

Microsoft Azure Sensor Pack by AutoMonX



Date	Change	Author			
20.01.2019	Initial Release	AutoMonX			
04.07.2019	Updated to cover the latest features	AutoMonX			
09.08.2019	Updated to cover the latest features, how to get your Azure credentials appendix	AutoMonX			
05.09.2019	Updated troubleshooting, new billing update functionality, Firewall and anti-virus requirements, License handling	AutoMonX			
05.11.2019	Updated new UI functionality	AutoMonX			
06.01.2020	Added the upgrade procedure	AutoMonX			
06.02.2020	Added Microsoft Insights monitoring	AutoMonX			
28.04.2020	Updated UI, Billing sensor updates	AutoMonX			
15.05.2020	Updates of troubleshooting, debugging and upgrade sections	AutoMonX			
26.05.2020	Updated Upgrade procedure, Monitoring Automation Timeout handling, Performance improvements				
08.06.2020	Supported platforms, New debug capabilities	AutoMonX			
12.09.2020	2020 Added support for PRTG native SNMP auto- discovery for Azure Virtual Machines. Support for new Sensor types (Service Plan and Logic App), Support for PRTG Cloud. Refreshed UI. Windows 2008R2 servers are no longer supported, Introduction of Hybrid Sensors				
24.01.2021	Multi-Tenant support, 8 new sensor types, UI Updates, Connection Profiles encryption, Upgrade procedures	AutoMonX			
03.07.2021	App Secret Key sensor added	AutoMonX			
04.08.2021	Azure Kubernetes (AKS) Sensor was added. Our UI now requires PRTG username and password instead of passhash.	AutoMonX			
23.09.2021	Updated security requirements	AutoMonX			



24.03.2022	Azure VM Multi-Disk sensor was added.	AutoMonX		
22.07.2022	New integration with PRTG, New discovery modes, Exclude sensors functionality, auto- deletion functionality	AutoMonX		
13.10.2022	New upgrade mode with installer.	AutoMonX		
22.01.2023	22.01.2023 Multiple granular discovery options, Discovery and Automation can use Azure tags to add relevant resources to PRTG, New CLI options, configuration values, REST API service, SQL Elastic Pools sensor			
24.04.2023	Multiple per subscription sensors introduced, as well as File Share and Host Pools sensors. New monitoring service for the AutoMonX processes.	AutoMonX		
18.10.2023 Added certificates for app registrations, Memory channels added to license sensor, U added test connection button to PRTG, new sensors: Alerts, AKS Upgrade Profiles, Topic Subscriptions, Auto Discovery speed improvements		AutoMonX		
06.11.2023	Added support for 7 new sensor types: Firewalls, Network Interfaces, Connections, Virtual Network Gateways, WAFs, Load Balancers, Network Watchers			
28.07.2024	Added support for sensors: SAML Certificate expiration, Site to Site VPN state. Azure Gov support			



Table of Contents

ICROSOFT AZURE SENSOR PACK OVERVIEW	
	8
IOW DOES IT WORK?	2
THE AZURE SENSOR PACK ARCHITECTURE THE AZURE SENSOR PACK – INTEGRATION WITH PRTG THE AZURE SENSOR PACK – ESSENTIAL TERMINOLOGY	2 2 3
GETTING STARTED WITH AZURE SENSOR PACK	3
SUPPORTED SOFTWARE VERSIONS AZURE SENSOR PACK - PORT REQUIREMENTS AZURE SENSOR PACK - ANTI-VIRUS REQUIREMENTS DOWNLOADING THE AZURE SENSOR PACK INSTALLING THE AZURE SENSOR PACK THE DIRECTORY STRUCTURE OF THE AZURE SENSOR PACK LOOKUP FILE HANDLING (ON-PREM PRTG) LOOKUP FILE HANDLING (PRTG CLOUD) REQUESTING A LICENSE FOR THE AUTOMONX AZURE SENSOR PACK O ACTIVATING THE AZURE SENSOR PACK LICENSE AZURE SENSOR PACK CONFIGURATION	3 4 4 6 10 14 15 18 18 18 20
PROXY SERVER CONNECTION CONFIGURATION CONFIGURING THE AUTOMONX AZURE SENSOR PACK AZURE SENSOR PACK - CONFIGURATION CHECK AZURE SENSOR PACK – UPGRADE INSTRUCTIONS AZURE SENSOR PACK SERVICE – INI CONFIG	21 21 23 25 30
NTRODUCING MULTI-TENANT AZURE MONITORING	33
Multi-Tenant License Types Explained Configuring Multi-Tenant Discovery Encryption of Connection Profile details The Azure Sensor hierarchy in PRTG Migration to Multi-Tenant Version	33 34 35 35 38
AUTO DISCOVERY AND MONITORING AUTOMATION	40
AUTOMATIC DISCOVERY OF AZURE RESOURCES PREVIOUS DISCOVERY RESULTS HANDLING SENSOR TYPES CREATED BY THE AZURE SENSOR PACK SELECTING AZURE SENSORS FOR MONITORING CONFIGURING PRTG GROUP SETTINGS AUTO-DISCOVERY OF AZURE VIRTUAL MACHINES SNMP SENSORS	40 41 42 43 44 46
	INCROSOFT AZURE SENSOR PACK OVERVIEW INCROSOFT AZURE SENSOR PACK ARCHITECTURE THE AZURE SENSOR PACK - INTEGRATION WITH PRTG THE AZURE SENSOR PACK - ESSENTIAL TERMINOLOGY SETTING STARTED WITH AZURE SENSOR PACK SUPPORTED SOFTWARE VERSIONS AZURE SENSOR PACK - PORT REQUIREMENTS AZURE SENSOR PACK - ANTI-VIRUS REQUIREMENTS DOWNLOADING THE AZURE SENSOR PACK INSTALLING THE AZURE SENSOR PACK THE DIRECTORY STRUCTURE OF THE AZURE SENSOR PACK LOOKUP FILE HANDLING (ON-PREM PRTG) LOOKUP FILE HANDLING (ON-PREM PRTG) LOOKUP FILE HANDLING (PRTG CLOUD) REQUESTING A LICENSE FOR THE AUTOMONX AZURE SENSOR PACK ACTIVATING THE AZURE SENSOR PACK LICENSE ZURE SENSOR PACK CONFIGURATION PREPARING FOR CONFIGURING THE AUTOMONX AZURE SENSOR PACK PROXY SERVER CONNECTION CONFIGURATION CONFIGURING THE AUTOMONX AZURE SENSOR PACK AZURE SENSOR PACK - OFIGURATION CONFIGURING THE AUTOMONX AZURE SENSOR PACK AZURE SENSOR PACK - OFIGURATION CHECK AZURE SENSOR PACK SERVICE – INI CONFIG NTRODUCING MULTI-TENANT AZURE MONITORING MULTI-TENANT LICENSE TYPES EXPLAINED CONFIGURING MULTI-TENANT DISCOVERY ENCRYPTION OF CONNECTION PROFILE DETAILS THE AZURE SENSOR HIERARCHY IN PRTG MIGRATION TO MULTI-TENANT VERSION XUTO DISCOVERY AND MONITORING AUTOMATION AUTOMATIC DISCOVERY OF AZURE RESOURCES PREVIOUS DISCOVERY OF AZURE VIRTUAL MACHINES SNMP SENSORS PREPARING FOR AUTO-DISCOVERY OF AZURE VIRTUAL MACHINES

AutoMonX Ltd © 2024 All Rights Reserved Web : http://www.automonx.com

Email : <u>support@automonx.com</u>

=	AutoMonX
	The Monitoring Automation Company

7.8	RUNNING AUTO-DISCOVERY OF AZURE VIRTUAL MACHINES	48
7.9	AUTOMATICALLY ADDING AZURE SENSORS TO PRTG	49
7.10	RESUMING ADDING SENSORS IN CASE OF TIMEOUTS	51
7.11	AUTOMATIC DISCOVERY OF AZURE APPLICATION INSIGHTS	51
7.12	USING CLI TO DISCOVER AZURE RESOURCES	54
7.13	AZURE SENSOR PACK DISCOVERY MODES EXPLAINED	55
7.14	AZURE RESOURCES DISCOVERY – CLI OPTIONS	56
7.15	QUICK DISCOVERY CLI OPTIONS	56
7.16	AZURE RESOURCES DISCOVERY REPORT	57
7.17	AZURE DELTA DISCOVERY REPORT	58
7.18	AZURE CHANNEL LIMITS REPORT	58
7.19	MONITORING AUTOMATION FILES	58
7.20	USING THE MONITORING AUTOMATION CLI	58
7.21	ADDING THE AZURE RESOURCE URL TO PRTG COMMENTS	59
7.22	MULTI-TENANT DISCOVERY SCHEDULER	59
8 SI	JPPORTED SENSOR TYPES	60
8.1	SQL DATABASE	60
8.2	SQL ELASTIC POOLS	60
8.3	Certificate	61
8.4	APP SERVICE PLAN (SERVER FARMS)	61
8.5	WEB SITES	62
8.6	STORAGE ACCOUNTS	62
8.7	CLOUD SERVICES	63
8.8	DATABASE ACCOUNTS	63
8.9	SERVICE BUS NAMESPACE	64
8.10	BATCH ACCOUNTS	64
8.11	Redis	65
8.12	Data Factory	65
8.13	Scheduler	66
8.14	SEARCH SERVICES	66
8.15	Service Bus Queues	67
8.16	Service Health	67
8.17	VIRTUAL MACHINES	68
8.18	AZURE BILLING	69
8.19	AZURE APPLICATION INSIGHTS	69
8.20	AUTOMONX LICENSE	70
8.21	Azure Logic App	70
8.22	Azure Disks	71
8.23	AZURE NOTIFICATION HUBS	71
8.24	AZURE NETWORK INTERFACES	72
8.25	AZURE EXPRESSROUTE CONNECTIONS	72
8.26	AZURE VPN GATEWAYS METRICS	73
8.27	Azure Backup Jobs	73
8.28	AZURE APP SECRET KEY AND/OR CERTIFICATE STATUS	74



8.29	AZURE KUBERNETES SERVICES (AKS)	75
8.30	AKS CLUSTER METRICS	75
8.31	AKS DEPLOYMENTS	75
8.32	AKS INTERNAL DEPLOYMENTS	76
8.33	AZURE VM MULTI-DISK	76
8.34	AZURE STORAGE FILE SHARE	76
8.35	AZURE SERVICE ISSUES	77
8.36	Azure Planned Maintenance	77
8.37	AZURE HEALTH ADVISORIES	78
8.38	Azure Advisor	78
8.39	Azure Quotas	79
8.40	Azure Host Pool	79
8.41	AZURE RESERVATIONS	80
8.42	AZURE ALERTS	80
8.43	AKS UPGRADE PROFILES	81
8.44	AZURE SERVICE BUS TOPIC SUBSCRIPTIONS	81
8.45	Azure Firewalls	82
8.46	AZURE APPLICATION GATEWAY (WAF)	82
8.47	AZURE (VPN) CONNECTIONS	83
8.48	IP ADDRESS	84
8.49	Azure Load Balancer	84
8.50	Azure Network Watcher	85
8.51	AZURE LOG ANALYTICS CUSTOM SENSOR	85
8.52	AZURE KUBERNETES COMBINED SENSOR	85
8.53	AZURE SHAREPOINT STATISTICS	86
8.54	AZURE ENTERPRISE SECRETS AND CERTIFICATES	87
8.55	AZURE SAML ENTERPRISE CERTIFICATES	87
9 TI	ROUBLESHOOTING	88
9.1	TROUBLESHOOTING THE AZURE SENSOR PACK INSTALLATION	88
9.2	TROUBLESHOOTING THE AZURE SENSOR CONFIGURATION	90
9.3	TROUBLESHOOTING AZURE DISCOVERY CONNECTION ERRORS	92
9.4	TROUBLESHOOTING AZURE DISCOVERY - PERMISSIONS	92
9.5	COLLECTING THE DISCOVERY FILES FOR AUTOMONX SUPPORT	92
9.6	TROUBLESHOOTING THE DISCOVERY OF AZURE METRICS	93
9.7	TROUBLESHOOTING THE DISCOVERY OF AZURE SERVICE HEALTH	93
9.8	COLLECTING AZURE SERVICE DEBUG INFORMATION	94
9.9	COLLECTING AZURE SENSOR DEBUG INFORMATION	94
9.10	COLLECTING IN-DEPTH AZURE SENSOR DEBUG INFORMATION	95
10 C	OMMAND LINE OPTIONS (CLI)	96
10.1	THE AZURE SENSOR PACK COMMAND LINE OPTIONS REFERENCE	96
10.2	FULLY AUTOMATED AZURE MONITORING	97
1	0.2.1 Automated Discovery and Monitoring	97
1	0.2.2 Automated Clean-Up of Un-Needed Resources from Monitoring	

Web : http://www.automonx.com Email : <u>support@automonx.com</u>



1	0.2.3 Automatically Pausing Un-Needed Resources	100
1	0.2.4 Automated Inclusion/Exclusion of Sensors and Channels	101
1	0.2.5 Automated Scan-Now Functionality	105
1	0.2.6 Automated Addition and Removal of Tenants	106
11 RI	EST API SERVICE	107
11.1	FUNCTION GET_TENANT_ID	107
11.2	FUNCTION INITIATE DISCOVERY VIA API	108
11.3	FUNCTION ADD_TENANT	108
11.4	FUNCTION REMOVE_TENANT	109
11.5	FUNCTION TO CHECK AUTO DISCOVERY STATUS	109
11.6	FUNCTION ADD_TO_PRTG	110
11.7	FUNCTION DISCOVER_AND_ADD	110
11.8	FUNCTION UPDATE_KEY	111
11.9	ALLOWING REMOTE CONNECTION TO API SERVER	111
12 M	ONITORING THE AUTOMONX AZURE SENSOR PACK PROCESSES	112
13 AI	PPENDIX A - OBTAINING THE AZURE APPLICATION AND TENANT IDS	113
13.1	RETRIEVING THE AZURE SECRET KEY	117
14 AI	PPENDIX B - ADDING PERMISSIONS	119
14.1	ENABLING APP SECRET KEYS FOR MONITORING	119
14.2	ENABLING RESERVATIONS FOR MONITORING	120
14.3	REGISTERING RESOURCE PROVIDERS	121
15 AI	PPENDIX C - ACTIVATING LOG ANALYTICS ON A VM	122
16 AI	PPENDIX D – CREATING CUSTOM LOG ANALYTICS SENSORS	123



1 Purpose

The purpose of this document is to provide a detailed explanation of the AutoMonX Microsoft Azure Sensor pack and how to deploy it.

2 Microsoft Azure Sensor Pack Overview

AutoMonX has developed the Microsoft Azure Sensor pack aimed at monitoring Microsoft Azure cloud environments for IT teams and service providers (MSPs and CSPs). The Azure Sensor pack can discover and monitor Azure resources located across multiple Azure tenants and multiple subscriptions. These unique sensors are monitoring the various aspects of Microsoft Azure's resources and services and have a tight integration with PRTG. The Azure Sensor pack currently supports auto-discovery and monitoring of 52 Azure resource types as seen below:

- Azure Kubernetes Deployments
- Azure Kubernetes Internal
 Deployments
- VM Multi-Disk
- SQL Elastic Pools
- Storage File Share
- Service Issues
- Planned Maintenance
- Health Advisories
- Azure Advisor
- Quotas
- Host Pool
- Reservations
- Firewalls
- Application Gateway (WAF)
- (VPN) Connections
- IP Address
- Load Balancer
- Network Watcher
- Alerts
- AKS Upgrade Profiles
- Service Bus Topic Subscriptions
- Custom KQL Log Analytics queries NEW
- SharePoint NEW
- Enterprise App Secrets NEW
- Enterprise SAML Certificates NEW

- SQL Databases (SAAS)
- Certificates
- App Service Plan (Server Farms)
- Web Sites
- Storage Accounts
- Cloud Services
- Database Accounts
- Service Bus Namespace
- Batch Accounts
- Redis
- Data Factory
- Scheduler
- Search Services
- Service Bus Queues
- Service Health
- Virtual Machines
- Azure Billing
- Application Insights
- Logic App
- Disks
- Notification Hubs
- Network Interfaces
- ExpressRoute Connections
- Virtual Network Gateways
- Backup Jobs
- App Secret Keys & Certificates
- Azure Kubernetes Cluster Metrics
- AutoMonX Ltd © 2024 All Rights Reserved

Web : http://www.automonx.com Email : support@automonx.com



The Azure Sensor pack supports auto-discovery and monitoring of many additional Azure resource types that their monitoring metrics are enabled via the generic API access.

3 How Does It Work?

The AutoMonX Azure Sensor Pack connects via REST API to the Microsoft Azure management environment and collects metrics, values and additional information. It reports back to the Azure Sensor application server the gathered data. The Azure Sensor Pack is tightly integrated with PRTG and provides metrics and custom error limits in a form that is understandable by PRTG. In order to integrate with the Azure Sensor pack, our Monitoring Automation autoconfigures PRTG by deploying HTTP Data Advanced sensors. These sensors connect to the Azure Sensor pack application server. As a result, PRTG displays the information gathered from Microsoft Azure by the Azure Sensor pack application server in a readable and clear way as seen in the picture below.

The EXE/Script Advanced sensor is the legacy option to integrate with PRTG and it also supported for backward compatibility.



🖌 Sensor Azure App Metrics 🕷 ★★★★★

Overview	Live Data 2 day	s 30 days	365 days Historic D	lata Log 🎤 Se	ttings Z	Notifications					
Last Message	8										
ок											
Last Scan: 6 m 6 s	Last Up: 6 m 6 s	L 1	ast Down: 4 d	Uptime: 95.9235%	Downtir 4.07659	ne: 6	Coverage: 98%	Sensor Type: EXE/Script Adva	nced		
Total da	itabase size			Blocked by Firew 0 #		CPU percentag 0 #	e 💦	Data IO percenta 0 #			
				Database size pe	\bigcirc	Deadlocks	\sim	DTU Limit			
12.648.4	148 Bvte	0	12.910.592 By	DTU percentage	\frown	DTU used	\bigcirc	Failed Connections	\bigcirc		
In-Memory 0 #	y OLTP	Log IO pe	ercentage	Sessions percent	\bigcirc	Successful Con	ne	Workers percent	\bigcirc		
Channel 🔻				ID Last Value		Minimum		Maximum		S	ettino
Blocked by F	irewall			9		0 #		0 #		0 #	*
CPU percent	age			13		0 #		0 #	0.0	07 #	*
Data IO pero	centage			10		0 #		0 #		0 #	\$
Database siz	ze percentage			5		0 %		0 %		0 %	*
Deadlocks				11		0 %		0 %		0 %	*
Downtime				-4							*
DTU Limit				3		10 #		0 #	1	10 #	*
DTU percent	tage			7		0 %		0 %		0 %	\$

3.1 The Azure Sensor Pack Architecture

The AutoMonX Azure Sensor Pack application server sends multiple requests to the Microsoft Azure Management API and therefore needs a managing service to efficiently handle the requests while minimizing the PRTG probe load. The managing service harnesses the advantages of threading technology to efficiently queue the sensor requests to the Microsoft Azure Management API in order to provide reliable, swift, and lightweight performance. The Azure Sensor Pack is highly scalable and flexible thus can monitor thousands of resources spread across multiple Azure Tenants, subscriptions and regions.

