

LICENCE PROFESSIONNELLE ENQUÊTEUR EN TECHNOLOGIES NUMÉRIQUES

Analyse des fichiers

« Event Trace Log »

Résumé

L'investigation numérique consiste à déceler les traces laissées par des attaquants sur un support numérique. Autrement dit, il s'agit de trouver des éléments pouvant constituer des preuves dans le but de déceler et reconstituer le scénario d'une infraction ou attaque informatique.

Le système d'exploitation Microsoft Windows et d'autres programmes enregistrent régulièrement ce qui se passe sur l'ordinateur afin de tenir un historique des actions effectuées, qu'elles soient initiées par le système lui-même, par des programmes installés ou par des utilisateurs.

La recherche d'artefacts Windows peut mener les enquêteurs vers des zones de recherches non traditionnelles.

L'objectif de ce mémoire est de présenter l'une de ces recherches non traditionnelles qui consiste en la collecte et l'analyse des différents fichiers de journalisation nommés « Event Trace Log », qui sont présents sur les systèmes d'exploitation Microsoft.

Mots clés :
Fichier « .etl »
Collecte en live
Collecte post-mortem

TABLE DES MATIÈRES

REMERCIEMENTS.....	6
INTRODUCTION	7
1 GENÈSE DE LA JOURNALISATION	8
1.1 Définition de la journalisation	8
1.2 But de la journalisation	8
1.3 Les différentes méthodes de journalisation	9
1.3.1 Fonctionnement de l'ETW	9
1.3.2 Les différents formats de journalisation d'événements	11
1.3.2.1 Format « .txt »	11
1.3.2.2 Format « .evt ».....	12
1.3.2.3 Format « .evtx ».....	13
1.3.2.4 Format « .etl ».....	13
2 JOURNAL DE SUIVI D'ÉVÉNEMENTS	14
2.1 Utilité des fichiers « .etl »	14
2.2 Durée de vie des fichiers « .etl ».....	14
2.3 Localisation des fichiers « .etl »	15
2.4 Intérêts pour l'investigation numérique (forensique)	17
Fichiers « ShutdownCKCL.etl » et « BootCKCL.etl »	17
Fichier « energy-ntkl.etl ».....	18
Fichier « ExplorerStartupLog.etl »	18
Fichier « CortanaTrace1.etl »	18
Fichier « Wifi.etl ».....	19
Fichier « LwNetLog.etl »	19

Fichiers « WindowsUpdate.date.hour...etl »	19
3 COLLECTE DES FICHIERS « .ETL »	20
3.1 Collecte en <i>live</i>	20
3.1.1 Prérequis.....	20
3.1.2 Désactivation de la politique de sécurité d'exécution des scripts	21
3.1.3 Obtention des droits « NT AUTHORITY\SYSTEM »	21
3.1.4 Collecte des fichiers « .etl »	23
3.1.5 Réactivation de la politique de sécurité d'exécution des scripts	24
3.1.6 Vérification de la collecte	24
3.2 Collecte <i>post-mortem</i>	26
3.2.1 Prérequis.....	26
3.2.2 Préparation avant collecte sous X-ways.....	26
3.2.3 Collecte des fichiers « .etl »	29
4 EXPLOITATION DES FICHIERS COLLECTÉS.....	31
4.1 Exploitation des fichiers « .etl » par application.....	31
4.1.1 L'Observateur d'événements	32
4.1.2 ETLParseur	34
4.1.3 Microsoft Message Analyzer (MMA).....	36
4.1.4 Windows Performance Analyzer (WPA).....	37
4.1.5 PerfView.....	41
4.1.6 Tela64	46
4.2 Avantages et inconvénients des applications.....	48
CONCLUSION	50
INDEX	51

LISTE DES FIGURES.....	52
LISTE DES TABLEAUX.....	55
BIBLIOGRAPHIE	56
ANNEXES.....	57
Annexe 1 – Script collecte en live	57
Annexe 2 – Exemple de fichier « list_file_elt.txt »	58

REMERCIEMENTS

Je souhaite avant tout remercier mon tuteur, pour le temps qu'il a consacré à m'apporter les outils méthodologiques indispensables à la conduite des cas tutorés. Sa rigueur et son exigence m'ont grandement stimulé.

L'enseignement de qualité dispensé par la licence « LPETN » a également su alimenter mes réflexions et a représenté une profonde satisfaction intellectuelle. Je remercie donc les enseignants qu'ils soient du CNFPJ, de l'UTT ou intervenants extérieurs, pour leurs conseils avisés dans les domaines techniques et juridiques.

Je souhaite exprimer ma gratitude à mes collègues, trop nombreux pour être cités, qui ont pris le temps de discuter de mon sujet. Chacun de ces échanges m'a permis de faire avancer mon analyse.

Je tiens à remercier ma hiérarchie, qui m'a donné la possibilité de suivre cette formation longue et exigeante pendant un an.

Je n'oublie pas mes camarades de promotion avec lesquels j'ai appris, échangé, partagé tant de choses d'un point de vue professionnel, académique et aussi amical. Je tiens à les remercier pour leur confiance et pour m'avoir intégré dans leurs univers professionnels et parfois personnels.

Enfin, mes derniers remerciements vont à mes chers amis pour leur précieuse aide à la relecture et à la correction de mon mémoire.

INTRODUCTION

Les techniques d'analyses inforensiques en environnement Microsoft se basent notamment sur différents artefacts tels que la base de registre, l'analyse de la mémoire vive, les **prefetchs** (Windows enregistre sur disque tout ce que fait chaque application exécutée pendant dix secondes), les **jumplists** (menu contextuel affichant la liste des raccourcis), la **MFT** (Master File Table, contient la liste de l'ensemble des fichiers stockés sur le disque dur), les journaux d'événements au format « **.evt** » et « **.evtx** ».

Les attaquants essayant de laisser le moins de traces possible de leurs actions malveillantes veilleront à effacer leurs traces, voire dans la mesure du possible, supprimer certains artefacts.

Dans l'hypothèse où les journaux d'événements seraient effacés, modifiés et supprimés, il existe malgré tout une autre source d'informations retraçant l'activité d'un système d'exploitation. Cette source d'informations assez récemment identifiée se matérialise sous forme de fichiers nommés « **ETL** », pour « **Event Trace Log** » (*journal de suivi d'événements*, en français).

Ce mémoire a pour but de présenter les fichiers « **.etl** », leur collecte et l'analyse de certains d'entre eux.

1 GENÈSE DE LA JOURNALISATION

1.1 Définition de la journalisation

La journalisation est l'action d'enregistrer dans un journal tout ou une partie des événements qui se produisent dans un système informatique pendant son fonctionnement.

Il est possible de définir deux types de journalisation :

- la **journalisation système** : ce type de journalisation désigne l'enregistrement chronologique des événements survenant au niveau des composants du système d'exploitation. Le niveau de cette journalisation peut être paramétré, afin de filtrer les différents événements selon leur catégorie de gravité ;
- la **journalisation applicative** : ce type de journalisation désigne l'enregistrement chronologique des opérations de la logique métier pendant le fonctionnement d'une application donnée. Notons que la procédure d'un journal applicatif peut, elle-même, correspondre à une exigence du métier. Le journal est alors comme une fonctionnalité faisant partie de la logique applicative, par exemple pour assurer la traçabilité des actions (approche DICT, Disponibilité, Intégrité, Confidentialité, Traçabilité). Par conséquent, la production du journal ne doit pas être arrêtée pendant le fonctionnement de l'application.

1.2 But de la journalisation

Le rôle et les buts de la journalisation sont multiples et souvent liés ou imposés par des contraintes de sécurité.

- **légal** : le but de la journalisation est alors de fournir un élément de preuve juridique. Par exemple, les fournisseurs d'accès à Internet doivent conserver et archiver les journaux concernant les connexions de leurs utilisateurs afin de pouvoir répondre à une réquisition judiciaire ;
- **statistique** : la journalisation permet aussi de fournir des informations statistiques concernant l'usage. Un administrateur de serveur web peut par exemple se demander « *Quelle est la page web de mon site qui est la plus visitée ?* », ou encore, « *D'où proviennent majoritairement les requêtes ?* » ;
- **analyse** : enfin, la journalisation permet d'analyser un problème et de trouver les enchaînements qui ont provoqué le problème. Par exemple « *Que s'est-il passé sur le réseau à 08H03 juste avant que le serveur DNS ne s'arrête ?* », ou encore, « *Quelle requête HTTPS a généré un dysfonctionnement de mon serveur web ?* ».

En informatique de manière générale, parler de « **log** », c'est se référer à une information de niveau plus ou moins haut rapportée par le système d'exploitation ou par une application spécifique. Elle sert à identifier ce qu'elle fait, y compris les erreurs, les problèmes ou les avertissements mineurs, en indiquant la date, l'heure et la seconde. Dans certains cas, on peut identifier la source, l'utilisateur, l'adresse IP et d'autres champs intéressants de l'événement.

Comme on peut le voir, les journaux peuvent nous aider à la compréhension du fonctionnement de notre système pour prévenir les fuites d'informations, ainsi que les comportements inappropriés ou anormaux qui provoquent des erreurs.

1.3 Les différentes méthodes de journalisation

Il existe plusieurs méthodes permettant de transporter et d'archiver les événements journalisés. Le choix et l'usage de ces méthodes sont fonction de l'environnement et de l'application.

- **Fichier plat** : c'est probablement la plus simple des méthodes. Les événements sont stockés dans un fichier en général au format texte.
- **Syslog** : c'est la méthode de journalisation historique utilisée dans le monde Unix. Cette méthode comprend un logiciel de collecte et d'archivage.
- **Base de données** : les événements de journalisation sont stockés dans une base de données.
- Les **sous-systèmes EventLog** et **Event Tracing for Windows (ETW)** : ces systèmes de journalisation sont utilisés dans le monde Microsoft. C'est un moyen standardisé et centralisé pour les applications (et le système d'exploitation) d'enregistrer des événements logiciels et matériels importants, à des fins d'administration du système.

1.3.1 Fonctionnement de l'ETW

Le sous-système « **ETW** » est un mécanisme de journalisation apparu à partir du système d'exploitation Windows 2000 pour le dépannage et les diagnostics. Il exploite un nombre très important d'événements générés par le système d'exploitation chaque seconde.

Il est configurable et permet d'activer ou de désactiver le suivi des événements de manière dynamique. Il facilite un suivi détaillé, dans un environnement de production, sans nécessiter le redémarrage de l'ordinateur ou de l'application.

L'interface de programmation d'application (*API Application Programming Interface*) de suivi d'événements est divisée en trois composants distincts :

- **les contrôleurs** (*controllers*) démarrent et arrêtent une session de suivi d'événements et activent les fournisseurs ;
- **les fournisseurs** (*providers*) fournissent les événements ;
- **les consommateurs** (*consumers*) traitent les événements renvoyés par les fournisseurs.

Le diagramme ci-dessous illustre le modèle de suivi d'événements :

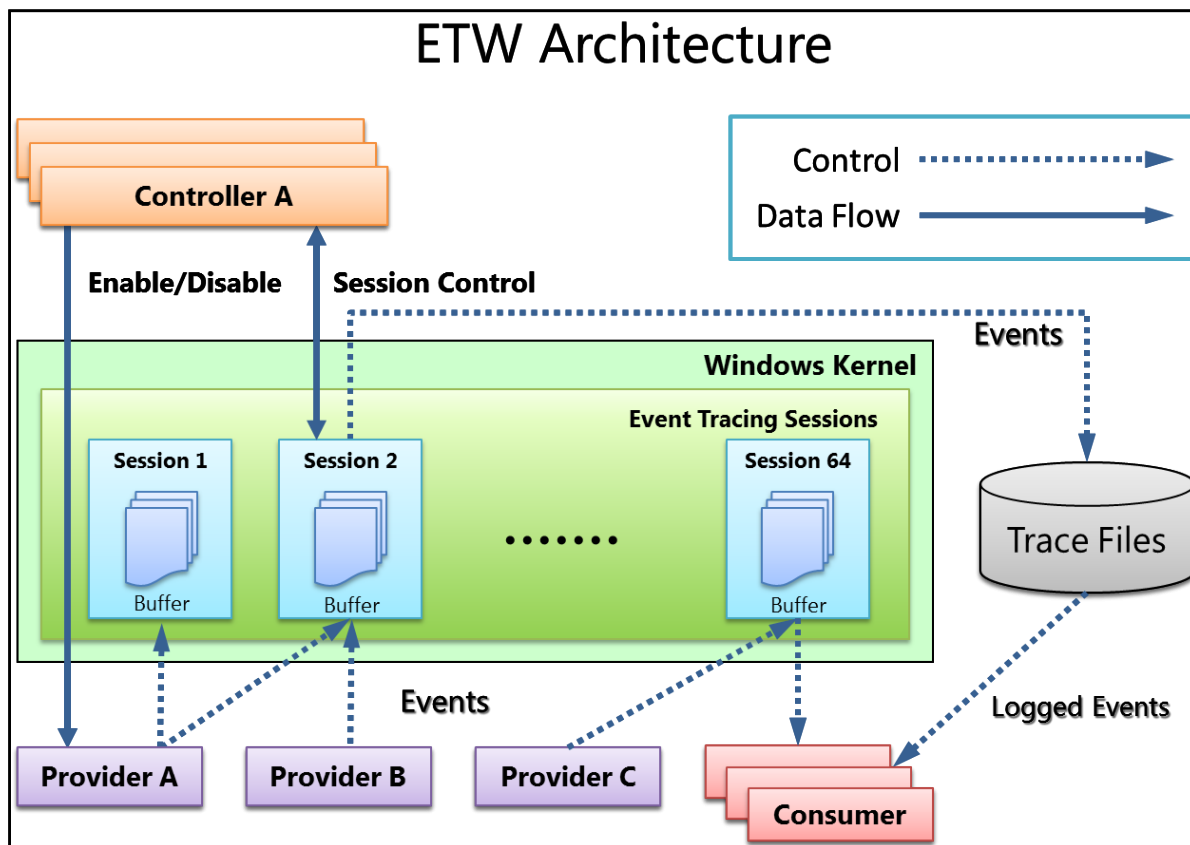


FIGURE 1 : [HTTPS://DOCS.MICROSOFT.COM/EN-US/WINDOWS-HARDWARE/TEST/WEG/INSTRUMENTING-YOUR-CODE-WITH-ETW](https://docs.microsoft.com/en-us/windows-hardware/test/weg/instrumenting-your-code-with-etw)

Les **contrôleurs** sont des applications qui définissent la taille et l'emplacement du fichier journal (.etl), démarrent et arrêtent les sessions de suivi des événements et activent les fournisseurs. Ils travaillent dans le but de consigner les événements, de gérer la taille du pool de mémoires tampons et d'obtenir des statistiques d'exécution pour les sessions.

Les **fournisseurs** sont des applications qui contiennent l'instrumentation de suivi d'événements. Une fois qu'un fournisseur s'est enregistré, un contrôleur peut y activer ou y désactiver le suivi des événements. Le fournisseur définit son interprétation de l'activation ou de la désactivation mais, de manière générale, un fournisseur activé génère des événements, contrairement à un fournisseur désactivé. Cela permet d'ajouter le suivi d'événements à une application sans qu'il ne soit obligé de générer des événements en permanence.

Bien que le modèle « **ETW** » sépare le contrôleur et le fournisseur en applications distinctes, une application peut inclure les deux composants.

Les **consommateurs** sont des applications qui sélectionnent une ou plusieurs sessions de suivi d'événements comme source d'événements. Un consommateur peut demander simultanément des événements de plusieurs sessions de suivi, et le système les réordonne automatiquement dans l'ordre chronologique. Les consommateurs peuvent recevoir des événements stockés dans des fichiers journaux ou des sessions qui délivrent des événements en temps réel. Lors du traitement des événements, un consommateur peut spécifier des heures de début et de fin, et seuls les événements qui se produisent dans le laps de temps spécifié sont remis.

1.3.2 Les différents formats de journalisation d'événements

Il existe 4 types de formats de journalisation d'événements dans les systèmes d'exploitation Microsoft :

- « **.txt** » ;
- « **.evt** » ;
- « **.evtx** » ;
- « **.etl** ».

Le programme rendant lisible entièrement ou partiellement ces 4 formats pour l'utilisateur se nomme « **Observateur d'événements** » (*Event Viewer*). Il est natif au système d'exploitation Windows depuis *Windows NT 3.1* en 1993.

1.3.2.1 Format « .txt »

Les enregistrements des événements Windows depuis les versions NT 3.1 jusqu'à Windows 98 se font dans des fichiers textes. Ces derniers sont situés principalement à la racine du système d'exploitation.

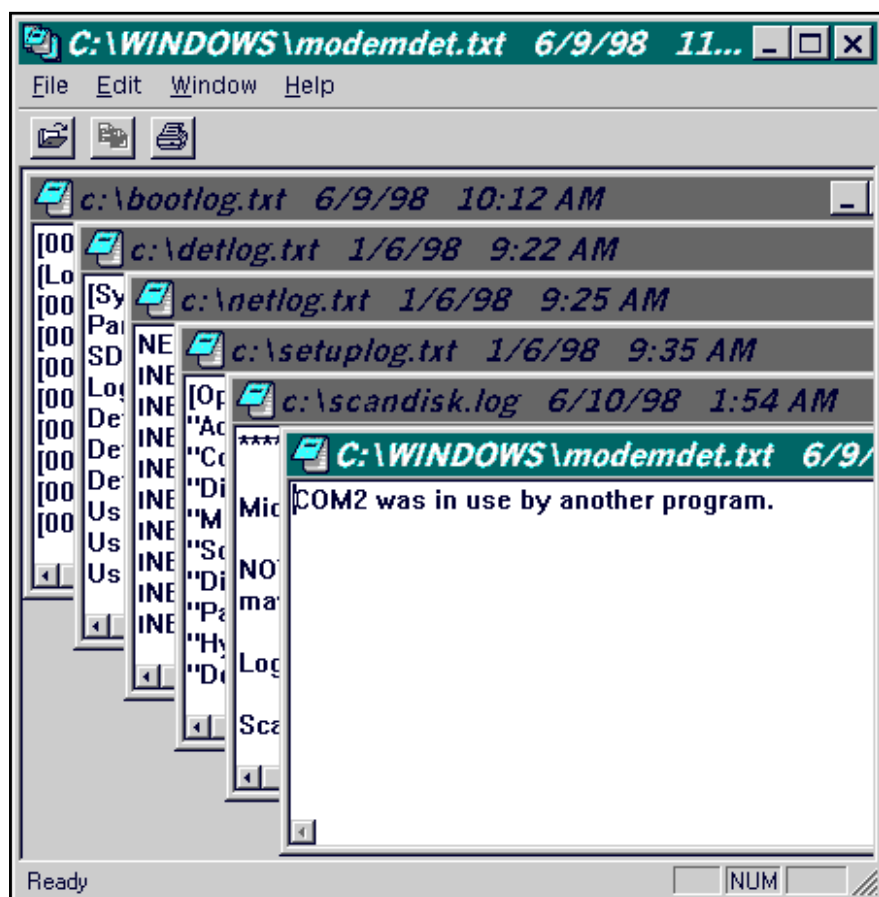
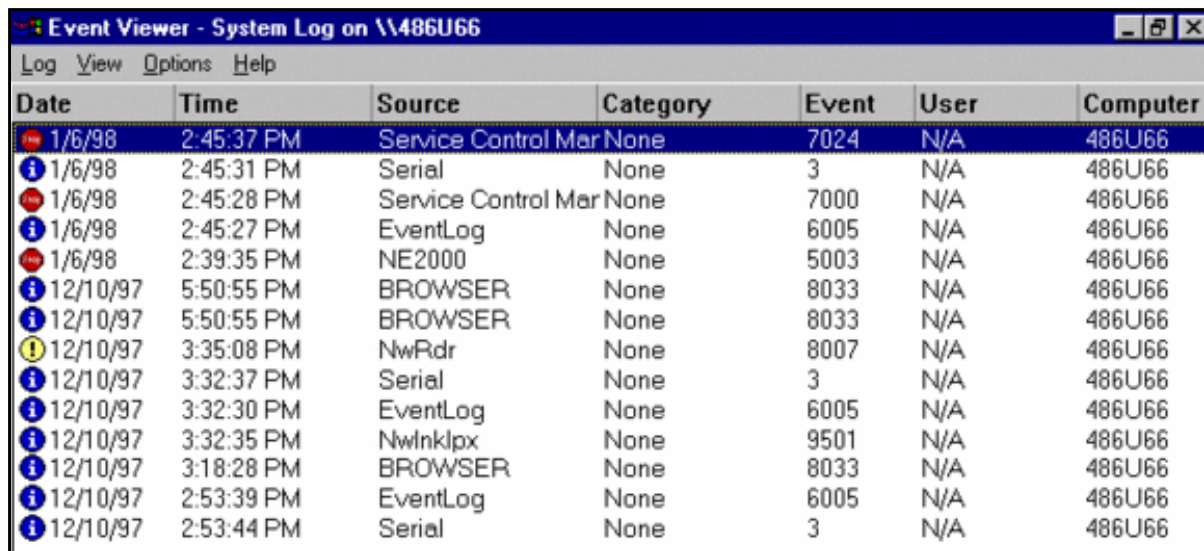


