# IMPP Framework for Instant Message and Presence Services

Marko Buuri, *University of Helsinki*

***Abstract*—Free and public instant messaging services were awarded with instant success right from their launch in the mid-1990's. An important factor to their appeal along with the actual chatting functionality was the so called buddy list feature that displays online presence information of one's friends. But as there were no universal standard for instant message and online presence protocols all such services were incompatible with the others. Internet Engineering Task Force founded the Instant Messaging and Presence Protocol Working Group in 1998 to address the need for an open unified protocol. This paper discusses the works published by the Working Group and analyzes them from a number of aspects to information security.**

***Index Terms*—Instant Messaging, Instant Message and Presence Protocol, Presence Services, Security**

## I. Introduction

INSTANT messaging – sending simple messages delivered immediately to online recipients – has come to ten years of age as service and applications as commonly understood today. But arguably the first widely adopted instant messaging system in the internet scale was a product of the late 1980's, the Internet Relay Chat (IRC). IRC quickly became de facto protocol for online chat services on the internet after its release. That was at the same time when SMTP-based e-mail started to gain larger user base, but many felt it didn't quite meet the real time requirements for lightweight chatting.

The first modern instant message services were launched by America Online (AOL) and Mirabilis after the mid-1990s. Both of these services were free to the end users. The software clients, the AOL Instant Messenger and the ICQ, were easy to use and had features like the buddy list, an address book extended with capability of displaying online presence status of one's friends. These features appealed strongly to the masses and both services rapidly grew their user base. AOL later acquired Mirabilis, and Yahoo! and Microsoft entered the market with their similar services in 1999. The active user base of then available services was already in tens of millions. The huge success even fueled visionaries to herald inescapable adoption of instant message services as a business communications tool.

All the free instant message services offered by commercial corporations have traditionally been based on closed proprietary protocols. This made the services incompatible with each other and left the users to take their pick from a number of offerings. For many users this translated to need to use several services in parallel to reach their entire network of online friends. Another problem for users were the security problems discovered from those proprietary protocols and client applications over the years. Early versions of proprietary instant messaging services gave, for example, no tools to prevent unsolicited messaging from previously unknown users.

In response to the growing popularity of instant message services, the Internet Engineering Task Force (IETF) founded the Instant Messaging and Presence Protocol Working Group (IMPP WG) to define instant messaging and online presence services and develop an open protocol for them. This paper introduces the IMPP framework and the requirements it places on behavior of instant message and presence services and protocols. The paper also analyzes those requirements from some commonly acknowledged aspects of information security, and compared them to the security mechanisms available to and commonly used for e-mail.

The following chapter discusses the history of IMPP Working Group as is needed for understanding the Request for Comments (RFC) documents it has published. Chapter III introduces the basic model and requirements of the IMPP framework. Chapter IV presents the interoperability requirements for instant message and presence services, and chapter V displays security mechanisms and decisions in the IMPP service model.

## II. IMPP Working Group's History

The work at the IETF was initiated when Microsoft submitted a draft of its Rendezvous Protocol for consideration in 1997. That started a discussion that led to the IMPP Working Group being formed in early 1998 [1]. According to the original IMPP WG's charter the working group was founded to "define protocols and data formats necessary to build an internet-scale end-user presence awareness, notification and instant messaging system [2]." The WG started its work on formalizing abstract instant message and online presence services, and behavior of the instant message and presence protocols. These foundational requirements were

Paper submitted September 16th, 2005. *Seminar on Instant Messaging and Presence Architectures in the Internet*, 2005 Department of Computer Science, University of Helsinki, Finland.

M. Buuri is a M.Sc. student of Computer Science at the University of Helsinki, Finland (e-mail: buuri@cs.helsinki.fi).

collected and published in two RFCs during 2000 [3, 4].

After the requirements for services were completed, the WG was set to develop actual protocol specifications. To get started with this, the WG requested initial suggestions from its members. There were several suitable instant message and presence protocols introduced to the WG each incompatible with the others and none superior in the face of the agreed requirements. The members of the WG found themselves unable to settle on any of the suggested approaches, and the WG split into three competing factions all developing similar protocols on their own [5, 6].

Some time later these factions were called together by the IETF to resolve the situation. The groups were set to find out if there was any common ground on the core functionality that could be standardized to allow some kind of interoperability between the different designs. That work resulted in common service and protocol profile requirements being published in 2004, along with common message formats for instant message and presence information. Those documents define common service behavior for instant message and presence services in minimum detail needed to build service gateways between incompatible IMPP-based designs [7–11].

