Tech Info Library

Open Transport 1.0.8: TCP/IP Features Q & A (3/96)

Revised: 3/26/96
Security: Everyone

Open Transport 1.0.8: TCP/IP Features Q & A (3/96)

===

Article Created: 27 October 1995
Article Reviewed/Updated: 26 March 1996

TOPIC ---

This article is a series of questions and answers on the TCP/IP features in Apple Open Transport 1.0.8.

Open Transport 1.1 is now available, and Apple recommends upgrading to it. Also refer to Open Transport 1.1 Reference Questions and Answers Tech Info Library articles for the most recent information.

DISCUSSION --

Question: What are some of the changes to the human interface for Open Transport/TCP?

Answer: The Open Transport/TCP configuration application represents a complete overhaul of the human interface from the MacTCP software it replaces. In addition to generic new features noted elsewhere (multiple saved configurations, recommended and required settings, on-line documentation, and so on), key new features include:

- direct entry of IP addresses and subnet mask in standard "dot notation";
- explicit selection of desired configuration method, now including DHCP;
- support for attachment to networks using Classless InterDomain Routing (CIDR);
- support for multiple entries in the router, name server, and explicit domain search lists; and
- improved support for large AppleTalk networks when using MacIP server/gateways.

Question: What are some of the upgraded features of Open Transport/TCP protocol stack?

Answer: With the broad adoption of TCP/IP -- and the tremendous excitement and visibility of the Internet -- Apple has made a significant investment in bringing a workstation-class implementation of TCP/IP protocols to the Mac OS. As with MacTCP, Open Transport/TCP is a full 32-bit stack. Open Transport/TCP adds support for:

- dynamic path MTU discovery, for more efficient network use in heterogeneous network topologies;
- Dynamic Host Configuration Protocol (DHCP), for centralized IP address configuration management. DHCP is an Internet Engineering Task Force (IETF) standards-track protocol;
- IP multicast, for participation as an MBone client;
- simultaneous TCP connections limited only by installed memory and processor power, for increased functionality as a Internet or other TCP/IP network server;
- ethernet version 2 and IEEE 802.3 framing, for better interoperability with a wider range of TCP/IP hosts;
- implicit and explicit domain name search paths, for increased control of domain name resolution; and,
- multiple IP routers with fail-over, for increased robustness in mission critical applications.

Question: How does the new support for Dynamic Path MTU discovery work?

Answer: Open Transport/TCP sets the "don't fragment" bit unless the packet size is larger than the MTU for the network. Intermediate routers are required by current RFCs to send back an "ICMP can't fragment" error when presented with a "don't fragment" packet that cannot be forwarded without fragmentation with that MTU size. In that event, Open Transport/TCP moves to the next smaller MTU size for that path and re-sends the packet, again with the "don't fragment" bit set. This process results in using the largest supported MTU size for off-subnet traffic.

Question: Which DHCP servers are supported by Open Transport/TCP?

Answer: Apple's implementation conforms to the current versions of the applicable specification documents (RFCs). To date, Open Transport/TCP has been tested with the following DHCP server implementations:

- Competitive Automation,
- FTP Software (http://www.ftp.com),
- Hewlett Packard HP-UX (http://www.hp.com),
- Microsoft Windows NT Advanced Server (see Network Planning and Administration),
- Silicon Graphics (http://www.sgi.com),
- Sun Solaris and SunOS (http://www.sun.com), and
- TGV (http://www.tgv.com).

Question: Does Open Transport/TCP support DHCP address leases?

Answer: Yes. Open Transport/TCP fully supports DHCP address leases. Open Transport/TCP will automatically attempt to renew any address lease that reaches it's Renewal Interval, which defaults to half of the lease's lifetime. (The Renewal Interval may be configured to a different value by making changes to the configuring DHCP server). Renewal will be attempted regardless of how many times the lease has already been renewed. Should an interface's IP address lease expire, the interface will be closed down.

