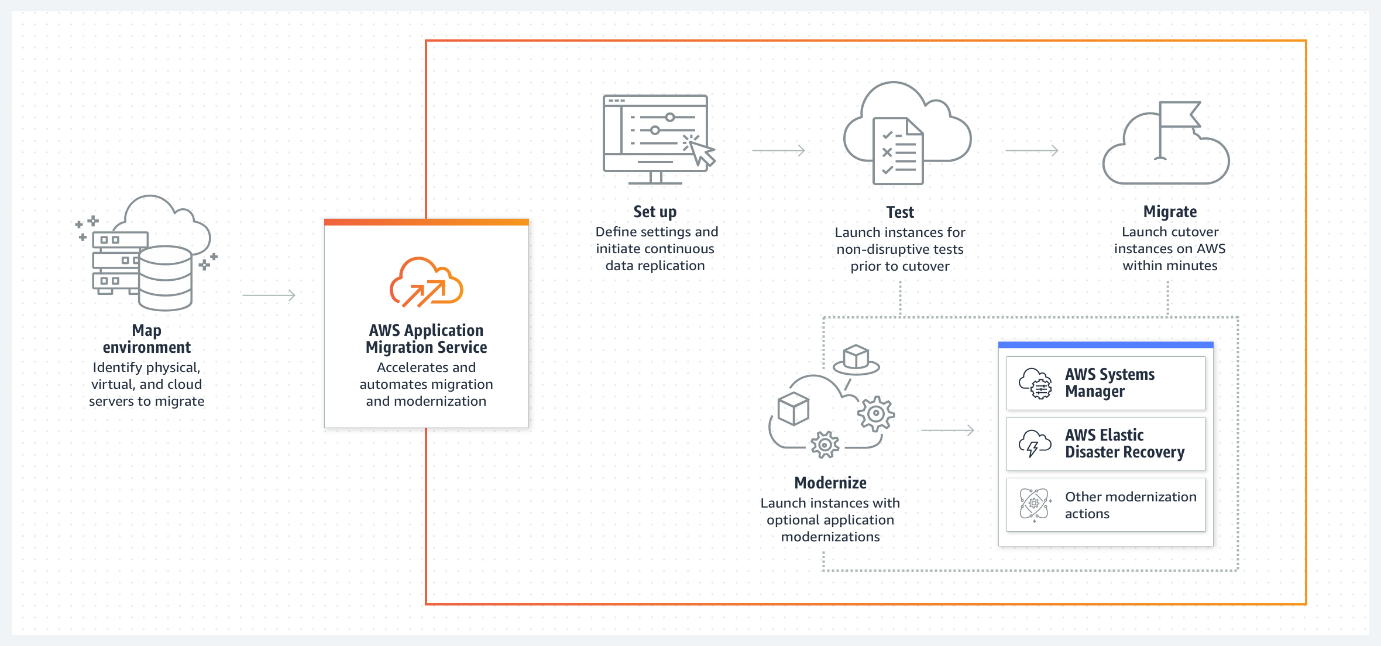
**Application Migration with AWS**

[Rehost AWS MGN]

**Introduction**

This strategy is also known as lift and shift. Using this strategy, you move your applications from your source environment to the AWS Cloud without making any changes to the application.



