FreeSOC - SOC basierend auf Open Source mitteln

FreeSOC ist ein full featured SOC welches auf Open Source mitteln beruht und dennoch nicht den Vergleich zu kommerziellen Produkten scheuen brauch.

- Wazuh Agent installieren auf Proxmox (PVE, PBS und evtl. PMR)

Einleitung

Was ist überhaupt ein SOC

SOC ist die Abkürzung für Security Operation Center. Dies ist eine Zusammenfassung die aus mehreren Komponenten besteht:

- Mitarbeiter die Umgang mit IT Sicherheit geübt oder besser noch geschult sind
- Produkte für die IT Sicherheit
- Einen Raum (mindestns) der für diesen Zweck vorgesehen ist und entsprechend eingerichtet ist (z.B. mit mehreren Monitoren zur Überwachung

Die üblichen Aufgaben der Mitarbeiter umfassen:

- Überwachen der Infrastruktur
- Betreuen der Infrastruktur
- Updaten der Infrastruktur
- Schulen der Kolleginnen und Kollegen
- Überwachen und Betreuen der Clouddienste
- Überwachen und Betreuen der Internetanbindung

Welche Produkte setzen wir in unserem FreeSOC ein

Das von mir zusammengefügte SOC nenne ich FreeSOC, da es auf freie Open Source Produkten beruht - die Projektseite ist https://freesoc.de und derzeit noch im Aufbau.

Dies sind die folgenden:

- WAZUH
- MISP
- The Hive
- Cortex
- Shuffle

Die Zentrale Komponente ist dabei Wazuh - dies ist ein Open Source SIEM mit XDR Funktionalitäten. Alle Daten laufen erst in WAZUH zusammen, von dort werden diese ausgewertet und weiterverarbeitet. Angebunden werden, können dort:

- Windows Clients/Server
- MacOS Clients
- Linux/Unix Clients/Server
- Geräte wie Firewalls, Switches, Router, Telefonanlagen, VPN Gateway, Mailrelays und mehr mehr über Syslog

Weiterhin können auch folgende Zentrale Clouddienste überwacht werden, wie z.B.:

- Github
- Office/Microsoft 365 (und natürlich Azure)
- Google
- AWS

Die Agent überwacht folgendes:

- Vulnarability
- Compliance (PCI, GDPR usw.)
- Status mit Berichtserstellung
- Und noch vieles mehr

MISP ist eine zentrale Malware and Sharing Plattform, welche mehrere Feeds anbieten mit vielen interessanten Informationen wie z.B.:

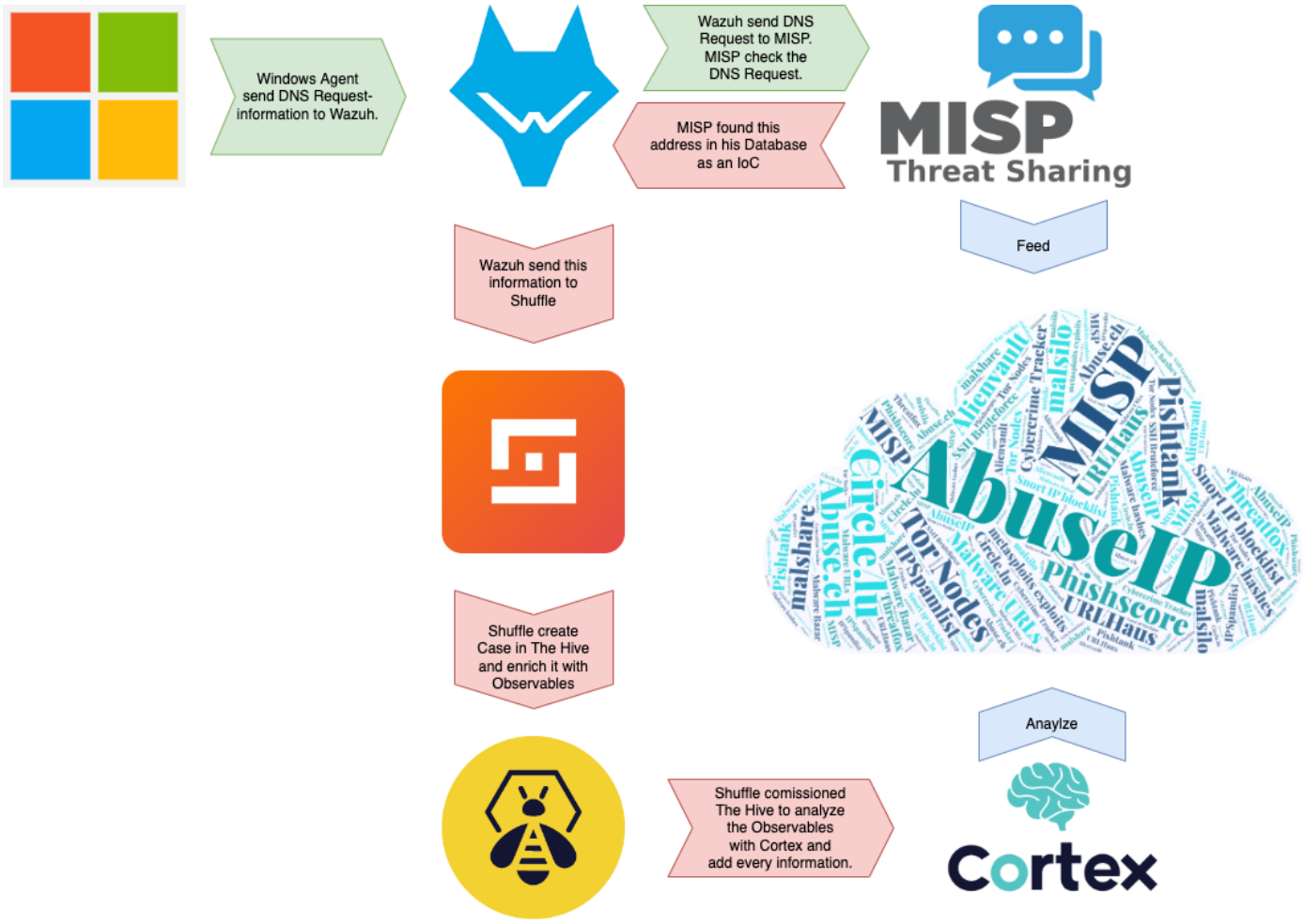
- Bad IP Adresses
- Spam Adressen
- Tor Exitnodes
- IoC wie URL, IP, Hashes, Dateinamen usw.
- Und vieles mehr

Hier werden die Datenbanken gepflegt, die uns unterstützen um z.B. IoC's zu finden.

The Hive ist ein SIRP, hier werden alle Informationen gemeldet und weiterverarbeitet. Mit Hilfe unserer automatisierungsplattform Shuffel sammeln wir alle IoCs ein und übergeben diese an Cortex, unserem Analyzer - der wieder rum nutzt mehrere Systeme wie MISP, Virustotal, AbuseIP usw. um dort Informationen zu sammeln und diese in dem Vorgang in The Hive anzureichern.

Nach der Bearbeitung können diese Vorgänge geschlossen und an MISP gegeben werden - um ggf. angebundenen Partnern zu Informieren.

Hier mal ein Beispiel vom gesamten Workflow:



Ein Beispiel wir unser SOC agiert seht ihr hier in dem Beispiel (leider erst mal nur in Englisch):

