Architecture & Infrastructure design

HPC Upstream Compute platform - PoC

**Document Signoff**

**Document Signoff**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Approved By | Approved Date | Approved By | Approved Date |
| 1.4 | **Name:** Jithendra Balakrishnan (Automation and DevOps Lead)  **Organisation:**  Dyson | 21/06/2023 | **Name:** Suryanarayana Vaddadi (Senior Manager Solution Architect)  **Organisation:**  Dyson | 23/06/2023 |
| Signature | Jithendra Balakrishnan | | Suryanarayana Vaddadi | |

Document Revision List

|  |  |  |  |
| --- | --- | --- | --- |
| Revision No. | Revision Date | Author | Revision Description |
| V 0.1 | 13/02/2023 | Prithwiraj | Initial Design Document |
| V 0.2 | 27/02/2023 | Aruneet | Inter-Connectivity within the Parallel Cluster |
| V 0.3 | 28/02/2023 | Jeelan | Updated the Implementation Section |
| V 0.4 | 01/03/2023 | Abdul | Updated the Configuration File for the Parallel Cluster |
| V 0.5 | 02/03/2023 | Ashwin | Updated the Terraform script to create the base infrastructure |
| V 0.6 | 03/03/2023 | Aruneet | Updated the Configuration Tags for the Head Node and Compute Nodes |
| V 1.0 | 03/03/2023 | Prithwiraj | Updated the Configuration Tags for the Slurm Job Queue |
| V 1.1 | 06/03/2023 | Prithwiraj | Updated few things as per design comments, verified the sizing details with AWS |
| V 1.2 | 06/03/2023 | Prithwiraj | Updated few things as per design comments from Dyson Cloud Architect, Added SGs |
| V 1.3 | 14/04/2023 | Prithwiraj | The cluster does not need to be domain joined, it should be able to query the Active Directory to validate users |
| V 1.3 | 14/04/2023 | Prithwiraj | Text changed from Windows 2022 to Windows Server 2022 |
| V 1.3 | 26/04/2023 | Prithwiraj | Updated the VPC name, and removed the place holder from section 4.1 |
| V1.4 | 09/06/2023 | Prithwiraj | Added a new section 8.1, 8.1.1, 8.1.2 for Custom GUI to create workstation as per user requirement  Also have added Sl#9 in Appendix-2 to show what is being provided in PoC and what will be given in post-PoC  Also have modified Sl#2 in Appendix-2 to introduce Custom GUI to support the ‘Workstation Selection’ criteria instead of EngineFrame |
| V1.4 | 19/06/2006 | Prithwiraj | Explained the 8.1 section to describe more about seamless workflow of the GUI application in post-PoC phase  Also have modified Sl#9 in Appendix-2 to describe features in PoC and Post-PoC |

Contents

1 Contents 4

2 Introduction 7

2.1 Purpose of the Document 7

2.2 AS-IS Model 7

2.3 Scope 8

2.4 Out of Scope 9

2.5 Approach 9

2.6 Design Consideration 9

2.7 Intended Audience 10

3 AWS Regions 11

4 AWS Accounts and Environments 12

4.1 AWS Account 12

4.2 Environment 12

5 Network Architecture and Connectivity 13

5.1 Overall Network Architecture 13

5.2 VPC/Subnet Structure for HPC workload using Parallel Cluster 14

5.3 Connectivity with Dyson on-prem 14

5.4 Connectivity for users using Dyson Laptop 15

5.5 IP Range and Integration with Dyson Network 16

5.6 IP CIDR Block for AWS Environment 16

5.7 Access to AWS Console 16

6 Inter-Connectivity within the Cluster 17

7 Security 18

7.1 AWS general approach for security 18

7.2 AWS Console and Service Access Management 18

7.3 Access Management using Azure AD 19

7.4 Patching and Antivirus 20

7.5 Vulnerability Check 20

7.6 Network Security 20

7.7 Governing AWS Environment 21

7.8 Data Encryption 21

7.9 Logging and Monitoring 21

8 AWS Infrastructure provisioning and Management 23

8.1 EC2SelfService Portal 23

8.1.1 Architecture 23

8.1.1 Workflow 25

9 Storage Considerations 26

9.1 Storage Sizing and Recommendations 26

10 Availability, Resilience and Capacity Management 27

10.1 Availability and Resilience 27

10.2 Capacity Management 27

11 Regulatory Compliance 28

12 Implementation 29

12.1 Approach 29

12.2 Implementation Details 30

12.3 App Installation 32

12.4 List Of Services 33

12.5 Subnet to Route Table Mapping 33

12.6 Route Table Details 33

12.7 Subnet to NACL Mapping 34

12.8 NACL Detail 34

12.9 Instance Details 34

12.10 Instance to Security Group Mapping 34

12.11 Security Group Detail 35

13 Appendix 36

Abbreviations

|  |  |
| --- | --- |
| Abbreviation | Description |
| ACL | Access Control List(s) |
| AWS | Amazon Web services |
| ISO | International standards organisation |
| ITSM | Information Technology Service Management |
| SOC | Security Operation Centre |
| TCS | Tata Consultancy Services |
| Dyson | Dyson Technology Limited |
| HPC | High Performance Computing |
| PoC | Proof Of Concept |
| AD | Active Directory |
| SSO | Single Sign On |
| SOCA | Scale Out Computing on AWS |

# Introduction

This document describes the AWS infra design specifications for DYSON-HPC-UPSTREAM Environment for PoC phase

* Provision the target environment in Ireland region under Dyson’s existing AWS landing zone
* Provision the target environment for Workstations, HPC, NICE DCV /Engine Frame
* Migrate the on-premises data to HPC environment for the Use Cases/Scenarios
* Deploy Applications in HPC environment for running the PoC Use Cases/Scenarios
* Test the Use Cases/Scenarios
* Provide support for the HPC and workstation environment during the use case execution.

## Purpose of the Document

The primary purpose of this document is to provide a holistic view of AWS Hosting platform and its functional delivery models.

## AS-IS Model

* **Existing Network Design:** Dyson’s corporate offices are connected to the AWS environment through SD-WAN. The AWS Account is sub-divided into OU level accounts. There are separations for Prod and Non-Prod accounts. The Transit Account is responsible for the SD-WAN connection at the AWS side and the Cisco CSRs are connected to the Transit Gateways as per Figure-1.

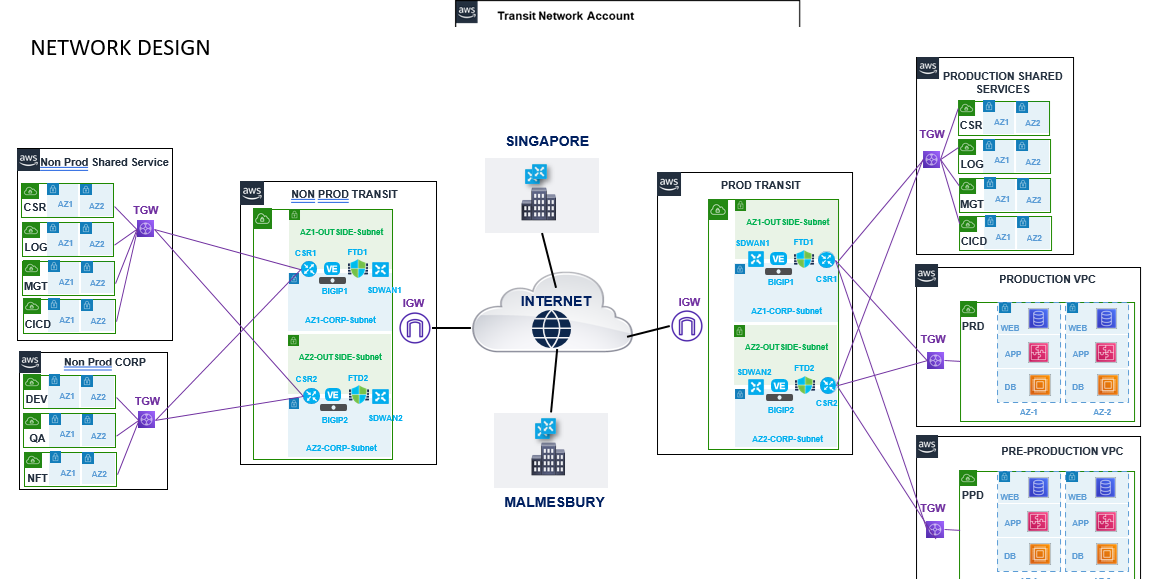


Figure 1 – DYSON Network Connectivity Overview

* **On-prem HPC Landscape:** There are two clusters in on-prem environment. The users are sitting with their local machine and connected to the clusters and underlying NFS storage. Domain-based authentication is performed using Azure AD as per Figure-2.

