IXIA CLOUDLENS WITH RIVERBED APPRESPONSE DEPLOYMENT GUIDE IN AZURE



DEPLOYMENT GUIDE

PROBLEM:

Organizations, even those not typically associated with technology, are migrating to the cloud. This trend is growing because the cloud offers increased flexibility and agility. With this mass migration, organizations have more segments to manage and more potential blind spots in their networks. Regardless of where infrastructure and applications reside, security and compliance needs remain the same. Organizations are finding that their traditional network visibility solutions are unable to meet their needs for visibility of cloud-based data.

SOLUTION:

CloudLens[™], Ixia's platform for public, private and hybrid cloud visibility addresses the challenges of granular data access in the cloud. CloudLens is the first network-level solution that provides Visibility-asa-Service (VaaS) through a Software-as-a-Service (SaaS). It is also the industry's first cloud serviceprovider agnostic visibility platform. This guide describes how to deploy Riverbed AppResponse together with CloudLens visibility in Azure.

KEY FEATURES:

- Elastically scales on-demand so visibility auto-scales horizontally along with the Virtual Machines monitored and the Virtual Machines that are needed to do the monitoring
- Automates cloud visibility management by providing it as a service so no architectural changes required
- Reduces errors by eliminating manual configuration
- Easy to use and setup with a drag and drop interface
- Reduces bandwidth to tools by filtering packets at the source Virtual Machines, eliminating unwanted traffic so tools operate optimally



SAMPLE DEPLOYMENT ARCHITECTURE

* Shown above is a sample deployment, while destination components need to in the same subnet, monitored sources instances can be located in any subnet. CloudLens Sensors run on customer Azure Virtual Mach ines, register up to the CloudLens SaaS which manages them and forwards desired traffic to the destination. Sources may be Linux and / or Windows Azure Virtual Machines. On the Destination subnet, a Linux Virtual Machine is required to terminate the CloudLens Tunnel. Another Virtual Machine runs Riverbed AppResponse.

PREPARE AZURE ENVIRONMENT

NOTE: IN THIS EXAMPLE WE ARE ASSUMING THE SOURCE VIRTUAL MACHINES ALREADY EXIST, YOU CAN THEN LOAD CLOUDLENS SENSORS ONTO THOSE VIRTUAL MACHIENS AS DESCRIBED ON P 5-6 OF THIS DOCUMENT. IN AZURE, EACH VIRTUAL MACHINES INBOUND PORT RULES MUST BE CONFIGURED TO ALLOWS PROPER FUNCTIONING OF CLOUDLENS AND APPRESPONSE, PLEASE SEE P 3-4 FOR DETAILS

YOU <u>DO</u> ALWAYS NEED TO CREATE ONE LINUX INSTANCE IN AZURE, THIS ONE WILL BE USED FOR TERMINATING THE CLOUDLENS TUNNELS, AND FORWARDING TRAFIC FROM CLOUDLENS TO RIVERBED VIA VXLAN AS DESCRIBED ON P 7. PLEASE CONSULT AZURE DOCUMENTATION FOR FULL DETAILS OF HOW TO CREATE A VM.

ALSO, WE ARE ASSUMING IN THIS EXAMPLE THAT AN INSTANCE OF RIVERBED APPRESPONSE IS ALREADY INSTALLED IN AZURE, PLEASE CONTACT RIVERBED FOR ASSISTANCE IF NEEDED.

LOAD CLOUDLENS SENSORS AND CONFIGURE VMS

Step 1 – not required but recommended, configure Tags for easier identification and grouping of Virtual Machines in CloudLens

