How to import Japanese Windows Event Logs into Sumo Logic

dev.classmethod.jp/articles/sumo-logic-sakumashogo-20230606

佐久間昇吾

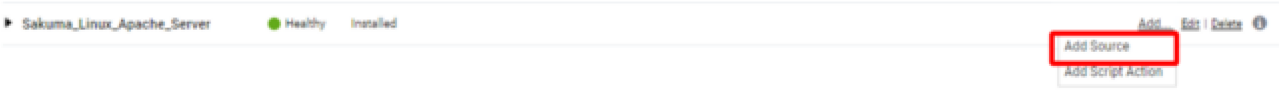
June 7, 2023

In this article, we will discuss how to import Windows Event Logs in a Japanese environment, the benefits of importing logs in JSON format, and event logs that you should keep from a security perspective.

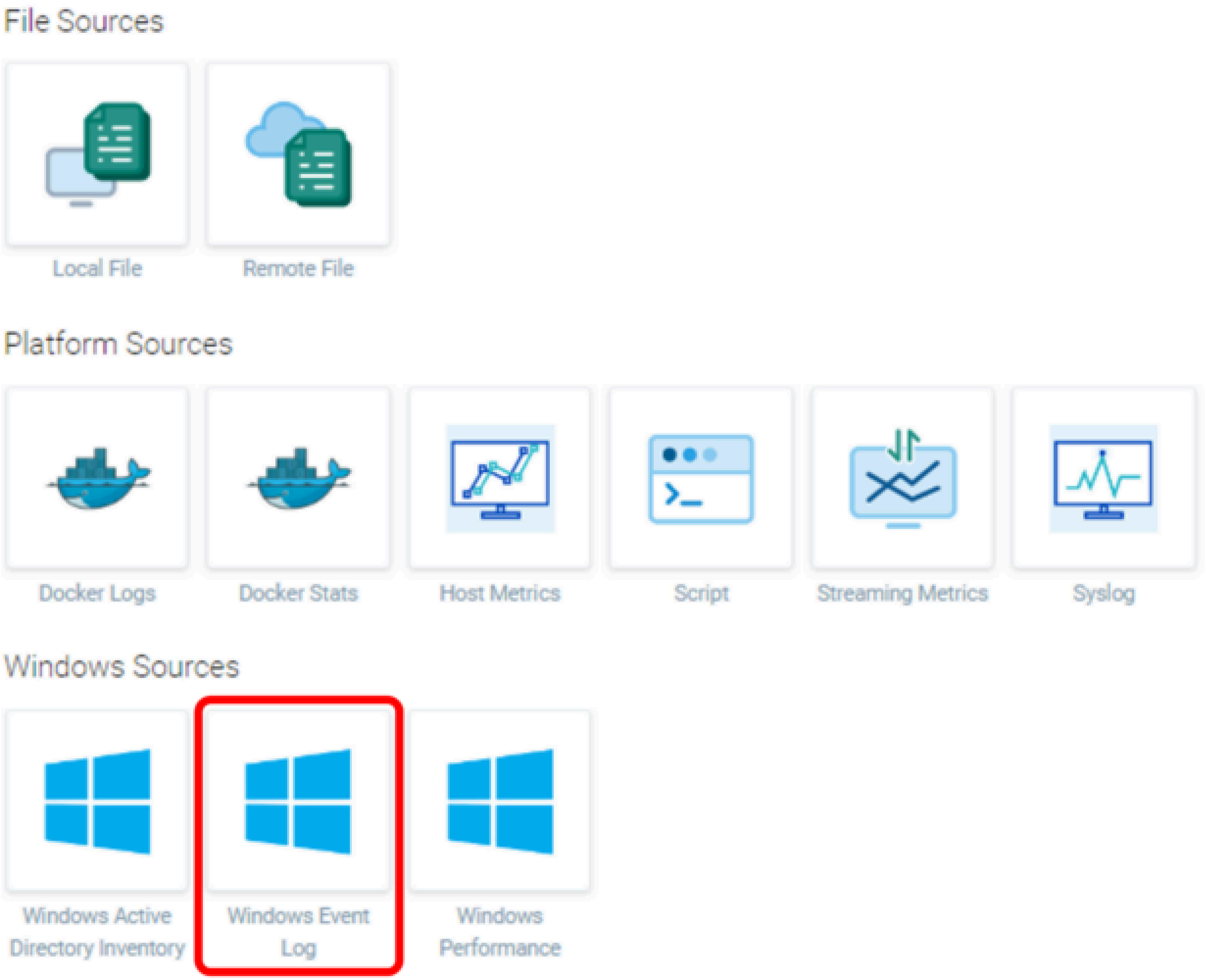
Configure a Windows Event Log Source

If you have not yet installed an Installed Collector on your Windows server, please follow the instructions below. Reference: [Install a Collector on Windows | Sumo Logic Docs]

