925d87c/registry-permissions-after-1703-update-unknown-account?forum=win10itprosetup

Viktik. (2015, 25 avril). Tutorial - Using Sysinternals Process Monitor to troubleshoot problems in Windows [Forum]. Dans MalwareTips Community. Récupéré de https://malwaretips.com/threads/using-sysinternals-process-monitor-to-troubleshoot-problems-in-windows.45250/

## Chapitres 6 et 7

Anonyme. (s. d.). Thumbcache Viewer - Extract thumbnail images from the thumbcache_*.db and iconcache_*.db database files. [Téléchargement]. Dans Github. Récupéré de https://thumbcacheviewer.github.io/

Broadcom. (2020a, 1er mai). Using Logs [Site corporatif]. Dans CA NetMaster® Network Management for TCP/IP 12.1. Récupéré de https://techdocs.broadcom.com/us/en/ca-mainframe-software/performance-and-storage/ca-netmaster-network-management-for-tcp-ip/12-1/using/using-logs.html

Broadcom. (2020b, 31 juillet). NCL Processing Environments [Site corporatif]. Dans CA NetMaster® Network Management for TCP/IP 12.1. Récupéré de https://techdocs.broadcom.com/us/en/ca-mainframe-software/performance-and-storage/ca-netmaster-network-automation/12-2/using/using-operator-console-services/ncl-processing-environments.html

Carrier, B. (2005). File system forensic analysis. Boston, Mass. ; London : Addison-Wesley.

Chen, G. (2019, 10 juin). Microsoft Office 365 (Windows Store Version) How load COMAddIn ? what's mean REGISTRY\WC** [Forum]. Dans Microsoft Community. Récupéré de https://answers.microsoft.com/en-us/msoffice/forum/msoffice_o365admin-mso_domains-mso_o365b/microsoft-office-365-windows-store-version-how/d826c9a8-522a-4450-974c-aac18b857493

Costas, K. (2020). An examination of Win10 ActivitiesCache.dbdatabase. Github. Récupéré de https://kacos2000.github.io/WindowsTimeline/WindowsTimeline.pdf

Darkdefender. (2019, 27 mai). Windows 10 Mail App Forensics. Dans Medium. Récupéré de https://medium.com/@melanijan93/windows-10-mail-app-forensics-39025f5418d2

DLL Files Database. (s. d.). Récupéré de https://dll.website/

Dubois, D. J., Kolcun, R., Mandalari, A. M., Paracha, M. T., Choffnes, D. et Haddadi, H. (2020). When Speakers Are All Ears: Characterizing Misactivations of IoT Smart Speakers. Proceedings on Privacy Enhancing Technologies, 2020(4), 255-276. doi: 10.2478/popets-2020-0072

extensionfile.net. (s. d.). File extension database [Site de connaissances]. Dans File extension darabase. Récupéré de https://extensionfile.net/database

Filext. (2000). FILExt - The File Extension Source [Base de données ouvertes]. Dans Filext. Récupéré de https://filext.com/

Heddings, L. (2019, 30 avril). SysInternals Pro: Understanding Process Monitor [Site de connaissances]. Dans How-To Geek. Récupéré de https://www.howtogeek.com/school/sysinternals-pro/lesson4/

Inconnu. (s. d.). File Signature Database:: D0CF11E0 File Signatures [Base de données ouvertes]. Dans File Signature. Récupéré de https://filesignatures.net/index.php?search=D0CF11E0&mode=SIG

Jenic, Y. (2019, 5 mai). What is CryptnetUrlCache directory and how do I remove it? Dans Windows Report | Error-free Tech Life. Récupéré de https://windowsreport.com/cryptneturlcache/

Khanse, A. (2017, 2 octobre). Local, LocalLow and Roaming folders in AppData on Windows 10 explained [Site de connaissances]. Dans The Windows Club. Récupéré de https://www.thewindowsclub.com/local-localnow-roaming-folders-windows-10

Mangan, T. (2017, 8 septembre). App-V 1703 Virtual Registry and Containers? – Confessions of a Guru [Blog de vulgarisation]. Blogs from TMurgent. Récupéré de https://www.tmurgent.com/TmBlog/?p=2692

Mgeeky. (2019, janvier). Procmon Operations. Récupéré de https://gist.github.com/mgeeky/f0d13172d557e5860c0301dbf847de60

Microsoft. (2018, 5 juillet). Package resource indexing and custom build systems - UWP applications [Site corporatif]. Dans Windows developer. Récupéré de https://docs.microsoft.com/en-us/windows/uwp/app-resources/pri-apis-custom-build-systems

Mikhailov, I. (2019, 10 avril). No Time to Waste: How Windows 10 Timeline Can Help Forensic Experts. Récupéré de https://www.group-ib.com/blog/windows10_timeline_for_forensics

Msuhanov. (2019, 4 janvier). What writes to the Syscache hive? [Blog de vulgarisation]. My DFIR Blog. Récupéré de https://dfir.ru/2019/01/04/what-writes-to-the-syscache-hive/

NXlog. (2020, 18 septembre). NXLog User Guide [Text]. Dans nxlog.co. Récupéré de https://nxlog.co/documentation/nxlog-user-guide/eventlog-about.html

Peart, D. (2016, 9 mai). Windows XP Boot Milestones & Behaviour. Récupéré de http://danielpeart.net/pdf/Windows%20XP%20Boot%20Milestones%20&%20Behaviour.pdf

Pr3cur50r. (2018, 3 mai). Windows 10 Timeline – Initial Review of Forensic Artefacts [Blog d'information]. Salt Forensics. Récupéré de https://salt4n6.com/2018/05/03/windows-10-timeline-forensic-artefacts/

Prabhu, S. (2015, 14 juillet). Active Directory - Overview of Active Directory files [Blog d'information]. Dans vembu.com. Récupéré de https://www.vembu.com/blog/active-directory-overview-active-directory-files/

Singh, B. et Singh, U. (2016). Leveraging the Windows Amcache.hve File in Forensic Investigations. J. Digit. Forensics Secur. Law. doi: 10.15394/JDFSL.2016.1429

Skulkin, O. (2019, 21 novembre). Hunting For Attackers' Tactics And Techniques With Prefetch Files [Site de connaissances]. Dans Forensic Focus. Récupéré de https://www.forensicfocus.com/articles/hunting-for-attackers-tactics-and-techniques-with-prefetch-files/

ThumbcacheViewer. (2018, 17 mai). Thumbcache Viewer - Extract thumbnail images from the thumbcache_*.db and iconcache_*.db database files. [Téléchargement]. Dans GitHub. Récupéré de https://thumbcacheviewer.github.io/

Victor. (2019). Comment télécharger et afficher une copie de vos données Windows 10 Cortana [Blog d'information]. Dans Victor, technologie, astuces, avis. Récupéré de https://victoriavette.com/fr/2917-how-to-download-and-view-a-copy-of-your-windows-10-cortana-data.html

Welcome.AI. (s. d.). Lyrebird [Corporatif]. Dans Welcome.AI. Récupéré de https://www.welcome.ai/lyrebird