Report di Cybersecurity

Analisi e valutazione della sicurezza informatica

Autore: Sofia Dianin
Data: dal 07/08/2025 al 10/08/2025
Organizzazione: Epicode

Report Tecnico di Cybersecurity - Analisi Log Splunk

1. Introduzione e Contesto

Questo report documenta l'analisi forense di log di sistema acquisiti da un ambiente di rete simulato, utilizzando la piattaforma Splunk per l'estrazione, il filtraggio e la correlazione degli eventi. L'obiettivo primario è l'identificazione di potenziali indicatori di compromissione (IoC), tentativi di intrusione e malfunzionamenti di sistema. L'approccio adottato è conforme alle linee guida del NIST SP 800-61 (Computer Security Incident Handling Guide) e ai principi della norma ISO/IEC 27035 per la gestione degli incidenti di sicurezza informatica.

2. Metodologia di Analisi

L'analisi è stata condotta mediante le seguenti fasi:
1. **Raccolta dei dati**: importazione dei file di log in Splunk senza specificare un index, per garantire l'applicabilità generica delle query.
2. **Estrazione dei campi**: utilizzo di espressioni regolari (comando `rex`) per individuare e salvare informazioni chiave quali indirizzi IP, nomi utente, porte di connessione e codici di stato HTTP.
3. **Filtraggio degli eventi**: impiego di clausole di ricerca e filtri logici per isolare specifiche categorie di eventi.
4. **Correlazione e statistica**: uso di `stats`, `where` e visualizzazioni tabellari per identificare pattern anomali o ricorrenti.
5. **Interpretazione dei risultati**: analisi qualitativa e quantitativa degli output per definire possibili scenari di attacco o malfunzionamenti.

3. Risultati dell'Analisi

3.1 Tentativi di accesso falliti - 'Failed password'

Query: source=”tutorialdata.zip:*” “Failed password” | rex “Failed password for (invalid user)? (?<username>\w+) from (?<src_ip>\d{1,3}(?:\.\d{1,3}){3})” | eval reason=”Failed password” | table _time, src_ip, username, reason | sort -_time

Descrizione: source=”tutorialdata.zip:*” “Failed password” cerca tutti I file contenuti nella zip e filtra i log che contengono la frase “Failed password”

| rex “Failed password for (invalid user)? (?<username>\w+) from (?<src_ip>\d{1,3}(?:\.\d{1,3}){3})” rex usa una regex (espressione regolare) per estrare campi personalizzati, estrae il nome Utente e l’indirizzo IP

| eval reason=”Failed password” aggiunge un a personalizzazione, serve per dare un’etichetta alla riga

| table _time, src_ip, username, reason specifica I campi scelti quali il timestamp, l?indirizzo IP, Utente e motivo

| sort - _time richiede di ordinare gli eventi in base all?ordine, - per descendente

Query per identificare I tentative di accesso falliti.

Vista tabellare degli eventi.

>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[5276]: Failed password for invalid user appserver from 194.8.74.23 port 3351 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[1039]: Failed password for root from 194.8.74.23 port 3768 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[5258]: Failed password for invalid user testuser from 194.8.74.23 port 3626 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[1165]: Failed password for apache from 194.8.74.23 port 4604 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[3760]: Failed password for invalid user mongodb from 194.8.74.23 port 2472 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[4998]: Failed password for mail from 194.8.74.23 port 1552 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[1930]: Failed password for games from 194.8.74.23 port 3007 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[5801]: Failed password for invalid user desktop from 194.8.74.23 port 2285 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[3759]: Failed password for nagios from 194.8.74.23 port 3769 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[5979]: Failed password for invalid user cyrus from 194.8.74.23 port 3417 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[4994]: Failed password for invalid user guest from 194.8.74.23 port 2294 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[2605]: Failed password for invalid user itmadmin from 194.8.74.23 port 4692 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure

Andamento temporale degli acessi falliti giornalieri

>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[5510]: Failed password for invalid user ubuntu from 212.58.253.71 port 4033 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[2054]: Failed password for invalid user mailman from 212.58.253.71 port 2692 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[2054]: Failed password for invalid user mailman from 212.58.253.71 port 2692 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[4791]: Failed password for invalid user system from 212.58.253.71 port 4234 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[4791]: Failed password for invalid user system from 212.58.253.71 port 4234 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[1246]: Failed password for invalid user jabber from 212.58.253.71 port 2772 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[1246]: Failed password for invalid user jabber from 212.58.253.71 port 2772 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[5023]: Failed password for invalid user amanda from 109.169.32.135 port 4272 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[5023]: Failed password for invalid user amanda from 109.169.32.135 port 4272 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[3132]: Failed password for invalid user admin from 109.169.32.135 port 1742 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[3132]: Failed password for invalid user admin from 109.169.32.135 port 1742 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure

2025-08-06 13:04:36	194.8.74.23	appserver	Failed password
2025-08-06 13:04:36	194.8.74.23	appserver	Failed password
2025-08-06 13:04:36	194.8.74.23	root	Failed password
2025-08-06 13:04:36	194.8.74.23	root	Failed password
2025-08-06 13:04:36	194.8.74.23	testuser	Failed password
2025-08-06 13:04:36	194.8.74.23	testuser	Failed password
2025-08-06 13:04:36	194.8.74.23	apache	Failed password
2025-08-06 13:04:36	194.8.74.23	apache	Failed password
2025-08-06 13:04:36	194.8.74.23	mongodb	Failed password
2025-08-06 13:04:36	194.8.74.23	mongodb	Failed password
2025-08-06 13:04:36	194.8.74.23	mail	Failed password
2025-08-06 13:04:36	194.8.74.23	mail	Failed password
2025-08-06 13:04:36	194.8.74.23	games	Failed password
2025-08-06 13:04:36	194.8.74.23	games	Failed password
2025-08-06 13:04:36	194.8.74.23	desktop	Failed password
2025-08-06 13:04:36	194.8.74.23	desktop	Failed password

Dettagli come Utente, indirizzo IP, data e ora

La query mirata ai messaggi 'Failed password' ha permesso di identificare numerosi tentativi di login non autorizzati, estrapolando indirizzo IP sorgente, nome utente e timestamp. Questi dati sono fondamentali per individuare attacchi di tipo brute force o attività di ricognizione.
Rischio: compromissione degli account utente tramite password guessing.
Mitigazione: implementare blocco automatico dell'account dopo N tentativi, abilitare l'autenticazione a più fattori (MFA) e mantenere blacklist di IP noti per attività malevole.

3.2 Sessioni SSH riuscite - utente 'djohnson'

Query: source=”tutorialdata.zip:*” “ Accepted password” “djohnson” | rex “Accepted password for (?<username>w+) from (?<src_ip>\d{1,3}(?:\.\d{1,3}){3}) port \d+ ssh2” | where username=”djohnson” | eval user_id=username | table _time, user_id | sort -_time

Descrizione: source=”tutorialdata.zip:*” “ Accepted password” “djohnson” cerca in tutti I log contenuti nella zip, cerca solo gli acessi riusciti e con Utente “djohnson”

| rex “Accepted password for (?<username>w+) from (?<src_ip>\d{1,3}(?:\.\d{1,3}){3}) port \d+ ssh2” estrae username e l?IP

| where username=”djohnson” restituisce il log solo se contenente “djohnso”

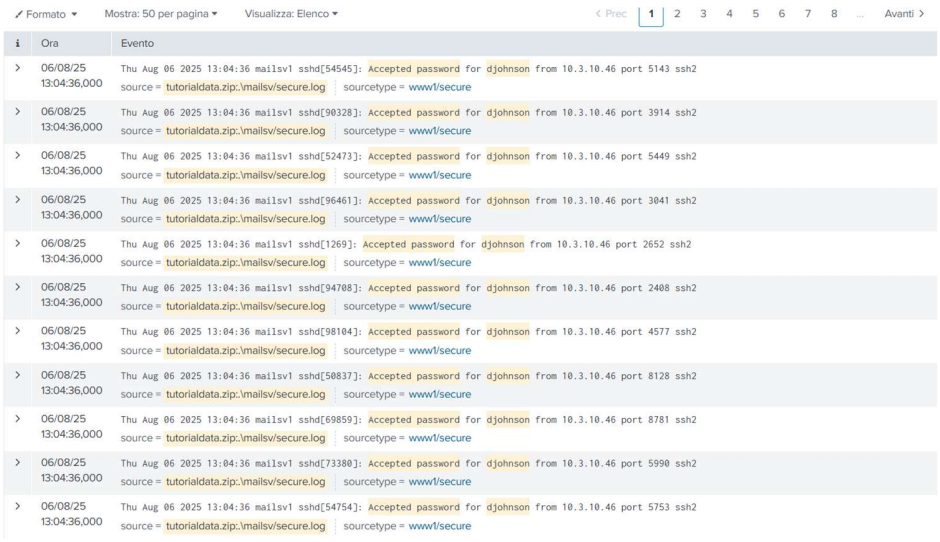
| eval user_id=username crea un campo di conferma dell’utente

