

# COMO CLUB SOLUTION ARCHITECTURE PRESENTATION

---

Carmen Yip, Chong Jie Mi, Jolene Loh,  
Lindy Lim, Nikki Poo, Wong Jiong Yu

# TABLE OF CONTENTS

01

BUSINESS NEED &  
SOLUTION  
OVERVIEW

---

02

DEMO

---

03

MAINTAINABILITY

---

04

AVAILABILITY

---

05

SECURITY

---

06

PERFORMANCE  
QUALITIES

---

# BUSINESS NEED



Middleware to manage the integration of ComoClub's numerous, siloed and disparate Business Units

# STAKEHOLDERS



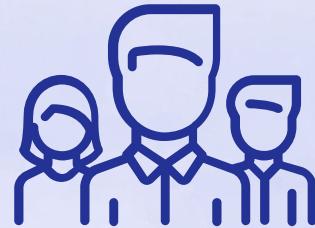
## GUESTS

- Become a member
- Browse Listings



## MEMBERS

- Book experiences
- Redeem items
- Earn points
- Pay with cards
- Change account details



## MERCHANTS

- Offering services/items

# STAKEHOLDER PERMISSIONS



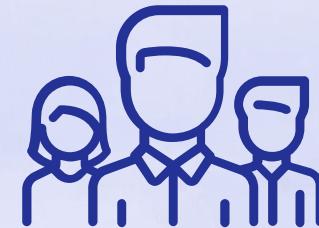
## GUESTS

- Create account
- View experiences and items



## MEMBERS

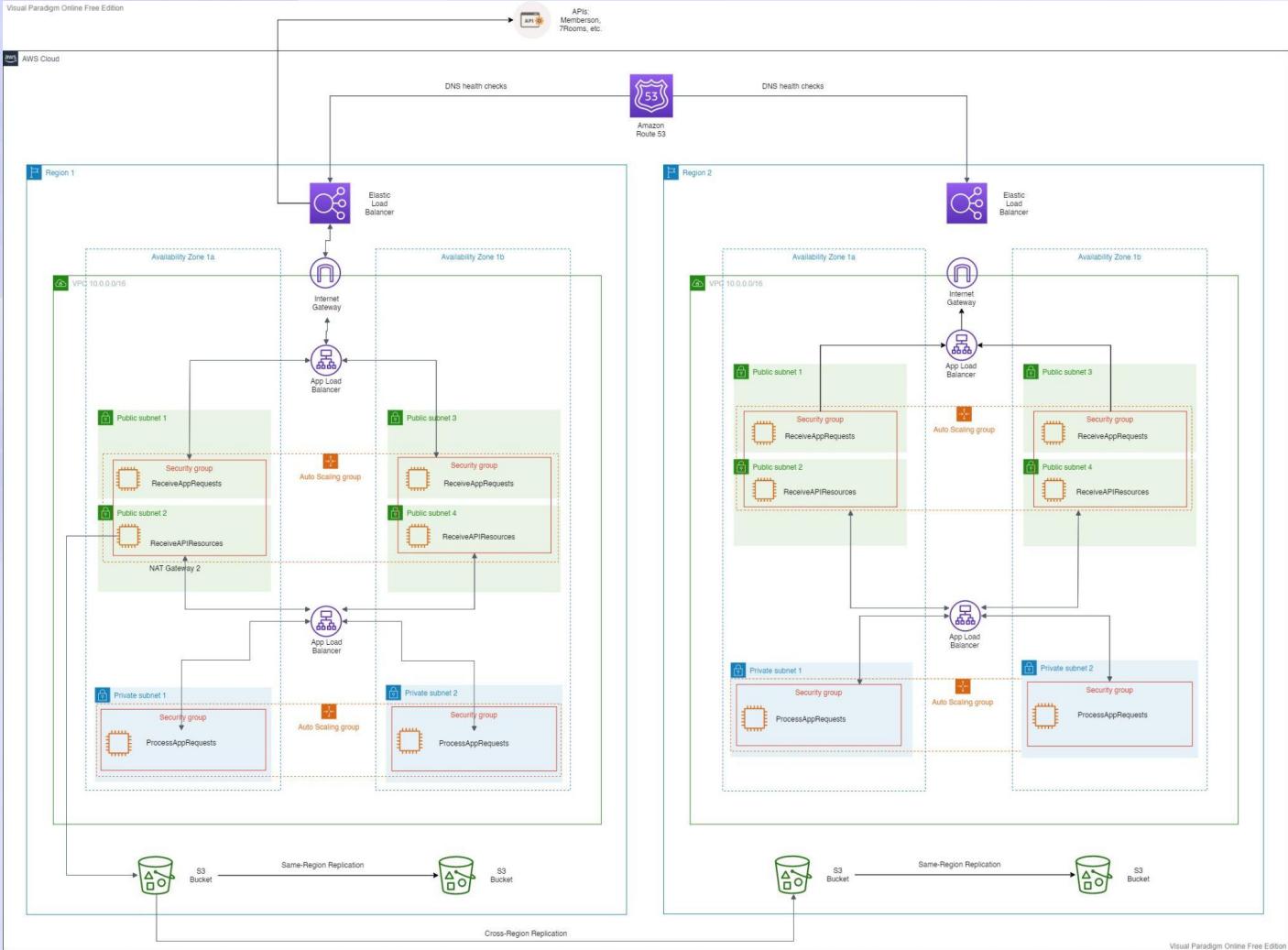
- Log in
- View experiences and items
- Create bookings
- Redeem items
- View wallet
- Update account details



