LIVRE BLANC

COMMENT DÉTECTER LES MENACES AVANCÉES ?

GATEWATCHER

Le marché de la sécurité a connu durant ces dernières années des révélations qui, bien que largement connues des initiés, ont permis toutefois de confirmer que **les attaques subies ne sont plus celles de simples pirates**, elles proviennent de puissances étrangères, d'agences gouvernementales, ou encore de sociétés privées.

Les conséquences sont importantes et nous réalisons, peut-être un peu tard, que **la défense du patrimoine informationnel est un enjeu majeur**. De nombreux contrats ont été perdus à cause d'un espionnage massif et de nombreuses technologies sont désormais copiées. En France la notion d'OIV (Organisme d' Importance Vitale) montre bien que **nos infrastructures doivent être protégées** au prix d'un effort renouvelé et coordonné de tous les acteurs concernés.

On constate cependant que la cyberdéfense couramment mise en œuvre s'appuie sur des paradigmes incorrects et des pratiques inadaptées :

- Elle fait **trop confiance** aux produits de sécurité régulièrement utilisés,
- Il est trop admis que les « amis » n'attaquent pas les « amis »,
- Une vision étriquée du fonctionnement d'une attaque ciblée prédomine,
- Les organisations fonctionnent en **mode réactif** faute de moyens mis en place et **l'analyse post mortem n'est qu'une fatalité**.

1.
LE SYSTÈME DE DÉTECTION TRACKWATCH®

Le système de détection Trackwatch®, édité par GATEWATCHER permet de lutter contre les attaques ciblées. Son historique permet de **comprendre les orientations technologiques de la solution**. A son origine, une question : **comment lutter contre les intrusions avancées et surmonter leur sophistication ?** Plusieurs voies semblaient ouvertes :

La Sandbox

Bien que très utilisée, la technologie sandbox est **plus un outil de forensic** destiné à l'investigation numérique légale qu'un outil automatisé de détection de malwares. En effet, il est **très aisé de contourner une sandbox** et de plus cette technique génère de nombreux faux positifs. La sandbox a vécu son heure de gloire, mais elle est **désormais totalement dépassée** et même dangereuse car **elle peut induire en erreur**. D'autre part elle ne peut par définition analyser que ce qui est exécutable ; **or les attaques avancées ne sont pas uniquement constituées de malwares**, elles sont souvent basées sur des payloads. Le malware est bien souvent un élément utilisé uniquement **pour la persistance**, d'où la notion d'APT (Advanced Persistent Threat).

L'analyse statistique réseau

Analyse Bayésienne, modélisation mathématique, machine learning … Le prototypage de ces techniques n'a pas été retenu pour les raisons suivantes :

- **L'analyse basée sur des probabilités d'apparition** (notamment bayésienne) est applicable à des segments « standard » d'un système d'information, autrement dit c'est **une technique extrêmement efficace** à partir du moment où les protocoles utilisés sont très standardisés et où **les attaques sont très prédictibles** ou ont tout du moins un objectif connu. C'est une **technique parfaite pour le SPAM** : les protocoles de messagerie ont peu évolué, sont standards (MIME/SMTP, POP, etc), et les objectifs des « spammeurs » sont assez communs (vente de Viagra, virements d'argent, etc…). Mais **face à une attaque ciblée qui est par définition non prédictible**, avec des objectifs inconnus, sur des protocoles qui ne sont pas forcément standards, le nombre de « **faux positifs** » entraînés par cette technique sera très élevé.

- **L'auto-apprentissage machine learning** est à priori séduisant car il promet des résultats intéressants en matière de détection. Cependant il faut **revenir aux bases d'une attaque ciblée** :
  - **l'attaque est discrète** car elle adopte généralement les modes de fonctionnement internes ;
  - les **déviations par analyse comportementale** concernent la plupart du temps la phase d'exfiltration des données s'il y en a une…. En résumé, c'est la fin de l'attaque. Il est alors déjà trop tard !
  - les **systèmes non standard ou évoluant régulièrement empêchent l'apprentissage**.

Prenons un exemple concret

Un pirate exploite une vulnérabilité 0-Days sur un serveur Apache. Il envoie une requête forgée contenant un shellcode lui donnant un reverse shell. Que verra réellement un système d'auto apprentissage ?

