

## RILEVAZIONE DI ESECUZIONE DI COMANDI MALEVOLI

Rilevare l'esecuzione di comandi sospetti o malevoli su sistemi Linux, aiuta a prevenire incidenti di sicurezza o identificare attività compromesse. Questo è possibile grazie all'integrazione di **Wazuh** con il framework **Audit** di Linux, che fornisce dettagli su azioni critiche eseguite nel sistema.

**Linux Audit** è un sistema di monitoraggio che registra eventi di *sicurezza* sul sistema operativo. Questo strumento è progettato per tracciare attività che possono rappresentare rischi, come l'accesso non autorizzato, l'esecuzione di comandi, modifiche a file sensibili o tentativi di privilege escalation. L'Audit opera attraverso:

- Audit daemon (auditd): che registra gli eventi.
- Regole di audit: configurazioni che specificano cosa tracciare (es. modifiche a file specifici o esecuzione di comandi).
- Log: gli eventi tracciati vengono salvati in file leggibili per ulteriori analisi.

Al fine di implementare la funzione di rilevazione di esecuzione di comandi malevoli installiamo innanzitutto **Auditd** sull'endpoint Linux che stiamo utilizzando:

```
vboxuser@Rosario:~$ su
Password:
root@Rosario:/home/vboxuser# apt install -y auditd
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libwpe-1.0-1 libwpebackend-fdo-1.0-1
Use 'apt autoremove' to remove them.
The following additional packages will be installed:
  libauparse0
Suggested packages:
  audispd-plugins
The following NEW packages will be installed:
  auditd libauparse0
0 upgraded, 2 newly installed, 0 to remove and 6 not upgraded.
Need to get 270 kB of archives.
After this operation, 876 kB of additional disk space will be used.
Get:1 http://it.archive.ubuntu.com/ubuntu jammy/main amd64 libauparse0 amd64 1:3.0.7-1build1 [58,0 kB]
Get:2 http://it.archive.ubuntu.com/ubuntu jammy/main amd64 auditd amd64 1:3.0.7-1build1 [212 kB]
-----
```

Qui invece è dove sono locati i file di tracking della sicurezza:

```
root@Rosario:/home/vboxuser# cd /var/log/audit
root@Rosario:/var/log/audit# ls
audit.log
root@Rosario:/var/log/audit#
```

Apriamo il file di configurazione dell'agent e aggiungiamo tra i localfile l'audit log di **Auditd** in modo da poterlo visualizzare sulla dashboard di Wazuh:

```
<localfile>
  <log_format>audit</log_format>
  <location>/var/log/audit/audit.log</location>
</localfile>
```

Ora configuriamo le regole nel file *audit.rules* in modo da monitorare ogni comando eseguito dall'utente root ed aggiorniamo il set di regole:

```
GNU nano 6.2 /etc/audit/audit.rules *
## This file is automatically generated from /etc/audit/rules.d
-D
-b 8192
-f 1
--backlog_wait_time 60000
-a exit, always -F euid=0 -F arch=b64 -S execve -k audit-wazuh-c
-a exit, always -F euid=0 -F arch=b32 -S execve -k audit-wazuh-c

root@Rosario:/var/log/audit# auditctl -R /etc/audit/audit.rules
No rules
enabled 1
failure 1
pid 11455
rate_limit 0
backlog_limit 8192
lost 0
backlog 3
backlog_wait_time 60000
backlog_wait_time_actual 0
enabled 1
failure 1
pid 11455
rate_limit 0
backlog_limit 8192
lost 0
backlog 0
backlog_wait_time 60000
backlog_wait_time_actual 0
enabled 1
failure 1
pid 11455
rate_limit 0
```

## Significato della regola

```
-a exit,always -F euid=0 -F arch=b64 -S execve -k audit-wazuh-c
```

### 1. **-a exit,always**

- **exit**: Specifica che la regola deve essere applicata quando una chiamata al sistema (syscall) termina (exit point). È il momento in cui si possono raccogliere i dettagli finali dell'operazione.
- **always**: Indica che questa regola deve essere applicata sempre, senza eccezioni.

### 2. **-F euid=0**

- **euid=0**: Filtra gli eventi per cui l'Effective User ID (eUID) è 0, cioè gli utenti con privilegi di root. Questo significa che verranno registrati solo i comandi eseguiti da root o da processi che hanno temporaneamente acquisito privilegi di root.

### 3. **-F arch=b64**

- **arch=b64**: Limita la regola alle chiamate di sistema a 64 bit. Questo è importante per sistemi che supportano sia architetture a 32 bit che a 64 bit, assicurando che la regola non registri eventi ridondanti.