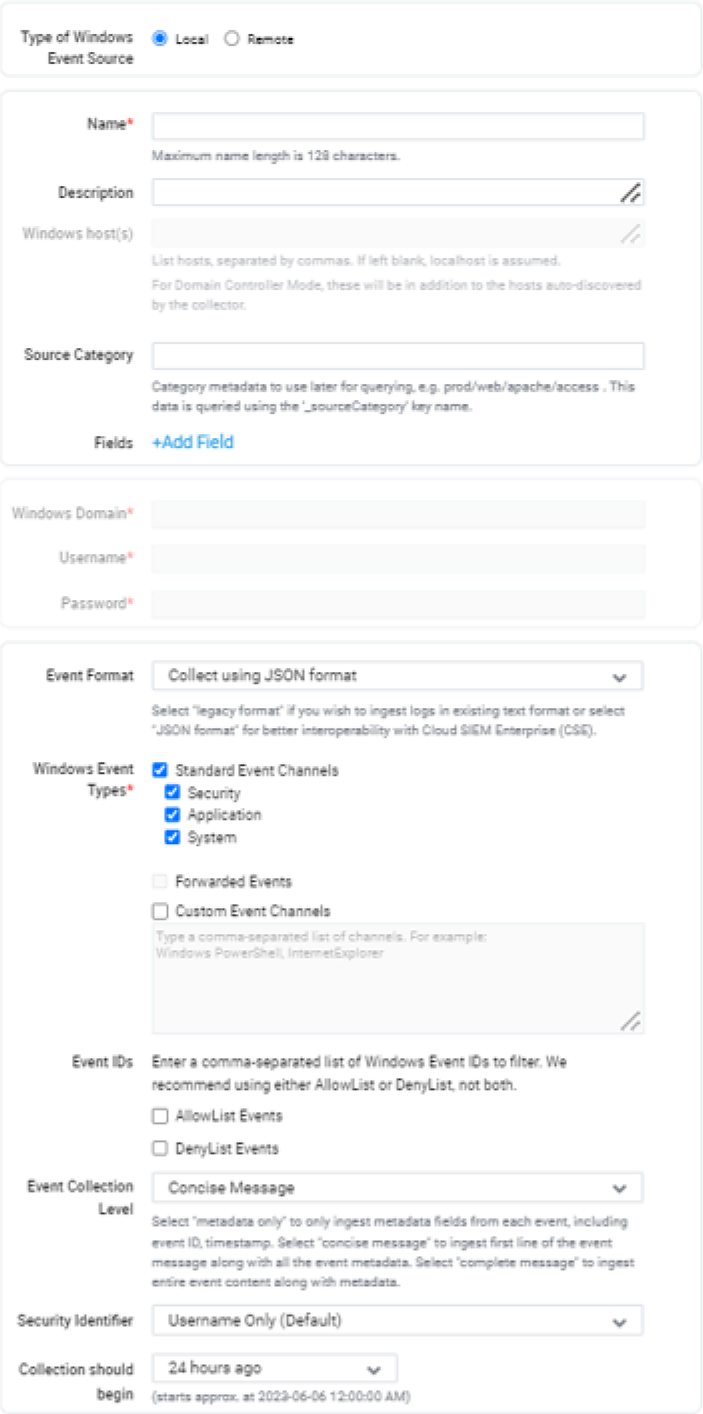
After installing a Collector, return to the Sumo Logic console and navigate to Manage Data > Collectors. Select the Collector > Add Source.



Next, create a Source on the Sumo Logic screen. Select Windows Source > **Windows Event Log to import event logs from Japanese environments without garbled characters.**



Then, fill in the required fields marked with a red asterisk. Enter an easy-to-understand name in Name. Enter Source Category separated by a slash (/) so that it is clear what data it is. We recommend using JSON format for Event Format (explained later). Use Windows Event Type to decide which channel logs to retrieve.