FIGURE 2 : EXEMPLE DE FICHIERS D'ÉVÉNEMENTS AU FORMAT TEXTE SOUS MICROSOFT WINDOWS 95



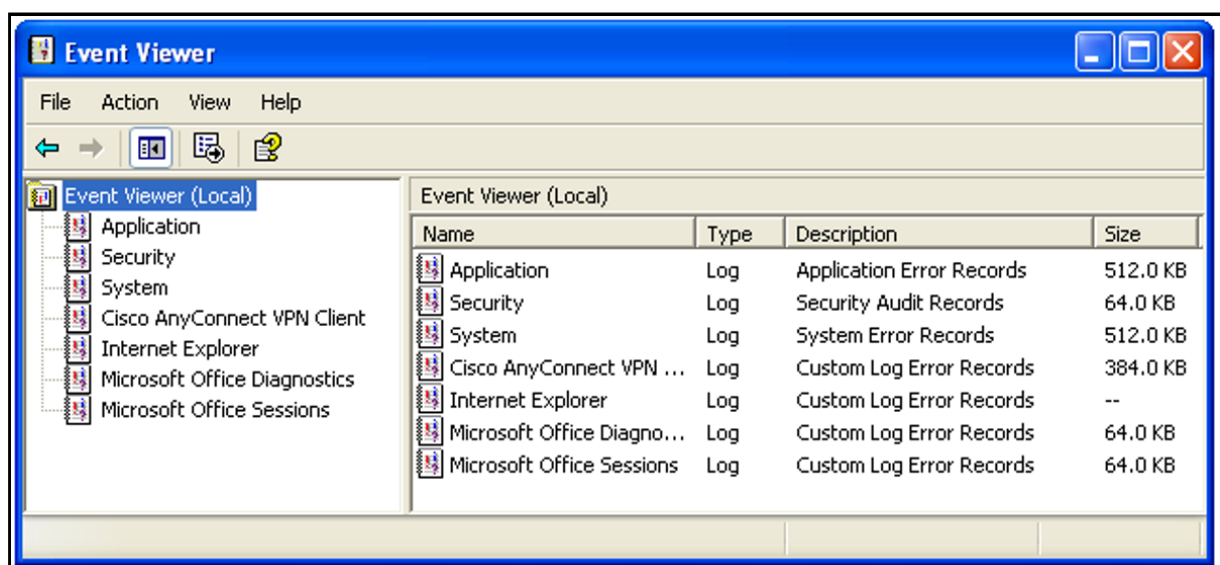
Date	Time	Source	Category	Event	User	Computer
1/6/98	2:45:37 PM	Service Control Mar	None	7024	N/A	486U66
1/6/98	2:45:31 PM	Serial	None	3	N/A	486U66
1/6/98	2:45:28 PM	Service Control Mar	None	7000	N/A	486U66
1/6/98	2:45:27 PM	EventLog	None	6005	N/A	486U66
1/6/98	2:39:35 PM	NE2000	None	5003	N/A	486U66
12/10/97	5:50:55 PM	BROWSER	None	8033	N/A	486U66
12/10/97	5:50:55 PM	BROWSER	None	8033	N/A	486U66
12/10/97	3:35:08 PM	NwRdr	None	8007	N/A	486U66
12/10/97	3:32:37 PM	Serial	None	3	N/A	486U66
12/10/97	3:32:30 PM	EventLog	None	6005	N/A	486U66
12/10/97	3:32:35 PM	Nwlnkpx	None	9501	N/A	486U66
12/10/97	3:18:28 PM	BROWSER	None	8033	N/A	486U66
12/10/97	2:53:39 PM	EventLog	None	6005	N/A	486U66
12/10/97	2:53:44 PM	Serial	None	3	N/A	486U66

FIGURE 3 : EXEMPLE D'ÉVÉNEMENTS DEPUIS L'OBSERVATEUR D'ÉVÉNEMENTS MICROSOFT WINDOWS 95

1.3.2.2 Format « .evt »

Les fichiers journaux d'événements Windows « **.evt** » (*Windows Event Log*) sont apparus sous Windows NT 5.0. Ce sont des fichiers au format binaire propriétaire.

Les versions Microsoft Windows 2000, XP et 2003 conservent généralement trois fichiers journaux d'événements : **Application**, **System** et **Security**. Ils se trouvent par défaut dans le répertoire « **%windir%\system32\config** ». Les versions serveur du système d'exploitation peuvent détenir davantage de journaux d'événements (*DNS Server.evt*, *Directory Service.evt*, *File Replication Service.evt*) selon d'éventuelles fonctionnalités additionnelles du serveur. D'autres journaux d'événements peuvent faire leur apparition en fonction de l'installation d'applications comme le pack Office ou Internet Explorer.



Name	Type	Description	Size
Application	Log	Application Error Records	512.0 KB
Security	Log	Security Audit Records	64.0 KB
System	Log	System Error Records	512.0 KB
Cisco AnyConnect VPN Client	Log	Custom Log Error Records	384.0 KB
Internet Explorer	Log	Custom Log Error Records	--
Microsoft Office Diagnostics	Log	Custom Log Error Records	64.0 KB
Microsoft Office Sessions	Log	Custom Log Error Records	64.0 KB

FIGURE 4 : EXEMPLE DE JOURNAUX DEPUIS L'OBSERVATEUR D'ÉVÉNEMENTS MICROSOFT WINDOWS XP

1.3.2.3 Format « .evtx »

Le format « **.evt** » a été abandonné par Microsoft depuis Windows Vista au profit du format « **.evtx** » (*Windows XML Event Log*). Ce sont des fichiers au format binaire propriétaire reposant sur une structure XML.

Les fichiers journaux d'événements Windows « **.evtx** » se trouvent par défaut dans le répertoire « **%windir%\system32\winevt\logs** ». Ils diffèrent donc des fichiers « **.evt** » par leur emplacement, leur structure et le fait qu'ils proposent un plus grand volume d'informations. Par exemple, Windows 7 peut avoir plus de 70 journaux d'événements uniques contre les trois présents sous Windows XP.

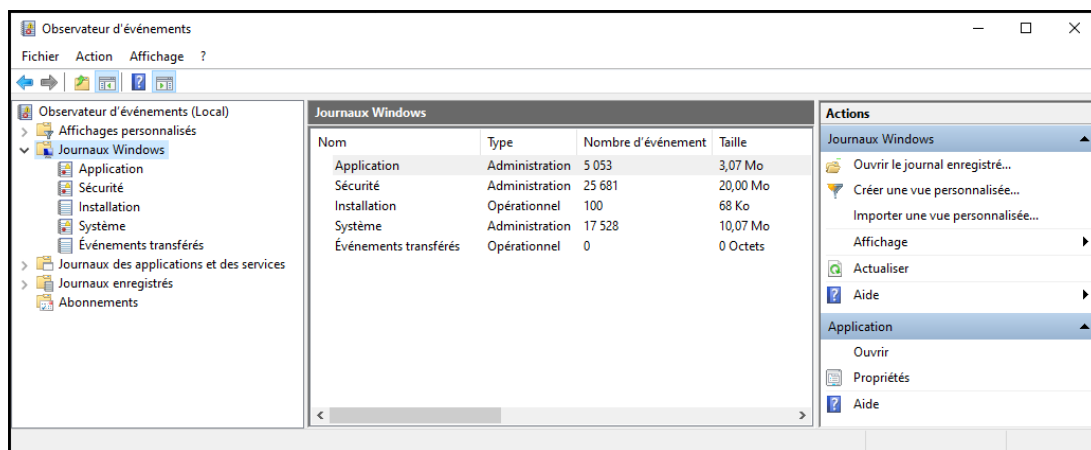


FIGURE 5 : EXEMPLE DE JOURNAUX D'ÉVÉNEMENTS DEPUIS L'OBSERVATEUR D'ÉVÉNEMENTS MICROSOFT WINDOWS 10

1.3.2.4 Format « .etl »

Le format « **.etl** » (*Event Trace Log*) est arrivé depuis le système d'exploitation Vista. Ces fichiers correspondent à une **session de suivi d'événements** (*event trace session*). Il en existe plusieurs centaines. Ils sont enregistrés à différents endroits du système, comme sous :

- **% SystemRoot%\ProgramData\Microsoft\Windows\Power Efficiency Diagnostics ;**
- **%windir%\System32\LogFiles\WMI ;**
- **%windir%\System32\SleepStudy\ScreenOn ;**
- **%windir%\Panther ;**
- **% SystemRoot%\Users\UserName\Tracing\WPPMedia ;**
- ...

Ce sont des fichiers au format binaire propriétaire compressé afin de réduire la quantité d'espace occupé. Leurs contenus doivent être décodés pour être consultés.

Il faut considérer les fichiers « **.etl** » comme un conteneur, de la même manière que les fichiers « **AVI** » sont des conteneurs qui imposent d'avoir les bons codecs pour la lecture des flux vidéo qu'ils contiennent. La lecture d'un fichier « **.etl** » est tout aussi difficile sans le bon outil.

2 JOURNAL DE SUIVI D'ÉVÉNEMENTS

2.1 Utilité des fichiers « .etl »

Les fichiers au format « **.etl** » sont assez peu documentés sur le web. Cela s'explique par le fait que Microsoft ne souhaite pas communiquer sur le fonctionnement du système d'exploitation, d'où la difficulté pour comprendre l'utilité de certains fichiers « **.etl** ».

Ces fichiers sont générés depuis le sous-système « **Event Tracing Windows** » par et pour les développeurs, dont le but est d'obtenir des informations sur le système d'exploitation, les applicatifs ou les comportements de l'utilisateur, en vue notamment :

- d'améliorer les performances de Windows ;
- de relever les dysfonctionnements du système et/ou des applications ;
- de récupérer des statistiques exécutées au démarrage et à l'arrêt de Windows ;
- d'étudier le comportement utilisateur ;
- de surveiller les tâches de mises à jour.

Il est possible à tout développeur, utilisateur ou administrateur de créer ses propres fichiers « **.etl** » pour journaliser une activité particulière sur son poste de travail ou serveur.

Par exemple, il peut être intéressant pour un développeur, pendant qu'une application est en cours d'exécution, de tracer tout problème à tout moment. Il est ainsi possible de surveiller tout ce qu'un développeur souhaite, et pas seulement un dysfonctionnement applicatif.

Le sous-système « **ETW** » est activé par défaut, mais ce qu'il enregistre et quand il enregistre dépend de divers facteurs, y compris la version du système d'exploitation et les applications installées. Les fichiers « .etl » déployés peuvent ainsi être différents d'une machine à l'autre en fonction de l'installation et de l'activité d'utilisation d'applicatifs.

Notons que la suppression des fichiers « **.etl** » n'affectera pas le fonctionnement ou les performances de l'ordinateur ou du serveur.

2.2 Durée de vie des fichiers « .etl »

Les fichiers « **.etl** » stockés sur le disque varient selon leur volatilité et les données qu'ils contiennent. Lorsqu'une session de trace est configurée pour la première fois, les paramètres utilisés déterminent comment les fichiers journaux sont stockés et quelles données y sont enregistrées. Des rotations de ces journaux peuvent être réalisées automatiquement, les anciennes données étant écrasées par de nouvelles lorsque la taille maximale du fichier est atteinte. Les anciens événements qui ont été écrasés ne sont plus récupérables. Les fichiers « **Wifi.etl** » ; « **BootCKCL.etl** » ou encore « **ShutdownCKCL.etl** » sont des exemples de fichiers journaux mettant en œuvre la rotation. D'autres ont des déclencheurs qui entraînent la réinitialisation du contenu du fichier journal, et certains ont plusieurs fichiers journaux pour chaque instance de l'événement.

Il y a également des sessions de suivi d'événements qui peuvent pré-allouer de l'espace disque et écrire des événements sur le disque lorsqu'elles sont déclenchées. Par exemple, l'application *Outlook*, lorsque les paramètres de débogage ont été configurés, écrira des événements dans un fichier journal à la fermeture de l'application *Outlook*.

Pour les fichiers « **.etl** » qui utilisent l'option « *nouveau fichier* », lorsqu'une taille de fichier maximale est atteinte, un nouveau fichier est créé en utilisant une valeur incrémentale comme nom du nouveau fichier.

Le système d'exploitation Windows fait lui aussi l'usage des fichiers « **.etl** », par exemple lorsque le système est arrêté, démarré, qu'un deuxième utilisateur s'est connecté au système, ou lors de l'exécution de mises à jour.

2.3 Localisation des fichiers « .etl »

Les fichiers « **.etl** » se trouvent à différents endroits du système. Le tableau 1 présente une liste non exhaustive des fichiers « **.etl** » et de leurs emplacements.

Emplacement sur le volume système	Fichier ETL
%ProgramData%\Microsoft\Windows\WER\ReportQueue*\	{guid}-WER-MMDDYYYY-####
%ProgramData%\Microsoft\Windows\wfp\	wfpdiag
%ProgramData%\USOShared\Logs\	NotificationUxBroker, UpdateSessionOrchestration
%LocalAppData%\Packages\Microsoft.CommsPhone_...\LocalCache\	CallsAppLog, CallsBackgroundTaskLog,
%LocalAppData%\Packages\Microsoft.Messaging_...\LocalCache\	MessagingBackgroundTaskLog
%LocalAppData%\Packages\Microsoft.Windows.Photos_...\LocalState\	PhotosAppTracing
%LocalAppData%\Diagnostics**	{guid}.Diagnose, {guid}.Repair, {guid}.Verify
%LocalAppData%\ElevatedDiagnostics**	{guid}.Diagnose.Admin, {guid}.Repair.Admin, {guid}.Verify.Admin
%LocalAppData%\Microsoft\Office\	OTeleData
%LocalAppData%\Microsoft\OneDrive\logs\Personal\	TraceCurrent, TraceArchive
%LocalAppData%\Microsoft\Windows\Explorer\	ExplorerStartupLog
%LocalAppData%\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\TempState\Traces	CortanaTrace1.etl
%LocalAppData%\Temp	{guid}.Repair, {guid}.Diagnose, {guid}.Verify
%UserProfile%\Tracing\WPPMedia\	Skype_MediaStack
%SystemRoot%\Logs\dosvc\	dosvc.YYYYMMDD_XXXXX
%SystemRoot%\Logs\NetSetup\	service.x
%SystemRoot%\Logs\SIH\	SIH.YYYYMMDD.XXXXX
%SystemRoot%\Logs\SystemRestore\	Restore.UI
%SystemRoot%\Logs\WindowsBackup\	WBEngine.x
%SystemRoot%\Logs\WindowsUpdate\	WindowsUpdate.YYYYMMDD.XXXXX
%SystemRoot%\Panther\	setup

Emplacement sur le volume système	Fichier ETL
%SystemRoot%\Performance\WinSAT\DataStore\	winsat
%SystemRoot%\ProgramData\Microsoft\Windows\Power Efficiency Diagnostics	energy-ntkl.etl
%SystemRoot%\security\logs\	SceSetupLog
%SystemRoot%\SoftwareDistribution\Download**	WindowsUpdate.YYYYMMDD.xxxx, SceSetupLog, LwtNetLog, Wifi, EtwRTDiagLog, EtwRTEventLog-Application, EtwRTEventLog-System, ShutdownCKCL, BootCKCL, SecondaryLogonCKCL, wfpdiag, KnobsCsp
%SystemRoot%\System32\LogFiles\WMI\	Wifi, FamilySafetyAOT, LwtNetLog, ...
%SystemRoot%\System32\LogFiles\WMI\RtBackup\	EtwRTDiagLog, EtwRTEventLog-System, EtwRTEventLog-Application, EtwRTUBPM, EtwRTEventlog-Security, EtwRTDefenderAuditLogger, EtwRTWFP-IPsec Diagnostics, EtwRTDiagtrack-Listener, EtwRTDefenderApiLogger, ...
%SystemRoot%\System32\NDF\	Eventlog
%SystemRoot%\System32\SleepStudy\	sleepstudy-trace-yyyy-mm-dd-hh-mm-ss, SleepStudyTraceSession
%SystemRoot%\System32\WDI\{guid}\{guid}\	snapshot
%SystemRoot%\System32\WDI\LogFiles\	SecondaryLogonCKCL, ShutdownCKCL,

Emplacement sur le volume système	Fichier ETL
	BootCKCL, ...
%SystemRoot%\System32\winevt\Logs\	AirSpaceChannel, ...

TABLEAU 1 : EMPLACEMENT DES FICHIERS ETL ET LEUR NOM

Ces emplacements et noms de fichier peuvent être différents en fonction des systèmes d'exploitation et des mises à jour.

2.4 Intérêts pour l'investigation numérique (forensique)

Les informations contenues dans les fichiers « **.etl** » peuvent être utilisées en investigation numérique dans une variété de scénarios, par exemple avec les objectifs suivants :

- identification des fichiers existants puis supprimés ;
- liste des contacts et autres informations sur Microsoft Lync (nouvellement Skype) ;
- liste des réseaux Wifi auxquels la machine s'est connectée et informations associées ;
- services et applications en cours d'exécution ;
- informations de configuration du système, y compris les informations sur les lecteurs physiques et logiques.

Rappelons que l'existence des fichiers « **.etl** » sur un système dépend de nombreux facteurs. De ce fait, ils peuvent ne pas exister sur un système en cours d'analyse. Lors de mes recherches, certains fichiers n'existaient pas, certains étaient vides et d'autres contenaient de très grandes quantités de données plus ou moins intéressantes.

Microsoft Lync, Office, OneDrive, SkyDrive et *Skype* peuvent également gérer leurs propres fichiers « **.etl** » contenant des informations de débogage et d'autres informations.

Il est à noter que les artefacts répertoriés dans ce mémoire, n'aborde qu'une infime partie de ce qui est renseigné dans les fichiers « **.etl** ». Dans les sections suivantes, une liste de fichiers « **.etl** » intéressants dans le domaine de l'investigation numérique est présentée.

Fichiers « ShutdownCKCL.etl » et « BootCKCL.etl »

Ces fichiers contiennent des informations sur le système que la session de suivi des événements connaît au moment où elle est arrêtée ou démarrée. Ils se situent à l'emplacement « **%windir%\System32\WDI\LogFiles** ». Ces 2 fichiers « **.etl** » fournissent des informations, notamment sur les traces d'exécution de fichiers DLL et système, en lien avec les exécutables, les processus, les threads et autres chargés au moment du démarrage ou de l'arrêt du système.

Le fichier « **BootCKCL.etl** » est écrasé à chaque démarrage du système.

Le fichier « **ShutdownCKCL.etl** » est écrasé à chaque arrêt du système.

Intérêt forensic : il est intéressant de savoir si un programme malveillant était exécuté lors du démarrage du système, s'il y a un mécanisme de persistance, l'exécution d'une tâche planifiée, une DLL chargée, ou autre.

Fichier « energy-ntkl.etl »

Ce fichier contient des informations telles que les paramètres d'alimentation, la configuration liée aux disques physiques présents sur le système, les informations de configuration liées aux cartes réseau, les informations de configuration liée aux processeurs, aux périphériques « **plug and play** » identifiés, et bien d'autres informations au moment du démarrage de la session du suivi des événements. Il se situe à l'emplacement « %
SystemDrive%\ProgramData\Microsoft\Windows\Power Efficiency Diagnostics ».

Intérêt forensic : il est intéressant de savoir si un ou plusieurs périphériques étaient connectés à un ordinateur ou serveur au moment d'un incident cyber, ou d'obtenir des informations détaillées sur les lecteurs internes ou externes.

Fichier « ExplorerStartupLog.etl »

Ce fichier est créé au démarrage du système. Il contient une grande variété d'événements sur les informations relatives au *shell* (interface utilisateur graphique), aux partages réseau, aux applications nécessitant des privilèges élevés et aux informations RunKey. Il se situe sous l'emplacement « %
SystemDrive%\Users\<UserName>\AppData\Local\Microsoft\Windows\Explorer ».

Intérêt forensic : il est intéressant de savoir si une ou plusieurs applications étaient exécutées au moment de l'ouverture d'une session utilisateur.

Fichier « CortanaTrace1.etl »

Ce fichier est créé au moment de l'exécution de l'application Cortana. Il se peut que ce fichier n'existe pas si l'application Cortana est inutilisée. Il contient les recherches effectuées vocalement à l'application. Il se situe à l'emplacement « %
SystemDrive%\Users\<UserName>\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\TempState\Traces ».

Intérêt forensic : il est intéressant de savoir quelles recherches ont été effectuées par le biais de l'application Cortana vocalement.

Fichier « Wifi.etl »

Ce fichier est créé sur un ordinateur ou un serveur possédant et utilisant une carte WiFi. Il contient les événements liés au réseau WiFi, plus particulièrement des événements sur la configuration automatique, les SSID du réseau à proximité ou ceux auxquels la carte wifi s'est connectée. On y retrouve également des informations sur les adresses MAC des points d'accès et de nombreuses autres informations. Ce fichier se situe à l'emplacement « **%windir%\System32\LogFiles\WMI** ».

