Computer Networks

Field of the Invention

The present invention relates to computer networks which use the IP protocol, and which include data link layer addresses, such as networks incorporating Ethernet and WaveLan technologies, as well as token ring and FDDI technologies.

Background to the Invention

All nodes on the Internet are identified by IP addresses. Unfortunately, due to the limited public IP address pool that is available, and to administrative and maintenance requirements, it is often not possible to allocate a static IP address to a user. Instead, a user may for example receive a different IP address for each Internet access session.

This can cause problems in identifying an end user on an intranet or on the Internet.

Various solutions to this problem have been provided, such as DDNS (Dynamic Domain Name System) which uses dynamic mapping of hostnames to IP addresses, X503 client certificates which have user information embedded within them, and even simple user login authentications.

These solutions, however, have their own limitations, such as additional client software support requirements and a dependency on end users to initiate communication.

An aim of the present invention is to provide a network system which facilitates the identification of an end user. Although the present invention may provide an alternative to the above-mentioned solutions, it need not be mutually exclusive with them.

The present invention is also particularly, though not exclusively, useful in respect of computer systems as described in International Patent Application PCT/SG00/00170, filed on 17 August 2000, and entitled "Reconfigurable Computer Networks", the contents of which are incorporated herein by reference.

Summary of Invention

According to a first aspect, the present invention provides a computer network system including a plurality of client nodes, each of which has a unique data layer link address, wherein the system includes at least one network device which is able to access the data link layer address of a client node, and wherein the network device uses the accessed data link layer address to uniquely identify the client node and to provide network layer or above (e.g. application layer) services to the node based on said data link layer address.

In accordance with the present invention, a router (as the network device) for example may monitor the data link layer information, e.g. as contained in a data packet transmitted by a client node, in order to extend the functionality of standard network and application services, and can use the data link layer address of a client node to uniquely identify a host and its associated end user.

The present invention contrasts with the traditional approach of using an IP address to provide such services and to map to a client node.

The data link layer address may typically comprise a medium access control (MAC) address. Every client node of a network includes at least one attached network interface card NIC (otherwise called a network card or network adapter card) in order to allow it to physically connect to the network. Each of these cards has a unique MAC address associated with it. These MAC addresses include a six-byte field. The first three bytes (the OUI – Organisational Unique Identifier) identify the vendor of the card and are globally assigned by the IEEE. The final three bytes are vendor assigned in a manner which ensures that each card has a unique address.

The rationale behind the present invention is that the hardware, that is the network cards, with their default hard-coded MAC addresses are less likely to change compared to other software controlled network settings, and so instead of relying only on IP addresses to identify a node in the network and application layers, the MAC address is used.

In a preferred embodiment, the system includes a link aware router which is able to determine the link layer address of a client node which has sent a data packet, and includes a database of link layer addresses and user information, which may be hosted by the router. The information may for example relate to the

department in which a user of the client node is registered and to their level of authority and security clearance.

Thus, when a request, such as a HTTP request, is received from a client node, the router is able to identify the client node from its MAC address and can implement one or more suitable network and application policies that a network administrator may define for the client nodes.

For example, in the case of a HTTP request, the router may provide a transparent proxy function to provide a different web page to two client nodes when they request the same URL. This may be dependent on for example the department in which the client nodes are registered. Alternatively, access to certain functions such as email, web pages, FTP, or the like may be denied and/or may be restricted to include or exclude certain domains.

In one preferred form, for example where the system is a reconfigurable computer network as disclosed in PCT/SG00/00170, the router may determine whether an end user's link layer address is included in the database, and may initiate a configuration procedure when said address is not in said database, e.g. by redirecting a HTTP request from the client node to a configuration page.

Further preferably, the router determines whether a client node's link layer address is included in the database, and initiates a security procedure when said address is not in the database, e.g. by redirecting a HTTP request to a web page warning that there is an unauthorised access attempt.

The system may also determine positional information of a client node and may record the information in the database. This information may for example be the number of the network card NIC (e.g. eth0 for Ethernet card number 0) of the router that data packets from the client node are received at. It may also comprise information regarding a client node's assignment to a virtual LAN, e.g. a VLAN ID. Thus, as well as MAC address determined services, the client node may be provided with location specific services, such as location specific web page content.

In a particularly preferred embodiment, the system includes a transparent DNS server, which may be hosted at the link aware router. When using a DNS server, the DNS server preferably assigns a unique name to a client node by mapping the client node's link layer address with the name. This name may be for example a fully qualified domain name (FQDN) or a hostname. It would also be possible to assign other attributes to the link layer address that might previously have

been associated with an IP address. This could include network resources, e.g. a printer and dynamic firewall rules (e.g. non-static IP).

