US 20060274741A1

(19) United States

(12) Patent Application Publication

(10) Pub. No.: US 2006/0274741 A1

(43) Pub. Date: Dec. 7, 2006

Wing et al.

(54) MANAGING DEVICES ACROSS NAT BOUNDARIES

(76) Inventors: **Daniel G. Wing**, San Jose, CA (US); **Shiva J. Shankar**, Bangalore (IN); **Ottalingam Satyanarayanan**, Fremont, CA (US)

Correspondence Address:
MARGER JOHNSON & MCCOLLOM, P.C.
210 SW MORRISON STREET, SUITE 400
PORTLAND, OR 97204 (US)

(21) Appl. No.: 11/147,713

(22) Filed: Jun. 7, 2005

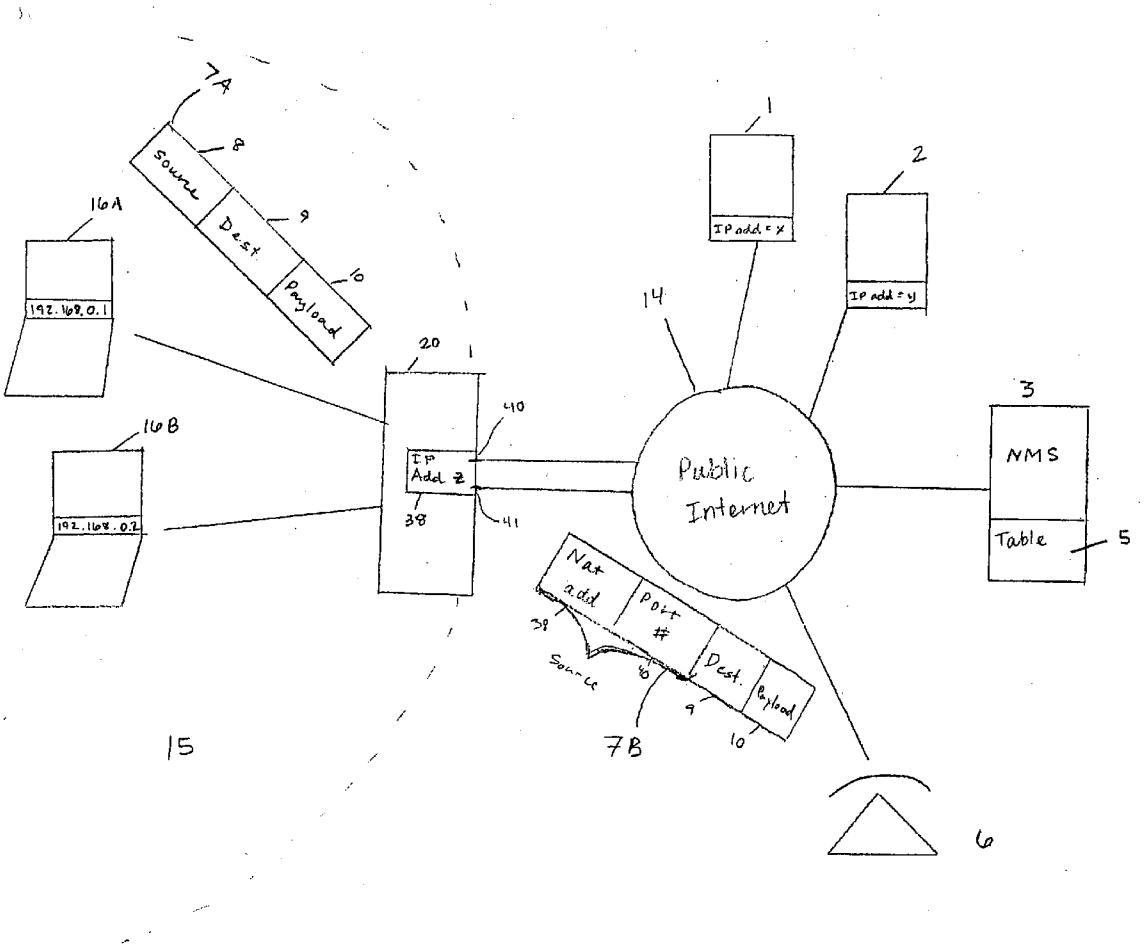
Publication Classification

(51) Int. Cl.
H04L 12/56 (2006.01)

(52) U.S. Cl. 370/389

(57) ABSTRACT

An address management scheme allows a Network Management System (NMS) to manage devices in a private network operating behind a Network Address Translator (NAT) boundary. A device operating in the private network sends a communication to a Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs) (STUN) server. The STUN server responds by communicating a public NAT IP address and a NAT port number back to the device. The device then provides the NMS with the public NAT IP address, a NAT port number associated with the device, a unique device identifier, and the private device IP address. The NMS stores this information in a table and then accesses this address information to manage the device in the private network. The device then uses the STUN server to identify any changes to the device address information and then sends the changes to the NMS.



