

# Documentazione Plugin PCF

Alex Frisoni

11 aprile 2025

## Introduzione

Questo documento racchiude la documentazione dei plugin SpiderFootScan e zapScan per la piattaforma PCF.

## Indice

<b>1</b>	<b>Plugin SpiderFootScan</b>	<b>1</b>
1.1	Panoramica . . . . .	1
1.2	Installazione . . . . .	2
<b>2</b>	<b>Plugin ZAP</b>	<b>2</b>
2.1	Panoramica . . . . .	2
2.2	Installazione . . . . .	2
<b>3</b>	<b>Punti critici</b>	<b>2</b>
3.1	Creazione service . . . . .	2
3.2	Aggiornamento services . . . . .	2
<b>4</b>	<b>Miglioramenti</b>	<b>3</b>

## 1 Plugin SpiderFootScan

### 1.1 Panoramica

Importa risultati da SpiderFoot organizzando:

- Host e servizi correlati
- Dati DNS/WHOIS
- Vulnerabilità con punteggio automatico
- Note strutturate per categoria evento

## 1.2 Installazione

```
1 # Copiare nella cartella plugins
2 cp -r spiderfootScan/ pcf/routes/ui/tools_addons/import_plugins/
3 # Installare dipendenze
4 pip install os json base64
```

## 2 Plugin ZAP

### 2.1 Panoramica

1. Validazione struttura report XML con schema XSD
2. Creazione relazioni IP-Porta
3. Generazione PoC dettagliati

### 2.2 Installazione

```
1 # Copiare nella cartella plugins
2 cp -r zapScan/ pcf/routes/ui/tools_addons/import_plugins/
3 # Installare dipendenze
4 pip install os json base64
```

## 3 Punti critici

### 3.1 Creazione service

```
1 {idPort: [ "0", "idHostname1"]}
```

### 3.2 Aggiornamento services

Esempio di con una sola porta e un solo hostname:

```
1 {"bc4467dc-5e83-470e-8a19-ff05e85ee13f": ["0", "076a64a2-6e66-49d0-8624-19ff36a886c2"]}
```

Esempio di con una sola porta e nessun hostname:

```
1 {"a0072228-a8d7-4fb1-b873-f5266147f8a6": ["0"]}
```

Esempio di con una sola porta e più hostname:

```
1 {"8b088405-b5fb-44b2-8fea-f84c82585337": ["0", "5b0b7655-7f33-4f2e-a8bc-d672d570a8c6", "312e8275-53c2-4b15-825d-787b744ba67f", "4ae05a9f-ad8a-4744-9940-821c2a912d45"]\}
```

## 4 Miglioramenti

- **Parsing del testo migliorato**
  - Riconoscimento di indirizzi IP, URL e formati speciali
  - Pulizia di caratteri non validi/spazi multipli
- **Controllo note duplicate**
  - Implementare un controllo anti-duplicati per le nuove note.
- **Migliorare il codice**
  - Suddivisione in moduli riutilizzabili
  - Gestione errori dettagliata con log
  - Ottimizzazione prestazioni per grandi file