By directly mapping a MAC address to a DNS A type resource record, it is possible to assign a FQDN or hostname to an end user computer without additional support from the client, and a node can always reach a peer correspondent node by its FQDN or hostname within a link aware DNS network, regardless of the peer's current IP address.

Further preferably, the system includes a Network Address Translator (NAT) for mapping a client node IP address to an assigned IP address. This then allows for transparent routing across address realms, in which for example the private IP addresses of a LAN are mapped to one or more public addresses, and possibly to one or more IP address and port combinations when using a Network Address and Port Translator (NAPT). For convenience, in the description herein the term NAT should be taken to cover NAPT.

Preferably, the Network Address Translator uses a client node's link layer address in the mapping of an IP address of said end node to an assigned node. This then allows client node's having the same IP address to be serviced by the NAT.

Where for example a configurable router as in PCT/SG00/00170 is provided the Network Address Translator may use physical location information for a client node in the mapping of an IP address of the client node. This may include the router NIC number at which packets from a client node are received as well as a VLAN ID.

Preferably, the Network Address Translator provides a reverse NAT operation in which an assigned IP address is mapped to a client node IP address and data link layer address.

Further preferably, the Network Address Translator provides a twice NAT operation in which mapping is provided both for a client node IP address to an assigned IP address and for an assigned IP address to a client node IP address and its data link layer address.

Viewed from a further aspect, the present invention provides a computer network system including a plurality of end systems, a NAT router, and a DNS server, wherein said router is able to determine the link layer address of the end systems, and wherein NAT and DNS procedures are based on said link layer addresses, each said link layer address being associated with a unique name.

From another aspect, the present invention provides a computer network system including a plurality of end systems and a NAT router, wherein said NAT router creates bindings between actual source end system IP addresses and apparent source end system IP addresses, and wherein said NAT router records the link layer addresses of said source end systems as a part of said binding.

Viewed from a still further aspect, the present invention provides a computer network system including a plurality of end systems, and a DNS server, wherein said DNS server maps link layer addresses of said end systems with unique names.

Viewed from another aspect, the present invention provides a computer network system including a plurality of end systems and one or more intermediate systems, and in which data is transferred between the end systems through the intermediate systems using network and link layer protocols, wherein the end systems are uniquely identified in the system by assigning a unique name to the link layer address of each end system, said unique name being used in the routing of data to the end systems by the intermediate systems.

Viewed from a further aspect, the present invention provides a computer network system including a plurality of client nodes and a proxy server, wherein the proxy server is configured to retrieve data link layer address information from data packets sent by said nodes and to service said client nodes in accordance with network level or above (e.g. application level) policy specifications based on said data link layer address information.

Viewed from a still further aspect, the present invention provides a computer network system including a plurality of network nodes having data link layer addresses and a router for routing traffic from and to the nodes, wherein the system includes a database of the data link layer addresses to attributes of the nodes, the router being configured to determine the data link layer addresses of the nodes and to perform look-ups of the database in order to service the nodes.

The present invention also extends to methods of administrating and servicing a computer system in accordance with the above features, and to software and hardware for implementing the same.

Brief Description of the Drawings

Various embodiments of the present invention will now be described, by way of example only, with reference to the accompanying drawings. It is to be understood that the particularity of the drawings does not supersede the generality of the preceding description of the invention.

Figure 1 is a schematic diagram of a possible network topology in accordance with an embodiment of the present invention;

Figures 2 and 3 are schematic diagrams illustrating outgoing and incoming traffic from and to a client node of a computer network to and from a link aware router with one or more intermediate systems according to an embodiment of the present invention;

Figure 4 is a schematic diagram of a link aware router hosting a link aware DNS server in accordance with an embodiment of the present invention;

Figures 5 to 7 are schematic diagrams illustrating how a router as shown in Fig. 5 may be used in the routing of traffic between two nodes of an intranet;

Figures 8 to 11 are schematic diagrams illustrating a router in accordance with an embodiment of the present invention connecting nodes across a routing realm boundary;

Figures 12 to 14 are schematic diagrams illustrating a router connecting nodes across a routing realm, wherein an address collision occurs between a receiving node and another node in the receiving node's intranet; and

Figures 15 to 17 are schematic diagrams illustrating communications between two nodes within an intranet which have the same IP address.

Detailed Description of the Invention

Fig. 1 illustrates one possible network topology for a computer network system incorporating a link aware router in accordance with one embodiment of the present invention, although of course many other topologies are possible.