New independent working groups developing IMXP (later APEX), PRIM (merged from PePP, PITP/IMTP, OneIM and SIMP) and SIMPLE protocols were founded to carry on developing the specification work. These working groups all declared their work to be compatible with the general IMPP design and jointly agreed interoperability requirements. Another somewhat IMPP-compatible XMPP protocol) specification working group (also known as Jabber) was founded some time later. The new WGs continued to work on their own directions and work at the IMPP Working Group was consequently terminated.

## III. IMPP Service Framework and requirements

The IMPP Working Group published two documents that specified abstract services, terminology and entities involved in an instant messaging and presence framework. The documents identify all the logically distinct components required in such services and define their roles in relation to each other, and set out the requirements that were meant as basis for the IMPP protocol specification [3, 4].

This chapter discusses the scope of the IMPP definitions and requirements from views of the formal framework, the service behaviors and the protocols.

### *A. Service Framework*

The service framework formalized by the IMPP Working Group considers presence service and instant message service as logically distinct services that must be able exist independently from each other. These services have their own purposes, requirements and protocols. There are several logical components to the both service frameworks.

The presence service framework consists of presentities (presence entities), watchers and a mediating presence service. These components are usually located on separate systems although that is not necessary by the requirements. Presentities produce presence information and provide it to watchers through a presence service. Presence information is an online availability statement pertaining to the principal controlling the presentity (that is, the user). A principal assigns her presentity to distribute her presence information to other principals using a presence user agent available to her. The other principals will use their watcher user agents to examine the presence information received by their watchers.

The presence service holds a copy of a recent version of each presentity's presence information and is therefore able to deliver that information to appropriate watchers even when a particular presentity is out of contact. Watchers can either actively fetch particular presence information from a presence service or subscribe to receive notifications of any future changes in the presence information of a given presentity. The subscriptions are handled by the presence service. A presence protocol is used to carry all presence information from presentities to service and from service to watchers.

The instant messaging service framework consists of senders, instant inboxes and a mediating instant message service. Like with the presence service model, these components are normally in practice located on separate systems although not required to. The principals create and send instant messages using their senders that they control with their sender user agents. Senders hand over instant messages to an instant message service for delivery. Instant messages are targeted to specific instant inboxes and an instant message service will try to locate the right recipients. An instant message protocol is used to carry instant messages.

A presence service or an instant message service can consist of from none to an arbitrary number of distinct server systems. If no servers – and thus service – are present, presence information and instant messages can be delivered directly from their sources to the targeted watchers or instant inboxes. If a service consists of more than one system, they together form a distributed system that acts as a single service when accessed from outside. The IMPP framework does not discuss how a distributed instant message or presence service could be implemented or the problems essential to distributed message services solved. The view is on service behavior on the interfaces while the internals of the services are left for consideration of implementers.

Typical instant message software combines many or all of the client end components into one client application – commonly called instant messaging clients or instant messengers – that can be used to sending and receiving messages and sharing presence information. Figure 1 illustrates how the logical components of the IMPP framework are located in typical client and service implementation. However, there are other ways to implement IMPP-compliant systems. For example, a three-tier design could be implemented using light version user agents implemented as web-downloadable software components, which could communicate to their main components in client-server manner.

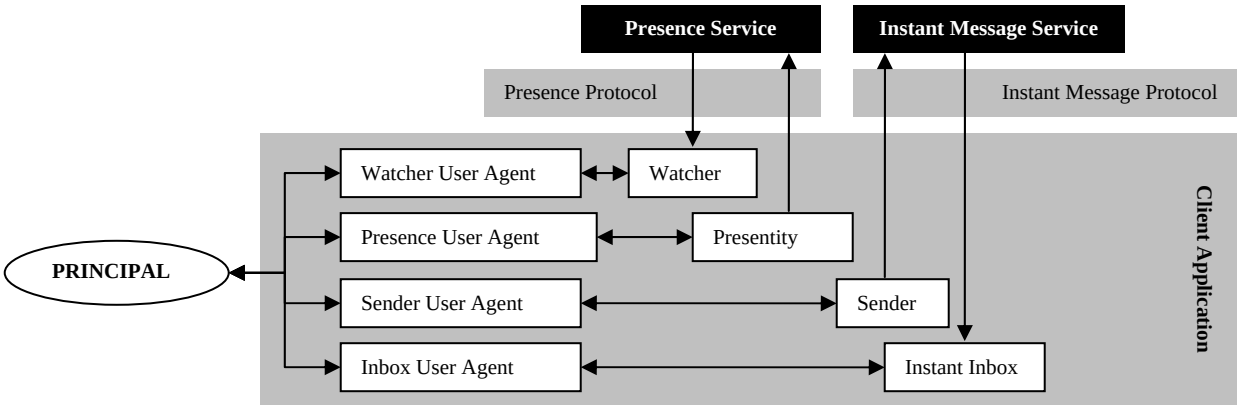


Figure 1: Entities of the IMPP framework placed in a typical client implementation.