| table _time, user_id mostra il timestamp e l?ID utente

| sort - _time ordina I risultati dal più recente

```
source="tutorialdata.zip:*" "Accepted password" "djohnson"
| rex "Accepted password for (?<username>\w+) from (?<src_ip>\d{1,3}(?:\.\d{1,3}){3}) port \d+ ssh2"
| where username="djohnson"
| eval user_id=username
| table _time, user_id
| sort -_time
```

Dettaglio dell?evento nella timeline



Log riscontarti con utente “djohnson”, indirizzo IP e porte

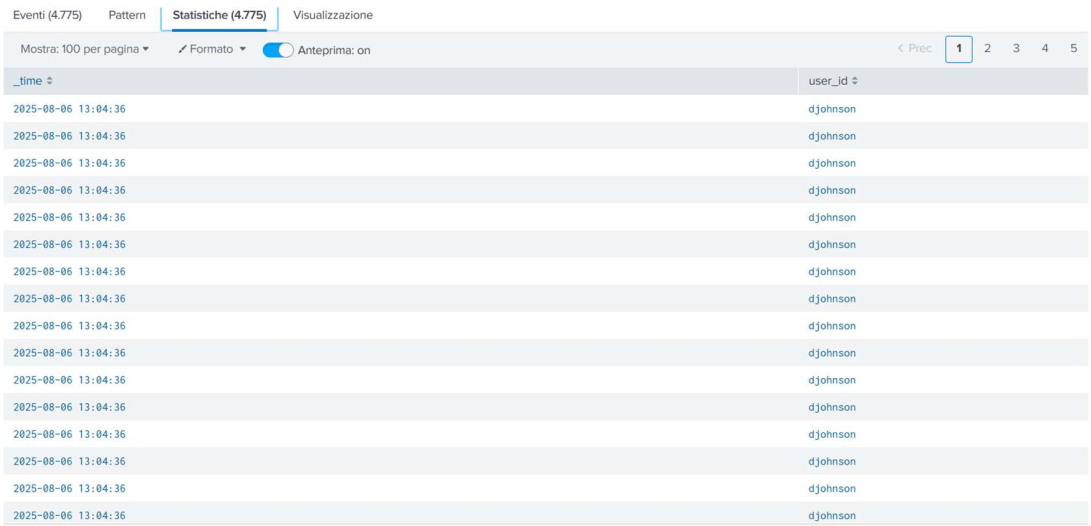


Tabella con data, orario e utente

La ricerca di eventi 'Accepted password for djohnson' ha consentito di monitorare un account specifico, verificando accessi riusciti con data, ora e identificativo utente. Il tracciamento di utenti privilegiati è critico per prevenire escalation di privilegi.
Rischio: uso non autorizzato di credenziali valide.

Mitigazione: MFA obbligatorio, controllo delle geolocalizzazioni degli accessi e alert in caso di orari inconsueti.