Intérêt forensic : il est intéressant de savoir sur quels hotspots wifi s'est connecté un ordinateur en vue de retracer les déplacements et l'agenda d'un utilisateur.

Fichier « LwNetLog.etl »

Ce fichier est créé sur le système dès que le service réseau est utilisé. Il contient les événements liés au réseau, comme les informations de configuration réseau, les adresses MAC des équipements contactés, et d'autres informations. Il se situe à l'emplacement « **%windir%\System32\LogFiles\WMI** ».

Intérêt forensic : il est intéressant de savoir quelle était l'activité réseau d'un utilisateur ou programme utilisant le service réseau.

Fichiers « WindowsUpdate.date.hour...etl »

Ces fichiers se créent lors d'une mise à jour du système d'exploitation. Alors que les journaux de Windows Update étaient stockés dans le fichier « *WindowsUpdate.log* » sous « **%windir%** » sur Windows 7 et Windows 8. Ce n'est désormais plus le cas depuis Windows 8.1. En effet, les journaux sont accessibles depuis le répertoire « **%windir%\logs\WindowsUpdate** ». De plus, ils ne sont plus générés dans le même format. Désormais, ils se présentent sous la forme de fichiers « **.etl** ». Ils contiennent des informations sur les mises à jour du système d'exploitation, la réussite ou échec des mises à jour, sur l'état de connexion du réseau, la version du client Windows Update, et bien d'autres informations utiles.

Intérêt forensic : il est intéressant de savoir si les mises à jour Windows sont actives, si elles s'installent correctement et surtout depuis quels serveurs elles sont téléchargées.

3 COLLECTE DES FICHIERS « .ETL »

La première phase de l'investigation est la collecte des informations à analyser. Pour ce faire, il existe deux types de collecte : la **collecte en live** (sur système en cours d'exécution) et la **collecte post-mortem** (sur système éteint).

La phase de collecte est très importante, car elle nécessite de respecter une procédure stricte qui n'altérera pas les données stockées.

3.1 Collecte en *live*

La phase de collecte des données en *live*, est la phase qui permet de copier des **données volatiles** et **non volatiles** d'un système en cours d'exécution, sur un support externe, dans le but de réaliser par la suite une analyse approfondie.

La collecte manuelle en *live* présentée, est adaptée pour un prélèvement sur un poste de travail ou serveur. La finalité de cette collecte vise à récupérer l'ensemble des fichiers « **.etl** » présents sur le support de manière récursive tout en respectant l'arborescence et la structure des dossiers du système.

3.1.1 Prérequis

La collecte en *live* des fichiers « **.etl** » nécessite en premier lieu la récupération de deux outils :

- l'outil « **PsExec64.exe** » de la suite « **SystinternalsSuite** » de Microsoft ;
- le script « **Collect_ETL.ps1** ».

A propos de « **PsExec64.exe** »¹ :

SHA-1 de « **PsExec64.exe** » : « **6c79d9ca8bf0a3b5f04d317165f48d4eedd04d40** ».

Ce soft nous permettra notamment d'obtenir les droits de l'utilisateur « **AUTORITÉ NT / Système** » nécessaires à la suite des opérations.

Dans l'exemple présenté, on considère enregistrer cet utilitaire sous « **C:\temp** ».

A propos du script « **Collect_ETL.ps1** »² :

Une fois les droits utilisateurs « **AUTORITÉ NT / Système** » obtenus, l'exécution du script permet d'automatiser la collecte des fichiers « **.etl** » sur l'ensemble du support visé de manière récursive.

SHA-1 du script : « **06950FAE3E4DD31899DFDE93CFDD653FB86684BA** ».

Dans l'exemple présenté, on considère enregistrer ce script sous « **C:\temp** ».

¹ Via le lien suivant : « <https://download.sysinternals.com/files/SysinternalsSuite.zip> ».

² Via le lien suivant : « https://github.com/Nyk0la5/Collect_ETL ».

3.1.2 Désactivation de la politique de sécurité d'exécution des scripts

Par défaut, dans un souci de sécurité et afin de ne pas rendre possible l'exécution de scripts sans autorisation de l'administrateur, la configuration de Windows ne permet pas l'exécution de scripts PowerShell.

Pour autoriser l'exécution des scripts PowerShell, il suffit de modifier la politique de sécurité appliquée en termes d'exécution des scripts.

- 1) Exécuter « **Windows PowerShell** » en tant qu'administrateur.
- 2) Vérifier la politique de sécurité appliquée à l'exécution des scripts en exécutant la commande :

```
PS C:\WINDOWS\system32> Get-ExecutionPolicy
```

Si la réponse retournée est « **Restricted** », cela signifie que les scripts sont interdits à l'exécution. Il faudra donc modifier cette restriction.

- 3) Autoriser l'exécution des scripts sans restriction en tapant la commande :

```
PS C:\WINDOWS\system32> Set-ExecutionPolicy Unrestricted
```

- 4) Valider le changement de politique de sécurité en tapant la lettre « **T** ».

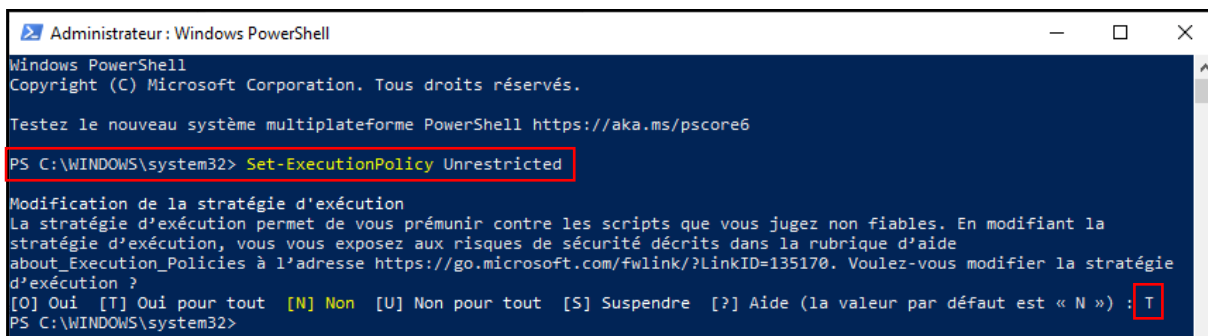


FIGURE 6 : MODIFICATION DE LA POLITIQUE DE SÉCURITÉ DE L'EXÉCUTION DES SCRIPTS

3.1.3 Obtention des droits « NT AUTHORITY\SYSTEM »

Pour la collecte des fichiers « **.etl** », l'utilisation du script *PowerShell* doit être exécuté avec les droits « **AUTORITÉ NT** » (*NT AUTHORITY*). Les utilisateurs « **AUTORITÉ NT** » sont des utilisateurs systèmes intégrés à Windows avec des privilèges plus élevés que celui du compte administrateur. Il existe plusieurs comptes « **AUTORITÉ NT** ». Ils sont utilisés par Windows pour lancer des processus systèmes ou services Windows. Celui qui nous intéresse est « **Système** » (*System*) car ce compte a les privilèges les plus élevés sur Windows. L'utilisation de ce compte permet de collecter l'ensemble des fichiers « **.etl** » sans restriction/blocage par rapport à leur emplacement dans l'arborescence du système.

Afin d'obtenir les privilèges de l'utilisateur « **NT AUTHORITY\SYSTEM** », il faut utiliser l'application « **PsExec64.exe** ».

Depuis une « **Invite de commandes** », exécutée en tant qu'administrateur, renseigner les commandes suivantes :

```
C:\WINDOWS\system32>cd %SystemDrive%\temp  
  
C:\temp>PsExec64.exe -u "NT AUTHORITY\System" -accepteula -i powershell
```

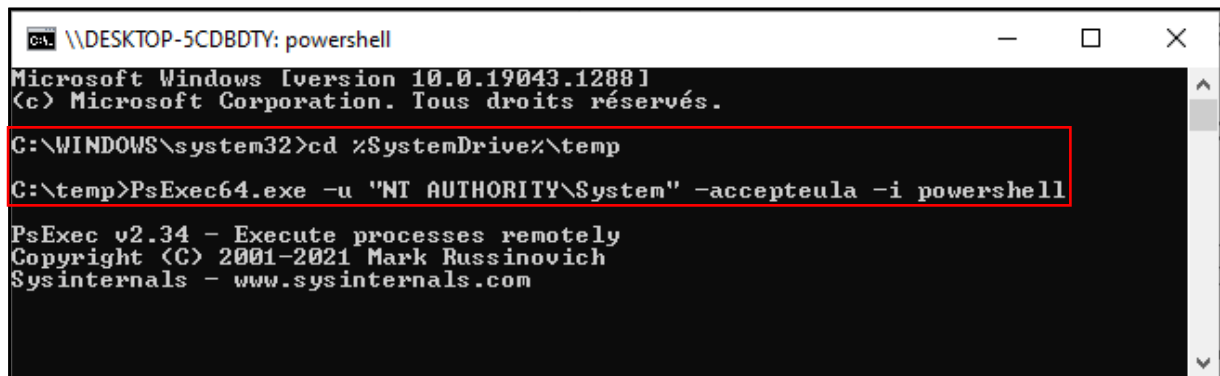


FIGURE 7 : OBTENTION DES DROITS "NT AUTHORITY\SYSTEM" ET EXÉCUTION DE POWERSHELL

Explications des commandes :

cd %SystemDrive%\Temp

Indique de se rendre dans le dossier « **Temp** » situé à la racine du disque.

PsExec64.exe -u "NT AUTHORITY\System" -accepteula -i powershell

Exécute le binaire « **PsExec64.exe** » en tant qu'utilisateur « **NT AUTHORITY\System** », tout en acceptant les termes du contrat de licence logiciel et finir par exécuter « **PowerShell** » avec les droits acquis précédemment (NT AUTHORITY\System).

Une nouvelle fenêtre PowerShell s'ouvre avec pour utilisateur « **NT AUTHORITY\System** ».

La vérification peut se faire grâce à la commande « **whoami** ».

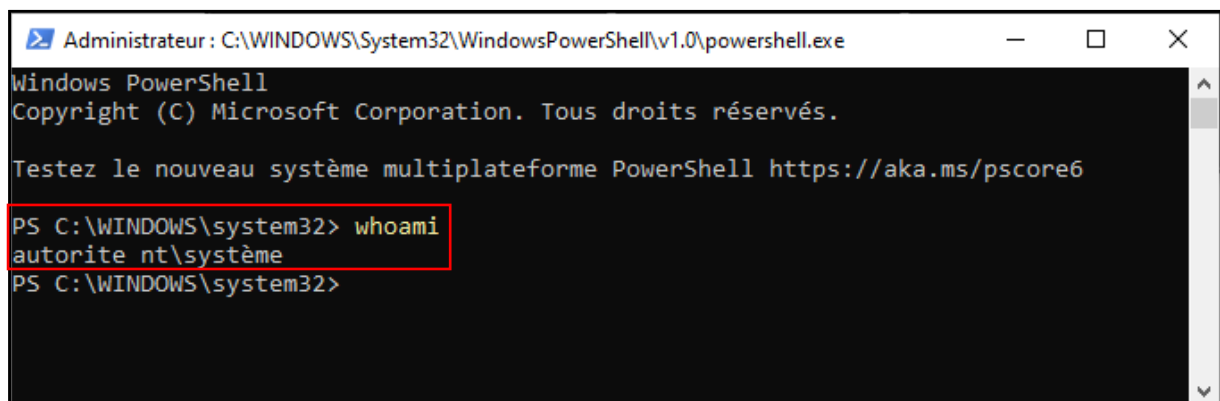


FIGURE 8 : EXÉCUTION DE POWERSHELL ET VÉRIFICATION DES DROITS "NT AUTHORITY\SYSTEM"

3.1.4 Collecte des fichiers « .etl »

Depuis la fenêtre PowerShell ouverte, renseigner la commande suivante :

```
PS C:\WINDOWS\system32> powershell C:\Temp\Collect_ETL.ps1
```

```

Administrateur : C:\WINDOWS\System32\WindowsPowerShell\v1.0\powershell.exe
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\WINDOWS\system32> powershell C:\Temp\Collect_ETL.ps1

Répertoire : C:\

Mode                LastWriteTime         Length Name
----                -
d-----         26/11/2021    10:44             ETL_DESKTOP-5CDBDVT

Répertoire : C:\ETL_DESKTOP-5CDBDVT\Program Files\UNP\SystemLogs

Mode                LastWriteTime         Length Name
----                -
-a-----         26/11/2021    10:44             0 UpdateNotificationPipeline.001.etl
-a-----         26/11/2021    10:44             0 UpdateNotificationPipeline.002.etl
-a-----         26/11/2021    10:44             0 UpdateNotificationPipeline.003.etl
-a-----         26/11/2021    10:44             0 UpdateNotificationPipeline.004.etl
-a-----         26/11/2021    10:44             0 UpdateNotificationPipeline.005.etl
-a-----         26/11/2021    10:44             0 UpdateNotificationPipeline.006.etl
-a-----         26/11/2021    10:44             0 UpdateNotificationPipeline.007.etl
-a-----         26/11/2021    10:44             0 UpdateNotificationPipeline.008.etl
-a-----         26/11/2021    10:44             0 UpdateNotificationPipeline.009.etl
    
```

FIGURE 9 : EXÉCUTION DU SCRIPT DE LA COLLECTE DE FICHIERS ETL

La collecte peut prendre quelques secondes à quelques minutes, en fonction du nombre de fichiers à collecter et leur taille. Exemple, sur un système Windows 10 professionnel d'utilisation bureautique, la collecte mettra une minute.

Une fois la collecte terminée, PowerShell sera en attente de nouvelles instructions et à partir de cet instant, il sera possible de fermer l'ensemble des applications ouvertes ayant servi à la collecte.

```

Administrateur : C:\WINDOWS\System32\WindowsPowerShell\v1.0\powershell.exe

PS C:\WINDOWS\system32>
    
```

FIGURE 10 : FIN DE LA COLLECTE, POWERSHELL EN ATTENTE D'INSTRUCTIONS

3.1.5 Réactivation de la politique de sécurité d'exécution des scripts

À la suite de la modification de la politique de sécurité concernant l'exécution des scripts sans restriction afin de faciliter la collecte de fichiers « **.etl** », il nous faut maintenant réactiver la politique par défaut.

1) Interdire l'exécution des scripts en tapant la commande :

```
PS C:\WINDOWS\system32> Set-ExecutionPolicy Restricted
```

2) Valider le changement de politique de sécurité en tapant la lettre « **T** ».

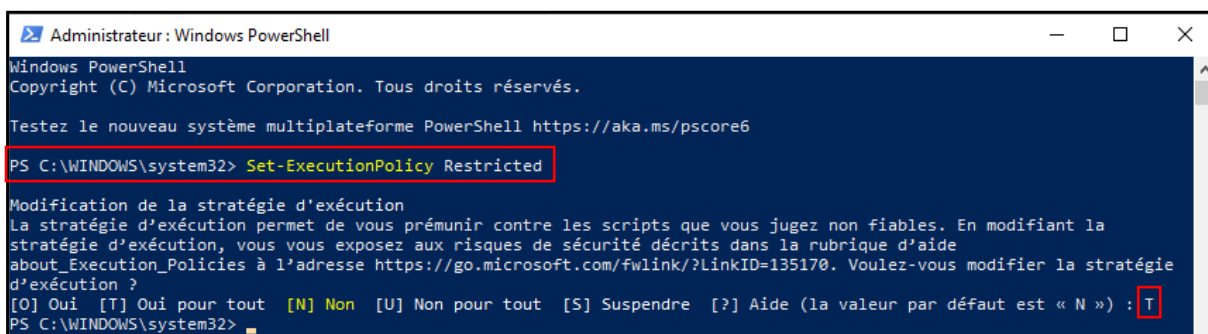


FIGURE 11 : RÉACTIVATION PAR DÉFAUT DE LA POLITIQUE DE SÉCURITÉ DE L'EXÉCUTION DES SCRIPTS

3.1.6 Vérification de la collecte

Un dossier est créé à la racine du système. Il se nomme « **ETL_suivi_du_Hostname** », *Hostname* correspondant au nom d'hôte de l'ordinateur ou du serveur qui a été prélevé.

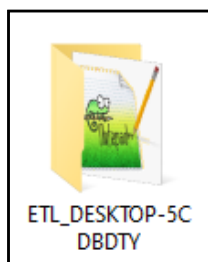


FIGURE 12 : EXEMPLE DE DOSSIER CRÉÉ À LA RACINE DU SYSTÈME

Ce dossier contient la collecte, à savoir l'arborescence des dossiers des fichiers « **.etl** », ainsi que les fichiers eux-mêmes. Il contient également un fichier texte nommé « **List_files_etl.txt** » (voir fichier en annexe 2), qui contient l'ensemble des fichiers « **.etl** » collectés et leur emplacement. Ce fichier permet de savoir rapidement si un fichier « **.etl** » particulier a été collecté.

Dans cet exemple, seulement deux dossiers sont générés. Ils contiennent d'autres sous-dossiers qui eux-mêmes contiennent des fichiers « **.etl** » dans l'état où ils étaient au moment de la collecte. Bien entendu, il se peut que d'autres dossiers soient créés, comme le dossier « **Users** ». Une remarque, il y aura *a minima* un seul dossier généré, celui de « **Windows** ».

Dans l'exemple de la figure 13, seulement deux dossiers sont générés. Ils contiennent des sous-dossiers qui eux-mêmes contiennent des fichiers « **.etl** » dans l'état où ils étaient au moment de la collecte. Bien entendu, il se peut que d'autres dossiers soient créés, comme le dossier « **Users** ». Une remarque, il y aura *a minima* un seul dossier généré, celui de « **Windows** ».

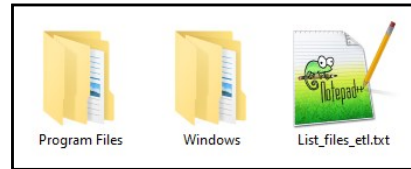


FIGURE 13 : EXEMPLE DE DOSSIERS ET FICHIER CRÉÉS PENDANT LA COLLECTE

L'exemple de la figure 14 montre plusieurs dossiers générés pendant la collecte :

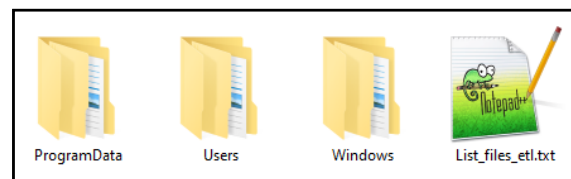


FIGURE 14 : AUTRE EXEMPLE DE DOSSIERS ET FICHIER CRÉÉS PENDANT LA COLLECTE

L'exemple de la figure 15 montre un aperçu du fichier nommé « **List_files_etl.txt** » :

Répertoire : C:\Program Files\UNP\SystemLogs				
Mode	LastWriteTime		Length	Name
----	-----	-----	-----	----
-a----	14/01/2021	15:54	327680	UpdateNotificationPipeline.001.etl
-a----	14/01/2021	15:30	327680	UpdateNotificationPipeline.002.etl
-a----	14/01/2021	14:52	327680	UpdateNotificationPipeline.003.etl
-a----	14/01/2021	14:52	327680	UpdateNotificationPipeline.004.etl
-a----	08/01/2021	11:49	327680	UpdateNotificationPipeline.005.etl
-a----	08/01/2021	08:10	327680	UpdateNotificationPipeline.006.etl
-a----	07/01/2021	13:44	327680	UpdateNotificationPipeline.007.etl
-a----	07/01/2021	08:39	327680	UpdateNotificationPipeline.008.etl
-a----	06/01/2021	07:58	327680	UpdateNotificationPipeline.009.etl
-a----	05/01/2021	13:01	327680	UpdateNotificationPipeline.010.etl
-a----	05/01/2021	07:50	327680	UpdateNotificationPipeline.011.etl
-a----	05/01/2021	07:50	327680	UpdateNotificationPipeline.012.etl
-a----	04/01/2021	09:30	327680	UpdateNotificationPipeline.013.etl
-a----	04/01/2021	09:14	327680	UpdateNotificationPipeline.014.etl
-a----	02/01/2021	18:00	327680	UpdateNotificationPipeline.015.etl
-a----	02/01/2021	17:59	327680	UpdateNotificationPipeline.016.etl
-a----	30/12/2020	19:48	327680	UpdateNotificationPipeline.017.etl

Répertoire : C:\Windows\Logs\NetSetup				
Mode	LastWriteTime		Length	Name
----	-----	-----	-----	----
-a----	09/11/2021	19:53	1245184	service.0.etl
-a----	06/11/2021	15:11	3211264	service.1.etl

FIGURE 15 : EXEMPLE DU FICHIER TEXTE APRÈS COLLECTE

L'exemple de la figure 16 montre un sous-dossier généré et des fichiers « **.etl** » :

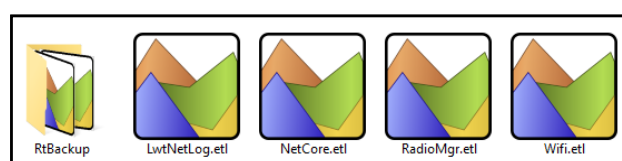


FIGURE 16 : EXEMPLE DE FICHIERS ET DOSSIER CRÉÉS

3.2 Collecte *post-mortem*

La phase de collecte *post-mortem* dite également **à froid**, se réalise une fois que le système est éteint. L'objectif est ici d'extraire les fichiers « **.etl** » du ou des supports de données d'un ordinateur ou serveur éteint, éventuellement en travaillant sur des copies de disque ou de machines virtuelles.

