

# CPTP0524 – W24D4 - Benchmark M6

## *Analisi del malware e Splunk*

### **Progetto**

Importate su Splunk i dati di esempio “tutorialdata.zip”:

- 1.** Crea una query Splunk per identificare tutti i tentativi di accesso falliti "Failed password". La query dovrebbe mostrare il timestamp, l'indirizzo IP di origine, il nome utente e il motivo del fallimento.
- 2.** Scrivi una query Splunk per trovare tutte le sessioni SSH aperte con successo. La query dovrebbe filtrare per l'utente “djohnson” e mostrare il timestamp e l'ID utente.
- 3.** Scrivi una query Splunk per trovare tutti i tentativi di accesso falliti provenienti dall'indirizzo IP “86.212.199.60”. La query dovrebbe mostrare il timestamp, il nome utente e il numero di porta.
- 4.** Crea una query Splunk per identificare gli indirizzi IP che hanno tentato di accedere ("Failed password") al sistema più di 5 volte. La query dovrebbe mostrare l'indirizzo IP e il numero di tentativi.
- 5.** Crea una query Splunk per trovare tutti gli Internal Server Error. Trarre delle conclusioni sui log analizzati utilizzando AI.

# Svolgimento

**1.** Crea una query Splunk per identificare tutti i tentativi di accesso falliti "Failed password". La query dovrebbe mostrare il timestamp, l'indirizzo IP di origine, il nome utente e il motivo del fallimento.

## Query:

```
source="tutorialdata.zip:*" host="tutorial-data-zip" index=main "Failed password" | rex "Failed password for (invalid user )?(?<user>\w+) from (?<src>\d{1,3}(\.\d{1,3}){3}) port \d+" | eval reason="Failed password" | table _time, src, user, reason
```

✓ 33.253 eventi (prima di 01/05/25 00:06:52,000)

Nessun campionamento degli eventi ▼

Processo ▼

||

■

→

📄

↓

Modalità dettagliata ▼

Eventi (33.253)

Pattern

Statistiche (33.253)

Visualizzazione

100 per pagina ▼

Formato

Anteprima ▼

< Prec

1

2

3

4

5

6

7

8

...