Type of Windows Event Source ● Local ○ Remote

Name*

Maximum name length is 128 characters.

Description

Windows host(s)

List hosts, separated by commas. If left blank, localhost is assumed.

For Domain Controller Mode, these will be in addition to the hosts auto-discovered by the collector.

Source Category

Category metadata to use later for querying, e.g. prod/web/apache/access . This data is queried using the '_sourceCategory' key name.

Fields +Add Field

Windows Domain*

Username*

Password*

Event Format: Collect using JSON format

Select "legacy format" if you wish to ingest logs in existing text format or select "JSON format" for better interoperability with Cloud SIEM Enterprise (CSE).

Windows Event Types*

☑ Standard Event Channels
- ☑ Security
- ☑ Application
- ☑ System

□ Forwarded Events

□ Custom Event Channels

Type a comma-separated list of channels. For example: Windows PowerShell, InternetExplorer

Event IDs: Enter a comma-separated list of Windows Event IDs to filter. We recommend using either AllowList or DenyList, not both.

□ AllowList Events

□ DenyList Events

Event Collection Level: Concise Message

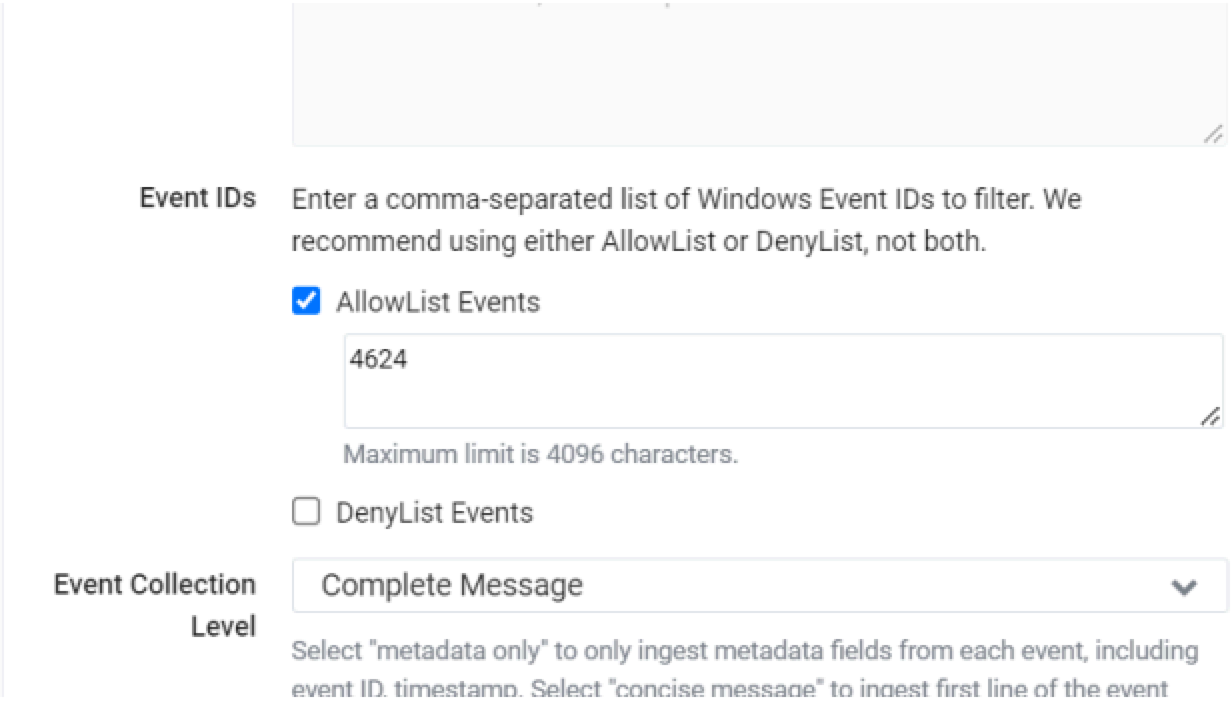
Select "metadata only" to only ingest metadata fields from each event, including event ID, timestamp. Select "concise message" to ingest first line of the event message along with all the event metadata. Select "complete message" to ingest entire event content along with metadata.