https://www.youtube.com/embed/hX5p9DQ3aBI?t=224s

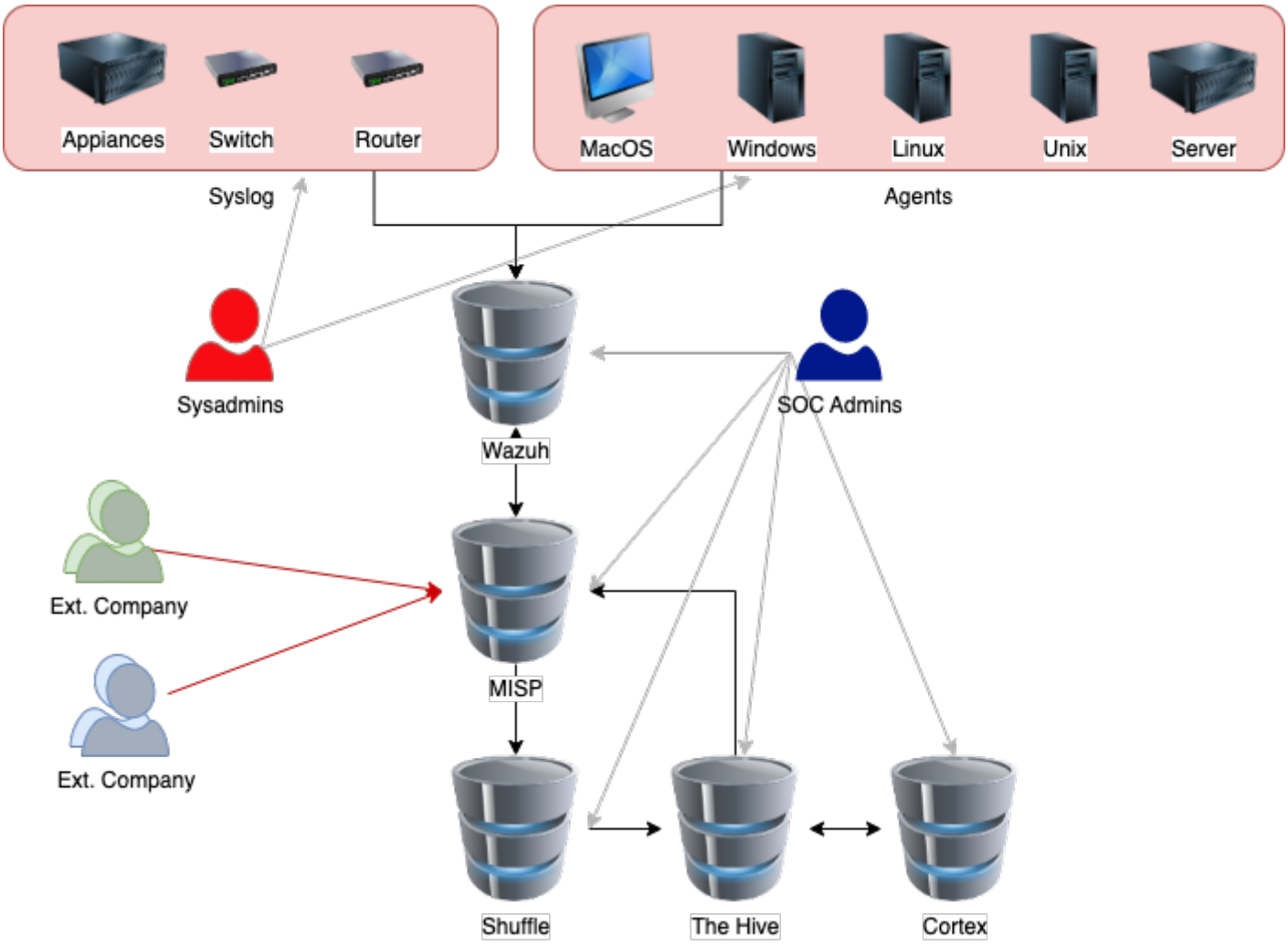
Möglicher Datenaustausch im FreeSOC

Der zentrale Datenbestand liegt im Wazuh - hier werden alle Daten hin geliefert. Hier benötigen Systemadministratoren und SOC Mitarbeiter Zugriffsrechte um ggf. Systeme hinzuzufügen oder zu entfernen.

Die Daten werden überwacht und zur weiteren Auswertung an MISP weitergegeben. MISP greift auf mehrere Datenquellen zu und kontrolliert ob diese bekannt sind. Wenn ja, dann wird ein IoC in Wazuh erstellt. Auf MISP benötigen Sicherheitsadministatoren und SOC Mitarbeiter Zugriff, damit diese MISP pflegen (Feeds/Quellen hinzufügen und ändern usw.).

Sollte ein IoC erstellt werden, wird Shuffle getriggert und ein Case in The Hive erstellt, Daten aus Wazuh geholt und mit in dem Case angereichert - mit diesen Daten werden diverse Analyzer in Cortex gestartet und alle Ergebnisse mit in den Case eingetragen. Auf The Hive und Cortex benötigen die SOC Mitarbeiter Zugriff.

Wurde das Ticket bearbeitet und abgeschlossen, kann dieses in MISP archiviert werden und ggf. mit anderen angebundenen Unternehmen geteilt werden.


Appiances
Switch
Router
MacOS
Windows
Linux
Unix
Server
Syslog
Agents
Sysadmins
SOC Admins
Wazuh
Ext. Company
MISP
Ext. Company
Shuffle
The Hive
Cortex

SOC einrichten

Vorbereitungen

Artikel folgt noch ...

Basis VM einrichten

Installieren eines Ubuntu 22.04 LTS Server als Minimalversion mit angepassten LVM Setup und aktivierten SSH Server.

Anschließend aktualisieren:

```
sudo apt update
sudp apt upgrade
sudo init 6
```

Sollte dies nicht der Wazuh Server werden, anschließend Wazuh Agent installieren mit zuordnen der entsprechenden Gruppen.

Danach kann dann wie folgt AutoUpdate eingerichtet werden:

https://www.lastbreach.de/blog/automatische-updates-fuer-linux-server

Hier ein Installationsvideo:

https://www.youtube.com/embed/L2ltLf_IMIM?t=6s

Solltet ihr noch einen Virtualisierer benötigen wie VMWare oder Proxmox, hilft euch dieses Installationsvideo zumindest bei VMWare weiter, Proxmox folgt noch:

https://www.youtube.com/embed/5smdQCtkVoQ?t=1s

Wazuh - SIEM

Wazuh ist ein SIEM mit XDR funktionalität und unsere zentrale Komponenten.

Wazuh installieren und einrichten, Agent installieren, Sysmon einrichten

https://www.youtube.com/embed/SRTW7nt4520?t=550s

Agent Update

Allgemeine Informationen:

- https://wazuh.com/install/
- https://documentation.wazuh.com/current/user-manual/agents/remote-upgrading/upgrading-agent.html
- https://documentation.wazuh.com/current/upgrade-guide/wazuh-agent/index.html

Agentupdates zentral über den Wazuh Server:

> Ich hatte hier jedoch massive Probleme, deshalb habe ich nachher die Agents selber auf dem entsprechenden Client selber.

> Um die Befehle auf dem Server oder den Clients zu starten benötigen sie entweder einen User der Gruppe Wazuh oder Root Rechte - alternativ können sie diese auch mit sudo ausführen (vor dem jeweiligen Befehl). Damit sie dies nicht bei mehreren Befehlen jedes mal machen müssen - können sie mit sudo bash eine Shell starten. Denken Sie auf jeden Fall daran, diese wieder mit exit zu beenden.

- Per SSH auf den Wazuh Server anmelden
- Sudo bash (alternativ ansonsten die Befehle einzeln mit sudo starten)
- /var/ossec/bin/agent_update -l
- /var/ossec/bin/agent_update -a CLIENT_ID

Agentupdate auf dem Linux-client selber:

> sudo bash
> curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | gpg --no-default-keyring --keyring gnupg-ring:/usr/share/keyrings/wazuh.gpg --import && chmod 644 /usr/share/keyrings/wazuh.gpg
> echo "deb [signed-by=/usr/share/keyrings/wazuh.gpg] https://packages.wazuh.com/4.x/apt/ stable main" | tee -a /etc/apt/sources.list.d/wazuh.list
> apt-get update
> apt-get upgrade

```
systemctl restart wazuh-agent
systemctl status wazuh-agent
```

Agentupdate auf dem Windows-client selber:

Download des aktuellen Agents (z.B. https://packages.wazuh.com/4.x/windows/wazuh-agent-4.3.10-1.msi) und anschließend starten der Installationsdatei.

Aktuelle Datei ist hier zu finden: https://documentation.wazuh.com/current/upgrade-guide/wazuh-agent/windows.html

Agentupdate auf dem Mac-client selber:

Download des aktuellen Agents (z.B. https://packages.wazuh.com/4.x/macos/wazuh-agent-4.3.10-1.pkg) und anschließend starten der Installationsdatei.

Aktuelle Datei ist hier zu finden: https://documentation.wazuh.com/current/upgrade-guide/wazuh-agent/macos.html

Syslog einrichten

Artikel folgt noch ...

Erstellen eines eigenen Decoders

Sollten Sie Systeme mit Syslog erfassen welche noch nicht unterstützt sind, dann sind diese zwar in den Archive Logs (wazuh-archive-*) können jedoch noch keine Alerts erzeugen.

Damit dies funktioniert, müssen sogenannte Decoder erstellt werden. Dies sind XML Dateien, welche die wazuh-archives-* Logs überwachen nach bestimmten Textmustern und dann diese in Alerts umwandeln, welche dann später sogar weiter interagieren können (z.B. Emailalerts, Integrations, Cases erstellen usw.).

Einen Decoder für die Sophos UTM erstelle ich gerade selber, da ich diesen benötige Die Dateien für erstelle ich später auf Github wenn dieser fertig ist.

Einen Beitrag von Wazuh selber, wie man Decoder selber erstellt findet ihr hier:

https://wazuh.com/blog/creating-decoders-and-rules-from-scratch/

Hier ist der aktuelle Stand (Stand 21.12.2022):

