Policy-Based Enhancements to the SNMP Framework

J. Saperia, JDS Consulting, Inc.

J. Schönwälder, TU Braunschweig

Technical Report 00-02

Computer Science Department
Technical University Braunschweig
Bültenweg 74/75
38106 Braunschweig
Germany

May 30, 2000

Abstract

This memo takes a look at some of the work on policy-driven networks currently being done in several IETF working groups. The proposed solutions (like COPS and PIBs) focus on solving local problems of policy management. It is the view of the authors that while local solutions like COPS can be more effective than more general ones, the advantages to effective management through the use of a single integrated management framework (an improved SNMP) outweigh the gains of locally optimized solutions.

This report is a slightly revised version of a position statement which was prepared for a two day interim IETF meeting organized by the Area Directors responsible for the IETF Policy Framework, the Resource Allocation Protocol and the Differentiated Services working groups. The meeting took place in Chicago in September 1999. The changes made to this document after the meeting are mainly of editorial nature (updated references, typos corrected, affiliations updated). This report thus reflects the thinking of the authors at the interim meeting in September 1999.

Contents

1 Introduction

With the addition of Differentiated Services and other technologies, the need for policy based management has grown. A number of working groups including the Policy Framework and Resource Allocation Protocol working groups needed support for the configuration of policy in the network. SNMP was evaluated by these working groups and in some cases and found lacking in a number of areas. As a result, new work was undertaken which led to COPS and PIBs among other things. These solutions focused on solving the local problems of policy management.

This document was prepared for a meeting involving some of the involved area directors and working group participants scheduled to take place in mid September 1999. The purpose of the meeting is to help the Area Directors gather information about the various proposals and the needs that they are to address. In addition, these needs will be evaluated against the features available in SNMP. This document has an additional purpose in addition to preparation for the meeting. It is background for a proposal to integrate the best features of COPS/PIBs and the best features of SNMP that is hoped will evolve from the Chicago meeting. Local solutions like COPS can be, if properly specified and implemented, more effective than general ones. However, the advantages to effective management through the use of a single integrated management framework (e.g., an improved SNMP) far outweigh the gains of locally optimized solutions.

2 Background

Over the years SNMP has evolved and most recently has undergone the addition of improved security and refinement of the specifications. In many important areas, such as the ability to retrieve large amounts of data and support complex configuration operations, many people felt that SNMP did not provide adequate solutions.

Given this view, people sought other solutions to fill in the gaps (real or perceived) that existed in SNMP.

The solutions range from formal protocol proposals like COPS to informal and proprietary solutions offered by vendors or implemented by network operations staffs. The result is a management environment that is fragmented. SNMP has been dominant in the areas of network status monitoring and statistics gathering, but has not been widely used for configuration management. This fundamental split makes problem resolution more costly and difficult.

The original focus of this draft was to have been cross referencing on how SNMP can meet the stated needs of the community. In the view of the authors, SNMP does need some enhancements in order to be effective in the policy area. These enhancements are not driven only by policy requirements since they can benefit all areas of management. We are advocates for effective management - not for a particular protocol. We felt that a good approach would be to work with the community to develop a clearly articulated list of requirements (which should not be to difficult), evaluate some of the principles that have been developed for COPS etc. and create a list of additions needed for SNMP. This is based on the belief that the Internet needs an integrated management approach to deal with the challenges of the coming years.

3 Statement of the Problem - Need for Integrated Management

It is a strongly held view by many that when configuration information (policy provisioning data is just one kind of configuration data) is in a different form than fault and other types of data, that management becomes more difficult. Imagine a network operator tasked with relating the failure of an interface which is the backup interface (in a sonet APS sense) to an SLA if the SLA has been expressed in a form which makes it difficult to associate the failed component. This can only be done when instance level information is known. This is not a dismissal of the powerful concept of classes found in the various framework and schema documents that have been published over the past few months. To the contrary, the concepts can be well used to augment the existing management framework. Using this approach a more integrated method of management can be achieved.

