AYI NEDJIMI Consultants

Registry Forensics : Guide Expert Analyse Securite

Catégorie : Forensics Lecture : 25 min Publié le : 07/12/2025 Auteur : Ayi NEDJIMI

analyse forensique du registre Windows : ruches NTUSER.DAT, SAM, SYSTEM, SOFTWARE, clés critiques, techniques d. Guide technique complet avec...

🔐 Registry Forensics Complet : Analyse Avancée du Registre Windows

Guide expert d'analyse forensique du registre Windows : ruches NTUSER.DAT, SAM, SYSTEM, SOFTWARE, clés critiques, techniques d'extraction, outils et méthodologies pour investigations numériques approfondies. La reponse aux incidents et l'analyse forensique requierent une expertise technique pointue et une methodologie rigoureuse. Les equipes DFIR sont confrontees a des defis croissants : volumes de donnees massifs, techniques d'evasion élaborées et environnements hybrides cloud. Cet article fournit un guide technique complet avec des procedures detaillees et des exemples concrets pour les professionnels de l'investigation numerique. L'investigation numérique exige rigueur et méthodologie. Registry Forensics : Guide Expert Analyse Securite couvre les aspects pratiques que les analystes forensics rencontrent sur le terrain. Nous abordons notamment : 🔐 registry forensics complet : analyse avancée du registre windows, introduction : le registre windows comme source d'evidence numérique et architecture et structure interne du registre windows. Les professionnels y trouveront des recommandations actionnables, des commandes prêtes à l'emploi et des stratégies de mise en œuvre adaptées aux environnements d'entreprise.

En cas d'incident, seriez-vous capable de retracer le parcours exact de l'attaquant ?

Introduction : Le Registre Windows comme Source d'Evidence Numérique

Le registre Windows constitue l'une des sources d'informations les plus riches et les plus complexes en investigation numérique. Cette base de données hiérarchique centralisée stocke la quasi-totalité des paramètres de configuration du système d'exploitation, des applications et des activités utilisateurs. Pour l'investigateur forensique, le registre représente une véritable mine d'or d'artefacts numériques, permettant de reconstituer avec précision les actions effectuées sur un système, les programmes exécutés, les périphériques connectés, et même les intentions des utilisateurs.

L'analyse forensique du registre Windows nécessite une compréhension approfondie de sa structure interne, de ses mécanismes de fonctionnement et des multiples pièges qui peuvent compromettre l'intégrité d'une investigation. Contrairement aux systèmes de fichiers traditionnels, le registre utilise une structure de base de données propriétaire avec ses propres mécanismes de journalisation, de mise en cache et de gestion transactionnelle. Cette complexité technique exige une approche méthodologique rigoureuse et l'utilisation d'outils spécialisés pour extraire, analyser et interpréter correctement les données.

📋 **Contenu de ce guide :**

- Architecture du registre Windows et structure binaire des ruches
- Analyse détaillée des ruches principales (NTUSER.DAT, SAM, SYSTEM, SOFTWARE)
- Clés forensiques critiques et leurs significations
- Techniques d'extraction sécurisée et méthodologies avancées
- Outils d'analyse et automatisation
- Cas pratiques d'investigation

Notre avis d'expert
L'analyse de la mémoire vive est devenue incontournable dans les investigations modernes. Les malwares fileless, les attaques living-off-the-land et les techniques d'injection en mémoire ne laissent souvent aucune trace sur le disque. Ignorer la RAM, c'est passer à côté de 60% des preuves.

Organisation Hiérarchique et Ruches du Registre

Le registre Windows s'organise selon une structure hiérarchique similaire à un système de fichiers, avec des clés (équivalentes aux dossiers) et des valeurs (équivalentes aux fichiers). Au niveau le plus élevé, le registre est divisé en cinq ruches racines principales, chacune ayant un rôle spécifique dans la gestion du système :

HKEY_LOCAL_MACHINE (HKLM) constitue le cœur du registre système, contenant les informations de configuration globale applicable à tous les utilisateurs. Cette ruche est physiquement stockée dans plusieurs fichiers situés dans `%SystemRoot%\\System32\\Config\\`, incluant SAM, SECURITY, SOFTWARE, SYSTEM et DEFAULT. Chacun de ces fichiers représente une sous-ruche avec des responsabilités distinctes dans la gestion du système.

HKEY_CURRENT_USER (HKCU) contient les paramètres spécifiques à l'utilisateur actuellement connecté. Cette ruche est une vue dynamique du fichier NTUSER.DAT de l'utilisateur, stocké dans son profil (`%UserProfile%\\NTUSER.DAT`). Elle inclut les préférences personnelles, les configurations d'applications et l'historique d'utilisation spécifique à cet utilisateur.

HKEY_USERS (HKU) regroupe les ruches de tous les profils utilisateurs chargés en mémoire. Chaque utilisateur est identifié par son Security Identifier (SID), permettant d'accéder aux configurations de plusieurs utilisateurs simultanément lors d'une analyse forensique.

HKEY_CLASSES_ROOT (HKCR) est une vue combinée de `HKLM\\SOFTWARE\\Classes` et `HKCU\\SOFTWARE\\Classes`, gérant les associations de fichiers, les enregistrements COM et les informations OLE. Cette ruche virtuelle facilite l'accès aux informations de classes sans nécessiter de naviguer entre les ruches utilisateur et système.

HKEY_CURRENT_CONFIG (HKCC) fournit un accès direct aux informations de configuration matérielle actuellement utilisées, extraites de `HKLM\\SYSTEM\\CurrentControlSet\\Hardware Profiles\\Current`.

Structure Binaire et Format des Fichiers de Ruche

Les fichiers de ruche utilisent un format binaire propriétaire complexe, organisé en pages de 4096 octets (4 Ko). Chaque fichier commence par une signature "regf" suivie d'un en-tête contenant des métadonnées critiques pour l'analyse forensique :

L'en-tête de base (Base Block) occupe les 4096 premiers octets et contient la signature, le numéro de séquence primaire et secondaire, l'horodatage de dernière modification, les versions majeures et mineures, le type de fichier, le format, le nom de la ruche et un checksum XOR pour validation de l'intégrité.

Les cellules de données (Hive Bins) suivent l'en-tête et contiennent les structures de données réelles du registre. Chaque bin commence par une signature "hbin", suivie de l'offset relatif au début du premier bin, de la taille du bin actuel et de séries de cellules contenant les clés, valeurs, listes de sous-clés, et descripteurs de sécurité.

