# Security & Business Considerations Critical Analysis and Response


## Group Member:

Archer Zhou B00806294
Junqiao Qu B00817232
Kessel Zhang B00809478


## Data Security:

We use AWS Cognito to authenticate user information. We created a user pool on Cognito and used the service they provided to add security to our website. It keeps user privacy (eg. email, phone, password) secure. They ensure data security by encryption at rest and encryption in transit [5].

We deploy our frontend file on Elastic Beanstalk. According to the document of AWS, the Elastic Beanstalk automatically provides the mechanism to prevent basic denial of service attacks. Amazon Web Services offers AWS Shield Standard at no extra charge to all of their customers. This service protects us from 96% of DoS threats, including SYN/ACK attacks, Reflection attacks, and HTTP slow reads [4]. However, there are still some vulnerabilities since the security provided by AWS Shield Standard is limited. For this part, we could use some professional services to prevent DoS attacks, like CloudFlare, AWS Shield Advanced.

The other vulnerability in our website is the access control to API. Currently, there is no authentication process in our API, which means everyone with a URL has access to it. With this access, the potential threat agent could easily revise the content of our webpage. A possible solution in future work is adding AWS IAM to create groups (administrator) and manage security tokens. For API, each request should contain a specific security token. Only users who are in a previously specified group could execute the code in lambda. It can prevent potential attacks on our website.

## Security Mechanism:

Hashing and Transport Layer Security (TLS) are used in our project to secure our data. We implemented these mechanisms by using AWS Cognito, according to the official document of AWS Cognito Transport Layer Security is required:

“All requests to Amazon Cognito must be made over the Transport Layer Security protocol (TLS). Clients must support Transport Layer Security (TLS) 1.0 or later. We recommend TLS 1.2 or later.”[5]

When we are using a public identity provider from AWS Cognito, according to the official website:

“Cognito Identity uses the token from the public identity provider to obtain a unique identifier for the user and then hashes it using a one-way hash so that the same user can be recognized again in the future without storing the actual user identifier.”[6]

It will use a one-way hash which means hashing mechanism is used.

## Budget Estimates:

To reproduce our architecture, something cloud service could be replaced.

1. The AWS elastic beanstalk will be replaced by a local server that has a public IP. It takes more than 100$-200$ to purchase a basic server.
2. The AWS S3 will also be replaced by a local storage machine. Commonly, a 500G mechanical hard disk costs about 20-40$.
3. The AWS Cognito could be replaced by a local database. But it is hard to compare the security.
4. DynamoDB will also be replaced by MySQL on the hard disk. The fee is up to the storage space we need on the hard disk.

Total cost: 150$-250$.

## Services that need attention:

The compute and storage services are most likely to cost most of the money, The cost of these two cloud mechanisms will rise rapidly when scaling up.

For the compute part, in our project, we are using elastic beanstalk to deploy our project, according to the Amazon AWS pricing Calculator, we can get these sample pricing[1]:

| vCPU | RAM(GiB) | SSD(GiB) | Pricing(USD/Month) |
|---|---|---|---|
| 8 | 32 | 50 | 98.81 |
| 16 | 32 | 50 | 192.61 |
| 16 | 64 | 50 | 287.07 |

| | | | |
|---|---|---|---|
| 16 | 64 | 100 | 292.07 |

As we can see, this is the price of one EC2 instance, when we come up with multiple EC2 instances and when we need more computational power, the price can grow very fast. When we need to scale up, we may deploy more EC2 instances and need more computational power, which is the reason why we need to monitor this part.

When it comes to serverless computing, we here use Lambda, the price will also grow up when there are more requests. According to the Amazon AWS pricing Calculator, we can get these sample pricing[3]:

| Number of requests(per minute) | Duration of each request (in ms) | Amount of memory allocated(MB) | Amount of ephemeral storage allocated(MB) | Pricing(USD/Month) |
|---|---|---|---|---|
| 100 | 500 | 128 | 512 | 5.44 |
| 200 | 500 | 128 | 512 | 10.88 |
| 200 | 1000 | 128 | 512 | 20 |
| 200 | 1000 | 256 | 512 | 38.25 |
| 200 | 1000 | 256 | 1024 | 38.39 |

When the number of requests grows up the price will have the same percentage increase, this is why we need to pay attention to it when scaling up.

For the storage part, S3 is mainly used for our products' image files, its price will go up when we have more products selling. But since we only use it to store the product images, we don't need super large storage for it even when scaling up. The problem is from DynamoDB, when we scale up, more users will be using our website, the more user means the more user information and operation log. According to the Amazon AWS pricing Calculator, we can get these sample pricing[2]:

| Storage(GiB) | Average File Size(KB) | Number of writes(Million/Month) | Number of reads(Million/Month) | Price(USD/Month) |
|---|---|---|---|---|
| 100 | 100 | 1 | 1 | 153.13 |
| 200 | 100 | 1 | 1 | 178.13 |
| 200 | 200 | 1 | 1 | 306.25 |
| 200 | 200 | 2 | 1 | 556.25 |
| 200 | 200 | 2 | 2 | 562.50 |

As the table shows, when we try to scale up, we need a better storage database. The price can grow very fast when we need better storage and we need to add monitoring to it.

## Reference:


[1] AWS Pricing Calculator. [Online]. Available: https://calculator.aws/#/createCalculator/EC2. [Accessed: 29-Mar-2022].

[2] AWS Pricing Calculator. [Online]. Available: https://calculator.aws/#/createCalculator/DynamoDB. [Accessed: 29-Mar-2022].

[3] AWS Pricing Calculator. [Online]. Available: https://calculator.aws/#/createCalculator/Lambda. [Accessed: 29-Mar-2022].

[4] J. Barr, “AWS Shield – Protect your Applications from DDoS Attacks,” Amazon Web Services. [Online]. Available: https://aws.amazon.com/cn/blogs/aws/aws-shield-protect-your-applications-from-ddos-attacks/. [Accessed: 29-Mar-2022].

[5] “Data Protection in Amazon Cognito - Amazon Cognito,” docs.aws.amazon.com. https://docs.aws.amazon.com/cognito/latest/developerguide/data-protection.html.[Accessed: 29-Mar-2022].

[6] “Amazon Cognito FAQ,” Amazon Web Services, Inc. https://www.amazonaws.cn/en/cognito/faq/ (accessed Mar. 29, 2022).[Accessed: 29-Mar-2022].