Use Cases and Considerations for SIP Client Configuration and Management
draft-channabasappa-sipua-config-mgmt-00.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/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 August 30, 2006.

Copyright Notice

Copyright (C) The Internet Society (2006).

Abstract

This document provides use cases for the configuration and management of SIP-based devices (SIP client and applications) based on available IETF protocols and related methodologies. The use cases have been made generic enough to take into account common network topologies and requirements in SIP service deployments.

Based on the use cases, we present a preliminary analysis of available IETF protocols that meet these requirements and highlight some of the limitations.
Finally, a summary section captures the main findings to generate further discussion with the relevant IETF working groups. It is the intent of the authors to seek general feedback from the IETF community on the approaches outlined in this document, especially in the OPS area and in SIPPING.

Table of Contents

1. Overview .......................................................... 3
2. SIP UA Config and Management - Basic Scenario and Use Cases . 4
3. Use Case 1: SIP client provisioning using XML-based configuration and data modeling ................................. 6
4. Use Case 2: Access Control Definitions and XML-based configuration methodologies .......................... 11
5. Use Case 3: Management of clients behind firewalls (and NATs) ....................................................... 15
6. Summary and Next Steps ............................................. 18
7. Acknowledgments ..................................................... 19
8. Security Considerations ............................................ 20
9. References ............................................................ 21
   9.1. Normative References ........................................ 21
   9.2. Informative References ...................................... 21
Authors' Addresses .................................................. 23
Intellectual Property and Copyright Statements .............. 24
1. Overview

A couple of IETF working groups, OPS area and BoF meetings have reported the various emerging models and requirements to configure devices or applications and perform some monitoring or element management of such devices or apps. Various working groups are also defining the new IETF toolkit available to end-users, product vendors, enterprises and service providers to meet these configuration and element management requirements, including NETCONF, ISMS, SIP, SIPPING, SIMPLE, CallHome BoF to cite a few.

At the same time, many IP networks are seeing a shift in the types and variety of SIP-based communication clients being used and the associated configuration and management needs are evolving. A communication client may be software or hardware-based, embedded into an access network bridge or router, or a standalone device; it may operate on diverse platforms (PDAs, PCs, Mobile phones, home router) and operating systems (real and non-real time), with differing requirements related to capabilities, memory and CPU constraints.

As described in XCAP, multiple end-users can share a client and multiple clients can be bound to an end-user.

In short, from a configuration and management standpoint, the above means:

- increased configuration requirements to handle multiple profiles and dynamic changes. Configuration changes may be generated by multiple sources: an end-user (based on application usage needs), a client (based on network conditions or profile usage), or an actor (admin user, enterprise IT or service provider sysadmin);

- support for configuration and management of clients with single or dual IP stacks with IPv4 and IPv6 support which may be behind NATs and firewalls.

This document attempts to describe a basic scenario for configuring and managing SIP devices by providing a couple of common use cases and looking at the available solutions in the IETF toolkit. Our goal is to show the design choices some folks are confronted with and underline open issues to support some of the current IETF work and, more importantly, outline some areas that potentially need more work.
2. SIP UA Config and Management - Basic Scenario and Use Cases

The basic scenario we use in this document is the one of an end-user, enterprise or service provider (an actor) wanting to configure and manage its SIP-based client and SIP-related application functionality by using available IETF protocols and existing element management tools (through various user interfaces: Command Line Interfaces, web-based views, or even management protocols via SNMP [RFC3411] for some categories of actors). The SIP client may be dual-stack IPv4 and IPv6 and may or may not be behind a home-based or network-based NAT: this special case is only secondary and invoked in the third use case.

We consider three use cases derived from this basic scenario:

- SIP client provisioning using XML-based configuration protocols with common management and data modeling techniques.
  The configuration of a SIP client is based on the use of IETF SIPPING Configuration framework ([ID-SIP-CFG]) and the IETF XCAP protocol ([ID-XCAP]) with the addition of commonly used MIB design methodologies using the Structure of Management Information (SMI) to solve the lack of XML data modeling. The use case illustrates the shift in client configuration methodology and elaborates on how SNMP configuration is evolving into XML-based configuration. We insist on some SNMP-based modeling techniques that could be of valuable to re-use for describing data elements for the purpose of management: especially but not exclusively when the SNMP protocol is used for management in conjunction with XML configuration.

- Management of access control for SIP client configuration based on SIPPING-config and XCAP based on XML data modeling:
  When using the IETF SIPPING configuration framework and XML-based configuration via XCAP, the actor should have flexible means to define the type of access on a particular object or XCAP target (read, write, execute, notify, etc.). Access control should also provide multiple access control levels for a particular object, similar to the UNIX access control lists: owner (user), group, others.

