AWS Cloud Project: OpenVPN Implementation

Luca Pasto

MIS 420-01-Spr2025

5/11/25

1. System Creation and Deployment

1.1 Project Setup

This section outlines the initial project setup for utilizing resources and determining what is necessary to successfully host a Virtual Private Network (VPN) within the AWS cloud to facilitate secure and encrypted connectivity over the internet. These first following steps involved creating a Virtual Private Cloud (VPC) and within it an EC2 instance labeled “mis420lp-openvpn”. This EC2 instance was then set to host the VPN via OpenVPN software installation and configuration on a Linux/Unix operating system.

1.2 Tasks and Significance to Information Security

Again, the project began by creating a Virtual Private Cloud (VPC) to segment other AWS resources, allowing for an enhanced security posture. The VPC labeled “mis420lp-vpc” was configured to have a public and private subnet for connectivity. Once the VPC was established, security groups were then created, acting as a virtual stateless firewall, which examines traffic based on predefined rules. These rules were configured to open and close ports for both inbound and outbound traffic. The security group was named “mis420lp-openvpn” which has inbound rules opening ports 22, 443, 943, 1143, and all ICMP traffic. This allows for the user to connect and establish configurations via SSH terminal as well. Outbound rules are defined to open ports 53, 443, and 80. This allows for HTTP/HTTPS connections and connection to DNS. If this is not explicitly configured for outbound traffic, the user would be able to connect to the VPN but would not be able to make any connections through it. The EC2 instance was then configured using an AMI to have the OpenVPN software installed on a Linux-based OS. The EC2 was then placed on the “mis420lp-vpc" and given the “mis420lp-openvpn” security group. Initial configurations for ports and login credentials were entered by connecting to the instance via Windows PowerShell.

To further bolster the availability of the EC2 instance, both an S3 bucket and a backup plan were configured to have an AMI that could be used to restore or duplicate the VPN. Utilizing the AWS Backups console, a backup plan was created to increase the redundancy of the EC2 and was scheduled for monthly backups. The S3 bucket served as another form of redundancy by providing a place to store AMI backups as well. This was done by manually creating an AMI snapshot of the EC2 instance, then utilizing AWS CloudShell to securely store it on the S3 bucket. This is important in case of a possible system failure or misconfiguration of the EC2. Maintaining a constant uptime is important for reliability and stability, ensuring that essential processes are not disrupted.

Through the AWS CloudWatch console, two different CloudWatch alarms were set to monitor the resources on the EC2 instance and alert for any suspicious activity. The first named "mis420lp-openvpn-usage" detects ingress traffic with a lower threshold to monitor anyone who initially connects to the VPN. The second alarm is named "mis420lp-networkout-usage," which becomes triggered once network out exceeds a 100,000-byte threshold to monitor unusually high outbound traffic. Once these alarms are triggered, AWS Simple Notification Service is enabled so that an email subscribed to the notifications receives an alert. The logs generated by CloudWatch are then backed up to an S3 bucket named "mis420lp-cloudwatch-logs" for future reference.

Lastly, roles and users were created through AWS IAM for the maintenance of the EC2 instance as well as for security auditing purposes. The user "mis420lp-ec2-user" is given permissions for EC2 access, allowing for maintenance, monitoring, and configurations. The second user “mis420lp-backup-user” was created for restoration purposes across all of the VPNs' related resources, including the EC2, S3s, and AWS Backup, to act as an administrator account. The two roles that were created are named “mis420lp-openvpn-buckets-role” with the necessary permissions to manage the AMIs and buckets associated with it, and “AWSServiceRoleForAmazonInspector2Agentless” which allows AWS Inspector to call services on behalf of the account owner.

1.3 Core Principles

To begin with, accountability within the project as a whole is addressed through three main categories, including IAM, CloudWatch, and with backups. Identity and access management allows for clearly defined users and roles, which facilitates auditing through tracing back specific actions within the account. Notably, this adheres to the principle of least privilege as users and roles are given only what is necessary for their specific tasks and job functions. This is important because it further improves the security posture of the system. In cases of account compromise, the account would only have limited access to resources that could disrupt business functions, effectively segmenting it off. CloudWatch enables monitoring of defined ingress and egress points, enabling a clear audit trail for any possible security events that might occur. This would allow any administrators of the system to recognize the suspicious activity and address it promptly with alerts. Each of the backups helps with accountability as well, since they could also be used as a golden image to recognize possible deviations. This baseline allows for business continuity and a way to restore in case of failure or misconfiguration.

To address confidentiality, encryption through S3 buckets and overall network segmentation ensures that any sensitive data is accessible only to those who are authorized. Deployment of resources through the VPC minimizes risk through traffic segmentation. This helps ensure that all traffic is clearly controlled, preventing unauthorized access to the system. S3 buckets, on the other hand, address confidentiality through encryption. All data stored inside the S3 bucket is encrypted by default, preventing a threat actor from compromising the data.

1.4 Rationale

Hosting a VPN on AWS provides the infrastructure and reliability to provide encrypted, secure connections for any remote user. OpenVPN hosted on an EC2 instance allows for a more scalable, cost-effective, and overall flexible solution that could be catered to companies of any size. There are several distinct advantages that come along with hosting on the AWS platform. Resources can be provisioned easily and adjusted to the current demand or workload of the organization attempting to implement it without any upfront cost for physical infrastructure or implementation. The cloud platform also gives many tools that are readily available and can be leveraged immediately, like AWS Inspector, CloudWatch, or IAM, providing essential functions that seamlessly integrate with the rest of the environment. Some of the notable disadvantages to this implementation are the overall dependency on AWS infrastructure and the possibility of misconfiguration leading to increased costs.

