

## VMware Carbon Black Cloud (using Azure Functions) connector for Microsoft Sentinel

Using Azure Functions to connect Microsoft Sentinel to your data source	<a href="https://learn.microsoft.com/en-us/azure/sentinel/connect-azure-functions-template?tabs=ARM">https://learn.microsoft.com/en-us/azure/sentinel/connect-azure-functions-template?tabs=ARM</a>
The docs for CB Cloud for Sentinel	<a href="https://learn.microsoft.com/en-us/azure/sentinel/data-connectors/vmware-carbon-black-cloud-using-azure-function">https://learn.microsoft.com/en-us/azure/sentinel/data-connectors/vmware-carbon-black-cloud-using-azure-function</a>
Carbon Black API Notes	<a href="https://developer.carbonblack.com/reference/carbon-black-cloud/authentication/#index-of-base-urls">https://developer.carbonblack.com/reference/carbon-black-cloud/authentication/#index-of-base-urls</a>

Get to know the variables you'll need to configure!	
<b>apId</b>	Carbon Black 'Endpoint Standard' API - This is for collecting the audit logs, but NOT the alerts! see below
<b>apiKey</b>	Carbon Black 'Endpoint Standard' API - key for the above app ID
<b>workspaceID</b>	you know this...
<b>workspaceKey</b>	you know this...
<b>uri</b>	Carbon Black web site: <a href="https://defense-prod05.confirdeploy.net/">https://defense-prod05.confirdeploy.net/</a>
<b>timeInterval</b>	leave this at 5 (minutes)
<b>CarbonBlackOrgKey</b>	8 alphanumeric value of your Carbon Black customer - see below.
<b>CarbonBlackLogTypes</b> (this is set in the ARM template)	Check off the 2 'Alert' boxes and the 'Audit' box (optionally check 'Event')
<b>s3BucketName</b>	in AWS, create this and give it a unique name
<b>EventPrefixFolderName</b>	in AWS, create this and just name it 'Events'
<b>AlertPrefixFolderName</b>	in AWS, create this and just name it 'Alerts'
<b>AWSAccessKeyId</b>	Needed to process the alerts(notifications)
<b>AWSSecretAccessKey</b>	Needed to process the alerts(notifications)
<b>SIEMapId</b> - <b>NOT Optional!!!</b> (even though the Microsoft notes say it is)	Carbon Black 'Endpoint Standard' API - this is ANOTHER api key you need for the alerts! - see below.
<b>SIEMapiKey</b> - <b>NOT Optional!!!</b> (even though the Microsoft notes say it is)	Carbon Black 'Endpoint Standard' API - key for the above app ID

Step #1: Get an AWS Administrator to create the S3 bucket/folders and provide you the 3 'AWS' variables above in blue.	
An AWS S3 Bucket with 2 folders named 'Alerts' and 'Events', along with a user key and secret	For temporarily storing the alert/events being collected from Carbon Black S3 buckets are needed (NOT optional for this connector!)
	AWS > Services > IAM > users > [S3 storage user] > Security Credentials > Access Keys - you can have 2. If you don't know the key's secret you'll have to create a new one.
This is where you get the AWSAccessKeyid and AWSSecretAccessKey:	AWS > Services > Storage > S3 > Create Bucket - leave defaults (block all public access)
This is where you create the S3 bucket and the 2 folders:	Create 2 folders named "Alerts" and "Events"

Step #2: Get the Carbon Black Administrator to create 2 API keys and provide you the customer OrgKey - for the variables above in red	
Creating the 2 API keys with 'API' and 'SIEM' permission types:	Settings > API Access > Access Levels tab. <a href="https://developer.carbonblack.com/reference/carbon-black-cloud/authentication/#index-of-base-urls">https://developer.carbonblack.com/reference/carbon-black-cloud/authentication/#index-of-base-urls</a>
Getting the OrgKey (8 alphanumeric characters)	Get this from the CB web console under Settings > API Access - it's in the top left corner of the screen.
The Carbon Black 'Audit Logs' and 'Notifications' APIs are needed to pull the logs from CarbonBlack. The api keys needed to access these APIs need special access levels. Carbon Black calls these access level names the 'API' and 'SIEM' access level names respectively.	Ask the CB Administrator for 2 API keys. One of the keys should be given the 'API' access level - these are the <b>ApId</b> and <b>ApiKey</b> The other is given the 'SIEM' access level - these are the <b>SIEMapiKey</b> and <b>SIEMapId</b>

Step #3 Deploy the Sentinel Connector for Carbon Black Cloud
In Sentinel, go to Connectors, find the Carbon Black Cloud data connector and deploy it. If you don't see the connector, go to the Content Hub and deploy the Carbon Black Cloud package first.