### *B. Protocols*

Presence and instant messaging services use their own protocols. The IMPP framework places several restrictions and requirements on both these protocols.

There is no distinct requirement in the IMPP model for supporting the Internet Protocol (IP) or any other named network protocol. But the authors predicted these types of services would become common among mobile users, and included a need to support unreliable high-latency low-bandwidth wireless access methods.

The protocols are required not to assume presence of both services in neither the client nor the service end, thus they must support separate identifiers to be used for addressing senders and instant inboxes, and addressing presentities and watchers. Further requirements include protocol level support for an arbitrary number of separate naming domains. The protocols are required to support millions of such domains each containing millions of distinct entities. This, however, is not a performance requirement for actual service implementations.

Both protocols are required to support proxies that can be used to bridge connections or to cache for performance. From the framework point of view these proxies are parts of the services. Proxies may be needed, for example, for bridging the instant message protocol from an IP network to lightweight mobile devices situated in non-IP networks. Proxies are required to support essential mechanisms for using presence and instant message services through them.

A presence protocol defines the interaction and how presence information is carried between presence service, presentities and watchers. Fetchers – watchers that retrieve presence information without subscription – may call for presence information pertaining to a particular principal from presence service, and subscribers can request a subscription to such, even when the presentity providing the information is out of contact. Presentitys and subscribers are, however, provided with information whether the other party is out of contact at that time. When an entity makes a change to the presence information it is publishing, it submits the change to the presence service which consequently notifies all watchers with a valid subscription to that information with the change. The notifications are trasmitted in rapid fashion, in speed comparable to sending an instant message to a subscriber.

An instant message protocol defines the interaction and how instant messages are carried between a instant message service, senders and instant inboxes. Senders deliver instant messages to an instant message service upon sending, and the service transmits the message to the targeted instant inboxes. It is not stated in the requirements of the IMPP framework whether the instant message service can, may or has to implement any kind of offline messaging.

There are additional security and privacy requirements to both type of protocols placed in the IMPP model. Those requirements are discussed in chapter VII. Data format requirements are detailed in chapter V.

## IV. Interoperability Requirements

The IMPP Working Group released several documents addressing interoperability issues that were present in competing IMPP-derived instant message and presence protocols. The issues existed due to the initial requirements that left room for different interpretation. Those interoperability documents define common service behavior and semantics needed to make gateways between the kinds of presence and instant message services implemented. The documents contain the minimum set of functionality all IMPP compliant services and protocols must implement.

Common Profile for Presence (CPP) specification defines the behavior of subscription and notification operations for presence services [7]. The required attributes for these operations are defined, but the specific data level protocols are left for consideration of designers and implementers. The operations serve the purpose and operations described in the IMPP requirements.

A subscription operation is initiated by a watcher towards a presentity. The watchers delivers the subscription message to the presence service. The message carries desired maximum subscription duration along with identities of the presentity and the watcher. A special type of subscription is that with requested duration of zero which is used by fetchers and

LISTING I
PRESENCE INFORMATION DATA FORMAT EXAMPLE

```
<?xml version="1.0" encoding="UTF-8"?>
<impp:presence
      xmlns="urn:ietf:params:xml:ns:pidf"
      xmlns:ext="urn:example-com:pidf-status-type"
      entity="pres:alice@example.com">
   <tuple id="mypres">
      <status>
         <basic>open</impp:basic>
         <ext:extdstatus>Busy</ext:extdstatus>
      </status>
      <contact priority="0.8">
         im:alice@example.com
      </contact>
      <contact priority="0.3">
         tel:+3585012345678
      </contact>
      <note xml:lang="en">
         Busy until 11 AM. Urgent messaging only.
      </note>
      <note xml:lang="fi">
         Kiirettä pitää kello 11 asti.
      </note>
      <timestamp>2005-08-14T09:14:37Z</timestamp>
   </tuple>
</presence>
```

pollers to retrieve presence information. That operation is also used to unsubscribe prematurely.

