Midcom Working Group
Internet Draft

Category: Standards Track
Expires on March
2002
September 2001

Sanjoy Sen
Cedric Aoun
Tom Taylor
Nortel Networks

MEGACO Middlebox Packages
<draft-sct-midcom-megaco-pkg-00.txt>

Status of this Memo

Abstract

This draft is work-in-progress, intended to satisfy some of the requirements in [1] that are not met by the Megaco base protocol as discussed in [2]. It defines three types of Packages:

- the base Middlebox Package containing properties and events supported by all Middlebox Terminations

- the Firewall Package, extending the base package, containing properties and events supported by Middlebox Terminations supporting firewall functions.

- the NAT Package, extending the base package, containing properties and events supported by Middlebox Terminations supporting NAT function

A generic model to extend the base Middlebox package and new command error codes for Middlebox control are also discussed.

Table of Contents

1 Introduction

This draft is work-in-progress, intended to satisfy some of the requirements in [1] that are not met by the Megaco base protocol as discussed in [2]. It defines three types of Packages:

- the base Middlebox Package containing properties and events supported by all Middlebox Terminations

- the Basic Firewall Package, extending the base Middlebox package, containing properties and events supported by Middlebox Terminations supporting basic packet-filtering functions.

- the Basic NAT Package, extending the base Middlebox package, containing properties and events supported by Middlebox Terminations supporting basic Address/Port translation functions.

A generic model to extend the Middlebox packages and new command error codes for Middlebox control are also discussed.

2 Conventions used in this document

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.

3 Midcom Terminologies and Concepts [3]

Middlebox: a device that has router functionality and either alters the content of the IP header or drops or forwards packets depending on the filtering rule that is applied.

Midcom Agent or Agent: an entity performing an application layer gateway (ALG) function, logically external to a Middlebox. Midcom agents possess a combination of application awareness and knowledge of the Middlebox function.

Ruleset: A logical Middlebox resource comprised of a matching expression for packet flows (flow descriptor) and the actions specified on the packets that match the flow descriptor (e.g., drop, modify certain fields of the IP header etc.)

Midcom protocol: The protocol between a Midcom agent and a Middlebox that allows the Midcom agent to gain access to Middlebox resources and allows the Middlebox to delegate application specific processing to Midcom agent.

The above terminologies are aligned with the terminologies currently used in the Midcom WG and may evolve in time. The draft will be updated to reflect any modification of the terminology.

4 ARCHITECTURE and REQUIREMENTS

[3] describes the general Midcom architecture consisting of the Agent and the Middlebox. When the Agent detects the initiation of an application session requiring Middlebox service, it requests the Middlebox to establish a ruleset for the application flow. The request should carry the following information at the minimum:

- suitable descriptor (5 elements minimum - source address, source port, destination address, destination port, protocol id) to identify the flow(s)
- actions (allow, drop, IP address/port translation, or other IP header manipulation) to be performed on the matched packets
- time-to-live(s) to be associated with the ruleset
- information (if required) for the Middlebox to determine the interface(s) with which the ruleset should be associated

NOTE: The properties discussed in this draft are for the purpose of illustration of key ideas and are likely to change with time. The Midcom WG is in the process of defining the minimum set of information to be carried by the protocol. The next version of the draft should reflect the consensus of the Working Group.

The Middlebox should be able to detect Events such as ruleset timer expiry, element failure etc., and report them to the Agent. It should also be able to collect relevant statistics, e.g., the number of packets on which a proposed action has been performed, for reporting them to the Agent. All these parameters are carried in Megaco requests and responses and are defined in these packages.

To model the Middlebox functions such as firewall, NAT etc., a new Middlebox Termination type is defined. Such a Termination can be associated with an interface and MUST contain the following parameters - flow descriptor and action(s). In order to allow multiple agents manipulate a ruleset (a key Midcom requirement), the latter is kept separate from the Termination. A Termination shall be associated with a single ruleset, but a ruleset may be associated with more than one Termination. Thus, a Termination can share a ruleset with another Termination, or have a ruleset partially overlapping with that of another Termination. This model allows two Agents, controlling two distinct Terminations manipulate the same or overlapping ruleset(s) as discussed in [2]. A Termination will also support an Event Timer.

At start-up or service change, the Middlebox capabilities, including all the Terminations and Packages supported, are queried using the AuditCapabilities command. It is assumed that a trust relationship between the Middlebox and the Agent has already been established at this stage (using IPSec, for example, as the underlying transport mechanism).

5 BASE MIDDLEBOX PACKAGE

PackageID: mb (serial number TBD)
Version: 1
Extends: None

Description: This package is supported by all Middlebox terminations. It contains the following properties associated with TerminationState descriptor: Ingress Realm, Source Address, Source Port, Egress Realm, Destination Address, Destination Port, Protocol Identifier, RTP Support, and Action. It also contains the following Events: Ruleset Expiry and Element Failure.