Security Identifier: Username Only (Default)

Collection should begin: 24 hours ago

(starts approx. at 2023-06-06 12:00:00 AM)

By the way, you can select the Event Log to import by Event IDs.



Event IDs Enter a comma-separated list of Windows Event IDs to filter. We recommend using either AllowList or DenyList, not both.

☑ AllowList Events

4624

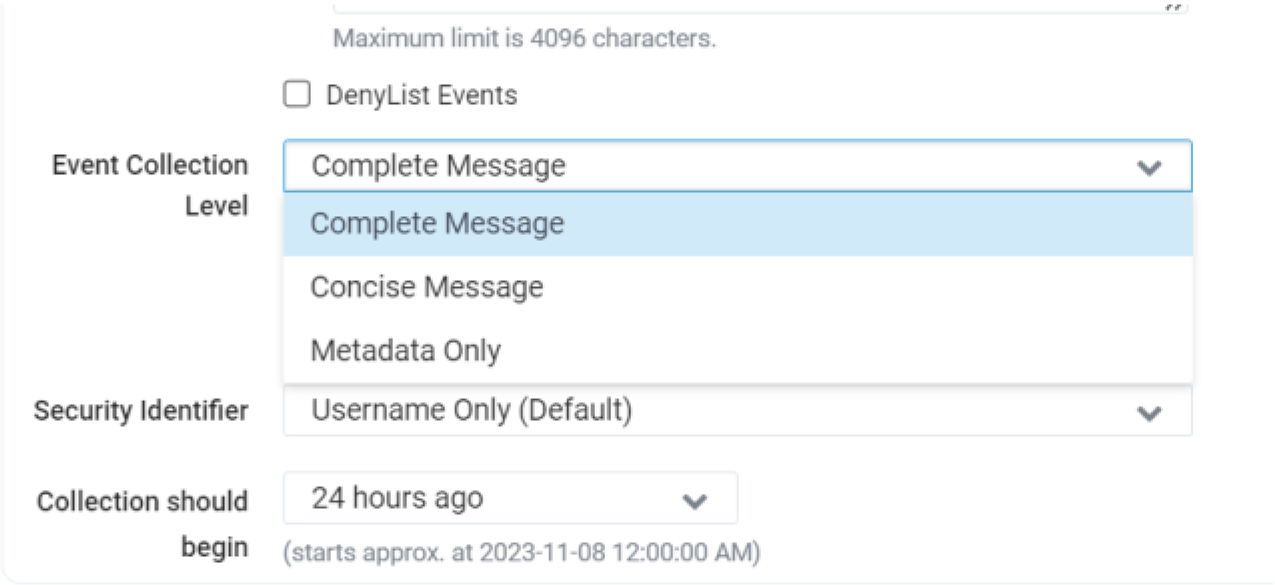
Maximum limit is 4096 characters.

☐ DenyList Events

Event Collection Level: Complete Message

Select "metadata only" to only ingest metadata fields from each event, including event ID, timestamp. Select "concise message" to ingest first line of the event

Also, if you set the Event Collection Level to Complete Message, all log information will be captured without being omitted.



Maximum limit is 4096 characters.