- La requête apparaît **valide**,
- **Le shellcode est indéchiffrable** car la plupart du temps il est **encodé,**
- Le reverse shell peut parfaitement s'assimiler à une connexion SSH. **En quoi est-ce suspicieux ?** Tous les administrateurs le pratiquent.

En bref, un système de machine learning ne verra rien contre une attaque avancée, ou au contraire, alertera sans arrêt. C'est éventuellement **un bon complément pour l'investigation post mortem, mais cela ne permet pas de bâtir un système fiable pour la détection des intrusions avancées.**

L'analyse par signatures

Aussi appelée **Threat Intelligence** par les équipes marketing, éprouvée durant ces dernières années, l'analyse par signature **a fait ses preuves contre les attaques classiques**. En revanche une attaque avancée, souvent à base de **0-Days, ne pourra par définition pas avoir de signature**. Les signatures n'ont donc qu'une utilité à posteriori dans le cas d'attaque avancée, mais restent cependant **un bon moyen de guetter les potentielles erreurs d'un attaquant** (déploiement de payload connus, etc…).
Les techniques existantes n'apportaient donc rien de convainquant contre les attaques avancées. GATEWATCHER a donc été conçu sur **des bases technologiques totalement différentes**.

Utiliser du machine learning sur un Cloud serait une hérésie...

Philippe Gillet (CTO, Gatewatcher)

2. PEUT-ON RÉELLEMENT FAIRE FACE AUX 0-DAYS ?

L'attaque avancée est très souvent **basée sur des 0-Days**. Il convient donc de se demander **comment détecter efficacement un 0-Days**. Lorsque l'on analyse les offres du marché on est surpris de constater que certaines solutions prétendent détecter **100% des 0-Days**. Cela est évidemment une contrevérité, **les vulnérabilités sont intrinsèques à tous les produits** et elles existeront toujours, il faut l'admettre. S'il en était autrement pourquoi ces mêmes solutions recommandent-elles, voire imposent-elles, des mises à jour ? **Le système infaillible relève du domaine du rêve**.

Faut-il pour autant penser que **le combat est perdu d'avance ?** Non, il faut simplement connaître les types d'attaques et les contre-mesures qui peuvent **prémunir les environnements sensibles**. Et pour partir sur de bonnes bases, il convient de reprendre la plupart des attaques de type 0-Days et trouver **un dénominateur commun**. Aujourd'hui, la plupart des 0-Days utilisent les techniques suivantes :

- Une exploitation type **ROP (Return Oriented Programming)** et dérivés (sROP – BROP – JOP…),
- Des exploits type **RCE** (Remote Code Exec),
- Une exploitation simple à base de **shellcode (encodé ou non)**,
- Des exploits type **Race Condition / Format String**,
- …

On considère actuellement qu'**environ 60 à 70% des exploits utilisent à minima des techniques type shellcodes/ROP**. Détecter ce type de technique peut permettre de repérer réellement des 0-Days. Mais **aucune signature n'existe contre les techniques listées plus haut**.

Quelques liens utiles renseignent sur ces techniques :

Exploit-DB

Stanford SCS : BROP

3. VULNÉRABILITÉS, ATTAQUES ET DÉFENSE

Nous n'évoquerons pas les vulnérabilités humaines qui ne peuvent être résolues que par une sensibilisation appropriée et la mise en œuvre de bonnes pratiques. En revanche nous aborderons la principale crainte des responsables de la sécurité : **la faille, la vulnérabilité** qui, malgré toutes les précautions, est présente et **prête à être exploitée**. Tous les patchs de sécurité ont été appliqués, tous les systèmes sont à jour, le réseau est équipé des meilleurs firewalls, une organisation efficace a été mise en place, des consultants ont validé l'ensemble… **Mais cela ne suffira pas !**

Pourquoi ? Nous avons déjà livré la réponse : **par principe un 0-Days agit sur des systèmes à jour et patchés**. Il fonctionnera même sur des systèmes durcis, ce n'est qu'une question de temps…

Pour lutter contre une attaque ciblée, une seule technologie n'est pas suffisante. Le système de détection Trackwatch®, édité par GATEWATCHER n'est pas magique. Mais nous verrons comment **la solution peut agir à certains niveaux en complémentarité avec d'autres contre-mesures**.

