

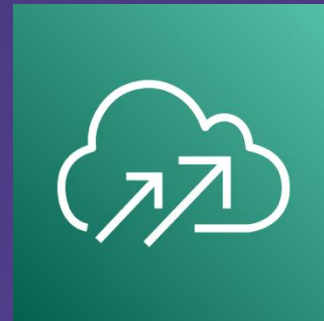
# *Taming the Tech Dinosaur: Overcoming Challenges in Migrating Legacy Servers to AWS*

André Vidigal

Cloud Consultant / Solutions Architect

tecRacer

AWS  
User Groups



# About Me

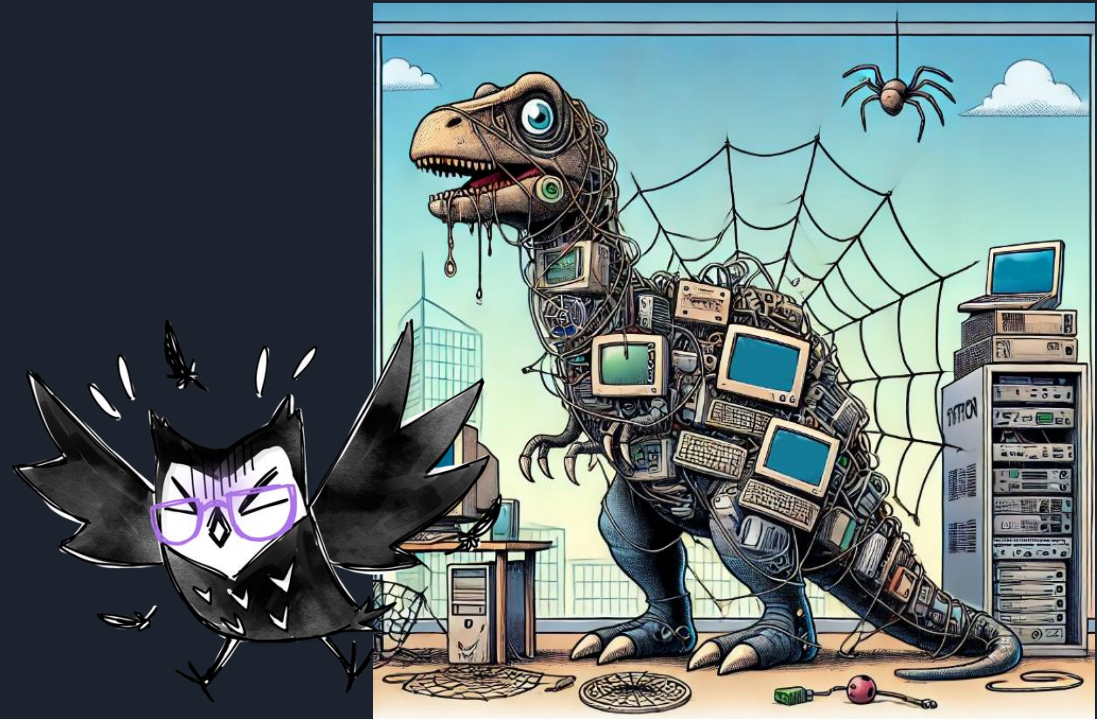
- André Vidigal
- In IT field since 2017!
- AWS Cloud consultant on tecRacer since 2022
- AWS Journey around the same time
- Last AWS service I worked on...



# Legacy Systems

SOMETIMES DATA CENTER = JURASSIC PARK

- A software / hardware can be considered a legacy system when is not updated for about 5 years nor receiving any more official support
- "If it's working, don't touch it"
- Jurassic Problems:
  - o Lack of compatibility
  - o Vulnerabilities
  - o Costs
  - o Etc.



