Towards an Open D-Star Reflector System

By Jonathan Naylor, G4KLX

Introduction

Many criticisms have been levelled at the D-Star system, mostly surrounding the use of the proprietary AMBE codec. However the availability of this codec in chip form for $20 does help to defuel this argument slightly. What is less discussed is the closed nature of the D-Star reflector system, which is not documented, and those who have studied it aren't usually willing to discuss it, often for political reasons. This is not a healthy state of affairs, and is certainly contrary to the spirit of amateur radio.

What this paper explores is a new D-Star reflector system, what distinguishes it from that system is that the design of this system is open, and independent implementation of the repeaters that use it, as well as the reflector itself is open to anyone to build, however this does not mean that the system will be insecure.

The Two Approaches

The current reflector system appears to use a centralised system which takes AMBE data from all the connected gateways and routes it to the appropriate gateway. This architecture also allows for all the gateways and hence repeaters to share what is in essence a global chat room, if configured that way. Dongle users can connected to either the reflector or the gateway running on the repeater itself.

The second approach is more akin to EchoLink where a central server holds information about all connected stations centrally, but the passed data is directly between the two gateways. Dongle users will have to connect to the gateway directly after querying the central server.

Each approach has its pros and cons.

The Repeater

Whichever approach is used, the repeater itself (as opposed to the gateway which is co-sited with the repeater) has only a simple job to perform. All incoming transmissions on the RF side are sent out as UDP packets to the gateway except in the case where the RPT2 callsign is set to the radio callsign or blank. Conversely incoming UDP packets are immediately transmitted unless an incoming radio transmission is already being received.

The header would also be transferred as a UDP packet and would contain a unique ID which covers this particular transmission, and each data packet will include this ID and a sequence number to ensure that out of sequence packets aren't transmitted. This general approach would apply to all data passed as UDP packets.

The Gateway

One of the basic tasks of the gateway, and common to both approaches would be that the gateway would be responsible for the connection between all of the repeaters operating at the same site. This

would be to ensure that setting RPT2 to one of the other repeater callsigns at the same site would do the right thing. The interesting part of the gateway is allowing for stations to communicate outside of the repeater complex.

For both approaches the gateway would connect to the reflector probably using SSL and after authenticating itself, would be accepted as a member of the network surrounding that reflector. The IP address could be extracted from the connection, and as part of the authentication, the gateway would pass the callsigns of the repeaters that it is responsible for. It would be expected that the reflector would re-authenticate the connection from time-to-time, and conversely the gateway would be expected to re-authenticate and pass its data again should there be a change in local circumstances, for example a change of IP address.

For the centralised approach, the gateway would simply pass all UDP packets from the repeater(s), bar the local ones, to the central reflector, probably using UDP. Conversely all incoming UDP packets would be sent to the appropriate local repeater. There would be very little local state, no need for local caching of addresses, callsigns, or anything else. The complexity would all be kept in the reflector.

For the decentralised approach things are more complex. The gateway would be responsible for querying the central reflector for two types of information, repeater callsign to IP address mapping, and individual callsign to repeater callsign, and IP mapping. With this information it would be possible to set up a UDP conversation to the appropriate remote gateway and pass the data. The remote gateway could potentially validate the incoming UDP packets by querying the reflector to match the IP address and callsign. It could also store the information held in the header to ensure that a reply could be sent without having to query the reflector.

For efficiency the gateway would locally cache the information from the reflector, deleting it a fixed time after the last use of that information, for example. In order to ensure that information is up to date, every time a change happens, an IP change of a gateway, or more importantly when a new station appears on a repeater, a message would be sent to the reflector. In turn the reflector would transmit this information to all connected gateways. If the gateway already had this information in the cache, it would be replaced, otherwise it would be ignored. Alternatively it could cache this information for efficiency, at the expense of local resources.

The Reflector

In the centralised approach, the central server would be receiving all data and routing it appropriately. It would inspect all the incoming headers to keep it's tables up to date, so very few explicit status messages would need to be passed between the gateways and the reflector. This means that it needs good network connectivity of high bandwidth, and any loss of the reflector would mean an end to the network.

The de-centralised server would simply be a database server, and not handling any other data so its bandwidth requirements would be lower. Another advantage would be that the loss of the reflector would not stop the network from operating, the gateways could use the cached data for longer and so maintaining some form of network connectivity.

Multiple Reflectors

The current system shares the load amongst several servers, there may also be load balancing within a single reflector but that is a separate issue.

For the centralised system, a request would be passed to the other reflectors, and the one which owns the requested gateway or station would reply, and from then on would receive the relayed UDP data which would then be sent to the required gateway.

For the de-centralised system, there must be a way for queries to be passed to other reflectors when the answer is not available from the directly connected one. The reflectors would connect to each other via SSL, and queries passed and answers passed back and relayed to the gateway making the query. There should be a way for a remote reflector to update another reflector, for example imagine a G station on holiday in the USA, this would require information about the G station to be deleted from the normal reflector serving the UK so that a query to all the other reflectors would be generated when a gateway wished to contact him.

DV-Dongle Users

The current system allows for DV-Dongle users to connect to either a gateway or a reflector. The centralised approach would continue this, maybe even using the existing DV-Dongle protocol.

The de-centralised version would only allow DV-Dongle users to connect directly to the appropriate gateway, although a query to the reflector would also be needed initially probably. A new set of protocols would probably be needed too.

Conclusion

I’m in favour of the de-centralised version, which would give most of the same benefits of the existing centralised structure, apart from the ability to relay general reflector traffic. I am very dubious about this facility anyway, having been in range of D-Star Repeaters that have been carrying such traffic and not being able to call through the local system due to it being blocked with such QSOs.