The service acknowledges the subscription request with a response carrying subscription status information, and in case of success, consequently invokes a notification operation. The notification message carries identities of presentity and watcher along with the presence information pertaining to the presentity. Subsequent notifications are sent when ever there is a change in the presence information of that presentity during the period of the subscription. The subscription operation will fail, if the identity of the presentity is unknown to the presence service or access controls prevent the operation. The service can make the subscription acceptance decision without contacting the presentity as the access control values are cached in the service.

Common Profile for Instant Messaging (CPIM) specification defines behavior of instant message services to message transmission operations [8]. A sending operation is used by a sender to transmit an instant message towards an instant inbox. The sender hands the message over to an instant message service which accepts the message and tries to deliver it to the destined instant inbox. Then the service initiates a response operation indicating whether sending was a success. A sending operation fails if the hop count of the message has been depleted, the instant message service cannot resolve the destination identity, the destination instant inbox is out of contact, or access control prevents the operation.

## V. COMMON DATA FORMATS

The IMPP Working Group has produced common presence and instant message formats that all entities can produce and consume. They are titled Presence Information Data Format (PIDF) and Common Profile for Instant Messaging (CPIM) data format [9, 10].

TABLE I
THE HEADERS IN THE CPIM INSTANT MESSAGE DATA FORMAT

| Header | Meaning | Example Value |
|---|---|---|
| From | Sender | Galileo <im:galileo@example.com> |
| To | Recipients | Leo <im:leonardo@example.com> |
| cc | Courtesy recipients | Gottfried <im:leibniz@example.com> |
| DateTime | Moment of sending | 2005-09-14T11:11:04+02:00 |
| Subject | Description of the message | I love rabbits |
| NS | Extension declaration | Ext <http://example.com/extensions> |
| Require | Required extension for application | Ext.KatakanaRenderer |

Presence information exchanged and carried by a presence protocol consist of elements called presence tuples that envelop presence information. Tuples are considered the smallest units of presence information. They consist of a status information, and optionally communication address and other presence markup. Status information is defined to have at least mutually-exclusive “open” and “closed” states that are used to communicate whether instant messages will be accepted at the defined communication address. This feature is useful to implement when a presence service is developed jointly with an instant message service. Status may be extended with an arbitrary number of other values. For example, while a principal is online and “open”, she might want to extend the presence information with a note that she is not immediately available for chatting. Communication address present in a tuple consists of communication means and a contact address. More than one tuple can be grouped together to presence information. This is desired, for example, when presence information for single principal is available from multiple sources.

PIDF is a common XML-based data format for representing presence information with the MIME content type “application/pidf+xml”. The format supports a minimal compatible feature set mandatory by the IMPP requirements but also contains additional features. These extensions include presentity’s possibility to include relative priority preferences over multiple contact means, and creation timestamps in presence information. One PIDF document can contain presence information in several presence tuples pertaining to a single presentity. A tuple contains a varying number of XML elements. It is possible to use custom XML namespaces to extend the status information beyond the “open” and “closed” modes.

**Listing I** contains a well-formed XML document example created according to the PIDF spesification. In the example the user controlling presentity "pres:alice@example.com" declares herself busy until certain time but makes herself available in case of emergencies. She uses two languages to communicate her message and provides two prioritized contact means.

Instant Messages exchanged and carried by instant message protocol are formatted according to CPIM data format in

LISTING II
INSTANT MESSAGE DATA FORMAT EXAMPLE

```
Content-type: Message/CPIM

From: MacGyver <im:macgyver@phoenixfoundation.org>
To: Pete <im:pete@phoenixfoundation.org>
To: Jack <im:jack@jackdalton.net>
DateTime: 1987-12-26T17:33:01Z
Subject: Urgent!
NS: Feats <mid:imfeats@phoenixfoundation.org>
Feats.MessageOption: Urgent-Flashing

Content-type: text/xml; charset=utf-8
Content-ID <88243a@phoenixfoundation.org>

<body>
Quick, send me some more bubble gum!
</body>
```

interoperability uses. The message data format is UTF-8 character encoded. Instant messages are organized in a header and body sections. A message body contains MIME-encapsulated message content. The MIME content type for instant messages is “Message/CPIM”. Headers contain the key-value paired meta-data relevant to end-to-end deliver of the messages. The defined header keys include “From”, “To, “cc” and “Subject” which have essentially the same meaning as with e-mail messages. “DateTime” marks the moment of sending the message in ISO 8601 representation [11]. Zero or more “NS” values point to namespace extensions for the header. Applications may insert non-standard header information through extensions. The locations of the extensions are defined as Universal Resource Identifiers (URI). “Require” keys designate the extensions that must be available to the receiving application. **Table I** contains a summary the headers keys and presents examples of possible values.

