

SIPPING
Internet-Draft
Intended status: Informational
Expires: May 22, 2008

H. Tschofenig
Nokia Siemens Networks
H. Schulzrinne
Columbia University
D. Wing
J. Rosenberg
Cisco Systems
D. Schwartz
Kayote Networks
November 19, 2007

A Framework to tackle Spam and Unwanted Communication for Internet
Telephony
draft-tschofenig-sipping-framework-spit-reduction-02.txt

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/lid-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 May 22, 2008.

Copyright Notice

Copyright (C) The IETF Trust (2007).

Abstract

Spam, defined as sending unsolicited messages to someone in bulk,

Internet-Draft

Framework for Reducing Spam

November 2007

might be a problem on SIP open-wide deployed networks. A number of solutions have been proposed for dealing with Spam for Internet Telephony (SPIT), for example, 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	3
2.	Terminology	3
3.	Framework	3
4.	Communication Patterns and User Groups	7
4.1.	Closed Groups	7
4.2.	Semi-Open Groups	7
4.3.	Open Groups	8
4.4.	Summary	9
4.5.	Usability	9
5.	Protocol Interactions	10
5.1.	End Host based Rule Enforcement	10
5.2.	Rule Enforcement via a Trusted Intermediary	10
5.3.	Incremental Deployment	10
6.	Privacy Considerations	12
7.	Example	13
8.	Security Considerations	15
9.	Acknowledgments	16
10.	References	16
10.1.	Normative References	16
10.2.	Informative References	17
Appendix A.	Outside the Scope	19
	Authors' Addresses	19
	Intellectual Property and Copyright Statements	21

[1.](#) Introduction

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

[I-D.ietf-sipping-spam] 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 form a solution to deal with SPIT. New building blocks can be added to provide more efficient SPIT handling, since there is no single solution that provides 100 % protection.

The main purpose of this document is largely to define 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. We focus only on the most important solution components and consider many other aspects either outside the scope of this work (see [Appendix A](#)) and postpone them for future work.

[2.](#) Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

[3.](#) Framework



Legend:

oooo: SIP message interaction

****: Protocol Interaction for authorizing the message sender

####: Management of authorization policies

Figure 1: Overview

Assume that an arbitrary sender transmits a message towards the recipients 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 only a few selected fields do most of the work. In this document, we argue that the

authenticated identity of the sender is the most valuable information item. In the future, when standardization progresses then, for example, reputation information obtained from social networks may offer additional input to the authorization process.

The protocol interaction for authorizing the message sender refers to the ability of the recipient or the proxy to interact with the sender to request authorization. The request for authorization is a pull model whereby the proxy or the recipient challenges the sender (e.g., via hash cash [[I-D.jennings-sip-hashcash](#)], or SIP payment [[I-D.jennings-sipping-pay](#)], or Completely Automated Public Turing Test to Tell Computers and Humans Apart (CAPTCHA) based robot challenges [[I-D.tschofenig-sipping-captcha](#)]) for authorization. SIP Identity on the other hand realizes as push model whereby authentication information is pushed towards the recipient.

Figure 2 shows this integration step. The conditions part of the rule offer a mechanisms to incrementally extend the overall framework with new SPIT prevention solution components. Depending on the rule evaluation the message may be rerouted 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.

