Firepower eNcore 運用ガイド

初版：2017 年 11 月 11 日
最終更新日：2021 年 7 月 6 日

目次

このマニュアルについて

作成者 (Author)	Seyed Khadem（skademd)
変更権限	シスコ アドバンスドサービス、セキュリティ & コラボレーション IDT、インプリメンテーション アメリカ
コンテンツ ID	
プロジェクト ID	

マニュアルの変更履歴

改定	日付	氏名またはユーザ ID	Comments
1.0	00/00/2021		初回リリース

表記法

このマニュアルでは、次の表記法を使用しています。

表記法	説明
太字 フォント	コマンド、キーワード、およびユーザが入力するテキストは、**太字**で示しています。
イタリック体 フォント	ドキュメント名、新規用語または強調する用語、値を指定するための引数は、*イタリック体*で記載されます。
[]	角かっこの中の要素は、省略可能です。
{x | y | z }	いずれか 1 つを選択しなければならない必須キーワードは波カッコで囲み、縦棒で区切って示しています。
[x | y | z]	いずれか 1 つを選択できる省略可能なキーワードは角カッコで囲み、縦棒で区切って示しています。

表記法	説明
文字列	引用符を付けない一組の文字。string の前後には引用符を使用しません。引用符を使用すると、その引用符も含めて string とみなされます。
`courier フォント`	システムが表示する端末セッションおよび情報は、`courier` フォントで示しています。
<>	パスワードのように出力されない文字は、山カッコで囲んで示しています。
[]	システム プロンプトに対するデフォルトの応答は、角カッコで囲んで示しています。
!、#	コードの先頭に感嘆符（!）またはポンド記号（#）がある場合には、コメント行であることを示します。

（注） 「注釈」です。役立つ情報やこのマニュアルに記載されていない参照資料を紹介しています。

注意： 「要注意」の意味です。機器の損傷またはデータ損失を予防するための注意事項が記述されています。

1 はじめに

2 ドキュメントの目的

このドキュメントでは、CLI、Splunk、および Sentinel 向けの eStreamer eNcore クライアントの背景情報や使用方法について説明し、ユーザによるインストールと実行を支援します。

2.1 バックグラウンド

Cisco Event Streamer（eStreamer）により、ユーザは Firepower Management Center または管理対象デバイス（eStreamer サーバ）から外部クライアント アプリケーションにシステム侵入、検出、接続などに関するデータをストリーミングできます。eStreamer は、高いパフォーマンスを促進する簡潔でコンパクトなバイナリ エンコード メッセージでクライアント要求に応答します。

これまで、eStreamer SDK は、個別の Perl アプリケーション（Splunk 向け Cisco eStreamer アプリケーションや CEF エージェントなど）を作成するために追加コードでラップされてきました。

2.2 アプリケーション サマリ

eStreamer eNcore は、Firepower Management Center バージョン 6.0 以降と互換性のあるマルチプラットフォーム、マルチプロセスの Python アプリケーションです。

eNcore は、汎用クライアントであり、eStreamer からあらゆるイベントを取得し、バイナリコンテンツを解析します。また、他のセキュリティ情報およびイベント管理（SIEM）ツールをサポートするために、さまざまな形式でイベントを出力できます。eNcore は、Python でスケーラブルかつ高速なマルチプロセス アーキテクチャを使用してゼロから構築されました。Firepower Management Center のバージョン 6.0 をサポートしています。CentOS 7 で構築およびテストされていますが、前提条件をサポートするすべての Linux ディストリビューションで通常は動作します。このソフトウェアは Windows 上で実行できますが、サポート対象外です。

eStreamer eNcore に関連する次の 3 つのパッケージがあります。

- eNcore CLI
- eNcore Add-on for Splunk
- eNcore Dashboard for Splunk

このガイドでは、3 つのパッケージすべてについて説明します。

3 Cisco eNcore CLI

eNcore CLI は、eStreamer eNcore のコマンドライン インターフェイスです。スタンドアロン アプリケーションとして動作し、Firepower Management Center eStreamer サーバからイベントを取得して、次のいずれかの形式でそれらのイベントを出力します。

- キーと値のペア：以前の Splunk コレクタとの互換性の維持を目的とします
- JSON
- CEF for Arcsight：以前の cef-agent との下位互換性の維持を目的とします

出力は、ファイル、TCP または UDP ネットワークポート、stdout にストリーミングできます。

3.1 eNcore CLI の前提条件

eNcore CLI は、前提条件をサポートするすべての Linux ディストリビューションで動作します。Windows 上でも実行できますが、実稼働には対応していません。

eNcore のインストール先となるプラットフォームには、主に次の 2 つの前提条件があります。

- Python 2.7 または Python 3.6 以降
- pyOpenSSL

eNcore の CLI バージョンは、Python 2.7 または Python 3.6 以降で実行できます。また、Firepower Management Center の PKCS12 ファイルを分割する手段も必要です。デフォルトの方法では、pyOpenSSL をインストールし、eNcore によって自動的に処理します。

（注） encore.sh スクリプトにより、これらすべての項目が確認されるため、すぐに作業を始めることもできますが、インストールの前にこれらの項目を把握しておくことをお勧めします。

Python 2.7 が存在するかどうかを確認するには、次のコマンドを使用します。

which python

Python 2.7 が存在する場所をテストするには、次のコマンドを使用します。

whereis python

Python がインストールされている場合、which python コマンドはインストールディレクトリのパスを示します。たとえば、コマンドの出力が /usr/bin/python の場合は、Python がインストールされています。インストールされている Python が v2.7 かどうかを確認するには、インストールディレクトリの親（上記の例では /usr/bin ディレクトリ）の内容をリストします。たとえば、次のようなエントリがリストに表示されているとします。

lrwxrwxrwx 1 root root 9 Dec 9 2015 python -> python2.7*

このエントリは、**python** が python2.7 ディレクトリにリンクされており、ここに Python v2.7 がインストールされていることを示しています。もう 1 つのコマンド **whereis python** を使用して、python2.7 ディレクトリが存在するかどうかを表示することもできます。

（注） Splunk を実行しているデバイスに CLI バージョンをインストールする場合は、Splunk に独自のバージョンの Python があることに注意してください。Splunk Python は、通常のディストリビューションとは異なる方法でコンパイルされています。具体的には、PyUnicodeUCS2 で構築されています。**encore.sh** スクリプトはこれを検出して警告を表示します。この問題が発

生した場合は、新しいユーザを作成し、そのユーザで eStreamer-eNcore を実行する必要があります。代わりに Splunk アドオンを実行することを検討する必要があります。

pyOpenSSL を確認するには、次のコマンドを使用します。

pip list | grep -i pyOpenSSL

代わりに python3 バージョンを使用すると、PyUnicodeUCS4 に関する複雑さに対処する必要がなくなります。python3 ブランチにアクセスするには、次のコマンドを実行します。

git checkout python3

3.1.1 Python 2.7 または Python 3.6 以降のインストール

Python を CentOS にインストールするには、次のコマンドを使用します。

sudo yum install python

3.1.2 pyOpenSSL のインストール

pyOpenSSL は Python 2.7 のインストールの一部としてインストールされている場合があります。インストールされているかどうかを確認するには、次のコマンドを使用します。

pip list | grep -i pyOpenSSL

pip がインストールされていない場合は、次のコマンドで CentOS にインストールできます。

sudo python get-pip.py

次のコマンドで pyOpenSSL をインストールします。

sudo yum install python-pip python-devel openssl-devel gcc
sudo pip install pyOpenSSL

python3 ブランチを使用している場合は、次のコマンドを実行します。
sudo pip3 install pyOpenSSL

3.1.3 RHEL の EPEL リポジトリの依存関係

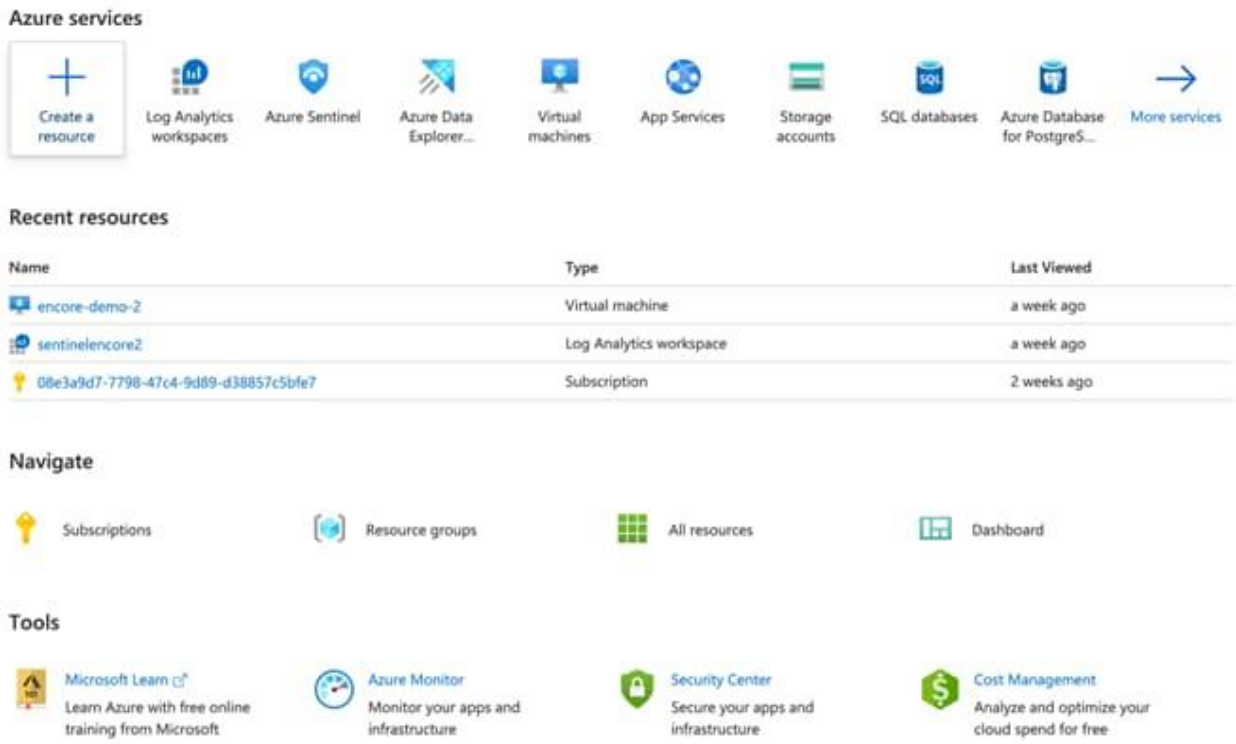
これらのパッケージのインストールに関して問題がある場合は、EPEL リポジトリの有効化が必要になることがあります。EPEL リポジトリをインストールして有効にする手順は、インターネットで公開されています。

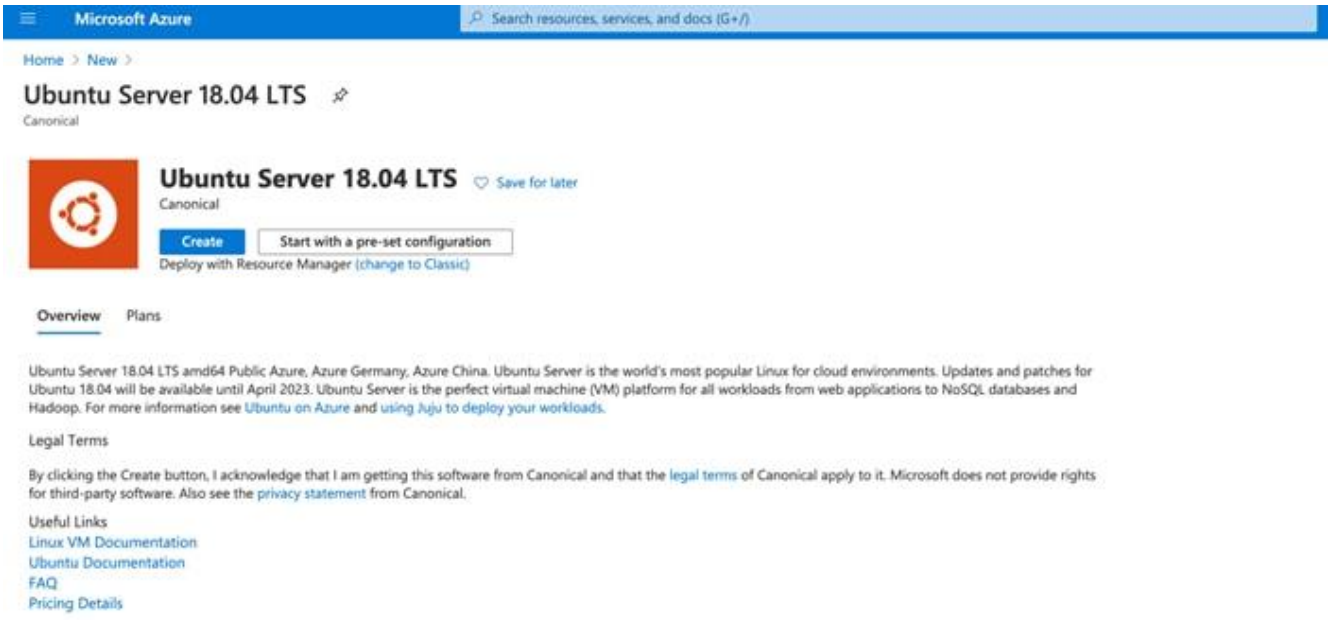
EPEL に関する Red Hat のガイド：

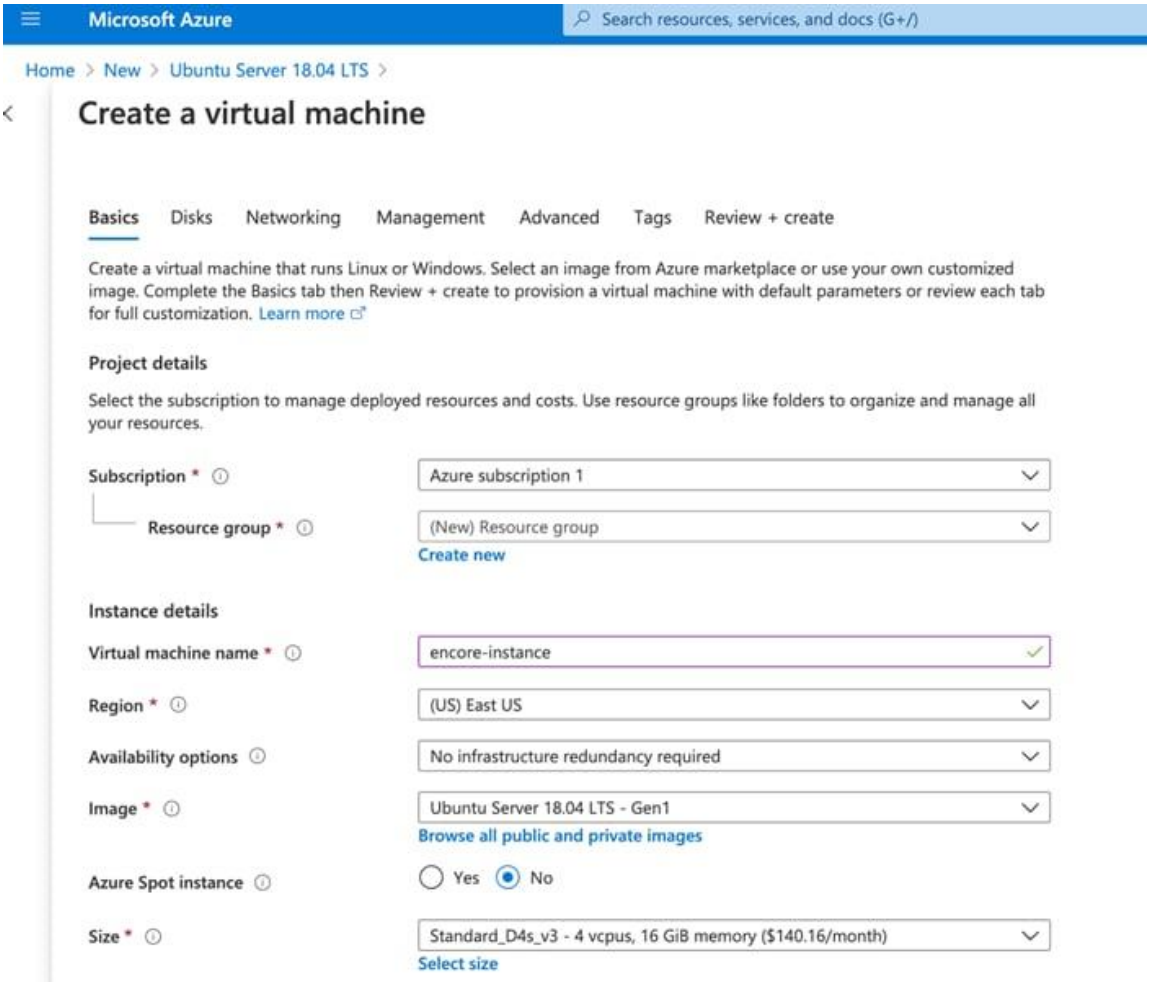
https://access.redhat.com/solutions/3358
https://www.redhat.com/en/blog/whats-epel-and-how-do-i-use-it

