

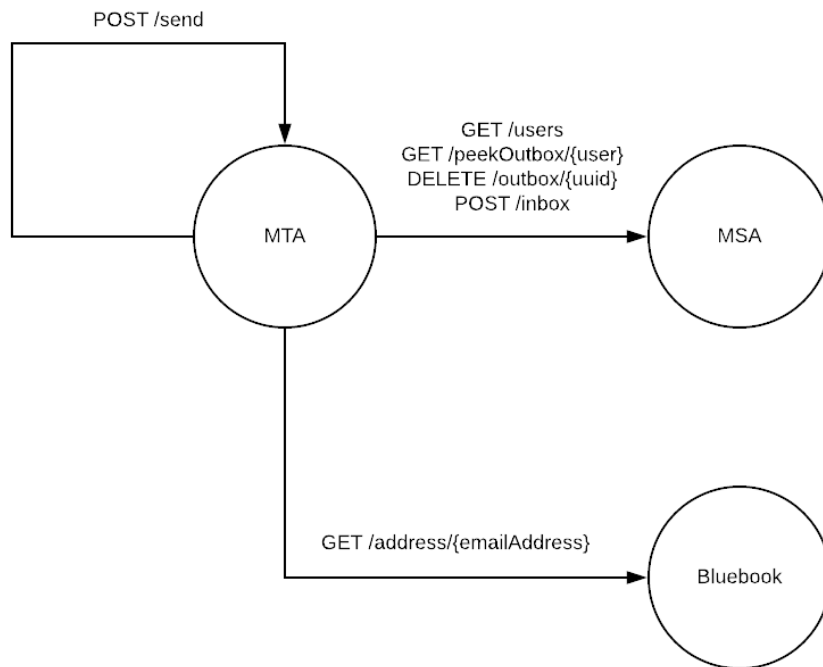
ECM3408 Enterprise Computing

Continuous Assessment

Email Microservice

Candidate Number: 123446

Microservice Architecture and HTTP requests



Bluebook

The bluebook microservice is a small service with a single REST endpoint. It contains mappings from email server domain names to their respective network addresses, e.g. "here.com" to "127.0.0.1:8888". A client makes a GET request with an email address as parameter and receives a network address in return. This is so that a Mail Transmission Agent (MTA) microservices of an email server can deliver emails to the correct MTAs of other email servers. The mapping between domain names and network addresses is initialised at the startup of the bluebook service and cannot be changed once it is running. The bluebook microservice does not make any HTTP requests.

Endpoints:

- GET /address/{emailAddress}

Status Code	Description
200 OK	The email address is legal (contains @) and there is a

Status Code	Description
	mapping from the domain to a network address.
500 Internal Server Error	The server could not marshal the network address in the response.
404 Not Found	The email address is illegal (doesn't contain @)
404 Not Found	The server has no network address for the domain

MSA

The Mail Submission Agent (MSA) is one of the two microservices that constitute an email server. The MSA contains all users and inboxes of an email server. It is responsible for the following:

- Adding a single email to a user's inbox
- Adding a single email to a user's outbox
- Listing all emails in a user's inbox
- Listing all emails in a user's outbox
- Listing the most recent message from a user's outbox
- Listing the most recent message from a user's inbox
- Listing all users on this email server
- Deleting a specific email from a user's inbox using the email's UUID
- Deleting a specific email from a user's outbox using the email's UUID

The MSA has a map where email addresses of users are mapped to their respective inboxes and outboxes. Users are added to the email server when the User Agent sends a request to add an email to the user's outbox for the first time. Users are uniquely identified by their email address. When an email is added to a user's outbox it is given a UUID. Although not required in the spec, the UUID is needed because two emails in a user's in/outbox might have the exact same source, destination, and body.

In addition to listing the entire outbox and inbox, clients can read the most recent message from a user's inbox or outbox. The MTA service uses these endpoints to move emails from outboxes of one email server to inboxes of another email server. The MSA does not send any HTTP requests.

Endpoints:

- POST /outbox
- POST /inbox
- GET /outbox/{user}

- GET /inbox/{user}
- GET /peekOutbox/{user}
- GET /peekInbox/{user}
- GET /users
- DELETE /outbox/{user}/{uuid}
- DELETE /inbox/{user}/{uuid}

POST /outbox

Status Code	Description
201 Created	The email was successfully added to the user's outbox
400 Bad Request	The body of the request could not be parsed or the source of the email does not match the domain
404 Not Found	The user does not exist on this email server

POST /inbox

Status Code	Description
201 Created	The email was successfully added to the user's inbox
400 Bad Request	The body of the request could not be parsed
404 Not Found	The user does not exist on this email server

GET /outbox/{user}

Status Code	Description
200 OK	The user exists on the server and their outbox is listed
404 Not Found	The user does not exist on this server
500 Internal Server Error	Error while marshaling the outbox to JSON

GET /inbox/{user}

Status Code	Description
200 OK	The user exists on the server and their inbox is listed
404 Not Found	The user does not exist on this server
500 Internal Server Error	Error while marshaling the inbox to JSON

GET /peekInbox/{user}

Status Code	Description
200 OK	The user exists on the server the most recent email from their inbox is listed
404 Not Found	The user does not exist on this server
500 Internal Server Error	Error while marshaling the inbox to JSON

GET /peekOutbox/{user}

Status Code	Description
200 OK	The user exists on the server the most recent email from their outbox is listed
404 Not Found	The user does not exist on this server
500 Internal Server Error	Error while marshaling the inbox to JSON

GET /users

Status Code	Description
200 OK	A list of users is returned
500 Internal Server Error	Error while marshaling the users to JSON

DELETE /outbox/{user}/{uuid}

Status Code	Description
202 Accepted	The email with the specific UUID was succesfully deleted from the user's outbox
404 Not Found	The user does not exist on this server

DELETE /inbox/{user}/{uuid}

Status Code	Description
202 Accepted	The email with the specific UUID was succesfully deleted from the user's inbox
404 Not Found	The user does not exist on this server

MTA

The Mail Transfer Agent (MTA) is the second microservice and together with an MSA makes up an email server. Every email server has an MTA and MSA. At set intervalls, the MTA of an email server requests a list of all users on its server polls the outboxes of all users found on its email server and delivers the emails to their destination MTA.

Endpoints:

- POST /send/{user}

POST /send/{user}

Status Code	Description
201 Created	The email was successfully sent to the users inbox
400 Bad Request	The email does not have the legal format
404 Not Found	The user does not exist on this server
500 Internal Server Error	Email could not be delivered or JSON could not be marshaled