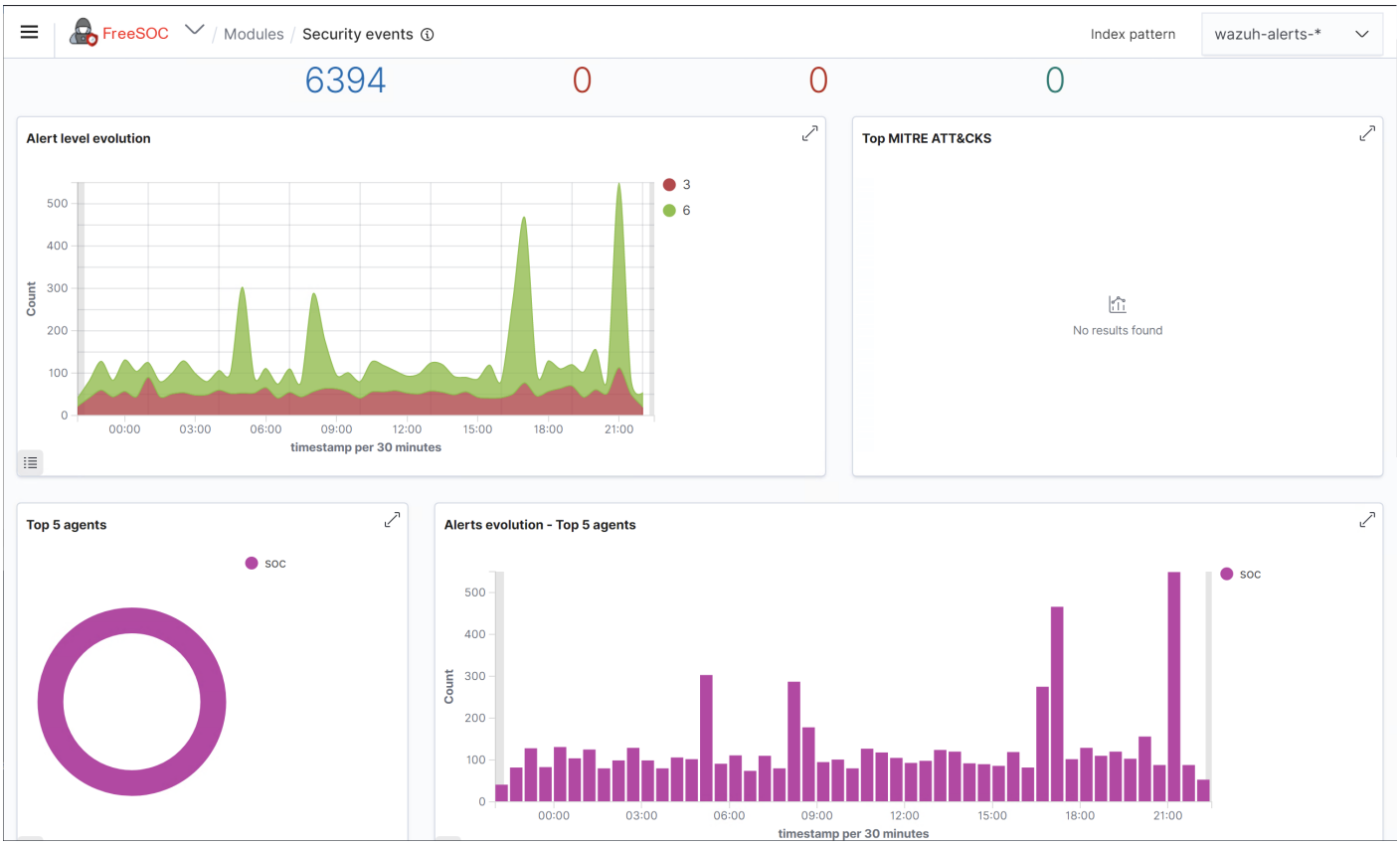
```
<decoder name="sophos-utm-custom">
  <prematch>^\d+:\d+:\d+-\d+:\d+:\d+\s\w+\s\w+[\d+]:\sid="\d+"</prematch>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex>(\d+:\d+:\d+)-(\d+:\d+:\d+)\s(\w+)\s(\w+)</regex>
  <order>date, time, hostname, module</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">name=\p(\w+\s\w+)</regex>
  <order>status</order>
</decoder>

<decoder name="sophos-utm-custom-child">
```

```
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">severity=\p(\w+\s\w+)</regex>
  <order>severity</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">sub=\p(\w+\s\w+)</regex>
  <order>sub</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">sys=\p(\w+\s\w+)</regex>
  <order>sys</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">srcip=\p(\d+.\d+.\d+.\d+)\p</regex>
  <order>srcip</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">dstip=\p(\d+.\d+.\d+.\d+)\p</regex>
  <order>dstip</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">srcport=\p(\d+)\p</regex>
  <order>srcport</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">dsport=\p(\d+)\p</regex>
  <order>dstport</order>
</decoder>
```

```
<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">profile=\p(\w+)\p</regex>
  <order>profile</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">filteraction=\p(\w+)\p</regex>
  <order>filteraction</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">url=\p(\S+)\p</regex>
  <order>url</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">referer=\p(\S+)\p</regex>
  <order>referer</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">reputation=\p(\w+)\p</regex>
  <order>reputation</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">categoryname=\p(\w+)\p</regex>
  <order>categoryname</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">user=\p(\w+)\p</regex>
```

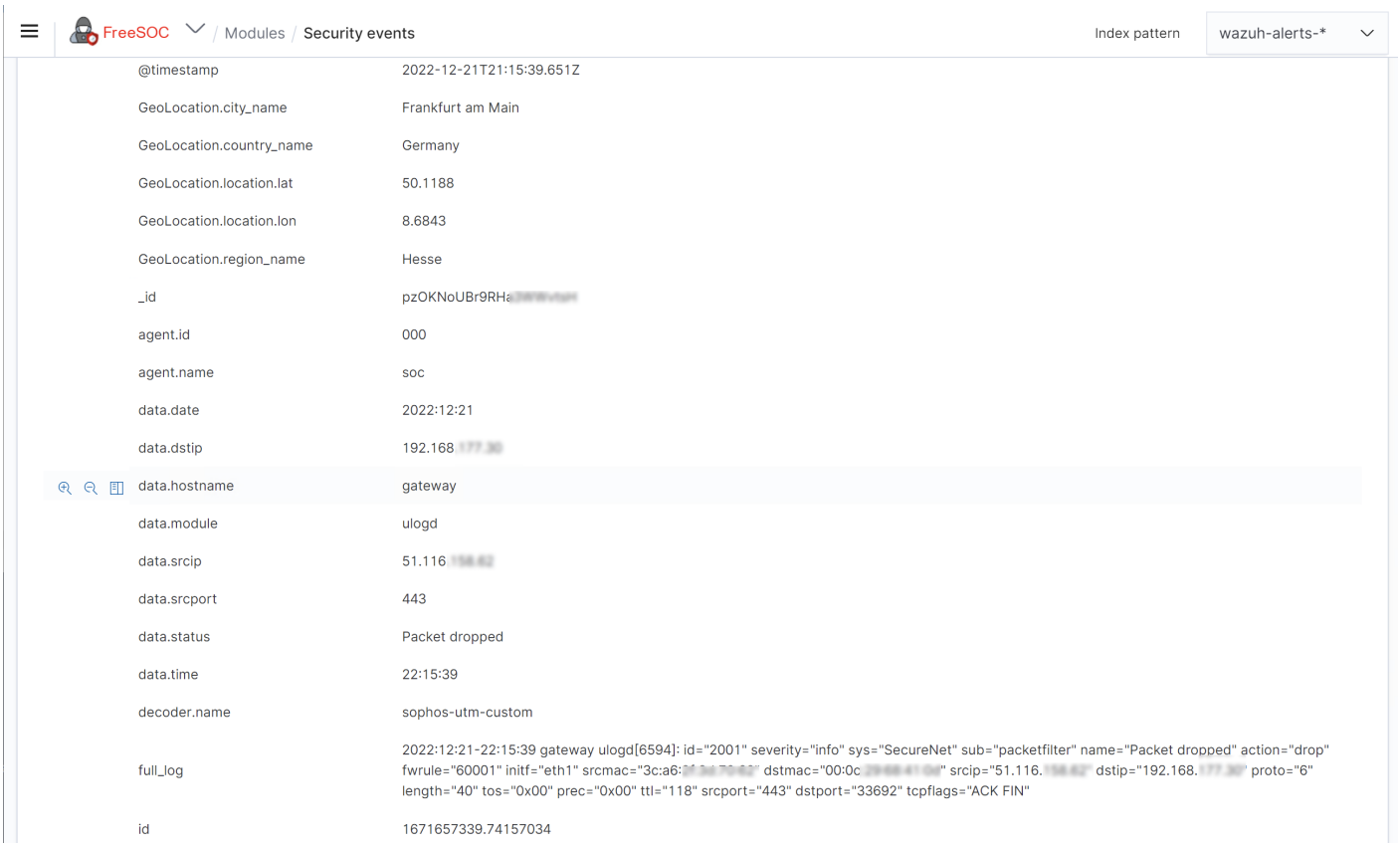
```
  <order>user</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">group=\p(\w+)\p</regex>
  <order>group</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">id=\p(\d+)\p</regex>
  <order>id</order>
</decoder>

<decoder name="sophos-utm-custom-child">
  <parent>sophos-utm-custom</parent>
  <regex offset="after_parent">reason=\p(\w+)\p</regex>
  <order>action</order>
</decoder>
```

Dazu ein paar Bilder, wie das in Wazuh aussieht:



Security Alerts

Time ↓	Agent	Agent name	Technique(s)	Tactic(s)	Description	Level	Rule ID
Dec 21, 2022 @ 22:15:39.651	000	soc			Sophos: Packet dropped on soc - Source 51.116.158.62 Destionation 192.168.177.30	6	117011
Dec 21, 2022 @ 22:15:39.651	000	soc			Sophos: Packet dropped on soc - Source 51.116.158.62 Destionation 192.168.177.30	6	117011
Dec 21, 2022 @ 22:15:39.651	000	soc			Sophos: Packet dropped on soc - Source 51.116.158.62 Destionation 192.168.177.30	6	117011
Dec 21, 2022 @ 22:15:13.590	000	soc			Sophos: Packet accepted on soc - Source 192.168.1.1 Destionation 192.168.50.10	3	117010
Dec 21, 2022 @ 22:15:13.590	000	soc			Sophos: Packet accepted on soc - Source 192.168.1.1 Destionation 192.168.50.10	3	117010
Dec 21, 2022 @ 22:15:07.461	000	soc			Sophos: AFC Alert on soc - ulogd: AFC Alert	3	117002
Dec 21, 2022 @ 22:14:52.698	000	soc			Sophos: AFC Alert on soc - ulogd: AFC Alert	3	117002
Dec 21, 2022 @ 22:14:50.150	000	soc			Sophos: Packet dropped on soc - Source 192.168.1.144 Destionation 224.0.0.1	6	117011
Dec 21, 2022 @ 22:14:22.354	000	soc			Sophos: Packet dropped on soc - Source 162.213.33.50 Destionation 192.168.177.30	6	117011



