



Network Protection for Convergence Applications: How FortiGate's Application-Aware Address Translation Technology Enables Secure VoIP

White Paper

August, 2002

Abstract

The increasing adoption of voice over IP (VoIP) technology in the enterprise environment is spawning renewed interest in converged voice/data applications. However, the limitations of conventional security technologies are a significant impediment to the rapid, broad roll-out of converged services. Specifically, conventional network address translation (NAT) technology, which is used almost universally by organizations to prevent unauthorized access to their private networks from the Internet, is incompatible with VoIP technology. This forces organizations to compromise security, convenience, or both – and in many cases renders converged applications impractical. The paper studies and reveals the fundamental cause the problem – the lack of content-level processing capability by conventional firewalls when performing header-only NAT – and describes how Fortinet's Accelerated Behavior and Content Analysis System (ABACAS™) technology enables secure VoIP and converged applications for enterprises and Service Providers. Using dedicated high performance hardware and patent-pending, ASIC powered content processing, Fortinet's FortiGate Network Protection Gateways enable businesses of all sizes to successfully build and deliver the highest class of security in their converged data and voice networks.

© 2002 Fortinet, Inc. All rights reserved.

The information contained in this document represents the current view of Fortinet, Inc. on the issues discussed as of the date of publication.

This document is for informational purposes only. FORTINET MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) or for any purpose, without the express written permission of Fortinet Corporation.

Fortinet may have patents, patent applications, trademarks, copyrights or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Fortinet, the furnishing of this document does not give you any license to these patents, trademarks, copyrights or other intellectual property.

Fortinet, FortiGate, FortiContent, are either registered trademarks or trademarks of Fortinet Inc., in the United States and/or other countries.

Fortinet Inc.

3333 Octavius Drive

Santa Clara, CA 95054; USA

Content

The Problem: Incompatibility Between VoIP and Conventional NAT	4
The Solution: FortiGate System's Intelligent, VoIP-Aware NAT	7
H.323 Case Study, Phase A: Call Setup	9
H.323 Case Study, Phase C: Logical Media Channel Setup	14
Summary	16
For More Information	17

An Introduction to VoIP and Conventional NAT

Voice Over IP (VoIP) – Lowering Costs, and Enabling New Applications

Organizations are accelerating the shift to running voice, data, and video on converged networks as the rapid development of the Internet has brought about the proliferation and maturation of Internet telephony. Voice over IP (VoIP) technology enables the Internet to be used as a transmission medium for telephone calls. For organizations with fixed-price Internet access of sufficiently high bandwidth and quality, Internet telephony provides extremely low cost telephone calls anywhere in the world. In addition, Internet voice communication uses the same infrastructure as Internet data communications, and is the key enabler for converged voice/data applications such as Web conferencing, teleworking, workgroup collaboration, distance learning, distributed call centers, and many others that are not well supported by conventional, dial-up voice communications. As a result, both enterprise users and service providers are finding a compelling value proposition for VoIP technology compared with traditional, Public Switched Telephone Network (PSTN)-based telephony. VoIP is enthusiastically embraced by both enterprise users and service providers, especially in Asia, and is rapidly spreading to other parts of the world.

VoIP Signaling Standards

Successful VoIP communications requires several key technologies, which includes the so-called “signaling and call control protocols” that are used to establish, route, transfer, and end calls. Today, there is no single standard signaling protocol for VoIP; In fact, there are 4 approaches in use today, one or more of which may be supported by each VoIP equipment vendor. The four protocols are:

- H.323
- Session Initiation Protocol (SIP)
- Media Gateway Control Protocol (MGCP - H.323(????))
- H.248/Megaco

A complete description and analysis of the four signaling protocols is beyond the scope of this paper. However, the following generalizations reflect the current status of the VoIP industry:

- The dominant signaling and call control protocol in IP voice networks today is H.323, which is governed by a set of International Telecommunications Union (ITU) specifications. Many service providers use H.323 to deploy packet voice services, because H.323 enables them to blend IP voice networks with the public switched telephone network (PSTN).
- Session Initiation Protocol (SIP) is finding its way into service provider networks to support new IP services, such as IP Centrex. SIP is considered by some to be the next-generation H.323, but simpler and more modular. SIP leverages the Web’s email address and URL structures in place of phone numbers.
- MGCP and Megaco were designed to provide an architecture where call control and services could be centrally added to a VoIP network. In that sense, an architecture using these protocols closely resembles the existing PSTN architecture and services.

Network Address Translation (NAT) Technology

Despite the attractive ROI and rapid development of converged networks set for running VoIP technology, there are still a few obstacles associated with VoIP. These obstacles include NAT traversal, lack of a standard signaling and call control protocol, and security.

