Authorization-Enhanced Mail System

Igor Zboran
izboran@gmail.com
June 24, 2021

Abstract

Electronic mail (email) is the most pervasive form of business information exchange. Email is often used not only as an interpersonal communication tool, but also as the default choice to send files. In this paper the User-Managed Access (UMA) [1, 2] authorization framework is proposed to overcome the data storage, access control and data transfer limitations of the current mail system. Today, outgoing mail is typically transferred from the source system to the destination system as a single text-encoded file using Simple Mail Transfer Protocol (SMTP). SMTP is a more than 40-year-old protocol that emerged long before World Wide Web became popular. Despite the fact that SMTP has been updated, modified and extended multiple times to increase security and efficiency, it still laggs behind modern web-based protocols. We propose to mirror the existing email ecosystem by creating a new secure and scalable web-based communication infrastructure. The web-based data transfer in combination with a decentralized API access management significantly leverages email security and enhances mail system utilization.

Introduction

The main components of the mail system have been designed between 1971 and 1992 by many inventors. In the course of time, email has become the most commonly used application of the Internet. Nowadays email is the only truly decentralized communication system of the Internet and the email infrastructure forms the backbone of the worldwide digital identity.

Problem

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

Current Situation

With the rising popularity of free email providers, such as Gmail or Outlook.com, web-browsers are increasingly being used to access the mail server. From a user standpoint, it is easy to read and send emails via web-browser on any device, from anywhere in the world. Centralized access to the mailboxes increases the security of web-based mail systems.

Current Flaws

Even though the main email service providers claim email accounts to be safe, the fact remains that major security and functional flaws are not fixed. There is still an attachments delivery dichotomy; bulky files are not transferred as an attachment but are shared via links. An “attachment sharing” is not natural for mail systems where each message with attachments is expected to be time-consistent. Shared links pose a consent phishing attack threat where attacker tricks users into granting a malicious application access to sensitive resources. This is known as an OAuth 2.0 authorization exploit. The Authorization-Enhanced Mail System is resistant to this security exploit as there are no direct user involvement in access granting.

Proposed Solution

Given that the current mail system is lagging behind modern communication and collaboration tools, we propose a web-based access control management and consequently a web-based data exchange channel for the new email ecosystem.

Motivation

Email still the most popular communication tool is lacking an important part of today's modern communications systems – an authorization framework. Understanding this lead us to implement the UMA authorization framework into the new email ecosystem. At the core of the proposed solution is an attempt to improve the usability of email – not only as an interpersonal communication tool, but also as the default choice to send and store files.

Main Concept

The Authorization-Enhanced Mail System (AEMS) is designed to follow the Identity and Access Management (IAM) best practices while mirroring the existing email ecosystem by creating a new secure and scalable web-based communication infrastructure. The UMA framework introduces a resource server, an authorization server and a requesting party client into the new mail system. To transfer data from sender to recipient, AEMS uses the Decentralized Identity-Based Access Control (DIBAC) [3] technology that is built around the UMA 2.0 protocol standard.

WebFinger

To decouple the mailbox from the email address AEMS uses the WebFinger protocol ...