@timestamp	2022-12-21T21:15:39.651Z
GeoLocation.city_name	Frankfurt am Main
GeoLocation.country_name	Germany
GeoLocation.location.lat	50.1188
GeoLocation.location.lon	8.6843
GeoLocation.region_name	Hesse
_id	pzOKNoUBr9RHa[redacted]
agent.id	000
agent.name	soc
data.date	2022:12:21
data.dstip	192.168.[redacted]
data.hostname	gateway
data.module	ulogd
data.srcip	51.116.[redacted]
data.srcport	443
data.status	Packet dropped
data.time	22:15:39
decoder.name	sophos-utm-custom
full_log	2022:12:21-22:15:39 gateway ulogd[6594]: id="2001" severity="info" sys="SecureNet" sub="packetfilter" name="Packet dropped" action="drop" fwrule="60001" initf="eth1" srcmac="3c:a6:[redacted]" dstmac="00:0c[redacted]" srcip="51.116.[redacted]" dstip="192.168.[redacted]" proto="6" length="40" tos="0x00" prec="0x00" ttl="118" srcport="443" dstport="33692" tcpflags="ACK FIN"
id	1671657339.74157034

Index Management

Wazuh pflegt seine Daten auf dem Server unter unter /var/ossec/logs

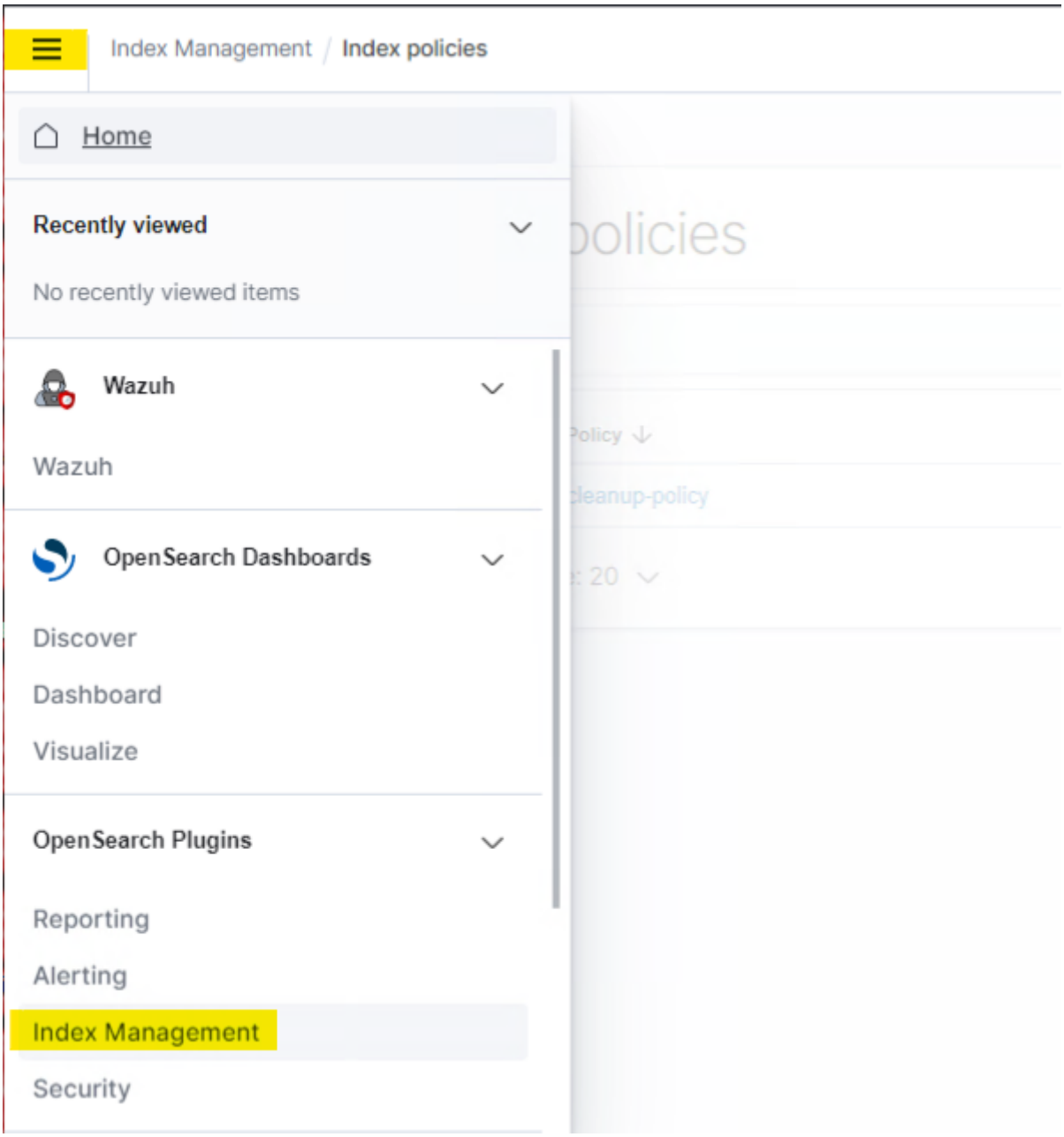
Die wichtigsten sind:

- /var/ossec/logs/alerts
- /var/ossec/logs/archives

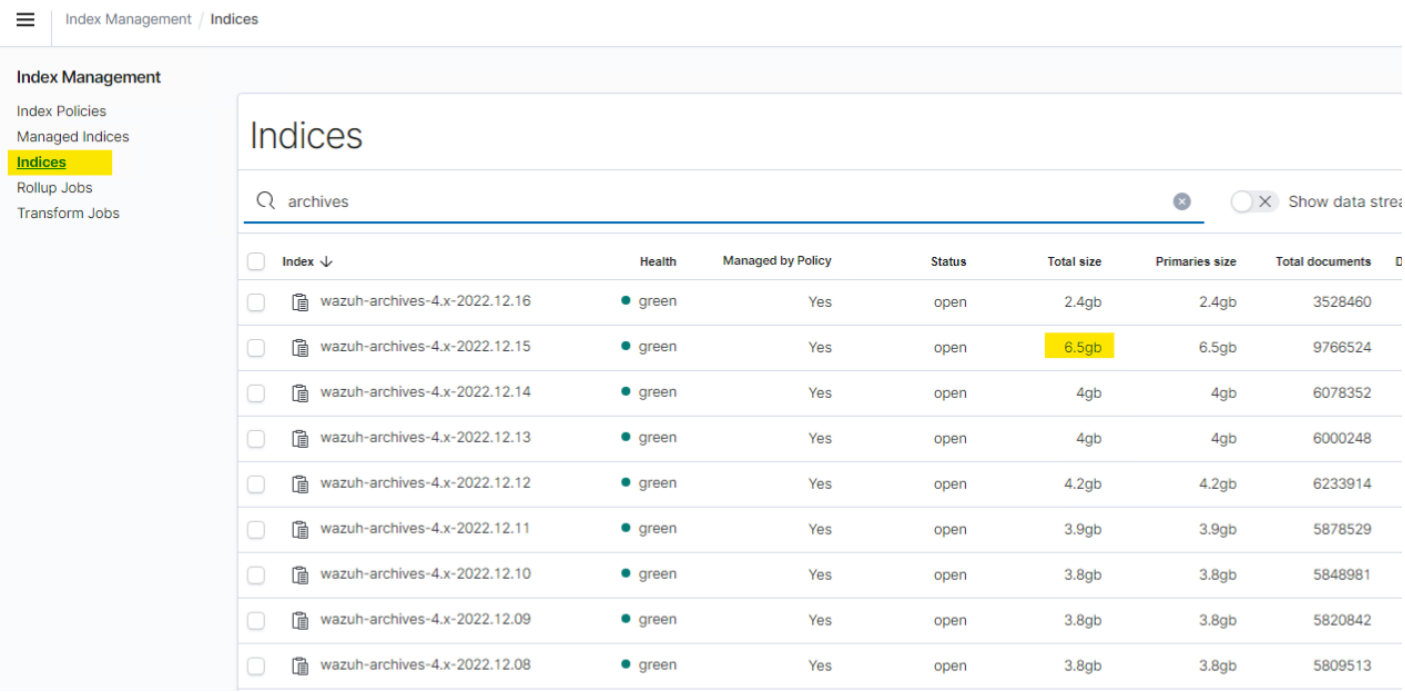
> Letzteren finden Sie, erst wenn die Sysog aktiviert haben und auch Gerät welche keine Agents nutzen können unterstützen wollen (z.B. VMWare, Switche, Firewalls, Gateway, VPNs usw.).

Die Indizies (Logs) finden Sie wie folgt:

- Loggen Sie sich im Wazuh Dashboard ein
- Gehen Sie oben Links auf die drei Balken
- Wählen Sie unten Index Management



- Dann unter Indicies finden sie die Indizies - besonders die unter wazuh-archives-* (für Syslogs) können sehr groß werden.



