FORCEPOINT

Forcepoint DLP and Azure Sentinel

Integration Guide

Michael Nevin
Mattia Maggioli
23 March 2020
Public

Version	Date	Author	Notes
0.1	31 December 2019	Michael Nevin	First draft
0.2	13 January 2020	Michael Nevin	Update
0.3	21 January 2020	Mattia Maggioli	Review
0.4	30 January 2020	Jonathan Knepher	Review
0.5	23 March 2020	Neelima Rai	Added troubleshooting chapter

Summary

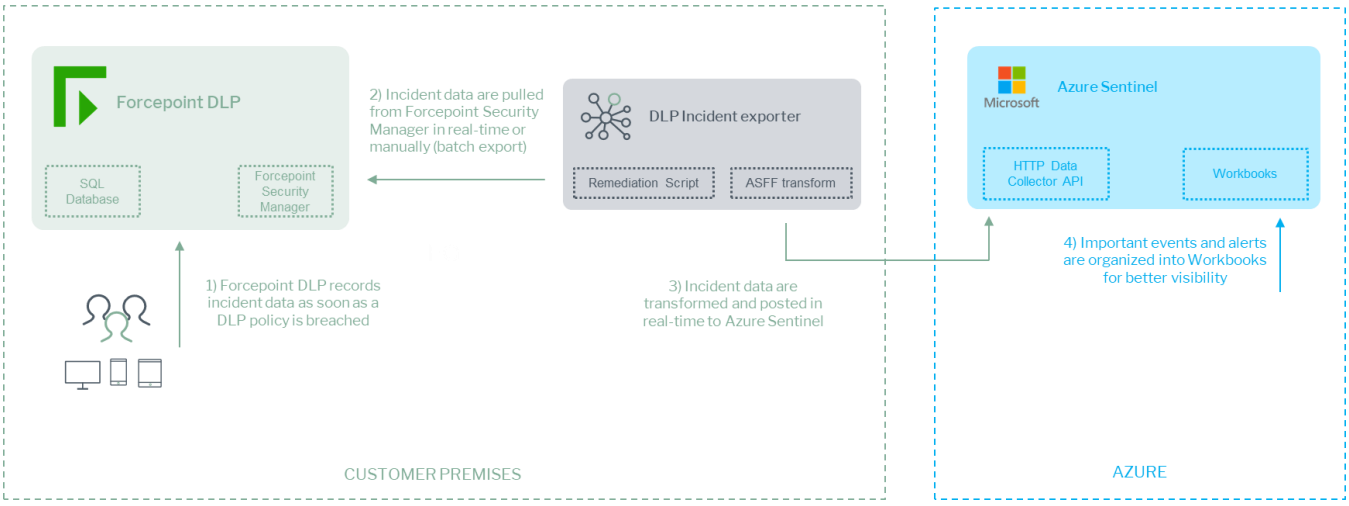
This guide provides step by step instructions to configure Forcepoint DLP and Azure Sentinel to export DLP incidents, transform data, and ingest them into Azure Sentinel.

The code and instructions provided enable system administrators to:

- Export incident data from Forcepoint DLP automatically in real-time
- Transform incident data into the format required by Azure Sentinel
- Ingest the data as custom logs into Azure Sentinel and query events

This interoperability enables customers to use Azure Sentinel for incident data provided by Forcepoint DLP, and to correlate incident events with other Findings from multiple sources including Azure workloads.

A description of the workflow between the components involved in this POC is depicted in this diagram:



Caveats

The integration described in this document is tested with the following product versions:

- Forcepoint DLP with Forcepoint Security Manager 8.5.x
- Azure Monitor with the HTTP Data Collector API (public preview)

Implementation

The solution described in this chapter requires the following files available at this link:

https://frcpnt.com/dlp-sentinel-latest

- fp-dlp-exporter-aws-azure-v1.zip

The archive **fp-dlp-exporter-aws-azure-v1.zip** contains all files necessary to setup and run all the services which enable the integration between Forcepoint DLP and Azure Sentinel:

- **FSM DB connection:** provides real-time export of DLP incidents, extracted from the database used by Forcepoint Security Manager

The solution allows for customizable levels of granularity (High, Medium, and Low severity levels) and performs the transformation and upload tasks, with minimal impact on the underlying storage.

We suggest deploying the solution on the machine which hosts Forcepoint Security Manager, the instructions provided in this document are based on this scenario. The machine hosting the Forcepoint Security Manager will be referenced in the rest of this document with the name "**FSM**".

The following software will be automatically installed by the **install.bat** script provided inside **fp-dlp-exporter-aws-azure-v1.zip**