Voici un tableau de synthèse non exhaustif **des attaques de types 0-Days et leurs contre-mesures** :

Techniques d'attaques et de compromission	Sondes IDS Next Gen- IDS	Firewall - WAF	Proxy	HIPS (agent)	Antivirus
ATTAQUES SYSTÈME + APPLICATIF					
attaques post 0-Days	+++++	N/A	N/A	++++	++
RCE (Remote Code Exec) avec 0-days	+++	N/A	N/A	+++++	+
Attaque type Code Reuse	+++++	N/A	N/A	+++	++
vulnérabilités type format string	+++++	N/A	N/A	+++++	+
exploitation vulnérabilité type race conditions	?	N/A	N/A	?	?
ATTAQUES WEB					
attaques post 0-Days	+++++	+++++	N/A	++	N/A
vulnérabilités sqli / nosql	++	+++++	N/A	++	N/A
vulnérabilités xss / csrf	++	+++++	N/A	++	N/A
vulnérabilités directory traversal	+++	+++++	N/A	+++	N/A
vulnérabilités LFI/RFI	++	+++++	N/A	++	N/A
vulnérabilités XML Injection	++	+++++	N/A	++	N/A
Malwares					
malware	++++	N/A	+++	+++	+++++
malware avancé (vuln embeded, stealth, etc)	+++++	N/A	++++	++++	++

Ce tableau est simplifié, car **certaines contre-mesures sont hybrides**, (comme les firewalls faisant fonction d'IDS, les proxy faisant antivirus, etc). Sa lecture permet cependant plusieurs remarques :

- L'IDS/BDS et l'HIPS sont les technologies les plus à même de **contrer des attaques avancées au niveau des systèmes et applicatifs**,
- Seul **le WAF est adapté aux technologies Web** et peut parer les attaques, via des règles plus ou moins élaborées (PCRE/REGEX),
- **L'antivirus ne sert à rien contre les malwares avancés**,
- **Le proxy est très utile contre les malwares avancés**. Il peut combattre des droppers, des URL malveillantes, et intègre facilement des IOC (Indicator Of Compromise) de type IP blacklistées,
- L'HIPS est certainement **la dernière ligne de défense**. Cependant les **contraintes de déploiement et de maintenance** en conditions opérationnelles rebutent de nombreuses entreprises. Il s'agit d'un moyen **à utiliser avec modération**.

La **corrélation** est une surcouche applicable aux techniques citées ci-dessus. Cependant **elle ne constitue pas une contre-mesure en soit** et il faut distinguer les outils qui produisent des logs de ceux qui effectivement agrègent et corrèlent les événements. **Une fausse croyance consiste à imaginer qu'un produit de corrélation peut découvrir simplement des attaques avancées, ou qu'il est par lui-même un outil de détection.**

Un SIEM s'avèrera inefficace si les événements traités ne sont pas pertinents ou complexes à relier entre eux.

Philippe Gillet (CTO, Gatewatcher)

