

Overview of Network Address Translation NAT in Windows XP

(Microsoft Corporation)

Introduction

As more homes and small businesses add computers they are finding that networking is an extremely powerful tool for sharing computer resources. An Internet connection is one of the more precious resources on the network and is likely to be shared. To do this and to enjoy an inexpensive, easy to manage, home or small office network, Internet gateways are being deployed. Internet gateways often provide network address translation (NAT) to connect multiple hosts to the Internet and share a single public IP address. Unfortunately, this solution breaks many types of networked applications—as will be described in this paper.

NAT Traversal technology has been created to allow network applications to detect the presence of a local NAT. Once detected, the application can then configure the NAT, defining the appropriate mappings so that the NAT will forward their traffic.

This paper is an overview to introduce consumers and developers of network applications to NAT, identify common NAT problems, and review how NAT Traversal can be used by applications to address these problems. Technical details of the NAT Traversal APIs are provided in the Windows Platform SDK. Developers are encouraged to review these resources for more detailed explanations of how to capitalize on these new operating system capabilities that also extend to third-party gateway devices.

NAT Traversal relies on the NAT supporting the UPnP technology. An important feature to look for in an Internet gateway device (IGD) is UPnP certification. Consumers purchasing or leasing an IGD from their Internet service provider (ISP) are strongly encouraged to consider only those devices that are UPnP certified for NAT traversal because this feature makes such an important difference with respect to customer satisfaction, lower support costs, and the use of more innovative services and applications.

Adding UPnP technology support for NAT traversal to an IGD is not a complex, expensive or time-consuming endeavor for the IGD vendor. By using UPnP technology, which is based on Internet standards and protocols, the IGD vendor can solve the problem of NAT traversal and have those benefits extend to most any application that traverses their device. This is in sharp contrast to other solutions that many application developers or gateway device vendors have to provide today to solve these problems. This paper is not a detailed guide for hardware vendors desiring to implement NAT Traversal in IGDs. For this information, please see the UPnP Forum Web site.

Knowledge of Windows architecture, networking and the UPnP architecture will be helpful, but not required, to fully understand this paper.

What is NAT?

Network address translation (NAT) is an Internet Engineering Task Force (IETF) standard used to allow multiple computers on a private network (using private address ranges such as 10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16) to share a single, globally routable IPv4 address. NATs are often deployed because public IPv4 addresses are becoming scarce. Internet Connection Sharing

Overview of Network Address Translation NAT in Windows XP

(Microsoft Corporation)

in Windows XP and Windows Me, along with many IGDs use NAT, particularly to connect to broadband networks through DSL or cable modems.

NAT is an immediate but temporary solution to the IPv4 address exhaustion problem that will eventually be rendered unnecessary with IPv6 deployment. IPv4 address exhaustion is a particular problem in Asia and other geographies around the world and will increasingly become an issue in North America.

In addition to reducing the number of public IPv4 addresses needed for worldwide Internet connectivity, NAT also provides a simple packet filtering function by forwarding only solicited traffic to private network hosts. Solicited traffic is traffic that was requested by a private network host. For example, when a private host computer accesses a Web page, the private host computer requests the page contents from the Web server. The traffic for the Web page contents is solicited traffic. By default, a NAT does not forward unsolicited traffic to private network hosts.

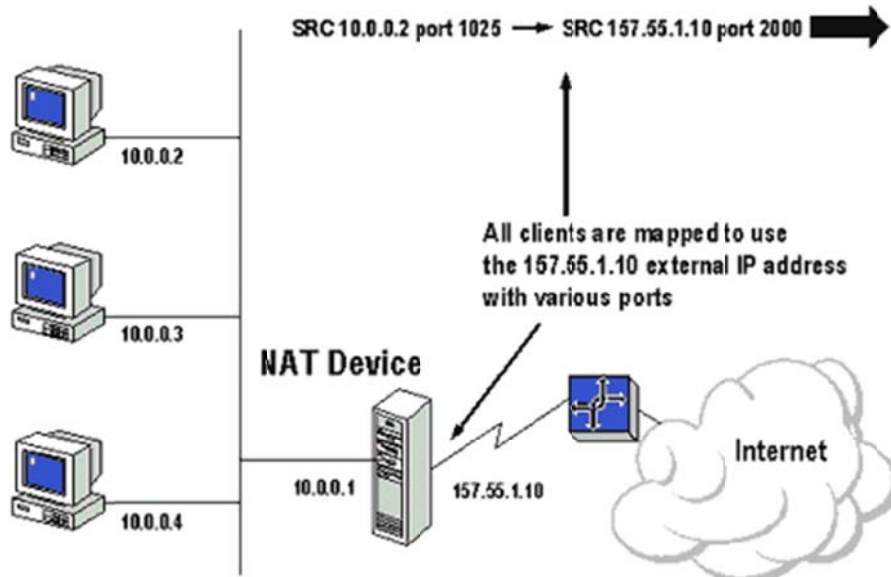


Figure 1: Example network using a NAT to communicate with the Internet

General NAT Operation

Clients behind a NAT are assigned private IP addresses, usually through the Dynamic Host Configuration Protocol (DHCP) or static configuration by an administrator. When communication outside of this private network takes place, the following things normally occur.