- Nssm 2.24

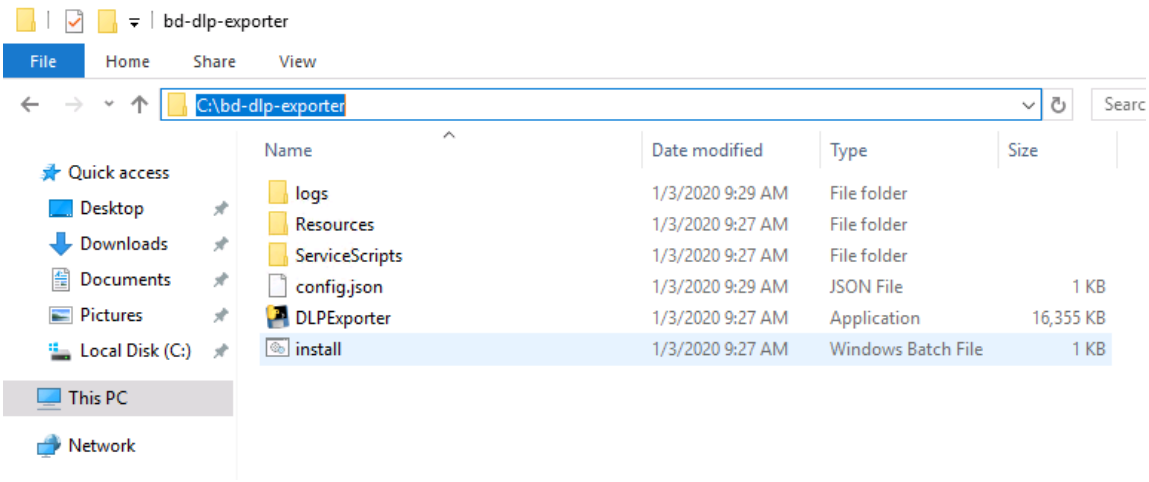
using the following command

START /WAIT powershell -command "[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12; Invoke-WebRequest "https://nssm.cc/release/nssm-2.24.zip" -Method Get -OutFile .\Resources\nssm.zip"

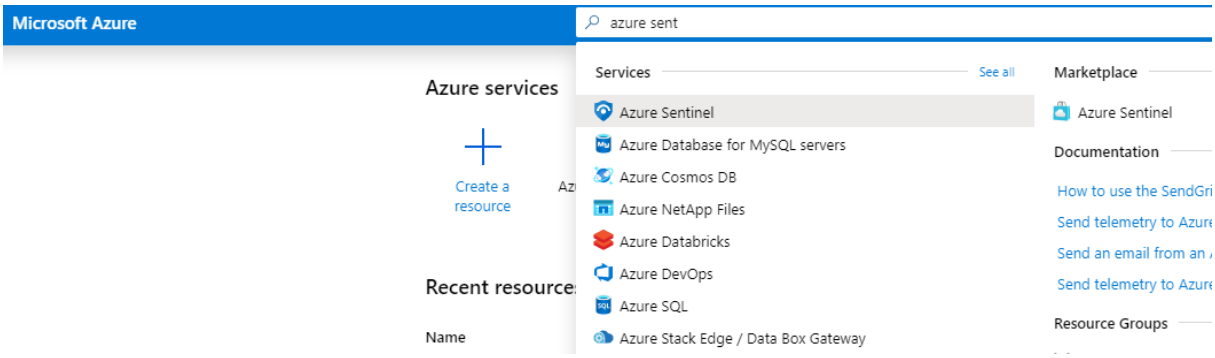
Step 1 – Unpack DLP Incident Exporter and setup Azure Sentinel

Interoperability with Azure Sentinel requires the activation of the service within Azure and obtaining credentials that will be used to send data using the **HTTP Data Collector** API. If both requirements are already satisfied skip to Step 2.

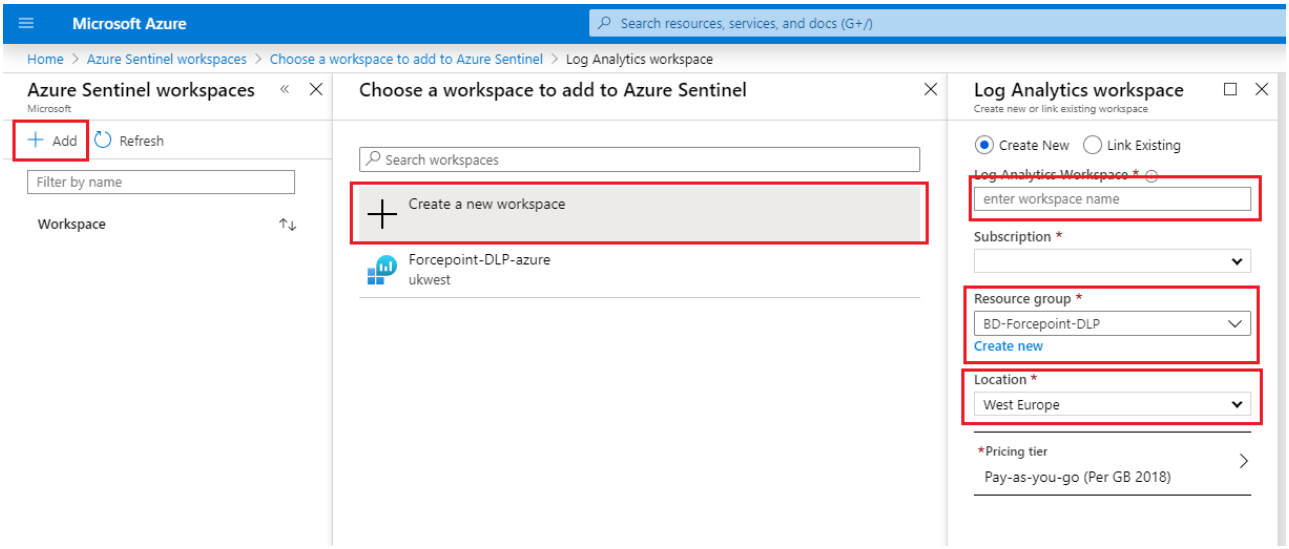
1. Login to the FSM machine and unzip **fp-dlp-exporter-aws-azure-v1.zip** into **C:\fp-dlp-exporter-aws-azure-v1**