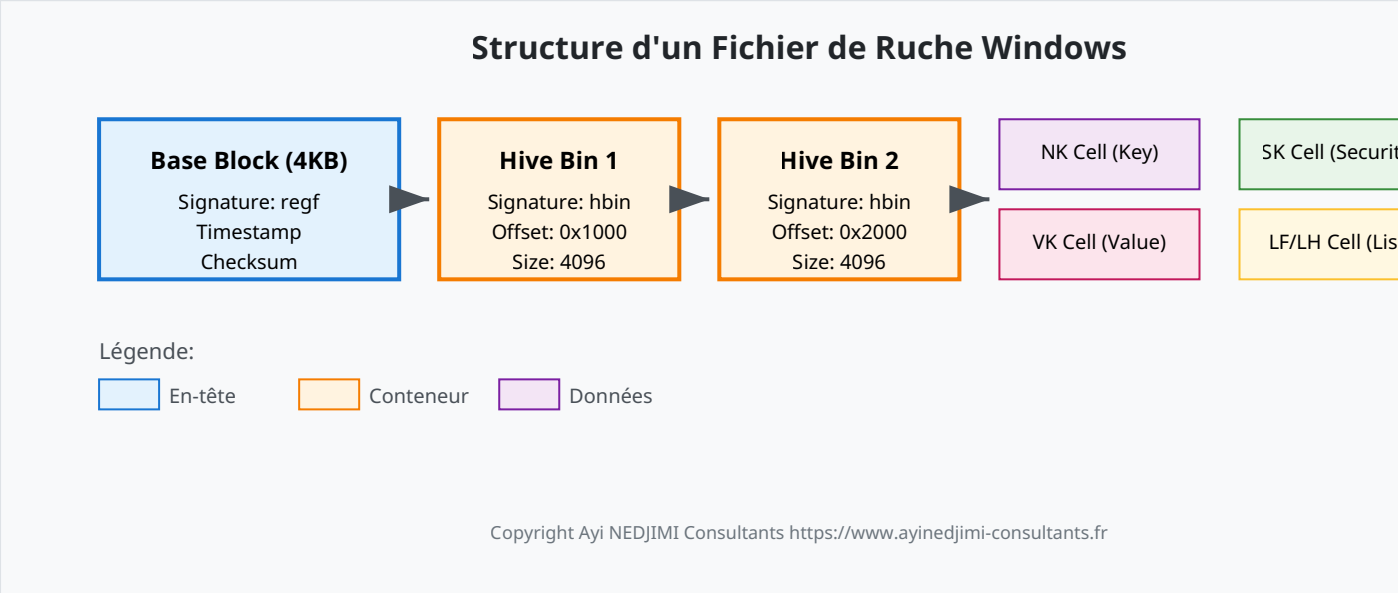

Illustration 1 : Structure Binaire d'un Fichier de Ruche Windows

Mécanismes de Transaction et Journalisation

Le registre Windows implémente un système complexe de journalisation transactionnelle pour garantir l'intégrité des données en cas de panne système. Ce mécanisme utilise des fichiers de log avec l'extension .LOG, .LOG1 et .LOG2, stockés aux côtés des fichiers de ruche principaux.

Le processus de journalisation suit le principe Write-Ahead Logging (WAL), où toute modification est d'abord écrite dans le journal avant d'être appliquée à la ruche principale. Cette approche permet une récupération cohérente en cas d'interruption brutale. Les fichiers .LOG contiennent les pages modifiées (dirty pages) et utilisent une structure de double buffer pour optimiser les performances tout en maintenant la cohérence.

Depuis Windows Vista, le Kernel Transaction Manager (KTM) et le Common Log File System (CLFS) ont introduit un support transactionnel natif au niveau du registre. Les transactions peuvent être atomiques, cohérentes, isolées et durables (ACID), permettant des modifications groupées qui sont soit toutes appliquées, soit toutes annulées.

Cas concret
L'investigation forensique après l'attaque Colonial Pipeline (2021) a permis au FBI de tracer et récupérer 2,3 millions de dollars en Bitcoin versés en rançon au groupe DarkSide. L'analyse des transactions blockchain et la coopération avec les exchanges ont démontré que les cryptomonnaies ne garantissent pas l'anonymat des cybercriminels.

Vos preuves numériques seraient-elles recevables devant un tribunal ?

Analyse Détaillée des Ruches Principales

NTUSER.DAT : Profil et Activités Utilisateur

Le fichier NTUSER.DAT représente le cœur du profil utilisateur dans Windows, stockant l'ensemble des préférences personnelles, configurations d'applications et traces d'activités. Cette ruche est chargée dynamiquement lors de la connexion de l'utilisateur et mappée sous HKEY_CURRENT_USER. Pour l'investigateur forensique, NTUSER.DAT constitue une source inestimable d'informations sur les comportements et actions de l'utilisateur.

La structure de NTUSER.DAT s'organise autour de plusieurs branches principales, chacune contenant des artefacts forensiques spécifiques. La branche `Software\\Microsoft\\Windows\\CurrentVersion` contient la majorité des traces d'activités utilisateur, incluant l'historique d'exécution des programmes, les recherches effectuées, les documents récents et les préférences d'interface.

L'analyse de la clé `Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\UserAssist` révèle un historique détaillé des programmes exécutés via l'interface graphique. Les entrées sont encodées en ROT13, un chiffrement par substitution simple où chaque lettre est remplacée par

celle située 13 positions plus loin dans l'alphabet. Chaque entrée contient un compteur d'exécution et un timestamp FILETIME de dernière exécution, permettant de reconstituer les habitudes d'utilisation.

SAM : Base de Données des Comptes Locaux

La ruche Security Account Manager (SAM) contient les informations critiques sur les comptes utilisateurs locaux, incluant les hashes de mots de passe, les appartenances aux groupes, les politiques de compte et les métadonnées de sécurité. Située dans `%SystemRoot%\\System32\\Config\\SAM`, cette ruche est verrouillée exclusivement par le processus SYSTEM pendant le fonctionnement normal de Windows, nécessitant des techniques spéciales pour son extraction en live forensics.

La structure de la SAM s'articule autour de deux branches principales : `SAM\\Domains\\Account` contenant les informations des comptes utilisateurs, et `SAM\\Domains\\Builtin` gérant les groupes de sécurité intégrés. Chaque compte utilisateur est identifié par un Relative Identifier (RID), un identifiant numérique unique ajouté au SID du domaine pour former le SID complet de l'utilisateur.

Points d'attention specifiques

⚠️ Sécurité des Hashes :

Les hashes NT et LM sont chiffrés avec une clé dérivée du RID et de la clé de boot système (SYSKEY). L'analyse forensique de la SAM permet de récupérer ces hashes pour des techniques de cracking offline, mais nécessite également l'extraction de la SYSKEY depuis la ruche SYSTEM.

L'analyse forensique de la SAM permet de reconstituer l'historique des comptes, identifier les comptes cachés ou suspects, détecter les escalades de privilèges non autorisées, et potentiellement récupérer les mots de passe via des techniques de cracking offline. Les timestamps contenus dans la SAM sont particulièrement précieux pour établir une timeline des activités d'administration et détecter les modifications suspectes de comptes.

