Cargomail — A Privacy-Aware Email System

Revision 1 (Draft)

Igor Zboran
izboran@gmail.com

***Abstract*—Electronic mail (email) is the most pervasive form of business information exchange. Email is used not only as an interpersonal communication tool but also as a "default choice" for sending files.**

This paper introduces Cargomail: a privacy-aware email system designed with the precept that the storage of email messages should be the responsibility of users and not of email providers. To succeed in this endeavor, we propose to add a logistics overlay on top of the existing email infrastructure.

Additionally, we explain the importance of separating the identifier (email address) and the locator (mailbox) in the design of the Cargomail architecture. Our paper concludes by highlighting the significant advantages of this design.

I. Introduction

The main components of the email system were designed between the early 1970s and mid-1990s by many inventors. Over time, email has become the most commonly used Internet application. Nowadays, email is the only ubiquitous communication system on the Internet that was built in a decentralized fashion. Moreover, the email infrastructure forms the backbone of the global digital identity.

II. Current Situation

Today, outgoing email is typically transferred from the source system to the destination system as a single text-encoded file using the Simple Mail Transfer Protocol (SMTP). SMTP is an over-40-year-old push-based protocol, and even though SMTP has been updated, modified, and extended multiple times to increase security and efficiency, it still lags behind modern web-based protocols.

III. Problems and Issues

Despite the importance of email infrastructure, the whole ecosystem still relies on more than 40-year-old architecture and protocol design. There have been spam and attachment issues from the very beginning. While conceptually sound as a communication means, the email system is structurally obsolete and functionally deficient.

A. Functional and Security Flaws

Even though the major email service providers claim their email services to be safe, the fact remains that fundamental security and functional flaws are not fixed. There is still a dichotomy of attachment delivery; bulky files are not transferred as an attachment but are shared via links. A "file sharing" is unnatural for the email system, where each message with attachments is expected to be time-consistent. Additionally, shared links can pose a consent phishing attack threat, where an attacker tricks users into granting malicious application access to sensitive resources.

B. Confidentiality and Privacy

Now, if we (as users) want to use a single email address, we have no choice but to use a single email service provider for all categories of communication. Information about every email we send or receive—"buying a car or a home, applying for a loan, taking out insurance, purchasing potato chips, requesting a government grant, getting turned down for credit, going to work, seeing a doctor" [1]—is funneled through the same service providers. This raises privacy concerns, leading to what we may call a One-Address-Fits-All privacy issue.

C. Hyperlinks to External Files

Documents, images, videos, and audio should be an integral part of the email. The concept of keeping a time-consistent, recipient-owned copy of the sender's files is critical in some industries. Here is the list of issues with hyperlinks to external files in email messages:

- expired, unknown, blocked, phishing, or malicious hyperlink
- masked hyperlink target using a URL shortener
- target updated – not the version it is expected
- target changed – forgery
- target encrypted – need a password
- access control – requires signup or sign in
- consent phishing attack

Given these points—You are buying a "pig in a poke" with each hyperlink to the external file in the email message.

D. Content Repository

The current email system is missing a content repository—a "file system on steroids" with the capability to create, store, locate, and exchange any content.

IV. Proposed Solution

Given that the current email system is lagging behind modern communication and collaboration tools, we propose to build a logistics overlay on top of the existing email infrastructure to address inefficiencies and vulnerabilities in the current email system.

A. Goals and Objectives

At the core of the proposed solution is an attempt to improve the usability of email as an interpersonal communication tool and as a default choice for sending files while supporting data sovereignty and governance. Our strategic decision is to ensure the new architecture is interoperable with the existing email architecture to enable end-to-end communication between early adopters and the current email system. This approach allows that the proposed solution can be deployed incrementally. We aim to provide a high-level overview of the solution rather than delving into the details. The topic is broad, and covering all design decisions and implementation details in one document is virtually impossible.