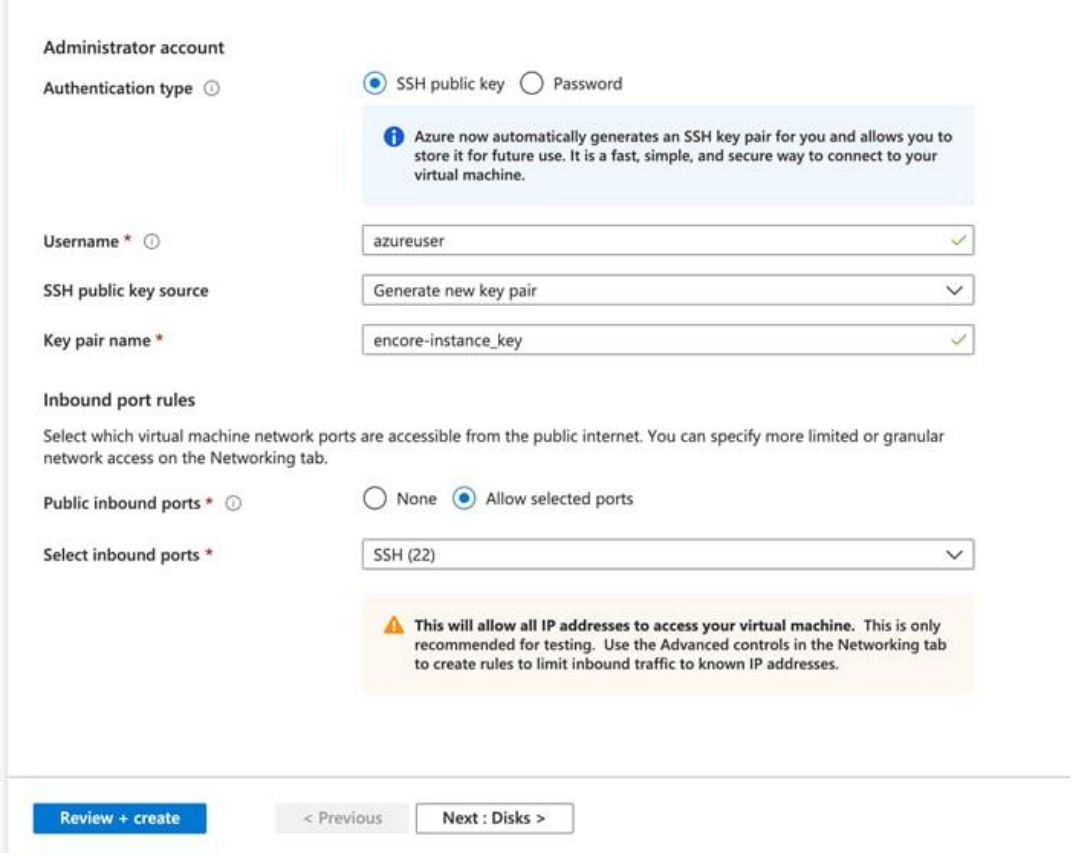
3.1.4 Azure での eNcore CLI の実

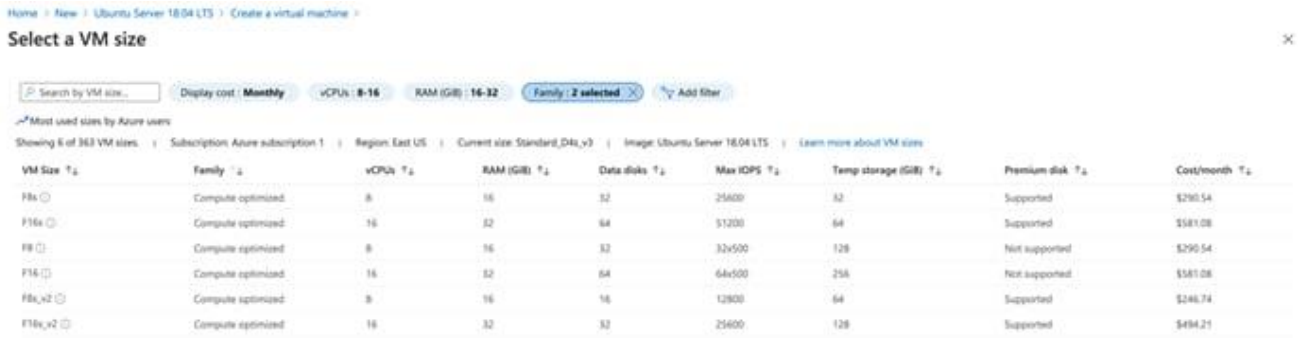
1 Ubuntu 18.04 LTS など、新しい Linux リソースを作成します。




Microsoft Azure
Search resources, services, and docs (G+/)
Home > New >
Ubuntu Server 18.04 LTS
Canonical
Ubuntu Server 18.04 LTS
Save for later
Canonical
Create
Start with a pre-set configuration
Deploy with Resource Manager (change to Classic)
Overview
Plans
Ubuntu Server 18.04 LTS amd64 Public Azure, Azure Germany, Azure China. Ubuntu Server is the world's most popular Linux for cloud environments. Updates and patches for Ubuntu 18.04 will be available until April 2023. Ubuntu Server is the perfect virtual machine (VM) platform for all workloads from web applications to NoSQL databases and Hadoop. For more information see Ubuntu on Azure and using Juju to deploy your workloads.
Legal Terms
By clicking the Create button, I acknowledge that I am getting this software from Canonical and that the legal terms of Canonical apply to it. Microsoft does not provide rights for third-party software. Also see the privacy statement from Canonical.
Useful Links
Linux VM Documentation
Ubuntu Documentation
FAQ
Pricing Details


Microsoft Azure
Search resources, services, and docs (G+/)
Home > New > Ubuntu Server 18.04 LTS >
Create a virtual machine
Basics
Disks
Networking
Management
Advanced
Tags
Review + create
Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. Learn more
Project details
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.
Subscription *
Azure subscription 1
Resource group *
(New) Resource group
Create new
Instance details
Virtual machine name *
encore-instance
Region *
(US) East US
Availability options
No infrastructure redundancy required
Image *
Ubuntu Server 18.04 LTS - Gen1
Browse all public and private images
Azure Spot instance
Yes
No
Size *
Standard_D4s_v3 - 4 vcpus, 16 GiB memory ($140.16/month)
Select size

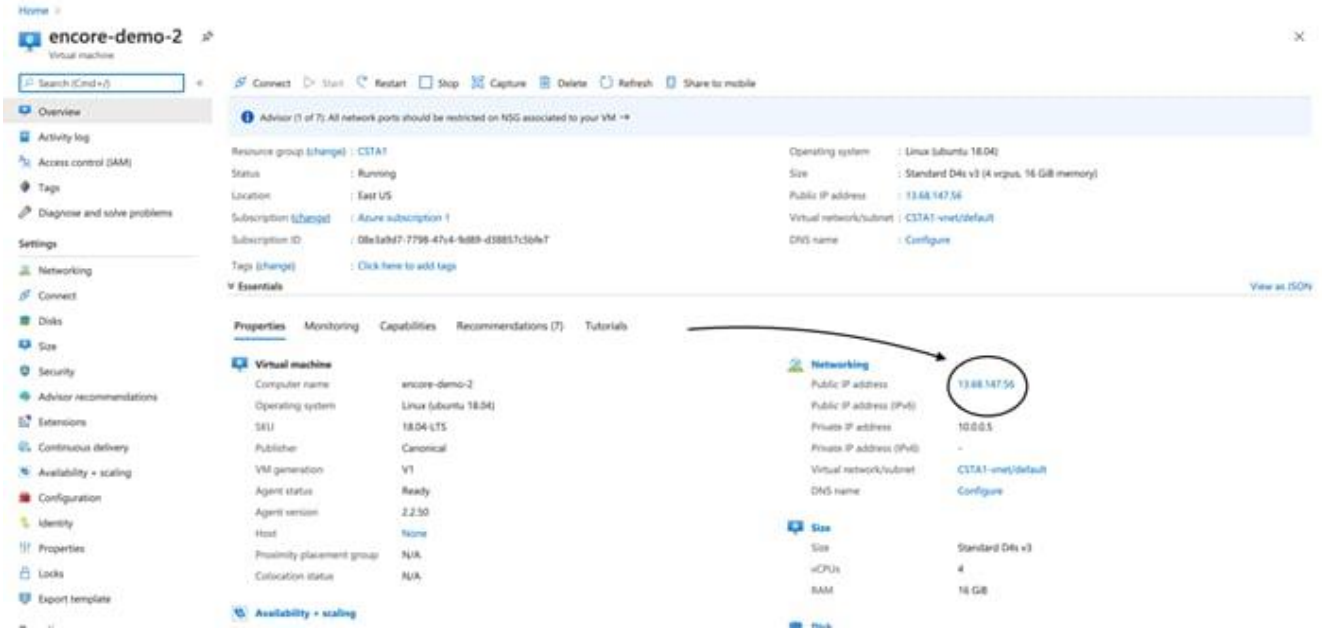




2 CPU を仮想インスタンスに割り当てます。eNcore CLI では最大 12 スレッドをサポートできます。8 ～ 16 コアの最適化されたコンピュータを使用することをお勧めします。eNcore CLI では、16 CPU F16s_v2 オプションを使用して、最大 7,000 イベント/秒をサポートできます。

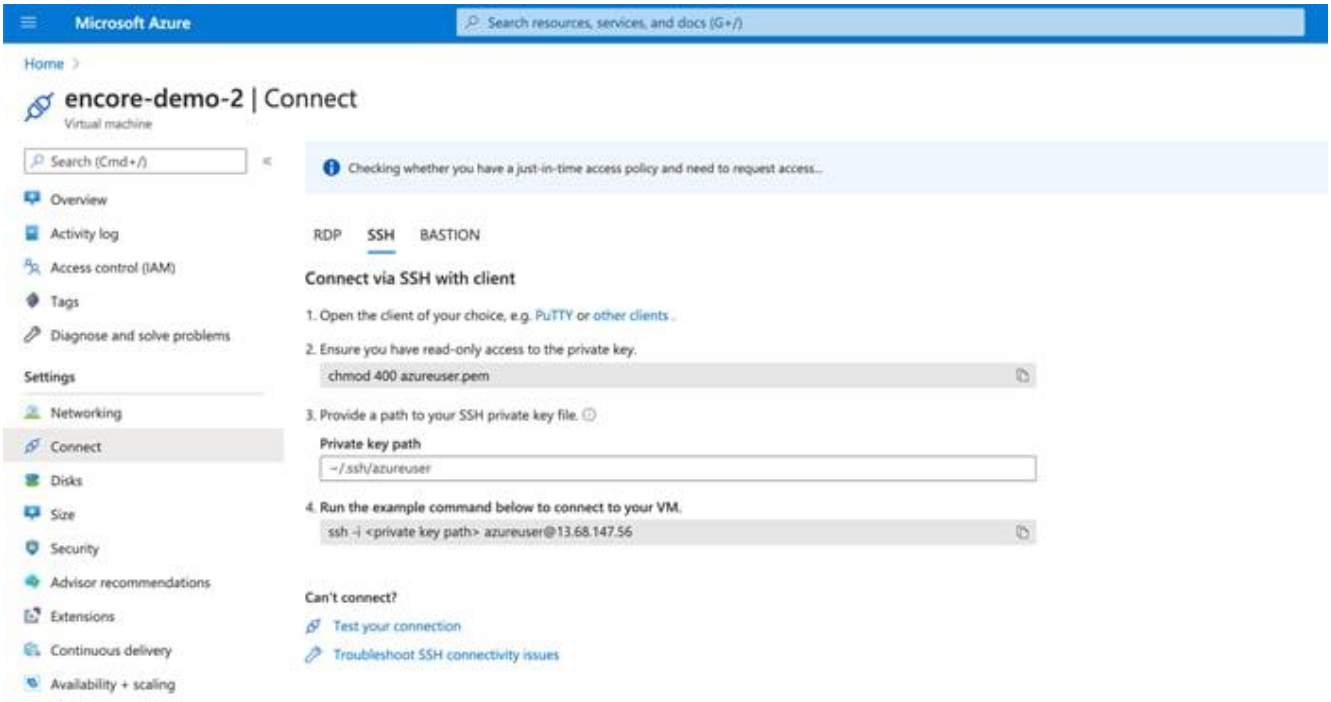
3 組織で想定されるボリュームに応じた規模にします。低ボリューム（500 イベント/秒未満）の運用では、推奨される最小 CPU 数は 4 です。

4 インスタンスに名前を付け、PEM 証明書をダウンロードします。

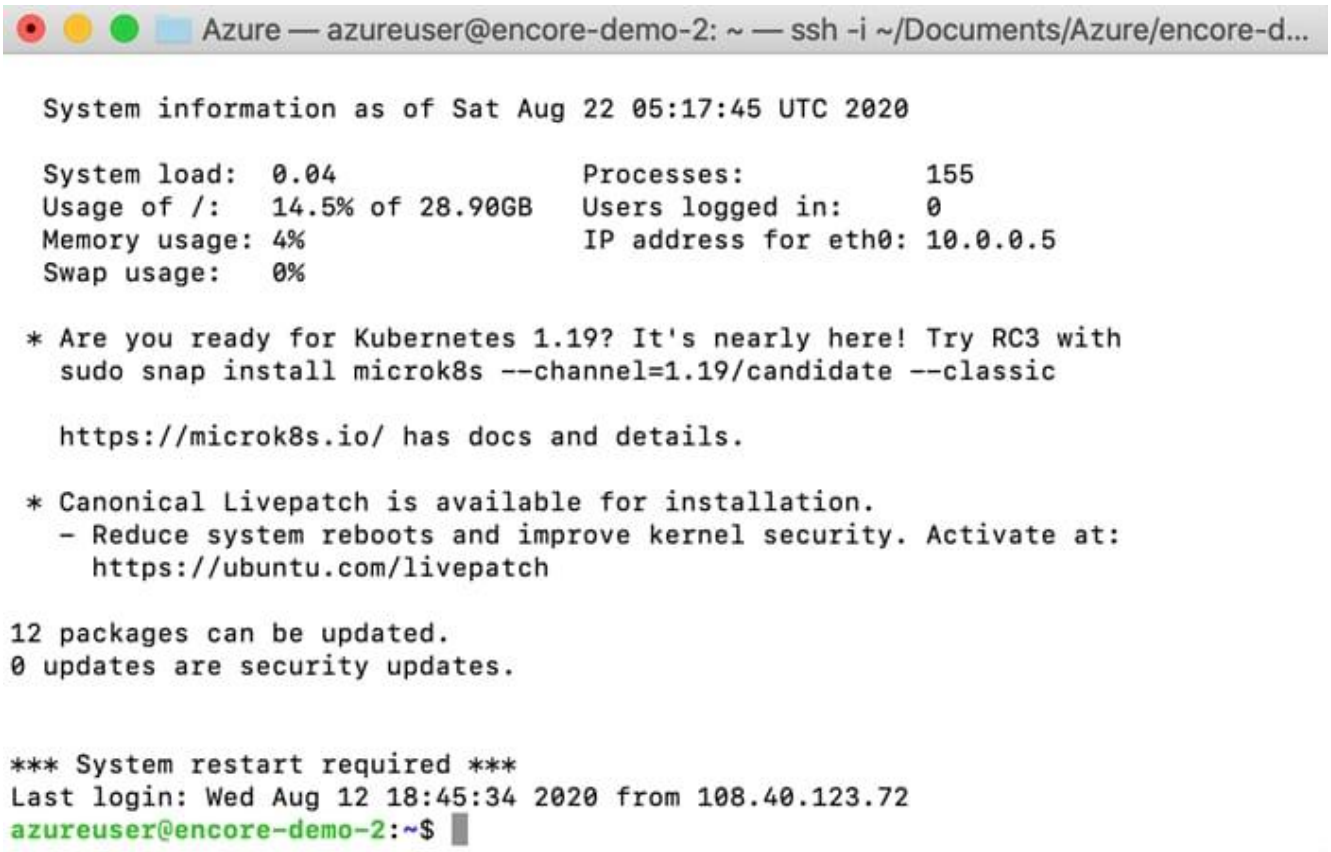


インスタンスに割り当てられたパブリック IP を控えておきます。この IP を使用して、Firepower Management Center eStreamer で証明書を作成します。

5 .pem ファイルを使用して、インスタンスのコマンドラインバージョンに接続します。これでインストールを開始できます。Azure には、クイックコマンドライン接続を有効にするショートカットもあります。



ssh -I <private key path> azureuser@<public ip>



3.1.5 Windows での eNcore CLI の実

Warning: Windows は、現時点では、実稼働環境での実行についてはサポート対象外です。ただし、CLI バージョンのインストールを試す場合は、次のコマンドを実行する必要があります。

pip install pyOpenSSL、pip install win-inet-pton

3.2 eStreamer eNcore CLI のインストール

3.2.1 ソースからの eNcore クライアントの構

次のコマンドを使用して、最新バージョンをターゲットクライアントにコピーします。

git clone https://github.com/CiscoSecurity/fp-05-firepower-cef-connector-arcsight

以前のバージョンの場合：https://github.com/CiscoSecurity/fp-05-microsoft-sentinel-connector/releases

3.2.2 PKCS12 ファイルの儀

eStreamer サーバは、クライアント接続を認証および許可できる必要があります。そのためには、eStreamer クライアントを識別する PKCS12 ファイルが eStreamer サーバ上に必要であり、このファイルを eNcore サーバにコピーする必要があります。

Firepower Management Center で PKCS12 ファイルを作成してダウンロードする方法については、「付録」を参照してください。

3.2.3 PKCS12 ファイルのインストール

次のコマンドを使用して、PKCS12 ファイルを eNcore CLI のインストール環境に安全にコピーします。

scp -i /path/to/pem/encore-demo-2_key.pem /local/path/<public ip>.pkcs12 azureuser@<Public Ip>:/tmp/

証明書を /tmp から Git プロジェクトのランタイムパスにコピーします。
cp /tmp/client.pkcs12 ~/fp-05-microsoft-sentinel-connector

3.2.4 Test

1 次のコマンドを使用して、作業ディレクトリを eStreamer-eNcore に変更します。

cd ~/fp-05-microsoft-sentinel-connector

2 **encore.sh** シェルスクリプトを実行します。追加の構成が表示されます。

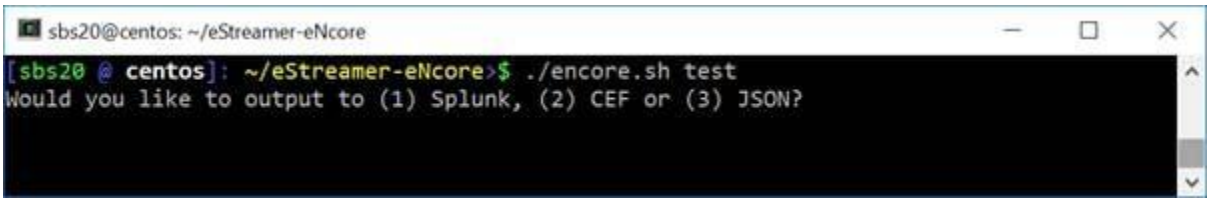
./encore.sh test

スクリプトでは、次の前提条件がインストールされていることが確認されます。

- Python 2.7、Python 3.6 以降には Git の「python3」ブランチが必要
- Python の正しいビルド
- pyOpenSSL
- client.pkcs12 ファイル
- 有効なホスト

3 Splunk、CEF、JSON のいずれのデータを出力するかを選択します。このガイドでは、CEF アウトプッタを使用しますが、将来のバージョンでは、使用されている Sentinel コネクタに応じて JSON またはその他のカスタム形式を使用する可能性があります。