SYSTEM : Configuration Matérielle et Services

La ruche SYSTEM constitue le centre névralgique de la configuration matérielle et des services Windows. Elle contient les ControlSets qui définissent la configuration de démarrage, les paramètres des pilotes de périphériques, la configuration des services système, et l'historique des configurations précédentes. Cette ruche est critique pour le démarrage du système et est chargée très tôt dans le processus de boot.

Windows maintient plusieurs ControlSets pour assurer la résilience du système. Le CurrentControlSet est un lien symbolique pointant vers le ControlSet actuellement utilisé, identifié par la valeur Current dans `SYSTEM\\Select`. Le ControlSet001 et ControlSet002 représentent généralement la dernière configuration connue fonctionnelle et la configuration de sauvegarde. La valeur LastKnownGood dans Select identifie le ControlSet à utiliser lors d'un démarrage en dernière bonne configuration connue.

📌 Artefacts USB dans SYSTEM :

La section `SYSTEM\\CurrentControlSet\\Enum\\USB` constitue une mine d'informations sur les périphériques USB connectés au système. Chaque périphérique est identifié par son Vendor ID (VID) et Product ID (PID), avec des sous-clés contenant le numéro de série unique. Les propriétés du périphérique incluent les timestamps de première installation, dernière connexion, et les pilotes associés.

L'analyse des services dans `SYSTEM\\CurrentControlSet\\Services` révèle non seulement les services légitimes mais aussi potentiellement les malwares installés comme services. Chaque entrée de service contient le type de démarrage (automatique, manuel, désactivé), le chemin de l'exécutable, les dépendances, et les paramètres de récupération. Les timestamps de modification peuvent indiquer quand un service a été installé ou modifié, crucial pour l'analyse de compromission.

SOFTWARE : Applications et Configurations Système

La ruche SOFTWARE, la plus volumineuse des ruches système, stocke les configurations de toutes les applications installées et de nombreux composants Windows. Cette ruche est particulièrement riche en artefacts forensiques, documentant l'inventaire logiciel, les associations de fichiers, les configurations réseau, et les politiques de sécurité appliquées.

La clé `SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Uninstall` contient l'inventaire des applications installées, chaque sous-clé représentant une application avec ses métadonnées : nom, version, éditeur, date d'installation, taille, et chemin d'installation. Cette information est cruciale pour établir la présence de logiciels spécifiques, incluant les outils potentiellement utilisés dans une attaque.

Les clés `SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Run` et `RunOnce` définissent les programmes lancés automatiquement au démarrage du système pour tous les utilisateurs. Ces emplacements sont fréquemment exploités par les malwares pour assurer leur persistance. L'analyse comparative avec les clés équivalentes dans NTUSER.DAT permet d'identifier les mécanismes de persistance au niveau utilisateur versus système.

Clés Forensiques Critiques et Techniques d'Analyse

UserAssist : Traçage de l'Exécution des Applications

La clé UserAssist, située dans `NTUSER.DAT\\Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\UserAssist`, représente l'un des artefacts les plus précieux pour reconstituer l'activité d'exécution des programmes. Cette clé maintient des statistiques détaillées sur les programmes lancés via l'interface Explorer, incluant les applications du menu Démarrer, les raccourcis du bureau, et les exécutions via la barre des tâches.

Les données UserAssist sont organisées sous deux GUIDs principaux : `{CEBFF5CD-ACE2-4F4F-9178-9926F41749EA}` pour les exécutables et `{F4E57C4B-2036-45F0-A9AB-443BCFE33D9F}` pour les raccourcis. Chaque entrée est encodée en ROT13 et contient une structure binaire de 72 octets (Windows 7+) incluant :

- **Offset 4** : Compteur d'exécution

- **Offset 12** : Compteur de focus (temps où l'application avait le focus)
- **Offset 60** : Timestamp de dernière exécution (format FILETIME Windows)

⚠️ **Attention aux Versions Windows :**

L'interprétation correcte nécessite de connaître la version de Windows. Windows XP utilise une structure de 16 octets, Vista étend à 72 octets, et Windows 7+ modifie légèrement les offsets. Le compteur d'exécution dans Windows 7+ commence à 5 par défaut, nécessitant une soustraction de 5 pour obtenir le nombre réel d'exécutions.

MUICache : Applications Exécutées et Descriptions

Le MUICache (Multilingual User Interface Cache) stocke les descriptions des applications exécutées, extraites de leurs ressources. Située dans `NTUSER.DAT\\Software\\Classes\\Local Settings\\Software\\Microsoft\\Windows\\Shell\\MuiCache` (Windows Vista+) ou `NTUSER.DAT\\Software\\Microsoft\\Windows\\ShellNoRoam\\MUICache` (Windows XP), cette clé fournit une liste des exécutables avec leurs descriptions localisées.

Contrairement à UserAssist, MUICache n'inclut pas de compteurs ou de timestamps, mais sa valeur réside dans sa couverture exhaustive. Toute application exécutée, même une seule fois, même si immédiatement supprimée, laissera une trace dans MUICache si Windows a pu extraire sa description. Cela inclut les applications portables, les malwares, et les outils d'administration qui pourraient ne pas apparaître dans d'autres artefacts.

ShellBags : Navigation et Préférences des Dossiers

Les ShellBags constituent un mécanisme de Windows pour mémoriser les préférences d'affichage des dossiers (taille, position, vue). Ces artefacts, stockés dans plusieurs emplacements du registre, documentent indirectement la navigation de l'utilisateur dans le système de fichiers, incluant les dossiers supprimés, les partages réseau, et les médias amovibles.

Les ShellBags sont principalement stockés dans `NTUSER.DAT\\Software\\Microsoft\\Windows\\Shell\\BagMRU` et `Bags`, avec des copies dans UsrClass.dat pour Windows 7+. La structure BagMRU forme un arbre binaire mimant la hiérarchie des dossiers, où chaque nœud contient un shell item identifiant le dossier. Les Bags contiennent les préférences d'affichage associées, référencées par un NodeSlot.

TypedURLs et TypedPaths : Historique de Navigation

Les clés TypedURLs et TypedPaths dans `NTUSER.DAT\\Software\\Microsoft\\Internet Explorer\\TypedURLs` et `NTUSER.DAT\\Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\TypedPaths` documentent respectivement les URLs saisies dans Internet Explorer et les chemins tapés dans l'Explorateur Windows.

TypedURLs maintient les 50 dernières URLs (par défaut) saisies manuellement dans la barre d'adresse d'Internet Explorer. Cela exclut les URLs accédées via favoris ou liens, se concentrant sur la navigation intentionnelle. Chaque entrée est numérotée (url1, url2, etc.) avec l'URL la plus

récente ayant le numéro le plus élevé. L'ordre chronologique peut être reconstitué en corrélation avec les timestamps de modification de la clé. Les recommandations de MITRE ATT&CK constituent une reference essentielle.

