Callsign Server for the KA9Q Internet Protocol Package

by

Douglas Thorn (N6OYU)

Dewayne Hendricks (WA8DZP)

INTRODUCTION

One of the problems we found with packet communications, after we did a few keyboard to keyboard connections and log-ins to BBS's, was what else does it do for us? We can send mail to all these people out there in packet land that we do not know, or we can watch messages fly by that don't interest us..... What is missing is a true user application. Hmm... We thought, there has to be something we can do for the packet community. One afternoon, we were logged into the Internet (a world wide computer network) and saw a message that a group of people were putting a project together to acquire a copy of the Amateur Radio Callsign database from the F.C.C. Great, here it is.... We could put together a service whereby people could contact a packet station, and lookup a callsign. This could be an interesting service, and if successful, provide something of use to the community. Since we knew one of the people involved in getting the data, we contacted him, and got a copy of the database. As it turned out, the actual data file is 108 Megabytes in size and contains over 435,000 callsigns. Unfortunately, it only contains US callsigns, but that's a great start. Each record of the file has in addition to the callsign, both the mailing and station addresses, the class of license, previous callsign, renewal/process/expiration dates, and even the persons birthdate! Wonderful.... now all we had to do was figure out how to give access to this data from a packet station, where was we going to store this huge database file, and how do we tell people about it....

ENTER THE KA9Q TCP/IP PACKAGE....

We were working on another project, porting the KA9Q TCP/IP package to the Macintosh, when several people in our area asked the same question. What are we going to do with TCP/IP? It is certainly a neat system, but without applications to make use of it, it suffered from the same old problem mentioned above. Then the light flashed, how about interfacing the callsign database to one of the TCP/IP servers.. . like the **finger** server. The **finger** server is a utility built-in to the TCP/IP package that allows a remote station to query your station for basic information. Great idea said Dewayne (WA8DZP, the programmer who did all the work!), and in a few days he had written the initial code to access the callsign datafile. It took several more weeks of debugging and testing, but finally we had an extension to the **finger** command.

IMPLEMENTATION

Before we can provide a service, we need to have a place to store the datafile, and of course the datafile itself. To start off, we contacted Rusty Carruth (N7IKQ) who headed up the project to acquire the data from the F.C.C. Rusty provided a copy of the data on a 9 Track magnetic tape. Since our plan was to put the data on a Macintosh, the next task was to get the datafile converted into a format that could be used. This was done with the help of several people at Apple Computer, who moved the data onto a DEC VAX system, from where it wa.s transferred (via Xmodem) to a Macintosh with a 300 Meg drive.

Now that the data was on a Macintosh, we moved it to an AppleShare™ fileserver at N6OYU's QTH. Since the Macintosh version of the KA9Q Internet Protocol Package supports multiple volume access, the next task became the implementation of the software to access the datafile, and return the requested callsign information via a packet station.

From here, Dewayne proceeded to develop the necessary routines required to access the datafile. Since the Finger client/server already provided much of the user interface we had in mind, it was used as a point of departure. Logic was added to the Finger server to look for a '%' as the first character in the parameter string passed to it from the user. When this character is received, control is passed to a new module which parses the callsign request and accesses the callsign database for the requested information.

This was all fine, but what about all the rest of the packet user's out there that do not have the TCP/IP package up and running. Well, as it turns out TCP/IP as implemented, will support AX25 connections as well, and even provides a mail box function. So additional code to extend the mail box function was added to allow an Inquiry of the database. This addition of the I)nquire command to the mail box code now provides the ax25 connects access to the database.