On the Client

When an application wants to talk to a server it will open a socket associated with a source IP address, source port, destination IP address, destination port and network protocol. This identifies both endpoints for the communication to take place. When the application transmits information using the socket, the client's private IP address (source IP address) and port (source port) are inserted into the source fields of the packet. The destination fields of the packet will contain the

Overview of Network Address Translation NAT in Windows XP

(Microsoft Corporation)

server's IP address (remote host – destination IP address) and port. Because this packet is destined for a location off of the private network, the client will forward this packet to the default gateway. The default gateway is the NAT.

Outgoing Packet at the NAT

The NAT will intercept this outgoing packet and create a port mapping using the destination IP address (server), destination port, external IP address of the NAT, external port, network protocol, and the internal IP address and port from the client.

The NAT will maintain a table of these mappings, storing this port mapping in the table. The external IP address and port are the public IP address and port to be used by for this data traffic in place of the internal client's IP address and port.

The NAT then "translates" the packet by swapping the source fields of the packet from the private, internal IP address and port of the client to the public, external IP address and port of the NAT.

The packet is then sent on the external network (the Internet) to eventually reach the intended server.

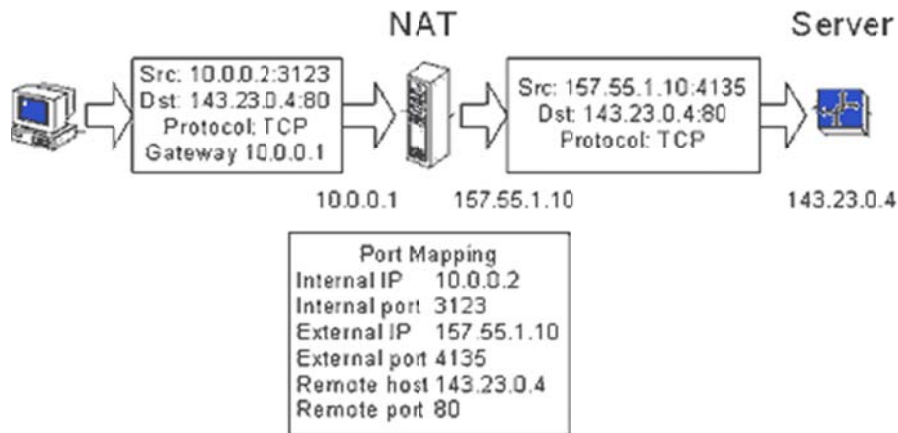


Figure 2: Example of an outgoing packet translation

At the Server

When the server receives the packet, it creates a socket with what appears to be a computer with a globally routable, public IP address. It will address response packets to the external IP address and port of the NAT, using its own IP address and port in the source fields.

Incoming Packet at the NAT

The NAT receives these packets from the server and compares them to its table of port mappings. If the NAT finds a port mapping where the source IP address, source port, destination port, and network protocol of the incoming packet match the remote host IP address, remote port, external port, and network protocol of the port mapping, the NAT will perform a reverse translation. The NAT replaces the external IP address and external port in the destination fields of the packet with

Overview of Network Address Translation NAT in Windows XP

(Microsoft Corporation)

the client's private IP address and internal port. This is an example of solicited incoming traffic. The NAT silently discards unsolicited incoming traffic that does not match a port mapping. The NAT then sends the packet on the internal network to the client.

The effect of NAT is the client will be able to communicate on the global Internet with a private IP address, without any extra effort on the part of the client application. This means the application will not have to call additional APIs and the client will not have to perform additional configuration. In this case, the NAT is transparent to both the client and the server application - everything just works. However, not all network applications use protocols that work with NAT.

Common Issues with NAT and Applications

Having clients use NAT to share a single public IP address works when the client initiates the contact and receives a reply on the same port. Some applications, however, do not work properly when a NAT is used to connect to the Internet.