🔁 🖅 🔨 Create a virtual machi	in X + V	- σ ×								
$\leftarrow \rightarrow$ 0 @ A h	https://portal.azure.com/#create/Microsoft.VirtualMachine	□☆ ☆ & ピ …								
Microsoft Azure	,D Search resources, services, and docs (G+/)	🖂 🕼 🖉 🖉 🙄 😳 gabriel.vartolomeu@ke								
a a	Home > Virtual machines > Create a virtual machine									
+ Create a resource	Create a virtual machine	×								
😭 Home										
Dashboard	Basics Disks Networking Management Advanced Tags Review + create									
🗄 All services	Tags are name/value pairs that enable you to categorize resources and view consolidated billing by applying the same tag									
	to multiple resources and resource groups. Learn more about tags	tay unaligite resources and resource groups. Learn more about tags								
Resource groups	Note that if you create tags and then change resource settings on other tabs, your tags will be automatically updated.									
🕎 Virtual machines	Name 🛈 Value 🛈 Resource									
S Disks	✓ : ✓ 11 selected ✓									
Retwork interfaces										
Public IP addresses										
Network security groups										
Virtual networks										
P Subscriptions										
S Virtual network gateways										
Azure Active Directory										
Storage accounts										
Diagnostics settings										
O Cost Management + Billing										
App registrations										
Activity log										
Help + support										
	Review + create < Previous Next : Review + create >									
	(>								

Step 2 – Configure Inbound Port Rules in Network Security Groups, and apply to Virtual Machines <u>Note:</u> Azure default for Outbound is open for All Traffic. But for Azure Virtual Machine Inbound Port Rules, a few ports numbers need to be explicitly opened:

Source Virtual Machines:

- UDP 19993 (CloudLens Tunnel) *
- TCP 22 (if Linux) **
- TCP 3389 (if Windows) **

CloudLens Termination Virtual Machine:

- UDP 19993 (CloudLens Tunnel) *
- UDP 4789 (VxLAN Tunnel) ***
- TCP 22 **

Riverbed AppResponse Virtual Machine:

- UDP 4789 (VxLAN Tunnel) ***
- TCP 22 **
- TCP 443 **

* Leave open all IP addresses, however if stricter controls are required contact Ixia support

** Specify IP addresses of customer administrators

*** Leave open all IP addresses, however if stricter controls are required contact Ixia support

Home > CloudLens - Inbound security rules								
CloudLens - Inbound secur	ity rules							
	🕇 Add 🔌 De	efault rules						
Overview	Priority	Name	Port	Protocol	Source	Destination	Action	
Activity log	100	UDP_19993	19993	UDP	Any	Any	Allow	
Access control (IAM)	110	Port_4789	4789	UDP	Any	Any	Allow	
Home > CloudLopsOpUburtu16	04 Notworking		k cocurity group					
	04 - Networking	·.	k security group					
Network interface	etwork secu	rity group						
,	* 🖪	Save 🗙 Discard						
Overview	Cverview Network security group							
Activity log	log CloudLens							
Home > CloudLensOnUbuntu16.04 - Netw	vorking							
CloudLensOnUbuntu16.04	- Networkin	g						
	🔷 Attach netw	ork interface 🛛 🛷 Detach networ	k interface					
Overview	Network Ir	nterface: UbuntuVMNic Eff	ective security rules	Topology				
Activity log	Activity log Virtual network/subnet: CloudLensVNET/Subnet NIC Public IP: ***********************************							
Access control (IAM)	Inbound por	t rules Outbound port rules	Application security of	roups Load balar	cing			
🧳 Tags	Network se	curity group Cloudlens (attached	to network interface	UbuntuVMNic)			Add inbound	nort rule
✗ Diagnose and solve problems	Impacts 0 su	bnets, 1 network interfaces		,			nad mooding	portraic
Settings	Priority	Name	Port	Protocol	Source	Destination	Action	
🖄 Networking	100	UDP_19993	19993	UDP	Any	Any	Allow	
Bisks	Bisks 110 Port_4789 4789 UDP Any Any O Allow							

Step 3 – 'Connect' to SSH of Source Virtual Machines from Azure Console (or use RDP for Windows)

