Sprint-wise Retrospective

DDoS Protection System for Cloud using AWS and Machine Learning

Panel No. 06
Supervisor Name
Dr. Balaji Srikaanth P, AP/NWC
Dr. S. Nagendra Prabhu, AP/CINTEL

Batch No. NW000156
Shaurya Singh Srinet – RA2111032010006
Shounak Chandra – RA2111032010026
Charvi Jain – RA2111047010113

Category: Research

Sprint 1 : Project Initialization and AWS Setup

Liked	Learned	Lacked	Longed For
Share aspects of the sprint that you enjoyed or found particularly effective.	*Discuss lessons learned, whether they are related to processes, technical aspects, or teamwork.*	*Identify areas where the team felt a lack of resources, support, or information.*	*Discuss any desires or expectations that the team had but were not met during the sprint.*
Successfully set up AWS environment with EC2 instances, Auto Scaling, and Load Balancer configurations without major issues.	Learned how to efficiently configure and manage AWS services like EC2, Auto Scaling, and Load Balancer for high availability.	Lack of sufficient documentation for Auto Scaling configuration in complex use cases.	Desired clearer guidelines or best practices for optimal security group configurations to ensure safe, scalable infrastructure.
The seamless SSH connectivity setup allowed for smooth remote management of the EC2 instances.	Gained hands-on experience in managing EC2 instances and setting up necessary permissions for SSH access.	Missing comprehensive examples of handling edge cases for SSH connectivity issues.	Wished for better support tools for troubleshooting connectivity issues more quickly.
The EC2 instances were set up to mirror real-world deployment environments, providing realistic configurations.	Realized the importance of thoroughly testing connectivity and system access before proceeding to other configurations.	Insufficient time was allocated to fully explore and configure advanced Auto Scaling policies.	Longed for more time to experiment with different AWS configurations and evaluate their impact on performance under varying loads.
The project's infrastructure was provisioned quickly, allowing the team to focus on higher-level tasks.	Understood the critical role of Load Balancers in ensuring traffic distribution and high availability across EC2 instances.	Lacked some advanced knowledge about integrating monitoring tools with AWS services during the setup phase.	Desired quicker feedback on the setup phase to detect potential misconfigurations earlier.
The team was able to set up the project's environment with minimal errors, keeping the sprint on schedule.	Realized the need for efficient cost management strategies to ensure long-term scalability.	Faced minor roadblocks due to limited access to additional AWS resources or credits.	Hoped for more resources or faster access to tools needed for more extensive system testing.

Sprint 2 : Traffic Simulation and Data Collection

Liked	Learned	Lacked	Longed For
Share aspects of the sprint that you enjoyed or found particularly effective.	*Discuss lessons learned, whether they are related to processes, technical aspects, or teamwork.*	*Identify areas where the team felt a lack of resources, support, or information.*	*Discuss any desires or expectations that the team had but were not met during the sprint.*
Successfully configured Apache JMeter to simulate real-world traffic scenarios, including benign and DDoS attacks.	Learned how to simulate various traffic patterns (normal vs. attack traffic) using JMeter.	Lack of more complex traffic patterns or real-time data to simulate attacks more realistically.	Desired more advanced attack simulation capabilities, especially for evolving DDoS tactics.
Network traffic data (traffic-data.csv) was successfully logged and formatted for later use.	Gained experience in configuring and logging network traffic data for anomaly detection purposes.	Some limitations in JMeter for fine-tuning traffic patterns for specific attack types.	Wanted more automated tools to validate and clean the generated traffic data.
The integration of benign and attack traffic scenarios was seamless.	Learned the importance of balancing real and attack traffic to ensure a diverse dataset for model training.	Lacked predefined templates or resources to simulate a broader range of attack vectors.	Hoped for real-time monitoring and automated analysis during data collection to quickly detect anomalies.
The traffic logs were comprehensive and provided enough data for the next sprint.	Realized how critical it is to organize the traffic data clearly and consistently for easier future analysis.	Insufficient time to explore deeper into traffic variation and multi-vector DDoS attacks.	Desired faster and more efficient ways to generate and analyze traffic data to meet the project deadlines.
Clear documentation was created for traffic simulation and data collection processes.	Understood the importance of thoroughly documenting the simulation settings for repeatability.	Limited access to tools for analyzing large-scale traffic data efficiently.	Wished for more time to fine-tune traffic patterns and better dataset validation processes.