Question: Does Open Transport/TCP support MacTCP "Server" addressing?

Answer: MacTCP Server mode addressing is a combination of the Bootstrap Protocol (BootP) and Reverse Address Resolution Protocol (RARP) configuration methods. When Server mode is selected, MacTCP will use BootP to attempt to acquire an IP address. If BootP fails to provide a valid address it would then try RARP. Whichever protocol was successful would be stored as a preference, and would be used first on next system startup. While this "fall-back" approach added a degree of robustness from the users point of view, it also added a degree of unpredictability from a network administrators point of view.

Based on customer feedback, Open Transport/TCP allows a network administrator to explicitly specify the single method they prefer to use. Thus while both RARP and BootP are supported, the Server mode does not appear as a choice in the Open Transport/TCP configuration utility.

Question: Does Open Transport/TCP support MacTCP "Dynamic" addressing?

Answer: No. MacTCP "Dynamic" mode addressing was based on an Apple-proprietary extension to TCP/IP protocols. It applied the address negotiation and assignment rules used by the AppleTalk protocols to TCP/IP networks, making it very easy to set-up a Macintosh only stand-alone TCP/IP network. Use of this Dynamic Addressing method in other scenarios, however, could create additional work for a network administrator.

The Internet community (the IETF) has since developed a multivendor standard for the dynamic assignment of IP addresses, known as Dynamic Host Configuration Protocol (DHCP). Apple has adopted the industry standard DHCP and dropped support for the earlier Apple "Dynamic" mode addressing with Open Transport/TCP.

Question: Does Open Transport/TCP support a local HOSTS file?

Answer: Yes. Open Transport/TCP supports a HOSTS file, stored in the System Preferences folder, that may be used to supplement and/or customize the Domain Name Resolver's initial cache of information. This file is parsed when Open Transport/TCP is initialized. As in MacTCP, the supported HOSTS file features follow a subset of the Domain Name System Master File Format (RFC 1035).

Should a HOSTS file be used, every effort should be made to keep it as small as possible, and to only include entries that will be accessed frequently. This reduces the total memory footprint required to cache the DNS information and minimizes the need to maintain and update the HOSTS files as system information changes over time.

In order to activate a HOSTS file, Open Transport/TCP must be configured using either the Advanced or Administrator mode to select the appropriate file. The text file must already exist, and can be created with any text editor or word processor. Also note that the HOSTS file selection is tied to the selected configuration. An administrator might, for example, specify different HOSTS files for use when a user connects via ethernet on the campus LAN and that same user when dialing-in from a remote location.

Supported features include blank lines, comments (indicated by a semicolon), and data entry. Comments may begin at any location in a line; they may follow data entry on the same line. A comment extends from the semicolon to the end of the line. Data entry must follow the format:

```
<domain-name> <rr> [<comment>]
```

where <domain-name> is an absolute or Fully Qualified domain name. The FQDN need not be terminated by a dot, but must contain at least one dot internally, and where

```
<rr> = [<ttl>] [<class>] <type> <rdata> OR [<class>] [<ttl>] <type> <rdata>
```

The only <class> currently supported is IN (Internet Domain); <ttl>, time to live, indicates the record's configured lifetime in seconds; and <type> can be A (host address), CNAME (canonical name of an alias), or NS (name server). If <ttl> is not present the entry is assumed to have an infinite lifetime; this may also be indicated by specifying a value of minus-one (-1). $INCLUDE and $ORIGIN are not supported.