図 1. 出力の選択



不足している項目がある場合は、説明が表示されます。次の図に説明の例を示します。

図 2：pkcs12 ファイルがない場合

4 Firepower Management Center の IP/FQDN と PKCS12 ファイルのパスワードを入力します。

図 3：パスワードの入力

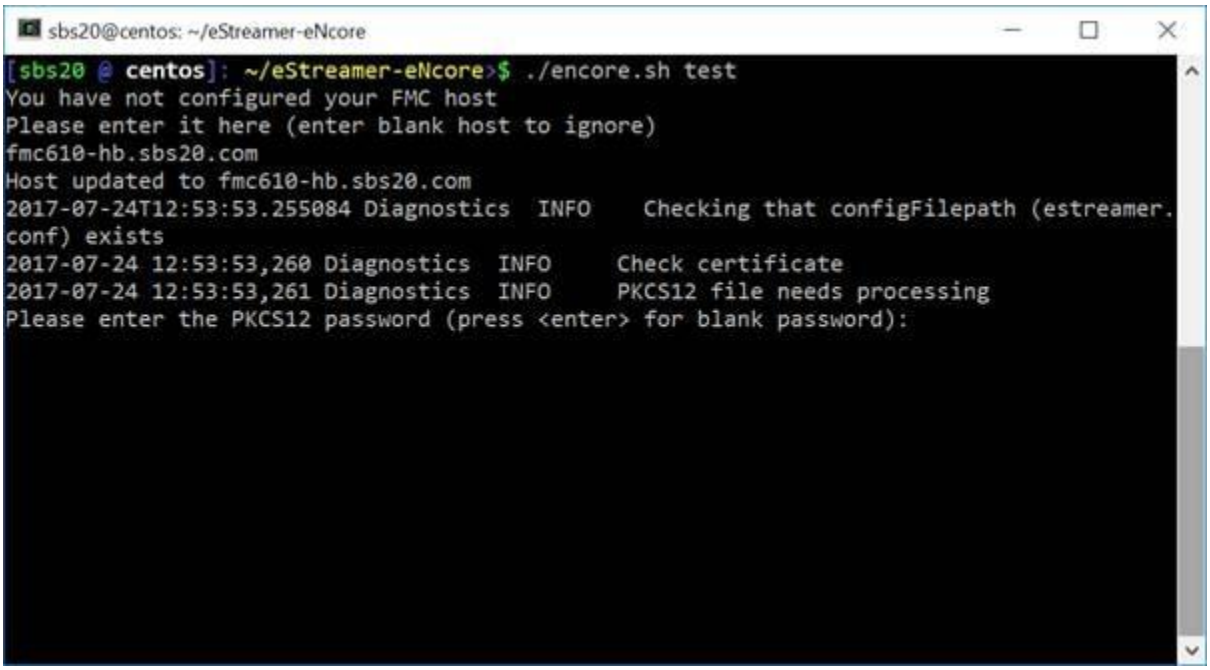
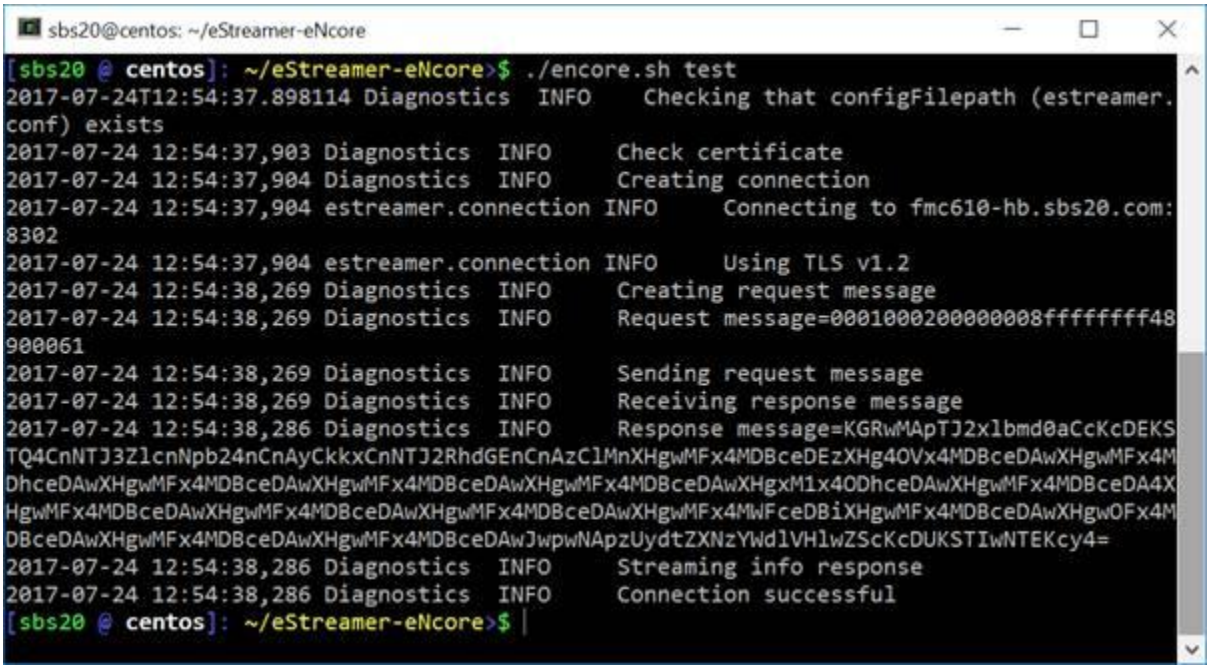


図 4：テストが成功した場合

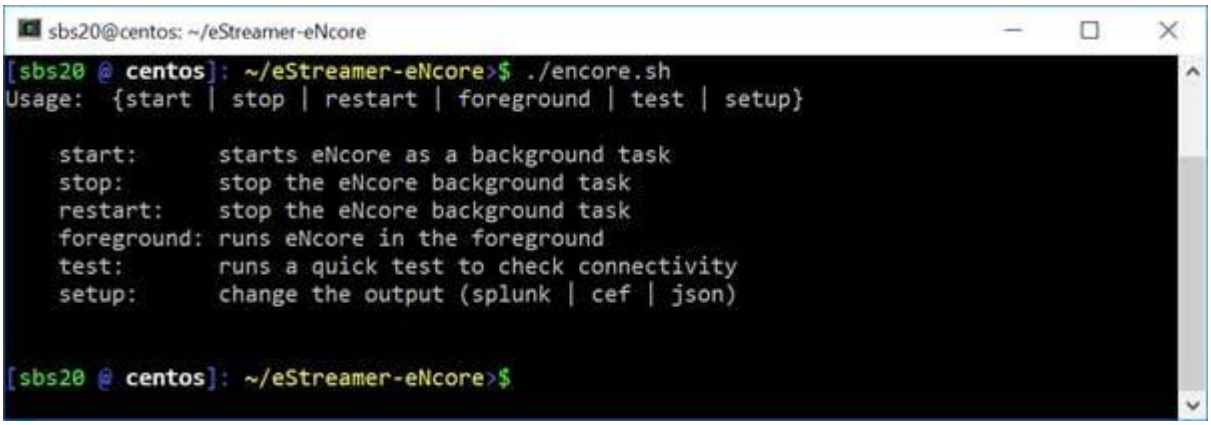


テストが成功した場合、eNcore CLI のインストールは完了です。

3.2.5 eNcore CLI の実

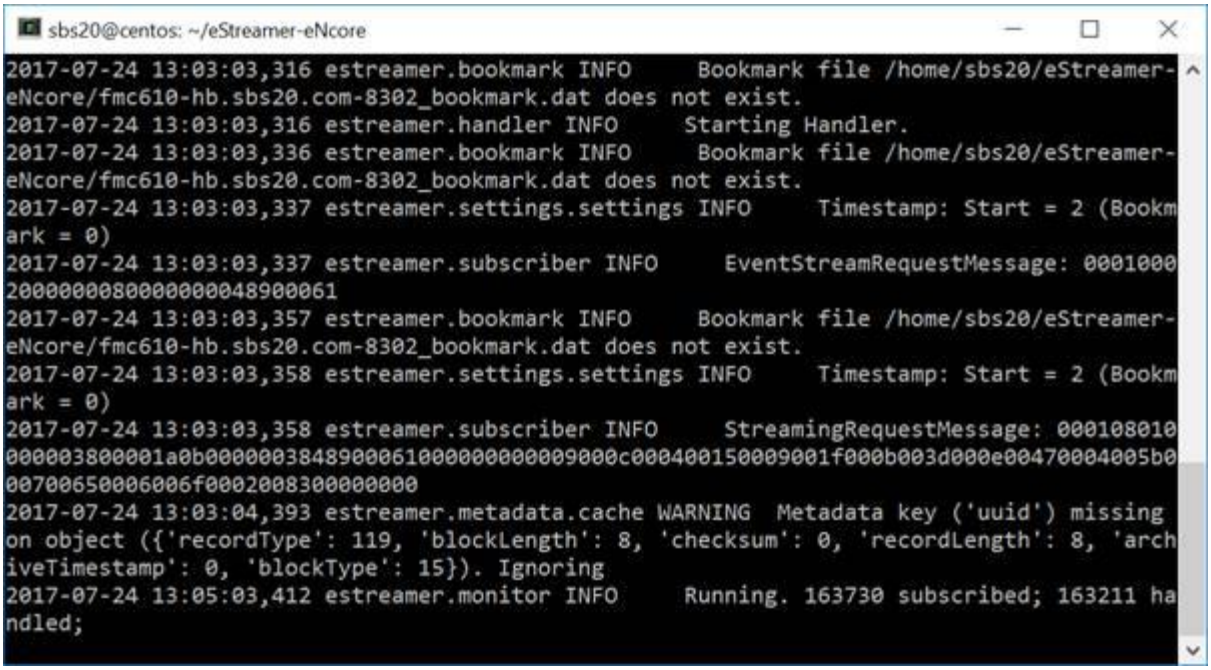
パラメータを指定せずに **encore.sh** を実行すると、簡単な説明が表示されます。

図 5：ヘルプ画面



初めて実行する際は、フォアグラウンドで実行すると、何が起きているかを確認できます。2 分ごとに、処理されたレコード数の情報が画面に表示されます。更新頻度を変更する場合は、**monitor.period** 構成を参照してください。

図 6：フォアグラウンドで実行して状況を監視



（注） フォアグラウンドプロセスを停止するには、Ctrl+C キーを押します。

3.3 eStreamer eNcore CLI の構

セクション 2.2 で説明されている eNcore CLI のインストールプロセスでは、Firepower Management Center eStreamer サーバへの接続を確立するために、Firepower Management Center の IP アドレスなどの基本

的な項目を設定する必要があります。このセクションでは、ソリューション要件を満たすためのアプリケーションの一般的な構成について説明します。

構成は、eStreamer-eNcore ディレクトリの **estreamer.conf** ファイルに保存されます。最初は、必要に応じて変更できるデフォルト設定が含まれています。ファイルは JSON 形式で、構成情報を示すキーが含まれています。ここでは、変更が必要になることの多いキーとセクションについて説明します。

デフォルトの構成ファイルは、すぐに実行できるように設定されています。カスタマイズできる各設定の簡単な説明を以下に示します。

3.3.1 サブスクリプションサーバ

これは、Firepower Management Center のホストおよび関連情報です。TLS に関する問題が発生した際にダウングレードする場合は、**tlsVersion** を 1.0 に変更できます。

（注） TLS バージョンのダウングレードは、デバッグやソフトウェアの動作確認に役立ちますが、長期的な戦略としては推奨されません。代わりに、根本原因を修正することをお勧めします。

subscription キーには、次の 2 つの主なサブセクションがあります。

- records セクションでは、Firepower Management Center eStreamer サーバへの接続時に eNcore が要求するイベントのタイプを選択できます。
- servers セクションには、Firepower Management Center のホスト IP と関連情報が含まれます。

このキーとその値の例を次に示します。

図 8：サブスクリプションサーバ画面

```
"subscription": {
    "records": {
      "@comment": [
        "Just because we subscribe doesn't mean the server is sending. Nor does it mean",
        "we are writing the records either. See handler.records[]"
      ],
      "archiveTimestamps": true,
      "eventExtraData": true,
      "extended": true,
      "impactEventAlerts": true,
      "intrusion": true,
      "metadata": true,
      "packetData": true
    },
    "servers": [
      {
        "host": "1.2.3.4",
        "port": 8302,
        "pkcs12Filepath": "client.pkcs12",
```

```
        "@comment": "Valid values are 1.0 and 1.2",
        "tlsVersion": 1.2
      }
  ], …
```

3.3.2 アウトプッタ

outputters セクションでは、eNcore がイベントを出力に書き込む方法を指定します。eNcore CLI では、次のいずれかの形式で出力できます。

- スプランク
- JSON
- CEF for Arcsight

出力は、ネットワーク接続を介して SIEM または別のコレクタに送信するか、ファイルに書き込むことができます。

次に例を示します。

- UDP 経由で出力を ArcSight コネクタに送信するように設定された ArcSight CEF アウトプッタ。
- 同じイベントをローカルファイルに書き込む ArcSight CEF アウトプッタ。URI の {0} の表記は、UNIX タイムスタンプをファイル名に挿入することを示します。

```
"outputters": [
  {
    "name": "CEF",
    "adapter": "cef",
    "enabled": true,
    "stream": {
      "uri": "udp://10.0.1.2:514",
    }
  },
  {
    "name": "CEFfile",
    "adapter": "cef",
    "enabled": true,
    "stream": {
      "uri": "relfile:///data/data.{0}.cef",
      "options": {
        "rotate": false,
        "maxLogs": 9999
      }
    }
  }
]
```

3.3.3 レコード数(Records)

records セクションでは、eNcore が処理するレコードを指定します。次の 2 つの方法で、どのイベントを処理するか（または処理から除外するか）を指定できます。

1. 接続（connections）など、目的とするイベントクラスの値を true に設定することで、そのクラスを処理するように指定できます。この例の場合、キーと値のペアは "connections": true になります。逆に、イベントクラスの値を false に設定して、そのクラスを処理しないように指定することもできます。

2. レコードタイプを include キーまたは exclude キーの値として記述することで、レコードタイプごとにイベントクラスの処理の例外を指定できます。JSON 配列では、複数の値をカンマで区切る必要があります。たとえば、レコードタイプ 98 および 170 を除外するには、exclude キーと値のペアは次のようになります。

   "exclude": [98, 170],

records キーと値のペアの例を次に示します。

（注） 処理するレコードのクラスは、最初に Firepower Management Center eStreamer 構成で選択する必要があることに注意してください。また、eNcore 構成の subscription セクションの records 部分でサブスクリプション用に設定する必要もあります。

```
"records": {
        "connections": true,
        "core": true,
        "excl@comment": [
          "These records will be excluded regardless of above (overrides 'include')",
          "e.g. to exclude flow and IPS events use [ 71, 400 ]"
        ],
        "exclude": [],
        "inc@comment": "These records will be included regardless of above",
        "include": [],
        "intrusion": true,
        "metadata": false,
        "packets": true,
        "rna": true,
        "rua": true
      }
```

3.3.4 イネーブル

eNcore で Firepower Management Center からイベントを取得し、ストリーミング操作を開始するには、enabled キーの値を true に設定する必要があります。このキーの例を次に示します。

"enabled": true,

3.3.5 実行

説明に従ってすべての項目を適切に設定したら、eNcore CLI を使用してイベントをストリーミングおよび書き込みできます。

さまざまなシェルスクリプトオプションを使用できます。

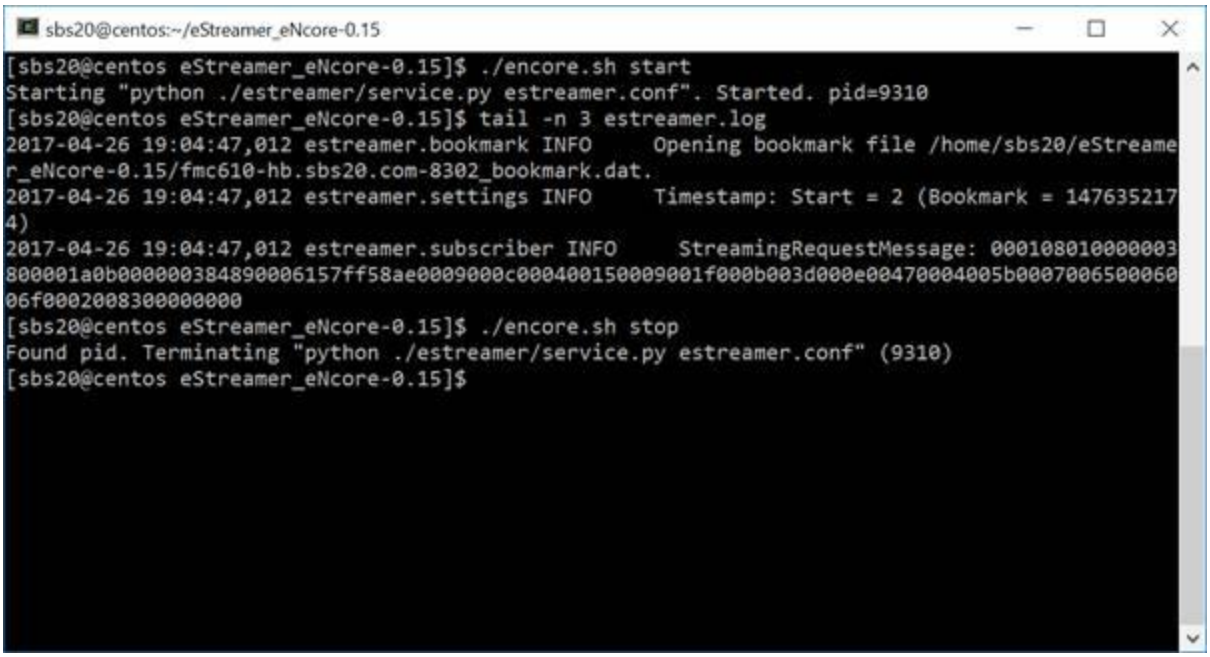
インストール時および初期設定時に（またはデバッグ目的で）、次のコマンドを実行すると便利です。

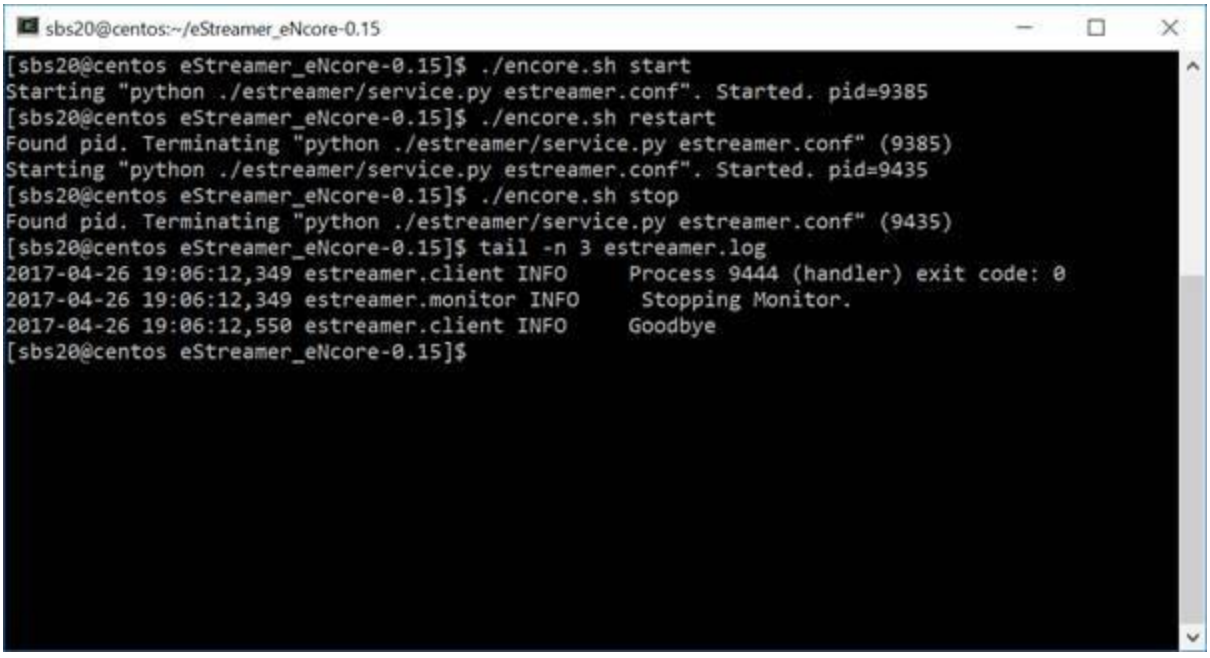
./encore.sh test
および
./encore.sh foreground