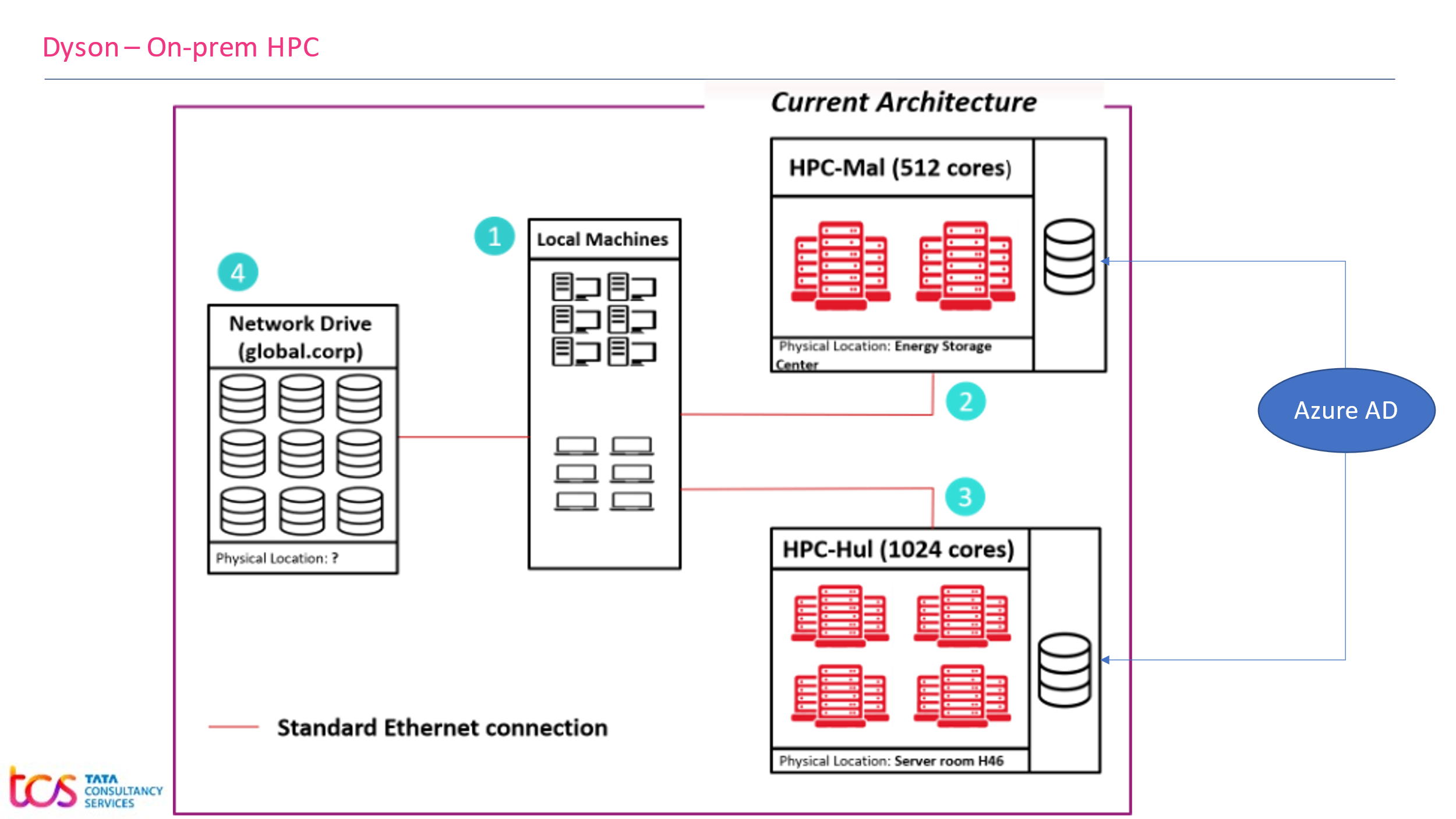


Figure 2 – DYSON On-Prem HPC Platform

* **On-Prem HPC Sizing Details:**

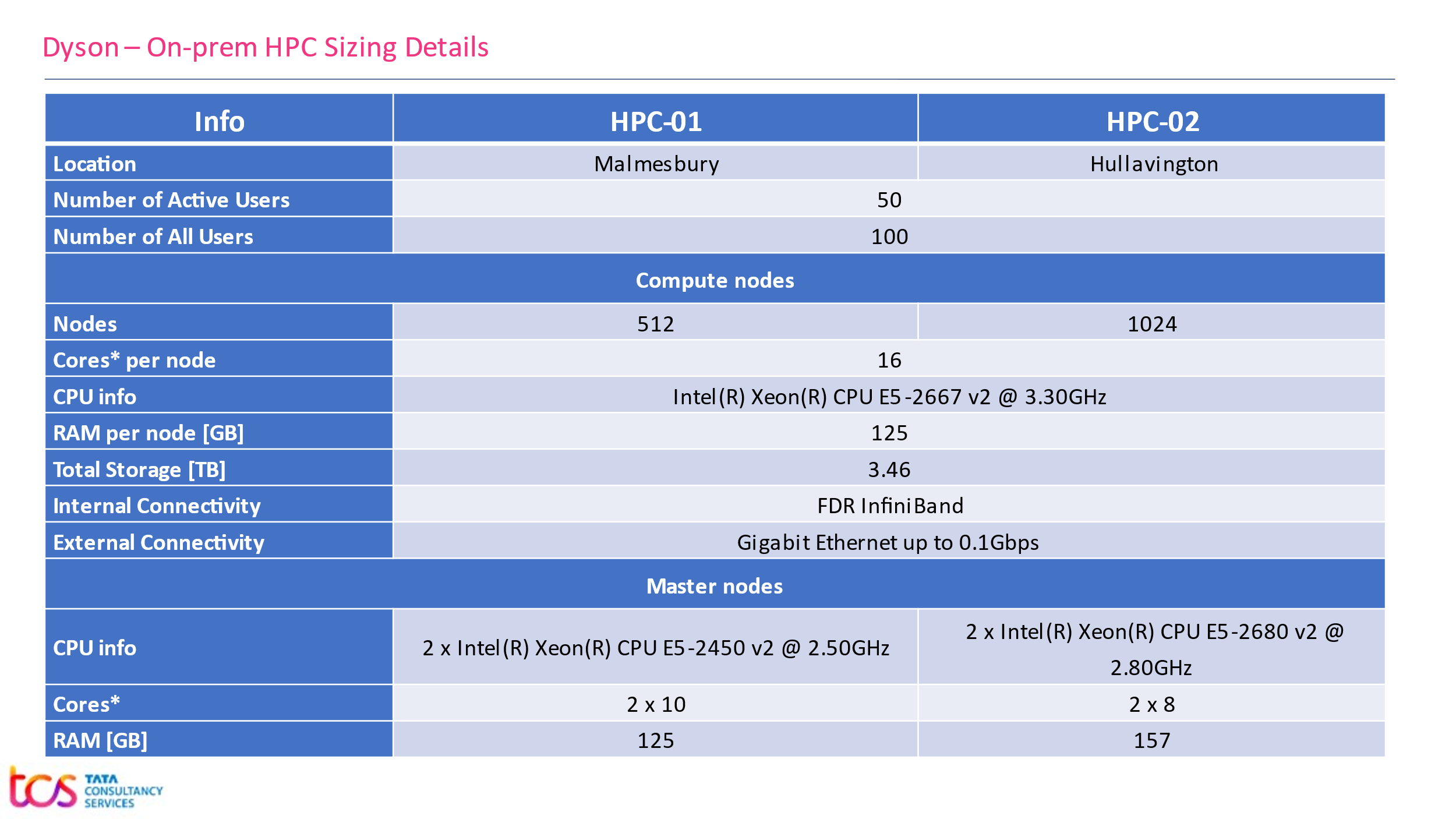


Figure 3 – DYSON On-Prem HPC Sizing

## Scope

Scope of this document is to describe the building blocks to setup AWS-HPC infrastructure for hosting DYSON-HPC-UPSTREAM platform for the PoC. This document describes key infrastructure components as

* AWS Regions and Locations
* Accounts and Subscriptions
* AWS Environments
* Network Connectivity
* VPC/Subnet Structure
* Authentication/Authorization
* Infrastructure as a Code
* HPC Cluster (AWS Parallel Cluster)
* Storage (EBS, FSx for NetApp ONTAP)
* Security and monitoring

## Out of Scope

* Design and implementation of any third-party tools
* Procurement of AWS Cloud services by making an Enterprise Agreement with AWS
* Procurement of network links/bandwidths, addressing/resolving issues with existing Dyson network
* Definition of security and compliance policies as per Dyson existing Cloud Standards
* Responsibility on any third-party vendor deliverables will remain with Dyson (Dyson will facilitate communications with any third-party vendor)
* Definition and implementation of ITSM process for cloud
* Configuration changes to ITSM tool
* Decommissioning of existing Infrastructure
* AI/ML related services

## Approach

Approach is based on the Dyson’s business requirement to provision a quick PoC model considering the Future Operating Model using AWS and AWS-HPC as the platform.

## Design Consideration

Consideration is made by analyzing two approaches – SOCA and AWS Parallel Cluster. Based on our analysis, AWS Parallel Cluster has been found as the best suited approach considering the business requirements, success criteria, future support and advisory from AWS.

## Intended Audience

This document will enable Dyson Stakeholders to understand TCS recommended technical design specifications.

Stakeholders:

* Solution Architect
* Cloud Architect
* Network Architect
* Domain Architects
* Security Architect
* IT Infrastructure Architect
* Service Delivery Owners

# AWS Regions

Regions in AWS are geographically distributed and isolated locations, consisting of one or more AWS Data centers connected to each other through low latency and high throughput network. Each AWS region is paired with another region within the same geography, together making a regional pair.

AWS region is determined based on the user’s geographical presence and considering the requirement criteria.

Dyson has their AWS footprints in **Ireland** region.

Below region is identified based on the hosting requirements of DYSON-HPC-UPSTREAM platform to AWS, such that AWS-HPC services availability, commercial implications, and feasibility to execute the given use cases as per the Business Requirement Document.

|  |  |  |
| --- | --- | --- |
| **Geography** | **Region** | **Primary/Secondary** |
| EU | Ireland | Primary (DC) |

TCS plans to host and manage the entire services to meet Dyson requirement in AWS EU – Ireland region across multiple zones as a primary data center. Our solution does not include any DR strategy for the PoC.

AWS Cloud EU region Data Centers are Tier 4, and every Availability Zone is equipped with High Availability compute clusters to provide the defined services and highly resilient network connected to data centers. The Multiple Availability Zones in a region are close enough for low latency connectivity.

# AWS Accounts and Environments

AWS organization structure is in place for Dyson. Dyson will be sharing an existing AWS account in “Non-Prod RDD” for this PoC. There are multiple shared VPCs available in the account however, for the sake of simplicity a dedicated VPC will be created for the PoC.

Managing the OU level control policy and guardrails are not considered here. Existing policies will be considered for the PoC.

## AWS Account

Shared account “Non-Prod RDD” (Account Id: **178920580136**) will be used for this PoC.

## Environment

DYSON-HPC-UPSTREAM PoC program we will leveraging the Non-Prod RDD account.

* **Non-Prod Account**

# Network Architecture and Connectivity

## Overall Network Architecture

Diagram depicts the existing network architecture of Dyson’s AWS environment and connectivity with their respective branches. A new VPC has been proposed under “Non-Prod RDD” account for the PoC which is highlighted below.