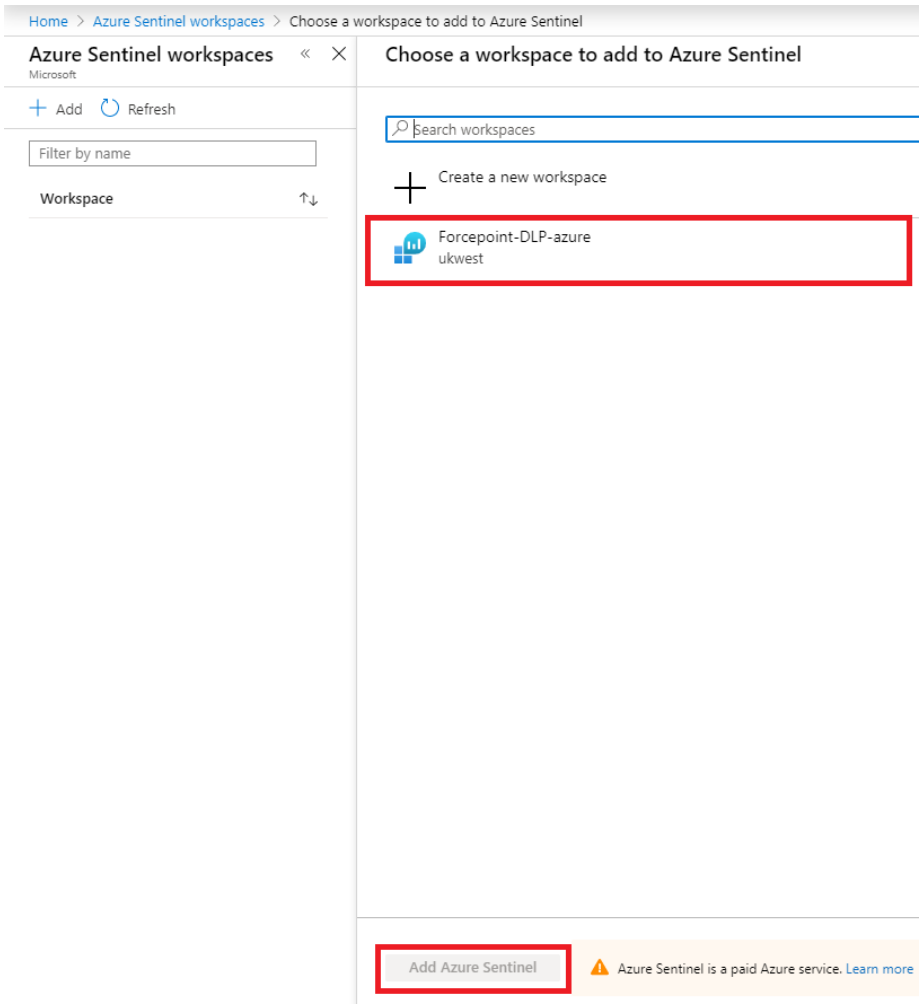
2. Login to your Microsoft Azure portal
3. Using the search bar search for “Azure Sentinel”



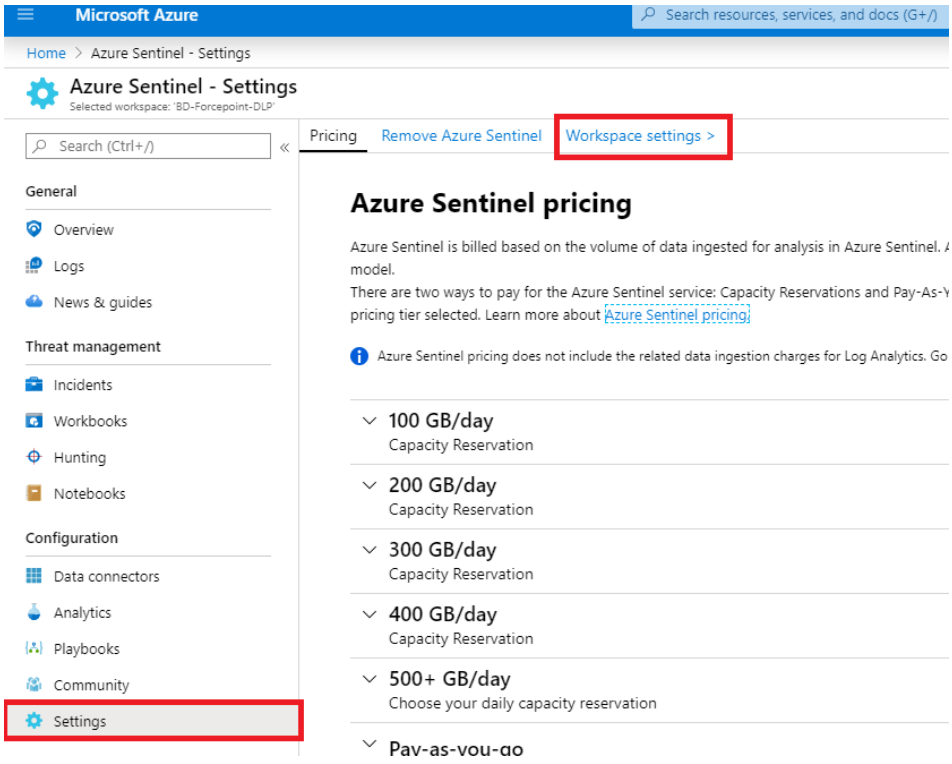
4. From the new window click **Create a new workspace**