**Listing II** contains an example instant message created according to CPIM message format specification. In the example the user controlling the sender “im:macgyver@phoenixfoundation.org” has created a message to recipients “im:pete@phoenixfoundation.org” and “im:jack@jackdalton.net”. The message requires a special urgent message rendering feature from the client application. The body of the message is an UTF-8 character encoded XML document.

## VI. SERVICE DISCOVERY

Operations included in the instant message and presence protocols contain source and destination identifiers. Service discovery for target identifiers is handled with a solution based on the internet’s Domain Name System (DNS) [12]. Internet Assigned Numbers Authority (IANA) has assigned a special URI schemes for identifying presentities, watchers, senders and instant inboxes. “Pres” URI identities are used for presentities and watchers. An example of such is “pres:alice@example.com”. “Im” URI identities are used for senders and instant inboxes. An example of such is “im:bob@example.com”. The identifiers resemble those used in e-mail systems consisting of a user name and a domain.

TABLE II
EXAMPLES OF DNS SERVICE RECORDS FOR PRESENCE AND INSTANT MESSAGE SERVICES

| Service Type | SRV Attributes | Service Host |
|---|---|---|
| _pres._pp.example.com | SRV 0 5 1111 | imservice.example.com. |
| _im._iprot.example.com | SRV 0 5 1112 | imservice.example.com. |

The user name is unambiguous within the specified domain or service. For example, identities “cecil@example.com” and “cecil@company.com” both refer to users named “cecil” in separate services. There is no method provided for identifying whether such two “cecils” are in fact the same principal.

The domain part of an identifier denotes the service. The domain name is interpreted as a fully qualified domain name (FQDN). FQDN is a unique root domain name resolvable through the public global DNS service. The domain has an arbitrary number of servers running presence and instant message services. The host names of gateways to these services are advertised as service location (SRV) resource records among domain’s other DNS information.

The name of a service host or gateway is created by starting with “_im”, and appending the protocol name and the domain name from the target identifier. The service host is then resolved in normal DNS lookup fashion. For example, when an instant message service wants to deliver an instant message to instant inbox “im:donald@example.com” using instant message protocol titled “IProt” registered with IANA, the service must resolve service location for “_im._iprot.example.com.”. Notice the trailing dot that makes it a FQDN. The lookup results in a DNS name or an IP address of the service gateway which consequently can be contacted for the delivery.

**Table II** has example service resource records for presence and instant message services on host imservice.example.com accessible with protocols “pp” and “iprot” on ports 1111 and 1112 respectively.

## VII. SECURITY AND PRIVACY

People have learned to be concerned of their privacy when they access and use online services. Any technology that could provide other people with information of one’s online presence or activities tends to raise concerns from aware users. That is also the case with computerized messaging and especially with providing online presence information. These types of privacy concerns can be witnessed in countries like Finland where the legislation places arguably stricter privacy laws controlling transmission and storage of electronic communications as to handling of traditional mail.

To protect the privacy of the public the authors of the IMPP requirements have discussed and included explicit mechanisms that give users ways to protect their presence information and messaging. These mandatory mechanisms are necessary as instant messages and presence information are likely to be carried over untrustworthy services, systems and networks such as the internet. This chapter analyzes the

requirements placed in the IMPP model from different classification aspects to information security mechanisms and threats that they thwart [13, 14]. There is no universally agreed classification for such mechanisms so a suitable one for purposes of this paper has been chosen. Instant messages and presence information send and received by users, delivered through services, are below jointly considered as sensitive information in need for protection. The end of this chapter compares differences in between security mechanisms available for e-mail and those in the IMPP framework.

### A. *Confidentiality Mechanisms*

Confidentiality mechanisms, as meant here, are used to protect the information while being transmitted between end users. The threat that these mechanisms counter is unauthorized interception of information. The interception can be executed anywhere on the information's path between the communicating end users. Common approaches used to implement confidentiality mechanisms are access controls and encryption.

An example of confidentiality breach is when a systems administrator or an operator with legitimate access to a system hosting instant message or presence service misuses her position of trust and obtains sensitive information pertaining to users of the service. Another example is a malign outsider who gains access to a network level device and coincidentally or deliberately also to the information routed through that device.