Push / Pull

AEMS uses the two-way push-pull data transfer …

Push email metadata

The …

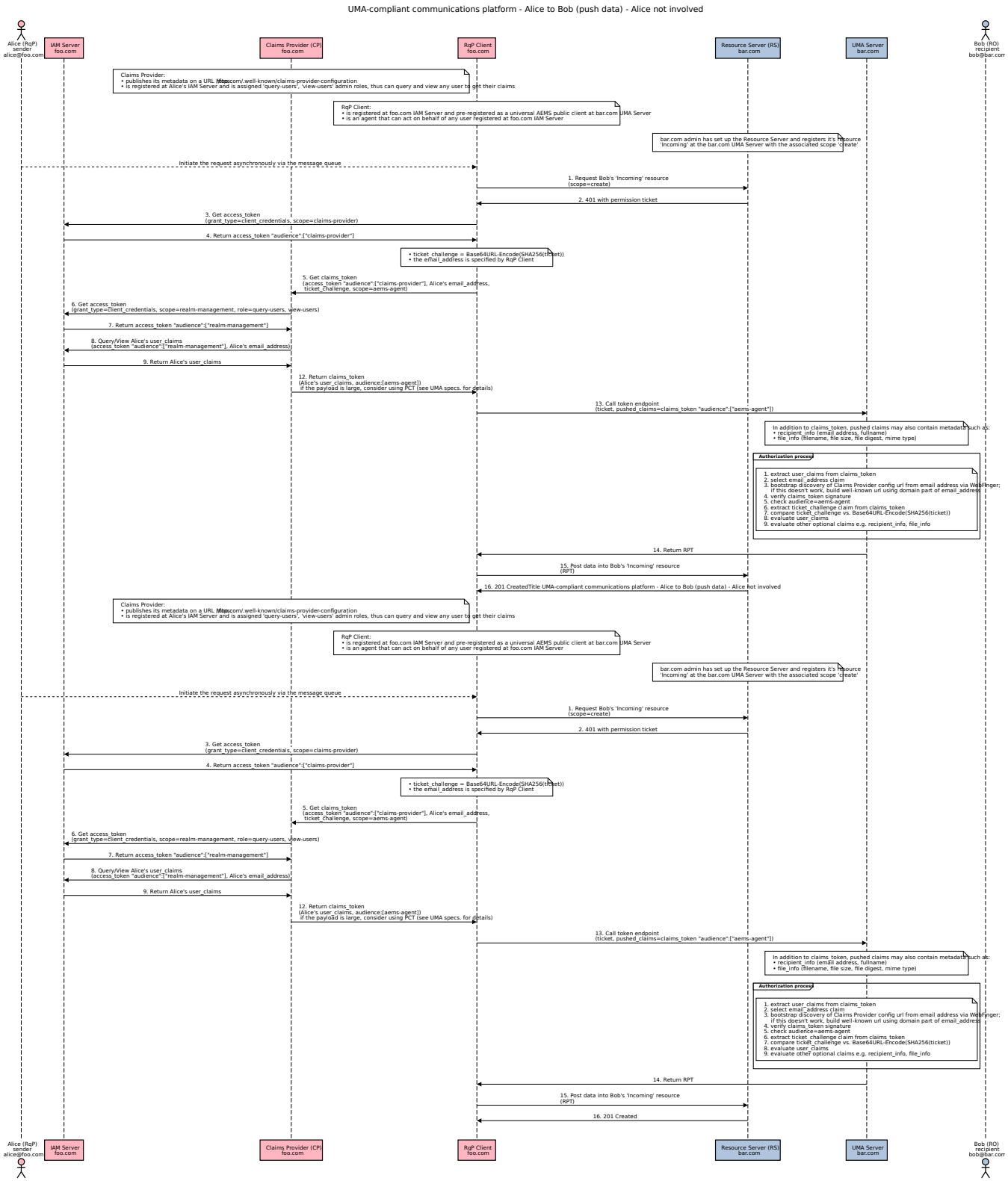

Figure 1. alice@foo.com to bob@bar.com – push email metadata

Pull email resources

The ...