NAT stands for Network Address Translation, an Internet standard that enables a local-area network (LAN) to use one set of IP addresses for internal traffic and a second set of addresses for external traffic. This is important because the Internet has a limited number of unique addresses, just as there are a limited number of unique, public telephone numbers. For conventional voice telephony, most companies have a small number of external phone lines, and designate a much larger number of private, internal extensions that reach each internal telephone. In the same way, the number of public Internet addresses assigned to most companies is far fewer than the number of private network addresses on their LAN. With NAT, a public IP address is assigned, usually dynamically, to support each Internet “connection” (e.g. Web or email request) initiated by an internal user – just as an available outside line is selected by a phone system when an employee attempts to make an outside call. The NAT function is sometimes provided by a stand-alone NAT device, but is more commonly a feature of another device, such as a router, firewall, or VPN gateway.

In addition to leveraging a small number of public addresses across a large user population, NAT provides several additional benefits. In particular, NAT provides an important security function, because the internal network addresses on the private LAN are hidden from the public network. With NAT, it is extremely difficult for someone outside the company to initiate an unsolicited data communications session with an internal PC or server. This eliminates a key entry point for potential intruders.

In a typical NAT implementation, the NAT gateway keeps a table of address mappings between internal addresses and ports and external addresses and ports. When an internal system sends a packet, the gateway rewrites the source address and port in the packet header using the mapping in the mapping table before forwarding the packet to the Internet. This makes it possible for return information (the response) to be routed back to the NAT gateway. When a packet arrives at the gateway from outside, a reverse process happens: The destination address is translated back to a private, internal LAN address and the packet is routed to the originating internal system.

The Problem: When VoIP Meets NAT...

In a customer data network environment where they have implemented a NAT-enabled firewall, H.323 as well as all other media signaling protocols must traverse the NAT device, which translates the IP and port address from private to public IP address.

As the name suggests, NAT is a network-level function: NAT address mapping decisions are made solely based on so-called network-level information in the “header” of the packet, which contains addressing information, but does not carry the application-level data “payload” of the packet. For most common Web protocols, such as HTTP (i.e. Web traffic), the NAT function does not need to use any information in the packet payload in order to function properly and does not interfere with communications between parties on either side of the NAT gateway.

The H.323 protocol is unique, because addressing information – specifically, the IP addresses and port numbers of the calling and called parties – are not contained in the packet header, but rather are contained in the payload portion of the packet. Since a standard NAT device only translates IP addresses in the message header, the signaling and Real-Time Protocol (RTP) media stream addresses are not translated properly. The consequence – turning NAT on blocks VoIP traffic from arriving at the intended destinations.

One approach to this problem is to leave “holes” in the firewall that allow the VoIP traffic to penetrate to the private network without using NAT. Such a practice can be easily exploited by intruders and flies in the face of even the most basic security practices. Another approach is to route VoIP phone calls through an encrypted VPN tunnel, but this also do not solve the problem for two reasons: First, IP telephony applications must be accessible at all times and cannot tolerate more than 20 milliseconds of latency, which renders many conventional VPN technologies inadequate. Second, many voice communications are ad-hoc, and do not take place between parties that have the kind of pre-established trust relationship that is typical with participants in a VPN. To limit secure VoIP only to VPN partners would be to severely limit the practical value of Internet telephony.

The Solution

To enable H.323 in conjunction with NAT requires a new, intelligent NAT solution, which goes beyond packet header address translation. Secure VoIP requires an application-aware NAT device that can recognize VoIP traffic and can process the application-layer information in the payload in real time.

The FortiGate Network Protection Gateways (NPGs) provide both network layer and application layer protection, and through their content-aware proxy architecture, understand and parse VoIP packets at the application layer, thereby solving the VoIP NAT traversal problem. FortiGate NPGs maintain high level security while enabling VoIP call setup and call transmission at very high performance. The products are available today with support for H.323 and SIP; additional VoIP signaling protocols will be added in future releases.

Note: The problems with H.323 and NAT affect all of the VoIP signaling protocols mentioned above: SIP, MGCP, and MegaCo. In the rest of this paper, H.323 will be used to illustrate the NAT traversal problem and how the FortiGate solution solves the problem. Similar technology is applied by FortiGate to solve the same problem faced by SIP and MGCP.

FortiGate NPGs Provide an Intelligent, VoIP-Aware NAT Solution

The FortiGate NPG provides a comprehensive set of network protection functions in a single unit. The ASIC-based FortiGate NPGs provide real-time antivirus protection, content filtering, firewall, VPN, intrusion detection, and traffic shaping. Because all FortiGate units are “application aware”, they also provide, intelligent, VoIP-aware NAT services. FortiGate units recognize H.323 messages based on the standard port 1720 port address and parse the body of H.323 messages. When it recognizes specific H.323 messages that carry the RTP/RTCP addresses, the FortiGate NPG creates the IP/Port mapping for the RTP traffic, which allows proper identification and address mapping of the RTP media stream between the calling and called devices in the private IP space and call host or IP gateways in the public space. (Note: Readers of this section are assumed to possess moderate level of knowledge of VoIP, RTP, H.323, H.225.0, H.245, VoIP’s deployment in an enterprise network, and basic NAT technology.)