In Fig. 1, a plurality of client nodes 1 is connected to a router 2, which in turn connects to an external network 3. The external network may take many different forms, and may for example comprise the Internet.

In this example, one of the client nodes, Node 1, connects to a first network card NIC I of the router 2 via a wireless base station 4 and a managed switch 5 having a VLAN tagged port, whilst a second client node, Node 2, connects to the same router card NIC I via a hub 6 and the managed switch 5. These nodes, Node 1 and Node 2, are in separate virtual LANs, VLAN A and VLAN B, corresponding to for example their departments within an organization running the network, and include identifiers in their outgoing data packets to this effect.

Another of the client nodes, Node 3, is connected directly to a second network card NIC II of the router 2.

The router 2 may provide proxy services to the private network containing the client nodes 1, for example by using various transparent proxies 7, and may funnel all user requests from the private network to the external network 3 and fan responses back out to the appropriate users. It may also provide other standard proxy services, such as web page and ftp caching, and application firewall protection.

The router may take the form of the reconfigurable router as described in International Patent Application No. PCT/SG00/00107 entitled "Reconfigurable Computer Networks". In this case, the client nodes 1 can be removed from and added to the network at the same or alternative logical and physical network locations at any time, with the router 2 updating its records appropriately at each connect/disconnect in order to allow seamless use of the client nodes 1 by the users of the network. Thus, the location of each client node can be determined from the incoming packet link information (e.g. the VLAN ID embedded in the packet data) and the network card of the router (e.g. NIC I or II) at which packets from the client nodes are received.

Each of the client nodes, Node 1, Node 2 and Node 3, has a unique MAC address, i.e. L1, L2 and L3.

This MAC address is a physical address (i.e. it is hardware specific), and is a unique serial number burned into the network adapter card of the client node when the network adapter is manufactured. The MAC address comprises a six-byte field of which the first three bytes (the OUI – Organisational Unique Identifier) identify the vendor of the card and are assigned by the IEEE, and the final three bytes are vendor assigned.

Generally, these addresses are used in the data link layer of a network, e.g. at the Ethernet or Token Ring MAC level.

In accordance with the present invention, however, these MAC addresses are used also to uniquely identify the client nodes at the network to application layers of the network, and the router 2 is configured so as to be aware of the MAC address of each of the nodes 1.

Thus, the router 2 includes a database 8 of MAC address to user information. This information may take any suitable form, and may relate to details of the user owning the client node, including their name, department and access privileges, as well as the client node current and permitted position information.

By providing such a database, and by enabling the router 2 to obtain the MAC addresses of the client nodes 1, the router 2 is able to provide intelligent service delivery by making available relevant specific network and application services to selective users based on their user profile obtained from the MAC address to user database and if desired also based on their logical and/or physical network location (e.g. through VLAN ID and incoming router NIC number).

Therefore, network and application policies can be enforced based on user attributes (e.g. user name, department and the like) derived from the MAC address, and network location derived from the incoming router NIC number and/or the VLAN ID. This may be in addition to any other standard criteria already supported in existing network applications (e.g. using IP addresses).

As an example, a company may maintain an intranet website, and all employees may be assigned a client node, e.g. a personal computer, with a dedicated network card having a MAC address. A database is then built of all the MAC addresses, together with the end users having these addresses and information relating to these end users. When employees from two different departments access the same URL from their assigned computers, they can be shown different web pages relevant to their departmental needs. This is possible even if they have the same IP address or connect to the same network point, because of the use of their unique MAC addresses to identify them.

Additionally, it is possible that for different physical network locations, the same employee, as identified by their MAC address, may be served different web pages for the same URL in cases where the page content is location dependent.

This example can be implemented using a transparent HTTP proxy with the router 2. Thus, every HTTP request may be intercepted by a firewall of the router 2 (e.g. by determining the destination port number 80,8080 or 3128 in the TCP header

of a packet, or by intercepting all outgoing TCP packets and analyzing the payload for HTTP requests, e.g. "GET http://*.*.* HTTP/1.0"), and will be redirected to the HTTP proxy (e.g. via destination IP address NAT, instead of traditional source IP address NAT). The proxy then retrieves the request's source MAC address, as well as VLAN ID and incoming router NIC information if applicable, and carries out a look-up of the related information based on the MAC address, and on the VLAN ID and incoming router NIC information if applicable. It then maps the original request to a different URL as necessary based on the information, and issues a proxy request to the new URL.