Pour cela, il est possible d'extraire les supports physiques, de les connecter à des bloqueurs en écriture (périphérique informatique permettant à un ordinateur de consulter le contenu d'un support numérique sans en modifier le contenu) et d'utiliser le script de récupération des fichiers « **.etl** » en modifiant les variables utiles afin que le script ne se focalise que sur les fichiers des supports physiques spécifiés.

Il est également possible de connecter les supports directement à des bloqueurs en écriture et d'utiliser l'application « **X-ways** » pour parcourir ces supports et permettre de filtrer sur les noms de fichier en « ***.etl** » pour les obtenir tous immédiatement. Cette solution est la plus simple et devrait être privilégiée.

3.2.1 Prérequis

- 1) Détenir l'application « **X-ways** » et sa licence matérielle (*dongle*).
- 2) Détenir une copie de disque ou un disque système derrière un bloqueur ou même une copie de machine virtuelle.
- 3) Créer un dossier de destinations pour recevoir les fichiers « **.etl** » collectés.

3.2.2 Préparation avant collecte sous X-ways

La méthodologie présentée utilise l'import d'une machine virtuelle (fichier .vmdk Windows 10), mais fonctionne également avec l'import d'un support numérique.

L'utilisation de l'application « **X-ways** » version 20.1 SR-9 x64 est utilisée dans le cadre de cette procédure. La langue paramétrée est l'anglais.

- 1) Création d'un cas sous : **File -> Create New Case**.
- 2) Attribuer un nom au cas : **Case title/number**.
- 3) Attribuer un dossier pour le cas : **Directory**.
- 4) Attribuer examinateur : **Examiner(s)**.
- 5) Importer un fichier de machine virtuelle : **File -> Add Image...** -> sélectionner l'image à importer.

6) Vérifier les condensats cryptographiques et le type des fichiers sur les partitions de l'image : sélectionner le dossier « **Case Root** » (afin que les actions suivantes soient prises en compte pour l'ensemble des partitions de l'image) -> **Specialist** -> **Refine Volume Snapshot...** -> cocher les cases « **Compute Hash** » et « **Verify file type** » -> **OK** -> **OK**.

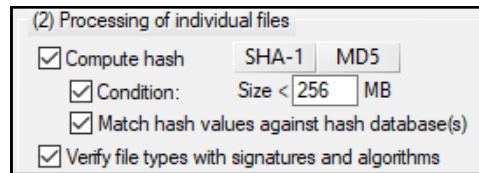


FIGURE 17 : CASES À COCHER - REFINE VOLUME SNAPSHOT

7) Filtrer sur l'ensemble des fichiers de type « **.etl** » : sélectionner l'icône du filtre de la colonne « **Type** » -> rechercher et cocher la case « **etl - Event trace log** » sous la section « **Windows Internals** » -> **Activate**.

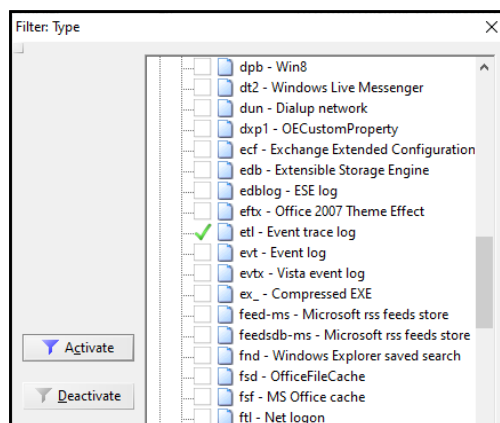


FIGURE 18 : SÉLECTION DU TYPE DE FICHIER À FILTRER

8) Exécuter la récursivité sur l'ensemble du « **Case Root** » : Clic droit sur « **Case Root** » -> sélectionner l'ensemble des partitions -> **OK**.

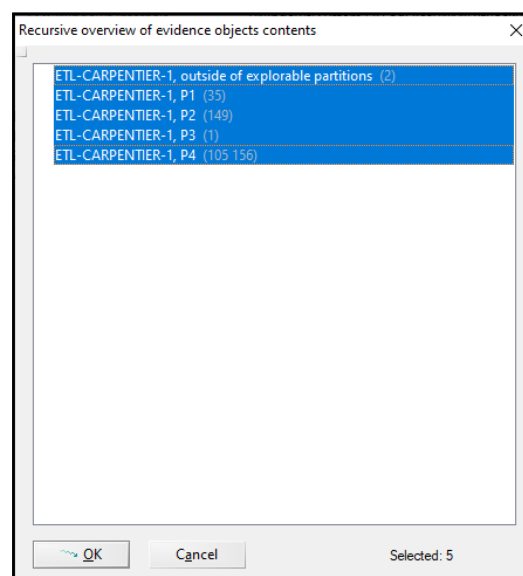


FIGURE 19 : RÉCURSIVITÉ DE LA RECHERCHE DES FICHIERS ETL

1 Case Root: ETL-CARPENTIER-1 Unpartitioned space, Partition 1, Partition 2, Partition 3, Partition 4					
Name	Size	Path	Type	Hash set	Hash
AirSpaceChannel.etl	4,0 KB	\\Windows\\System32\\winevt\\Logs	etl		C8A12847B86063473F25C517C305E15AE983B560
AutoLogger-Diagtrack-Listener.etl	128 KB	\\ProgramData\\Microsoft\\Diagnosis\\ETLLogs\\Sh...	etl		8ED96ED1B9286DD7CE5555C3B5C86D45BD7F4BD
BootCKCL.etl	12,0 MB	\\Windows\\System32\\WDI\\LogFiles	etl		C31925B8F317FCF42A867E838460A59484F46CF
dosvc.134546.1.etl	64,0 KB	\\Windows\\Logs\\dosvc	etl		1CF35F7D06D04DF9B36C924D6A1BB49F46D0ADC9
dosvc.136125.1.etl	64,0 KB	\\Windows\\Logs\\dosvc	etl		BA9EBE85187E6E97B01D79A1DDAD7AFDC1FEA699
dosvc.137421.1.etl	64,0 KB	\\Windows\\Logs\\dosvc	etl		0FCA0174AE1FB2390603671C6D288780E0A2D1B
dosvc.140453.1.etl	64,0 KB	\\Windows\\Logs\\dosvc	etl		1D6D0B81FECB6EA1E4FED50EA2FAD5D0F95CE7CE
dosvc.141312.1.etl	64,0 KB	\\Windows\\Logs\\dosvc	etl		6B740F93CF795162BCAC9996F627E43780A51139
dosvc.149873.1.etl	64,0 KB	\\Windows\\Logs\\dosvc	etl		F3D89AC72AECF6BFF497D1EA05603FF2AEBDCFFE
dosvc.168265.1.etl	64,0 KB	\\Windows\\Logs\\dosvc	etl		C2F88F28B02255831C3CDDA0059DAD46A1535954
dosvc.5638359.1.etl	64,0 KB	\\Windows\\Logs\\dosvc	etl		E110DD7656B2419685A1A5D3D302392C6792ECFF
EtwRTDefenderApiLogger.etl	0 B	\\Windows\\System32\\LogFiles\\WMI\\RtBackup	etl		
EtwRTDefenderAuditLogger.etl	72 B	\\Windows\\System32\\LogFiles\\WMI\\RtBackup	etl	NSRLFile	0D0E47938F6E00166E7352732DDFB7C610F44DB2
EtwRTDiagLog.etl	376 B	\\Windows\\System32\\LogFiles\\WMI\\RtBackup	etl		B56A2035AC5FC14BAB93AE842DE4096D51524999
EtwRTEventLog-Application.etl	72 B	\\Windows\\System32\\LogFiles\\WMI\\RtBackup	etl		DDF98936E4D95A52825067D1A45DC407BD386939
EtwRTEventLog-Security.etl	72 B	\\Windows\\System32\\LogFiles\\WMI\\RtBackup	etl	NSRLFile	0D0E47938F6E00166E7352732DDFB7C610F44DB2
EtwRTEventLog-System.etl	496 B	\\Windows\\System32\\LogFiles\\WMI\\RtBackup	etl		75797508C68DC65A2D75B00C62A6F97CC15FC96D
EtwRTUBPM.etl	72 B	\\Windows\\System32\\LogFiles\\WMI\\RtBackup	etl	NSRLFile	0D0E47938F6E00166E7352732DDFB7C610F44DB2
EtwRTWFP-IPsec Diagnostics.etl	0 B	\\Windows\\System32\\LogFiles\\WMI\\RtBackup	etl		
ExplorerStartupLog.etl	448 KB	\\Users\\user\\AppData\\Local\\Microsoft\\Windows...	etl		8BFB4C2AEAFF3414494F79A6E42E8B7A1FA083399
ExplorerStartupLog.etl	424 KB	\\Users\\Administrateur\\AppData\\Local\\Microsoft...	etl		E563E024F65CC918619D9FE8CCBCA3429492E01
LwtNetLog.etl	576 KB	\\Windows\\System32\\LogFiles\\WMI	etl		20E20A689B4619425400A38F6BCC279AE6681E22
Microsoft-RMS-MSIPC%4Debug.etl	4,0 KB	\\Windows\\System32\\winevt\\Logs	etl		C8037C44949292F4F4F4AA43EC8CA1DD31D20242
netcfgx.0.etl	768 KB	\\Windows\\INF	etl		CF5EA2A28B4B2B0D201201AF53952692AD6C5C94
PhotosAppTracing_BGTASK.etl	64,0 KB	\\Users\\user\\AppData\\Local\\Packages\\Microsoft...	etl		120FFB0A157BE4FAF4FDAC7B7831294983C6ADA3
PhotosAppTracing_BGTASK.last.etl	64,0 KB	\\Users\\user\\AppData\\Local\\Packages\\Microsoft...	etl		6E2B5931D257EBE079F2C78EA85AAC7F08EA4D9
SecSetupLog.etl	48,0 KB	\\Windows\\security\\logs	etl	NSRLFile	0A953863D43CE93D0D9BFC5563B964DB45838BE
SecondaryLogonCKCL.etl	11,8 MB	\\Windows\\System32\\WDI\\LogFiles	etl		F07ECE1F36B420D5347AD98BDA4E1A57A5A3E36B0
setup.etl	316 KB	\\Windows\\Panther	etl		0160BD228EACBC7D743D31982587B6FE7DD0051
ShutdownCKCL.etl	1,7 MB	\\Windows\\System32\\WDI\\LogFiles	etl		D72FA2611249B18C2520A800617D54D2A2395855
SIH.20210824.135800.984.1.etl	8,0 KB	\\Windows\\Logs\\SIH	etl		B76FF60B3F0799E84E6359087D231A48D0C0F570
SIH.20210824.173217.393.1.etl	8,0 KB	\\Windows\\Logs\\SIH	etl		2C45DB19B6E5EFD53B1F455142B5AE2300E81BAD
SIH.20210928.155109.052.1.etl	8,0 KB	\\Windows\\Logs\\SIH	etl		79E59905CA832C34FF4F50572A01CE999E8BD49B
SIH.20211012.102113.684.1.etl	8,0 KB	\\Windows\\Logs\\SIH	etl		90E4E0C73C82E71C2652786528078013C5B82CFE
SIH.20211028.100033.933.1.etl	8,0 KB	\\Windows\\Logs\\SIH	etl		022CA2D5B1047423038C332613881759F57900AE

FIGURE 20 : RÉSULTAT DU FILTRE SUR LES FICHIERS DE TYPE ETL

9) En fonction du besoin, il est possible de filtrer uniquement sur les noms de certains fichiers « .etl » : sélectionner l'icône du filtre sur la colonne « **Name** » -> cocher la case « **Match against full name** » -> renseigner le ou les noms des fichiers « .etl » à rechercher -> **Activate**.

Filter: Name

☒ Match against full name, allow * wildcards (e.g. *.jpg) and : for NOT
☐ Substring search

wifi.etl
setup.etl
boot*.etl
shutdown*.etl
lw*.etl

☐ NOT ☐ Match case

Activate

Deactivate

Help

FIGURE 21 : FILTRE SUR LE NOM DE FICHIERS ETL

2 Case Root: ETL-CARPENTIER-1 Unpartitioned space, Partition 1, Partition 2, Partition 3, Partition 4			
Name	Size	Path	Type
BootCKCL.etl	12,0 MB	\Windows\System32\WDI\LogFiles	etl
LwtNetLog.etl	576 KB	\Windows\System32\LogFiles\WMI	etl
setup.etl	316 KB	\Windows\Panther	etl
ShutdownCKCL.etl	1,7 MB	\Windows\System32\WDI\LogFiles	etl
Wifi.etl	400 KB	\Windows\System32\LogFiles\WMI	etl

FIGURE 22 : RÉSULTAT DU FILTRE SUR LES NOMS DE FICHIERS ETL

3.2.3 Collecte des fichiers « .etl »

La collecte des fichiers « **.etl** » consiste maintenant à récupérer une copie de ces fichiers en utilisant la fonction « **Recover/Copy...** ».

- 1) Sélectionner le ou les fichiers « **.etl** » à récupérer.
- 2) Clic droit sur cette sélection et choisir : « **Recover/copy** ».

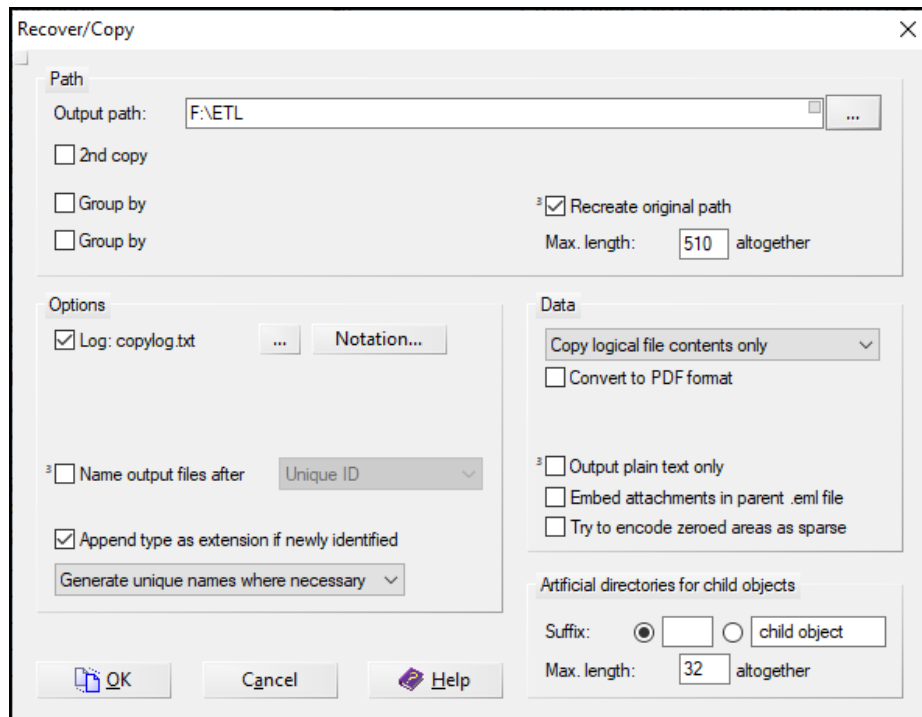


FIGURE 23 : RECOVER/COPY SUR LA SÉLECTION DES FICHIERS ETL

- 2) Sélectionner le dossier de récupération des fichiers « **.etl** » depuis l'option « **Output path:** ».
- 3) Recréer l'arborescence de l'emplacement des fichiers récupérés si besoin : cocher la case « **Recreate original path** » -> **OK**.

Une remarque, cette option est intéressante à activer, afin que certains fichiers « **.etl** » portant le même nom, ne soient renommés par l'application X-ways en sortie pour éviter les doublons.

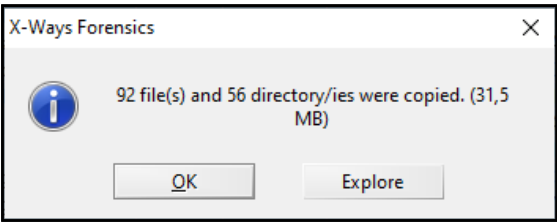


FIGURE 24 : RÉSULTAT DE L'ACTION DE COPIE PAR X-WAYS

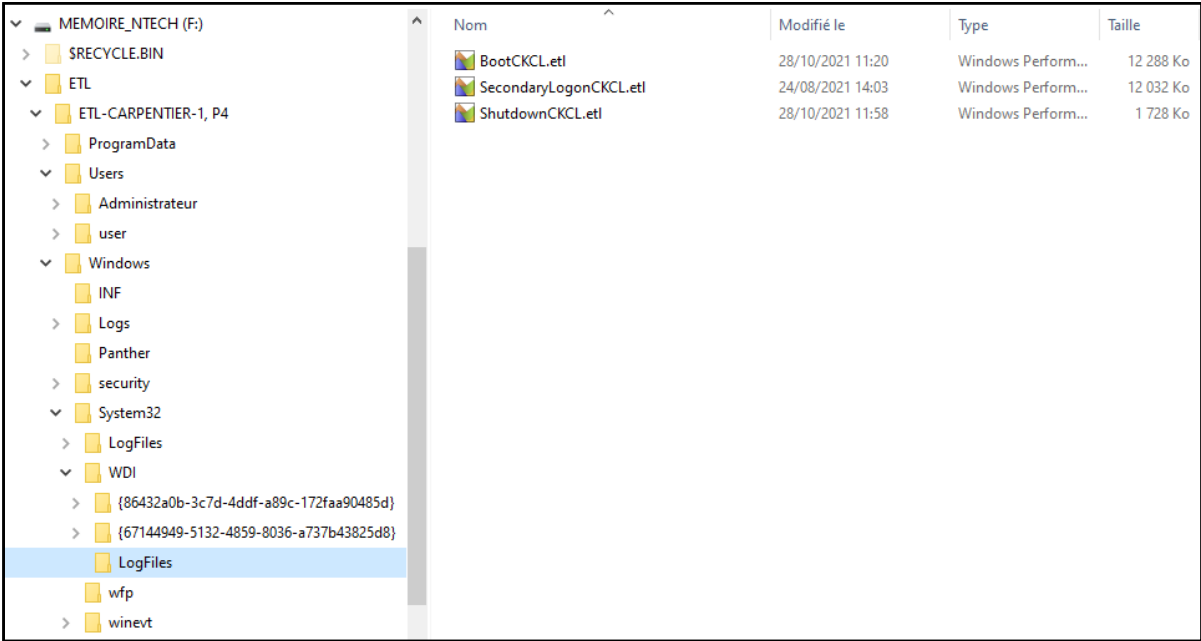


FIGURE 25 : RÉSULTAT DE LA COPIE DES FICHIERS ETL DEPUIS L'EXPLORATEUR WINDOWS

4 EXPLOITATION DES FICHIERS COLLECTÉS

Plusieurs logiciels d'exploitation ou de visualisation de fichiers « **.etl** » existent, certains étant fournis par d'autres entités que Microsoft.

Après plusieurs tests et analyses effectués par le biais de ces outils, il apparaît que ces derniers ne sont pas en mesure de décoder l'intégralité des fichiers « **.etl** », même ceux fournis par Microsoft. Il n'y a donc pas de logiciel à ce jour capable d'ouvrir l'intégralité des fichiers « **.etl** », ce qui rend les analyses complexes.

En plus de l'exploitation incomplète des fichiers « **.etl** » par ces outils, ils ne sont pas très intuitifs et sont donc difficile à prendre en main.

Il s'avère également qu'ils ne sont pas dimensionnés pour exploiter un volume conséquent de fichiers « **.etl** », les outils se concentrant principalement sur l'exploitation d'un seul fichier à la fois.

4.1 Exploitation des fichiers « **.etl** » par application

Citons les applications permettant d'exploiter les fichiers « **.etl** » :

- « **Observateur d'événements** » de Microsoft ;
- « **ETLParser** » de Forensiclunch³ ;
- « **Microsoft Message Analyzer** » de Microsoft⁴ ;
- « **Windows Performance Analyzer** » de Microsoft⁵ ;
- « **Perfview** » de Microsoft⁶ ;
- « **Tela64** » de TZWorks⁷.

Chacun de ces outils va être détaillé pour voir lequel serait le plus adapté à une investigation numérique en exploitant les fichiers « **.etl** ».