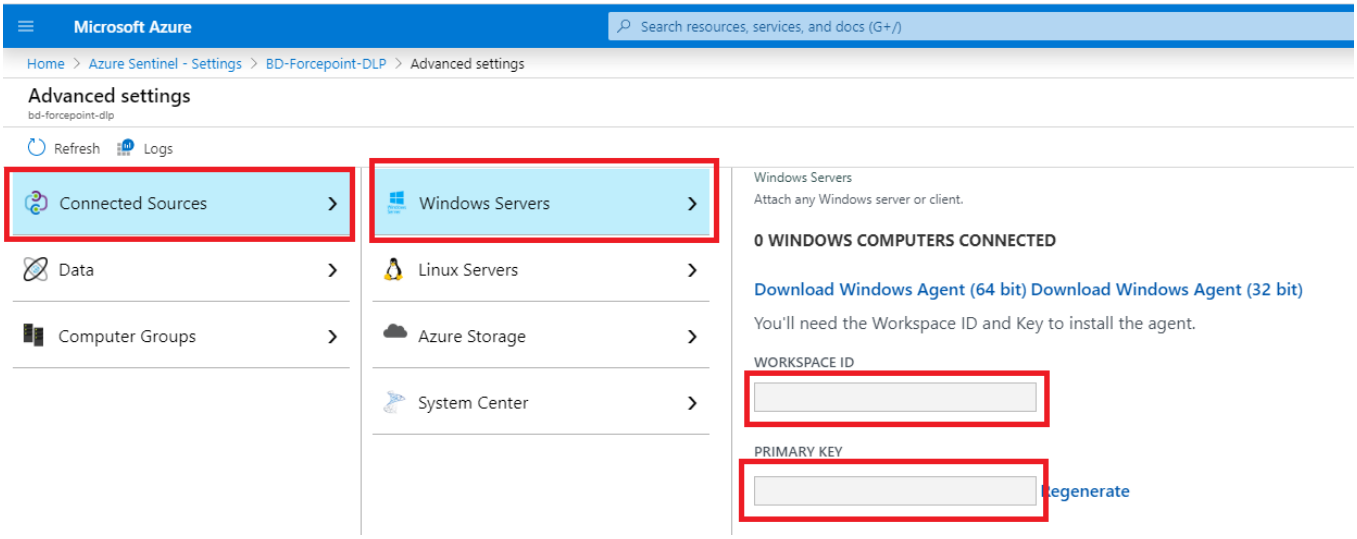
5. Once the new workspace is created, select the workspace and click **Add Azure Sentinel**



6. Select the workspace from the workspace pane. On the next page go to **Configuration** > **Settings** then in the new pane click **Workspace Settings**



7. Go to **Settings** > **Advanced settings** and then **Connected Sources** > **Windows Servers** and store in a secure location the values of **WORKSPACE ID** and **PRIMARY KEY**



Step 2 – Installing the DLP Incident Exporter

1. On the FSM machine navigate to **C:\fp-dlp-exporter-aws-azure-v1**

   Open **config.json** with a text editor edit the settings needed by the **DLP Incident Exporter**.

   Explanation of all settings is in Appendix A of this document.

```
{
  "file_location": "/XMLFileCopy",
  "HIGH": true,
  "MEDIUM": false,
  "LOW": false,
  "Database_Connection": {
    "Server": "sqlserver-hostname",
    "Database": "wbsn-data-security",
    "Trusted_Connection": "yes",
    "UID": "username",
    "PWD": "password"
  },
  "AzureCustomerId": "6d93c191e90efae5cee36d93c19+a6f84246e6c0c212",
  "AzureSharedKey": "6d93c191e90efae5cee36d93c19+a6f84246e6c0c212",
  "LogName": "ForcepointDLPEvents"
}
```

Once **config.json** is edited with all necessary values, double click **install.bat** to run it: the installer will display a few messages as it progresses through the installation steps .

2. The installer will pause at **Creating Service: DLPExporter** and wait for user input:

   - **Please enter your username:** enter the username of an account with administrator access to the FSM machine. Username must be entered according to the format

     DOMAIN\username if using a domain account
     .\username if using a local account

   - **Please enter your administrator password:** enter the password of the account with administrator access

   Once both values are entered the installer will progress until a successful completion.

```
Creating Python Service: DLPSecurityHub
---------------------------
Please enter your username: .\Administrator
Please enter your administrator password:ExamplePassword
Service "DLPSecurityHub" installed successfully!
Set parameter "AppDirectory" for service "DLPSecurityHub".
Set parameter "AppStdout" for service "DLPSecurityHub".
Set parameter "AppStderr" for service "DLPSecurityHub".
Set parameter "ObjectName" for service "DLPSecurityHub".
DLPSecurityHub: START: The operation completed successfully.

PS C:\bd-dlp-aws-master>
```

Once completed, the **DLP Incident Exporter** will run as a service on the FSM machine and DLP incidents will be exported to Azure Sentinel automatically.