The access controls available in the service protocols of the IMPP framework are discussed later in this chapter.

When considering security of messaging and online presence applications where mediating services are used, transport level security solutions should be left out of consideration as they will not protect from confidentiality attacks taking place within the systems that the service consists of. The CPIM agreement accomplishes end-to-end security with mandatory support for Secure Multipurpose Internet Mail Extensions (S/MIME) specification originally developed for e-mail [16]. Possibility to use OpenPGP for MIME security is also mentioned but not required [17]. Use of S/MIME or OpenPGP adds end-to-end cryptographic security to instant message and presence services. Consequently all implementations compatible to CPIM must support several digest and encryption algorithms required by S/MIME.

The body of an instant message or PIDF presence content can be secured using S/MIME's EnvelopedData content type for encryption. The SignedData content type is used for digital cryptographic signatures. Use of Advanced Encryption Standard (AES) cryptographic algorithm is recommended in CPIM specification, but not required there or in current version of S/MIME.

Support for end-to-end message security effectively counters threats against confidentiality when untrustworthy networks and services are used for transmitting. Interception attacks are made useless as bodies of messages transferred cannot be interpreted without access to the encryption keys used. However, the knowledge of messages being send and received remains unprotected. Message header information cannot be encrypted end-to-end when an external delivery service is used. Since knowledge of messaging taking place between given parties is in itself valuable information to some unauthorized third party, the substance of this particular matter should be considered.

### B. *Authentication Mechanisms*

Authentication mechanisms, as meant here, ensure that the origins of all information are correctly identified. As with any messaging system an instant message service could hardly be useful without proper authentication mechanisms. Lack of such mechanisms would likely lead to message fabrication in extent the common e-mail systems suffer from today. Fabrication attacks are executed by sending counterfeit communications that appear to originate from legitimate sources.

An example of an attack against authentication is when a user sends an instant message to the accounting department in the name of the company's CFO requesting a money transfer to her personal account.

The IMPP specification counters threats against authentication by requiring the protocols to implement necessary means to identify valid messages from fraudulent ones do exist for all information transmitted. The digital signature means provided with S/MIME do solve this problem partially as that the digital signature mechanism will identify who created the presence information or the instant message. However, the digital signature algorithms in S/MIME do not provide means to cryptographically secure who the intended receiver of the information is [18]. This enables forwarding of messages as a recipient cannot verify of whether the message was originally intended to her.

### C. *Integrity Mechanisms*

Integrity mechanisms provide functionality that will reveal any tampering with information being transmitted. The threats that these mechanisms thwart is modification of the transmitted information by an unauthorized party, and replaying of messages. An integrity mechanism will alert if a violation of integrity is detected, but it doesn't need to know how to fix it. A common approach to protect integrity of transmitted information today is to use digital signature algorithms. Message replaying is countered with message identifiers, timestamps or counters in protocols.

An example of a modification attack is when a malicious user interrupts her boss' instant message to human resources department concerning her raise, alters the figures, and resends the e-mail.

Instant messaging and presence protocols are by IMPP service model required to provide means to ensure integrity of all messaging as well as protection against malicious replaying of valid messaging. Using just digital signature algorithms available in S/MIME the integrity of all transmitted information is well monitored. The replaying of presence

information can be observed if the optional timestamp element was used. With the instant messages the "Content-ID" key provides a way to counter replying. In both cases altering the content will also break the cryptographic signature which will not go unnoticed from the end user.

### D. Nonrepudiation Mechanisms

Nonrepudiation mechanisms determine that neither the sender nor the receiver of information can deny that the transmission of information took place. There are two distinct threats to be addressed: repudiation of origin and denial of receipt. Both of these threats may be significant when messaging systems used in business or governmental institutions are considered. A common solution against repudiation of origin is to require use of digital signature algorithms. There are no widely adopted mechanisms to counter denial of receipt threat today.

An example of attack against nonrepudiation is when a person sends an instant message order for one-time password to an online service. She receives the password and uses the service. Later she denies having sent the order and refuses to pay for it.

In IMPP requirements protocols must provide the receiver means to verify the sender. The repudiation of origin is in a way thwarted using suitable digital signature algorithms available in S/MIME. If asymmetric encryption was not used, the creator can still deny that the message was originally intended to some recipient. There are no requirements or suggestions in the model that would prevent the denial of receipt.

### E. Access Control Mechanisms

Access control mechanisms, as meant here, provide ability to place constraints on who can access particular information from given services. These mechanisms rely on identifying the principal and therefore need authentication mechanisms to be available. The main threat against access control is that an authenticated user can bypass her constraints and access information beyond expectations.

