Requêtes de recherche orbitale de base pour l'analyse des menaces

Table des matières

Introduction

Ce document décrit les requêtes de recherche orbitale de base pour l'analyse des menaces.

Conditions préalables

Exigences

Cisco vous recommande de bien connaître l'intérêt que vous portez à la compréhension des menaces et des programmes malveillants, ainsi que les bases des tables SQL (Structured Query Language).

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Connecteur Secure Endpoint Connector version 7.1.5 ou ultérieure pour Windows
- Connecteur de terminal sécurisé version 1.16 ou ultérieure pour Mac
- Connecteur de terminal sécurisé version 1.17 ou ultérieure pour Linux

- Le rôle d'administrateur doit être attribué à l'utilisateur Secure Endpoint pour déployer Orbital

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l’incidence possible des commandes.

Informations générales

Les requêtes personnalisées sont exploitées, ce qui doit vous aider à apprendre rapidement la puissance d'Orbital et d'osquery pour la chasse aux menaces.

Orbital utilise les tables de stock osquerys en plus des tables spécifiques à Orbital. Les résultats renvoyés via Orbital peuvent être envoyés à d'autres applications, telles que Secure Endpoint, Secure Malware Analytics et SecureX Threat Response, et peuvent être stockés dans des magasins de données distants (RDS), tels qu'Amazon S3, Microsoft Azure et Splunk.

Utilisez la page Orbital Investigate afin de créer et d'exécuter des requêtes en direct sur des terminaux afin d'en collecter davantage d'informations. Orbital utilise osquery, qui vous permet d'interroger vos périphériques comme une base de données avec des commandes SQL de base.

Voici un exemple simple : SELECT column1, column2 FROM table1, table2 WHERE column2='valeur'.

Dans cet exemple, column1 et column2 sont les noms de champ de la table dans laquelle vous souhaitez sélectionner des données. Afin de choisir tous les champs disponibles dans la table, utilisez cette syntaxe : SELECT * FROM table1.

Accès

Ouvrez Orbital directement sur ces sites :

