Scenario

Notre client a été victime d'une attaque de défiguration sur son site internet. Le client utilise Splunk comme SIEM. Notre rôle est de mener une enquête afin d'identifier l'auteur de l'attaque et de prendre les mesures nécessaires.

Investigation d'une Attaque avec methode cyber kill chain

1. Reconnaissance

Objectif : Identifier comment l'attaquant a collecté des informations sur la cible avant l'attaque.

Vu que c'est un attack sur un site internet la premiere chose que je vais faire c' est d' aller voir les requette http donc je fais une recherche sur Splunk avec cet fonction `index=botsv1 imreallynotbatman.com sourcetype="stream:http"`.

Dans la section src_ip (source ip) j'ai trouvé 2 ip qui ont envoyé beaucoup de requete

Valeurs	Nombre	%
40.80.148.42	17 483	93,402 %
23.22.63.114	1 235	6,598 %

Je prends note des deux IP et je vais examiner de plus près la première. Il est évident que cette IP appartient à l'attaquant, vu le nombre de requêtes, mais pour en être certain, je vérifie avec Suricata `index=botsv1 imreallynotbatman.com src_ip="40.80.148.42" sourcetype=suricata`.

10 premières valeurs	Nombre	%
ET WEB_SERVER Script tag in URI, Possible Cross Site Scripting Attempt	103	21,776 %
ET WEB_SERVER Onmouseover= in URI - Likely Cross Site Scripting Attempt	48	10,148 %
ET WEB_SERVER Possible XXE SYSTEM ENTITY in POST BODY.	41	8,668 %
SURICATA HTTP Host header invalid	35	7,4 %
ET WEB_SERVER Possible SQL Injection Attempt SELECT FROM	33	6,977 %
ET WEB_SERVER SQL Injection Select Sleep Time Delay	32	6,765 %
ET WEB_SERVER Possible CVE-2014-6271 Attempt	18	3,805 %
ET WEB_SERVER Possible CVE-2014-6271 Attempt in Headers	18	3,805 %
ET WEB_SERVER PHP tags in HTTP POST	13	2,748 %
GPL WEB_SERVER global.asa access	12	2,537 %

Comme vous pouvez voir il y a beaucoup de alert de suricata.

Cependant, la question que je me pose est : qu'est ce que l'attaquant veut faire ?

Dans la liste des informations je trouve quelque choose qui attire mon intérêt src_headers 100+

```
Acunetix-Product: WVS/10.0 (Acunetix Web
Vulnerability Scanner - Free Edition)
Acunetix-Scanning-agreement: Third Party
Scanning PROHIBITED
Acunetix-User-agreement:
http://www.acunetix.com/wvs/disc.htm
Accept: */*
```

J'ai trouvé de nombreuses requêtes provenant d'un scanner de vulnérabilités nommé Acunetix. Nous avons ainsi identifié comment l'attaquant a effectué sa phase de reconnaissance.

T1595 l'attaquant scanne les ports ouverts, les services et les failles de sécurité.

2. Weaponization

Objectif : Identifier les outils ou exploits utilisés par l'attaquant pour préparer son attaque.

Vous avons deux IP suspect et je veux voir plus claire donc je vais utiliser cet fonction
`index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" http_method=POST`

Avec ça, je peux voir toutes les requêtes envoyées au site web.
Dans le champ URI, je trouve un grand nombre de tentatives de connexion au compte administrateur.


```
index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" http_method=POST uri="/joomla/administrator/index.php" | table _time uri src_ip dest_ip form_data
```

Je **utilise** cette **fonction** pour me créer une liste des requêtes dans l'**URI** et les **IP**. Je trouve un nombre très élevé **d'essais** de mots de passe, donc **il s'agit d'une attaque par force brute**.

```
index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" http_method=POST form_data=*username*passwd* | rex field=form_data "passwd=(?<creds>\w+)" | table _time src_ip creds
```

Avec cette fonction, je vais filtrer et nettoyer les informations superflues. J'ai une liste complète des mots de passe utilisés pour le compte administrateur.

_time	src_ip	creds
2016-08-10 23:48:05.858	40.80.148.42	batman
2016-08-10 23:46:51.394	23.22.63.114	rock
2016-08-10 23:46:51.154	23.22.63.114	cool
2016-08-10 23:46:51.156	23.22.63.114	sammy
2016-08-10 23:46:50.873	23.22.63.114	august
2016-08-10 23:46:50.634	23.22.63.114	phantom
2016-08-10 23:46:50.627	23.22.63.114	williams
2016-08-10 23:46:50.621	23.22.63.114	private
2016-08-10 23:46:50.640	23.22.63.114	baby

Vu la vitesse de saisie des mots de passe, l'attaquant a utilisé un logiciel. Nous pouvons aussi voir qu'il y a eu une seule connexion avec l'IP 40.80.148.42, probablement utilisée pour s'identifier.