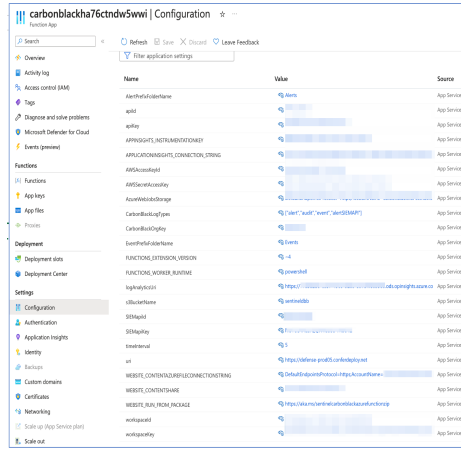
Example Configuration of the Carbon Black ARM Template	
<div><div><div>Log Types</div><div><input type="checkbox"/> Event Supported with AWS S3 Bucket Credentials</div><div><input checked="" type="checkbox"/> Audit Supported with API Credentials</div><div><input checked="" type="checkbox"/> Alert Supported with SIEM API Credentials</div><div><input checked="" type="checkbox"/> Alert Supported with AWS S3 Bucket Credentials</div></div></div> <div><div>API Credentials</div><div>API Key * <input type="text" value="first api key from Carbon Black with 'API' permissions"/></div><div>API Id * <input type="text" value="first api id"/></div><div>Uri * <input type="text" value="https://def-prod05.confirdeploy.net/"/></div><div>SIEM API Credentials</div><div>SIEM API Key * <input type="text" value="second api key from Carbon Black with 'SIEM' permissions"/></div><div>SIEM API Id * <input type="text" value="api id"/></div><div>Time Interval * <input type="text" value="5"/></div><div>AWS S3 Bucket Credentials</div><div>Carbon Black Org Key * <input type="text" value="carbon black Org key"/></div><div>S3 Bucket Name * <input type="text" value="create an S3 bucket and put the name here"/></div><div>AWS Access Key Id * <input type="text" value="Services &gt; IAM &gt; users &gt; [your user] &gt; Security Credentials &gt; Access K..."/></div><div>AWS Secret Access Key * <input type="text" value="secret key from your IAM key above"/></div><div>Alert Prefix Folder Name * <input type="text" value="Alerts (create this folder in your AWS bucket)"/></div></div>	

### Example Variables as seen in the operational Azure Function App

Here is a screenshot of ALL of the variables that should be set inside your Azure Function once the ARM template is complete.

These variables are found here: Azure Functions > [your function app] > configuration

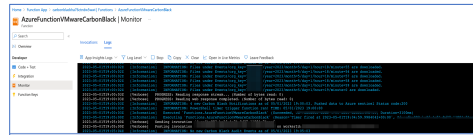
You can also make changes to your function app's variables here