Open Transport/TCP is somewhat more stringent regarding the format and content of the HOSTS file than was MacTCP. MacTCP permitted violation of the Fully Qualified requirement for <domain-name>; this feature was often used to avoid the necessity for entering CNAME records by associating an address directly with a non-fully qualified name. For instance, this format:

```
charlie               A        128.1.1.1
```

which was acceptable to the MacTCP DNR, is no longer permitted because of the use of domain search lists in Open Transport/TCP (charlie could potentially exist in any or all of the configured domains). To accomplish the same effect, use this format instead:

```
charlie               CNAME     myhost.mydomain.edu
myhost.mydomain.edu   A        128.1.1.1
```

This associates the local alias charlie with the fully qualified domain name myhost.mydomain.edu, and resolves it to the address 128.1.1.1. Use of local aliases is limited to CNAME entries; NS and A entries must use fully qualified domain names.

Question: How does the new Open Transport/TCP Domain Name Resolver work?

Answer: When a client of the DNR requests a name-to-address mapping, the DNR checks for a "." at the end of the name. If it exists the name is assumed to be fully qualified (RFCs 1034 and 1035), and the DNR will search for that name. If the name contains at least one "." internally but does not end with "." it is considered to be provisionally fully qualified. The DNR will begin a search for these names without further manipulation.

Otherwise the name is assumed to be partially qualified. The DNR will begin a search for the name in the domain name configured in the "Default Domain name:" field. For example, an attempt to resolve joe with a Default Domain of tech.support.apple.com would look for joe.tech.support.apple.com.

If the requested name is not found and an optional Admin Domain has been configured using either the Advanced or Administrator mode, implicit searches will take place next. Continuing with the example, with a Default Domain of tech.support.apple.com and an Admin Domain of apple.com, a search for the name joe would look for the following additional names:

joe.support.apple.com
joe.apple.com.

Implicit searching will stop when the name is resolved, or when the FQDN becomes equal to the name-to-be-resolved concatenated with the Admin Domain (that is in the example no implicit search would be made for joe.com). Implicit searching will not be attempted unless an Admin Domain is explicitly configured and the Default Domain is a subdomain of the Admin Domain (per RFC 1535).

If the name is still not found, the explicit Search Domains are searched. For each search domain the configured name server(s) are contacted in the order specified in the Name Servers field. If the name is resolved in the first search domain that answer is returned; other Search Domains will not be checked. If an authoritative answer that the "name-does-not-exist" is returned the DNR immediately begins the search in the next configured Search Domain. The search continues through the configured Search Domains. If still no match is found, the DNR will search the root domain if it makes sense to do so.

The DNR has an overall time-out of 2 minutes, after which it will abandon its search.

Question: What is MacIP?

Answer: MacIP, sometimes also referred to as KIP (Kinetics Internet Protocol), is a protocol specification developed as a method for carrying TCP/IP traffic on AppleTalk only networks -- originally these would have been LocalTalk networks. MacIP is today frequently used in conjunction with AppleTalk Remote Access Protocol (ARAP) to provide mobile users access to TCP/IP network services. MacIP specifies encapsulation of TCP/IP datagrams in AppleTalk packets for transmission over such connections.

Use of MacIP requires a MacIP gateway. AppleTalk encapsulated IP packets are sent to the MacIP gateway using AppleTalk protocols (DDP). The gateway strips off the AppleTalk encapsulation and places the IP packet on the TCP/IP LAN. When packets are destined for the MacIP end-node, that gateway provides the needed encapsulation services.

MacIP gateway support is most frequently offered as an integrated service within a multiprotocol router. The gateway (router) attaches to both an AppleTalk and a TCP/IP network, acting as a middleman between the MacIP end-node and the appropriate TCP/IP-based hosts on the LAN or WAN.

Open Transport includes end-node support for MacIP. A end-node is configured to use MacIP using the TCP/IP configuration utility by selecting "AppleTalk (MacIP)" in the "Connect via:" pop-up menu. The user (or network administrator) must also specify where on the network (in which zone) to look for the MacIP gateway. Once selected, TCP/IP will be encapsulated in AppleTalk and will be sent out the "Connect via:" interface selected using the AppleTalk configuration utility.

Article Change History:
26 Mar 1996 - Added statement on Open Transport 1.1 release.

Tech Info Library Article Number:18832