4 Requirements

For effective management to take place, there are a number of requirements relevant to the current discussion which can not be ignored. Below are some of the areas which any policy management paradigm must address to successfully integrate into currently deployed Internet management environments.

4.1 Multiple Management Stations

Multiple management systems exist today for concurrent access to managed elements for a wide variety of reasons. In modern networks service and performance interruptions must be kept to a minimum - this applies to the management infrastructure as well. As a result, multiple standby management components are often on hot stand-by and in full sync with their primary counterparts to avoid extra delay for connection re-establishment in the case of failure. These standby components are often geographically dispersed.

Beyond the basic issues of multiple redundant components, multiple management systems may need to be in communication with managed devices (often concurrently) for other reasons. In some cases, different organizations within the management community perform different configuration functions (e.g., routing, versus provisioning of the hardware). These different aspects are often related to policy.

4.2 Multiple Users in Multiple Roles

From the previous item, it should be clear that multiple users must have access to a single system - often concurrently. What is also true is that not all management personnel are given the same privileges with respect to access to information. Some people are allowed only read access to certain configuration information while others are permitted to have full access to a wider range of management objects inside the network elements. A restricted few have wide and full access. Even in these cases, there may be some systems within a particular administrative domain that have even more restrictive access. Policy/Configuration information is sensitive and management operations staffs require flexibility in the permissions given people in different roles.

4.3 Connection-oriented and Connection-less Management

A great deal has been written and tried over time for management with connection-oriented and connection-less approaches. What may be needed is a combination of the two since neither one meets all types of requirements. Connection-less polling for short term information is far less costly than maintaining many (potentially thousands) of connections for the collection of this type of data - connection-oriented management systems have always been problematic with regard to scale. Status polling for certain types of information also has some scale problems, so asynchronous notification is a viable scalable alternative.

Connection-oriented management for the retrieval of large amounts of data or the transmission of complex configuration information has proved to be of great value over time.

4.4 Hierarchically Managed Networks/Overlapping Domains

Networks of size must provide for some level of hierarchical management and overlapping management domains. Often management has a functional flavor where the WAN folks deal with part of the network, and the local people deal with another part of the network. To some extent, even in the policy area there will be overlap. Imagine an edge device which is subject to the policies of multiple administrative domains. Hierarchy is not always exactly straight up and down and it is often the case that policy does not have clean boundaries and systems must to some degree have multiple masters.

4.5 Policy and Instance Specific Information - Required Integration

The idea of saving data transfer to the PEP by avoiding the granularity of instance level data is a helpful concept. There is often some configuration information that is required on a per instance basis. A good example would be the IP address of an interface. It is essential that these two types of information, general policy rules/roles and specific instance related information be well connected — otherwise detecting and correcting configuration errors will be more difficult.

One of the side effects of requiring the PEP to convert to instance level data is that while the transmission of the data to the machine would be less resource intensive than if SNMP based instance level data were sent, there will be considerable additional processing needed to make the conversion. In addition it will require a global knowledge of the system and 'smarts' to make the conversion that are not currently available in many systems. Please also see next point.

4.6 Compatibility with SNMP Management Data

If a card or some other hardware or software component fails, it is important to tie this to the SLA or SLAs that are in jeopardy. In large systems especially this will require the tight integration of the information described in the various policy documents with other standard information. The failure will always be expressed in terms of instance specific information. In fact, only by evaluation of the instance specific information can a determination be made about the impact to an SLA.

Similarly, data about the utilization of various resources in the network is necessarily collected via SNMP based mechanisms. It will be impossible to make good policy decisions without this information. The conversion of this (SNMP) data to another format could be costly.

4.7 Multiple Configurations and Switching Configurations

Configuration management systems in general have to support the notion of multiple configurations. There are usually at least two configurations stored in networking devices: The currently active configuration and a configuration which is stored in stable storage and reloaded upon re-initialization.

Some devices contain additional configuration sets so that changes can be made without risking disruption to existing services and to allow operators to program automated switching between different complex configurations (e.g. based on the time of day). Such a switch between configurations may affect lots of devices and it should therefore be supported in an effective way.

