Banking API Infrastructure

Hosted on AWS using the following:

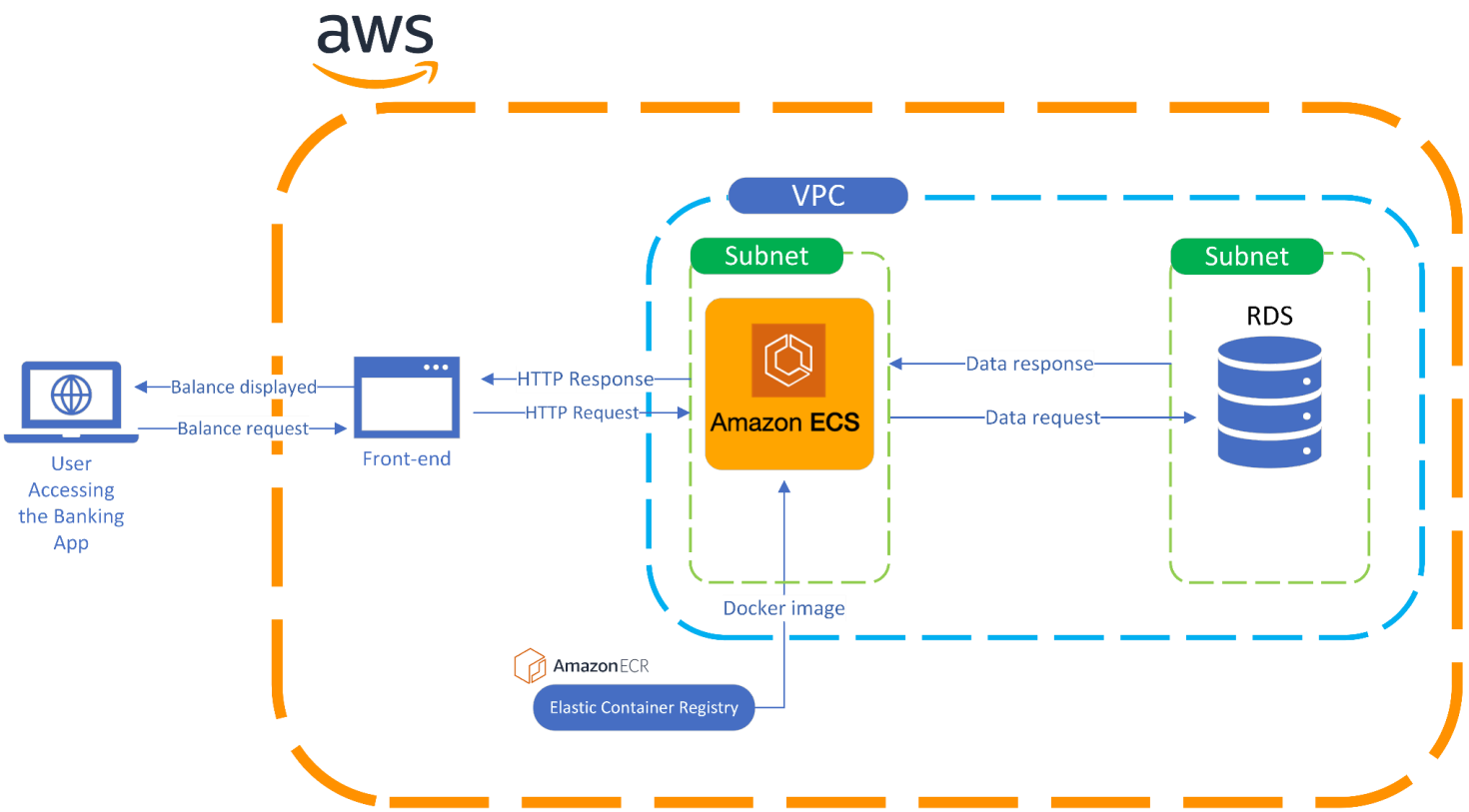
- ECS for the container instance
- RDS for the database
- Docker image in ECR
- Contained within VPC and subnet
- Wrapped in security groups to control traffic

ECS (Elastic Container Service) is an ideal choice for hosting the containerised application because it is a fully managed service which takes away the need to specify the underlying infrastructure (like you would need to with self-managed Kubernetes) and allows containerised apps to be hosted quickly and at scale. Like many cloud resources it also features pay-as-you-go pricing which takes away the cost of having the underlying infrastructure up and running 24/7.

RDS (Relational Database Service) gives a wide choice of database engines to choose from, so any flavour of SQL could be used to host the data for the API. AWS Aurora is the fully managed RDS service for MySQL and PostgreSQL which provides a highly available and secure option for hosting data in the cloud, as well as making set up simpler than a self-managed option.

As this is a banking API the data being handled is highly sensitive and would need to be extremely secure. To ensure this we would host both of these services within a Virtual Private Cloud (VPC) to ensure that we can control any traffic through user defined gateways. By containing the resources within subnets we can logically separate them, then use security groups attached to the resources to set specific rules on ingress/egress for those resources. This will provide a strict set of rules that will keep the data secure in both the database and the containers using it, and while the data is in transit.

The combined use of all of these resources comes with all the benefits of hosting in the cloud, including high availability and scalability, geographical redundancy for our valuable data, reduced costs compared to hosting on site and strong security measures at our disposal.


aws
VPC
Subnet
Subnet
RDS
Amazon ECS
Balance displayed
Balance request
HTTP Response
HTTP Request
Data response
Data request
User Accessing the Banking App
Front-end
Docker image
AmazonECR
Elastic Container Registry