FortiGate Core Intelligence: H.323 Message Parsing

This section together with the following several sections describe H.323 message parsing and show how a typical Microsoft Netmeeting session setup is translated. Netmeeting uses the H.323 protocol for signaling and data transmission. VoIP calls through an IP PBX operate in a slightly different but similar fashion.

[NOTE: IN ALL PICTURES, PLEASE CHANGE LABEL TO “FORTIGATE NETWORK PROTECTION GATEWAY” vs. FORTIGATE SECURITY GATEWAY]

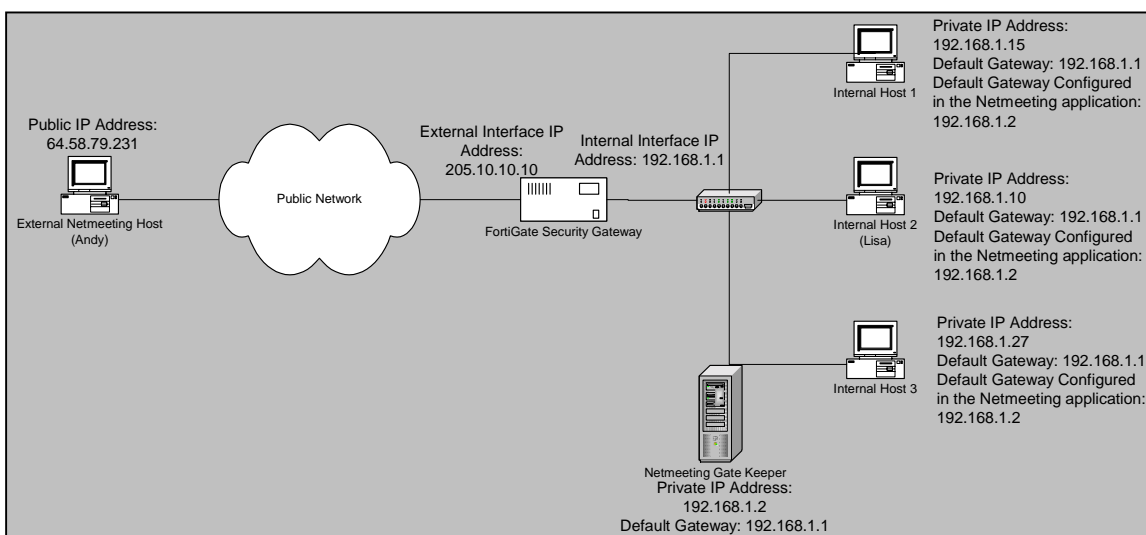


Figure 1: Network environment for the case study

Figure 1 shows the network topology involved in the Netmeeting session. In the middle sits a FortiGate unit, which acts as a network protection gateway between the external network (“Public Network” as it is called in the diagram) and the internal network, which has multiple local hosts (named Internal Host 1 through 3 in the diagram). For convenience, we have named Internal Host

2 as “Lisa”. Also sitting on the private network protected by the FortiGate NPG is the NetMeeting “Gatekeeper” server.

Introduction to the H.323 Gatekeeper

The Gatekeeper (GK) is used in the context of the H.323 protocol (and is used in other media control protocols as well). The GK is considered the “brains” of an H.323 signaling system. Typically a GK is required when more than one local host needs to receive or initiate VoIP calls or host/participate in Netmeeting sessions. The Gatekeeper keeps a registration table where the private IP addresses of the local hosts are mapped to the “Alias” of that particular local host. In a Netmeeting session, the Alias is used to identify the party involved in a call. In our example, assuming the email addresses of the users of the internal hosts are used in Netmeeting to uniquely identify the participants, the table that the GK keeps will look like the following:

Local Host Private IP	User Alias	Other User Information
192.168.1.15	doug.smith@lebeis.com
192.168.1.10	Lisa.Kelly@lebeis.com
192.168.1.27	Joe.han@lebeis.com
.....

Note that we assume that doug.smith@lebeis.com, lisa.Kelly@lebeis.com, and joe.han@lebeis.com are the email addresses of the three users using the three machines on the local network as illustrated in the above chart. On receiving an H.323 related message directed to it by the FortiGate gateway, the Gatekeeper will use the above mapping table to direct the message to the appropriate local host based on the User Alias value, which is used in the Netmeeting session as an identifier of a person to call. The table in the Gatekeeper is empty when no Netmeeting client has been started on the local network. Note that if multiple Gatekeepers are deployed across a company’s WAN, they synchronize between themselves for registered user information.