4.8 Evolution of Configuration Management Data

Configuration management data models are not fixed for all time and are subject to evolution like any other management data model. It is therefore necessary to anticipate changes in the configuration data model and to provide mechanisms that can deal with changes effectively without causing interoperability problems or having to replace/update large amounts of fielded networking devices.

5 Issues with COPS/PIB Approach

This section lists some problems with the current COPS/PIBs approach. The focus here is on conceptual problems rather than technical details of the specifications.

5.1 Incomplete SPPI Definition

The COPS policy provisioning protocol [1] requires a new language called Structure of Policy Provisioning Information (SPPI) which is used to define concrete policy objects in a Policy Information Base (PIB). SPPI is defined by refering to the SMIv2 standard and making several changes as described in [2]. There are several problems with this approach.

First, the ad-hoc definition of SPPI leaves many issues unspecified. It is for example unclear whether the usage of the Counter32 SMIv2 base type is allowed in PIBs. It is also unclear how SMIv2 base data types with restricted SMIv2 access interact with the POLICY-ACCESS clause. Another example are the SMIv2 macros to describe compliance requirements. Are they also applicable to PIBs? Or is every implementation required to implement a full PIB even if some of the PIB objects do not make sense on a particular device?

These are only examples of questions that arise since there is no real definition of the Structure of Policy Provisioning Information (SPPI). Many more can be found be carefully analyzing the SMIv2 definition in light of the COPS/PIB approach to policy provisioning.

5.2 PIB Instance Identification

The SPPI and the COPS protocol requires that all instances are indentified by what is called a Policy Rule Instance Identifier (PRID). The POLICY-FRAMEWORK-PIB [3] introduces the PolicyInstanceId textual convention derived from Unsigned32 to identify policy rule instances.

It is at least unclear whether more complex PRIDs are legal. Simple-minded PRIDs introduce a need to map MIB instance identifiers, which are in many cases complex and meaningful, to PRID values used in PIBs. Furthermore, such a simple PRID identification schemes makes direct translation of PIBs into MIBs in many cases useless since table indexing in MIBs is usually optimized for typical retrieval operations and/or efficient access control.

5.3 Missing Contexts

The SNMP evolution has lead to the introduction of contexts which contain collections of MIB variables. Contexts turn out to be a very useful thing in order to handle multiple instantiations of MIB trees, which would map to multiple configurations in the policy provisioning scenario.

The COPS/PIB approach is missing a similar mechanism. Instead, each instance of a rule class can only exist once. It is therefore not possible to easily describe and manipulate multiple configurations, which makes policy changes at for example a particular time during a day costly. In this scenario, COPS/PIBs require to remove all policy data from all PEPs and to install new policy data on them.

5.4 Evolution of PIBs

SMIv2 MIBs and the SNMP protocol have the important property that individual definitions can be added, updated or removed without having to deprecate complete tables. This allows for a smooth evolution of SMIv2 MIBs and experience with for example the Interfaces MIB [4] shows that changes are unavoidable.

The PIBs in contrast do not support such an evolution of PIBs. The reason is basically that the install/remove/notify operations are bound to a PIB class (which corresponds to a MIB table) and that the COPS-PR protocol only transfers a sequence of values. Thus, changes like the addition of an element to a class (table) requires to completely redefine the whole table.

Furthermore, it can be expected that some implementations will not support all elements of a class (table) definition. There is no mechanism to deal with this situation in an explicit way. Hence, a PEP which does not support all elements of a class (table) still has to conform to the interface and it has to provide/accept dummy values during notify or install operations. This can lead to serious problems in troubleshooting situations since dummy values are hard to identify.

5.5 Configuration Management for exisiting MIBs

The COPS/PIBs proposal can not be directly applied for policy provisioning with existing MIBs. A good example are SNMP access control tables [5] or IP forwarding tables [6] for which well known MIB definitions do exist.