TypedPaths fonctionne similairement pour les chemins saisis dans la barre d'adresse de l'Explorateur, documentant l'accès direct aux dossiers locaux, partages réseau, et chemins UNC. Cette information est particulièrement précieuse pour identifier l'accès intentionnel à des ressources spécifiques, potentiellement en préparation d'exfiltration de données.

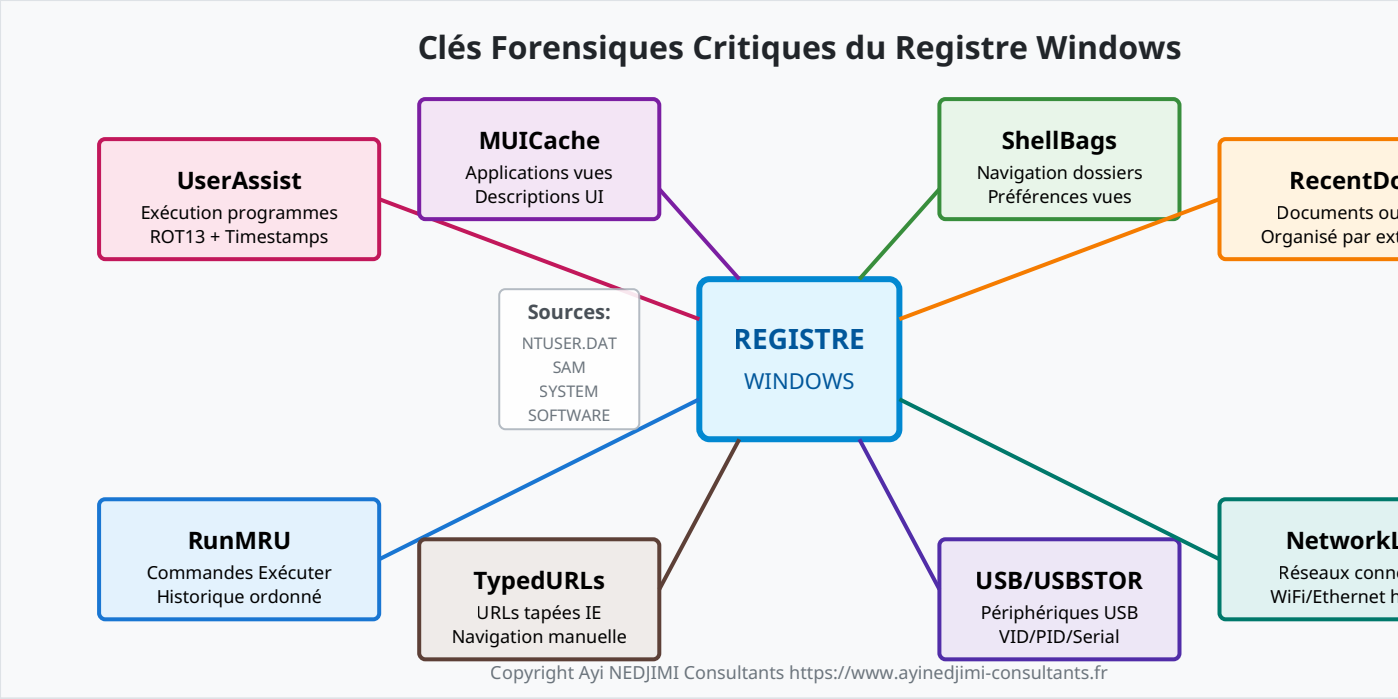

Illustration 2 : Cartographie des Clés Forensiques Critiques du Registre Windows

ComDlg32 : Historique des Boîtes de Dialogue

La clé ComDlg32, située dans `NTUSER.DAT\\Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\ComDlg32`, maintient l'historique des interactions avec les boîtes de dialogue communes Windows (Ouvrir, Enregistrer sous). Cette clé est structurée en deux sections principales : OpenSavePidlMRU pour les fichiers et LastVisitedPidlMRU pour les dossiers.

OpenSavePidlMRU organise l'historique par extension de fichier, chaque sous-clé correspondant à une extension spécifique. Les entrées contiennent des shell items encodés représentant les fichiers sélectionnés, avec leurs métadonnées incluant taille, dates MAC (Modified, Accessed, Created) au moment de la sélection. Cette information peut révéler l'existence de fichiers même après leur suppression.

LastVisitedPidlMRU corrèle les applications avec les derniers dossiers visités via leurs boîtes de dialogue. Chaque entrée combine le nom de l'exécutable et le chemin du dossier, permettant de reconstituer le contexte d'utilisation. Cette information est particulièrement utile pour identifier les patterns d'accès aux données et potentielles tentatives d'exfiltration.

Techniques Avancées d'Extraction et d'Analyse

Extraction Sécurisée des Ruches

L'extraction forensique des ruches du registre nécessite une méthodologie rigoureuse pour préserver l'intégrité des preuves et éviter la contamination. Plusieurs approches sont disponibles selon le contexte de l'investigation et l'état du système.

Extraction depuis un Système Hors Ligne
Pour l'extraction depuis un système hors ligne, l'approche la plus sûre consiste à monter le disque en lecture seule et copier directement les fichiers de ruche. Les fichiers principaux se trouvent dans `%SystemRoot%\\System32\\Config\\` pour les ruches système et dans `%UserProfile%\\` pour NTUSER.DAT. Il est crucial de capturer également les fichiers de transaction (.LOG, .LOG1, .LOG2) pour permettre la reconstruction des modifications non committées.

Extraction depuis un Système En Ligne
L'extraction depuis un système en ligne présente des défis supplémentaires car les ruches sont verrouillées par le système. La méthode Volume Shadow Copy Service (VSS) permet d'accéder à une copie instantanée cohérente du système. Les outils comme vssadmin ou des solutions forensiques spécialisées peuvent créer et monter des shadow copies pour extraction. Pour approfondir, consultez Livre Blanc : Sécurisation.

💡 **Méthodes d'Extraction Live :**

- **Volume Shadow Copies (VSS)** : Copie instantanée cohérente du système
- **API RegSaveKey** : Export via API Windows (nécessite privilèges BACKUP)
- **Extraction mémoire** : Via outils comme Volatility (hivelist, hivedump)
- **Raw disk reading** : Lecture directe du disque (privilèges élevés requis)

Validation de l'Intégrité
La validation de l'intégrité post-extraction est critique. Le calcul de hashes cryptographiques (SHA-256 minimum) doit être effectué immédiatement après l'extraction. La vérification du checksum interne de la ruche (XOR-32 dans l'en-tête) peut détecter les corruptions. La comparaison des numéros de séquence primaire et secondaire dans l'en-tête identifie les écritures incomplètes.

Parsing et Reconstruction des Structures

Le parsing des ruches nécessite une compréhension approfondie du format binaire et la capacité de reconstruire les structures même en présence de corruption. Les outils modernes implémentent des algorithmes robustes de récupération.

