SIPPING
Internet-Draft
Intended status: Informational
Expires: January 15, 2009

H. Tschofenig
Nokia Siemens Networks
H. Schulzrinne
Columbia University
D. Wing
J. Rosenberg
Cisco Systems
D. Schwartz
XConnect

July 14, 2008

A Framework to tackle Spam and Unwanted Communication for Internet Telephony
draft-tschofenig-sipping-framework-spit-reduction-04

Status of this Memo

Abstract

Spam, defined as sending unsolicited messages to someone in bulk, is likely to become a problem on SIP open-wide deployed networks. A number of solutions have been proposed for dealing with Spam for Internet Telephony (SPIT) and unwanted communication, such as content filtering, black lists, white lists, consent-based communication,

reputation systems, address obfuscation, limited use addresses, turing tests, computational puzzles, payments at risk, circles of trust, and many others.

This document describes the big picture that illustrates how the different building blocks fit together and can be deployed incrementally.

Table of Contents

1. Introduction

The problem of Spam for Internet Telephony (SPIT) is an imminent challenge and only the combination of several techniques can provide a way to deal with unwanted communication attempts.

[RFC5039] provides four core recommendations that need to be considered for a SPIT solution, namely

- o Strong Identity
- o White Lists
- o Solve the Introduction Problem
- o Don't Wait Until its Too Late

This document illustrates how existing building blocks can be put together to be able to recognize unwanted communication attempts and to execute appropriate actions. Ideally, a framework should allow new building blocks to be added as adversaries become more sophisticated. Since there are strong economical incentives for adversary to exploit communication networks that are widely deployed it only possible to detect and react on unwanted communication attempts in such a way that the total number of unwanted communication attempts reaches a level that is acceptable for the end user considering false positives and the additional burden for the users using these mechanisms.

The purpose of this document defines a model of internal device processing, protocol interfaces, and terminology to illustrate a way in which SPIT prevention techniques can be added in a seamless fashion. This document focuses on the descripion of how to combine different building blocks in an architectural fashion. No specific pre-selection is being provided on what mechanism should be standardized or implemented by various parties. This is left to the parties deploying these mechanisms and, when it comes to standardization, subject of a separate document to pick an initial set of mechanisms to start with.

2. Terminology

This document does not contain normative language.

3. Framework

Figure 1 shows the interaction between the end host and a SIP proxy belonging to its VoIP provider. One important part of the overall solution is the ability to make authorization decisions based on

incoming communication attempts. The entity that writes these authorization rules is referred as Rule Maker. A human, acting as the Rule Maker, might enter policies via some form of graphical user interface; some other policies may be generated automatically by observing the behavior of the user. Furthermore, in certain deployment environments an initial rule set will be provided by some third party entity, such as the enterprise system administrator or the VoIP service provider.

Policies are processed by corresponding module within the SIP proxy, called Authorization Engine, that interacts with the message routing component. By following this architectural approach the Policy Decision Point (PDP) and the Policy Enforcement Point (PEP) are closely combined. As such, authorization policies are stored at at a SIP proxy rather than the SIP UA client itself. The implications of relocating these two functions, PDP and PEP, to the SIP UA client are described in Appendix A.

```
             +---------------------------------------------------------+
             |                 Authorization                           |
             |  re-route       Policy                   +------------+ |
             |      ^          (implicit)         ######| Rule Maker | |
             |      o          +#######+          #     +------------+ |
             |      o          #       #          #                    |
             |  +---o----------#-------#--+       # Authorization      |
             |  |   o          #       #  |<####### Policy             |
 +--------+  |  |   o   Proxy  #       #  |                            |
 |        |  |  |   o          #       #  |<*******************+       |
 | Sender |<***>|+-------+     v       #  |                    *       |
 |        |  |  ||Msg.   |   +-----------+| Authorization      *       |
 +--------+  |  ||Routing|   |  Authz.   || Policy (explicit)  *       |
   ^    o    |  ||Engine |<->|  Engine   |<#################+  *       |
   *    o    |  |+-------+   +-----------+|                 #  *       |
   *    o    |  +-^--*--^-----------------+                 #  v       |
   *    o    |    o  *  o                              +-------------+ |
   *    o    |    o  *  o                              |             | |
   *    +oooo|oooo+  *  +ooooooooooooooooooooooooooooo>|  Recipient/ | |
   +**************************************************>|  Rule Maker | |
             |                                         +-------------+ |
             |                                                         |
             |                                                         |
             +-------------------Domain Boundary-----------------------+
```