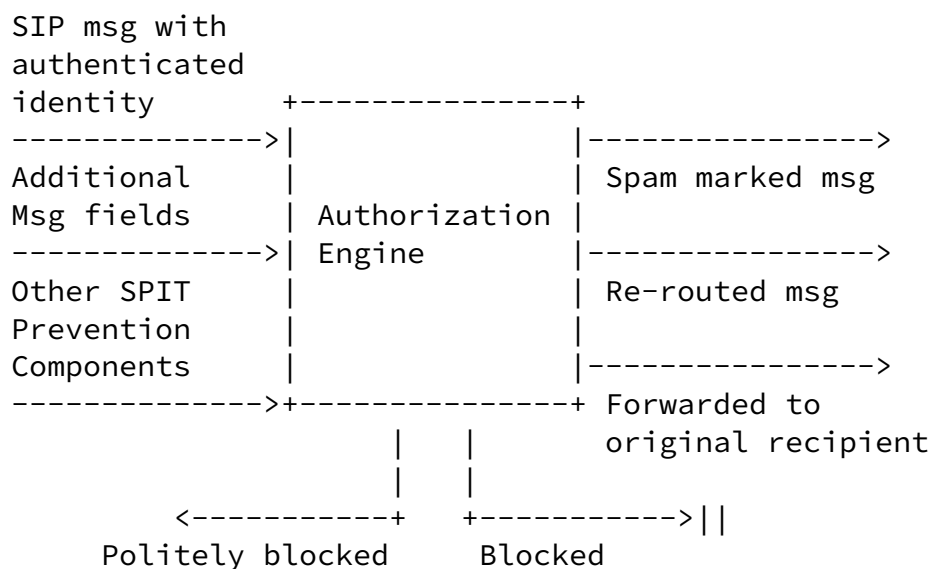


Figure 2: Message Filtering and Routing

Note that some traffic analysis and consequently some form of content filtering (e.g., of MESSAGES) can be applied locally within the VoIP provider's domain also under the control of the end user. For example, consider a Voice over IP provider that wants to utilize a statistical analysis tool for Spam prevention. It is not necessary

to standardized the algorithms; 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 for SPIT filtering 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). Conveying Spam marking is proposed in [\[I-D.schwartz-sipping-spit-saml\]](#), in [\[I-D.niccolini-sipping-feedback-spit\]](#) and in [\[I-D.wing-sipping-spam-score\]](#). A Rule Maker may decide to act with an appropriate action on a certain level of Spam marking.

In a minimalistic SPIT framework only authenticated identities in combination with authorization policies are used. This should serve as a starting point for future work.

Authenticated Identities:

SIP Identity [[RFC4474](#)] is assumed to be used to provide the receiver of a communication attempt with the authenticated identity. SIP Identity is a reasonable simple specification and does not rely on a huge amount of infrastructure support.

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.

Authorization Policies:

When the white lists are stored and managed only at the end host then the authorization policies and the protocol to modify the policies do not need to be standardized since they are purely implementation specific details. Even if the authorization policies are managed centrally or some amount of policy enforcement is done by trusted intermediaries then still there might not be a need to standardize an authorization policy language if the policies can be modified via a webpage. This type of policy handling is done in many cases today already for various applications.

Unfortunately, this approach tends to become cumbersome to manage 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.

In some cases the above-described approach is not sufficient whereby it is necessary to define a standardized protocol, for

example, if policies are used by different entities, created and modified in an automatic way and when multiple entities manipulate policies (potentially on behalf of the person affected by the policies), e.g., the user may have multiple devices.

An example solution for authorization policies providing Spam prevention capabilities are described in [[I-D.tschofenig-sipping-spit-policy](#)] with the requirements

detailed in [[I-D.froment-sipping-spit-requirements](#)].

The white list needs 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 are typically known.

[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., for example in a company environment). For members outside the company the communication patters depend on the role of the 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 Instant Messaging or automated 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 machine 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) 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. 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.

Statistical analysis may be helpful in some cases to deliver additional information about the calling party. Social networks might provide similar techniques based on reputation based systems.

[4.4.](#) Summary

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

- o 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.
- o 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:

- o 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.
- o 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.
- o 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. We will describe two different environments, namely one where rule enforcement happens at the end host and another one where a trusted network intermediary performs the actions.

[5.1.](#) End Host based Rule Enforcement

- o SIP Identity [[RFC4474](#)] is mandatory to implement at the end host and used to determine the authenticated identity of the sending side.
- o Authorization policies are purely implementation specific matter.

Since a purely end host based rule enforcement suffers from a number of drawbacks, rule enforcement by a trusted intermediary is also offered.

[5.2.](#) Rule Enforcement via a Trusted Intermediary

- o SIP Identity [[RFC4474](#)] or a corresponding mechanism is mandatory to implement at the trusted intermediary (e.g., the immediate VoIP provider) and it determines the authenticated identity of the sending side.
- o 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 [[I-D.ietf-simple-presence-rules](#)], which is based on Common Policy [[RFC4745](#)], can be accomplished and is highly desirable.
- o 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.