L'analyse commence par la validation de la signature "regf" et l'extraction des métadonnées de l'en-tête. Le parsing des hive bins suit, identifiant chaque bin par sa signature "hbin" et parcourant les cellules qu'il contient. La reconstruction de l'arbre de clés nécessite de suivre les offsets relatifs entre les cellules, construisant progressivement la hiérarchie.

La reconstruction des transactions incomplètes utilise les fichiers de log pour appliquer ou annuler les modifications pending. L'analyse différentielle entre les pages du log et de la ruche principale révèle les modifications en cours. Cette technique peut exposer des tentatives de modification avortées ou des activités malveillantes interrompues.

Analyse Temporelle et Reconstruction Chronologique

L'horodatage dans le registre Windows est complexe et multifacette, nécessitant la corrélation de plusieurs sources temporelles pour une reconstruction chronologique précise.

Les timestamps au niveau des clés utilisent le format FILETIME Windows (100-nanoseconde intervals depuis 1601). Chaque clé maintient un LastWriteTime indiquant sa dernière modification. Cependant, ce timestamp est hérité : la modification d'une valeur met à jour le timestamp de sa clé parente, mais pas des clés ancêtres. Cette propriété permet d'identifier les branches récemment modifiées mais complique la datation précise des changements.

⚠️ **Pièges des Timestamps :**

- Le LastWriteTime n'indique PAS quand une action spécifique s'est produite
- Les timestamps sont hérités aux clés parentes, pas aux ancêtres
- Les valeurs n'ont généralement pas de timestamps directs
- Corréler avec d'autres sources (Event Logs, MFT) est impératif
- Les anomalies temporelles peuvent indiquer du timestomping

La corrélation multi-sources est essentielle pour une timeline précise. Les timestamps du registre doivent être croisés avec les journaux d'événements Windows, les timestamps du système de fichiers (MAC times), les entrées de Prefetch, et les artefacts de journalisation des applications. Cette approche holistique permet de valider les timestamps du registre et détecter les manipulations temporelles.

Artefacts Spécifiques et Cas d'Usage Avancés

Analyse de la Persistance des Malwares

Le registre Windows constitue un vecteur privilégié pour l'établissement de la persistance des malwares. L'analyse forensique doit examiner systématiquement les multiples emplacements exploités par les acteurs malveillants.

Les clés Run classiques (HKLM et HKCU `\\Software\\Microsoft\\Windows\\CurrentVersion\\Run`) restent les plus couramment exploitées. L'analyse doit inclure les variantes : RunOnce pour exécution unique, RunServices et RunServicesOnce pour exécution en tant que service, et les clés `Policies\\Explorer\\Run` souvent négligées. La comparaison entre les versions HKLM et HKCU peut révéler des persistances ciblant des utilisateurs spécifiques.

Les services malveillants dans `SYSTEM\\CurrentControlSet\\Services` méritent une attention particulière. Les indicateurs incluent des noms de services imitant des services légitimes, des chemins d'exécutable pointant vers des emplacements temporaires ou utilisateur, des descriptions manquantes ou génériques, et des dépendances inhabituelles. L'analyse des timestamps peut identifier les services récemment ajoutés corrélés avec l'incident.

Les techniques d'évasion incluent l'utilisation de caractères null dans les noms de clés pour éviter l'affichage dans regedit, le stockage de payloads encodés ou chiffrés dans des valeurs binaires, et l'exploitation de clés CLSID pour le hijacking de COM objects. L'analyse doit utiliser des outils capables de détecter ces techniques d'obfuscation.

Investigation des Connexions Réseau

Le registre documente extensivement les activités réseau, fournissant des preuves cruciales pour les investigations impliquant des accès non autorisés ou des exfiltrations de données.

La clé `SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion\\NetworkList\\Profiles` contient les profils détaillés de tous les réseaux auxquels le système s'est connecté. Chaque profil inclut le GUID du réseau, son nom (ProfileName), le type (domaine, privé, public), la catégorie (wired, wireless, broadband), et les timestamps de première et dernière connexion. Les Signatures associées contiennent les détails techniques incluant les adresses MAC des passerelles.

Les connexions VPN laissent des traces dans `SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Internet Settings\\Connections` et dans les profils RAS (Remote Access Service). Les configurations incluent les serveurs VPN, les protocoles utilisés, et potentiellement des credentials cached. L'analyse peut révéler des connexions à des infrastructures d'attaquants ou des tentatives de contournement de la sécurité périmétrique.

🔌 **Artefacts de Connexion Réseau :**

- **NetworkList\\Profiles** : Historique complet des réseaux WiFi/Ethernet
- **Network (HKCU)** : Lecteurs réseau mappés persistants
- **MountPoints2** : Tous points de montage incluant partages temporaires
- **Terminal Server Client** : Historique de connexions RDP
- **Internet Settings\\Connections** : Configurations VPN et proxies

Les partages réseau mappés sont documentés dans `HKCU\\Network` pour les drives persistants et dans `HKCU\\Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\MountPoints2` pour tous les points de montage, incluant les partages temporaires. Les chemins UNC complets révèlent les serveurs accédés et peuvent exposer des mouvements latéraux dans le réseau.

Analyse des Périphériques de Stockage

L'historique complet des périphériques de stockage connectés au système est minutieusement enregistré dans plusieurs emplacements du registre, fournissant une piste d'audit détaillée pour l'investigation d'exfiltration de données ou d'introduction de malwares.

`SYSTEM\\CurrentControlSet\\Enum\\USBSTOR` catalogue tous les périphériques de stockage USB connectés. Chaque périphérique est identifié par une combinaison Fabricant&Produit&Version, avec une sous-clé pour chaque numéro de série unique. Les propriétés incluent les timestamps

d'installation et de dernière connexion, les capabilities du périphérique, et les pilotes associés. La corrélation avec les Volume Serial Numbers permet de lier les périphériques aux lettres de lecteur assignées.

La clé `SYSTEM\\MountedDevices` mappe les identifiants de volumes aux lettres de lecteur et points de montage. Les valeurs `\\DosDevices\\` correspondent aux lettres de lecteur, tandis que les `\\??\\Volume{GUID}` représentent les volumes. Le parsing des données binaires révèle les signatures de disque et les offsets de partition, permettant la reconstruction de la configuration de stockage même après la déconnexion des périphériques.

`SOFTWARE\\Microsoft\\Windows Portable Devices\\Devices` documente les périphériques mobiles modernes (smartphones, tablettes) connectés via MTP/PTP. Les identifiants incluent les modèles de périphériques, les numéros de série, et les timestamps de synchronisation. Cette information est cruciale pour les investigations impliquant des périphériques mobiles comme vecteurs de données.

Méthodologies d'Investigation et Corrélation

Construction de Timelines Multi-Sources

La construction d'une timeline comprehensive nécessite l'agrégation et la corrélation de timestamps provenant de multiples artefacts du registre et d'autres sources système.