Amérique du Nord - https://orbital.amp.cisco.com

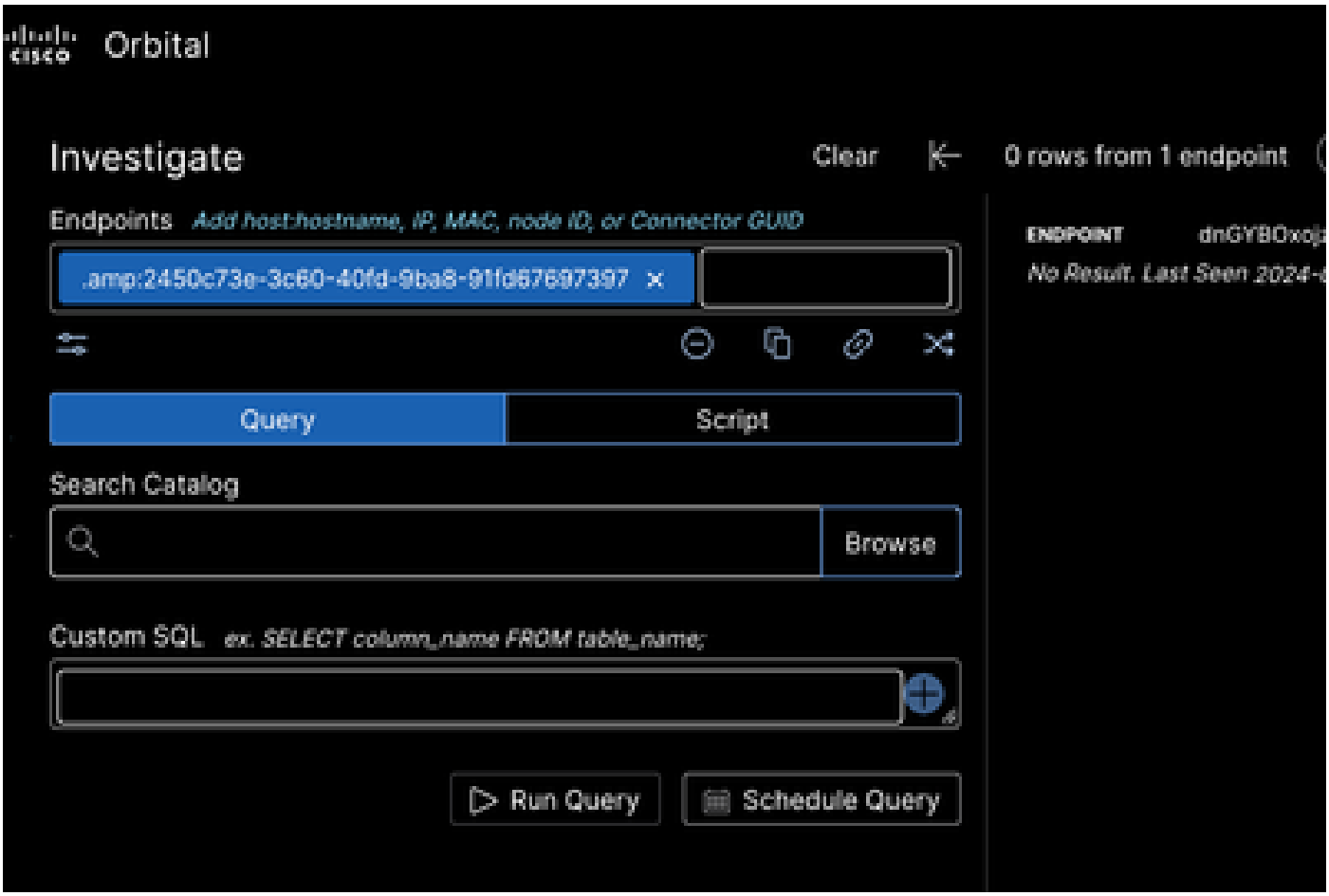
Europe - https://orbital.eu.amp.cisco.com