☐ DenyList Events

Event Collection Level: Complete Message

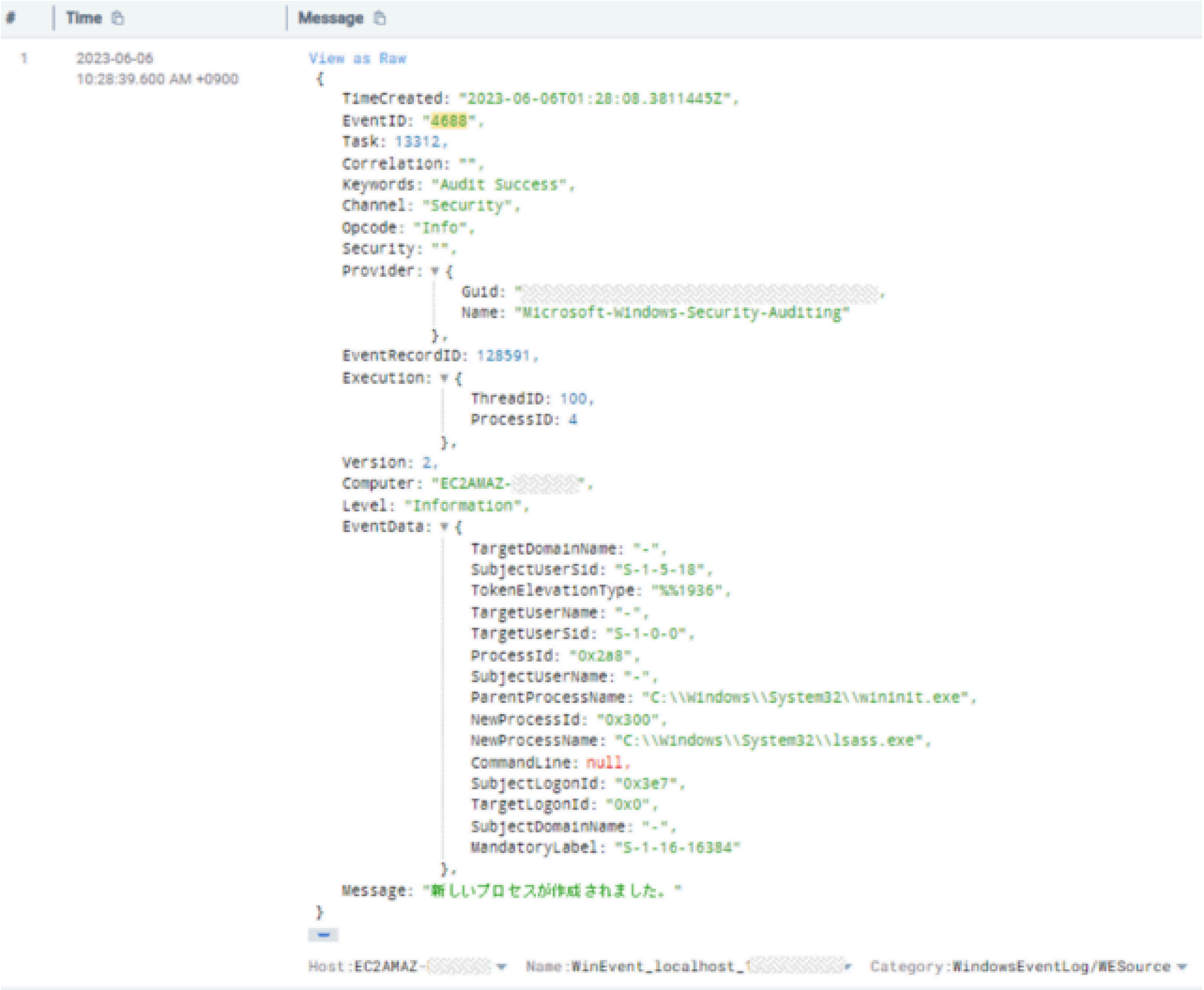
Security Identifier: Username Only (Default)

Collection should begin: 24 hours ago

(starts approx. at 2023-11-08 12:00:00 AM)

Once you've completed the configuration, click Save. At this point, the .evtx format logs have been converted to json and imported into Sumo Logic. The logs should be ingested after a while.

After waiting a little while, if you run a query, Japanese will also be output as shown below.



```
#   Time                       Message
1   2023-06-06                 View as Raw
    10:28:39.600 AM +0900      {
                                   TimeCreated: "2023-06-06T01:28:08.3811445Z",
                                   EventID: "4688",
                                   Task: 13312,
                                   Correlation: "",
                                   Keywords: "Audit Success",
                                   Channel: "Security",
                                   Opcode: "Info",
                                   Security: "",
                                   Provider: ▾ {
                                             Guid: "[redacted],
                                             Name: "Microsoft-Windows-Security-Auditing"
                                         },
                                   EventRecordID: 128591,
                                   Execution: ▾ {
                                              ThreadID: 100,
                                              ProcessID: 4
                                          },
                                   Version: 2,
                                   Computer: "EC2AMAZ-[redacted]",
                                   Level: "Information",
                                   EventData: ▾ {
                                              TargetDomainName: "-",
                                              SubjectUserSid: "S-1-5-18",
                                              TokenElevationType: "%%1936",
                                              TargetUserName: "-",
                                              TargetUserSid: "S-1-0-0",
                                              ProcessId: "0x2a8",
                                              SubjectUserName: "-",
                                              ParentProcessName: "C:\\Windows\\System32\\wininit.exe",
                                              NewProcessId: "0x300",
                                              NewProcessName: "C:\\Windows\\System32\\lsass.exe",
                                              CommandLine: null,
                                              SubjectLogonId: "0x3e7",
                                              TargetLogonId: "0x0",
                                              SubjectDomainName: "-",
                                              MandatoryLabel: "S-1-16-16384"
                                          },
                                   Message: "新しいプロセスが作成されました。"
                               }
                               −
                               Host:EC2AMAZ-[redacted] ▾   Name:WinEvent_localhost_[redacted] ▾   Category:WindowsEventLog/WESource ▾
```

