



Linux Logging for SOC

TryHackMe

Informações do documento	
Referência	CTF de estudo – Gabriel Garritano de Mendonça Villela
Nº Revisão	1
Data de publicação	30/09/2025
Link	https://tryhackme.com/room/linuxloggingforsoc

Redação	Gabriel Garritano	Estudante
Revisão	Nome do revisor	Orientador
Aprovação	Nome do aprovador	Diretor

Histórico de revisões		
Nº	Entregas	Descrição
0	DD/MM/AAAA	Produção
1	DD/MM/AAAA	Revisão
2	DD/MM/AAAA	Aprovação

Informações do CTF	
Nível de Dificuldade	Easy
Tipo de acesso	Gratuito
Conceitos envolvidos	WebHacking
Plataforma	TryHackMe
Área	Blue Team

CONTEXTUALIZAÇÃO

Linux has long been a leader in servers and embedded systems, and now its use is even more widespread with the growth of cloud adoption. As a SOC analyst, you are now very likely to investigate Linux alerts and incidents, either from traditional on-premises servers or from cloud-native containerized workloads. In this room, you will explore the most common Linux logs sent to SIEM and learn how to view them directly on-host.

CONCEITOS UTILIZADOS

- Explore authentication, runtime, and system logs on Linux
- Learn the commands and pitfalls when working with logs
- Uncover how tools like auditd monitor and report the events
- Practice every learned log source in the attached VM

DESENVOLVIMENTO

WORKING WITH TEXT LOGS

A primeira tarefa do CTF envolve a extração de informações a partir da consulta do log do sistema. É necessário a exploração da `/var/log` para identificar as seguintes informações: Nome de um servidor e mensagem deixada pelo usuário Yama. Para isso, utilizam-se os comandos `less` e `grep`. Os comandos utilizados estão visíveis no print.

Respostas:

`ntp.ubuntu.com`

Becoming mindful.

```
ubuntu@thm-vm:~$ less /var/log/syslog | grep server
2025-08-13T13:41:48.185573+00:00 thm-vm systemd[1]: Listening on ssh.socket - OpenBSD Secure Shell server socket.
2025-08-13T13:42:17.819639+00:00 thm-vm systemd-timesyncd[275]: Contacted time server 185.125.190.58:123 (ntp.ubuntu.com).
2025-08-13T13:44:21.248447+00:00 thm-vm systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
2025-08-13T13:44:21.337390+00:00 thm-vm systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
2025-08-13T13:57:19.919423+00:00 thm-vm systemd[1]: Listening on ssh.socket - OpenBSD Secure Shell server socket.
2025-08-13T13:57:49.388678+00:00 thm-vm systemd-timesyncd[268]: Contacted time server 185.125.190.58:123 (ntp.ubuntu.com).
2025-08-13T13:59:39.116136+00:00 thm-vm systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
2025-08-13T13:59:39.164709+00:00 thm-vm systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
2025-08-28T14:02:07.694115+00:00 thm-vm systemd-timesyncd[282]: Contacted time server 185.125.190.58:123 (ntp.ubuntu.com).
2025-08-28T14:02:07.701729+00:00 thm-vm systemd[1]: Listening on ssh.socket - OpenBSD Secure Shell server socket.
2025-08-28T14:02:07.702490+00:00 thm-vm systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
2025-08-28T14:02:07.745248+00:00 thm-vm systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
2025-09-09T13:45:41.665964+00:00 thm-vm systemd-timesyncd[266]: Contacted time server 185.125.190.58:123 (ntp.ubuntu.com).
2025-09-09T13:45:41.666994+00:00 thm-vm systemd[1]: Listening on ssh.socket - OpenBSD Secure Shell server socket.
2025-09-09T13:45:41.816961+00:00 thm-vm systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
2025-09-09T13:45:42.071277+00:00 thm-vm systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
2025-09-30T19:16:37.058928+00:00 thm-vm systemd[1]: Listening on ssh.socket - OpenBSD Secure Shell server socket.
2025-09-30T19:16:37.122594+00:00 thm-vm systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
2025-09-30T19:16:37.276379+00:00 thm-vm systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
ubuntu@thm-vm:~$

ubuntu@thm-vm:~$ less /var/log/syslog | grep Yama
2025-08-13T13:41:48.176653+00:00 thm-vm kernel: Yama: becoming mindful.
2025-08-13T13:57:19.908956+00:00 thm-vm kernel: Yama: becoming mindful.
2025-08-28T14:02:07.691523+00:00 thm-vm kernel: Yama: becoming mindful.
2025-09-09T13:45:41.659300+00:00 thm-vm kernel: Yama: becoming mindful.
2025-09-30T19:16:37.056146+00:00 thm-vm kernel: Yama: becoming mindful.
```