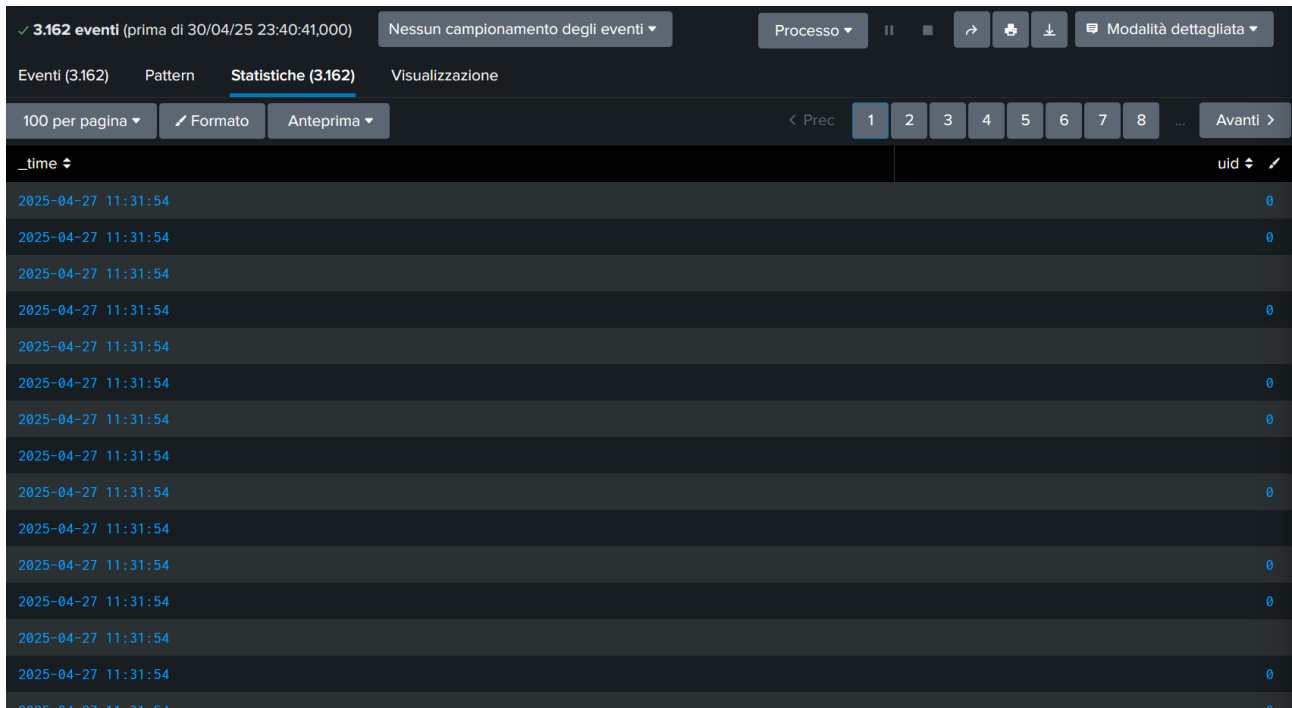
Avanti >

_time ↕	src ↕	user ↕	reason ↕
2025-04-27 11:31:54	194.8.74.23	appserver	Failed password
2025-04-27 11:31:54	194.8.74.23	root	Failed password
2025-04-27 11:31:54	194.8.74.23	testuser	Failed password
2025-04-27 11:31:54	194.8.74.23	apache	Failed password
2025-04-27 11:31:54	194.8.74.23	mongodb	Failed password
2025-04-27 11:31:54	194.8.74.23	mail	Failed password
2025-04-27 11:31:54	194.8.74.23	games	Failed password
2025-04-27 11:31:54	194.8.74.23	desktop	Failed password
2025-04-27 11:31:54	194.8.74.23	nagios	Failed password
2025-04-27 11:31:54	194.8.74.23	cyrus	Failed password
2025-04-27 11:31:54	194.8.74.23	guest	Failed password
2025-04-27 11:31:54	194.8.74.23	itmadmin	Failed password
2025-04-27 11:31:54	194.8.74.23	inet	Failed password
2025-04-27 11:31:54	194.8.74.23	operator	Failed password
2025-04-27 11:31:54	194.8.74.23	irc	Failed password
2025-04-27 11:31:54	194.8.74.23	harrison	Failed password

**2.** Scrivi una query Splunk per trovare tutte le sessioni SSH aperte con successo. La query dovrebbe filtrare per l'utente "djohnson" e mostrare il timestamp e l'ID utente.

**Query:**

```
source="tutorialdata.zip:*" host="tutorial-data-zip" "djohnson" | table _time, uid
```



The screenshot shows the Splunk search results interface. At the top, it indicates 3,162 events. The search bar contains the query: `source="tutorialdata.zip:*" host="tutorial-data-zip" "djohnson" | table _time, uid`. The interface is set to 'Processo' mode. Below the search bar, there are tabs for 'Eventi (3.162)', 'Pattern', 'Statistiche (3.162)', and 'Visualizzazione'. The 'Statistiche' tab is selected. The results are displayed in a table with two columns: '\_time' and 'uid'. The table shows a list of events, all with the same timestamp '2025-04-27 11:31:54' and a 'uid' of '0'. The table is paginated, showing 100 results per page.

_time	uid
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0
2025-04-27 11:31:54	0

**3.** Scrivi una query Splunk per trovare tutti i tentativi di accesso falliti provenienti dall'indirizzo IP "86.212.199.60". La query dovrebbe mostrare il timestamp, il nome utente e il numero di porta.