Gatekeepers communicate with end points (local hosts in this case) using a protocol called RAS, or Registration, Admission, and Status. RAS is a straightforward protocol used by endpoints to register with a Gatekeeper for the purposes of call setup with another H.323 endpoint. In our example, as soon as a user, let’s say, Lisa, starts her Netmeeting client on her local host 192.168.1.10, her private IP address and her User Alias (stored in the Netmeeting client) lisa.Kelly@lebeis.com will be entered into the gatekeeper’s table.

The following sections outlines the entire process of Netmeeting setup using H.225.0, a sub-protocol of the H.323 protocol designed for call setup and teardown. Eight steps are described in Figure 2 that constitute the initial part of the Netmeeting setup process. An analysis of the 8 steps highlights the problems are involved in VoIP NAT traversal, and shows how the FortiGate content-aware packet and content stream inspection solves the problem to allow VoIP traffic to traverse NAT.

In the example, we assume that Andy, the external user, is hosting a Netmeeting session. Andy has a public IP address of 64.58.79.231. Andy’s email address is

andy.huang@external.com. Lisa Kelly will be joining the Netmeeting session from behind the FortiGate gateway.

The call model of H.323

The H.323 call model supports many methods for setting up and routing calls, such as direct point-to-point (no Gatekeeper), point-to-point with routing via a Gatekeeper, multi-point, etc. This example examines the “Gatekeeper routed” model in the context of a point-to-point Netmeeting call through H.323.

The call model of H.323 consists of five phases:

1. Call setup (Phase A)
2. Initial communication between endpoints and terminal capability exchange (Phase B)
3. Establishment of audio and/or visual communication between endpoints (Phase C)
4. Request and negotiation of Call Services (Phase D)
5. Call termination (Phase E)

In the following sections we will focus on the initial stage of Phase A and part of Phase C to illustrate the potential problems VoIP call setup/data transmission will experience and show how FortiGate technology solves the problems.

H.323 Case Study, Phase A: Call Setup

The initial stage of the call setup process is illustrated below in Figure 2.