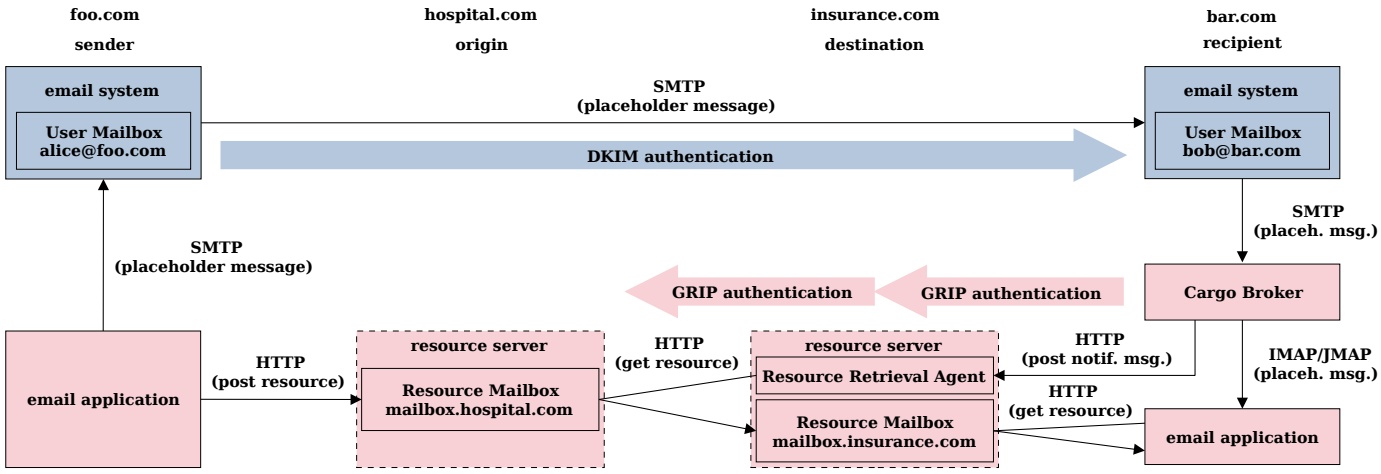

Fig. 1. Cargomail architecture (the *sender-resource_server-resource_server-recipient* topology)

B. Concept

In the current email system, SMTP RFC 5321 [2] defines an email address as a character string that identifies a user to whom mail will be sent or a location into which mail will be deposited. In that sense, the terms mailbox and email address can be used interchangeably. However, Cargomail uses an architecture that separates the identifier and the locator of the address/mailbox pair to introduce a new type of mailbox—a Resource Mailbox. This approach offers a different way of organizing email communication. In our design, each email includes a placeholder message stored in the User Mailbox and related external digital resources (e.g., documents, books, images, videos, audio) that are kept in the Resource Mailbox on the email-specific resource server. The actual correspondence takes place between the Resource Mailboxes, while the default User Mailbox of the current email system is used to send the placeholder message, as shown in Figure 1. A newly designed service called Cargo Broker coordinates email delivery. To reference external resources, Cargomail uses content-addressed identifiers instead of URLs, as the location of the resources differs between the sender and recipient.

The approach depicted in Figure 1. separates the identifier (email address) and locator (Resource Mailbox URL). By decoupling the locator and identifier, data can be exchanged between diverse Resource Mailboxes using one email address per user. Cargomail exchanges referenced resources between the Resource Mailboxes using the Global Reference Identity Protocol (GRIP) [3] authentication mechanism. This enables both the sender and recipient to use multiple Resource Mailboxes while still using their single email address.

Moreover, Cargomail does not use attachments or links to external files. Instead, it uses an editor that allows users to reference documents, images, and videos in the message body by their content via a cryptographic hash value while keeping the respective resources in a content-addressed Resource Mailbox storage. The final download link is constructed in the email client using the destination Resource Mailbox URL and the cryptographic hash value of the referenced resource conveyed in the placeholder message, see Figure 2. The content of the external resources can be displayed directly in the message body.

C. Key Points

• Each email consists of a placeholder message stored in the User Mailbox and related external digital resources stored in the Resource Mailbox.

