Network Working Group Eliot Lear
INTERNET-DRAFT Cisco Systems
Category: Informational

<draft-lear-middlebox-arch-00.txt>
January 3, 2001

A Middle Box Architectural Framework

1 Status of this Memo

2. Copyright Notice

2 Abstract

It used to be reasonable to expect that any two points connected to the Internet to have the ability to hold any communication. Such an expectation has not be reasonable for quite some time, thanks to firewalls, NATs, and other intermediate devices. Today, we acknowledge a new architecture and we name the functional blocks of that architecture as well as several ways to get ends to communicate, and how two devices could expect to communicate with each other. This document does not define the protocols involved.

3 Introduction

The IPv4 Internet consists of a network of interconnected networks that may use public or private address space. The use of private address space [BCP5] has broken the classic connection model that applications use to speak to other devices. Similarly, many networks are separated by firewalls.

Now we consider the components necessary for end to end communications in this environment, as well as the forms of signaling necessary for those communications to occur in a reliable way.

The reader should be familiar with RFCs 1918, 2663, and 2775. We do not intend to fix all problems related to NATs or firewalls with this architecture. For instance, one will not find below a way to save a TCP connection in the face of a NAT failure.

3.1 Motivation

The currently envisioned sets of applications to make use of this architecture are voice and video conferencing and telephony. Others may follow. Today voice and video capture a small percentage of total network usage. As demand for these uses grows it will become more important that we have in place mechanisms that have proper scaling properties.

3.2 Goals, Terms, and Limits

Here are our design goals:

1. Enable hosts within different administrative domains and address realms to have end to end sessions.
2. The architecture must allow for multiple middle boxes that connect one realm to multiple realms.
3. Enable this with a minimal amount of signaling from the end host, and no new signaling beyond administratively defined boundaries.
4. The recommended methods must not radically alter the end host stack below the application.
5. Work within the existing interior and exterior routing framework.
6. The mechanisms used must easily integrate with the existing Internet mechanisms and services.

The three middle box processes we consider in this document are discovery, diagnostics, and signaling. Discovery means determining that one or more middle box sits between a host end the remote end of a session. Once that box is discovered, either an end host or its proxy will exchange information with it in order for the communication to proceed. Diagnostics is the untimely discovery of a problem involving a middle box. Discovery and diagnostics differ in that an end host may receive a diagnostic message from one middle box while it might need to discover a separate middle box in order for the communication to proceed. The architecture we propose elides the two functions.

No changes are made to the layer 3 routing mechanism, other than the possible addition of a new ICMP message to be multiplexed back to the responsible application. To do otherwise would dramatically increase implementation cost and complexity, as well as the cost of troubleshooting problems. While the host may signal for additional information to and from the application layer, it cannot rely on new layer 3 routing facilities within the network.

Nothing this architecture proposes may stop two oHOSTs from communicating with each other without the help of a middle box. Similarly, this architecture must not prevent two iHOSTS from communicating, just as they would have previously. However, for all the reasons listed in RFC 2775 it is not possible for a middle box to enable all forms of communication between iHOSTs and oHOSTs. The recommended method to clear at least some of these roadblocks is the wide deployment of IP version 6.

4 Architectural Components and Terms

We now define components of the architecture based on the following diagram:

```
______________________________
|                             |
|   Private Realm             |
|                             |
|                     [AC]    |
|                             |           -------        [oHOST]
|                             |          /        \
|  [iHOST]              [middle box]    |Internet|
|                             |          \        /
|            [iHOST]          |           \------/
|                             |
-------------------------------
```

A private realm consists either of hosts that are numbered within the space defined by BCP 5 or hosts that sit within a single security domain. An iHOST is a host that sits within that realm. An oHOST sits outside the private address realm. It may be assigned public address space or private address space. In the latter case it would sit within its own private address realm. An AC is an application controller. An application controller may be an application layer gateway, or it may merely arrange for communication between two hosts, be they iHOSTs or oHOSTs. An AC may itself be a middle box.

A middle box is a device that sits on the edge of a private realm. It is the responsibility of the middle box to police or transform sessions from its private realm to the outside world. Middle boxes consist of firewalls, NATs and other devices that sit within the flow of packets and may impact a session from an iHOST to an oHOST. At the border of a private realm there may be multiple middle boxes that connect the realm to other realms.

The remainder of this document refers to communications between an iHOST and an oHOST.

4.2 Communication Models

No matter the model we assume that iHOSTS may only signal to middle boxes that exist within the same administrative domain.

