Query di ricerca orbitali di base per l'analisi delle minacce

Sommario

Introduzione

Questo documento descrive le query orbitali di base per l'analisi delle minacce.

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dell'interesse nella comprensione delle minacce e del malware e una conoscenza di base delle tabelle SQL (Structured Query Language).

Componenti usati

Le informazioni fornite in questo documento si basano sulle seguenti versioni software e hardware:

- Secure Endpoint Connector versione 7.1.5 o successiva per Windows
- Secure Endpoint Connector versione 1.16 o successiva per Mac
- Secure Endpoint Connector versione 1.17 o successive per Linux
- All'utente dell'endpoint sicuro deve essere assegnato il ruolo di amministratore per distribuire

Orbital

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

Premesse

Le query personalizzate sono sfruttate che devono aiutarvi a imparare rapidamente la potenza di Orbital e osquery per la caccia alle minacce.

Orbital utilizza tavole delle scorte osquerys oltre a quelle specifiche di Orbital. I risultati restituiti tramite Orbital possono essere inviati ad altre applicazioni, ad esempio Secure Endpoint, Secure Malware Analytics e SecureX Threat Response, e possono essere archiviati in archivi dati remoti, ad esempio Amazon S3, Microsoft Azure e Splunk.

Utilizzare la pagina Indagine orbitale per creare ed eseguire query dinamiche sugli endpoint al fine di ottenere ulteriori informazioni. Orbital utilizza osquery, che consente di eseguire query sui dispositivi come un database con comandi SQL di base.

Di seguito è riportato un semplice esempio: SELECT colonna1, colonna2 FROM tabella1, tabella2 WHERE colonna2='valore'.

In questo esempio, colonna1 e colonna2 sono i nomi dei campi della tabella da cui si desidera scegliere i dati. Per scegliere tutti i campi disponibili nella tabella, utilizzare la sintassi seguente: SELECT * FROM tabella1.

Accesso

Aprire Orbital direttamente nei seguenti siti:

Nord America - https://orbital.amp.cisco.com

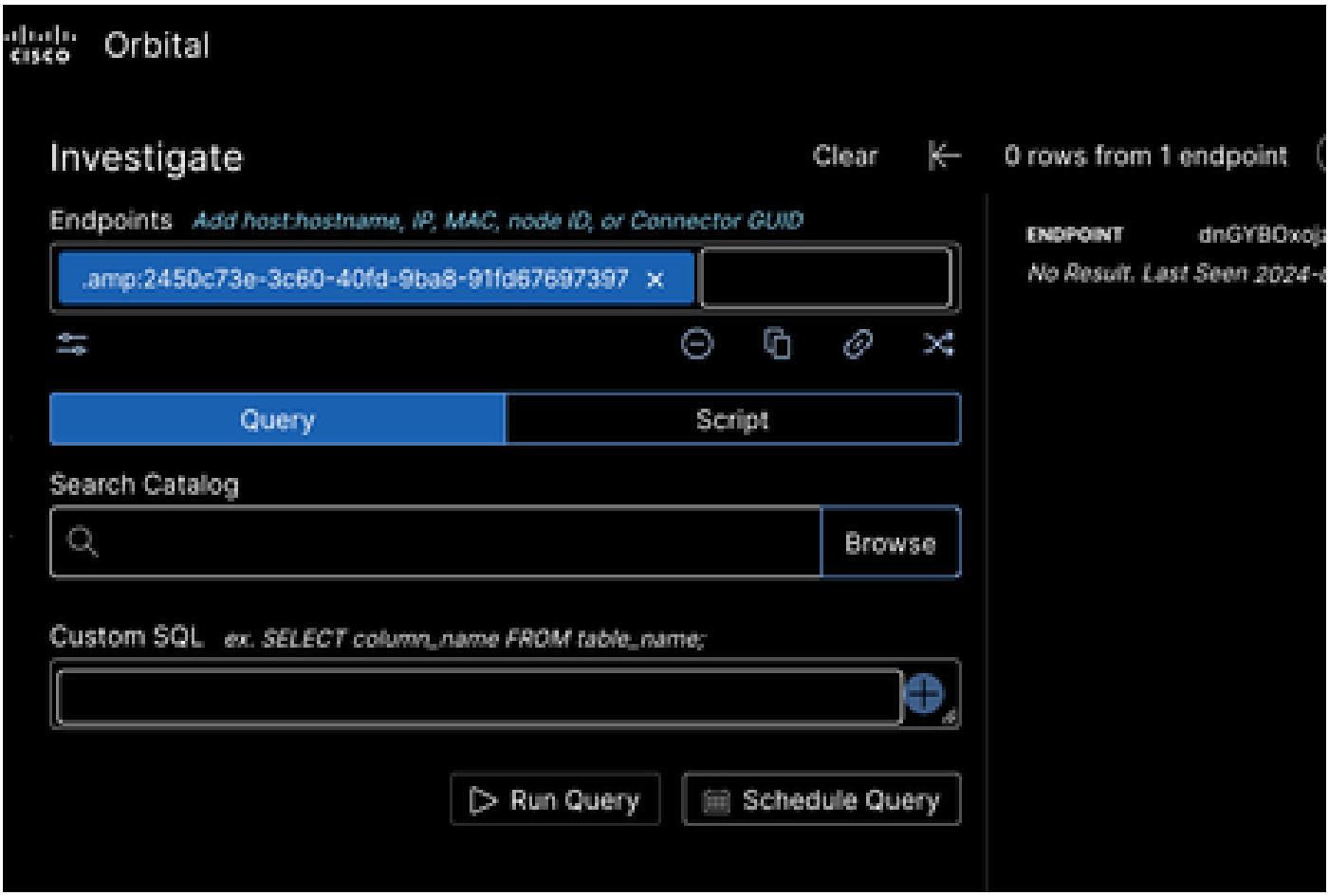
Europa - https://orbital.eu.amp.cisco.com