Appendix A - Description of config.json settings

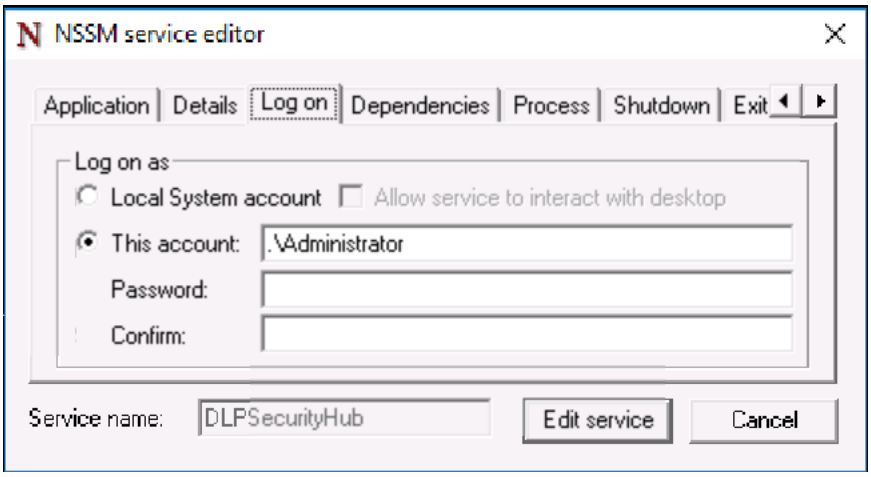
PARAMETER	DESCRIPTION	CHANGE REQUIRED
file_location	Location used by the **DLP Incident Exporter** to store XML files with incident data before upload to Azure. Used when log export is done using the manual method based on remediation script	NO
HIGH **MEDIUM** **LOW**	These parameters allow filtering of DLP incidents, upload only logs whose severity matches the levels set to TRUE.	YES
Database_Connection	These parameters are needed to connect to the SQL database used by Forcepoint Security Manager to store data of DLP incidents. **Server**: hostname or IP address of the SQL database **Database**: name of the database hosting the FSM data **Trusted_Connection**: only “yes” or “no” are possible • yes - if it is a trusted connection • no - if username and password will be used to connect **UID**: username used to login to the database **PWD**: password used to login to the database	YES
AzureCustomerId	Obtained from step 1.1 WORKSPACE ID	YES
AzureSharedKey	Obtained from step 1.1 PRIMARY KEY	YES
LogName	This will be the name of the log that Azure Sentinel will receive from DLP as “custom log”. “_CL” will be appended automatically to the log name once the file is received by Azure Sentinel e.g. “LogName_CL”	YES

Appendix B – Service scripts

The **DLP Incident Exporter** service is managed by the NSSM tool.

Navigate to **C:\fp-dlp-exporter-aws-azure-v1\ServiceScripts**. There are four scripts provided.

PARAMETER	DESCRIPTION
changePassword	This script opens the UI of NSSM to provide an easy way to change or update the password. The password is editable from the **Log on** tab of NSSM (see below)
removeService	This script will remove the **DLPExporter** service from the server and stop it from running
restart	Restarts the **DLPExporter** service
stopService	Stops the **DLPExporter** service (Note this has not removed the service only stopped it from running)



Appendix C – Logs of DLP Incident Exporter

Logs of **DLP Incident Exporter** operations are stored into **C:\fp-dlp-exporter-aws-azure-v1\logs**.

Example message

2020-01-03 09:29:25 - DLPExporter - INFO - Azure is configured on

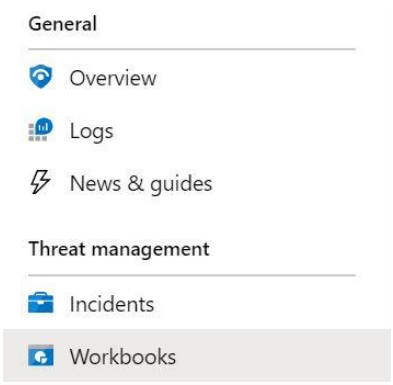
Log structure

Date and time	Service Name	Message Type	message
2019-12-13 17:56:35.055756	**DLPExporter**	INFO DEBUG CRITICAL ERROR WARNING	Azure is configured on

Appendix D – Create a Workbook into Azure Sentinel

Workbooks combine text, Analytics queries, Azure Metrics and parameters into rich interactive reports.

1. Login to Azure Sentinel portal
2. Select **Workbooks** from the left-hand menu, under **Threat management** section. This launches a workbook gallery



3. Click on **Add workbook**, to open a new workbook
4. Click on **Edit**, to edit workbook sections

5. Click **Add query**, to launch a new Log Analytics workspace Logs Query
6. Insert the following query

```
ForcepointDLPEvents_CL
| where TimeGenerated > ago(3d)
| summarize count(RuleName_1_s) by RuleName_1_s, SourceIpV4_s
```

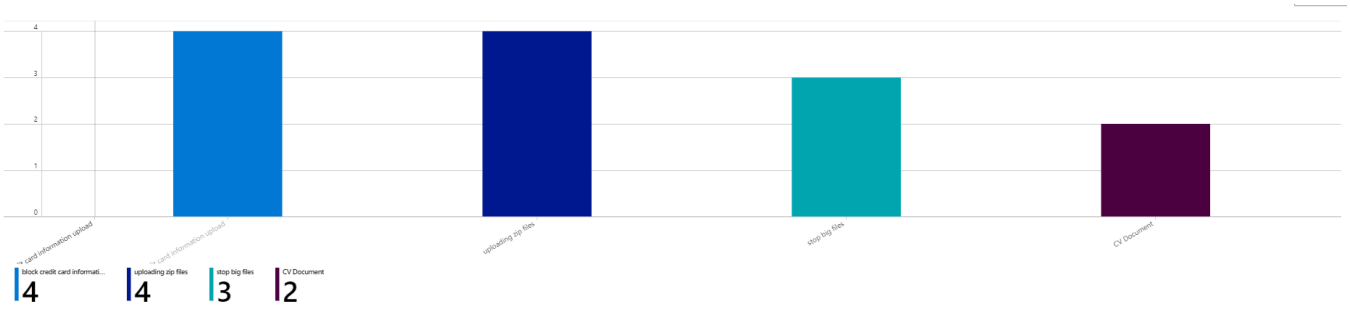
The above query searches for rules triggered in the last three days. The query provides an output similar to this