In the above, the firewall may do a destination IP address NAPT, e.g. a destination address and port of www.antlabs.com, 80, may be NAPT to a destination IP address of a local host and destination TCP port 3128 where the proxy resides. A TCP socket connection may then be established between the client and the transparent proxy, instead of with the web server www.antlabs.com. Each TCP socket will store the end nodes session information augmented to include the client MAC address and the incoming NIC logical index number (as each VLAN ID monitored by a NIC is assigned a new logical number, it is possible to derive the NIC and VLAN ID from the logical number). For each intercepted TCP packet read by the proxy, a socket option call may be made to obtain the client MAC address and NIC logical number of the selected socket.

In another example, where the router has the zero configuration capabilities as described in PCT/SG00/00170, the combination of both technologies can be used to configure a new network appliance remotely via a web-based GUI.

Thus, the router 2 can detect HTTP requests from an alien MAC address not listed in the database, and for example originating from a router NIC that is assigned for remote configuration, and can redirect the end user to a default start-up configuration page.

Thus, for each new user detected, a start-up page can be used to register the user's information, such as name, department, preferences (e.g. future start-up page), preferred printer, etc. This feature may also be used to configure a new network appliance such as a web server (a "black box" web-based administrated appliance rather than a traditional computer server), for which it is problematic to provide an initial IP address. With the present invention, there is no need to set an initial IP address, instead, the web server can simply be plugged in and its web

browser started. The start-up configuration page will then be presented, and from there web based administration can set an IP for WAN connection and other appliance configurations, etc.

Further, using a similar approach, the router 2 may be used as a security access control device. Thus, instead of typical packet based firewall controls based on IP addresses, which are susceptible to IP spoofing, any HTTP request from an unregistered client node network adapter card (i.e. from a card without its MAC address registered in the database) can be redirected to a web page warning, which informs the end user that they are attempting an unauthorised access.

In this manner, and unlike existing application firewalls, security controls can be provided all the way from the physical socket connection to the user (identified by their MAC address) for each individual application.

As well as providing the service delivery, the use of the MAC address allow the router 2 to make suitable return packet forwarding decisions by the direct use of the MAC address.

Figures 2 and 3 illustrate how outgoing and incoming traffic from and to a client node 1 could be configured in e.g. the system of Figure 1, although the figures here show the possibility of other routers also provided between the client node 1 and the link aware router 2.

In general, at least one of the client nodes 1 between two communicating nodes (in this example a node with MAC address L1 and IP address N1) should be one router hop from the link aware router 2. There can, however, be multiple intervening layer 2 switches (VLAN or otherwise) between the client node 1 and link aware router 2. There may also be co-operating routers 9 (e.g. transparent routers or a router in accordance with PCT/SG00/00170) between the client node 1 and the link aware router 2, that may for example provide the information necessary for the link aware router 2 to be able to inform a co-operating edge router 10 to generate and send the correct outgoing packet layer 2 link header back to the client node.

The intermediate routers 9 route packets to the link aware router 2 to allow it to determine policy decisions. If desired, they can make routing decisions for outgoing traffic, e.g. based on the destination IP address, it is possible for traffic to be sent to a different link aware router. They may also enable routing from one VLAN to another.

As can be seen in Figs. 1 and 2, the client node MAC, L1, may be embedded in a data packet between a link header (containing source and destination physical addresses) and a network header (containing source and destination IP addresses). Alternatively, in a switched based implementation, where the source MAC address does not change, the client MAC may form a part of the link header. These are only two examples, and other means of providing the client MAC address to the link aware router 2 and to the edge router 10 are possible, e.g. with out-of-band messages.

Routing operations between the link aware router 2 and the next hop router 11 on the e.g. Internet side of the router 2 will proceed as is known in the art, e.g. with the router 11 and the like determining the destination of the data packet from the IP address destination in the network header and routing the packet according to routing tables for the IP addresses.

The link aware router 2 may also provide NAT (Network Address Translation) or NAPT (Network Address and Port Translation), as is known in the art, in order to for example map private IP addresses to public IP addresses, and to prevent problems with address collisions. NAPT allows the mapping of port numbers also, and for the purposes of the present application, references to NAT should be taken to include reference to NAPT and other variants of NAT (e.g. twice NAPT used for port forwarding, and destination NAT used by transparent proxies) also.

In the present invention, a link aware router, such as the router 2 of Fig. 1, may provide a NAT capability which also incorporates the MAC address of a client node in the mapping.

Thus, the NAT function may operate in the same manner as standard NAT functions (including any variants, such as destination NAT and e.g. as noted in "IP Network Address Translator (NAT) Terminology and Considerations", RFC 2663, P. Srisuresh, M. Holdrege, August 1999) except for the following:

1) For every new incoming session that is NAT, the client MAC address, as well as if applicable the incoming router NIC number and any other optional location specific information (e.g. VLAN ID), is stored in the binding of the original and given IP addresses in addition to other normally required NAT information.

