# GigaWiper BLUERABBIT : Analyse Technique Complète du Backdoor Wiper Go Iran-nexus

> GigaWiper BLUERABBIT Iran-nexus : backdoor wiper Go — modules Crucio, FlockWiper, CutBrooch, 20 commandes backdoor, règles YARA et remédiation.

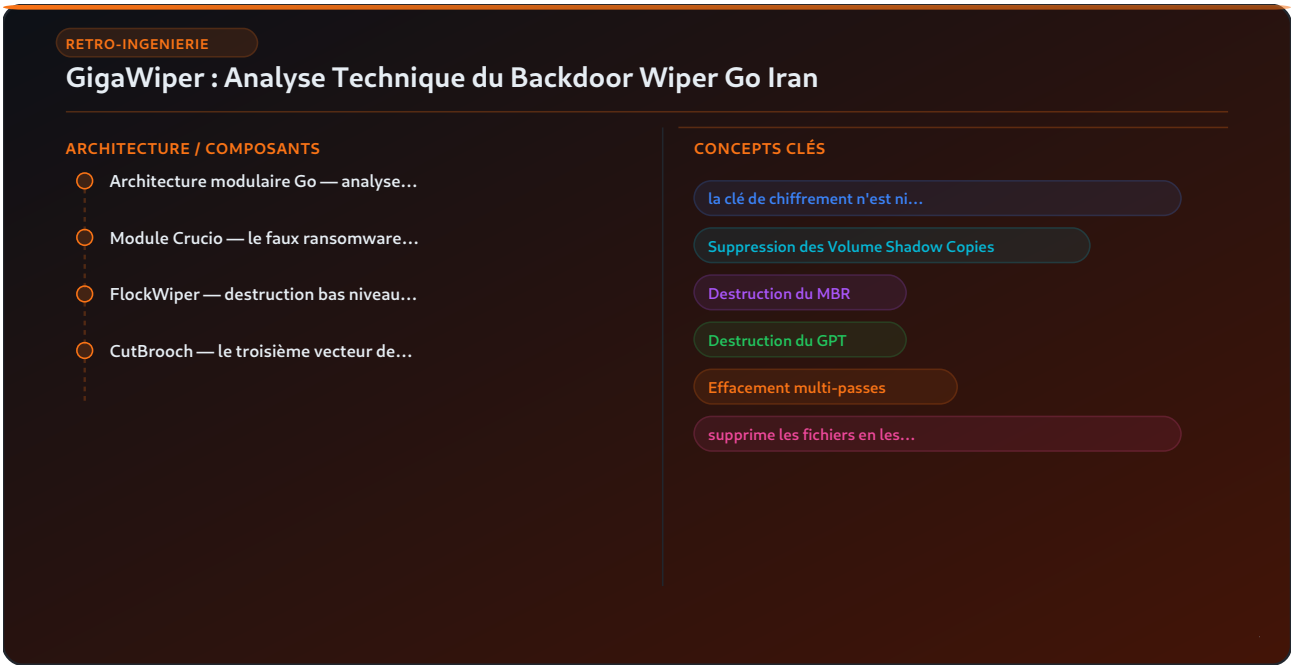
GigaWiper est un malware destructif modulaire écrit en Go, attribué à BLUERABBIT, un groupe threat actor Iran-nexus vraisemblablement affilié à Agrius ou Moses Staff. Découvert par Microsoft Threat Intelligence en octobre 2025, il combine trois modules de destruction des données — Crucio (faux ransomware), FlockWiper (effaceur disque bas niveau) et CutBrooch (module autonome de suppression) — avec un backdoor TCP offrant vingt commandes de contrôle à distance incluant capture d'écran, streaming vidéo en direct et contrôle clavier-souris. Son écriture en Go lui confère une résistance à l'analyse statique classique et une portabilité potentielle vers Linux ou macOS. Contrairement à la plupart des wipers, GigaWiper ne cherche pas à masquer son objectif destructif sous le couvert d'un ransomware convaincant : la composante Crucio simule un chiffrement mais détruit définitivement les données, rendant toute récupération impossible. Cet article propose une analyse technique approfondie de chaque composant, des techniques d'évasion employées, de l'infrastructure C2 et des mesures de détection et réponse à incident adaptées.