• The resources owned by the sender, stored in the origin Resource Mailbox, are temporarily shared with recipients. Following a successful sharing process, a placeholder message is sent to each recipient through the standard email system. The placeholder message contains the origin Resource Mailbox URL, the cryptographic hash values of the referenced resources (Content-IDs), and the category of correspondence, e.g., personal, business, or healthcare, as illustrated in Figure 2.

Placeholder Message

```
From: alice@foo.com
To: bob@bar.com
Cc:
Subject: Test message
X-Origin-Resource-Mailbox-URL: mailbox.hospital.com;
expiration="Sa, 4 Mar 2023 14:05:01 +0100 (CEST)"
X-Correspondence-Category: healthcare
X-Destination-Resource-Mailbox-URL: mailbox.insurance.com
Content-Type: multipart/mixed; boundary=42
MIME-Version: 1.0

--42
Content-Type: message/external-body;
        access-type=content-addressed-uri;
        hash-algorithm=sha256;
        size=2563

Content-Type: text/plain
Content-Disposition: inline
Content-ID: <2f77668a9… ...32b2f91>

--42
Content-Type: message/external-body;
        access-type=content-addressed-uri;
        hash-algorithm=sha256;
        size=87328426

Content-Type: video/mp4
Content-Disposition: attachment
Content-ID: <d534be829… ...bfefbc6>

--42--
```

Header added by Cargo Broker

Fig. 2. Placeholder message

• After receiving the placeholder message, the recipient's Cargo Broker determines (according to the user's preferences and the category of correspondence) which destination Resource Mailbox will be used for communication. Once the destination Resource Mailbox is determined, the Cargo Broker creates the notification message with the origin Resource Mailbox URL and the cryptographic hash values of the referenced resources and posts it to the relevant destination resource server using the GRIP authentication mechanism. The Cargo Broker then adds the header with the destination Resource Mailbox URL to

the placeholder message, as illustrated in Figure 2., and delivers it to the email application using IMAP or JMAP protocol.

• The Resource Retrieval Agent at the destination resource server gets the origin Resource Mailbox URL and the cryptographic hash values of the referenced resources in the notification message. Using the GRIP authentication mechanism, the agent tries to retrieve the external resources from the origin Resource Mailbox. After successful authentication, the data is retrieved and stored in the destination Resource Mailbox. Finally, the email application downloads the relevant data from the destination Resource Mailbox and reconstructs the original email according to the placeholder message template.

D. GRIP Authentication

To enable controlled access to data within the Cargomail logistics layer, we need an authentication/authorization mechanism designed in a decentralized manner.

Design decision: The initial idea was to use an authentication mechanism based on the OAuth 2.0 framework. However, this approach turned out to be inappropriate due to the fact that in the OAuth 2.0 framework, the roles of the authorization server, resource server, and client are co-located in the same security domain. Therefore, we developed a new identity propagation mechanism to address this issue. The main idea behind the GRIP authentication concept is that a cross-domain identity propagation system should work much like an email system. We were inspired by the fact that in a special case—as a de facto part of the globally accepted signup and password recovery mechanism—an email with a confirmation link works as a "token" in MIME format. Equally important for us was the finding that such a "token" is forwardable. We can say that these "tokens" are self-issued (the issuer is a Message Transfer Agent), DNS-bound (via the DKIM signing mechanism), chained (via the sender/recipient headers), and nested (forwarded as an attachment). With this in mind, we designed GRIP, a token-based zero-trust security protocol that authenticates service requests between untrusted hosts across the Internet.

Context conveyance: To meet the Cargomail architecture needs, we use self-issued DNS-bound tokens to propagate the user/broker context from the Cargo Broker to the destination resource server and the user/broker/agent context from the destination resource server to the origin resource server, all in different security domains. Proper chaining and nesting of tokens, as illustrated in Figure 3, is crucial to convey identity claims about the user, broker, and agent across the network.