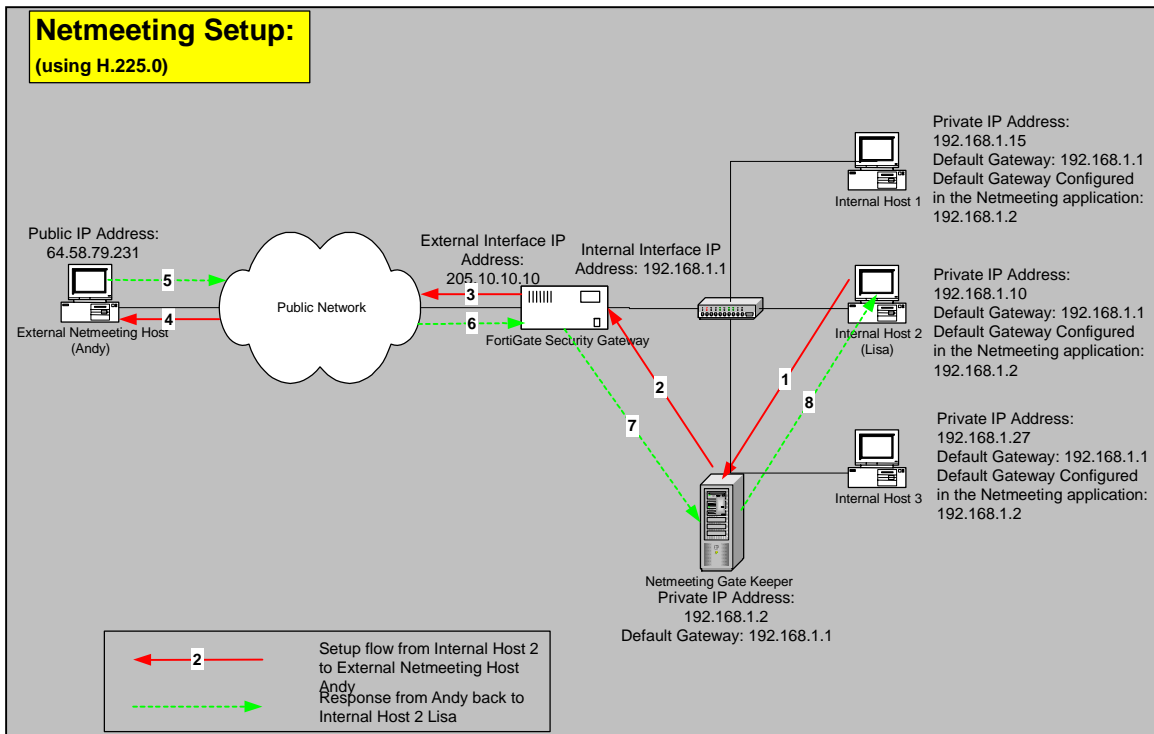


Figure 2: Netmeeting Setup

Step 1: User Registration in the Gatekeeper through RAS

As discussed earlier, to join the netmeeting hosted by Andy Huang, Lisa opens the Netmeeting client. Earlier, she enabled the “Advanced Calling Options/Use a Gatekeeper to Place a Call” in the Netmeeting client, where she entered 192.168.1.2, or the private IP address of the Gatekeeper on her local area network for her Netmeeting packets to be routed through (see Figure 3 as show below). The table in the gatekeeper will then add the mapping between Lisa’s private IP address to her alias (lisa.Kelly@lebeis.com) as a new entry. This process is completed through UDP protocol and port 1719.

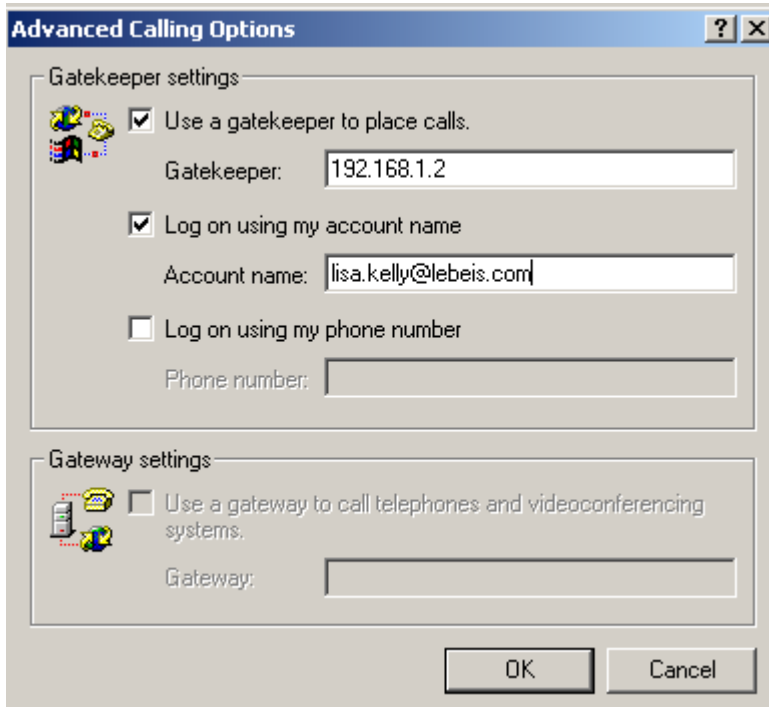


Figure 3: Netmeeting advanced call setup – Using a Gatekeeper to route calls and using account name to log onto the Gatekeeper

Once Lisa started the Netmeeting client (which in the background caused her to be registered onto the Gatekeeper), she types in Andy’s public IP address (64.58.79.231) which is used to find him as shown in Figure 4, and clicks on “Call”

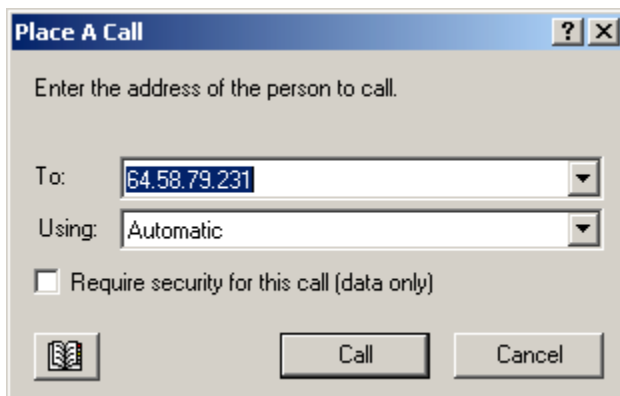


Figure 4: Lisa types in Andy's public IP address to connect to the netmeeting hosted by Andy

What happens in the background is that a H.225.0 call setup TCP packet is sent by Lisa's Netmeeting to the Gatekeeper. Note that Lisa's Netmeeting client is set up in such a way that all the call related packets (call setup, call data transmission, call teardown, etc.) will be routed to the Gatekeeper before being relayed to the network gateway connected to the external network. The TCP packet looks like this:

(Header) Source IP	Source Port	Destination IP	Destination Port
192.168.1.10	2427	64.58.79.231	1720
Message Body			
<p>.....</p> <p>Source IP: 192.168.1.10 Source Port: 2427</p> <p>...</p> <p>Alias = lisa.kelly@lebeis.com</p> <p>...</p>			

Figure 5: H.225.0 call setup TCP packet

The source IP address shows Lisa's private IP and the destination IP shows Andy's public IP address; the destination port is 1720, the H.225.0 call setup port that must be open on Andy's machine.

