Address Resolution

As mentioned earlier, different layers of the protocol stack use different kinds of addresses. We can now see that the Transport Layer (TCP) uses port addresses to route data to the correct process, the Network Layer (IP) uses IP Addresses to route data units across the network, and the Data Link Layer uses MAC addresses to identify the destination NIC on a particular link. The task of determining the appropriate addresses at any stage in the process is called Address Resolution (or sometimes Name Resolution).

Resolution Types

NetBIOS name to MAC Address
NetBIOS name to IP Address
IP Address to MAC Address
Private IP Address to Public IP Address
Domain Name to IP Address

Resolution Mechanisms

There are two fundamental approaches to Address/Name Resolution. The simplest is to simply install a table on every host containing all the translations it might ever need. On a small network which does not connect to the Internet this is quite practical. It requires, however, that the Network Administrator maintain these tables. If hosts are added or deleted from the network, than all the tables in every host need to be manually updated.

The most common such tables are HOSTS and LMHOSTS. The HOSTS table is used for recording domain names and their corresponding IP Addresses. The LMHOSTS table us used for translating computer (NetBIOS) names to IP Addresses.

The second method for performing Address Resolution is to keep a single table on a single host and have a server on that host deliver the translations as required.

1For this reason, IP Addresses are frequently referred to as Layer 3 addresses and MAC Addresses are referred to as Layer 2 addresses. Furthermore, the devices that work with these addresses are similarly referenced: Routers are layer 3 devices while switches are layer 2 devices.
WINS (Windows Internet Name Service) replaces the LMHOSTS table and provides a central repository of NetBIOS names and their IP Addresses. DNS (Domain Name Service) is implemented on the Internet and translates domain names to IP Addresses. The data base for this, of course, is gigantic, so DNS’ data base is distributed across the network on many Domain Name Servers.

NetBIOS name to MAC Address resolution is accomplished by broadcasting a ‘query’ request to all hosts on the network and receiving a ‘recognition’ response from the machine with the matching name.

IP Address to MAC Address is accomplished through the use of a protocol called ARP (Address Resolution Protocol). ARP accomplishes its task by broadcasting query requests on the link similarly to the mechanism described in the previous paragraph. ARP also keeps a cache of recent translations in each host so that recent translations are remembered and don’t have to be continuously redone.

Private IP Address to Public IP Address is required in those environments where hosts on a LAN connect to the Internet only through a single dedicated interface host, called a gateway, or proxy server. On the LAN side of the gateway there may be a large number of hosts, each with their own IP Address. These IP Addresses, however, are private in the sense that the Internet is unaware of them. In fact, they need not be registered and may duplicate IP Addresses elsewhere in the world (but not, of course, in the same LAN.) On the Internet side of the proxy server is just a single interface to the Internet with a single, unique, IP Address.

The Proxy server uses a protocol called Network Address Translation (NAT) to translate the Private IP Addresses to its public address, and to determine how to do the reverse translation when response come back from the Internet to one of the LAN hosts.

Domain Name Service

The Domain Name Service is the Internet’s mechanism for providing an IP Address for a known domain name. It relies on the hierarchical nature of domain names to distribute the work needed for such translations, and the Domain Name Servers which implement the service are themselves arranged hierarchically.

At the top of the DNS hierarchy are the root name servers. These servers are aware, primarily of the top-level domain names. At the bottom of the hierarchy are local name servers. The local name servers are aware of all the domain names and IP addresses for their local networks. These servers may also be the Authoritative Name Servers:
the servers which are known to have the translations for a zone. The root name servers know the authoritative name servers for all top level domains. Between the local name servers and the root name servers are other domain servers which may support a particular domain or subdomain, plus regional name servers which are aware of second-level domain names.

The DNS data base is a collection of resource records. There are different kinds of resource records for different translations that DNS is able to perform.

<table>
<thead>
<tr>
<th>Resource Record</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOA</td>
<td>Start of Authority Identifies this server as an authoritative server</td>
</tr>
<tr>
<td>NS</td>
<td>Name Server Identifies another authoritative name server</td>
</tr>
<tr>
<td>A</td>
<td>Address Contains a domain name to IP address translation</td>
</tr>
<tr>
<td>CNAME</td>
<td>Canonical Name Contains an alias to domain name translation</td>
</tr>
<tr>
<td>MX</td>
<td>Mail Exchange Identifies a mail server</td>
</tr>
<tr>
<td>PTR</td>
<td>Pointer Contains an IP address to domain name translation (reverse lookup)</td>
</tr>
</tbody>
</table>

DNS Name Resolution

As a rule, the resolver formulates the initial query for a domain name translation. Such a query may result in getting the actual IP address for the requested domain name, in which case the query is known as a recursive query. On the other hand, The response to an initial query might be the IP address of another name server to which an additional query must be made. This kind of query is known as an iterative query.