Asia-Pacifico - https://orbital.apjc.amp.cisco.com

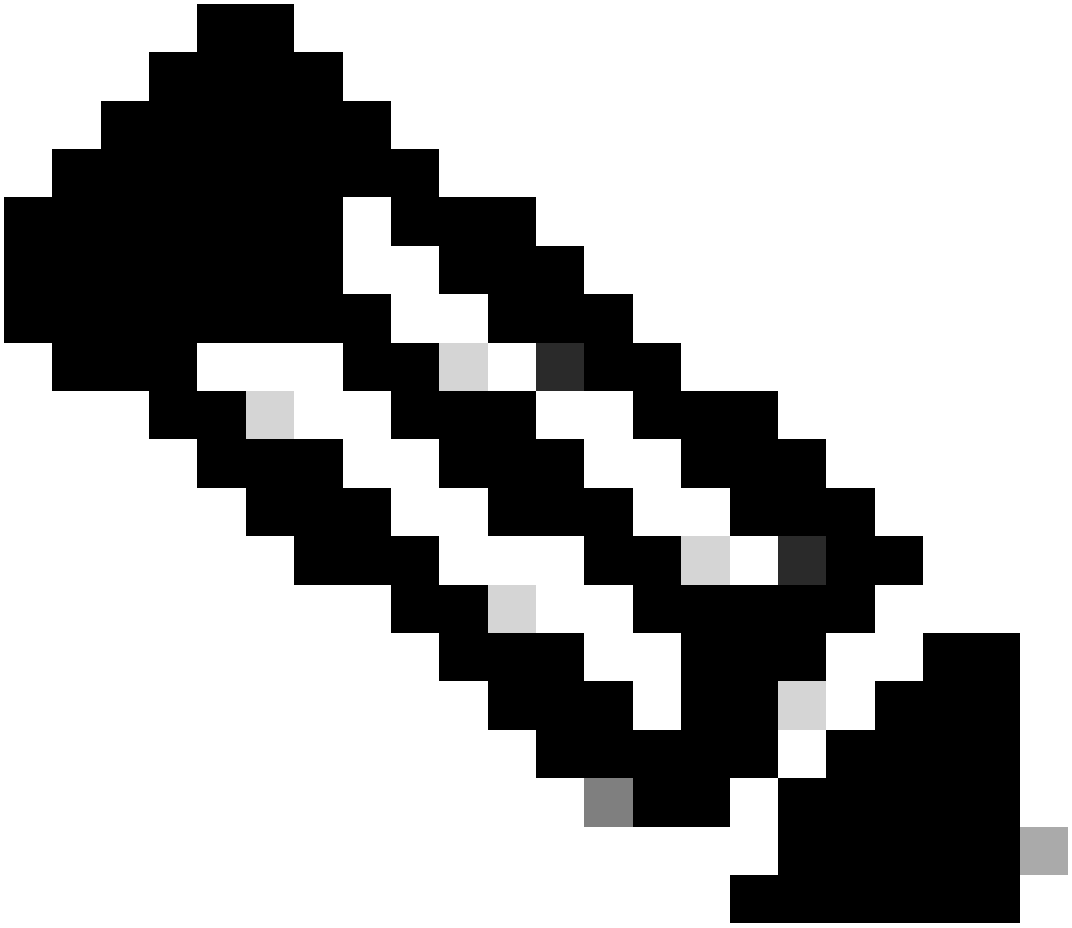
O

In Secure Endpoint Console, scegliere il sistema host interessato e fare clic su Investigate in Orbital.