There are two models of communication we will consider. The first model may not require end host application changes, but rather requires changes to servers that assist end hosts in communicating with each other. The second model allows end hosts themselves to exchange information with the intermediate devices. These two models differ significantly.

4.2.1 Proxy Signaling

In the first case, iHOSTs use something akin to an application layer gateway to first arrange a session. We will generically refer to such a box as an application controller (AC). An example would be H.225 signaling between an H.323 end point and a gate keeper. As the gatekeeper processes call setup information it would communicate with any middle box necessary for the end points to establish end to end connectivity. This model is best employed when there already exists some sort of application controller, since no additional signaling may be necessary between the iHOST and the AC. Either the end application is required to know of the AC or the AC must reside within the data path between the iHOST and oHOST.

```
______________________________
|                             |
|   Private Realm             |
|                             |
|          --- 1 --> [AC]->v  |
|         /                v  |
|        /                 2  |
|       |                  v  |           -------        [oHOST]
|       |                  v  |          /       \         ^
|  [iHOST]-----3-------->[middle box]--->|Internet|---3---->^
|                             |          \       /
|              [iHOST]        |           \------/
|                             |
-------------------------------
```

In order for the end hosts to continue sessions with some level of reliability the AC must maintain topological awareness, so that it can determine which middle boxes are involved. It might do this by monitoring and examining link state information from the IGP. However, no new signaling would be required between iHOSTs and middle boxes.

4.2.2 Direct Signaling

In the second case, no AC is required. Instead, an iHOST must determine that a middle box exists, signal to it for end to end configuration information, and then proceed. Furthermore, the iHOST must determine when the path to the oHOST has changed during a session. The application should recover as circumstances dictate.

In this model it is important for the end host to determine that not only has a failure occurred, but that the failure occurred due to something the middle box could see or control.

```
 ______________________________
|                             |
|   Private Realm             |
|                             |
|                             |
|                             |           -------        [oHOST]
|        >------2--------v    |          /        \
|  [iHOST]<-----1--------[middle box]    |Internet|
|       ^<------3--------v    |          \        /
|                             |           \------/
|                             |
-------------------------------
```

1.   Diagnostic/discovery, a message or messages that indicate that the middle box requires attention. This message is sent in response to an attempt by iHOST to start a session with oHOST that the middle box knows will not succeed.
2.   Signaling request from iHOST to middle box.
3.   Response from middle box.

Note that in this model there is no AC.

4.2.3 Current use of Application Layer Gateways

An alternative model would be for end hosts to use application layer gateways to access external resources. This model requires no new generic signaling, but a method for each iHOST to determine when it should use an application layer gateway, and when it should communicate directly with another iHOST.

In this case, because the ALG is a middle box, it follows that this method requires the ALG to reside on the perimeter of the administrative domain.

We mention this model not merely for completeness, but because it is the operative model for many applications that would eventually use one of the other two models. Its clear benefit is that it exists today. Furthermore, there may be benefits to having ALGs reside on the perimeter. For instance, these devices will be able to statefully inspect each and every packet to and from an internal network.

4.3 Which model should we build?

The astute reader will notice that the direct model is very close to a superset of the proxy model. The proxy model needs a signaling mechanism between the AC and the middle box. There is no reason that signaling mechanism couldn't be the same one used by the direct model.

Indeed as diagnostics are introduced they can enhance both the proxy model and the ALG model by returning decent diagnostics to the end user when iHOST not properly configured or the AC or ALG unexpectedly falls outside the data stream.

5 Diagnostics and Discovery

As mentioned in the introduction, we elide the diagnostics and discovery functions in this architecture. No matter which of the above models is implemented diagnostics are required to indicate to the users and application when a failure has occurred. These failures can take numerous forms, but here are some examples:

- A middle box might reboot and lose state of which holes are meant to have been open.
- The traffic may be rerouted to an unsuspecting middle box.
- The application on iHOST may be unaware of a middle box for which an information exchange is required.
- An AC may be misconfigured to prevent the iHOST from establishing a session with an oHOST.

The diagnostic mechanism used on the Internet is ICMP. When a middle box detects an attempt by an iHOST to start a session to an oHOST that will not succeed, it should send a message back to the iHOST indicating that a failure is likely. The content of that ICMP message is discussed separately in [BLOCKER].

Of the two models mentioned in Section 4, discovery may be different. With the proxy model it is fairly easy to configure by hand a relatively small number of ACs. Indeed either ACs must know about the middle boxes or the middle boxes must know about them, since ACs do not sit within the packet flow.