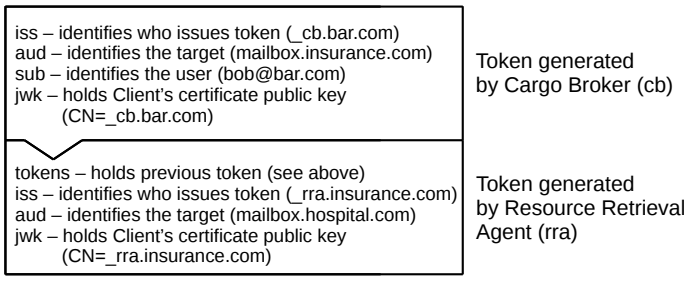

Fig. 3. Nested, chained, self-issued DNS-bound tokens

Actors authenticity: In some service-to-service communication scenarios, three identities are employed: user, client, and server identities. Fundamentally, mutual TLS (mTLS)/TLS certificates resolve client and server identities, while tokens resolve client and user identities. A DNS-bound token is a self-issued assertion in a JWT format signed by an mTLS private key that the first service uses to authenticate to the second service. The mTLS public key hash is published in the first service domain using the DNS TXT record, where the CN attribute of the mTLS public key certificate is used as a global client identifier in respect of the service it represents. Given that, to ensure that the proper context is conveyed and tokens cannot be replayed, the Cargo Broker and the Resource Retrieval Agent, as well as the Resource Retrieval Agent and the origin Resource Mailbox, are authenticated through mTLS.

In summary: The zero-trust concept implemented in the GRIP authentication mechanism improves the usability and security of the proposed Cargomail system.

V. Advantages compared to the Current Email System

Cargomail has several decisive advantages over the current email system. These advantages encompass improved security and privacy, increased efficiency, and ease of use. Overall, Cargomail provides users with a more reliable and effective way of exchanging information.

A. Spam Protection

With Cargomail architecture, the email address and mailbox are separated. The actual email correspondence is exchanged between Resource Mailboxes, or "locators," while the email address, or "identifier," is only used to deliver a placeholder message. This architecture does not protect against unsolicited emails—anyone can send you an email—it allows for a more detailed assessment of the sender's reputation by individually evaluating their email address identifier and origin Resource Mailbox locator.

B. Data Sovereignty

The Resource Mailbox is decoupled from the user's email address. This separation allows a user with a single email address to use multiple Resource Mailboxes side by side. With a single email address and numerous mailboxes, Cargomail can keep official, business, healthcare, and personal correspondence separately on designated resource servers, ensuring compliance with relevant laws and regulations.

C. Data Residency and Data Governance

Email messages and their resources are kept together in chronological tamper-resistant records. A content-addressed Resource Mailbox storage ensures the authenticity and integrity of email messages and their resources. The cryptographic hash value guarantees the storage of only a single instance of a resource. With the capability to create, store, locate, and exchange any content, including documents, images, videos, and audio, the proposed solution gives users full control over data residency and governance.

VI. Implementation Concerns

This section deals with the issues of implementing the proposed data exchange mechanism into the existing email infrastructure.

A. MIME External-Body Subtype

In the placeholder message, Cargomail utilizes the standardized MIME message/external-body subtype defined in RFC 2046 [4] to indicate that the actual body data are not included but merely referenced. Although this is a standardized method of referencing external data, not every email provider handles it correctly.

B. Decentralized Notification System from Scratch

Instead of relying on an SMTP-based email infrastructure, a newly designed web-based decentralized notification system may be worth considering. The system would use the GRIP authentication mechanism to send placeholder messages in the sender's context.

VII. Exchange Network for Digital Resources

The logistics layer of the Cargomail system creates an opportunity to build a global ecosystem around digital resources. The GRIP authentication mechanism (i) allows using a user-centric approach to manage access to individual resources, (ii) enables trust among all parties, and (iii) ensures interoperability among resource servers.

A. Cargomail Logistics Topology

Cargomail has the capability to operate in several logistics topology scenarios. We will highlight three of these scenarios to demonstrate its architecture's versatility.

Two distinguished resource servers: Each resource server operates within its own security domain, which we refer to as the *sender-resource_server-resource_server-recipient* topology. The email provider has no access to the exchanged data.