Services on the Internal Network

Many network services or servers assume that if they establish a listening socket, any client on the Internet can initiate contact with them. If there is a NAT on the edge of the network, NAT requires that a port-mapping exist in order to forward incoming traffic to services on an internal network. Because of this, the service only works for clients on the private network - it is unavailable to the rest of the Internet.

The most common work around for this issue is to manually configure a port mapping that will cause the NAT to forward traffic addressed to a specific external IP address and port of the NAT to the internal IP address and port used by the service.

With this port mapping in place, services can receive incoming packets – making the service accessible to clients external to the private network. Until the port mapping is made, the service is inaccessible from the Internet.

Manually configuring this mapping is usually complicated and requires a more experienced user in order to configure the mapping correctly. As a result, many consumers or small business users are not able to use the applications or services they desire unless they contact customer support of their broadband ISP, computer manufacturer, retailer, or Internet gateway vendor trying to sort out the source of and solution to the problem. This also results in a less restrictive mapping – any external client can use this mapping to initiate contact with the server.

Embedded Addresses or Ports

Some network applications assume the IP address and port the client has been assigned will always be globally routable and can be used on the Internet directly. In many cases they are private IP addresses from IETF reserved address ranges. The application will include this private

Overview of Network Address Translation NAT in Windows XP

(Microsoft Corporation)

IP address or port in the payload of packets sent to the server. The server may use this embedded address as the address to contact the client.

If the server attempts to reply using the embedded IP private address and port instead of the mapped address and port supplied by the NAT, the packet is dropped. This occurs because the embedded IP address is non-routable. If the network application could discover the presence of a NAT, and retrieve the external IP address and external port mapping to be used, the application could embed the right information in the packet.

Applications Using Disparate Sockets

Other network applications send traffic to a server or peer using a socket on one port "X" and expect to receive traffic from the server to a separate listening socket on port "Y". The NAT sees the outgoing traffic and creates a port mapping for port "X", but does not know how to make a port mapping for the return packets addressed to port "Y". Incoming packets addressed to port "Y" are dropped.

Expecting Ports to be Available

Some network protocols assume that a globally routable, well-known port will always be available to them. When multiple clients share an IP address, only one client can use the well-known port at one time. For example only one web service can use the external port 80 on a local network at a time. If this were not the case, the NAT would be unable to determine which client the external request applied to. Even with the aid of a user configuring port mappings, special measures must be taken if multiple clients are to be discovered from outside the local network.

Multiple NATs

If a client is behind a NAT which is behind another NAT, new problems beyond the scope of this paper appear.

Impact on Customers and Industry

The previous paragraphs describe the technical phenomena associated with NAT traversal. The impact of this from a user perspective is simple: people cannot use the services or applications they want to use when NAT interferes.

Most users today do not even realize they've been subject to NAT issues. All they know is that when they try to enjoy multi-player gaming or engage in peer-to-peer applications, such as real time communications, or use some other application, they cannot. They may see some sort of "cannot connect" error message on their computer or perhaps their application will attempt to work and then just fail.

In some cases, a user with a dial-up modem connection to the Internet will have no issues with these experiences while using the dial up modem. Then, when the user signs up for broadband service and begins using a DSL or cable modem device with NAT, the problems occur. Expecting

Overview of Network Address Translation NAT in Windows XP

(Microsoft Corporation)

to enjoy a faster Internet experience, these users, in particular, can be baffled by the NAT issue that suddenly inhibits their ability to play games or enjoy other services.

This causes customer dissatisfaction, which can be directed at the computer vendor, the ISP, the Internet gateway vendor, or others. Often, the customer does not know what the source of the problem is and technical support staffs do not always know how to troubleshoot these problems over the phone.

This is not just an issue for the user. It also is an issue for the vendors that provide products and services to the user. The support calls the customer makes to try to resolve these NAT-induced problems cost money and can reduce or eliminate a vendor's or retailer's profitability. These issues can cause some users to be less interested in new services or applications due to lack of satisfaction with previous services the user has attempted, so NAT can be an inhibitor to more innovative product/service offerings and adoption. Given these factors, solving this NAT issue is an important task for the industry.

What is NAT Traversal?

NAT Traversal is a set of capabilities that allows network-aware applications to discover they are behind a NAT, learn the external IP address, and configure port mappings to forward packets from the external port of the NAT to the internal port used by the application – all in an automated fashion so the user does not have to manually configure port mappings or other such mechanisms.

This is a more holistic solution to the connectivity issues caused by NAT than other application-specific methods that have been employed to-date. Such specialized solutions to-date required either technical knowledge on the part of the user, special development arrangements of development efforts on the part of application developers or Internet gateway vendors, or all the above.