3.2 The Azure Sensor Pack – Integration with PRTG

The AutoMonX Azure Sensor Pack is tightly integrated with PRTG via two sensor types: the HTTP Advanced sensor and the custom EXE/XML sensor (for backward compatibility). Starting with version 4.2.x, HTTP Advanced sensor is the recommended (and the default) way to integrate PRTG and the Azure Sensor pack. Such integration is automatically created by the Monitoring Automation feature of the Azure Sensor Pack that pushes all the required configuration settings into PRTG.



3.3 The Azure Sensor Pack – Essential Terminology

The Azure Sensor Pack automatically adapts PRTG web interface for displaying Microsoft Azure resources. Below are some essential terms that are used through this deployment guide:

<u>**Group**</u> – PRTG group of devices. The Azure Sensor pack monitoring automation automatically organizes the Azure resources it discovers by creating groups of Azure resources (SQL, WebFarm, Network Interface etc) Read more about the automatically created hierarchy in <u>section 6.4</u>.

Device – Each Azure resource is represented in PRTG as a device (for example a Virtual Machine, an SQL database, or a Network Interface).

<u>Sensor</u> – Created under every PRTG device. The main sensors that are available for most Azure resource types are Azure App Metrics and Azure Service Health. Additional sensor types are available, read more about it in <u>section 7.3</u>.

<u>Channel</u> – The PRTG sensor channels (App Metrics) represent a single Azure resource performance metric.

<u>Azure Tags</u> – These are Labels in the Microsoft Azure Portal for logical grouping of resources. The AutoMonX Azure Sensor Pack can utilize these tags to include or exclude the desired resources. Read more in <u>section 10.2.4</u>.

4 Getting Started with Azure Sensor Pack

4.1 Supported Software versions

The Azure Sensor pack has been tested to support the following software:

Software Type	Versions	Comments
Windows OS	2012R2, 2016, 2019	Standard and Enterprise editions
Virtual Infrastructure	VMWare Cloud or on-prem, Azure VM, AWS VM	
PRTG Core and Probe deployments	20.x, 21.x, 22.x, 23.x, 24.x	All On-Prem PRTG license types supported
PRTG Hosted (Cloud)	Supported with Azure Sensor pack 3.31 and higher	



4.2 Azure Sensor Pack - Port requirements

The AutoMonX Azure Sensor Pack requires the following ports to be open for it to function correctly. Please make sure that the local firewall / anti-virus and the external firewalls are configured as required to allow the Sensor Pack to function correctly.

Port / URL	Purpose	Direction
https://management.azure.com https://login.windows.net https://login.microsoftonline.com https://management.core.windows.net https://graph.microsoft.com https://api.applicationinsights.io https://api.loganalytics.io	Azure API connection	From PRTG Probe to Azure
TCP 443, 80	Connect to Azure, PRTG API	From PRTG Probe to Azure and PRTG Core
SNMP (UDP 161), ICMP	For monitoring Azure- based Virtual Machines	From PRTG Probe to Azure VMs
TCP 8148 TCP 8092 TCP 8075	Internal service ports. Make sure these ports are not occupied by other programs on the server.	No need to open FW rules.

4.3 Azure Sensor Pack – Anti-Virus Requirements

The AutoMonX Azure Sensor Pack initiates many processes and threads during its normal execution. Configure your anti-virus and/or anti-malware software to exclude the AutoMonX directory in *<drive>:\Program Files (x86)\PRTG Network Monitor\Custom Sensors\EXEXML* from on-access scanning. This would greatly improve the general performance of the Azure Sensor Pack.

4.4 Downloading the Azure Sensor pack

Obtain the software by downloading it from the AutoMonX web site at http://www.automonx.com/downloads



Production Releases

Product Name		Updated On	Download	Comments	MD5 Fingerprint
Cisco ACI Monitoring Pack	2.2.7	11.10.2023	Download		1c01f1d38e936b5fa864c7b0d82f9c9(
3PAR/Primera Sensor Pack	2.7.4.6.1	28.12.2021	Download		a3a9aaa4ffff6034c9e95bf8d4e4da50
Azure Sensor Pack	4.3.10	28.08.2023	Download	New License for versions prior to v4.0.26	5e61415d244fab6aabc211fe8515cc6
Linux Sensor Pack	3.1	15.12.2021	Download		1a26d016bbb283bef70d1fe9141236e

The Azure Sensor Pack is optimally deployed via an installer (strongly recommended).

Product Name		Updated on	Download	Comments	MD5 Fingerprint
Azure Sensor Pack	4.2.14 (Preview)	14.01.2023	Download	Preview Release!	e35453393fe205b790694a45f083adb4
Azure Sensor Pack	4.2.8 (Stable)	23.10.2022	Download	Upgrade supported from v4.0.28 and higher	21e8f673711df30460f5307967a7c7ac
Linux Sensor Pack	3.1	15.12.2021	Download	New installations Only!	971393ec712168a9225acf446ad4474f



4.5 Installing the Azure Sensor Pack

Download the latest Azure Sensor Installer from

https://www.automonx.com/downloads

Start the installer and follow the instructions.



Configure the PRTG connection information. Make sure to mark "Enable HTTPS" if relevant.



p - AutoMonX Azure Sensor Pack	- [
RTG Web Credentials	
This information is critical for the immediate success of this i	installation
User Name:	
prigaumin	
Password:	
•••••	
IP:	
127.0.0.1	
Port:	
443	
✓ Enable HTTPs	

	Back	Next	Cancel
Ohanna if was wish to install the additional Davi			
Choose if you wish to install the additional Back	cend Servic	ce (Monitor	rs the

Sensor Pack and provides API capability).

AutoMonX The Monitoring Automation Company

 Setup - AutoMonX Azure Sensor Pack Additional Monitoring Features You can enable additional monitoring features from the list below AutoMonX Azure Sensor Pack Automonx Azure Backend Service 	,		
	Back	Next	Cancel
Setup - AutoMonX Azure Sensor Pack Select Additional Tasks Which additional tasks should be performed?			
Select the additional tasks you would like Setup to perform while ins click Next. Additional shortcuts: Create desktop icon to the UI for AutoMonX Azure Sensor Pack	stalling AutoMonX	(Azure Sensor Pa	ιck, then
	Back	Next	Cancel

AutoMonX Ltd $\ensuremath{\textcircled{C}}$ 2024 All Rights Reserved

Web : http://www.automonx.com Email : <u>support@automonx.com</u>



Setup - AutoMonX Azure Sensor Pack Ready to Install		_
Setup is now ready to begin installing AutoMonX Azure	Sensor Pack on your computer.	- ()
Click Install to continue with the installation.		
	Back Install	Cancel
Setup - AutoMonX Azure Sensor Pack		
Installing Please wait while Setup installs AutoMonX Azure Senso	or Pack on your computer.	
Installing Please wait while Setup installs AutoMonX Azure Senso Uninstalling existing services	or Pack on your computer.	
Installing Please wait while Setup installs AutoMonX Azure Senso Uninstalling existing services	or Pack on your computer.	
Installing Please wait while Setup installs AutoMonX Azure Senso Uninstalling existing services	or Pack on your computer.	
Installing Please wait while Setup installs AutoMonX Azure Senso Uninstalling existing services	or Pack on your computer.	
Installing Please wait while Setup installs AutoMonX Azure Senso Uninstalling existing services	or Pack on your computer.	
Installing Please wait while Setup installs AutoMonX Azure Senso Uninstalling existing services	or Pack on your computer.	
Installing Please wait while Setup installs AutoMonX Azure Senso Uninstalling existing services	or Pack on your computer.	
Installing Please wait while Setup installs AutoMonX Azure Senso Uninstalling existing services	r Pack on your computer.	
Installing Please wait while Setup installs AutoMonX Azure Senso Uninstalling existing services	r Pack on your computer.	

AutoMonX The Monitoring Automation Company

O Setup - A	utoMonX Azure Sensor Pack			– 🗆 X
Auto	IonX Azure Sensor Pack installation	n conclusion		\bigcirc
Th	ank you for installing the AutoMonX Az	ure Sensor Pack		
Se	nsor pack files copied	¥		
Se	rvice installation	×		
Re	load Lookup Files in PRTG	×		
De	tails required for evaluation license	request:		
The The The	e host name for the machine is: 'Azure e IP address for the machine is: '10.3 e MAC address for this machine is: '60	.Test'. 0.16' 0-45-BD-82-7C-65'.		
Co Or	ntact us to get the evaluation license mail sales@automonx.com.	<i>i</i> a automonx.com		
			Next	

If an error occurred while updating the Lookup files, update them manually <u>as</u> <u>explained in section 4.6.</u>

4.6 The Directory Structure of the Azure Sensor Pack

The installer will extract all the Azure Sensor pack files and sub directories to the following directory on the PRTG Probe server:

"<drive>:\Program Files (x86)\PRTG Network Monitor\Custom Sensors\EXEXML\"

The Azure Sensor pack will not function anywhere else. The extracted files will create a directory structure as seen below. The root directory under EXEXML will look like the picture below:

Name	Date modified	Туре	Size	
🕹 AutoMonX	23/07/22 1:51 PM	File folder		
🔊 Automonx_AzureSensor.cmd	28/05/19 6:54 PM	Windows Comma	1 KI	3
🔊 Automonx_AzureSensorEngine.cmd	15/08/21 5:09 PM	Windows Comma	1 KI	3

AutoMonX directory content:



Contrib
Data
Logs
Queue
Reports
ResponderREQ
Scheduler

AutoMonX Root directory would include the following files

Filename	Purpose
AutoMonX_AzureSensorEngine.cmd	Used for invoking the Sensor by PRTG
AutoMonX_AzureSensor.cmd	

Common directory would include the following files:

Filename	Purpose
ExecutableActivation.dll	Discovery and monitoring User Interface
ExecutableActivation.pdb	files
FileHelpers.dll	
FileHelpers.xml	
Newtonsoft.Json.dll	
Newtonsoft.Json.xml	
Renci.SshNet.dll	
Renci.SshNet.xml	
SensorAutoDisco_UI.exe	
SensorAutoDisco_UI.exe.config	
SensorAutoDisco_UI.ini	
SensorAutoDisco_UI.Lib.dll	
SensorAutoDisco_UI.Lib.pdb	
SensorAutoDisco_UI.pdb	
LicDetailsLocator.exe	Utility to gather the required details for
	license generation
AutoMonX_PRTG_Automation.exe	Monitoring Automation module files
AutoMonX_PRTG_Automation.ini	
AMX_PRTG_senors_issues.exe	
exclude_mon - Example.csv	Filtering logic files for Monitoring
exclude_mon.csv	Automation
include_mon.csv	
down_sensors_filter.ini	
pause_sensors_filter.ini	
AutoMonX_ReqFetch.dll	DLLs required for Monitoring Automation
libcrypto-1_1-x64.dll	
libgcc_s_seh-1.dll	



libssh2-1.dll	
libssl-1_1-x64.dll	
libstdc++-6.dll	
libwinpthread-1.dll	
zlib1.dll	

Azure directory would include the following files:

Filename	Purpose
Contrib Data Logs Queue	Sub-directories required for the Azure sensor operation
Reports ResponderREQ Scheduler	
AutoMonX_AzureCollector.exe AutoMonX_AzureResponder.exe AutomonX_AzureSensor.exe AutoMonX_AzureSensorRun.exe automonx_azuresensorsumm_starter.exe Azure_Backend_Service.exe	Azure Sensor Pack executables
AutoMonX_AzureSensor.ini AzureConnProfiles.ini	Azure Sensor Pack main configuration file and Azure Connection profiles configuration file (for multiple Tenants)
AutoMonX_AzureLicense.dat AutoMonX_AzureTenantLicense.dat	Azure sensor license file – Sensors Azure Tenants license file
libcrypto-1_1-x64.dll libgcc_s_seh-1.dll libssl-1_1-x64.dll libstdc++-6.dll libwinpthread-1.dll zlib1.dll	Azure sensor DLL files

OVL directory content:

Filename	Purpose
automonx.azure.availability.ValueLookup.ovl	PRTG custom lookup file for the Azure
automonx.azure.availabilitymetric.ValueLookup.ovl automonx.azure.certificate.ValueLookup.ovl automonx.azure.jobbackupstatus.ValueLookup.ovl	Sensor Pack
automonx.azure.quota.ValueLookup.ovl automonx.azure.status.ValueLookup.ovl	
automonx.azure.taskstatus.ValueLookup.ovl automonx.azure.connectionstatus.ValueLookup.ovl	



Contrib directory content:

Filename	Purpose
Amx_Discovery_Addition.cmd Amx_Discovery_Addition.ps1 Amx_Sensor_Deletion.cmd	Example scripts for running automated discovery and addition to PRTG. Please don't edit these files – instead copy to a different folder, as they will be overwritten upon upgrade.

Reports directory content – here you will find the Auto Discovery reports in HTML format.

Scheduler directory content:

Filename	Purpose
Scheduler.csv	Sample settings for the scheduler functionality that allows running API commands for the Azure Sensor Pack automation such as discovery, monitoring automation and more



4.7 Lookup File Handling (on-prem PRTG)

You need the AutoMonX Azure Sensor Lookup files to properly display the sensor output in PRTG. Copy the following files from the OVL folder, located in the zip file to the following folder on the PRTG Core server.

"<drive>:\Program Files (x86)\PRTG Network Monitor\lookups\custom"

Below are the OVL files you need to copy:

- automonx.azure.availability.ValueLookup.ovl
- automonx.azure.availabilitymetric.ValueLookup.ovl
- automonx.azure.certificate.ValueLookup.ovl
- automonx.azure.jobbackupstatus.ValueLookup.ovl
- automonx.azure.quota.ValueLookup.ovl
- automonx.azure.status.ValueLookup.ovl
- automonx.azure.taskstatus.ValueLookup.ovl
- automonx.azure.connectionstatus.ValueLookup.ovl

After copying the Lookup files to the PRTG Core Server, you would need to reload the PRTG Lookup database by the following action:

From the PRTG upper menu -> Setup -> System Administration -> Administrative Tools -> Reload Lookup Files



4.8 Lookup File Handling (PRTG Cloud)

You can now manually add the custom Automonx lookup files into hosted PRTG. After installation on the probe you can find them in the folder: PRTG Network Monitor\Custom Sensors\EXEXML\AutoMonX\OVL

Subscription (Overview		+ CREATE NEW SUBSC	RIPTION
Q SEARCH BY COMMA	SEPARATED STRINGS			·= =
\oslash				
Subscription Owner	Company	Subscription Plan Hosted 1000 (1 Year)	Your Order No.	
OPEN PRTG ACTIONS	1			8
\oslash				
Subscription Owner	Company	Subscription Plan Hosted 1000 (1 Year)	Your Order No.	
OPEN PRTG ACTIONS	1			0
\oslash				
Subscription Owner	Company	Subscription Plan Hosted 500 (1 Year)	Your Order No.	
OPEN PRTG ACTIONS				



Upload Custom Files

=	AutoMonX
	The Monitoring Automation Company

My Custom Files

Upload your custom files. Upload your own created custom device templates here if the predefined custom templates you need are not available in the list above.

 You cannot upload files that are larger than 1MB. 	
MIB/ .MIB, .mib, .my	
devicetemplates/.odt	•
lookups/custom/.ovi	•
snmplibs/.oidlib	•
webroot/icons/devices/.svg,.png	•
UPLOAD	

Custom Files

- 1. Click to the ovl file type and browse for the path to the file in the **File Explorer**.
- 2. Select the file and click **Open**.

My Custom Files	
Upload your custom files. Upload your own created custom device templates here if the predefined custom templates you need are not available in the list above.	
You cannot upload files that are larger than 1MB.	
MIB/.MIB, .mib, .my	÷
devicetemplates/.odt	+
lookups/custom/ .ovl	+
 snmplibs/.oidlib 	+
Custom_file.oidlib	
webroot/icons/devices/ .svgpng	+
UPLOAD	

Custom File

3. Click **Upload** to upload the custom ovl files.





Previous way to handle hosted PRTG lookups:

Since custom lookup files are not available in the PRTG Cloud edition, the Azure Sensor pack can also function with the built-in lookup files. To enable the PRTG Cloud mode, you need to edit the AutoMonX_AzureSensor.ini file.

Open the *AutoMonX_AzureSensor.ini* file in Notepad and modify the following line from false to true (support PRTG Cloud)

CLOUD_VLOOKUP=true

Save the file and restart the AutoMonX Azure Sensor Service.



4.9 Requesting a License for the AutoMonX Azure Sensor pack

The initial license file used by the Azure Sensor Pack, part of the downloaded zip file, is empty and functions as a place holder. You must activate the sensor by obtaining a license.

To successfully activate the Azure Sensor Pack, you must contact AutoMonX Ltd either by filling the license request form at <u>http://www.automonx.com/azure</u>

Or by sending an email to <u>sales@automonx.com</u> and provide the following information:

- Your first and last name
- Your contact details (email, phone)
- Your business addresses.
- The hostname of the PRTG Probe server machine
- The IP address of the PRTG Probe server

Important: The hostname is case sensitive. Please use the LicDetailsLocator.exe utility to obtain the hostname and IP address

AutoMonX would provide you with a fully functional software evaluation license (two strings) valid for 30 days.

At the end of the evaluation period, you would need to purchase a license to continue monitoring your Azure infrastructure.

4.10 Activating the Azure Sensor pack License

In version 3.23 and higher, you can activate the licenses of the Azure Sensor Pack by <u>opening our UI</u> and selecting the Settings Tab. Select "Azure" from the Product drop-down list (if not selected) and paste the license string you have received via email.

You can also activate the Azure Sensor Pack by editing the following files via Notepad, pasting the relevant license string(s) you have received via email and saving the files.

AutoMonX_AzureLicense.dat – For Azure Sensor pack resources monitoring license for single Tenant scenarios.

AutoMonX_AzureTenantLicense.dat – For Azure Sensor pack multi-tenant license for multiple Azure tenants monitoring scenarios.



AutoMonX	AutoMonX	Discovery And Autom	ation For PRTG	i	
Settings Discovery Devic	e Discovery Results SNM	P Discovery PRTG Group Setting	gs Add sensors to PRT	G	
	PRTG Installation Path	Configuration And Licensi	ing Netw		
	Product:	Azure	~		
	License:	/O95zP5Jcsv9WpIZYdiU9jd1	IEe7e		
	LogPath:				
	Multi Tenant License:	Cq69ki80yJB7zAKCxZHB5V	mJkrc		
		Update			
All Rights Reserved © AutoMonX Ltd 2	020 - V1.12.8			Back	Next



5 Azure Sensor Pack Configuration

5.1 Preparing for Configuring the AutoMonX Azure Sensor pack

The AutoMonX PRTG Azure Sensor pack connects to Microsoft Azure via a service principal, that at least must have read permissions. You need to obtain the following information for the AutoMonX Azure Sensor Pack to properly function:

- APP Identification key (Application Id) AZURE_AD_ID
- AD password KEY (SECRET KEY) AZURE_PASS
- Tenant Identification key AZURE_TENANT

The connection information of the <u>first</u> Azure tenant added to monitoring is always stored in AutoMonX_AzureSensor.ini.

