Network Working Group
Internet Draft
Document: draft-ietf-MIDCOM-requirements-01.txt
Expires: November 2001
Category: draft

R P Swale
BTexact Technologies
P A Mart
Marconi Communications
P Sijben
Lucent Technologies
May 2001

Middlebox Control (MIDCOM) Protocol Architecture and Requirements

< draft-ietf-MIDCOM-requirements-01.txt >

Status of this Memo

Abstract

Networks connecting to and forming part of the Internet are increasingly deploying specialised network functions to address security, quality of service and other administrative policy requirements. These devices are a subset of what can be referred to as 'Middleboxes'.

While certain applications may operate unaffected in this environment, other types of application – or certain operational scenarios of a given application – may either fail to work completely or malfunction in some way. These classes of session-oriented application are typified by the use of separated data transfer and control sessions. Existing solutions to these problems deploy application specific intelligence within individual Middleboxes. However this leads to many difficulties including poor scalability of the solution and problems in reusing generic Middlebox functionality. In addition, trusted third parties are also increasingly being asked to make such policy decisions on behalf of the various entities participating in an application's operation. A need has therefore emerged for session-orientated applications to be able to express their communications needs to the devices in the network that provide enforcement of transport policy. This document therefore focuses on the requirements for the protocol between the requesting device and

the Middlebox and any associated policy entity. It is desirable that existing IETF protocols be used to implement these requirements where practical to do so.

Table of Contents

1. Introduction

A need has emerged for session-orientated applications to be able to express and communicate their needs for communications to devices in the network that provide, or represent, enforcement of transport policy. Examples of these devices include:

- firewalls,
- network address translators,
- intrusion detection systems (e.g. signature management)

These devices are a subset of what can be referred to as 'Middleboxes'. To enable the end-to-end operation of session-orientated applications in this environment, trusted third parties have become required to make policy decisions on behalf of the various entities participating in an application's operation. This document focuses on the requirements for the protocol between the requesting device and the Middlebox and any associated policy entity. It is desirable that existing IETF protocols be used to implement these requirements where practical to do so.

1.1. Background

For a variety of reasons, Middlebox devices effectively delineate domains of administrative policy. Such policies may include translation of IP addresses (e.g. NAT) or admission control (e.g. firewalling). Examples of this are where a firewall may be used to delineate the security policy of an end user's LAN from a service provider's wide area network link to the Internet.

While certain applications may operate unaffected in this environment, other types of application – or certain scenarios of a specific application – may either fail to work completely or malfunction in some way. Some of these examples are illustrated in [MIDBOXSCEN].

Session orientated applications represent an example of one class of application that encounters deployment difficulties in networks involving Middleboxes. This is because there are implicit relationships between individual packets that must be maintained between hosts across the network. One solution that has emerged to this problem has been to develop proprietary extensions (known as an Application Level Gateway) for a given Middlebox that enable it to handle a specific application. Unfortunately while this works for a small set of simple applications, it does not scale well. Neither can this approach easily handle increasingly complex scenarios such as those associated with multimedia applications. This is because these typically involve a multiplicity of media and control streams that originate from a single host but may have individual packet streams terminating on a diverse set of hosts.

Trusted third parties are therefore increasingly being asked to make policy decisions on behalf of the various entities participating in an application's operation. Consequently a need has developed for applications to be able to express their needs to the devices in the network providing transport policy enforcement.

Decomposing applications requiring policy decisions by removing application logic from the Middlebox and instead providing a generalized communications interface (by use of the 'MIDCOM Protocol') provides a number of benefits, including:-

- improved performance,
- lower hardware and software development costs
- lower maintenance costs,
- improved ability to support traversal of packet filters by complex protocols,
- easier deployment of new applications, and
- the ability to consolidate management functions.

For example, by moving stateful inspection of protocols such as H.323 and SIP out of firewalls, it is possible to improve performance and scalability while reducing development time and costs.

1.2. Scope

This document presents a discussion of requirements for the MIDCOM protocol.

The network environment envisaged for MIDCOM includes one or more Middleboxes in the data path between communicating clients, servers or peers as described in [MIDBOXFRAME] and [MIDBOXSCEN].