2) For any matching reverse session traffic that maps the client IP address (and additionally transport port number for NAPT) back, the router for the

traffic forwarding uses the previously stored link information to generate the packet and transmit the packet to the client, and bypasses the normal route processing.

For step 2 above, if the client is mobile and movement detection is supported (e.g. as in PCT/SG00/00170), the router NIC number to which the outgoing traffic is sent may not be the same as that previously recorded in step 1, and this information should be updated by the presence detection process. The MAC address will not however change.

The link aware NAT procedure allows two clients with the same IP address to use NAT even if they are in the same link broadcast domain since they can be discriminated from one another by their MAC addresses, and as the return routing does not need to for example issue an ARP request.

In order for communications between two nodes with the same IP address, there must be a separate unique end-to-end form of identification, e.g. a fully qualified domain name (FQDN).

Both examples of link aware NAT procedures and the provision of FQDNs are discussed below in relation to a further feature of the present invention which is link aware dynamic domain name servers.

As discussed, the MAC address of a network interface card attached to a client node 1 can be used to uniquely identify the client node 1 and its associated end user (rather than using an IP address to map to a client node as in the prior art).

Extending this concept to the DNS system, it is possible within a network serviced by a link aware dynamic DNS system to assign a fully qualified domain name (FQDN), e.g. john.antlabs.com, or a hostname, e.g. john, to an end user computer without additional support from the client.

By directly mapping a MAC address to a DNS A type resource record, a node can always reach a peer correspondent node by its FQDN or hostname within a link aware DNS network, regardless of the peer's current IP address and without the need for a client node to register its IP address.

In a link aware DNS system, each node is identified by its MAC address, and by monitoring the network traffic and implementing movement detection e.g. as described in PCT/SG00/00170, a link aware router can determine the location of each client node and their associated MAC and IP address.

As shown in Fig. 4, a link aware DNS system may consist of a link aware router 2 and a link aware DNS server, which may be hosted on the link aware router 2, and includes a pre-configured MAC address to FQDN table 12.

For an internal LAN with no address collision, the simplest mechanism would be for the link aware router to dynamically set-up host based routes and update the DNS with the actual IP address of each individual client it discovers.

When a client resolves a FQDN or hostname, the actual current IP address of the host is returned. Any traffic destined for the corresponding host will be forwarded by the established dynamic host based routes (which may change if the client node moves).

Fig. 5 illustrates such an internal LAN. When traffic is generated by a client node 1, a routing table 13 and a DDNS table 14 on the link aware router 2 are updated. For example, if a packet with a MAC address A and a source IP address 1.1.1.1 is received on the router NIC 1, a host-based route is set in the routing table 13. The router then does a lookup in the link aware DNS table 12 for the MAC address A, and this returns the FDQN "albert.a.com". The DDNS table 14 is then updated to reflect the IP address 1.1.1.1 for the FQDN "albert.a.com".

When a client node 1, as in Fig. 6, carries out a DNS query, e.g. for albert.a.com, the query is intercepted by the transparent link aware DNS server of the link aware router 2. If the FQDN is found in the DDNS table 14, then the current IP address of the FQDN, e.g. 1.1.1.1, is returned in the DNS reply. If it is not found, then recursive DNS resolution is employed, as per standard known DNS operations.

Once the FQDN is resolved, as shown in Fig. 7, traffic can be routed between the sending and receiving nodes 1 through the router 2 due to the pre-established host based routes of routing table 13 shown in Fig. 5.

In cases where address collision exists, or where communications occur across two or more routing realms, a form of NAT can be applied with the link aware DNS server.

For example, in the case of cross routing realm communications without network collisions, a permanent FDQN or hostname can be used to find a client node's IP address and to send traffic to that IP address. When the client correspondent node receives a datagram from a sending node, there is no need for it to do a similar name lookup on the sender's FDQN or hostname, as the destination IP address (i.e. the IP address of the sender) is already known by the correspondent

node from the datagram received. Therefore, in this case, name lookup need only be done in one direction.

If the IP address of the node that initiates the name lookup is unique, i.e. there is no address collision), then the link aware router need only support Domain Name Server Application Level Gateway (DNS_ALG), and if there is no address collision in the private network, then NAT with DNS_ALG will suffice. The link aware DNS server will still need to update the correspondent node IP address in the DDNS table 14 on client discovery.

