Open Pluggable Edge Services
Internet-Draft
Expires: June 18, 2007

M. Stecher
Secure Computing

December 15, 2006

Integrity, privacy and security in OPES for SMTP
draft-ietf-opes-smtp-security-02

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

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 Internet-Draft will expire on June 18, 2007.

Copyright Notice

Abstract

The Open Pluggable Edge Services (OPES) framework is application agnostic. Application specific adaptations extend that framework. Previous work has focussed on HTTP and work for SMTP is in progress. These protocols differ fundamentally in the way data flows and it turns out that existing OPES requirements and IAB considerations for OPES need to be reviewed with regards to how well they fit for SMTP adaptation. This document analysis aspects about the integrity of SMTP and mail message adaptation by OPES systems and privacy and security issues when the OPES framework is adapted to SMTP and lists requirements that must be considered when creating the "SMTP adaptation with OPES" document.

Table of Contents

1. Terminology 3
2. Introduction 4
2.1 Differences between unidirectional and bidirectional application protocols 4
2.2 Non-standardized SMTP adaptations at SMTP gateways 4
2.3 Non-OPES issues of SMTP 5
2.4 Opportunities of OPES/SMTP to address some issues 5
2.5 Limitations of OPES in regards to fixing SMTP issues . . . 5
3. Integrity, privacy and security considerations 7
3.1 Tracing info in OPES/SMTP 7
3.2 Bypass in OPES/SMTP 7
3.3 Compatibility with Cryptographic Protection Mechanisms . . 8
4. Protocol requirements for OPES/SMTP 10
5. IAB Considerations for OPES/SMTP 11
5.1 IAB Consideration (2.1) One-Party Consent 11
5.2 IAB Consideration (2.2) IP-Layer Communications 11
5.3 IAB Consideration (3.1) Notification 11
5.4 IAB Consideration (3.2) Notification 11
5.5 IAB Consideration (3.3) Non-Blocking 12
5.6 IAB Consideration Application Layer Addresses (4.x) . . . 12
5.7 IAB Consideration (5.1) Privacy 12
5.8 IAB Consideration Encryption 13
6. Security Considerations 14
7. References 15
7.1 Normative References 15
7.2 Informative References 15
Author's Address 16
A. Acknowledgements 17
Intellectual Property and Copyright Statements 18

1. Terminology

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [1]. When used with the normative meanings, these keywords will be all uppercase. Occurrences of these words in lowercase comprise normal prose usage, with no normative implications.

2. Introduction

Because OPES is a protocol that is built over application layer transports, its security may depend on the specifics of the transport. OPES designs are guided by the IAB considerations for OPES document [2], and those considerations are revisited here in the context of the SMTP protocol.

2.1 Differences between unidirectional and bidirectional application protocols

The IAB listed considerations for Open Pluggable Edge Services (OPES) in [2] and OPES treatment of those considerations has been discussed in [3]. Both documents make use of HTTP as an example for the underlying protocol in OPES flows, and focus on web protocols that have requests and responses in the classic form (client sends a request to a server that replies with a response of the same protocol within a single protocol transaction).

RFC 3914 [3] already indicates that other protocols may not fit in this context, for example in section 5.3: "Moreover, some application protocols may not have explicit responses...".

When using SMTP there are still client and server applications and requests and responses handled within SMTP, but email messages are sent by the data provider to the recipients (data consumers) without a previous request; on that abstraction layer, email delivery via SMTP is a unidirectional process and different from the previously handled web protocols such as HTTP. For example: Bypass has been defined for OPES so far by allowing the data consumer to request an OPES bypass by adding information to the application protocol request; the OPES system can then react on the bypass request in both the application request and response. For SMTP, the data consumer (email recipient) cannot request in-band that the OPES bypass handling of his/her messages.

The IAB considerations need to be revisited and special requirements may be needed for OPES handling of SMTP.

2.2 Non-standardized SMTP adaptations at SMTP gateways

A large number of email filters are deployed at SMTP gateways today; in fact all usecases listed in "OPES SMTP Use Cases" [6] are already deployed, often in non standardized ways. This opens a number of integrity, privacy and security concerns that are not addressed, and SMTP itself does not provide effective measures to detect and defend against compromised implementations.

OPES will most likely not be able to solve these issues completely, but at least might be able to improve the situaton to some extent.