FreeWheel, Ex. 1022

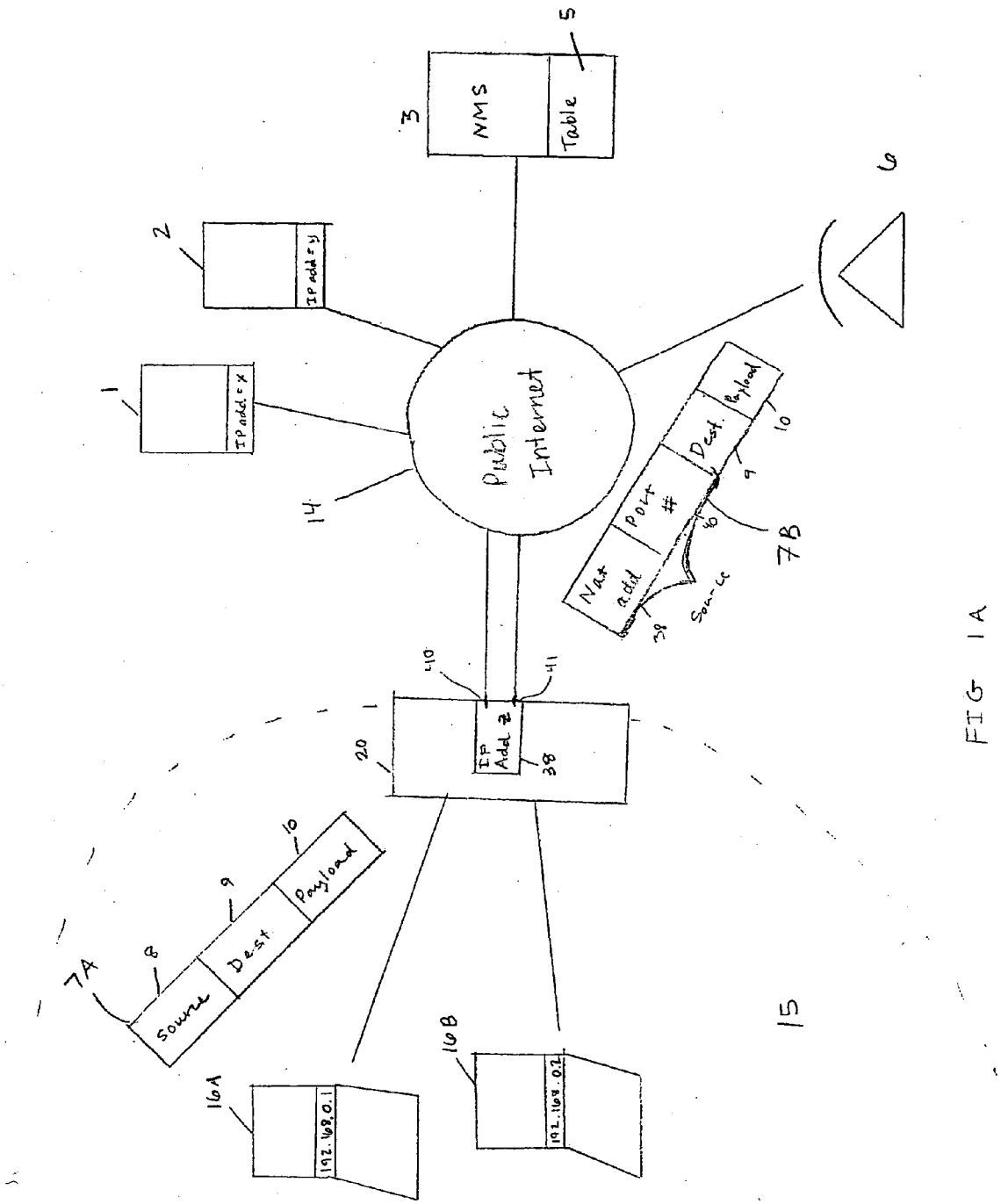

NMS
Table
IP add = y
IP add = x
Public Internet
Nat add
Port #
Dest.
Payload
Source
IP Add z
Source
Dest
Payload
192.168.0.1
192.168.0.2
1
2
3
5
6
7A
7B
8
9
10
14
15
16A
16B
20
38
40
41

FIG 1A

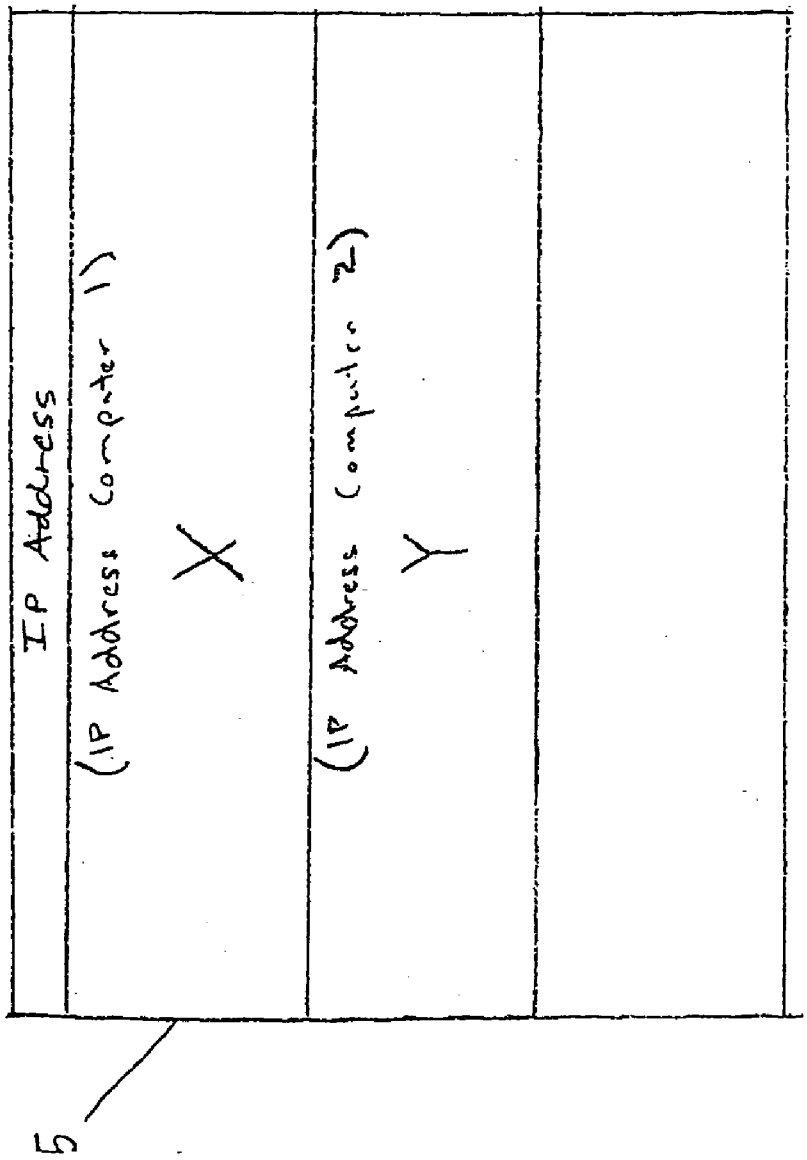


FIG 1B

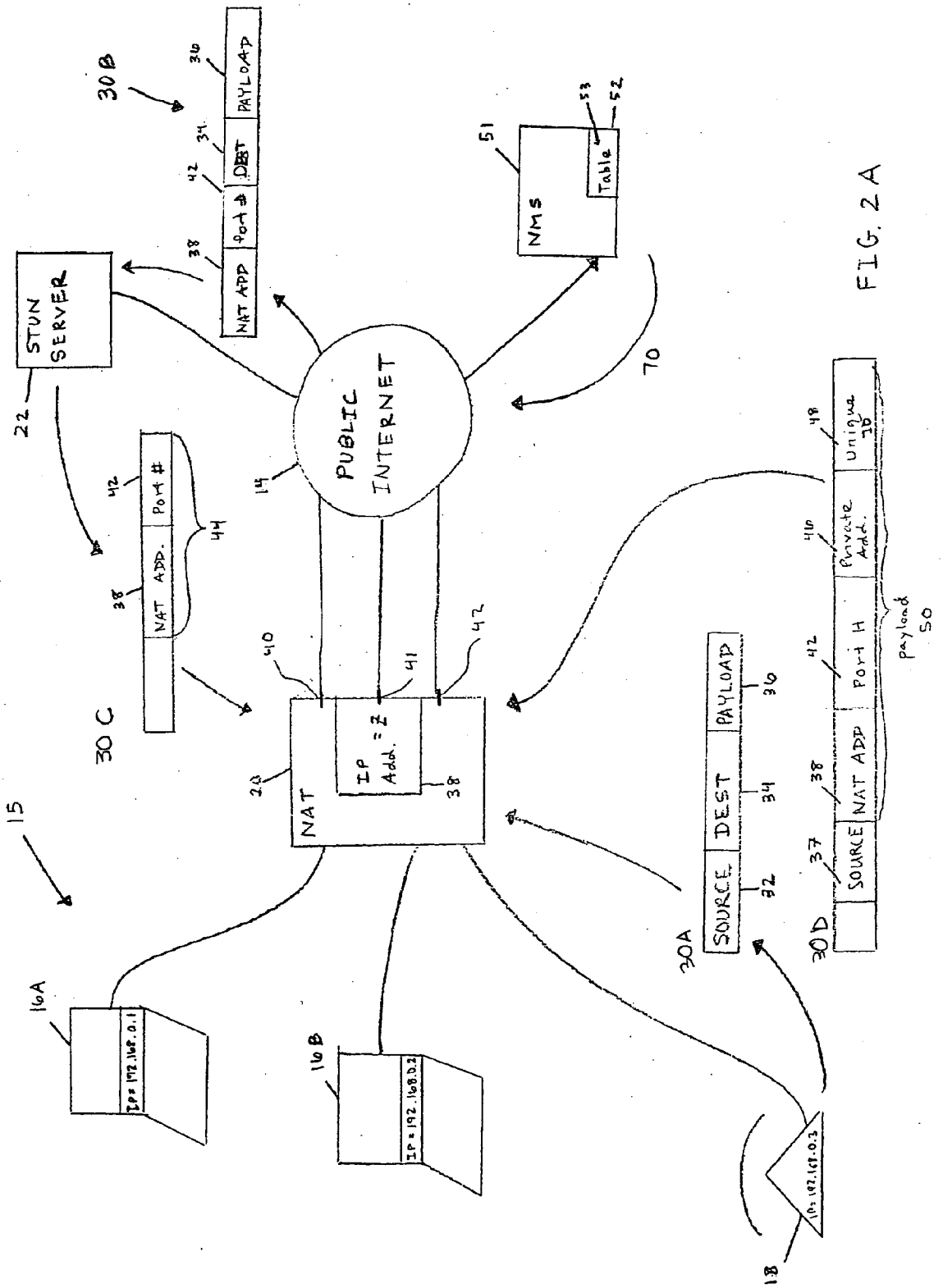

STUN SERVER
22
30B
NAT ADD
Port #
DEST
PAYLOAD
38
42
34
36
NMS
51
Table
53
52
70
PUBLIC INTERNET
14
30C
NAT ADD.
Port #
44
15
NAT
20
IP Add. = Z
38
40
41
42
16A
IP = 192.168.0.1
16B
IP = 192.168.0.2
SOURCE
DEST
PAYLOAD
32
34
36
20A
SOURCE
NAT ADD
Port #
Private Add.
Unique ID
37
38
42
40
48
30D
Payload 50
IP = 192.168.0.3
18