RuleName_1_s	SourceIpV4_s	count_RuleName_1_s
stop big files	192.168.122.2	3
block credit card information upload	192.168.122.2	4
CV Document	192.168.122.2	2
uploading zip files	192.168.122.2	4

7. Click **Done Editing**
8. Move to the next section of the workbook and click **Edit**
9. Add the following query to display a Bar Chart which provides a visual overview for rules triggered in the last three days

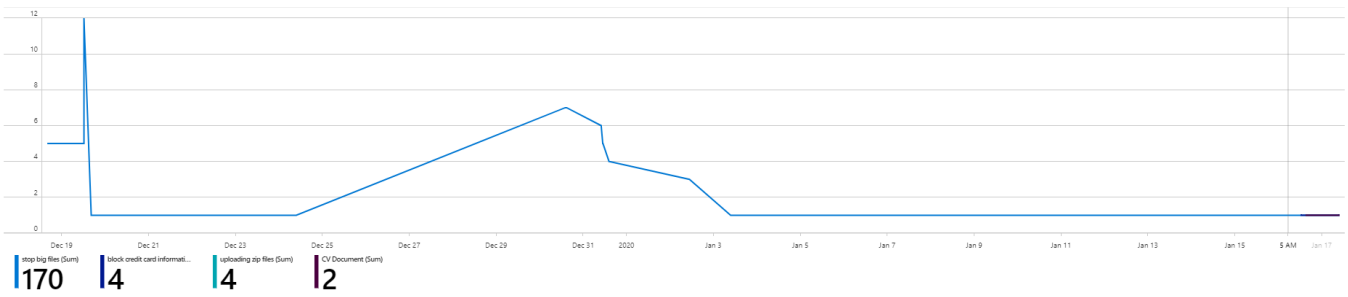
```
ForcepointDLPEvents_CL
| where TimeGenerated > ago(3d)
| summarize count(RuleName_1_s) by RuleName_1_s, SourceIpV4_s
| render barchart
```

10. Click **Done Editing**. The result displayed will be similar to this



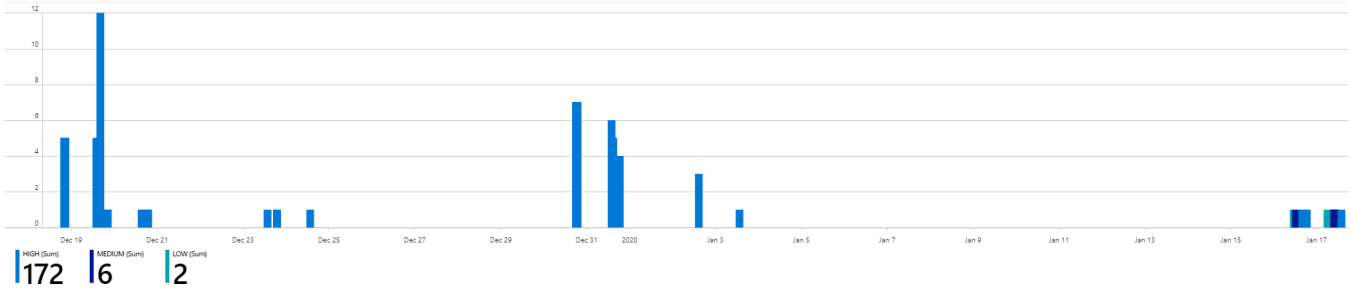
Another query to display rules triggered over time (past 90 days) generated is

```
ForcepointDLPEvents_CL
| where TimeGenerated > ago(90d)
| sort by CreatedAt_t asc nulls last
| summarize count(RuleName_1_s)  by  CreatedAt_t, RuleName_1_s
| render linechart
```