[5.3.](#) 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
```

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>
    <aid>new-spit-policy-example</aid>
    <aid>xcap-caps</aid>
  </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>
```

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 as 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.

Tschofenig, et al.

Expires May 22, 2008

[Page 11]

Internet-Draft

Framework for Reducing Spam

November 2007

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
```

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>
```

```
    <aid>spit-policy</aid>
```

```
    <aid>xcap-caps</aid>
```

```
    <aid>hashcash</aid>
```

```
    <aid>statistical-analysis</aid>
```

```
  </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>
```

Second XCAP Response with the supported Capabilities

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

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. Furthermore, 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

```

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. >>>>
[Editor's Note: In a future version an XML example
policy document will be listed here.]

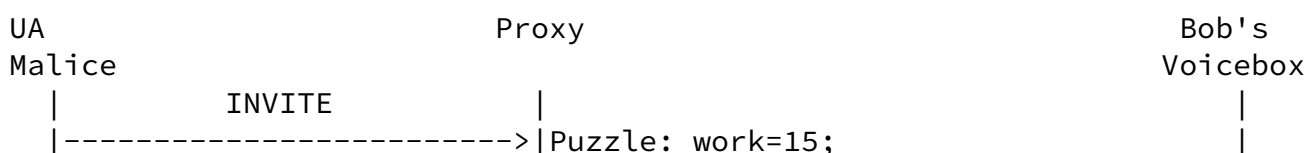
```

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.