Legend:

oooo: SIP message interaction
****: Protocol Interaction for authorizing the message sender
####: Management of authorization policies

Figure 1: Overview

Assume that an arbitrary entity transmits a message to a specific URI that finally hits the SIP proxy on the recipients side. Information provided within that message are used as input to the rule evaluation. Any part of the message may serve as input to the evaluation process but for practical reasons a few selected fields do most of the work. There are three aspects to consider when it comes to the rule evaluation:

Where does identity information come from?

Authentication information can come in different forms, depending on the chosen SIP security mechanism (e.g., P-Asserted-ID [RFC3325] or SIP Identity [RFC4474]). Additionally, the interworking with the privacy mechanisms, such as [RFC3323] or [I-D.ietf-sip-ua-privacy] need to be considered.

An example of how these different mechanisms are being considered during the rule evaluation is described in Common Policy [RFC4745] and Presence Authorization Policy [RFC5025].

What is the quality of the authentication procedure?

When evaluating authorization policies with respect to an incoming request the identity information of the entity sending the message may provide enough information when the recipient authorized that specific sender's identity. However, when the authorization policies refer to entire domains instead of individual users then it would be valuable to know how easily users within that specific domain are able to aquire their identities and how strong the authentication procedure actuallly is. Consider the following example: an enterprise network provisions entities to employees only and the authentication credentials are based on a smart card based mechanism. In an other case new identities can be created on the fly using a protocol interaction with an email-address based return routability check without additional verification. As such, in these two examples the chances to hold a real-world person accountable for their actions is very likely to be different in case that abuse reports are received by the two VoIP providers. Unfortunately, information about such a process is often not available when the authorization decision is being made.

Who creates the policies?

Identity based authorization rules may contain entries for specific users or for entire domains. Such policies may be configured by the end host as a Rule Maker or by the VoIP provider themself. Particularly the later part is likely to be attractive for VoIP providers since they may be able to form federations of VoIP providers that fulfill certain preconditions with respect to their VoIP / IM usage. These type of federations are also the basis for getting SIP SAML [I-D.ietf-sip-saml] to work since a valid digital signature together with the presence of certain assertions statements is insufficient as a basis for trusting their content.

As illustrated above, there are various possible actions that may be taken by the receipient or it's VoIP provider to authorize the message sender. Some of these mechanisms may require interaction with the sender. The request for authorization might require the message sender to be challenged (e.g., via hash cash [I-D.jennings-sip-hashcash], via SIP payment [I-D.jennings-sipping-pay], or via CAPTCHAs [I-D.tschofenig-sipping-captcha]). Some other mechanisms, such as SIP Identity do not require the verifying entity to challenge the

authentication service since the identity assertion is pushed towards the recipient.

Additionally, it is possible to utilize mechanisms the Consent Framework [I-D.ietf-sipping-consent-framework] or the Information Event Package [RFC3857] to allow the recipient to authorize a request.

Figure 2 shows this integration step. The conditions part of the rule offer a mechanisms to incrementally extend the overall framework with new components. Depending on the outcome of the rule evaluation, the message may be re-routed to another entity, such as an answering machine, to the recipient, rejected or other actions are triggered. The latter aspect is particularly interesting since it allows further solution components to be executed.

```
SIP msg with
authenticated
identity      +---------------+
-------------->|               |---------------->
Additional     |               | Spam marked msg
Msg fields     | Authorization |
-------------->| Engine        |---------------->
Other SPIT     |               | Re-routed msg
Prevention     |               |
Components     |               |---------------->
-------------->+---------------+ Forwarded to
                     |   |       original recipient
                     |   |
         <-----------+   +----------->||
     Politely blocked     Blocked
```

Figure 2: Message Filtering and Routing

Note that some traffic analysis and consequently some form of content filtering (e.g., of MESSAGEs) message be applied locally within the VoIP provider's domain also under the control of the end user. However, this is largely an implementation-specific technique without protocol impact. For example, consider a VoIP provider that wants to utilize a statistical analysis tool for Spam prevention. It is not necessary to standardized the algorithms nor protocols; the impact for the authorization policies is mainly the ability to allow the Rule Maker to enable or to disable the usage of these statistical techniques and potentially to map the output of the analysis process to value range from 0 (i.e., the message is not classified as Spam) and 100 (i.e., the message was classified as Spam). A Rule Maker may decide to act with an appropriate action on a certain level of Spam

marking.