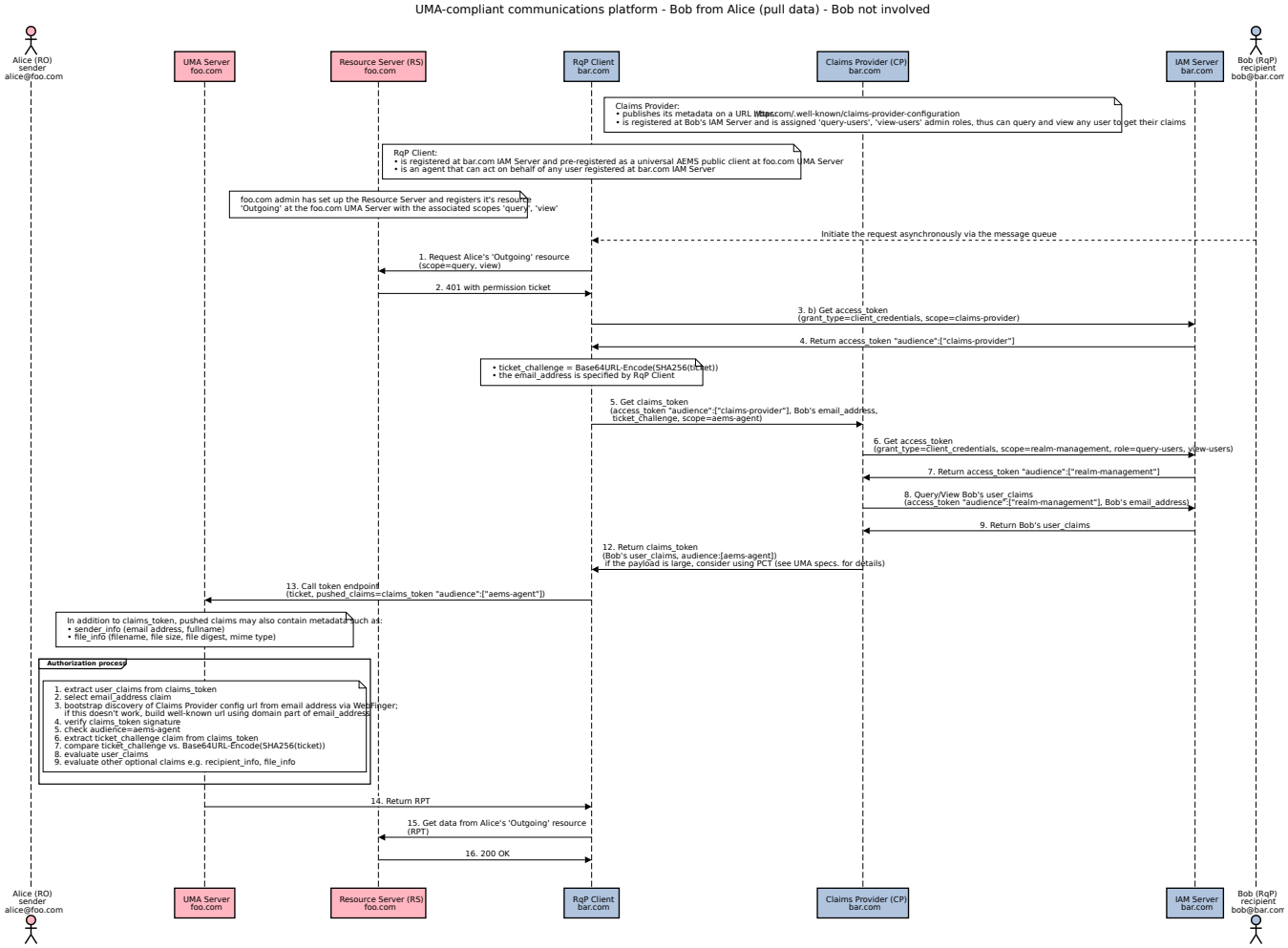

Figure 2. bob@bar.com from alice@foo.com – pull email resources

Features and Comparison with Current Mail System

The novelty of the proposed solution approach can be assessed by comparison with the current mail system.

New Features

The proposed AEMS solution provides several new features that are lacking in the current mail system:

- Intrinsic privacy-preserving properties. Each user can have their mailbox running on a separate server. The user can also run his own AS.
- Single mailbox for both incoming and outgoing emails.
- Multiple mailboxes per one email address.
- Built-in cross-domain autonomic (without conscious user intervention) access control using the standardized UMA protocol.
- Efficient data exchange channel.
- No attachments size limit. Attachments are transferred as separate files without size limit.
- Linked content using a clickable hyperlinks.
- Instant messages. Messages and attachments are transferred separately, there is no need to wait for incoming bulky message-with-attachments file. Attachments-stripped bare messages are transferred with a higher priority.

