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. 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.

Executive Summary

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.
AWS Overview and Key Concepts

AWS uses multiple access methods to deploy, manage, and monitor AWS resources such as virtual machines, storage, and networking. AWS provides multiple methods of interfacing for this access: via CLI, through the AWS portal GUI, or through a REST API. Many programming languages are integrated for use with AWS such as Java and Python. Many other languages are supported. 5

AWS uses Identity and Access Management (IAM) to manage access to AWS services and resources. Like traditional IAM, AWS uses the following classifications:

- **User** – a user or service account authorized to access.
- **Group** – a collection of Users with similar access permissions.
- **Role** - tied to AWS resources, authorized to access other AWS services or resources. For example: a role tied to an EC2 instance authorized to access a S3 bucket.
- **Policy** – a list of resources and permissions that can be associated with a User, Group or Role.

AWS uses three different types of load balancing to meet the needs of their customers: 6

1. **Classic Load Balancer**
   A legacy load balancer to be used with only EC2-Classic Network, not the Virtual Private Cloud (VPC).

2. **Application Load Balancer**
   An HTTP/HTTPS layer load balancer.

3. **Network Load Balancer**
   A TCP/UDP layer load balancer.

---

5 https://aws.amazon.com/developer/
6 https://docs.aws.amazon.com/AmazonECS/latest/developerguide/load-balancer-types.html
Virtual Private Cloud (VPC)
Amazon Virtual Private Cloud allows you to provision a locally segmented network with an AWS region, which allows connected resources to communicate. You have complete control over the networking of the VPC, allowing you to define either a private or public IP address range, divided in subnets.⁷

It is important to remember that while VPCs are isolated from each other, they can have overlapping IP address ranges. In the case where you need to communicate across VPCs or to another network, the IP address range can not overlap.

VPC Components
All VPCs share the same networking components:
• **Subnets** – identifies a part of VPC CIDR block.
• **Public Subnet** – a subnet is associated with a route table that has a route entry to the internet via an internet gateway.
• **Private Subnet** – this subnet does not have a direct route to public internet. Typically instances in a Private Subnet can make outbound connectivity to the public internet via a NAT gateway.
• **Route Table** – this provides a layer 3 routing mechanism. Subnets are associated with a route table.

Virtual Firewall
Multiple firewall options are available including a Web Application Firewall (WAF), a host-based firewall, or an in-line firewall.⁸

Other important security capabilities include:
• **Network Access Control Lists (NACL)**
  – Control routing within the VPC.
  – Associated with a subnet.
  – NACLs are stateless. They must have explicit inbound and outbound rules.
• **Security Groups**
  – Associated with the instance’s network interface.
  – Stateful, which allows bidirectional traffic for the given rule.

A NACL is applied first, which is followed by Security Groups.

AWS Networking
AWS provides substantial flexibility to meet the networking needs of its customers:

**On-Prem Connectivity**
AWS provides two types of connectivity to an on-prem datacenter:
• **VPN Gateway** – IPSec VPN connectivity through public Internet.
• **DirectConnect** – Dedicated WAN connectivity.

**VPC Peering**
This allows VPCs to communicate with each other. As mentioned earlier, remember that IP addresses should not overlap. All traffic flows within AWS itself, not via the public internet. VPC peering allows you to create different network architectures, including:

---

⁷ /28 or larger
⁸ https://aws.amazon.com/answers/networking/vpc-security-capabilities/
- **Transit VPC** – a hub and spoke VPC architecture.
  - **Note:** this is a deployment model, not an AWS service (unlike Transit Gateway).
  - Appliances are typically deployed in Transit VPC.

- **Transit Gateway** – a central gateway to manage all network communication.
  - Managed Service provided by AWS.
  - Used to connect VPCs, on-prem etc.

---

Figure 1: AWS Transit VPC Architecture

Figure 2: AWS Transit Gateway Architecture

---

VPC Flow Logs

VPC flow logs enable the capture of IP traffic flow metadata in the VPC. Because flow log data is collected outside of the normal network traffic path, there is zero impact on both latency and throughput. Flow logs can be enabled at the VPC level, the subnet level, and at the network interface level. Lastline Defender is able to analyze flow logs in order to detect security events such as anomaly detection (for example: abnormal communication between hosts), reputation-based events (for example: traffic seen travelling to a known bad IP address or other threat intelligence derived information), among others.

Flow logs are formatted like this:

- `<version> <account-id> <interface-id> <srcaddr> <dstaddr> <srcport> <dstport> <protocol> <packets> <bytes> <start> <end> <action> <log-status>`

VPC Traffic Mirroring

VPC traffic mirroring enables the capture and copying of network traffic from an elastic network interface (ENI) that is attached to an EC2 instance. From a security perspective, this is used for both threat monitoring and content inspection. It also allows you to filter network traffic in order to only provide the information you are interested in monitoring. The traffic mirroring target can be another ENI or a network load balance (NLB). A Lastline sensor deployed under NLB is the target for traffic mirroring sessions.