Note also that in the message body, there's also information about the Source IP and the Source Port – Andy, as the Netmeeting host will be using the this information in the message body to reply to Lisa to try to setup the call. This constitutes the first major VoIP NAT traversal problem that will be described in step 2. Lisa's alias is also embedded in the message body for future use.

Step 2: Call Setup Failure Using a Conventional NAT Gateway

The Gatekeeper is pre-configured to have 192.168.1.1, or the internal IP address of the FortiGate gateway as it's default gateway to the Internet. All packets it receives from the internal hosts will be relayed to the FortiGate unit (except for the initial caller registration information).

If the FortiGate handled NAT like a conventional firewall, the FortiGate unit would translate all internal host IP addresses into a public address (typically the external interface IP address of 205.10.10.10, or if an external IP address pool is available – into one of the addresses from the pool). The internal addresses such as 192.168.1.10 are not routable outside of the internal network.

A security gateway such as a conventional firewall that understands and translates only the header of the packets will change the packet into the following:

(Header) Source IP	Source Port	Destination IP	Destination Port
-----------------------	-------------	----------------	------------------

205.10.10.10	10000	64.58.79.231	1720
Message Body			
.....			
Source IP: 192.168.1.10			
Source Port: 2427			
...			
Alias = lisa.kelly@lebeis.com			
...			

Figure 6 H.225.0 call setup TCP packet NAT by conventional firewalls

Note that the source IP and the Source Port numbers have been changed from their original internal host (Lisa’s) into the external IP address of FortiGate (205.10.10.10); similarly, the Port number may or may not be changed (in this case, it’s changed from 2427 into 10000). Note the Source IP and Source Port in the message body are not touched because conventional firewalls don’t look beyond packet headers.

The problem arises when Andy’s Netmeeting application tries to talk back to Lisa to set up the call – remember we mentioned earlier that it is the Source IP and Source Port embedded in the message body that will be used to locate the host at the other end of Andy’s Netmeeting call. Andy’s application will then try to send a TCP packet back to destination address 192.168.1.10 with destination port = 2427. This packet will be lost because the address 192.168.1.10 is not routable outside of Lisa’s internal network. The result – the reply packet sent back by Andy will be lost and the Netmeeting call attempt will be aborted.

In essence, the VoIP NAT traversal problem arises because the media transmission and signaling protocols such as H.323, SIP, MGCP, and MegaCo make extensive use of the information embedded in the message body in packets rather than rely on the packet header for routing. Conventional firewalls understand and process only the header with network address translation and don’t touch the message body. The other side of the media session will be using the wrong information to route its reply back and therefore communication attempts will fail.

Step 2 Revisited: Call Setup Success Using a FortiGate Gateway

The FortiGate unit’s application-aware NAT technology avoids the problems experienced using a conventional, network-level NAT solution. The FortiGate unit receives the TCP Netmeeting call setup packet from the Gatekeeper, and by parsing the message, recognizes the H.323 call setup request. As a result, besides the normal network address translation that FortiGate unit performs on every outgoing packet, it performs a “message body address translation” as well by replacing the source IP and source port in the **message body** with an externally routable IP and port number, as illustrated in Figure 7.

(Header) Source IP	Source Port	Destination IP	Destination Port
205.10.10.10	10000	64.58.79.231	1720
Message Body			

```

.....
Source IP: 205.10.10.10
Source Port: 10000
...
Alias = lisa.kelly@lebeis.com
...

```

Figure 7: H.225.0 call setup TCP packet following content aware NAT by the FortiGate unit. Note that the “Source IP” address in the message body has been translated by the FortiGate unit to the publicly routable external address of the FortiGate gateway.

All the above processing is completed within the proprietary purpose-built FortiOS™ operating system kernel and therefore latency introduced as a result of the packet parsing is negligible.

Step 3, 4, 5, and 6: Andy Receives and Responds to the Call Setup Request

Once the header-level NAT is completed together with the address translation in the message body, the call setup request (TCP packet destined towards Andy’s port 1720) will be sent to the external network that FortiGate is connected to and routed to Andy’s machine at public IP address 64.58.79.231. Andy’s machine is listening on port 1720 for call setup requests. It picks up the setup request from Lisa, removes the header, and looks into the message body to pick out the Source IP and Source Port from there. It understands that the Lisa can be reached at Source IP address 205.10.10.10 (which is the external interface address of FortiGate) and at port 10000. It also understands that Lisa is called on her in Netmeeting by her Alias lisa.Kelly@lebeis.com. It then sends a TCP call setup reply back to Lisa as illustrated in Figure 8:

(Header) Source IP	Source Port	Destination IP	Destination Port
64.58.79.231	1720	205.10.10.10	10000
Message Body			
<pre> Source IP: 64.58.79.231 Source Port: 1720 ... Call host alias = andy.huang@ocsic.com Call Participant Alias = lisa.kelly@lebeis.com ... </pre>			

Figure 8: H.225.0 call setup TCP packet following content-aware NAT by the FortiGate unit

The reply gets routed through the public network and arrives at the FortiGate external interface.