Sono disponibili opzioni per l'utilizzo del Catalogo orbitale (fare clic su Sfoglia) o Immettere le query personalizzate nella sezione SQL personalizzato, come indicato di seguito:



Query personalizzate

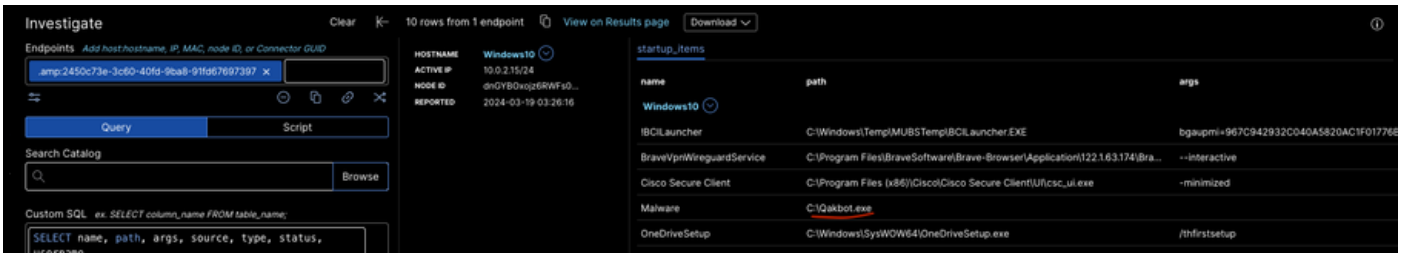


Nota: Il sistema host si trova nella rete lab e si tenta di mantenere inalterato il sistema/la rete.

1. Elementi di avvio

Gli elementi di avvio possono essere sfruttati dagli utenti malintenzionati per mantenere la persistenza in un sistema compromesso, ovvero il software dannoso continuerà a funzionare o verrà riavviato automaticamente a ogni riavvio del sistema. Nell'esempio successivo, Qakbot.exe è in esecuzione nel sistema host.

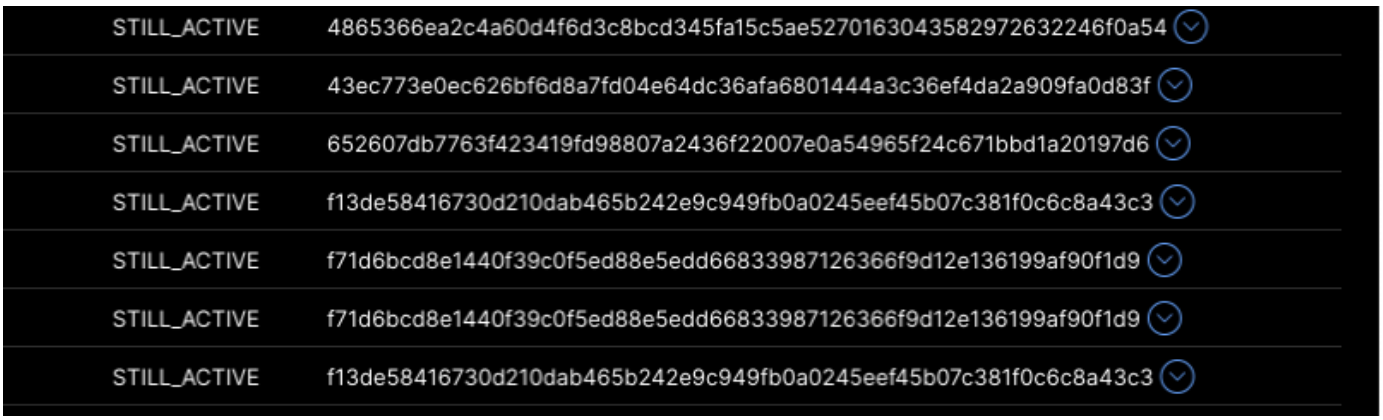
```
SELECT name, path, args, source, type, status, username
FROM startup_items;
```