Ici, nous pouvons voir que l'attaquant a utilisé Python pour mener une attaque par force brute.

Je click sur Mozilla et je tombe sur ça

Dans la recherche effectuée précédemment, on a vu que c'était le dernier login et mot de passe dans la liste. Vu qu'il n'y en a plus d'autres, cela veut dire qu'il a trouvé les identifiants.

ID: T1110 brute force

---

3. Delivery

Objectif : Comprendre comment la charge utile (malware, exploit) a été livrée à la victime.

index=botsv1 sourcetype=suricata http_method=POST dest_ip="192.168.250.70" *.exe

Avec cet foncions j'ai regardé quel executable a été envoyé a la victime

J'ai trouvé que deux options /_vti_bin/shtml.exe et 3791.exe le premier semble être un fichier Microsoft FrontPage, le deuxième semble suspect. Je vais essayer de trouver la source avec cette fonction index=botsv1 sourcetype=suricata http_method=POST dest_ip="192.168.250.70" *.exe filename="3791.exe" et je tombe sur l'IP de l'attaquant 40[.]80[.]148[.]42 donc c'est un executable envoyé dans le site web.

À ce moment-là, je vais regarder les processus créés dans Sysmon pour trouver plus d'informations. Plus de détails dans la phase d'installation.

T1105 - Ingress Tool Transfer Téléversement d'un fichier malveillant via HTTP POST.

---

4. Exploitation

Objectif : Identifier comment la vulnérabilité a été exploitée pour compromettre le système.

L'attaquant a simplement trouvé comment accéder à la page back-end du site web, puis il a effectué une attaque par force brute. Après cela, il a téléversé le fichier 3791.exe, un malware de type backdoor.

ID: T1110 brute force

---

5. Installation

Objectif : Déterminer si un malware ou une porte dérobée (backdoor) a été installé.

index = botsv1 sourcetype="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational" 3791.exe J'utilise cette fonction pour rechercher les processus liés à l'exécutable et je vais examiner le champ **"commandline"**.

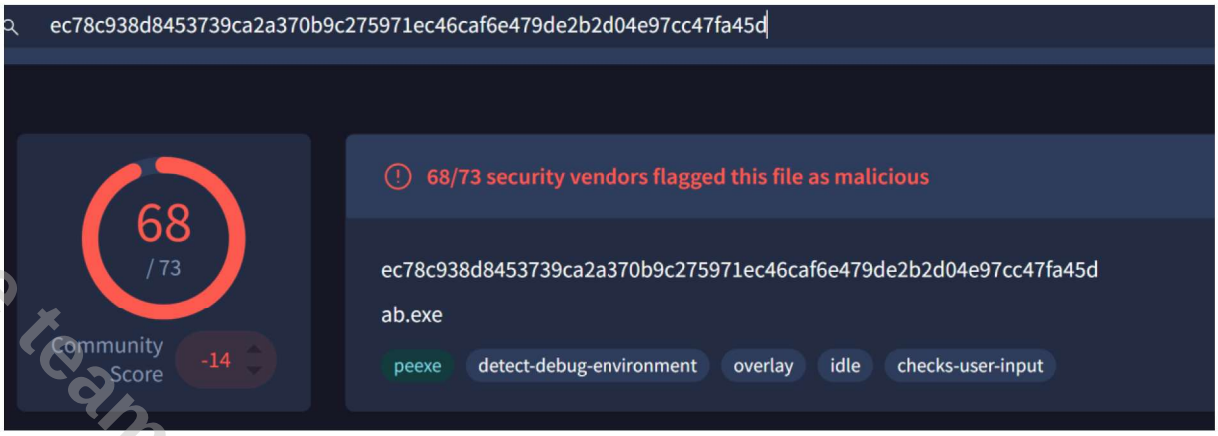
C:\Windows\system32\cmd.exe	2	40 %
3791.exe	1	20 %
\??\C:\Windows\system32\conhost.exe 0xffffffff	1	20 %
cmd.exe /c "3791.exe 2>&1"	1	20 %

ici je vois tout les action lié au executable, je peux deduire que il a été executé dans cmd. Je vais dans 3791.exe pour plus de details.