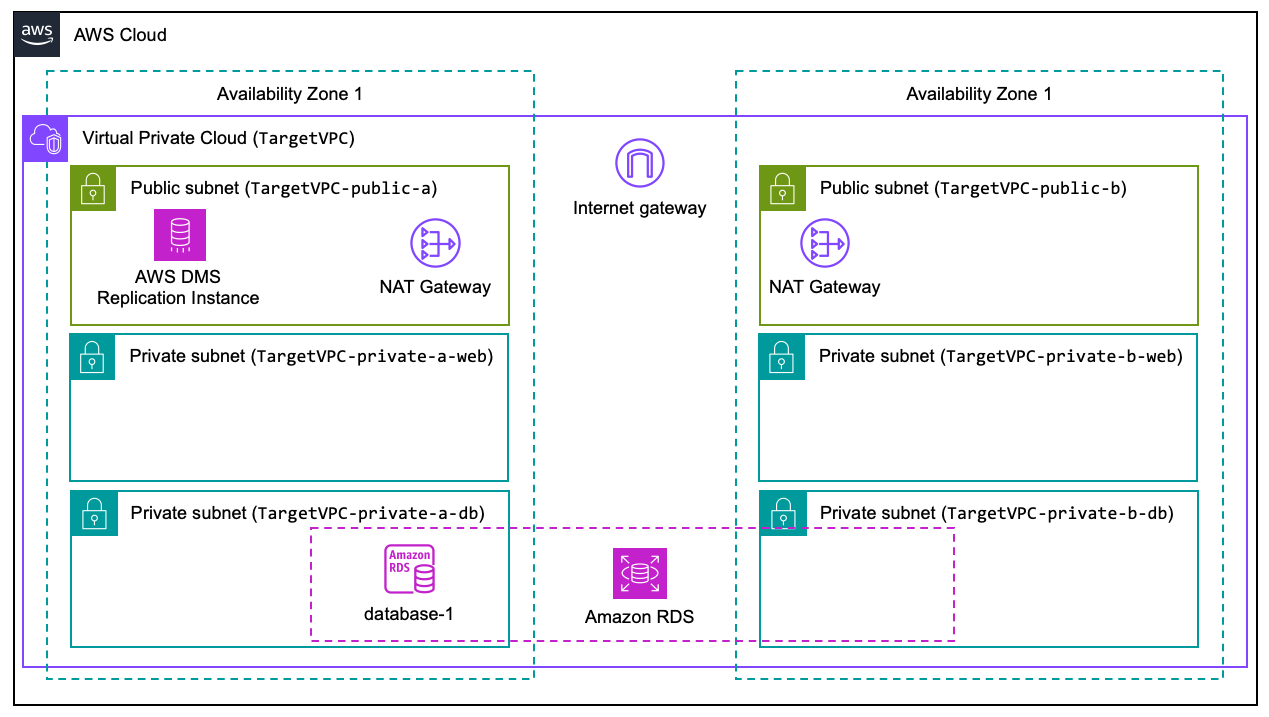
**A screenshot of a computer

Description automatically generated**

**Project Description – Scenario**

This project is a continuation from my previous project... I will use AWS MGN to rehost a web application from the source environment to the target environment, leveraging the Amazon Elastic Compute Cloud (EC2) as the destination compute platform.