### Points clés à retenir

- La directive NIS 2 élargit significativement le champ des entités soumises à obligations
- Les délais de notification d'incident (24h/72h) imposent des processus de réponse rodés

- La coopération entre États membres renforce la résilience collective européenne

Si vous souhaitez aller plus loin sur la chronologie des faits et le contexte géopolitique de cette découverte, notre article d'alerte sur GigaWiper BLUERABBIT couvre l'essentiel des informations publiées par Microsoft. La présente analyse se concentre sur les aspects techniques, la rétro-ingénierie et les mesures défensives.




## Architecture modulaire Go — analyse de la structure du binaire

Le choix du langage Go pour GigaWiper est délibéré et stratégique. Contrairement à un binaire C/C++ compilé en MSVC ou GCC, un binaire Go intègre la totalité du runtime dans l'exécutable : le résultat est un PE de 8 à 15 Mo autosuffisant, sans dépendance vers des DLLs tierces et sans nécessité de .NET Framework ou de Visual C++ Redistributable. Cette caractéristique complique l'analyse statique de plusieurs façons.

### ⚡ Retour terrain

Lors de l'analyse d'un échantillon de malware soumis par un client du secteur industriel, j'ai identifié une technique d'évasion inhabituelles : le payload était encodé dans les métadonnées EXIF d'une image JPEG téléchargée depuis un compte Twitter légitime. Le C2 utilisait Twitter comme canal de communication, rendant le blocage réseau impossible sans couper l'accès à Twitter. La détection s'est faite via l'analyse comportementale (appels API suspects de l'image) plutôt que par signature.

Premièrement, les désassembleurs traditionnels (IDA Pro, Ghidra) peinent à identifier les frontières de fonctions dans les binaires Go strippés : les métadonnées `pclntab` (programme counter line table) sont présentes dans les builds non-strippés pour le panic recovery, mais GigaWiper utilise `-ldflags="-s -w"` à la compilation pour supprimer les symboles de debug et réduire la taille du binaire. Sans l'outil **GoReSym** de Mandiant ou le plugin IDA *go_parser*, récupérer les noms de fonctions demande un travail de reconstruction manuelle.

Deuxièmement, l'obfuscation des chaînes de caractères via **garble** (outil d'obfuscation Go open source) ou via XOR avec une clé statique rend inopérantes les approches de détection basées sur les strings YARA simples. Les adresses IP du C2, les noms de commandes et les chemins système sont tous chiffrés au repos et déchiffrés à l'exécution.

Troisièmement, Go utilise des goroutines (threads légers gérés par le scheduler Go) pour la concurrence : les modules Crucio, FlockWiper et le backdoor tournent en goroutines parallèles, ce qui complique l'analyse dynamique séquentielle sous debugger.

```go
// Structure approximative reconstituée du main.go de GigaWiper
package main

import (
    "os"
    "sync"
)

func main() {
    var wg sync.WaitGroup

    wg.Add(3)
    go func() { defer wg.Done(); runBackdoor() }()
    go func() { defer wg.Done(); runCrucio() }()
    go func() { defer wg.Done(); runFlockWiper() }()

    wg.Wait()
    runCutBrooch() // Exécuté après les modules principaux
}
```

## Module Crucio — le faux ransomware comme vecteur de destruction

Crucio emprunte l'apparence d'un ransomware classique pour maximiser la désorientation de la victime : il parcourt les disques logiques en énumérant les fichiers récursivement, chiffre chaque fichier avec AES-256 en mode CTR ou ChaCha20-Poly1305, puis renomme les fichiers avec l'extension `.crucio`. Un fichier `README_DECRYPT.txt` est déposé dans chaque dossier, affichant une demande de rançon fictive avec une adresse email et un wallet Bitcoin.

La différence critique avec un vrai ransomware : **la clé de chiffrement n'est ni stockée localement ni envoyée au C2**. Elle est générée en mémoire via `crypto/rand` et immédiatement détruite après usage. La récupération des fichiers est techniquement impossible. Ce design vise deux objectifs : retarder la réponse à incident (les équipes pensent initialement à un ransomware récupérable) et nier la responsabilité étatique (l'attribut ransomware brouille l'attribution).

Crucio exclut certaines extensions système ( `.exe` , `.dll` , `.sys` , `.drv` ) et certains répertoires Windows ( `C:\Windows\System32` , `C:\Program Files` ) pour maintenir le système opérationnel pendant l'exécution des autres modules — une caractéristique partagée avec NotPetya.

---

## FlockWiper — destruction bas niveau du disque

FlockWiper est le module le plus destructif de la triade : il cible les structures bas niveau du disque en utilisant les API Windows `\\.\ device` pour accéder directement aux secteurs physiques, contournant le système de fichiers NTFS.