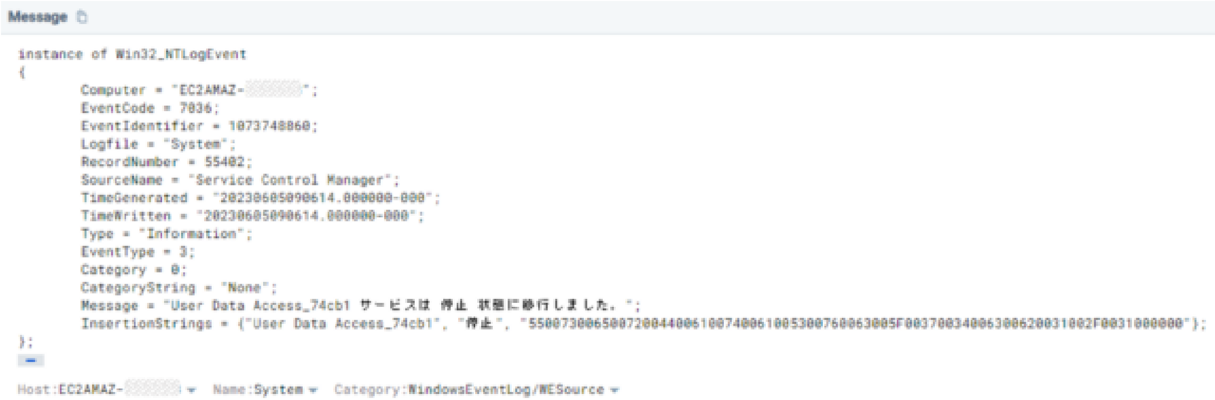
In this way, you can send log data to Sumo Logic without any special local configuration regarding log output.

That's all for now on the settings. Next, I will explain why we should use JSON for the Event Format, as I mentioned earlier.

Benefits of importing JSON format logs

- **Logs output to the Message tab are now easier to read.**
  If it is in json format, all nested data will be collapsed, like the query results we mentioned in the setup instructions.

  On the other hand, if the format is not JSON, nested data will be displayed without line breaks.