FIG. 2A

Device	Public IP	Port #	Unique ID	Private IP
IP phone 18	NAT 20 IP = Z 38	NAT 20 42	IP phone 18 Mac Addr. = Q 48	IP phone 18 192.168.0.3 46
Computer 16A	NAT 20 IP = Z	NAT 20 40	Computer 16A Mac Addr = R 48	Computer 16A 192.168.0.1
Computer 16B	NAT 20 IP = Z	NAT 20 41	Computer 16B Mac Addr. = S 48	Computer 16B 192.168.0.2

52

53

FIG 2B

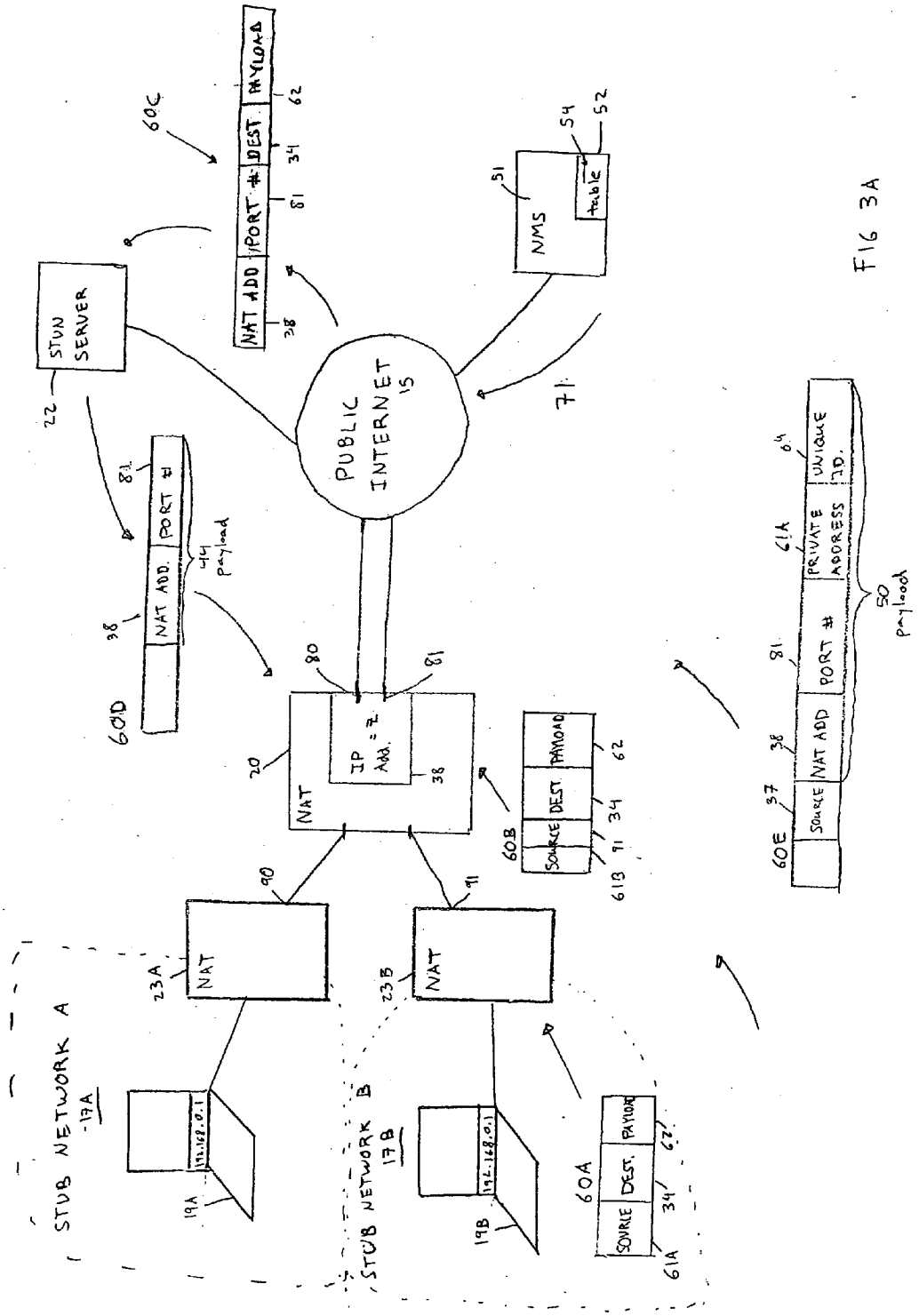

PUBLIC INTERNET 15
STUN SERVER
22
NAT ADD
PORT #
DEST
PAYLOAD
38
81
34
62
60C
NMS
table
51
54
52
71
NAT ADD.
PORT #
44 payload
60D
NAT
IP Add. = Z
20
80
81
38
SOURCE
DEST
PAYLOAD
60B
61B
91
34
62
NAT
23A
NAT
23B
90
91
STUB NETWORK A 17A
STUB NETWORK B 17B
19A
19B
192.168.0.1
SOURCE
DEST.
PAYLOAD
60A
61A
34
62
SOURCE
NAT ADD
PORT #
PRIVATE ADDRESS
UNIQUE ID.
60E
37
38
81
61A
64
50 payload

FIG 3A

Device	Public IP (38)	Port #	Unique ID (48)	Private IP (61A)
IP Phone 18	NAT 20 IP = Z	NAT 20 42	IP phone 18 Mac. add = Q	IP phone 18 192.168.0.3
16A	NAT 20 IP = Z	NAT 20 40	Computer 16A Mac. add = R	Computer 16A 192.168.0.1
16B	NAT 20 IP = Z	NAT 20 41	Computer 16B Mac. add = S	Computer 16B 192.168.0.2
19A	NAT 20 IP = Z	NAT 20 80	Computer 19A Mac. add = T	Computer 19A 192.168.0.1
19B	NAT 20 IP = Z	NAT 20 81	Computer 19B Mac. add = U	Computer 19B 192.168.0.1