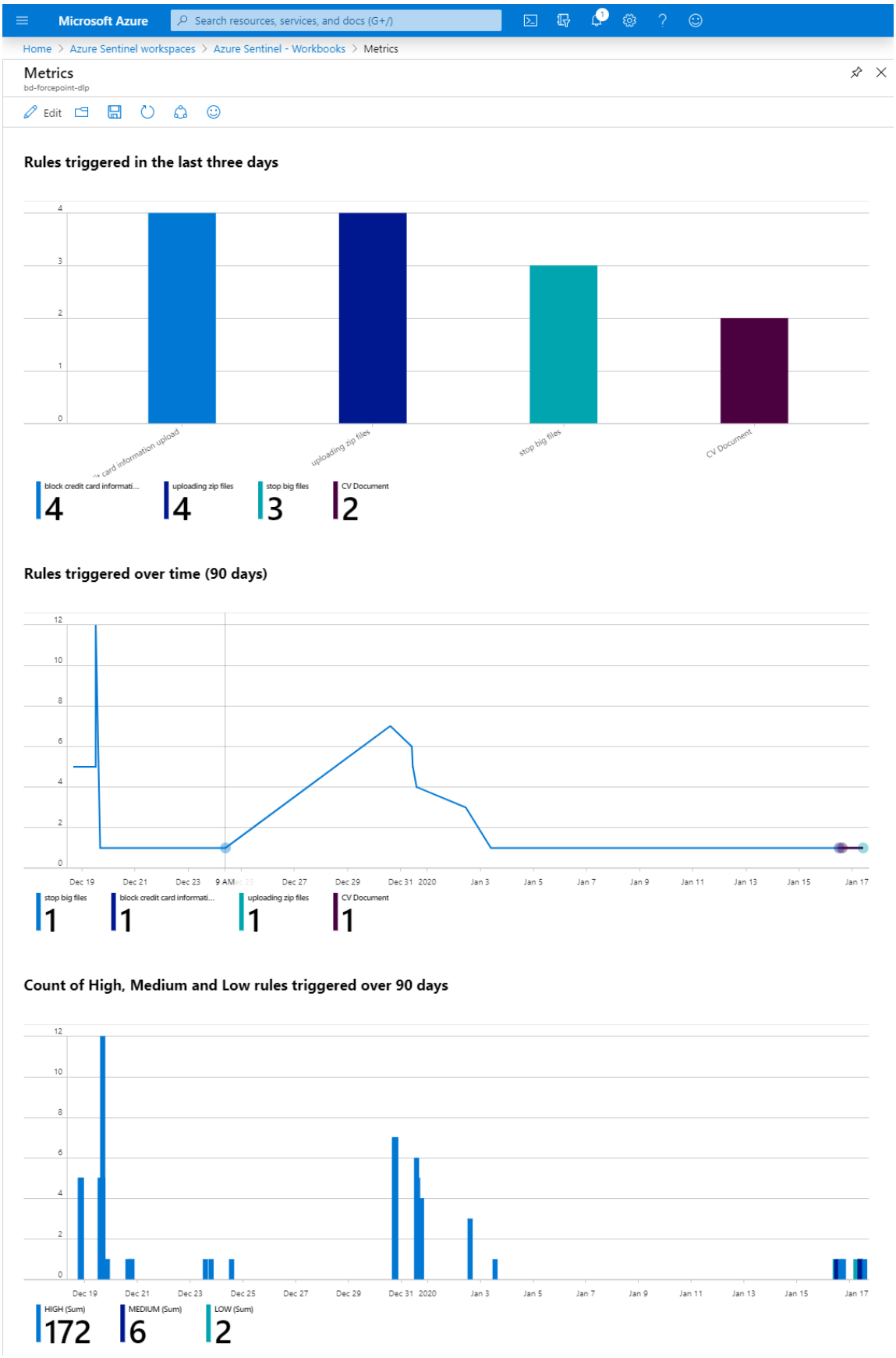
Another query to display counts of High, Medium and Low rules triggered over 90 days

```
ForcepointDLPEvents_CL
| where TimeGenerated > ago(90d)
| sort by CreatedAt_t asc nulls last
| summarize count(Severity_s)  by  CreatedAt_t, Severity_s
| render barchart
```



11. Once finished editing queries click **Done Editing** on the top left corner and on the save icon to save the workbook

Multiple queries can be used to populate a workbook with tables and chart, enabling powerful visualization of events and security related activities obtained from Forcepoint DLP.


Microsoft Azure
Search resources, services, and docs (G+/)
Home > Azure Sentinel workspaces > Azure Sentinel - Workbooks > Metrics
Metrics
bd-forcepoint-dlp
Edit
Rules triggered in the last three days
4
3
2
1
0
uploading zip files
stop big files
CV Document
block credit card informati...
4
uploading zip files
4
stop big files
3
CV Document
2
Rules triggered over time (90 days)
12
10
8
6
4
2
0
Dec 19
Dec 21
Dec 23
Dec 25
Dec 27
Dec 29
Dec 31
2020
Jan 3
Jan 5
Jan 7
Jan 9
Jan 11
Jan 13
Jan 15
Jan 17
stop big files
1
block credit card informati...
1
uploading zip files
1
CV Document
1
Count of High, Medium and Low rules triggered over 90 days
12
10
8
6
4
2
0
Dec 19
Dec 21
Dec 23
Dec 25
Dec 27
Dec 29
Dec 31
2020
Jan 3
Jan 5
Jan 7
Jan 9
Jan 11
Jan 13
Jan 15
Jan 17
HIGH (Sum)
172
MEDIUM (Sum)
6
LOW (Sum)
2

Troubleshooting

Follow these steps to identify issues impacting the normal operation of the integration described in this document.