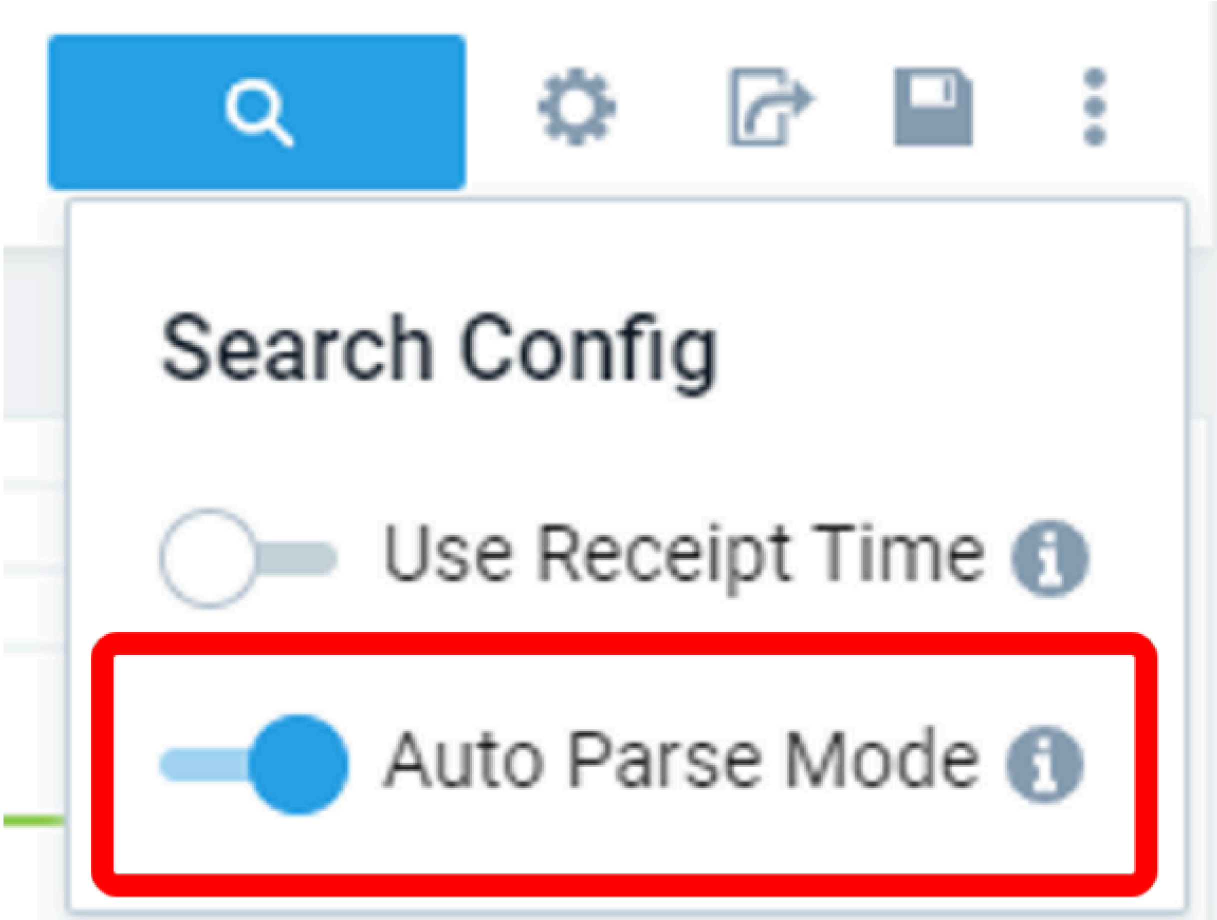


Message

```
instance of Win32_NTLogEvent
{
        Computer = "EC2AMAZ-[redacted]";
        EventCode = 7036;
        EventIdentifier = 1073748860;
        Logfile = "System";
        RecordNumber = 55402;
        SourceName = "Service Control Manager";
        TimeGenerated = "20230605090614.000000-000";
        TimeWritten = "20230605090614.000000-000";
        Type = "Information";
        EventType = 3;
        Category = 0;
        CategoryString = "None";
        Message = "User Data Access_74cb1 サービスは 停止 状態に移行しました。";
        InsertionStrings = {"User Data Access_74cb1", "停止", "55007300650072004400610074006100530076006300 5F003700340063006200310 02F0031000000"};
};
```

Host:EC2AMAZ-[redacted] Name:System Category:WindowsEventLog/WESource

- **Using Fields**
  If data is ingested in json format, Auto Parse Mode will parse the json format for you.