52

54

FIG. 3B

FIG. 4

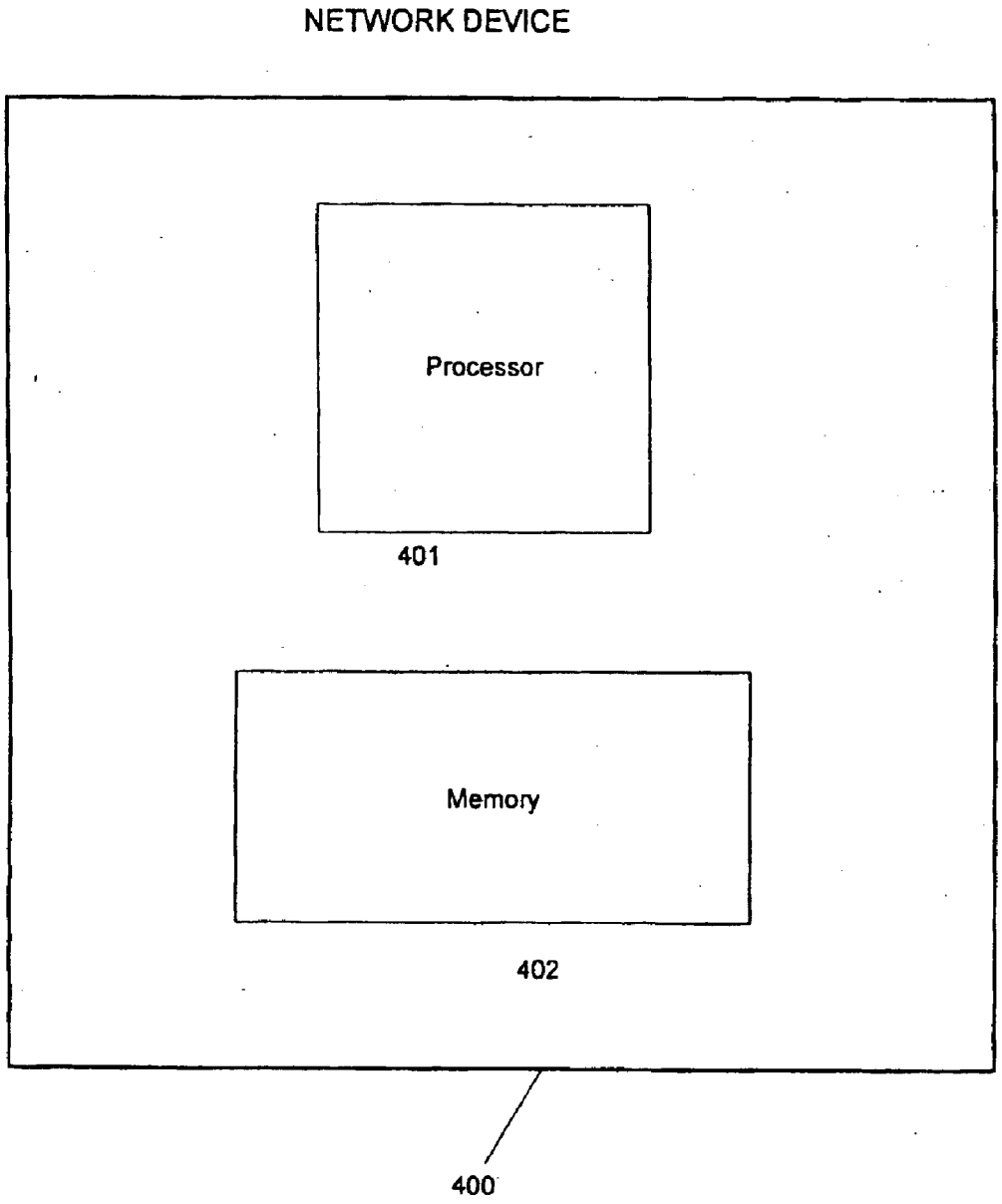
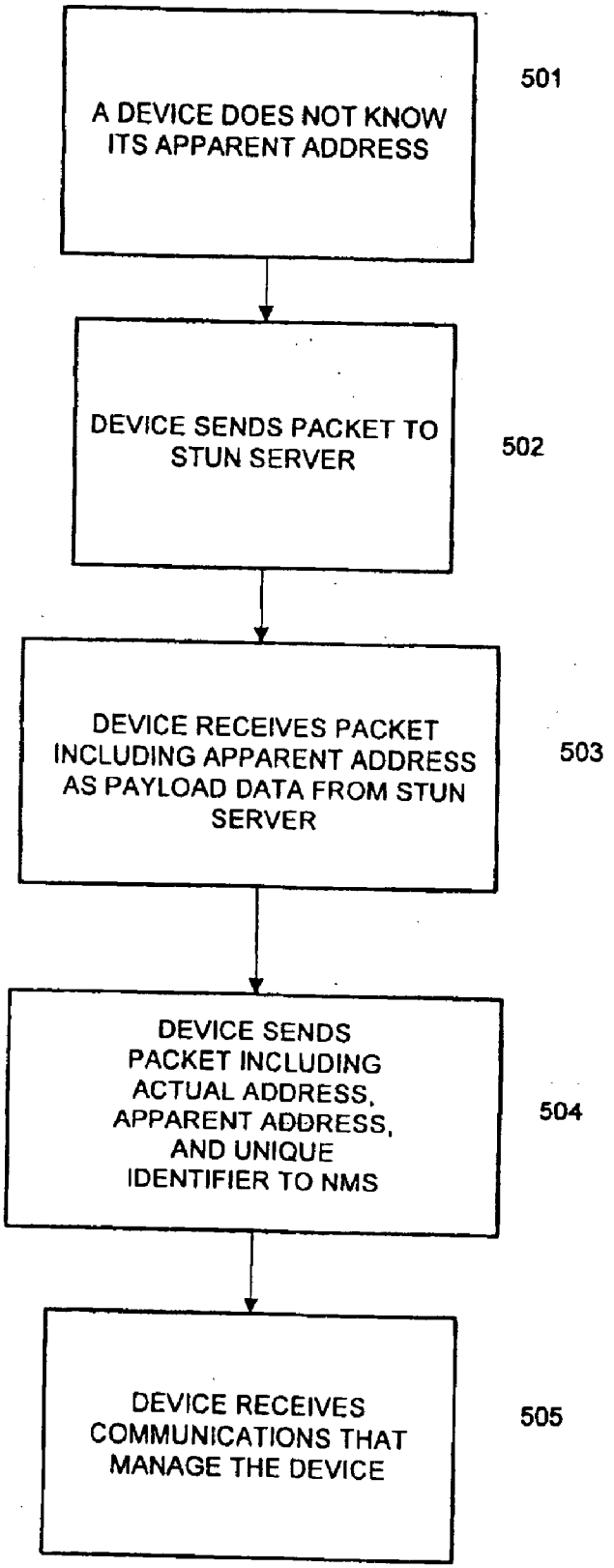


FIG. 5


A DEVICE DOES NOT KNOW ITS APPARENT ADDRESS
501
DEVICE SENDS PACKET TO STUN SERVER
502
DEVICE RECEIVES PACKET INCLUDING APPARENT ADDRESS AS PAYLOAD DATA FROM STUN SERVER
503
DEVICE SENDS PACKET INCLUDING ACTUAL ADDRESS, APPARENT ADDRESS, AND UNIQUE IDENTIFIER TO NMS
504
DEVICE RECEIVES COMMUNICATIONS THAT MANAGE THE DEVICE
505