2. Hash Sha256 dei processi in esecuzione

Gli hash SHA256 non sono intrinsecamente associati all'esecuzione di processi nel relativo stato naturale. Tuttavia, il software di sicurezza e gli strumenti di monitoraggio del sistema possono calcolare l'hash SHA256 di un processo in esecuzione del file eseguibile per aiutarne a verificarne l'integrità e l'autenticità.

```
 SELECT
p.pid, p.name, p.path, p.cmdline, p.state, h.sha256
FROM processes p
INNER JOIN hash h
ON p.path=h.path;
```

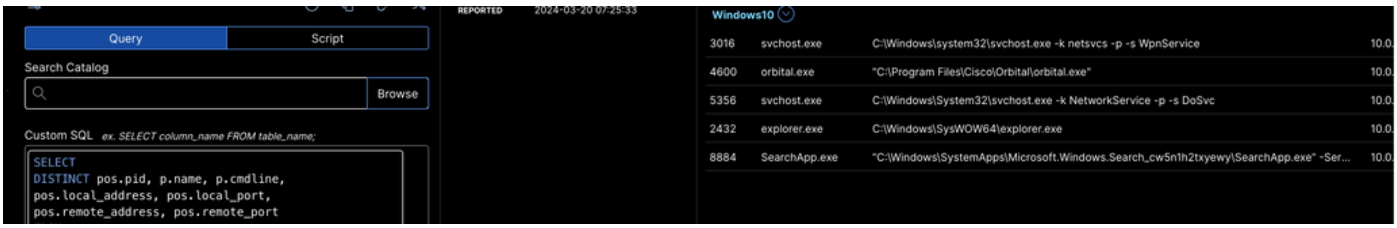


se l'hash associato di un file è dannoso, sarà possibile identificarsi con questa query.

3. Processo con connessioni di rete

I processi con connessioni di rete sono programmi o servizi di sistema che utilizzano attivamente l'interfaccia di rete per comunicare con altri dispositivi in una rete o tramite Internet.

```
SELECT
DISTINCT pos.pid, p.name, p.cmdline, pos.local_address, pos.local_port, pos.remote_address, pos.remote_
FROM processes p
JOIN process_open_sockets pos USING (pid)
WHERE
pos.remote_address NOT IN ("", "0.0.0.0", "127.0.0.1", "::", "::1", “0");
```



4. Processo privilegiato con connessione di rete non localhost

Esecuzione di programmi o servizi con autorizzazioni elevate (come quelli di un account amministratore o di sistema) e comunicazione in rete con un dispositivo o un servizio esterno, ovvero qualsiasi indirizzo IP diverso da 127.0.0.1 (localhost) o ::1 (localhost IPv6).

```
SELECT DISTINCT p.name, p.cmdline, pos.pid, pos.local_address, pos.local_port, pos.remote_address, pos.
FROM processes p JOIN process_open_sockets pos USING (pid)
WHERE pos.remote_address NOT IN ("", "0.0.0.0", "127.0.0.1", "::", "::1")
```

Una volta che si dispone dell'elenco PID (Packet Identifier), è possibile aggiungerlo di conseguenza nelle query personalizzate.

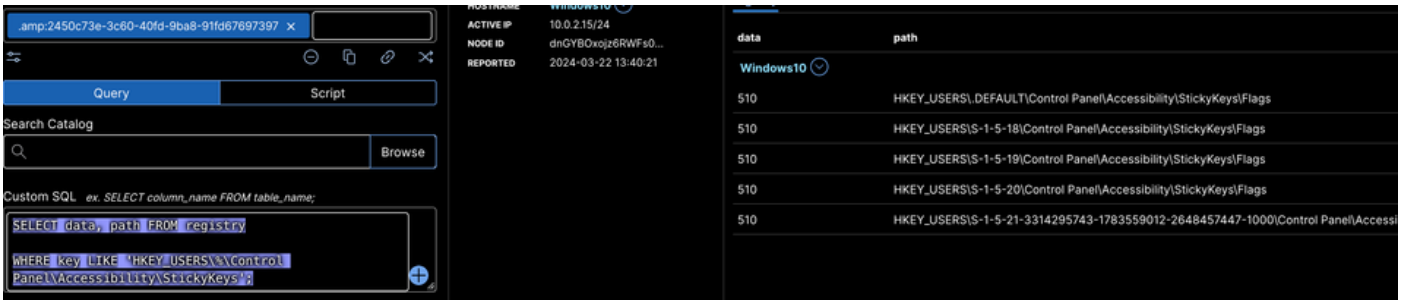
```
SELECT DISTINCT p.name, p.cmdline, pos.pid, pos.local_address, pos.local_port, pos.remote_address, pos.
FROM processes p JOIN process_open_sockets pos USING (pid)
WHERE pos.remote_address NOT IN ("", "0.0.0.0", "127.0.0.1", "::", "::1") and p.uid=1436
```