4.
LE FONCTIONNEMENT DE TRACKWATCH

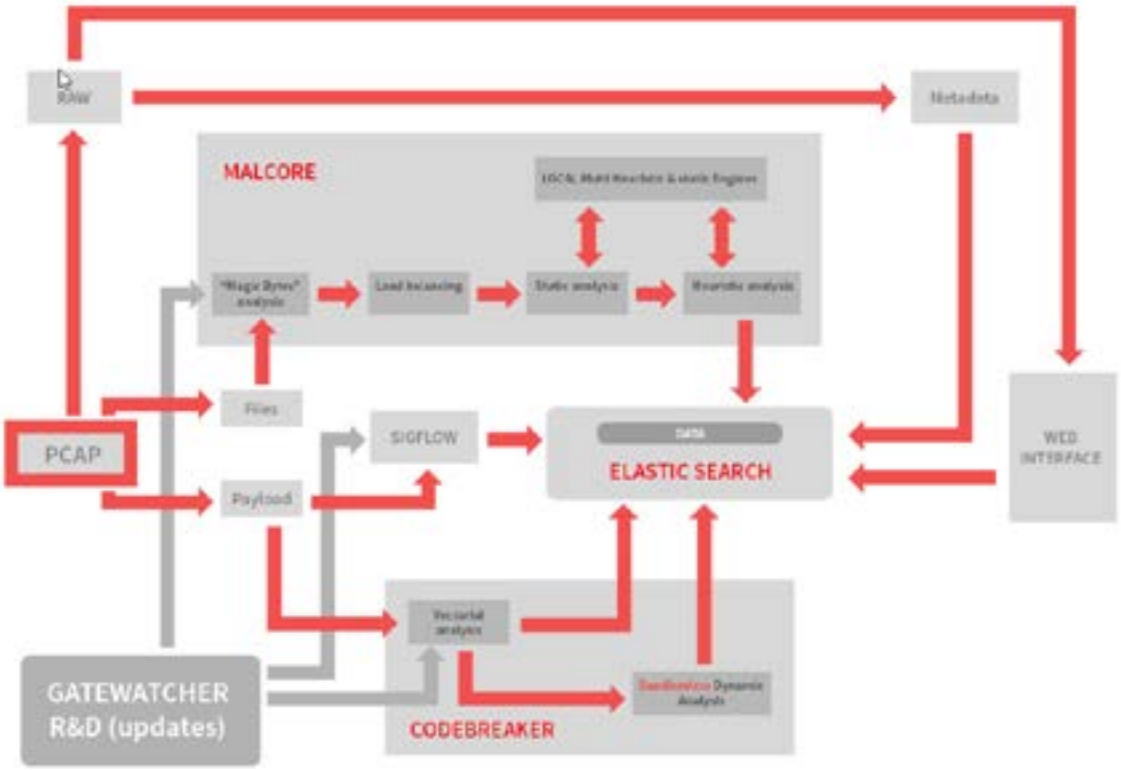
Alors comment détecter les attaques avancées ? L'approche de GATEWATCHER provient de la réponse à une unique question : **bien qu'il n'y ait aucun point commun entre toutes les vulnérabilités, quel est celui entre tous les 0–Days ?** La réponse est simple : ce sont **les techniques d'exploitation**. En effet, si l'on étudie l'historique du problème posé, on remarque que la plupart des 0-Days systèmes et applicatifs, hors applicatifs web, sont exploités par **des techniques dont les fondements varient peu**. Ces techniques évoluent, deviennent de plus en plus complexes du fait notamment du **durcissement des environnements ciblés**, mais elles retiennent sur le fond un même principe simple : **manipuler la mémoire pour en prendre le contrôle.**

Les techniques d'exploitation permettant d'opérer un tel contrôle sont devenues si complexes qu'il est **impossible de créer des signatures**. L'époque où l'on pouvait signer les shellcodes ou détecter avec snort des attaques type buffer overflow via des séries de NOP est révolue ; elle nous renvoie dix années en arrière ! C'est pourquoi le système de détection Trackwatch® adresse **l'ensemble du spectre fonctionnel des techniques d'attaques**, à base de fichiers (principalement via des malwares) ou s'appuyant sur des payloads. Dans cet objectif TRACKWATCH est composé de quatre moteurs, deux concernant les payload, et deux pour le traitement des fichiers :

Sur les payload
- SIGFLOW
- CODEBREAKER

Sur les fichiers
- MALCORE
- RETROACT

Voici le schéma résumant **le mode de fonctionnement de TRACKWATCH** :



GATEWATCHER est également **adapté de par sa conception aux environnements confinés** (CD-SD-SO-DR-DRSF, etc), pour ce type d'environnement l'accès aux mises à jour ne peut se faire directement sur Internet.

Par ailleurs, mettre en place un système de détection qui voit transiter **l'ensemble des flux et des fichiers** et qui envoie tout ou une partie dans un cloud externalisé est dans certains cas une hérésie sur le plan de la sécurité : **l'ensemble des analyses peuvent alors être réalisées localement par Trackwatch** sans que rien ne sorte du système installé chez le client. Les mises à jour peuvent être effectuées par des interventions hors connexion afin d'**éviter la communication de Trackwatch avec Internet**.

GATEWATCHER

DÉTECTION EN **TEMPS RÉEL** DE MENACES AVANCÉES

www.gatewatcher.com