The fact that there are two different languages to describe the same information seems like a costly way to operate networks and it will lead to duplication of work within the IETF. It will also lead to more complexity in typical implementations.

5.6 Manual Translations between PIBs and MIBs

There is no algorithm that can automatically convert a PIB into a MIB. There are also no provisions to leverage existing MIBs and to make them available for more efficient configuration management without having to manually redefine the MIB as a PIB.

The authors believe that it is desirable to have a single and coherent language and format for defining management information. In particular, it should be possible to use the same definition for monitoring and configuration management purposes. The introduction of multiple languages with manual translations between them introduces costs and another source of possible inconsistencies.

6 Possible Changes to SNMP

In this section, we will briefly outline changes to the SNMP Framework that will enhance the SNMP Framework to support Policy-Based management. Some of the changes listed below are actually being worked on for other reasons, mainly to improve the efficiency of bulk data transfers.

6.1 SMIv2 Extensions to Define Operations

The SMIv2 should be extended to allow the definition of operations on collections of managed objects (typically on table rows). A definition of an operation should include the operation signature, which includes the operation name (a registered OID value), the arguments of the operation (a list of object-types), the results produced by the operation (a list of object-types) and operation specific exceptions. It is important to reduce the degree of freedom that currently exist with the SNMP set operation. Furthermore, it is necessary that SMIv2 defined operations can be mapped to APIs in frequently used implementation languages automatically in order to make the development of (configuration) management applications more efficient and cost effective.

An SMIv2 extension for operations will allow the definition of install/remove/notify operations on table rows for existing as well as future MIB modules. Care must be taken to allow proper versioning of operation definitions since MIB definitions (such as conceptual table) can and will change over time.

6.2 SNMP Transaction Support

There is a need for enhanced transaction support across sequences of simple SNMP operations. In its simplest form, a transaction will be made up of a sequence of SNMP operations targeted at the same SNMP entity.

6.3 Transport over TCP

An SNMP over TCP transport mapping is needed in order to enable more efficient bulk data transfers and to ease the implementation of larger transactions. A document defining SNMP over TCP is under development by the Network Management Research Group (NMRG) [7].

It is important to realize that an SNMP over TCP transport mapping complements the SNMP over UDP transport mapping and does not replace it.

6.4 Additional Protocol Operations

The SNMP protocol could be extended with additional PDUs if necessary that realize operations on collections of object-types. This minimally requires the introduction of a new "CallOperation" PDU. Support for larger transactions may require the introduction of additional PDUs unless one uses other mechanisms for transaction handling (such as contexts or transport system mechanisms).

6.5 Efficiency Improvements

Bulk data transfers are not very efficient with SNMP since the payload contains large amounts of redundant OID fragments. This is caused by common OID prefixes and instance identifiers in the payload. A compression scheme is one potential solution to increase the bandwidth efficiency of SNMP. Such a mechanism may have advantages over alternate encoding schemes since the receiving application still has access to full instance-level names rather than just a sequence of values. This provides for robustness since ordering or omission problems in the encoded values, which may be caused by legal changes to the MIB definition or simply implementations that support only subsets of the standardized configuration information, can be detected and dealt with.

7 Comments on draft-kzm-policy-protocomp-00.txt

This section comments on the document [8] which was intended as a comparison between COPS and SNMP. The introduction read in part, "it has been requested that a comparison be undertaken as to why COPS is better suited to this than a (modified if necessary) version of SNMP."

At first glance, it seems to be well written and appears to make a number of good points. However, upon closer examination, several of the points made in that document deserve further scrutiny in order to arrive at a more balanced view.

7.1 Section 3.3 Exclusive Access by the PDP

An early major point in the protocol comparison document [8] is that different requirements have led to different design choices for COPS/PIBs vis-a-vis SNMP/MIB. One of these requirements differences is with regard to the “single PDP” versus “multiple manager” dimension and the design choices that spring therefrom.

The COPS document considers the possibility of multiple PDPs operating on a system which had multiple sets of policy. It was not made clear how this can or should/will work within a single managed system.