## MERCHANTS

- Connect with COMOClub to create, update and delete experiences

# SOLUTION OVERVIEW



# OVERALL COSTS - DEVELOPMENT

SERVICE	DESCRIPTION	COST
<b>Working Hours</b>	Time spent implementing new software, modifications & integration and creating Middleware	6 weeks of labour x a team of 6
<b>AWS</b>	Services used to host and run the middleware application in a development environment	Approx \$400
<b>Designing Solution Architecture</b>	Designing, describing, and managing the solution engineering in relation to specific business problems	2hrs x 4 weeks x 6 students
<b>Developing Security</b>	Implementing SSL and TLS	1hr x 4 weeks x 6 students
<b>Developing CI</b>	Time spent creating, testing and iterating	2hrs x 4 weeks x 6 students
<b>Developing CD</b>	Time spent developing & deploying	3hrs x 4 weeks x 6 students

# OVERALL COSTS - PRODUCTION

SERVICE	DESCRIPTION	COST
Virtual Private Cloud (VPC)	<ul style="list-style-type: none"><li>- 8 NAT Gateway</li><li>- 30 GB Data Processing per NAT Gateway</li></ul>	\$273.60 USD per month
EC2 Instance Saving Plan	<ul style="list-style-type: none"><li>- 12 Instances</li><li>- Instance type</li><li>- 100% Utilization/month</li><li>- Storage: 30GB</li></ul>	\$24 USD per month
Elastic Load Balancing	<ul style="list-style-type: none"><li>- No. of Application Load Balancers: 4</li></ul>	\$65.70 USD per month
S3 Bucket Caching	<ul style="list-style-type: none"><li>- Amazon Simple Storage Service: 2 S3 buckets</li><li>- S3 Standard storage: 30GB</li></ul>	\$6.60 USD per month

# SOLUTION OVERVIEW DEMO

---

- Key Use Case Scenarios
- End-to-end features

# Key Use Cases



## Use Case 1

Experience Booking  
(7 Rooms)