**Query:**

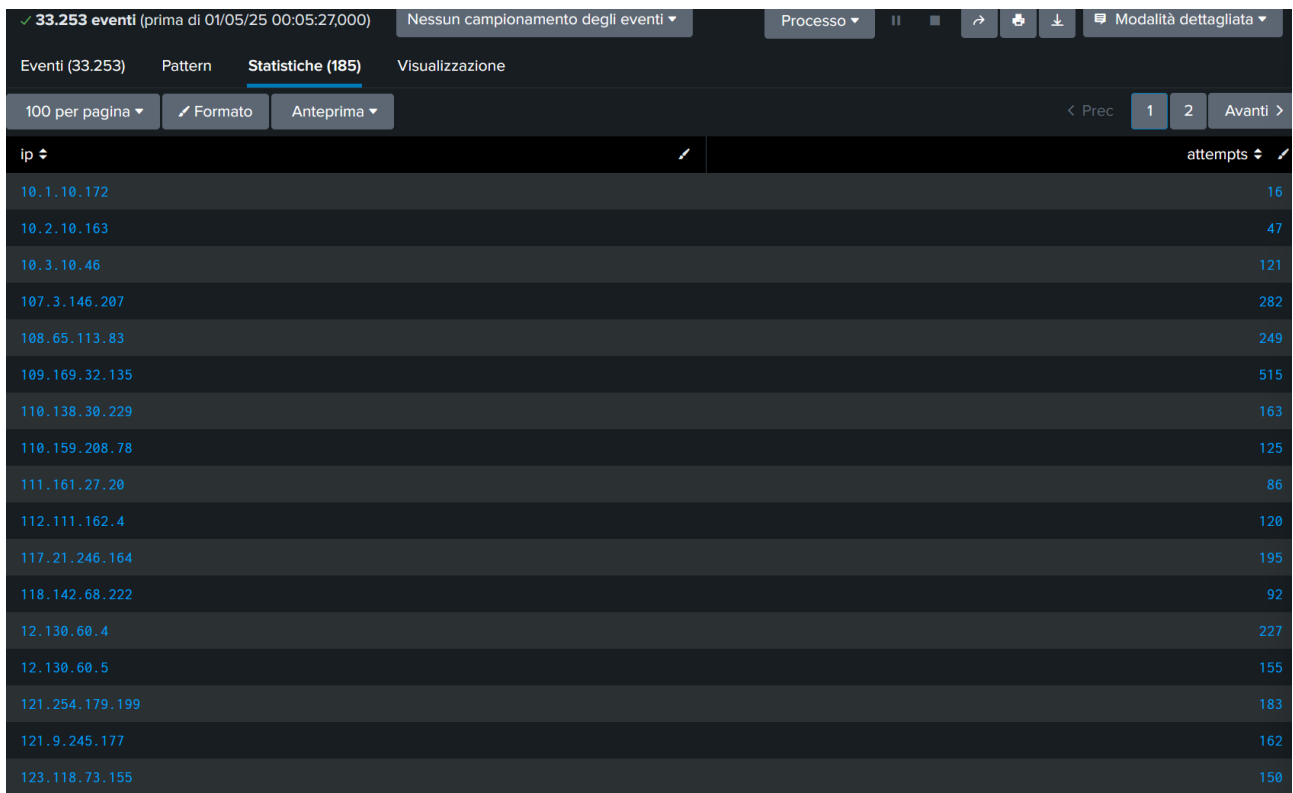
```
source="tutorialdata.zip:*" host="tutorial-data-zip" index=main "Failed password" "86.212.199.60" | rex "Failed password for (invalid user )?(?<user>\w+) from 86\.212\.199\.60 port (?<port>\d+)" | table _time, user, port
```

Eventi (158)   Pattern <b>Statistiche (158)</b> Visualizzazione			
100 per pagina ▾	Formato	Anteprima ▾	< Prec   1   2   Avanti >
_time ↕	user ↕		port ↕
2025-04-27 11:31:54	agushto		3692
2025-04-27 11:31:54	tomcat		1464
2025-04-27 11:31:54	desktop		3518
2025-04-27 11:31:54	yp		2856
2025-04-27 11:31:54	mail		1054
2025-04-27 11:31:54	apache		2630
2025-04-27 11:31:54	services		4740
2025-04-27 11:31:54	irc		1203
2025-04-27 11:31:54	mysql		4802
2025-04-27 11:31:54	pmuser		1775
2025-04-27 11:31:54	ventrilo		1465
2025-04-27 11:31:54	system		3305
2025-04-27 11:31:51	db		2690

4. Crea una query Splunk per identificare gli indirizzi IP che hanno tentato di accedere ("Failed password") al sistema più di 5 volte. La query dovrebbe mostrare l'indirizzo IP e il numero di tentativi.

### Query:

```
source="tutorialdata.zip:*" host="tutorial-data-zip" index=main "Failed password" | rex "from (?<ip>\d{1,3}(?:\.\d{1,3}){3})" | stats count as attempts by ip | where attempts > 5
```



The screenshot shows the Splunk search results interface. At the top, it indicates 33,253 events. The search bar contains the query: `source="tutorialdata.zip:*" host="tutorial-data-zip" index=main "Failed password" | rex "from (?<ip>\d{1,3}(?:\.\d{1,3}){3})" | stats count as attempts by ip | where attempts > 5`. The results are displayed in a table with two columns: 'ip' and 'attempts'. The table shows 18 rows of data, listing various IP addresses and their corresponding number of failed password attempts.

ip	attempts
10.1.10.172	16
10.2.10.163	47
10.3.10.46	121
107.3.146.207	282
108.65.113.83	249
109.169.32.135	515
110.138.30.229	163
110.159.208.78	125
111.161.27.20	86
112.111.162.4	120
117.21.246.164	195
118.142.68.222	92
12.130.60.4	227
12.130.60.5	155
121.254.179.199	183
121.9.245.177	162
123.118.73.155	150

5. Crea una query Splunk per trovare tutti gli Internal Server Error. Trarre delle conclusioni sui log analizzati utilizzando AI.