### 4. **-S execve**

- **execve**: È il nome della syscall da monitorare. **execve** è usata per eseguire comandi e lanciare nuovi processi. Questa regola, quindi, cattura ogni esecuzione di comandi effettuata con privilegi di root.

### 5. **-k audit-wazuh-c**

- **-k**: Specifica una chiave personalizzata per identificare gli eventi registrati da questa regola. In questo caso, la chiave è **audit-wazuh-c**. Può essere utile per filtrare o analizzare i log correlati in modo rapido usando questa etichetta.

## Obiettivo

Questa regola è pensata per:

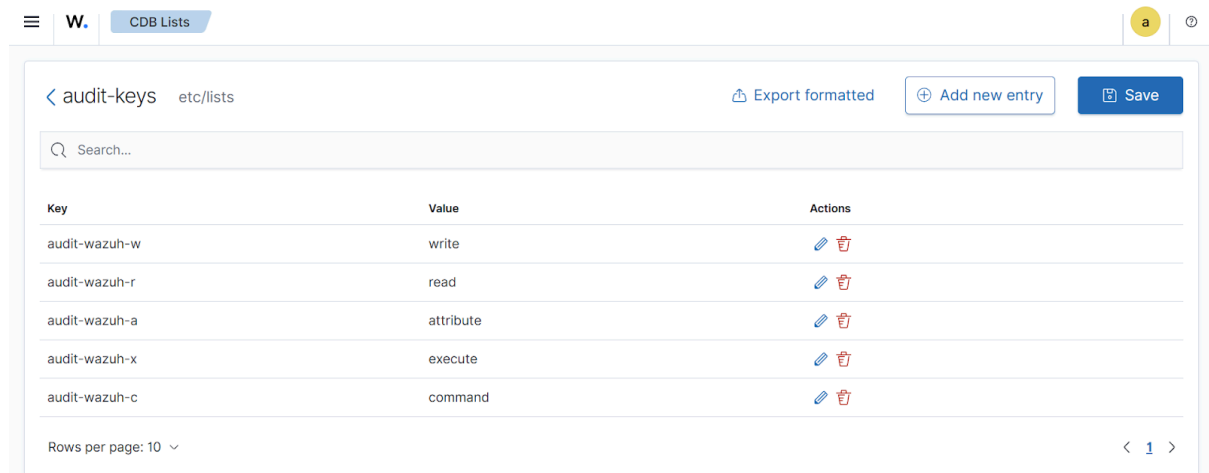
- Monitorare tutti i comandi eseguiti con privilegi di root.
- Registrare le attività sospette o non autorizzate.
- Integrare i log generati da auditd con soluzioni come **Wazuh**, che può analizzare e correlare tali eventi per il rilevamento di potenziali minacce.

## Output nel file di log di auditd

Quando questa regola viene attivata, genera un evento nei log di auditd (generalmente in **/var/log/audit/audit.log**) con dettagli come:

- Il comando eseguito.
- L'utente effettivo.
- Il timestamp.
- L'identificativo della regola (chiave **audit-wazuh-c**).

Sulla dashboard di **Wazuh** modifichiamo le audit keys in modo che coincidano con quelle indicate nelle regole del modulo **Auditd**:



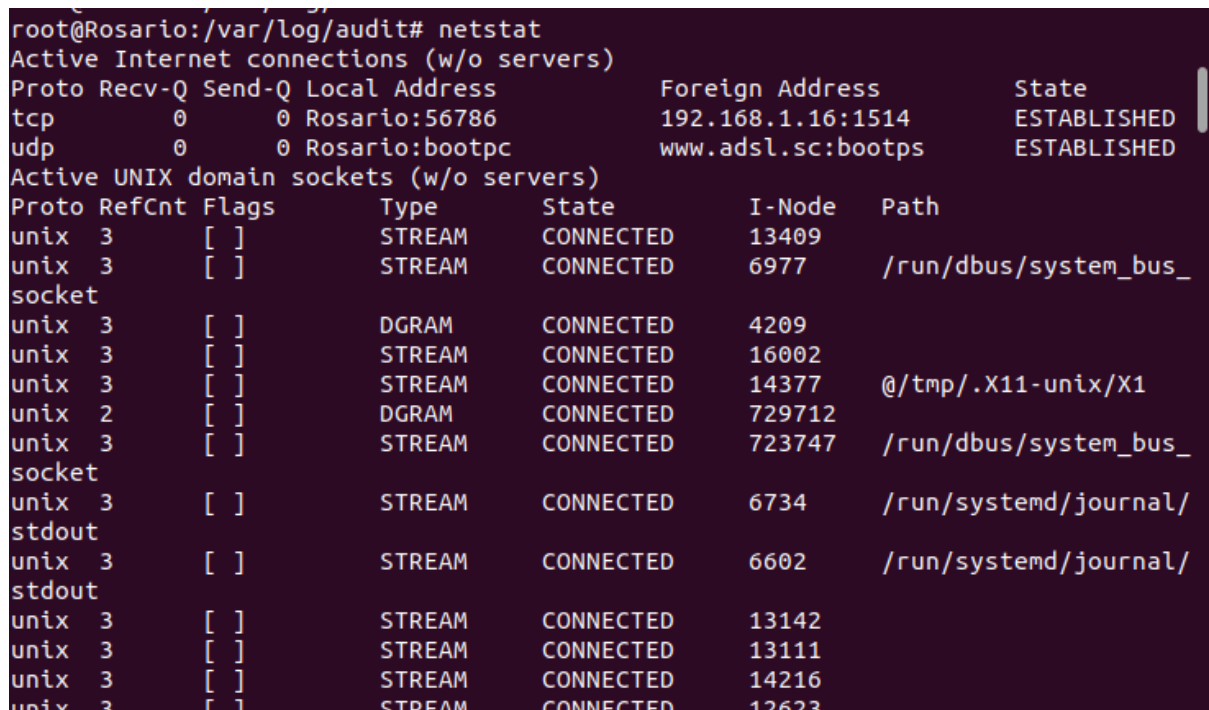
The screenshot shows the Wazuh CDB Lists interface. The breadcrumb is 'audit-keys etc/lists'. There are buttons for 'Export formatted', 'Add new entry', and 'Save'. A search bar is present. The table below lists the audit keys and their values, with edit and delete icons for each.

Key	Value	Actions
audit-wazuh-w	write	
audit-wazuh-r	read	
audit-wazuh-a	attribute	
audit-wazuh-x	execute	
audit-wazuh-c	command	

Rows per page: 10

< 1 >

Proviamo ad eseguire un semplice comando *netstat* da utente root per vedere se viene rilevato da **Wazuh** e se riusciamo a visualizzarlo sulla Dashboard:



```
root@Rosario:/var/log/audit# netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 Rosario:56786           192.168.1.16:1514      ESTABLISHED
udp        0      0 Rosario:bootpc          www.adsl.sc:bootps     ESTABLISHED
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags               Type                   State                  I-Node   Path
unix  3      [ ]                  STREAM                CONNECTED              13409
unix  3      [ ]                  STREAM                CONNECTED              6977     /run/dbus/system_bus_
socket
unix  3      [ ]                  DGRAM                CONNECTED              4209
unix  3      [ ]                  STREAM                CONNECTED              16002
unix  3      [ ]                  STREAM                CONNECTED              14377    @/tmp/.X11-unix/X1
unix  2      [ ]                  DGRAM                CONNECTED              729712
unix  3      [ ]                  STREAM                CONNECTED              723747   /run/dbus/system_bus_
socket
unix  3      [ ]                  STREAM                CONNECTED              6734     /run/systemd/journal/
stdout
unix  3      [ ]                  STREAM                CONNECTED              6602     /run/systemd/journal/
stdout
unix  3      [ ]                  STREAM                CONNECTED              13142
unix  3      [ ]                  STREAM                CONNECTED              13111
unix  3      [ ]                  STREAM                CONNECTED              14216
unix  3      [ ]                  STREAM                CONNECTED              12623
```

Su **Wazuh** risulterà la seguente situazione:

	↓ timestamp	agent.name	rule.description	rule.level
	Nov 27, 2024 @ 21:37:52.026	ubuntucione	Listened ports status (netstat) changed (new port o...	7
	Nov 27, 2024 @ 21:37:47.511	ubuntucione	Host-based anomaly detection event (rootcheck).	7
	Nov 27, 2024 @ 21:37:47.450	ubuntucione	Host-based anomaly detection event (rootcheck).	7
	Nov 27, 2024 @ 21:37:45.074	ubuntucione	Wazuh agent started.	3
	Nov 27, 2024 @ 21:13:29.817	ecommerceserver		4
	Nov 27, 2024 @ 19:20:20.469	ecommerceserver	Listened ports status (netstat) changed (new port o...	7
	Nov 27, 2024 @ 19:17:01.094	ubuntucione	Listened ports status (netstat) changed (new port o...	7

Se guardiamo il campo rule.description noteremo che il comando eseguito sull’endpoint è stato rilevato da **Wazuh**.