Starting with Azure Sensor Pack version 4.0, when monitoring a multi-tenant Azure environment, the location of the connection parameters for any additional Azure tenant(s) has moved to a new file called AzureConnProfiles.ini.

The Azure connection settings are modified by using the AutoMonX UI. Check the <u>Configuring the Azure Sensor Pack</u> section for more information.

Note: The Azure Sensor Pack is backward compatible and will be able to read the Azure connection details from their previous location.

The Tenant ID is the Directory ID which can be found in the Properties submenu in the Azure Active Directory resource.

Use the following guide to create a service principal to get an Application Id and password:

https://docs.microsoft.com/en-us/azure/azure-resource-manager/resourcegroup-create-service-principal-portal

It is recommended to follow the entire guide in order to prepare all the relevant settings for the AutoMonX Azure Sensor Pack.



5.2 Proxy Server Connection Configuration

The following is required to configure the proxy data. In the file "AutoMonX_AzureSensor.ini" add the following lines:

[PROXY] PROXY_SERVER= PROXY_PORT= For example:

Automonx_AzureSensor.ini - Notepad

File Edit Format View Help

[PROXY] PROXY_SERVER=64.225.8.142 PROXY_PORT=8443

[AUTH]

5.3 Configuring the AutoMonX Azure Sensor pack

You need to start the Azure Sensor pack configuration UI by running as Administrator a file called SensorAutoDisco_UI.exe from the AutoMonx\Common folder.

Use the configuration UI to fill the required details for the Azure Sensor Pack to connect to Azure API as the initial ini file contains dummy data.

	Azure Discovery Setting	S	
Product:	Azure	✓ Update Config ①	
	Connection Profile		
Tenant Name:	Tenant1	- 🕂 🔟 💉	
App ID:		Encrypt 🕕	
Secret Key:		Encrypt	
Tenant:		Encrypt	
Subscription:	All	✓ All	
Offer Id:		Billing	
Currency:			
Country Code:			
Threads:	5 🔹 🚺		
Install Service Con	fig Check Apply Dis	cover Discover Insights	



Important: Press the Apply button to save your changes and press the "Install Service" button to install the Azure Sensor Pack service.

Azure Billing Information: Starting with version 4.2.8 of the Azure Sensor pack, it is no longer required to add the Azure billing information. The Azure billing information will be automatically discovered.

Below is an example how the UI would look like if the service is already installed and started.

gs Discovery De	vice Discovery Results	SNMP Discovery	PRTG Group Settings	Add sensors to PRTG	
	Product	Azure D	Discovery Settings	Update 👔	
	Troduct.	Azure			
	Tenent Name	Conn	ection Profile		
	renant Name:	My Tenant	~	+ 🛛 🖍	
	App ID:	15d2660c-66	6af-4086-87cb-8ae0d	Encrypt	
	Secret Key:	ifn7Pf~4D~4	E14entVRHI7ktDF5F	Encrypt	
	Tenant:	5d33d3aa-4	7ff-4b98-9c02-a026c1	Encrypt	
	Subscription:	All			
	Offer Id:			Billing	
	Currency:				
	Country Code:				
	Threads:	5 ~	0		
	_				
	Confi	g Check A	pply Discover	Discover Insights	•

In some cases, where the Azure Sensor Pack service is not started, the UI would make visible the "Start Service" button so you can start the Sensor Pack service from our UI

=	AutoMonX
	The Monitoring Automation Company

	Azure Discovery Settings	
Product:	Azure ~	
	Connection Profile	
Tenant Name:	My Tenant ~	+ 🔟 💉
App ID:	15d2660c-66af-4086-87cb-8ae0d	□ Encrypt
Secret Key:	FyYUtQ4i~.i~a3zAK3C-SkZu_2R0	Encrypt
Tenant:	5d33d3aa-47ff-4b98-9c02-a026c1	Encrypt
Subscription:	All	All
Offer Id:		Billing
Currency:		
Country Code:		
Threads:	10 ~ 🕕	
		Discourse la si stata
Start Service Confi	g Check Apply Discover	Discover Insights

5.4 Azure Sensor pack - Configuration Check

To verify your Azure Sensor Pack installation, start the UI, go to AutoMonX\Azure\Common directory and run as Administrator the SensorAutoDisco_UI.exe file.

Fill in all the required information. Press the Config Check button to initiate a self-check to make sure everything was configured correctly. Successful test will look like the screen below:





When a Multi-Tenant license is available, Config check will report the number of Tenant licenses:





5.5 Azure Sensor pack – Upgrade Instructions

If you are required to upgrade an existing installation of the AutoMonX Azure Sensor Pack, please follow the steps below.

Upgrade Notes:

The license of any version prior to **v4.0.28**, needs to be migrated. Please contact <u>sales@automonx.com</u> before upgrading to the latest releases.

When upgrading from version **4.0.17** or lower, update the OVL files in the relevant directory and reload them via the PRTG Web Administration page. Delete all the AutoMonX_Azure_types.dat files and run full discovery, otherwise the metrics polling will not function properly. This is automated with the installer in later versions.

Starting from version **4.2.0**, auto-discovery by default uses the HTTP Data Advanced sensors as the integration point with PRTG. This is by design as it replaces the previous EXE/Script Advanced sensor. You can fallback to EXE type sensors by changing the HTTP_SENSOR_MODE value in the AutoMonX_AzureSensor.ini file.

Make sure to follow the upgrade procedure carefully and avoid copying INI and the license (.dat) files from the zip file unless instructed. It is suggested to pause the Azure Sensor Pack sensors until the upgrade is completed.

Using the Azure Sensor Pack Installer is highly recommended. The installer automatically upgrades all the Sensor pack files. Automatic upgrade to the latest version is supported starting from version 4.0.17 of the Azure Sensor pack.

- Download the latest Azure Sensor Installer from <u>https://www.automonx.com/downloads</u>
- Make sure to pause the Azure root group in PRTG.
- Add the PRTG passhash to the configuration file (To smoothly update the lookup files. You may delete this later). For example:



AutoMonX_PRTG_Automation.ini - Notepad

File Edit Format View Help FIRST_CHECK_TIMEOUT=15 SECOND_CHECK_TIMEOUT=5

[Connections]
PRTG_USER=prtgadmin
PRTG_SERVER=127.0.0.1
PRTG_PORT=443
HTTPS_CONNECTION=1
PRTG_PASSHASH=4224444444

- Make a backup of the entire AutoMonX folder.
- Start the installer and follow the instructions.



When upgrading make sure to tick the AutoMonX Azure Backend Service every time if it is already installed:

AutoMonX The Monitoring Automation Company

Setup - AutoMonX Azure Sensor Pack Additional Monitoring Features You can enable additional monitoring features from the list below	,		- · ×
☑ <u>A</u> utoMonX Azure Sensor Pack			
□ <u>Automonx Azure Backend Service</u>			
	<u>B</u> ack	<u>N</u> ext	Cancel
Setup - AutoMonX Azure Sensor Pack			- 🗆 X
Select Additional Tasks Which additional tasks should be performed?			
Select the additional tasks you would like Setup to perform while ins click Next.	stalling AutoMon≻	Azure Sensor Pa	ack, then
Additional shortcuts:			
□ Create desktop icon to the UI for AutoMonX Azure Sensor Pack			

AutoMonX Ltd $\ensuremath{\mathbb{C}}$ 2024 All Rights Reserved

Web : http://www.automonx.com Email : <u>support@automonx.com</u>



Setup - AutoMonX Azure Sensor Pack Ready to Install Setup is now ready to begin installing AutoMonX Azure Sensor Pack	on your compute	er.	
Click Install to continue with the installation.			
	Deale	L	Quart
	Dack	Install	Cancer
Setup - AutoMonX Azure Sensor Pack			>
Installing Please wait while Setup installs AutoMonX Azure Sensor Pack on yo	our computer.		
Installing Please wait while Setup installs AutoMonX Azure Sensor Pack on yo Uninstalling existing services	our computer.		
Installing Please wait while Setup installs AutoMonX Azure Sensor Pack on yo Uninstalling existing services	our computer.		
Installing Please wait while Setup installs AutoMonX Azure Sensor Pack on yo Uninstalling existing services	our computer.		
Installing Please wait while Setup installs AutoMonX Azure Sensor Pack on yo Uninstalling existing services	our computer.		
Installing Please wait while Setup installs AutoMonX Azure Sensor Pack on yo Uninstalling existing services	our computer.		
Installing Please wait while Setup installs AutoMonX Azure Sensor Pack on yo Uninstalling existing services	our computer.		
Installing Please wait while Setup installs AutoMonX Azure Sensor Pack on yo Uninstalling existing services	our computer.		
Installing Please wait while Setup installs AutoMonX Azure Sensor Pack on yo Uninstalling existing services	our computer.		
Installing Please wait while Setup installs AutoMonX Azure Sensor Pack on yo Uninstalling existing services	our computer.		
AutoMonX The Monitoring Automation Company

Setup - AutoMonX Azure Sensor Pack		_	
AutoMonX Azure Sensor Pack inst	allation conclusion		
Thank you for installing the AutoMo	onX Azure Sensor Pack		
Sensor pack files copied	~		
Service installation	✓		
Reload Lookup Files in PRTG	×		
Details required for evaluation l	cense request:		
The host name for the machine is: The IP address for the machine is The MAC address for this machine	'AzureTest'. . '10.3.0.16' e is: '60-45-BD-82-7C-65'.		
Contact us to get the evaluation lid Or mail sales@automonx.com.	ense via automonx.com		
		Next	

- If an error occurred while updating the Lookup files, update them manually <u>as explained in section 4.6.</u>
- Resume the sensors in PRTG.



5.6 Azure Sensor pack Service – INI config

The following table shows the configurable settings in the AutoMonX_AzureSensor.ini file.

Parameter	Default Value	Details
AZURE_AD_ID	Empty	This is the application connection ID. Can be retrieved using the following guide: <u>https://docs.microsoft.com/en-</u> <u>us/azure/azure-resource-</u> <u>manager/resource-group-create-service-</u> <u>principal-portal#get-application-id-and-</u> <u>authentication-key</u> Or refer to this section: <u>Troubleshooting</u> <u>Retrieving Azure application ID and</u> <u>tenant ID</u>
AZURE_PASS	Empty	This is the application connection authentication key Can be retrieved using the following guide: <u>https://docs.microsoft.com/en-</u> <u>us/azure/azure-resource-</u> <u>manager/resource-group-create-service-</u> <u>principal-portal#get-application-id-and-</u> <u>authentication-key</u> Or refer to this section: <u>Troubleshooting</u> <u>Retrieving Azure Application Pass</u>
AZURE_TENANT	Empty	This is the Tenant Id of the Active Directory Azure Application. Can be retrieved using the following guide: <u>https://docs.microsoft.com/en-</u> <u>us/azure/azure-resource-</u> <u>manager/resource-group-create-service-</u> <u>principal-portal#get-tenant-id</u> Or refer to this section: <u>Troubleshooting</u> <u>Retrieving Azure application ID and</u> <u>tenant ID</u>
IGNORE_INVALID_DATA_RETRIES	2	How many times should the Sensor Pack ignore invalid data messages from Microsoft Azure
IGNORE_ERROR_RETRIES	2	How many times should the Sensor Pack ignore temporary errors conditions



SEND_ERROR_RETRIES	20	Retry count for getting data from an unresponsive resource.
CONNECTION_AND_TOKEN_TIMEOUT	60	Data fetch timeout from Azure Portal
REQUEST_TIMEOUT	80	This setting controls for how long a sensor exe will stay alive until it reaches a timeout. Should set higher as more sensor are added.
PREDATA_VALID_SEC	700	Sensor data validation intervals.
PROCESSING_TIMEOUT	200	Sensor data fetch timeout in seconds.
DATA_CACHING_SEC	200	Minimum time to wait between the same Azure API calls, to lower the number of fetches in case of excessive PRTG calls.
THREAD_NUMBER	8	The number of threads the Azure Sensor will use for running. More threads mean better Sensor efficiency but higher CPU. See <u>Service Threads</u> section for more information.
SERVICE_RESTART_MIN	1440	This setting controls how long in minutes to keep the service running before a controlled restart.
AZURE_SELF_MON_SERVICE_CHECK	60	The interval in seconds between performing the service functionality check.
AZURE_SELF_MON_MEMORY_LIMIT_MB	1000	The limitation in MB of the amount of RAM taken up by the AutoMonX processes. Lowering this might cause unwanted frequent restarts to the service.
DEBUG_LOG_DIR	LOG_DIR	This setting tells the Sensor Pack where to store its debug log files. The default LOG_DIR points to the AutoMonX\Azure\Log folder. Leave blank to point to the PRTG log folder (Usually: C:\ProgramData\Paessler\PRTG Network Monitor\Logs (Sensors)
SERVICE_DEBUG	0	This setting runs the service in debug mode. Requires a service restart for every change of setting. The debug log is written to the AutoMonX_ServiceDebugLogger file. This file should be reviewed by AutoMonX support for any assistance.



		This setting should be turned off when a debug mode is not necessary.
HTTP_SENSOR_MODE	1	Discovery will generate sensors of type HTTP Data Advanced. To fallback to EXE/Script Advanced type sensors, turn this value to 0.
SET_SENSOR_NOT_FOUND_TO_WARN	false	When set to TRUE, will automatically mark removed resources as warning instead of error.
CLOUD_VLOOKUP	false	Support for non-custom lookup when using PRTG cloud. Configure this setting before adding to PRTG.
AZURE_DEPRECATED_METRICS	OFF	When set to TRUE, will allow the discovery of deprecated metrics from Azure Portal.
DISCOVER_BACKUP_JOBS_SUMMARY	False	Change to True if you with the Backup jobs to be discovered in a summary sensor per backup vault.
SET_AZURE_HEALTH_CRITICAL_STATE	any	When set to non_user, will mark any virtual machines stopped by user as warning instead of error.
DISABLE_DEFAULT_THRESHOLDS	FALSE	When set to TRUE, will disable all default thresholds from channels (not including lookups)
AZURE_USED_REGIONS		Used for displaying the relevant channels in the quota sensors. List the used regions by your organization separated by commas to limit the channels to the regions you would like to see.
AZURE_QUOTA_LIMIT	80	The limit to alert quota usage.
<metric_name>_Aggregation_Type</metric_name>		Custom parameters to edit the aggregation type of specific metrics.
AZURE_GOV_ENABLED=TRUE AZURE_GOV_COUNTRY=USA	FALSE	Support for government costumers
VM_NAME_FOR_SNMP	TRUE	Set to TRUE to activate SNMP discovery by PRTG for VMs in the SNMP section in the UI. Set to FALSE to remove the "Node" label from the VM name.



6 Introducing Multi-Tenant Azure Monitoring

The AutoMonX Azure Sensor Pack is capable to automatically discover and monitor multiple Azure Tenants from a single PRTG probe. This major improvement allows CSPs/MSPs and large enterprises to monitor their entire Azure estate without being limited by Tenant boundaries.

Important License information: To support the Multi-Tenant version, new license types are available as specified below. If you require to monitor multiple tenants, make sure to contact AutoMonX sales <u>sales@automonx.com</u> to obtain the most suitable license type.

6.1 Multi-Tenant License Types Explained

Additional license types are available to facilitate the monitoring of resources in multiple Azure tenants.

- The license is bound to PRTG probe machine
- Single-tenant Azure Sensor pack licenses are still available
- Multi-Tenant licenses are sold in packs:
 - o 5 Tenants
 - o 10 Tenants
 - o 15 Tenants
 - o 20 Tenants
 - o 25 Tenants
 - o 50 Tenants
- Each Multi-tenant license pack allows you to monitor an unlimited number of subscriptions and unlimited number of sensors within the number of tenants you have purchased.

• License boundaries/limitations:

- Your PRTG License
- The number of Tenants in the Multi-Tenant license you have purchased (5,10,25,50)
- The license is bound to a specific PRTG Probe
- The PRTG Probe physical capabilities



6.2 Configuring Multi-Tenant Discovery

You need to configure the details of each tenant you wish to add to Azure Sensor Pack auto-discovery by using our UI.

		Azure Discovery Settings		
	Product:	Azure 👻	Update Config 🕕	
		Connection Profile		
	Tenant Name:	Tenant1 -	+ 🔟 💉	
	App ID:		Encrypt 🕕	
	Secret Key:		Encrypt	
	Tenant:		Encrypt	
	Subscription:	All	✓ AII	
	Offer Id:		Billing	
	Currency:			
	Country Code:			
	Threads:	5 -		
Install	Service Conf	ig Check Apply Discover	Discover Insights	

Tenant1 •	+	匬	Í	>
	Tenant1 -	Tenant1 🔹 🕂	Tenant1 🔹 🕂 🛈	Tenant1 🔹 🕂 🔟 🌶

Connection Profiles have been introduced for quick and easy identification of tenants in our UI and in PRTG. You can edit these labels by using the editing icons. You can move between different Connection Profiles by using the drop-box menu.



Important: Once a connection profile was added and its respective tenant subscriptions and their Azure resources have been added to PRTG, it is better to keep the same Connection Profile names and not modify them, as it may create duplicate entries in PRTG during additional rounds of auto-discovery and Monitoring automation activities.

Connection Profile editing icons:



- Plus sign: Add new connection profile details
- Trashcan: Delete a connection profile from the list
- Pencil: Modify tenant label

6.3 Encryption of Connection Profile details

Version 4.x introduces the ability to encrypt the Azure connection profile details. You can choose to encrypt any of the connection details: App ID, Secret Key or Tenant ID by ticking the check boxes on the right as seen below:

App ID:	🔲 Encrypt 🚺
Secret Key:	Encrypt
Tenant:	Encrypt

Important: Once the connection details of a profile are encrypted, there is no way to decrypt them via our software for you to see. This is by design and aimed at protecting your Azure connection details. It is recommended to store the connection details in a safe place or password management software in case you would need to enter them again.

Encryption via CLI is available with the command:

AutoMonX_AzureCollector.exe -enc_data <secret_key>

6.4 The Azure Sensor hierarchy in PRTG

Our Monitoring Automation creates a tree-structured hierarchy based on some initial configuration made in our UI. You need to provide the top-level group for all Azure assets (needs to be manually created in PRTG) and specify the labels for each Tenant in our UI. The rest of the hierarchy would be automagically created for you in PRTG as seen below:

PRTG Probe (where the Azure sensor is installed)



- **Azure** (The top-level PRTG group you need to manually create, could be any name)
 - Tenant label (in a format of <Tenant Label>-<Last 4 digits of Tenant id) i.e. AutoMonX-3234. The group must be created manually.
 - Subscription name (in a format of <sub name>-sub. i.e. Prod – sub)
 - Resource Group(s) Automatically generated by Monitoring Automation (i.e. SQL, LogicApp, Storage etc)

See below an example of the tree structure created automatically by our Monitoring automation:



	Automonx_Azure	
Ŧ	🔄 prod - Sub	
	W 2 Sens ✓ 247 Se U 1 Sens	
Ð	microsoft partner network - Sub1	
	⊞ 'æ Web <mark>√7 Sens</mark>	
	⊞ 🔄 Sql	
	V T Sens	
	⊞ Logic ✓ 2 Sens ✓	
	E E Storage	
	 	l
	🛱 microsoft azure sponsorship - Sub	
	⊞ Instwork ✓8 Sens	
	⊞ ⊡ Disk	
	 B Compute ✓ 13 Sen 	



6.5 Migration to Multi-Tenant Version

If you are planning to monitor multiple Azure tenants via the Multi-Tenant functionality and your existing Azure Sensor Pack version is prior to v4.x, follow the steps below:

- 1. **Upgrade the software version**: Complete the software upgrade steps as explained in <u>Upgrade instructions</u>.
- 2. Add new Azure tenant(s): Start our Configuration UI and add the new Azure Tenants you want to auto-discover and monitor.
- 3. **Modify Tenant label(s)**: Your existing Tenant will get a default label: MyTenant. You can modify it via the UI to a label of your choice.
- 4. **Update the Configuration**: Use the "Update Config button" to start the configuration update.