Dans un souci d'équité dans le comparatif, les tests d'exploitation sont réalisés avec les mêmes fichiers « **.etl** » pour chaque application.

³ Via le lien suivant : « <https://github.com/forensiclunch/ETLParser> » de Nicole Ibrahim.

⁴ Via le lien suivant : « <https://web.archive.org/web/20191104120853/https://www.microsoft.com/en-us/download/confirmation.aspx?id=44226> ».

⁵ Via le lien suivant : « <https://go.microsoft.com/fwlink/p/?LinkId=845298> ».

⁶ Via le lien suivant : « <https://github.com/microsoft/perfview> ».

⁷ Via le lien suivant : « https://tzworks.com/prototype_page.php?proto_id=40 ».

4.1.1 L'Observateur d'événements

L'observateur d'événements (*event viewer*) est un composant du système d'exploitation de la famille Windows NT de Microsoft. Il permet aux utilisateurs et aux administrateurs de visualiser les journaux d'événements de l'ordinateur local ou sur une machine distante.

Il permet ainsi de conserver et de consulter la trace d'activités logicielles et matérielles, afin d'aider au diagnostic ou à l'optimisation d'un système.

Il prend en compte les fichiers aux formats « **.evt** » ; « **.evtx** » et « **.etl** » de façon unitaire. Il n'est pas possible de lui fournir plusieurs fichiers au format « **.etl** » à traiter en une seule fois. Il faudra donc les importer les uns à la suite des autres.

Exemple d'exploitation d'un fichier « **.etl** » grâce à l'observateur d'événements :

1) Ouvrir l'observateur d'événements.

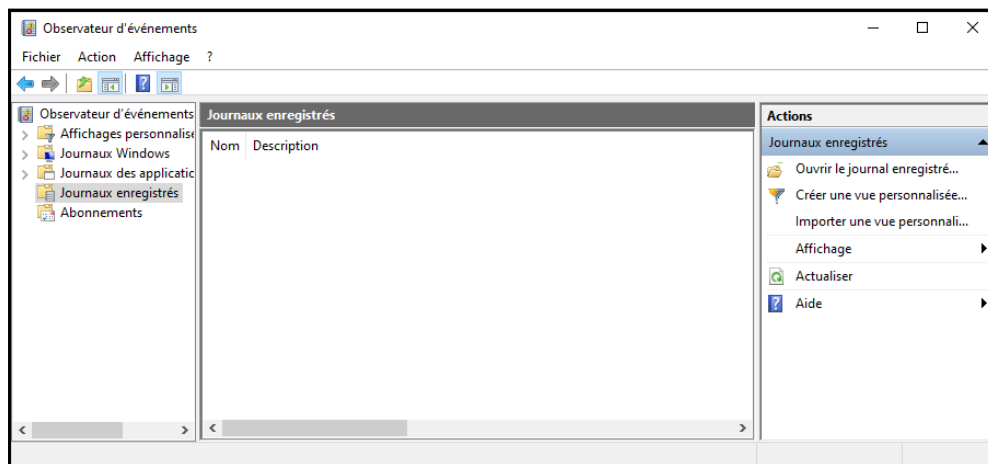


FIGURE 26 : OBSERVATEUR D'ÉVÉNEMENTS MICROSOFT WINDOWS 10

2) Importer un fichier « **.etl** » -> « **Ouvrir le journal enregistré...** » et sélectionner le fichier à analyser -> « **Ouvrir** ». (« **LwNetLog.etl** » contient les événements liés au réseau, comme les informations de configuration réseau, les adresses MAC des équipements contactés, et d'autres informations.)

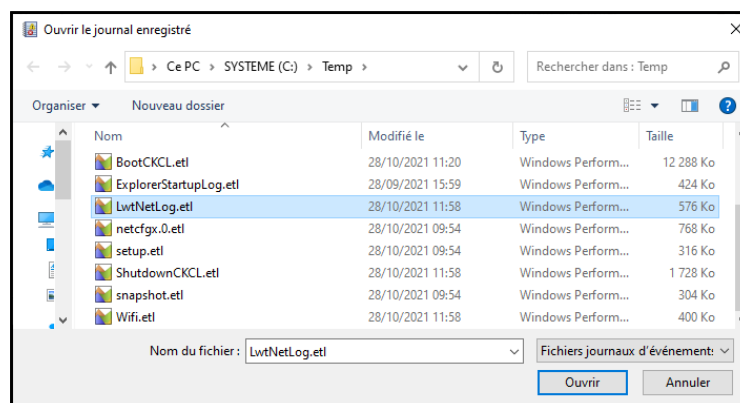


FIGURE 27 : SÉLECTION D'UN FICHIER ETL À ANALYSER

3) Convertir au format de journal d'événements « **.evtx** » -> « **Oui** ».

La conversion permet de traiter en profondeur le fichier « **.etl** » donc de fournir d'avantage d'éléments à l'exploitation.

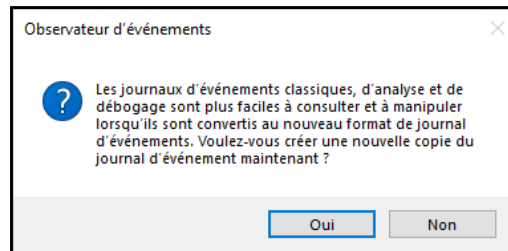


FIGURE 28 : CONVERSION AU FORMAT EVTX DU FICHIER ETL

4) Nommer et donner une description au fichier « **.etl** » importé -> « **OK** ».

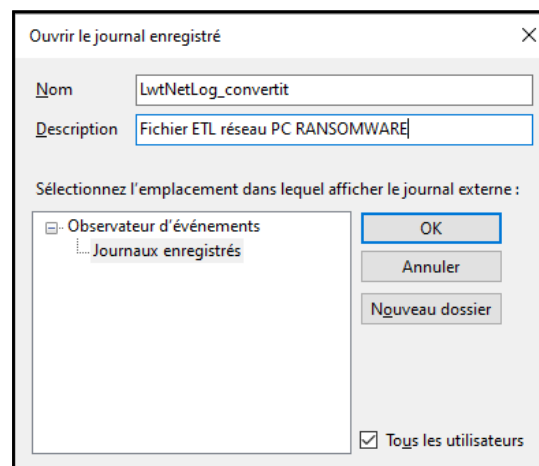


FIGURE 29 : NOMMAGE ET DESCRIPTION DU FICHIER ETL IMPORTÉ

5) Visualiser et exploiter les événements.

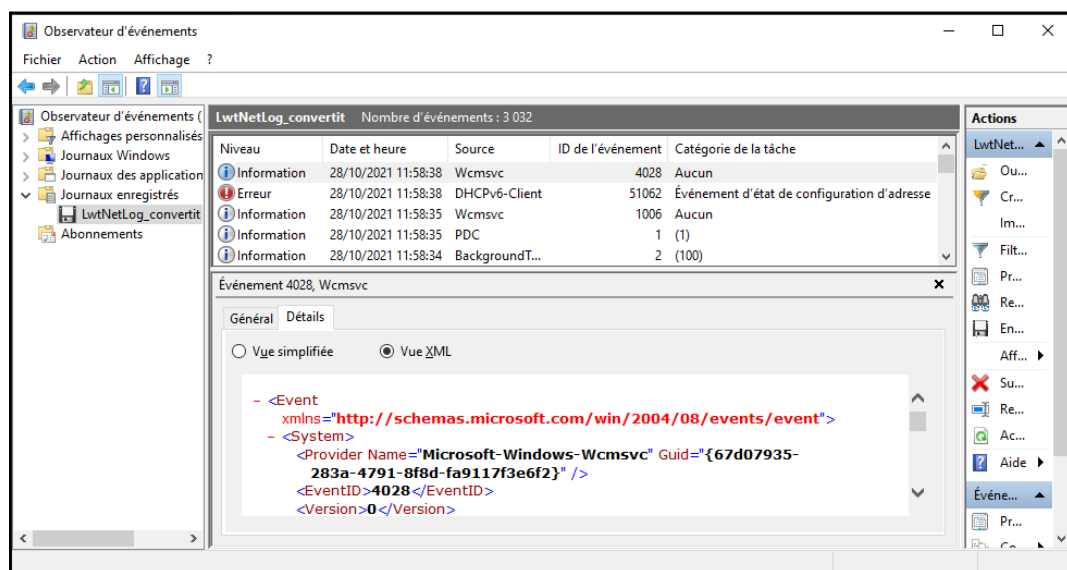


FIGURE 30 : VISUALISATION ET EXPLOITATION DES ÉVÉNEMENTS

6) Pour une exploitation plus rapide ou des recherches par mot clé, il est possible d'exporter les événements aux formats « **.csv** » ou « **.xml** » -> « **Enregistrer tous les événements sous...** ». La conversion peut prendre du temps.

Le fichier « **LwNetLog.etl** » permettra par exemple de retrouver des adresses IP, des requêtes DNS, des adresses MAC, des informations sur les cartes réseaux, leurs paramètres, des numéros de processus et d'autres informations. Ces informations sont très utiles lors d'une analyse post-compromission, par exemple pour identifier les communications d'une machine compromise vers un serveur de contrôle et de commandes.

4.1.2 ETLParser

« **ETLParser.exe** » est un exécutable en ligne de commande pour analyser les fichiers « **.etl** ».

Il est très simple d'utilisation et n'impose aucune limitation en entrée sur le nombre de fichiers « **.etl** » à transcoder. Les formats de sortie après traitement par l'exécutable sont « **.sqlite** » et « **.csv** ».

L'inconvénient de cet exécutable est qu'il prend uniquement en charge que des fichiers « **.etl** » stockés dans un même dossier. Aucune recherche par récursivité de dossier en dossier. Ce qui est gênant en sortie de collecte. En effet, la structure d'arborescence des dossiers étant conservée, il faudra rechercher, sélectionner, copier et coller les fichiers « **.etl** » intéressants pour l'analyse dans un seul dossier avant le lancement de l'exécutable.

Exemple d'exploitation de fichiers « **.etl** » grâce à l'exécutable « **ETLParser.exe** » :

1) Récupérer l'exécutable depuis le site « <https://github.com/forensiclunch/ETLParser> », « **Version 0.3** ».

SHA-1 de l'exécutable : « **5f298495d325ab19bd848e9f11a4901498aaafe5** »

2) Placer l'exécutable sous l'emplacement « **C:\Temp** » accompagné des fichiers « **.etl** » à traiter.

3) Créer le dossier de sortie dans lequel les fichiers « **.sqlite** » et « **.csv** » seront enregistrés.

Nom	Modifié le	Type	Taille
ETL_RANSOMWARE	19/11/2021 10:45	Dossier de fichiers	
Wifi.etl	28/10/2021 11:58	Windows Perform...	400 Ko
snapshot.etl	28/10/2021 09:54	Windows Perform...	304 Ko
ShutdownCKCL.etl	28/10/2021 11:58	Windows Perform...	1 728 Ko
setup.etl	28/10/2021 09:54	Windows Perform...	316 Ko
netcfgx.0.etl	28/10/2021 09:54	Windows Perform...	768 Ko
LwtNetLog.etl	28/10/2021 11:58	Windows Perform...	576 Ko
ExplorerStartupLog.etl	28/09/2021 15:59	Windows Perform...	424 Ko
BootCKCL.etl	28/10/2021 11:20	Windows Perform...	12 288 Ko
ETLParser.exe	08/06/2018 01:07	Application	13 641 Ko

FIGURE 31 : EXEMPLE PRÉREQUIS ETLPARSER

4) Depuis une « **Invite de commandes** », exécutée en tant qu'administrateur, renseigner les commandes suivantes :

```
C:\WINDOWS\system32>cd %SystemDrive%\temp
C:\temp>ETLParser.exe -c RANSOMWARE -s .\ -o .\ETL_RANSOMWARE
```

Explications des commandes :

cd %SystemDrive%\temp

Indique de se rendre dans le dossier « **Temp** » situé à la racine du disque.

ETLParser.exe -c RANSOMWARE -s .\ -o .\ETL_RANSOMWARE

Exécute le binaire « **ETLParser.exe** ». L'option, « **-c RANSOMWARE** » mentionne le nom du cas, « **-s .** » mentionne l'emplacement des fichiers « **.etl** » à parser, « **-o .\ETL_RANSOMWARE** » mentionne le nom du dossier de sortie.

```
Administrateur : Invite de commandes - ETLParser.exe -c RANSOMWARE -s .\ -o .\ETL_RANSOMWARE
Microsoft Windows [version 10.0.19043.1288]
(c) Microsoft Corporation. Tous droits réservés.

C:\WINDOWS\system32>cd %SYSTEMDRIVE%\Temp
C:\Temp>ETLParser.exe -c RANSOMWARE -s .\ -o .\ETL_RANSOMWARE

ETL Parser v0.3, Runtime: 11/19/2021 09:45:22 UTC
=====
Parsing files.....

[BEGIN_PARSE] 11/19/2021 09:45:22 UTC File 1 of 9. Started parsing BootCKCL.etl.
[PARSE_FINISHED] 11/19/2021 09:46:30 UTC 47595 events parsed.

[BEGIN_PARSE] 11/19/2021 09:46:30 UTC File 2 of 9. Started parsing ETLParser.exe.
[PARSE_ERROR] 11/19/2021 09:46:30 UTC Unable to parse.

[BEGIN_PARSE] 11/19/2021 09:46:30 UTC File 3 of 9. Started parsing ExplorerStartupLog.etl.
```

FIGURE 32 : EXÉCUTION DE L'EXÉCUTABLE ETLPARSER

Une fois le traitement terminé, la ligne « *Finished parsing. Total Events Parsed: xxx* » s'affiche et l'invite de commandes attend de nouvelles instructions.

```
Administrateur : Invite de commandes

[BEGIN_PARSE] 11/19/2021 09:46:52 UTC File 5 of 9. Started parsing netcfgx.0.etl.
[PARSE_FINISHED] 11/19/2021 09:46:52 UTC 35 events parsed.

[BEGIN_PARSE] 11/19/2021 09:46:52 UTC File 6 of 9. Started parsing setup.etl.
[PARSE_FINISHED] 11/19/2021 09:46:54 UTC 1164 events parsed.

[BEGIN_PARSE] 11/19/2021 09:46:54 UTC File 7 of 9. Started parsing ShutdownCKCL.etl.
[PARSE_FINISHED] 11/19/2021 09:47:07 UTC 9334 events parsed.

[BEGIN_PARSE] 11/19/2021 09:47:07 UTC File 8 of 9. Started parsing snapshot.etl.
[PARSE_FINISHED] 11/19/2021 09:47:10 UTC 1700 events parsed.

[BEGIN_PARSE] 11/19/2021 09:47:10 UTC File 9 of 9. Started parsing Wifi.etl.
[PARSE_FINISHED] 11/19/2021 09:47:10 UTC 98 events parsed.

-----
Finished parsing. Total Events Parsed: 66175
C:\Temp>
```

FIGURE 33 : FIN DU TRAITEMENT D'ETLPARSER

Résultat du traitement :

 RANSOMWARE_ETL.sqlite	19/11/2021 10:47	Fichier SQLITE	90 125 Ko
 RANSOMWARE_Parsed_ETL.csv	19/11/2021 10:47	Fichier CSV Micro...	76 984 Ko

FIGURE 34 : RÉSULTAT DU TRAITEMENT ETLPARSER

La taille des fichiers en sortie est élevée. Il y avait moins de 17 Mo en entrée de fichiers « **.etl** ».

Il est possible d'ouvrir le fichier « **.sqlite** » grâce à l'application « **DB BROWSER for SQLite** ».

La figure 35 présente un exemple d'export du fichier « **LwNetLog.etl** ».

Payload	Timestamp
138.0.0.35	Filtre
'Interface: 4', 'Protocol: IPv4', 'DadState: 0...	2021-10-28 09:20:55.588421 UTC
'Interface: 4', 'Protocol: IPv4', 'DadState: ...	2021-10-28 09:20:58.632854 UTC

FIGURE 35 : RECHERCHE TEXTE SUR UNE ADRESSE IP DEPUIS DB BROWSER FOR SQLITE

4.1.3 Microsoft Message Analyzer (MMA)

« **Microsoft Message Analyzer** » (MMA) est un outil permettant de capturer, d'afficher et d'analyser le trafic de protocole de messagerie, les événements et d'autres messages système ou d'application dans le cadre du dépannage du réseau. MMA permet également de charger, d'agrégier et d'analyser les données des fichiers journaux et de trace enregistrés. C'est le successeur de « *Microsoft Network Monitor 3.4* ».

Il est principalement utilisé pour exploiter les fichiers « **.etl** » en lien avec le réseau.

« **Microsoft Message Analyzer** » est retiré des sites « microsoft.com » depuis le 25 novembre 2019. Il n'y a actuellement aucun remplacement à cet outil en développement pour le moment. Si « **Microsoft Message Analyzer** » est déjà installé, il est possible de continuer à l'utiliser. L'analyse des traces « **ETW** » continuera également à fonctionner comme avant. Depuis le 25 novembre 2019, « **MMA** » une fois exécuté tente de se connecter au site de « *Microsoft* » pour vérifier ses mises à jour. Un message d'erreur s'affiche, il ne faut pas en tenir compte.

La récupération de cet outil a été réalisée grâce au site « <https://web.archive.org/> » (WayBackMachine). La page permettant de récupérer cet exécutable est « <https://web.archive.org/web/20191104120853/https://www.microsoft.com/en-us/download/confirmation.aspx?id=44226> ».

Il prend en compte les fichiers « **.etl** » de façon unitaire. Il n'est pas possible de lui fournir plusieurs fichiers au format « **.etl** » à traiter en une seule fois. Il faudra les importer les uns à la suite des autres.

Exemple d'exploitation d'un fichier « **.etl** » grâce à Microsoft Message Analyzer :

- 1) Ouvrir Microsoft Message Analyzer.
- 2) Importer un fichier « .etl » réseau -> « **File** » -> « **Open** » -> « **From File Explorer** » -> sélectionner le fichier -> dans cet exemple « **LwtNetLog.etl** » -> « **Ouvrir** ».
- 3) Une fois le traitement terminé, il est possible d'exécuter des requêtes comme sous « **Wireshark** » (outil de capture et d'analyse de paquets réseau).

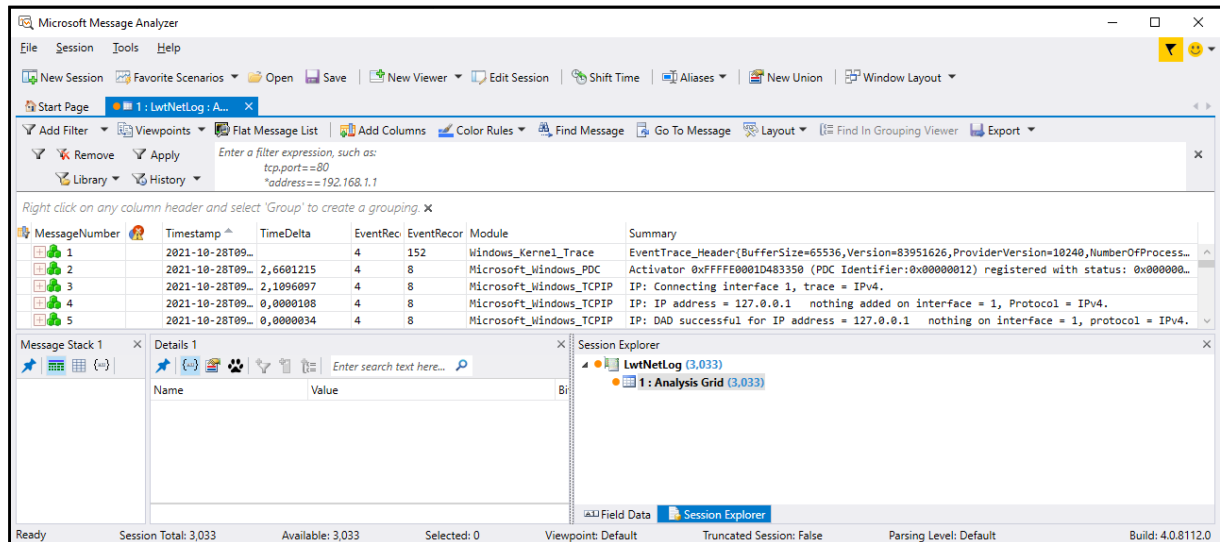


FIGURE 36 : EXPLOITATION D'UN FICHIER ETL SOUS MMA

Cette application a toutefois du mal à traiter les nouveaux fichiers « .etl ». De ce fait, certaines requêtes ne pourront aboutir. Pour palier partiellement à cet inconvénient, il faut exporter les données au format « .csv ».

- 4) Export au format « .csv » des données -> sélectionner « **Export** » -> « **All** » -> renseigner un nom et un emplacement -> « **Enregistrer** ».