他のすべてのケースでは、eNcore はバックグラウンドで実行されることが想定されます。これには次のコマンドが該当します。

./encore.sh start
./encore.sh stop
./encore.sh restart

図 12：start、tail log、stop





3.3.6 ロギング

デフォルトでは、eNcore は estreamer.log アプリケーションを出力して、ログレベル INFO で作業ディレクトリにログを記録します。ログファイルの形式は、logging.format 構成を使用して調整できます。レベルも調整できます。デフォルト設定は、実稼働環境での実行のために、そのままにしておくことをお勧めします。

4 Cisco eStreamer eNcore for Sentinel

4.1 Sentinel へのデータの送信

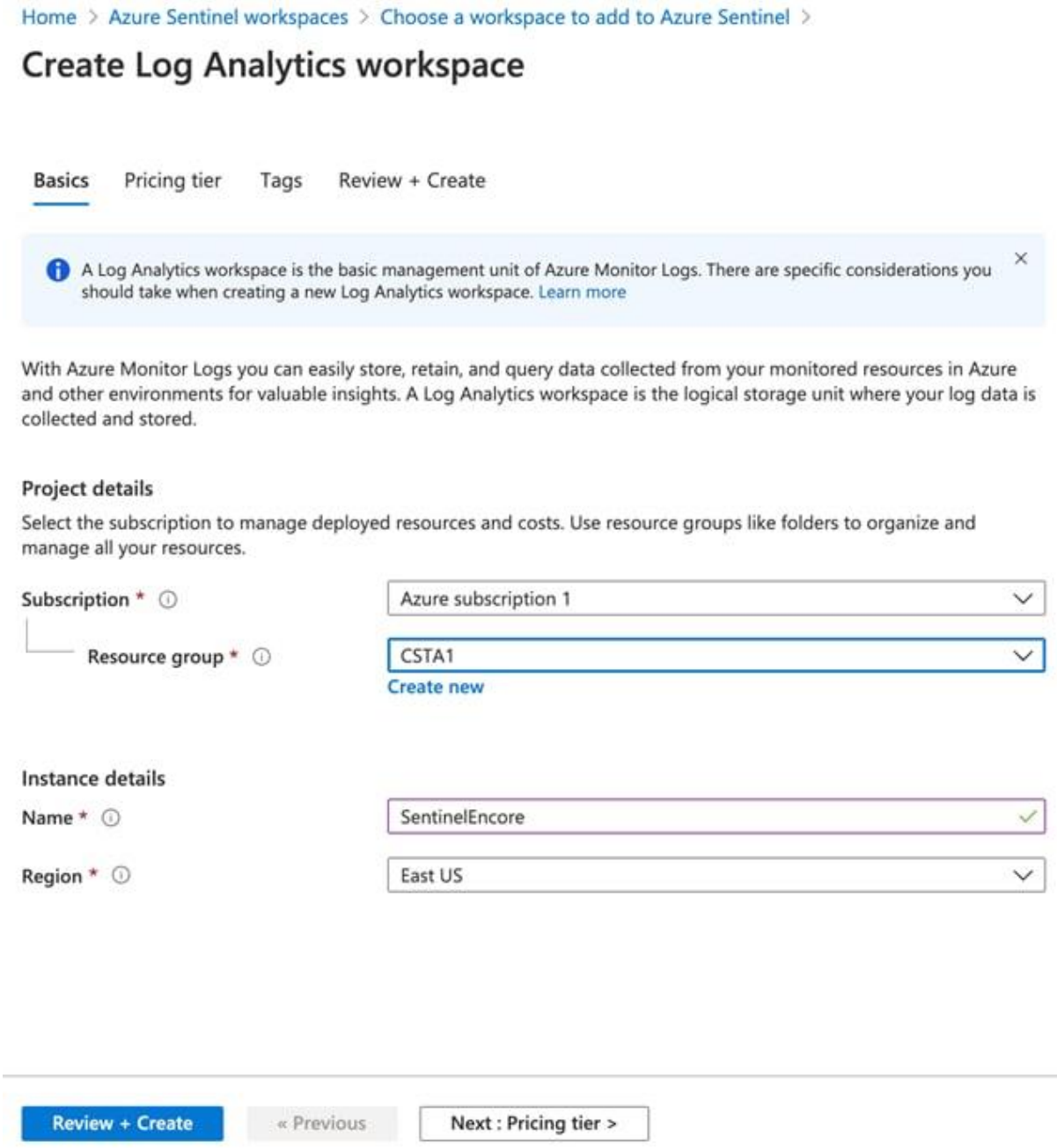
4.1.1 UDP をストリーミングするための encore の設定

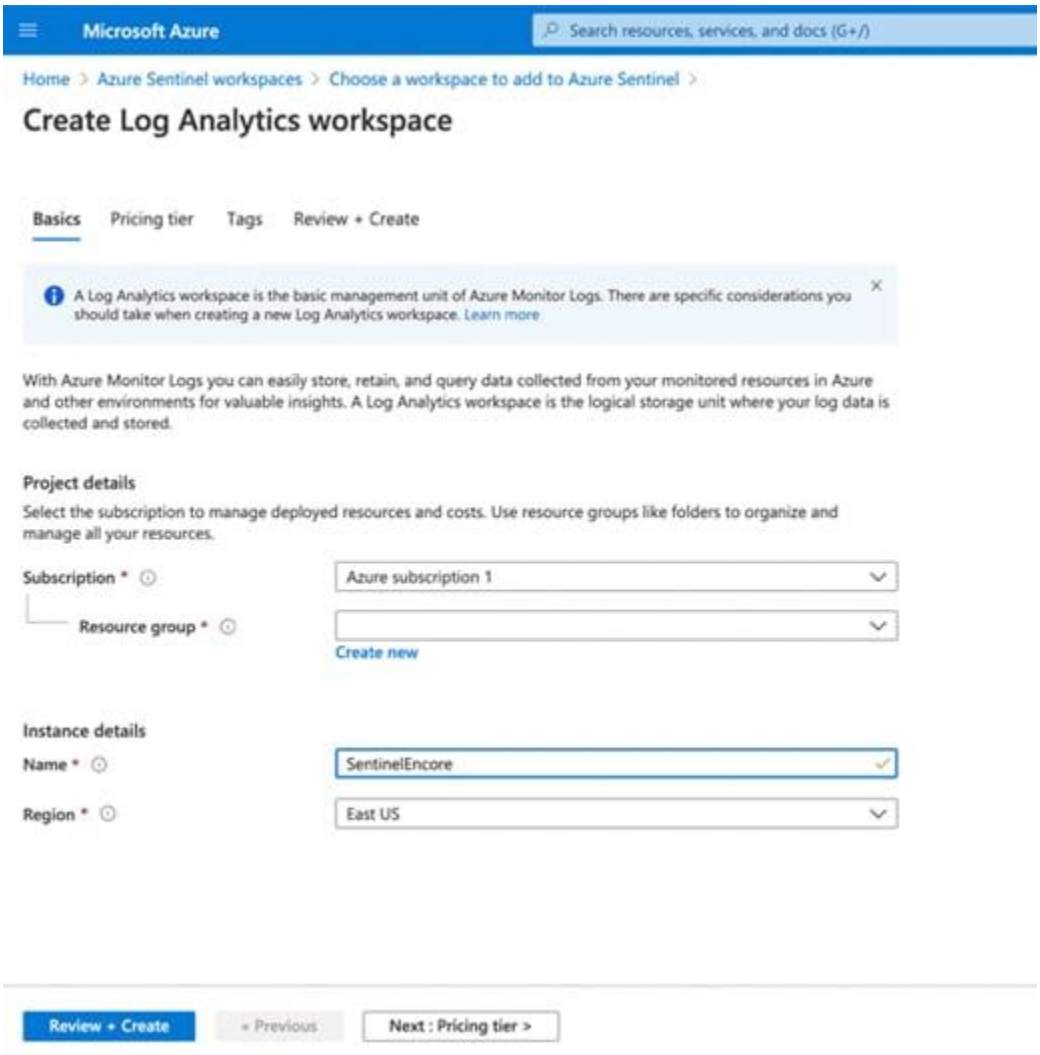
ポート 25226 で UDP を使用して CEF データをストリーミングするように encore を設定します。encore がすでに処理中の場合は、**encore.sh stop/start** コマンドを使用して encore を再起動します。

4.1.2 Sentinel ワークスペースの構成

Firepower Management Center と Azure インスタンスの間に有効な eNcore 接続を確立したら、エージェントコレクタを使用してデータ出力を Sentinel にルーティングできます。

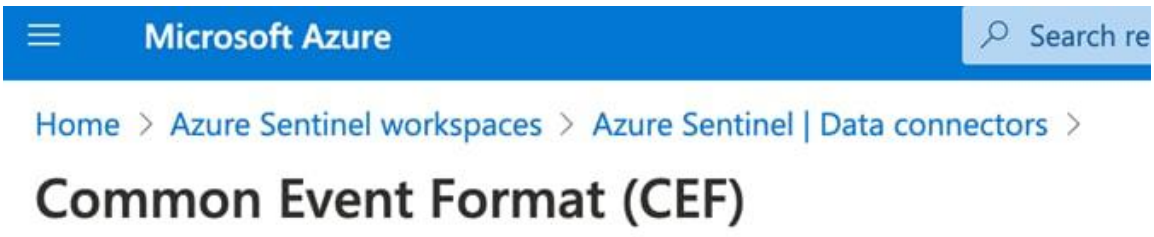
Sentinel ワークスペースがない場合は、次の手順を実行します。





4.1.3 CEF データコネクタの設

Firepower Management Center と Azure インスタンスの間に有効な eNcore 接続を確立したら、エージェントコレクタを使用してデータ出力を Sentinel にルーティングできます。Microsoft 公式のガイド（https://docs.microsoft.com/en-us/azure/sentinel/connect-cef-agent?tabs=rsyslog）を参照してください。



ドキュメントおよび事前入力されたコマンドには Azure インスタンスに固有のワークスペースとプライマリキーの情報が含まれるため、Sentinel からコネクタのドキュメントガイドに直接アクセスすることをお勧めします。
以下の手順は、Azure Sentinel セットアップガイドから直接参照したものです。

（注） エージェントコレクタのインストール時に実行する必要がある正確なコマンドとワークスペース/プライマリ ID が含まれているため、Sentinel プラットフォームで直接ドキュメントを使用することをお勧めします。

次のように、展開スクリプトを実行します。

1. Azure Sentinel のナビゲーションメニューで、[データコネクタ（Data connectors）] をクリックします。
2. コネクタのリストから、[Common Event Format（CEF）（Common Event Format (CEF)）] タイルをクリックし、右下にある [接続ページを開く（Open connection page）] ボタンをクリックします。
3. [1.2 LinuxマシンへのCEFコレクタのインストール（1.2 Install the CEF collector on the Linux machine）] で、[次のスクリプトを実行してCEFコレクタをインストールして適用する（Run the following script to install and apply the CEF collector）] の下に示されているリンク、または下記のテキストをコピーします。

sudo wget
https://raw.githubusercontent.com/Azure/Azure-Sentinel/master/DataConnectors/CEF/cef_installer.py&&sudo python cef_installer.py [ワークスペース ID] [ワークスペース プライマリキー]

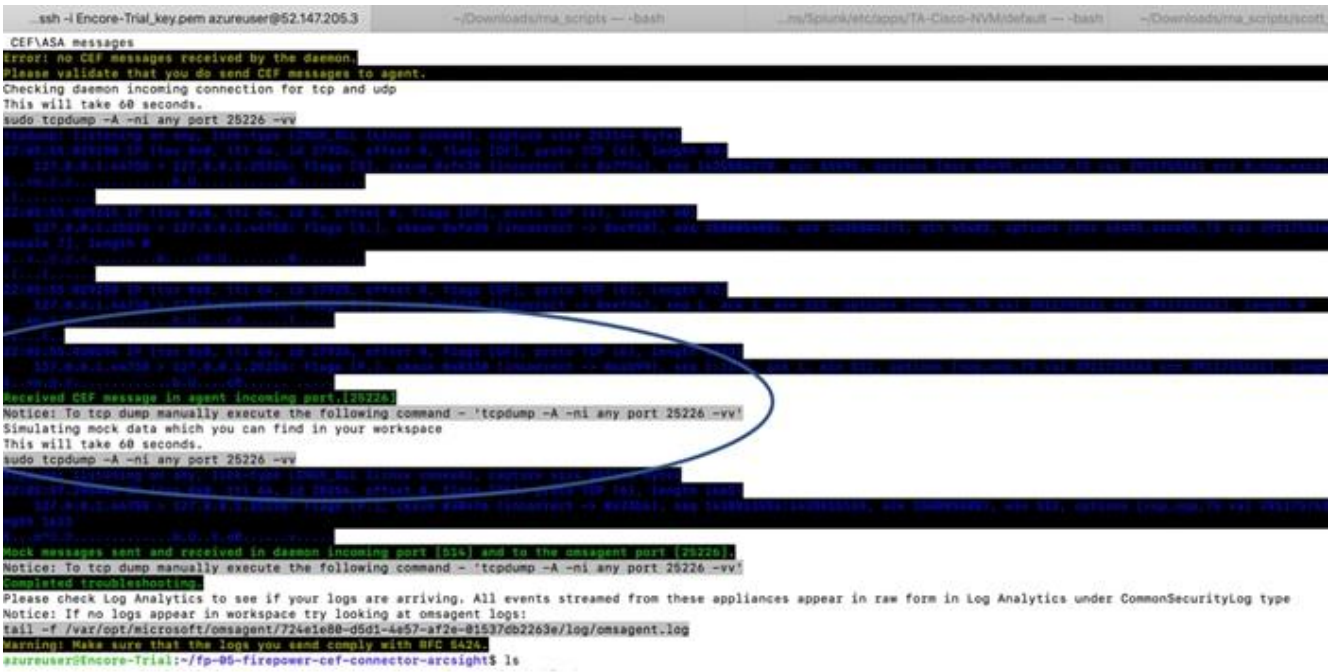
4. スクリプトの実行中に、エラーまたは警告メッセージが表示されないことを確認します。

（注） 同じマシンを使用して、プレーン Syslog および CEF メッセージを転送してください。

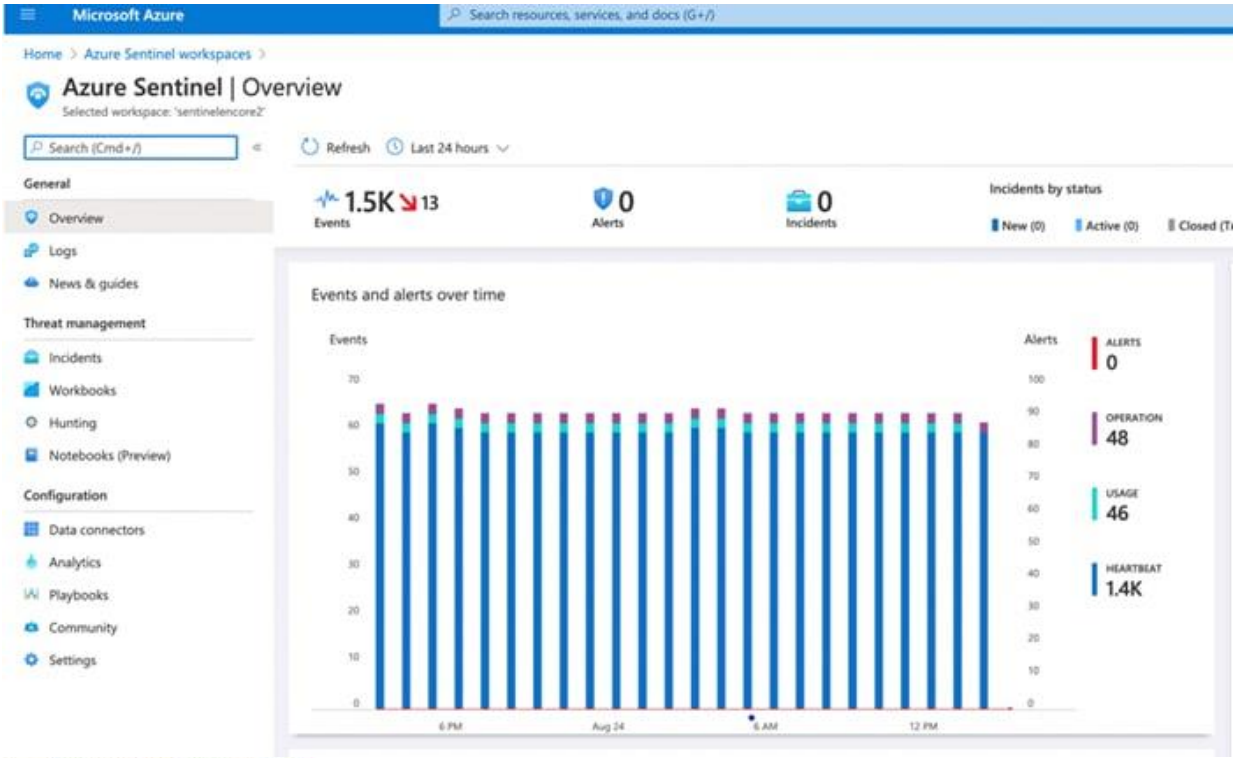
このログフォワーダマシンを使用して Syslog メッセージと CEF を転送する場合は、Syslog および CommonSecurityLog テーブルに対するイベントの重複を避けるために、次の手順を実行する必要があります。