	Azure Discov	ery Settings	
Product:	Azure		Update Config 🕕
The following warning	will be shown:		
Update Configuration			
You are about to update y The process may take set Make sure to update the A See the deployment guide	our Azure sensor settin veral hours but monitori Azure sensor pack files of details	gs to a new format. ng will continue norm to the latest version b	ally. efore starting the process.
	Update	Cancel	

Press on Update button to continue with the process or Cancel it.

5. **Upgrade Process**: The Azure Sensor Pack will start to re-discover the tenants in a sequential order. During the re-discovery, monitoring will continue without interruptions



6. **Finalizing the update**: At the end of the process that may take several hours, the following messages will be shown:

Finalize?
The Azure Sensor pack configuration has completed. Status: Ended successfully. In order to finalize the update press on Finalize button. This will stop and start the Azure sensor service. You can Finalize the process later by restarting the UI and pressing Update Config again.
Finalize Cancel

Press Finalize button to complete the update process. During this step the Azure Sensor pack service will stop and start automatically.

7. Adding the new Tenant(s) to PRTG: Use our UI to add the new Tenant(s) to PRTG as explained in the next section of this guide.



7 Auto Discovery and Monitoring Automation

7.1 Automatic Discovery of Azure Resources

The AutoMonX Microsoft Azure Sensor Pack needs to scan the Azure environment for any resources it can monitor. In auto-discovery mode, the Sensor Pack will generate a list of all the Azure resources in your environment that it can monitor. It would also provide you with the required sensor configuration to monitor these resources.

Important: Starting version 4.2.x, the default sensor type used for integrating the Azure Sensor Pack with PRTG is HTTP Data Advanced as it has much lower performance hit on the PRTG Probe. If you wish to continue utilizing the EXE/Script Advanced type sensors, change the HTTP_SENSOR_MODE configuration in *the AutoMonX_AzureSensor.ini* file prior to running discovery

There are two discovery modes:

- 1. Thorough Discovery
- 2. Quick Discovery

The Quick discovery is currently only available via CLI, read more here.

Press the "Discover" button to start the Azure resources discovery. At this stage, the auto discovery will take place.

Note: Depending on the network connection, the Azure API response time and taking into account the size of your Azure deployment, it can take between a few minutes to several hours to complete.

	Config Check Apply Discover Discover Insights Cancel
	Discovery may take some time, please wait
	1.923% Completed
A	W Rights Reserved © AutoMonX Ltd 2019 - V1.9.5



You can cancel the discovery process while it is running by pressing the "Cancel" button.

The discovery process can take some time, follow-up the progress by checking the message area at the bottom of the screen.

Timeout messages may appear sometimes during the discovery process, but you can safely ignore them if they last no longer than 10 minutes.

When auto-discovery has completed, the following window will pop-up. Now you can move to the next tab and examine the discovery results.



7.2 Previous Discovery Results handling

In cases there are previous auto-discovery results, the UI will offer to use them instead of re-discovering again the Azure resources, which can be time consuming.



Before starting auto-discovery, the UI will backup any previous discovery results and pop-up the following window:

	Files are being backed up	x
0	Backing up your previous discovery files in: C:\Program Files (x86)\PRTG Network Monitor\Custom Sensors\EXEXML\automonx\common\backup	
	ОК	



7.3 Sensor types created by the Azure Sensor pack

The Azure Sensor Pack auto-discovery will create several sensor types, most of them are custom sensors:

Sensor Type	Description	Actions in Azure Portal / PRTG
Azure App Metrics	Resource-specific performance metrics	Performance Metrics must
	available via the Azure API, multiple	be enabled per each
		resource via the Azure portai
Azure Service Health	Resource health as reported in Azure	Resource health needs to be
	portal, two channels	configured via the Azure
		portal
Azure VM Multi-Disk	For Virtual Machines only, disk metrics	Log Analytics must be
	polled with Log Analytics.	enabled on the virtual
		machine.
Azure LogicApp	The workflow status of a Logic App.	
Azure DataFactory	The Data Factory Status	
DataSet		
Appa Saarat Kay	The evolution of each accretion	
Status	under available Apps	
Olalus		
Azure Billing	Azure Billing data, several channels	
	that cover supported and un-supported	
	resources	
AutoMonX License	Self-monitoring sensor that shows	N/A
	license consumption and days left for	
	maintenance and license to expire (if	



7.4 Selecting Azure Sensors for Monitoring

Press "Next" to move to the next tab. All the discovered Azure resources would be presented:

ettings	Discovery Device Discove	ry Results SNMP Discove	ery PRTG Group Settings	Add sensors to PRTG	
Selected	Sensor	Device	ParentGroup		
	Azure Costs	microsoft azure sponso	microsoft azure sponso		
	Function - timetrigger24	microsoft partner netwo	Web&ParentParentGro		
	Azure Costs	microsoft azure sponso	microsoft azure sponso		
	Azure Service Health	ipl-adconnect_(VirtualM	Compute&ParentParent		
	Azure Service Health	ipldiag3167be85735_(St	Storage&ParentParent		
	Azure App Metrics	ipldiag3167be85735_(St	Storage&ParentParent		
	Azure Service Health	ipldisks521caf93e66_(S	Storage&ParentParent		
	Azure App Metrics	ipldisks521caf93e66_(S	Storage&ParentParent		
	Azure Service Health	elgonitvpn01_(Connecti	Network&ParentParent		
	Azure Service Health	elgonitvnetgw01_(Virtua	Network&ParentParent		
	Azure App Metrics	elgonitvnetgw01_(Virtua	Network&ParentParent		
	Azure Certificate	space_(Certificate)	Web&ParentParentGro		
	Azure Service Health	elgonMyDPRights-Servi	Web&ParentParentGro		
	Azure App Metrics	elgonMyDPRights-Servi	Web&ParentParentGro		
	Azure Service Health	elgonmydprightsbe_(Sit	Web&ParentParentGro		
	Azure App Metrics	elgonmydprightsbe_(Sit	Web&ParentParentGro		
	Azure Service Health	elgonmydprightswk_(Sit	Web&ParentParentGro		
	Azure App Metrics	elgonmydprightswk_(Sit	Web&ParentParentGro		
	Azure Service Health	elgonmydprightskeyvaul	KeyVault&ParentParent		
	Azure App Metrics	elgonmydprightskeyvaul	KeyVault&ParentParent		
	Azure App Metrics	elgonmydprights_(Sql S	Sql&ParentParentGrou		
	Azure Service Health	elgonmydprights/gdpr_(Sql&ParentParentGrou		
	Azure App Metrics	elgonmydprights/gdpr_(Sql&ParentParentGrou		
	Azure App Metrics	elgonmydprights/maste	Sql&ParentParentGrou		
	Azure Certificate	space_(Certificate)	Web&ParentParentGro		
	Azure Certificate	space (Certificate)	Web&ParentParentGro		

Select the sensors you want to add to PRTG by clicking on the relevant checkbox on the left side of the table. You can also click on "Select All" to mark all the sensors. There is also an option to present only certain sensors by using the Search window as seen below:



Aut	toMonX Di	iscovery And Automa	ation For PRTG	
	А	utoMonX Disc	overv And Automa	ation For PRTG
Disc	overy Dev	ice Discovery Results	Add sensors to PRTG	
	Selected	Sensor	Device	ParentGroup
•		Azure Certificate	3EC4#ProdRG-WestEu	Web&Parentprod&Pare
		Azure Certificate	5A73##WestEurope# (Web&Parentprod&Pare
		Azure Certificate	7771#ProdRG-WestEur	Web&Parentprod&Pare
		Azure Certificate	A0E5#ProdRG-WestEu	Web&Parentprod&Pare
	Sel	ect All	certificate	Search Apply
	hts recorded	© AutoMonX Ltd 2019		Back Next
, ar ragi	into reserveu v			

Click "Appply" to save your settings. A confirmation window will pop-up. Click "OK" to confirm or "Cancel".

Please confirm	
Adding 11 sensors to PRTG	
OK Cancel	

Press "Next" to proceed to the "Add sensors to PRTG" tab.

See section <u>6.10</u> for the location of the Discovery results report.

7.5 Configuring PRTG Group Settings

The PRTG Group Settings window provides you with the option to change the scanning intervals of the sensors under the Azure PRTG group. By default, the sensors are added with a 300 second interval (5 minutes). You can tick on the "inherit" checkbox to add the sensors in the Azure groups with the PRTG interval inherit system. More information can be seen it the bottom of the window.

This is an optional setting. Use the "Next" button to move to the next tab



				Settings	PRIG Group	Settings	Add sensors t	o PRTG		
Name	Subscript	tion	Inherit	Scanni	ng_Interval					
Default	test - Sub)		300						
Here	you can de	fine the	e scanni	ng interv	al of PRTG de	evices that	are part of a c	ertain group	о. Ар	ply
Here	you can de se "Inherit"	fine the	⇒scannii ∧ant the	ng interv group to	al of PRTG de take the scar	evices that	are part of a c igs from an up	ertain group	p. Ap	ply
Here Choo All th	you can de se "Inherit" e values arc	fine the if you w ∋ in Sec	e scannii <i>r</i> ant the conds.	ng interv group ta	al of PRTG de take the scar	evices that	are part of a c ngs from an up	ertain group per-level gro	o. Ap	ply



7.6 Auto-Discovery of Azure Virtual Machines SNMP Sensors

Starting with version v3.32 of the Azure Sensor Pack, it is possible to automatically discover and monitor the SNMP performance metrics and Ping (ICMP) of Virtual Machines (Windows or Linux) hosted in Azure, along with the Azure-based metrics, already avilable via the Azure Sensor Pack.

Such hybrid approach will allow you to enjoy the best of two worlds – the rich functionality of Azure metrics monitoring coupled with OS-based SNMP performance metrics under a single device in PRTG.

7.7 Preparing for Auto-Discovery of Azure Virtual Machines

You need to perform the following tasks before initiating Auto-Discovery of SNMP metrics:

• DNS – Configure a DNS record for each VM in the Azure Portal



 Networking Ports – Configure and open networking ports for SNMP (UDP 161) and ICMP

Networking

🖉 Attach network inte	erface 🖉 Detach network interf	ace			
IP configuration ① ipconfig1 (Primary)	~				
Network Interface Virtual network/subnet:	Effe vvw-vnet/default NIC Public I	ctive security rules Top P: NIC Pr	ivate IP: Accelerated	networking: Disabled	
Inbound port rules	Outbound port rules Appli	cation security groups	Load balancing		
Network security gr Impacts 0 subnets, 1	roup automonxwin32-nsg (attac network interfaces	hed to network interface: a	automonxwin32888)		
Priority	Name	Port	Protocol	Source	Destination
300	A RDP	3389	ТСР	Any	Any
310	Port_161	161	Any	Any	Any
320	A Port_icmp	Any	ICMP	Any	Any



- **SNMP Service** install the SNMP service on each Virutal Machine you wish to monitor via the SNMP Discovery.
- **Community String in PRTG**: You must configure the SNMP community string in the PRTG group where plan to create the Azure Virtual Machines

Credentials for SNMP Devices	SNMP Version	() v1
OD inherit from '듦 Linux Monitor SSH (SNMP Version: V2, SNMP Port: 161, SNMP Timeou)		 v2c (recommended) v3
	Community String	public
	SNMP Port	161
	SNMP Timeout (Sec.)	5



7.8 Running Auto-Discovery of Azure Virtual Machines

Make sure you have completed the preparations for monitoring the Virtual Machines. Initiate a new Azure resources discovery using our UI.

In the Discovery Results tab, select the Virtual Machines resources it identified and press "Next". In the SNMP Discovery tab choose the Virtual Machines that have been prepared for monitoring via SNMP and press "Apply"

AutoMonX The Montering Automotic Compared AutoMonX Discovery And Automation For PRTG	- X
Discovery Device Discovery Results SNMP Discovery PRTG Group Settings Add sensors to PRTG Select VMs for SNMP discovery:	• •
Select All	2
All Rights Reserved ® AutoMonX Ltd 2020 - V1.11.3 Back Next	

Pos 🔻	Sensor ≑	Status ≑	Message	Graph
1 .	✓ Azure Service Health	Up	ОК	Service Healtl Available
₽ 2.	Azure App Metrics	Up	ОК	Percentage Ci 4 %
4 3.	V Ping	Up	ОК	Ping Time 150 msec
4 .	Vptime	Up	ОК	System Uptim 5 d 2 h 41 m
4 5.	✓ (002) Local Area Connection* Traffic	Up	ОК	Traffic Total 0 kbit/s
4 6.	CPU Load	Up	ОК	Total 0 %
4 7.	✓ Disk Free: C:\ Label:Windows Serial N	Up	ОК	Free Space 82 %
4 8.	✓ Disk Free: D:\ Label:Temporary Storag	Up	ОК	Free Space 74 %
. € 9.	Memory: Physical Memory	Up	ОК	Percent Avail: 60 %
4 10.	Memory: Virtual Memory	Up	ОК	Percent Avail: 60 %
ф 11.	Service Task Scheduler	Up	ОК	Operating Sta Active



7.9 Automatically Adding Azure Sensors to PRTG

Important:

- Fill-in your PRTG credentials and make sure that the PRTG Web interface connection details (username, password, server IP, port and if HTTPs was enabled) for this step to succeed
- When using <u>Hosted PRTG</u> password is not supported, only passhash.
- You need to <u>manually</u> create a target group in PRTG that will contain the Azure resources sensors before running "Add sensors to PRTG". The default group that our Monitoring Automation is configured to use is AutoMonX_Azure. You can create a group with a name of your choice and indicate it in the "PRTG Group" field.
- In case of **Multiple Tenants**, make sure to create groups with <u>different</u> <u>names</u> in PRTG. Discover each tenant from its respective PRTG Probe and make sure to point our UI to the relevant group (per the Azure tenant you have discovered)

AutoMonX	AutoMonX Disc	overy And Autom	ation For PRTG	= ×
Settings Discovery Device Discovery	ery Results SNMP Disco	very PRTG Group Setting	gs Monitoring Automati	on
	Mor	nitoring Automation		
	I	PRTG Integration		
	PRTG Username:	admin	0	
	PRTG Server:	127.0.0.1	0	
	PRTG Port:	443	0	
	Password		0 🗌 Show Password	
	○ Passhash		🕕 🗌 Show Passhash	
	HTTPs :	Enabled ~	0	
	PRTG Root Group:		* optional	
	Apply	Test Connection	Add To Monitoring	
	- Abbild	rest connection		
				^
				~
All Rights Reserved © AutoMonX Ltd 2023 - V1.18	3			Back Next



Press "Apply" to save your settings.

You can also test the connection to PRTG to make sure everything is correct.

AutoMonX Discovery And Autom	nation For PRTG
Settings Discovery Device Discovery Results SNMP Discovery PRTG Group Setting	Monitoring Automation
Monitoring Automation	
PRTG Integration	
PRTG Licomony Instandaria Instand	 Show Password Show Passhash show Passhash o toptional
Copy to ClipBoard OK	^

Press "Add to PRTG" to add the device and the sensors to PRTG. Confirm the group in PRTG that the Azure resources would be added to.



Allow the AutoMonX Monitoring Automation to add the resources and their sensors to PRTG. This could take several minutes depending on the size of the PRTG installation and the number of sensors to be added. When the process has successfully completed you can close the UI.



7.10 Resuming Adding Sensors in case of Timeouts

Important: The PRTG API sometimes fails to timely respond to the Monitoring Automation API calls. It may cause a timeout and the process of adding sensors may fail. In such cases, wait a few minutes and resume the addition of devices and sensors by pressing the "Add to PRTG" button. The Monitoring Automation will continue from the point it has stopped.

7.11 Automatic Discovery of Azure Application Insights

The AutoMonX Azure Sensor Pack has the capability to monitor the Azure Application Insights in your environment. The Application Insights sensor is used to monitor Azure resources that perform requests. Request information is saved in Azure Insights. The Sensor uses the insight identification and key to monitor the Insights.

The process is very similar to adding other Azure resources to monitoring through PRTG.

After you discover the resources in your environment using the Auto Discovery, another button would appear in the buttons line "Discover Insights" as seen below:



Clicking the "Discover Insights" button will open a window showing all the discovered insights in your environment.



Appinsigntivame	ApiKey	SubName
Site1	FillApiKey	prod2 - Sub
Site2	FillApiKey	prod2 - Sub
Login1	FillApiKey	prod2 - Sub
VM1	FillApiKey	prod2 - Sub
Site3	FillApiKey	prod - Sub
VM2	FillApiKey	prod - Sub

You will need to fill the API keys according to each Application Insight you want to monitor. Create an API key for Application Insights via the Application Insight page in your Azure Portal environment as seen below.

Important: Make sure to grant the API Key permissions to Read Telemetry information.