AUTHENTICATION LOGS

Para a seguinte etapa é necessário explorar os registros de identificação. do sistema. O payload a ser criado utiliza-se dos mesmos conceitos dos anteriores. Felizmente, a sala do TryHackMe oferece informações necessárias para os argumentos adicionais a serem buscados, junto de explicações de como cada tipo de authentication log funciona no Linux. É necessário encontrar qual usuário tentou conectar-se repetidas vezes via protocolo SSH e falhou e qual usuário foi adicionado ao servidor. Para isso, utilizou-se os comandos abaixo:

```
less /var/log/auth.log | grep -E 'ssh|Failed'
```

```
less /var/log/auth.log | grep -E 'useradd|sudo'
```

Respostas:

10.14.94.82

xerxes

```
025-08-13T15:56:18.903048+00:00 thm-vm sshd[1176]: Failed password for root from 10.14.94.82 port 57696 ssh2
025-08-13T15:56:22.556480+00:00 thm-vm sshd[1176]: Connection closed by authenticating user root 10.14.94.82 port 57696 [preauth]
025-08-13T15:56:27.989388+00:00 thm-vm sshd[1192]: Invalid user admin from 10.14.94.82 port 57697
025-08-13T15:56:29.897034+00:00 thm-vm sshd[1192]: pam_unix(sshd:auth): check pass; user unknown
025-08-13T15:56:29.897173+00:00 thm-vm sshd[1192]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ss ruser= rhost=10.14.94.82
025-08-13T15:56:31.754409+00:00 thm-vm sshd[1192]: Failed password for invalid user admin from 10.14.94.82 port 57697 ssh2
025-08-13T15:56:34.766528+00:00 thm-vm sshd[1192]: pam_unix(sshd:auth): check pass; user unknown
025-08-13T15:56:36.311951+00:00 thm-vm sshd[1192]: Failed password for invalid user admin from 10.14.94.82 port 57697 ssh2
025-08-13T15:56:37.349154+00:00 thm-vm sshd[1192]: Connection closed by invalid user admin 10.14.94.82 port 57697 [preauth]
025-08-13T15:56:37.349605+00:00 thm-vm sshd[1192]: PAM 1 more authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=10.14.94.82
025-08-13T15:56:43.069229+00:00 thm-vm sshd[1194]: Invalid user support from 10.14.94.82 port 57698
025-08-13T15:56:44.867125+00:00 thm-vm sshd[1194]: pam_unix(sshd:auth): check pass; user unknown
025-08-13T15:56:44.867248+00:00 thm-vm sshd[1194]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ss ruser= rhost=10.14.94.82
025-08-13T18:04:29.425939+00:00 thm-vm usermod[1458]: add 'xerxes' to group 'sudo'
025-08-13T18:04:29.426146+00:00 thm-vm usermod[1458]: add 'xerxes' to shadow group 'sudo'
025-08-13T18:35:09.196614+00:00 thm-vm sudo: pam_unix(sudo:session): session closed for user root
025-08-13T18:35:28.361941+00:00 thm-vm sudo: ubuntu : TTY=pts/0 ; PWD=/home/ubuntu ; USER=root ; COMMAND=/usr/bin/su
```