## Use Case 2

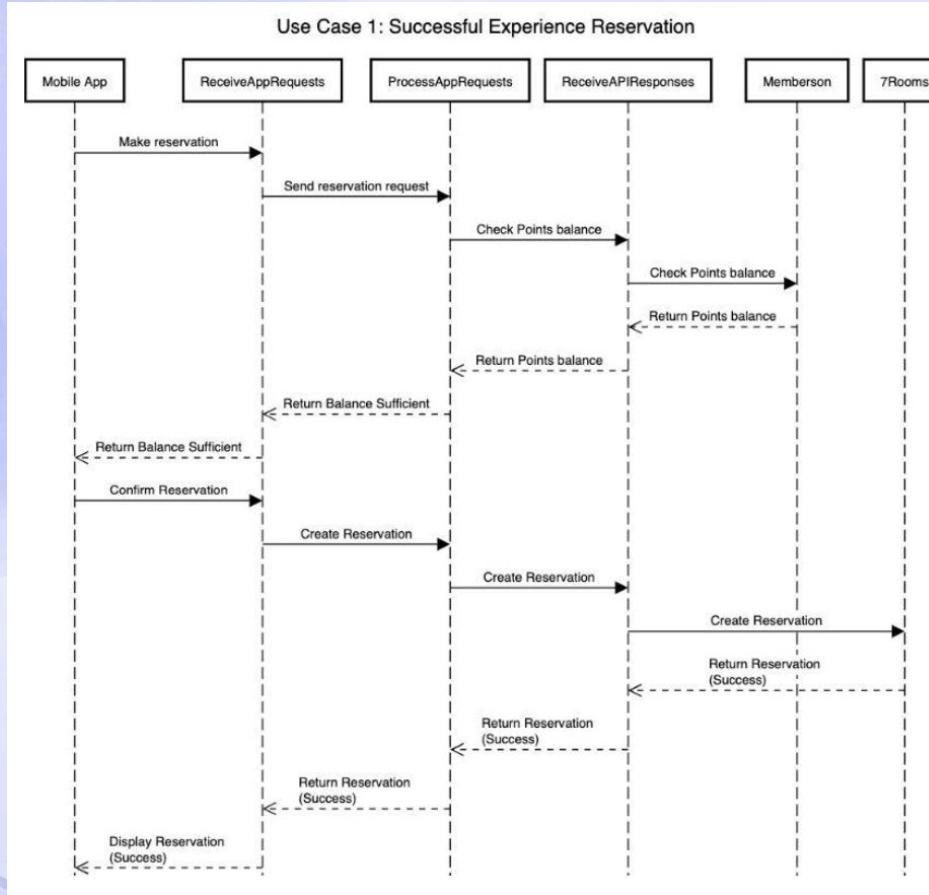
Item Redemption



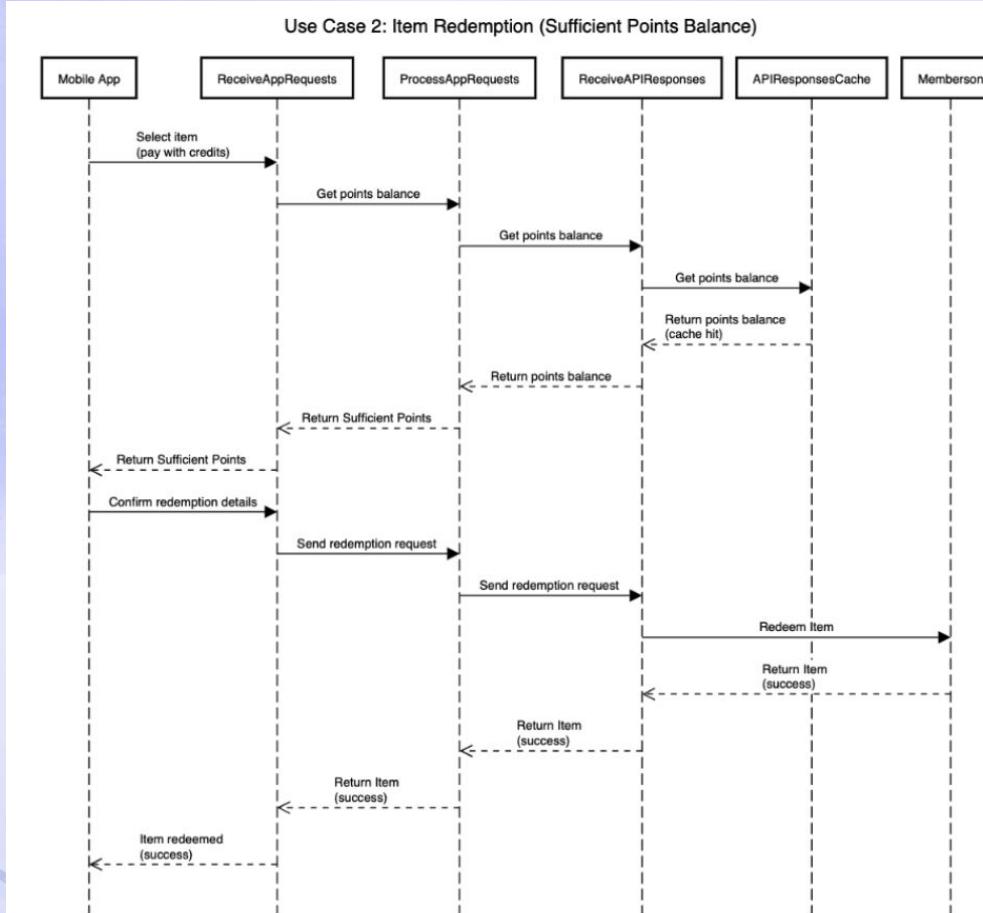
## Use Case 3

Creating a new  
membership account

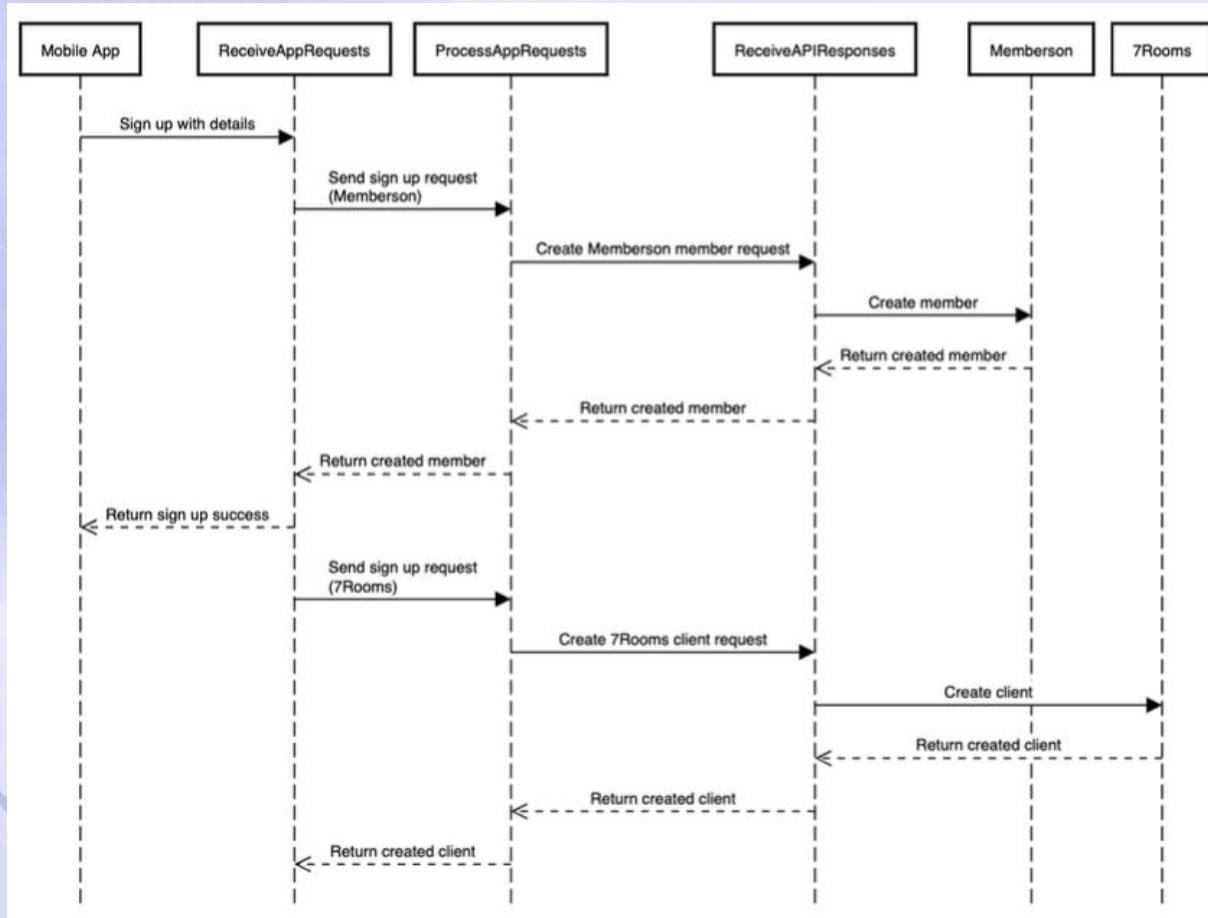
# Use Case 1: Experience Booking



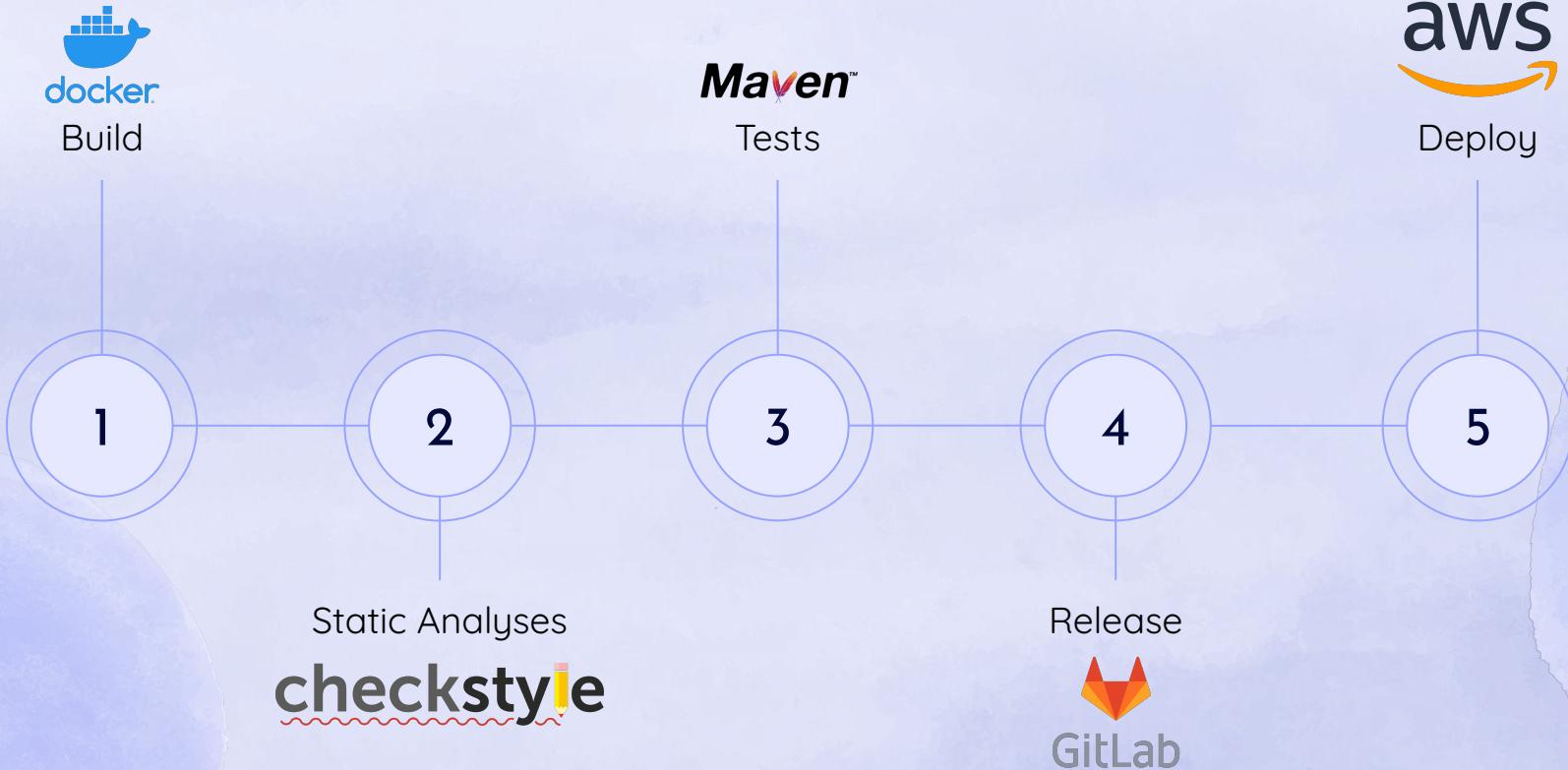
# Use Case 2: Item Redemption



# Use Case 3: Creating new member



# MAINTAINABILITY



# 02

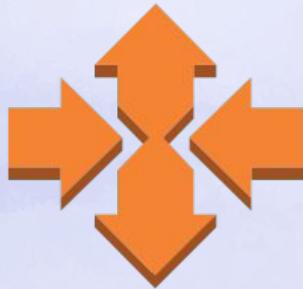