7.2 Section 3.5 COPS Takes Advantage of TCP

A second major point is that of choice of transport: TCP versus UDP. It has a number of sub-points.

The first is maximum message size. The document contrasts the maximum message size of a COPS object (64KB) with the minimum size an SNMP implementation must support (484 bytes). It fails to acknowledge that the maximum message size of an SNMP message is also 64KB (actually virtually unlimited – limited to 2 billion varbinds and the practical limit comes from the buffering capability of the underlying operating system and maximum message size of the transport which goes up to 64KB for UDP). The document does correctly state that messages this large are rare. One could quibble over the stated value of 1500 bytes but it is not of importance for this discussion.

One reason this is of no importance is that the SNMP-based framework allows a large transaction to be broken into groups of multiple, smaller, transactions. Each transaction-let can be transported independently and then made to go active simultaneously. Not only can all columns of an entire row be made active simultaneously, multiple rows can be made active simultaneously. This is a MIB and application design issue, not a protocol issue, as it properly should be. The related portions of policy can be conveyed independently and made active simultaneously, hence SNMP-based management can be used to convey policy information (or any other information) without concern for the maximum message size. Operational experience indicates that it is possible to move a management script consisting of multiple tens of kilobytes via SNMP/MIB and then to activate the script. Although it is not suggested that SNMP will replace FTP over TCP, the applicability to moving other management information, such as policy configuration data, is obvious.

Section 3.5 talks about the problems with incremental creation of a read-create table and how atomic access is easier to implement and test. In many of today’s modern SNMP implementations, the code to implement incremental and atomic row creation and deletion are a part of a subroutine library and the grungy details are isolated from the developer who uses it and who may not be an SNMP expert. As a result of wrapping a powerful API and automatically generated code around the row creation, implementation and test is not as difficult as represented and movement to COPS’ atomic creation does not yield the stated advantages.

Finally, with regard to incremental creation, there is an additional important point worthy of discussion. One reason for the ability to handle incremental creation is that there is a possibility of version skew in the version of the information known by the sender and the information known by the receiver as would occur because of evolution of the PIB, such evolution to be expected. COPS/PIB needs a robust mechanism for dealing with this evolution.

Another point made in the discussion of TCP versus UDP is that COPS can afford to rely on TCP because the design choices for COPS assume that the network is up and working. SNMP’s typical use of UDP is an appropriate one because research and field experience have shown that UDP, with an appropriate application layer retransmission strategy is more robust, and provides a lower mean time to conduct a transaction, than does TCP with it one-size-fits-all retransmission algorithm. This is particularly true when the network is at or near congestion collapse.

It seems obvious that if you ever want SNMP to work, you want it to work when the network is at or near congestion collapse, and the choice of a UDP transport with an intelligent application-layer retransmission algorithm would be an appropriate design choice.

Similarly, it seems obvious that if you ever want policies to work, especially RSVP and diffserv-related policy, you want it to work when the network is at or near congestion collapse, and the choice of a UDP transport with an intelligent application-layer retransmission algorithm would again be an appropriate design choice. Such a choice would be important in minimizing the mean time to complete those transactions that could be used to bring the network back from the brink of congestion collapse.

The protocol comparison document [8] failed to make the critical distinction between transport reliability obtained through retransmission and transactional integrity.

A third major point is that SNMP operations are more complex than necessary for COPS/PIBs.

In order to make this point, the authors use an example from the User-based Security Model (USM) for creating a row in a user table, along with appropriate secrets. Aspects of this table that are not explained in the document are that the creation of secrets on the new user comes from a manipulation of the keys of a user being "cloned" from. These operations allow the creation of a new user and new secrets to be done in the clear, but without disclosing either the new or the old secrets. Consequently, neither the old nor the new secret can be read, and the operation must be idempotent. The operation must be performed at least once and at most once, therefore exactly once even during periods of network instability. Therefore, this is fundamentally a harder problem than typically encountered.