Damit nun die Festplatte nicht voll läuft, sollten das Index Management eingerichtet werden.

Hierzu bietet Wazuh einen Blogartikel an: https://wazuh.com/blog/wazuh-index-management/

> Dieser Artikel beruht auf einer älteren Version und sieht ein wenig anders aus - das ist jedoch nicht schlimm, wir benötigen nur den unteren Textteil.

Richten wir nun eine *Index Management Policy* ein:

- Gehen Sie links auf dem Punkt Index Policies
- Wählen Sie oben rechts Create Policy
- Wählen sie bei *Configuration Method* den *JSON Editor* aus
- Vergeben sie nun eine Policy ID wie z.B. *cleanup-policy*
- Ersetzen Sie den Inhalt durch den unten und wählen Sie *Create*

```
{
    "id": "cleanup-policy",
    "seqNo": 3244,
    "primaryTerm": 1,
    "policy": {
        "policy_id": "cleanup-policy",
        "description": "Cleanup Indicies Rule. Set after 15 Days to cold (Read Only) and delte
it after 60 Days.",
```

```
"last_updated_time": 1669980175085,
"schema_version": 12,
"error_notification": null,
"default_state": "hot",
"states": [
    {
        "name": "hot",
        "actions": [
            {
                "replica_count": {
                    "number_of_replicas": 0
                }
            }
        ],
        "transitions": [
            {
                "state_name": "cold",
                "conditions": {
                    "min_index_age": "15d"
                }
            }
        ]
    },
    {
        "name": "cold",
        "actions": [
            {
                "read_only": {}
            }
        ],
        "transitions": [
            {
                "state_name": "delete",
                "conditions": {
                    "min_index_age": "60d"
                }
            }
        ]
    },
    {
```

```
                "name": "delete",
                "actions": [
                    {
                        "delete": {}
                    }
                ],
                "transitions": []
            }
        ],
        "ism_template": [
            {
                "index_patterns": [
                    "wazuh-*"
                ],
                "priority": 100,
                "last_updated_time": 1669967389155
            }
        ]
    }
}
```

> Sollten Sie mehrere Wazuhserver bzw. Knoten haben - dann können Sie auch mehrere Replicas wählen - maximal die Anzahl der Knoten, die sie haben.

> Die Indizies, die aktuell genutzt werden sind *Hot*. *Cold* hingegen sind nur noch zum lesen vorhanden und belegen so weniger Systemressourcen (Ram und CPU, nicht Festplattenplatz). *Delete* gibt hingegen an, ab wann die Indizies (also die Dateien!) gelöscht werden. Diese sind danach nicht mehr zu nutzen, sofern sie diese nicht ausgelagert oder gesichert haben.

Ab nun an, werden die Indicies auf die Zukunft gepflegt. Sollten Sie diese Regel zu Anfang einrichten - sind sie nun fertig.

> Sollten Sie jedoch schon Indizies haben und möchten diese bereinigen (sehen Sie bitte davon ab, diese einfach von der Festplatte zu löschen - da sonst Wazuh nicht mitbekommt, das diese entfernt wurden), gehen sie einfach links auf Indicies und geben bei Search *wazuh-alerts* ein - wählen sie alle aus und klicken sie oben auf Apply Policy und wählen die eben erstellte Policy aus.

>

Sollte die Festplatte schon vollgelaufen sein und sie virtualisieren - dann können Sie in der Virtualisierung die Festplatte vergrößern und mit Tools wie Gparted booten und damit die virtuelle Festplatte vergrößern. Sollten sie ein physiches System benutzen könnten Sie eine weitere Platte einsetzen, diese formatieren und temporär mounten - die Dienste beenden, danach die Dateien rüberkopieren und anschließend den alten Ordner löschen. Nachdem sie die fstab angepasst haben und neugestartet haben, sollten wieder alles in Ordnung sein. **Sie sollten jedoch unbedingt vorher Sicherungen erstellen!**

Anpassen des Designs

Dieser Artikel folgt noch ...

Office 365 einrichten

Artikel folgt noch ...

Github einrichten

Artikel folgt noch ...

Wazuh Denoising

> [Rule 510]: Trojaned version of file detected (/bin/diff)

Lösung:

- Copy https://github.com/ossec/ossec-hids/blob/master/src/rootcheck/db/rootkit_trojans.txt to /var/ossec/etc/shared/ on your hub server.
- upgrade from source out of master

Weitere Informationen: https://github.com/ossec/ossec-hids/issues/2020

> [Rule 92213]: Executable file dropped in folder commonly used by malware (cleanmgr.exe)

Lösung:

Erstellen/erweitern der overwrites-custom-warnings.xml mit folgenden Inhalt (ggf. Rule ID anpassen):

```
  <rule id="110030" level="4">
    <if_sid>92204</if_sid>
    <field name="win.eventdata.image"
type="pcre2">C:\\\\Windows\\\\system32\\\\sdiagnhost\.exe</field>
    <options>no_full_log</options>
    <description>CleanMGR - Downlevel Info</description>
  </rule>
```

> [Rule 510]: Host-based anomaly detection event (/var/lib/docker/overlay2 und /var/lib/docker/volume)

Anpassen der Konfigurationsdatei im Bereich Rootcheck, anschließend Manager neu starten:

```
  <ignore>/var/lib/docker/overlay2/</ignore>
  <ignore>/var/lib/docker/volume/</ignore>
```

>

> Keine Vulnerability detects für Ubuntu 22.04

Hinzufügen zu der Konfiguration im Abschnitt Vulnerability-detector / Provider Canonical - anschließend speichern und Manager neustarten:

```
    <os>jammy</os>
```

> [Rule: 92201]: powershell.exe created a new scripting file under Windows Temp or User data folder (PSPolicyScript)

Lösung:

Erstellen/erweitern der overwrites-custom-warnings.xml mit folgenden Inhalt (ggf. Rule ID anpassen):

```
<rule id="110031" level="0">
    <if_sid>92213</if_sid>
    <field name="win.eventdata.image"
type="pcre2">C:\\\\Windows\\\\system32\\\\wsmprovhost.exe</field>
    <field name="win.eventdata.targetFilename"
type="pcre2">AppData\\\\Local\\\\Temp\\\\__PSScriptPolicyTest_*</field>
    <description>PSScript PolicyTest ignorieren</description>
  </rule>
```

Quelle:
https://www.reddit.com/r/Wazuh/comments/174enng/create_exclusion_for_false_positive/?rdt=50834