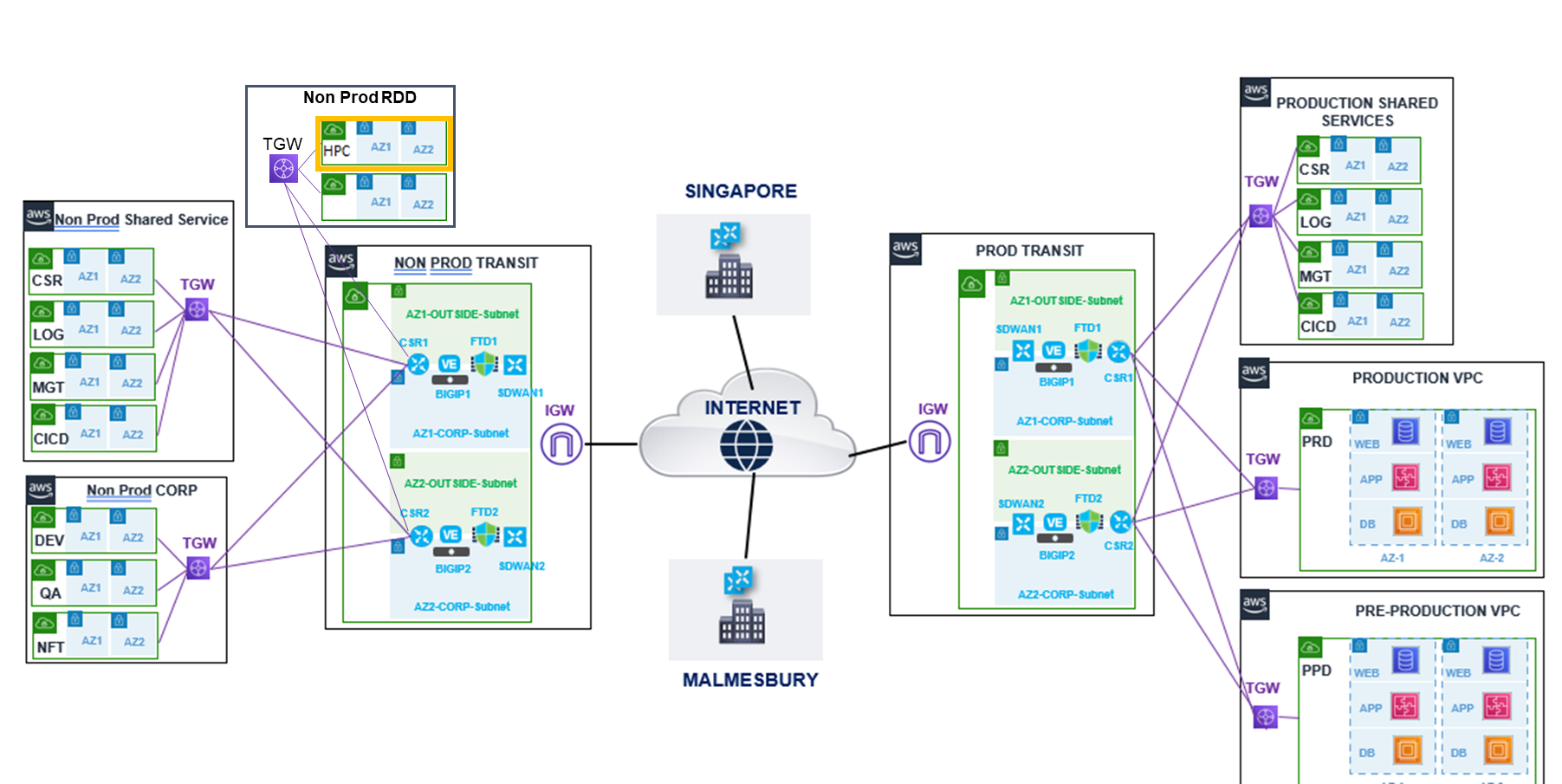


Figure 4 – DYSON-HPC-UPSTREAM Network Connectivity Overview

**Key points covered as part of the initial architecture design**

1. Centralized Transit Network account acts as a Network Hub. All VPCs within AWS environment are connected to Transit Gateway. The Transit Gateway is connected to SD-WAN to establish the connectivity between the on-prem and AWS environment
   * + The “Non-Prod Transit Network” account is being used for all non-prod workloads i.e., DEV, QA, NFT, PoC etc.
     + The “Prod Transit Network” account is being used for all production workloads.
2. The VPC under Transit Network account acts as an entry point from Dyson to AWS environment.
3. HPC VPC will be created under “Non-Prod RDD” account for the said PoC.
4. Users are connected to AWS environment as below

* Dyson users are connected to AWS environment through SD-WAN.
* TCS Development team will be connected to AWS environment through Client VPN.

1. To migrate data from Dyson to AWS, the SD-WAN is being used.

## VPC/Subnet Structure for HPC workload using Parallel Cluster

The following diagram depicts VPC and Subnet structure of HPC workload for the PoC

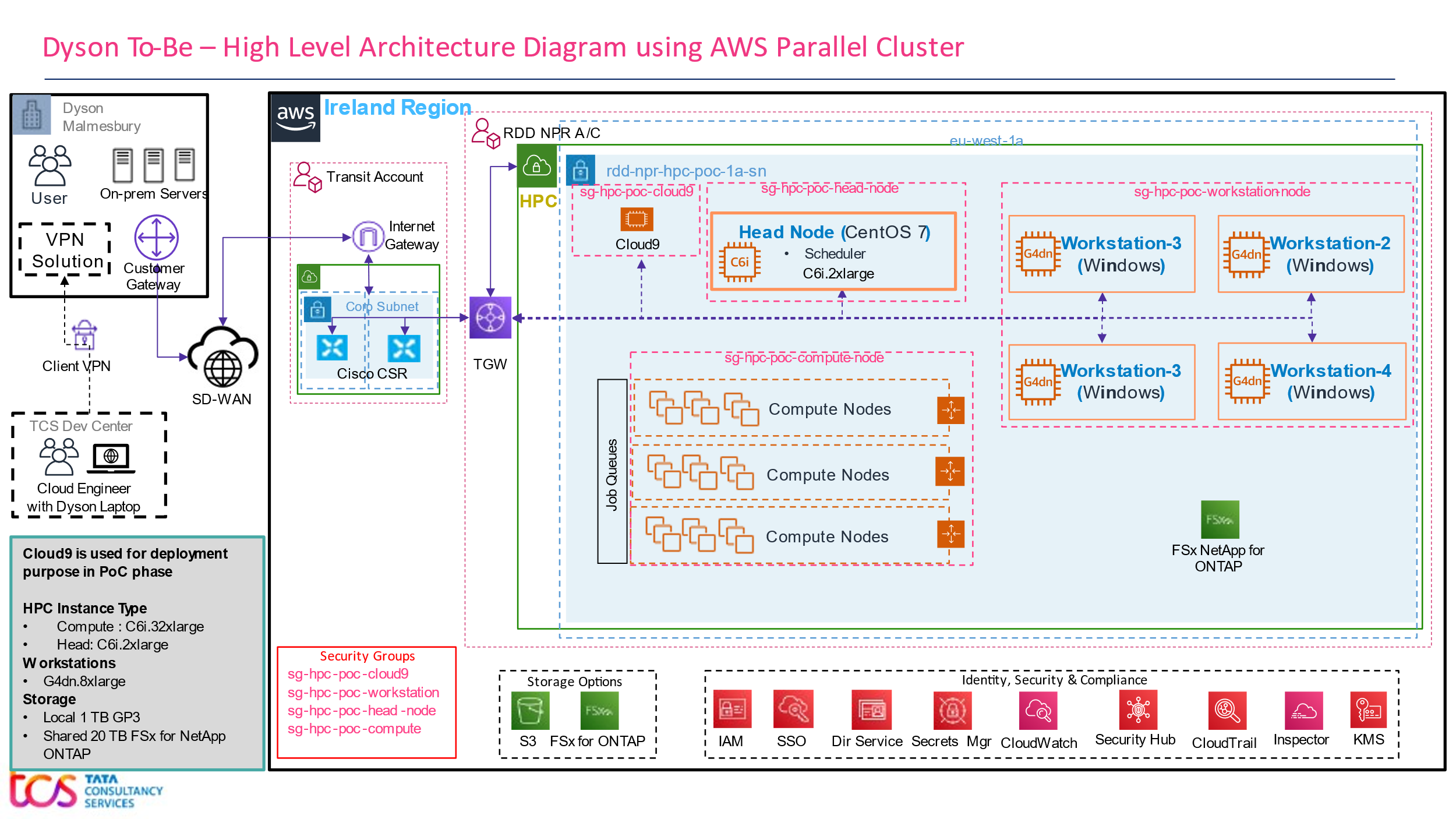


Figure 5 – DYSON-HPC-UPSTREAM VPC Overview

Below are some of the key points covered as part of the initial architecture design

