Securing AWS Instances

BAHx Security

Angelica Espinoza, Edwin Hyatt, Nathen Mammen, Roberto Basurto, Zach Kaylor

George Mason Volgenau School of Engineering, 4400 University Dr, Fairfax, VA 22030

Booz | Allen | Hamilton

Project Overview

Goal: The project involves securing multiple EC2 instances in an AWS environment, with each instance being categorized by the low-risk, moderate-risk or high-risk data it holds

Classification: The data will be classified as low, moderate or high in order to properly mitigate their respective risk and place them in appropriate security groups

Defense: The different data classifications will require different security controls to be implemented on each data-protecting instance

Support: A fourth EC2 instance will be utilized to send out updates to the three data-protecting instances and receive logs

Tools & Objectives

Fail2ban: intrusion prevention software framework that protects computer servers from brute-force attacks

OpenVAS: Periodically scans the instance for any known vulnerabilities while also creating a baseline and monitoring anything outside that baseline

Bro: open source framework that is used to analyze network traffic and detect any anomalies

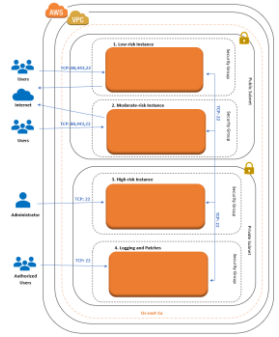
Snort: open source network intrusion detection system

Rsync: fast and secure file-copying tool for use across remote machines that supports data transport over SSH

Google Authenticator: Uses 2-factor authentication on mobile devices to allow access to certain services and privileges

OpenDLP: A data loss prevention solution that will protect the data stored within the instance from being exfiltrated.

AWS Environment



EC2 Instances Architecture

Security Architecture

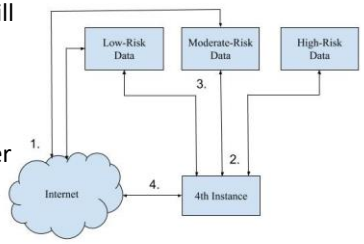
The AWS environment consists of four instances in a Virtual Private Cloud (VPC). The fourth instance is used for logging and patching. The instances are configured into separate security groups and placed into different subnets. Users would authenticate to the instances through the utilization of public/private-key pairing and each instance has limited open ports depending upon its risk level.

Data Model

1. Logs: Logs generated from the three instances will be sent to the fourth instance for storage and analysis.

2. Fetching/Patching: Once the fourth instance has fetched updates from the primary mirrors, the other three instances will fetch packages from the fourth.

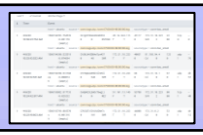
4. Fourth Instance: Connects to the Internet and is responsible for periodically checking for necessary



EC2 Instances Model

Solutions

The team uses 4 AWS EC2 instances on an AWS organization account. The AWS console allows for visuals of network activity,



Above shows all the entries of the conn.log files that were uploaded from a single zip file. On the left side are the fields that were extracted from each of the files and will be used to create a visualization of the IP addresses in the form of graphs.

Tool Implementation

Bro Logs showing network connection logs

Fail2ban showing failed brute-force attack

SSHD showing the instance access logs

Lastb showing failed login attempts

Results

The team created bash scripts to run on each instance. There were four scripts made to fit the low, moderate, and high-risk data-protecting instances, as well the fourth logging/patching instance. To start, the scripts echo instructions to the user, then proceed to update and upgrade the instances. The script then installs necessary repositories before installing Linux packages of the chosen security tools and configuring them.

Conclusion

In order to secure the data-protecting instances the team reviewed NIST Special Publications for security controls in relation to a cloud environment and then selected the controls that were appropriate to the project requirements. Open-source tools with the needed functionality to fit the selected security controls were researched and implemented onto the instances using a bash script. Each script implements different levels of security depending on the risk-level of the presented instance. All data-protecting instances were secured and the script allows for changes to be made in configuration is needs change.