Authenticated Identities:

Initial VoIP provider are likely to secure their SIP signaling using Transport Layer Security (TLS) or IP security (IPsec) between neighboring providers and use P-Asserted-ID [RFC3325].

Note: SIP Identity is comparable to DomainKeys Identified Mail (DKIM) [I-D.ietf-dkim-overview] used for associating a "responsible" identity with an email message and provides a means of verifying that the association is legitimate.

SIP Identity [RFC4474] is a proposal for stronger security mechanisms used to provide the verification service with the authenticated identity. SIP Identity is a reasonably simple specification and does not rely on a huge amount of infrastructure support.

This framework does not assume a specific mechanism for asserting identities to be used but a strong identity mechanism is a pre-requisity for authorization policy handling to be successful.

Authorization Policies:

Even if policy decision making and policy enforcement is done outside the SIP UA client then still there might not be a need to standardize an authorization policy language if the policies can be modified via a webpage. This approach of policy handling is done in many cases today already for various applications.

Unfortunately, this approach tends to become cumbersome for end users and therefore it is better to hide a lot of policy details from the end user itself and to make use of context information, for example, address books and authorization policies available already created for presence based systems.

Additionally, a user may have multiple devices and a consistent view of the policies should be provided.

An example solution for authorization policies for dealing with reducing unwanted communication is described in [I-D.tschofenig-sipping-spit-policy] with the requirements detailed in [I-D.froment-sipping-spit-requirements].

There is still one significant problem unsolved: since white lists need to be created somehow and hence there is an introduction problem. Section 4 discusses this aspect in more details.

4. Communication Patterns and User Groups

When communication takes place then at least three types of groups can be identified.

4.1. Closed Groups

People in this group communicate only with the peers in their group. They do not appreciate communication attempts from outside. Communication is possible only for people within this list. Here is an example of a closed group: Consider parents that do not want their children from getting contacted by strangers. Hence, they may want to create a white list containing the identifies of known friends, parents and other relatives on behalf of their kids.

The usage of authorization policies for usage with closed groups is straight forward. The introduction problem is also not considered very large given that the identities of the individual entities are typically known in an out-of-band fashion.

4.2. Semi-Open Groups

In a semi-open environment all members of the same group are allowed to get in contact with everyone else (e.g., persons working within the same company are allowed to contact each other without restrictions). For the communication with persons outside the company the communication patters depend on the role of the specific person (e.g., standardization people, sales people, etc.) and on the work style of the person.

For this category we distinguish a number of (non-spam) message sources based on their characteristics:

- o "friends" or "acquaintances", i.e., those we have communicated with before.
- o strangers, divided into 'interesting' and 'uninteresting'. The latter are messages from people that someone does not care to have a conversation with or respond to, at least at that particular moment.

Strangers can be defined by individual names or whole domains. A special class of 'stranger' messages are transaction-related communications, such as automated messages or calls from an airline or shipping company.

One way to deal with the introduction problem is to make use of techniques like hash cash [I-D.jennings-sip-hashcash] or Completely Automated Public Turing Test to Tell Computers and Humans Apart

(CAPTCHA) based robot challenges [I-D.tschofenig-sipping-captcha]. Alternatively, a communication attempt may also be forwarded to an answering maschine or alternative ways of establishing the initial interaction may be proposed.

The usage of authorization policies for usage with Semi-Open Groups is challenging but is considered manageable.

4.3. Open Groups

People in this type of group are not allowed to limit communication attempts. Help desks, certain people in governmental agencies, banks, insurance companies, etc.