5.1 PROPERTIES

1) Ingress Realm

PropertyId: inrealm (0x0001)

Description: indicates the realm from which the flow enters the Middlebox. This property can be specified, left unspecified or wildcarded (ALL). The Ingress Realm property, in conjunction with Source Address, is used by the MB to determine the ingress interface(s) with which the ruleset shall be associated. This determination is governed by the following rules:

I. If both the Ingress Realm and the Source Address are specified, the MB should be able to uniquely determine the ingress interface with which the ruleset shall be associated.

II. If the Ingress Realm is specified and the Source Address is wildcarded, the ruleset shall be associated with all ingress interfaces under the Ingress Realm.

III. If the Ingress Realm is left unspecified by the Agent, the ruleset must NOT be associated with any interface unless the Egress Realm is specified.

IV. If the Ingress Realm is wildcarded with ALL, the Agent is requesting the MB to determine its interface with which the ruleset shall be associated (from routing table). Note: this assumes that the Source Address be globally routable. If not, the Agent is required to know the Realm.

Type: string - syntax TBD

Values: as set by the Network Administrator. Can be specified, left unspecified or wildcarded (only ALL).

Defined in: TerminationState descriptor

Characteristics: read/write

2) Source Address

PropertyId: srcaddr (0x0002)

Description: indicates the source address or range of addresses for identifying flow(s). Source Address can be used in conjunction with the Ingress Realm to determine the interface(s) with which a ruleset shall be associated (See above).

Type: string - syntax TBD

Values: Can be either specified (as a complete address or address range) or wildcarded (only ALL).

Defined in: TerminationState descriptor

Characteristics: read/write

3) Source Port

PropertyId: srcport (0x0003)

Description: indicates the source port or range of ports for identifying flow(s).

Type: integer

Values: Can be either specified (as a complete address or address range) or wildcarded (only ALL).

Defined in: TerminationState descriptor

Characteristics: read/write

4) Egress Realm

PropertyId: egrealm (0x0004)

Description: indicates the destination realm of the flow from the MB. This property can be specified, left unspecified or wildcarded (ALL). The Egress Realm property, in conjunction with Destination Address, is used by the MB to determine the egress interface(s) with which the ruleset shall be associated. This determination is governed by the following rules:

I. If both the Egress Realm and the Destination Address are specified, the MB should be able to uniquely determine the egress interface with which the ruleset shall be associated.

II. If the Egress Realm is specified and the Destination Address is wildcarded, the ruleset shall be associated with all egress interfaces under the Egress Realm.

III. If the Egress Realm is left unspecified by the Agent, the ruleset must NOT be associated with any interface unless the Ingress Realm is specified.

IV. If the Egress Realm is wildcarded with ALL, the Agent is requesting the MB to determine its interface with which the ruleset shall be associated (from routing table). Note: this assumes that the Destination Address be globally routable. If not, the Agent is required to know the Realm.

Type: string - syntax TBD

Values: as set by the Network Administrator. Can be specified, left unspecified or wildcarded (only ALL).

Defined in: TerminationState descriptor

Characteristics: read/write

5) Destination Address

PropertyId: destaddr (0x0005)

Description: indicates the destination address or range of addresses for identifying flow(s). Destination Address can be used in conjunction with the Egress Realm to determine the interface(s) with which a ruleset shall be associated (See above).

Type: string - syntax TBD

Values: Can be either specified (as a complete address or address range) or wildcarded (only ALL).

Defined in: TerminationState descriptor

Characteristics: read/write

6) Destination Port

PropertyId: destport (0x0006)

Description: indicates the destination port or range of ports for identifying flow(s).

Type: integer

Values: Can be either specified (as a complete address or address range) or wildcarded (only ALL).

Defined in: TerminationState descriptor

Characteristics: read/write

7) Protocol Identifier

PropertyId: protoid (0x0007)

Description: identifies the protocol datagram being carried in the IP packet

Type: string

Values:

Defined in: TerminationState descriptor

Characteristics: read/write

8) RTP Support

PropertyId: rtp (0x0008)

Description: Specifies whether or not an RTCP flow will be associated with an RTP packet flow in opposite direction. This translates into the MB allocating port bind or opening pinhole for the port consecutive to the RTP port, and that the address translation result is as follows: RTP address a/portx, RTCP address a/portx +1 <-> RTP address b/porty, RTCP address b/porty + 1. It is assumed that if an RTP flow is allowed, the corresponding RTCP flow will always be allowed. The default value is set to FALSE.
Type: Boolean

Values: TRUE, FALSE

Defined in: TerminationState descriptor

Characteristics: read/write

9) Action

PropertyId: action (0x0009)

Description: Specifies the action that should be applied by the Middlebox on the matched packets. Extension to this Package will add possible values to action.

Type: Enumeration

Values:

Defined in: TerminationState descriptor

Characteristics: read/write

5.2 EVENTS

1) Ruleset Expiry

EventID: rule-expiry (0x0001)

Description: Indicates that the ruleset-timer associated with a Termination has expired.