1. VPCs will be spanned in one Availability Zone in Ireland region in PoC phase.
2. VPC and Subnet structure

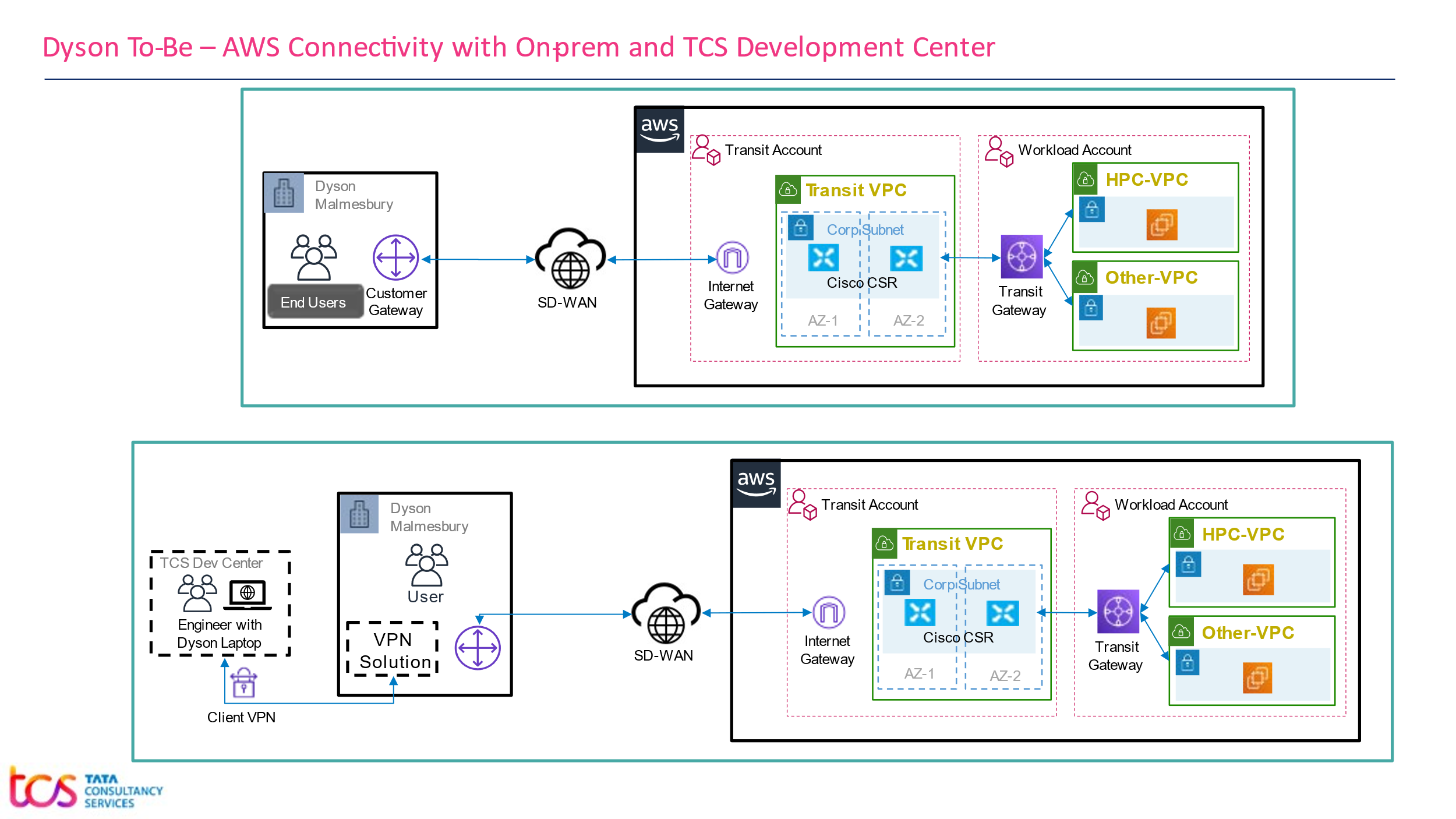
* No Public Subnet:
* 1 Private Subnet:
  1. One Head Node (CentOS 7) will have Job Scheduler, NICE DCV server will be installed on it.
  2. Four Workstations (Windows); NICE DCV server will be installed.
  3. CentOS based on-demand Compute Nodes under auto-scaling group

1. All internet traffic will pass through firewall deployed in the Transit Account
2. Access to Workstation instances will be enabled through AWS SSO authentication through console. Alternatively – the user should be able to RDP/DCV to the Windows based Workstations.

## Connectivity with Dyson on-prem

Below are some of the key points to access AWS resources from Dyson office location (Malmesbury):

* Dyson’s Malmesbury location is connected to the Internet Gateway inside the Transit Account using SD-WAN over internet.
* Cisco CSRs from the Transit Account are connected to the Transit Gateway in the Workload Account where the HPC cluster will be created.
* RDP/DCV ports to be opened from Dyson to AWS VPC.
* Dyson user may use the DCV client from their local machine to remotely visualize the workstations.
* Alternatively, Dyson user may log on to their corresponding Workstations and access the cluster.



*Figure 6 – High Level Diagram of Connectivity from Dyson to AWS over SD-WAN*

* Data transfer between Dyson and AWS will happen over intranet.

## Connectivity for users using Dyson Laptop

Below are some of the key points to access AWS resources by the users having Dyson Laptop:

* User will have his/her Dyson laptop with Client VPN installed, which will be used to connect the Dyson’s N/W, and from there AWS cloud can be accessed.
* User will have their User ID created in Dyson’s existing Active Directory.
* DCV ports to be opened from Dyson to AWS VPC.

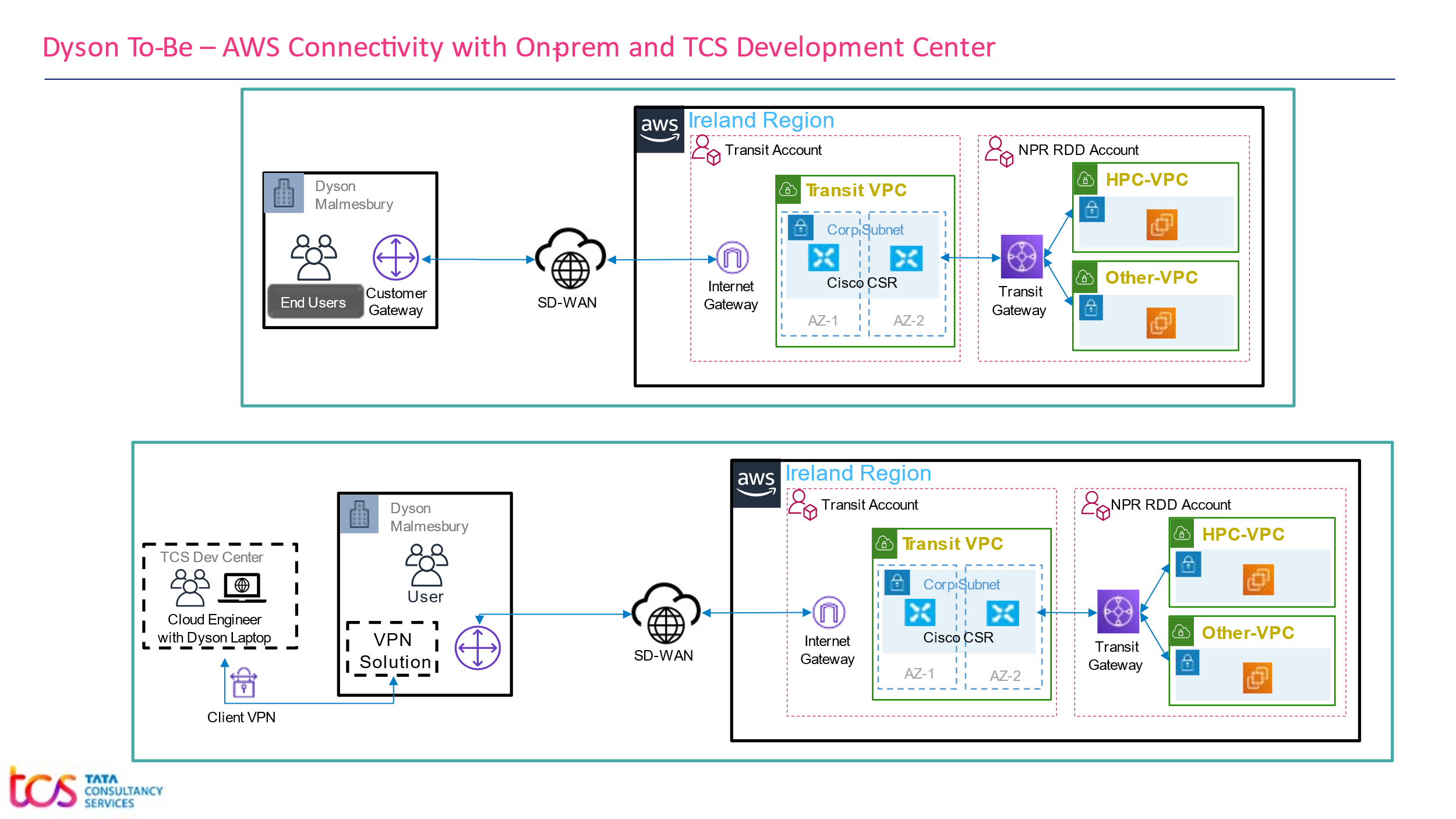


Figure 7 – Connectivity between AWS and Users using Client VPN

**Dependency to establish Connectivity**

To establish the connectivity between AWS and Dyson Corporate Network, need to take the following points into consideration

* Involvement of cloud team to attach the Transit Gateway to the newly created VPC for HPC Cluster to establish on-Prem to AWS connectivity for Dyson user.
* IP range of Dyson Corporate N/W should not overlap with CIDR Block **10.221.90.0/24** which is planned to be used for the AWS HPC Cluster for this PoC.

## IP Range and Integration with Dyson Network

As per proposed architecture all services like Head Node, Workstations will be available over Dyson’s corporate network using SD-WAN. Data Transfer will be done using SD-WAN, its inbuilt encryption to ensure data security while in transit.

AWS resources must meet Dyson specific security/compliance requirements, e.g., fully patched, Anti-Virus and Firewall Protection in order to establish the connection with Dyson Corporate Network. Domain join may be considered during the PoC phase to establish the connection between Dyson on-premises resources and AWS Cloud.

## IP CIDR Block for AWS Environment

IP CIDR Block for AWS HPC VPC: **10.221.90.0/24**

|  |  |
| --- | --- |
| **AWS VPC** | **(EU-West-1)** |
| HPC VPC | **10.221.90.0/24** |

## Access to AWS Console

AWS Console will be accessible from Dyson Corporate Network over SD-WAN.

* Console access will be centrally managed through AWS Single Sign On.
* MFA will be enabled for all Console Users.
* Least Privilege strategy will be enforced by using AWS Role and Policy.

# Inter-Connectivity within the Cluster

Key points to describe the workflow and cluster inter-connectivity.