For open groups a solution for providing SPIT prevention is far more complicated. Consider a person working on a customer support helpdesk. Ideally, they would like to receive only calls from friendly customers (although the motivation for calling is most likely a problem they experience) and the topic of the calls only relates to problems they are able to solve. Without listening to the caller they will have a hard time to know whether the call could be classified as SPIT or not. Another extreme case is a Public Safety Answering Point where emergency service personell is not allowed to reject calls either.

Many SPIT prevention techniques might not be applicable since blocking callers is likely not possible and applying other techniques, such as turing tests, might not be ideal in an case of open groups.

Providing additional information about the caller may be helpful from the called party VoIP provider but cannot be considered sufficient. A more promising approach is the ability to provide abuse reporting in the style of [XEP-0161] to provide the ability for punishment in case of misuse. This approach is helpful if an honest VoIP provider has to deal with a small number of adversaries within their network and the abuse reporting entity is trusted by that VoIP provider as well. This technique is not helpful when VoIP provider itself is convolated in sending spam messages or has some other financial benefits from not holding the adversary accountable. Another possible approach is to establish blacklisted domains within a federation, as this is common practice within the email domain.

4.4. Summary

Based on the discussions regarding communication patters and groups the following observations can be made:

- A single person very likely has many roles and they may have an impact on the communication patterns.
  For example, consider a person who is working in a company but also want to be available for family members.
- The context in which a person is may change at any time. For example, a person might be available for family members while at work except during an important meeting where communication attempts may be rejected. Switching a context has an impact for reachability and the means for communicating with a specific recipient, based on enabled rule sets.

From an authorization policy point of view it is important to be able to express a sphere (i.e., the state a user is in) and to switch between different spheres easily by thereby switching to a different rule set.

4.5. Usability

An important aspect in the usage of authorization policies is to assist the user when creating policies. Ideally, the policies should be established automatically. Below, there are a couple of examples to illustrate the idea given that these aspects are largely implementation issues:

- It must be possible for the proxy to automatically add addresses on outbound messages and calls to the rule set. This approach is similar to stateful packet filtering firewalls where outbound packets establish state at the firewall to allow inbound packets to traverse it again.
- Already available information in the address book can be used for building the policy rules there is quite likely already a relationship available with these persons existent.
- A large amount of email is non-personal, automated communication, such as newsletters, confirmations and legitimate advertisements. These are often tagged as spam by content filters. This type of correspondence is usually initiated by a transaction over the web, such as a purchase or signing up for a service. [I-D.shacham-http-corr-uris], for example, defines an HTTP header for conveying future correspondence addresses that can be integrated in the rule set.

5. Protocol Interactions

This section describes the necessary building blocks that are necessary to tie the framework together.

5.1. Rule Enforcement via a Trusted Intermediary

- Some from of strong identity assurance is required to build the basis for identity-based authorization. SIP Identity [RFC4474] or P-Asserted-ID [RFC3325] are examples of available mechanisms. These mechanisms allow the authenticated identity of the sending party to be determined.
- Authorization Policies based on the Common Policy framework [RFC4745], as extended in [I-D.tschofenig-sipping-spit-policy] for the purpose of SPIT prevention, are mandatory to implement at the end host side and at the trusted intermediary. The implementation of the rule evaluation engine might only be necessary on the trusted VoIP proxy. Harmonization with the work done for presence authorization [RFC5025], which is based on Common Policy [RFC4745], can be accomplished and is highly desirable.
- XML Configuration Access Protocol (XCAP) [RFC4825] is used to create, modify and delete authorization policies and is mandatory to implement at the end host side and at the trusted intermediary.

5.2. Incremental Deployment

An important property is incremental deployment of additional solution components that can be added and used when they become available. This section aims to illustrate how the extensibility is accomplished, based on an example.

Consider a VoIP provider that provides authorization policies that provide the following functionality equivalent to the Common Policy framework, i.e., identity-based, sphere and validity based conditions initially. For actions only 'redirection' and 'blocking' is provided. In our example we give this basic functionality the AUID 'new-spit-policy-example' with the namespace 'urn:ietf:params:xml:ns:new-spit-policy-example'.