With the direct signaling model it is possible to piggyback discovery on top of the diagnostic message discussed earlier.

6 Signaling

Once the application has determined that it must communicate with a middle box in order for a communication to properly proceed, either the AC or the iHOST initiates an exchange with the middle box. The nature of this exchange depends on the function of the middle box. For instance, if the middle box is a firewall, the application is in essence asking permission from the firewall for the communication to proceed. If, on the other hand, the box is a NAT or NAPT, the middle box may merely need to know the mapping between the two addressing realms for the communication. These two functions can be combined, and it is reasonable to do so in order to reduce signaling overhead.

Any signaling requests to reserve address space or open pinholes must be matched with similar requests to undo what was done. However, firewalls will as a matter of policy not trust that all went well. Indeed they should be fairly conservative to reduce the risk that pinholes have not been left open beyond their legitimate purposes.

6.1 Firewalls

Firewalls require sufficient information about the communication to determine whether or not it is authentic and whether or not it is authorized. The firewall may query the application for specific information necessary for authorization, but we can assume that during the initial contact the firewall will need at least the following information:

1. Protocol to be used during a session
2. IP address and source port of the iHOST
3. IP address and source port of the oHOST
4. One or more methods to indicate when the session has ended.

The firewall may also wish to know what person is initiating the request, the application that is being used, or even perhaps a token of some form. While it might wish to have arbitrary amounts of information in order to make its decision, applications need to be aware of the sorts of information the firewall will demand. Thus, the names and formats of the values of the requested information must be standardized, and should be managed by IANA.

If a firewall is going to respond to a request from an iHOST or AC it SHOULD do so with a request for all the information it needs in response to an initial request from either the iHOST or AC.

The format of the query the firewall makes must be standardized, as should the names and formats of the individual attributes.

A firewall may accept, reject, or ignore such signaling requests. If the firewall accepts a request it should respond with the protocol, source and destination IP addresses and ports, confirming the way the communication shall be terminated. In addition, it may return additional information, as requested. This brings us to NATs.

Note that even if a communication is initially authorized, nothing we state here should prevent or even discourage further stateful inspection of any communication.

6.2 NATs

An AC or iHOST may request the port and IP address mapping between two realms as part of the query discussed above. The middle box may return the appropriate mappings. If it does so, it MUST also return connection termination conditions.

6.3 Termination Conditions

It is important for the middle box and the application (AC or iHOST) to agree on when a session has ended. In the case of a firewall it is critical that it properly close the pinhole it opened. In the case of a NAT, once a session has terminated the NAT may reallocate addressees and ports to another iHOST.

Termination conditions can be one of several methods:

1. A period of time, similar to a TTL used by DHCP.
2. A requirement that the application tell the middle box when the communication has ended.
3. An easily discernable in stream method to determine that the session is over. For instance, a TCP session ends with the exchange of FINs, followed by their acknowledgement. Such methods should be described in an RFC.

While an application might request one or more method it is up to the middle box to decide which method to employ. If more than one method is contained within a request or a response, the termination condition that occurs soonest will be used.

For example, an H.323 gateway might request a UDP connection from the iHOST on port 7499 to oHOST on port 8233. The AC might also request that it tell the middle box when the session is terminated. A firewall might respond that it has opened a pinhole as described, and the termination conditions are that the AC will indicate the completion of the session AND the session will be considered closed after five minutes.

Prior to the expiration of the TTL, if the call is still active, the AC might further request additional time from the middle box.

6.4 Communication between middle boxes

Although it is theoretically possible for middle boxes to exchange connection state amongst each other, the overhead for doing so may well prove quite high, and the value is dubious. If a middle box fails it is possible that a hot spare would be able to take over its responsibilities. There exists at least one document that considers this possibility [YAKOV et.al]. However, we choose not to standardize this function at this time.

More likely the case will be that any existing connections will fail due to a topological change, either the middle box failing or a route to the middle box failing. Therefore, it is important that end hosts be able to re-establish communications and retain state above the transport layer, as is necessary and appropriate.

6.5 Signaling Protocol Choice

Returning to the discussion of the two different models discussed in Section 4, we note that there are subtle differences in expected protocol characteristics between the proxy and direct models. In the case of the direct model an iHOST could expect to issue a single request per session. However, in the case of a proxy, it is likely to make many requests to the middle box on behalf many client iHOSTs.

The signaling protocol should allow for ease of failover. In addition, the protocol should also take minimal resources on both client and middle box. The client itself may be a middle box. In any event the signaling end points - both middle box and client - MUST MUST MUST implement appropriate congestion control mechanisms.

