INTERNET-DRAFT
<draft-aol-imx-00.txt>
Expires: December 15, 2000

E. Aoki
A. Wick
AOL

June 15, 2000

The IMX Architecture
Interoperability with America Online's Instant Messaging Services

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This memo provides information for the Internet community.

Copyright Notice

Abstract

The ability to exchange instant messages and presence information provides users with a powerful mechanism for communicating in real-time.

This document outlines an architecture for interoperability among Instant Messaging Systems which allows disparate systems to exchange messages and presence information while being relatively easy to implement and maintaining a high standard of security and scalability.

Table of Contents

1. Introduction

Today's instant messaging systems are typically comprised of a client, through which the end-user interacts, and servers which relay information between compatible clients. Tight integration between clients and servers allows instant messaging services to provide a secure, reliable channel through which authentication, presence, and messaging information is passed between users and the service.

As the number of instant messaging providers has grown, there is increased interest in enabling IM users to exchange presence and messaging information not only with users on their system, but with those on other systems as well Some vendors have responded by creating "multi-headed clients," clients which can simultaneously communicate with servers on disparate instant messaging systems.

Such clients achieve interoperability at a high price, however. Since each service has its own feature set, clients may advertise features that do not work across systems. Since each service implements its own security model, multi-headed clients must often resort to mechanisms that circumvent security or require the user to provide passwords to third parties. Inconsistent terms of service also make it difficult to enforce anti-spam measures or encourage equitable resource sharing. And vendors are forced to constantly upgrade clients to keep up with changes in features and services across the instant messaging universe.

An alternative approach is to provide a mechanism for the services themselves to interoperate in a peering arrangement, much like the Internet mail system works today. In such a system, the interaction between instant messaging clients and their associated servers would remain much as it is today, but servers could communicate with other servers to exchange presence information, messages, or other data. This approach helps to preserve existing models for security and allows Instant Messaging Service providers to manage client authentication, service policy, and privacy.

This document describes how America Online intends to develop such an architecture to allow its services to discover other servers (and be discovered by other servers), exchange data, and ensure security. It also describes implications that any architecture for interoperability may have for the spread of unsolicited instant messaging (spam).

2. Requirements

The authors set out to design a system that would be flexible, yet practical to implement. In that vein, many of our design goals, listed below, are the same as or similar to those specified in [RFC 2779], but with additional consideration paid to implementation issues.

The primary requirements were to design a system which:

1) is scalable to hundreds of millions of instant messaging users.

2) is scalable to hundreds of thousands (or perhaps millions) of individual instant messaging systems and domains, so every company or ISP could have its own instant messaging system.

3) allows existing instant messaging implementations to manage their own user namespace.

4) is self managed (like the Internet mail system) such that new servers and new systems could be added without administration

by some central authority and without manual administration by other service providers.

5) initially supports at least five main end-user features:

- Requesting/Renewing/Canceling presence subscriptions
- Sending/Receiving presence notifications (in response to a subscription)
- Routing and delivery of instant messages
- Retrieval of named user attributes (at minimum an "alias" and current presence state)
- Retrieval of named domain attributes

6) allows traffic between users in a single instant messaging system to stay within that system.

7) makes it possible to implement interoperability between instant messaging systems by adding gateways to existing systems without rearchitecting existing core systems.

8) is extensible, so new features can be added incrementally without requiring redesign and while allowing for backwards compatibility.

9) has a well thought-through security strategy such that:

- Messaging data or state can't be easily spoofed or replayed by a third party
- Messaging data or state can't be easily intercepted, hijacked, or stolen by a third party
- More advanced security measures such as end-to-end encryption or signing can be layered on top of the initial implementation, but are not required in the initial implementation.

10) easily supports clients that are inside of a common company firewall (e.g. incoming connections are often refused).

11) easily supports international usage.

12) leverages existing standards in as many places as practical.

3. Terminology

[RFC 2778] specifies a common terminology for the discussion of Internet Messaging and Presence Protocol information; however, that document defines terms which are considerably more granular than are required by this document. Accordingly, this document uses the following terms:

Aggregator - A special kind of Gateway, which services multiple domains and routes messages between other IMX Servers and Servers which service a particular domain. The protocol between the Aggregator and each Server is arbitrary. An example of an Aggregator might be a Gateway which acts as a front-end to multiple, privately-labeled Instant Messaging Services.

Attributes - metadata about an End User, such as a nickname or alias; or about a domain, such as a timeout value. Attributes consist of a key, which is scoped at a domain, and a value. It may be desirable to have certain attribute keys which are global to the IMX architecture and interpreted identically by all participating services.

Data - any of Instant Messages, Attributes, Notifications, Subscriptions, or requests for these that are exchanged between Servers.

End User - a human or other entity whose presence information is reflected by the service and who can send and receive Instant Messages through an Instant Messaging Service.

Instant Message - a short, real-time or near-real-time message which is sent between Instant Messaging clients. While this document does not prescribe a definition for "short," the intent is to prevent streams of arbitrary length from being sent as Instant Messages. This is consistent with the definition of an INSTANT MESSAGE in [RFC 2778].

Instant Messaging Client (or Client) - a User Agent which provides an End User with the ability to initiate and receive Instant Messages, and request Subscriptions and receive Notifications for Presence Information. This term is used in this document to encompass a SENDER, INSTANT INBOX, PRESENTITY, and WATCHER in [RFC 2778], and is roughly consistent with the generic description of an "instant messenger" in section 2.7 of that document.

Instant Messaging Gateway (or Gateway) - a special type of Instant Messaging Server which does not communicate directly with Clients but sits between Servers which service a particular domain and the world of other IMX servers. Gateways act essentially as routers, potentially performing protocol translation between an Instant Messaging Service's local protocols and the IMX protocol. An example of a Gateway would be a Server which routes messages to an already existing, private Instant Messaging Service.

Instant Messaging Server (or Server) - an entity which maintains Presence Subscriptions for and delivers Instant

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.