4.1.4 Windows Performance Analyzer (WPA)

« **Windows Performance Analyzer** » est un outil qui crée des graphiques et des tables de données de suivi d'événements pour Windows « **ETW** ». « **WPA** » peut ouvrir tout fichier journal de suivi des événements « **ETL** » à des fins d'analyse.

Cet outil est très performant. En revanche, sa prise en main n'est pas intuitive.

Il prend en compte les fichiers « .etl » de façon unitaire. Il n'est pas possible de lui fournir plusieurs fichiers au format « .etl » à traiter en une seule fois. Il faudra les importer les uns à la suite des autres.

L'exécutable « **Windows Performance Analyzer** » fait partie d'un panel d'applications appartenant à la suite « **SDK** » (software development kit) de Microsoft. Pour récupérer cette

suite, il faut télécharger l'exécutable « **winsdksetup.exe** »⁸. Après avoir téléchargé, exécuté, renseigné la destination d'installation, veillez à ne pas autoriser l'envoi de données à Microsoft et accepter la licence. Il faudra au minimum choisir « **Microsoft Performance Toolkit** » pour l'installation. L'installation se fera par téléchargement.

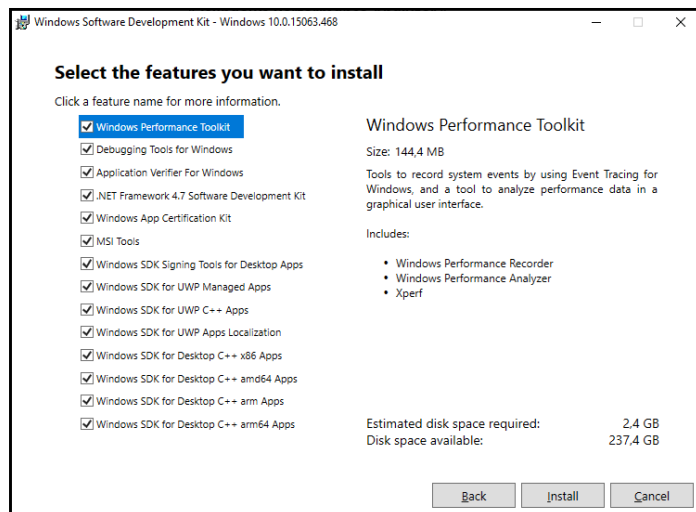


FIGURE 37 : CAPTURE D'ÉCRAN D'INSTALLATION DE WINDOWS SOFTWARE DEVELOPMENT KIT

Exemple d'exploitation d'un fichier « **.etl** » grâce à Windows Performance Analyzer :

- 1) Ouvrir Windows Performance Analyzer.
- 2) Importer un fichier « **.etl** » -> « **File** » -> « **Open...** » -> sélectionner le fichier -> dans cet exemple « **LwtNetLog.etl** » -> « **Ouvrir** ».
- 3) Il est possible d'afficher les propriétés de la trace -> « **Trace** » -> « **Trace Properties** ». pour obtenir des informations sur le système d'exploitation et sur le processeur de la machine d'où provient le fichier « **.etl** ». Pour récupérer la date et l'heure au moment de la capture, il convient de se rendre sous le nouvel onglet nommé « **System Configuration** ».

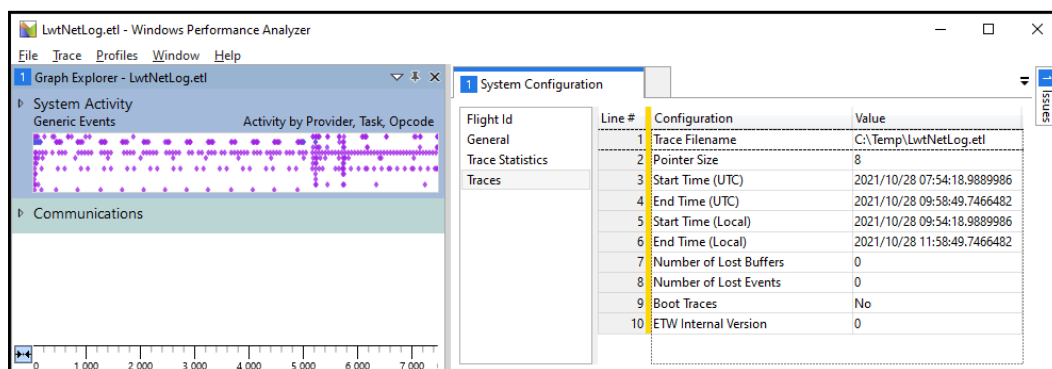


FIGURE 38 : INFORMATIONS SUR LE FICHIER ETL DEPUIS WINDOWS PERFORMANCE ANALYZER

⁸ Via le lien suivant : « <https://go.microsoft.com/fwlink/p/?LinkID=845298> ».

4) L'analyse des traces de suivi d'événements se réalise en sélectionnant une catégorie dans le bandeau de gauche comme « **System Activity** » ou « **Communications** » -> faire un clic-droit sur l'une des sous catégories comme « **Generic Events** » -> « **Add graph to Analysis View** ». Un nouveau graphique apparaît. Il est possible d'afficher l'intégralité des graphiques d'une catégorie en sélectionnant « **Add all System Activity graphs to Analysis View** ».

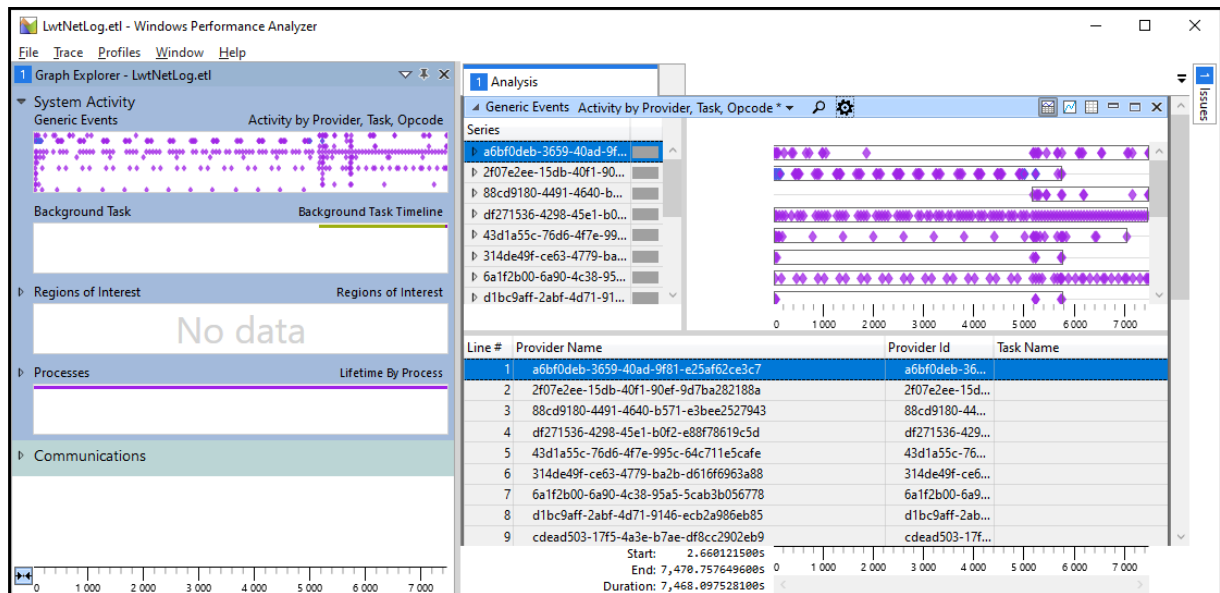


FIGURE 39 : VUE GRAPHIQUE DE LA SOUS-CATÉGORIE GENERIC EVENTS DE SYSTEM ACTIVITY

5) Rechercher dans les graphiques les informations intéressantes à l'enquête.

Afin de montrer la puissance de cet outil, le fichier « **BootCKC.etl** » est utilisé dans l'exemple suivant :

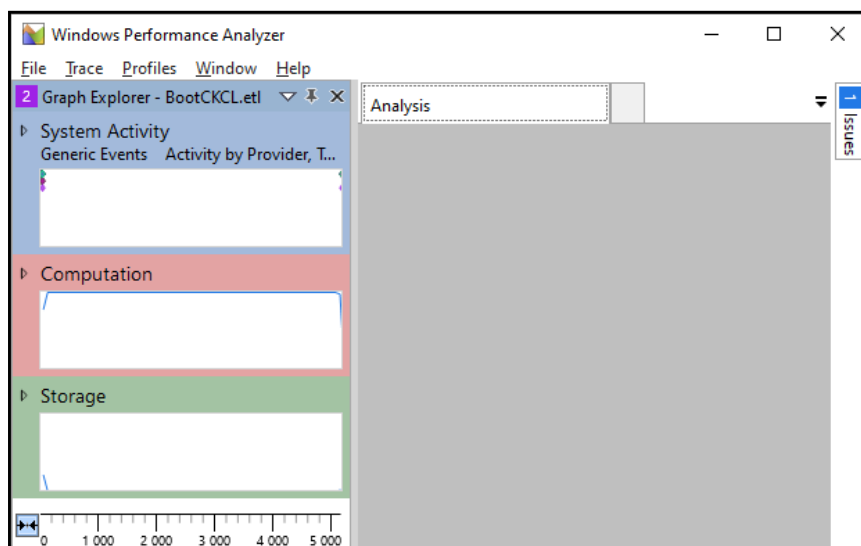


FIGURE 40 : EXPLOITATION DU FICHIER BOOTCKCL.ETL SOUS WINDOWS PERFORMANCE ANALYZER

Ce fichier « **.etl** » fournit des informations, notamment sur les traces d'exécution de fichiers DLL et système, en lien avec les exécutables, les processus, les threads et autres, chargés au

moment du démarrage du système, et ce jusqu'à l'enregistrement des événements soit arrêté.

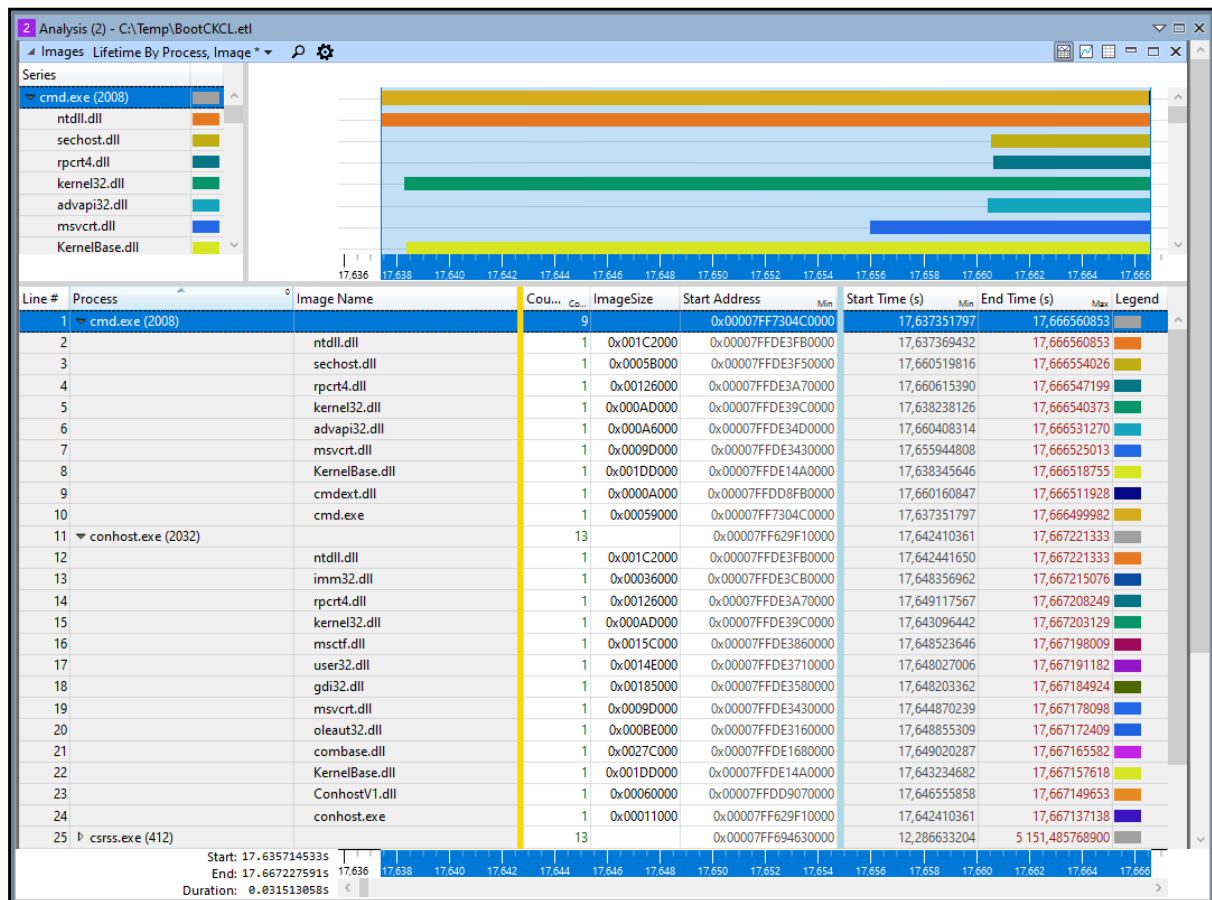


FIGURE 41 : GRAPHIQUE DE LA SOUS-CATÉGORIE IMAGES DE LA CATÉGORIE SYSTEM ACTIVITY

Il est facile de retrouver les processus exécutés au démarrage du système d'exploitation et des sessions utilisateurs ainsi que les fichiers liés aux processus tels que les fichiers DLL et système. Le fichier est intéressant car il fournit des informations qu'il est difficile d'obtenir sans une collecte de la mémoire d'un ordinateur ou serveur, telles que la durée d'exécution des processus, les adresses mémoires, et bien d'autres informations.

Il est aussi possible d'obtenir des graphiques sur l'utilisation du CPU, de l'utilisation du disque, etc. La figure 42 ci-dessous est un exemple de graphique ayant pour but de montrer le taux d'utilisation du disque de stockage en pourcentage en fonction du temps en seconde.

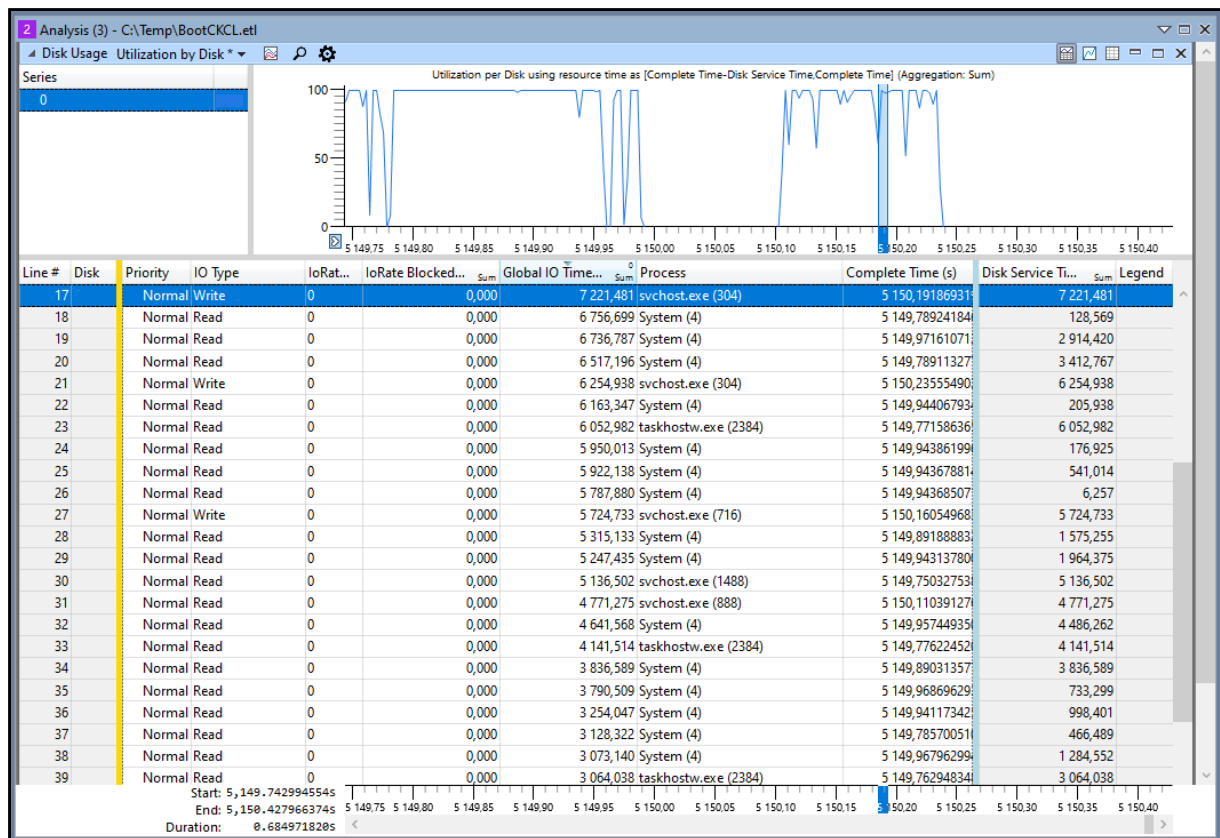


FIGURE 42 : GRAPHIQUE DE LA SOUS-CATÉGORIE DISK USAGE DE LA CATÉGORIE STORAGE

4.1.5 PerfView

« **PerfView.exe** » est un exécutable graphique pour analyser les fichiers « **.etl** ».

Le Guide de l'utilisateur de « **PerfView** » fait partie de l'application elle-même. Il suffit de cliquer sur le lien « **Users Guide** » depuis le menu « **Help** ».

Il prend en compte les fichiers « **.etl** » de façon unitaire. Il n'est pas possible de lui fournir plusieurs fichiers au format « **.etl** » à traiter en une seule fois. Il faudra les importer les uns à la suite des autres.

Exemple d'exploitation de fichiers « **.etl** » grâce à l'exécutable :

1) Récupérer l'exécutable depuis le site « <https://github.com/microsoft/perfview> », « **Version 1.9.0.0** ».

SHA-1 de l'exécutable : « **43725a34e624deac259259a8383dd2da0c74cebd** »

2) Sélectionner un fichier « **.etl** » comme « **ShutdownCKCL.etl** » en parcourant l'arborescence, grâce à l'icône du dossier tout en bas suivi de deux petits points.

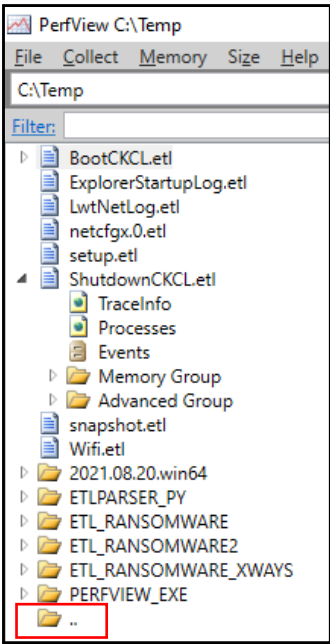


FIGURE 43 : EXEMPLE DE RECHERCHE DE FICHIERS ETL

Le fichier « *TraceInfo* » permet d’obtenir des informations sur la capture de trace.

Machine Name	
Operating System	
OS Build Number	
UTC offset where data was collected	2,00
UTC offset where PerfView is running	2,00
Delta of Local and Collection Time	0,00
OS Boot Time	10/28/2021 09:54:18.486
Trace Start Time	10/28/2021 09:54:18.988
Trace End Time	10/28/2021 11:58:50.059
Trace Duration (Sec)	7 471,1
CPU Frequency (Mhz)	1 800
Number Of Processors	4
Memory Size (Meg)	0
Pointer Size	8
Sample Profile Interval (MSec)	1,00
Total Events	8 018
Lost Events	0
ETL File Size (MB)	1,8
No data collection log file found	

FIGURE 44 : TRACEINFO DU FICHIER SHUTDOWNCKCL.ETL

Le fichier « *Processes* » permet d’obtenir des informations sur l’exécution des processus en cours sur le système d’exploitation lors de la capture de la trace.

Processes for ShutdownCKCL.etl in Temp (C:\Temp\ShutdownCKCL.etl)

Back Forward Click in Window, Ctrl-F for find

Process Summary

- [View Process Data in Excel](#)
- [View Process Modules in Excel](#)

Processes that did not live for the entire trace.