overwrites-custom-warnings.xml

```
<!-- Modify it at your will. -->
<group name="overwrites-custom-warnings">
  <rule id="60107" level="4" overwrite="yes">
    <if_sid>60104</if_sid>
    <field name="win.system.eventID">^577$|^4673$</field>
    <options>no_full_log</options>
    <description>Failed attempt to perform a privileged operation.</description>
  </rule>
<rule id="110001" level="4">
    <if_sid>92213</if_sid>
    <field name="win.eventdata.image"
type="pcre2">C:\\\\(?i)Windows\\\\(?i)system32\\\\(?i)cleanmgr\.exe</field>
    <description>CleanMGR - Downlevel Info</description>
</rule>
   <rule id="110002" level="4">
      <if_sid>510</if_sid>
      <field name="file">/usr/sbin/apachectl</field>
      <description>Ignoring the rootcheck alert for the file: $(data.file).</description>
  </rule>
  <rule id="110003" level="4">
    <if_sid>550</if_sid>
    <field name="syscheck.path"
type="pcre2">\/etc\/cups\/subscriptions\.conf|\/etc\/cups\/subscriptions\.conf\.O</field>
    <description>Known file</description>
  </rule>
  <rule id="110004" level="4">
    <if_sid>533</if_sid>
    <description>Netstart Warnings move down</description>
  </rule>
  <rule id="110005" level="4">
```

```
    <if_sid>92151</if_sid>
    <field name="win.eventdata.image" type="pcre2">C:\\\\Program
Files\\\\Veeam\\\\\.+\\\\Veeam\.\w+\.\w+\.exe</field>
    <description>Powershell started by Veeam</description>
  </rule>
   <rule id="110006" level="4">
    <if_sid>92152</if_sid>
    <field name="win.eventdata.imageLoaded"
type="pcre2">C:\\\\(?i)Windows\\\\(?i)System32\\\\spool\\\\drivers\\\\x64\\\\3\\\\PrintConfig.
dll</field>
    <description>Printer Powershell commands</description>
  </rule>
  <rule id="110007" level="4">
    <if_sid>92204</if_sid>
    <field name="win.eventdata.image" type="pcre2">C:\\\\Program Files|Program Files
\(x86\)\\\\Google\\\\Chrome\\\\Application\\\\chrome\.exe</field>
    <field name="win.eventdata.imageLoaded"
type="pcre2">C:\\\\Users\\\\.+\\\\AppData\\\\Local\\\\Temp\\\\.+\\\\software_reporter_tool\.ex
e</field>
    <description>Chrome Software Reporter</description>
  </rule>
  <rule id="110008" level="4">
    <if_sid>92204</if_sid>
    <field name="win.eventdata.image" type="pcre2">C:\\\\Program Files|Program Files
\(x86\)\\\\Google\\\\Update\\\\GoogleUpdate\.exe</field>
    <field name="win.eventdata.imageLoaded"
type="pcre2">C:\\\\Users\\\\.+\\\\AppData\\\\Local\\Temp\\\\.+\.exe</field>
    <description>Chrome Update</description>
  </rule>
  <rule id="110009" level="6">
    <if_sid>5402</if_sid>
    <description>Successful sudo to ROOT executed (Higher leveled).</description>
  </rule>
    <rule id="110010" level="6">
    <if_sid>5501</if_sid>
    <description>PAM: Login session opened (Higher leveled).</description>
  </rule>
    <rule id="110011" level="6">
    <if_sid>5502</if_sid>
    <description>PAM: Login session closed (Higher leveled).</description>
```

```
  </rule>
  <rule id="110012" level="6">
    <if_sid>5403</if_sid>
    <description>First time user executed sudo (Higher leveled).</description>
  </rule>
  <rule id="110013" level="6">
    <if_sid>5715</if_sid>
    <description>sshd: authentication success (Higher leveled).</description>
  </rule>
  <rule id="110014" level="4">
    <if_sid>5104</if_sid>
    <field name="description">Interface entered in promiscuous(sniffing) mode.</field>
    <description>Interface entered in promiscuous(sniffing) mode - Cortex Analyzer
working</description>
  </rule>
  <rule id="110015" level="4">
    <!-- Test: <if_sid>5402</if_sid> -->
    <if_sid>110009</if_sid>
    <field name="agent.name">pihole</field>
    <field name="data.srcuser">www-data</field>
    <description>Successful sudo to ROOT executed (Higher leveled).</description>
  </rule>
      <rule id="110016" level="4">
    <if_sid>5502</if_sid>
    <field name="name.agent">pihole</field>
    <field name="data.srcuser">www-data</field>
    <description>PAM: Login session closed (Down leveled).</description>
  </rule>
    <rule id="110017" level="4">
    <if_sid>92151</if_sid>
    <field name="win.eventdata.image" type="pcre2">C:\\\\Program
Files\\\\Veeam\\\\Backup365\\\\Veeam\.Archiver\.Service\.exe/gm</field>
    <description>Veeam starts Powershell commands</description>
  </rule>
  <rule id="110018" level="4">
    <if_sid>100651, 100653</if_sid>
    <field name="win.eventdata.parentimage" type="pcre2">C:\\\\Program Files|Program Files
\(x86\)\\\\TeamViewer\\\\Update\\\\update\.exe</field>
    <options>no_full_log</options>
    <description>TeamViewer Update</description>
```

```
</rule>
<rule id="110019" level="4">
    <if_sid>550</if_sid>
    <field name="syscheck.path" type="pcre2">/etc/pihole/\.+</field>
    <description>Ignoring PIHOLE Updates config</description>
  </rule>
  <rule id="110021" level="3">
    <if_sid>92151</if_sid>
    <field name="win.eventdata.user">BITSYSTEMS-
GMBH\\\\svc_monitoring|RESDOM\\\\svc_prtg</field>
    <description>Powershell started by PRTG</description>
  </rule>

  <rule id="110022" level="3">
    <if_sid>92151</if_sid>
    <field
name="win.eventdata.image">C:\\\\(?i)Windows\\\\(?i)System32\\\\(?i)ServerManager\.exe</field>
    <description>Powershell started by Servermanager</description>
  </rule>
  <rule id="110023" level="3">
    <if_sid>92151</if_sid>
    <field name="win.eventdata.image">C:\\\\Program Files \(x86\)\\\\Trend Micro\\\\Security
Agent\\\\utilCmdletWrapper\.exe</field>
    <description>Powershell started by TrendMicro</description>
  </rule>
  <rule id="110024" level="3">
    <if_sid>92151</if_sid>
    <field name="win.eventdata.image" type="pcre2">\w:\\\\Exchange
Server\\\\Bin\\\\Microsoft.Exchange.Store.Worker.exe</field>
    <description>Powershell started by Exchange Server</description>
  </rule>
  <rule id="110025" level="3">
    <if_sid>92204</if_sid>
    <field name="win.eventdata.image">C:\\\\Program Files (x86)\\\\Citrix\\\\ICA
Client\\\\receiver\\\\Receiver.exe</field>
    <field
name="win.eventdata.targetFilename">C:\\\\Users\\\\\w+\\\\AppData\\\\Local\\\\Temp\\\\\d\\\.+\
\\\CitrixReceiverUpdater.exe</field>
    <description>Citrix Receiver Update</description>
  </rule>
```

```
  <rule id="110026" level="3">
    <if_sid>119003</if_sid>
    <field name="misp.value">127.0.0.1|aka.ms</field>
    <description>Misp IoC's downgrade (false/positive)</description>
  </rule>
  <rule id="110027" level="3">
    <if_sid>9224</if_sid>
    <field
name="win.eventdata.image">C:\\\\Users\\\\\w+\\\\Downloads\\\\MicrosoftEdgeSetup\.exe</field>
    <field
name="win.eventdata.targetfilename">C:\\\\Users\\\\\.+\\\\AppData\\\\Local\\\\Temp\\\\\.+\\\\\
.+\\\\MicrosoftEdgeUpdate\w+\.exe</field>
    <description>Edge Update</description>
  </rule>
  <rule id="110028" level="3">
    <if_sid>510</if_sid>
    <field name="file">/var/tmp/tmccinstcheck\.dat</field>
    <description>TrendMicro Mac Updatecheck</description>
  </rule>
  <rule id="110029" level="3">
    <if_sid>510</if_sid>
    <field name="file">/tmp/ubuntu-advantage/candidate-version</field>
    <description>Ubuntu Advantage File</description>
  </rule>
  <rule id="110030" level="4">
    <if_sid>92204</if_sid>
    <field name="win.eventdata.image"
type="pcre2">C:\\\\(?i)Windows\\\\(?i)system32\\\\sdiagnhost\.exe</field>
    <options>no_full_log</options>
    <description>CleanMGR - Downlevel Info</description>
  </rule>
  <rule id="110031" level="0">
    <if_sid>100652</if_sid>
    <field name="win.eventdata.image"
type="pcre2">C:\\\\(?i)Windows\\\\(?i)system32\\\\schtasks\.exe</field>
    <field name="win.eventdata.CurrentDirectory" type="pcre2">C:\\\\Program Files\\\\Common
Files\\\\Microsoft Shared\\\\ClickToRun</field>
    <description>Office 365 Softwareupdate</description>
  </rule>
   <rule id="110032" level="3">
```

```
    <if_sid>92900</if_sid>
    <field name="win.eventdata.sourceImage"
type="pcre2">C:\\\\ProgramData\\\\Microsoft\\\\Windows
Defender\\\\Platform\\\\.+\\\\MsMpEng\.exe</field>
    <field name="win.eventdata.targetImage"
type="pcre2">C:\\\\(?i)Windows\\\\(?i)system32\\\\(?i)lsass\.exe</field>
    <description>Defender Scan lsass.exe</description>
  </rule>
   <rule id="110033" level="3">
    <if_sid>92201</if_sid>
    <match>PSScriptPolicyTest</match>
    <description>Powershell PSScriptPolicyTest</description>
  </rule>
  <rule id="110034" level="4">
    <if_sid>92213</if_sid>
    <field name="win.eventdata.image"
type="pcre2">C:\\\\(?i)Windows\\\\(?i)system32\\\\ServerManager\.exe</field>
    <description>ServerManager - Downlevel Info</description>
  </rule>
  <rule id="110035" level="0">
    <if_sid>100652</if_sid>
    <field name="win.eventdata.image"
type="pcre2">C:\\\\(?i)Windows\\\\(?i)system32\\\\(?i)schtasks\.exe</field>
    <field name="win.eventdata.parrentImage" type="pcre2">C:\\\\(?i)Program
Files\\\\(?i)Common Files\\\\(?i)microsoft
shared\\\\(?i)ClickToRun\\\\(?i)officesvcmgr\.exe</field>
    <description>Office 365 Softwareupdate</description>
  </rule>
  <rule id="110036" level="3">
    <if_sid>92900</if_sid>
    <field name="win.eventdata.sourceImage"
type="pcre2">C:\\\\(?i)Windows\\\\(?i)system32\\\\svchost\.exe|C:\\\\(?i)Windows\\\\(?i)system
32\\\\(?i)MRT\.exe</field>
    <field name="win.eventdata.targetImage"
type="pcre2">C:\\\\(?i)Windows\\\\(?i)system32\\\\(?i)lsass\.exe</field>
    <description>SVChost/Defender access lsass.exe</description>
  </rule>
  <rule id="110037" level="3">
    <if_sid>510</if_sid>
    <field name="file" type="pcre2">/tmp/filter.lock</field>
```

```
    <description>Downlevel Anomalidetection</description>
  </rule>
  <rule id="110038" level="3">
    <if_sid>87702</if_sid>
    <srcip>192.168.33.240</srcip>
    <description>OPNsense: $(agent.name) on $(hostname) - Source $(srcip) Destionation
$(dstip)</description>
  </rule>
   <rule id="110039" level="4">
      <if_sid>510</if_sid>
      <field name="file"> /bin/diff</field>
      <description>Ignoring the rootcheck alert for the file: $(data.file).</description>
  </rule>
  <rule id="110040" level="4">
    <if_sid>92213</if_sid>
    <field name="win.eventdata.image" type="pcre2">C:\\\\(?i)Windows\\\\(?i)Application
Compatibility Scripts\\\\(?i)acregl\.exe</field>
    <description>ServerManager - Downlevel Info</description>
  </rule>
  <rule id="110041" level="0">
    <if_sid>92213</if_sid>
    <field name="win.eventdata.image"
type="pcre2">C:\\\\Windows\\\\system32\\\\wsmprovhost.exe</field>
    <field name="win.eventdata.targetFilename"
type="pcre2">AppData\\\\Local\\\\Temp\\\\__PSScriptPolicyTest_*</field>
    <description>Ignore PSScript PolicyTest</description>
  </rule>
  <rule id="110042" level="0">
    <if_sid>92213</if_sid>
    <field name="win.eventdata.image"
type="pcre2">C:\\\\(?i)Windows\\\\(?i)system32\\\\(?i)wsmprovhost\.exe|C:\\\\(?i)Windows\\\\(?
i)system32\\\\(?i)sdiagnhost\.exe</field>
    <field name="win.eventdata.targetFilename"
type="pcre2">C:\\\\(?i)Users\\\\\w+\\\\(?i)AppData\\\\(?i)Local\\\\(?i)Temp\\\\__PSScriptPolic
yTest_\w+\.\w+\.ps1</field>
    <description>Ignore PSScript PolicyTest</description>
  </rule>
  <rule id="110043" level="5">
    <if_sid>92657</if_sid>
    <field name="agent.name" type="pcre2">MGMT-SRV</field>
```

```
    <description>MGMT-SRV RDP Logon downlevel</description>
  </rule>
  <rule id="110044" level="3">
    <if_sid>60602</if_sid>
    <field name="win.eventdata.library"
type="pcre2">C:\\\\(?i)Windows\\\\(?i)system32\\\\(?i)ntdsperf\.dll</field>
    <description>Windows Perflib downlevel</description>
  </rule>
  <rule id="110045" level="3">
    <if_sid>100652</if_sid>
    <field name="win.eventdata.parentimage" type="pcre2">C:\\\\(?i)Program Files\\\\(?i)Common
Files\\\\(?i)microsoft shared\\\\(?i)ClickToRun\\\\(?i)officesvcmgr\.exe</field>
    <description>Schedular Officeupdate downlevel</description>
  </rule>
</group>
```