Furthermore, sometimes writers of specifications intentionally elect to write algorithms in "simpleminded" baby steps in the interest of reader understanding rather than implementation efficiency. This algorithm is such an example. As a result, it is inappropriate to compare a COPS-based approach with no exception handling to a non-optimized approach with exception handling. A better comparison would be to document the steps one would use with a COPS-based approach to perform an idempotent operation and to be able to determine if the operation succeeded or failed in the presence of a race condition between the loss of the management connection and the conclusion of the transaction/sending of the response.

Of course, it is possible that a COPS proponent might say that it is not necessary or appropriate to perform operations that must be conducted exactly once. If that is the case, then it is not appropriate to compare with an example that achieves that which is unnecessary.

7.3 Section 3.7 COPS over TCP (Revisited)

Section 3.7 claims that the use of COPS over TCP is different from and superior to SNMP/MIB with respect to the volatility of provisioned policy information. The document states that COPS over TCP can invoke an expire timeout on provisioned policy information after loss of communications with a PDP. The expire timeout information comes as a part of the policy information.

Sometimes the protocol comparison document [8] uses "SNMP" to mean the entire SNMP-based Internet-standard Management Framework (including the SMI, MIB, and Protocol with security and administration) and at other times uses "SNMP" to mean merely the protocol portion of the management framework. This is unfortunate and sometimes misleads the reader.

The section goes on to say that SNMP is different whereas in reality it is the same. In the SNMP-based management framework, the volatility of MIB information to be stored, as the document states, is communicated along with the MIB information. Having volatility information or timer information in MIB data which could be directly applied to expiry rules is neither new nor unusual. One of the oldest examples is ipRouteAge, which dates back to MIB I and 1988. More recently, it has become customary for MIB documents to use the StorageType textual convention, which designates data to be volatile, non-volatile, permanent, etc. It is obviously easy to use a similar textual convention to associate a time-based MIB object which conveys expiry information with MIB-based policy data in an manner that is exactly analogous to that of COPS over TCP. In this case, the expiry would be invoked when communications between the PDP and PEP were lost.

Contrary to the claim, SNMP is no different and COPS over TCP offers no advantages in this regard.

7.4 Section 3.8 Modifying Message Formats

Section 3.8 makes the point that COPS messages use a TLV format, making COPS superior to SNMP. SNMP similarly uses a TLV format.

Section 3.8 also speaks to the fact that in SNMP the varbindlist in a response is identical to that of the corresponding request. This is an important design feature. This feature was added in order to insure that systems (including managers and agents) designed around the SNMP-based management framework were able to attain transactional integrity. It is important to distinguish between transactional integrity and transport reliability.

7.5 Section 3.9 Initiation by the PEP

It is interesting to note that the document compares apples and oranges in that it contrasts the COPS approach wherein the PEP, such as a router, querying the PDP, i.e., pulling the policy data. However, in the SNMP approach, it has the PDP doing set requests on the PEP, i.e., pushing the policy data. Section 3.9 dismisses this approach without a fair analysis.

The SNMP protocol has had, since its invention, a mechanism to send unsolicited notifications. SNMPv3 includes a confirmed operation to submit notifications. It is therefore very well possible that a PEP can initiate a configuration over SNMP in much the same way a configuration can be initiated using COPS by sending a notification to the PDP which in turn installs the relevant MIB objects on the PEP. The statement that a PEP router likely requires a command generator for this purpose seems to ignore this possibility and may mislead readers.

The analysis fails to consider the use of logical contexts to sort out the subset of relevant policies for a given router and throws in SQL as a red herring. According to the last paragraph of the section, the COPS approach is for the PEP to indicate its capabilities and status to the PDP. The PDP then sends the relevant policies via COPS. It is equally practical for the PEP to use an SNMP/MIB-based approach to indicate its capabilities and status to the PDP and for the PDP to provide the relevant policies via SNMP/MIB.