2.3 Non-OPES issues of SMTP

The SMTP specifications [4] require that NDRs (Non Delivery Reports) be sent to the originator of an undeliverable mail that has been accepted by an SMTP server. But it has become common practice for some sorts of mail (spam, worms) to be silently dropped without sending an NDR, a violation of the MUST statement of SMTP (see section 3.7 of [4]). While the user of a web protocol notices if a resource cannot be fetched, neither the email sender nor email recipient may notice that an email was not delivered. These kind of issues already exist and are not introduced by OPES.

2.4 Opportunities of OPES/SMTP to address some issues

Adding SMTP adaptations with OPES allows us to define a standardized way for SMTP gateway filtering, to offload filtering services to callout servers and address a number of the integrity, privacy and security issues. OPES offers methods to add OPES tracing information and to request bypass of filtering, and by that can make email gateway filtering a more reliable and standardized function. But OPES won't make email delivery via SMTP a reliable communication.

2.5 Limitations of OPES in regards to fixing SMTP issues

The biggest concerns when adding OPES services to a network flow are that compromised, misconfigured or faulty OPES systems may change messages in a way that the consumer can no longer read them or that messages are not longer delivered at all.

Defining a standard way to mark mails that have been handled by OPES systems is fairly simple and does not require new techniques by SMTP gateways; they already today MUST leave tracing information by adding "Received" headers to mails. Therefore, recipients receiving broken mail have a fair chance of finding the compromised OPES system by using the trace information. There is still no guarantee, as the email have been broken in a way that makes even the tracing information unreadable; but the chance will be even better than with other protocols such as HTTP, because most email clients allow the user to display mail headers, while many browsers have no mechanism to show the HTTP headers that might include tracing info.

Email that cannot be delivered because a compromised OPES system prevented the delivery of legitimate mail, MUST result in a an NDR to be sent to the originator of the mail according to the SMTP specifications [4]. OPES should not be forced to fix the issue that

NDRs are not reliable over SMTP.

3. Integrity, privacy and security considerations

3.1 Tracing info in OPES/SMTP

Tracing is an important requirement for OPES systems. Tracing information added to mails should follow a similar syntax and structure to that defined for OPES/HTTP in HTTP Adaptation with Open Pluggable Edge Services [5], and with the same guidelines as the SMTP specifications [4] define for the "Received" headers.

Trace information is then seen by mail recipients when the mails reach the recipient. Mail that cannot be delivered or that is blocked by the OPES service will either be rejected or cannot be delivered after it has been accepted by an SMTP server. In the latter case SMTP specifications [4] require that a NDR MUST be sent to the originator; OPES requires that if a NDR is sent that the report MUST also contain information about the OPES system so that the sender gets informed. If an email is rejected, an OPES system MUST also include trace data in the SMTP response so that the originator can find out why and where the mail was rejected.

3.2 Bypass in OPES/SMTP

If a mail message was rejected or could not be delivered (and a NDR was sent), the originator of the message may want to bypass the OPES system that blocked the message.

If the recipient of a message receives a mail with OPES trace information, he may want to receive a non-OPES version of the message. Although there is no direct in-band request from the recipient back to the OPES system, the recipient can contact the sender and ask her to send the message again and to add a bypass request for the OPES system.

An OPES system MAY also define out-of-band methods to request a bypass, for example a web interface or an email message sent to it that results in the creation of a white list entry for the sender/ recipient pair. Examples for these out-of-band methods are email systems that keep a copy of the original email in a quarantaine queue and only send the recipient a block notification plus either a direct link, or a digest notification with the ability to retrieve the original message from quarantaine.

OPES MUST implement methods to request a bypass but there cannot be a guarantee that the bypass request will be approved. The security needs of the receiver or the receiver's network may demand that certain filters must not by bypassed (such as virus scanners for example). In general, the receiver should be able to configure a

client centric OPES system, i.e. the receiver should be able to indicate if he/she wants to receive a non-OPES version if it is available.

Bypass requests could be added to the mail message or within the SMTP dialog. Bypass request data added to the mail message cannot bypass OPES services that operate on other SMTP dialog commands, which are sent before the mail message has been received (such as RCPT commands).

Bypass request data sent at the beginning of a SMTP dialog may not reach the OPES system if intermediate SMTP relays do not support those bypass request commands and don't forward that information.

3.3 Compatibility with Cryptographic Protection Mechanisms

Cryptography can be used to assure message privacy, to authenticate the originator of messages, and to detect message modification. There are standard methods for achieving some or all these protections for generic messages ([9], [10], [11]), and these can be used to protect SMTP data without changing the SMTP protocol.