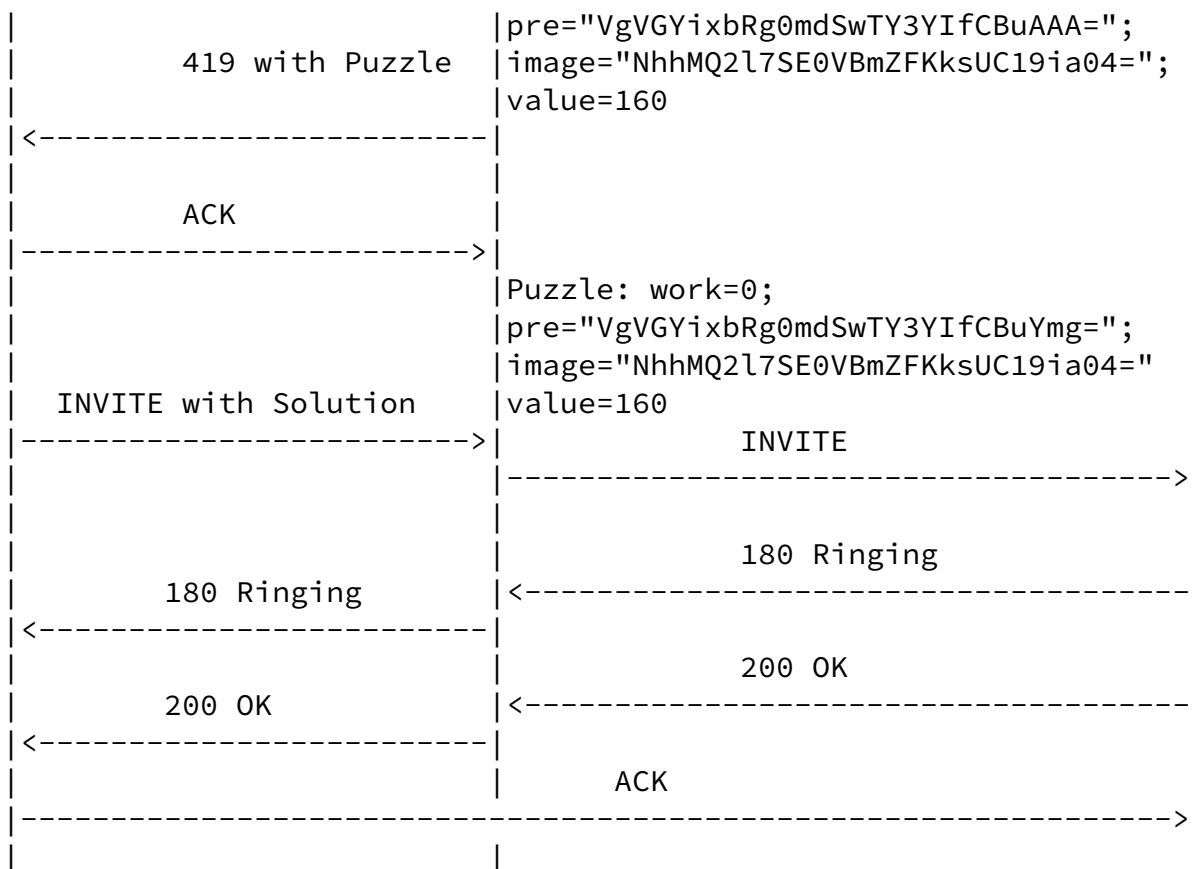


Figure 9: Example Exchange: Malice contacts Bob

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.

10. References

10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997.
- [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.
- [RFC4825] Rosenberg, J., "The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)", [RFC 4825](#), May 2007.

Internet-Draft

Framework for Reducing Spam

November 2007

[10.2.](#) Informative References

- [I-D.ietf-sipping-spam]
Rosenberg, J. and C. Jennings, "The Session Initiation Protocol (SIP) and Spam", [draft-ietf-sipping-spam-05](#) (work in progress), July 2007.
- [I-D.ietf-simple-presence-rules]
Rosenberg, J., "Presence Authorization Rules", [draft-ietf-simple-presence-rules-10](#) (work in progress), July 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., "Spam Score for SIP", [draft-wing-sipping-spam-score-00](#) (work in progress), August 2007.
- [I-D.ietf-sip-consent-framework]
Rosenberg, J., Camarillo, G., and D. Willis, "A Framework for Consent-based Communications in the Session Initiation Protocol (SIP)", [draft-ietf-sip-consent-framework-03](#) (work in progress), November 2007.
- [I-D.ietf-dkim-overview]
Hansen, T., Crocker, D., and P. Hallam-Baker, "DomainKeys Identified Mail (DKIM) Service Overview", [draft-ietf-dkim-overview-07](#) (work in progress), November 2007.
- [I-D.tschofenig-sipping-spit-policy]
Tschofenig, H., "A Document Format for Expressing Authorization Policies to tackle Spam and Unwanted Communication for Internet Telephony", [draft-tschofenig-sipping-spit-policy-01](#) (work in progress), July 2007.

[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]

Tschofenig, et al.

Expires May 22, 2008

[Page 17]

Internet-Draft

Framework for Reducing Spam

November 2007

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]

Froment, T., "Requirements for Authorization Policies to tackle Spam and Unwanted Communication for Internet Telephony", [draft-froment-sipping-spit-requirements-01](#) (work in progress), July 2007.

[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. and E. Leppanen, "Completely Automated Public Turing Test to Tell Computers and Humans Apart (CAPTCHA) based Robot Challenges for the Session Initiation Protocol (SIP)", [draft-tschofenig-sipping-captcha-00](#) (work in progress), July 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/press_e/20051028-dsbk-voip.htm", Oktober 2005.

[Law3] "Working Party 29 Opinion 2/2006 on privacy issues related

Tschofenig, et al.

Expires May 22, 2008

[Page 18]

Internet-Draft

Framework for Reducing Spam

November 2007

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.

Appendix A. Outside the Scope

We consider the following aspects outside the scope of this document:

- o Mechanisms to publish SPIT causing parties on the Internet, i.e.,
information about domains that create SPIT.
- o Determining the source of unwanted traffic in real-time.
- o Pushing packet filters and authorization policies towards the SPIT
sending domain.

Authors' Addresses

Hannes Tschofenig
Nokia Siemens Networks
Otto-Hahn-Ring 6
Munich, Bavaria 81739
Germany

Email: Hannes.Tschofenig@nsn.com

URI: <http://www.tschofenig.com>

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

Phone:
Email: dwing@cisco.com

Tschofenig, et al.

Expires May 22, 2008

[Page 19]

Internet-Draft

Framework for Reducing Spam

November 2007

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

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

David Schwartz
Kayote Networks
Malcha Technology Park
Jerusalem 96951
Israel

Email: david.schwartz@kayote.com

Full Copyright Statement

Copyright (C) The IETF Trust (2007).

This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS

OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).