When a name server receives a recursive query it tries to return the requested translation from its own data base. If it can’t find the domain name in its data base it creates a query of its own to another name server in the hierarchy. When a name server is finally reached which contains the domain name, the IP address is returned back along the same path from name server to name server until it reaches the original requester.

When a name server receives an iterative query and it does not contain the requested domain name, it passes the IP address of the next name server back to the original requester. That requester now generates a new DNS request to the next name server, which repeats the process. In this case, when the name server containing the desired domain name is queried, it returns the IP address directly to the original requester.

Every DNS server which processes a DNS request maintains a cache of recent

---

2 A zone is a subset of a domain or subdomain. It defines the region of the domain the name server is responsible for.
requests and the results. Thus, if a requester makes a second request for the same domain name the local server will have the translation in its cache and can immediately return it. Such translations are kept in the cache for a limited amount of time, called the Time To Live (TTL) and is specified in the SOA record of the name server.

**Dynamic Host Configuration Protocol (DHCP)**

DHCP is not an address resolution method, but a method for assigning Logical (IP) Addresses to hosts. Without DHCP, the system administrator has to manually assign an IP address to each host on the LAN. Assuming that these hosts are visible on the Internet, than these manually assigned addresses are registered with ICANN and were supplied by the LAN’s ISP or other supplier of IP addresses to the organization owning the LAN.

When IP addressed are manually assigned this way, they are known as static addresses - they do not change unless the network administrated explicitly changes them. In some cases this is important. For instance, if one of the hosts is running a web server, than that web server must have the IP address associated with the web sites domain name. It may not be changed.

On the other hand, if it is not important that the IP addresses be fixed, then it makes since to put in place a mechanism that automatically assigns IP addresses to hosts without an administrator’s intervention. This is what DHCP does.

DHCP requires a DHCP server on the network which monitors the network and automatically assigns addresses to hosts as they are added to the network. The DHCP is configured with two parameters. The first of these is the scope. The scope is a range of IP addresses from which the DHCP server can select address to assign to hosts. The scope is usually set to be either 1) a block of addresses assigned to this organization by ICANN or an ISP, or 2) a block of addresses from the private addresses specified by ICANN.

Recall that IP Address can be either Class A, Class B, or Class C addresses. The address class can be determined by looking at the first decimal integer of the IP Address. Within each class, a range of addresses is designated as private; these private addresses are never assigned to anyone for use on the Internet, and are intended to be used by LANs which either don’t need to be addressed by other hosts on the Internet, or reside behind gateways or proxy servers and are thus invisible to the Internet. The following table shows the private addresses associated with each class.
I will use the generic term ‘gateway’ to include routers that connect LANs to the Internet as well as Proxy Servers and other gateway devices.

### Network Address Translation (NAT)

Hosts on a LAN generally communicate with the Internet through a designated host on the LAN called a gateway. Gateways present just a single host image to the Internet despite the fact that there are many hosts on the LAN; that is, the Internet sees just a single IP address where there are, in fact, many IP addresses on the LAN. Software or hardware within the Gateway most handle the translation of internal LAN IP addresses to and from the single IP address visible to the Internet. The protocol which defines how this is done is **NAT**.

The NAT device (gateway) maintains a table which is used to perform the required translations. The table contains the following information.

<table>
<thead>
<tr>
<th>LAN IP Address</th>
<th>LAN port</th>
<th>Internet IP Address</th>
<th>NAT port</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.0.10</td>
<td>3205</td>
<td>127.43.254.72</td>
<td>44444</td>
</tr>
</tbody>
</table>

*The LAN IP Address*

The private address of the host on the LAN. This will be replaced in the IP datagram header by the gateways own Internet-visible IP address.

---

3 I will use the generic term ‘gateway’ to include routers that connect LANs to the Internet as well as Proxy Servers and other gateway devices.
The LAN port
The port number assigned to the request by TCP\(^4\).

Internet IP Address
The destination IP Address from the IP header of the client datagram

NAT port
A port number chosen by the NAT device. It replaces the LAN port in the TCP header of the packet before it is sent out onto the network.

When a packet leaves the LAN the NAT device chooses a new, random, port number (from the non-reserved, or ethereal range of port numbers) and replaces the incoming port number with the new one in the TCP header. At the same time it replaces the source IP address in the IP header with its own IP address as known to the Internet. The original IP address and port number (socket) are place in the table along with the original destination address and the NAT-chosen port number.

When a response is returned from the remote host the table is searched for the Internet IP address (in the incoming IP datagram's source address field) and matching NAT port number (from the incoming TCP packet's destination address). When a match is found, the original substitutions are reversed and the packet forwarded onto the LAN.

\(^4\) The LAN IP Address and the LAN port together constitute an interface known as a socket. A socket uniquely identifies a process on a network. A socket is usually specified by catenating the IP Address and the port, separated by a colon. In the current example, the socket would be specified as 192.168.0.10:3205.