The content of encrypted mail messages cannot be inspected by OPES systems because only the intended recipient has the information necessary for decryption. The IAB and others have suggested that users might want to share that information with OPES systems, thus permitting decryption by intermediates. For most cryptographic systems that are compatible with email, this would require end users to share their most valuable keys, in essence their "identities", with OPES machines. Some key management systems, particularly those which have centralized administrative control of keys, might have trust models in which such sharing would be sensible and secure.

Once having decrypted the message, if the OPES box modifies the content, it would be faced with the task of re-encrypting it in order to maintain some semblance of "end-to-end" privacy.

If OPES/SMTP had a way to interact with end users on a per message basis, it might be possible to communicate cryptographic key information from individual messages to end users, have them compute the message encrypting key for particular message, and to send that back to the OPES box. This would perhaps ameliorate the need to share a user's "master" message decrypting key with the OPES box. This kind of communication has not been defined for OPES.

Message protection systems generally include some message integrity mechanisms by which recipient can check for message modification that may have occurred after the sender released the message. This

protection can be applied to encrypted or plaintext messages and can be accomplished through either symmetric or asymmetric cryptography. In the case of symmetric cryptography, the key sharing problem is exactly similar to the encryption case discussed previously. If the OPES box modified the content, then the message integrity (or authentication) code would have to be re-calculated and included with the modified message.

For asymmetric cryptography the situation is more complicated. The message integrity is tied to the sender's public key, and although anyone who can get the sender's public key can also check for message modification, no one but the sender can compute the sender's signature on a modified message. Thus, an OPES system could not modify messages and have them appear to come from the purported sender. The notion of sharing the sender's signing key with the OPES system is unpalatable, because few trust models would be compatible with sharing digital identities across organization boundaries. However, if the OPES system doing the modification were under the control of the sender's local administration, the sharing might be sensible (as discussed for decryption, above).

OPES/SMTP systems could present modified content showing the modified regions in a form that permits authentication of the original message and authentication of the OPES modifications (assuming the OPES box had a digital signature identity and key). One method for doing this is outlined in [12], but to our knowledge this method is not in any standard.

There are security risks associated with sharing cryptographic keys that must be addressed by implementors. Because this is not a simple task, it is not a requirement for OPES/SMTP.

4. Protocol requirements for OPES/SMTP

In addition to other documents listing requirements for OPES, the discussion in this document implies specific requirements for designing and implementing SMTP adaptations with OPES:

- OPES Systems MUST add tracing headers to mail messages

- If an email message that has been accepted by an OPES system cannot be delivered, the non delivery report MUST include trace information of the OPES system.

- The OPES/SMTP specifications MUST define a bypass request option that can be included in mail messages.

- The OPES/SMTP specifications MUST define a bypass request option as an extension for SMTP dialogs.

5. IAB Considerations for OPES/SMTP

This section lists the IAB considerations for OPES [2] and summarizes how OPES/SMTP addresses them.

5.1 IAB Consideration (2.1) One-Party Consent

The IAB recommends that all OPES services be explicitly authorized by one of the application-layer end-hosts (that is, either the data consumer application or the data provider application). For OPES/SMTP this means consent of either the email message sender or the recipient.

The application agnostic architecture of OPES [7] requires that "OPES processors MUST be consented to by either the data consumer or data provider application" (OPES processor is the email gateway for OPES/SMTP). This cannot prevent the consent-less introduction of OPES processors by incompliant OPES entities.

5.2 IAB Consideration (2.2) IP-Layer Communications

The IAB recommends that OPES processors must be explicitly addressed at the IP layer by the end user (data consumer application).

This requirement has been addressed by the architecture requirements in section 2.1 of [7] and has been further clarified in section 2.2 of [3].

5.3 IAB Consideration (3.1) Notification

"The overall OPES framework needs to assist content providers in detecting and responding to client-centric actions by OPES intermediaries that are deemed inappropriate by the content provider" [2].

For OPES/SMTP this translates into assistance for the email message sender to detect and respond to recipient-centric actions that are deemed inappropriate by the sender.

This has been addressed in Section 3.1 and by the second tracing requirements in Section 4. As discussed in Section 2.3 OPES/SMTP cannot prevent that NDRs are not sent or get blocked before reaching the sender of the original message.

5.4 IAB Consideration (3.2) Notification

"The overall OPES framework should assist end users in detecting the behavior of OPES intermediaries, potentially allowing them to

identify imperfect or compromised intermediaries" [2].