An example of attack against access control is when a user has an account to a system, but is able to escalate her privileges through security vulnerability in the system to gain information not meant for her.

The IMPP model contains two types of rule-based access control mechanisms used to limit access to information. Access rules are used to limit how presence service makes presence information available to watchers. By taking use of these rules a user can set the watchers allowed to observe and subscribe to her presentity's presence information. Visibility rules limit how watcher information is made available to watchers. Many users of an online presence system are interested in who is fetching, polling to or subscribing to their presence information. Users take advantage of visibility rules by defining access to watcher information of their watchers. The types of access control as defined in the model are summarized in **Table III**. The table also includes delivery rules discussed below.

TABLE III
TYPES OF ACCESS CONTROL DEFINED IN IMPP MODEL

| Rule Type | Rule Function |
|---|---|
| Access Rule | Constraints on how a presence service makes presence information available to watcher. |
| Delivery Rule | Constraints on how an instant message service delivers received instant messages to instant inboxes. |
| Visibility Rule | Constraints on how a presence service makes watcher information available to watcher. |

### F. Availability Mechanisms

Availability mechanisms prevent attacks that are executed to cause loss or reduction in availability of select services and information within. Two different approaches to such attacks are interruption of transmission and denial of service attack.

An example of interruption attack is when a malicious person gains access to a network device and reconfigures it to fail for select services or users. Another type of attacking availability is to aim so much service requests to a service that it would be required to reject legitimate service requests under the amount.

Requirements of the IMPP framework include delivery rules used to limit how instant messages are delivered to instant inboxes. Users are expected to take use of these to filter out messaging from unauthorized senders to prevent vast amounts of unsolicited messaging such as present in e-mail services. This resolves the denial of service where individuals are targeted with so much messaging that the substantive information would effectively be buried among them. Even if the system did cope with the load, the attack on availability of the legitimate messages would be feasible without delivery rules.

There are no requirements for mechanisms in IMPP model that counter interruption of communication attacks.

### G. Security Mechanisms in Comparison to E-mail

To get an idea on how well IMPP requirements address given security threats, they can be compared to e-mail security. There are many similarities from information security point of view between commonly used e-mail services based on Simple Mail Transport Protocol (SMTP) and abstract IMPP-defined instant message and presence services. In both cases message body confidentiality and integrity can be achieved with S/MIME or PGP, but the headers will remain subject to confidentiality and integrity threats. Transport level security setups are commonly used between e-mail clients and servers (also between servers in different domains), and can be used by IMPP service model as well.

Use of S/MIME and PGP implement asymmetric cryptographic mechanisms that counter authentication threats effectively, when used correctly. There are known ways to use these technologies in a wrong manner that adds lesser value than perhaps expected [18]. Also the issues of key management that trouble all public key cryptosystems are

naturally present in these types of application as well. Common e-mail services suffer from the same denial of receipt problems as services by the IMPP framework.

One security aspect where IMPP framework surpasses SMTP e-mail is availability. The ability to deny messaging from previously unknown senders is functionality that makes e-mail anti-spam services useless. This feature adds enormous value to end users and is implemented in most of the publicly available instant message services today.

## VIII. Conclusion

The IMPP Working Group of IETF never produced the protocol specifications for instant message and presence services it intended. But it did produce the first public formalized framework and requirements that were influential to designs of online presence and messaging protocols and architectures that followed.

The authors of IMPP requirements predicted many foreseeable security threats that such services are likely to be subjected to and some went unnoticed. When compared to common e-mail services used today the differences are less than striking. But with the delivery rules it provides for users to set, the IMPP instant message service effectively obliterates the one most annoying problem e-mail users are faced with, unsolicited messaging.

The three children of IMPP Working Group carried on with designing their own protocols. Application Exchange (APEX) Working Group started from SMTP e-mail paradigm as basis framework for developing an instant message infrastructure. They have published three RFCs describing their design. The PRIM Working Group chose a lightweight design strategy building their protocol operations on top of TCP. They are yet to publish any specifications. SIMPLE and XMPP technologies are still under active work. The work on the XMPP core functionality has been concluded in IETF after total of four RFCs. The on-going work is targeted on extensions to the protocol. SIMPLE Working Group has today published four RFCs and several more Internet Drafts. The last two technologies are the best candidates today for an instant message and presence service technology standard in the internet.