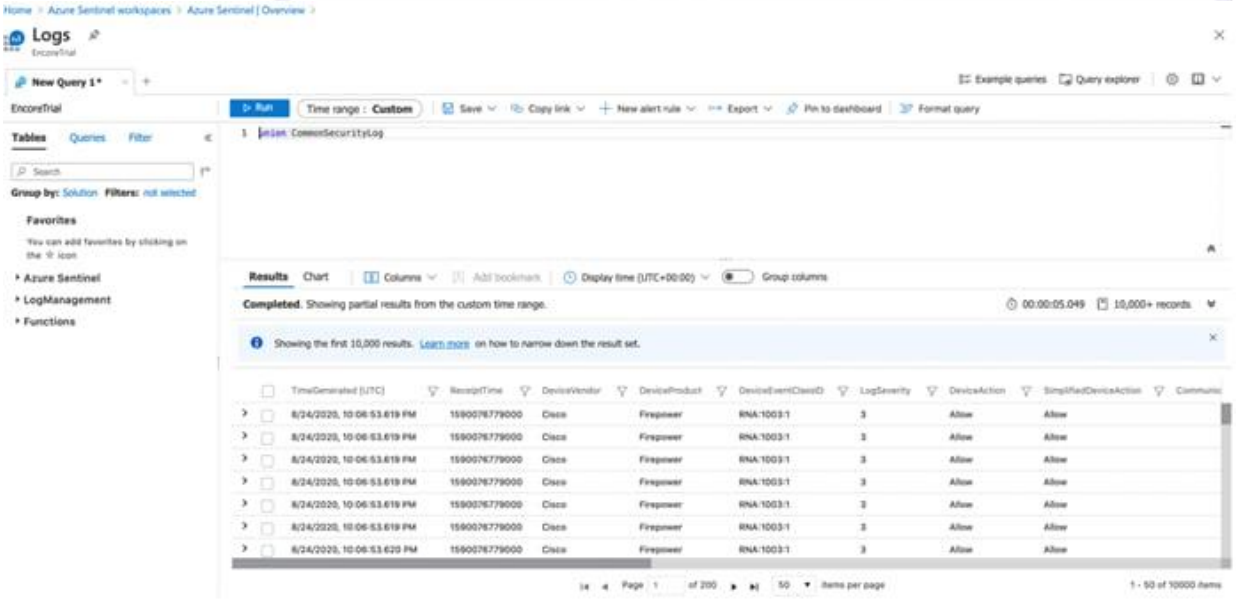
- CEF 形式でフォワーダにログを送信する各ソースマシンで、Syslog 構成ファイルを編集して、CEF メッセージの送信に使用されているファシリティを削除します。これにより、CEF で送信されるファシリティは、Syslog でも送信されなくなります。これを行う方法の詳細な手順については、「Configure Syslog on Linux agent」を参照してください。
- これらのマシンで次のコマンドを実行して、Azure Sentinel の Syslog 構成とエージェントの同期を無効にします。これにより、前の手順で行った構成の変更が上書きされなくなります。
  sudo su omsagent -c 'python /opt/microsoft/omsconfig/Scripts/OMS_MetaConfigHelper.py --disable'
  https://docs.microsoft.com/en-us/azure/sentinel/connect-cef-agent?tabs=rsyslog

検証スクリプトを実行すると、Azure Sentinel Analytics 画面にデータが表示されます。



(注) 「エージェント (着信ポート25226) でCEFメッセージを受信 (Received CEF message in agent incoming port [25226]) 」というメッセージが表示されている場合、エージェントの検証と構成が成功したことを示します。


Microsoft Azure
Search resources, services, and docs (G+/)
Home > Azure Sentinel workspaces >
Azure Sentinel | Overview
Selected workspace: 'sentinelencore2'
Search (Cmd+/)
Refresh
Last 24 hours
General
Overview
Logs
News & guides
Threat management
Incidents
Workbooks
Hunting
Notebooks (Preview)
Configuration
Data connectors
Analytics
Playbooks
Community
Settings
1.5K 13
Events
0
Alerts
0
Incidents
Incidents by status
New (0)
Active (0)
Events and alerts over time
Events
Alerts
ALERTS
0
OPERATION
48
USAGE
46
HEARTBEAT
1.4K
6 PM
Aug 24
6 AM
12 PM


Logs
New Query 1*
Example queries
Query explorer
EncoreTrial
Run
Time range : Custom
Save
Copy link
New alert rule
Export
Pin to dashboard
Format query
Tables
Queries
Filter
Group by: Solution Filters: not selected
Favorites
Azure Sentinel
LogManagement
Functions
1 union CommonSecurityLog
Results
Chart
Columns
Add bookmark
Display time (UTC+00:00)
Group columns
Completed. Showing partial results from the custom time range.
00:00:05.049
10,000+ records
Showing the first 10,000 results. Learn more on how to narrow down the result set.
TimeGenerated [UTC]
ReceiptTime
DeviceVendor
DeviceProduct
DeviceEventClassID
LogSeverity
DeviceAction
SimplifiedDeviceAction
8/24/2020, 10:06:53.619 PM
1590076779000
Cisco
Firepower
RNA:1003:1
3
Allow
8/24/2020, 10:06:53.620 PM
Page 1 of 200
50 items per page
1 - 50 of 10000 items

5 Cisco eStreamer eNcore Add-on for Splunk 8.1+（TA-eStreamer）

Cisco eStreamer eNcore Add-on for Splunk（TA-eStreamer）

eStreamer eNcore Add-on for Splunk は、コア eNcore eStreamer クライアントコードと以下を含むテクノロジーアドオンです。

- データ、ログ、およびステータスのデータ入力（inputs.conf）
- 解析ヒント（props.conf）
- eNcore が Splunk とともに開始および停止することを可能にする拡張機能

(注) Splunk 用 eNcore アドオンは、Windows 向け Splunk ではサポートされていません。

Cisco eStreamer eNcore Dashboard for Splunk（eStreamer ダッシュボード）

これは、以前の Splunk 向け Cisco eStreamer アプリケーション(https://splunkbase.splunk.com/app/1629/)と同じユーザインターフェイス要素を含むアプリケーションです。ただし、コードやコレクタ要素は含まれていません。これは、定義済みの検索、
マクロ、イベントタイプ、およびワークフローアクションを備えたシンプルな UI アプリケーションです。

5.1 前提条件

eNcore Add-on for Splunk と eNcore Dashboard for Splunk には、特別な前提条件は必要ありません。これらは Splunkbase からダウンロード可能で、他のアドオンやアプリケーションと同じ方法で検索ヘッドにインストールされます。

eNcore Add-on for Splunk には、Python 3.6 以降および openSSL が必要です。最新の Splunk 8.1 リリースには Python3 が含まれていますが、openSSL の Python mod は含まれていないため、この更新で説明されている追加の構成手順が必要です。Splunk のインストールがカスタマイズされていて、一方または両方のコンポーネントがない場合は、アドオンを機能させるためにインストールする必要があります。

5.2　インストール

（注） Splunk 用 eNcore アドオンは、Windows 向け Splunk ではサポートされていません。

5.2.1　eNcore Add-on for Splunk（TA-eStreamer）のインストール

eNcore Add-on for Splunk をインストールするには、次のいずれかを実行します。

- http://apps.splunk.com/app/3662 からアドオンをダウンロードし、Splunk の [ファイルからアプリをインストール（Install app from file）] 機能を使用してアドオンをアップロードおよびインストールします。
- Splunk の [他のアプリを参照(Browse more apps)] 機能を使用して eNcore を検索し、検索結果で Cisco eStreamer Add-on for Splunk を見つけ、そのアドオンの [インストール (Install)] をクリックします。

Splunk サーバの PKCS12 証明書をインストールする必要があります。これにより、eNcore クライアントが Firepower Management Center に接続してセキュアトンネルを確立するときに、Firepower Management Center がアドオンの ID を認証できるようになります。

- Firepower Management Center で PKCS12 証明書を作成します。
- 証明書をダウンロードします。
- Splunk サーバの次の 2 つの場所に証明書をコピーします（client.pkcs12 に名前を変更します）。

  $SPLUNK_HOME/etc/apps/TA-eStreamer/bin/encore/client.pkcs12

  $SPLUNK_HOME/etc/apps/TA-eStreamer/bin/client.pkcs12

PKCS12 証明書の作成および Splunk サーバへのコピーの詳細については、「付録」を参照してください。

5.2.2 eNcore Dashboard for Splunk（eStreamer ダッシュボード）のインストール

eNcore Dashboard for Splunk をインストールするには、次のいずれかを実行します。

- http://apps.splunk.com/app/3663 からアプリケーションをダウンロードし、Splunk の [ファイルからアプリをインストール（Install app from file）] 機能を使用してアドオンをアップロードおよびインストールします。
- Splunk の [他のアプリを参照（Browse more apps）] 機能を使用して「eNcore」を検索し、検索結果で Cisco Firepower eNcore App for Splunk を見つけ、そのアプリケーションの [インストール（Install）] をクリックします。

5.3 eNcore Add-on for Splunk のセットアップ構成

5.3.1 データ入力の有効化

eNcore Add-on for Splunk は、インストールのデータディレクトリ内のログファイルにイベントを書き込みます。このディレクトリからイベントを読み取るデータ入力を使用して、Splunk を設定する必要があります。

これを行うには、[設定（Settings）] > [データ入力（Data Inputs）] > [ファイルとディレクトリ（Files & Directories）] に移動し、パス $SPLUNK_HOME/etc/apps/TA-eStreamer/data およびソースタイプ（Source type） cisco:estreamer:data を使用してデータ入力を有効にします。

5.3.2 スクリプトの有効化

eNcore Add-on for Splunk には、重要な操作を実行する次の 3 つのスクリプトがあります。

- **cisco:estreamer:clean**：出力はありませんが、12 時間以上経過したデータファイルを削除するために使用します。
- **cisco:estreamer:log**：eNcore の stdout を使用してプログラムログデータを取得します。これは、想定どおりに処理が進んでいない場合に役立ちます。さらに重要な点は、eStreamer eNcore プロセスを開始するスクリプトであるということです。
- **cisco:estreamer:status**：プログラムが実行されているかどうかの明示的なステータスを保持するために定期的に実行します。

スクリプトを有効にするには、[設定（Settings）] > [データ入力（Data Inputs）] > [スクリプト（Scripts）] に移動し、3 つの TA-eStreamer スクリプトの [有効化（Enable）] をクリックします。

5.3.3 eNcore アドオンのセットアップ構成

$SPLUNK_HOME/etc/apps/TA-eStreamer/bin にある TA-eStreamer bin ディレクトリに移動します。ここで、$SPLUNK_HOME は、Splunk ヘビーフォワーダのインストールのホームディレクトリを表します。

SPLUNK_HOME インストール変数のホームパス（SPLUNK_HOME）を設定するには、次のコマンドを実行します。

```
export SPLUNK_HOME=/opt/splunk
```

/opt/splunk は、Splunk インストールのホームロケーションです（必要に応じて変更してください）。

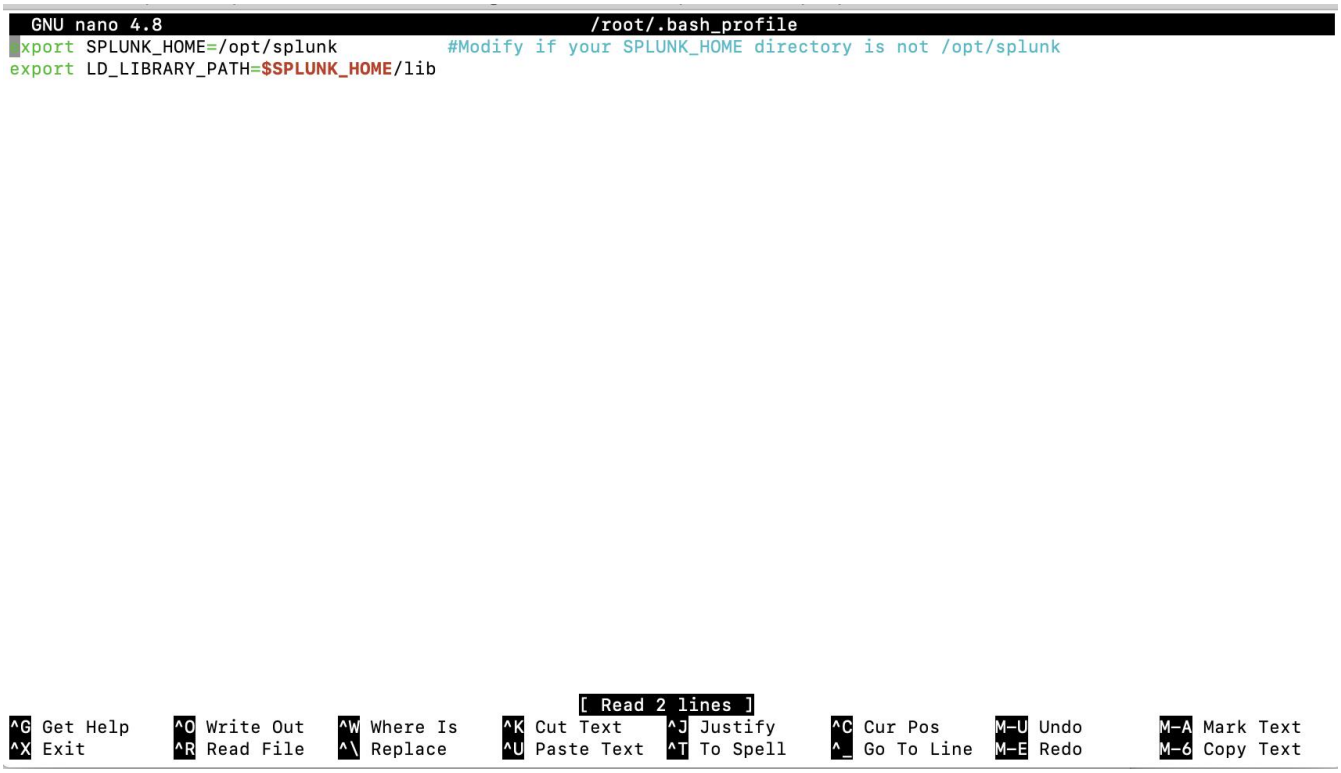
起動/テストスクリプトを実行すると、次のエラーが表示される場合があります。

/opt/splunk/bin/openssl: error while loading shared libraries: libssl.so.1.0.0: cannot open shared object file: No such file or directory

これを解決するには、Splunk Lib パスのセットアップ変数を 1 つ追加します。これはスクリプト内でコメントアウトされているため、セットアップスクリプトを実行する前に次のコマンドを実行する必要があります。

export LD_LIBRARY_PATH=$SPLUNK_HOME/lib

SPLUNK_HOME および LD_LIBRARY_PATH の設定はローカル端末セッションに含まれます。これらの値を保持するには、次の手順を実行します。



Ubuntu の場合：

1 ~/.bash_profile ファイルを編集します。

2 上記の export 変数を次のようにして追加します。

- export SPLUNK_HOME = /opt/splunk
- export LD_LIBRARY_PATH=SPLUNK_HOME/lib

3 ファイルを保存し、source ~/.bash_profile を実行します。

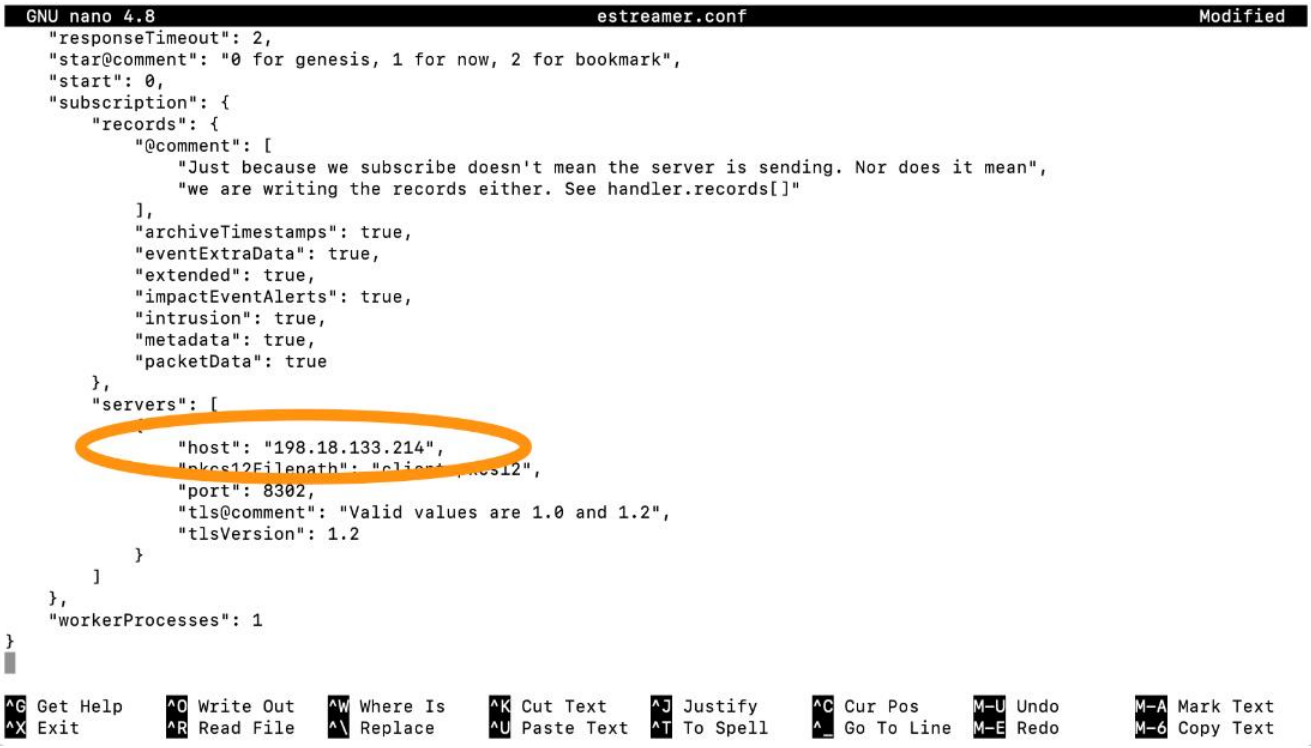
```
root@splunk-8-1:~# nano ~/.bash_profile
root@splunk-8-1:~# source ~/.bash_profile
root@splunk-8-1:~#
```

CentOS の場合：

1 bash プロファイルは別のエイリアスを使用している場合があります。~./profile または ~/.bashrc を試してください。

2 ファイルを編集し、保存して、この source コマンドを実行します。

https://community.splunk.com/t5/Developing-for-Splunk-Enterprise/How-to-get-Splunk-Python-on-CentOS-to-use-SSL-Crypto/m-p/310051
Firepower Management Center サーバのホスト IP アドレスを指すように estreamer.conf ファイルを変更します。