**Architecture – Overview – Current Status**

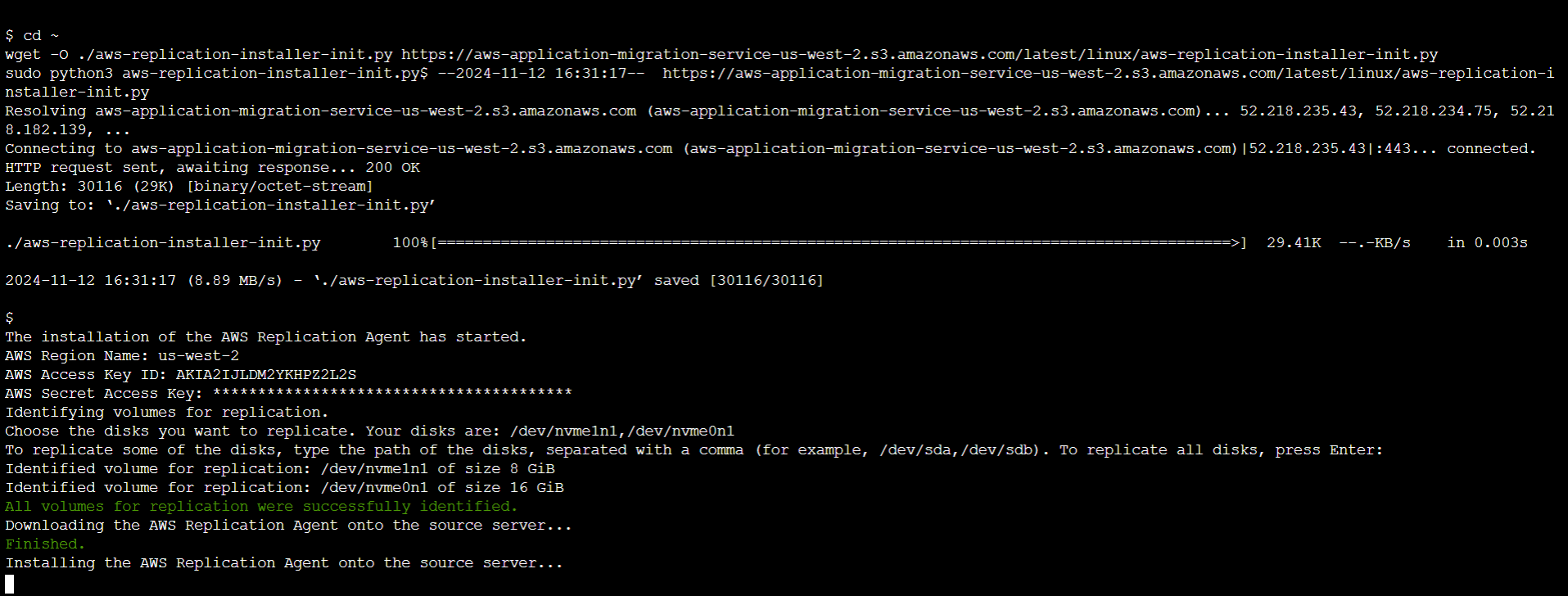


**Set up Application Migration Service**

The service must first be initialized. The initialization involves creation of IAM Service-Linked Roles and the Replication Server template. Replication servers are lightweight Amazon EC2 instances that are used to replicate data between your source servers and AWS, and the configuration is based on that template.

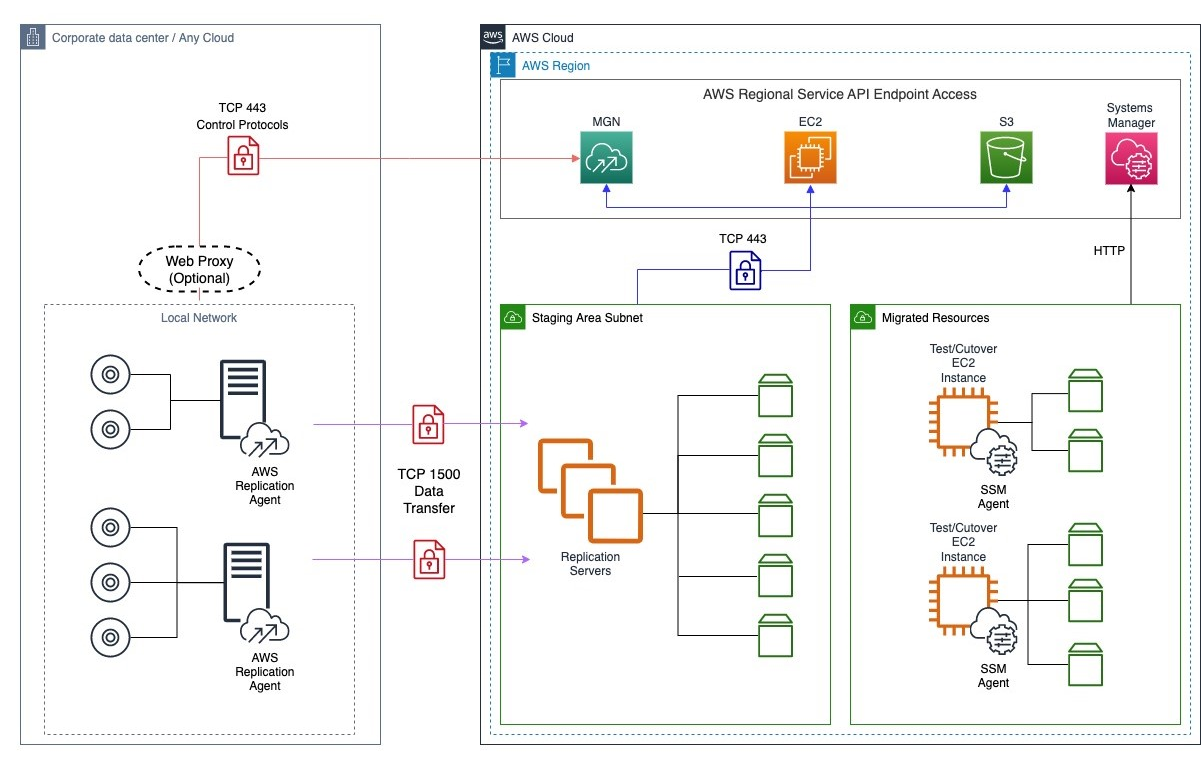
**Agent installation**

AWS Application Migration Service (MGN) rehosts the servers to AWS by replicating block-level data from the source target to the replication server.