This is addressed in Section 3.1 and by the first tracing requirement in Section 4. It must be noted that some email systems do not make the email headers available to the end user although the headers belong to the payload that is transferred via SMTP. Building an OPES architecture with those email systems should be avoided or requires that the tracing information is made available to the end users in a different way.

5.5 IAB Consideration (3.3) Non-Blocking

"If there exists a "non-OPES" version of content available from the content provider, the OPES architecture must not prevent users from retrieving this "non-OPES" version from the content provider" [2].

For OPES/SMTP this has been discussed in Section 3.2 and is addressed by the two bypass requirements of Section 4.

5.6 IAB Consideration Application Layer Addresses (4.x)

While "most application layer addressing revolves around URIs" (section 8 of [2]), SMTP uses email addresses, for which the considerations apply to some degree only.

The SMTP use cases document [6] includes a use case for Mail Rerouting and Address Rewriting. Alias and email list address resolution are standard function of an email gateway described in [4].

Translating the reference validity consideration regarding inter- and intra-document reference validity to SMTP, OPES services mapping internal to external email addresses MUST ensure to properly map addresses in all affected email headers.

5.7 IAB Consideration (5.1) Privacy

This consideration recommends that the overall OPES framework must provide for mechanisms for end users to determine the privacy policies of OPES intermediaries.

The application agnostic part for OPES and has been discussed in section 10 of [3]. Email specific trace information that will be added to OPES/SMTP according to the requirements in Section 4 may raise additional privacy issues that MUST be added to the privacy policy description of the OPES system.

5.8 IAB Consideration Encryption

"If OPES was compatible with end-to-end encryption, this would effectively ensure that OPES boxes would be restricted to ones that are known, trusted, explicitly addressed at the IP layer, and authorized (by the provision of decryption keys) by at least one of the ends" [2].

This has been discussed in Section 3.3.

6. Security Considerations

The document itself discusses security considerations of OPES/SMTP. General security threats of OPES are described in Security Threats for OPES [8]

Section 3.3 (about compatibility with cryptographic protection mechanisms) mentions that an OPES system could eventually deal with cryptographic keys. This raises security issues (such as availability and storage of cryptographic keys) that must be addressed by the implementer.

7. References

7.1 Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[2] Floyd, S. and L. Daigle, "IAB Architectural and Policy Considerations for Open Pluggable Edge Services", RFC 3238, January 2002.

[3] Barbir, A. and A. Rousskov, "Open Pluggable Edge Services (OPES) Treatment of IAB Considerations", RFC 3914, October 2004.

7.2 Informative References

[4] Klensin, J., "Simple Mail Transfer Protocol", RFC 2821, April 2001.

[5] Rousskov, A. and M. Stecher, "HTTP Adaptation with Open Pluggable Edge Services (OPES)", RFC 4236, November 2005.

[6] Stecher, M. and A. Barbir, "Open Pluggable Edge Services (OPES) SMTP Use Cases", RFC 4496, May 2006.

[7] Barbir, A., Penno, R., Chen, R., Hofmann, M., and H. Orman, "An Architecture for Open Pluggable Edge Services (OPES)", RFC 3835, August 2004.

[8] Barbir, A., Batuner, O., Srinivas, B., Hofmann, M., and H. Orman, "Security Threats and Risks for Open Pluggable Edge Services (OPES)", RFC 3837, August 2004.

[9] Elkins, M., Del Torto, D., Levien, R., and T. Roessler, "MIME Security with OpenPGP", RFC 3156, August 2001.

[10] Housley, R., "Cryptographic Message Syntax (CMS)", RFC 3852, July 2004.

[11] Eastlake, D., Reagle, J., and D. Solo, "(Extensible Markup Language) XML-Signature Syntax and Processing", RFC 3275, March 2002.

[12] Orman, H., "Data Integrity for Mildly Active Content", Proceedings of the Third Annual International Workshop on Active Middleware Services p.73, August 2001.

Author's Address

Martin Stecher
Secure Computing Corporation
Vattmannstr. 3
33100 Paderborn
Germany

Email: martin.stecher@webwasher.com
URI: http://www.securecomputing.com/

Appendix A. Acknowledgements

Many thanks to everybody who provided input and feedback for this document. Very special thanks to Hilarie Orman for her input and suggestions, especially for the content of Section 3.3 (about compatibility with cryptographic protection mechanisms).

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.