3.3 Tentativi di accesso falliti da IP specific

Query: source="tutorialdata.zip:*" "Failed password" "86.212.199.60" | rex "Failed password for (invalid user)? (?<username>w+) from 86\.212\.199\.60 port (?<port>\d+)" | table _time, username, port | sort -_time

Descrizione: source="tutorialdata.zip:*" "Failed password" "86.212.199.60" cerca I log nella zip mostrando gli eventi che contengono "Failed password", restringendo agli eventi in cui compare IP

| rex "Failed password for (invalid user)? (?<username>w+) from 86\.212\.199\.60 port (?<port>\d+)" aggiunge delle etichette di selezione e cattura il numero di porte

| table _time, username, port aggiunge etichette quali tempo, Utente e porte

| sort - _time ordina in base decrescente I log

Query focalizzata sull?IP 86.212.199.60

Timeline ed eventi per IP specifici

Dettagli di eventi con Utente e porte

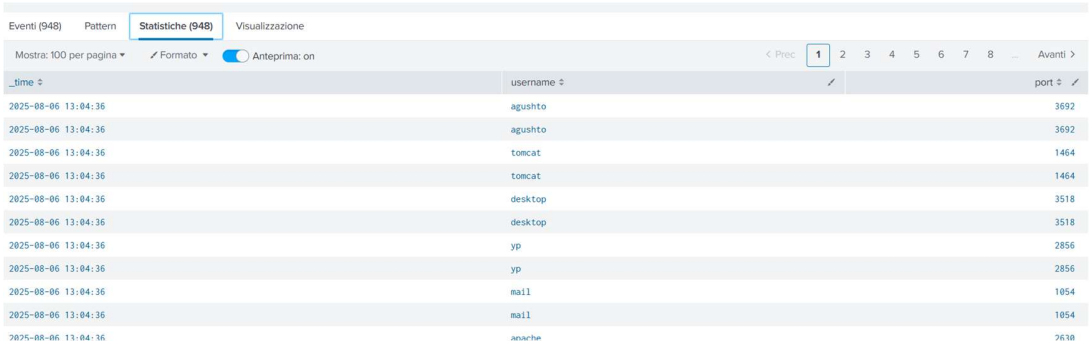


Tabella con data, ora , Utente e porte

Individuati tentativi di accesso falliti provenienti dall'IP 86.212.199.60, con dettaglio di utente e porta. Questo approccio consente di isolare campagne di attacco mirate da singole sorgenti.
Rischio: attacco persistente e mirato verso un asset specifico.
Mitigazione: blocco IP a livello firewall, segnalazione all'ISP, analisi dell'eventuale rete di provenienza.

3.4 IP con più di 5 tentativi falliti

Query: source=”tutorialdata.zip:*” “Failed password” | rex “Failed password for(invalid user)? \w+ from (?<src_ip>\d{1,3}(?:\.\d{1,3}){3})” | stats count AS tentative by src_ip | where tentativi > 4 | sort -tentativi

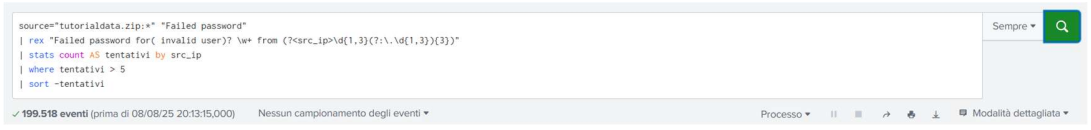
Descrizione: source=”tutorialdata.zip:*” “Failed password” limita la ricercar ai soli accessi falliti nella ricercar dei log nella zip

| rex “Failed password for(invalid user)? \w+ from (?<src_ip>\d{1,3}(?:\.\d{1,3}){3})” ester indirizzi IP da ogni riga