**Deploying Agents to source Server – Depiction**

The [AWS Identity and Access Management (IAM**)**](https://aws.amazon.com/iam/) credentials with proper permissions are needed to communicate with AWS MGN APIs.



**Initial synchronization – Status**

**A screenshot of a computer

Description automatically generated**

**Parameter Store:**

This parameter store will store the configuration needed to use with AWS CloudWatch agent.

**CloudWatch & SSM agents are both active**

**A screenshot of a computer program

Description automatically generated**

**Validating the rehosted source server**

Validate if MGN can lift-and-shift it properly and confirm if the workload is fully functional after migration

**Launch test instance | Ready for testing**

Theserver has been successfully added to AWS MGN and data replication has started.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Launch test instance | Test in progress**

A test instance is currently being launched.

A screenshot of a computer

Description automatically generated

**The post launch actions**

The post launch actions can control and automate actions performed after the server has been launched in AWS. This capability could be used to execute automate testing, operating system conversion, tooling deployment, etc.

A screenshot of a computer

Description automatically generated

**CloudWatch Agent failed:** Unable to store the configuration needed to use with AWS CloudWatch agent, this means that I will not be able to publish any metrics from the migrated instance such as memory & swap utilization.

**SSM Agent Install:** Allow executing actions on launched servers.

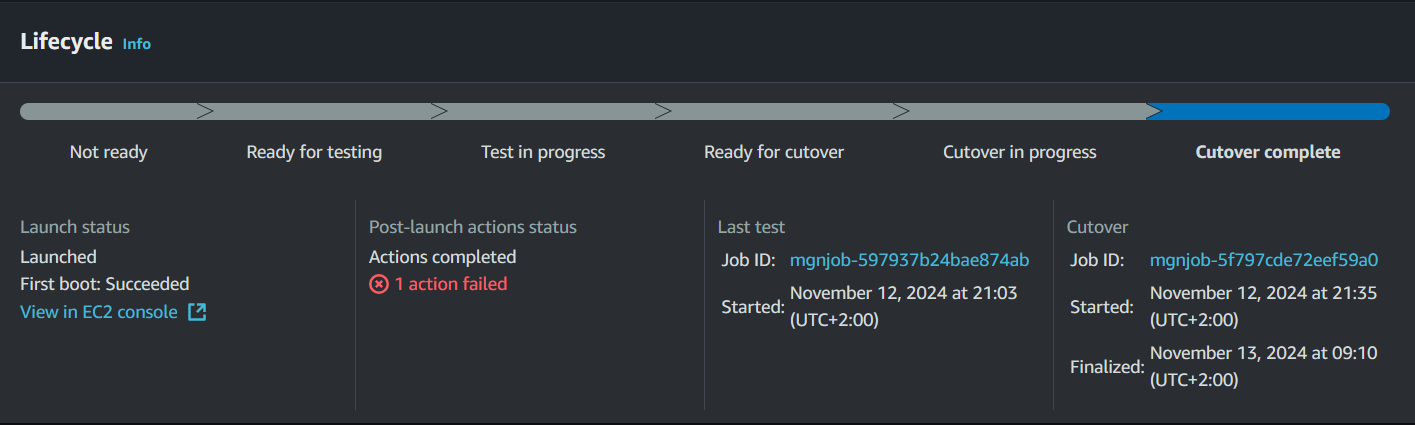
**Launch test instance | Cutover in progress**

A cutover instance is currently being launched for this server.