■ Microsoft Azure	℅ Search resources, services, and docs (G+/)
Home > API Access	
Application Insights	
₽ Search (Ctrl+/)	« 2 + Create API key 🗊 Delete API key 🗹 Help
∓ Funnels	Application ID ③
Ser Flows	
🛂 Retention	API key description Last Used Created On
 Impact 	You haven't set up any ADI keys. Click 'Create ADI key' to get started
🏜 Cohorts	- The never (set up any Arrikeys, click create Arrikey to get statted.
··· More	
Configure	
Properties	
💐 Smart Detection settings	
Usage and estimated costs	
💮 Continuous export	
Performance Testing	
📍 API Access 1	
📒 Work Items	
	A Search resources, services, and c
Home > faf3aga - API Access > Create API ke	29
Create API key	
Create an API key to read Application Insights data.	
API keys are used by applications outside the b Your API keys should be managed like passwo	prowser to access this resource. rds. Keep them secret.
Provide a description to help you identify thi	is API key in the future. 🕥
AutoMon Insgihts Acess	
Choose what this API key will allow apps to do	:
Write annotations 🗊	
Authenticate SDK control channel 🗊	
Generate key	



Place the newly generated key in the API Key cell for each Application Insight you want to monitor.

ease add the correct Al	² I key to the following appli ApiKey	Cations: SubName
Site1	FillApiKey	prod2 - Sub
Site2	FillApiKey	prod2 - Sub
Login1	FillApiKey	prod2 - Sub
VM1	FillApiKey	prod2 - Sub
Site3	FillApiKey	prod - Sub
VM2	FillApiKey	prod - Sub

When you finish, click the OK button and let the Application Insights discovery to run. The next steps are similar to the addition of any other sensor to PRTG as described in section <u>6.3 Selecting the Azure sensors for monitoring</u>

7.12 Using CLI to Discover Azure Resources

The AutoMonX Microsoft Azure Sensor Pack needs to scan the Azure environment for any resources that it can monitor. In the discovery mode, the Sensor Pack will generate a list of all the Azure resources in your environment that it can monitor and provide you with the required sensor configuration to monitor these resources. There are several discovery modes of Azure resources:

- 1. Thorough Discovery (Resources and Metrics)
- 2. Quick Discovery of Azure Resources only
- 3. Quick Discovery of Azure Resource Metrics
- 4. Quick Discovery with Inclusion/Exclusion

AutoMonX Ltd © 2024 All Rights Reserved

Web : http://www.automonx.com Email : <u>support@automonx.com</u>



7.13 Azure Sensor Pack Discovery Modes explained

Since there are multiple discovery options, below is a table that compares the discovery functionality and their benefits:

Discovery	Functionality	Pros and Cons			
Mode					
Thorough	Default discovery mode available in our UI. Discovers all the subscriptions, resources and metrics	Easy to use in a UI-interactive mode. Longer time to see full results. Great for the initial discovery			
		their metrics. Push all the metrics in two simple steps to monitoring			
Quick – Resources only	Quick discovery of Resources only (no metrics), available in CLI only	Great for initial discovery of your Azure estate without spending time on metrics discovery			
Quick – Metrics only	Quick discovery of the metrics of Resources	Great to focus on specific resources only and get their metrics ready to be pushed to monitoring.			
		Uses the Azure resources discovered in a previous step and discovers only their metrics.			
Quick – New	Quick discovery of new	Great for when you have a			
		discover and add only new resources to monitor			
Quick – Discovery with Inclusion or exclusion of specific resource types	Quick discovery using the exclude and include configuration files. Can be used with any of the above options.	Great for even quicker discovery when you have a good grasp of what you want to monitor. Might exclude resources you do wish to monitor.			



7.14 Azure Resources Discovery – CLI Options

Below are some examples for running Auto discovery of Azure resources using the CLI.

• Discover all subscriptions and resources (single Tenant)

AutoMonX_AzureCollector.exe -discovery -sub -All

• Discover resources of a specific subscription (single Tenant)

AutoMonX_AzureCollector.exe -discovery -sub <Subscription Name>

• Discover a specific Azure Tenant in a Multi-tenant environment. The tenant number is configured in the file "AzureConnProfiles.ini". For the first tenant configured in the file "AutoMonX_AzureSensor.ini" use 0 or run discovery without the tenant option as in the previous examples.

AutoMonX_AzureCollector.exe -discovery -sub -All -tenant <tenant_number>

7.15 Quick Discovery CLI Options

Depending on the size of your Azure environment, the full discovery process can take a long time. For shorter discovery (50% or more in time savings) use the following CLI options.

• Azure Resources-only discovery (no metrics, quicker results):

AutoMonX_AzureCollector.exe -discovery -sub -All -resources

Azure Metrics-only discovery on supported resources.
 Important: For first time, use this option only after running the resource discovery or after a full discovery was performed:

AutoMonX_AzureCollector.exe -discovery -sub -All -metrics

Running a discovery without the options resources and metrics will result in a thorough discovery.

• Azure new resources only discovery (only resources that weren't discovered in a previous discovery):

Important: For first time, use this option only after running the metrics discovery or after a full discovery was performed:

AutoMonX_AzureCollector.exe -discovery -sub -All -new_resources



• Include/Exclude discovery

Using the option below would substantially narrow down your discovery time only for specific resource types or resources with a predefined Azure Tag. It instructs the discovery process to consult two configuration files: include_mon.csv and exclude_mon.csv. See <u>this section</u> for more details.

AutoMonX_AzureCollector.exe -discovery -sub -All -whitelist

7.16 Azure Resources Discovery Report

Running the discovery commands generates a report that contains the following information:

- Subscription names.
- List of Microsoft Azure resources for each subscription.
- Command line parameters for the AutoMonX Azure PRTG Sensor pack.
- List of unsupported resource types.
- List of Azure resources that don't have their monitoring metrics enabled.

The commands to monitor the supported resources will be found in **Sub Name>_AutoMonX_Azure_Report.html** in the PRTG EXEXML program folder under the Azure folder.

Note: The PRTG EXEXML program folder is usually located in the path: <Drive>: \Program Files (x86)\PRTG Network Monitor\Custom Sensors\EXEXML Or:

<Drive>: \Program Files\PRTG Network Monitor\Custom Sensors\EXEXML

If the -sub -all parameters were used, the report name will be set to ALL_AutoMonX_Azure_Report.html.

Below is a sample report:

PRTG Sensor pack for Microsoft Azure by Automonx - Discovery Results

Subscription microsoft azure sponsorship

Discovered Resources:

Resource Name	Sensor Azure Type	Configuration Line
microsoft azure sponsorship_(Subscription)	Azure Billing	-sub "microsoft azure sponsorship" -cons
WindowsSRV2_(VirtualMachines)	Azure Service Health	-type Microsoft.Compute/virtualMachines_health -resgrp MS_Sponsored -res "WindowsSRV2" -sub "microsoft azure sponsorship"
WindowsSRV2_(VirtualMachines)	Azure App Metrics	-type Microsoft.Compute/virtualMachines -resgrp MS_Sponsored -res "WindowsSRV2" -sub "microsoft azure sponsorship"
linuxsrv167_(NetworkInterfaces)	Azure App Metrics	-type Microsoft.Network/networkInterfaces -resgrp MS_Sponsored -res "linuxsrv167" -sub "microsoft azure sponsorship"
WindowsSRV2_disk1_d5a2d7a8cc3a4c3bb09c732a6175847e_(Disk)	Azure App Metrics	-type Microsoft Compute/disks -resgrp MS_SPONSORED -res "WindowsSRV2_disk1_d5a2d7a8cc3a4c3bb09c732a6175847e" -sul



7.17 Azure Delta Discovery Report

For customers with a constantly expanding Azure environment, who want to track the newly added resources, a Delta discovery report was added. The delta reports shows only the new resources added during last discovery.

<Sub Name>_AutoMonX_Azure_Discovery_Delta.html



Azure Auto discovery - New Results

Tenant: My Tenant (5d3

Discovery started at: 22 January 2023 16:12

Subscription	New resources	Total resources
microsoft partner network	1	21

Discovery Type: Quick, Metrics Discovery completed at: 22 January 2023 16:33 Total Running time: 0 hours, 20 min

Resource Name

7.18 Azure Channel Limits Report

This report is created automatically based on your sensors and shows the out of the box limits configured for different types of resources. The purpose of this report is to give you a clear picture of the thresholds the AutoMonX sensors have configured.

AutoMonX\Azure\Logs\ChannelLimits.csv

7.19 Monitoring Automation Files

The Monitoring Automation files are created in the Azure Sensor pack settings folder and contain the commands that allow to add the discovered Azure resources to PRTG.

After a successful discovery, a file will be generated in the Data folder for each Azure subscription. The files are named according to the following format: <TenantID>_<subscription name>DiscoveryData.csv

7.20 Using the Monitoring Automation CLI

The Azure Sensor Pack contains a command line interface that automates the addition of Azure resources as sensors to the PRTG system.

To use the automation CLI, first you must edit the file below that is in the Azure\Common folder:



AutoMonX_PRTG_Automation.INI PRTG_USER=<prtg_administrative_user> PRTG_SERVER=<prtg_server_name>

- You need a PRTG user with read and write permissions to operate the program.
- You will need to create a target group in PRTG that will contain the Azure resources sensors.

Below is an example how to use the Monitoring Automation CLI:

AutoMonX_PRTG_Automation.exe -file <DiscoveryData>.csv -p <passhash> -group <target_group>

7.21 Adding the Azure Resource URL to PRTG Comments

Adds the full URL to access the Azure resource in the Azure Portal. The URL is added automatically to the device comments tab in PRTG. This feature can help monitoring teams to rapidly access the relevant resource in the Azure Portal.

To activate this feature, add the option resource_id to the discovery via CLI. Any newly created sensors will show the full resource URL in the device comments:

AutoMonX_AzureCollector.exe -discovery -sub -All -resource_id

Devices Local Probe	 Automonx_/ 	Azure * micros	oft azure * Net	work * DefaultRe	sourceGroup-EUS2-vnet-bastion_(★★☆☆	BastionHosts) 🍷			
Dence Delautice	sourcedrou	ip-cosz-viiet-	.pastion_(past	ioninosts)					
Overview	2 days	30 days	365 days	Alarms	O System Information	🔲 Log	Settings	Notification Triggers	Comments
https://portal.azur	e.com/#@	.com/resource/s	subscriptions/38¢		/resourceGroups/D	efaultResourceG	oup-EUS2/providers/1	/licrosoft.Network/bastionHosts/D	efaultResourceGroup-EUS2-

7.22 Multi-tenant Discovery Scheduler

For automatic discovery scheduling especially in large multi-tenant environments, see the following scheduling function.

Configuration is done in the file "AutoMonx\Azure\Scheduler\Scheduler.csv".

You can also find an example file in the same directory. Adjust the "Time To Run" (Must be in the format Mon HH:mm), and the tenant id you want to run the discovery on. Please refer to the <u>API guide</u> for explanation about the API Params.



AutoMonX The Monitoring Automation Company

8 Supported sensor types

8.1 SQL Database

The Azure SQL Database resource is a fully managed relational cloud database. Its Metrics measure the database's health and performance.



8.2 SQL Elastic Pools

The SQL Elastic Pool is a simple, cost-effective solution for managing and scaling multiple databases that have varying and unpredictable usage demands. The databases in an elastic pool are on a single server and share a set number of resources at a set price. The SQL Elastic pool sensor is used to monitor such pools.





8.3 Certificate

The Azure Certificate resource verifies a secure connection to an internet resource. Its Metric measure its expiration date and host name binding status.



8.4 App Service Plan (Server Farms)

An App Service Plan consists of the underlying virtual machines that will host the Azure App Services. The App Service Plan defines the region of the physical server where your app will be hosted on and the amount of storage, RAM, and CPU the physical servers will have. Below is an example of such service (i.e. Server Farm)





8.5 Web Sites

The Azure Web Apps resource is a scalable web application. Its Metrics measure the Web-App's health and performance.



8.6 Storage Accounts

The Azure Storage Account resource is a massively scalable object storage for unstructured data. Its Metrics measure the storage's health and performance.





8.7 Cloud Services

Azure Cloud Services is an example of a platform as a service (PaaS). This technology is designed to support applications that are scalable, reliable, and inexpensive to operate.



8.8 Database Accounts

The Azure Database account resource manages scalability of databases. Its Metrics measure the account's health and the databases requests.

Last Message: OK								
Last Scan: 5 m 12 s	Last Up: 5 m 12 s	Last Down:		Uptime: 100.0000%	Dow 0.00	ntime: C 00% 1	overage: 00%	Sensor Type: EXE/Script Advanced
Metadata Requests		Mongo Request 0 #		Mongo Requests 0#		Total Request Un 4#		
				Total Requests				
				4#				
∩#	~	0	1 #					



8.9 Service Bus Namespace

The Azure Service Namespace resource manages messaging communication between applications using a bus. Its Metrics measure the Namespace's health and performance.

Successful Requests	Abandoned Messages	ActiveConnections	Completed Messages	Connections Closed.
X	0#	123 #	40 #	0#
	Connections Opened.	Count of active messages in	Count of dead-lettered messa	. Count of messages in a Queu
	0#	0#	81#	81#
N	Count of scheduled message	Incoming Messages	Incoming Requests	Outgoing Messages
0 # 5,151 # 0 # 5,151 #	0#	20 #	1,720 #	20 #
Pending Checkpoint Operatio Sensor Health	Server Errors.	Server Send Latency.	Size	Throttled Requests.
0 # Available	0#	13 Msecs	152 КВ	0#
User Errors.				
20#				

8.10 Batch Accounts

The Azure Batch Account resource manages job tasks. Its Metrics measure the Batch Manager's health and performance.

Last Message: OK									
Last Scan: 201 s	Last Up: 201 s	Last Down: 4 h 38 m		Uptime: 100.0000%	Dowr 0.000	ntime: C 00% 10	overage: 00%	Sensor Type: EXE/Script Adv	anced
Dedicated Core Count			Creating Node C 0 #		Dedicated Node . 0#		Idle Node Count 0#	\bigcirc	
				Leaving Pool No 0#		LowPriority Core . 0 #		Low-Priority Nod 0 #	
0 #		0 1#		Offline Node Co 0#		Preempted Node. 0#		Rebooting Node 0#	
Reimaging Node 0#		Running Node C 🗸	-11 -12	Start Task Failed 0#		Starting Node C 0#		Unusable Node 0 #	
Waiting For Start	\bigcirc								


8.11 Redis

The Azure Redis Database resource is a managed key/value database. Its Metrics measure the database's health and performance.

Last Message: OK									
Last Scan: 237 s	Last Up: 237 s	Last Down		Uptime: 100.0000%	Down 0.000	time: 0%	Coverage: 100%	Sensor Type: EXE/Script Ad y	anced
Connected Clie	ents			Cache Hits		Cache Read		Cache Write	
			662#	-14 (0	69,119 /s	-a 45	28,249 /s	-14 10	
				CPU		Evicted Keys		Expired Keys	
			x	1%		0#		4#	-04-05
				Gets		Operations Per S	S	Server Load	
65 #		0	71# 🐺 🕸	777 #	-14 (3)	35#	-11 Q	0%	
Sets		Total Keys		Total Operations		Used Memory		Used Memory RSS	
237 #	-11 (3)	615,706 #	-11 (3)	1,105 #	-14 (2)	1,206,222,668 Byte	e 🔪	1,257,559,040 Byte	-14 (2)

8.12 Data Factory

The Azure Data Factory resource performs analytical operations on data from various data sources using pipelines. Its Metric measure the pipelines status.





8.13 Scheduler

The Azure Scheduler resource manages scheduling of job tasks. It Metrics measure the task's Health Status



8.14 Search Services

The Azure Search Services resource manages search services performed on applications. Its Metrics measure the search service's health and operations





8.15 Service Bus Queues

Azure Service Bus is a fully managed enterprise message broker with message queues and publish-subscribe topics (in a namespace). Service Bus is used to decouple applications and services from each other. Azure Service Bus supports reliable message queuing and durable publish/subscribe messaging.



8.16 Service Health

The Azure Service Health sensor checks the current service health status of a resource. An error status will show as red with the provided error. Unknown statuses are shown as Warning. For Virtual Machines only, an additional channel VM Status will show with a text description.





8.17 Virtual Machines

The Azure Virtual Machine sensor provides performance information regarding virtual machines hosted in Azure. An example is seen below. Additionally, our monitoring automation can invoke PRTG-native SNMP sensors that can also be automatically discovered and added under the same Virtual Machine device.

Descentere CDU		Channel 🗢	ID 🏺	Last Value 🕈
Percentage CP0		OS Disk Max Burst Bandwidth	9	170,000,000 #
		OS Disk Target Bandwidth	20	25,000,000 #
		Disk Write Bytes	24	2,220 KB
		OS Disk Write Bytes/Sec	13	37 KB
x		OS Disk Max Burst IOPS	30	3,500 #
1 %	0 % 100 %	CPU Credits Remaining	27	159 #
Disk Read Bytes	Disk Read Operations/Sec	OS Disk Target IOPS	19	120 #
		Premium OS Disk Cache Read Hit	26	100 %
OS Disk IOPS Consumed Per	OS Disk Max Burst Bandwidth	Inbound Flows	12	57 #
\bigcirc		Outbound Flows	29	57 #
0%	170,000,000 # 0 OS Disk Used Burst BPS Cred	Inbound Flows Maximum Creati	16	10 #
OS Disk Target IOPS		Outbound Flows Maximum Cre	17	8 #
120 #	1 0%	OS Disk Write Operations/Sec	5	6.54 #
Premium Data Disk Cache Re	Premium Data Disk Cache Re	Disk Write Operations/Sec	28	3.27 #
0%	x_0%	Percentage CPU	7	2 %
VM Uncached Bandwidth Cor	I	VM Cached IOPS Consumed Pe	23	1 %
0%		CPU Credits Consumed	21	0.01 #
		Data Disk Bandwidth Consume	40	0 %
		Data Disk IOPS Consumed Perc	45	0 %
		Data Disk Max Burst Bandwidth	46	0 #
		Data Disk Max Burst IOPS	36	0 #



8.18 Azure Billing

The Azure Billing sensor provides information on the daily cost of a subscription. Can be used by providing an offer ID – offered parameter in the discovery process. Billing will be displayed in the currency relevant to your subscription and covers only the resources that the Azure API provides billing information about them. Resources that are not covered by the sensor are shown as "Missing Resources".

✓ .	Sensor Azure ^{ок}	Billing [⊨] ★★★	r☆☆					
_	Overview	(•)) Live Data	2 days	30 days	365 days	Historic Data	🔳 Log	🌣 Sel
	Subscription Billi	ng		Missing Resource	Types	Missing Resources		
				0 types	✓ ↓	0 resources		
				Resource Types		Resources		
			X	13 types	0.1	26 resources		
	11 USD	0	11 USD					

8.19 Azure Application Insights

Application Insights, a feature of <u>Azure Monitor</u>, is an extensible Application Performance Management (APM) service for developers and DevOps professionals. Use it to monitor your live applications. The AutoMonX Azure Sensor Pack can help you to Auto-discover and monitor Azure Insights.

<i>nsor</i> Function - time ок	triggerGrp	- TimerTrigge	er2 [₽] ***☆☆				"	₽ ⊠ 2
(•) Overview Live Data	2 days	30 365 days days	Historic Data	E Log	¢ Settings	A Notification Triggers	Q Comments	Histor
Last Scan: 111 s Coverage: 100%	Last Up: 111 s Sensor Type EXE/Script /	: Advanced	Last Down: Dependency: Parent	Uptii 100. Inter 5 m	me: 0000% val:	Downtim 0.0000% ID: #2590	ie:	0
ast Request Status		Memory - Avail 743,324 KByte Process - Priva 234,360 KByte	able Bytes te Bytes te Bytes Call and the b	e Processor Time	e Proces	s • % Processor Time_ Proce	te	o
Channel -	ID \$	Last Value ≑		Minimum 🗘		Maximun	n ¢	
Channel - Downtime Last Request Status	ID \$ -4 2	Last Value ≑		Minimum 🍣		Maximun	n ≎ 100	•



8.20 AutoMonX License

The License sensor is an additional sensor type aimed at monitoring the AutoMonX licensing and maintenance status. This sensor will let you know if your license or maintenance period is about to expire and helps you to renew it on time. This will help you to enjoy continues monitoring and prepare for renewal, without the license ending unexpectedly. It also includes the memory usage of AutoMonX.



8.21 Azure Logic App

Azure Logic App is a cloud service that helps you schedule, automate, and orchestrate tasks, business processes, and workflows when you need to integrate apps, data, systems, and services across enterprises or organizations. Below is a screenshot of typical metrics you will get in PRTG:





8.22 Azure Disks

Azure makes available several disk types for attaching them to virtual machines for usage such as OS or Data disks. The sensor below provides the metrics available for such disks:



8.23 Azure Notification Hubs

Azure Notification Hubs is a massively scalable mobile push notification engine for quickly sending millions of notifications to iOS, Android, Windows, or Kindle devices, working with APNs (Apple Push Notification service), GCM (Google Cloud Messaging), WNS (Windows Push Notification Service), MPNS (Microsoft Push Notification Service), and more.