overwrites-custom-misp.xml

```
<!-- Custom overwrites for MISP -->

<group name="overwrites-misp,">
  <rule id="116001" level="4">
    <if_sid>119003</if_sid>
    <field name="misp.value" type="pcre2">cdn.discordapp.com|discord.com|discord.gg</field>
    <description>Ignoring MISP IoC for $(misp.value)</description>
  </rule>
  <rule id="116002" level="4">
    <if_sid>119003</if_sid>
    <!-- <field name="misp.source.description" type="pcre2">Sysmon - Event 22: DNS Query for
google.com by C:\\\\Program Files\\\\Google\\\\Chrome\\\\Application\\\\chrome.exe</field> -->
    <field name="misp.value" type="pcre2">google.com|www.google.com</field>
    <description>Ignoring MISP IoC for $(misp.value)</description>
  </rule>
  <rule id="116003" level="4">
    <if_sid>119003</if_sid>
    <field name="misp.source.description" type="pcre2">.+C:\\\\Program Files \(x86\)\\\\PRTG
Network Monitor\\\\PRTG Probe\.exe</field>
    <description>Ignoring MISP IoC for $(misp.value)</description>
  </rule>
    <rule id="116004" level="4">
    <if_sid>119003</if_sid>
    <field name="misp.value">dc</field>
    <description>Ignoring MISP IoC for $(misp.value)</description>
  </rule>
  <rule id="116005" level="4">
    <if_sid>119003</if_sid>
    <field name="misp.type">sha256</field>
    <field
name="misp.value">e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855</field>
    <description>Ignoring MISP IoC for $(misp.value)</description>
  </rule>
```

```
  <rule id="116006" level="4">
    <if_sid>119003</if_sid>
    <field name="misp.type.value" type="pcre2">127\.0\.0\.1|0\.0\.0\.0|192\.168\.1\.\d</field>
    <description>Ignoring MISP IoC for $(misp.value)</description>
  </rule>
  <rule id="116007" level="4">
    <if_sid>119003</if_sid>
    <field name="misp.value" type="pcre2">php.net|www.php.net</field>
    <description>Ignoring MISP IoC for $(misp.value)</description>
  </rule>
</group>
```

overwrites-custom-o365.xml

```
<!-- Modify it at your will. -->
<group name="overwrites-custom-o365,">
   <rule id="118001" level="4">
    <if_sid>91725</if_sid>
    <field name="office365.objectid" type="pcre2">(?i).+\.prod\.outlook\.com\/Microsoft
Exchange Hosted Organizations\/.+\.onmicrosoft\.com\/DiscoverySearchMailbox{.+-.+-.+-.+-
.+}</field>
    <description>Office 365: User got FullAccess permissions in Exchange (Downlevel
DiscoverySearchMailbox)</description>
  </rule>
  <!-- Funktioniert noch nicht -->
<rule id="118002" level="12">
    <if_group>office365</if_group>
    <field name="GeoLocation.country_name">!Germany</field>
    <description>Office 365 used in: $(GeoLocation.country_name).</description>
  </rule>
</group>
```

overwrites-custom-sophos.xml

```
<!-- Modify it at your will. -->
<!-- Rules for Sophos UTM Custom -->

<group name="syslog,sophos,">
  <rule id="117001" level="3">
    <decoded_as>sophos-utm-custom</decoded_as>
    <description>Sophos: log without rule</description>
  </rule>

  <rule id="117002" level="3">
    <if_sid>117001</if_sid>
    <status>Authentication successful|AFC Alert|strict TCP</status>
    <description>Sophos: $(status) on $(hostname) - $(module): $(status)</description>
  </rule>

  <rule id="117003" level="12">
    <if_sid>117001</if_sid>
    <status>Authentication failed</status>
    <description>Sophos: $(status) on $(location) - User $(dstuser) [$(srcip)]</description>
  </rule>

    <!-- rule id="117004" level="12">
    <if_sid>117001</if_sid>
    <sub>up2date</sub>
    <description>Sophos: $(hostname) Service $(sub) - Status $(name)</description>
  </rule -->

  <rule id="117009" level="12">
    <if_sid>117001</if_sid>
    <hostname>smtp</hostname>
```

```
    <description>Sophos: $(status) on $(hostname) - User $(dstuser) [$(srcip)]</description>
  </rule>

   <rule id="117010" level="3">
    <if_sid>117001</if_sid>
    <status>Packet accepted</status>
    <description>Sophos: $(status) on $(hostname) - Source $(srcip) Destionation
$(dstip)</description>
  </rule>

  <rule id="117011" level="6">
    <if_sid>117001</if_sid>
    <status>Packet dropped</status>
    <description>Sophos: $(status) on $(hostname) - Source $(srcip) Destionation
$(dstip)</description>
  </rule>

  <rule id="117012" level="6">
    <if_sid>117001</if_sid>
    <status>Packet dropped (GEOIP)</status>
    <description>Sophos: $(status) on $(hostname) - Source $(srcip) Destionation
$(dstip)</description>
  </rule>

  <rule id="117013" level="6">
    <if_sid>117001</if_sid>
    <match>/var/chroot-httpd/var/webadmin/extra/httpd_session_cleanup</match>
    <description>Sophos: httpdcleanup on $(hostname)</description>
  </rule>

</group>
```

overwrites-unifi-udm-custom.xml

```
<!-- Modify it at your will. -->
<group name="syslog,unifi,">
  <rule id="114001" level="3">
    <decoded_as>unifi-udm-custom</decoded_as>
    <description>Unifi: log without rule</description>
  </rule>
</group>
```

MISP - Malware Sharing Plattform

MISP installieren

Artikel folgt noch ...

MISP einrichten

Artikel folgt noch ...

Wazuh an MISP anbinden

Artikel folgt noch ...

The Hive - SIRP

The Hive installieren

Artikel folgt noch ...

The Hive einrichten

Artikel folgt noch ...

The Hive updaten

Bevor sie updaten, stellen sie bitte sicher das sie in aktuelles Backup haben oder ein Snapshot erstellt haben - sofern sie Virtualisierung nutzen.

Loggen sie sich auf dem The Hive Server ein per SSH oder Console.

Rest folgt noch ...

The Hive - bekannte Probleme

In der Version bis 5.0.23-1 hatte ich das Problem, das ich Cases und Alerts unter bestimmen Umständen nicht mehr löschen konnte. Dies wurde gemeldet und soll mit der Version 5.1.1. behoben sein.

Cortex - Analyzer

Cortex ist ein Analyzer, welchen wir an the Hive angebunden haben. Dieser kann jedoch auch alleinstehend genutzt werden oder gar dank der ResAPI auch in anderen Produkten wie Shuffle, Wazuh usw.

Cortex installieren

Artikel folgt noch ...

Cortex einrichten

Artikel folgt noch ...

Shuffle - Automation Plattform

Shuffle installieren

Artikel folgt noch ...

Shuffle einrichten

Artikel folgt noch ...

Workflow: Case in The Hive erstellen mit Daten aus Wazuh

Artikel folgt noch ...

Allgemeine Sicherheit

Datensicherung

Im Open Source Bereich sieht es im Thema Datensicherung nicht so gut aus. Jedoch bieten auch einige Hersteller abgespeckte Versionen ihrer Professionellen Produkte kostenlos an.