La méthodologie commence par l'extraction systématique de tous les timestamps disponibles : LastWriteTime des clés, timestamps intégrés dans les valeurs (UserAssist, ShellBags), et métadonnées temporelles dans les structures binaires. Chaque timestamp doit être normalisé en UTC et documenté avec sa source et son contexte.

L'enrichissement de la timeline intègre les timestamps du système de fichiers, les journaux d'événements, les fichiers de Prefetch, et les logs d'applications. La corrélation révèle les séquences d'actions : une entrée RunMRU suivie d'une création de processus dans les logs, puis d'une entrée UserAssist confirme l'exécution intentionnelle d'un programme.

📊 Sources de Timestamps à Corréler :

Source	Type d'Information	Précision
Registre (LastWriteTime)	Modification de clé	100-nanosec
UserAssist	Dernière exécution	100-nanosec
MFT (NTFS)	Création/Modif/Accès	100-nanosec
Prefetch	Dernières 8 exécutions	Seconde
Event Logs	Événements système	Milliseconde
AmCache/ShimCache	Exécution programmes	Variable
VSS Snapshots	État système passé	Snapshot time

L'analyse des patterns temporels identifie les comportements anormaux. Les burst d'activité en dehors des heures normales, les séquences d'actions automatisées (intervals réguliers suggérant des scripts), ou les gaps temporels (suggesting anti-forensics) méritent une investigation approfondie. Les clusters de modifications simultanées dans différentes ruches peuvent indiquer l'installation de logiciels ou des modifications système majeures.

Détection d'Anti-Forensiques

Les techniques anti-forensiques ciblant le registre sont abouties et variées, nécessitant des approches de détection avancées.

Le nettoyage sélectif des artefacts laisse des traces indirectes. L'absence suspecte d'entrées attendues (UserAssist vide malgré une utilisation évidente), les gaps dans les séquences MRU, ou les timestamps de clés parentes sans valeurs correspondantes suggèrent un nettoyage. La comparaison avec les Shadow Copies peut révéler les suppressions.

⚠️ **Indicateurs d'Anti-Forensiques :**

- **Nettoyage sélectif** : Absence d'artefacts attendus, gaps dans MRU
- **Timestomping** : Timestamps anachroniques, incohérences avec autres sources
- **Obfuscation** : Caractères null/non-imprimables, encoding custom
- **Données chiffrées** : Valeurs binaires suspectes contenant payloads
- **Rootkits** : Divergences entre analyse online et offline
- **Clés cachées** : Exploitent des bugs d'affichage de regedit

La manipulation des timestamps via SetRegTime ou l'édition directe des structures de ruche peut être détectée par l'analyse des incohérences. Les timestamps de clés enfants plus anciens que les parents, les modifications sans traces correspondantes dans les logs système, ou les patterns temporels statistiquement improbables sont des indicateurs.

L'utilisation de rootkits modifiant la vue du registre en mémoire peut être détectée par la comparaison entre l'analyse online et offline. Les divergences entre les APIs Windows et l'analyse directe des fichiers de ruche révèlent les manipulations. L'analyse mémoire peut identifier les hooks et modifications de structures kernel affectant l'accès au registre.

Outils et Automatisation de l'Analyse

Outils Commerciaux et Open Source

L'écosystème d'outils pour l'analyse du registre Windows est riche et diversifié, offrant des capacités allant de l'extraction basique à l'analyse correlative avancée.

Registry Explorer (Eric Zimmerman)
Registry Explorer de Eric Zimmerman représente l'état de l'art en matière d'analyse manuelle. Il offre une navigation intuitive, le décodage automatique des structures connues (UserAssist, ShellBags, etc.), la recherche avancée avec expressions régulières, et l'affichage des timestamps supprimés. Les bookmarks et l'export détaillé facilitent la documentation des findings.

RegRipper (Harlan Carvey)
RegRipper de Harlan Carvey automatise l'extraction d'artefacts via un système de plugins. Avec plus de 300 plugins couvrant les artefacts forensiques majeurs, il génère des rapports structurés facilitant l'analyse. Le framework extensible permet le développement de plugins custom pour des besoins spécifiques. L'approche modulaire facilite l'intégration dans des workflows automatisés.

```
# Exemple d'utilisation de RegRipper
rip.exe -r NTUSER.DAT -p userassist
rip.exe -r SYSTEM -p usbstor
rip.exe -f system -a # Tous plugins pour ruche SYSTEM
```

Volatility Framework
Volatility Framework excelle dans l'analyse du registre depuis la mémoire. Les plugins hivelist, hivedump, printkey, et hashdump permettent l'extraction et l'analyse sans accès au disque. L'avantage unique est la capacité d'analyser les modifications non committées et les clés temporaires existant uniquement en mémoire.

```
# Extraction de ruches depuis dump mémoire
volatility -f memory.dmp --profile=Win10x64 hivelist
volatility -f memory.dmp --profile=Win10x64 printkey -K "Software\\Microsoft\\Windows\
\CurrentVersion\\Run"
volatility -f memory.dmp --profile=Win10x64 hashdump
```

🛠️ **Suite d'Outils Recommandés :**

- **Registry Explorer** : Analyse manuelle interactive (GUI)
- **RegRipper** : Automatisation via plugins (CLI)
- **Volatility** : Analyse depuis la mémoire
- **X-Ways Forensics / EnCase / FTK** : Suites commerciales complètes
- **python-registry** : Bibliothèque Python pour scripting custom
- **FRED (Registry Editor)** : Viewer cross-platform
- **ShellBags Explorer** : Analyse spécialisée ShellBags

Scripts et Automatisation Python

Le développement de scripts Python personnalisés permet une analyse ciblée et l'automatisation de tâches répétitives. La bibliothèque python-registry de Willi Ballenthin fournit un framework robuste pour le parsing programmatique.

```
from Registry import Registry
import codecs

# Ouverture d'une ruche
reg = Registry.Registry("NTUSER.DAT")

# Navigation vers UserAssist
key = reg.open("Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\UserAssist")

# Parcours des GUIDs
for guid_key in key.subkeys():
    count_key = guid_key.subkey("Count")

    for value in count_key.values():
        # Décodage ROT13
        decoded_name = codecs.decode(value.name(), 'rot_13')

        # Extraction des données binaires
        data = value.value()
        if len(data) >= 72:
            exec_count = int.from_bytes(data[4:8], 'little')
            timestamp = int.from_bytes(data[60:68], 'little')

            print(f"{decoded_name} - Execs: {exec_count-5} - Last: {timestamp}")
```

L'extraction automatisée des artefacts clés peut être implémentée efficacement. Un script parcourant systématiquement les emplacements forensiques connus, extrayant et décodant les valeurs (ROT13 pour UserAssist, shell items pour ShellBags), et générant des rapports structurés accélère significativement l'analyse. L'intégration avec des bases de données permet le stockage et la recherche efficace à travers plusieurs cas.