3 設定 **false** を編集して **true** に変更します。

4 **./splencore.sh test** コマンドを実行します。

```
root@splunk-8-1:/opt/splunk/etc/apps/TA-eStreamer/bin# ./splencore.sh test
2021-06-14T19:25:01.680552 Diagnostics  INFO    Checking that configFilepath (estreamer.conf) exists
2021-06-14 19:25:01,692 Diagnostics  INFO     Check certificate
2021-06-14 19:25:01,693 Diagnostics  INFO     PKCS12 file needs processing
Please enter the PKCS12 password (press <enter> for blank password):
2021-06-14T19:25:13.998455 Diagnostics  ERROR   [no message or attrs]:

# Splunk 8.1+ Python3 does not natively support openssl, please perform the following

Run the following two commands, alernatively you can use the command line version of OpenSSL

$SPLUNK_HOME/bin/openssl pkcs12 -in "client.pkcs12" -nocerts -nodes -out "/opt/splunk/etc/apps/TA-eStreamer/bin/encore/19
8.18.133.214-8302_pkcs.key"
$SPLUNK_HOME/bin/openssl pkcs12 -in "client.pkcs12" -clcerts -nokeys -out "/opt/splunk/etc/apps/TA-eStreamer/bin/encore/1
98.18.133.214-8302_pkcs.cert"

Note:  If you are using python3 the command to install OpenSSL is as follows, using python3 with openssl will
       not require manual commands above and this script will automatically extract and process certificate files

    sudo apt install python3-openssl


root@splunk-8-1:/opt/splunk/etc/apps/TA-eStreamer/bin#
```

5 client.pkcs 証明書のパスワードを入力します。最初は失敗し、次のコマンドを入力するように求められます。

コマンド 1：

$SPLUNK_HOME/bin/openssl pkcs12 -in "client.pkcs12" -nocerts -nodes -out "/opt/splunk/etc/apps/TA-eStreamer/bin/encore/198.18.133.214-8302_pkcs.key"

コマンド 2：

$SPLUNK_HOME/bin/openssl pkcs12 -in "client.pkcs12" -clcerts -nokeys -out "/opt/splunk/etc/apps/TA-eStreamer/bin/encore/198.18.133.214-8302_pkcs.cert"

（注） 198.18.133.214 を Firepower Management Center サーバのホスト IP アドレスに置き換えてください。

```
root@splunk-8-1:/opt/splunk/etc/apps/TA-eStreamer/bin#
root@splunk-8-1:/opt/splunk/etc/apps/TA-eStreamer/bin# $SPLUNK_HOME/bin/openssl pkcs12 -in "client.pkcs12" -nocerts -nodes -out "/opt/splunk/etc/apps/TA-eS
treamer/bin/encore/198.18.133.214-8302_pkcs.key"
WARNING: can't open config file: /opt/splunk-home/openssl/openssl.cnf
Enter Import Password:
MAC verified OK
root@splunk-8-1:/opt/splunk/etc/apps/TA-eStreamer/bin# $SPLUNK_HOME/bin/openssl pkcs12 -in "client.pkcs12" -clcerts -nokeys -out "/opt/splunk/etc/apps/TA-e
Streamer/bin/encore/198.18.133.214-8302_pkcs.cert"
WARNING: can't open config file: /opt/splunk-home/openssl/openssl.cnf
Enter Import Password:
MAC verified OK
root@splunk-8-1:/opt/splunk/etc/apps/TA-eStreamer/bin#
```

6 各プロンプトの後に Firepower Management Center の client.pkcs 証明書のパスワードを入力します。成功した場合、各コマンドの後に「MAC 検証 OK（MAC verified OK）」というテキストが表示されます。

```
client.pkcs12  configure_handler.py  configure.sh  encore  setup.xml  splencore.sh
root@ubuntu-splunk:/opt/splunk/etc/apps/TA-eStreamer/bin# ./splencore.sh test
2021-06-28T16:34:53.884468 Diagnostics  INFO    Checking that configFilepath (estreamer.conf) exists
2021-06-28 16:34:53,896 Diagnostics  INFO     Check certificate
2021-06-28 16:34:53,896 Diagnostics  INFO     PKCS12 file needs processing
Please enter the PKCS12 password (press <enter> for blank password):
2021-06-28T16:34:57.314872 Diagnostics  ERROR   [no message or attrs]:

# Splunk 8.1+ Python3 does not natively support openssl, please perform the following

Run the following two commands, alernatively you can use the command line version of OpenSSL

$SPLUNK_HOME/bin/openssl pkcs12 -in "client.pkcs12" -nocerts -nodes -out "/opt/splunk/etc/apps/TA-eStreamer/bin/encore/198.18.133.1
94-8302_pkcs.key"
$SPLUNK_HOME/bin/openssl pkcs12 -in "client.pkcs12" -clcerts -nokeys -out "/opt/splunk/etc/apps/TA-eStreamer/bin/encore/198.18.133.
194-8302_pkcs.cert"

Note:  If you are using python3 the command to install OpenSSL is as follows, using python3 with openssl will
       not require manual commands above and this script will automatically extract and process certificate files

    sudo apt install python3-openssl


root@ubuntu-splunk:/opt/splunk/etc/apps/TA-eStreamer/bin# $SPLUNK_HOME/bin/openssl pkcs12 -in "client.pkcs12" -nocerts -nodes -out
 "/opt/splunk/etc/apps/TA-eStreamer/bin/encore/198.18.133.194-8302_pkcs.key"
WARNING: can't open config file: /opt/splunk-home/openssl/openssl.cnf
Enter Import Password:
MAC verified OK
root@ubuntu-splunk:/opt/splunk/etc/apps/TA-eStreamer/bin# $SPLUNK_HOME/bin/openssl pkcs12 -in "client.pkcs12" -clcerts -nokeys -out
 "/opt/splunk/etc/apps/TA-eStreamer/bin/encore/198.18.133.194-8302_pkcs.cert"
WARNING: can't open config file: /opt/splunk-home/openssl/openssl.cnf
Enter Import Password:
MAC verified OK
root@ubuntu-splunk:/opt/splunk/etc/apps/TA-eStreamer/bin#
```

7 **./splencore.sh test** コマンドを実行すると、次のように表示されます。

```
root@ubuntu-splunk:/opt/splunk/etc/apps/TA-eStreamer/bin# ./splencore.sh test
2021-06-28T16:36:30.492874 Diagnostics  INFO    Checking that configFilepath (estreamer.conf) exists
2021-06-28 16:36:30,506 Diagnostics  INFO     Check certificate
2021-06-28 16:36:30,506 Diagnostics  INFO     Creating connection
2021-06-28 16:36:30,506 Connection   INFO     Connecting to 198.18.133.194:8302
2021-06-28 16:36:30,506 Connection   INFO     Using TLS v1.2
2021-06-28 16:36:30,544 Diagnostics  INFO     Creating request message
2021-06-28 16:36:30,544 Diagnostics  INFO     Request message=b'00010002000000008ffffffff48900061'
2021-06-28 16:36:30,544 Diagnostics  INFO     Sending request message
2021-06-28 16:36:30,544 Diagnostics  INFO     Receiving response message
2021-06-28 16:36:30,553 Diagnostics  INFO     Response message=b'gAN9cQAoWAcAAAB2ZXJzaW9ucQFLAVgLAAAAbWVzc2FnZVR5cGVxAk0DCFgGAAAAbG
VuZ3RocQNLMFgEAAAAZGF0YXEEQzAAABOJAAAACAAAAAAAAAAAAAATiAAAAAgAAAAAAAAAAAAAAGgsAAAAIAAAAAAAAAABxBXUu'
2021-06-28 16:36:30,554 Diagnostics  INFO     Streaming info response
2021-06-28 16:36:30,554 Diagnostics  INFO     Connection successful
root@ubuntu-splunk:/opt/splunk/etc/apps/TA-eStreamer/bin#
```

5.4 動作

セクション 4 の説明に従ってすべての項目を適切に設定した後、アドオンセットアップページの [有効化（Is enabled）] チェックボックスをオンにし、[名前を付けて保存（Save as）] をクリックして（セクション 4.3 の説明を参照）、eNcore Add-on for Splunk を起動します。

実行後、次のように、ステータス、ログ、データのイベントを検索することでアドオンの動作をモニタリングできます。

- ステータスを確認するには、**sourcetype="cisco:estreamer:status"** を検索します。
- 詳細なログ出力を確認するには、**sourcetype="cisco:estreamer:log"** を検索します。
- eStreamer データを探すには、**sourcetype=" cisco:estreamer:data"** を検索します。

Firepower イベントをより詳しく分析するには、Cisco Firepower App for Splunk をインストールすることを検討してください。

6 Splunk 用Firepower ダッシュボード

6.1 インバウンド/アウトバウンドサブネットの構

プラットフォームごとに eNcore によって用意されたデフォルト設定が **estreamer.conf** ファイル内にあります。これは、多くの展開に最適な構成を提供します。ただし、状況により、ユーザが調整する必要があるオプションもあります。ここでは、これらのオプションについて詳しく説明します。

6.2 レコード数(Records)

records セクションでは、eNcore が処理するレコードを指定します。次の 2 つの方法で、どのイベントを処理するか（または処理から除外するか）を指定できます。

- 接続（connections）など、目的とするイベントクラスの値を true に設定することで、そのクラスを処理するように指定できます。

この例の場合、キーと値のペアは **"connections": true** になります。逆に、イベントクラスの値を false に設定して、そのクラスを処理しないように指定することもできます。

- レコードタイプを include キーまたは exclude キーの値として記述することで、レコードタイプごとにイベントクラスの処理の例外を指定できます。JSON 配列では、複数の値をカンマで区切る必要があります。

たとえば、レコードタイプ 98 および 170 を除外するには、exclude キーと値のペアは次のようになります。

"exclude": [98, 170],

records のキーと値のペアの例を次に示します。

(注) 処理するレコードのクラスは、最初に Firepower Management Center eStreamer 構成で選択する必要があることに注意してください。また、eNcore 構成の subscription セクションの records 部分でサブスクリプション用に設定する必要もあります。

```
"records": {
        "connections": true,
        "core": true,
        "excl@comment": [
          "These records will be excluded regardless of above (overrides 'include')",
          "e.g. to exclude flow and IPS events use [ 71, 400 ]"
        ],
        "exclude": [],
        "inc@comment": "These records will be included regardless of above",
        "include": [],
        "intrusion": true,
        "metadata": false,
        "packets": true,
        "rna": true,
        "rua": true
      }
```

6.3 モニタ

モニタは、モニタリングタスクとメンテナンスタスクを実行する独立したスレッドです。デフォルトでは、2 分ごとに実行されます。これは、処理されたイベントの数を eNcore ログに書き込み、サブプロセスのステータスをチェックします。サブプロセスに問題がある場合、モニタはクライアントをエラー状態にし、クライアントはシャットダウンします。

モニタスレッドによってログに書き込まれるメッセージの例を次に示します。

```
2018-08-30 05:09:15,026 Monitor      INFO     Running. 2296400 handled; average rate 578.86 ev/sec;
2018-08-30 05:11:15,684 Monitor      INFO     Running. 2296400 handled; average rate 561.87 ev/sec;
2018-08-30 05:13:15,384 Monitor      INFO     Running. 2296400 handled; average rate 545.86 ev/sec;
```

ログメッセージのいくつかの項目は、次の場所にある estreamer.conf 構成ファイルの monitor セクションで設定できます。

Splunk：**$SPLUNK_HOME/etc/apps/TA-eStreamer/bin/encore/estreamer.conf**

Sentinel/CEF：**/ fp-05-firepower-cef-connector-arcsight/estreamer.conf**

設定できる項目は次のとおりです。

- period：モニタがサブプロセスのチェックを実行し、ステータスメッセージをログに書き込む間隔（秒単位）。
- bookmark：true の場合、ブックマーク（Unix 時刻形式での最新イベントの時刻）が各モニタログメッセージに含まれます。
- handled：true の場合、eNcore が起動後に処理したイベントの数。
- details：true の場合、モニタによってログに書き込まれる簡単なステータスメッセージに加えて、eNcore クライアントの操作に関連する多くのステータス項目を含む詳細なメッセージも書き込まれます。

estreamer.conf ファイル内のこれらのパラメータの構成例を次に示します。

```
"monitor": {
   "period": 120,
   "bookmark": false,
   "handled": true,
   "details": true
},
```

6.4 Start Time

クライアント要求では、eStreamer サーバに対して開始時間を指定する必要があります。この開始時間以降に発生したイベントのみが Firepower Management Center から送信されます。3 つのオプションがあります。

- 0：Firepower Management Center で使用可能な最も古い時点からすべてのイベントを送信します。
- 1：クライアント要求の受信後に発生したすべてのイベントを送信します。
- 2：ブックマークを使用して、中断した場所から再開します。最初の実行では、0 からになります。

estreamer.conf ファイル内の start 構成の例を次に示します。

```
"@startComment": "0 for genesis, 1 for now, 2 for bookmark",
   "start": 2,
```

6.5 アウトプッタ

デフォルトでは、Splunk アウトプッタのみが有効になっています。データは相対的なファイルの位置に書き込まれますが、別の場所にデータを出力することもできます。これを変更するには、**stream.uri**

プロパティを **file:///absolute/file/path/filename{0}.ext** に変更します。{0} はタイムスタンプのプレースホルダです。

estreamer.conf ファイル内の outputters 構成の例を次に示します。

```
"outputters": [
  {
    "name": "Splunk default",
    "adapter": "splunk",
    "enabled": true,
    "stream": {
      "uri": "relfile:///data/splunk/encore.log{0}",
      "options": {
        "rotate": true,
        "maxLogs": 9999
      }
    }
  },

"outputters": [
  {
    "name": "Arcsight",
    "adapter": "cef",
    "enabled": true,
    "stream": {
      "uri": "relfile:///data/cef/encore{0}.cef",
      "options": {
        "rotate": true,
        "maxLogs": 9999
      }
    }
  },
```

6.6　パフォーマンスの調整

マルチプロセッシングの追加により、バージョン 4.x では、Splunk 用 eNcore アドオンのパフォーマンスが向上しています。デフォルトでは、4 つのワーカープロセスが着信メッセージを処理して、より高いスループットを実現します。

複数のプロセスでパフォーマンスを大幅に向上させることができますが、プラットフォームごとに処理のボトルネックが異なるため、このパフォーマンス向上はプラットフォームに大きく依存します。複数のプロセスではタスクの分散を管理するための追加のオーバーヘッドも必要となるため、CPU コ

ア数の少ないプラットフォームでは、プロセス数を増やすと、実際にはパフォーマンスが低下する可能性があります。

ワーカープロセスの数は、**estreamer.conf** ファイルの **workerProcesses** パラメータで設定できます。この数は 1 ～ 12 の範囲で設定できます。一般に、プラットフォームの性能が高いほど（CPU コア数が多い、I/O が優れているなど）、より多くのワーカープロセスによって、より高いスループットが実現されます。ただし、信頼できる唯一の方法は、1、2、4、8、12 などのさまざまな設定でパフォーマンスをテストすることです。多くの場合は、ワーカープロセスを 1 つだけにすると、プロセスマーシャリングが不要となるため、最適なパフォーマンスが得られます。

テストのシナリオ例を次に示します。

1. テスト中には同じイベントが何度も要求されるため、Splunk でアドオンのデータ入力を無効にします。
2. workerProcesses の設定数（8 など）を指定し、start パラメータ 0（発生時点から取得）または少なくとも古い開始時間で eNcore を起動します。
3. Firepower Management Center に対して接続イベントを要求します（または別の方法で、多数のバックログイベントを送信するよう Firepower Management Center に要求します）。
4. estreamer.log ファイルでモニタプロセスによって報告されたイベントレートを確認します。
5. workerProcesses の数を変えてテストを繰り返します。
6. 最適な数を特定したら、workerProcesses をその数に設定し、アドオンのデータ入力を有効にして実稼働環境での運用を再開します。

estreamer.conf ファイル内の workerProcesses 構成の例を次に示します。

イベントレート（1 秒あたり）	**ワーカースレッド数**	**バッチサイズ（推奨）**
100 未満	1	2
100 - 2000	1	100（デフォルト）
2000-4000	4	100（デフォルト）
4000-6000	8	250
8000+	12	500

一般的な Splunk ヘビーフォワーダでは 1 秒あたり平均 4,000 ～ 5,000 件のイベントを処理できますが、このレートはオペレーティングシステムで利用可能なリソース、追加のバックグラウンドタスク、そ

の他の TA（テクノロジーアドオン）などの影響を受けるため、実際のパフォーマンスは低くなる可能性があります。

専用 VM を使用する場合、仕様は処理するボリュームに大きく依存します。1 秒あたり約 4,000 イベントを処理する一般的なインストール環境については、3.6 GHz の 8 コア CPU、32 GB RAM のマシンまたは (c5.2x large ec2 インスタンス) にインストールする必要があります。1 日あたり少量のイベント (100 件未満のイベント）しか処理しない軽量クライアントの場合、eNcore は、最小仕様 4 コア、1 GB RAM で動作することがテストおよび確認済みです。

6.7　バッチサイズ

Splunk 用 eNcore アドオンでは、受信したイベントをバッチ処理し、バッチのしきい値に達したときにのみ出力に書き込むことで、パフォーマンスの向上を図ります。デフォルトのバッチサイズは 100 イベントです。

イベントレートが非常に低い場合、バッチサイズが 100 イベントであると、Splunk でのイベントの表示に望ましくない遅延が生じる可能性があります。

たとえば、処理される唯一のイベントが侵入イベントであり、侵入イベントレートが 1 時間あたり平均 100 イベントである場合、バッチの最初のイベントは、通常、バッチが完了してディスクに書き込まれるまでに 1 時間以上遅延することになります。このような遅延を減らすには、batchSize の設定値を小さくします。また、完全に遅延をなくすには、batchSize を 1 に設定します。

batchSize を 1 に設定することの欠点は、高スループット環境では全体的なイベントレートが低くなることです。さらに、イベントが常にディスクに書き込まれるため、ファイルのロックとローテーションが問題になる可能性があります。そのため、batchSize は 2 以上に設定することを強く推奨します。estreamer.conf ファイル内の batchSize 構成の例を次に示します。

"batchSize": 50

高ボリュームの構成では、batchSize を 500 程度に設定すると、最適なパフォーマンスを実現できます。繰り返しになりますが、batchSize を大きくすると、クライアントがディスクに書き込む頻度が減り、ファイル I/O が少なくなります。これにより、コンピューティングの負荷が軽減しますが、イベントの処理に若干の遅延が生じます。