When a client queries the capabilities of a SIP proxy in the VoIP providers network using XCAP the following exchange may take place.

```
GET   /xcap-caps/global/index HTTP/1.1
Host: xcap.example.com
```

Figure 3: Initial XCAP Query for Capabilities

```
HTTP/1.1 200 OK
  Etag: "wwhha"
  Content-Type: application/xcap-caps+xml

  <?xml version="1.0" encoding="UTF-8"?>
  <xcap-caps xmlns="urn:ietf:params:xml:ns:xcap-caps">
    <auids>
        <auid>new-spit-policy-example</auid>
        <auid>xcap-caps</auid>
    </auids>
    <namespaces>
      <namespace>urn:ietf:params:xml:ns:xcap-caps</namespace>
      <namespace>urn:ietf:params:xml:ns:spit-policy</namespace>
      <namespace>urn:ietf:params:xml:ns:common-policy</namespace>
    </namespaces>
  </xcap-caps>
```

Figure 4: Initial XCAP Response with the supported Capabilities

As shown in the example above, Common Policy and the example SPIT extension is implemented and the client can upload rules according to the definition of the rule set functionality.

Later, when the VoIP provider updates the functionality of authorization policies as more sophisticated mechanisms become available and get implemented the functionality of the authorization policy engine is enhanced with, for example, hashcash and the ability to perform statistical analysis of signaling message. The latter functionality comes with the ability to mark messages are Spam and the ability for end users to enable/disable this functionality. We use the namespaces 'urn:ietf:params:xml:ns:hashcash' and 'urn:ietf:params:xml:ns:statistical-analysis' for those.

A end user could now make use of these new functions and a capability query of the SIP proxy would provide the following response.

```
GET   /xcap-caps/global/index HTTP/1.1
Host: xcap.example.com
```

Figure 5: Second XCAP Query for Capabilities

```
HTTP/1.1 200 OK
  Etag: "wwhha"
  Content-Type: application/xcap-caps+xml

  <?xml version="1.0" encoding="UTF-8"?>
  <xcap-caps xmlns="urn:ietf:params:xml:ns:xcap-caps">
    <auids>
        <auid>spit-policy</auid>
        <auid>xcap-caps</auid>
        <auid>hashcash</auid>
        <auid>statistical-analysis</auid>
    </auids>
    <namespaces>
      <namespace>urn:ietf:params:xml:ns:spit-policy</namespace>
      <namespace>urn:ietf:params:xml:ns:common-policy</namespace>
      <namespace>urn:ietf:params:xml:ns:hashcash</namespace>
      <namespace>urn:ietf:params:xml:ns:statistical-analysis</namespace>
    </namespaces>
  </xcap-caps>
```

Figure 6: Second XCAP Response with the supported Capabilities

New SPIT handling functionality may extend condition, actions and/or transformation elements of a rule.

5.3. Botnets

A botnet is a large number of compromised maschines that are used to create and send spam or viruses or flood a network with messages as a denial of service attack.

Such a botnet represents a significant challenge for a VoIP infrastructure and also for the mechanisms proposed in this document. Recently observed attacks indicated that some botnets tried to steal credentials to distribute messages with "real" identities. To deal with the threat it is useful to classify the behavior of these bots into three categories, namely

- o The botnet does not have access to the user's credentials. In this case identity-based white lists provides adequate protection.
- o The botnets does have access to user's credentials of compromised maschines but distributes messages in a random fashion. In this case identity-based white lists provides adequate protection since it is unlikely that the recipient will have that person in their whitelist.
- o In this category the botnet has access to the user's credentials and utilizes addresses from the user's addressbook. In this case whitelists do not provide a proper protection. Since the

recipient knows the sender of the message it would, in many cases, be able to get in contact with him or her and report the observed problem. This approach does not work with a pure maschine-to-maschine communication environment without user involvement.

6. Privacy Considerations

This document does not propose to distribute the user's authorization policies to other VoIP providers nor is the configuration of policies at SIP proxies other than the trusted user's VoIP provider necessary. Furthemore, if blocking or influencing of the message processing is executed by the VoIP provider then they have to be explicitly enabled by the end user. Blocking of messages, even if it is based on "super-clever" machine learning techniques often introduces unpredictability.