This document will only deal with needs arising from the use of firewalls and network address translation although the solution must enable extension to encompass other types of device in the future.

This document may form the basis of subsequent protocol development or the development of profiles of existing protocols.

The discovery of Middleboxes in the path of an application instance and communications between Middleboxes is outside the scope of the MIDCOM protocol and consequently this document.

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 [1].

The terminology describing Middleboxes used in this document is introduced in [MIDBOXFRAME].

3. Definitions

Flow	a sequence of IP packets communicated from one host and port to another host and port.
Pin hole	An association established across a Middlebox enabling a flow to be established that would otherwise have been prevented by normal or default behaviour of the Middlebox.
Application Control flow	A flow representing the command and control aspects of an application.
Application Content flow	A flow representing the transfer of the information or media associated with an application.

Application Content may include voice, video, data or other application specific media.

4. Specific functions assumed

The MIDCOM environment [MIDBOXFRAME] involves three main entities; the Middlebox, the MIDCOM Agent and the Policy Server as shown in figure 1. The following functionality is assumed within these components from the view point of the MIDCOM Protocol.

```
                          +-----------+    +--------------+
                          |  MIDCOM   |    |    Policy    |
                          |   Agent   |    |    Server    |
                         +-----------+    +--------------+
                           ^     ^                ^
     Application           |     | MIDCOM         | Policy protocol
       Control             |     | Protocol       |
        Flow               |     |                |
                           |     |                |
                           |     |                |
                           v     v                v
+----------+             +-------------------------+      +---------+
|  Host    |             |        Middlebox        |      |  Host   |
|          |<------->|          Device         |<----->|         |
+---------+              +----------------------+       +-------+
```

Figure 1. The MIDCOM Protocol and Environment

4.1. Middlebox functions

As described in [MIDBOXFRAME], a Middlebox may provide a number of processing functions for IP packets traversing it. Essentially, a Middlebox forwards packets from one of its physical interfaces to another. In doing so it may manipulate the headers of IP packets as requested by a MIDCOM Agent. However the Middlebox must not alter packet information unless it appears in the packet header. This is to ensure that there is an appropriate separation of concerns between the MIDCOM Agent and the Middlebox.

It is assumed that the functions provided by a Middlebox are externalised from a control point of view to a MIDCOM Agent through the use of protocol(s) meeting the requirements of the MIDCOM protocol.

The Middlebox must support the explicit forwarding of application control session information received on one, or more, destination address/port combinations to an appropriate MIDCOM Agent(e.g. an application gateway) as described in section 7.

A Middlebox may establish whether the association with a specific MIDCOM Agent should be allowed via a Policy Server. The Middlebox may also communicate with a Policy Server to determine whether the actions requested by a specific MICOM Agent are permissible.

4.2. MIDCOM Agent functions

A MIDCOM Agent, as described in [MIDBOXFRAME], embodies application specific intelligence external to a Middlebox. In conjunction with an associated Middlebox, a MIDCOM Agent enables specified end-to-end flows between application clients and/or servers in accordance with the needs of the application in question.

Control information is passed between a MIDCOM Agent and associated Middlebox by the protocol(s) meeting the requirements of the MIDCOM protocol.

4.3. Policy Server functions

The MIDCOM environment allows for a Policy Server function to externalise policy decisions from a Middlebox. A Policy Server is a management entity interfacing with Middlebox functions to establish policies concerning authorization of MIDCOM agents requesting access to the Middlebox and its resources. Its purpose may determine the flows supported across one or more Middleboxes.

A MIDCOM agent may be pre-configured on a Middlebox as a trusted entity or the trust relationship may established dynamically. In the case where a MIDCOM agent is not pre-configured, the Middlebox will need to consult a Policy Server to validate requests from the agent.

A policy server may add or remove the subscription of individual MIDCOM agents registered onto a Middlebox.

The protocol facilitating the communication between a Middlebox and a Policy Server need not be part of the MIDCOM protocol.

4.4. Distribution of MIDCOM elements

To enable the solution to scale, it must be possible for the functional components within the MIDCOM framework to be distributed.