Although NAT traversal addresses some of the problems with NAT, it is not a panacea, and does not solve everything. Still, NAT traversal in this automatic fashion represents a significant step forward with regard to improving customer satisfaction, reducing customer support calls and enabling new, innovative services and applications, particularly in a home network situation.

NAT traversal should be thought of as a coping mechanism that should be used when needed, but will not work in all situations. NAT and therefore NAT traversal will no longer be needed in an IPv6 world where every client has a globally routable IP address. Forecasts vary with regard to how quickly IPv6 will enjoy pervasive deployment. The industry, including Microsoft, is making significant investments to move forward with IPv6, but the NAT traversal solution described in the remainder of this document can make a real difference now and for the next few years for consumers and small business users who want to overcome NAT issues.

Overview of Network Address Translation NAT in Windows XP

(Microsoft Corporation)

NAT Traversal Operation

NAT Traversal relies on discovery and control protocols that are part of the UPnP architecture specifications. The UPnP Forum has a working committee focused on defining the control protocol for IGDs and defining the services for these devices.

IGDs that support the required elements of the IGD control protocol will advertise their presence and publish XML description documents to control points on their local network. From these XML description documents, it is possible for control points to learn what actions to call to determine if an Internet Gateway has a NAT enabled, get the external IP address of the NAT, and create port mappings.

The NAT Traversal API in Windows abstracts the need to use UPnP technology directly, providing interfaces to detect, manage and configure the NAT.

The NAT Traversal API

When a network application needs to detect the presence of a NAT and adjust behavior of that device, the application can use the NAT Traversal API offered in Windows (fully documented in the Platform SDK) to provide the following functionality:

- Determine if a NAT is present
- Get the external IP address of the NAT.
- Get the static port mapping information for a specific external port, if it is mapped.
- Add a static port mapping, unless the external port is previously assigned.
- Enable or disable a specific port mapping without deleting it
- Edit the user-friendly description of a static port mapping
- Delete a static port mapping.
- Obtain a list of static port mappings for the local network.

With this functionality, applications can work around many of the problems created by the presence of NAT. Note that Windows NAT traversal APIs support port mappings only of infinite duration, otherwise known as static port mapping, at this time.

NAT Traversal APIs in Windows XP

NAT Traversal APIs are installed by default on Windows XP. These APIs also can be installed on machines running Windows Me and Windows 98 by using a tool on the Windows XP CD called the "Network Setup Wizard". The NAT Traversal APIs also require users to install Internet Explorer version 6.0 for the additional XML parser support provided. NAT Traversal is not currently supported on Windows 2000.

Supporting NAT Traversal in Internet Gateways

Internet gateways support NAT traversal by supporting the IGD (IGD) specification defined by the Internet Gateway Working Committee of the Universal Plug and Play Forum. Gateways vendors

Overview of Network Address Translation NAT in Windows XP

(Microsoft Corporation)

also should be aware that NAT traversal APIs in Windows make the following assumptions about IGDs.

- IGDs only advertise one external interface at a time. Though it is technically acceptable for IGDs to advertise multiple external interfaces, the NAT Traversal APIs will only use the first one.
- IGDs support port mappings that allow any remote IP address to send packets to internal clients.
- IGDs support port mappings with the broadcast address listed as the client
- IGDs support different numbers for the external port of the NAT and internal port of the client.
- IGDs will advertise with a version number of 1.

Static port mappings (or port mappings with a duration set to infinity) will persist indefinitely, surviving reboots, IP address changes, and the presence of the client on the server.

As more manufacturers of IGDs understand the benefits of using UPnP technology to address this issue and as more consumer and small business users become aware of the issues associated with NAT and the viability of these UPnP certified NAT traversal solutions, there is an expectation that UPnP certification for NAT traversal will become a checkbox item or market requirement for devices in this category.

Internet gateway vendors should become members of the UPnP Forum to learn how to make their IGD compliant with UPnP technology.

It should be noted that Internet Connection Sharing on Windows XP supports version 0.9 of the UPnP IGD standard. It is anticipated that version 1.0 will be compatible with version 0.9.

How Applications Make Use of NAT Traversal

How an application uses NAT Traversal will depend upon several factors, including how long-lived a port mapping needs to be and whether the port is used by multiple clients or services. It is very important that applications clean up any static port mappings they create to avoid orphaned mappings and depletion of ports for use by other applications.