6.8 接続の維持

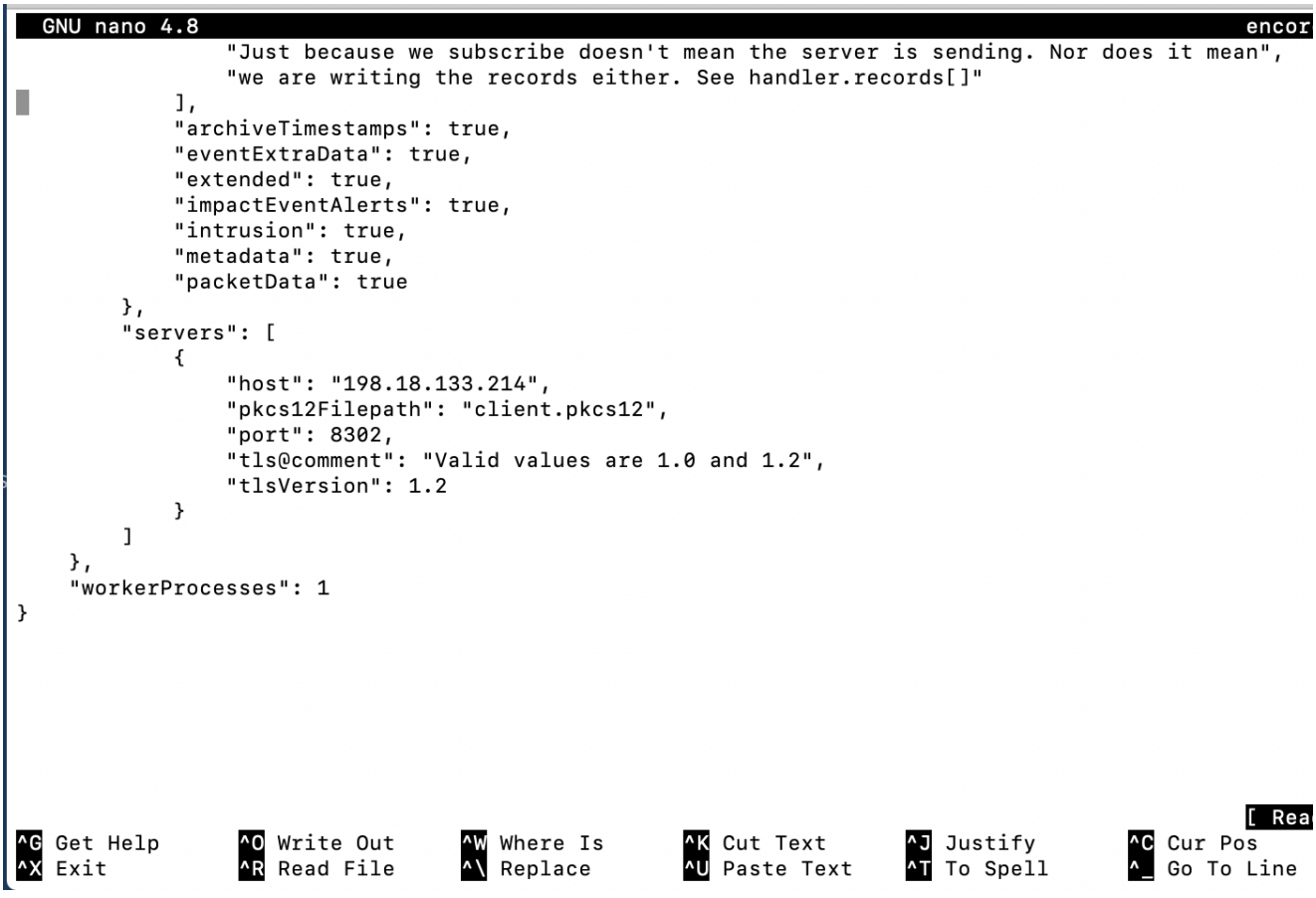
クライアントを無期限に維持することは、Firepower Management Center からのデータストリームをリッスンするためや、Splunk の停止後に eNcore を自動的に再起動するために役立つことがあります。これは、次の構成値を true に設定することで実現できます。

"alwaysAttemptToContinue": true

6.9 ホスト

デフォルトでは、汎用のプレースホルダが estreamer.conf ファイル内で定義されています。必要に応じて、これを Firepower Management Center の IP またはホスト名に変更できます。本書の作成時点では、サポートされているのは IPv4 アドレスのみです。

“host”: “1.2.3.4”



6.10 高度な構成

次の表に、**estreamer.conf** ファイルのキー定義を示します。

キー	定義
alwaysAttemptToContinue	true | false。CLI プロセス が終了した場合でも eNcore クライアントで接続を維持するかどうかを制御します。
batchSize	メモリに格納するイベントの数。この数を超えると、ディスクに書き込まれます。デフォルトは 100 です。このしきい値に達するまで、イベントはメモリ内のキューに保持されるため、トラフィック量が少ない場合はこの値を小さくします。 高ボリュームの実装では、この値を大きくすると、ファイル I/O アクセス要求の数が限定され、クライアントのパフォーマンスが向上しますが、イベントの遅延が生じる可能性があります。遅延の程度は取り込みレートに依存します。たとえば、取り込みレートが 100 イベント/秒で、batchSize が 500 の場合、5 秒ごとにデータがディスクに書き込まれます。
enabled	true | false。eNcore を実行するかどうかを制御します。
connectTimeout	クライアントが接続の確立を待機する時間（秒単位）。この時間を超えると、エラーになります。
responseTimeout	クライアントが応答を待機する時間（秒単位）。この時間を超えると、タイムアウトになります。
monitor.period	モニタタスクを実行する間隔（秒単位）。デフォルトは 120 です。数値が小さいほどデバッグに役立ちますが、ログトラフィックが増加します。
monitor.velocity	true | false。true の場合、クライアントがレコードを処理する速度が表示されます。正の値は、eStreamer によってイベントが送信されるよりも高速にクライアントがイベントを処理していることを意味します。負の値はそれよりも低速ということです。最新の状態になると、ゼロ近辺で推移します。
monitor.bookmark	true | false。true の場合、最後のブックマークのタイムスタンプが表示されます。これは、eNcore クライアントの処理がどの程度遅れているかを確認するのに役立ちます。
monitor.subscribed	true | false。true の場合、サブスクライブされたイベントの合計数が報告されます。

キー	定義
monitor.handled	true | false。true の場合、出力に書き込まれたイベントの合計数が報告されます。
Start	0 は、利用可能な最も古いデータを指定します。 1 は、現時点のデータを指定します。 2 は、ブックマークの使用を指定します。
logging.level	レベルには、FATAL、ERROR、WARNING、INFO、DEBUG、VERBOSE、TRACE があります。要件に応じてロギングのレベルを選択します。 実稼働環境では INFO よりも上のレベルは使用しないことを強くお勧めします。 DEBUG は非常に大きなログファイルを生成し、TRACE はパフォーマンスに大きく影響します。
logging.format	ログの形式と保存方法を示します。メッセージ形式のデフォルト構成は、“{date-time}{name of module}-{level of logging-message}” です。
logging.stdOut	true | false。ログ出力が標準出力にも表示されるかどうかを示します。
logging.filepath	アプリケーションログの場所を指定します。
maxQueueSize	バッファされるメッセージの最大数。この数を超えると、スロットリングが行われます。基本的には、この値はバッファサイズです。この数値が大きいほど、シャットダウンに要する時間が長くなります。デフォルト構成は 100 です。変更しないでください。
subscription.servers[]	これは配列ですが、eNcore で現在サポートできるサーバは 1 つだけです。配列は、将来、複数のホストに接続する機能に対応することを目的としています。
server.host	Firepower Management Center（eStreamer サーバ）の IP アドレス。デフォルト構成は 1.2.3.4 です。eNcore の実行後にホストエントリを変更すると、新しいキャッシュ、ブックマーク、およびメタデータファイルが生成されます。
server.port	接続先のサーバポート。デフォルトは 8302 です。

キー	定義
server.pkcs12Filepath	PKCS12 ファイルパスの場所。すでに eNcore を実行している状態でこれを変更する場合は、キャッシュされた公開キーと秘密キーを削除する必要もあります。削除しないと、eNcore でそれらのキーが引き続き使用されます。これらのキーの名前は {host}-{port}_pkcs.cert と {host}-{port}_pkcs.key です。
server.tlsVersion	有効なオプションは 1.0 と 1.2 です。
subscription.records	この値は変更しないでください。
handler.records.metadata	true | false。（タイムスタンプ情報がないため）メタデータの出力を除外する場合は、この値を false に設定します。
handler.records.flows	true | false。接続フローレコードを除外する場合は、この値を false に設定します。
handler.outputters[]	eNcore による書き込みの動作と形式を定義するアウトプッタコントローラの配列。
outputter.name	人間が読める形式の名前です。便宜上用意されています。コードでは使用されません。
outputter.adapter	データは eStreamer から読み取られ、構造化された内部形式で保存されます。アダプタは、データを必要な形式に変換します。使用可能な値は次のとおりです。 • splunk • json
outputter.enabled	true | false。アウトプッタは一度に複数指定できます。特定のアウトプッタを無効にする場合は、このフラグを false に設定します。すべてのアウトプッタが false の場合(またはアウトプッタがない場合)、シンクとして動作します。
outputter.passthru	true | false。true の場合、転送データは復号化とメタデータの処理をバイパスします。この場合、非常に高速になりますが、用途は限られます。主な目的はデバッグです。
outputter.stream.uri	出力を保存する場所を指定します。ファイル URI は、絶対パス（file:///absolute/path/to/file など）または相対パス（relfile:////relative/path/to/file など）で指定できます。 現在、ファイル URL のみがサポートされています。
outputter.stream.options	ファイルベースのストリームには追加オプションが必要です。

キー	定義
option.rotate	true	false。ログローテーションが必要な場合に設定します。デフォルト構成は true です。eNcore では古いファイルの削除は行われないことに注意してください。これを行うには、そのスクリプトを別途作成し、スケジュールする必要があります。 例： これを cron ジョブから呼び出します。 #!/bin/bash find /opt/splunk/etc/apps/eStreamer/log/* -mmin +1440 -exec rm {} \;
option.maxLogs	ログのサイズ (行数) を指定します。デフォルト構成は 10,000 です。少数の大きなファイル (50,000 行など) を作成することもできます。

7 トラブルシューティング

7.1 エラー メッセージ

Splunk 用 eNcore アドオンは、ユーザにとってわかりやすいエラーメッセージを提供するように設計されています。エラーメッセージの例を次に示します。

eStreamer サービスが接続を閉じました。考えられる原因は複数あり、この時点までのエラーログに表示されています。 (The eStreamer service has closed the connection - There are a number of possible causes which may show above in the error log.)

エラーがまったく表示されない場合は、次のことが考えられます。

- サーバがシャットダウンしている。
- クライアント認証に失敗した（アウトバウンド IP アドレスが、証明書に関連付けられているものと一致していることを確認してください。デバイスが NAT の対象である場合、証明書 IP はアップストリーム NAT IP と一致している必要があります）。
- サーバに関する問題がある。Firepower Management Center v6.0 を実行している場合は、「Sourcefire 3D Defense Center S3 Hotfix AZ 6.1.0.3-1」のインストールが必要になることがあります。

意味のわからないエラーや詳しい説明が必要なエラーが発生した場合は、サポートまでご連絡ください。問題の修正、およびエラーメッセージの改善を行います。

7.2 eNcore のよくある問題

問題： 以下の情報は、Firepower eNcore for Splunk TA を使用する場合に起こる一般的な問題をすばやく解決するのに役立ちます。報告された多数の問題から、共通のテーマは安定性、接続性、構成に関する問題であることがわかっています。以下のリストでは、このようなシナリオの一部を取り上げ、迅速に解決するための方法を示していますが、問題が引き続き発生する場合は、TAC サポートチケットを作成してください。Microsoft Sentinel Agent のインストールで問題が発生する場合は、次のとおりです。

推奨事項：

Microsoft エージェントを Azure にインストールしてから、OMS を再インストールしてみてください(手順：https://support.microsoft.com/en-us/help/4131455/how-to-reinstall-operations-management-suite-oms-agent-for-linux）。

問題： Splunk にデータが送られてこない。

推奨事項：

- 次の場所にある Splunk TA のデータディレクトリを調べてください。

  $SPLUNK_HOME/etc/apps/TA-eStreamer/bin/encore/data/splunk（デフォルト構成）

- 見つからないレコードを検索してください。一般的な方法は grep です。

  cat "encore*" | grep "rec_type=400"（400 は侵入イベント）

結果が表示されない場合は、証明書の問題であるか、特定のイベントタイプを除外している可能性があります。estreamer.log でエラーやディスク障害を示すメッセージがないか調べてください。

また、inputs.conf ファイルを調べて、モニタが上記のデータディレクトリをポイントしていることを確認してください（デフォルトでは次のようになります）。

Where data is written to

[**monitor**://$SPLUNK_HOME/etc/apps/TA-eStreamer/bin/encore/data]

disabled = 0

source = encore

sourcetype = cisco:estreamer:data

crcSalt = <SOURCE>

問題： データが多すぎるため、clean ユーティリティではデータを適切に消去できない。

推奨事項：

Splunk を使用している場合は、次のように、inputs.conf monitor スタンザを batch に変更できます。これにより、取り込み時にファイルが削除されます。

[**monitor**://$SPLUNK_HOME/etc/apps/TA-eStreamer/bin/encore/data]

上記を下記に変更します。

Where data is written to

[**batch**://$SPLUNK_HOME/etc/apps/TA-eStreamer/bin/encore/data]

disabled = 0

source = encore

sourcetype = cisco:estreamer:data

crcSalt = <SOURCE>

Splunk を再起動して、変更を有効にします。

CLI/CEF Arcsight エディションでは、./encore.sh clean ユーティリティによってある程度の効果を得られますが、これは非常に基本的な機能であるため、オペレーティングシステム、ファイルローテーショ

ン、特定の頻度で消去を行う cron でより堅牢なファイル保持ポリシーを使用する必要があります。または、代わりに Apache Kafka をファイル/ディスク管理に使用できます。

問題： 接続を確立できない。Splunk にデータが送られてこない。

推奨事項：

次のように、接続を確立する手順は複数あります。

- Firepower Management Center からアクセス可能なクライアント証明書を確立します。<トレーニングガイドへのリンク>
- 証明書の配置後、./splencore.sh test または ./encore.sh test ルーチンを実行して、接続が有効になっているかどうかを確認できます。クライアントの状態は背後にあるネットワークに左右されます。一般に、その接続全体が無効になっているか、NAT IP エラーが発生しています。このような場合は、エンドポイントから Firepower Management Center に ping スクリプトを実行して、ネットワーク上で接続がブロックされていないことを確認します。ポート 8042 が eStreamer プロトコル用に開いていることに注意してください。

eNcore のデータ配信に関するもう 1 つの問題は、データがロードバランサに転送されているか、TCP を使用してネットワークで転送されていることです。データをクライアントにローカルに保存し、専用サービスを使用してネットワークで配信することを強くお勧めします。eNcore は、estreamer.conf で次の変数を使用して「永続」モードで動作させることができます。

“alwaysAttemptContinue”: true

ロードバランサによって接続が終了された場合や、ネットワーク上で eNcore が再起動された場合、このモードでは、自動的に Python プロセスを開始して通信を再開します。

問題： 多数の侵入イベントが突然発生し、その後何時間、何日間も何も表示されない。

推奨事項：

クライアントで処理しているイベントの量が非常に少ない可能性があります。デフォルトでは、eNcore は 100 イベントごとにイベントをディスクに書き込む、またはストリーミングするように設定されています。このように、ファイル I/O 操作を限定することでパフォーマンスを最大化するように設計されています。これは、estreamer.conf の次の変数を使用して、構成で調整できます。

“batchSize”: 2.

batchSize を 2 に設定すると、イベントがすぐに配信されますが、パフォーマンス上の負荷が高くなります。逆に、batchSize を大きくするとパフォーマンスは向上しますが、イベントがディスクに書き込まれるタイミングが遅くなります。これは、受信されるイベント数が 1 秒あたり 2,000 件を超える場合に役立つことがあります。この場合、batchSize を 500 にすると、実装環境により適したしきい値となることがあります。

また、送信されるイベントの量が非常に少ない場合にバッチサイズとワーカースレッド数を大きくすると、バッチのしきい値に達するまでデータがメモリ内のキューに保持されるため、データが Splunk に送信されるまでに数日間や数週間かかるなど、遅延の問題が発生する可能性があります。

サーバでサポートできる作業プロセスの数は、プロセッサコアの数と速度、およびサーバの負荷によって決まります。

イベントレート（1 秒あたり）	**ワーカースレッド数**	**バッチサイズ（推奨）**
100 未満	1	2
100 - 2000	1	100（デフォルト）
2000-4000	4	100（デフォルト）
4000-6000	8	250
8000+	12	500

Q：batchSize を 1 に設定する必要がありますか。

A：これは、このドキュメントの以前のバージョンで推奨されていましたが、多大な影響が生じる可能性があります。

batchSize を 1 に設定すると、Firepower Management Center からのすべての情報が即座にディスク/ストリームに書き込まれます。これは環境によっては役立つ場合もありますが、clean や monitor などの他のコマンドが実行されたときにファイルシステムでデッドロックが発生する可能性があります。そのため、書き込みプロセスの間にデータストアで他のアクションを実行できるように、この値を 2 以上に設定することをお勧めします。

問題： 時間の経過とともにイベントデータに欠落が繰り返し見られる。グラフを作成すると、鋸歯状のグラフになる。

推奨事項：

この現象が取り込み分析で観察される場合は、考慮すべき重要な事項がいくつかあります。分析で取り込み時間またはイベント時間が表示されているかどうかを確認してください。Splunk の取り込み、特に複数のデータソースの取り込みは、最も効率的なものに基づいて行われます。Splunk では、アプリケーションの制御外の複数の要因に従って取り込みに優先順位が付けられます。

また、一部のイベントが頻繁に消去されている可能性もあります。データ保持ポリシーを調べ、Firepower Management Center からの負荷にクライアントが対応していることを確認してください。遅延が長くなると、イベントが Splunk によって取り込まれる前に消去される場合があり、結果として、その期間の欠落が生じます。

estreamer.conf データディレクトリ内の raw データファイルを調べ、分析の欠落がある時間範囲内のイベントを検索してください。raw データが存在する場合は、Splunk の取り込みに問題があるか、分析の結果からデータが除外されている可能性があります。

問題： ./encore.sh スクリプトを実行できない。

推奨事項：

これは権限の問題である可能性があります。ログインしているユーザと、Git ダウンロードのファイルおよびディレクトリに対する権限を調べてください。権限の問題がある場合は、sudo を使用して、テストおよび起動スクリプトを実行できます。

次のスクリーンショットは、./encore.sh スクリプトでテストスクリプトを実行しようとしたときの権限エラーの例、および昇格された権限でコマンドの実行方法を変更した結果を示しています。

7.3 よく寄られる質問

データを別のサーバに出力できますか。

はい現在、eNcore ではファイルシステムにのみ書き込みを行いますが、NFS または SMB 共有をマウントし、そのパスを上記のように指定できます。これはパフォーマンスに影響する可能性があります。

複数のインスタンスを実行できますか。

はい。CLI バージョンを使用して実行できます。

ただし、encore.sh シェルスクリプトは、 現在 1 つのインスタンスしかサポートしていません。基盤となる Python プログラムは、一時ファイル（メタデータ、証明書、ブックマークなど）にホストとポートをプレフィックスとして付加します。また、データの衝突を避けるために、アウトプッタの場所（[Splunk] … directory = splunk など）を更新する必要があります。