Name	ID	Parent ID	Bitness	CPU MSec	Ave Procs Used	Duration MSec	Start MSec	Exit Code	Command
SearchFilterHost	3864	2556	64	0	0,000	2 775,138	7 456 894,277	0x0	"C:\Windows\system32\SearchFilterHost.exe"
SearchProtocolHost	3064	2556	64	0	0,000	2 806,971	7 456 866,974	0x0	"C:\Windows\system32\SearchProtocolHost.exe" Global\UsGthrCtrlPipeMssGthrPipe7 1 -21- Search" "Mozilla/4.0 (compatible; MSIE 6.0; ; "C:\ProgramData\Microsoft\Search\Data\Tem
rundll32	3640	716	64	0	0,000	14,065	7 455 486,871	0xc000026b	C:\Windows\System32\rundll32.exe shell32.d F6A8-11CF-A442-00A0C90A8F39} -Embedc
SearchIndexer	2556	612	64	0	0,000	7 461 097,046	0,000	0x0	C:\Windows\system32\SearchIndexer.exe /Em
taskhostw	668	888	64	0	0,000	7 456 556,188	0,000	0x40010004	taskhostw.exe {222A245B-E637-4AE9-A93F-
sihost	3444	888	64	0	0,000	7 455 563,565	0,000	0x40010004	sihost.exe
svchost	3252	612	64	0	0,000	7 459 648,285	0,000	0x0	C:\Windows\System32\svchost.exe -k wsappx
MsMpEng	1776	612	64	0	0,000	7 460 018,371	0,000	0x0	"C:\Program Files\Windows Defender\MsMpE
vmtoolsd	1748	612	64	0	0,000	7 459 136,770	0,000	0x0	"C:\Program Files\VMware\VMware Tools\vr

FIGURE 45 : PROCESSES DU FICHIER SHUTDOWNCKCL.ETL

Ce fichier permet également d'exporter ce tableau sous Excel, ainsi qu'un autre tableau contenant, en plus, les fichiers utilisés par les processus, tels que les fichiers DLL et système.

La fonction rechercher (ctrl + f) fonctionne depuis le fichier « **Processes** ».

Name	ID	Parent ID	Bitness	CPU MSec	Ave Procs Used	Duration MSec	Start MSec	Exit Code	Command
taskhostw	4644	888	64	0	0,000	7 456 556,188	0,000	0x40010004	taskhostw.exe {222A245B-E637-4AE9-A93F-
firefox	4460	1384	64	0	0,000	7 456 556,188	0,000	0x0	"C:\Program Files\Mozilla Firefox\firefox.exe" -contentproc MapSize 241171 -jsInit 1120 286204 -parentBuild tab
ShellExperienceHost	4028	716	64	0	0,000	7 456 556,188	0,000	0x0	C:\Windows\SystemApps\ShellExperienceHost_cw5n1h2t ws\System32\RuntimeBroker.exe -Embedding
RuntimeBroker	3744	716	64	0	0,000	7 456 556,188	0,000	0x0	C:\Windows\System32\RuntimeBroker.exe -Embedding
explorer	1388	1660	64	0	0,000	7 456 556,188	0,000	0x0	C:\Windows\Explorer.EXE
firefox	1384	508	64	0	0,000	7 456 556,188	0,000	0x0	"C:\Program Files\Mozilla Firefox\firefox.exe"
vmtoolsd	320	1388	64	0	0,000	7 452 984,084	0,000	0x0	"C:\Program Files\VMware\VMware Tools\vmtoolsd.exe"
SearchUI	3040	716	64	0	0,000	7 455 125,921	0,000	0x1	"C:\Windows\SystemApps\Microsoft.Windows.Cortana_cw OneDrive
OneDrive	284	1388	32	0	0,000	7 453 190,110	0,000	0x40010004	"C:\Users\Administrateur\AppData\Local\Microsoft\OneDr
svchost	1488	612	64	0	0,000	7 459 299,469	0,000	0x0	C:\Windows\System32\svchost.exe -k utcsvc
svchost	1428	612	64	0	0,000	7 459 654,108	0,000	0x0	C:\Windows\system32\svchost.exe -k NetworkServiceNetw
autodiscover	4996	1388	32	0	0,000	7 454 030,924	-1 124,215	0x40010004	"C:\Users\Administrateur\Downloads\autodiscover.exe"
svchost	3576	612	64	0	0,000	7 454 023,781	-1 124,215	0x40010004	C:\Windows\system32\svchost.exe -k UnistackSvcGroup

FIGURE 46 : FONCTION RECHERCHE DEPUIS LE FICHIER PROCESSES

Le conteneur « **Events** » permet d'examiner l'intégralité des événements enregistrés lors de la capture de la trace.

Sélectionner l'intégralité des « **Event Types** » dans le bandeau de gauche et appuyer sur la touche « entrée » pour afficher l'intégralité des événements (voir figures 47 et 48 ci-dessous).

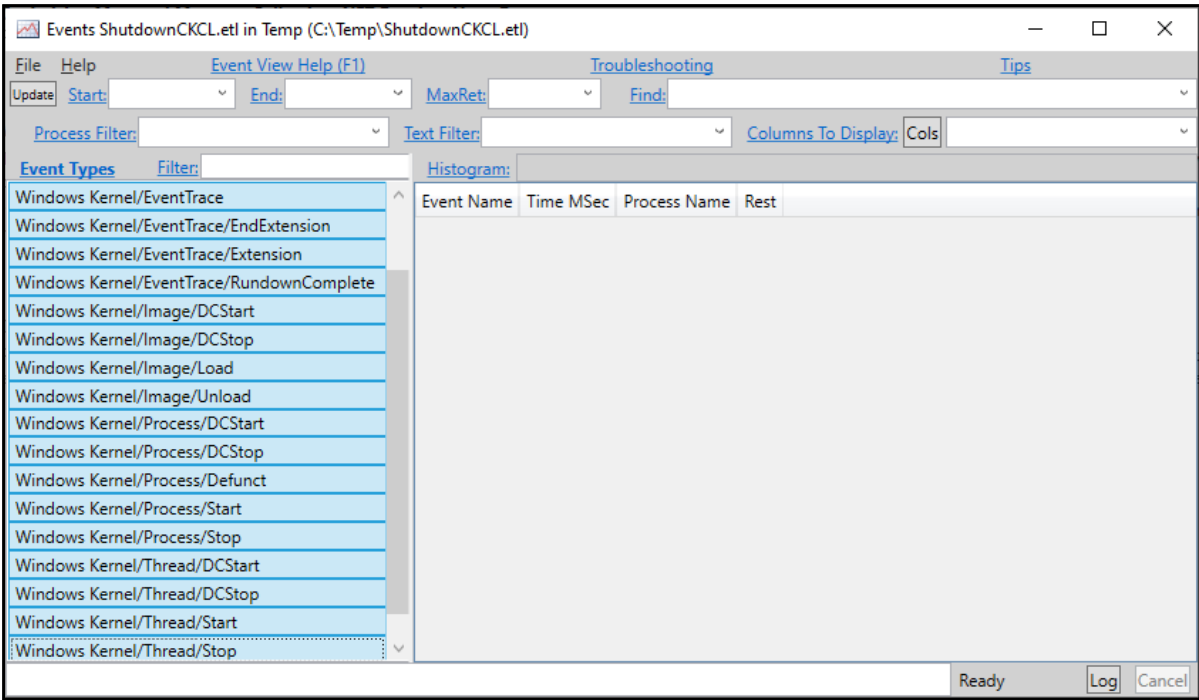


FIGURE 47 : SÉLECTION DE L'ENSEMBLE DES EVENT TYPES

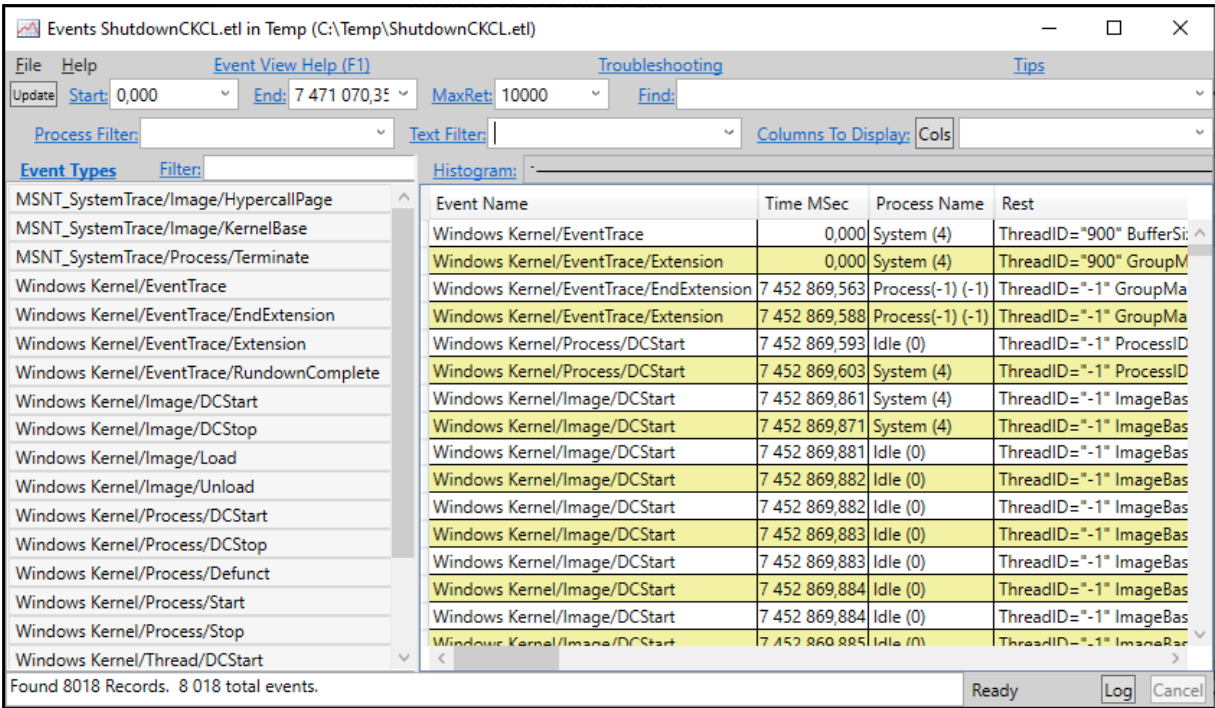


FIGURE 48 : AFFICHAGE DE L'ENSEMBLE DES ÉVÉNEMENTS

Il est possible d'effectuer une recherche par mot clé grâce au filtre « **Text Filter** » comme le montre la figure 49 ci-dessous.

The screenshot shows the Windows Event Viewer window titled 'Events ShutdownCKCL.etl in Temp (C:\Temp\ShutdownCKCL.etl)'. The 'Text Filter' is set to 'autodiscover'. The 'Event Types' list on the left includes various system and application events. The main pane displays a table of events filtered by the 'autodiscover' process.

Event Name	Time MSec	Process Name	Rest
Windows Kernel/Thread/Stop	7 452 885,988	autodiscover (4996)	ThreadID="2 940" St
Windows Kernel/Thread/Stop	7 452 886,101	autodiscover (4996)	ThreadID="5 108" St
Windows Kernel/Thread/Stop	7 452 886,106	autodiscover (4996)	ThreadID="4 572" St
Windows Kernel/Thread/Stop	7 452 886,430	autodiscover (4996)	ThreadID="3 588" St
Windows Kernel/Thread/Stop	7 452 886,534	autodiscover (4996)	ThreadID="4 176" St
Windows Kernel/Thread/Stop	7 452 886,539	autodiscover (4996)	ThreadID="4 620" St
Windows Kernel/Thread/Stop	7 452 887,915	autodiscover (4996)	ThreadID="3 664" St
Windows Kernel/Thread/Stop	7 452 887,925	autodiscover (4996)	ThreadID="4 684" St
Windows Kernel/Image/Unload	7 452 903,652	autodiscover (4996)	ThreadID="-1" Imag
Windows Kernel/Image/Unload	7 452 903,686	autodiscover (4996)	ThreadID="-1" Imag
Windows Kernel/Image/Unload	7 452 903,705	autodiscover (4996)	ThreadID="-1" Imag
Windows Kernel/Image/Unload	7 452 903,715	autodiscover (4996)	ThreadID="-1" Imag
Windows Kernel/Image/Unload	7 452 903,738	autodiscover (4996)	ThreadID="-1" Imag
Windows Kernel/Image/Unload	7 452 903,753	autodiscover (4996)	ThreadID="-1" Imag
Windows Kernel/Image/Unload	7 452 903,770	autodiscover (4996)	ThreadID="-1" Imag
Windows Kernel/Image/Unload	7 452 903,791	autodiscover (4996)	ThreadID="-1" Imag

Found 74 Records. 74 total events.

FIGURE 49 : FILTRE SUR AUTODISCOVER

Un référencement des processus est réalisé automatiquement et il est possible de filtrer sur un processus en particulier comme le montre la figure 50 ci-dessous.

The screenshot shows the Windows Event Viewer window titled 'Events ShutdownCKCL.etl in Temp (C:\Temp\ShutdownCKCL.etl)'. The 'Text Filter' is set to 'autodiscover'. The 'Event Types' list on the left includes various system and application events. The main pane displays a table of events filtered by the 'autodiscover' process.

Event Name	Time MSec	Process Name	Rest
Windows Kernel/Image/Unload	7 452 888,036	svchost (3576)	ThreadID="-1" ImageBase="0x7ff6e2ce0000" Imag
Windows Kernel/Image/Unload	7 452 888,062	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdbed30000" Imag
Windows Kernel/Image/Unload	7 452 888,076	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdbed60000" Imag
Windows Kernel/Image/Unload	7 452 888,134	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdbed80000" Imag
Windows Kernel/Image/Unload	7 452 888,170	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdbedc0000" Imag
Windows Kernel/Image/Unload	7 452 888,184	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdbee00000" Imag
Windows Kernel/Image/Unload	7 452 888,198	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdbee70000" Imag
Windows Kernel/Image/Unload	7 452 888,212	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdbec00000" Imag
Windows Kernel/Image/Unload	7 452 888,226	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdbf040000" Imag
Windows Kernel/Image/Unload	7 452 888,255	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdbf060000" Imag
Windows Kernel/Image/Unload	7 452 888,316	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdc6890000" Imag
Windows Kernel/Image/Unload	7 452 888,344	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdc0c60000" Imag
Windows Kernel/Image/Unload	7 452 898,229	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdd14e0000" Imag
Windows Kernel/Image/Unload	7 452 898,258	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdd2730000" Imag
Windows Kernel/Image/Unload	7 452 898,299	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdd35a0000" Imag
Windows Kernel/Image/Unload	7 452 898,312	svchost (3576)	ThreadID="-1" ImageBase="0x7ffdd35d0000" Imag

Found 2027 Records. 2 027 total events.

FIGURE 50 : FILTRE AUTOMATIQUE SUR UN PROCESSUS PARTICULIER RÉFÉRENCÉ

Il est très facile de retrouver les processus exécutés au démarrage du système d'exploitation et des sessions utilisateurs, ainsi que les fichiers liés aux processus tels que les fichiers DLL et système. Le fichier « **ShutdownCKCL.etl** » est intéressant car il fournit des informations qu'il est difficile d'obtenir sans une collecte de la mémoire d'un ordinateur ou serveur. La durée d'exécution des processus, les adresses mémoires, et bien d'autres informations.

4.1.6 Tela64

« **Tela64.exe** » est un outil de ligne de commande qui analyse les fichiers « **.etl** ». Cet outil payant a été développé par la société « **TZWorks** ».

Bien qu'il existe divers outils de Microsoft pour analyser ces fichiers journaux, le but de « **Tela64** » est de fournir un outil indépendant de l'API Windows qui peut analyser ces journaux rapidement.

L'autre objectif est de pouvoir analyser bon nombre de ces journaux par lots, tout en restituant leur contenu aux formats « **.csv** » (par défaut) ou « **.txt** ».

Exemple d'exploitation d'un fichier « **.etl** » grâce à l'exécutable :

1) Récupérer l'exécutable depuis le site « https://tzworks.com/prototype_page.php?proto_id=40 », « **Version 0.24** ».

SHA-1 de l'exécutable : « **13103cb5f1923529a77c91d299296af3ffaeb770** »

2) Placer l'exécutable sous l'emplacement « **C:\Temp** » ainsi que le fichier « **.etl** » à parser.

3) Créer le dossier de sortie dans lequel le fichier « **.csv** » sera enregistré.

4) Depuis une « **Invite de commandes** », exécutée en tant qu'administrateur, renseigner les commandes suivantes :

```
C:\WINDOWS\system32>cd %SystemDrive%\temp
```

```
C:\temp>tela64.exe -log .\ShutdownCKCL.etl > .\Files_csv\ShutdownCKCL_etl.csv
```

Explications des commandes :

cd %SystemDrive%\temp

Indique de se rendre dans le dossier « **Temp** » situé à la racine du disque.

tela64.exe -log .\ShutdownCKCL.etl > .\Files_csv\ShutdownCKCL_etl.csv

Indique d'exécuter le binaire « **tela64.exe** », « **-log .\ShutdownCKCL.etl** » indique l'emplacement et le nom du fichier « **.etl** » à traiter, « **> .\Files_csv\ShutdownCKCL_etl.csv** » redirige le fichier de sortie après traitement à tel emplacement et avec tel nom.



FIGURE 51 : EXEMPLE D'EXÉCUTION DU PROGRAMME "TELA64"

Une fois le traitement terminé, l'invite de commandes attend de nouvelles instructions.

5) Analyse du fichier de sortie.

Enregistrement automatique

ShutDownCKCL_etl.csv

Fichier

Accueil

Insertion

Mise en page

Formules

Données

Révision

Affichage

Aide

Rechercher (Alt=Q)

nicolas carpentier

Commentaires

Partager

Obtenir des données

A partir d'un fichier Texte/CSV

A partir de données d'une base de données

Requêtes et transformations des données

Sources récentes

Données existantes

Actualiser tout

Requêtes et connexions

Visualiser les données

Types de données

Trier

Filtrer

Avancé

Requêtes et connexions

Conversion

Supprimer les doublons

Validation des données

Outils de données

Consolider

Représentage interactif

Analyse séquentielle

Feuille de prévision

Prévision

Gérer le modèle de données

Plan

Groupier

Dissocier

Sous-total

B1

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

tela - full ver: 0.24

License #16785b786023340 is authenticated for business use and registered to French GCA / CAUD (Single Facility)

run time: 11/22/2021 08:26:19 [UTC]

cmdline: tela64.exe -log .\ShutDownCKCL.etl

file: .\ShutDownCKCL.etl

orig source path: C:\Windows\system32\LogFiles\ShutDownCKCL.etl

created [utc or wtc]

provider guid

provider name

session name

activity id

event id

process id

thread id

channel

level

opcode

task id

keywords

processor time

processor name

10/28/2021 07:54:18.98895650

header info only

0x00000004

0x00000004

0x00000004

0x000000000000005e

d

10/28/2021 09:58:31.85864560

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000b4

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85864620

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000b8

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85864730

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000bc

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85864790

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000c0

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85864900

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000c8

0

0

0

0

0x0000000000000000

0x000000000000011b

3 d

10/28/2021 09:58:31.85865020

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000d0

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85865070

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000dc

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85865190

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000e0

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85865240

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000e4

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85865300

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000e8

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85865360

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000ec

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85865410

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000f0

0

0

0

0

0x0000000000000000

0x0000000000000044

3 d

10/28/2021 09:58:31.85865530

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000f4

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85865580

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000f8

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85865640

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x000000fc

0

0

0

0

0x0000000000000000

0x0000000000000059

3 d

10/28/2021 09:58:31.85865700

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x00000100

0

0

0

0

0x0000000000000000

0x0000000000000066

3 d

10/28/2021 09:58:31.85865750

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x00000104

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85865810

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x00000108

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85865870

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x00000110

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85865930

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x00000114

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85866040

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x00000118

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85866100

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x0000011c

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85866150

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x00000120

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85866200

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x00000124

0

0

0

0

0x0000000000000000

0x0000000000000015

3 d

10/28/2021 09:58:31.85866320

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x00000128

0

0

0

0

0x0000000000000000

0x000000000000003b

3 d

10/28/2021 09:58:31.85866380

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x0000012c

0

0

0

0

0x0000000000000000

0x000000000000003f

3 d

10/28/2021 09:58:31.85866440

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x00000130

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

10/28/2021 09:58:31.85866500

3d6fabd1-fe05-11d0-9dda-00c04f7ba7c

Kernel-Thread-Trace

Circular Kernel Context Logger

00000000-0000-0000-0000-000000000000

0x000503

0x00000004

0x00000134

0

0

0

0

0x0000000000000000

0x0000000000000000

3 d

ShutDownCKCL.etl

Prêt

FIGURE 52 : ANALYSE DU FICHIER DE SORTIE SHUTDOWNCKCL_ETL.CSV

L'utilisation de ce binaire n'est pas pertinent car la sortie est très peu exploitable. Les seuls éléments qui ressortent sont les « *event_id* », « *process_id* », « *thread_id* » ainsi que les « *processort_time* ». Toutes ces valeurs étant données en hexadécimal. Le test a été effectué sur plusieurs fichiers « *.etl* » et cette conclusion reste inchangée.