# The AWS "Ambar"

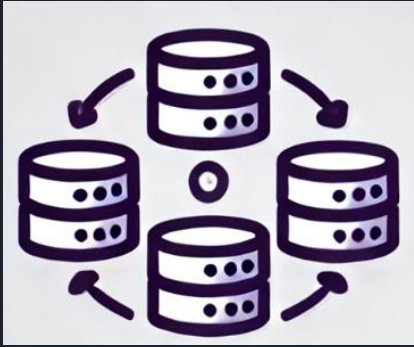
BRINGING DINOS TO THE CLOUD

## - AWS Application Migration Service

- Replicate your On-Premises systems onto AWS.
- Can be used to not only migrate legacy systems (legacy Agent) but to also migrate updated systems.
- Multi-OS Support.
- Maintain normal business operations while the replication happens.
- Free Period of 90 days (2,160 hours)



# Why AWS?



Replication



Upgradability



Managed Services



Modernization

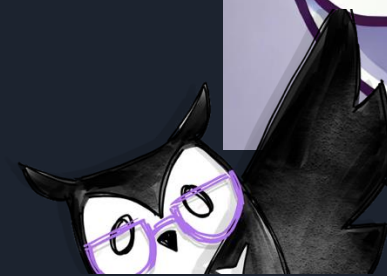




# Pre-requirements

## BEFORE DIGGING

- Ensure that the source [server is supported](#) by AWS
  - o If it's a legacy system (i.e., Windows Server 2003, 2008, etc.) you will need the legacy agent
- 4GB free disk and at least 300MB of free RAM
- In case of legacy Windows systems must have .NET Framework version 3.5 or 4.5 depending on the system
- For legacy, Non-Nitro based instances only.
- Create a IAM user or IAM role with the policy "[AWSApplicationMigrationAgentInstallationPolicy](#)"
- Open TCP Port 443 and TCP Port 1500



# First Steps

LET'S START DIGGING

[Application Migration Service](#) > Set up Application Migration Service

## Set up Application Migration Service

### Service initialization

In order to use Application Migration Service in this region, the service must be initialized on your AWS account. Once the service is initialized, you can modify the default settings.

By continuing, you are allowing application Migration Service to create a replication and the launching of migrated servers. [Learn more](#)

[View roles](#)

### AWS Replication Agent installation

#### 1. Select your operating system

☐ Linux

☒ Windows

☒ Legacy OS: Windows Server 2003, Windows Server 2008 or Windows Server 2008 R2

#### 2. Select your replication preferences [Info](#)

Replicate all disks

#### 3. Provide the required credentials [Info](#)

Create an IAM role or user with the `AWSApplicationMigrationAgentInstallationPolicy` policy.

IAM access key ID

IAM secret access key

This form does not send the secret – it only adds it to the installation command you can copy

[Show](#)

Session token

Session token is only required when using temporary credentials

#### 4. User provided resource id - *optional* [Info](#)

#### 5. Download the [installer](#) [↓](#) onto your source server (or copy it there after downloading)

If you need to validate the installer hash, the correct hash can be found here:

[https://aws-application-migration-service-hashes-eu-west-1.s3.eu-west-1.amazonaws.com/latest/windows\\_legacy/AwsReplicationWindowsLegacyInstaller.exe.sha512](https://aws-application-migration-service-hashes-eu-west-1.s3.eu-west-1.amazonaws.com/latest/windows_legacy/AwsReplicationWindowsLegacyInstaller.exe.sha512)

#### 6. Copy and input the command below into the PowerShell command-line on your source server

```
.\AwsReplicationWindowsLegacyInstaller.exe --region eu-west-1
```

[Copy](#)