7.6 Section 3.10 Deleting Policy

Section 3.10 rightly states that MIB objects can be used to delete multiple instances to be deleted and that RowStatus is a common mechanism for deleting a particular row. If there is a need for a mechanism to delete all rows in a table, definition and implementation is done through the MIB, i.e., this is a MIB definition issue. The fact that objects to delete all rows in a table are rare is simply a reflection of the fact that the requirement for such a capability has been rare.

7.7 Additional Sections

Later sections of the document discuss hypothetical scenarios. During the meeting we can discuss a wide range of opportunities for modification of SNMP and integrating value add features from

COPS/PIBs.

8 Conclusions

The publication of COPS as a proposed standard without consideration of how that standard is to relate to the IETF standard management protocol, SNMP, will have a number of undesired effects. This would be particularly unfortunate if the benefits of the COPS/PIB approach were not as significant as claimed in [8]. The undesired effects include fragmenting focus and resources in the IETF, vendors and customers, and adding confusion in an already difficult area. For example, how does the policy information in COPS relate to the instance based fault information which is widely deployed using SNMP?

The already difficult problem of management will be made more so if we start configuring policy with COPS/PIBs and do other management with SNMP. Some excellent work has taken place in the Policy area. Some of the COPS ideas, and a good deal of the work described in the Policy Framework and other documents are very helpful - they do not require a new protocol. Implementing these ideas with SNMP where changes are necessary to support them would result in a unified management environment utilizing the best features of both.

In the end it is necessary to read performance information via SNMP. Policy information can be read by SNMP. Configuration of device instance specific information via SNMP is and will continue to be done. The only item remaining is configuration of policy based information. It makes sense to evolve SNMP so that it can meet this important requirement.

The differences between the COPS/PIB approach and the SNMP/MIB approach are not as great as presented. The advantages of COPS/PIB over SNMP/MIB do not outweigh the importance of a single, consistent, and coordinated management approach which is complementary to the vast installed base of SNMP-based management.

A Suggested Plan of Action

Many of the best proposals are the result of a concentrated effort by interested and involved individuals. We recommend that this approach be used in this case. Realizing that time is important and that pragmatic trade-offs are necessary the following is the recommended course of action:

1. Within a short time of the meeting the involved area directors and working group chairs should reach agreement on details with regard to the participants. That is, they will have selected (or be in the process of selecting) a small group of individuals representing many disciplines. This approach has been successfully employed many times in a variety of circumstances in the IETF.

2. The first step will be the laundry listing of policy related features which are deemed mandatory. Input for this effort will naturally come from COPS/PIBs and much of the other related work that has been done. This work should be completed in time for the IETF in Washington.

3. The Area Directors and Working Group Chairs should work together to make good presentations to the working groups involved so that the community can add anything to the requirements.

4. Within 2 weeks after the close of the IETF, requirements gathering should close. The closure of this phase is not a prerequisite to starting the other phases of the project listed below.

5. After the requirements have been identified, the changes necessary to SNMP should be isolated.

6. These changes should then be integrated into a revised set of the impacted SNMP documents and put out for a review about 1 month prior to the Spring IETF.

7. Issues can be discussed at the Spring IETF and then closed on the mailing lists within 4 - 6 weeks after the meeting.

References

[1] K. H. Chan, D. Durham, S. Gai, S. Herzog, K. McCloghrie, F. Reichmeyer, J. Seligson, A. Smith, and R. Yavatkar. COPS Usage for Policy Provisioning. Internet Draft <draft-ietf-rap-cops-pr-02.txt>, Nortel Networks, Intel, Cisco Systems, IPHighway, Extreme Networks, March 2000.

[2] K. McCloghrie, M. Fine, J. Seligson, K. Chan, S. Chan, A. Smith, and F. Reichmeyer. Structure of Policy Provisioning Information (SPPI). Internet Draft <draft-ietf-rap-sppi-00.txt>, Cisco Systems, Nortel Networks, Intel, Extreme Networks, IPHighway, March 2000.