COMMON LINUX LOGS

Esta etapa ensina conceitos essenciais sobre funcionamento e exploração de logs genéricos do sistema, aplicações e como visualizar o histórico do bash de outros usuários. É necessário encontrar a versão do programa unzip funcionando no sistema e a mensagem escondida no histórico de um dos usuários. Para a primeira tarefa bastou seguir o mesmo protocolo de comando utilizado nas etapas anteriores, apenas alterando o parâmetro buscado pelo grep para 'unzip'. A segunda demandou buscar qual usuário havia deixado a mensagem, Primeiro foi realizada busca nos usuários ubuntu (aquele que estamos conectados) e no xerxes, em nenhum deles foi encontrado nada. Como o usuário que estamos utilizando possui privilégio de administrador, foi possível buscar o histórico dentro do usuário root. Utilizando o comando `less /home/root/.bash_history` encontrou-se o segredo.

Respostas:

6.0-28ubuntu4.1 T

HM{note_to_remember}

```
ubuntu@thm-vm:~$ less /var/log/dpkg.log | grep 'unzip'
2025-08-12 16:41:24 install unzip:amd64 <none> 6.0-28ubuntu4.1
2025-08-12 16:41:24 status half-installed unzip:amd64 6.0-28ubuntu4.1
2025-08-12 16:41:25 status unpacked unzip:amd64 6.0-28ubuntu4.1
2025-08-12 16:41:25 configure unzip:amd64 6.0-28ubuntu4.1 <none>
2025-08-12 16:41:25 status unpacked unzip:amd64 6.0-28ubuntu4.1
2025-08-12 16:41:25 status half-configured unzip:amd64 6.0-28ubuntu4.1
2025-08-12 16:41:25 status installed unzip:amd64 6.0-28ubuntu4.1
```

```
nano /etc/ssh/sshd_config
exit
ll -h /var/log
echo "THM{note_to_remember}" >> notes.txt
apt install auditd
cd /etc/audit/
ls
cd rules.d/
nano audit.rules
ll `which python3`
systemctl restart auditd
systemctl status auditd
nano audit.rules
cd /etc/logrotate.d
ls
rm *
exit
```

USING AUDITD

Para a última etapa da sala é necessário explorar os registros de Audit, um programa de monitoramento comumente utilizado por times de SOC. A sala ensina como utilizar o comando ausearch e a maneira no qual os logs são registrados. Para montar o payload, basta combinar as informações requeridas com a lógica de busca com grep utilizada nas etapas anteriores.

Respostas:

08/13/25 18:36:54

naabu_2.3.5_linux_amd64.zip

192.168.50.0/24

```
ubuntu@thm-vm:~$ ausearch -i -k file_thmsecret
Error opening config file (Permission denied)
NOTE - using built-in logs: /var/log/audit/audit.log
-----
type=PROCTITLE msg=audit(08/13/25 13:41:41.595:56) : proctitle=/sbin/auditctl -R /etc/audit/audit.rules
type=CWD msg=audit(08/13/25 13:41:41.595:56) : cwd=/
type=SOCKADDR msg=audit(08/13/25 13:41:41.595:56) : saddr={ saddr_fam=netlink nlk-fam=16 nlk-pid=0 }
type=SYSCALL msg=audit(08/13/25 13:41:41.595:56) : arch=x86_64 syscall=sendto success=yes exit=1084 a0=0x3 a1=0x7ffd77e095f0 a2=0x43c a3=0x0 ite
ms=1 ppid=293 pid=342 audit=unset uid=root gid=root euid=root suid=root fsuid=root egid=root sgid=root fsgid=root tty=(none) ses=unset comm=audit
ctl exe=/usr/sbin/auditctl subj=unconfined key=(null)
type=CONFIG_CHANGE msg=audit(08/13/25 13:41:41.595:56) : audit=unset ses=unset subj=unconfined op=add_rule key=file_thmsecret list=exit res=yes
-----
type=PROCTITLE msg=audit(08/13/25 13:57:13.530:51) : proctitle=/sbin/auditctl -R /etc/audit/audit.rules
type=CWD msg=audit(08/13/25 13:57:13.530:51) : cwd=/
type=SOCKADDR msg=audit(08/13/25 13:57:13.530:51) : saddr={ saddr_fam=netlink nlk-fam=16 nlk-pid=0 }
type=SYSCALL msg=audit(08/13/25 13:57:13.530:51) : arch=x86_64 syscall=sendto success=yes exit=1084 a0=0x3 a1=0x7ffc0e38b1b0 a2=0x43c a3=0x0 ite
ms=1 ppid=281 pid=334 audit=unset uid=root gid=root euid=root suid=root fsuid=root egid=root sgid=root fsgid=root tty=(none) ses=unset comm=audit
ctl exe=/usr/sbin/auditctl subj=unconfined key=(null)
type=CONFIG_CHANGE msg=audit(08/13/25 13:57:13.530:51) : audit=unset ses=unset subj=unconfined op=add_rule key=file_thmsecret list=exit res=yes
-----
type=PROCTITLE msg=audit(08/13/25 18:36:54.574:1600) : proctitle=cat /secret.thm
type=CWD msg=audit(08/13/25 18:36:54.574:1600) : cwd=/root
type=SYSCALL msg=audit(08/13/25 18:36:54.574:1600) : arch=x86_64 syscall=openat success=yes exit=3 a0=AT_FDCWD a1=0x7ffd133b973a a2=0_RDONLY a3=
0x0 items=1 ppid=1542 pid=1578 audit=ubuntu uid=root gid=root euid=root suid=root fsuid=root egid=root sgid=root fsgid=root tty=pts1 ses=30 comm=
cat exe=/usr/bin/cat subj=unconfined key=file_thmsecret
-----
type=PROCTITLE msg=audit(08/13/25 18:29:14.700:1523) : proctitle=wget https://github.com/projectdiscovery/naabu/releases/download/v2.3.5/naabu.2.3.5_linux_amd64.zip -O /tmp/naabu.zip
type=CWD msg=audit(08/13/25 18:29:14.700:1523) : cwd=/
type=EXECVE msg=audit(08/13/25 18:29:14.700:1523) : argc=4 a0=wget a1=https://github.com/projectdiscovery/naabu/releases/download/v2.3.5/naabu.2.3.5_linux_amd64.zip a2=0 a3=/tmp/naabu.
type=SYSCALL msg=audit(08/13/25 18:29:14.700:1523) : arch=x86_64 syscall=execve success=yes exit=0 a0=0x252f61107a8 a1=0x6252f6110708 a2=0x6252f6110708 a3=0x8 items=2 ppid=1499 pid=150
root egid=root sgid=root fsgid=root tty=pts1 ses=30 comm=wget exe=/usr/bin/wget subj=unconfined key=proc_wget
-----
ubuntu@thm-vm:~$ ausearch -i -k proc_all | grep 'naabu'
Error opening config file (Permission denied)
NOTE - using built-in logs: /var/log/audit/audit.log
type=PROCTITLE msg=audit(08/13/25 18:29:25.214:1524) : proctitle=unzip naabu.zip
type=EXECVE msg=audit(08/13/25 18:29:25.214:1524) : argc=2 a0=unzip a1=naabu.zip
type=PROCTITLE msg=audit(08/13/25 18:29:45.152:1526) : proctitle=chmod +x naabu
type=EXECVE msg=audit(08/13/25 18:29:45.152:1526) : argc=3 a0=chmod a1=+x a2=naabu
type=PROCTITLE msg=audit(08/13/25 18:29:51.986:1527) : proctitle=./naabu -host 192.168.50.0/24 -top-ports
type=EXECVE msg=audit(08/13/25 18:29:51.986:1527) : argc=4 a0=./naabu a1=-host a2=192.168.50.0/24 a3=-top-ports
```

CONCLUSÃO

O desafio aborda diversos tipos de registros diferentes comuns no dia a dia de um agente de segurança SOC. O CTF simula a exploração realizada por funcionários de blue team na manutenção e segurança de um servidor. Os conceitos aprendidos apesar de possuírem payloads relativamente simples na execução são essenciais para o monitoramento da segurança de um servidor.