| stats count AS tentative by src_ip conta quanti tentative ha fatto ogni IP

| where tentativi > 5 mostra solo quelli con più di 5 tentativi

| sort -tentativi ordina da numero di tentative



Timeline degli accessi falliti

Lista dei log dei tentative

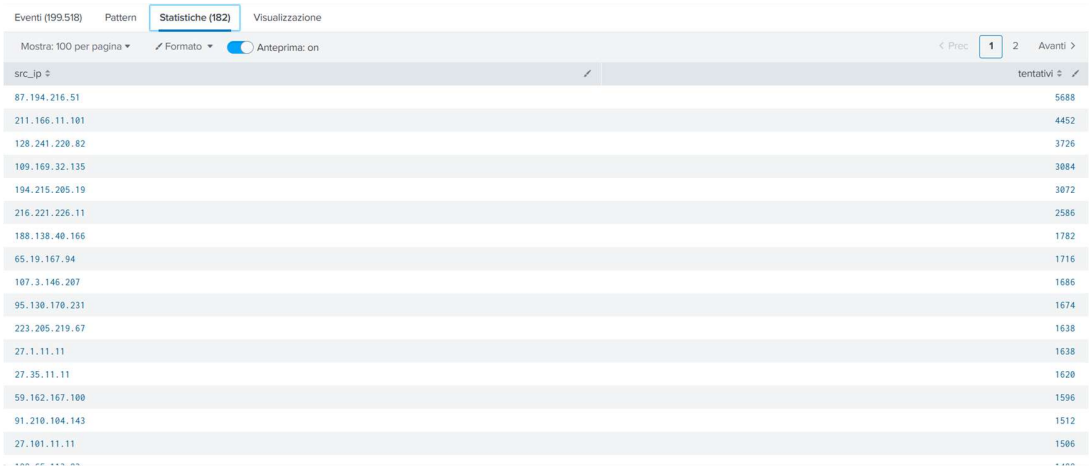


Tabella con indirizzo IP e numero di tentativi

Identificati indirizzi IP che hanno superato la soglia di 5 tentativi falliti, potenzialmente indicativi di attività automatizzate. Questa statistica permette l'implementazione di regole IDS/IPS per blocchi dinamici.
Rischio: attacchi di forza bruta distribuiti.
Mitigazione: tarare le soglie di blocco, integrare sistemi di rate limiting e allertare il SOC in caso di picchi.

3.5 Errori 'Internal Server Error'

Query: source=”tutorialdata.zip:*” “500” | rex “(?<status>\b500\b)” | table _time, host, source, status | sort -_time

Descrizione: source=”tutorialdata.zip:*” “500” cerca in tutti I log all’interno del file zip e cerca I log che contengono I codici HTTP 500(Internal Server Error)

| rex “(?<status>\b500\b)” estate il codice in un campo

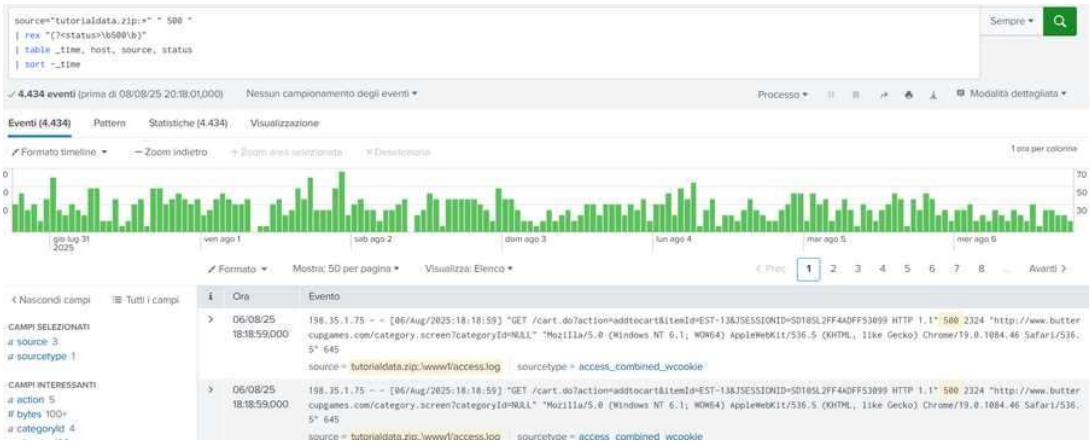
| table _time, host, source, status | sort -_time ordina le informazioni essenziali

>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[5276]: Failed password for invalid user appserver from 194.8.74.23 port 3351 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[1039]: Failed password for root from 194.8.74.23 port 3768 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[5258]: Failed password for invalid user testuser from 194.8.74.23 port 3626 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[1165]: Failed password for apache from 194.8.74.23 port 4604 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[3760]: Failed password for invalid user mongodb from 194.8.74.23 port 2472 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[4998]: Failed password for mail from 194.8.74.23 port 1552 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[1930]: Failed password for games from 194.8.74.23 port 3007 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[5801]: Failed password for invalid user desktop from 194.8.74.23 port 2285 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[3759]: Failed password for nagios from 194.8.74.23 port 3769 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[5979]: Failed password for invalid user cyrus from 194.8.74.23 port 3417 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure
>	06/08/25 13:04:36,000	Thu Aug 06 2025 13:04:36 mailsv1 sshd[4994]: Failed password for invalid user guest from 194.8.74.23 port 2294 ssh2 source = tutorialdata.zip:.\mailsv/secure.log sourcetype = www1/secure