Name	Value	Source
AccountName	AccountName	App Service
AccountKey	AccountKey	App Service
AccountKey2	AccountKey2	App Service
AccountKey3	AccountKey3	App Service
AccountKey4	AccountKey4	App Service
AccountKey5	AccountKey5	App Service
AccountKey6	AccountKey6	App Service
AccountKey7	AccountKey7	App Service
AccountKey8	AccountKey8	App Service
AccountKey9	AccountKey9	App Service
AccountKey10	AccountKey10	App Service
AccountKey11	AccountKey11	App Service
AccountKey12	AccountKey12	App Service
AccountKey13	AccountKey13	App Service
AccountKey14	AccountKey14	App Service
AccountKey15	AccountKey15	App Service
AccountKey16	AccountKey16	App Service
AccountKey17	AccountKey17	App Service
AccountKey18	AccountKey18	App Service
AccountKey19	AccountKey19	App Service
AccountKey20	AccountKey20	App Service
AccountKey21	AccountKey21	App Service
AccountKey22	AccountKey22	App Service
AccountKey23	AccountKey23	App Service
AccountKey24	AccountKey24	App Service
AccountKey25	AccountKey25	App Service
AccountKey26	AccountKey26	App Service
AccountKey27	AccountKey27	App Service
AccountKey28	AccountKey28	App Service
AccountKey29	AccountKey29	App Service
AccountKey30	AccountKey30	App Service
AccountKey31	AccountKey31	App Service
AccountKey32	AccountKey32	App Service
AccountKey33	AccountKey33	App Service
AccountKey34	AccountKey34	App Service
AccountKey35	AccountKey35	App Service
AccountKey36	AccountKey36	App Service
AccountKey37	AccountKey37	App Service
AccountKey38	AccountKey38	App Service
AccountKey39	AccountKey39	App Service
AccountKey40	AccountKey40	App Service
AccountKey41	AccountKey41	App Service
AccountKey42	AccountKey42	App Service
AccountKey43	AccountKey43	App Service
AccountKey44	AccountKey44	App Service
AccountKey45	AccountKey45	App Service
AccountKey46	AccountKey46	App Service
AccountKey47	AccountKey47	App Service
AccountKey48	AccountKey48	App Service
AccountKey49	AccountKey49	App Service
AccountKey50	AccountKey50	App Service
AccountKey51	AccountKey51	App Service
AccountKey52	AccountKey52	App Service
AccountKey53	AccountKey53	App Service
AccountKey54	AccountKey54	App Service
AccountKey55	AccountKey55	App Service
AccountKey56	AccountKey56	App Service
AccountKey57	AccountKey57	App Service
AccountKey58	AccountKey58	App Service
AccountKey59	AccountKey59	App Service
AccountKey60	AccountKey60	App Service
AccountKey61	AccountKey61	App Service
AccountKey62	AccountKey62	App Service
AccountKey63	AccountKey63	App Service
AccountKey64	AccountKey64	App Service
AccountKey65	AccountKey65	App Service
AccountKey66	AccountKey66	App Service
AccountKey67	AccountKey67	App Service
AccountKey68	AccountKey68	App Service
AccountKey69	AccountKey69	App Service
AccountKey70	AccountKey70	App Service
AccountKey71	AccountKey71	App Service
AccountKey72	AccountKey72	App Service
AccountKey73	AccountKey73	App Service
AccountKey74	AccountKey74	App Service
AccountKey75	AccountKey75	App Service
AccountKey76	AccountKey76	App Service
AccountKey77	AccountKey77	App Service
AccountKey78	AccountKey78	App Service
AccountKey79	AccountKey79	App Service
AccountKey80	AccountKey80	App Service
AccountKey81	AccountKey81	App Service
AccountKey82	AccountKey82	App Service
AccountKey83	AccountKey83	App Service
AccountKey84	AccountKey84	App Service
AccountKey85	AccountKey85	App Service
AccountKey86	AccountKey86	App Service
AccountKey87	AccountKey87	App Service
AccountKey88	AccountKey88	App Service
AccountKey89	AccountKey89	App Service
AccountKey90	AccountKey90	App Service
AccountKey91	AccountKey91	App Service
AccountKey92	AccountKey92	App Service
AccountKey93	AccountKey93	App Service
AccountKey94	AccountKey94	App Service
AccountKey95	AccountKey95	App Service
AccountKey96	AccountKey96	App Service
AccountKey97	AccountKey97	App Service
AccountKey98	AccountKey98	App Service
AccountKey99	AccountKey99	App Service
AccountKey100	AccountKey100	App Service

### Troubleshooting Azure Functions (for Sentinel)

Azure Functions > [your function app] > functions > Monitor > Logs  
Here you can see a 'live log tail'. New logs will show up every 5 minutes when the function is triggered.



### Carbon Black API Types - A bit more explanation on the CB 'API' and 'SIEM' Access Levels for your APIs

Sentinel needs to pull from the 'Audit Logs' api and the 'Notifications' apis  
This requires 2 API keys from Carbon Black.  
Carbon Black has pre-configure Access levels for these 2 APIs, named 'API' and 'SIEM'

<https://developer.carbonblack.com/reference/carbon-black-cloud/authentication/#index-of-base-urls>

Endpoint Standard	Carbon Black Product Name	Access Level Name
Audit Logs	/integrationServices/v3/auditlogs	API (Low Response is deprecated) auditlogs
Device Control	/device_control/	Custom (with appropriate permissions) (hostname/device_control/v3org_key)
Enriched Events Search	/investigate/	Custom (with appropriate permissions) (hostname/investigate/v3org_key)
Notifications	/integrationServices/v3/notifications	SIEM (hostname/integrationServices/v3notification)
Recommendation	/recommendation-service/	Custom (with appropriate permissions) (hostname/recommendation-service/v3org_key)