8.24 Azure Network Interfaces

A network interface enables an Azure Virtual Machine to communicate with internet, Azure, and on-premises resources.



8.25 Azure ExpressRoute Connections

ExpressRoute lets you extend your on-premises networks into the Microsoft cloud over a private connection with the help of a connectivity provider.





8.26 Azure VPN Gateways Metrics

A VPN gateway is a specific type of virtual network gateway that is used to send encrypted traffic between an Azure virtual network and an on-premises location over the public Internet. You can also use a VPN gateway to send encrypted traffic between Azure virtual networks over the Microsoft network. Each virtual network can have only one VPN gateway. However, you can create multiple connections to the same VPN gateway. When you create multiple connections to the same VPN gateway, all VPN tunnels share the available gateway bandwidth. The Azure Sensor Pack can monitor various VPN Connection types such as: S2S (Site to Site), P2S (Point to Site) and M2S (Multi-site) connections.



8.27 Azure Backup Jobs

The Backup Job sensor monitors the status of the backup jobs under every Recovery Service Vault. By default, the AutoMonX Azure Sensor Pack supports the daily backup. If you have a weekly backup configured, please add the following configuration in the file "AutoMonX_AzureSensor.ini": BACKUP_JOB_VAULT_<*vault_name>_*HOUR=168 And restart the service for the changes to take effect.





8.28 Azure App Secret Key and/or Certificate Status

The Azure Apps Secret Key is used for authenticating with the Azure API for a specific Azure App (App Registrations). Each secret key has an expiry date. After the secret key has expired it can no longer be used. The Azure App Secret Key sensor displays the remaining time until its expiry date. The Certificates for the App will also appear as a channel in this sensor

To make the Azure Secret Keys discoverable and available for monitoring, you need to follow the procedure in <u>Chapter 14.1</u>.





8.29 Azure Kubernetes Services (AKS)

Starting with release 4.0.15, The Azure Sensor Pack can discover and monitor the Azure Kubernetes Services (AKS). If the sensors are not discovered, enable the insights monitors (follow <u>this guide</u>). Four new sensor types were developed to support AKS:

- Azure Kubernetes Cluster Metrics
- Azure Kubernetes Deployments
- Azure Kubernetes Internal Deployments

8.30 AKS Cluster Metrics

The Azure Kubernetes Metrics sensor shows the general resource consumption of the entire AKS cluster (average usage figures)



8.31 AKS Deployments

The Azure Kubernetes Deployments sensor shows per each deployment the number of running pods (pod is a core building block of a deployment managed by Kubernetes) in the AKS cluster.





8.32 AKS Internal Deployments

The Azure Kubernetes Internal Deployments sensor shows the Azure-specific (internal) Kubernetes deployments and their pod count.



8.33 Azure VM Multi-Disk

After activating Insights for a VM, available disk space and size can be monitored through the Log Analytics API. Please follow <u>this guide</u> to activate this in the Azure portal.



8.34 Azure Storage File Share

This sensor shows the usage of your file shares.





8.35 Azure Service Issues

Azure Service Issues sensor monitors the active issues and events per Azure subscription per region as seen in Service Health in the Azure portal. We recommend setting your own limits.



8.36 Azure Planned Maintenance

The Azure Planned Maintenance sensor monitors the Azure-planned maintenance that my impact your resources. It shows the Azure regions and the number of planned maintenances in each region.





8.37 Azure Health Advisories

Azure Health Advisories monitors the number of Azure advisories per the geographic locations where your Azure resources are located.



8.38 Azure Advisor

Per subscription Advisor score as seen in Azure portal, showing each category score to assess how well-architected your workloads are.





8.39 Azure Quotas

The Azure Quotas sensor monitors the number of resources that have reached their quota per resource type (i.e Compute, Storage, Vmware, Networking etc). The quota limit is configured in an INI parameter AZURE_QUOTA_LIMIT in AutoMonX_AzureSensor.ini. Regions with no usage will not be displayed.



8.40 Azure Host Pool

Monitors the Virtual Desktop Hosts under Host Pools.





8.41 Azure Reservations

This sensor is created per Azure subscription. It measures the percentage of the subscriptions' usage of the reservations you have set and the reservation expiry time.

To discover this resource, you must add a permission to the service principle as described in <u>appendix B 2</u>.

Note - Archived reservations are not discovered.



8.42 Azure Alerts

Allows a general overview of the number of active alerts by their severity and their short description. It allows you to quickly get notified when something major happens in your Azure estate.





8.43 AKS Upgrade Profiles

Allows to monitor the version of your AKS deployment and generate an alert when a new software upgrade is available. Preview versions will not trigger an alert.



8.44 Azure Service Bus Topic Subscriptions

This sensor extends our Azure Service Bus sensor. It adds the ability to monitor the messages counts of each subscription to a Service Bus topic.





8.45 Azure Firewalls

Azure Firewall is a cloud-native and intelligent network firewall security service that provides the best of breed threat protection for your cloud workloads running in Azure. This sensor collects the metrics available for Azure firewalls and its health state.



8.46 Azure Application Gateway (WAF)

Azure Web Application Firewall is a cloud-native service that protects web apps from common web-hacking techniques such as SQL injection and security vulnerabilities such as cross-site scripting. This sensor provides the vital metrics of the Azure WAF





8.47 Azure (VPN) Connections

Site to Site and VNet to VNet VPN Gateway Connection metrics & status.







8.48 IP Address

The Azure IP Address sensor provides an in-depth view of the public Azure IP Address metrics.



8.49 Azure Load Balancer

Load balancer distributes inbound flows that arrive at the load balancer's front end to backend pool instances. The backend pool instances can be Azure Virtual Machines or instances in a Virtual Machine Scale Set. The Azure Load Balancer sensors provides some interesting performance metrics to monitor its functionality.





8.50 Azure Network Watcher

Azure Network Watcher provides a suite of tools to monitor, diagnose, view metrics, and enable or disable logs for Azure IaaS (Infrastructure-as-a-Service) resources.



8.51 Azure Log Analytics Custom Sensor

Custom sensor created by your own KQL query in Azure Log Analytics. The sensor configuration is explained in <u>Appendix D</u>.



8.52 Azure Kubernetes combined sensor

For Clients that need to optimize sensor use, this sensor combines all AKS specific sensors listed above. To activate this sensor discovery the following configuration must be added to the file "AutoMonX_AzureSensor.ini"

COMBINE_AKS_SENSORS=TRUE



Overview	((•)) Live Data	2	days	30 days	365 days	📥 Historic Data	🗏 Log 🌼 S
ama-logs-rs			ama-logs-agent		azure-ip-masq-agent	cloud-node-manager	Cluster Available Major Versi
			1 Pods		1 Pods	a 1 Pods	0#
			Cluster Upgrade I	Profiles	coredns-autoscaler	csi-azuredisk-node	csi-azurefile-node
			Warning	U a a	1 Pods	a 1 Pods	1 Pods
		x	Current Cluster V	ersion	konnectivity-agent	kube-dns	kube-proxy
1 Pods	0 Pods	1 Pods	0 #	V	2 Pods	2 Pods	1 Pods
Max Region Version	metrics-server		Node 1 Last Resp	oonded	Node Count	Node Pool Available Major Ve	Node Pool Upgrade Profiles
(\	0 Dada		40 Casanda		1 Nodos	0.#	Warning

8.53 Azure SharePoint Statistics

Displays the usage statistics for your active sites as seen in the SharePoint Admin center.

To allow the Application to discover the SharePoint metrics, please add the Reports.Read.All API Permission for the Application (Similarly to the process explained in <u>Chapter 14.1</u>)





8.54 Azure Enterprise Secrets and Certificates

The Azure App Secret Keys and Certificates are used for authenticating with the Azure API for a specific Azure App (Enterprise Application that do not appear in App Registrations). Each key has an expiry date. After the secret key has expired it can no longer be used. The Azure App Enterprise Key sensor displays the remaining time until its expiry date.

To make the Azure Secret Keys discoverable and available for monitoring, you need to follow the procedure in <u>Chapter 14.1</u>.



8.55 Azure SAML Enterprise Certificates

The Azure SAML (Security Assertion Markup) certificates are used for authenticating with the Azure Entra to establish federated single sign-on (SSO) to your software as a service (SaaS) Enterprise Applications. Each Azure SAML Certificate sensor can monitor a single application.

Each SAML certificate has an expiry date. After the SAML certificate has expired it can no longer be used. The Azure SAML Certificates sensor displays the status of the certificate and the remaining time until its expiry date.





9 Troubleshooting

9.1 Troubleshooting the Azure Sensor pack Installation

Problem Description	Troubleshooting Steps
Azure Sensor pack Service is not starting	 Run Check Config via the UI (as administrator), check the results and fix any problems. Refer to <u>Troubleshooting the Azure Configuration</u> Make sure your Azure User is set up OK. Refer to <u>Troubleshooting the Azure Connection Error</u> Make sure the PRTG probe is open to the Internet and can access Microsoft Azure Make sure that the Product Key is valid Use the service debug mode to check service errors.
	Refer to <u>Debug Using Service Debug Mode</u>
results	 Make sure that the Azure Sensor Pack service is running Make sure your Azure User has enough permissions to the desired subscription. Refer to <u>Troubleshooting the Azure</u> <u>Connection Permission</u> Submit a support request via <u>support@automonx.com</u> and send the following log files: Refer to <u>Sending the Discovery Files to the support team.</u>
Discovery provides partial results	 Make sure your Azure User has enough permissions to the desired subscription. Refer to <u>Troubleshooting the Azure</u> <u>Connection Permission</u> Make sure Diagnostics is enabled for the desired resource. Refer to <u>Troubleshooting Missing Azure Resource Metrics</u> Submit a support request via <u>support@automonx.com</u> and send the following log files: Refer to <u>Sending the Discovery Files to the support team.</u>
Discovery is not able to discover the Azure Billing resources	Make sure your Azure User has enough permissions to the desired subscription. Refer to <u>Troubleshooting the Azure</u> <u>Connection Permission</u>
Discovery is not able to discover Azure App metrics sensors	Make sure Diagnostics is enabled for the desired resource.



	Refer to Troubleshooting Missing Azure Resource Metrics
Discovery is not able to discover	Make sure service health is enabled for all the Azure resources
Azure Service Health metrics sensors	Refer to Troubleshooting Missing Service Health
Azure Sensors are Down with	Make sure that the Azure connection parameters are correct (use
error message: The AutoMonX	the AutoMonX UI and run Config Check
Azure service is down, cannot connect	Make sure that Azure is not blocked by a proxy server or Firewall of your organization



9.2 Troubleshooting the Azure Sensor Configuration

In order to analyze the status of the connection to Azure and the Azure Sensor pack configuration, use one of the following options:

- AutoMonX Configuration UI
- Config Check command line utility

These tools perform various checks of the Azure Sensor Pack configuration, its service and the connection to Azure and displays vital information that may assist in checking for issues. Through the Automation UI "Config Check" Button:

AutoMonX The Montoring Automation Compare AutoMonX Discovery And Automation For PRTG							
Settings Discovery PRTG Group Set	ettings PRTG Group Settings Add sen	sors to PRTG					
	Azure Discovery Settings						
Product:	Azure 🗸						
App ID:	38888db-8888-8888-8888-88882						
Secret Key:	888888888888888888888888888888888888888						
Tenant:	8888ddb-8888-8888-8888-888891						
Subscription:	All	II All					
Offer Id:		Billing					
Currency:							
Country Code:							
Threads:	5 ¥ 🛈						
Config Check Apply Discover Discover Insights							
IRights Reserved [©] AutoMonX Ltd 2019 - V1.9.5		Back Next					

Through command line: AutoMonX_AzureCollector.exe -config_check



Below is an example of a successful configuration check:



The Azure sensor was able to connect to Azure using the supplied information, the service is up and running and subscriptions were found in the Azure account.



9.3 Troubleshooting Azure Discovery Connection Errors

In case of a failed connection to Azure, an error will be displayed in the messages area of the Configuration UI.

Possible causes can be that Azure is unreachable due to limitations of network access, incorrect connection information such as App ID, Secret Key, or Tenant ID.

Check the Azure sensor settings using the instructions in <u>Section 6.3</u>.

9.4 Troubleshooting Azure Discovery - Permissions

When no subscriptions are found during discovery, it means that the current credentials set doesn't have permissions to view any subscriptions and their resources under your Azure account.

Please assign at least a **Reader** role for the connection ID on the subscriptions you want to monitor. Please refer to <u>Microsoft Azure - assign-the-application-to-a-role</u>

9.5 Collecting the Discovery Files for AutoMonX Support

In case of any other problems encountered during Azure discovery, open a case with our support team at support@automonx.com. You would need to provide the following information:

- Discovery log file AutoMonX_Discovery_out.log
- Discovery results in a form of CSV files with names such as: AutoMonX_Azure_automation_<Sub>.csv
- The Azure types file: AutoMonX_Azure_types.dat

These files are located in the following directory:

Program Files (x86)\PRTG Network Monitor\Custom Sensors\EXEXML\AutoMonX\Azure

• The output of the following command:

AutoMonX_AzureCollector.exe -discovery -sub -All



9.6 Troubleshooting the Discovery of Azure Metrics

When the discovery process can't discover one or more Azure resource's App Metrics data, make sure that the Azure diagnostics was enabled for this resource.

To enable the resource diagnostics, navigate to the Azure resource settings page in the Azure admin portal.

On the left side of the resource page, click on Diagnostics. Setup the metrics collections settings and make sure they are turned on (enabled)

Microsoft Azure		𝒫 Search resources, services, and docs	∑ 4 ⊊ ¢ ² ⊗ ?
	Home > Myres - Diagnostics		
+ Create a resource	MEMSQL - Diagnostics		
🛧 Home	Virtual machine (classic)	Sava Y Discard	
🖪 Dashboard	O Search (Ctrl+/)	Jare Distard	
i≡ All services	😂 Disks 🔨		
- 🛨 FAVORITES	Security		
All resources	Endpoints	* Storage Account	
📦 Resource groups	Extensions	spineinsqusig	
🔇 App Services	IP addresses	Basic metrics	1h, 1m
👼 SQL databases	💠 Load balanced sets	Boot diagnostics	
📬 SQL data warehouses	Migrate to ARM		
🬌 Azure Cosmos DB	Network security group		
👰 Virtual machines	🧕 Size		
💠 Load balancers	ackup 🕈		
🧾 Storage accounts	11 Properties		
Virtual networks	Locks		
🚸 Azure Active Directory	Monitoring		
🖰 Monitor	Diagnostics		
🔷 Advisor	Advisor recommendations		

9.7 Troubleshooting the Discovery of Azure Service Health

When the discovery process can't discover the Azure Service Health sensors, make sure service health was enabled in the Azure portal for these resources in the link below:

https://portal.azure.com/#blade/Microsoft_Azure_Health/AzureHealthBr owseBlade/resourceHealth



9.8 Collecting Azure Service Debug information

To activate the service debugger, you would need to set the SERVICE_DEBUG variable to 1 (default is 0) in the AutoMonX_AzureSensor.ini file. This setting will activate the service debug mode upon the next start of the service.

During debug mode, a special log file is created. This file tracks all the Azure sensor service operations. This file needs to be examined by the AutomonX support team to detect any issues. Open a case with our support team at support@automonx.com. You would need to provide the following file:

• Service Debugger file - AutoMonX_ServiceDebugLogger

The files are in the following directory:

C:\Program Files (x86)\PRTG Network Monitor\Custom Sensors\EXEXML\AutoMonX\Azure\Logs

9.9 Collecting Azure Sensor Debug Information

To activate the debug logs of the Azure sensors, add the -debug argument to the parameters of the AutoMonx_AzureSensor.cmd in the PRTG sensor settings as seen below:

Sensor Settings	
	The EXE file has to run on the computer where the parent probe is installed, not on the par directory for EXE files is the probe directoryvbs files, .ps1 files, or other script files may us directories.
EXE/Script	Automonx_AzureSensor.cmd
Parameters ⁽¹⁾	-type Microsoft.Logic/workflows -resgrp mylogicapps -res "firstlogicapp" -sub "mpn" -debug
Environment	Default Environment
	O Set placeholders as environment values

Please note – when the sensor is of type HTTP, spaces will cause an error – you must add %20 instead:



Basic Sensor Settings	Sensor Name 🖲	Azure Billing
	Parent Tags 🖲	
	Tags 🕄	httpsensor 🗙 O
	Priority ^①	黄黄黄 ☆☆
HTTP Specific	Timeout (Sec.) 🖲	900
	URL [®]	http://127.0.0.1:8092/get?exeline=-sub%20°microsoft%20partner%20network*%20-cons <mark>%20-debug</mark>

This setting will activate the sensor debug mode upon the next run of the sensor. The Azure sensor debug logs are created in the default PRTG sensors log directory. You can change the location of these logs by modifying the following variable in the AutoMonX_AzureSensor.ini file:

DEBUG_LOG_DIR=C:\Temp

Typically, the log file name will look like the example below: *AutoMonX_Azure_Microsoft.DocumentDb_databaseAccounts_res_prod.log*

Please note – make sure to keep track of the sensors you activated debug for to revert after completion. Otherwise, the log file will keep growing and take up space on your probe machine.

9.10 Collecting In-Depth Azure Sensor Debug Information

To activate the in-depth debug logs of the Azure sensors, add the following parameters to the AutoMonx_AzureSensor.cmd in the PRTG sensor settings.

-debug verbose

This setting will activate the sensor in-depth debug mode upon the next run of the sensor. The Azure sensor debug logs are created in the default PRTG sensors log directory. You can change the location of these logs by modifying the following variable in the AutoMonX_AzureSensor.ini file:

DEBUG_LOG_DIR=C:\Temp





10 Command Line Options (CLI)

10.1 The Azure Sensor Pack Command Line Options Reference

Option	Details
-install	Installs the AutoMonX Azure Sensor Service. The service communicates with the Azure environments, retrieves and stores the Azure communication token.
<pre>-config_check [-tenant <tenant_number>]</tenant_number></pre>	Checks the service communication to the Microsoft Azure environment.
-discovery -sub { <subscription_name> -All}</subscription_name>	Discovers all the resources in an AZURE environment and creates a report in a HTML
-discovery -sub SubName [-tenant	format. The report is AutoMonX_Azure_Report-
<tenant_number>] [-outmode] [-whitelist] [-</tenant_number>	_{.html in the PRTG EXEXML}
resource_id] [-verbose]	program folder.
-discovery -sub SubName -resources	Use the resources flag or the metrics for a quicker discovery, that doesn't get all metrics.
-discovery -sub Subivarile -metrics	flags will perform the thorough discovery.
-discovery -sub SubName -new_resources	
-create_conn_profile -tenant_label	For multi-tenant users. Creates a new
<display_label> -azure_appid <app_id> -</app_id></display_label>	credentials set in the file
azure_secretkey <secret_key> -</secret_key>	"AzureConnProfiles.ini".
azure_tenantid <tenant_id> [-encrypt_secret]</tenant_id>	
<pre>-remove_tenant -tenant <tenant_number></tenant_number></pre>	For multi-tenant users. Deletes all information related to a specific tenant. Use with caution !
-install_backend	Installs a complementary service for REST API
	access to configuration and discovery.
-get_tenant_id -tenant_label <display_label> /</display_label>	Given a tenant label or full tenant id returns the
-azure_tenantid <tenant_id></tenant_id>	configured internal tenant id.
-status_discovery	Activate with the same parameters as the discovery to get a progress status for the discovery process.
-update_key -tenant_label <display_label> -</display_label>	Updates existing App Key. To be used before
azure_secretkey <secret_key> [- encrypt_secret]</secret_key>	the key expires. Will accept tenant_label, azure_tenantid or tenant (internal id)



-type <res_type> -resgrp <res_group> -res <res_name> -sub <res_subscription> [-debug] -type Azure/Type -resgrp ResGroup -res ResName -sub SubName</res_subscription></res_name></res_group></res_type>	These are the arguments required for the Sensor Pack to monitor a resource in the Microsoft Azure environment. This is only used with the AutoMonX_AzureSensorRun executable.
-cons -sub SubName	These are the arguments required for the Sensor Pack to monitor the billing of a subscription. This is only used with the AutoMonX_AzureSensorRun executable.
-licstatus	These are the arguments required for the Sensor Pack to monitor the AutoMonX License status. This is only used with the AutoMonX_AzureSensorRun executable.
-enc_data <secret_key></secret_key>	Encrypts a given string. You can use it to encrypt the Azure secret key to safely store in the credentials file.
-ovl_map	Create a report of OVL values mapping
-version	Displays the program's version.
-help	Displays the command option list.

