**About the project:**

* **Multi-tier web application stack (VPROFILE)**
* **Host & run-on AWS cloud for production**
* **Lift & shift strategy**

SCENARIO:

Application services running on physical/virtual machines

Work load is in your datacentre.

**PROBLEM:**

* Complex management.
* Scale up/down complexity.
* Upfront capital expenditure & regular operations expenditure
* Manual process
* Difficult to automate
* Time consuming

**Solution**:

Cloud computing setup

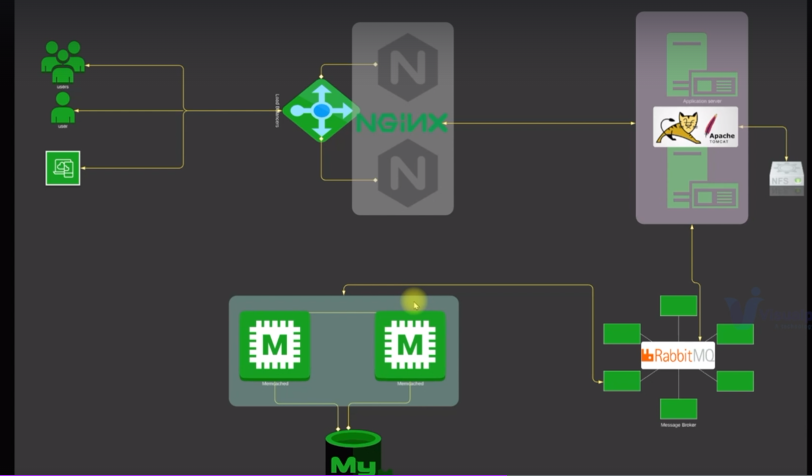
**AWS Services**

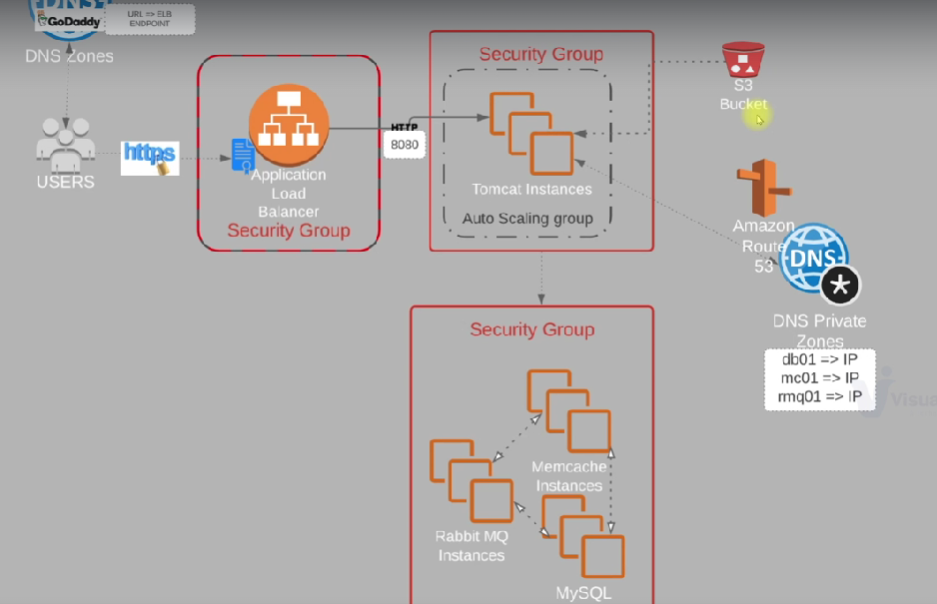
* EC2 instances: VM for TOMCAT, RabbitMQ, Memcached, MySQL
* ELB (Load Balancer)
* Autoscaling: Automating VM scaling
* S3/EFS Storage: Shared storage
* ROUTE 53: For private DNS Service
* IAM, ACM, EBS also used

**OBJECTIVE**:

* Flexible Infra
* No upfront cost
* Modernize effectively using AWS services
* IAAC

**Architecture of AWS services for the project**





Users will access our website by using a URL and that URL we be pointing to an end point; this entry will be mentioned in GoDaddy DNS.

User browsers or the app will use this end point then to connect to the load balancer, by using https Certificate for https encryption will be mentioned in ACM Amazon Certificate Manager Service. So, user will access application load balancer endpoint. Our load balancer will be in a security group and will only allow https traffic.

And then our application load balancer, will route the request to Tomcat instances. Apache tomcat service, will be running on some set of instances which will be managed by our auto scaling group. So as for high or low load, these instances capacity will be scaled out or scaled in.

These ec2 instances where Tomcat is running, will be in a separate security group and will only allow traffic on Port 8080 only from a load balancer.

And our application needs backend servers, which are MySQL, Memcached and RabbitMQ. Information of backend services or the backend server IP address will be mentioned in Route 53 private DNA zone.

So, Tomcat instances will access back server with a name which will be mentioned in Route 53 private dns where the private IP address of our background servers will be mentioned. These backend ec2 instances, which will be running MySQL, RabbitMQ, Memcached will be in a separate security group.

So, the AWS sources, which are in use over here are first Amazon certificate manager for a certificate application load balancer. Set of ec2 instances for Tomcat, Memcache, RabbitMQ and MySQL, three separate security groups. Amazon Route 53 for DNS Private Zones. And also, there's Amazon S3 bucket to store our software artifacts.