Validate the prerequisites

Make sure the prerequisites described in the **Summary** chapter are all satisfied:

- Check the versions of Forcepoint DLP with Forcepoint Security Manager and 3rd party products/services in use are listed as compatible

  Forcepoint DLP with Forcepoint Security Manager 8.5.x
  Azure Monitor with the HTTP Data Collector API (public preview)

- Verify the integration component is hosted on a Windows 10 or Windows Server machine
- User must have administrator access to the Windows machine in order to run and complete the installation successfully. Username and password will be requested at the time of install.
- The machine running the **DLPExporter** must have network connectivity to the SQL server
- Check the user has permissions to **Invoke-WebRequest** and **Expand-Archive** in Powershell

Check network connectivity

Make sure firewalls or other security appliances are not impacting the network connectivity necessary for the operation of all components involved into this integration:

- Check the windows machine has network connectivity to AWS:

  The user can check this from the logs created in **C:\fp-dlp-exporter-aws-azure-v1\logs** in the log file named **ForcepointDLPEvents**

  and check the log file has a message similar to below:

  2020-02-28 13:06:06 - DLPExporter - INFO - Azure is configured on

- Check the windows machine has network connectivity to the SQL server:

  The user can check this from the logs created in **C:\fp-dlp-exporter-aws-azure-v1\logs** in the log file named **ForcepointDLPEvents**

  and check the log file has a message similar to below:

  2020-02-28 13:06:06 - DLPExporter - INFO - Database Connection established

Check all components are configured and running properly

Make sure the products and services involved into this integration are configured as expected and they

are running:

- Check SQL connectivity: If you get messages similar to below, that means you either have no SQL connectivity or are entering wrong credentials:

  2020-02-28 13:04:21 - DLPExporter - ERROR - [08001] [Microsoft][ODBC SQL Server Driver][DBNETLIB]SQL Server does not exist or access denied. (17) (SQLDriverConnect); [08001] [Microsoft][ODBC SQL Server Driver][DBNETLIB]ConnectionOpen (Connect()). (53)
  Traceback (most recent call last):
  File "DLPExporter.py", line 135, in <module>
  KeyboardInterrupt
  [18468] Failed to execute script DLPExporter
  2020-02-28 13:09:35 - DLPExporter - ERROR - [28000] [Microsoft][ODBC SQL Server Driver][SQL Server]Login failed for user 'g'. (18456) (SQLDriverConnect); [28000] [Microsoft][ODBC SQL Server Driver][SQL Server]Login failed for user 'g'. (18456)

- In case the user provided wrong credentials for SQL server connection, you can follow the following steps:

  1. Go to **C:\fp-dlp-exporter-aws-azure-v1** and edit the **configs.json** file to add the correct SQL Server connection credentials
  2. Go back to **C:\fp-dlp-exporter-aws-azure-v1\ServiceScripts** and double click on **restart** script. This will restart the **DLPExporter**
  3. Check the **ForcepointDLPEVents** log in **C:\fp-dlp-exporter-aws-azure-v1\logs** and see if the database connection is established.

- The install.bat file should only be run once. If anything goes wrong, you need to go back to the Service scripts to make changes.
- If a wrong password for the administrator account was entered during the first run of the **install.bat** file to install **DLPExporter**, use the following steps to change it:

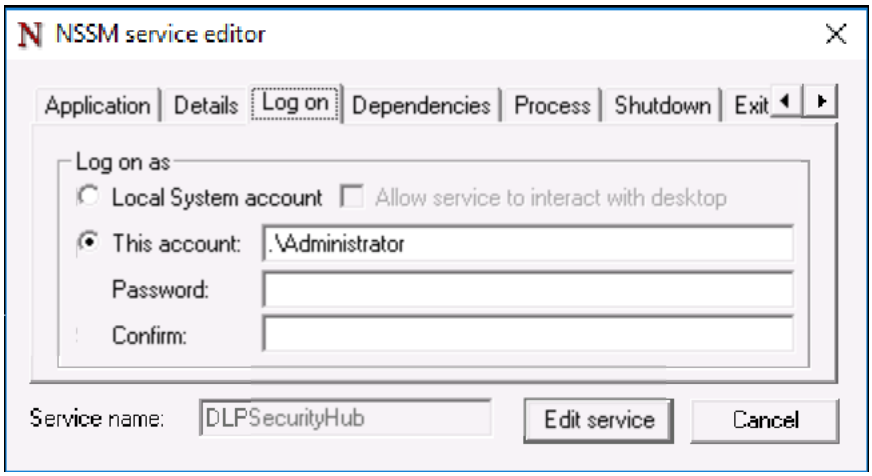
```
Creating required directories
-----------------------------


Creating Service: DLPExporter[0m
-----------------------------
Please enter your username: example
Please enter your administrator password: example
Service "DLPExporter" installed successfully!
Set parameter "AppDirectory" for service "DLPExporter".
Set parameter "AppStdout" for service "DLPExporter".
Set parameter "AppStderr" for service "DLPExporter".
Failed to look up the SID for username example!
LsaLookupNames(): No mapping between account names and security IDs was done.

Failed to look up the SID for username example!
LsaLookupNames(): No mapping between account names and security IDs was done.

Failed to grant the "Log on as a service" right to account example!
Error setting parameter "ObjectName" for service "DLPExporter"!
DLPExporter: START: The operation completed successfully.
Press any key to continue . . .
```

1. Go to **C:\fp-dlp-exporter-aws-azure-v1\ServiceScripts** and double click on **changePassword** script. A window will pop up where the user can enter the correct password



2. Go back to **C:\fp-dlp-exporter-aws-azure-v1\ServiceScripts** and double click on **restart** script. This will restart the **DLPExporter**.

- If the **install.bat** file was run multiple times, the **DLPExporter** service might still be running in the background (even if **removeService** script was run afterwards). Follow the steps below in order to remove the service completely:

1. Open the cmd prompt as administrator.
2. Go to the C:\fp-dlp-exporter-aws-azure-v1\Resources folder
3. Execute the command: nssm
4. Execute the command: nssm stop DLPExporter
5. Execute the command: nssm remove DLPExporter confirm

6. Execute the command: nssm status DLPExporter