### Query:

```
source="tutorialdata.zip:*" host="tutorial-data-zip" index=main index=main  
status=500 | table _time, host, source, _raw
```

Eventi (733)   Pattern   Statistiche (733)   Visualizzazione			
100 per pagina ▾	Formato	Anteprima ▾	< Prec   1   2   3   4   5   6   7   8   Avanti >
_time ▾	host ▾	source ▾	_raw ▾
2025-04-27 18:18:59	tutorial-data-zip	tutorialdata.zip:/www1/access.log	198.35.1.75 - - [27/Apr/2025:18:18:59] "GET /cart.do?action=addtocart&itemId=EST-13&JSESSIONID=SD10SL2FF4ADFF53099 HTTP 1.1" 500 2324 "http://www.buttercupgames.com/category.screen?categoryId=NULL" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.46 Safari/536.5" 645
2025-04-27 18:18:55	tutorial-data-zip	tutorialdata.zip:/www1/access.log	198.35.1.75 - - [27/Apr/2025:18:18:55] "GET /product.screen?productId=SF-BVS-G01&JSESSIONID=SD10SL2FF4ADFF53099 HTTP 1.1" 500 2809 "http://www.buttercupgames.com/cart.do?action=view&itemId=EST-14" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.46 Safari/536.5" 370
2025-04-27 17:42:03	tutorial-data-zip	tutorialdata.zip:/www2/access.log	125.89.78.6 - - [27/Apr/2025:17:42:03] "POST /cart.do?action=changequantity&itemId=EST-16&JSESSIONID=SD10SL8FF3ADFF52952 HTTP 1.1" 500 1165 "http://www.buttercupgames.com/product.screen?productId=SF-BVS-G01" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_4) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.46 Safari/536.5" 230
2025-04-27 17:17:00	tutorial-data-zip	tutorialdata.zip:/www1/access.log	194.146.236.22 - - [27/Apr/2025:17:17:00] "POST /product.screen?productId=SF-BVS-G01&JSESSIONID=SD4SL3FF2ADFF52813 HTTP 1.1" 500 299 "http://www.buttercupgames.com/cart.do?action=remove&itemId=EST-13" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.2.28) Gecko/20120306 YFF3 Firefox/3.6.28 (.NET CLR 3.5.30729; .NET4.0C)" 749
2025-04-27 17:15:13	tutorial-data-zip	tutorialdata.zip:/www3/access.log	121.254.179.199 - - [27/Apr/2025:17:15:13] "POST /product.screen?productId=SF-BVS-G01&JSESSIONID=SD0SL9FF10ADFF52799 HTTP 1.1" 500 2243 "http://www.buttercupgames.com/oldlink?itemId=EST-13" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_4) AppleWebKit/536.5 (KHTML, like Gecko) Chrome/19.0.1084.46 Safari/536.5" 642
2025-04-27 16:54:07	tutorial-data-zip	tutorialdata.zip:/www3/access.log	76.89.103.115 - - [27/Apr/2025:16:54:07] "GET /cart.do?action=remove&itemId=EST-17&JSESSIONID=SD1SL9FF9ADFF52693 HTTP 1.1" 500 2606 "http://www.buttercupgames.com/category.screen?categoryId=NULL" "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; Trident/4.0; .NET CLR 2.0.50727; MS-BTC LM 8.0; TeFePath 23" 158

## Conclusioni tratte dai log (Internal Server Error)

Dall'analisi dei log access.log dei server www1, www2 e www3, sono stati rilevati numerosi eventi con codice di risposta **HTTP 500**, indicanti **errori interni del server**.

### 🔴 Osservazioni principali:

- Gli errori 500 si verificano su diverse rotte applicative, in particolare:
  - /cart.do
  - /product.screen
  - /category.screen
- Gli errori provengono da **diversi indirizzi IP** internazionali, indicando che **non si tratta di un attacco mirato** ma probabilmente di **problemi applicativi**.
- Tutti gli host (www1, www2, www3) sono coinvolti, il che suggerisce un problema **a livello di codice condiviso o backend comune** (es. database, API, sessioni).

### **Ipotesi AI-based:**

- Potrebbe trattarsi di **bug nel codice** che gestisce il carrello o la visualizzazione dei prodotti.
- I problemi potrebbero essere legati alla gestione della sessione (JSESSIONID) o a risorse non disponibili al momento della richiesta.
- È raccomandato analizzare i **log dell'applicazione** (non solo Apache), eseguire **test sulle API**, e monitorare l'uso di risorse per identificare colli di bottiglia o errori di runtime.

### **Azioni consigliate:**

- Analisi approfondita del codice e dei log applicativi relativi agli endpoint colpiti.
- Aggiungere controlli di errore (es. try/catch, fallback logici).
- Testare il comportamento con carichi simulati (stress test).

Francesco Rinaldi