Fig. 8 shows an example of a no collision communication across a routing realm between a node 15 of a public network 3 such as the Internet and a client node 1 of a private network having a registered domain name "a.com" and a link aware router 2 which is NAT capable, supports DNS_ALG and is the authoritative name server for the domain "a.com".

As shown in Fig. 8, the link aware router 2 has discovered the client node MAC address A and has updated the routing table 13 (for the incoming NIC of the router 2 on which the packet from client node 1 was received) and the DNNS table 14 for "albert.a.com".

As shown in Fig. 9, when the public network node 15, with IP address 10.0.0.3 tries to resolve "albert.a.com", the DNS query is sent to the link aware router 2 since it is the authoritative name server for the "a.com" domain. The DNS_ALG thus queries the private network link aware DNS server, i.e. the DDNS table 14, which returns the result 1.1.1.1. The DNS_ALG then sets up an address binding 16 to map the client node's IP address 1.1.1.1 to a public IP address which belongs to the link aware router 2 for NAT purposes, in this case the IP address 11.0.0.1, and transmits the DNS reply of 11.0.0.1 to the requesting node 15.

Once the DNS resolution is completed, communication between the two nodes 15 and 1 proceeds as standard, as documented in "DNS extensions to Network Address Translators (DNS_ALG)", RFC 2694, P. Srisuresh, G. Tsirtsis, P. Akkiraju, and A. Heffernan, September 1999.

Fig. 10 shows the resulting NAT operations for traffic from the node 15 with IP address 10.0.0.3 to the correspondent node 1 with IP address 1.1.1.1, whilst Fig. 11 shows the NAT operations for reply traffic from the correspondent node 1 to the public node 15.

Turning to the possibility of address collisions, one such collision is an address collision between a correspondent client node (the target of a name lookup by an external node) and another client node on the same private network. This may be addressed by having the link aware router support reverse link aware NAT. This requires the use of the client nodes' link layer information.

In this situation, as communication is established by a public node outside of the private network rather than by one of the private nodes 1 with the collision problem, the reverse link aware NAT link layer information may be established when the binding information is created by the DNS_ALG.

For communications across routing realms, e.g. from the Internet 3 to a private network, there is a need to assign a public IP address to the correspondent node 1 of the private network.

For all scenarios, a unique IP address (not necessarily public) that does not collide with any other possible correspondent nodes must be used.

Fig. 12 shows a link aware DNS table 12 and a DDNS table 14 for the private network a.com with two client nodes 1 having the same IP address 1.1.1.1.

As in the no-collision embodiment of Figures 8-11, and as shown in Fig. 13, when a public node 15 transmits a DNS request for "albert.a.com", the DNS_ALG interrogates the DDNS server to obtain the IP address 1.1.1.1 of the correspondent node having the domain name "albert.a.com", and then sets up a DNS_ALG binding 16 of a public IP address 11.0.0.1 with the private client node address 1.1.1.1.

Additionally, in order to enable reverse link aware NAT, the DNS_ALG informs the link aware DNS server of the public IP address 11.0.0.1 assigned to the correspondent node's FQDN. Using this information, the link aware DNS server is then able to build a reverse link aware NAT table 17 mapping the assigned IP address 11.0.0.1 to the appropriate client node characterised by its MAC address A. This can be produced as the MAC address to FQDN/hostname mapping (of the link aware DNS table 12 of Fig. 12) is unique (one-to-one), and so a reverse look up from the FQDN/hostname to retrieve the correspondent node's MAC address is possible.

As the router 2 monitors the link level and incoming NIC information for each client node 1, in order to update the link aware DNS system, the link aware router 2 can establish the IP address and the network location of the correspondent node 1. With the correspondent node's actual and dynamically assigned (by DNS_ALG) IP

addresses, its MAC address and incoming NIC interface information (together with VLAN ID information if any), reverse link aware NAT can be carried out.

Fig. 14 illustrates the NAT operations for traffic from the node IP address 10.0.0.3 to the correspondent node address 1.1.1.1 with the reverse link aware NAT. Reply traffic from the correspondent node 1.1.1.1, MAC A to the node IP address 10.0.0.3 can be performed using NAT as per standard methods.

Figs. 15 – 17 show an embodiment in which an IP collision exists between nodes 1, 1' within a link aware DNS serviced private network, with no cross routing realm communications (all communicating nodes are one hop from the link aware router 2).

Where such an IP collision exists between communicating nodes, then link aware twice NAT can be used, i.e. link aware NAT is carried out for both the source and destination IP addresses.