If an application is a network service, like a Web server, and requires the use of a well known port for the duration of its life time, its installation program can use the NAT Traversal APIs to configure a static port mapping. Assuming that other applications, network administrators, the network topology remains constant and clean-up mechanisms leave the mapping alone, external clients will be able to contact the service for the life of the service. The application's uninstall service is responsible for deleting this mapping. In the event of a crash, the static port mappings will persist

Overview of Network Address Translation NAT in Windows XP

(Microsoft Corporation)

in the absence of the service. If the external IP address changes, the static port mapping will automatically pick up the change.

If the application is not always going to be running, or is less trusting of the network to maintain its static port mappings, it might reserve a particular well known port every time it launches and return the resource every time it shuts down. This can be done by running a script in parallel. An alternative to adding and deleting the port mapping is for the application to enable and disable the mapping as appropriate. The application can also leave the static port mapping up all the time and simply refresh the mapping whenever the application launches.

Again, if the external IP address changes the static port mapping automatically picks up the change.

If multiple applications on different clients on the private network use the same internal port number, the applications will require modification to support multiple clients running. Only a single client can use this internal port number for an external port mapping. The recommended behavior here is first client wins. The other clients should request asymmetric port mappings where the internal port number is different than the external port.

There is a special case where multiple clients can listen on the same external port for the sole purpose of being discovered by remote hosts. Incoming packets can be translated to use a broadcast address for the internal client IP address, instead of a particular clients address. Clients that are listening on that port will be able to reply by initiating their own connection to the remote host. This in is not recommended for general use, because incoming packets to this address will be received by and affect every client on the network.

If a service needs to listen to a random port for a short time, it should request a static port mapping from within the application and not with a script. It should clean up after itself as soon as it is done (delete the mapping). The application should keep a record of its outstanding port mappings. This way if the application were to crash without closing the mappings, it will be able to retrieve the information necessary to clean up the port mappings the next time it is launched.

If an application should leave the network without cleaning up its port mappings, the mappings will remain and cleanup responsibility will fall on the user. There is currently no clean up mechanism in Windows, as it is difficult to tell when an application is done using a mapping.

Limitations of NAT Traversal

While NAT Traversal solves several problems associated with connecting through NATs, several issues remain or exist as a result of NAT Traversal. These issues include:

NAT traversal has an open trust model. This means that all application on the private network have access to all the port mappings on a NAT. This allows for a great amount of flexibility of multiple points of administration, but applications do not have exclusive ownership of their mappings.

Overview of Network Address Translation NAT in Windows XP (Microsoft Corporation)

Conflict resolution is the responsibility of applications. If an application tries to map a port that is already mapped to another client, it is up to the application to either find another port or overwrite the application.

NAT traversal does not solve the problem of an ISP distributing private addresses and using NAT to let clients connect. In this case the NAT is outside of the IGD, actually sitting within the ISP's network. NAT traversal within the home or small business will fail if the NAT on the client's network is behind another such NAT. As a result, ISPs are encouraged not to deploy NAT within their networks.

Applications don't get NAT traversal for free; they will have to be modified to call APIs or ship with scripts to make the solution happen. This is a manageable development effort for most developers, particularly considering that once these NAT traversal mechanisms have been incorporated into an application, the application can work automatically with a variety of IGDs.

Applications are responsible for cleaning up after themselves when they are done with a port mapping. Static mappings persist indefinitely and are most appropriately used by services that intend to listen on well-known ports for the life of the application.

The Internet gateway providing the NAT must support Universal Plug and Play IGD Spec version of at least .9.

Conclusion

NAT is an IETF-approved solution to the problem of IPv4 address space exhaustion. Internet gateways that use NAT are often used in homes and small offices. They are used because they are cheap, easy to manage, and don't require users to install special software.

The downside to using NAT is that many chat, multiplayer games and peer-to-peer applications break. This is because their network protocols make assumptions about the network architecture that are no longer true.

NAT Traversal provides a way for applications to discover the presence of the NAT, discover the shared, globally routable IP address and configure static port mappings to solve some of the connectivity problems. The NAT traversal solution does not solve all of the problems associated with NAT, but alleviates some of the problems.

Key items of this article are the following:

- IGD vendors should implement support for UPnP technology in their devices to support NAT Traversal.

Overview of Network Address Translation NAT in Windows XP

(Microsoft Corporation)

- Network application developers should use the Windows NAT Traversal APIs to detect the presence of NAT and enable their applications to traverse the NAT when necessary.
- Consumers should use IGDs that support UPnP technology and NAT Traversal to ensure the best application behavior.
- DSL and cable modem ISPs should specify, sell, and lease IGDs that support UPnP technology for NAT traversal.