10.2 Fully Automated Azure Monitoring

Starting with version 4.1.7, fully automated monitoring of your Azure estate is available as part of the Azure Sensor pack. To deploy it, you need to adapt a simple batch file, that can run on a scheduled and non-interactive fashion and cover the entire cycle of automatically discovering, adding to monitoring and even deleting dead/orphan resources from monitoring.

10.2.1 Automated Discovery and Monitoring

Depending on the structure of your Azure estate (Tenants, subs etc) you can easily adapt the contributed batch file below and create a Windows scheduled task for periodic discovery and automatic addition of discovery results for monitoring.

An example is available in: "AutoMonX/Azure/Contrib/Amx_Discovery_Addition.cmd". Please provide the pass hash and group name as parameters to the script.

Amx_Discovery_Addition.cmd <passhash> <PRTG_group>



You may update the script to run on only specific subscription. Please make sure the scheduled task is run with highest privileges:

\bigcirc	Run	only	when	user is	s logge	d on
\sim		<i>j</i>				

Run whether user is logged on or not

Do not store password. The task will only have access to local computer resources.

Run with highest privileges

We recommend you add the -metrics flag to the discovery line after the initial run to shorten the discovery time.



10.2.2 Automated Clean-Up of Un-Needed Resources from Monitoring

USE WITH EXTRA CAUTION: Please note that deleting sensors is irreversible and will delete the sensors and all their historic data. This feature is bound to our EULA agreement. AutoMonX LTD will not be responsible for any damages direct or collateral due to usage of any of our products or their features. The deletion of sensors also deletes devices without any sensors and empty sub-groups to avoid phantom resources

This feature allows you to automatically delete any sensors in PRTG that no longer provide any useful monitoring data and seen in Down state. Typically, such state is seen because the Azure resource was deleted or shut down via the Azure Portal.

Make sure that the file "down_sensors_filter.ini" exists in the AutoMonX/Common folder. Otherwise copy it from the latest Azure Sensor Pack version zip archive and insert the message text of the sensors you wish to delete.

You can set a periodic sensor deletion of removed resources with a Windows scheduled task running a batch file. An example is available in "AutoMonX/Azure/Contrib/Amx_Sensor_Deletion.cmd". Please provide the pass hash and PRTG group(s) name(s) as parameters to the script.

Amx_Sensor_Deletion.cmd <passhash> <PRTG_group(s)>

If the group name contains spaces, make sure to add quotation marks enclosing the "group name". It can also be run on several groups, separated by commas (for example: Web,Logic,"Azure test").

By default, it will delete sensors in a Down state that display an error message that contains the text "Resource Not Found".

To adjust this functionality to delete sensors with other errors types, based on their last message text, you need to edit the *down_sensors_filter.ini* file.

Add new line of text per the relevant sensor error message(s) as seen in PRTG, one per each line. Make sure to be as specific as possible.

The auto-deletion option can be used to clean-up any PRTG sensor types.



10.2.3 Automatically Pausing Un-Needed Resources

This feature allows you to automatically pause any sensor in PRTG that no longer provides any useful monitoring data and seen in Down state. Typically, such state is seen because the Azure resource has been deleted or shut down via the Azure Portal. It is useful for NOC teams that wish to investigate the source of the problem rather than automatically delete the sensor(s).

Make sure that the file "pause_sensors_filter.ini" exists in the AutoMonX/Common folder. Otherwise copy it from the latest Azure Sensor Pack version zip archive and insert the relevant messages of the sensors you wish to pause.

You can set a periodic sensor pausing of removed resources with a Windows scheduled task running a batch file. Example:

AutoMonX_PRTG_Automation.exe -pause_sensors -p 11111111 -grouplist AutoMonX_Azure

If the group name contains spaces, make sure to add quotation marks enclosing the "group name". It can also be run on several groups, separated by commas (for example: Web,Logic,"Azure test").

By default, it will pause sensors in a Down state that display an error message that contains the text "Resource Not Found".

To adjust this functionality to pause sensors with other errors types, based on their last message text, you need to edit the *pause_sensors_filter.ini* file.

Add new line of text per the relevant sensor error message(s) as seen in PRTG, one per each line. Make sure to be as specific as possible.

The auto-pause feature can be used to pause any PRTG sensor types.

10.2.3.1 Automatically Setting Removed Resources to Warning State As an alternative to pausing the sensors, you may activate automatic setting of the sensor to Warning in PRTG (instead of Down) for Azure resources that have been deleted.

Update the following configuration value in *AutoMonX_AzureSensor.ini* file and restart the service Azure Sensor Pack service:

SET_SENSOR_NOT_FOUND_TO_WARN=true

This will not require setting up any additional scheduled tasks but will only work for removed resources and not for user-specified phrases.


10.2.4 Automated Inclusion/Exclusion of Sensors and Channels

Most of the Azure resources contain multitude of monitoring information which yields lots of channels in PRTG, some of them are less useful for certain IT teams. Another frequently seen scenario is when the Azure monitoring teams wish to add only specific resource types to PRTG (i.e. only the production Databases). The purpose of this feature is to allow fine granularity of which Azure Resources and metrics you want to add to PRTG.

This section explains how to utilize the Exclude and Include Functions of AutoMonX Azure Sensor Pack to control the addition of Azure Resource Groups, Azure Resources (devices), Azure sensors and Azure Metrics (channels) to PRTG. This functionality can replace the selection of sensors in the AutoMonX UI and allow full automation of the discovery and monitoring automation.

Important:

1. Make sure to run full discovery before applying any configuration to the filter files

2. The discovery process creates a csv file for each Azure Subscription with all discovered Resources, their sensors and channels in a format of the *include* and *exclude* files. Use this file to create your include or exclude filters. Typically, the name of the file would like "*AutoMonXVAzureVLogsVAutoMonX_Channel_Report-*<*sub>.csv*".

3. Monitoring Automation will firstly process the *Include* file and then the *Exclude* file.

4. Edit the *exclude_mon.csv* and *include_mon.csv* files only with a simple text editor such as Notepad or Notepad++. Don't save these files in a non-textual format such as XLX or XLXS.

10.2.4.1 Excluding an Azure Resource Group from Monitoring Under the <u>PRTG Group</u> column, enter the Azure Resource group you wish to exclude (i.e. Compute, Network, Web etc.).

The configuration below will exclude the group named Batch and any Azure Resources (devices) and their sensors below that:

PRTG Group	DeviceType_Category	SensorName	ChannelsBlackList	TagName	TagValue
Batch	any	any	any	any	any



10.2.4.2 Excluding an Azure Resource

Under the <u>DeviceType_Category</u> column enter the type of the Azure Resource (device) you wish to exclude. Use the resource type located in parentheses (i.e. *VirtualMachines* if the Azure Resource name *is Windowstest_(VirtualMachines)*), or part of the device name.

The configuration below will exclude Azure Resources of type Vaults under the group RecoveryServices

PRTG Group	DeviceType_Category	SensorName	ChannelsBlackList	TagName	TagValue
RecoveryServices	Vaults	any	any	any	any

10.2.4.3 Excluding an Azure Sensor Type

In order to exclude specific Azure sensor types, specify the exact sensor type under the <u>SensorName</u> column. Entering "any" in this column will result in exclusion of all sensor types under this Azure Resource type (device). Which would effectively not add this Azure resource to PRTG.

The configuration below will not add sensors of type "Azure Service Health" for Azure resources (devices) of the type ServerFarms under the group Web.

 PRTG Group
 DeviceType_Category
 SensorName
 ChannelsBlackList
 TagName
 TagValue

 Web
 ServerFarms
 Azure Service Health
 any
 any
 any

10.2.4.4 Excluding Azure Metrics (Channels)

In order to exclude specific Azure metrics (channels), add the full channel name under the <u>Channels</u> column separated by the sign "~". Another option is to specify a single Azure Metric per line.

The configuration below excludes the "OS Disk Max Burst IOPS" and "Disk Read Bytes" metrics for all VirtualMachines.

 PRTG Group
 DeviceType_Category
 SensorName
 ChannelsBlackList
 TagName
 TagValue

 Compute
 VirtualMachines
 Azure App Metrics
 OS Disk Max Burst IOPS~Disk Read Bytes
 any
 any

Azure Tags Include/Exclude

10.2.4.5 Excluding by Azure Tags

If you utilize Azure Tags in your Azure estate, you can leverage them by adding the Azure Tag under the column TagName and its value under TagValue in order to exclude any resources marked with this tag. Otherwise leave empty.



Important – Only Azure Tags that are assigned directly to Azure resources in the Azure Portal can be used for filtering. Azure Tags set on a Resource group level will be ignored.

The configuration below excludes the all resources with Azure Tag and Value pair MonitorWithPRTG:FALSE

PRTG Group	DeviceType_Category	SensorName	ChannelsBlackList	TagName	TagValue
any	any	any	any	MonitorWithPRTG	FALSE

10.2.4.6 Further Information and Troubleshooting of Exclusion

You may find that the **include** functionality is more suitable for your needs. Make sure to configure the file *include_mon.csv*, in ways like explained in the paragraphs above. You would be able to add specific Groups of Azure Resources, Azure Resources and Azure Metrics to PRTG. Both exclude and include functions can be used in conjunction to tailor the automation for your needs.

Important:

1. Partial names may not be filtered correctly, except where stated.

2. The "Sensor Health" channel cannot be excluded

3. Make sure not to list the same sensors in both files – this might result in an unwanted behavior.

4. Excluding channels may result in gray channels, 0 value channels or sensors shown in error, this by design in PRTG. Please delete and re-add the sensors via the AutoMonX UI or the <u>PRTG monitoring automation CLI</u>.

Debugging the discovery process:

For Quicker discovery you can apply the Include/Exclude policies to the discovery itself, as described in <u>section 7.15</u>. Refer to the file "AutoMonX_Discovery_out.log" under the folder Azure/Logs. If a resource was filtered by either policy, you will see the lines:

Device: MyServer_(SqlDatabases) filtered by blacklist

Device: MyServer_(SqlDatabases) filtered by whitelist



Debugging addition to PRTG:

Refer to the file "AutoMonX_Automation_Progress_out.old.log" under the Common folder. In it you will see the complete PRTG Addition logs, where in the top of the file you will see the number of filters applied, and if the sesnors passed of failed the include_mon policy. Example:

Notice: 3 Filters are applied via C:/Program Files (x86)/PRTG Network Monitor/Custom Sensors/EXEXML/AutoMonX/Common\include_mon.csv file

Found device: Dev:Name:AutoMonX_LicStatus

dev:name:automonx_licstatus - Removed by Whitelist

For each Group, Device and Sensor that pass the Whitelist, you will see if it was successfully added to PRTG or excluded. Example:

GroupName:Web - DeviceName:MyTest(Sites) - SensorName:Azure App Metrics - This Sensor would be skipped by the **exclude_mon.csv policy at line:2**

Skipping Sensor Azure App Metrics by Blacklist exclusion

Debugging Channel Exclusion:

When debugging a sensor, you will see in the generated debug file the excluded channels. Read more about sensor debugging in <u>section 9.9</u>.

In the folder "Common" you can find the file "exclude_mon – Example.csv" with the examples provided previously.



10.2.5 Automated Scan-Now Functionality

In some PRTG versions, we have noticed certain functionality issues for some types of custom sensors. The most frequently observed are the "Undefined lookup value" situations and sometimes "out of range" values (i.e. 10000% values). The solution we found for these issues was running a manual rescan of the sensor. Obviously, it is not a scalable solution for large environments, and this is the primary reason why this feature was introduced.

10.2.5.1 Automatic scan-now functionality

To allow automatic sensor scanning, add the desired groups names to check in the "AutoMonX_PRTG_Automation.ini" file:

[RESCAN] PRTG_GROUPS=AutoMonX_Azure

The feature is activated automatically by the Azure Sensor pack every 20 minutes and checks if new sensors were added. It will go over the PRTG groups configured in the INI file and look for sensors in Down or Warning states that have specific last message text values. Then, the feature will perform an automatic re-scan of these sensors in batches of 5 sensors per minute to avoid lags in the PRTG core.

10.2.5.2 Manually activating the scan-now functionality:

Delete the file "sensors_not_to_scan.txt" if exists.

AMX_PRTG_senors_issues.exe -grouplist AutoMonX_Azure,"Azure test"



10.2.6 Automated Addition and Removal of Tenants

For customers who manage large number of tenants such as MSPs or CSPs, we have added a fully automated CLI support for adding and removing tenants.

10.2.6.1 Adding a new tenant:

The command below creates a new Azure Tenant connection profile in the file "AzureConnProfiles.ini".

AutoMonX_AzureCollector.exe -create_conn_profile -tenant_label <display_label> azure_appid <app_id> -azure_secretkey <secret_key> -azure_tenantid <tenant_id>

After adding a new Azure Tenant, run a configuration check so that the Azure Sensor Pack service will immediately pick-up the Azure API token for the new tenant, after which you can start the discovery process:

AutoMonX_AzureCollector.exe -config_check

10.2.6.2 Deleting a tenant

The command below will delete the Tenant connection profile, log files and historical data related to the specified tenant. It does not delete the PRTG sensors and groups – This should be done prior, because this action will not delete the tenant if there are active sensors in it. **Use with outmost care!**

AutoMonX_AzureCollector.exe -remove_tenant -tenant 1



11 REST API Service

To fully automate the AutoMonX Azure Sensor Pack, in version 4.2.14 and later, a new API service was introduced. The main functions of the REST API service are to initiate auto-discovery and activate the monitoring automation to add newly discovered Azure resources to PRTG. Additionally, adding and removing Tenants actions are also available.

The service can be installed via the Installer under additional services. For manual installation of the API Service use the following command:

AutoMonX_AzureCollector.exe -install_backend

The API service is listening on TCP Port 8075 only on the localhost IP address. The service will not accept any attempts to access it from outside the machine it was installed on.

To list all supported API options, use any web browser and access the following URL: <u>http://127.0.0.1:8075/api</u>

When running the PRTG Probe on an Azure Virtual Machine, you can activate these options remotely through the Azure Portal run-command feature. Example:

az vm run-command create --name "AutoMonX_Discovery" --vm-name "PRTG_Probe" --resource-group "Test" --script "Invoke-RestMethod -Uri 'http://127.0.0.1:8075/api/discovery/start?tenant_id=1&outmode'"

11.1 Function get_tenant_id

Provides the Azure tenant internal id as written in the AzureProfiles.ini file. You need to provide either the tenant label or the full tenant id. Example: <u>http://127.0.0.1:8075/api/get_tenant_id?tenant_label=MyTenant</u>

Param	Required	Description
tenant_label		Name of the tenant as saved in the file
		"AzureConnProfiles.ini".
azure_tenantid		Full ID of the tenant as appears in Azure Portal
		and saved in the file "AzureConnProfiles.ini".

Response on success:

The number of the tenant (i.e. 2)

Response on failure:

- No tenant with the name MyTenant
- Failed getting tenant id

Web : http://www.automonx.com Email : support@automonx.com



• Missing tenant_label or azure_tenantid parameter

11.2 Function Initiate Discovery via API

Starts a discovery process for a tenant. Example: http://127.0.0.1:8075/api/discovery/start?tenant_id=1&outmode

Param	Required	Description
tenant_id	\checkmark	The tenant id as returned in the previous API.
sub=subcription		The subscription name.
outmode		When passed will generate a log file.
resources		Quick discovery of resources.
metrics		Quick discovery of metrics.
whitelist		Quick discovery with Inclusion/Exclusion.
resource_id		Adds the resource URL to the comments in
		PRTG
add_comments		Adds the resource URL to the comments in
		PRTG for existing sensors

Response on success:

Initiated: "C:\Program Files (x86)\PRTG Network Monitor\Custom Sensors\EXEXML\AutoMonX\Azure\AutoMonX_AzureCollector.exe" -discovery tenant 1 -sub "-All" -outmode

Response on failure:

Missing tenant_id param

11.3 Function add_tenant

This function adds a new tenant. Example:

http://127.0.0.1:8075/api/add_tenant?tenant_label=NewTenant&azure_tenantid= xxxx&azure_appid=xxxx&azure_secretkey=xxxx

Param	Required	Description
azure_tenant_id	\checkmark	The full tenant id as appears in Azure Portal.
tenant_label	\checkmark	The name of the tenant to be saved.
azure_appid	\checkmark	The App ID.
azure_secretkey	\checkmark	The Secret Key.
encrypt_secret		When passed will encrypt the secret key in
		the file.

Response on success:

Initiated: "C:\Program Files (x86)\PRTG Network Monitor\Custom Sensors\EXEXML\AutoMonX\Azure\AutoMonX_AzureCollector.exe" -

AutoMonX Ltd © 2024 All Rights Reserved

Web : http://www.automonx.com Email : support@automonx.com



create_conn_profile -azure_tenantid xxxx -tenant_label "NewTenant" - azure_appid xxxx -azure_secretkey xxxx

Response on failure:

- Missing azure_tenantid parameter
- Missing tenant_label parameter
- Missing azure_appid parameter
- Missing azure_secretkey parameter

11.4 Function remove_tenant

Removes all data related to a tenant. Use with care!

Example:

http://127.0.0.1:8075/api/remove_tenant?tenant_id=2

Param	Required	Description
tenant_id	\checkmark	The internal tenant id

Response on success:

Tenant 4 removed successfully.

Response on failure:

- Missing tenant_id parameter
- Failed getting command status
- Error: Deletion Failed. Active sensors use this tenant

11.5 Function to check Auto discovery status

Gets the status of a specific discovery process. Example: http://127.0.0.1:8075/api/discovery/status?tenant_id=1&metrics

Param	Required	Description
tenant_id	\checkmark	The internal tenant id
sub=subcription		The subscription name.
outmode		When passed will generate a log file.
resources		Quick discovery of resources.
metrics		Quick discovery of metrics.

