SpamOverflow

Software Architecture

May 13, 2025

Evan Hughes & Richard Thomas

1 Brief

Design a service for scanning and filtering spam/malicious emails. Specially the service needs to support:

- Scanning an email via an API request.
- Providing access to a specified REST API, e.g. for use by front-end interfaces and internal teams.
- Remaining responsive while scanning emails.

Task You are working for SpamOverflow, a new competitor in the email security space. SpamOverflow uses a microservices based architecture to implement their new malicious email filtering platform. The CEO saw on your resume that you are taking Software Architecture and has assigned you to design and implement a service. This service must be scalable to cope with a large influx of emails.

Requirements Email filtering software can filter email as it arrives or after. SpamOverflow will implement a service that does not impede the flow of traffic (i.e. does not prevent email from arriving). It will receive an API call when the mail server receives an email message. The service then pulls the email from the user's inbox, as fast as it can, to prevent the user from seeing the malicious email or clicking any links.

Commercial email providers send an API request for each email received. For optimal performance this service needs to be able to handle a large number of requests in a short period of time.

Since these emails can be dangerous, the service must be able to report that it is bad or good in a timely manner. Though genuine emails that are incorrectly marked as dangerous should be returned to the user as quickly as possible.

Persistence is an important characteristic of the platform. Customers will want to analyse why emails were flagged after the fact. Upon receiving an email scan request, and after filtering, the system must guarantee that the data has been saved to persistent storage before returning a success response.

2 Outline

Introduction (5 minutes)

Introduction to the brief and resources, including the API specification[1] and quality scenarios in section 3. A tool will be used to scan emails for malicious content, called SpamHammer[2].

Planning (10 minutes)

In small groups, discuss the following issues or any others you think are relevant to designing the service.

1. What are the key requirements introduced by the quality scenarios?

[1] https://csse6400.uqcloud.net/api/spamoverflow
[2] https://github.com/CSSE6400/SpamHammer

2. What strategies can you implement to support these scenarios?
3. What AWS resources would prove helpful?
4. What are the likely bottlenecks?

Design (20 minutes)

In your group, design an appropriate architecture for SpamOverflow. You need to consider the flow of an API request through your service, use the API specification and quality scenarios in section 3 to ensure all use cases have been considered.

Presentation (15 minutes)

In the remaining time, each group should present their proposed architecture design. This is an opportunity for discussion amongst the class to point out limitations of the proposed system designs.

3 Quality Scenarios

Q1: Steady Stream Steady receipt of email messages at a rate of *M* per minute, fairly evenly spread across all customers. Approximately 20% of the messages are malicious.

Q2: Bad 'News' Stream Steady receipt of email messages at a rate of *N* per minute, fairly evenly spread across all customers. Approximately 80% of the messages are malicious.

Q3: Peaks and Troughs Periods of receipt of a high volume of email messages, followed by periods of low volume.

Q4: High Value Customer The Department of Defence (DoD) has adopted SpamOverflow. They are a high value customer and you must ensure that their requests are handled quickly. All other customers have a service level agreement (SLA) that guarantees a certain level of responsiveness. You cannot ignore their requests to only prioritise the DoD's requests.

Q5: Leaked Directory A bad actor has managed to get the email addresses of all employees of <Large Company>. They have sent a phishing email to all of the users advertising a pay raise with a link to a fake login page. The email is sent to all 10,000 employees at the same time.

Q6: Personalised Attack A bad actor has trained an AI model using social media profiles of targeted victims. They can generate personalised phishing email messages based on personal information. As the messages are personalised, they can be of greatly different lengths and contain different content. These messages can only be identified by SpamHammer. They have sent these phishing messages to 2,000 users at the same time.