Lastline Reference Architecture Series: Cloud Security for Azure
Executive Summary

Many enterprises are rapidly moving the entirety of their computing workloads to the Cloud, and one would be hard pressed to find an enterprise that isn't using the Cloud in some fashion. The reasons for doing so are myriad: the replacement and modernization of in-house applications, the rapid pace of innovation, and the attractive cost savings that can be found by no longer having to architect, purchase, deploy, and maintain physical infrastructure are some of the key benefits of Cloud-based projects. Couple that with the flexibility to scale workloads up or down literally in moments, and experiment with cutting edge technologies such as Artificial Intelligence (AI) and Machine Learning (ML) and it's easy to see why Cloud Computing is leading the digital transformation charge and growth continues to charge along at a breakneck pace.

Regardless of the Cloud infrastructure provider you use, they all share similar holistic properties: they allow technology professionals to provide new ways to connect with customers and partners and do it at speeds we could not have envisioned even a decade ago. But with all of these applications and the highly sensitive data both created and stored by enterprises, the security challenges are not small.

What initially started as Infrastructure-as-a-Service (IaaS) has quickly shifted to what is known as Platform-as-a-Service (PaaS). In fact, PaaS is the single largest component of Cloud usage at the time of this writing.¹ Modern workloads are utilizing PaaS more and more, making the security of PaaS a substantial concern. Regardless of whether enterprises are using SaaS, IaaS, or PaaS, well over half of all workloads are reported to have moved to the Cloud already.²

¹ https://assets.kpmg.com/content/dam/kpmg/xx/pdf/2017/02/the-creative-cios-agenda-journey-to-cloud.PDF
The Shared Responsibility Model for Cloud Security

As you would expect, when an enterprise stays with the more traditional on-premises IT model, where they manage all of their own infrastructure, securing the entirety of that infrastructure as well as the applications and data that operate inside it is entirely up to the enterprise. This is not the case when they move to a cloud computing model. The enterprise surrenders some security controls to the cloud service provider (CSP) but keeps other security controls under their roof. Both parties share responsibility for security – just different aspects of it.

Depending on the cloud computing model being used, the cloud provider may be responsible for securing the basic underlying cloud infrastructure such as the underlying network infrastructure and the storage disks. The enterprise user will find itself being responsible for securing their actual data and the applications they are using in the cloud. In most cases they are also responsible for the security of the operating systems and the software being used to support the applications they are using in the Cloud. Think of it in another way: the CSP takes security responsibility for the cloud itself, and the enterprise is responsible for the security of what is inside the cloud. But that is not written in stone: each CSP may have slightly different shared responsibility models. It is essential to understand what you are responsible for, and what the CSP has agreed to provide security for. Amazon Web Services (AWS) maintains a very short and easy to read page on their expectations around shared responsibility\(^3\), as does Microsoft for their Azure platform.\(^4\)

\(^3\) [https://aws.amazon.com/compliance/shared-responsibility-model/](https://aws.amazon.com/compliance/shared-responsibility-model/)
About Lastline Defender

Lastline Defender, a Network Detection and Response (NDR) platform, detects and contains sophisticated threats before they disrupt your business. It delivers the cybersecurity industry’s highest fidelity insights into advanced threats entering or operating in your on-premises and cloud network, enabling your security team to respond faster and more effectively to threats.

The Defender Platform uses a combination of three complementary AI-powered technologies to detect the advanced threats that other tools miss and significantly reduce false positives:

- **Behavioral analysis** to detect malicious content attempting to enter your network via web or email
- **Network Traffic Analysis (NTA)** to detect lateral movement of evasive threats already inside your network
- **Intrusion Detection/Prevention (IDPS)** to detect known threats

This unique combination enables deterministic detections and eliminates most false positives. You can respond faster and more effectively, with fewer resources. Most AI-based network security products implement less accurate techniques. These probabilistic approaches lead to many false positives and hours of follow-up investigation.

Not All Artificial Intelligence is the Same

Applying AI to network traffic will inevitably detect anomalous patterns of behavior, because that is what it’s designed to do. Unfortunately, it is virtually impossible for these other AI-based tools to understand if the detected anomaly is malicious or benign. After all, not all anomalous activity is malicious, and not all malicious activity is anomalous. Lastline is different.

We utilize AI that is automatically trained both on network traffic and malicious behaviors:

- Lastline Defender applies unsupervised Machine Learning (ML) to your network traffic to detect protocol and traffic anomalies.
- It applies supervised ML to automatically create classifiers that recognize malicious network behaviors and previously unknown malware

The Lastline Defender platform provides a detailed understanding of a threat’s scope by identifying compromised systems, communication between local and external systems, and data sets that might have been accessed or uploaded. It facilitates hunting of latent threats resulting from file downloads, website content, and email attachments that are now hiding in your network.
Azure Overview and Key Concepts

Microsoft Azure uses the Azure Resource Manager (ARM) to deploy, manage, and monitor key Azure resources like virtual machines, storage, networking, etc. Azure gives developers multiple communications options in which they can interface with Azure:

- PowerShell
- Command Line Interface
- GUI Portal
- REST API

ARM templates in JSON format are used for repeatable designs in Azure. Azure supports multiple programming languages (depending on Azure runtime version):

- C#
- JavaScript
- F#
- Java
- PowerShell
- Python
- TypeScript

**Azure Resource Groups** are logical containers/grouping of resources. Azure resources belong to only one resource group. These are used to segregate deployments (for example: applications A, B, C, D; Development, QA, Production, etc.)