- Management of SIP devices behind NATs and firewalls:
  All types of actors want their SIP clients or devices to transparently operate behind well-behaved NATs and firewalls. The use case takes the specific instance of a service provider or an enterprise wanting to manage IPv4-IPv6 SIP devices using CLI or protocols like SNMP or HTTP over secure transport. We look at how management sessions for clients behind firewalls (and may be NATs) require from the use of SIP signaling stimulus.
The next sections provide more details on the three distinct use cases and indicate potential solutions based on IETF protocols.
3. Use Case 1: SIP client provisioning using XML-based configuration and data modeling

In this section, we consider a SIP client being provisioned using XML-based configuration protocols (IETF SIPPING configuration framework and IETF XCAP protocol). We investigate how re-using some of the commonly used MIB design methodologies may help to solve the lack of XML data modeling.

While the intent of the use case is generic, taking a specific example of cable service providers should help provide some context. This may also be applicable to other actors, enterprises and service providers in particular. The main point is not to insist on the use of SNMP but more importantly, to underline how engineers and implementers have used the SMI data model associated with SNMP to design configuration files.

Traditionally, devices in cable networks (e.g. DOCSIS(r) cable modems, PacketCable VoIP clients) have used SMI-based ([RFC1155],[RFC2578]) SNMP MIB modules as a means to represent both configuration and management data elements. Device management is performed using the IETF SNMP protocols and recommendations. The initial device configuration relies on a configuration file downloaded via TFTP or HTTP whose content includes TLV attributes (Type, Length, Value): each TLV in a typical client configuration file represents an SNMP variable binding: a configuration parameter and its initial value.

This process is illustrated in Figure X1. A cable device or client obtains its configuration file, populates the configuration parameters by 'internally' setting the corresponding SNMP MIB objects and proceeds to initialization of the supported services and is ready to be managed by element management systems based on SNMP.

Note that the configuration file was provided using TFTP (or HTTP) on a one-time basis (during bootstrapping) as opposed to using a configuration protocol that can support dynamic changes.
A couple of points are worth noting in the data modeling process:
- Modeling data using SMI to create a MIB module may not always be perceived as adequate, elegant or efficient by many, but there are
many design guideline documents available and it is used by the MIB designers in the IETF community. The experience gained over the years is invaluable;
- there are many tools available to check the syntax of MIB modules and enforce strict adherence to the SMI (e.g. boundaries for integer values, default values, formatting of the display of strings, re-usability of textual conventions facilitating interoperability and eliminating the parsing of free form strings, etc.);
- there are numerous data models built on SMI that would benefit from being available in the XML-based data modeling world in support of configuration or management.

Moving to the use case at hand, the increased client configuration requirements, the recognition that SNMP is rarely used for configuration (NETCONF), and the good work done by the IETF SIPPING working group on the configuration framework ([ID-SIP-CFG]) has led many to explore an XML-based configuration model.

The configuration of SIP-based clients and devices in this example is based on the configuration proposals available in the IETF for SIP User Agents (UAs) configuration, the eXtensible Markup Language Configuration Access Protocol (XCAP) is chosen as the configuration protocol of choice for this use case.

Client management is left unexplored by many proposals: it is not in scope of XCAP or not fully addressed by NETCONF. In this use case, we propose to use SMI for three main reasons:
1. to overcome the lack of XML data modeling for element management and configuration,
2. to leverage the guidelines and experience in doing configuration and management modeling using well-defined techniques,
3. to provide continuity for some actors that may manage devices using SNMP.

This may be where the use case becomes more specific but we believe that even without the use of SNMP, re-using SMI brings some clear advantages.

For these reasons, SMI-to-XML Schema transformations are accomplished to provide XML Schemas ([IR-SMI-XML]). This enables the same operation as shown in Figure X1, but supports a full-fledged SIPPING and XCAP configuration protocols and allows SNMP management (SNMP management means SNMP for monitoring here (e.g. GET), not for configuration (e.g. SET)).

Given the IETF does not have any standard XML-based data modeling proposals, the SMI-to-XML conversion mechanism seems to provide a stop-gap solution for supporting some element configuration. It is important to realize that folks who do not want to deal with SMI or ASN.1 will obtain the derived XML schemas. Rather than defining
those schemas from scratch, SMI is used as an intermediary step. This may be seen as short-sighted but in the short term, and potentially long-term, it may provide a viable solution.