5. Monitoraggio del Registro di sistema per il backup/ripristino

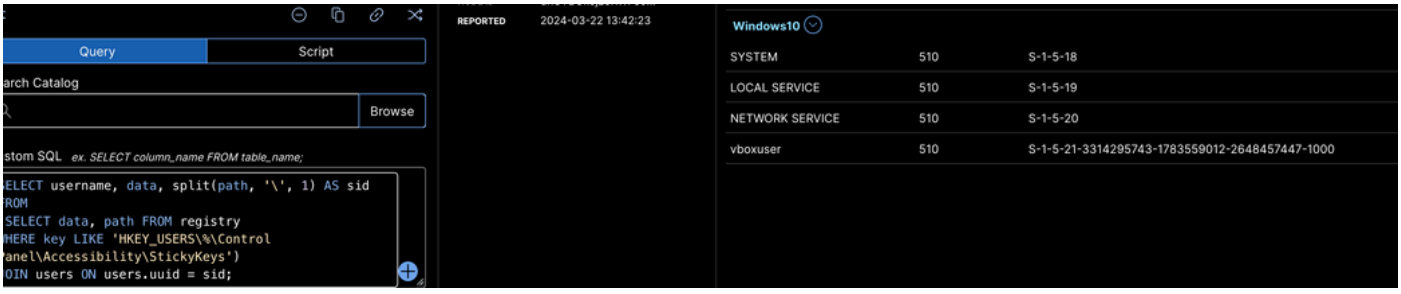
Registrazione degli eventi in cui vengono apportate modifiche al Registro di sistema di Windows tramite operazioni di backup o ripristino. Il Registro di sistema di Windows è un database gerarchico che memorizza le impostazioni e le opzioni di configurazione nei sistemi operativi Microsoft Windows.

```
SELECT key AS reg_key, path, name, data, DATETIME(mtime, "unixepoch") as last_modified
FROM registry
WHERE key LIKE "HKEY_LOCAL_MACHINE\system\currentcontrolset\control\backuprestore\filesnottosnapshot";
```

```
SELECT data, path FROM registry
WHERE key LIKE 'HKEY_USERS\%\Control Panel\Accessibility\StickyKeys';
```




```
SELECT username, data, split(path, '\', 1) AS sid
FROM
(SELECT data, path FROM registry
WHERE key LIKE 'HKEY_USERS\%\Control Panel\Accessibility\StickyKeys')
JOIN users ON users.uuid = sid;
```



6. Ricerca di file

Consente agli utenti di individuare file e cartelle nel computer utilizzando vari criteri, ad esempio il nome del file, il contenuto, le proprietà o i metadati.

```
SELECT
f.directory, f.filename, f.uid, f.gid,
f.mode, f.device, f.size,
f.atime,
f.mtime,
f.ctime,
f.btime,
f.hard_links, f.symlink, f.file_id, h.sha256
```

```
FROM file f
LEFT JOIN hash h on f.path=h.path
WHERE
f.path LIKE (SELECT v from __vars WHERE n="file_path") AND
f.path NOT LIKE (SELECT v from __vars WHERE n="not_file_path");
```

Selezionare PARAMETRI > Percorso file e fare clic su %.dll o %.exe o %.png.