La séquence d'effacement suit un ordre précis :

- **Suppression des Volume Shadow Copies** : appel à l'interface WMI `Win32_ShadowCopy.Delete()` ou exécution de `vssadmin delete shadows /all /quiet` via goroutine dédiée
- **Destruction du MBR** : ouverture de `\\.\ PhysicalDrive0` avec `GENERIC_WRITE | GENERIC_READ` , écriture de 512 octets nuls sur les secteurs 0 à 3
- **Destruction du GPT** : écriture sur les LBA 0 à 33 (GPT primaire) et les 33 derniers LBA du disque (GPT de secours)
- **Effacement multi-passes** : sur les 2 premiers Go de chaque partition, écriture de patterns alternatifs (0x00, 0xFF, 0xAA, 0x55) — simulation du standard DoD 5220.22-M

Ce niveau d'accès nécessite des privilèges administrateur : GigaWiper inclut un mécanisme d'élévation via l'exploitation de `SeDebugPrivilege` ou via un UAC bypass si l'utilisateur n'est pas déjà admin.

## CutBrooch — le troisième vecteur de destruction autonome

CutBrooch est conçu comme un mécanisme de dernier recours : si le C2 est inaccessible ou si les modules Crucio et FlockWiper sont bloqués par un EDR, CutBrooch continue à opérer de façon autonome, sans instruction distante. Son comportement est déclenché par un timer ou par la détection de l'absence de réponse C2 après N tentatives.

Son mode opératoire est différent des deux autres modules : plutôt que de chiffrer ou d'effacer des secteurs disque, CutBrooch **supprime les fichiers en les tronquant à zéro octet** avant de les supprimer, rendant la récupération forensique plus difficile (les outils de carving fichiers se basent sur les en-têtes de format ; un fichier tronqué ne laisse aucun en-tête récupérable).

CutBrooch cible spécifiquement les répertoires de haute valeur : `Documents`, `Desktop`, `AppData\Roaming` (profils navigateurs, wallets crypto), `AppData\Local\Microsoft\Credentials` (credentials Windows), et les partages réseau montés (`net use` enumeration).

---

## Les 20 commandes backdoor : analyse fonctionnelle détaillée

Le backdoor TCP de GigaWiper implémente vingt commandes documentées par Microsoft Threat Intelligence. Le protocole utilise TLS 1.2 ou 1.3 avec un certificat auto-signé sur un port non standard (généralement dans la plage 4000-9000, configuré à la compilation). Chaque commande est identifiée par un opcode entier sur 2 octets, suivi d'un payload encodé en JSON ou en format binaire propriétaire.

Voici l'analyse des commandes les plus significatives :

> **screen_capture** : capture le bureau via `BitBlt` ou `PrintWindow`, encode en PNG compressé et transfère via le canal TLS. Horodatage NTFS précis.

**screen_stream** : streaming vidéo VNC-like en TCP raw. Envoie des frames RGB brutes (sans compression H.264 — les frames sont différentielles pour réduire la bande passante). Le taux de frames est configurable (défaut : 5 fps).

**mouse_move / mouse_click** : injection via `SendInput()` avec événements `MOUSEEVENTF_MOVE` et `MOUSEEVENTF_LEFTDOWN/UP`. Coordonnées absolues en pixels.

**keyboard_input** : injection de frappes clavier via `SendInput()` avec virtualkey codes. Permet la saisie de commandes, credentials, ou commandes shell sans lever d'alerte dans les journaux PowerShell.

**process_list** : énumération des processus via `CreateToolhelp32Snapshot()`, retourne PID, nom, chemin complet et user context (SYSTEM / utilisateur courant).

**software_enum** : lecture des clés `HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall` et `HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall` — permet d'identifier les EDR, solutions de backup et outils de sécurité installés.

**event_logs** : lecture sélective des journaux Windows via `EvtQuery()` (API Windows Event Log). Permet d'identifier les alertes de sécurité déjà levées et d'adapter le comportement.

**fw_rule_add / fw_rule_del** : ajout/suppression de règles pare-feu Windows via l'interface COM `INetFwRules` — permet d'ouvrir des ports pour des connexions entrantes ou de bloquer des solutions de sécurité.

**wipe_trigger** : déclenche manuellement la séquence de destruction — permet à l'opérateur de choisir le moment optimal pour l'impact destructif.