[3] M. Fine, K. McCloghrie, J. Seligson, K. Chan, S. Hahn, A. Smith, and F. Reichmeyer. Framework policy information base. Internet Draft <draft-ietf-rap-frameworkpib-00.txt>, Cisco Systems, Nortel Networks, Intel, Extreme Networks, IPHighway, March 2000.

[4] K. McCloghrie and F. Kastenholz. The Interfaces Group MIB using SMIv2. RFC 2233, Cisco Systems, FTP Software, November 1997.

[5] B. Wijnen, R. Presuhn, and K. McCloghrie. View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP). RFC 2575, IBM T. J. Watson Research, BMC Software, Cisco Systems, April 1999.

[6] F. Baker. IP Forwarding Table MIB. RFC 2096, Cisco Systems, January 1997.

[7] J. Schönwälder. SNMP over TCP Transport Mapping. Internet Draft <draft-irtf-nmrg-snmp-tcp-04.txt>, TU Braunschweig, April 2000.

[8] K. McCloghrie and M. Fine. A Comparison of Policy Provisioning Protocols. Internet Draft <draft-kzm-policy-protcomp-00.txt>, Cisco Systems, October 1999.

Technische Universität Braunschweig
Informatik-Berichte ab Nr. 96-04

96-04	J. Adámek, J. Koslowski, V. Pollara, W. Struckmann	Workshop Domains II (Proceedings)
96-05	F.-J. Grosch	A Syntactic Approach to Structure Generativity
96-06	E. H. A. Gerbracht, W. Struckmann	Zur Diskussion elementarer Funktionen aus algorithmischer Sicht
96-07	H.-D. Ehrich	Object Specification
97-01	A. Zeller	Versioning Software Systems through Concept Descriptions
97-02	K. Neumann, R. Müller	Implementierung von Assertions durch Oracle7-Trigger
97-03	G. Denker, P. Hartel	TROLL – An Object Oriented Formal Method for Distributed Information System Design: Syntax and Pragmatics
97-04	F.-J. Grosch	M - eine typisierte, funktionale Sprache für das Programmieren-im-Grossen
97-05	J. Küster Filipe	Putting Synchronous and Asynchronous Object Modules together: an Event-Based Model for Concurrent Composition
97-06	J. Küster Filipe	A categorical Hiding Mechanism for Concurrent Object Systems
97-07	G. Snelting, U. Grottker, M. Goldapp	VALSOFT Abschlussbericht
98-01	J. Krinke, G. Snelting	Validation of Measurement Software as an application of Slicing and Constraint Solving
98-02	S. Petri, M. Bolz, H. Langendörfer	Transparent Migration and Rollback for Unmodified Applications in Workstation Clusters
98-03	M. Cohrs, E. H. A. Gerbracht, W. Struckmann	DISKUS - Ein Programm zur symbolischen Diskussion reeller elementarer Funktionen
98-04	C. Lindig	Analyse von Softwarevarianten
98-05	Gregor Snelting, Frank Tip	Reengineering Class Hierarchies Using Concept Analysis
98-06	Juliana Küster Filipe	On a Distributed Temporal Logic for Modular Object Systems
98-07	J. Schönwälder, M. Bolz, S. Mertens, J. Quittek, A. Kind, J. Nicklisch	SMX - Script MIB Extensibility Protocol Version 1.0
98-08	C. Heimann, S. Lauterbach, T. Förster	Entwurf und Implementierung eines verteilten Ansatzes zur Lösung langrechnender Optimierungsprobleme aus dem Bereich der Ingenieurwissenschaften
99-01	A. Zeller	Yesterday, my program worked. Today, it does not. Why?
99-02	P. Niebert	A Temporal Logic for the Specification and Verification of Distributed Behaviour
99-03	S. Eckstein, K. Neumann	Konzeptioneller Entwurf mit der Unified Modeling Language
99-04	T. Gehrke, A. Rensink	A Mobile Calculus with Data
00-01	T. Kaiser, B. Fischer, W. Struckmann	The Modula-2 Proving System MOPS
00-02	J. Saperia, J. Schönwälder	Policy-Based Enhancements to the SNMP Framework