# The "Dinosaur"

WORKING ON THE SOURCE SERVER

```
Administrator: Windows PowerShell (3)
PS C:\Users\Administrator\Desktop> .\AwsReplicationWindowsLegacyInstaller.exe --region eu-west-1 --aws-access-key-id AKI
AW3MEBDKNCW4EZBI2 --aws-secret-access-key [REDACTED] --no-prompt
The installation of the AWS Replication Agent has started.
Verifying that the source server has enough free disk space to install the AWS Replication Agent (a minimum of 2 GB of f
ree disk space is required).
Identifying volumes for replication.
Disk to replicate identified: c:\ of size 127 GiB
All volumes for replication were successfully identified.
Downloading the AWS Replication Agent onto the source server...
Finished.
Installing the AWS Replication Agent onto the source server...
Finished.
Syncing the source server with the Application Migration Service Console...
Finished.
The following is the source server ID: s-0a5d566f054461956.
You now have 1 active source server out of a total quota of 150.
Learn more about increasing source servers limit at https://docs.aws.amazon.com/mgn/latest/ug/MGN-service-limits.html
The AWS Replication Agent was successfully installed.
PS C:\Users\Administrator\Desktop> S_
```





# Migration Status

Data replication status [Info](#)

Initiating

Replication progress

0%

Replication type

Agent based

Total replicated storage

0 of 127 GiB

Lag

-

Backlog

-

Replication initiation steps

1

Create security groups

2

Launch Replication Server

3

Boot Replication Server

4

Authenticate with service

5

Download replication software

6

Create staging disks

7

Attach staging disks

8

Pair Replication Server with AWS Replication Agent

9

Connect AWS Replication Agent to Replication Server

10

Start data transfer

Last start time

October 29, 2024 at 12:38 (UTC+0:-00)

Next attempt time

October 29, 2024 at 14:27 (UTC+0:-00)

Actions ▼

Replication ▼

Test and cutover ▲

Testing

Launch test instances

Mark as "Ready for cutover"

Revert to "Ready for testing"

Cutover

Launch cutover instances

Finalize cutover

Revert to "Ready for cutover"

Other

Edit Launch Settings

Edit post-launch settings

Terminate launched instances

# Some rocks are bound to be found

HERE'S SOME HELP

Unfortunately, things could go wrong... But there's always hope and coffee ☺

- Check network and firewalls
  - Use VPN if necessary
  - Security Groups on AWS side
- In case of legacy systems like Windows, check pre-installed drivers or install through registry
- Look out for Lag / Backlog (Read/Write VS Bandwidth)

Lag

-

Backlog

-



# My Jurassic Adventure

REAL-WORLD PROBLEMS

Connect

Instance state ▾

Actions ▲

Launch instances ▾


Get instance screenshot [Info](#)

Instance screenshot

🔄

Download

i-0dc544a1b7d3b18b2 (sigodbdev) on 2024-09-24 at T14:28:22.092 +01:00



EC2 > Snapshots > snap-067f7e3cf7b4064a6

snap-067f7e3cf7b4064a6

Snapshot ID

📄 snap-067f7e3cf7b4064a6

Started

📄 Tue Oct 29 2024 14:56:05 GMT+0000 (Western Eur Standard Time)

▼ Source volume

Volume ID

📄 vol-09f42b20dbce06050

▼ Encryption

Encryption

Not encrypted

🔄

Delete

Actions ▲

Create volume from snapshot

Create image from snapshot

Copy snapshot

Snapshot settings ▶

Archiving ▶

Owner

📄 471112686234

Description

📄 volume

KMS key ARN

-

AWS

User Groups

# Next Steps

"IT IS NOT THE STRONGEST OF THE SPECIES THAT SURVIVES, NOT THE MOST INTELLIGENT THAT SURVIVES. IT IS THE ONE THAT IS THE MOST ADAPTABLE TO CHANGE"



Refactor and Upgrade



Modernization



# Final thoughts

- Not a perfect solution, especially for older systems.
- Be prepared for troubleshooting and trial and error.
- If able, perform any kind of backup / snapshot on source servers.
  - o To help with backlog, have a replica of the source server on premise.
- Don't be afraid to test and recreate replicated instances



# Thank you!

## Q&A Time



LinkedIn