The Middlebox therefore needs to support multiple simultaneous MIDCOM Agents. Each Agent may be acting on behalf of one or more applications. A specific Middlebox should not assume any relationship between MIDCOM Agents.

5. General Protocol Requirements

The MIDCOM protocol needs to enable a MIDCOM Agent requiring the services of a Middlebox to establish an association between itself and the Middlebox for the purposes of requesting pin-hole services from the Middlebox and obtaining statistics and other reporting information.

The MIDCOM protocol needs to address three distinct aspects:-

- The association of MIDCOM Agents and Middleboxes
- The establishment of an Application Control flow in conjunction with the Middlebox and an appropriate MIDCOM Agent
- The establishment of an Application Content Flow through a Middlebox according to rules determined by an associated MIDCOM Agent

The syntax and semantics of the MIDCOM Protocol shall be independent of the application(s)requesting Middlebox services.

The syntax and semantics of the MIDCOM Protocol need to be extensible to allow the requirements of future applications to be adopted.

The MIDCOM Protocol should allow the aggregation of commands, requests and responses.

5.1. Application based requirements

The MIDCOM protocol supports control-orientated interactions between a Middlebox and a requesting entity known as a MIDCOM Agent [MIDBOXFRAME]. Applications may require:-

- Translation of IP Addresses and/or port numbers from one side of the Middlebox to the other and knowledge of the specific translations used.
- Specific assured bandwidth from one side of the middebox to the other, as determined by packet size and transmission frequency.

The MIDCOM protocol needs to enable a MIDCOM Agent to express the direction of application packets. This must allow, for example, a MIDCOM Agent to specific that a flow will traverse a Middlebox from its inside realm to its outside realm.

5.2. Pin-hole requirements

5.2.1. Pin-Hole specification

In conjunction with an appropriate MIDCOM Agent, an application shall be able to request a pin-hole through a Middlebox. To achieve this, the MIDCOM protocol needs to enable the MIDCOM Agent and Middlebox to work together to support the desired flow.

A Pin-Hole needs to be identified by a suitable Pinhole-Descriptor which may uniquely identify a flow denoted by the tuple of FlowDirection, SourceAddress, DestinationAddress, Protocol, SourcePort and DestinationPort. The attributes associated with a Pinhole-Descriptor may be specific to a given Middlebox or MIDCOM Agent.The MIDCOM Protocol should also support the concept of an aggregated Pinhole-Descriptor comprising a multiple of individual flows to be treated as an aggregate.

To enable a Middlebox to reliably establish a pin-hole at the request of a MIDCOM Agent, the MIDCOM Agent must express the nature of the pin-hole required to the Middlebox. The MIDCOM Protocol shall allow the MIDCOM Agent to describe the desired behaviour of the Middlebox for the purposes of a requested flow(s).

The requests communicated between a MIDCOM Agent and a Middlebox need to permit pin-holes to be specified in terms of packet size (maximum and minimum length, inter-packet arrival time and other parameters including:

For IPv4: source and destination IP address or group of them determined by a netmask, protocol number, TOS field

For IPv6: source and destination IP address or group of them determined by a netmask, next header fields (Note: multiple fields may need to be traversed until a match is found)and traffic class field

UDP: source and destination port numbers or group of them

TCP: source and destination port numbers or group of them, "SYN packets" permission

ICMP: type and code

IGMP: type

The protocol shall permit the expression of direction to be associated with a pin-hole. The direction shall be specified in terms that apply to external view of the Middlebox. This directionality shall be expressed as "in", "out" or "Loopback" (meaning both "in" and "out" of the same side of the same Middlebox realm).

The protocol shall enable the association of a "Packet modifier" with a pin-hole to describe the required re-writing of header fields of accepted packets. The "Packet modifier" must be able to change the following protocol header fields

IPv4: IP addresses, TOS field

IPv6: IP addresses, traffic class field

UDP: port numbers

TCP: port numbers

Note: If modifiers are used, packet checksums must be recalculated.

5.2.2. Pin-hole identification

The Middlebox must be able to relate flows across it to requests for pin-holes from specific MIDCOM Agents.