Sprint 3 : Anomaly Detection Implementation

Liked	Learned	Lacked	Longed For
Share aspects of the sprint that you enjoyed or found particularly effective.	*Discuss lessons learned, whether they are related to processes, technical aspects, or teamwork.*	*Identify areas where the team felt a lack of resources, support, or information.*	*Discuss any desires or expectations that the team had but were not met during the sprint.*
The anomaly detection model using Isolation Forest was successfully implemented and produced meaningful results.	Learned how to apply the Isolation Forest algorithm for identifying anomalies in network traffic.	Lack of more complex anomaly detection algorithms for further model comparison.	Desired access to other advanced anomaly detection techniques for model diversification.
Results were visualized through anomaly plots, confirming the model's accuracy in detecting DDoS traffic.	Gained insights into how data visualization can be used to evaluate model effectiveness.	Needed more test data with varying attack patterns to validate model robustness.	Hoped for a more intuitive interface for visualizing and comparing model performance over time.
Team collaboration on code review and feature refinement was effective.	Understood the value of continuous validation to prevent overfitting in machine learning models.	Lacked extensive documentation on the parameters and setup for tuning the Isolation Forest model.	Wished for faster testing feedback on model adjustments to refine detection accuracy quicker.
The anomaly detection script was easy to integrate with the existing simulation setup.	Learned how to ensure model scalability by integrating it seamlessly into a simulated environment.	Limited computational resources hindered the speed of model testing and validation.	Desired better computational resources to enable faster iterations of the model tuning process.
Clear documentation of the anomaly detection setup helped replicate and extend the approach.	Realized the importance of hyperparameter tuning for achieving optimal model accuracy.	Insufficient automated tools to handle model testing and error detection at scale.	Longed for more extensive testing scenarios, especially for low-intensity DDoS attacks, to gauge the model's sensitivity.

Sprint 4 : Performance Testing and Integration

Liked	Learned	Lacked	Longed For
Share aspects of the sprint that you enjoyed or found particularly effective.	*Discuss lessons learned, whether they are related to processes, technical aspects, or teamwork.*	*Identify areas where the team felt a lack of resources, support, or information.*	*Discuss any desires or expectations that the team had but were not met during the sprint.*
Conducted performance tests using JMeter to evaluate system scalability under different traffic conditions.	Gained hands-on experience in stress testing cloud-based infrastructure and anomaly detection systems.	Lacked access to real-time traffic during tests to better simulate real-world load and attack conditions.	Desired real-time performance monitoring and alerting tools to track system performance during attacks.
Integrated CloudWatch alarms to monitor system performance and configured alerts for traffic anomalies.	Learned how to integrate AWS monitoring tools with the performance testing setup to track system health.	Needed more advanced testing cases to evaluate system response under complex attack scenarios.	Hoped for automated scaling adjustments based on traffic patterns to ensure system resilience under heavy load.
The integration of monitoring tools provided clear insights into system behavior during attacks.	Understood how dynamic scaling and load balancing can help mitigate DDoS attack effects.	Lack of pre-configured templates for performance testing made it harder to quickly assess various conditions.	Desired more fine-grained traffic alerting and feedback loops to quickly respond to issues during tests.
The system's response to DDoS attacks was effectively monitored and analyzed.	Learned the importance of continuous monitoring and analysis during performance testing to ensure system reliability.	Limited resources for simulating larger-scale attacks and evaluating system response in real-time.	Wished for quicker feedback from the testing phase to address issues before the next round of simulations.
The test documentation was well-organized, ensuring smooth setup and configuration for future tests.	Realized the need for balancing testing and optimization to ensure system efficiency during attacks.	Insufficient time allocated to testing all potential attack scenarios, limiting coverage of edge cases.	Longed for more advanced visualization tools to help interpret system performance data and attack mitigation effectiveness.