**Identity and Access Management**

Role-Based Access Control (RBAC) is used to manage access to Azure resources. In order to make assigning roles easier for customers, Azure comes with dozens of well-defined built-in roles (for example: Owner, Contributor, SQL DB Contributor, etc.) There is also the ability to create and define custom roles based on the customer’s unique requirements (for example, a “Security Appliance Administrator”).

**VNet Peering**

VNet peering allows VNets to communicate with each other. Remember, IP address ranges should not overlap. Traffic between VNets flows through Microsoft’s internal backbone, not the public internet. Two types of VNet peering are available:

- VNet peering – within the same Azure region.
- Global VNet peering – across Azure regions.

**Virtual Network (VNet)**

A VNet is a logically segmented network within Azure. It allows connected resources to communicate with each other. It can contain either a private or public IP address range, divided into appropriate subnets. VNets are isolated from each other, but can have an overlapping IP address range. Similar to AWS, if resources or applications need to communicate across VNets or other networks, IP address ranges can not overlap.

**VNet Traffic Routing**

By default, resources within a VNet can communicate with each other and the internet. There are two types of routes available:

- System routes – these are automatically created and managed by Azure.
- Custom routes – these are managed by the customer and they override system routes.
**Lastline uses User Defined Routes (UDR) (a part of custom routes) to route network traffic to the Lastline sensor.**

**Network Security Group (NSG)**
A virtual firewall is provided by Azure, and it can be associated to a specific subnet or to an individual network interface. It consists of a prioritized list of rules, which define the policies for the flow of network traffic. Rules are defined by priority order, traffic source, traffic destination, port/port range, direction (inbound or outbound), protocol, and action (allow or deny).

**On-Premises Connectivity**
Azure provides two types of connectivity to an on-premises datacenter:
- **VPN Gateway** – IPSec VPN connectivity through the public internet.
- **ExpressRoute** – dedicated WAN connectivity.

**Load Balancing**
Azure provides customers with three types of load balancing:
- **Azure Traffic Manager** – DNS distributes traffic across multiple Azure regions or data centers.
- **Azure Application Gateway** – HTTP/HTTPS layer load balancer.
- **Azure Load Balancer** – TCP/UDP layer load balancer. Lastline sensors are deployed under Azure Load Balancer for both high availability and scalability.

**Lastline Sensor Design Details**
The Lastline sensor is a virtual appliance readily available in Azure Marketplace. Lastline provides both a Bring-Your-Own-License (BYOL) or Pay-as-you-Go (PAYG) licensing model for customers.

The recommended VM size for a Lastline sensor is:
- 16 cores
- 32 GB memory
- ~7 to 10Gbps network interface
- Azure VM Standard_F16 or higher.
- Accelerated networking must be enabled in the sensor VM.

**Important Architectural Considerations and Questions:**

**Q:** Can a sensor inspect traffic in real-time?
**A:** Yes, sensors are deployed in Azure to monitor and protect network traffic “inline”.

**Q:** Can I configure Azure vTAP (“mirrored” traffic) to work with a sensor?
**A:** No. As of December 2019, Azure vTAP is still in “preview” mode.

**Q:** Can a sensor scale up or down based on VNet traffic?
**A:** Yes, if the sensor is deployed under Azure internal load balancer.

**Q:** I have workloads in Azure China. Can I use the sensor in Azure China?
**A:** Azure China is a separate entity from other Azure regions. The recommended approach is to deploy the Lastline Sensor in another region, then transfer the sensor image to Azure China.

**Q:** What capabilities does the Lastline sensor in Azure provide?
**A:** All of the expected capabilities of an on-premise Lastline NDR deployment (malicious content, anomaly detection, signature analysis, etc.)

**Q:** Can a Lastline sensor analyze encrypted traffic?
**A:** Yes, but with encrypted traffic your sensor capabilities will be limited. It is recommended to route unencrypted traffic to the Lastline sensor to take full advantage of the capabilities NDR offers.
Questions to Ask Yourself to Determine Number of Sensors Needed:

1. How many VNets are in your Azure environment?
2. How many Azure regions are used? If you have workloads in many Azure regions, and you choose to deploy a Lastline sensor in only one region, this will result in an additional data transfer cost and increased latency.
3. How many of these VNets run critical workloads that you wish to protect using a Lastline sensor?
4. What is the average network traffic bandwidth in your VNet(s)?

Lastline recommends the following for most efficient and accurate results:

- Deploy a sensor in each VNet that are traffic intensive or contain a mission-critical workload.
- Deploy a sensor in Hub VNet or Transit VNet.
- Ensure the Lastline sensor VM’s network interface has sufficient bandwidth to handle traffic, keeping in mind that if necessary, sensors can scale horizontally if deployed under Azure load balancer.

Design Considerations

- Deploy your Lastline sensors under Azure load balancer for both high availability and scalability.
- Deploy your Lastline sensors in their own subnets. This will prevent possible ‘routing loops’.
- Use UDR to route traffic to a Lastline sensor. Use NextHop type as ‘Virtual Appliance’.
- Depending on your requirements, any traffic can be routed to a sensor appliance (north-south and east-west traffic).
- “Enable IP Forwarding” option should be enabled in the sensor VM’s network interface so that sensor can forward the traffic it receives. Failure to do this will cause the traffic to be dropped.
Deployment Options for Azure

Depending on your workload size and your answers to the above questions, there are multiple deployment options:

- A sensor protecting a single VNet.
- A sensor protecting multiple VNets.
- A sensor deployed in Hub VNet or Transit VNet.

Figure 1: n-tier application in Azure, with Lastline sensor.
Figure 2: Lastline sensor in multiple VNets.
Figure 3: Lastline sensor in Hub and Transit VNet.