Asie-Pacifique - https://orbital.apjc.amp.cisco.com

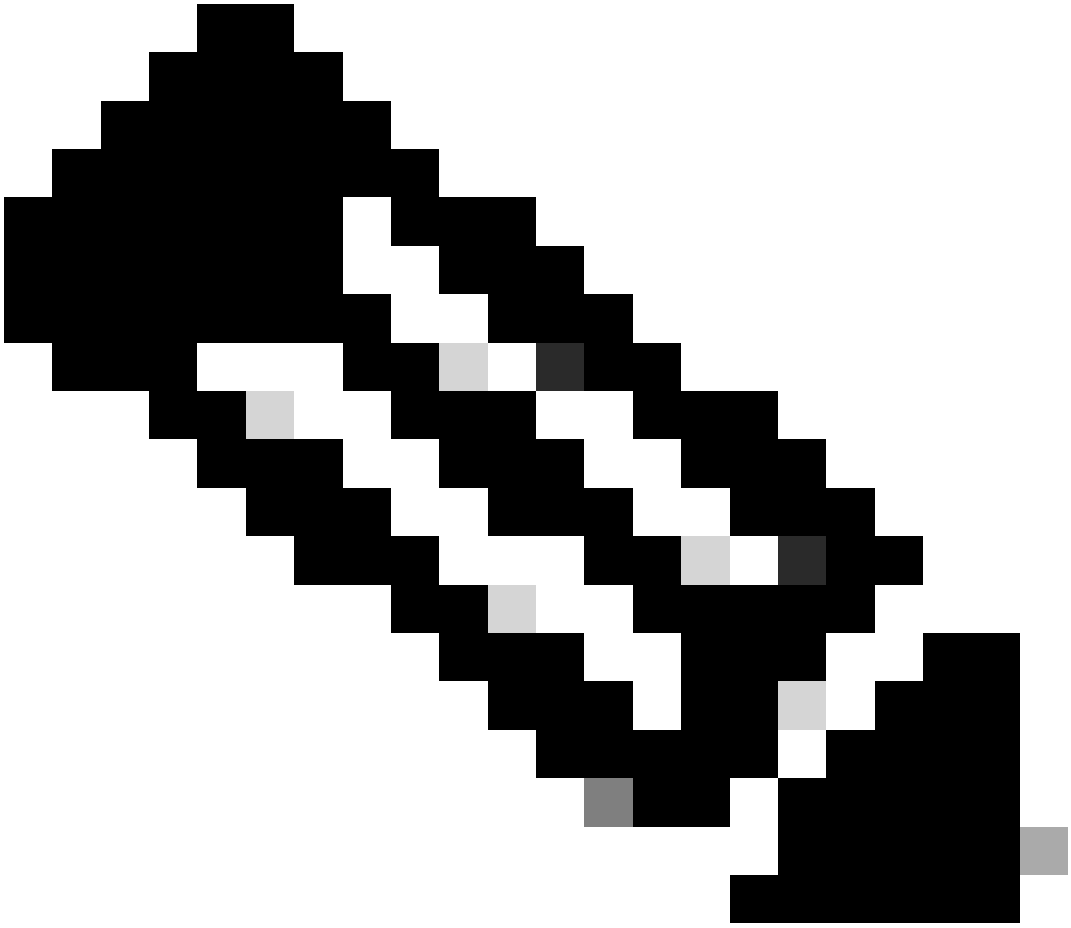
OU

Sur la console Secure Endpoint, sélectionnez le système hôte concerné et cliquez sur Investigate in Orbital.

Il existe des options pour utiliser le catalogue orbital (cliquez sur Parcourir) ou entrez les requêtes personnalisées sous SQL personnalisé section comme mentionné :



Requêtes personnalisées

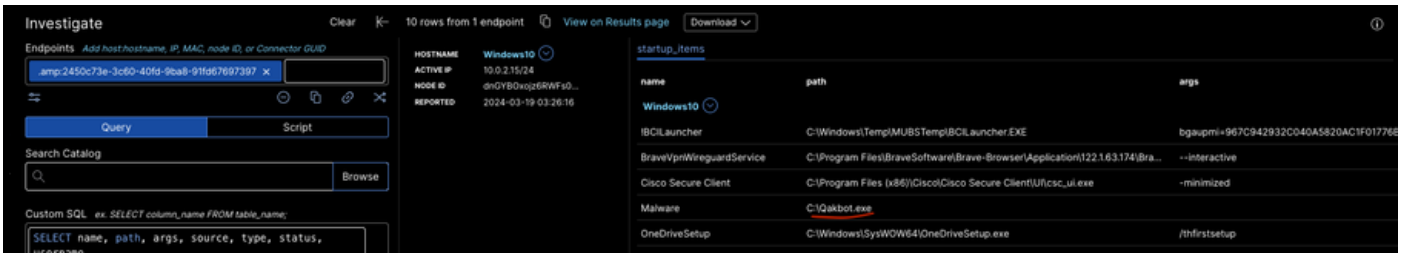


Remarque : Le système hôte se trouve sur le réseau des travaux pratiques et on tente de le maintenir intact.

1. Éléments de démarrage

Les éléments de démarrage peuvent être exploités par les pirates pour maintenir la persistance sur un système compromis, ce qui signifie que le logiciel malveillant continuera à s'exécuter ou à être relancé automatiquement à chaque redémarrage du système. Dans l'exemple suivant, Qakbot.exe s'exécute dans le système hôte.