Two shared resource servers: Each resource server shares a security domain with its respective sender's or recipient's email provider, which we refer to as the *sender/resource_server-resource_server/recipient* topology. This logistic topology is appropriate for corporate environments operating on-premises where the corporate email system may access the exchanged data.

Single distinguished resource server: A single resource server operates within its own security domain. We called it the *sender-resource_server-recipient* topology. The email provider has no access to the exchanged data, and exchanged data never leaves the resource server.

B. Applications and Use Patterns

Cargomail overcomes the limitations of traditional email when working with digital resources. Its architecture offers new, not typical applications and use patterns.

Digital signatures: Using public key cryptography to sign a placeholder message provides a reliable method to guarantee both the authenticity and integrity of all digital resources referenced by the message. This ensures that the recipient can have complete confidence in the origin and content of the linked resources.

End-to-End Encryption (E2EE): Encrypting digital resources can be tricky. The cryptographic hash value of the symmetrically encrypted digital resource must be included in the placeholder message, along with the corresponding symmetric encryption key. The placeholder message body must be selectively encrypted using public key cryptography, where only the symmetric encryption keys are encrypted, leaving some metadata unencrypted. Each symmetric encryption key must be separately encrypted using the recipient's corresponding public key.

Publish-subscribe patterns: TBD
TBD
TBD
TBD

VIII. Models and Scenarios

Although Cargomail can be integrated into the email system of any email service provider, we slightly digress to introduce two visionary models of what a global email ecosystem might look like in the future.

A. Estonian Model

In this model, the government provides email services with User Mailboxes. To avoid the risk of governmental surveillance, Cargomail allows citizens to use non-governmental Resource Mailboxes, e.g., from financial institutions or healthcare providers. Using non-governmental, sector-specific Resource Mailboxes increases the privacy of individual citizens, as the government cannot obtain detailed information about their activities.

B. Postal Model

According to UPU research [5], more than 93% of postal operators provide some form of digital postal service either directly or in partnership with other companies. Cargomail allows postal operators to expand and become public email service providers or innovate their existing email services and provide the User Mailbox services with the ability to attach the Resource Mailboxes from the government as well as other institutions and organizations.

IX. Related Work

While this proposal seems closely related to the Internet Mail 2000 [6] concept proposed by Daniel J. Bernstein, we differ in the new idea that the storage of email messages and their resources should be the responsibility of users and not of email providers.

X. Conclusion

Cargomail can play an essential role in communication across various industries in the public and private sectors.

A. Overall Summary

Combining Cargomail with the current email system creates a hybrid architecture that meets the needs of a modern communication tool. Exchanging data between the appropriate mailboxes eliminates privacy concerns.

B. Future Work

Cargomail brings content-addressed storage and a new data exchange mechanism into the email ecosystem, predestining the proposed system to become more than a bare messaging tool. It would be fascinating to build a prototype of the proposed solution to serve as a proof of concept.

XI. Discussion

This proposal offers a fresh and uncharted outlook on the email system. The main question is whether it is worthwhile to integrate Cargomail into the current email system or to build up a new communication network based on the Cargomail architecture and GRIP mechanism.

References

[1] Jeffrey Rothfeder. 1992. Privacy for Sale: How Computerization Has Made Everyone's Private Life an Open Secret (pp. 22-23). Simon & Schuster Trade.
[2] J. Klensin, "Simple Mail Transfer Protocol", IETF RFC 5321, 2008, https://www.ietf.org/rfc/rfc5321.txt.

[3] I. Zboran, "Global Reference Identity Protocol (GRIP)," GitHub repository, March 2023, https://github.com/cargomail-org/grip.
[4] N. Freed, Innosoft, N. Borenstein, First Virtual, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", IETF RFC 2046, 1996, https://www.ietf.org/rfc/rfc2046.txt.
[5] Universal Postal Union/Activities/Digital Services, accessed 27 March 2023, https://www.upu.int/en/Universal-Postal-Union/Activities/Digital-Services.
[6] Internet Mail 2000, accessed 23 April 2023, https://en.wikipedia.org/wiki/Internet_Mail_2000.