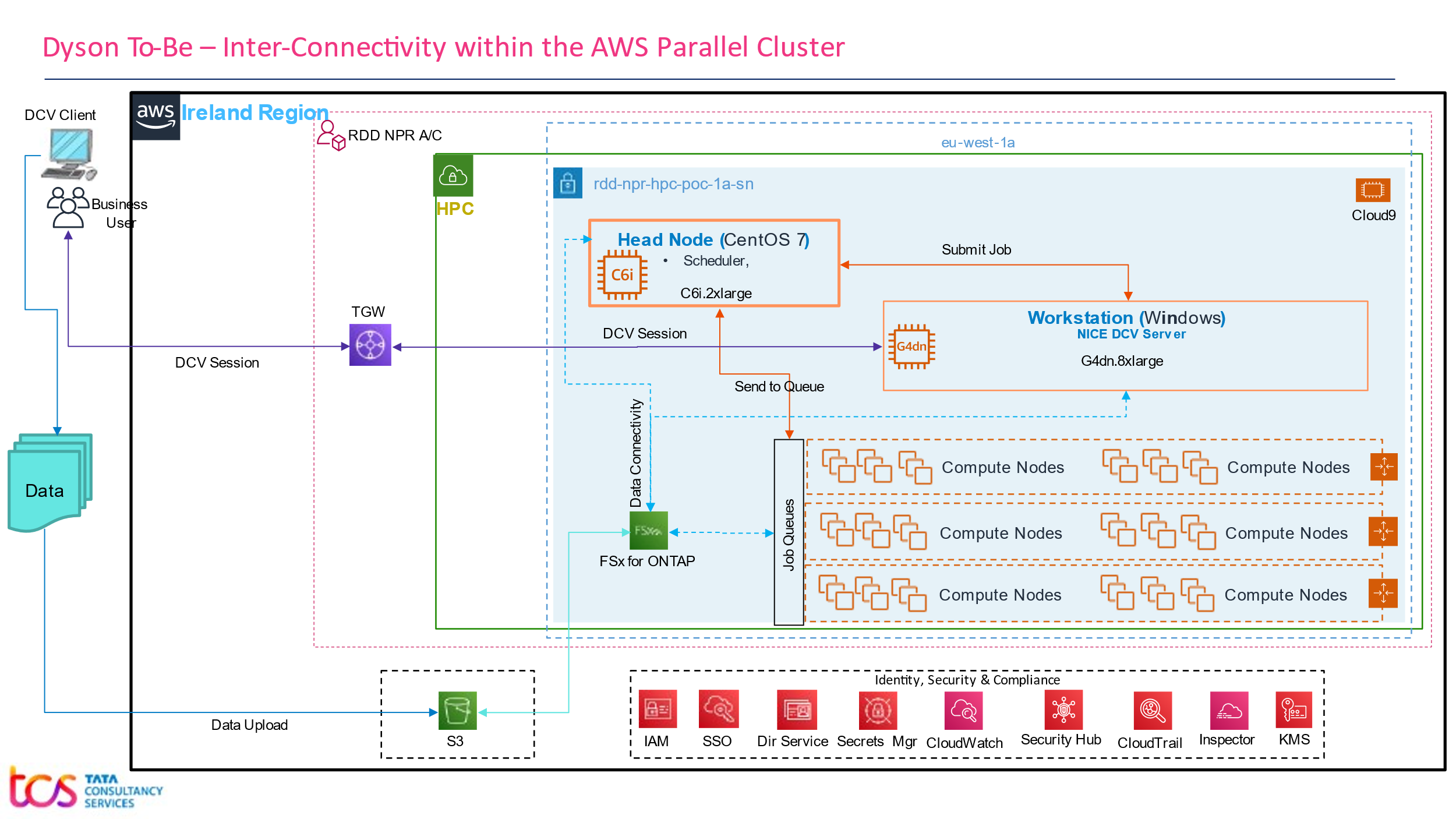


Figure 8 – Inter-Connectivity within the Cluster

* Workstations will have NICE DCV Server installed (note: free on AWS EC2).
* User may use the DCV Client and establish the remote session to their respective workstation.
* Windows based workstations will have required applications installed.
* Users will upload their data in a S3 bucket **euw1-rdd-npr-hpc-poc,** and from there it can be transferred to the shared storage (FSx NetApp Ontap). The shared storage will be accessible throughout the cluster.
* App support team will fetch applications from the app repository and upload them to S3 and from S3 to workstation and head node.
* Users will be able to access the cluster from the workstations.
* Scheduler will schedule the submitted job to the job queue based on scheduling policy.
* Compute Nodes will be provisioned on-demand and will be scaled up and down as per the scaling policy.

# Security

## AWS general approach for security

Proposed architecture will be based on AWS Security best practices. The entire AWS environment (Landing Zone) is already setup with Governance, Compliance and Security. SSO and MFA are also in place to ensure full centralized control over Access Management.

Actions taken by a user, on any AWS services are recorded as events in CloudTrail. Events include actions taken in the AWS Management Console, AWS Command Line Interface, and AWS SDKs and APIs.

Centralized log management for efficient monitoring and management functionality using CloudWatch, CloudTrail, VPC Flow Logs.

## AWS Console and Service Access Management

AWS SSO is only for Console Users (Developers, Administrator).

AWS SSO is being used to give permission/monitor access rights to various users. Some of the SSO related implementation details are given below

* Console access is centrally managed through AWS SSO and configured in the master/root account.
* Once the SSO user authenticated against the AD and MFA, the respective accounts will be listed to the user based on their privileges.
* Access to individual account is based on role policy defined and created in the respective AWS accounts.
* Third party soft token MFA (Google authenticator, Azure authenticator etc) can be used as an MFA device.
* Least privilege model is followed to grant necessary access to the accounts for individual console/CLI users.
* Service control policies can be configured to audit/restrict higher access privileges

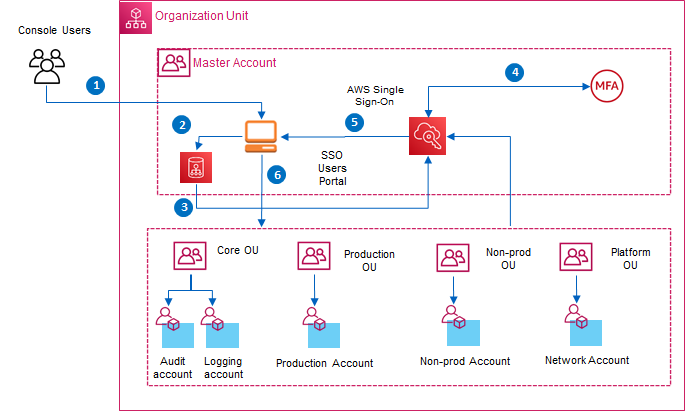


Figure 9 – AWS SSO Authentication Flow

**Authentication Flow steps:**

1. Console login user access the AWS SSO user’s portal
2. Users will be redirected to the Identity Provider if not already logged in.
3. Identity Provider response created and sent to AWS SSO
4. Second level authentication against MFA token
5. Based on the existing policies defined in the AWS accounts and attached to the SSO user, available AWS accounts will be listed to user
6. Select the account and will be redirected to the respective management console.

## Access Management using Azure AD

Dyson’s existing Active Directory i.e. Azure AD will be leveraged for authentication, AD is already integrated with AWS SSO as per the figure given below. Domain join is required to get the connectivity with the on premise license server, PLM server, and NFS.

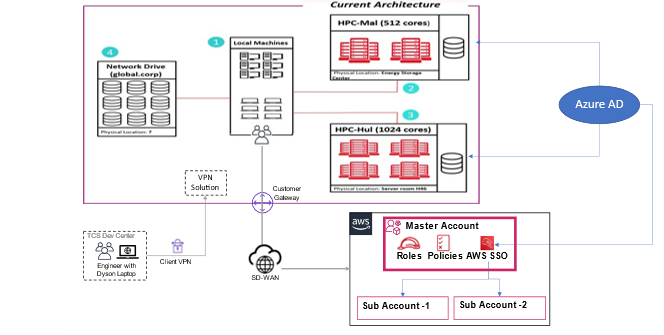


Figure 10 – AWS SSO and Microsoft AD Authentication

## Patching and Antivirus

As per vulnerability assessment report patching will be done.

## Vulnerability Check

Qualys is a cloud-based solution that detects vulnerabilities on all networked assets, including servers and instances. MS Defender or Qualys will be installed to check the vulnerability.

## Network Security

Below are the key components to manage Network Security.

1. All internet traffic will pass through existing firewall solution.
2. Access will be restricted in Firewall to blacklist all other sources except the traffic coming from the Dyson Corporate Network.
3. All EC2 Instances will be protected by **Security Group** (Stateful Access Control list). Only required port will be opened.
4. **Network Access Control List** are being used to filter traffic at Subnet level in the Transit A/C.
5. Head Node, Workstations and Compute Nodes will be kept in private Subnets.

## Governing AWS Environment

Governance for overall AWS environment will be taken care by **Guardrail**. Through guardrails, AWS Control Tower implements *preventive* or *detective* controls that help organization to govern resources and monitor compliance across groups of AWS accounts.

The existing **Guardrail** will be applied.

## Data Encryption

**Data in Rest:**

* 256-bit Advanced Encryption Standard (AES-256) will be used to encrypt your data in rest. EBS , S3 and all backup data will also be in encrypted form.
* All Amazon FSx file systems are encrypted at rest with keys managed using AWS Key Management Service (AWS KMS).

**Data in Transit:**

* Transfer of sensitive data or migration of data from Dyson to AWS will happen only through SDWAN over internet.
* Amazon FSx automatically encrypts data in transit using SMB encryption as you access your file system

**Key Management:**

* Key management will be done through AWS Key Management Service (AWS KMS). Key can be managed either by AWS or Dyson. During the PoC, let’s use AWS managed key.

## Logging and Monitoring

Existing Logging and Monitoring strategies will be used as per the proposed architecture.

**CloudWatch:**

* Application and infrastructure monitoring from CloudWatch console.
* Required logs will be collected and metrics will be created for monitoring and alerting.
* Logs will be stored in CloudWatch for 7 days during PoC phase
* Based on metrics threshold setting/system state changes, alarms will be triggered into Support team during managed support.

**CloudTrail:**

Logs user and API activities into S3 for Governance and are already integrated with Sentinel for compliance and operational auditing.