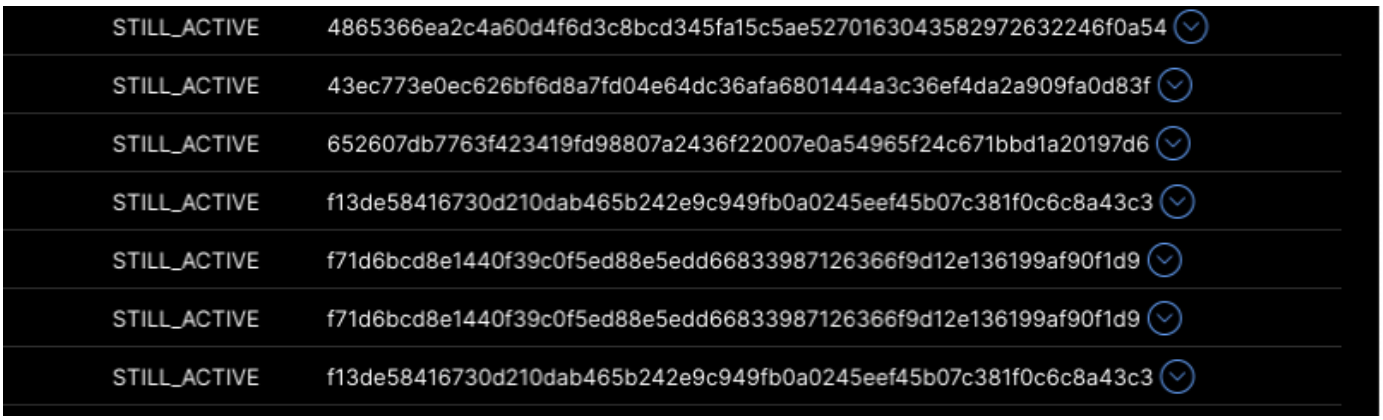
```
SELECT name, path, args, source, type, status, username
FROM startup_items;
```



2. Sha256 Hachages des processus en cours d'exécution

Les hachages SHA256 ne sont pas associés de manière inhérente à l'exécution de processus dans leur état naturel. Cependant, les logiciels de sécurité et les outils de surveillance du système peuvent calculer le hachage SHA256 d'un processus en cours d'exécution du fichier exécutable afin de vérifier son intégrité et son authenticité.

```
 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;
```

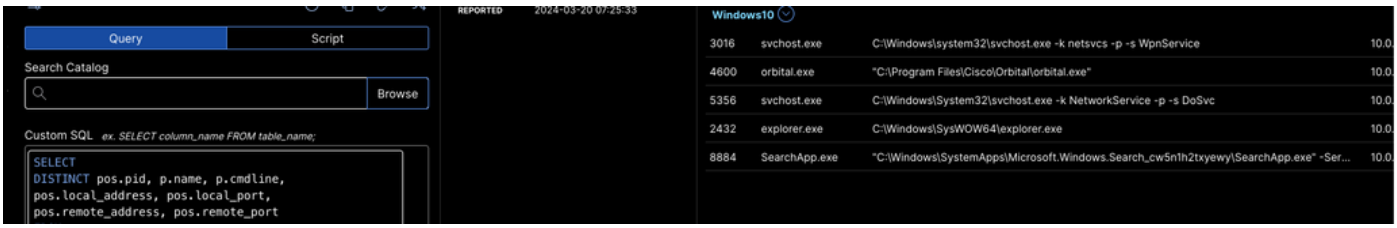


si le hachage associé d'un fichier est malveillant, vous pourrez vous identifier avec cette requête.

3. Processus avec les connexions réseau

Les processus avec des connexions réseau sont des programmes ou des services système qui utilisent activement l'interface réseau afin de communiquer avec d'autres périphériques sur un réseau ou sur 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. Processus privilégié avec une connexion réseau non locale

Exécution d'un programme ou d'un service disposant d'autorisations élevées (comme celles d'un administrateur ou d'un compte système) et communiquant sur le réseau avec un périphérique ou un service externe, ce qui signifie toute adresse IP autre que 127.0.0.1 (localhost) ou ::1 (IPv6 localhost).

```
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")
```

Une fois que vous avez la liste PID (Packet Identifier), vous pouvez l'ajouter en conséquence dans les requêtes personnalisées.

```
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
```