Legal norms from fields of law can take regulative effects in the context of SPIT processing, such as constitutional law, data protection law, telecommunication law, teleservices law, criminal law, and possibly administrative law. See, for example, [Law1], [Law2] and [Law3]. For example, it is mandatory to pass full control of SPIT filtering to the end user, as this minimises legal problems.

An overview about regulatory aspects can be found in [Spit-AL].

7. Example

This section shows an example whereby we consider a user Bob@company-example.com that writes (most likely via a nice user interface) the following policies. We use a high-level language to show the main idea of the policies.

```
RULE 1:
     IF identity=alice@foo.example.com THEN ACCEPT
     IF identity=tony@bar.example.com THEN ACCEPT

RULE 2:
     IF domain=company-example.com THEN ACCEPT

RULE 3:
     IF unauthenticated THEN
            EXECUTE hashcash

RULE 4:
     IF <hashcash result="success"/>
     THEN
        REDIRECT sip:voicebox@company-example.com

RULE 5:
     IF <hashcash result="failure"/>
     THEN
        block
```

Figure 7: Example of Bob's Rule Set

At some point in time Bob uploads his policies to the SIP proxy at his VoIP providers SIP proxy.

```
PUT
/spit-policy/users/sip:bob@company-example.com/index/~~/ruleset

HTTP/1.1
Content-Type:application/spit+xml
Host: proxy.home-example.com

 <<<< Added policies go in here. >>>>
```

Figure 8: Uploading Policies using XCAP

When BoB receives a call from his friends, alice@foo.example and tony@bar.example.com, then all the rules related to the spit policy are checked. Only the first rule (rule 1) matches and is applied. Thus, the call is forwarded without any further checks based on Rule 1. The rules assume that the authenticated identity of the caller has been verified.

When Bob receives a call from a co-worker, Charlie@company-example.com, Rule 2 is applied since the domain part in the rule matches the domain part of Charlie's identity.

Now, when Bob receives a contact from an unknown user, called Mallice in this example. Rule 3 indicates that an extended return-routability test using hashcash [I-D.jennings-sip-hashcash] is used with the call being redirected to Bob's voicebox afterwards. This exchange is shown in Figure 9.

```
  UA                                                          Bob's
Malice                     Proxy                            Voicebox
  |         INVITE           |                                       |
  |------------------------->|Puzzle: work=15;                       |
  |                          |pre="VgVGYixbRg0mdSwTY3YIfCBuAAA=";    |
  |         419 with Puzzle  |image="NhhMQ2l7SE0VBmZFKksUC19ia04="; |
  |                          |value=160                              |
  |<-------------------------|                                       |
  |                          |                                       |
  |         ACK              |                                       |
  |------------------------->|                                       |
  |                          |Puzzle: work=0;                        |
  |                          |pre="VgVGYixbRg0mdSwTY3YIfCBuYmg=";    |
  |                          |image="NhhMQ2l7SE0VBmZFKksUC19ia04="   |
  |  INVITE with Solution    |value=160                              |
  |------------------------->|             INVITE                    |
  |                          |-------------------------------------->|
  |                          |                                       |
  |                          |             180 Ringing               |
  |        180 Ringing       |<--------------------------------------|
  |<-------------------------|                                       |
  |                          |             200 OK                    |
  |        200 OK            |<--------------------------------------|
  |<-------------------------|                                       |
  |                          |      ACK                              |
  |----------------------------------------------------------------->|
  |                          |                                       |
```

Figure 9: Example Exchange: Malice contacts Bob

Depending on the outcome of the exchange the call is forwarded to a mailbox sip:voicebox@company-example.com (in case Malory returned the correct solution, see Rule 4) or blocked in case an incorrect response was provided. It might be quite easy to see how this rule set can be extended to support other SPIT handling mechanisms as well (e.g., CAPTCHAs, SIP Pay, etc.).

8. Security Considerations

This document aims to describe a framework for addressing Spam for

Internet Telephony (SPIT) in order to make it simple for users to influence the behavior of SIP message routing with an emphasis on SPIT prevention.

The framework relies on three building blocks, namely SIP Identity, authorization policies based on Common Policy and Presence Authorization Policy, and XCAP.