**VPC Flow Logs:**

VPC **Flow Logs is a feature** that enables you to capture information about the IP traffic going to and from network interfaces in a VPC. Flow logs can help to determining the direction of the traffic to and from the network interfaces

# AWS Infrastructure provisioning and Management

TCS will leverage Infrastructure as a code in the way of using Terraform, Cloud formation templates to drive the entire PoC phase using a Cloud9 instance. However, TCS will leverage the fully automated way in the form of Terraform and CloudFormation Stack to provision the infrastructure.

TCS will adopt few additional components which will provide more feature rich cluster in later phase of PoC which may include Engine Frame etc.

## EC2SelfService Portal

TCS has analysed the user’s requirement in more detail and created a custom GUI through which the user should be able to spin new workstation according to their need and can manage their workstation as well

* The solution comprises of –
  1. Html based UI hosted in IIS under EC2 instance
  2. API Gateway
  3. AWS Lambda
* The user is prompted the login page powered by Windows Authentication
* After Authentication, a list of Workstations is being shown to the user
* User can create new Workstations according to their need
* User should receive email notification with the Workstation detail
* The workstation will be domain joined manually

However, full functionalities as given below shall be provided in Post-PoC environment.

* The GUI can be deployed in AWS Amply to get the full advantage of AWS serverless computing
* The Single sign-on can be implemented for user authentication
* The portal should be restricted to specific AD groups
* Secret mgr can be used to store the service account (having lambda execution role) credential
* GUI should read the service account credential from the Secret Manager
* GUI should be able call lambda using the above credential
* No need of API Gateway layer at all
* The admin should be able to spin new workstation and can manage all the workstations
* The normal users should be able to manage the workstations those are assigned to them only
* The workstations should be tagged with user email address
* Managed operations are – DCV Session Connect, Start, Stop, Hibernate (if supported)

EngineFrame, AWS Workspace, AWS AppStream may be explored later as per the user’s need.

### Architecture

* PoC Design

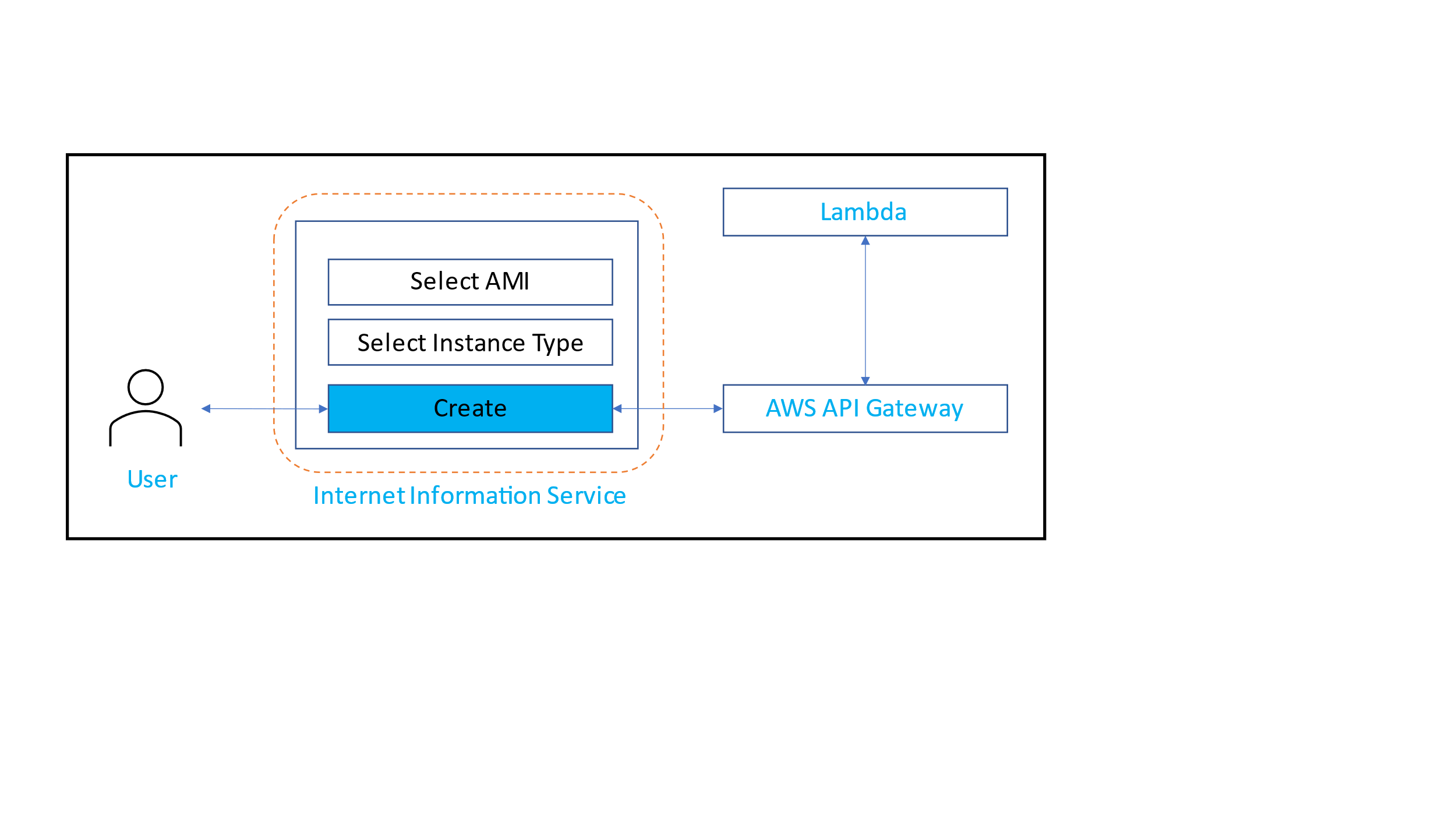


Figure 11 – EC2Self Service Portal Workflow

* The user should be prompted the login page and then EC2SelfService Portal to create new Workstations or to list down existing Workstations
* The UI should pass the required information to the Lambda through API Gateway
* The Lambda will process the request and response back to the GUI through API gateway
* Post-PoC Design

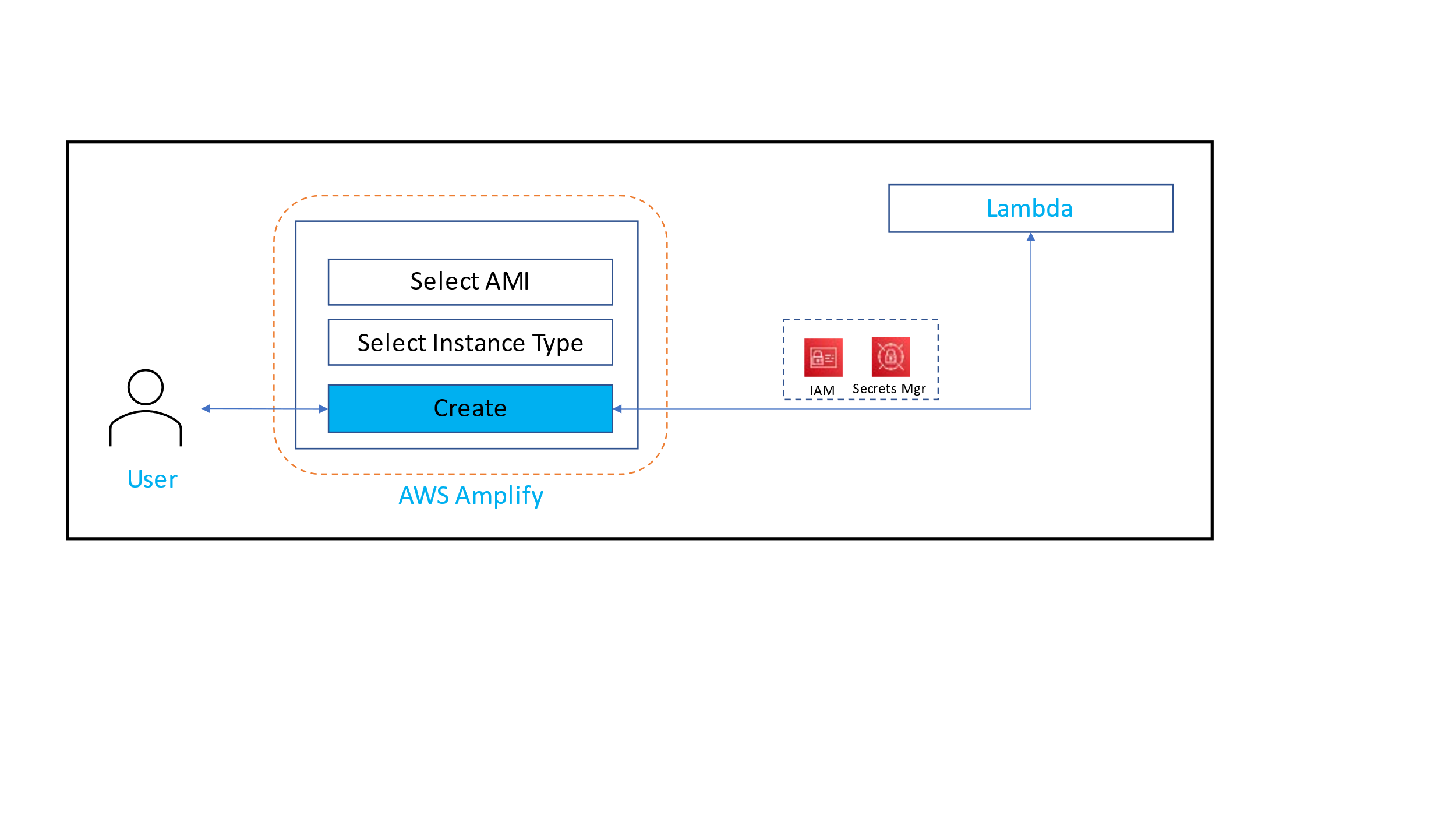


Figure 12 – EC2Self Service Portal Workflow

* The user should be prompted the SSO login page
* The UI should pass the required information to the Lambda
* The Lambda will process the request and response back to the GUI

### Workflow

Figure 13 – EC2Self Service Portal Workflow

* In PoC: There are couple of manual steps involved within this process such as domain join the workstations and installing security agents etc
* In Post-PoC: TCS will reconsider some of the manual steps towards automation, such as automatic domain join using script etc

# Storage Considerations

TCS will use the following options for storage in AWS

**Elastic Block Storage (EBS) for EC2 instances:**

Amazon Elastic Block Store (Amazon EBS) provides block level storage volumes for use with EC2 instances. EBS volumes behave like raw, unformatted block devices. You can mount these volumes as devices on your instances.