## MAINTAINABILITY DEMO

---

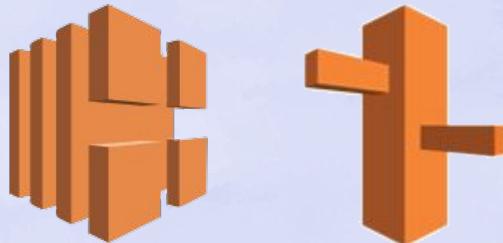
# AVAILABILITY



Route requests to different copies of instances created for redundancy  
(Load Balancing)



Scale up instances when there's more traffic  
(Auto Scaling)



Route requests over to different regions  
(Multi-region failover)

# 03 AVAILABILITY DEMO

---

# SECURITY

## 4 Key Vulnerabilities



### Password Security

Applied SHA-512 Hash Function



### DNS Spoofing / Cache Poisoning

Modified subnets' security groups to control traffic to our EC2 instances



### Layers of Attack Surface

Isolate Components  
Limit attack surface



### Man in the Middle Attack

- Digital Signature
- Transport Layer Security (TLS)

# SECURITY DEMO

---

04



## Password Security

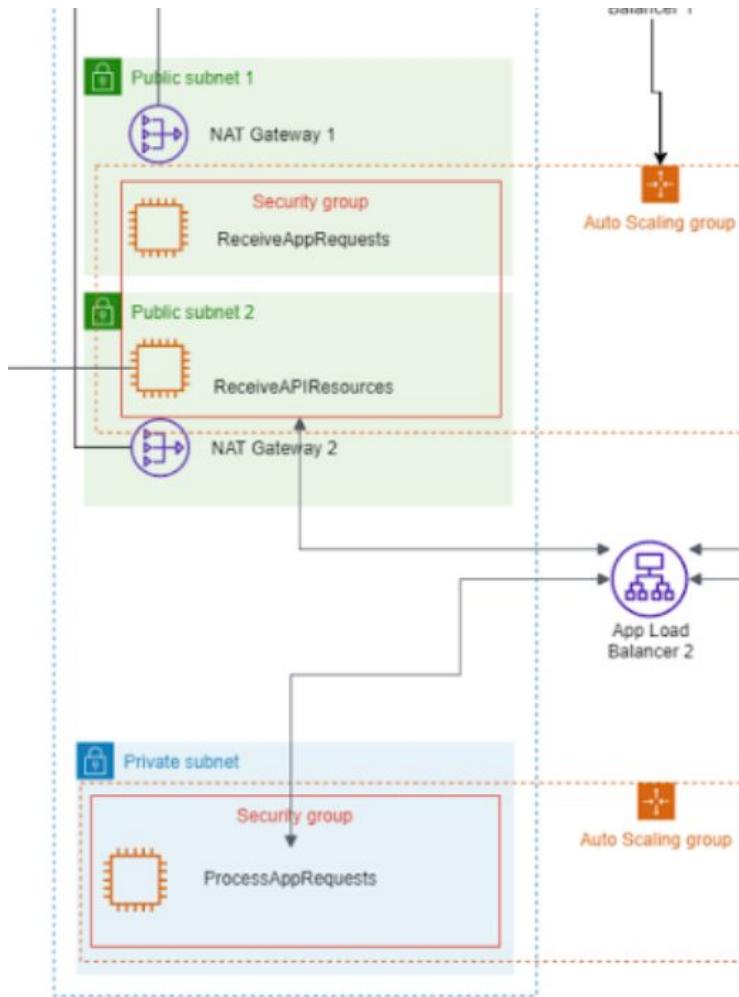
Applied SHA-512 Hash  
Function

```
<script type = "text/javascript">
    function hashpw(obj) {
        var pwdObj = document.getElementById('password');
        var hashObj = new jsSHA("SHA-512", "TEXT", {numRounds: 1});
        hashObj.update(pwdObj.value);
        var hash = hashObj.getHash("HEX");
        pwdObj.value = hash;
    }
</script>
```