Pièges d'Interprétation et Bonnes Pratiques

Erreurs Communes d'Analyse

L'interprétation des artefacts du registre est semée d'embûches pouvant conduire à des conclusions erronées. La conscience de ces pièges est essentielle pour maintenir la rigueur forensique.

Sur-interprétation des Timestamps
La sur-interprétation des timestamps représente l'erreur la plus fréquente. Le LastWriteTime d'une clé n'indique pas nécessairement quand une action spécifique s'est produite, seulement quand la clé a été modifiée pour la dernière fois. Une clé Run peut avoir été modifiée par une mise à jour système sans que le programme référencé ait été exécuté. La correlation avec d'autres sources est impérative.

Ignorance du Contexte Système
L'ignorance du contexte système conduit à des faux positifs. Les entrées de registre peuvent être créées par des installations légitimes, des mises à jour Windows, ou des politiques d'entreprise. Une clé suspecte peut être standard dans certains environnements corporate. La baseline de l'environnement normal est cruciale pour identifier les vraies anomalies.

Confusion Existence vs Utilisation
La confusion entre preuves d'existence et preuves d'utilisation est problématique. La présence d'une entrée MUICache prouve qu'un programme a été exécuté, mais pas nécessairement par l'utilisateur (peut être via scheduled task ou service). Les ShellBags prouvent l'existence d'un dossier, pas nécessairement son accès intentionnel (peut être via application automatique).

⚠️ **Pièges Critiques à Éviter :**

- Assumer que LastWriteTime = quand l'action s'est produite
- Ignorer le contexte (environnement corporate, politiques GPO)
- Confondre existence (artefact présent) et utilisation (action intentionnelle)
- Croire que l'absence d'artefact = preuve de non-occurrence
- Ne pas considérer les limitations (MRU size, retention policies)
- Sur-interpréter un artefact isolé sans corroboration
- Ignorer les explications alternatives légitimes

Assumptions sur la Persistence
Les assumptions sur la persistence des données sont dangereuses. Les MRU lists ont des tailles limitées et écrasent les anciennes entrées. Les politiques de nettoyage peuvent purger automatiquement certains artefacts. L'absence d'une entrée n'est pas preuve de non-occurrence, seulement de non-persistence.

Validation et Corroboration

La validation rigoureuse des findings du registre est essentielle pour assurer l'admissibilité légale et la précision technique.

Corroboration Multi-Sources
La corroboration multi-sources est fondamentale. Chaque artefact significatif du registre doit être validé par au moins une source indépendante. Une exécution dans UserAssist corroborée par Prefetch, Amcache, et journal d'événements établit une preuve robuste. Les incohérences entre sources méritent une investigation approfondie.

Documentation de la Chaîne de Custody
La documentation de la chaîne de custody est critique. Chaque étape de l'extraction, du processing, et de l'analyse doit être documentée avec les outils utilisés, les paramètres appliqués, et les hashes de validation. Les screenshots, exports, et logs détaillés supportent les conclusions et permettent la revue par des pairs. Pour approfondir, consultez Ransomware Forensics : Identifier la Souche.

📋 **Checklist de Documentation :**

- Hashes cryptographiques (SHA-256) de tous fichiers sources
- Outils utilisés (nom, version, paramètres exacts)
- Commandes et requêtes exécutées (scripts préservés)
- Timestamps d'extraction et d'analyse
- Captures d'écran des findings clés
- Exports complets des clés pertinentes

- Notes détaillées sur observations et interprétations
- Considération des limitations et incertitudes

Tests de Reproductibilité
Les tests de reproductibilité valident les méthodologies. L'analyse doit être reproductible par un investigateur indépendant utilisant les mêmes données et outils. Les scripts et requêtes utilisés doivent être préservés. Les environnements de test permettent la validation des interprétations sur des systèmes contrôlés.

Considération des Explications Alternatives
La considération des explications alternatives maintient l'objectivité. Pour chaque finding, les scénarios alternatifs légitimes doivent être considérés et éliminés méthodiquement. L'analyse doit reconnaître les limitations et incertitudes plutôt que de sur-interpréter les preuves disponibles.

Cas Pratiques et Études de Cas

Investigation d'Exfiltration de Données

Un cas typique d'exfiltration de données illustre l'application pratique de l'analyse du registre. L'investigation commence par l'identification des vecteurs d'exfiltration potentiels à travers les artefacts du registre.

Phase 1 : Identification du Vecteur
L'analyse des périphériques USB via `SYSTEM\\CurrentControlSet\\Enum\\USBSTOR` révèle un dispositif de stockage inconnu connecté durant la fenêtre d'incident. Le numéro de série unique permet le traçage à travers MountedDevices pour identifier la lettre de lecteur assignée. Les timestamps confirment la connexion pendant les heures non-ouvrées, augmentant la suspicion.

Phase 2 : Reconstruction de l'Activité
Les ShellBags dans le profil du suspect montrent la navigation vers des répertoires sensibles, incluant des dossiers contenant des données confidentielles. La correlation avec RecentDocs révèle l'ouverture de fichiers spécifiques maintenant manquants du système. Les entrées ComDlg32 OpenSavePidlMRU confirment l'interaction avec ces fichiers via des boîtes de dialogue, suggesting une copie intentionnelle.

Phase 3 : Analyse Réseau
L'analyse réseau via NetworkList révèle la connexion à un hotspot mobile personnel durant l'incident, contournant le monitoring réseau corporate. Les TypedPaths montrent l'accès direct aux partages réseau contenant les données ciblées. La timeline consolidée démontre un pattern d'activité méthodique consistent avec l'exfiltration planifiée.

Phase 4 : Détection Anti-Forensiques
Les techniques anti-forensiques sont évidentes : des gaps dans UserAssist suggèrent l'utilisation d'outils de nettoyage, mais les traces résiduelles dans Amcache confirment leur exécution. Les tentatives de timestomping sont trahies par les incohérences entre les timestamps du registre et les journaux système. La récupération depuis les Shadow Copies révèle les artefacts supprimés.

Analyse Post-Compromission APT

L'investigation d'une compromission par un Advanced Persistent Threat démontre l'utilité du registre pour comprendre les techniques poussées d'attaquants.

Details techniques supplementaires

Persistance Multi-Niveaux
La persistance multi-niveaux est évidente dans le registre. Au-delà des clés Run standard, l'analyse révèle des services malveillants déguisés avec des noms imitant des services Windows légitimes. Les AppInit_DLLs contiennent des références à des DLLs malveillantes pour injection globale. Les clés WMI Event Consumers révèlent des persistances WMI pour exécution fileless.