In such an approach, a unique IP address (for which there is no address collision), in this case 11.0.0.2 is co-located with the sender node 1' that does the name lookup. The link aware NAT records the co-located IP address 11.0.0.2 and matches it to the sender node IP address 1.1.1.1 and its MAC address B, and incoming NIC information and VLAN ID if any. This is stored in a link aware NAT table 18.

DNS_ALG then returns another unique IP address 11.0.0.1 that is co-located with the corresponding node 1 in the DNS reply to the sending node 1'. This IP address 11.0.0.1 is stored by the link aware NAT along with the client node's IP address 1.1.1.1 and MAC address A, and NIC interface information and VLAN ID, if any. This can be derived in the same manner as in the embodiment of Figs. 12-14, and is stored in a reverse link aware NAT table 17.

These two co-located IP addresses 11.0.0.1 and 11.0.0.2 are valid for the duration of any existing communications between the two nodes 1 and 1', before another name lookup.

Any communications between the two nodes 1,1' are twice NAT. The difference with traditional twice NAT being that the available link layer and NIC information removes the need for any routing table lookup (which breaks when there is an address collision) or ARP requests (which can generate multiple replies when there are address collisions) in order to construct and deliver a packet to any one of the nodes 1,1'.

The DNS_ALG bindings 16, the link aware NAT table 18 and the link aware reverse NAT table 17 are set up when the correspondent node with MAC address B issues a DNS query.

Fig. 16 illustrates the NAT operations for traffic from the client node 1' with MAC address B to the correspondent client node 1 with MAC address A with link aware twice NAT, whilst Fig. 17 illustrates the NAT operations from the correspondent node 1 with MAC address A to the client node 1' with MAC address B with link aware twice NAT.

In the case of cross routing realm communications (initiated from a private network to the Internet or vice versa) and address collision occurring between the private node and the public node, a unique IP address within the private network must be co-located with the public correspondent node, a unique public IP address belonging to the link aware router must be co-located with the private node, and twice NAT with DNS_ALG is employed to enable communication as documented in "DNS Extensions to Network Address Translators (DNS_ALG)", RFC 2694, Section 6. If address collisions also occur within the private network, link aware twice NAT for traffic from public node to private node but not for traffic from private node to public node NAT should be used.

Overall, the present inventive features, especially when used with the reconfigurable network system of PTC/SG00/00170, provide many advantages.

The technology is intelligent enough to provide network access on demand to end users without needing administrator intervention, and is configurable when necessary. That which would require reconfiguring end users' systems individually, can be configured and deployed by an administrator for a whole network in real-time, and configuration change can even be applied up to individual user granularity. For example, from a web-based server administration page, a network administrator could remotely: Switch to a new SMTP; change the Intranet web server machine or URL; install and make accessible to all users a new network resource, e.g. a printer; assign a new fully qualified domain name or internal hostname to an end user's computer; or assign a publicly accessible IP address to an end user's computer.

Further, the technology is not only able to connect client nodes to the network, but can be the launch pad for intelligent service delivery. Servers can have a complete database of every client system in a network from their user information to their current location, and an administrator can configure specific services to be

made available to selected users, and can specify access to such services based on physical location and user profile.

Permanent human understandable virtual FQDNs or hostnames can be used to identify client nodes, instead of depending on unfriendly IP addresses, and an end user can reach a peer correspondent node by its assigned name, even if the peer changes IP addresses or roams to a new location.

For example, a sales staff with a colour laser printer quota can wirelessly connect to a LAN and specify delivery of an urgent colour brochure to a colour laser printer (restricted access) nearest to a colleague who is in some unknown part of a building. Also, a sales presentation could be digitally transmitted to the first vacant meeting room (with no client nodes detected) and information could be fed back via an automated memo to the specific presenter the instant the presenter is connected to the network.

Security controls can go all the way from a physical socket connection to an actual users individual applications access and monitoring. Firewall rules, access control and network policies can be implemented from a centralised server with network configurable point-to-point security, and, at the link layer, rules can be imposed on individual clients' link layer MAC addresses and network point of origins. At the network and transport levels, a server can operate like a host-based firewall with dynamic firewall rules, insertion and deletion based on a client's physical location and user classification, etc. A web-based authentication page can further grant access to specific services to the exact port a user is connected to in real-time. A dynamic firewall could automatically delete the access privileges when a client is disconnected from the network.

At the application level, common application requests like DNS, email (SMTP), and web (HTTP) may be monitored and can have their behaviour controlled based on a physical connection, link frame or network address basis/criteria.