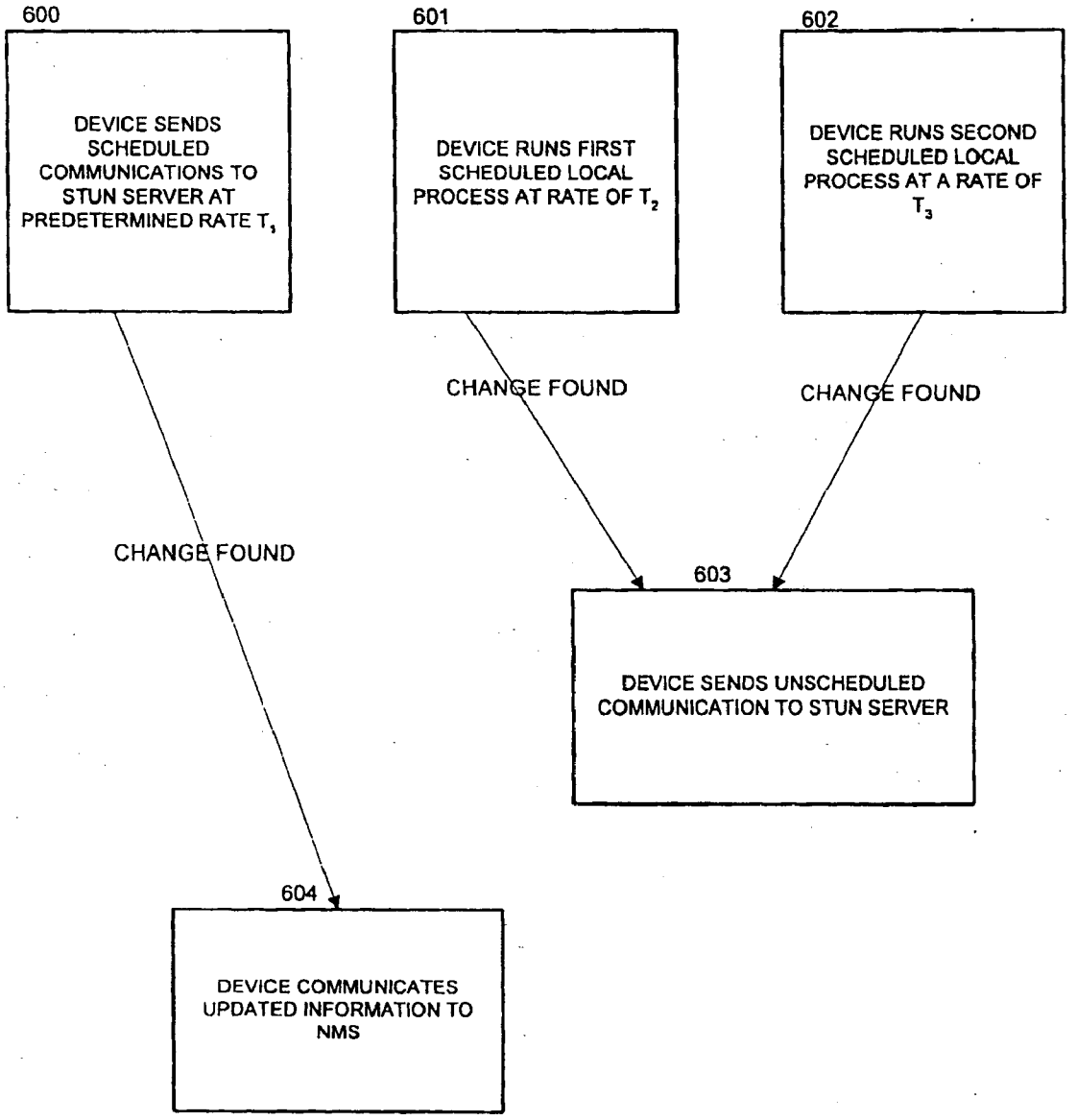

600
DEVICE SENDS SCHEDULED COMMUNICATIONS TO STUN SERVER AT PREDETERMINED RATE T_1
601
DEVICE RUNS FIRST SCHEDULED LOCAL PROCESS AT RATE OF T_2
602
DEVICE RUNS SECOND SCHEDULED LOCAL PROCESS AT A RATE OF T_3
CHANGE FOUND
CHANGE FOUND
CHANGE FOUND
603
DEVICE SENDS UNSCHEDULED COMMUNICATION TO STUN SERVER
604
DEVICE COMMUNICATES UPDATED INFORMATION TO NMS