### More information about the Powershell script that makes this Azure Function Work

Azure functions depend on a script to perform the core logic.  
<https://github.com/Azure/Azure-Sentinel/blob/master/Solutions/VMware%20Carbon%20Black/>  
<https://github.com/Azure/Azure-Sentinel/tree/master/Solutions/VMware%20Carbon%20Black/>

This section of the script defines the audit logs being fed to Sentinel:

Notice the specific url used to call the API

apicsecretkey/apid - for the 'audit' logs

SIEMapikey/SIEMapid - for the 'notification' logs - these are the actual CB alerts

```
SentinelHeaders = @{
    "X-Auth-Token" = "$($apiSecretKey)/$($apiId)"
}

AuditLogResult = Invoke-RestMethod -Headers $SentinelHeaders -Uri ($System.Uri::new("$($hostName)/integrationServices/v3/auditlogs"))

SentinelHeaders = @{
    "X-Auth-Token" = "$($SIEMapiKey)/$($SIEMapiID)"
}

NotificationResult = Invoke-RestMethod -Headers $SentinelHeaders -Uri ($System.Uri::new("$($hostName)/integrationServices/v3/notification"))
```

### Some KQL Query Examples

default sample queries are here

<https://learn.microsoft.com/en-us/azure/sentinel/data-connectors/vmware-carbon-black-cloud-using-azure-function>

```
CarbonBlackNotifications_CL
| where threatHunterInfo_responseSeverity_d > 8
| project threatInfo_summary_s, threatInfo_threatCause_actorName_s, threatHunterInfo_threatCause_actor_s,
threatInfo_threatCause_reason_s, threatHunterInfo_threatCause_actorName_s, threatHunterInfo_score_d,
threatHunterInfo_summary_s, threatHunterInfo_responseSeverity_d, threatHunterInfo_targetPriority_s, type_s,
deviceInfo_internalIpAddress_s, deviceInfo_externalIpAddress_s, deviceInfo_groupName_s, deviceInfo_deviceName_s,
deviceInfo_deviceVersion_s, deviceInfo_email_s
rfo_summary_s, threatHunterInfo_targetPriority_s, type_s, deviceInfo_internalIpAddress_s, deviceInfo_externalIpAddress_s,
deviceInfo_groupName_s, deviceInfo_deviceName_s, deviceInfo_deviceVersion_s, deviceInfo_email_s
```

```
CarbonBlackNotifications_CL
| summarize dcount(threatInfo_summary_s) by deviceInfo_deviceName_s
| where dcount_threatInfo_summary_s > 1
| join CarbonBlackNotifications_CL on deviceInfo_deviceName_s
| sort by deviceInfo_deviceName_s
| project deviceInfo_deviceName_s, threatHunterInfo_responseSeverity_d, threatInfo_summary_s,
threatInfo_threatCause_actorName_s, threatHunterInfo_threatCause_actor_s, threatInfo_threatCause_reason_s,
threatHunterInfo_threatCause_actorName_s, threatHunterInfo_score_d, threatHunterInfo_summary_s,
threatHunterInfo_targetPriority_s, type_s, deviceInfo_internalIpAddress_s, deviceInfo_externalIpAddress_s,
deviceInfo_groupName_s, deviceInfo_deviceVersion_s, deviceInfo_email_s
```

```
CarbonBlackNotifications_CL
| where threatHunterInfo_responseSeverity_d != ""
| summarize count() by deviceInfo_deviceName_s
| where count_ > 1
| join CarbonBlackNotifications_CL on deviceInfo_deviceName_s
| sort by deviceInfo_deviceName_s
| project threatHunterInfo_responseSeverity_d, threatInfo_summary_s, threatInfo_threatCause_actorName_s,
threatHunterInfo_threatCause_actor_s, threatInfo_threatCause_reason_s, threatHunterInfo_threatCause_actorName_s,
threatHunterInfo_score_d, threatHunterInfo_summary_s, threatHunterInfo_targetPriority_s, type_s, deviceInfo_internalIpAddress_s, deviceInfo_externalIpAddress_s,
deviceInfo_groupName_s, deviceInfo_deviceVersion_s, deviceInfo_email_s
```

### Testing the Carbon Black API with curl

#### Get Alerts/Notifications

```
curl -H "X-Auth-Token:SIEMapikey/SIEMapiID" "https://defense-prod05.conferdeploy.net/integrationServices/v3/notification"
```

#### Get Audit logs

```
curl -H "X-Auth-Token:Apikey/Apid" "https://defense-prod05.conferdeploy.net/integrationServices/v3/auditlogs"
```