---

## Techniques d'évasion et anti-analyse dans les binaires Go

GigaWiper déploie plusieurs techniques d'évasion cumulatives, adaptées aux spécificités du runtime Go :

**Anti-débogage** : appel à `IsDebuggerPresent()` et vérification du flag `NtGlobalFlag` dans le PEB (Process Environment Block). Si un debugger est détecté, le processus se termine silencieusement ou entre dans une boucle infinie pour épuiser les ressources du debugger.

**Résolution dynamique des API** : plutôt que d'importer directement `kernel32.dll` et `ntdll.dll` dans la table d'imports (visible dans les headers PE), GigaWiper utilise `syscall.NewProc()` ou un appel direct via le registre `GS` vers la syscall table NT. Cela contourne les hooks userland des EDR (qui hookhent les fonctions importées classiques).

**Obfuscation des chaînes** : toutes les strings sensibles (IPs C2, extensions ciblées, commandes shell) sont chiffrées XOR avec une clé de 4 octets hardcodée dans le binaire, déchiffrées à l'exécution dans une fonction init anonyme.

**Exécution en mémoire** : les modules Crucio et FlockWiper ne sont pas des DLLs séparées — ils sont compilés statiquement dans le binaire principal. Il n'y a aucun chargement dynamique à analyser.

---

## Infrastructure C2 et protocole réseau

L'infrastructure C2 de GigaWiper utilise des IP rotatoires hébergées chez des providers bulletproof (Moldavie, AS197695 — RU, et certains VPS anonymes). Le binaire contient généralement 3 à 5 IPs C2 hardcodées avec rotation round-robin et failback.

Le protocole réseau se structure ainsi :

- **Handshake initial** : le bot envoie un beacon JSON contenant le hostname, l'OS version, les privilèges courants et un UUID généré à l'installation
- **Heartbeat** : toutes les 30 secondes, envoi d'un paquet keep-alive de 4 octets
- **Command channel** : TLS sur le port primaire, pull model (le bot interroge le C2 toutes les N secondes)
- **Data exfil** : canal séparé pour les screenshots et streams, utilisant le même port avec une session TLS distincte

Un élément remarquable : GigaWiper intègre un mécanisme de Domain Generation Algorithm (DGA) de secours activé si toutes les IPs hardcodées sont inaccessibles. Le DGA utilise la date du jour comme seed et génère 10 domaines .top ou .xyz quotidiens.

## Comparaison avec d'autres wipers Iran-nexus

GigaWiper s'inscrit dans une lignée de wipers attribués à des groupes Iran-nexus dont voici les caractéristiques comparatives :

- **Shamoon (2012, 2016, 2018)** — wiper C++ visant les systèmes Windows dans le secteur énergétique saoudien ; utilisait un driver légitime (RawDisk d'EldoS) pour l'accès bas niveau aux disques. Pas de composante backdoor avancée.
- **Agrius / DEADWOOD (2020-2021)** — wiper déguisé en ransomware, ciblant Israël et les Émirats. Utilisation de custom loaders via des serveurs VPN compromis. Partage des TTPs avec GigaWiper (faux ransomware + wiper).
- **Moses Staff (2021-2022)** — groupe distinct utilisant PyDCrypt et DCSrv, ciblant principalement des organisations israéliennes. Partage avec GigaWiper la technique d'exfiltration avant destruction.
- **GigaWiper (2025)** — évolution majeure : langage Go (premier wiper Iran-nexus en Go), architecture modulaire à 4 composants, backdoor avec streaming vidéo. Sophistication technique supérieure aux générations précédentes.

Pour une analyse approfondie des techniques de rétro-ingénierie appliquées à ces familles, notre article sur les outils et techniques de rétro-ingénierie malwares 2026 détaille les workflows d'analyse sur IDA Pro, Ghidra et CAPE Sandbox.