![Diagram](attachment:image.png)

Figure X2: Configuration of SIP clients via XCAP using SMI for data modeling
In conclusion, the co-authors believe that a well-defined XML data model for configuration and management is required. While it has surfaced in some working groups (netconf [ID-NETCONF] for example), it is not available today. The long-term solution may be to define a new model in IETF or elsewhere but in the short-term, given the availability of SMI design guidelines and tools, and the experience that can be leveraged from the SNMP community on configuration data design, the IETF should continue the work initiated on SMI-to-XML to provide a standard means to perform quick and reliable syntax conversions.
4. Use Case 2: Access Control Definitions and XML-based configuration methodologies

Access control definitions allow data model designers to choose access rights and restrictions for the defined data elements. Most everyone is familiar with the UNIX ACL management associated with the file system: each file has read/write/execute access rights for the owner, group, and others. The View-based Access Control Model (VACM) for SNMP is also a good reference as it is applied to configuration ([RFC3415]).

In this use case, we look at the change in data ownership that is introduced by XML-based configuration models and protocols (XCAP as an example) and conclude that access control definitions for XML-based data models is a necessity. This use may be viewed as an extension of the first one and it also shows some areas that are not addressed by the SMI-to-XML conversions.

Use case description:
A SIP client is configured using the SIPPING configuration framework and XCAP. Changes in the configuration data elements may occur at multiple levels (write access): based on end-user settings on the SIP device, remote configuration changes (end-user accessing a web-based interface to change settings or a service operator updating the config), and many other means (software updates from device manufacturer changing some default configuration parameters, etc.). Furthermore, the presentation of some configuration data (read access) may also have to be controlled: not all users of a client may be allowed to change the settings, some settings may be hidden by the protocol stacks, device manufacturers or service providers so that end-users may not misconfigure the device and disrupt the service. Note that the mechanism by which the configuration change is not a factor in this use case: a configuration change may be operated via a custom user interface on the SIP device or application, an end-user or operator controlled CLI, remote web-based interface.

Problem statement:
How does the designer of the XML schema specify access rights for each data element and for each level (owner, group, others) in an interoperable manner?

To provide more context to the use case, we compare this problem statement with the SNMP or SMI-based model even though, as stated above, the problem statement is independent of SNMP. Clients with an SNMP agent store configuration data as instances of the SNMP MIB objects. Changes to such data by authorized network elements is accomplished via SNMP through EMS and NMS systems. Any change is authorized by the SNMP agent based on the SMI access control
This arrangement is suitable when dynamic configuration changes are minimal: the configuration data is stored on the client and the model works well when there is a single source of changes. There are examples of MIB modules that have been defined where an object was not writable via SNMP but since the object value could be changed during runtime or by the end-user: a note was put in the DESCRIPTION clause.

This is illustrated in Figure Z1.

---

<table>
<thead>
<tr>
<th>Network</th>
<th>Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Configuration Changes</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Client</td>
</tr>
<tr>
<td></td>
<td>(SNMP agent)</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Config Data</td>
</tr>
<tr>
<td></td>
<td>Configuration changes</td>
</tr>
<tr>
<td></td>
<td>&lt;------------------------&gt;</td>
</tr>
<tr>
<td></td>
<td>(SNMP Manager)</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Management Data</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure Z1: Example of Configuration and Management using SNMP

With the use of a configuration protocol like XCAP, this paradigm has shifted as illustrated in Figure Z2.
Figure Z2: Configuration and Management using various protocols

A separate configuration server (for example, the XCAP Server in the above figure) is responsible for the storage and modification of data. This configuration data is accessible by clients and authorized network elements alike.

The importance of access control can be illustrated by many examples: assume a data element that is accessible by both the client and a network element, but is modifiable only via the network element (e.g.: my speed dial list or favorite friends' URI list is modifiable via a web-based application talking to the "Network Element"). One would expect the XCAP server, the 'keeper' of the data, to enforce this access control requirement. Should this be expressed in the XML Schema?
Further, multiple data profiles can be built out of the same set of data elements, but with different access control settings. There may also be multiple entity types that share the same access control sets. All this hints at UNIX-like or VACM-like grouping of access controls.

Additionally, the fact that the same XML Schema definitions may call for run-time change in access privileges should also be taken into consideration. This may call for the ability to provide access control definitions internal or external to the XML data models (for example via XPATH).

In conclusion, as we hope this use case illustrates, the co-authors believe there is a need for standardization of Access Control directives to enable configuration and management in general, and XCAP in particular, to be effectively deployed at wide scale and for a wide range of applications.
5. Use Case 3: Management of clients behind firewalls (and NATs)

The issues of establishing SIP sessions behind NATs and firewalls have been extensively discussed in IETF in SIPPING, BEHAVE, MIDCOM and other groups. We focus on the establishment of a management session between a SIP client and a management station 'separated by a firewall (and NAT) device. Any command or management operation should be initiated by the client.