5.3. Pin-hole operations

The MIDCOM protocol needs to support two basic operations; the opening an closing of pin-holes across a Middlebox.

5.3.1. Open Pin-Hole

An operation is required that enables a MIDCOM Agent to explicitly request a specified pin-hole be established across a Middlebox.

If the Middlebox is able to support a Pin-hole request from an authenticated MIDCOM Agent, it needs to provide the MIDCOM Agent with information that may be used to identify the flow in subsequent operations.

5.3.2. Close Pin-Hole

An operation is required to enable a MIDCOM Agent to tear-down an active pin-hole it has requested.

A pin-hole may be closed such that ICMP reporting for subsequently discarded packets is either enabled or disabled.

Where the communication relationship between a Middlebox and a MIDCOM Agent are broken, all existing pin-holes must be closed.

5.4. Pin-Hole reporting

The Middlebox may need to report, to the associated MIDCOM Agent, information concerning the status and behaviour of a flow through an established pin-hole.

5.5. Error handling

The MIDCOM protocol must enable the Middlebox and/or MIDCOM Agent to determine when an Application Stream Session is no longer valid. This is to address cases such as when either a Middlebox or MIDCOM Agent re-starts, or where resource contention and prioritisation take place.

The MIDCOM Protocol must allow the MIDCOM Agent access to sufficient information as to enable it to notify the application when a pin-hole has become invalid.

Other aspects of error handling are discussed in section 10.

6. Agent and Middlebox association

Prior to servicing any requests from a MIDCOM Agent for Middlebox services, a Middlebox must first establish a valid and appropriately trusted relationship with the MIDCOM Agent.

A given Middlebox may support multiple concurrent trust relationships with a number of MIDCOM Agents. Resources, such as a Pinhole-Descriptor, may be shared across MIDCOM Agents and a Middlebox. As Pinhole-Descriptors may be shared across Middlebox functions, a Pinhole-Descriptor may be created by one function, and terminated by a different one.

6.1. MIDCOM Agent Registration

Prior to permitting MIDCOM agents access to Middlebox resources, a registration process must take place between the MIDCOM Agent requiring Middlebox Services and the Middlebox intended to provide those services. This should appropriately authenticate each entity to establish the trust relationship between the two entities for subsequent interactions.

Registration is a different process from establishing a transport connection. The former requires exchanging the MIDCOM Agent’s profile information with the Middlebox i.e. establishing what kind of MIDCOM Agent it is. The latter refers to establishing a MIDCOM transport connection and exchanging security credentials adhering with the registered transport service profile.

A MIDCOM Agent may be pre-authorised or post-authorised to access a given Middlebox.

6.1.1. Pre-authorised registration

It is possible for a MIDCOM Agent to be pre-authorised to access Middlebox resources. In this case, the Middlebox needs to be able to validate registration requests originating from a previously identified MIDCOM Agent.

6.1.2. Post-authorised registration

Where a Middlebox is not previously aware of the authorisation status of a given MIDCOM Agent, it will need to establish these details as part of a dynamic registration process.

6.2. Deregistration

Once the need for the relationship between a MIDCOM Agent and a Middlebox established during the registration process has lapsed, a deregistration process will be required. This will break the trust relationship between the specific Middlebox and MIDCOM Agent. Once this has been done, a Middlebox must not service any further requests from a given MIDCOM Agent without the MIDCOM Agent first re-registering to re-establish the appropriate trust relationship.

MIDCOM session disconnection may be prompted by a successful de-registration request or a failure of some sort, such asa communications failure between the Middlebox and MIDCOM agent.

At the end of a MIDCOM session between a Middlebox and MIDCOM Agent, it should be possible for either the Middlebox or the MIDCOM agent to initiate the disconnection of the relationship.

As part of the de-registration process, any pin-holes or other Middlebox resources requested by the MIDCOM Agent must be immediately released by the Middlebox.

6.3. Error handling

The MIDCOM protocol must enable the Middlebox and/or MIDCOM Agent to determine when a registration session is no longer valid. This is to address cases such as when either a Middlebox or MIDCOM Agent re-starts.