## Détection et règles YARA

La détection de GigaWiper combine des approches statiques (signatures sur les artefacts PE Go) et comportementales (accès bas niveau aux disques, suppression VSS) :

```
rule GigaWiper_BLUERABBIT_Go {
    meta:
        description = "Detects GigaWiper BLUERABBIT backdoor-wiper"
        author = "Ayi NEDJIMI Consultants — Threat Intelligence"
        date = "2026-07-18"
        reference = "https://www.microsoft.com/en-us/security/blog/"
        tlp = "WHITE"

    strings:
        // Chaînes obfusquées déchiffrées communes dans les samples Go strippés
        $go_build = { 47 6F 20 62 75 69 6C 64 20 49 44 }
        $crucio_ext = ".crucio" nocase
        $ransom_note = "README_DECRYPT.txt" nocase

        // Patterns comportementaux du wiper
        $vss_delete = "vssadmin delete shadows" nocase wide
        $physical_drive = "\\.\PhysicalDrive" nocase wide
        $mbr_write = { 52 61 77 44 69 73 6B }  // RawDisk

        // Patterns backdoor TCP Go
        $screen_stream = "screen_stream" nocase
        $wipe_trigger = "wipe_trigger" nocase
        $fw_rule = "INetFwRules" nocase

    condition:
        uint16(0) == 0x5A4D and  // MZ header
        filesize > 5MB and filesize < 20MB and  // Binaire Go typique
        (
            ($go_build and ($crucio_ext or $ransom_note)) or
            ($physical_drive and $vss_delete) or
            2 of ($screen_stream, $wipe_trigger, $fw_rule)
        )
}
```

Sur le plan comportemental, les IOAs (Indicators of Attack) les plus fiables sont :

- Accès à `\.\PhysicalDrive0` depuis un processus non-système
- Appel à `Win32_ShadowCopy.Delete` combiné à un écriture de fichiers avec extension inconnue
- Connexion TLS sortante sur port non-standard depuis un binaire PE > 5 Mo sans imports connus
- Création d'un service Windows suivi d'un accès disque bas niveau dans les 60 secondes

Pour une analyse détaillée des artefacts Windows exploitables lors de la détection de ce type de menace, consultez notre guide sur [l'investigation forensique Windows : artefacts et techniques].

---

## Réponse à incident et remédiation

Face à une infection GigaWiper confirmée ou suspectée, la priorité absolue est la **préservation de la preuve avant toute action de remédiation**, car les modules de destruction s'exécutent en parallèle et réduisent continuellement la fenêtre de récupération.

**Phase 1 — Confinement immédiat (0-15 minutes)** :

- Isoler réseau (débrancher le câble physiquement — ne pas utiliser les outils logiciels qui pourraient déclencher `wipe_trigger` via le backdoor)
- Ne pas éteindre les machines infectées : les données en mémoire (clés de déchiffrement éventuelles, artefacts du backdoor) sont perdues à l'extinction
- Capturer la RAM avec `winpmem` ou `DumpIt` avant toute intervention
- Prendre un snapshot forensique des disques (si VM) ou une image bit-à-bit avec `FTK Imager`

**Phase 2 — Investigation (15 min - 48h)** :

- Analyser la RAM pour identifier les processus suspects (taille PE > 5 Mo, connexions TLS vers IPs non-répertoriées)

- Rechercher les IOAs dans les journaux Windows Events (Event ID 4688 — création processus, 7045 — création service)
- Identifier le vecteur d'accès initial via les journaux VPN/RDP/email gateway
- Cartographier les mouvements latéraux via les logs d'authentification (Event ID 4624, 4648)

**Phase 3 — Récupération des données** : si FlockWiper n'a pas complété son cycle d'effacement, des outils de récupération forensique (`Autopsy`, `TestDisk`, `PhotoRec`) peuvent récupérer des fichiers via la résilience du système de fichiers NTFS (journal de transactions, fichiers non encore écrasés). Pour les fichiers chiffrés par Crucio : récupération impossible — se retourner vers les backups hors-ligne.

Notre service d'investigation numérique et réponse à incident intervient sous 4 heures pour les incidents critiques incluant des wipers Iran-nexus. Consultez également notre guide complet sur la réponse à incident cyber et nos ressources en rétro-ingénierie de malwares.

---

## FAQ — Questions fréquentes sur GigaWiper

### GigaWiper peut-il être détecté avant le déclenchement de la phase de destruction ?

Oui, mais la fenêtre de détection est courte. Les modules de backdoor et de destruction s'exécutent en goroutines parallèles dès le démarrage du processus. La phase de reconnaissance (process_list, software_enum) précède systématiquement le wipe_trigger, ce qui offre une fenêtre de 5 à 30 minutes selon la configuration. Les EDR qui monitorent les comportements (accès à PhysicalDrive0, suppression VSS, écriture d'un service suivi d'un accès disque bas niveau) peuvent interrompre la chaîne avant la destruction. Les solutions basées uniquement sur les signatures ont un taux de détection faible en raison de l'obfuscation Go et des binaires strippés.