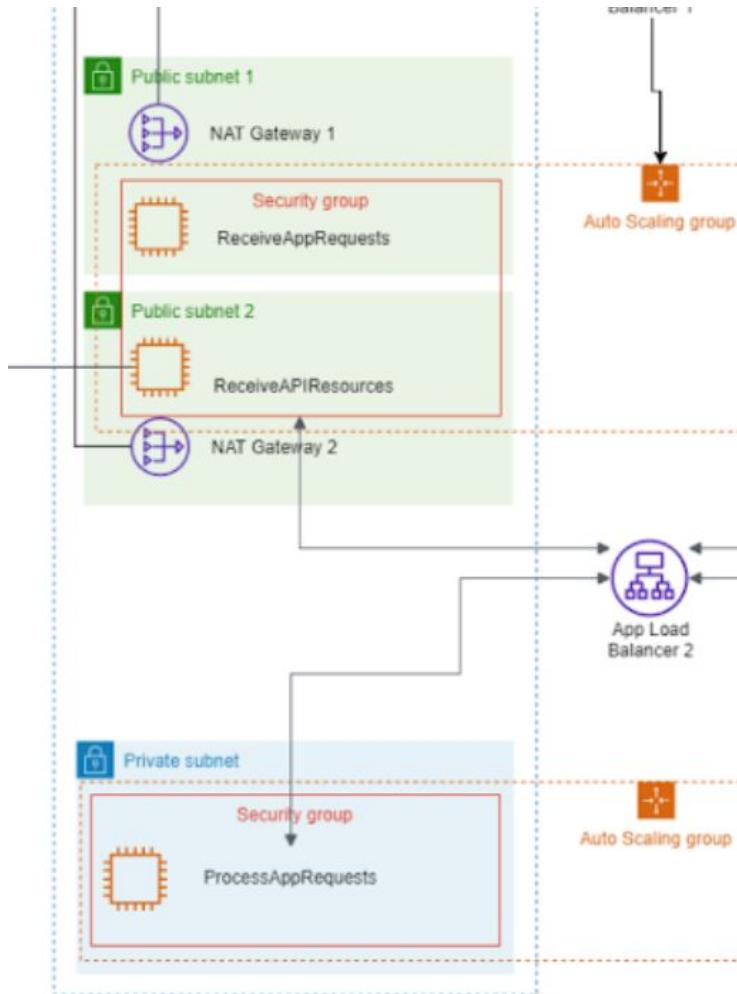
## DNS Spoofing / Cache Poisoning

Modified subnets' security groups to control traffic to our EC2 instances



## Layers of Attack Surface

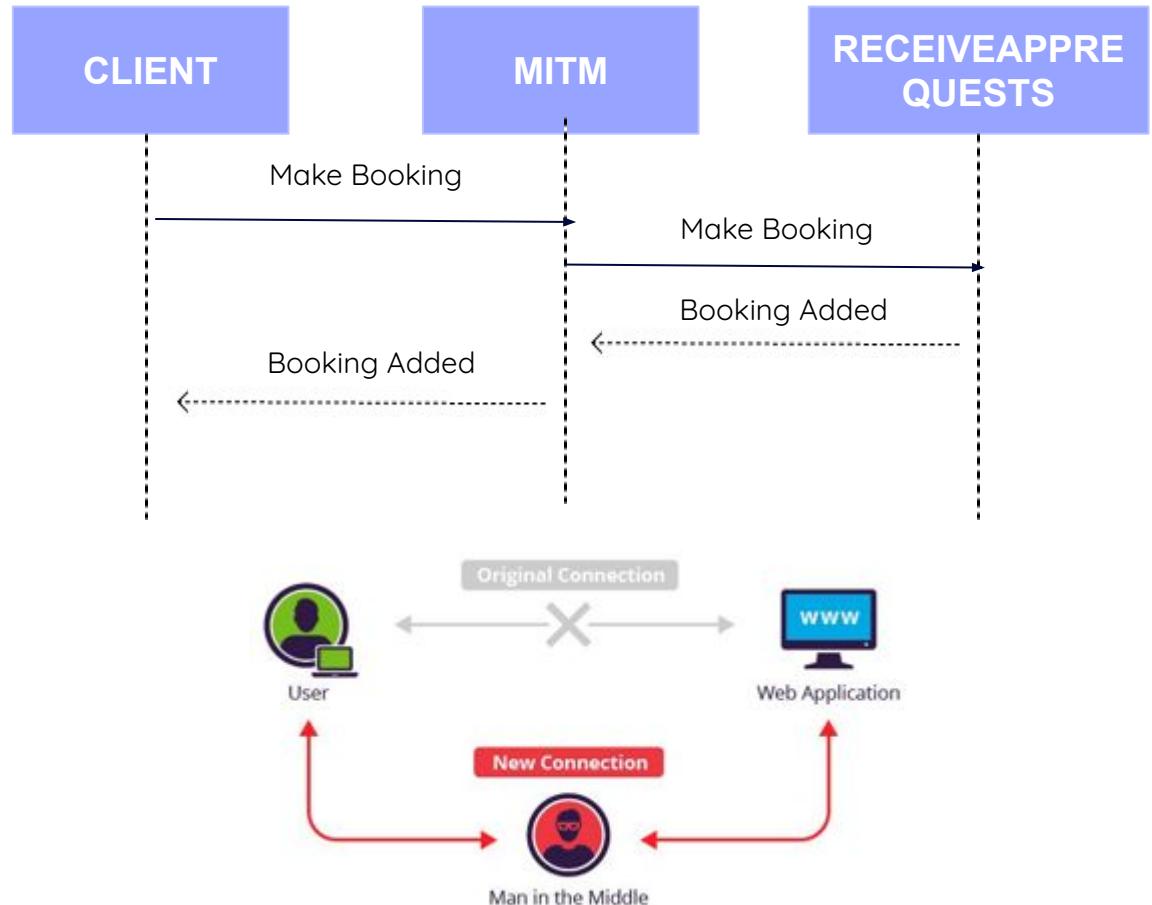
Isolate Components  
Limit attack surface



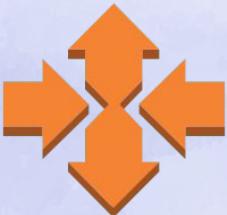
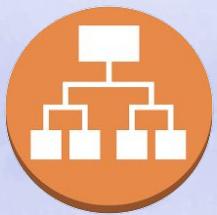


## Man in the Middle Attack

- Digital Signature
- Transport Layer Security (TLS)