2. Security Enhancements

2.1 Improvements Discussion

There are several areas in which the current system could be improved upon that allow for greater security or efficiency. One example of this would be the use of Lambda functions to automate workflows involved with S3 buckets. AWS Lambda could be used to automate EC2 and CloudWatch log backups, which would be more efficient and secure since there would be no chance for human error in the process. Another improvement that could be made is the overall permissions given to each user account. There are many different sets of permissions that each encompass a segment of the overarching service it is a part of. These permissions could be fine-tuned to further adhere to the principle of least privilege, improving the overall security posture. One last notable improvement would be the use of more alarms beyond CloudWatch. Other AWS services, like CloudTrail, could be utilized to closely monitor user actions and access.

2.2 Implementing MFA

The AWS platform allows for several different methods of multi-factor authentication, including biometric keys, authenticator apps, and hardware tokens. This works by having each user enter their login credentials, verify through a CAPTCHA system that they are human, and as a final check, having them enter the code provided through their software/hardware token or biometric, like fingerprint or face scan. Each of these could be attached to a user account through the IAM console and notify the administrative users which accounts have it enabled or disabled. The current root account has a multi-factor authentication set utilizing the authenticator app Duo Mobile. This app generates a software token every 30 seconds that could be used to login and gain access to the account.

2.3 Hardening Practices

There are several methods by which the VPN can be hardened on the AWS platform. Some of these are readily available and built into AWS with services such as Inspector and the Well-Architected Framework. Inspector completes comprehensive scans of the currently used resources and ranks them from most to least critical. This helps aid in hardening by giving priority to configurations that need to be adjusted. The Well-Architected Framework is a set of Amazon-provided tools that help customers evaluate and implement designs. These guidelines encompass six pillars addressing operational excellence, security, reliability, performance efficiency, cost optimization, and sustainability. Each of these pillars can be incorporated into the platform through the AWS Well-Architected Tool, which is a mechanism that can regularly identify issues, evaluate workloads, and suggest improvements to the current system.

2.4 Role/Policy Restrictions

There are several methods by which policies and roles can be applied to resources within the AWS platform. Beyond granting permissions to individual users, roles and policies can be attached to resources or groups as well. For instance, a group can be granted specific permissions for a particular department in an organization, allowing for simplified management. Users can then be put into the group to complete their corresponding job functions and removed once no longer necessary. Policies can also be attached to a particular resource, like an S3 bucket, defining how it should be used or accessed. These policies explicitly define the exact actions that can be performed and under what conditions. This enables organizations to tailor permissions to their resources' needs.

3. System Support Plan

This comprehensive 3-month support plan will give a high-level overview of the actions and steps that should be taken and prioritized in a real-world implementation of the VPN hosted on AWS. A higher priority objective for maintaining the system would be establishing a golden image utilizing Amazon Inspector to update and configure both the EC2 instance operating system and OpenVPN software. Amazon Inspector helps by prioritizing critical vulnerabilities in the EC2, as well as the current software being run on the instance. This data should be utilized to establish a clear base, which could then be backed up and stored with an AWS backup plan or through the use of an S3 bucket. The AMIs from which the instances are initially generated lack current updates, so the systems must be updated to patch known vulnerabilities before use. This could then be transitioned to a weekly approach that frequently tests patches and applies them once verified.

Another high-level objective that could be addressed within these three months is the implementation of MFA for all users. Multi-factor authentication severely reduces the risk of unauthorized account access because authentication is handled in a multi-layered approach. Multiple forms of authentication make it so that if one approach becomes compromised, like a password, the account is still secure through other forms. This should be done through the many previously listed options, like Duo Mobile authenticator or a hardware token for users on a case-by-case basis.

Lastly, from a day-to-day operational perspective, proactive monitoring through AWS CloudWatch and CloudTrail would be more critical. This would allow constant fine-tuned monitoring of resources and IAM user access attempts, as well as their actions. Logs from these sources should be regularly reviewed to identify unauthorized access attempts and anomalous behavior on the network. This, in tandem with a security information and event management tool (SIEM), would allow aggregation of the logs and use of dashboards to enhance monitoring capabilities, facilitating efficient administrative response.

4. Incident Response Plan

This incident response plan will emphasize guidelines from the NIST 800-61r3 as well as the AWS Security Incident Response User Guide. This will outline the response lifecycle that should be followed, but in the context of the current VPN system hosted on AWS. The preparation phase of the incident response plan should outline clear roles and permissions necessary prior to any event. A team should be created with specific roles and responsibilities, should an event occur. In the context of the current system, this might be provisioning more accounts specifically for incident response through AWS IAM.

The next phase of the lifecycle will encompass detecting and analyzing any security findings through the use of Amazon GuardDuty, CloudWatch, and CloudTrail. Should any suspicious activity be deemed as an event, the event should be escalated and moved on to containment. In this case, the EC2 instance should be immediately isolated from the rest of the resources on the VPC. This could be done by editing security groups to block off any inbound or outbound traffic. The instance could then be removed from the public subnet onto a private subnet and dissociate the elastic IP that is currently assigned to it. Doing so prevents any further exposure to critical information that could have been on EC2, such as PII, user credentials, and more. This could further compromise the rest of the AWS environment if any of the credentials provide more access to the system. In this case, all compromised users should have their accounts suspended.

After isolation is complete, the cause of the incident should be fully removed from the affected instance in an eradication phase. Tools such as Inspector could aid in this process by thoroughly scanning for any vulnerabilities or malicious elements. Once these affected components are identified, they must be completely removed or have their configurations returned to the golden image. Once all of these steps have been sufficiently completed, current backups and configurations should go under review to determine if they are still vulnerable prior to a complete restoration. In conclusion, a more comprehensive document outlining all processes and steps should be made to review what was successful and what wasn’t for future incidents.