Home > Virtual machines > CloudL	ensOnUbuntu16.04				
	6.04				
, Search (Ctrl+/)	« Connect 🕨	Start 🤇 Restart	📕 Stop 🛛 🕄 C	apture <u> </u> Delete	👌 Refresl
Overview	Advisor (1 of 3	3): Enable virtual machin	e backup to protect	your data from corrupt	ion and accid
Activity log	Resource group (c	change) : azure-rm-ub	ountu1604-with-cl	oudlens7cd9	Cc
🗳 Access control (IAM)	Status	: Running			Ol
and Tags	Location	: West US			Si
• 10gs	Subscription (char	nge) : azure-poc-c	loudlens		Ep

Step 4 - Install CloudLens Container on Linux Source Virtual Machines (if Windows, follow Step 5)

AS WELL AS installing CloudLens container on Linux Virtual Machine used for CloudLens Termination

Note: Customers are assumed to have already created an account at <u>http://ixia.cloud</u> before completing the next step. If you don't have an account, you can sign up for a 45-day free trial. After login please create or view your CloudLens Project, and make note of the Project Key (aka API key) which you will need in Step B below.

Step A – If Docker Engine is not already present, install Docker from https://docs.docker.com/install/

e.g. in the case of Ubuntu, docker can be installed as follows;

sudo apt update

sudo apt-get install -y docker.io

Step B - Run the CloudLens container

Note: You will need to substitute your CloudLens Project Key (aka API key) here

sudo docker run \

--name cloudlens \

- -v /:/host \
- -v /var/run/docker.sock:/var/run/docker.sock \
- -d --restart=always \
- --net=host \
- --privileged \
- ixiacom/cloudlens-agent:latest \
- --server agent.ixia.cloud \
- --accept_eula y \
- --apikey <substitute your CloudLens Project Key here> \

Step 5 - where applicable Install CloudLens Sensor onto Windows Virtual Machines

Note: Customers are assumed to have already created an account at <u>http://ixia.cloud</u> before completing the next step. If you don't have an account, you can sign up for a 45-day free trial. After login please create or view your CloudLens Project, and make note of the Project Key (aka API key) which you will need in Step C below

Step A – Download Ixia's CloudLens agent from the link provided on Window Server

https://agent.ixia.cloud/updates/windows/latest

Step B – Install CloudLens agent (click through defaults – until Step C shown next page)

ix his CloudLens – – × ≝ Ixia CloudLens

Options Install Close

Step C – The Windows CloudLens Sensor should be associated with a Project previously created in <u>https://ixia.cloud</u>. The value of Host: agent.ixia.cloud . You must also specify your own Project Key (aka API key)

	This IXIA SOFTWARE agreement between and/or dicking the "I download and/or inst SOFTWARE installed SOFTWARE on hardw the SOFTWARE and/ applicable, Licensee if Licensee does not access or use the SC	E END USER LICENSE AGR you ('Licensee') and Ixia Accept' or similar box or tallation process, and/or t (for example, and not by ware owned, controlled, controlled, controlled, for by activating the SOFT is agreeing to all the term agree to be bound by the VFTWARE in any way, and	EEMENT (this "Agreement , a California corporation button at the beginning wi installing the SOFTWA way of limitation, by per or operated by Licensee) WARE with any association as and conditions of this Agreement I Licensee (either itself of	t ⁺) is a legal ('Dia'). By checking of the SOFTWARE RE or having the mitting Dia to install the mitting Dia to install the and/or by downloading ted license key, as greement. t, Licensee may not register through any of its	,
	<				>
	Accept EULA				
Host:	agent.ixia.cloud				
API Key:	1				

Step D – Finish CloudLens sensor installation



Step E – Restart the instance and verify the instance is associated with the CloudLens project created.



Ixia CloudLens

Installation Successfully Completed

You must restart your computer before you can use the software.