As a high-level overview, the framework allows the user to control end-to-end connectivity at the SIP message routing level whereby the glue that lets all parts fit together is based on authorization policies. Several other solution components can be developed independently and can be plugged into the framework as soon as available.

It must be avoided to introduce Denial of Service attacks against the recipient by misguiding him or her to install authorization policies that allow senders to bypass the policies although that was never intended by the recipient. Additionally, it must not be possible by extensions to the authorization policy framework to create policies to block legitimate senders or to stall the processing of the authorization policy engine.

9. Acknowledgments

We would like to thank

Jeremy Barkan, Dan York, Alexey Melnikov, Thomas Schreck, Eva Leppanen, Cullen Jennings, Marit Hansen and Markus Hansen for their review comments to a pre-00 version.
Jeremy Barkan, Eva Leppanen, Michaela Greiler, Joachim Charzinski, Saverio Niccolini, Albert Caruana, and Juergen Quittek for their comments to the 00 version.
Otmar Lendl, Jan Seedorf, Saverio Niccolini, Kai Fischer, Joachim Charzinski, Dan York, Peter Saint-Andre, Brian Azzopardi, Martin Stiemerling, and Juergen Quittek for their comments to the -01/-02 version.

10. References

10.1. Normative References

10.2. Informative References

[RFC3323] Peterson, J., "A Privacy Mechanism for the Session Initiation Protocol (SIP)", RFC 3323, November 2002.

[RFC4474] Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 4474, August 2006.

[RFC4745] Schulzrinne, H., Tschofenig, H., Morris, J., Cuellar, J., Polk, J., and J. Rosenberg, "Common Policy: A Document Format for Expressing Privacy Preferences", RFC 4745, February 2007.

[RFC3325] Jennings, C., Peterson, J., and M. Watson, "Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks", RFC 3325, November 2002.

[RFC4825] Rosenberg, J., "The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)", RFC 4825, May 2007.

[RFC5039] Rosenberg, J. and C. Jennings, "The Session Initiation Protocol (SIP) and Spam", RFC 5039, January 2008.

[RFC5025] Rosenberg, J., "Presence Authorization Rules", RFC 5025, December 2007.

[I-D.jennings-sip-hashcash]
Jennings, C., "Computational Puzzles for SPAM Reduction in SIP", draft-jennings-sip-hashcash-06 (work in progress), July 2007.

[I-D.wing-sipping-spam-score]
Wing, D., Niccolini, S., Stiemerling, M., and H. Tschofenig, "Spam Score for SIP", draft-wing-sipping-spam-score-02 (work in progress), February 2008.

[I-D.ietf-sipping-consent-framework]
Rosenberg, J., "A Framework for Consent-Based Communications in the Session Initiation Protocol (SIP)", draft-ietf-sipping-consent-framework-05 (work in progress), June 2006.

[I-D.ietf-dkim-overview]
Hansen, T., Crocker, D., and P. Hallam-Baker, "DomainKeys Identified Mail (DKIM) Service Overview", draft-ietf-dkim-overview-10 (work in progress), July 2008.

[I-D.tschofenig-sipping-spit-policy]
Tschofenig, H., Wing, D., Schulzrinne, H., Froment, T., and G. Dawirs, "A Document Format for Expressing

Authorization Policies to tackle Spam and Unwanted
Communication for Internet Telephony",
draft-tschofenig-sipping-spit-policy-03 (work in
progress), July 2008.

[I-D.schwartz-sipping-spit-saml]
Schwartz, D., "SPAM for Internet Telephony (SPIT)
Prevention using the Security Assertion Markup Language
(SAML)", draft-schwartz-sipping-spit-saml-01 (work in
progress), June 2006.

[I-D.shacham-http-corr-uris]
Shacham, R. and H. Schulzrinne, "HTTP Header for Future
Correspondence Addresses", draft-shacham-http-corr-uris-00
(work in progress), May 2007.

[I-D.jennings-sipping-pay]
Jennings, C., "Payment for Services in Session Initiation
Protocol (SIP)", draft-jennings-sipping-pay-06 (work in
progress), July 2007.