FIG. 6

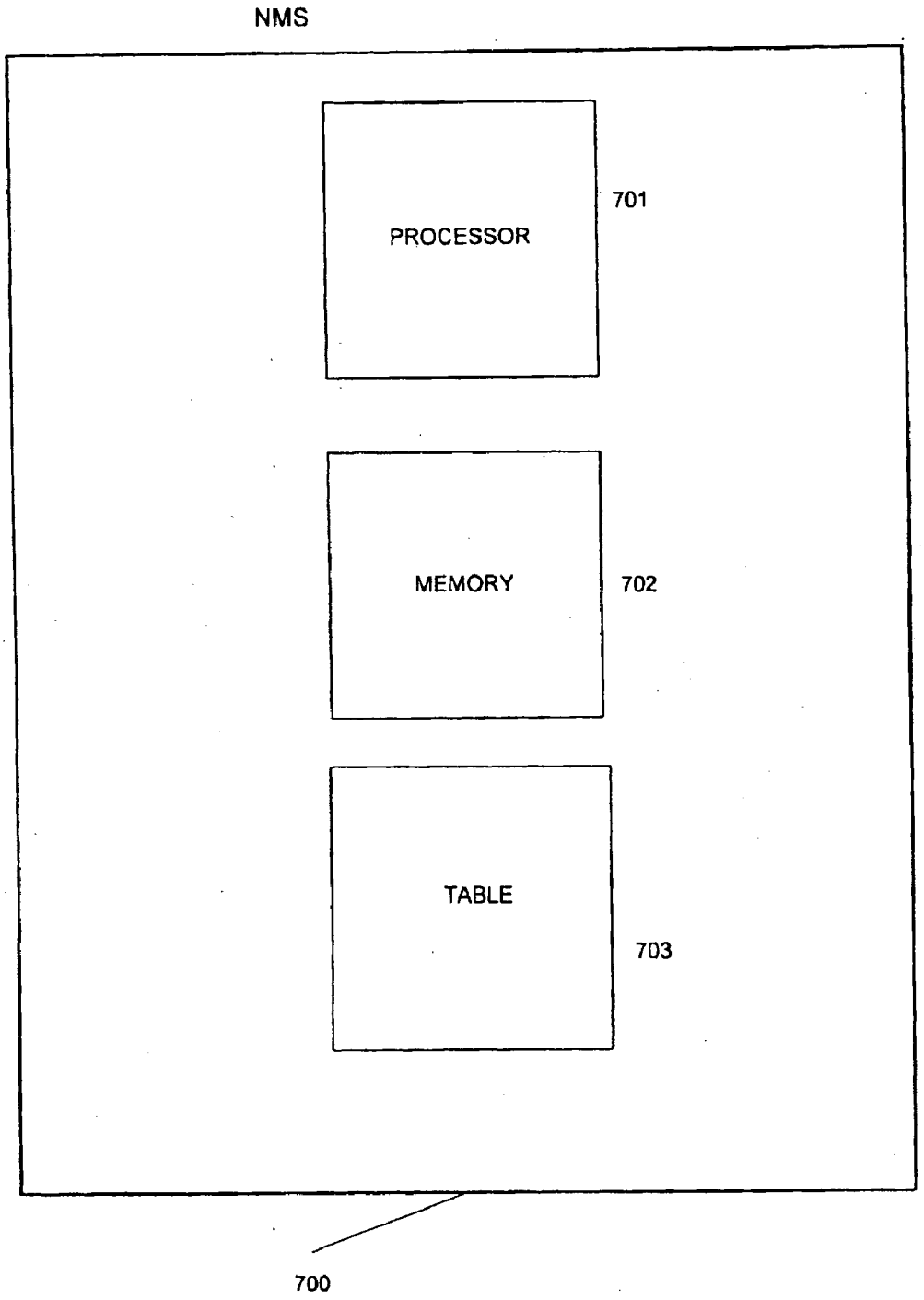


FIG. 7

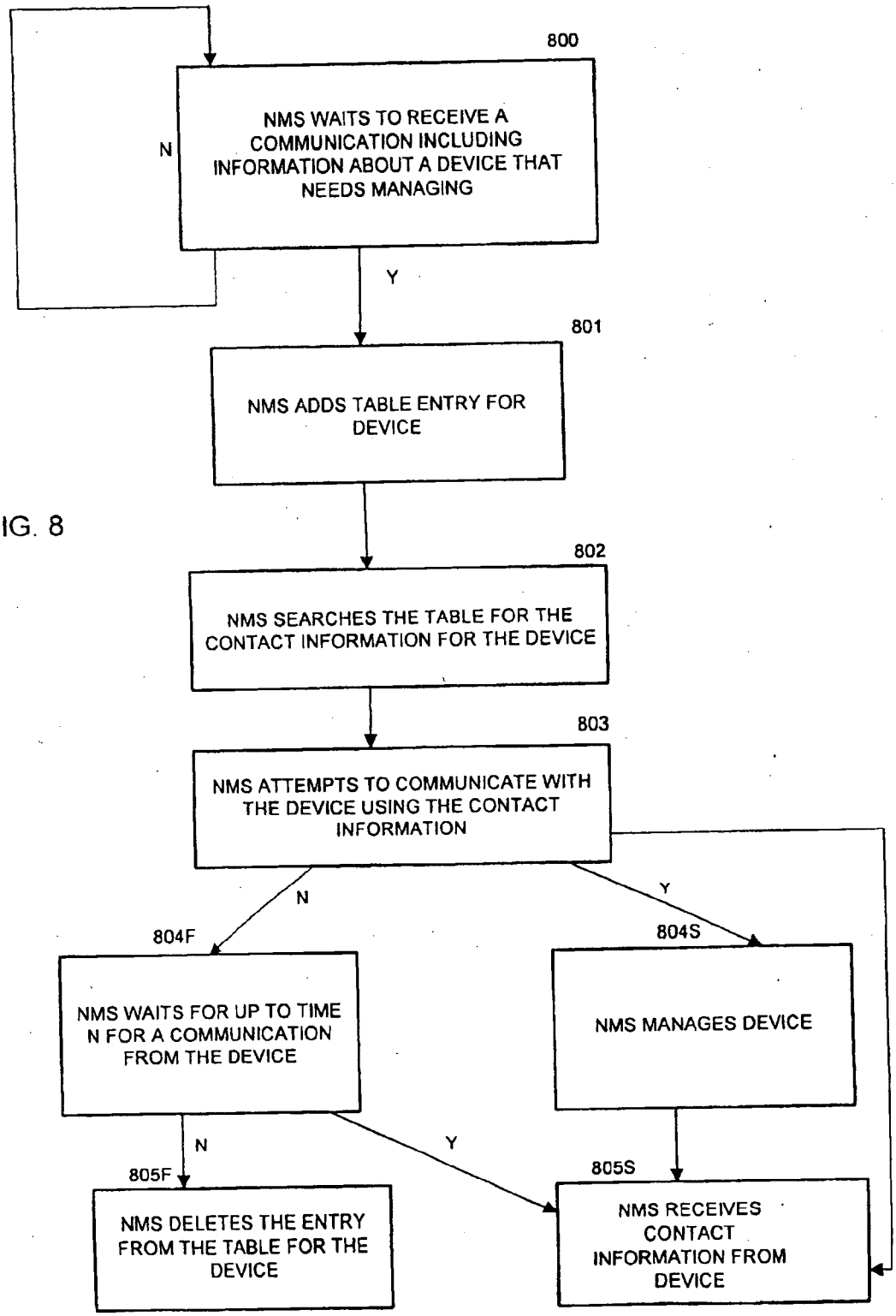

800
NMS WAITS TO RECEIVE A COMMUNICATION INCLUDING INFORMATION ABOUT A DEVICE THAT NEEDS MANAGING
N
Y
801
NMS ADDS TABLE ENTRY FOR DEVICE
802
NMS SEARCHES THE TABLE FOR THE CONTACT INFORMATION FOR THE DEVICE
803
NMS ATTEMPTS TO COMMUNICATE WITH THE DEVICE USING THE CONTACT INFORMATION
N
Y
804F
NMS WAITS FOR UP TO TIME N FOR A COMMUNICATION FROM THE DEVICE
804S
NMS MANAGES DEVICE
N
Y
805F
NMS DELETES THE ENTRY FROM THE TABLE FOR THE DEVICE
805S
NMS RECEIVES CONTACT INFORMATION FROM DEVICE

FIG. 8

MANAGING DEVICES ACROSS NAT BOUNDARIES

BACKGROUND OF THE INVENTION

[0001] This invention relates generally to network management and more particularly to managing devices across Network Address Translator (NAT) boundaries using Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs) (STUN) servers.

[0002] A Network Management System (NMS) can manage many devices including computers and Internet Protocol (IP) telephones. Management can include network management, changing system settings, recording failures of network devices, discovering what hardware components are installed in network devices, discovering what software is installed on the device, etc.

[0003] **FIG. 1A** shows an NMS **3** used for managing computers **1** and **2**. A table **5** stores the IP addresses of the devices managed by the NMS **3**. The table **5** is shown in more detail in **FIG. 1B** and includes entries listing the IP addresses of the computers **1** and **2** managed by NMS **3**. The NMS **3** communicates with the computers **1** and **2** using the public IP addresses X and Y, respectively. For example, the NMS uses IP address X in table **5** to communicate with computer **1**.

[0004] Referring back to **FIG. 1A**, Network Address Translator (NAT) **20** and computers **16**A and **16**B reside within a private network **15**. The NAT **20** has a public IP address **38** and assigns private IP addresses to computers **16**A and **16**B. The NAT **20** is designed for IP address simplification and conservation, by enabling the private IP network **15** to use non-registered (private) IP addresses. The NAT **20** operates as a router connecting the private network **15** together with the public network **14**. The NAT **20** translates the private (not globally unique) addresses used in the private network **15** into public IP addresses. As part of this functionality, NAT **20** can be configured to advertise only one public address to the public network **14** that represents the entire private network **15**.

[0005] For example, computers **16**A and **16**B communicate over Internet network **14** using the public IP address **38** provided by the NAT **20**. The NAT **20** receives a packet **7**A from a device on private network **15**, such as computer **16**A. The packet **7**A includes a private source address **8** and a destination IP address **9** for an endpoint such as IP phone **6**. Packet **7**A also includes a payload **10**. The NAT **20** reformats packet **7**A into a packet **7**B that replaces the private source address **8** with the NAT's public IP address **38** and a port number **40** that the NAT **20**, associates with computer **16**A. The NAT **20** then forwards the reformatted packet **7**B to IP phone **6**.

[0006] The IP phone **6** sends packets (not shown) back to the computer **16**A that includes the public IP address **38** and port number **40** for the NAT **20**. The NAT **20** receives and forwards the packet from IP phone **6** to computer **16**A based on the port number **40**.

[0007] The NMS **3** cannot manage computers **16**A and **16**B behind NAT **20** for several reasons. First, the table **5** in NMS **3** only includes public IP device addresses. The NMS **3** does not have the ability to obtain the private IP addresses and port numbers needed for communicating with computers **16**A and **16**B. Even if the NMS **3** could obtain the private IP addresses and port numbers associated with of computers **16**A and **16**B, these addresses are not routable from the NMS. Additionally, the private IP addresses may be dynamically reassigned whenever the NAT **20** is reset. Port numbers are also typically refreshed in unison with the private IP address reassignment.

[0008] Because of the forgoing limitations, network management servers are unable to manage devices operating in private networks behind NATs. The disclosure that follows solves this and other problems associated with the prior art.

SUMMARY OF THE INVENTION

[0009] An address management scheme allows a Network Management System (NMS) to manage devices in a private network operating behind a Network Address Translator (NAT) boundary. A device operating in the private network sends a communication to a Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs) (STUN) server. The STUN server responds by communicating a public NAT IP address and a NAT port number back to the device. The device then provides the NMS with the public NAT IP address, a NAT port number associated with the device, a unique device identifier, and the private device IP address. The NMS stores this information in a table and then accesses this address information to manage the device in the private network. The device then uses the STUN server to identify any changes to the device address information and then sends the changes to the NMS.

[0010] The foregoing and other objects, features and advantages of the invention will become more readily apparent from the following detailed description of a preferred embodiment of the invention which proceeds with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] **FIG. 1A** is a diagram of an NMS managing two devices.

[0012] **FIG. 1B** is a diagram of a table used by the NMS in **FIG. 1A**.

[0013] **FIG. 2A** is a diagram of an NMS managing devices in a private network.

[0014] **FIG. 2B** is a table used by the NMS in **FIG. 2A**.