**FSx NetApp ONTAP:**

Amazon FSx for NetApp ONTAP is a fully managed service that provides highly reliable, scalable, high-performing, and feature-rich file storage built on NetApp’s popular ONTAP file system.  With FSx for ONTAP:

* Support for petabyte-scale datasets in a single namespace
* Multiple gigabytes per second of throughput per file system
* Security: AWS KMS, SMB Kerberos session keys, On-demand antivirus scanning
* ISO, PCI DSS, SOC, HIPAA compliant

**Amazon S3 Buckets:**

In proposed architecture Amazon S3 will be used to

* Store the data files temporarily. The user will upload the datafiles required for PoC from Dyson to S3 bucket, and then from S3 bucket to AWS HPC cluster.
* Store different Snapshots (i.e., EC2) for backup purpose.
* Store different logs and API activities.

## Storage Sizing and Recommendations

Actual sizes are given in the appendix-1 as part of the System requirements for PoC.

Following are few recommendations for this PoC

* + 256-bit Advanced Encryption Standard (AES-256) will be used, AWS-KMS will also be used.
  + Primary location of data will be AWS **Ireland** region

# Availability, Resilience and Capacity Management

## Availability and Resilience

The AWS global infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected with low-latency, high-throughput, and highly redundant networking.

According to proposed architecture:

* For simplicity in the PoC – the Head Node and the Workstations will be placed in a single AZ
* Compute Instances will be distributed in multiple Availability Zone. Multi-AZ instances available with a Monthly Uptime Percentage of at least 99.99%. Compute instances will be auto scaled across the Azs.
* The S3 Standard storage class is designed for 99.99% availability. Apart from that Cross regional replication will be enabled.

## Capacity Management

Capacity resizing will be done based on the performance test output and recommendation.

**AWS Trusted Advisor** inspects our AWS environment, and then makes recommendations when opportunities exist to save money, improve system availability and performance, or help close security gaps.

# Regulatory Compliance

All the existing regulatory compliances of Dyson will be adhered.

# Implementation

## Approach

* Below is the overall implementation approach for PoC phase. Using the Terraform script, the baseline infrastructure will be created along with a Cloud9 instance. Then from that Cloud9, the entire cluster will be created as per the figure given below.

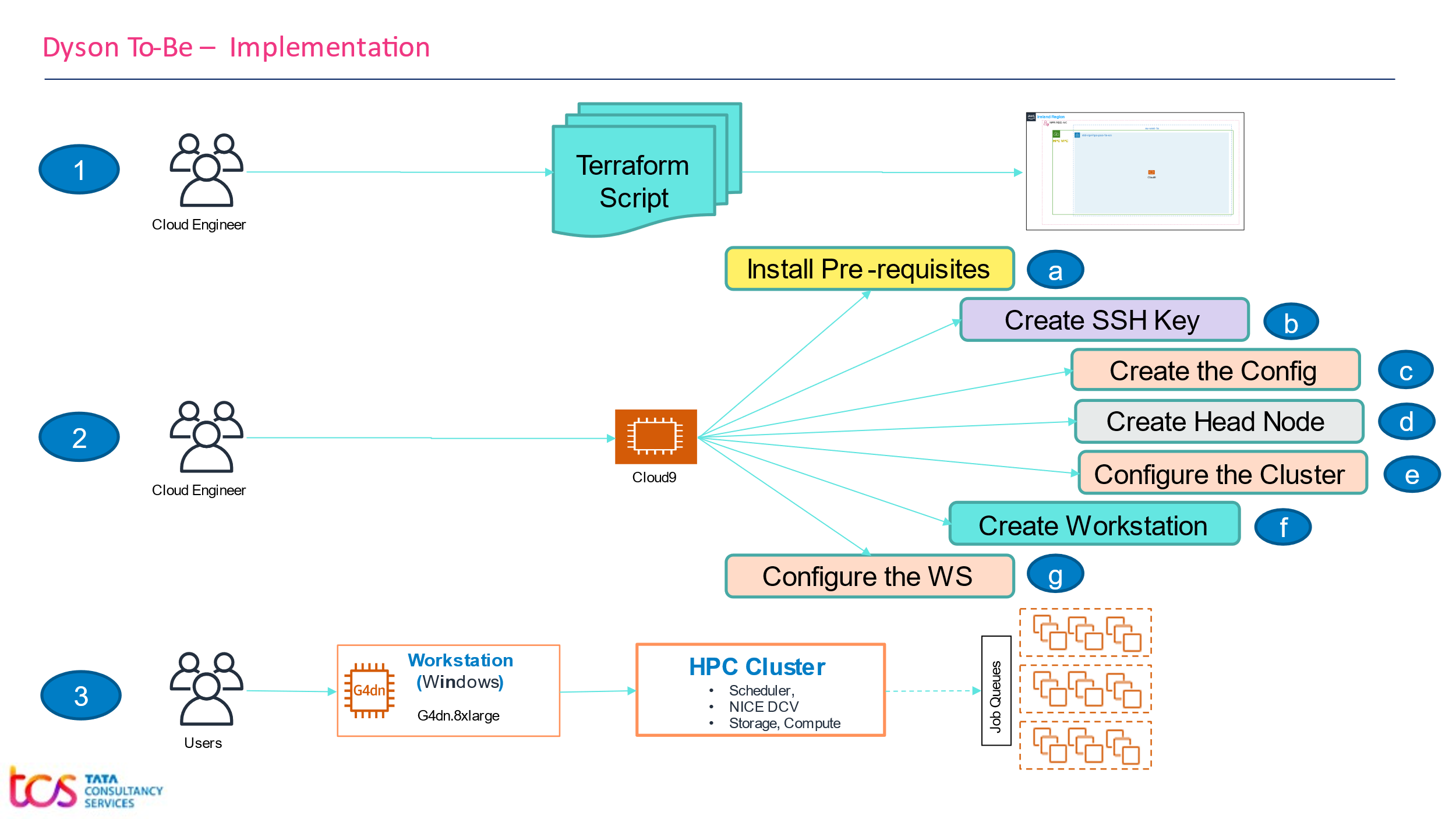


Figure 14 – Implementation plan for PoC phase

* **However, the Parallel Cluster will be created/provisioned using a fully automated IaC script in post PoC phase. However, linux/bash script can be used where ever Terraform module does not exist. However, Console/Manual steps can be performed where ever linux/bash script does work.**
  + Create the **SSH** key file using Terraform script.
  + Create the cluster config file and store it in **euw1-rdd-npr-hpc-poc** bucket in **yaml** format.
  + Create the **baseline infrastructure** such as VPC, Private Subnets and a Cloud9 using Terraform script. Please note that – we can remove the Cloud9 after the cluster is created.
  + Attach a start-up script with the Cloud9 instance. The script will download the cluster configuration file from **euw1-rdd-npr-hpc-poc** S3 location, then installs the cluster.
  + Log on to the **Head Node**, and configure the cluster including patch upgrade, anti-virus, Windows Defender or Qualys agent, Domain join.
  + Create a **Workstation** from an AWS Machine image using the Terraform script.
  + Log on to the Workstation and **configure, install required apps, Windows Defender or Qualys agent** etc.
  + Create a **golden image** and store in **euw1-rdd-npr-hpc-poc** S3 using Terraform script
  + Create Other **workstations from the golden image**, implement domain join to be a part of Dyson Corporate N/W.
  + Go to the cluster and configure the cluster according to our need.

## Implementation Details

Below are the key steps to implement the Parallel Cluster in AWS:

1. TCS Cloud Engineer will use their Dyson laptop to create the Terraform script to deploy the base infrastructure, VPC, subnets, and the Cloud9 as per the below figure inside the NPR RDD account.



Figure 15 – AWS base infrastructure

1. Here is the code snippet of the Terraform Script

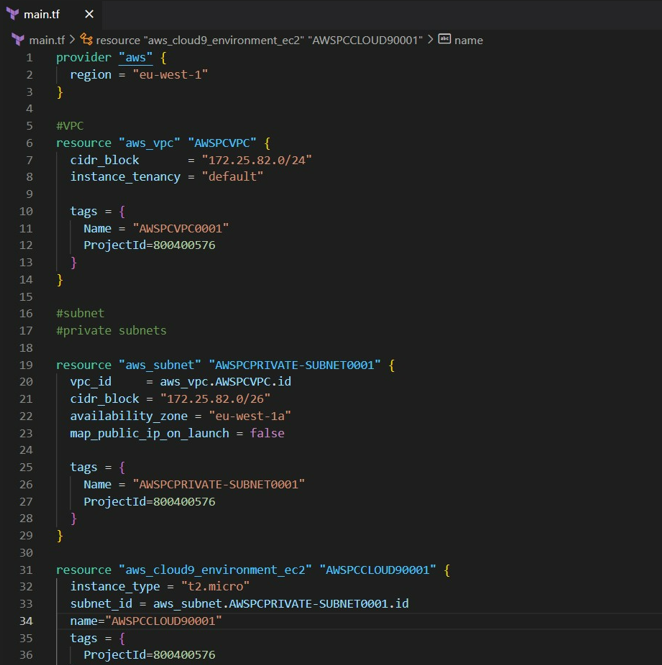


Figure 16 – Terraform Code Snippet

1. TCS Cloud Engineer will then log into the Cloud9 and will install the pre-requisites to ensure the latest version of AWS-CLI.
2. TCS Cloud Engineer will create the configuration file, here is the code snippet for the configuration file.
3. Store the config file which is ***config.yaml*** in Cloud9
4. Create a SSH key-pair to create the cluster, name it as **Dyson-HPC-PoC-Key.pem**