Below are examples of how this all works. (Bold-Italic character are typed by the user)

```
net>  finger  %wa8dzp@n6oyu<cr>
SYN sent
Established
[N6OYU.norcal.ampr.org]

Name:                 DEWAYNE L. HENDRICKS
License:              WA8DZP         License Class: E
Mail address:         43730 VISTA DEL MAR, FREMONT, CA 94539-
0000
Station address:      43730 VISTA DEL MAR, FREMONT, CA
Effective date:       May. 17, 1988  Expiration date: May. 17,
1998
Previous Callsign:                   Previous Class : A
```

```
Birthdate:              Oct. 11, 1949     Process date: May. 17,
1988

Close wait
Last ACK
Closed (Normal)
```

Note the use of the **%callsign@n6oyu** syntax in this command. The typical finger command looks like this: **finger doug@n6oyu**. This will look for a text file named 'doug' on the system diskette, and copy it's contents to the TNC. With the Callsign server extension, we added the % to tell the finger server to lookup the following callsign in the database, and return that information to the TNC.

As it turns out, TCP/IP allows the use of another command to query information provided with the **finger** command. This involves using the **telnet** command (telnet is the command used for keyboard to keyboard communications with another TCP/IP station). This gets fairly complicated, but suffice to say that it works. Below is another example of using the **telnet** command to get the same information:

```
net> telnet n6oyu 79<cr>
SYN sent
Established
[N6OYU.norcal.ampr.org]
%ka9q<cr>

Name:                   PHILIP R. KARN JR
License:                KA9Q            License Class: E
Mail address:           25B HILLCREST RD, WARREN, NJ 07060-0000
Stat ion address:       25B HILLCREST RD, WARREN, NJ
Effective date:         Sep. 27, 1988   Expiration date: Sep. 27,
1998
Previous Callsign:                      Previous Class:
Birthdate:              Oct. 4, 1956        Process date: Sep. 27,
1986

Close wait
Last ACK
Closed (Normal)
```

In the above example the **79** tells the **telnet** server to forward the request to the **finger** server, which in this case is the **% ka9q** on the next line. The server processes the request as before.

Below is a example of an AX25 request:

```
Connect N6OYU<cr>
Conn pending
Connected
<Carriage Return>
```

```
[NET-$]
Welcome to the N6OYU.norcal.ampr.org TCP/IP Mailbox
 (C)hat (I)nquire (S)end (B)ye >
I  K6LLK<cr>

Name:                JOHN D. CRONIN JR.
License:             K6LLK          License Class: E
Mail address:        1543 FORDHAM CT, MOUNTAIN VIEW, CA 94040-
0000
Station address:     1543 FORDHAM CT, MOUNTAIN VIEW, CA
Effective date:      Dec. 9, 1986   Expiration date: Dec. 9,
1996
Previous Callsign:                  Previous Class:
Birthdate:           Jan 1, 1944       Process date: Dec. 9,
1986

 (C)hat (I)nquire (S)end (B)ye >
B<cr>
Disconnected
```

Now all packet user's have access to the Callsign server, via several different mechanisms. Since the bringing up of the sever, there have been over 3500 accesses over a 6 month period. The service proved especially useful during the last Field Day exercises, with several hundred requests during the weekend. An additional observation is to see what each new user does with the server. First, almost without exception, everyone looks up their own callsign!.... Then they look up their friends....

What is all this running on. Well, the database file is on a 300 Megabyte Hard Disk drive which in turn is connected to a Macintosh Plus computer running Apple's AppleShare™ fileserver software. The radio is a Yaesu FT-211RH connected to an AEA PK-232 TNC and another Macintosh Plus computer running the Macintosh version of the KA9Q TCP/IP package. The two computers are connected together via LocalTalk™ (Apple's networking system). Additionally a Macintosh IIx color system and a LaserWriter IINTX printer also share the network.

Future Directions

The future holds many changes for the service. The first will be, of course, a more current data file. Next, we plan on changing the method of access. As it stands currently, the only was to get callsign information is to directly connect to the station via either ax.25 or TCP/IP. What we plan on is similar to the white pages lookup now available in the PBBS network. You will be able to send a message to the system, with whatever means you have, and the system will send a reply message with the callsign information you requested. Now back to the coding....