[0015] **FIG. 3A** is a diagram showing the NMS managing two devices in a nested NAT configuration.

[0016] **FIG. 3B** is a diagram of the table used by the NMS in **FIG. 3A**.

[0017] **FIG. 4** is a detailed diagram of the device operating in a private network.

[0018] **FIG. 5** is a flowchart showing how the device operates in the private network.

[0019] **FIG. 6** is a flowchart showing how the device monitors local address changes.

[0020] **FIG. 7** is a diagram of the NMS shown in **FIGS. 2A and 3A**.

[0021] **FIG. 8** is a flowchart showing how the NMS shown in **FIGS. 2A and 3A** manages devices in a private network.

DETAILED DESCRIPTION

[0022] **FIG. 2A** shows an NMS **51** that manages devices **16**A, **16**B, and **18** in the private network **15** behind NAT **20**. The NMS **51** includes a table **52** that includes information necessary for the NMS **51** to manage devices **16**A, **16**B, and **18**. The devices **16**A, **16**B, and **18** utilize Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs) (STUN) in a novel way to ensure that the information in table **52** is adequate for the NMS **51** to manage the NAT devices **16**A, **16**B and **18**. The devices **16**A, **16**B and **18** also automatically update table **52** with any local address changes.

[0023] STUN is described in Request for Comment (RFC) 3489 which is herein incorporated by reference. STUN typically is used for NAT traversal during voice communications and with Internet telephony signaling protocols such as Session Initiated Protocol (SIP).

[0024] In the example below, IP phone **18** is managed by NMS **51**. A STUN server **22** is used by the IP phone **18** to determine the public NAT IP address **20** and to determine what NAT port number is associated with the IP phone **18**. IP phone **18** also uses the STUN server **22** to determine when the public IP address for NAT **20** has changed or when the NAT **20** has reassigned the IP phone **18** to a different port number.

[0025] The IP phone **18** sends an IP packet **30**A to the STUN server **22**. The NAT **20** replaces the private device source address **32** with the public NAT IP address **38** and NAT port number **42** before forwarding the reformatted packet **30**B to the STUN server **22**. The STUN server **22** inserts the public NAT address **38** and NAT port number **42** into the payload **44** of a return packet **30**C and sends the packet **30**C back to NAT **20**. The NAT **20** then forwards the return packet **30**C back to the IP phone **18**.

[0026] Once the NAT address **38** and port number **42** are acquired, the IP phone **18** sends this information to the NMS **51**. The IP phone **18** sends a packet **30**D to the NMS **51**. In one embodiment, a payload **50** for the packet **30**D includes the public NAT IP address **38**, the NAT port number **42**, a private IP address **46** for the IP phone **18**, and a unique identifier **48** for the IP phone **18**. The NAT IP address **38** and port number **42** may also be provided to the NMS **51** through a source address field **37** in packet **30**D instead of in the payload **50**, or in addition to inclusion in the payload **50**. The unique identifier **48** associated with the IP phone **18** may be a MAC address, certificate, user name, or any other identifier that is unique to IP phone **18** in private network **15**.

[0027] Once the packet **30**D is received, the NMS **51** adds an entry **53** for IP phone **18** to a table **52**, shown in more detail in **FIG. 2B**. Entry **53** includes NAT public IP address **38**, NAT port number **42**, private IP address **46** of the IP phone **18**, and unique identifier **48** for IP phone **18**. The unique information in entry **53** allows NMS **51** to conduct management operations **70** (**FIG. 2A**) with the IP phone **18** including changing system settings, monitoring, phone operations, etc.

[0028] Referring back to **FIG. 2A**, the NAT **20** may change the port number associated with the IP phone **18**. The IP phone **18** communicates these changes, such as the public NAT IP address **38** and the NAT port number **42**, to the NMS **51**. These changes are detected by the IP phone **18** and notified to the NMS **51** using additional communications similar to **30**A-**30**D. The frequency of the additional communications similar to **30**A-**30**D may correspond with how often the NAT **20** typically changes the port number associated with the IP phone **18**. STUN provides a method of determining an adequate frequency for additional communications as described in Request for Comment (RFC) 3489.

[0029] The IP phone **18** can also periodically determine whether the NAT **20** has reassigned the private IP address of the IP phone **18** and whether the unique identifier **48** of the IP phone **18** has changed. In one embodiment, the IP phone **18** uses STUN to determine if the private IP address of the IP phone **18** has changed. In another embodiment, the IP phone **18** locally determines if the private IP address **46** or unique identifier **48** has changed by running local processes. The optional local processes will be described in more detail below.

[0030] If any of the information including the public NAT IP address **38**, the NAT port number **42**, the private IP address **46** of the IP phone **18**, or the unique identifier **48** of the IP phone **18** changes, the IP phone **18** sends a packet similar to **30**D to the NMS **51** updating the new information in entry **53** of table **52**.

[0031] Referring now to **FIG. 3A**, the NMS **51** can also manage devices in a private network containing nested NATs. Computer **19**A and computer **19**B reside in different stub networks **17**A and **17**B, respectively. In this example, NAT **23**A has currently assigned the IP address 192.168.0.1 to computer **19**A and NAT **23**B has currently assigned the same IP address 192.168.0.1 to computer **19**B.

[0032] Computers **19**A and **19**B communicate with the STUN server **22** to determine the public NAT IP address **38** and their assigned port numbers **80** and **81**, respectively assigned by NAT **20**. For example, computer **19**B sends an IP packet **60**A to NAT **23**B. The source address **61**A is replaced with the IP address **61**B and port number **91** for NAT **23**B before the reformatted packet **60**B is forwarded to NAT **20**. The NAT **20** replaces the source address **61**B and the port number **91** with the public NAT IP address **38** and port number **81** before forwarding the reformatted packet **60**C to the STUN server **22**.

[0033] The STUN server **22** inserts the public NAT IP address **38** and port number **81** into the payload **44** of return packet **60**D and sends the return packet **60**D back to NAT **20** at port number **81**. Because the return packet **60**D is received at port number **81**, NAT **20** forwards the return packet **60**D to NAT **23**B at port number **91**. Because the return packet **60**D is received at port number **91**, NAT **23**B forwards the return packet **60**D to computer **19**B.

[0034] Once the contact information including the NAT address **38** and the port number **81** are received in payload **44**, the computer **19**B forwards the contact information to the NMS **51** in packet **60**E. In one embodiment, the payload **50** of packet **60**E includes the IP address **38** and the port number **81** for NAT **20**, the private IP address **61** A for the computer **19**B, and a unique identifier **64** for the computer **19**B. Optionally the NAT address **38** and port number **81** may be provided to the NMS **51** via the source field **37** instead of in the payload **50**. In one embodiment the unique identifier **48** for the computer **19**B is a MAC address.

[0035] Once the packet **60**E is received, the NMS **51** adds an entry **54** to table **52** for computer **19**B. Table **52** is shown

in more detail in **FIG. 3B**. The information in entry **54** allows NMS **51** to manage computer **19**B. Computer **19**A provides information to the NMS **51** in the same manner as computer **19**B so that the NMS **51** can also manage computer **19**A.

[**0036**] The NAT address **38** for computers **19**A and **19**B is the same since both computers are located behind NAT **20**. In this example, the private IP address values **61** A for computers **19**A and **19**B also happen to be the same. However, the port numbers **80** and **81** for computers **19**A and **19**B, respectively, are different. This provides unique contact information to NMS **51** for computers **19**A and **19**B.

[**0037**] Even though the NMS **51** has different contact information for computers **19**A and **19**B, the unique identifiers **48** allow the NMS **51** to reliably distinguish between each computer operating in the private network. This is because the NAT **20** may from time to time reassign port numbers for computers **19**A and **19**B. For example, NAT **20** may reassign port numbers when rebooted. Even if NAT **20** swaps the port numbers associated with computers **19**A and **19**B, NMS **51** can still identify computers **19**A and **19**B via their associated MAC address as shown by unique identifiers **48** in **FIG. 3B**.