Figure 17 – Snippet of Configuration File

1. Create the Cluster by the command given below

pcluster create-cluster –cluster-name <Cluster Name> --cluster-configuration *config.yaml*

1. Head Node will be created along with all the required components such as DCV Server, Job Scheduler etc.
2. Configure the Head Node – Patch upgrade, Anti-Virus, Windows Defender or Qualys agent. Install the required s/w to execute the use cases.
3. Create the Workstations from Amazon Machine Images using Terraform script, the image will provide certain s/w preinstalled.
4. Configure the Workstation – Patch upgrade, Anti-Virus, Widows Defender or Qualys agent, install required s/w to execute the use cases.
5. Domain join FSx and Windows workstations.
6. Integrate the Parallel Cluster with Active Directory.
7. Configure the FSx netapp to be accessible from Windows workstation and Parallel Cluster as well.
8. Upload data files from on-prem to shared storage – FSx, S3 bucket

## App Installation

* The existing app or s/w could be taken from Dyson repository, rest can be downloaded from the vendor’s website
* Applications downloaded through internet need to go through an antivirus check.
* Once the installer files are ready, app support team will upload it to the S3 bucket (app team would have access to S3 buckets).
* To login to the workstation, the app team would be using DCV Client or the connection URL, through which they would be able to access the DCV Server installed on the workstation.
* Once they are logged into the workstation, team will download the installer files from S3 bucket and install it.
* Windows Defender or Qualys will be installed for vulnerability check
* AMI will be created out of the workstations to create other workstations to execute the user cases.
* For app installation over the HPC – the users will be given the local credential of the Head Node. They can simply get into the VMs using Putty.
* Using those installers, installation of application would be done.

## List Of Services

|  |  |  |  |
| --- | --- | --- | --- |
| **Resource List of RDD-NPR in** **EU-West-1** | | | |
| **Service Type** | **CIDR** | **Name Tag** | **Name** |
| **VPC** | 10.221.90.0/24 | 2001493-HPC Upstream Compute | aws-rdd-npr-hpc-poc-vpc-euw1 |
| **Subnet** | 10.221.90.0/26 | 2001493-HPC Upstream Compute | rdd-npr-hpc-poc-1a-sn |
| **Route Table** |  | 2001493-HPC Upstream Compute | rdd-npr-hpc-poc-rt |
| **NACL** |  | 2001493-HPC Upstream Compute | acl-05cf88990acf77d71 |
| **S3 bucket** |  | 2001493-HPC Upstream Compute | euw1-rdd-npr-hpc-poc |
| **FSx For OnTAP** |  | 2001493-HPC Upstream Compute | rdd-npr-hpc-poc-fsx-netapp-ontap |
| **Key-Pair** |  | 2001493-HPC Upstream Compute | rdd-npr-hpc-poc.eu-west-1 |
| **Secret Manager** |  | 2001493-HPC Upstream Compute | rdd-hpc-poc-domain-credential |

## Subnet to Route Table Mapping

|  |  |
| --- | --- |
| **VPC: aws-rdd-npr-hpc-poc-vpc-euw1 in NPR-RDD account in EU-West-1** | |
| **Availability Zone – eu-west-1a** | |
| **Subnet Name** | **Route Table** |
| rdd-npr-hpc-poc-1a-sn | rdd-npr-hpc-poc-rt |

## Route Table Details

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Subnet Type** | **Subnet Name** | **Route Table** | **Target Network** | **Next Hop Type** |
| Private Subnet | rdd-npr-hpc-poc-1a-sn | rdd-npr-hpc-poc-rt | 0.0.0.0/0 | tgw-060b743a2e75f3c12 |
| 10.221.90.0/24 | local |

## Subnet to NACL Mapping

|  |  |
| --- | --- |
| **Availability Zone – eu-west-1a** | |
| **Subnet Name** | **NACL** |
| rdd-npr-hpc-poc-1a-sn | acl-05cf88990acf77d71 |

## NACL Detail

|  |  |  |  |
| --- | --- | --- | --- |
| **NACL** | **Allowed Inbound CIDR** | **Allowed Outbound CIDR** | **Description** |
| acl-05cf88990acf77d71 | All traffic 0.0.0.0/0 | All traffic 0.0.0.0/0 |  |

## Instance Details

|  |  |  |  |
| --- | --- | --- | --- |
| **VPC: aws-rdd-npr-hpc-poc-vpc-euw1 in NPR-RDD account in EU-West-1** | | | |
| **Subnet** | **EC2 Instance** | **Node Name** | **Operating System** |
| rdd-npr-hpc-poc-1a-sn | Cloud9 | AWSPCVMCloud9 | Amazon-Linux |
| Head Node | AWSPCVMHN0001 | CentOS 7 |
| Workstation-1 | AWSPCVMWN0001 | Windows Server 2022 |
| Workstation-2 | AWSPCVMWN0002 | Windows Server 2022 |
| Workstation-3 | AWSPCVMWN0003 | Windows Server 2022 |
| Workstation-4 | AWSPCVMWN0004 | Windows Server 2022 |
| Compute Node | AWSPCVMCN0001 | CentOS 7 |

## Instance to Security Group Mapping

|  |  |  |
| --- | --- | --- |
| **Instance to Security Group Mapping** | | |
| **Instance Name Tag** | **Security Group Name Tag** | **Description** |
| AWSPCVMCloud9 | sg-hpc-poc-cloud9 |  |
| AWSPCVMHN0001 | sg-hpc-poc-head-node |  |
| AWSPCVMWN0001 | sg-hpc-poc-workstation-node |  |
| AWSPCVMWN0002 | sg-hpc-poc-workstation-node |  |
| AWSPCVMWN0003 | sg-hpc-poc-workstation-node |  |
| AWSPCVMWN0004 | sg-hpc-poc-workstation-node |  |
| AWSPCVMCN0001 | sg-hpc-poc-compute-node |  |

## Security Group Detail

|  |  |  |  |
| --- | --- | --- | --- |
| **Security Group** | **Allowed Inbound Port** | **Allowed Outbound Port** | **Description** |
| sg-hpc-poc-cloud9 | All traffic from Dyson n/w, All Port | All traffic to internet | Consider this as an exception during PoC phase, later we will be able to identify the exact security and will be able to consider it for the production |
| sg-hpc-poc-head-node | All traffic from Dyson n/w, All Port | All traffic to internet |
| sg-hpc-poc-workstation-node | All traffic from Dyson n/w, All Port | All traffic to internet |
| sg-hpc-poc-compute-node | All traffic from Dyson n/w, All Port | All traffic from Dyson n/w, All Port |

# Appendix

**Appendix 1:** System requirements for PoC

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **System requirements for PoC** | | | | | | | |
| **Sl. No** | **Description** | **Environment** | **Instance Type** | **vCPU** | **RAM (GB)** | **Storage** | **Size (GB)** |
| 1 | Head Node | NPR RDD | C6i.2xlarge | 8 | 16 | EBS | 50 |
| 2 | Workstation-1 | NPR RDD | G4dn.16xlarge | 64 | 256 | EBS | 500 |
| 3 | Workstation-2 | NPR RDD | G4dn.8xlarge | 32 | 128 | EBS | 150 |
| 4 | Workstation-3 | NPR RDD | G4dn.8xlarge | 32 | 128 | EBS | 40 |
| 5 | Workstation-4 | NPR RDD | G4dn.8xlarge | 32 | 128 | EBS | 2000 |
| 6 | Compute Node | NPR RDD | C6i.32xlarge | 128 | 256 | EBS | 1200 |
| 7 | FSx for NetApp ONTAP | NPR RDD | Shared Storage | | | FSx | 20000 |

**Appendix 2:** Functionalities during PoC and Post PoC?

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl. No** | **Functionalities** | **PoC** | **Post PoC** |
| 1 | Deployment | Using Console, Cloud9, AWS-CLI | Fully automated IaC script |
| 2 | Workstation Selection | Manually created instances | User will be given the custom GUI to create and manage workstation |
| 3 | Patching/AV | In-Scope for Windows, evaluate the feasibility for CentOS | In-Scope |
| 4 | Domain Join | In-Scope | In-Scope |
| 5 | Multi-region, Data Replication | Not In-Scope | Can be Implemented based on the requirement |
| 6 | Backup | Not In-Scope | In-Scope |
| 7 | Disaster Recovery | Not In-Scope | Can be Implemented |
| 8 | Managed Support | Not In-Scope | In-Scope |
| 9 | Custom GUI to create Workstation | In-Scope with limited functionalities | In-Scope with full functionalities |
| Components | HTML+ Javarscript, API Gateway, Lambda, IIS | HTML+ Javarscript, Secrets Manager, IAM, Lambda, AWS Amplify |
| Authentication /Authorization | Windows authentication for the UI | SSO based authentication for UI. Service account and IAM based lambda execution. AWS Secrets Manager to store service account credential |
| User Persona | Not In-Scope | Active Directory based user personalization depending on which groups the user belongs to |
| Metadata list for image, size | Hardcoded at UI side | Will analyze further whether to store them in a database table or will hard code in another lambda |
| Deployment and upgrade | Deployed in IIS Using Console | Deployed in AWS Amplify, fully automated with Terraform script |
| Support and Maintenance | TCS AWS Team | TCS AWS Team |

**Appendix 3:** Data Transfer from on premise to AWS cloud using S3 bucket. This will be validated by Dyson security.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case** | **Type of data** | **Extension** | **Size/ Volume** | **Purpose** |
| 1 | Various file types of ANSYS Workbench Project | .cas, .dat, .msh, .wbpj, .jpg, and other | 2-4GB | To test ANSYS WB automated design of experiment. Fluent, Meshing, CFD-Post, Workbench. |
| 2 | Binary file format input to ANSYS FLUENT | \*.cas  \*.dat | 4.5 GB  5.4 GB | To run the ANSYS FLUENT simulation |
| 3 | Binary file format input to ANSYS FLUENT like 2. | \*.cas  \*.dat | 4.5 GB  5.4 GB | To run the ANSYS FLUENT simulation |
| 4 | Binary file format input to PowerFLOW sim | \*.case  \*.cdi | 1 GB  0.5 GB | To run PowerFLOW CFD simulation |