Type	Champ	Valeur
Sélectionné	CommandLine	3791.exe
	host	we1149srv
	source	WinEventLog:Microsoft-Windows-Sysmon/Operational
	sourcetype	XmlWinEventLog:Microsoft-Windows-Sysmon/Operational
Événement	Computer	we1149srv.waynecorpinc.local
	CurrentDirectory	C:\inetpub\wwwroot\joomla\
	EventChannel	Microsoft-Windows-Sysmon/Operational
	EventCode	1
	EventDescription	Process Create
	EventID	1
	Hashes	SHA1=65DF73D77324D008C83C3E57B445DF0FD43A3A51,MD5=AAE3F5A29935E6ABCC2C2754D12A9AF0,SHA256=EC78C938D8453739CA2A370B9C275971EC46CAF6E479DE2B2D04E97CC47FA45D,IMPHASH=481F47BBB2C9C21E108D65F52B04C448
	IMPHASH	481F47BBB2C9C21E108D65F52B04C448
	Image	C:\inetpub\wwwroot\joomla\3791.exe
	IntegrityLevel	High
	Keywords	0x8000000000000000
	Level	4
	LogonGuid	{E500B0EA-219E-57AA-0000-0020E3030000}
	LogonId	0x3e3
	MD5	AAE3F5A29935E6ABCC2C2754D12A9AF0

J'ai trouvé l'**host**. Le processus qui nous intéresse a le code **1**, indiquant la création d'un processus. J'utilise ensuite la **valeur de hachage** pour faire de l'**OSINT**, ce qui me permet

d'obtenir ces informations.



Le fichier est un **Trojan.Swrort/Cryptz**, utilisé pour créer une **porte dérobée** (*backdoor*).

T1204 - User Execution of malware

---

6. Command & Control (C2)

Objectif : Identifier comment l'attaquant contrôle le système compromis.

Puisque nous avons trouvé un malware conçu pour une **backdoor** et la **persistance**, nous allons vérifier s'il y a des communications avec des **canaux C2** (*Command & Control*).

index=botsv1 sourcetype=suricata src_ip="192.168.250.70" Un site web ne devrait normalement pas envoyer de requêtes vers des **IP externes**. En général, ce sont les **utilisateurs** qui envoient ou téléchargent des fichiers. Dans les **alertes Suricata**, nous avons identifié **trois destinations suspectes**.

Top valeurs Top valeurs par heure Valeurs rares

Événements avec ce champ

Valeurs	Nombre	%
40.80.148.42	10 317	81,874 %
23.22.63.114	1 294	10,269 %
192.168.250.40	758	6,015 %
192.168.2.50	214	1,698 %
108.161.187.134	12	0,095 %
192.168.250.255	3	0,024 %

À ce moment-là, je vais enquêter sur les trois IP pour vérifier s'il y a des anomalies, et voici ce que je trouve.

Valeurs	Nombre	%
/joomla/administrator/index.php	1 235	95,736 %
/joomla/agent.php	52	4,031 %
/poisonivy-is-coming-for-you-batman.jpeg	3	0,232 %

J'ai trouvé **trois requêtes** provenant de l'IP de la victime, qui a téléchargé un **.jpeg** depuis le domaine de l'attaquant. J'ai vérifié s'il y avait d'autres liens avec le domaine ou le fichier, mais je n'ai rien trouvé. **J'en déduis donc que ce fichier est responsable de la défiguration du site web.**

T1071.001 - C2 Over Web Protocols (HTTP/S)

---

7. Actions on Objectives

Objectif : Déterminer l'objectif final de l'attaquant et les actions effectuées.

L'attaquant avait pour objectif de **défigurer** le site web, probablement afin de **nuire à la réputation de l'entreprise**.

Dans un premier temps, il a effectué un **scan de vulnérabilités**. Ensuite, il a utilisé une **attaque par force brute** pour s'authentifier.

Une fois connecté, il a envoyé un **malware** pour créer une **backdoor**, lui permettant de prendre le **contrôle du back-end**. Grâce à cet accès, il a téléchargé le fichier **un.jpeg**, qui a servi à **défigurer le site web**.

T1491.001 - Defacement: Internal Defacement

Lessons Learned – Enseignements Tirés

1- Protection des identifiants

- **MFA (Authentification Multi-Facteurs)**
- **Restriction des tentatives de connexion** avec un verrouillage temporaire après plusieurs échecs.

2- Amélioration de la surveillance des logs

- **Des règles SIEM plus strictes** (alertes sur scans et tentatives de brute force).
- **Un monitoring Suricata plus réactif** pour identifier rapidement les connexions C2.

3- Renforcement des contrôles d'upload

- Mettre en place un filtrage strict des extensions autorisées et utiliser une sandbox pour analyser les fichiers uploadés.