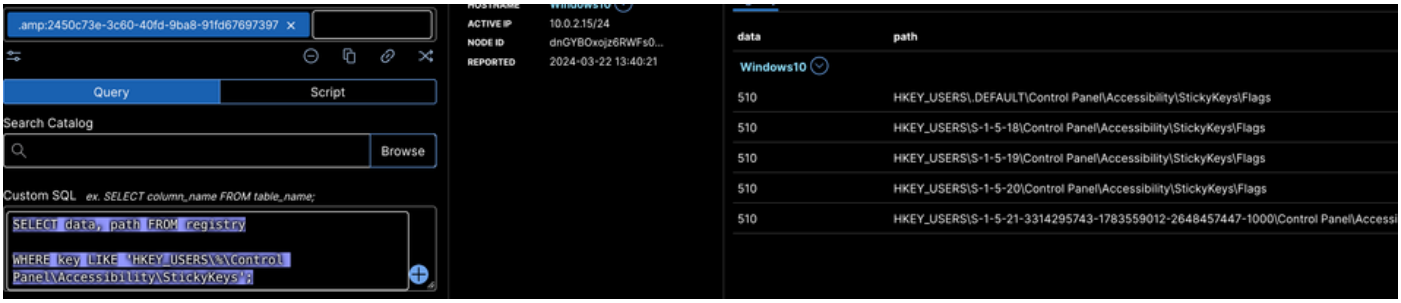
5. Sauvegarde/restauration de la surveillance du registre

Suivi des événements au cours desquels des modifications sont apportées au Registre Windows par le biais d'opérations de sauvegarde ou de restauration. Le Registre Windows est une base de données hiérarchique qui stocke les paramètres et les options de configuration sur les systèmes

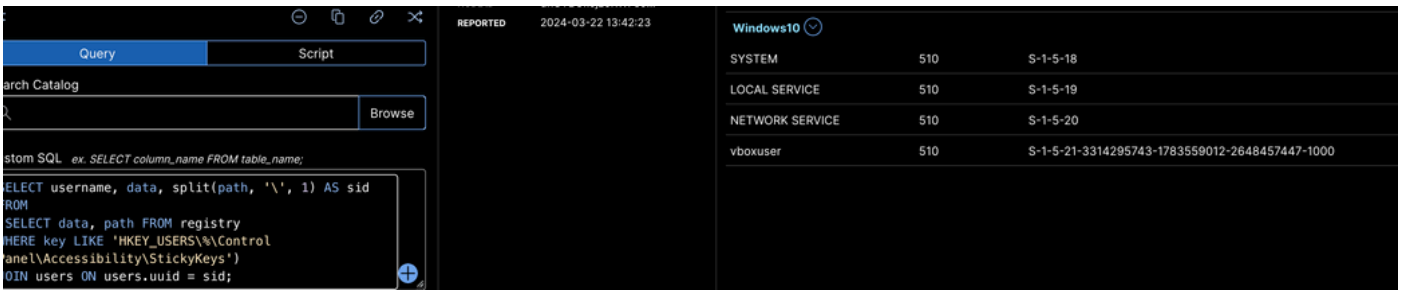
d'exploitation Microsoft Windows.

```
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. Recherche de fichiers

Permet aux utilisateurs de localiser des fichiers et des dossiers sur leur ordinateur en utilisant divers critères tels que le nom de fichier, le contenu, les propriétés ou les métadonnées.