## Acknowledgment

The author thanks Mark Day for insight he provided on the history of the IMPP Working Group.

## References

[1] Y. Kohda, H. Sugano, and S. Okuyama, “IMPP: A New Instant Messaging Standard and Its Impact on Internet Business,” in *Fujitsu Scientific and Technical Journal*, 36(2), Dec. 2000, pp. 147–153. Available: http://magazine.fujitsu.com/us/vol36-2/paper06.pdf

[2] IMPP Working Group, “Instant Messaging and Presence Protocol (IMPP) Charter,” Internet Engineering Task Force, 1998. Available: http://www.ietf.org/html.charters/OLD/impp-charter.html

[3] M. Day, J. Rosenberg, and H. Sugano, “A Model for Presence and Instant Messaging,” Request for Comments 2778, Internet Engineering Task Force, Feb. 2000. Available: http://www.ietf.org/rfc/rfc2778.txt

[4] S. Aggarwal, M. Day, and J. Vincent, “Instant Messaging / Presence Protocol Requirements,” Request for Comments 2779, Internet Engineering Task Force, Feb. 2000. Available: http://www.ietf.org/rfc/rfc2779.txt

[5] D. Atkins and M. Day, “IMPP Working Group History and De Facto Charter,” Internet Draft, Internet Society, 2003. Available: http://www.watersprings.org/pub/id/draft-day-atkins-impp-defacto-00.txt

[6] D. Crocker, A. Diacakis, C. Huitema, G. Klyne, F. Mazzoldi, M. T. Rose, J. Rosenberg, R. Sparks, and H. Sugano, “A Framework for Moving IMPP Forward ,“ Internet Draft, Internet Engineering Task Force, Aug. 2000. Available: http://www.iptel.org/info/players/ietf/presence/outdated/draft-rosenberg-impp-differences-00.txt

[7] J. Peterson, “Common Profile for Presence (CPP),” Request for Comments 3859, Internet Engineering Task Force, Aug. 2004. Available: http://www.ietf.org/rfc/rfc3859.txt

[8] J. Peterson, “Common Profile for Instant Messaging (CPIM),” Request for Comments 3860, Internet Engineering Task Force Aug. 2004. Available: http://www.ietf.org/rfc/rfc3860.txt

[9] A. Bateman, W. Carr, S. Fujimoto, G. Klyne , J. Peterson, and H. Sugano, “Presence Information Data Format (PIDF): Message Format,” Request for Comments 3863, Internet Engineering Task Force, Aug. 2004. Available: http://www.ietf.org/rfc/rfc3863.txt

[10] D.Atkins and G. Klyne, “Common Presence and Instant Messaging (CPIM): Message Format,” Request for Comments 3862, Internet Engineering Task Force, Aug. 2004. Available: http://www.ietf.org/rfc/rfc3862.txt

[11] G. Klyne and C. Newman, “Date and Time on the Internet: Timestamps,” Request for Comments 3339, Internet Engineering Task Force, Jul. 2002. Available: http://www.ietf.org/rfc/rfc3339.txt

[12] J. Peterson, “Address Resolution for Instant Messaging and Presence,” Request for Comments 3861, Internet Engineering Task Force, Aug. 2004. Available: http://www.ietf.org/rfc/rfc3861.txt

[13] J. Hildebrand, “Nine IM Accounts and Counting,” in *ACM Queue*, 1(8), Nov. 2003, pp. 45–50. Available: http://www.acmqueue.com/modules.php?name=Content&pa=showpage&pid=93

[14] W. Stallings, “Cryptography and Network Security: Principles and Practice – 2nd Ed.,” Prentice-Hall, Inc., 1999.

[15] M. Bishop, “Computer Security: Art and Science,” Pearson Education, Inc., 2003.

[16] B. Ramsdell, “S/MIME version 3 message specification,” Request for Comments 2633, Internet Engineering Task Force, Jun. 1999. Available: http://www.ietf.org/rfc/rfc2633.txt

[17] M. Elkins, D. Del Torto, R. Levien, and T. Roessler, “MIME security with OpenPGP,” Request for Comments 3156, Internet Engineering Task Force, Aug. 2001. Available: http://www.ietf.org/rfc/rfc3156.txt

[18] D. T. Davis, “Defective Sign & Encrypt in S/MIME, PKCS#7, MOSS, PEM, PGP, and XML,” in *Proceedings of the USENIX Technical Conference*, Jun. 2001, pp. 65-78. Available: http://world.std.com/~dtd/sign_encrypt/sign_encrypt7.PDF