Other aspects of error handling are discussed in section 10.

7. Application Control Flow Services

Application Control flows constitute the control aspect of an application. To enable a Middlebox to correctly handle such flows, they will need to be routed to a specific MIDCOM Agent that is able to handle the given application. These flows may be routed between a physical port on a Middlebox and a specific a MIDCOM Agent. The establishment of this association will requested via the MIDCOM protocol or may be determined by policy otherwise applied to the Middlebox.

7.1. Middlebox service requests

A MIDCOM Agent shall be able to request a pin-hole from a Middlebox for the purposes of establishing a presence on the external network.

A Middlebox may require a pin-hole request from a MIDCOM Agent to be authorised by an external Policy Server prior to granting the request.

A MIDCOM Agent may be notified when a pin-hole previously granted has become invalid, such as through a system restart.

A MIDCOM Agent may be able to refresh a pin-hole that has previously been granted.

The MIDCOM communication protocol needs to allow for selective communication between a specific MIDCOM agent and one or more Middlebox functions on the interface to which it is connected.

The MIDCOM protocol needs to allow either the MIDCOM Agent or the Middlebox to terminate the communications relationship between the MIDCOM Agent and Middlebox. This allows either entity to close the session for maintenance, security or other reasons.

7.1.1. Application message handling

For a MIDCOM Agent to be able to perform its function correctly, it will need to be able to intercept the flow of specific application control streams across an associated Middlebox. To do this, the MIDCOM Agent will need to request that the Middlebox intercept specific datagrams and direct these to the MIDCOM Agent. This enables the MIDCOM Agent to appear in the packet path of the application and consequently communicate with the Middlebox to establish the packet manipulations required for application media packets. The MIDCOM Agent may also need to pass manipulated packets back to the Middlebox for onward transmission.

7.2. Error handling

The MIDCOM protocol must enable the Middlebox and/or MIDCOM Agent to determine when an Application Control Session is no longer valid. This is to address cases such as when either a Middlebox or MIDCOM Agent re-starts.

Other aspects of error handling are discussed in section 10.

8. Application Content Flow Services

To enable the separation of application specific behaviour from a Middlebox, an external MIDCOM Agent needs to be able to express the requirements it has for the Middlebox to support a specific flow. The MIDCOM approach resolves this issue by identifying the need for a Control Language. The MIDCOM Control Language describes the desired behaviour of the Middlebox for the purposes of a requested flow.

To enable the solution to be reused across a number of different applications, the syntax and semantics of the MIDCOM Control Language shall be independent of a given application.

The syntax and semantics of the MIDCOM Control Language need to be extensible.

An operation is needed to permit a flow, with a given flow specification and address translation, across the Middlebox. It shall be possible to indicate whether packets should be carried or discarded when a given flow specification is exceeded.

9. Resource control requirements

On receipt of a request to create a pin-hole to allow a flow with a particular flow specification, the Middlebox needs to determine that appropriate resources are available that will allow the flow, as described to the Middlebox, to be carried.

The MIDCOM protocol must enable a Middlebox to be able to ensure the integrity of the network on both its inside and outside ports. This may cause the Middlebox to refuse requests from MIDCOM Agents where necessary.

The Middlebox needs to report or otherwise make information available to the associated MIDCOM Agent when a granted flow contravenes the parameters agreed. The Middlebox should not permit the flow of any packets contravening a prior agreement as this may adversely impact other flows traversing the Middlebox.

The Middlebox may provide QoS transport facilities for packet flows, for example, by setting DS-bytes or MPLS mappings.

Other than the communication of control information between Middlebox and an associated MIDCOM Agent, no packets will be allowed to flow across the Middlebox unless requests by a MIDCOM Agent. i.e. the default operation is that "no packets should be allowed to cross the Middlebox".

10. Operational and Management requirements

10.1. Status reporting

The Middlebox may report the status of any pin-hole to the associated MIDCOM agent.

The Middlebox may report the observed behaviour of a nominated flow through a specific pin-hole.

The MIDCOM Agent may report its status to a Middlebox with which it is associated.