7. Monitoraggio della cronologia di Powershell

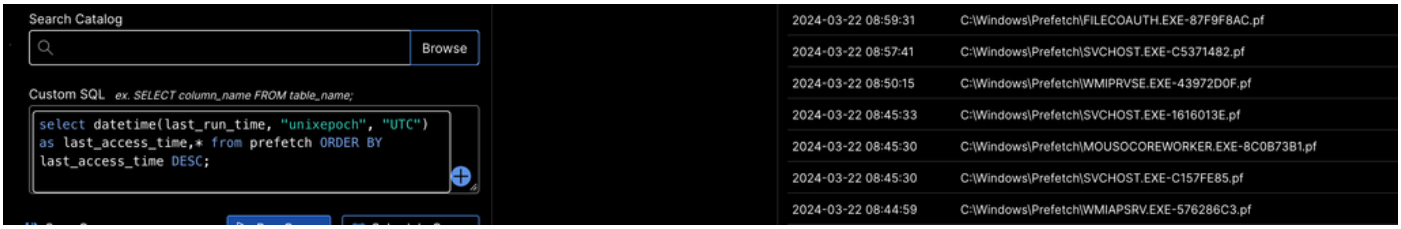
Procedura che consente di tenere traccia dei comandi eseguiti nelle sessioni di PowerShell. Il monitoraggio della cronologia di PowerShell può essere particolarmente importante per motivi di sicurezza e conformità.

```
SELECT time, datetime, script_block_id, script_block_count, script_text, script_name, script_path
FROM orbital_powershell_events
ORDER BY datetime DESC
LIMIT 500;
```

8. Query di prelettura

Funzionalità che accelera il caricamento delle applicazioni. La prelettura comporta l'analisi del modo in cui il software viene caricato ed eseguito su un sistema e quindi l'archiviazione di informazioni su questo in file specifici.

```
select datetime(last_run_time, "unixepoch", "UTC") as last_access_time,*
from prefetch
ORDER BY last_access_time DESC;
```

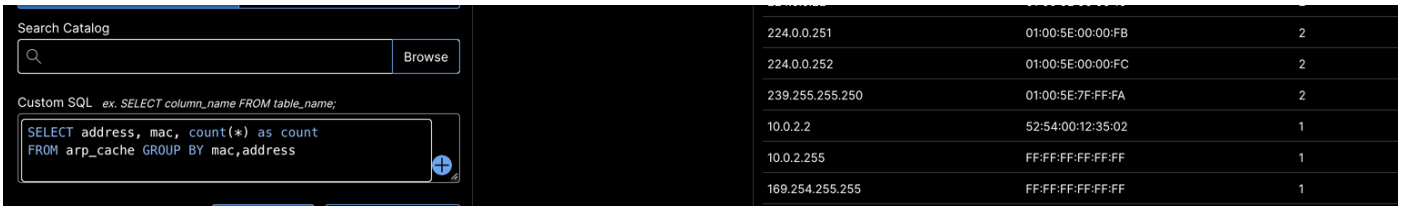


La prelettura è un meccanismo con il quale SQL Server può generare molte richieste di I/O in parallelo per un join di loop nidificato.

9. Ispezione cache ARP (Address Resolution Protocol)

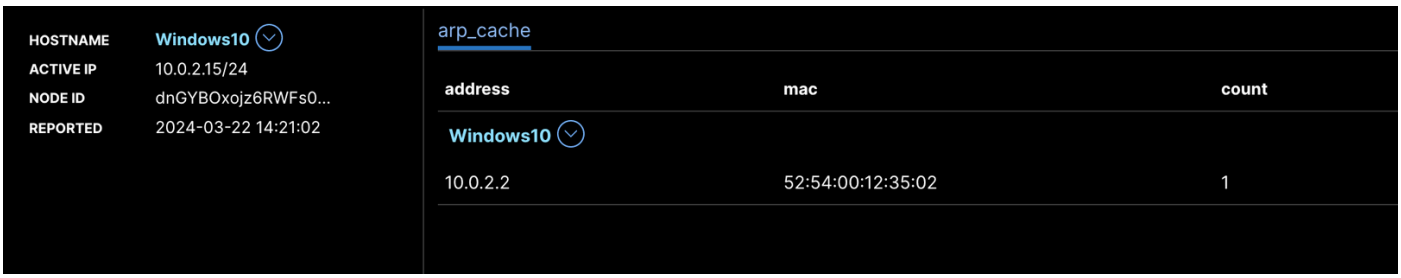
Richiede l'analisi del contenuto della cache ARP in un computer o dispositivo di rete. La cache ARP è una tabella che memorizza i mapping tra gli indirizzi IP e gli indirizzi MAC corrispondenti.

```
SELECT address, mac, count(*) as count
FROM arp_cache GROUP BY mac,address;
```



Nell'esempio seguente viene calcolato l'indirizzo MAC sospetto e il relativo conteggio dalla cache ARP.

```
SELECT address, mac, count(*) as count
FROM arp_cache GROUP BY mac,address
HAVING COUNT(mac) >= (SELECT count FROM arp_cache WHERE count>=1)
AND mac LIKE (SELECT mac FROM arp_cache WHERE mac="52:54:00:12:35:02");
```



Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l’accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).