Step 6-7-8: The FortiGate Unit and Gatekeeper Implement Intelligent H.323 Call Setup

When the call setup reply arrives at the FortiGate unit’s external interface, several things happen:

1. The FortiGate unit parses all incoming packets and recognizes the H.225.0 call setup reply packet by looking at the header of the TCP packet – all Source Port = 1720 packets are by design H.225.0 call setup packets.

2. The FortiGate implements “Port Forwarding,” – in this case it has been configured to route all packets with a Source address = 1720 to the Gatekeeper.
3. The Gatekeeper, on receiving the packet, understands that the packet is for call setup and is a reply from the host of the Netmeeting session. The Gatekeeper parses the message and understands that the call participant’s alias is lisa.kelly@lebeis.com. It then looks it up in its mapping table and finds the private IP for Lisa at 192.168.1.10. The reply packet is then relayed onto Lisa’s machine.
4. Once the call setup reply from Andy arrives at Lisa’s machine and is appropriately processed, the initial H.225.0 call setup is completed.

What Happens If Andy Calls In as A Participant and Lisa Hosts a Netmeeting...

Up to now, we’ve focused on the example where Lisa calls out to join Andy’s netmeeting. What if Lisa hosts a Netmeeting and Andy calls in to join? In this situation, all that we need to do is to turn on the H.323 support in FortiGate gateway – the same as when we want to enable Lisa to call out. No port will have to be permanently opened in the firewall to enable Andy’s request to get through.

When Andy initiates a Call setup request to join Lisa’s meeting the request TCP packet will arrive at the FortiGate gateway with a destination port of 1720. The FortiGate unit will use port address forwarding to redirect the packet to the Gatekeeper, which will in turn forward it onto Lisa’s host (based on Lisa’s alias of lisa.kelly@lebeis.com) for further processing. Lisa’s reply will be going through similar address and port translation as described above. This way security is not jeopardized while incoming calls are allowed.

H.323 Case Study, Phase C: H.323 Establishment of Media Communication – H.245 Logical Channel Setup

The initial stage of the call setup process is illustrated below in Figure 9.

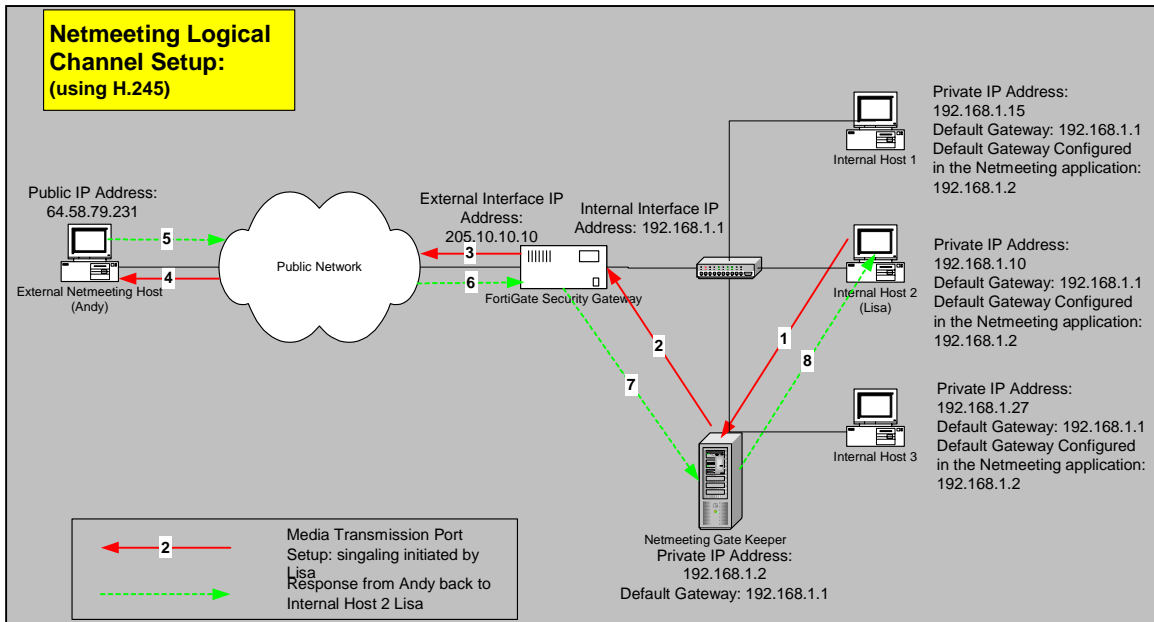


Figure 9: Netmeeting call logical channel setup

Once the initial call setup is completed, both sides can exchange information to set up the Master/Slave status in the call and also tell each other what kind of capabilities (video, audio, etc.) are available on both sides. The next stage would be to set up the logical channel for RTP, or Real Time Protocol, which carries the voice and/or video channels. The most important part in logical channel setup is the port address exchange so that both sides know which port to direct the media traffic to in media transmission. As before, a step by step analysis highlights the relevant FortiGate operations that make facilitates the port setup process.