```
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");
```

Accédez à PARAMETERS > File Path et cliquez sur %.dll ou %.exe ou %.png.

7. Surveillance de l'historique Powershell

Pratique consistant à effectuer le suivi des commandes qui ont été exécutées dans les sessions PowerShell. La surveillance de l'historique PowerShell peut s'avérer particulièrement importante pour des raisons de sécurité et de conformité.

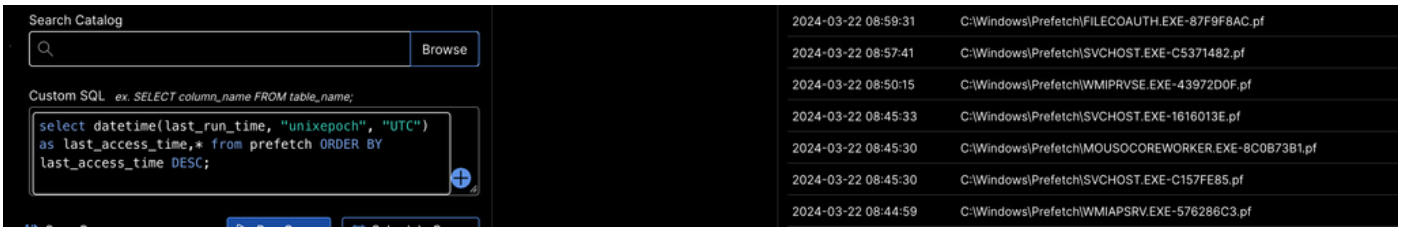
```
SELECT time, datetime, script_block_id, script_block_count, script_text, script_name, script_path
FROM orbital_powershell_events
ORDER BY datetime DESC
LIMIT 500;
```

8. Requête de prérécupération

Fonction de performance qui accélère le chargement des applications. La préextraction consiste à analyser la façon dont le logiciel est chargé et exécuté sur un système, puis à stocker des informations à ce sujet dans des fichiers spécifiques.

```
select datetime(last_run_time, "unixepoch", "UTC") as last_access_time,*
from prefetch
```

```
ORDER BY last_access_time DESC;
```

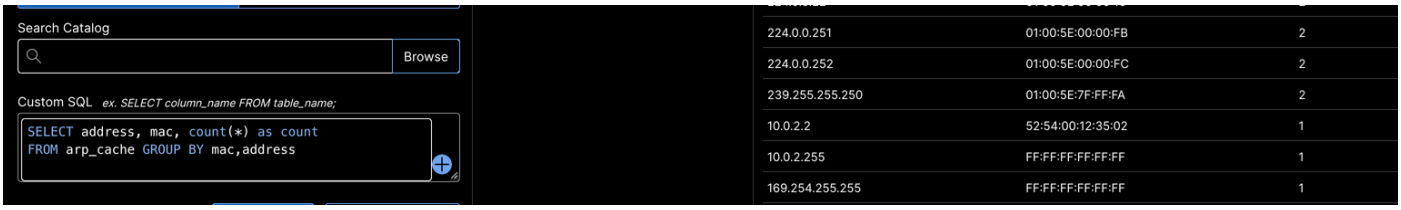


Prefetch est un mécanisme avec lequel SQL Server peut lancer de nombreuses requêtes d'E/S en parallèle pour une jointure par boucle imbriquée.

9. Inspection du cache ARP (Address Resolution Protocol)

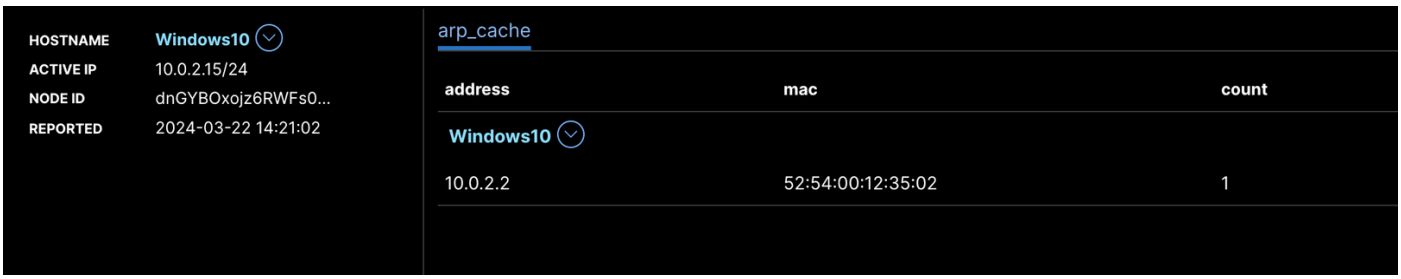
Comprend l'examen du contenu du cache ARP sur un ordinateur ou un périphérique réseau. Le cache ARP est une table qui stocke les mappages entre les adresses IP et leurs adresses MAC correspondantes.

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



L'exemple suivant calcule l'adresse MAC suspecte et son nombre à partir du 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");
```



À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.