Ich bevorzuge die Produkte vom Hersteller Veeam - so bietet er z.B.die folgenden Produkte an:

- Sicherung Windows / Linux Rechner
- Sicherung VMWare / HyperV Hypervisor
- Sicherung Office /Microsoft 365
- Und noch vieles mehr ...

Die Downloads findet ihr hier.

Allgemeine Sicherheit

BSI bietet auch kostenlos Unterstützung an

Das BSI bietet mehrere Mittel um zu Unterstützen.

So bietet es für Privatpersonen einen Newsletter an welcher über Software und Geschehnisse berichtet für den privaten Sektor.

Diese Newsletter bietet das BSI auch den Kommerziellen Bereich an, hier wird jedoch eher auf Firmenprodukte eingegangen.

Weiterhin bietet es ein Magazin an, welches viele Informationen rund um die IT - und Cyber - Sicherheit bietet: BSI - BSI-Magazin (bund.de).

Auch eine interessanter Dienst, ist die Allianz für Cybersicherheit - hier kann man einen Bündnis beitreten, welches viele Informationen und auch eine Anlaufstelle bietet rund um das Thema Cybersicherheit. Hier gibt es jedoch nicht nur Hilfe zur Prävention, sondern auch wenn schon was vorgefallen ist.

Containersicherheit

Fast immer nutzt man Docker Container, die von anderen bereitgestellt werden. Dabei ist man darauf angewiesen, das diese sich um die Sicherheit kümmern.

So gibt es mehrere Punkte zum Thema Sicherheit die man beachten sollte:

- Zum einen sollten die verwendeten Quellen keine Sicherheitslücke (CVE) aufweisen.
- Viele Komponenten setzen Web- oder Datenbankkomponente ein, hier gibt es auch bestimmte Probleme mit SQL Injection, Cross Site Scripting usw.
- Es könnten Keys oder fest eingerichtete User oder gar Backdoors eingerichtet sein.

Ein Open Source Tool, welches als CLI installiert werden kann direkt auf den Docker host selber ist z.B. Triny.

Installation

```
sudo apt-get install wget apt-transport-https gnupg lsb-release
wget -qO - https://aquasecurity.github.io/trivy-repo/deb/public.key | gpg --dearmor | sudo tee /usr/share/keyrings/trivy.gpg > /dev/null
echo "deb [signed-by=/usr/share/keyrings/trivy.gpg] https://aquasecurity.github.io/trivy-repo/deb $(lsb_release -sc) main" | sudo tee -a /etc/apt/sources.list.d/trivy.list
sudo apt-get update
sudo apt-get install trivy
```

Befehle

```
# Beispiel
trivy image --ignore-unfixed --scanners vuln <image> > <dateiname>
Scan:
trivy image --ignore-unfixed --scanners vuln vaultwarden/server:latest > /home/pleibling/240420_vaultwarden_report.txt
```

Formulare

Erstellen sie eine lokale Formatvorlage, als Beispiel könnte diese hier dienen:
https://github.com/aquasecurity/trivy/blob/main/contrib/html.tpl

Passen sie diese Vorlage ihren wünschen entsprechend an und speichern sie diese ab.

Mit dem folgenden Befehl können Sie dann diese Vorlage verwenden:

```
# Vorlage:
trivy image --format template --template "@html.tpl" -o <dateiname> <image>
# Beispiel:
trivy image --format template --template "@html.tpl" -o report.html phpmyadmin
```

Weitere Informationen

- https://www.heise.de/hintergrund/Marktuebersicht-Sicherheitsscanner-fuer-Container-Images-9682078.html?seite=all
- https://github.com/aquasecurity/trivy
- https://medium.com/@maheshwar.ramkrushna/scanning-docker-images-for-vulnerabilities-using-trivy-for-effective-security-analysis-fa3e2844db22
- https://aquasecurity.github.io/trivy/v0.48/docs/configuration/filtering/
- https://www.youtube.com/watch?v=Em_DdKkPUR8

Agent Installationen

Wazuh Agent installieren

Meldet euch im Wazuh an und wählt dann unter den Agents > Deploy Agent.

Alternativ aber auch hier noch mal.

Windows Wazuh Agent installieren:

```
Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.6.0-1.msi -OutFile ${env.tmp}\wazuh-agent; msiexec.exe /i ${env.tmp}\wazuh-agent /q WAZUH_MANAGER='192.168.50.10' WAZUH_AGENT_GROUP='default,SERVERS,WINDOWS' WAZUH_REGISTRATION_SERVER='192.168.50.10'
```

Windows Agent starten:

```
NET START WazuhSvc
```

Linux Agent installieren:

Folgt noch.

Proxmox Agent installieren:

Installation für Proxmox ist besser wenn man diese manuell ausführt, eine Anleitung findet ihr hier: https://wiki.leibling.de/books/freesoc-soc-basierend-auf-open-source-mitteln/page/wazuh-agent-installieren-auf-proxmox-pve-pbs-und-evtl-pmr.

Elastic Agent installieren

Elastic Agent installieren:

```
$ProgressPreference = 'SilentlyContinue' Invoke-WebRequest -Uri
https://artifacts.elastic.co/downloads/beats/elastic-agent/elastic-agent-8.10.4-windows-
x86_64.zip -OutFile elastic-agent-8.10.4-windows-x86_64.zip Expand-Archive .\elastic-agent-
8.10.4-windows-x86_64.zip -DestinationPath . cd elastic-agent-8.10.4-windows-x86_64 .\elastic-
agent.exe install --url=https://192.168.50.20:8220 --enrollment-
token=bG9RUG9Zc0JnN1Q5NzctWExPZEI6MzVTaGpiR0pTd3VhTUN1aXdFVWZ6Zw== --insecure
```

Wazuh Agent installieren auf Proxmox (PVE, PBS und evtl. PMR)

Die aktuellen Proxmox Versionen nutzen Debian 12 (Stand 04/2024). Diese sollten normalerweise unterstützt werden - jedoch lassen sie diese nicht so einfach installieren, bzw. in Betrieb nehmen.

Ich hatte beispielsweise das Problem, das sich der Agent zwar installieren aber nicht starten ließ (z.B. fehlte der User Wazuh, was man in der /etc/passwd sehen konnte - aber es fehlen auch noch andere Dinge, die Wazuh benötigt.

Sollte eine zuvor versuchte Installation noch vorhanden sein, dann diese wieder entfernen:

```
apt remove --purge wazuh-agent
```

Danach müssen wir dann den Client erst mal runterladen (aktuelle Übersicht der Downloadadressen findet ihr hier: https://documentation.wazuh.com/current/installation-guide/packages-list.html - hier könnt ihr mit der rechten Maustaste auf den Link gehen und dann die Adresse kopieren), z.B.

```
 wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.7.3-1_amd64.deb
```

Anschließend dann die Vorraussetzungen erfüllen und folgende Komponenten laden:

```
apt install lsb-base lsb-release
```

Danach dann die benötigen Variablen laden, die der Agent bei der Installation verwendet (bitte verwendet natürlich eure eigenen Daten:

```
WAZUH_MANAGER='192.168.1.1'
WAZUH_AGENT_GROUP='default,LINUX,SERVER,PROXMOX'
```

Danach könnt ihr dann auch schon den Agent installieren mit:

```
dpkg -i wazuh-agent_4.7.3-1_amd64.deb
```

Anschließend den Dienst registrieren und starten:

```
systemctl daemon-reload
systemctl enable wazuh-agent
systemctl start wazuh-agent
```

Sollte der Start einen Fehler ausgeben, dann könnt ihr den wie folgt kontrollieren:

```
systemctl status wazuh-agent
```

Sollte dort angegeben werden, das der Manager nicht gefunden wurde, dann hat das Setup die Daten die variablen nicht richtig übernommen, kontrolliert bitte ob in der Daten /var/ossec/etc/ossec.conf die Adresse verwendet wird und nicht der Name MANAGER_IP - ihr könnt auch direkt die Gruppen und das Debian Profil kontrollieren - es sollte ungefähr so aussehen:

```
<ossec_config>
  <client>
    <server>
      <address>192.168.1.1</address>
      <port>1514</port>
      <protocol>tcp</protocol>
    </server>
    <config-profile>debian, debian12</config-profile>
    <notify_time>10</notify_time>
    <time-reconnect>60</time-reconnect>
    <auto_restart>yes</auto_restart>
    <crypto_method>aes</crypto_method>
    <enrollment>
      <enabled>yes</enabled>
      <groups>default,PROXMOX,LINUX,SERVERS</groups>
      <authorization_pass_path>etc/authd.pass</authorization_pass_path>
    </enrollment>
  </client>
```

Wenn ihr dies geändert habt, dann könnt ihr wieder den Agent erneut starten und kontrollieren ob er gestartet wurde:

```
systemctl start wazuh-agent
systemctl status wazuh-agent
```

Der Agent sollte nun nach kurzer Zeit in eurer Wazuh Agent Übersicht auftauchen.

Noch ein wenig schneller geht es, wenn ihr den Teil mit den Variablen laden überspringt - während das bei fast allen Linux Systemen funktioniert, scheint es bei den Proxmox Systemen nicht zu funktionieren) und auch bevor ihr den Dienst registriert/startetn schon vorher die ossec.conf kontrolliert und ggf. anpasst. So sollte dann jeder Agent in ca. 3 Minuten installiert sein.