It is to be understood that various alterations, additions and/or modifications may be made to the parts previously described without departing from the ambit of the present invention, and that, in the light of the teachings of the present invention, the various network components and functions may be implement in software and/or hardware in a variety of manners.

Claims

1. A computer network system including a plurality of client nodes, each of which has a unique data layer link address, wherein said system includes at least one network device which is able to access the data link layer address of a said client node, and wherein said network device uses said accessed data link layer address to uniquely identify the said client node and to provide network layer or above services to the node based on said data link layer address.

2. The system of claim 1, wherein said system includes at least one router which is able to determine the link layer address of a client node which has sent a data packet, and includes a database of link layer addresses and user information.

3. The system of claim 2, wherein said router conducts one or more policies for said client nodes based on said link layer address and database information.

4. The system of claim 2 or 3, wherein said router determines whether a client node's link layer address is included in said database, and initiates a configuration procedure when said address is not in said database.

5. The system of claim 2, 3 or 4, wherein said router determines whether an end user's link layer address is included in said database, and initiates a security procedure when said address is not in said database.

6. The system of any preceding claim, wherein said system determines positional information of a said client node and records said information in said database.

7. The system of any preceding claim, including a DNS server, wherein said DNS server assigns a unique name to a client node by mapping the client node's link layer address with said unique name.

8. The system of claim 7, wherein the DNS server includes a database of link layer addresses to fully qualified domain names.

9. The system of claim 7, wherein the DNS server includes a database of link layer addresses to hostnames.

10. The system of claim 7, 8 or 9, wherein when traffic is generated by a client node, the router determines the unique name for the client node from its link layer address, and updates the IP address for said name in a DDNS table.

11. The system of any preceding claim, wherein said system includes a Network Address Translator for mapping a client node IP address to an assigned IP address.

12. The system of claim 11, wherein said Network Address Translator uses a client node's link layer address in the mapping of an IP address of said end node to an assigned node.

13. The system of claim 11 or 12, wherein said Network Address Translator uses physical location information for a client node in the mapping of an IP address of the client node.

14. The system of claim 11, 12 or 13, wherein said Network Address Translator provides a reverse NAT operation in which an assigned IP address is mapped to a client node IP address and data link layer address.

15. The system of any of claims 11 to 14, wherein said Network Address translator provides a twice NAT operation in which mapping is provided both for a client node IP address to an assigned IP address and for an assigned IP address to a client node IP address and its data link layer address.

16. A computer network system including a plurality of end systems, a NAT router, and a DNS server, wherein said router is able to determine the link layer addresses of the end systems, and wherein NAT and DNS procedures are based on said link layer addresses, each said link layer address being associated with a unique name.

17. A computer network system including a plurality of end systems and a NAT router, wherein said NAT router creates bindings between actual source end system

IP addresses and apparent source end system IP addresses, and wherein said NAT router records the link layer addresses of said source end systems as a part of said binding.

18. A computer network system including a plurality of end systems, and a DNS server, wherein said DNS server maps link layer addresses of said end systems with unique names.

19. A computer network system including a plurality of end systems and one or more intermediate systems, and in which data is transferred between the end systems through the intermediate systems using network and link layer protocols, wherein the end systems are uniquely identified in the system by assigning a unique name to the link layer address of each end system, said unique name being used in the routing of data to the end systems by the intermediate systems.

20. A computer network system including a plurality of client nodes and a proxy server, wherein the proxy server is configured to retrieve data link layer address information from data packets sent by said nodes and to service said client nodes in accordance with network level or above policy specifications based on said data link layer address information.

21. A computer network system including a plurality of network nodes having data link layer addresses and a router for routing traffic from and to said nodes, wherein said system includes a database of said data link layer addresses to attributes of the nodes, said router being configured to determine the data link layer addresses of said nodes and to perform look-ups of said database in order to service said nodes.

22. A method of servicing nodes on a computer network which includes a plurality of such nodes and a router for routing traffic from and to said nodes, including the steps of:

obtaining data link layer addresses for said nodes;

constructing a database of said addresses to attributes of the users of said nodes;

performing a look-up of the said address of a node in said database when traffic in received from said node; and

servicing said node based on the attribute information obtained from said look-up.

23. A method of NAT in a computer system which includes a plurality of nodes have data link layer addresses and a router for routing traffic from and to said nodes, including the steps of:

determining the link layer address of a said node; and

creating a binding between an IP address assigned by the router and the IP address of the node and its link layer address.

24. A method of identifying a node in a computer network which includes a plurality of nodes having data link layer addresses, and a router for routing traffic from and to said nodes, including the step of:

providing said network with a DNS server which maps a unique name to the data link layer address of a said node.