A Middlebox may report its status to a MIDCOM Agent with which it is associated.

10.2. Unsolicited messages

The protocol needs to support unsolicited messages, for event reporting and propagation of alarms.

10.3. Protocol Failure

To enable management systems to interact with the MIDCOM environment, the protocol needs to include failure reasons that allow the MIDCOM Agent behaviour to be modified as a result of the information contained in the reason. Failure reasons need to be chosen such that they do not make an attack on security easier.

10.4. MIDCOM Agent Authentication Requirements

The MIDCOM Protocol must allow communications between Middleboxes and MIDCOM Agents to be authenticated.

The MIDCOM Protocol shall describe the desired behaviour of the Middlebox for the purposes of a requested flow(s).

The syntax and semantics of the Authentication aspects of the MIDCOM Protocol shall be independent of the application(s)requesting packet processing services.

The syntax and semantics of the Authentication aspects of the MIDCOM Protocol need to be extensible.

Where possible, existing IETF protocols shall be used for authenticating MIDCOM entities such as Middleboxes and MIDCOM Agents.

10.5. MIDCOM Policy Requirements

The MIDCOM Policy Protocol shall be used between MIDCOM Agents and MIDCOM Policy Servers.

The MIDCOM Policy Protocol shall describe the desired behaviour of the MIDCOM Agent for the purposes authorising requested flow(s).

The syntax and semantics of the MIDCOM Policy Protocol shall be independent of the application(s)requesting Middlebox services.

The syntax and semantics of the MIDCOM Policy Protocol shall be extensible.

Where possible, existing IETF protocols shall be used for communicating policy information and decisions between MIDCOM entities such as Middleboxes and MIDCOM Agents.

10.6. Asynchronous behaviour

Since Middleboxes and MIDCOM Agent(s) are likely to operate asynchronously, the protocol needs to permit the synchronisation of state machines between peer entities by adding state information to messages in normal operations or by the use of explicit query messages.

Either the agent or the Middlebox can choose to initiate a connection prior to any data traffic. Alternately, either party (Middlebox or the MIDCOM agent) may choose to initiate a connection only upon noticing application specific traffic.

10.7. Protocol reliability

The protocol needs to provide per message acknowledgements, in order to achieve reliable communication.

10.8. Protocol Extensibility

The MIDCOM protocol needs to contain version inter-working capabilities to enable subsequent extensions to support different types of Middlebox and future requirements of applications not considered at this stage.

If a peer does not understand an option it must be clear whether the action required is to proceed without the unknown attribute being taken into account or the request is to be rejected. Where attributes may be ignored if not understood, a means may be provided to inform the client about what has been ignored.

This may be provided as a version and capability exchange mechanism.

10.9. Protocol Performance

The protocol needs to support time-constrained operation to enable scalability of the complete solution. This may include, amongst other things, the use of short messages and minimal messages per operation.

11. Protocol Transport Requirements

11.1. MIDCOM Protocol transport

Communications between a MIDCOM Agent and a Middlebox constitute a flow of requests for services and associated responses. The transport of this information may present a security concern and should be appropriately protected.

11.2. Application protocol transport

To enable the MIDCOM Agent to implement the desired application level functionality, it may be necessary for the Middlebox to forward IP packets received on specific IP addresses and/or ports directly to the MIDCOM Agent.

11.3. Protocol Transport performance

The protocol must not assume 0 (zero) delay in communication paths to enable realistic deployment scenarios. The protocol must therefore be capable of being used over low speed links.

12. Security Requirements

The security of the interfaces will be one of the primary focuses of the work, and potential vulnerabilities on the interfaces and in the architecture along with defenses against those threats need to be identified.

To protect MIDCOM messages from being tampered with, individual message authentication should be used in addition to host authentication. Further message confidentiality may be administered by employing appropriate techniques to MIDCOM messages, when a higher level of security is required. Simple Source-address based security is the least form of security and should be permitted only to the most trusted hosts.

The MIDCOM protocol must enable a trust relationship to be established between a specific MIDCOM Agent and a specific Middlebox for the purposes of permitting flows between the hosts that flow via the Middlebox.