Restart Close

CONFIGURE CLOUDLENS TERMINATION VIRTUAL MACHINE

Step 1 – Connect to SSH of Linux Virtual Machine

Step 2 - Configure VxLAN tunnel between CloudLens and Riverbed AppResponse (e.g. below)

Create a vxlan interface on the VM

sudo ip link add vxlan0 type vxlan id 42 group 239.1.1.1 dev eth0 dstport 4789

- Create a bridge between the vxlan interface and the destination IP of the AR11

sudo bridge fdb append to 00:00:00:00:00:00 dst Riverbed_AR_destination_IP dev vxlan0

- Add an address on the vxlan0 interface

sudo ip addr add 192.168.200.1/24 dev vxlan0

- Enable the interface

sudo ip link set up dev vxlan0

USING CLOUDLENS SAAS PORTAL

Step 1 – Log into http://ixia.cloud and open your previously created CloudLens Project

Step 2 – Click on Instances counter, find your Source Virtual Machines(s), then create an 'Instance Group' for the instances you wish to monitor (optionally you may create multiple Tap Groups for different types of Source VMs – if you previously added Tags to your VMs this can help with grouping)

CloudLens × +								-	
← → C 🔒 catadev.ixia-cloudlens.net/	project/8299526a7135409ab5b4	96c93092d3b4				1	¥ O 🖸 📼	0 🗷	6 :
🚹 IXFOOD 🚾 release 🚾 checkin ઉ GitSw	arm 🧕 Dashboard 📋 AWS	Azure 🗶 Phantom	[Jenkins]CLPrivate	R Ixia Licensing	🛱 JIRA 🤤 IxTracker	🖨 Test_Import 🔲	OpenStack Docs		*
CloudLens > Demo_Project							Acco	unt: Demo	User 🗯
DEFINE GROUP LAUNCH AGENT						2 instances	1 groups 0 to	ols 0 Mt <u>SHOW P</u>	ops traffic ROJECT KEY
INSTANCE GROUPS							MONITORING TOO	GROUPS	
Tap Group 1 instances 0 Mbps									
For support please contact us at clou	idlens@kevsight.com.								

Step 3 - Configure your CloudLens Termination node. Click on the Virtual Machine that is acting as the instance which terminates the CloudLens tunnel, and bridges via VxLAN to the Riverbed AppResponse.

- This time configure the Group as a 'Tool'
- You must specify the Aggregation Interface to match what you setup earlier e.g. vxlan0

ALL FILTERS ACTIVE FILTERS	1 instances Ø					
A	#	TAG: ROLE	INSTANCE ID	INSTANCE TYPE		
type filter name	1	tool	SAVE SEARCH	andard_D2s_v3		
Tag: role	Choose	Columns	Save as an instance group			
tool			Save as a tool			
Instance Id			Name Tool Group			
e2489384-c14d-47aa-9776-b2462f55105f			Aggregation Interface			
bac69577-9b98-4fe1-bbc0-244f0647e434			Vxlan0			
Instance Type			Comment			

Step 4 – Drag a secure visibility path between source and tool groups

a CloudLens × +				- 🗆 ×
C 🔒 catadev.ixia-cloudlens.net/	t/project/8299526a7135409ab5b496c930	92d3b4		☆ 🗘 🖾 🗠 💹 🌚 :
IXFOOD 🚾 release 🚾 checkin G GitSi	warm 🧕 Dashboard 👔 AWS 👫 Azu	re 🗶 Phantom 🧕 [Jenkins]CLPrivate	R Ixia Licensing 🍟 JIRA 🤤 IxTrack	er 🖨 Test_Import 🚺 OpenStack Docs 🔹 👋
loudLens > Demo_Project				Account: Demo User 🦿
DEFINE GROUP LAUNCH AGENT				2 instances 1 groups 1 tools 0 Mbps traffic
				SHOW PROJECT KEY
INSTANCE GROUPS				MONITORING TOOL GROUPS
Tap Group 1 instances 0 Mbps				Tool Group 1 instances 0 Mbps
For support please contact us at <u>clo</u>	oudlens@keysight.com.			

Step 5 - login to Riverbed AppResponse hosted in Azure

Verify traffic from Source Virtual Machines are available in Riverbed AppResponse



WHERE TO GET HELP

If you experience technical difficulties, please email <u>cloudlens@keysight.com</u> for assistance