[I-D.froment-sipping-spit-requirements]
Tschofenig, H., Dawirs, G., Froment, T., Wing, D., and H.
Schulzrinne, "Requirements for Authorization Policies to
tackle Spam and Unwanted Communication for Internet
Telephony", draft-froment-sipping-spit-requirements-03
(work in progress), July 2008.

[I-D.niccolini-sipping-feedback-spit]
Niccolini, S., "SIP Extensions for SPIT identification",
draft-niccolini-sipping-feedback-spit-03 (work in
progress), February 2007.

[I-D.tschofenig-sipping-captcha]
Tschofenig, H., Leppanen, E., Niccolini, S., and M.
Arumaithurai, "Completely Automated Public Turing Test to
Tell Computers and Humans Apart (CAPTCHA) based Robot
Challenges for SIP", draft-tschofenig-sipping-captcha-01
(work in progress), February 2008.

[I-D.ietf-sip-ua-privacy]
Munakata, M., Schubert, S., and T. Ohba, "UA-Driven
Privacy Mechanism for SIP", draft-ietf-sip-ua-privacy-01
(work in progress), February 2008.

[RFC3857] Rosenberg, J., "A Watcher Information Event Template-
Package for the Session Initiation Protocol (SIP)",
RFC 3857, August 2004.

[I-D.ietf-sip-saml]
Tschofenig, H., Hodges, J., Peterson, J., Polk, J., and D. Sicker, "SIP SAML Profile and Binding", draft-ietf-sip-saml-03 (work in progress), November 2007.

[Spit-AL] Hansen, M., Hansen, M., Moeller, J., Rohwer, T., Tolkmitt, C., and H. Waack, "Developing a Legally Compliant Reachability Management System as a Countermeasure against SPIT, Third Annual VoIP Security Workshop, Berlin, available at https://tepin.aiki.de/blog/uploads/spit-al.pdf", June 2006.

[Law1] "Bundesnetzagentur: Eckpunkte der regulatorischen Behandlung von Voice over IP (VoIP), available at http://www.bundesnetzagentur.de/media/archive/3186.pdf", September 2005.

[Law2] "70. Konferenz der Datenschutzbeauftragten des Bundes und der Laender: Entschliessung Telefonieren mit Internettechnologie (Voice over IP - VoIP), available at http://www.datenschutzzentrum.de/material/themen/presse/20051028-dsbk-voip.htm", Oktober 2005.

[Law3] "Working Party 29 Opinion 2/2006 on privacy issues related to the provision of email screening services, WP 118, available at http://ec.europa.eu/justice_home/fsj/privacy/docs/wpdocs/2006/wp118_en.pdf", February 2006.

[XEP-0161]
Saint-Andre, P., "Abuse Reporting", XSF XEP 0161, May 2007.

Appendix A. Authorization Engine in SIP UA

When white lists are stored and managed only at the SIP UA client then the authorization policies language and the protocol to modify the policies do not need to be standardized; they are purely implementation specific details.

While this appears to be an advantage there are various drawbacks including the inability to synchronize policies among different devices. Additionally, some information that is typically available to the Policy Decision Point may not be available to the end host. To avoid standardizing the exchange of such type of information an abstract form of Spam marking is proposed in [I-D.wing-sipping-spam-score].

Authors' Addresses

Hannes Tschofenig
Nokia Siemens Networks
Linnoitustie 6
Espoo 02600
Finland

Phone: +358 (50) 4871445
Email: Hannes.Tschofenig@gmx.net
URI: http://www.tschofenig.priv.at

Henning Schulzrinne
Columbia University
Department of Computer Science
450 Computer Science Building
New York, NY 10027
US

Phone: +1 212 939 7004
Email: hgs@cs.columbia.edu
URI: http://www.cs.columbia.edu

Dan Wing
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134
USA

Email: dwing@cisco.com

Jonathan Rosenberg
Cisco Systems, Inc.
600 Lanidex Plaza
Parsippany, New York 07054
USA

Email: jdrosen@cisco.com
URI: http://www.jdrosen.net

David Schwartz
XConnect
Malcha Technology Park
Jerusalem, 96951
Israel

Email: dschwartz@xconnect.net