4.2 Avantages et inconvénients des applications

Nom de l'application	Avantages	Inconvénients
Observateur d'événements	<ul style="list-style-type: none"> - Natif à Windows - Lecture de l'ensemble des fichiers « .etl » - Simple d'utilisation - Possibilité d'exporter aux formats « .csv » et « .xml » - Recherches simples depuis les formats « .csv » et « .xml » 	<ul style="list-style-type: none"> - Traitement incomplet - Importation des fichiers « .etl » de façon unitaire - Recherches/filtres difficiles depuis l'observateur d'événements
ETLParser	<ul style="list-style-type: none"> - Lecture de l'ensemble des fichiers « .etl » - Simple d'utilisation - Importation des fichiers « .etl » de façon unitaire, par sélection de plusieurs fichiers ou dans leur intégralité - Export aux formats « .csv » et « .sqlite » - Recherches simples depuis les formats « .csv » et « .sqlite » 	<ul style="list-style-type: none"> - Traitement incomplet - Pas de récursivité pour l'importation des fichiers « .etl ». Ils doivent tous être dans un même emplacement avant traitement. Problème pour plusieurs fichiers « .etl » qui portent le même nom
Microsoft Message Analyzer	<ul style="list-style-type: none"> - Application développée par Microsoft - Utile pour l'exploitation des traces de suivi d'événements réseau - Possibilité d'effectuer des recherches simples grâce à des requêtes comme sous Wireshark - Possibilité d'exporter aux formats « .csv » et « .cap » 	<ul style="list-style-type: none"> - Plus officiellement disponible en téléchargement - Importation des fichiers « .etl » de façon unitaire - Utile principalement pour les fichiers « .etl » concernant les traces de suivi d'événements réseau
Windows Performance Analyzer	<ul style="list-style-type: none"> - Application développée par Microsoft - Lecture de l'ensemble des fichiers « .etl » 	<ul style="list-style-type: none"> - Importation des fichiers « .etl » de façon unitaire - Prise en main complexe

Nom de l'application	Avantages	Inconvénients
	<ul style="list-style-type: none"> - Présence importante d'informations 	
PerfView	<ul style="list-style-type: none"> - Application développée par Microsoft - Lecture de l'ensemble des fichiers « .etl » - Simple d'utilisation - Recherche simple (ctrl+f) ou depuis les filtres d'affichage - Possibilité d'exporter au format « .xls » - Informations/statistiques sur les traces de suivi d'événements - Présence importante d'informations 	<ul style="list-style-type: none"> - Importation des fichiers « .etl » de façon unitaire
tela64	<ul style="list-style-type: none"> - Importation des fichiers « .etl » de façon unitaire, par sélection de plusieurs fichiers ou dans leur intégralité - Simple d'utilisation - Possibilité d'exporter aux formats « .csv » et « .txt » 	<ul style="list-style-type: none"> - Traitement très incomplet - Soumis à licence

TABLEAU 2 : AVANTAGES ET INCONVÉNIENTS DES APPLICATIONS

Les applications à privilégier sont donc « **PerfView** », « **Windows Performance Analyzer** » et « **ETLParser** ».

CONCLUSION

Les analyses forensiques conventionnelles actuelles d'environnement Microsoft ne prennent pas en compte les fichiers « **.etl** ». Pourtant ces fichiers sont une mine d'informations très riche car ils contiennent les actions effectuées par le noyau ainsi que certaines actions utilisateurs.

Une connaissance plus approfondie de ces fichiers et de leurs structures permettrait d'obtenir d'avantage d'informations à exploiter lors de futures investigations et d'acquérir des réponses plus précises à nos interrogations.

A ce jour, les documentations sur ce sujet sont quasi inexistantes et peu d'applications permettent d'ouvrir et d'analyser ces fichiers, ce qui rend les analyses (par ailleurs très enrichissantes) longues et complexes.

En complément et en l'absence des journaux d'événements, des prefetchs, des captures réseaux ou mémoires, ces fichiers sont aujourd'hui indispensables lors des analyses forensiques.

Un autre atout de ces fichiers est qu'ils peuvent être générés et analysés en temps réel par des analystes ou administrateurs et non exclusivement par des développeurs grâce à l'application de Microsoft « Windows Performance Recorder ». Ce qui permet de rechercher rapidement des traces de malveillances. Grâce à ces artefacts, des nouvelles pistes d'analyses et de détections basées sur les fichiers « **.etl** » voient actuellement le jour.

Les applications privilégiées au traitement des fichiers « **.etl** » sont : « PerfView », « Windows Performance Analyzer » et « ETLParser ».

Les prochains travaux consisteront à détecter les nouveaux fichiers « **.etl** » et les éléments de preuve qu'ils contiennent avec l'apparition du nouveau système d'exploitation Microsoft Windows 11. Il serait également possible d'élaborer une procédure de collecte de masse sur tout un parc informatique Microsoft. Un autre axe de recherche porte sur la définition d'un outil approprié au traitement de l'ensemble des fichiers « **.etl** ».

INDEX

B

Base de données, 9

BootCKCL.etl, 19

C

Collecte en live, 22

Collecte post-mortem, 28

Consommateurs, 10

Consumers, 10

Contrôleurs, 10

Controllers, 10

CortanaTrace1.etl, 20

E

Energy-ntkl.etl, 20

ETLParser, 37

ETW, 10

Event Tracing for Windows (ETW), 9

EventLog, 9

ExplorerStartupLog.etl, 20

F

Fichier plat, 9

Format « .etl », 14

Format « .evt », 13

Format « .evtx », 14

Format « .txt », 12

Fournisseurs, 10

J

Journal de suivi d'événements, 16

Journalisation, 8

L

L'Observateur d'événements, 34

LwNetLog.etl, 21

M

Microsoft Message Analyzer (MMA), 39

P

PerfView, 44

Providers, 10

S

ShutdownCKCL.etl, 19

Syslog, 9

T

Tela64, 49

W

Wifi.etl, 20

Windows Performance Analyzer (WPA), 41

WindowsUpdate.date.hour...etl, 21

X

X-ways, 28

LISTE DES FIGURES

Figure 1 : https://docs.microsoft.com/en-us/windows-hardware/test/weg/instrumenting-your-code-with-etw	10
Figure 2 : Exemple de fichiers d'événements au format texte sous Microsoft Windows 95	11
Figure 3 : Exemple d'événements depuis l'Observateur d'événements Microsoft Windows 95	12
Figure 4 : Exemple de journaux depuis l'Observateur d'événements Microsoft Windows XP	12
Figure 5 : Exemple de journaux d'événements depuis l'Observateur d'événements Microsoft Windows 10	13
Figure 6 : Modification de la politique de sécurité de l'exécution des scripts.....	21
Figure 7 : Obtention des droits "NT AUTHORITY\System" et exécution de Powershell	22
Figure 8 : Exécution de PowerShell et vérification des droits "NT AUTHORITY\System" ...	22
Figure 9 : Exécution du script de la collecte de fichiers ETL.....	23
Figure 10 : Fin de la collecte, PowerShell en attente d'instructions.....	23
Figure 11 : Réactivation par défaut de la politique de sécurité de l'exécution des scripts	24
Figure 12 : Exemple de dossier créé à la racine du système	24
Figure 13 : Exemple de dossiers et fichier créés pendant la collecte	25
Figure 14 : Autre exemple de dossiers et fichier créés pendant la collecte	25
Figure 15 : Exemple du fichier texte après collecte	25
Figure 16 : Exemple de fichiers et dossier créés	25
Figure 17 : Cases à cocher - Refine Volume Snapshot	27
Figure 18 : Sélection du type de fichier à filtrer	27
Figure 19 : Récursivité de la recherche des fichiers ETL.....	27
Figure 20 : Résultat du filtre sur les fichiers de type ETL	28
Figure 21 : Filtre sur le nom de fichiers ETL.....	28

Figure 22 : Résultat du filtre sur les noms de fichiers ETL.....	29
Figure 23 : Recover/copy sur la sélection des fichiers ETL.....	29
Figure 24 : Résultat de l'action de copie par X-ways.....	30
Figure 25 : Résultat de la copie des fichiers ETL depuis l'explorateur Windows	30
Figure 26 : Observateur d'événements Microsoft Windows 10	32
Figure 27 : Sélection d'un fichier ETL à analyser	32
Figure 28 : Conversion au format EVTX du fichier ETL	33
Figure 29 : Nommage et description du fichier etl importé.....	33
Figure 30 : Visualisation et exploitation des événements	33
Figure 31 : Exemple prérequis ETLParser	34
Figure 32 : Exécution de l'exécutable ETLParser	35
Figure 33 : Fin du traitement d'ETLParser	35
Figure 34 : Résultat du traitement ETLParser.....	36
Figure 35 : Recherche texte sur une adresse IP depuis DB BROWSER for SQLite.....	36
Figure 36 : Exploitation d'un fichier ETL sous MMA.....	37
Figure 37 : Capture d'écran d'installation de Windows Software Development Kit	38
Figure 38 : Informations sur le fichier ETL depuis Windows Performance Analyzer	38
Figure 39 : Vue graphique de la sous-catégorie Generic Events de System Activity	39
Figure 40 : Exploitation du fichier BootCKCL.etl sous Windows Performance Analyzer	39
Figure 41 : Graphique de la sous-catégorie Images de la catégorie System Activity.....	40
Figure 42 : Graphique de la sous-catégorie Disk Usage de la catégorie Storage	41
Figure 43 : Exemple de recherche de fichiers ETL.....	42
Figure 44 : TraceInfo du fichier ShutdownCKCL.etl	42
Figure 45 : Processes du fichier ShutdownCKCL.etl	43

Figure 46 : Fonction recherche depuis le fichier Processes	43
Figure 47 : Sélection de l'ensemble des Event Types	44
Figure 48 : Affichage de l'ensemble des événements	44
Figure 49 : Filtre sur autodiscover	45
Figure 50 : Filtre automatique sur un processus particulier référencé	45
Figure 51 : Exemple d'exécution du programme "tela64"	46
Figure 52 : Analyse du fichier de sortie ShutdownCKCL_etl.csv	47

LISTE DES TABLEAUX

Tableau 1 : Emplacement des fichiers ETL et leur nom 17

Tableau 2 : Avantages et inconvénients des applications..... 49

BIBLIOGRAPHIE

- gc-nibrahim. (2018, 06 08). *GitHub - forensiclunch/ETLParser: Binary commandline executable to parse ETL files*. Récupéré sur GitHub: Where the world builds software · GitHub: <https://github.com/forensiclunch/ETLParser>
- Ibrahim, N. (2018, 06 07). *ETW Event Tracing for Windows and ETL Files - Hacking Exposed Computer Forensics Blog*. Récupéré sur Hacking Exposed Computer Forensics Blog: <http://www.hecfblog.com/2018/06/etw-event-tracing-for-windows-and-etl.html>
- jaimeo, M. /. (2021, 09 15). *Fichiers journaux de Windows Update - Windows Deployment / Microsoft Docs*. Récupéré sur Microsoft – Cloud, ordinateurs, applications et jeux: <https://docs.microsoft.com/fr-fr/windows/deployment/update/windows-update-logs>
- Les utilisateurs AUTORITE NT - malekal.com*. (2017, 03 13). Récupéré sur malekal.com - site informatique Windows11, Windows 10, Virus et Linux: <https://www.malekal.com/utilisateur-autorite-nt/>
- Microsoft. (2021, 07 01). *About Event Tracing - Win32 apps | Microsoft Docs*. Récupéré sur Microsoft – Cloud, ordinateurs, applications et jeux: <https://docs.microsoft.com/en-us/windows/win32/etw/about-event-tracing>
- Microsoft. (2021, 09 15). *Suivi d'événements - Win32 apps | Microsoft Docs*. Récupéré sur Microsoft – Cloud, ordinateurs, applications et jeux: <https://docs.microsoft.com/fr-fr/windows/win32/etw/event-tracing-portal>
- Microsoft. (s.d.). *Download Microsoft Message Analyzer from Official Microsoft Download Center*. Récupéré sur WayBackMachine: <https://web.archive.org/web/20191104120853/https://www.microsoft.com/en-us/download/confirmation.aspx?id=44226>
- Roccia, T. (2019, 11 28). *Collectez les informations de base pour l'investigation - Menez une investigation d'incident numérique forensic - OpenClassrooms*. Récupéré sur Formations en ligne et cours en accès libre - OpenClassrooms: <https://openclassrooms.com/fr/courses/1750151-menez-une-investigation-d-incident-numerique-forensic/6472896-collectez-les-informations-de-base-pour-l-investigation>
- TZWorks. (2021). *Trace Event Log and Analysis*. Récupéré sur TZWorks LLC (www.tzworks.com) Homepage: https://tzworks.com/prototype_page.php?proto_id=40

ANNEXES

Annexe 1 – Script collecte en live

Nom : « **Collect_ETL.ps1** »⁹

SHA-1 : « **06950FAE3E4DD31899DFDE93CFDD653FB86684BA** »

```
# Script pour lister l'ensemble des fichiers ".etl" dans un fichier texte nommé "list_file_etl.txt" et copier les
fichiers "*.etl" dans un dossier nommé "ETL_$ComputerName" en conversant l'arborescence récursive des
dossiers dans lesquels les fichiers ".etl" ont été trouvés.
# Nicolas CARPENTIER
# avril 2021
# version 1
# testé sur version powershell 5 et environnement windows 10 familial et professionnel
# Exécution de Powershell en tant qu'administrateur

# Récupération de la lettre du lecteur où l'OS est installé en tant que variable et déplacement à la racine de
cette même lettre
$LetterOS= (Get-WmiObject Win32_OperatingSystem).SystemDrive
$SourceDir= "$LetterOS\"
cd $SourceDir

# Dénomination du dossier de collecte
$ComputerName= hostname
$TargetDir= "ETL_$ComputerName"

# Création du dossier nommé "ETL_$ComputerName" afin de recevoir les éléments collectés
New-Item -Path $SourceDir -Name $TargetDir -ItemType directory

# Création du fichier nommé "List_files_etl.txt" sous "$SourceDir\ETL_$ComputerName" qui contiendra la
liste des fichiers ".etl" trouvés et leurs emplacements avant copie
Get-ChildItem $SourceDir -Filter *.etl -Recurse | Out-File $SourceDir$TargetDir\List_files_etl.txt

# Copie des fichiers ".etl" sous "$SourceDir\ETL_$ComputerName\" en conservant l'arborescence récursive
des dossiers dans lesquels les fichiers ".etl" ont été trouvés
Get-ChildItem -Path $SourceDir -Recurse -Include *.etl | `
    foreach{
        $targetFile = "$TargetDir\" + $_.FullName.SubString($SourceDir.Length);
        New-Item -ItemType File -Path $targetFile -Force;
        Copy-Item $_.FullName -destination $targetFile
    }
}
```

⁹ Disponible depuis le site Github : « https://github.com/Nyk0la5/Collect_ETL ».

Annexe 2 – Exemple de fichier « list_file_elt.txt »

Répertoire : C:\Program Files\UNP\SystemLogs			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	14/01/2021 15:54	327680	UpdateNotificationPipeline.001.etl
-a----	14/01/2021 15:30	327680	UpdateNotificationPipeline.002.etl
-a----	14/01/2021 14:52	327680	UpdateNotificationPipeline.003.etl
Répertoire : C:\Temp\ETLPARSER_PY\etl-parser-master\tests\example			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	23/07/2020 10:55	393216	AMSITrace.etl
-a----	23/07/2020 10:55	22020096	BootPerfDiagLogger.etl
-a----	23/07/2020 10:55	24576	lxcore_kernel.etl
-a----	23/07/2020 10:55	55050240	NetTrace.etl
-a----	23/07/2020 10:55	3211264	ShutdownPerfDiagLogger.etl
Répertoire : C:\Windows\Logs\NetSetup			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	22/11/2021 10:46	262144	service.0.etl
-a----	22/11/2021 08:33	3211264	service.1.etl
Répertoire : C:\Windows\Logs\SIH			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	18/11/2021 10:58	12288	SIH.20211118.105829.591.1.etl
-a----	19/11/2021 08:28	12288	SIH.20211119.082819.654.1.etl
-a----	22/11/2021 08:24	8192	SIH.20211122.082359.625.1.etl
Répertoire : C:\Windows\Logs\waasmedic			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	22/11/2021 12:59	12288	waasmedic.20211122_115820_835.etl
-a----	22/11/2021 14:03	20480	waasmedic.20211122_130218_132.etl
-a----	22/11/2021 14:56	12288	waasmedic.20211122_135529_011.etl
Répertoire : C:\Windows\Logs\waasmediccapsule			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-----	14/11/2021 15:31	131072	WaasRemediation.001.etl
Répertoire : C:\Windows\Logs\WindowsUpdate			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	22/11/2021 13:08	131072	WindowsUpdate.20211122.125646.516.1.etl
-a----	22/11/2021 14:12	32768	WindowsUpdate.20211122.140217.294.1.etl
-a----	22/11/2021 15:05	28672	WindowsUpdate.20211122.145527.536.1.etl

Analyse des fichiers ETL

Répertoire : C:\Windows\Panther			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	14/01/2021 15:54	17707	DITel.Merge.00001.etl
-a----	14/01/2021 16:42	700416	setup.etl
-a----	14/01/2021 16:24	10992	WinRETel.etl.Merge.00001.etl
Répertoire : C:\Windows\Performance\WinSAT\DataStore			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	02/01/2021 15:41	147456	2021-01-02 15.39.53.185.winsat.etl
Répertoire : C:\Windows\security\logs			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	14/01/2021 16:40	65536	SceSetupLog.etl
Répertoire : C:\Windows\System32\LogFiles\WMI			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-----	22/11/2021 09:41	16777216	LwtNetLog.etl
-a----	22/11/2021 09:59	23068672	NetCore.etl
-a----	19/11/2021 13:51	6291456	RadioMgr.etl
-----	22/11/2021 09:42	8437760	Wifi.etl
Répertoire : C:\Windows\System32\LogFiles\WMI\RtBackup			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	06/11/2021 15:19	72	EtwRTDefenderApiLogger.etl
-a----	06/11/2021 15:19	72	EtwRTDefenderAuditLogger.etl
-a----	17/10/2021 14:08	72	EtwRTDiagLog.etl
-a----	17/10/2021 14:07	72	EtwRTEventLog-Application.etl
-a----	17/10/2021 14:08	72	EtwRTEventlog-Security.etl
-a----	17/10/2021 14:08	72	EtwRTEventLog-System.etl
-a----	17/10/2021 14:10	0	EtwRTSgrmEtwSession.etl
-a----	17/10/2021 14:08	72	EtwRTUBPM.etl
-a----	17/10/2021 14:08	0	EtwRTWFP-IPsec Diagnostics.etl
Répertoire : C:\Windows\System32\Logs			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	18/08/2021 10:36	131072	UpdateHealthTools.001.etl
-a----	07/06/2021 18:25	131072	UpdateHealthTools.002.etl
-a----	26/04/2021 22:04	131072	UpdateHealthTools.003.etl
-a----	01/03/2021 09:00	131072	UpdateHealthTools.004.etl
Répertoire : C:\Windows\System32\NDF			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	03/02/2021 17:21	786432	eventlog.etl

Analyse des fichiers ETL

Répertoire : C:\Windows\System32\SleepStudy			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	18/11/2021 13:45	9762	user-not-present-trace-2021-11-18-13-45-33.etl
-a----	18/11/2021 15:27	4469	user-not-present-trace-2021-11-18-15-27-33.etl
-a----	19/11/2021 14:10	35805	user-not-present-trace-2021-11-19-14-10-22.etl
-a----	22/11/2021 13:20	16167	user-not-present-trace-2021-11-22-13-20-18.etl
-----	22/11/2021 13:20	2097152	UserNotPresentSession.etl
Répertoire : C:\Windows\System32\SleepStudy\ScreenOn			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	17/11/2021 17:21	393216	ScreenOnPowerStudyTraceSession-2021-11-17-00-28-00.etl
-a----	18/11/2021 17:31	327680	ScreenOnPowerStudyTraceSession-2021-11-17-17-21-57.etl
-a----	19/11/2021 16:59	262144	ScreenOnPowerStudyTraceSession-2021-11-18-17-31-00.etl
-----	19/11/2021 16:59	0	ScreenOnPowerStudyTraceSession-2021-11-19-16-59-10.etl
Répertoire : C:\Windows\System32\WDI\LogFiles			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	17/10/2021 14:09	28311552	BootPerfDiagLogger.etl
-a----	22/11/2021 09:17	20971520	ShutdownPerfDiagLogger.etl
Répertoire : C:\Windows\System32\WDI\{533a67eb-9fb5-473d-b884-958cf4b9c4a3}\{9041f38b-adb8-489f-bc24-1d3a36531bdc}			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	09/11/2021 19:49	2097152	snapshot.etl
Répertoire : C:\Windows\System32\WDI\{533a67eb-9fb5-473d-b884-958cf4b9c4a3}\{db7af707-ceb6-495d-aecf-f8415af6116e}			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	18/11/2021 10:53	2097152	snapshot.etl
Répertoire : C:\Windows\System32\winevt\Logs			
Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a----	02/03/2021 14:40	1048576	AirSpaceChannel.etl
-a----	10/02/2021 23:11	4096	DebugChannel.etl