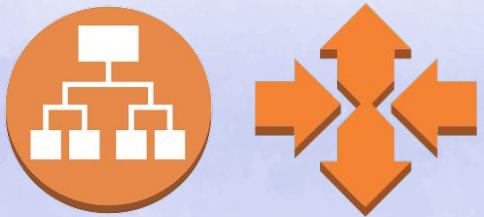
# PERFORMANCE



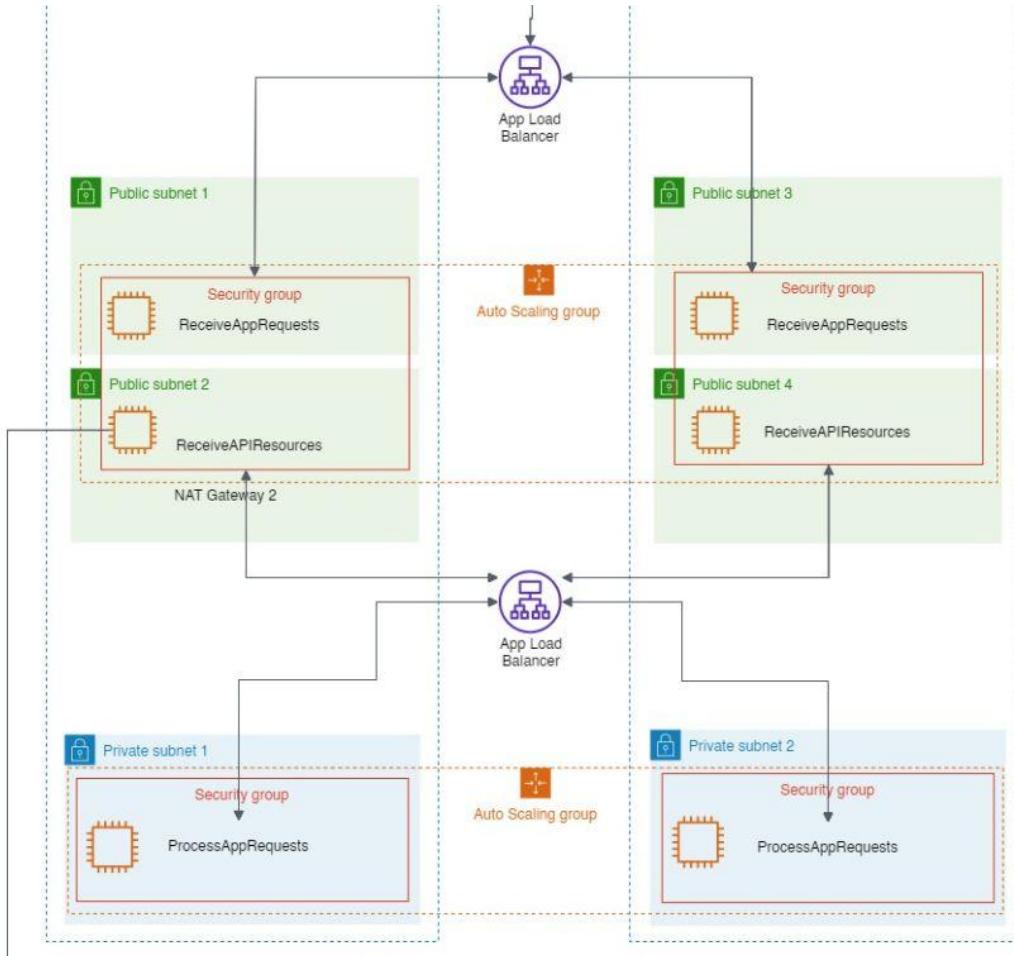
Load balancing and  
auto-scaling



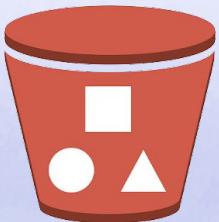
Caching frequently  
used information (S3  
Bucket)



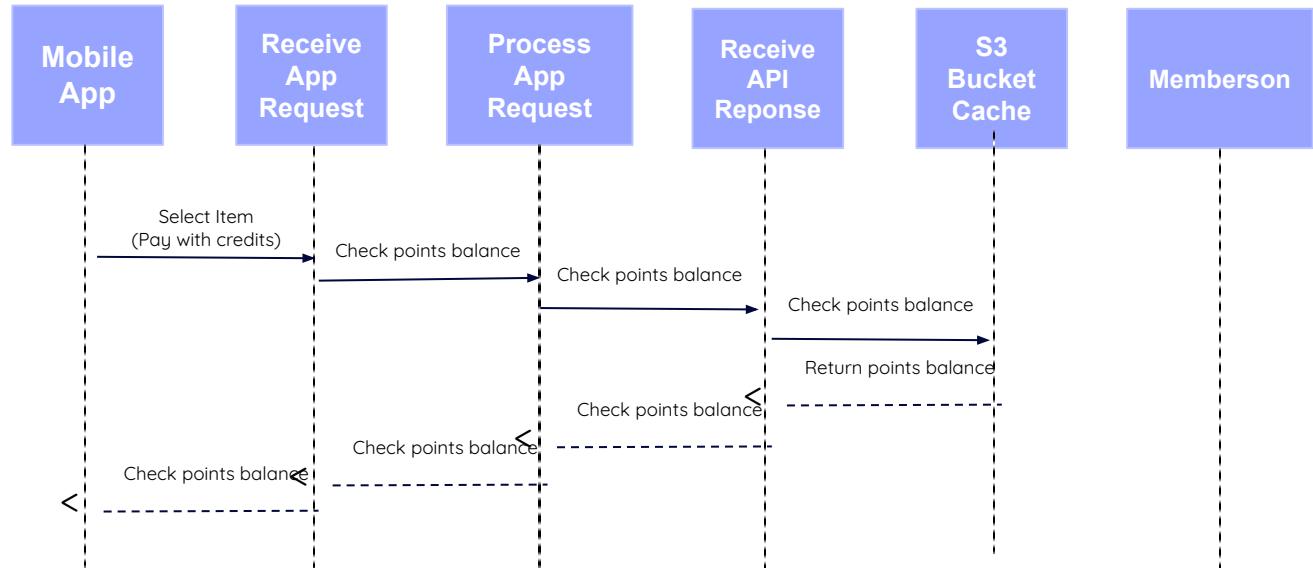
Load balancing and  
auto-scaling



# Use Case: Item Redemption



Caching frequently used information (S3 Bucket)



05

# PERFORMANCE DESIGN DEMO

---

**THANK  
YOU**