Comparison with Current Mail System

The use of AEMS has many advantages over the current mail system. The architecture of AEMS guarantees more control over potential security and privacy issues such as leakage of intellectual property or loss of confidential content and makes the system compatible with enterprise security policies. In the following we highlight the main advantages of the proposed solution compared to the current mail system.

1. Security and Privacy

User correspondence takes place between separated mailboxes decoupled from email addresses. This allows a user with a single email address to use simultaneously multiple mailboxes. To separate official, business, personal and healthcare correspondence, AEMS provides the flexibility for storing emails according to various criteria within an appropriate mailbox provider.

2. Usability

With the capability to store, locate, send and receive any content including documents, images, audios and videos, the proposed solution can be considered a promising platform for a file storage.

3. Integrations

AEMS provides a standardized RESTful application interface to ease the integrations with external systems.

Conclusion

AEMS can play an important role in communication across various industries in the public and private sectors. Consolidation of repository, communication and identity represents a pillar of the web-based human-centric Next Generation Internet [4].

Overall Summary

The new email technology in combination with the UMA framework creates a composite architecture that meets the needs of the modern communication tool. The proposed solution can be used as a content services platform to provide the e-records storage, exchange and retrieval system protected by the standardized authorization framework utilized by users through the email application.

A consolidated access control and a new data exchange mechanism leverages email security and enhances the mail system utilization. The question arises as to whether the standard implementation of UMA 2.0 will fit into the current mail system and how difficult it will be to build the UMA email extension.

Future Work

The UMA framework brings into the new email ecosystem a web-based data storage and exchange technology that predestine the proposed system to become more than a bare messaging tool.

The following are potential future R&D areas:

- Consider a Consent mechanism extension design.
- Explore linked content using a clickable hyperlinks – linking content across the business.
- Design an extension for exchanging tagged messages and attachments – grouping content across the business.
- Design an attachment versioning extension – managing the attachment content changes.
- Explore health information exchange between healthcare professionals and inspect use of email communication between patients and healthcare professionals.
- Employ regular email clients and applications using JMAP protocol to support a standardized email API.
- Design and use a proprietary GraphQL API to replace the poorly adopted JMAP protocol.
- Incorporate an electronic mailing/discussion list system into the proposed solution to extend the basic email functionality.

A prototype implementation of the proposed solution, working as a proof of concept, would be interesting to build.

Acknowledgements

We thank the User Managed Access Work Group [5] for invaluable comments and feedback.

About the Author

Igor Zboran is a mechanical engineer by education with professional experience as a software engineer and solutions architect. He'd like to transform his knowledge into a useful system or service that people would love to use.

Igor received Ing. degree in Mechanical Engineering from the University of Žilina, Slovakia in 1988. After graduating, he worked in several small private companies as a software developer. From 2008 to 2009, he provided expert advice to Prague City Hall IT department, Czech Republic as an external consultant. He invented a new decentralized Identity-Based Privacy (IBP) trusted model built around OAuth2 and OpenID Connect standards. Igor is a strong proponent of open source software and open standards.

References

[1] User-Managed Access (UMA) 2.0 Grant for OAuth 2.0 Authorization https://docs.kantarainitiative.org/uma/wg/rec-oauth-uma-grant-2.0.html.
[2] Federated Authorization for User-Managed Access (UMA) 2.0 https://docs.kantarainitiative.org/uma/wg/rec-oauth-uma-federated-authz-2.0.html.
[3] Decentralized Identity-Based Access Control (DIBAC) proposal https://github.com/dibac/proposal.
[4] Next Generation Internet https://www.ngi.eu.
[5] WG - User Managed Access https://kantarainitiative.org/confluence/display/uma/Home.