7 Multiple Middle Boxes

When used with the direct model, a diagnostic message such as ICMP allows the application on an iHOST to determine not only that a middle box is in its path, but also which middle box is in its path. Once the iHOST identifies the middle box it can signal to it. Should the data path change so that another middle box is chosen, the iHOST will once again receive a notification.

Depending on the application or environment it may be possible for an iHOST to fail over between two middle boxes that are sharing state. Such failures must be transparent to the iHOST at all layers.

The matter of multiple middle boxes is somewhat more complex with the proxy model. Because the AC is not in the data flow it must go to some additional measures to determine that a middle box has failed. Indeed once the AC has determined that a particular middle box has failed, or that a path has changed, it must communicate appropriate information back to the iHOST. Thus, unless the application already anticipates appropriate failure and restart conditions, modifications may be required, defeating the usefulness of the proxy model.

There is one additional case to be considered. When an iHOST attempts to communicate across several concentric boundaries it might require several rounds of signaling before a session could proceed. The fastest way for signaling to proceed within the direct model is for the middle box to forward a packet that has generated a diagnostic, so that the very same packet could cause the next middle box to generate a diagnostic as well, etc.

Note that there are a number of deployment scenarios one could create in which the signaling itself could create a diagnostic from a middle box. We declare such scenarios a matter of bad implementation and deployment.

8 The Stack Simplification Act of 2001

Some mechanisms such as [RSIP] significantly complicate the host stack by not giving sufficient guidance as to when the mechanism should be used.

We propose that there are only three alternatives:

1. the application relies on the host's network layer to get the packets directly to the other end;
2. the application communicates with a service that identifies failures and optionally enables a flow;
3. the application is explicitly configured to use an application layer gateway.

Option 2 is of great concern, in as much as the host must properly multiplex any messages to an application, and those messages may need to be received out of band of the normal communication.

9 Future Work

The first effort necessary is to conclude in fact whether or not additional diagnostics are necessary. If so, we must next determine the exact mechanisms to deliver those diagnostics. We must further agree on whether or not additional discovery mechanisms should be employed.

This document focuses on unicast based applications. We believe it provides sufficient flexibility to allow for design of multicast applications that take advantage of those building blocks, but more study is clearly needed.

10 Security Considerations

There are numerous security considerations that middle boxes will encounter, and the ones we list below should be viewed as far from complete.

Any time a request is made for information or for a configuration change it should be viewed with great suspicion. It is as of yet unclear all the attacks that can be made using either the signaling mechanism proposed in this document or the diagnostic messages proposed in [BLOCKER].

To minimize the risk of attacks, this mechanism should only be used in conjunction with strong authentication and a conservative authorization model. The lower bound of risk is likely that of today's model, where sites generally allow outbound TCP connections.

The signaling, diagnostics, and discovery discussed in this draft are useful only within the boundaries of a single administrative domain. The middle boxes on the borders of that domain should prevent external devices from participating by not transmitting diagnostic messages outside, and by not listening for signaling requests on interfaces external to that domain.

Furthermore, intrusion detection systems would be well advised to look for such requests as an indication of either configuration error or a possible attack.

The period of time between the termination of a communication and the termination of pinholes in firewalls may allow for mischief. End hosts must be prepared to ignore unsolicited traffic to the ports involved.

11 IANA Considerations

The names of attributes and the format of their contents that a middle box can either furnish or request will need to be held in a registry.

12 References

[1] Rekhter et. al., "Address Allocation for Private Internets", RFC 1918, February 1996.

[4] H.323

[6] Carpenter, B., ..., RFC 2775

[7] Postel, J., "Internet Control Message Protocol", RFC 792, USC/Information Sciences Institute, September 1981.

[9] Postel, J., ed., "Transmission Control Protocol - DARPA Internet Program Protocol Specification", RFC 793, USC/Information Sciences Institute, NTIS AD Number A111091, September 1981.

[11] Postel, J., "User Datagram Protocol", RFC 768, USC/Information

Sciences Institute, August 1980.

RFC 2663
BLOCKER
RSIP
DHCP
NAT-FAILOVER (Yakov, et. al).

13 Author's Address

Eliot Lear
Cisco Systems, Inc.
170 W. Tasman Dr.
San Jose, CA 95134-1706
Email: lear@cisco.com
Phone: +1 (408) 527 4020

14 Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

16 Full Copyright Statement

17 Expiration Date

This memo is filed as <draft-lear-middlebox-arch-00.txt>, and expires July 3, 2001.