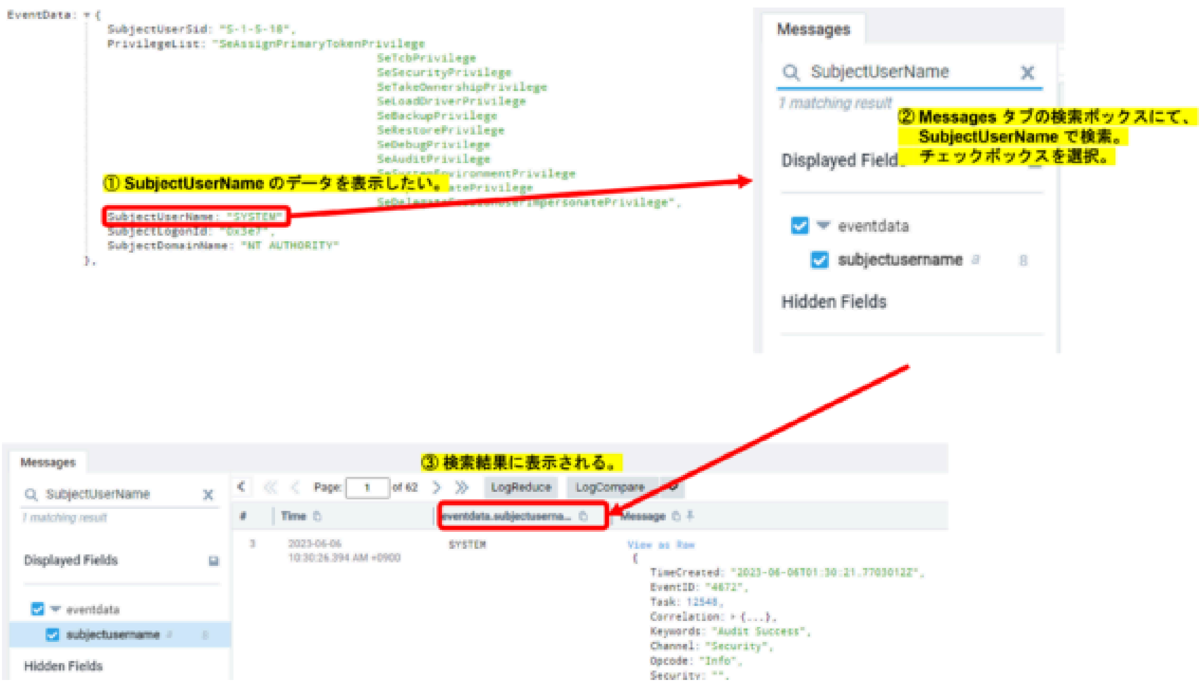


Search Config

○ Use Receipt Time

● Auto Parse Mode

This makes it easier to search using Field.



```
EventData: ▾ {
    SubjectUserSid: "S-1-5-18",
    PrivilegeList: "SeAssignPrimaryTokenPrivilege
                    SeTcbPrivilege
                    SeSecurityPrivilege
                    SeTakeOwnershipPrivilege
                    SeLoadDriverPrivilege
                    SeBackupPrivilege
                    SeRestorePrivilege
                    SeDebugPrivilege
                    SeAuditPrivilege
                    SeSystemEnvironmentPrivilege
                    SeImpersonatePrivilege
                    SeDelegateSessionUserImpersonatePrivilege",
    SubjectUserName: "SYSTEM",
    SubjectLogonId: "0x3e7",
    SubjectDomainName: "NT AUTHORITY"
},
```

① SubjectUserName のデータを表示したい。

② Messages タブの検索ボックスにて、
SubjectUserName で検索。
チェックボックスを選択。

③ 検索結果に表示される。

```
3    2023-06-06                       SYSTEM    View as Raw
     10:30:26.394 AM +0900                      {
                                                  TimeCreated: "2023-06-06T01:30:21.7703012Z",
                                                  EventID: "4672",
                                                  Task: 12548,
                                                  Correlation: ▸ {...},
                                                  Keywords: "Audit Success",
                                                  Channel: "Security",
                                                  Opcode: "Info",
                                                  Security: "",
```

On the other hand, if the log is not in JSON format, you will need to write a query using the Parse operator like the one below to parse it. Also, if the log contains line breaks (\r\n), you will need to use escape characters, which can make the query statement more complex.

```
_sourceCategory=Labs/Apache/Access
| parse "GET * " as url
```

Additionally, View Search Results for JSON Logs | Sumo Logic describes how to use JSON data around the console.

Parse JSON Formatted Logs | Sumo Logic explains how to parse JSON using JSON operators, including extracting values and arrays. We hope this is helpful.

There are advantages to importing data in JSON format like this, so we recommend it.

Event logs that are useful for security purposes

There are some important Windows Events that are disabled by default.

- **Event ID: 4688 (Process Creation)**
  This is output when a new process is created. If an attacker launches an attack such as user discovery, script execution, permission change, or credential reading, some kind of process will be launched on the PC. Therefore, it is a good idea to obtain the Process Creation log.

Additionally, because attackers often use PowerShell to execute malicious attacks, it would be a good idea to also collect logs of PowerShell modules and script blocks. Also, it is important to check that business PCs may have monitoring software such as Sysmon installed or enabled during kitting.

summary

While researching this article, I discovered that the default file format output by the Windows Event Log is the Windows-specific .evtx file, which appears to be a similar issue not only with Sumo Logic but also with other SIEM products. In Sumo Logic, I was able to easily obtain the data by creating a Windows Event Log Source. However, I found that the previous Local File Source resulted in garbled characters, so I recommend using the Windows Event Log Source. I hope this helps.