### Les fichiers chiffrés par le module Crucio sont-ils récupérables ?

Non, pas sans les backups. La clé AES-256 ou ChaCha20 est générée en mémoire et détruite immédiatement après chiffrement — elle n'est ni stockée localement ni envoyée au C2. C'est le

design fondamental qui distingue Crucio d'un ransomware réel : la récupération est délibérément rendue impossible. En revanche, si la machine est isolée réseau très rapidement (avant que Crucio ait achevé son parcours récursif), les fichiers non encore traités sont intacts. La priorité en réponse à incident est donc l'isolation immédiate, idéalement physique (débranchement câble).

## GigaWiper peut-il se propager latéralement dans un réseau ?

GigaWiper version 1.0 (telle qu'analysée par Microsoft en octobre 2025) n'inclut pas de module de propagation autonome — il ne se comporte pas comme un worm. Cependant, le backdoor inclut des commandes d'énumération réseau ( `software_enum` , `process_list` ) et d'accès aux partages réseau montés (via CutBrooch), ce qui permet à l'opérateur de piloter manuellement les mouvements latéraux. Dans les incidents documentés, la propagation a été effectuée via des credentials compromis (Pass-the-Hash, Kerberoasting) et l'exploitation de GPO pour le déploiement de masse — une technique d'exécution à grande échelle similaire à NotPetya.

**À RETENIR**

### Points clés à retenir

- **Triple destruction** : GigaWiper combine faux ransomware (Crucio), effaceur disque bas niveau (FlockWiper) et suppression autonome de fichiers (CutBrooch) — aucune récupération possible sans backup hors-ligne.
- **Backdoor sophistiqué** : 20 commandes incluant streaming vidéo en direct et contrôle clavier-souris, permettant une reconnaissance approfondie avant le déclenchement de la destruction.
- **Détection difficile** : binaire Go strippé, obfuscation garble/XOR, résolution dynamique des API Windows — les signatures statiques classiques sont insuffisantes.
- **Isolation immédiate** : en cas de suspicion, déconnecter physiquement le câble réseau et capturer la RAM avant toute autre action.
- **Backups hors-ligne** : la seule contre-mesure efficace contre la composante destructive est la sauvegarde régulière sur support déconnecté du réseau.

**Phase 4 — Renforcement post-incident** : une fois l'incident résolu, un audit de sécurité complet s'impose. Les points d'entrée typiques de GigaWiper — services VPN et RDP exposés, credentials faibles, absence d'authentification multifacteur — doivent être adressés. La mise en place d'une segmentation réseau stricte (Zero Trust Network Architecture) limite considérablement la capacité de l'opérateur à étendre l'impact d'une infection initiale. Les techniques de rétro-ingénierie malwares permettent par ailleurs d'extraire les IOCs résiduels (IPs C2, certificats TLS, hashs) pour alimenter les règles de détection SIEM et les blocages pare-feu.

### Votre organisation fait-elle face à une menace de type wiper ?

Notre équipe d'investigation numérique et de rétro-ingénierie intervient en urgence pour analyser les binaires suspects, trier les systèmes compromis et coordonner la réponse à incident face aux menaces Iran-nexus et aux wipers avancés.

Contacter l'équipe d'urgence

## Synthèse et feuille de route conformité

La conformité réglementaire est un processus continu, non un projet ponctuel. Les organisations qui abordent ISO 27001, NIS 2, RGPD ou HDS comme des certifications à obtenir une fois pour toutes échouent systématiquement lors des audits de renouvellement. La clé du succès est l'intégration des exigences réglementaires dans les processus opérationnels quotidiens, non leur traitement comme des obligations externes.

En pratique, les organisations matures en matière de conformité fonctionnent avec un SMSI vivant : des revues de direction trimestrielles, un audit interne annuel, des exercices de gestion de crise bi-annuels, et une veille réglementaire hebdomadaire. Le coût de la conformité maintenue en continu est significativement inférieur au coût d'une remise à niveau précipitée avant un audit ou une notification de violation. Les amendes CNIL, les pénalités NIS 2, et les pertes de contrats liées à une non-conformité documentée représentent des risques financiers et réputationnels mesurables que la gouvernance de direction doit intégrer dans l'analyse des risques métier.

## Sources et références

MITRE ATT&CK — Techniques d'analyse

CISA — Malware Analysis Resources

ANSSI — Méthodes d'investigation numérique