In the trusted case, the trust is derived from some aspect of the session in which the trusted party is involved, for example the security association. In the untrusted case, trust is derived from the presentation and authentication of credentials that are offered for each request.

Entities communicating using this protocol should be authenticated in order for message exchange to proceed.

Trust relationships within the MIDCOM architecture must be explicitly established.

In order to prevent hacking into the Middlebox by intercepting communication, the MIDCOM protocol must allow for strong encryption to be used to protect the trustworthiness and privacy of the requests and responses.

12.1. Trust relationships

The establishment of appropriate trust relationships between a specific Middlebox and MIDCOM Agent pair are essential to maintain the integrity of the MIDCOM solution. A Middlebox may have either a Trusted or an Un-trusted relationship with a MIDCOM Agent. A Middlebox must establish an appropriate level of trust with any MIDCOM Agents with which it communicates before it processes any requests that would cause pin-holes to be established, maintained or removed from the Middlebox.

The Middlebox should only accept Middlebox service requests from MIDCOM Agents with which it has a trust relationship.

Based upon the established identity of the requesting MIDCOM Agent, a Middlebox shall establish trust and authorise requests for Middlebox services from that MIDCOM Agent. The Middlebox shall authorise the request based upon either local policies or policies from the MIDCOM Policy Server.

Where no trust relationship is maintained between a MIDCOM Agent and a Middlebox, each request received by the Middlebox shall be authorised, including any necessary authentication, before the Middlebox takes any action on the request.

MIDCOM agents, their trust level and accessibility (i.e., the MIDCOM agent profile) may be pre-registered with the Middlebox while provisioning the Middlebox function.

12.1.1. Trusted MIDCOM Agent

In the case where it is trusted, the Middlebox will treat the request from the MIDCOM Agent as authoritative and respond accordingly.

12.1.2. Un-trusted MIDCOM Agent

Where no trust relationship is maintained between a MIDCOM Agent and a Middlebox, each request received from the MIDCOM Agent will need to be authorised, before the Middlebox takes any action on the request. This authorisation process will need to include any necessary authentication of the MIDCOM Agent.

In the case where it is untrusted, the Middlebox will have to verify within the parameters determined by any local security policy that it is authorised to complete the request. This authorisation could be non-existent or come from a policy server either separated from the Middlebox or one that is an integral part of it.

12.2. General Security considerations

Security mechanisms may be specified as provided in underlying transport mechanisms, such as IPSEC. The protocol, or such mechanisms, must:

a. Allow for mutual authentication at the start of an MIDCOM Agent-Middlebox association

b. Allow for preservation of the control messages once the association has been established.

c. Allow for optional confidentiality protection of control messages. (If provided the mechanism should allow a choice in the algorithm to be used.)

d. Operate across un-trusted domains between the MIDCOM Agent and Middlebox in a secure fashion.

e. Support non-repudiation for a customer-located Middlebox communicating with a network operator’s MIDCOM Agent.

g. Define mechanisms to mitigate denial of service attacks.

h. Define mechanisms to mitigate replay attacks on the control messages.

Note: any associated protocol document will need to include an extended discussion of security requirements, offering more precision on each threat and giving a complete picture of the defense including non-protocol measures such as configuration.

13. Copyright

14. Intellectual Property

15. Acknowledgements

The authors would like to acknowledge the contributions from members of the MIDCOM working groups, specifically Jiri Kuthan, Jonathan Rosenberg, Andrew Molitor, Pyda Srisuresh and Melinda Shore.

16. References

[MIDBOXFRAME] Middlebox Communication Architecture and Framework – work in progress.
[MIDBOXSCEN]   MIDCOM Scenarios – work in progress

17. Author's Addresses

Richard Swale
BTexact Technologies
Callisto House
Adastral Park
Ipswich United Kingdom   Email: richard.swale@bt.com

Paul Sijben
Lucent Technologies EMEA BV
Huizen
Netherlands   Email: sijben@lucent.com

Philip Mart
Marconi Communications Ltd.
Edge Lane
Liverpool
United Kingdom   Email: philip.mart@marconi.com