**A screenshot of a computer

Description automatically generated**

**Launch test instance | Finalized cutover complete**

****

After launching the cutover instance, the AWS Application Migration Service (MGN) created the final EC2 instance in the target environment. This instance was created based on the data from the source server, which means that the workload was still configured to the source environment I had to update the application configuration to use the database from the target environment, which was previously migrated to Amazon RDS database

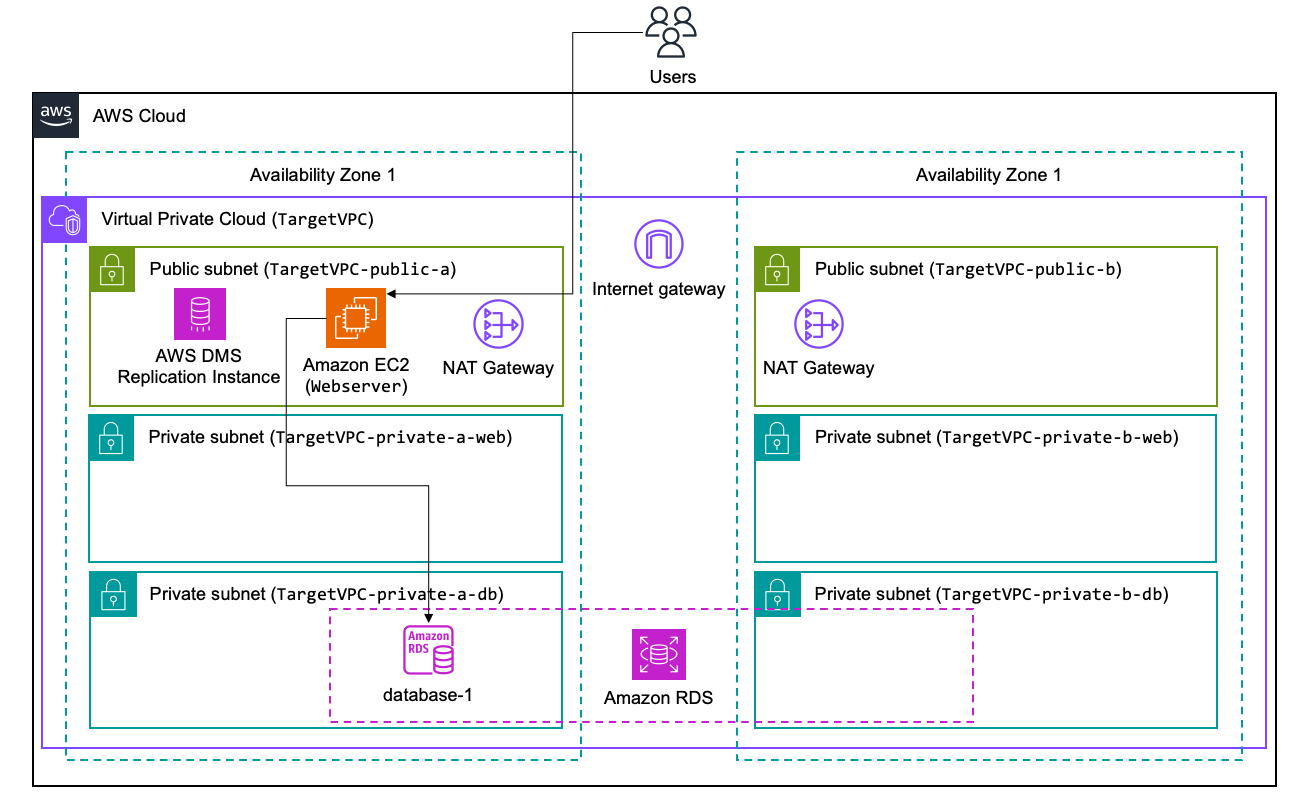
*Note: this is a continuation of a Previous Project.*

*[Created in the*[*Database Migration > Replatform - AWS Database Migration Service (DMS)*](https://catalog.us-east-1.prod.workshops.aws/workshops/c6bdf8dc-d2b2-4dbd-b673-90836e954745/en-US/02-database-migration/01-dms/)*project]*

**Summary**

I was able to rehost the source server to the target AWS environment using AWS Application Migration Service (MGN). After adjusting its configuration, the sample application I was able to successfully access the migrateddatabase with the updated informationand is now fully functional.

**Architecture Overview | Target environment**



**Conclusion | Re-cap**

1. Initialized and configured AWS Application Migration Service (MGN)
2. Deployed MGN Agent to the source application server
3. Customized the EC2 Launch Template to fine tune how the rehosted server would be configured in the target AWS environment
4. Optionally configured a post-launch action
5. Launched test instance with MGN to validate the rehost process was functional
6. Launched the cutover instance, and reconfigured the application to the new target environment
7. Finalized the cutover process, and executed housekeeping on AWS Application Migration Service (MGN) inventory

**Done!**