複数のインスタンスを実行する場合は、eStreamer-eNcore の追加コピーを抽出し、個別に設定して、encore.sh の変更を避けることをお勧めします。

複数の Firepower Management Center に接続できますか。

現在、単一のインスタンスではできません。ただし、上記のように複数のインスタンスを設定できます。

Firepower のレコードタイプにはどのようなものがありますか。

Firepower には 500 を超えるレコードタイプがあります。ここでは、クイックリファレンスの表を示します。詳細なガイドへのリンクは表の下にあります。

レコードタイプ	ブロックタイプ（Block Type）	シリーズ	説明	レコード ステータス	データ形式の参照先…
2	該当なし	該当なし	パケット データ（バージョン 4.8.0.2 以上）	現在（Current）	パケット レコード 4.8.0.2 以上
4	該当なし	該当なし	プライオリティのメタデータ	現在（Current）	プライオリティ レコード

9	20	1	侵入の影響アラート	レガシー	侵入影響アラート データ
9	153	1	侵入の影響アラート	現在 (Current)	侵入の影響アラート データ 5.3 以上
62	該当なし	該当なし	ユーザ メタデータ	現在 (Current)	ユーザ レコード
66	該当なし	該当なし	ルール メッセージのメタデータ (バージョン 4.6.1 以上)	現在 (Current)	4.6.1 以上のルール メッセージのレコード
67	該当なし	該当なし	分類のメタデータ (バージョン 4.6.1 以上)	現在 (Current)	4.6.1 以上の分類 レコード
69	該当なし	該当なし	相関ポリシーのメタデータ (バージョン 4.6.1 以上)	現在 (Current)	相関ポリシー レコード
70	該当なし	該当なし	相関ルールのメタデータ (バージョン 4.6.1 以上)	現在 (Current)	相関ルール レコード
104	該当なし	該当なし	侵入イベント (IPv4) レコード 4.9 ~ 4.10.x	レガシー	製品の旧バージョン
105	該当なし	該当なし	侵入イベント (IPv6) レコード 4.9 ~ 4.10.x	レガシー	製品の旧バージョン
110	4	2	侵入イベント追加データ (バージョン 4.10.0 以上)	現在 (Current)	侵入イベント追加データレコード

111	5	2	侵入イベント追加データのメタデータ (バージョン 4.10.0 以上)	現在 (Current)	侵入イベント追加データのメタデータ
112	128	1	5.1 ~ 5.3.x の相関イベント	レガシー	5.1 ~ 5.3.x の相関イベント
112	156	1	5.4 以上の相関イベント	現在 (Current)	5.4 以上の相関イベント
115	14	2	セキュリティ ゾーン名のメタデータ	現在 (Current)	セキュリティ ゾーン名レコード
116	14	2	インターフェイス名のメタデータ	現在 (Current)	インターフェイス名レコード
117	14	2	アクセス コントロール ポリシー名メタデータ	現在 (Current)	アクセス コントロール ポリシー名のレコード
118	15	2	侵入ポリシー名のメタデータ	現在 (Current)	侵入ポリシー名レコード
119	15	2	アクセス コントロール ルール ID のメタデータ	現在 (Current)	アクセス コントロール ルール ID レコードのメタデータ
120	該当なし	該当なし	アクセス コントロール ルール アクションのメタデータ	現在 (Current)	アクセス コントロール ルール アクション レコード メタデータ

121	該当なし	該当なし	URL カテゴリのメタデータ	現在 (Current)	URL カテゴリ レコード メタデータ
122	該当なし	該当なし	URL レピュテーション メタデータ	現在 (Current)	URL レピュテーション レコード メタデータ
123	該当なし	該当なし	管理対象デバイスのメタデータ	現在	管理対象デバイス レコードのメタデータ
125	該当なし	2	マルウェア イベント レコード (バージョン 5.1.1 以上)	現在 (Current)	マルウェア イベント レコード 5.1.1 以上
125	24	2	マルウェア イベント (バージョン 5.1.1 以上)	現在 (Current)	マルウェア イベント データ ブロック 5.1.1.x
125	33	2	マルウェア イベント (バージョン 5.2.x)	レガシー	マルウェア イベント データ ブロック 5.2.x
125	35	2	マルウェア イベント (バージョン 5.3)	レガシー	マルウェア イベント データ ブロック 5.3
125	44	2	マルウェア イベント (バージョン 5.3.1)	レガシー	マルウェア イベント データ ブロック 5.3.1

125	47	2	マルウェア イベント (バージョン 5.4 以上)	現在	マルウェア イベントのデータ ブロック 5.4 以上
127	14	2	集合型セキュリティ インテリジェンス クラウド名のメタデータ（バージョン 5.1 以上)	現在	集合型セキュリティ インテリジェンス クラウド名のメタデータ
128	該当なし	該当なし	マルウェア イベント タイプのメタデータ（バージョン 5.1 以上)	現在 (Current)	マルウェア イベント タイプのメタデータ
129	該当なし	該当なし	マルウェア イベント サブタイプのメタデータ（バージョン 5.1 以上)	現在 (Current)	マルウェア イベント サブタイプのメタデータ
130	該当なし	該当なし	FireAMP ディテクタタイプのメタデータ (バージョン 5.1 以上)	現在	FireAMP ディテクタタイプのメタデータ
131	該当なし	該当なし	FireAMP ファイルタイプのメタデータ（バージョン 5.1 以上)	現在	FireAMP ファイルタイプのメタデータ
132	該当なし	該当なし	セキュリティ コンテキスト名	現在（Current)	セキュリティ コンテキスト名

207	該当なし	該当なし	侵入イベント(IPv4) レコード 5.0.x ~ 5.1	レガシー	侵入イベント（IPv4）レコード 5.0.x ~ 5.1
208	該当なし	該当なし	侵入イベント(IPv6) レコード 5.0.x ~ 5.1	レガシー	侵入イベント（IPv6）レコード 5.0.x ~ 5.1
260	19	2	ICMP タイプ データのデータ ブロック	現在（Current）	ICMP タイプのデータ ブロック
270	20	2	ICMP コードのデータ ブロック	現在（Current）	ICMP コードのデータ ブロック
400	34	2	侵入イベント レコード 5.2.x	レガシー	侵入イベント レコード 5.2.x
400	41	2	侵入イベント レコード 5.3	レガシー	侵入イベント レコード 5.3
400	54	2	侵入イベント レコード 5.3.1	レガシー	侵入イベント レコード 5.3.1
400	45	2	侵入イベント レコード 5.4 以上	現在	侵入イベント レコード 5.4 以上
500	32	2	ファイル イベント（バージョン 5.2.x)	レガシー	ファイル イベント 5.2.x
500	38	2	ファイル イベント（バージョン 5.3)	レガシー	ファイル イベント 5.3
500	43	2	ファイル イベント（バージョン 5.3.1)	レガシー	ファイル イベント 5.3.1
500	46	2	ファイル イベント（バージョン 5.4 以上)	現在（Current）	ファイルイベント 5.4 以上

502	32	2	ファイル イベント（バージョン 5.2.x)	レガシー	ファイル イベント 5.2.x
502	38	2	ファイル イベント（バージョン 5.3)	レガシー	ファイル イベント 5.3
502	43	2	ファイル イベント（バージョン 5.3.1)	レガシー	ファイル イベント 5.3.1
502	46	2	ファイル イベント（バージョン 5.4 以上)	現在（Current）	ファイルイベント 5.4 以上
該当なし	27	2	5.3 以上のファイル イベント SHA ハッシュ	現在（Current）	5.3 以上のファイル イベント SHA ハッシュ
510	該当なし	該当なし	5.3 以上のファイル タイプ ID のメタデータ	現在（Current）	5.2 以上のルール ドキュメントのデータ ブロック
511	40	2	5.3 以上のファイル イベント SHA ハッシュ	現在（Current）	5.3 以上のファイル イベント SHA ハッシュ
520	28	2	5.2 以上の位置情報のデータ ブロック	現在（Current）	5.2 以上の位置情報のデータ ブロック
530	該当なし	該当なし	ファイル ポリシー名	現在	ファイル ポリシー名
600	該当なし	該当なし	SSL ポリシー名	現在（Current）	SSL ポリシー名
602	該当なし	該当なし	SSL 暗号スイート	現在（Current）	SSL 暗号スイート（SSL Cipher Suite）
604	該当なし	該当なし	SSL バージョン	現在（Current）	SSL バージョン（SSL Version）

605	該当なし	該当なし	SSL サーバ証明書ステータス	現在（Current）	SSL サーバ証明書ステータス
606	該当なし	該当なし	実際の SSL アクション	現在（Current）	[実際のSSLアクション（SSL Actual Action）]
607	該当なし	該当なし	予期された SSL アクション	現在（Current）	[予期されたSSLアクション(SSL Expected Action）]
608	該当なし	該当なし	SSL フロー ステータス	現在（Current）	SSL フロー ステータス
613	該当なし	該当なし	SSL URL カテゴリ	現在（Current）	[SSLURLカテゴリ（SSL URL Category）]
614	50	2	5.4 以上の SSL 証明書の詳細のデータ ブロック	現在（Current）	5.4 以上の SSL 証明書の詳細のデータ ブロック
700	該当なし	該当なし	ネットワーク分析ポリシー レコード	現在（Current）	ネットワーク分析ポリシー レコード

Firepower Event Streamer Integration Guide：

https://www.cisco.com/c/en/us/support/security/defense-center/products-programming-reference-guides-list.html

SIEM コストを低く抑えるために、eNcore でデータの重複を排除できますか。

いいえ。ただし、Splunk の dedup コマンドを調べて、重複するエントリを排除するコマンドを作成することはできます。これは当然ながら、手間がかかり、エラーが発生しやすくなります。

イベントの重複排除に関する Splunk のガイド：
https://docs.splunk.com/Documentation/SCS/current/SearchReference/DedupCommandExamples

HA ペアで eNcore の 2 つのインスタンスを実行できますか。

技術的には、2 つのインスタンスをサイドバイサイドで実行することは可能です。ただし、これらのインスタンスは完全に個別に動作し、出力されるデータが 2 倍になります。したがって、プライマリクライアントの状態および構成データが定期的にセカンダリクライアントにコピーされるホットスタンバイ構成などで実行することをお勧めします。

該当の状態および構成データは次のとおりです。estreamer.conf、x.x.x.x-port_bookmark.dat、x.x.x.x-port_cache.dat、x.x.x.x-port_pkcs.cert、x.x.x.x-port_pkcs.key、x.x.x.x-port_status.dat

ロギングの詳細度を上げることはできますか。

はい。.conf ファイルの logging.level を変更してください。

このレベルを VERBOSE に上げることは可能ですが、パフォーマンスへの影響が大きくなることに注意してください。DEBUG は便利ですが、低速です。実稼働環境での通常の実行では INFO までのレベルにすることを強く推奨します。

8 シスコ サポート

サポートは Cisco TAC によって提供されます。

アプリケーションは無料で使用でき、コミュニティでサポートされています。疑問点などがございましたら、encore-community@cisco.com まで電子メールにてご連絡ください。

9 リンクとリソース

CSTA パートナーの最新リストは次のとおりです。

https://www.cisco.com/c/m/en_us/products/security/technical-alliance-partners.html

9.1 役立つリンク

- Splunkbase：https://splunkbase.splunk.com/
- eNcore TA for Firepower 6.x のお客様：https://splunkbase.splunk.com/app/3662/
- Firepower App for Splunk（2019）：https://splunkbase.splunk.com/app/4388/#/overview

- Firepower 5.4 用の FTA およびダッシュボード （2014）：https://splunkbase.splunk.com/app/1629/
- FTD TA：https://splunkbase.splunk.com/app/3955/
- FTD ダッシュボード：https://splunkbase.splunk.com/app/4010/
- Cisco Security Suite：https://splunkbase.splunk.com/app/525/
- Sourcefire TA for Firepower 5.4：https://splunkbase.splunk.com/app/1808/
- eNcore CLI バージョン：https://github.com/CiscoSecurity/fp-05-firepower-cef-connector-arcsight
- Firepower App for Splunk Overview on SalesConnect：https://www.cisco.com/c/dam/en/us/products/collateral/security/solution-overview-c22-741993.pdf

10 付録

10.1 Firepower Management Center eStreamer クライアント証明書の作成

eStreamer クライアント証明書を生成する手順は次のとおりです。

1 Firepower Management Center の Web インターフェイス (https://fmc-ip-address) に移動し、Firepower Management Center のログイン情報を使用してログインします。

2 Firepower Management Center 6.x の GUI で、[システム（System）] > [統合（Integration）] > [eStreamer（eStreamer）] に移動します。

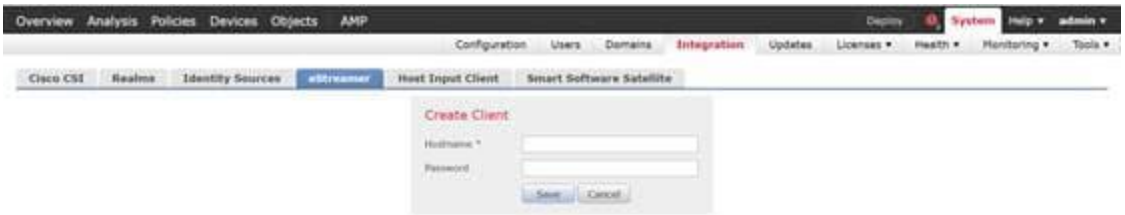
3 [クライアントの作成（Create Client）] をクリックします。

4 ホスト名とパスワードを入力します。

(注) これは、Firepower Management Center からイベントデータを収集するクライアントの IP である必要があります。ここで入力するパスワードは、eStreamer eNcore を初めて実行するときに必要になります。

ここで入力する IP アドレスは、Firepower Management Center の観点における eStreamer-eNcore クライアントの IP アドレスである必要があることに注意してください。つまり、クライアントが NAT デバイスの背後にある場合、IP アドレスはアップストリーム NAT インターフェイスの IP アドレスである必要があります。

5 クライアントのホスト名とパスワードの画面を作成します。



6 [保存（Save）] をクリックします。

7 クライアントの保存画面を作成します。

8 右側にある [ダウンロード（Download）] アイコンをクリックして、PKCS12 ファイルをダウンロードします。

9 ダウンロード画面が表示されます。

10 ターゲットデバイスの目的の場所に PKCS12 ファイルをコピーします。

デフォルトでは、eStreamer-eNcore は /path/eStreamer_eNcore/client.pkcs12 を探します。別のファイル名を使用する場合は、estreamer.conf ファイルを編集する必要があります。

10.2 設定ファイルの例

Splunk 用 eNcore アドオンには、デフォルトの estreamer.conf ファイルが付属しています。参考のために、構成ファイルの例を以下に示します。

```
{
  "connectTimeout":  10,
  "responseTimeout":  10,

  "@startComment": "0 for genesis, 1 for now, 2 for bookmark",
  "start": 2,

  "monitor": {
    "period": 120,
    "bookmark": false,
    "handled": true,
    "details": true
  },

  "logging": {
    "@comment": "Levels include FATAL, ERROR, WARNING, INFO, DEBUG, VERBOSE
and TRACE",
    "level": "INFO",
    "format": "%(asctime)s %(name)-12s %(levelname)-8s %(message)s",
    "stdOut": true,
    "filepath": "estreamer.log"
  },

  "@queueComment": [
    "Maximum number of messages buffered before throttling takes place. The more
powerful",
    "your CPU and more RAM you have, the larger this number can be. It's essentially
a",
    "buffer size. Beyond a certain size you won't see any performance gain and it will",
    "just take longer to stop"
  ],
  "maxQueueSize": 100,

  "subscription": {
    "servers": [
      {
        "host": "1.2.3.4",
        "port": 8302,
        "pkcs12Filepath": "client.pkcs12",
        "@comment": "Valid values are 1.0 and 1.2",
        "tlsVersion": 1.2
      }
    ],

    "records": {
      "@comment": [
        "Just because we subscribe doesn't mean the server is sending. Nor does it
mean",
```

```
        "we are writing the records either. See handler.records[]"
      ],
      "packetData": true,
      "extended": true,
      "metadata": true,
      "eventExtraData": true,
      "impactEventAlerts": true,
      "intrusion": true,
      "archiveTimestamps": true
    }
  },

  "handler": {
    "records": {
      "core": true,
      "metadata": true,
      "flows": true,
      "packets": true,
      "intrusion": true,
      "rua": true,
      "rna": true,

      "@includeComment": "These records will be included regardless of above",
      "include": [],

      "@excludeComment": [
        "These records will be excluded regardless of above (overrides 'include')",
        "e.g. to exclude flow and IPS events use [ 71, 400 ]"
      ],
      "exclude": []
    },

    "@comment": "If you disable all outputters it behaves as a sink",
    "outputters": [
      {
        "name": "Splunk default",
        "adapter": "splunk",
        "enabled": true,
        "stream": {
          "uri": "relfile:///data/splunk/encore.log{0}",
          "options": {
            "rotate": true,
            "maxLogs": 9999
          }
        }
      },
      {
        "name": "JSON",
```

```
            "adapter": "json",
            "enabled": false,
            "stream": {
              "uri": "relfile:///data/json/log{0}.json",
              "options": {
                "rotate": true,
                "maxLogs": 9999
              }
            }
          }
        ]
      }
    }
```

11 免責事項

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報と推奨事項は、すべて正確であると考えていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザ側の責任となります。

対象製品のソフトウェア ライセンスと限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

シスコが導入する TCP ヘッダー圧縮は、カリフォルニア大学バークレー校（UCB）により、UNIX オペレーティング システムの UCB パブリック ドメイン バージョンの一部として開発されたプログラムを適応したものです。

ここに記載されている他のいかなる保証にもよらず、各社のすべてのマニュアルとソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコおよびこれら各社は、商品性の保証、特定目的への準拠の保証、および権利を侵害しないことに関する保証、あるいは取引過程、使用、取引慣行によって発生する保証をはじめとする、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、シスコおよびその供給者は、このマニュアルの使用または使用できないことによって発生する利益の損失やデータの損傷をはじめとする、間接的、派生的、偶発的、あるいは特殊な損害について、あらゆる可能性がシスコまたはその供給者に知らされていても、それらに対する責任を一切負わないものとします。

このマニュアルで使用している IP アドレスと電話番号は、実際のアドレスと電話番号を示すものではありません。マニュアル内の例、コマンド表示出力、ネットワーク トポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

ハード コピーおよびソフト コピーの複製は公式版とみなされません。最新版はオンライン版を参照してください。

シスコは世界各国 200 箇所にオフィスを開設しています。各オフィスの住所、電話番号、FAX 番号は当社の Web サイト（www.cisco.com/go/offices）をご覧ください。