VPC traffic monitoring has some very compelling benefits: it gives you the ability to do deep packet inspection (DPI). It also provides access to any traffic (north-south or east-west). Conversely, it is absolutely essential that the ENI have sufficient bandwidth to handle traffic mirroring. If it does not, traffic mirroring traffic will be dropped. Using traffic mirroring may require the redeployment of larger virtual machines in order to handle the increased traffic and load.

VPC Ingress Routing

VPC ingress routing is a new feature in AWS that was announced during the re:Invent conference in December 2019. For enterprises who wanted to enforce the same security policies on their VPCs as their on-premises infrastructure, this was a welcome announcement. VPC ingress routing allows you to send all your network traffic to a specific EC2 instance (which will run network security tools such as Lastline Defender) before that traffic reaches your workloads. This allows you to block any malicious traffic in-line before it can cause an incident. Lastline is one of the initial launch partners.

11 https://docs.aws.amazon.com/vpc/latest/userguide/flow-logs.html
12 https://docs.aws.amazon.com/vpc/latest/mirroring/what-is-traffic-mirroring.html
Figure 3: VPC Ingress Routing Architecture
Lastline Sensor Design Details

Lastline sensors are virtual appliances and are readily available in the AWS Marketplace. Lastline provides both a Bring-Your-Own-License (BYOL) or Pay-as-you-Go (PAYG) licensing model for customers.

Recommended VM size for a Lastline sensor is:
- 16 cores
- 64 GB memory
- 10Gbps network interface
- m5.4xlarge or higher

Important Design Considerations:
- You should deploy the Lastline sensor under Network Load Balancer for HA and scalability.
- It is best practice to deploy the Lastline Sensor in its own subnet. This will prevent possible routing loops.
- Use AWS ingress routing to route north-south "inline" traffic to the Lastline Sensor.
- Use AWS traffic mirroring to route east-west "mirrored" traffic to the Lastline Sensor. The traffic mirroring session target should be the private IP address of the network load balancer.
- Traffic mirroring needs to be configured at each ENI in the VPC.
- Enabling traffic mirroring in ENI will double the network throughput. Make sure the ENI has sufficient bandwidth to handle traffic mirroring or traffic will be dropped.

Questions to Ask Yourself to Determine Number of Sensors Needed:
- How much network traffic are we generating in AWS?
  - Make sure the Lastline sensor VM’s network interface has sufficient bandwidth to handle traffic.
  - The sensors can scale horizontally if deployed under the Load Balancer and Auto Scaling Group.
- How many VPCs are in your AWS environment?
- How many AWS regions are you operating in?
- How many of these VPCs are running critical workloads that you want to protect?
- What is the average bandwidth seen in the ENIs in these VPCs?
  - This information is critical to ensure your existing ENIs can handle the bandwidth requirements of traffic mirroring.
  - If existing ENIs are not able to handle the doubling in bandwidth for traffic mirroring, you will need to plan to redeploy more powerful or “beefier” VMs to support traffic mirroring.

Lastline recommends that you deploy a sensor in each VPC that is traffic intensive and you deploy a sensor in either the Transit VPC or Transit Gateway model.
Deployment Options
Depending on your answers to the previous questions and the size of your workload in AWS, there are three deployment options. They are:

- A single sensor protecting a single VPC. This is recommended for customers who only wish to protect a single VPC.
- A single sensor protecting multiple VPCs. Depending on workload sizes and bandwidth requirements this can be an option for a customer who wishes to protect a small number of VPCs, typically 2 to 4.
- A sensor deployed in the Transit VPC or the Transit Gateway. This is the recommended deployment for those customers who wish to protect a large number of VPCs.

Single VPC Deployment Model:
- Sensor protects single VPC.
- Public subnet has Application Load Balancer (ALB) and NAT gateway.
- Private subnet has EC2 instances deployed with Auto Scaling Group (ASG).
- SOC subnet has Lastline Sensors deployed under Network Load Balancer (NLB1) and ASG.
- Each ENI has a traffic mirroring session configured with its destination as NLB1.
- Virtual Firewall protecting communication within VPC.

Figure 4: Single VPC Deployment Model
Multiple VPC Deployment Model:

- Sensor protects multiple VPCs (in this example the Production and Dev/QA VPCs)
- Public subnet has Application Load Balancer and NAT gateway.
- Private subnet has EC2 instances deployed with Auto Scaling Group (ASG).
- SOC subnet has Lastline sensors deployed under Network Load Balancer (NLB1) and ASG in Production VPC.
- Each ENI has a traffic mirroring session configured with its destination as NLB1.
- Virtual Firewall protecting communication within VPC.
Transit VPC Deployment Model:
- Sensor protects multiple VPC, and on-prem traffic.
- Lastline sensors deployed in dedicated VPC.
- Transit gateway, Transit VPC architecture used.

Figure 6: Transit VPC Deployment Model