EventDescriptor Parameters:

Timer
ParameterID: timer (0x0001)
Description: timer associated with the Termination
Type: integer
Possible values: in sec

ObservedEventDescriptor Parameters: None added to this Package

2) Element Failure

EventID: mbfail

Description: Indicates a failure in the processing of the Middlebox function

EventDescriptor Parameters: none added by this package

ObservedEventDescriptor Parameters:

Error code
ParameterID: ec
Description: describes the failure reason
Type: integer, 0 to 99
Possible values:
1 Firewall failure
2 NAT failure

5.3 STATISTICS

None

5.4 SIGNALS

None

5.5 PROCEDURES

The Agent creates a new Termination in a Context when it wants to create a new ruleset on behalf of the application. It subtracts the Termination from the Context when the ruleset is no longer needed.

The Agent associates a Timer Event with a Termination (and implicitly, with a ruleset). Thus, by virtue of the one-to-many association between the ruleset and Terminations (i.e., when a ruleset is shared by multiple Agents), a ruleset may be associated with multiple Timers, each controlled by an Agent. When a Timer expires, the Agent is notified of that Event by the Middlebox. The Agent may choose to refresh the ruleset by sending a MODIFY command to the Termination.

6 BASIC FIREWALL PACKAGE

PackageID: bas-fw (serial number TBD)
Version: 1
Extends: mb

Description: This package describes the properties required by the Middlebox Termination to perform basic packet filtering function.

6.1 PROPERTIES

The Property Action in the Base Package is extended to specify possible packet-filtering actions: "Allow" and "Drop".

6.2 EVENTS

None

6.3 STATISTICS

1) Packets Dropped

ParameterID: pktsdrop (0x0001)

Description: Number of packets dropped by the Termination in a session

Units: in packets

Defined in: Statistics descriptor

7 BASIC NAT PACKAGE

PackageID: bas-nat (serial number TBD)
Version: 1
Extends: mb

Description: This package provides the properties required by the Middlebox Termination to perform address and port translation (NAPT) function

7.1 PROPERTIES

1) NAT Action

PropertyId: nat-action (0x00010)

Description: used by the MB to specify whether only address translation or both address and port translation can be performed by the Termination on matched packets

Type: Enumeration

Values: "Address", "Address-port"

Defined in: TerminationState descriptor

Characteristics: read only

2) Bind Values

PropertyID: Bindvals (0x00011)

Description: Allows the MB to specify the translated address/port information to the MA. Also allows the MA to offer hint to the MB about the translated address/port.

Type: String - detailed syntax TBD

Values:

Defined in: TerminationState descriptor

Characteristics: read/write

7.2 EVENTS

None

7.3 STATISTICS

1) Packets Translated

ParameterID: trans (0x0002)

Description: Number of packets translated by the Termination in a session

Type: Double integer

Units: in packets

Defined in: Statistics descriptor

8 NEW COMMAND ERROR CODES

Errors consist of an IANA registered error code and an explanatory

string. Megaco consists of a list of IANA registered error codes. Following are the new ones that need to be added to that list for the purpose of Midcom:

582 Ports unavailable
Description: used by a Middlebox NAPT to indicate to the Agent about unavailability of ports for translation.

583 Address and port already in use
Description: used by a Middlebox NAPT to indicate to the Agent that the requested Address/port is already in service

584 Port already in use
Description: used by a Middlebox NAPT to indicate to the Agent that the requested port is already in service

585 Resource already in use
Description: used to indicate contention when multiple Agents attempt to access/modify the same ruleset

9 Package creation model for new Middlebox functions

The protocol should be able to incorporate several new types of Middlebox functions. All new functions can be modeled as extensions to the base Middlebox package. The new package will follow the structure of the standard Megaco packages as defined in [4].

10 Security Considerations

Please refer to [3] for discussions.

11 IANA Considerations

The document describes new Packages for Middleboxes providing firewall and NAT functionality. The document also describes new command error codes. Both of the above will need IANA registration.

12 References

[1] Brim et. al., "Midcom Requirements", midcom-reqs-bullets-010910.txt, work in progress
[2] Sen, Aoun, Taylor, "Applicability of Megaco for Middlebox Control", draft-sct-midcom-megaco-00.txt, work in progress
[3] Srisuresh, Kuthan, Rosenberg," MIDCOM Architecture & Framework", Internet draft, draft-ietf-midcom-framework-03.txt
[4] "MEGACO Protocol Version 1.0", RFC 3015

13 Acknowledgments

The authors would like to thank Mark Watson for his useful comments related to this draft.

14 Author's Address

Sanjoy Sen
Nortel Networks
sanjoy@nortelnetworks.com

Cedric Aoun
Nortel Networks
cedric.aoun@nortelnetworks.com

Tom Taylor
Nortel Networks
taylor@nortelnetworks.com

15 Intellectual Property Statement

16 Full Copyright Statement