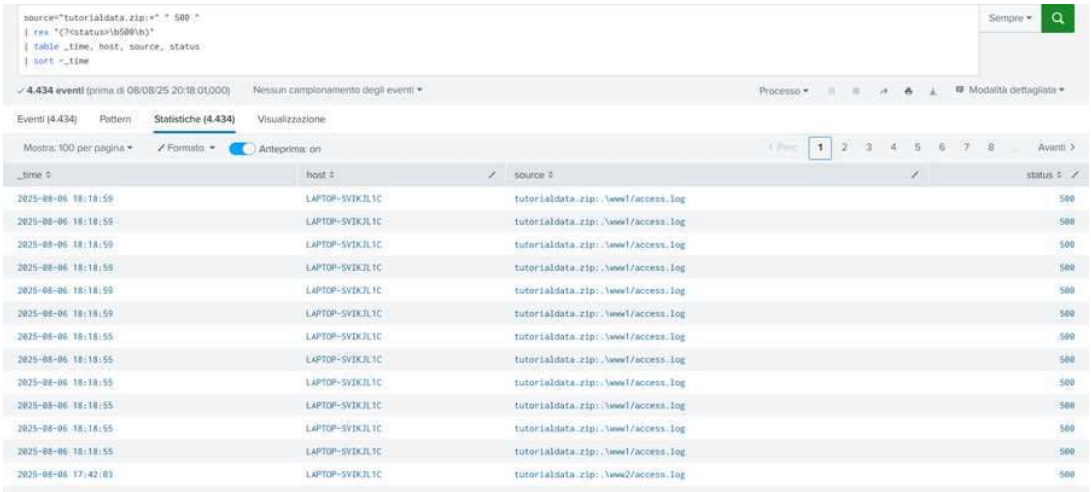
Elenco degli eventi HTTP 500 nel tempo

Password errata (utente valido)	root	87.194.216.51	328
Utente non valido	operator	87.194.216.51	240
Password errata (utente valido)	root	211.166.11.101	232
Password errata (utente valido)	root	59.162.167.108	216
Utente non valido	admin	87.194.216.51	216
Utente non valido	operator	211.166.11.101	208
Utente non valido	administrator	128.241.220.82	200
Password errata (utente valido)	mail	87.194.216.51	192
Password errata (utente valido)	root	128.241.220.82	184
Utente non valido	admin	109.169.32.135	176
Utente non valido	administrator	194.215.205.19	176
Utente non valido	db	87.194.216.51	176
Utente non valido	mailman	87.194.216.51	176
Utente non valido	admin	211.166.11.101	168
Utente non valido	operator	194.215.205.19	168
Password errata (utente valido)	squid	87.194.216.51	160
Utente non valido	email	87.194.216.51	160
Utente non valido	administrator	109.169.32.135	152
Utente non valido	irc	87.194.216.51	152
Utente non valido	sys	87.194.216.51	152

Statistiche per host, indirizzo IP



Timeline con errori 500



Esempi ripetuti di risposte 500

L'analisi dei log HTTP con codice di stato 500 ha evidenziato errori interni del server, potenzialmente legati a vulnerabilità applicative o configurazioni errate. Questi errori possono essere sfruttati da attaccanti per raccogliere informazioni sensibili o compromettere il sistema.
Rischio: esposizione di vulnerabilità applicative.
Mitigazione: revisione del codice, test di penetrazione mirati, correzione delle configurazioni errate.

4. Conclusioni e Raccomandazioni

L'analisi ha permesso di evidenziare tentativi di intrusione, attività sospette e problematiche tecniche. Si raccomanda di:

- Implementare controlli proattivi basati su regole di correlazione.
- Integrare dashboard di monitoraggio in tempo reale.
- Adottare un framework di sicurezza come NIST CSF o ISO 27001.
- Svolgere periodicamente vulnerability assessment e penetration test.

FONTI:

Tutorialdata.zip