<table>
<thead>
<tr>
<th></th>
<th>Client</th>
<th>NAT-FW</th>
<th>EMS/NMS/web</th>
<th>EMS/NMS/web</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Device</td>
<td>Mgmt Station</td>
<td></td>
</tr>
</tbody>
</table>

| X<----------------- | Retrieve Operations |
|                   | (Management host initiated) |
|                   |                               |
|                   |                               |
|                   |                               |
|                   |                               |

<p>| X&lt;----------------- | Modification Operations |
|                   | (Management host initiated) |
|                   |                               |
|                   |                               |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Notifications</td>
</tr>
<tr>
<td></td>
<td>(Client initiated)</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------</td>
</tr>
</tbody>
</table>

Figure Y2: Management operations affected in the presence of firewalls (and sometimes NATs depending on the protocol used)

A management session may be established by the client for an undefined period of time (session stays up which could be resource intensive). For efficiency, it should be initiated as needed, based on a stimulus from the management host. For SIP clients, this solution may be made possible by using the same signaling mechanisms proposed in the SIPPING Configuration framework, for e.g. using the SIP SUBSCRIBE/NOTIFY framework. A client could subscribe to
management events and would get notified to initiate a management session. This is illustrated in Figure Y3.

---
<table>
<thead>
<tr>
<th>Client</th>
<th>NAT</th>
<th>EMS/NMS</th>
</tr>
</thead>
</table>
-------------        -------        ---------------
|                |                  |   Previously established SIP session (e.g. SIP SUBSCRIBE/NOTIFY) |
| <- - - - - - - |----------------->|   Notification (e.g. SIP NOTIFY) |
|                |                  |   Management Session establishment (CLI or SNMP over ssh, HTTPS, etc.) |
| <- - - - - - - >|-----------------|   Retrieve Operations (Management initiated) |
|                |                  |   Modification Operations (Management initiated) |
| <- - - - - - - >|-----------------|   Notifications (Client initiated) |
| - - - - - - ->|------------------|

Figure Y3 Initiating management sessions in the presence of firewall - NAT devices

In conclusion, the use case highlights the need to define a generic mechanism for establishing a client-initiated management connection.
when a SIP session pre-exists. The components of such a solution may include:
- defining how SIP can be used to establish a management session,
- defining a SIP event package for conveying the relevant management session parameters (IP and transport addresses of the management station, the type of transport protocol to use for management - HTTPS, SNMP over SSH, etc.).
6. Summary and Next Steps

In summary, this document presents three use cases for the configuration and management of SIP clients or devices using IETF protocols. The proposed approaches lead to following findings and suggestions:

- An XML data modeling specification for configuration and management is required. The experimental SMI to XML conversion methods provide an elegant, short term solution for some actors. It leverages the experience and tools for building modules for management purposes. It allow the reuse of some existing data models based on SMI with XML based configuration protocols like XCAP ([ID-XCAP]). There have been proposals in the past ([IR-SMI-XML]) that have been implemented, and, based on some prototyping experimentation, the co-authors of this document recommend that this proposal be revisited.

- XML-based configuration protocols can greatly benefit from a standardized access control definition methodology. This would foster effective deployment of XML based configuration protocols.

- Solutions developed in some OPS working groups should address the management of clients behind NATs and firewalls, so that IPv4 and dual-stack IPv4-IPv6 clients may be managed efficiently. An example specific to SNMP would be to invite proposals to the ISMS WG to help address the NAT and firewall traversal within the scope of modified transport ([ID-SNMP-SSH]) and security models ([ID-SNMP-TMSM]).

- The SIP community may benefit from defining solutions to enable client-initiated management connection when a SIP session exists. A new SIP event package may be used part of the solution ([RFC3265]).
7. Acknowledgments

The authors of this draft wish to thank members of the CableLabs PacketCable focus teams and various other IETF participants who contributed directly or indirectly to the content presented in this draft. Specifically, the following individuals are recognized: Josh Littlefield, Eugene Nechamkin, Paul Duffy, Thomas Clack, Harindranath P.R Nair.
We also thank Juergen Schoenwaelder and Frank Strausse for the email exchanges.
8. Security Considerations

FFS.
9. References

9.1. Normative References

[ID-SIP-CFG]

[ID-SNMP-SSH]

[ID-SNMP-TMSM]


[IR-SMI-XML]


9.2. Informative References

[ID-NETCONF]

December 2002.
Authors' Addresses

Sumanth Channabasappa
CableLabs
858 Coal Creek Circle
Louisville, CO 80027
USA

Email: sumanth@cablelabs.com

Jean-Francois Mule
CableLabs
858 Coal Creek Circle
Louisville, CO 80027
USA

Email: jfm@cablelabs.com
Full Copyright Statement

Copyright (C) The Internet Society (2006).

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 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 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 currently provided by the Internet Society.