[**0038**] **FIG. 4** shows a device **400** operating in a private network that provides contact information to the NMS **51**. The device **400** includes a processor **401** and a memory **402**. The memory **402** includes instructions that, when executed by the processor **401**, perform the functions described in the flowcharts of **FIGS. 5 and 6**.

[**0039**] Referring to **FIG. 5**, the device **400** in block **501** does not know its apparent address. An apparent address is either the actual public IP address of the device **400**, or if the device is behind a NAT, the public NAT IP address and port number. The device **400** generates and sends a first packet to the STUN server in block **502**. In block **503**, the device **400** receives a response packet back from the STUN server that includes the apparent address of the device. The apparent address will either be the actual address of the device **400** or the IP public address and port number of an intervening NAT. In a nested NAT configuration when there is more than one intervening NAT, the apparent address will be the public IP address and port of the NAT connected the public IP network.

[**0040**] After receiving a response back from the STUN server, the device **400** in block **504** sends a second packet including the apparent address, the actual address of the device **400**, and a unique identifier for the device **400** to the NMS. The device **400** in block **505** then receives management communications from the NMS. In one embodiment, the management communications are made according to Simple Network Management Protocol (SNMP). The device **400** also detects changes in the apparent address, the actual address, and the unique identifier as shown in the flowchart in **FIG. 6**.

[**0041**] Referring to **FIG. 6**, a NAT frequently changes port assignments and sometimes the NAT public IP address. The device **400** in block **600** accordingly, either periodically, or based on detected events, sends communications to the STUN server to monitor any changes in the apparent address. Whenever an event such as a device or NAT reset is detected, the device **400** sends another message to the STUN server. The device **400** also may send pre-scheduled communications to the STUN server at predetermined rates. The frequency with which the device sends scheduled communications to the STUN server can be determined according to RFC **3489**.

[**0042**] In one embodiment, the device in block **600** sends scheduled communications to the STUN server at a predetermined rate T_1. The device can also run two scheduled local processes at predetermined rates to determine whether the actual address or the unique identifier has changed. Since the unique device identifier is relatively static in this embodiment, the device runs a first scheduled local process at a slow rate of T_2 in block **601** to determine whether the unique device identifier has changed. The device runs a second scheduled local process at a faster rate of T_3 in block **602** to determine whether the actual address assigned to the device by NAT has changed. Port number reassignment by the NAT often occurs in unison with actual address reassignment by the NAT. Thus, when the second scheduled process **602** determines that the actual address assigned to the device by the NAT has changed, the NAT port number has probably also changed. Therefore, if the second scheduled local process **602** detects that the actual address has changed, the device makes an unscheduled communication with the STUN server in block **603**.

[**0043**] Because the unscheduled communications **603** detect port number reassignment, the frequency of scheduled communications **600** to the STUN server may be reduced. If any scheduled communication **600** or unscheduled communication **603** with the STUN server detect change to the apparent address, the actual address, or the unique identifier, the device updates the NMS in block **604**.

[**0044**] Referring now to **FIGS. 7 and 8**, an NMS **700** that manages devices either directly or across NAT boundaries is shown. The NMS **700** includes a processor **701**, a memory **702**, and table **703**. The memory **702** includes instructions that, when executed by a processor, perform functions described in the flowchart of **FIG. 8**.

[**0045**] Referring to **FIG. 8**, in block **800** the NMS **700** waits to receive information about a device that needs to be managed. When the information is received, the NMS **700** adds a table entry for the device in block **801**. The table entry includes an apparent address of the device, an actual address of the device and a unique identifier. In block **802**, the NMS **700** searches the table for an entry associated with a device that needs to be managed. In block **803**, the NMS **700** attempts to communicate with the device based on the contact information in the associated table entry. If the communication is successful, the NMS **700** manages the device in block **804**S.

[**0046**] If communication with the device fails, the NMS **700** in block **804**F waits for an amount of time N for updated contact information from the device. If an update is received within time N in block **805**S, the NMS **700** re-attempts to communicate with the device in block **803**. Optionally, if the NMS **700** fails to receive an update within time N, the NMS **700** deletes the entry for that device in block **805**F.

[**0047**] The system described above can use dedicated processor systems, micro controllers, programmable logic devices, or microprocessors that perform some or all of the operations. Some of the operations described above may be implemented in software and other operations may be implemented in hardware.

[0048] For the sake of convenience, the operations are described as various interconnected functional blocks or distinct software modules. This is not necessary, however, and there may be cases where these functional blocks or modules are equivalently aggregated into a single logic device, program or operation with unclear boundaries. In any event, the functional blocks and software modules or features of the flexible interface can be implemented by themselves, or in combination with other operations in either hardware or software.

[0049] Having described and illustrated the principles of the invention in a preferred embodiment thereof, it should be apparent that the invention may be modified in arrangement and detail without departing from such principles. I claim all modifications and variation coming within the spirit and scope of the following claims.

What is claimed is:

1. A Network Management System (NMS) managing devices across a Network Address Translator (NAT) boundary, comprising:

a processor receiving a public Internet Protocol (IP) address and a port number for a NAT located between the NMS and one of the devices and managing the device across the NAT boundary according to the received public IP address and the port number for the NAT.

2. The NMS according to claim 1 wherein the processor stores the public IP address and port number for the NAT in a table entry associated with the device.

3. The NMS according to claim 1 wherein the processor receives a unique identifier for the device and uses the unique identifier to communicate with the device over the NAT boundary.

4. The NMS according to claim 3, wherein the processor receives private IP addresses and unique identifiers for multiple devices operating behind the NAT boundary and further receives public IP addresses and port numbers for the NATs connected to the multiple devices, the processor using the IP addresses, port numbers and unique identifiers to communicate with the multiple devices.

5. The NMS according to claim 4 wherein some of the multiple devices include a same private IP address or NAT port number in the table and the processor uses the unique identifier to access the devices.

6. A network device comprising:

a processor sending a first communication to a Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs) (STUN) server and receiving back a response including a port number and a public IP address for a NAT located between the processor and the STUN server, the processor sending a second communication containing the port number and the public IP address for the NAT to a Network Management System (NMS) that enables the NMS to send receipt of management communications to the processor.

7. The network device of claim 6 wherein the processor sends additional communications to the STUN server that identify any changes in the port number or the public IP address of the NAT and then sends any identified changes to the NMS.

8. The network device of claim 6 wherein the processor also sends a private device IP address or a unique device identifier in the second communication.

9. The network device of claim 8 wherein the processor determines locally whether the private device IP address or the unique device identifier have changed and sends any identified changes to the NMS.

10. A method for providing contact information to a Network Management System (NMS), comprising:

sending a message to a Simple Traversal of User Datagram Protocol Through Network Address Translators (STUN) server;

receiving a response back from the STUN server including a public Network Address Translator (NAT) IP address and NAT port number; and

sending the NAT IP address and NAT port number to the NMS.

11. The method of claim 10 including sending a unique identifier address to the NMS that allows the NMS to uniquely access devices in a nested NAT network.

12. The method of claim 10 including:

sending additional communications to the STUN server;

receiving responses back from the STUN server;

examining the responses from the STUN server to identify any changes to the public NAT IP address or the NAT port number; and

sending any identified changes to the NMS.

13. A method for managing a device in a private network comprising:

receiving a communication from the device including a Network Address Translator (NAT) port number and public IP address;

storing the NAT port number and public IP address in a table entry associated with the device in the private network; and

sending management communications to the device in the private network using the NAT public IP address and NAT public port number.

14. The method of claim 13 including receiving a unique device identifier and using the unique device identifier along with the NAT port number and IP address to communicate with the device.

15. The method of claim 13 including receiving periodic updates to the public NAT IP address, NAT port number, or unique device identifier and using the updates to then communicate with the device.

* * * * *