Mouvement Latéral
Le mouvement latéral est tracé via les artefacts réseau. Terminal Server Client montre des connexions RDP vers d'autres systèmes internes avec des comptes compromis. Les MountPoints2 révèlent l'accès à des partages administratifs (C$, ADMIN$) sur plusieurs machines. Les credentials cached dans Credential Manager indiquent la récolte et réutilisation d'identifiants.

Techniques d'Évasion
Les techniques d'évasion incluent l'utilisation de clés avec caractères null pour éviter la détection, le stockage de payloads chiffrés dans des valeurs binaires apparemment bénignes, et la modification de clés de sécurité pour désactiver les défenses. L'analyse des timestamps révèle des modifications effectuées via des outils automatisés, avec des patterns temporels réguliers impossibles manuellement.

Reconstruction de la Timeline APT
La reconstruction de la timeline démontre une compromission de longue durée. Les phases distinctes sont visibles : reconnaissance initiale (queries réseau, énumération), établissement de persistances multiples, élévation de privilèges (modifications de comptes dans SAM), et exfiltration continue (patterns réguliers d'accès aux données). Les artifacts résiduels suggèrent l'utilisation d'un framework d'attaque avancé.

Considérations Légales et Reporting

Admissibilité et Standards

L'utilisation des preuves du registre dans un contexte légal nécessite le respect de standards stricts pour assurer l'admissibilité.

Conformité aux Standards
La conformité aux standards industriels est essentielle. Les méthodologies doivent suivre les guidelines établies (NIST SP 800-86, ISO/IEC 27037). Les outils utilisés doivent être validés et acceptés dans la communauté forensique. La documentation doit démontrer l'adhérence aux meilleures pratiques reconnues.

📚 **Standards et Guidelines :**

- **NIST SP 800-86** : Guide to Integrating Forensic Techniques into Incident Response
- **ISO/IEC 27037** : Guidelines for identification, collection, acquisition and preservation
- **RFC 3227** : Guidelines for Evidence Collection and Archiving
- **ACPO Guidelines** : Good Practice Guide for Digital Evidence (UK)
- **SWGDE** : Scientific Working Group on Digital Evidence standards

Qualification de l'Expert
La qualification de l'expert est cruciale pour l'admissibilité. L'analyste doit démontrer la formation, l'expérience, et les certifications pertinentes. La capacité à expliquer les concepts techniques complexes en termes compréhensibles pour un jury est essentielle. La préparation pour le témoignage inclut l'anticipation des challenges à la méthodologie.

Considérations de Privacy et Scope
Les considérations de privacy et de scope sont critiques. L'analyse doit respecter les limites légales de l'investigation, évitant l'examen de données hors scope. Les informations personnelles non pertinentes doivent être protégées et exclues des rapports. La conformité avec les régulations de protection des données (GDPR, CCPA) est requise.

Documentation et Présentation

Rapports Techniques
Les rapports techniques détaillent la méthodologie complète, les outils et versions utilisés, les commandes et paramètres exacts, et tous les artefacts analysés avec leurs emplacements. Les findings sont présentés avec les données brutes supporting, les interprétations proposées, et les niveaux de confiance. Les limitations et incertitudes sont explicitement déclarées.

Analyse complementaire

Executive Summaries
Les executive summaries traduisent les findings techniques en impact business. Les conclusions clés sont présentées sans jargon technique, focalisées sur le qui, quoi, quand, où, et comment. Les implications pour l'organisation et les recommandations actionnables sont prioritaires. Les visualisations (timelines, diagrams) facilitent la compréhension.

Annexes Techniques
Les annexes techniques préservent les détails pour revue experte. Les exports complets des clés pertinentes, les logs d'analyse, les scripts utilisés, et les données de validation sont inclus. Cette documentation permet la validation indépendante et supporte les challenges légaux potentiels.

Peut-on recuperer des cles de registre supprimees ?

Oui, la recuperation de cles de registre supprimees est possible grace a l'analyse des fichiers transaction logs (.LOG1, .LOG2) et des dirty pages. Des outils comme Registry Explorer de Zimmerman ou RegRipper permettent d'extraire des traces d'entrees supprimees, offrant des preuves precieuses en investigation forensique.

Combien de temps les artefacts registre sont-ils conserves ?

La duree de conservation des artefacts registre varie selon le type d'artefact et l'activite du systeme. Les timestamps MRU et UserAssist persistent jusqu'a la reinitialisation du profil, tandis que les traces de services desinstalles peuvent persister indefiniment dans les ruches systeme si elles ne sont pas nettoyees manuellement.

Quels outils open source utiliser pour Registry Forensics : Guide Expert Analyse Securite ?

Les incontournables sont Autopsy, Volatility 3, Plaso/log2timeline et RegRipper. Ils couvrent l'analyse disque, mémoire, timeline et registre sans coût de licence.

Pour approfondir, consultez les ressources officielles : SANS White Papers, NVD - NIST et ANSSI.

Sources et références : SANS SIFT · MITRE ATT&CK

Articles connexes

- Telemetry Forensics - Guide Pratique Cybersecurite
- Network Forensics : Analyse PCAP Avancee : Guide Complet

Conclusion et Perspectives Futures

L'analyse forensique du registre Windows demeure un pilier fondamental de l'investigation numérique, évoluant continuellement avec les nouvelles versions de Windows et les techniques d'attaque élaborées. La maîtrise de cette discipline exige non seulement une compréhension technique approfondie, mais aussi une approche méthodologique rigoureuse et une conscience des pièges d'interprétation.

Les développements futurs promettent d'enrichir encore les capacités d'analyse. L'intégration du machine learning pour la détection d'anomalies, l'automatisation accrue via des frameworks d'orchestration, et l'amélioration des techniques de récupération de données supprimées étendent continuellement les possibilités investigatives. Parallèlement, les défis émergent avec les nouvelles techniques anti-forensiques, le chiffrement accru, et les architectures cloud hybrides compliquant l'accès aux artefacts traditionnels.

L'investigateur forensique moderne doit maintenir une veille technologique constante, adapter ses méthodologies aux évolutions de Windows, et développer une expertise approfondie dans l'interprétation nuancée des artefacts. La corrélation multi-sources, la validation rigoureuse, et la documentation méticuleuse restent les fondements d'une investigation réussie et légalement défendable.

Le registre Windows, dans sa complexité et sa richesse, continue d'offrir une fenêtre incomparable sur les activités système et utilisateur. Sa maîtrise représente non seulement une compétence technique essentielle, mais aussi un art nécessitant expérience, intuition, et rigueur méthodologique. Pour l'investigateur déterminé, il reste une source inépuisable de vérité numérique, révélant les secrets les plus profonds des systèmes Windows et les actions de ceux qui les utilisent.

Ressources open source associées :

- awesome-cybersecurity-tools — Liste de 100+ outils de cybersécurité

Ayi NEDJIMI Consultants — Expert cybersécurité offensive & intelligence artificielle

ayinedjimi-consultants.fr · ayi@ayinedjimi-consultants.fr