Response on success:

- Discovery is not running
- Discovery is still running

Response on failure:

- Missing tenant_id parameter
- Failed getting discovery status

AutoMonX Ltd © 2024 All Rights Reserved

Web : http://www.automonx.com Email : support@automonx.com



11.6 Function add_to_prtg

Initiates the monitoring automation that automatically adds groups, devices, and sensors in PRTG for a specific Azure tenant. Example:

http://127.0.0.1:8075/api/add_to_prtg?tenant_id=1&passhash=xxxx

Param	Required	Description
tenant_id	\checkmark	The internal tenant id.
group_name		The PRTG group name to add the sensors to
passhash		The passhash of the saved user to connect to
		PRTG. You can also store it on the server.
file=file_name		The exact file name you want to add to PRTG,
		from the file under the Data directory.

Response on success:

Initiated add to PRTG automation for tenant 1

Response on failure:

Missing tenant_id parameter

11.7 Function discover_and_add

Starts a discovery process for a tenant and after it is completed, initiates the automatic groups, devices, and sensor creation in PRTG for the tenant. Example:

http://127.0.0.1:8075/api/discover_and_add?tenant_id=1&metrics&passhash=xx xx&group_name=Azure_Sensors

Param	Required	Description
tenant_id	\checkmark	The internal tenant id.
sub=subcription		The subscription name.
outmode		When passed will generate a log file.
resources		Quick discovery of resources.
metrics		Quick discovery of metrics.
quick		Quick discovery of both resources & metrics.
new_resources		Discovers only new resources
whitelist		Quick discovery with Inclusion/Exclusion.
resource_id		Adds the resource URL to the comments in
add comments		Adds the resource URL to the comments in
		PRTG for existing sensors
group_name		The PRTG group name to add the sensors to
passhash		The passhash of the saved user to connect to
		PRTG. You can also store it on the server.



Response on success:

Initiated discover and add to PRTG for tenant 1

Response on failure:

Missing tenant_id parameter

11.8 Function update_key

Updates the secret key for a specific tenant. To be used before the key is expired. Example:

http://127.0.0.1:8075/api/update_key?tenant_label=OldTenant&azure_secretkey =xxxx

Param	Required	Description
tenant_label	\checkmark	The name of the tenant to be updated.
azure_secretkey	\checkmark	The new secret key.
encrypt_secret		When passed will encrypt the secret key in the
		file.
azure_tenantid		Full ID of the tenant as appears in Azure Portal
		and saved in the file "AzureConnProfiles.ini".
tenant_id		The internal tenant id.

Response on success:

Initiated: "C:\Program Files (x86)\PRTG Network Monitor\Custom Sensors\EXEXML\AutoMonX\Azure\AutoMonX_AzureCollector.exe" update_key -tenant_label "NewTenant" -azure_secretkey xxxx

Response on failure:

- Missing tenant_label, internal tenant id, or azure_tenantid parameter
- Missing azure_secretkey parameter

11.9 Allowing remote connection to API server

The API Server can receive requests from addresses configured in the file "AzureSensor_ACL.ini":

UPDATE_WIN_FW=TRUE ALLOWED_IP=8.8.8.8.8.8.4.4

Add all clients into the value of ALLOWED_IP. A relevant Windows Firewall rule will be created to allow the connection and will be updated with every service restart if you add clients to the list.

If you wish to manually update the Windows Firewall, change the value of UPDATE_WIN_FW to FALSE.



12 Monitoring the AutoMonX Azure Sensor Pack Processes

To provide our customers with an effortless experience using the AutoMonX Sensor Pack, we added an additional service that with no additional overhead will monitor all the processes and restart the service when needed.

The service can be installed via the Installer under additional services. For manual installation of the API Service use the following command:

AutoMonX_AzureCollector.exe -install_backend

Configure the service using the following INI parameters:

AZURE_SELF_MON_SERVICE_CHECK – How often to run the full check in seconds. Default is 1 minute. If you wish to turn off this feature without stopping the API server, change this value to 0.

AZURE_SELF_MON_MEMORY_LIMIT_MB – Limiting the AutoMonX Processes RAM usage. The default is 1000 MB. Please note that lowering this might cause frequent unwanted restarts to the service.

The Service will log all errors and restart times into the file "Backend_Error_Log.log".



13 Appendix A - Obtaining the Azure Application and Tenant IDs

This section describes how to retrieve an Azure application ID required to monitor your Azure environment.

Select Azure Active Directory -> App registrations -> New registration



In the next windows:

- Choose an application name (for example PRTG_app).
- Select Accounts in this organizational directory only (Single tenant)
- Select Web and fill and <u>https://myapp.com/auth</u> And Click Register



Microsoft Azure	✓ Search resources, services, and docs
«	Home > Default Directory - App registrations > Register an application
+ Create a resource	Register an application
🛧 Home	
🖽 Dashboard	* Name
i≡ All services	The user-facing display name for this application (this can be changed later).
- 🛨 FAVORITES	azure-monitor1 1
All resources	
📦 Resource groups	Supported account types
🔇 App Services	Who can use this application or access this API?
🧟 SQL databases	Accounts in this organizational directory only (Default Directory only - Single tenant)
😹 Azure Cosmos DB	O Accounts in any organizational directory (Any Azure AD directory - Multitenant)
Virtual machines	O Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
💠 Load balancers	Help me choose
Storage accounts	
↔ Virtual networks	Redirect LIRI (optional)
Azure Active Directory	We'll return the authentication response to this LIRI after successfully authenticating the user. Providing this now is optional and it can be
😬 Monitor	changed later, but a value is required for most authentication scenarios.
🐢 Advisor	Web V https://myapp.com/auth V
🧧 Security Center	
Oost Management + Billing	
Help + support	By proceeding, you agree to the Microsoft Platform Policies 🔀
	Register 4

Next step: Assign subscription permissions to the newly created application ID.

Repeat this procedure for all the subscriptions you wish to discover and monitor:

- Write Subscriptions in the search box
- Click the Subscriptions Item
- Perform the grant permissions process for each subscription



Microsoft Azure			Subscription 1			
«	Home > Subscriptions		Services			
+ Create a resource	Subscriptions		↑ Subscriptions 2			
🛧 Home	+ Add		Event Grid Subscriptions			
😐 Dashboard	-		(🔊) Resource groups			
i≡ All services	My role 6	OManage subscriptions in the Billing/Account Center				
+ FAVORITES	8 selected		Resources			
All resources	Apply		No results were found.			
😭 Resource groups	Show only subscriptions sele	cted in the global subs				
🔇 App Services	♀ Search to filter items		-			
👼 SQL databases	SUBSCRIPTION NAME	^↓ su	J Searching all subscriptions. Change			
🧷 Azure Cosmos DB	🤗 Pay-As-You-Go	3	Accou			

The process of granting permissions for accessing each subscription is shown below:

	\mathcal{P} Search resources, services, and docs	<u>>_</u>	Ŗ	¢″ ⊗	? 🙂			
Home > Subscriptions > Pay-As-You-Go - Access control (IAM) Add role assignment								
Pay-As-You-Go - Acc	ess control (IAM)	4	Role	O				
Search (Ctrl+/)			Assic	n access to A	or service principal			
Overview Activity log	Add role assignment Add role assignment Add role assignment		Sele	it 🙃	or service principar			
Access control (IAM)	manage access to Azure resources for users, groups, service principals and managed identities at this scope by creating role assignments. Learn more 🛽	6	azu	re-monitor1				
X Diagnose and solve proble	Name Type Role Name Name or empil	7	7	azure-m	onitor1			
Security Events	Group by 🕐	_	L					
Cost Management	0 items							
🍬 Cost analysis	NAME TYPE ROLE							
③ Budgets	No user assignments exist							
🍖 Advisor recommendations			Sele	ted members:	d Carach fan and i			
Billing			mem	ibers you want to	o assign to the role			
Invoices			Lear	n more about RE	AC			
External services								
😇 Payment methods								
Partner information								
Settings								
Programmatic deployment								
Resource groups								
Resources	~		° ∟	Save Dis	card			



- Select Access control (IAM)
- Select Add button and select Add role assignment
- Select in the Add role assignment box the role of Reader.
- In the Assign Access to box the service principal option.
- Fill in the select box the name of Application ID connection
- Select the application id you created earlier
- Click the save button.

Now we can retrieve the Application ID and Tenant ID:

We go to Azure Active Directory -> App registrations -> Select the monitor application connection.





In the next page copy the Application ID and Tenant ID

Microsoft Azure		${\cal P}$ Search resources, services, and docs
«	Home > Default Directory - App registra	tions > azure-monitor1
+ Create a resource	azure-monitor1	
🛧 Home	C Search (Ctrl) A	🛅 Delete 🌐 Endpoints
🛄 Dashboard	D Search (Clrt+7)	
E All services €	👯 Overview	Welcome to the new and improved App registrations. Looking to learn
– 🛨 FAVORITES ————————————————————————————————————	📣 Quickstart	Display name : azure-monitor1 2
		Application (client) ID 312d1101-76c9-4eab-8acc-b775c13bb816
Airresources	Manage	Directory (tenant) ID 1 6921c24c-1545-4bb8-8b20-ed5e5a51dd05
🗊 Resource groups	🔤 Branding	Object ID : 70b6039b-eacf-473d-bb34-f94ab1cc1cea
🔇 App Services	Authentication	

13.1 Retrieving the Azure Secret Key

The next step is obtaining the Azure secret key:

Navigate to Azure Active Directory -> App registrations -> Select the monitor application connection.





In the next page perform the following:

- Click the Certificates & secrets menu
- Select under Client secrets the New client secret button
- Fill a in the Description
- Select when the token will expire. The longest expiry time is two years, therefore, monitoring this is important (explained in section 13).
- Select Add

Home > automonx.com App registr	ations > Lily Test	Add a client secret	×
Search		Description Expires	Monitor Custom V
 Overview Quickstart Integration assistant 	Credentials enable confidential applications to identify themselves to the authentication service when receiving toket scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.	Start End	03/05/2023
Manage Branding & properties	Application registration certificates, secrets and federated credentials can be found in the tabs below.		
 Authentication Certificates & secrets Token configuration API permissions 	Certificates (0) Client secrets (1) Federated credentials (0) A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as ap	Add Cancel	

In the next page copy the secret key because you won't be able to retrieve it afterwards.

Client secrets							
A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.							
+ New client secret							
DESCRIPTION	EXPIRES	VALUE					
demo secret	5/14/2020	nWu9HVZ7Rnj.2y7XSkVyUngZ][x9Z:e 🎦	Ō				



14 Appendix B - Adding permissions

14.1 Enabling App Secret Keys for monitoring

The App Secret monitoring requires additional permissions to run properly. The addition of permissions is done through the Azure Active Directory (Entra) panel.

From the App registrations panel select the app that is used for the AutoMonX Azure access and perform the following:



Click API Permissions and click the Add a permission button. From there go to Microsoft Graph and add a permission from type Application named:

Application.Read.All

Make sure to give it Admin consent.





14.2 Enabling Reservations for monitoring

Make sure your user has role assignment permission:

Go to Azure Active Directory (Entra) and under Manage select Properties. Toggle Yes under Access management for Azure resources. Press Save.

	automonx.com Pro	operties
	*	🔚 Save 🔀 Discard 🔗 Got feedback?
Ma	inage	
8	Users	Tenant properties
2	Groups	Name *
Û	External Identities	automonx.com
2.	Roles and administrators	Country or region
3	Administrative units	Israel
•	Delegated admin partners	EU Model Clause compliant datacenters
	Enterprise applications	Notification language
	Devices	English V
	App registrations	Tenant ID
۵	Identity Governance	5d33
	Application proxy	Technical contact
2	Custom security attributes	✓
	(Preview)	Global privacy contact
ĥ	Licenses	V
٩	Cross-tenant synchronization	Privacy statement URL
•	Azure AD Connect	✓
	Custom domain names	Access management for Azure resources
ر ک	Mobility (MDM and MAM)	can manage access to all Azure subscriptions and management groups in this
•	Descword reset	tenant, Learn more
		Ves No
	Company branding	Manage security defaults
<u> 2</u> 2	User settings	
111	Properties	

Now open Azure PowerShell CLI.

Find the App id using this command: Get-AzADServicePrincipal -SearchString <app name>

PS /home/	PS /home/ > Get-AzADServicePrincipal -SearchString "Microsoft Azure PRTG"						
DisplayName Microsoft Aze	Id ure PRTG 3e529dad*^	AppId 4aced4fa					
PS /home/	>						



Cope the Id value.

Run the following commands:

Import-Module Az.Accounts

Import-Module Az.Resources

New-AzRoleAssignment -Scope "/providers/Microsoft.Capacity" -PrincipalId <id>-RoleDefinitionName "Reservations Reader"



Now the Azure Sensor Pack will be able to discover the Reservations.

Note – Archived reservations are not discovered.

14.3 Registering Resource Providers

If you encounter the following error:

409: {"code":"AuthError","message":"An error occured in authentication or authorization. Make sure the resource provider has been registered with this subscription"}

The resource Automonx encountered a permissions issue.

The following resource provider must be registered in each subscription: Microsoft.ResourceHealth

Subscription AutoMonX_Prod Resource providers ★ …								
	🤇 Register 🏷 Unregister 🖒	Refresh 🔗 Feedback						
Management - certificates	\checkmark resourcehealth \times	Status : All Registration Policy : All						
∞ My permissions								
š≡ Resource providers	Provider ↑	Status	Registration Policy					
Deployments	Microsoft.ResourceHealth	📀 registered	RegistrationRequired 🕦					



15 Appendix C - Activating Log Analytics on a VM

Go to the desired VM in the Azure portal and navigate to the Insights tab under the Monitoring section. Click "Enable" to activate this monitoring feature.



Give it a few minutes to collect data. Then it will look like this:

1	Virtual machine									
۶	Search (Ctrl+/) «	Resource Group Mo	nitoring O Azu	re Monitor 🤌 F	Run Diagnostic	s 🕐 Refresh	🙂 Provide F	eedback		
	LOCKS	We have released a	new version of Azure	Monitor for VMs. P	lease upprade r	now! →				
Ор	perations									
х	Bastion P	Performance Ma	p Health (pre	view)						
O	Auto-shutdown									
2	Backup	Time range: Last ho	ur as of 20 Mar 10	5:46						
2	Disaster recovery	Logical Disk Perfo	rmance							
ş	Guest + host updates	DISK	CURRENT SIZE (GB)	CURRENT USED (%)	P95 IOPs READ	P95 IOPs WRITE	P95 IOPs TOTAL	P95 MB/s READ	P95 MB/s WRITE	P95 MB/s TOTAL
-	Inventory	/	28.9	12%	0.17	2.97	3.37	0	0.02	0.02
	Change tracking	/mnt	3.87	6%	0	0	0	0	0	0
Ħ	Configuration management (Preview)	Total	32.78	12%	0.17	2.97	3.37	0	0.02	0.02
۲	Policies									
4	Run command	CDI L Utilization 9/						Available	Moment	
Mo	onitoring	1m granularity	0	Av	g Min 50th	90th 95th	Max 🖍	1m granula	rity	
۲	Insights	100%								
	Alerts	80%						476 <u>.8MB</u>		~~~~
ſά	Metrics	60%						286.1MB		
-	Diagnostic settings	40%						190.7MB		
₽	Logs	20%						95 <u>.4MB</u>		
	Connection monitor (classic)	0%						08		
~	Workbooks	Average 0.97 %	95th 2.30 %	04:15	04	50	04:45	Avera 494	ge 5th 9 MB 49	алема 3.9 мв 494 мв
Au	tomation									

Now you can re-run the discovery process and pick the "Azure VM Multi-Disk" sensor when it is complete.



16 Appendix D – Creating Custom Log Analytics Sensors

To allow customers to create custom sensors with Azure KQL queries

- 1. Create the query in Azure portal and make sure it returns values in a specific format.
- 2. Save the query into a QueryPack
- 3. Add the Azure resource to perform the query on, in the Query description field.

Creating the query

Go to the resource (for example, Virtual Machine). Under Monitoring select Logs and write your own query. Query must include UTF-8 characters only. Adjust it so that the returned values will be of the following format:

- 1. Single value Relevant for state or count queries. The channel name will be "Value". The value must be numeric.
- 2. Channel name and value Make sure that the first column will always return the same name for the channel name. Limited to 40 results. The value must be numeric.
- 3. Channel name, value, and unit the third column can represent the custom unit to show in PRTG. Limitations as listed above.

AzureTest Logs Virtual machine	☆	
₽ Search	~	📮 New Query 1* 🛛 × 🕂
😹 Inventory	*	AzureTest Select scope Azure Time range : Last 24 hours Save ∨
🖣 Change tracking		» 1 InsightsMetrics
Monitoring		where Varigin == Vm.azm.ms where Namespace == 'Processor' and Name == 'UtilizationPercentage'
Insights		4 top 1 by TimeGenerated desc 5 project Name, Val
Alerts		
🖬 Metrics		
Diagnostic settings		
🗭 Logs		Results Chart
🕵 Connection monitor (classic)		Name Val
Workbooks		> UtilizationPercentage 1.072379

In this example the query is:

InsightsMetrics | where Origin == 'vm.azm.ms' | where Namespace == 'Processor' and Name == 'UtilizationPercentage' | top 1 by TimeGenerated desc | project Name, Val



Saving the Query

After you reached the desired output, press Save to Save as query. In the popped-out window, name the query – this will be the name of the sensor. The most important part is the **Description** field. Here you must specify the specific resource to perform this query on. This must include the full Resource ID, you can find in under the Configuration -> Properties tab of the resource.

_
Ð



Save as query

Query name *

CPUutilization

Description

7dcde0db3272/resourceGroups/prod_rg/ providers/Microsoft.Compute/virtualMac hines/AzureTestVM

 \times

 \checkmark

٠

Ŧ

Path



Save to the default query pack 🛈

Tags

Resource type

Virtual machines

Category

0 selected

Label

0 selected

Create new label

Save Cancel



Creating the sensor

The sensors will be automatically created like regular AutoMonX sensors during the discovery process. It will be placed under the resource's device (given the resource id was provided in the description).

Editing the query

To adjust or add new resources in the description, open the QueryPack you saved the query into, press on the Export templates tab, and press Deploy.



Now press Edit template:



Custom deployment

Deploy from a custom template

🧭 New! Deployment Stacks let you manage the lifecycle of your deployments. Try it now $\,
ightarrow\,$

Basics	Review + create			
Template	•			
3 C	Custom template 🖙 resources	Edit template	D Edit parameters	<mark>у</mark> Visualize

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * 🕡	Microsoft Azure Sponsorship (
Resource group * (i)	LogAnalyticsDefaultResources ~
	Create new
Instance details	
Region * (i)	(US) East US 2
Querypacks_Default Query Pack_name	DefaultQueryPack 🗸
Virtual Machines_Azure Test_externalid	/subscriptions/
Virtual Machines_Azure Test VM_externalid	/subscriptions/

To add several resources, separate them with a comma.



After you finish editing the query body or description, press Save and continue to Create the template. The query in the sensor will be updated automatically, but to create the sensors for new resources the discovery must be ran again.