Step 1, 2, and 3: Lisa Initiates a Logical Channel Setup Message

RTP is the protocol used in H.323 for media data transmission, including both video and audio. Logical channel setup focuses on exchanging UDP port information to the parties involved in the call to set up the port for media data exchange. This process is accomplished through protocol H.245, a sub-protocol in H.323.

In our example, Lisa will send a logical channel setup UDP packet to Andy. The packet is illustrated in the following figure (9):

(Header) Source IP	Source Port	Destination IP	Destination Port
192.168.1.10	5005	64.58.79.231	...
Message Body			
.....			
Source IP: 192.168.1.10			
Source (RTP) Port: 5005			
...			

Figure 9: H.245.0 logical channel setup UDP packet illustration

Note that the message body carries the source IP address as well as the port address to be used for RTP data transmission. Similar to Phase A in Call Setup, without appropriate replacement of the IP and Port in the message body, Andy won't know the appropriate port number for Lisa and therefore the media data transmission piece won't go through. This would be the case if a conventional NAT/firewall were used, and thus the call would fail.

The FortiGate unit inspects both the header and message body of contents, and through its proprietary masquerading function, it will generate an IP address/Port pair that will be used to replace the Source IP/Source Port in the message body as well as in the header. The mapping will be entered into its mapping table as illustrated below:

Source IP	Source Port	Translated IP	Translated Port
192.168.1.10	5005	205.10.10.10	10000
...

The UDP packet after the translation will look like the following:

(Header) Source IP	Source Port	Destination IP	Destination Port
205.10.10.10	10000	64.58.79.231	...
Message Body			
..... Source IP: 205.10.10.10 Source (RTP) Port: 10000 ...			

This new packet carrying translated address and port information will then be sent to Andy. Note that the ports that opened for the media transmission session will be opened only for the duration of the session and will be dynamically closed immediately after the session is over. This eliminates the severe security vulnerabilities that are introduced when pre-defined ports are left open permanently, as is often the case using conventional NAT/firewalls.

Step 4, 5, 6: Andy's Netmeeting Client Responds to Lisa's Logical Channel Setup Request

On receiving Lisa's logical channel setup request, Andy's Netmeeting Client will respond by sending a reply carrying its own port address, and will send the packet to address 205.10.10.10 and port 10000 as indicated in Lisa's message body. The response will travel across the public network and arrive at the FortiGate gateway.

Step 7, 8: Andy's Response Intelligently Routed to Lisa

Once the response arrives at FortiGate, it will use the mapping table to translate it into the Private IP as well as the private port address corresponding to Lisa's machine and Netmeeting application.

Similar to Phase A Call setup, using port address forwarding the logical channel setup response from Andy will then be forwarded to the Gatekeeper, which will relay (without further processing) the response to the correct port on Lisa's machine, thus completing the logical channel setup process and enabling a secure VoIP session.

Summary

Voice over IP is a major enabler of high-value, converged voice/data applications. However, widespread use of VoIP is at odds with conventional security technology: Network-level devices, such as NAT gateways, lack the intelligence to recognize and properly process the critical signaling protocols that enable VoIP calls to be established and managed. Until now, extreme compromises have been required to enable flexible VoIP applications: Companies have been forced to either compromise their security, by opening holes in their perimeter security to allow VoIP signaling to pass, or have been forced to purchase expensive, single-function call-proxy systems that focus on solving the NAT traversal problem. Another approach – using VoIP only in conjunction with VPN tunnels – greatly limits the reach of Internet telephony. None of these so-called solutions achieves the goal of enabling widespread, secure use of VoIP technology.

FortiGate Network Protection Gateways have been specifically designed to provide the application-level intelligence and ASIC-accelerated performance required for secure Internet telephony and mixed voice/data applications. FortiGate gateways support secure VoIP services for any size company – from SOHO users through large enterprises and service providers – and enable any organization to take full advantage of the benefits of converged voice/data applications.

For More Information

In the constantly evolving network security environment, the FortiGate Network Protection Gateways provide the strongest and most effective protection available for your network. More information about Fortinet's products is available from the following sources.

Business Information

Please visit us at www.fortinet.com.

Potential Partners

Please contact us at partner@fortinet.com or visit us at www.fortinet.com.

Additional Resources

Please contact us at (1) 408-235-7700 for engineering/technical support.