NRSE - Executive Summary

Andy Liow, Keiko Tada, Richard Smith, Toshihiro Aiyoshi

Overview

The GRS (GRID Resource Scheduling) project aims to enable GRID users to micro-manage QoS reservations at the edges of their networks on a per-flow basis. We have designed and implemented the NRSE (Network Resource Scheduling Entity) component of GRS. The NRSE provides a service interface, specifying an XML-based signalling protocol for service level requests and notifications. The service levels provided are a subset of IETF DIFFSERV; the NRSE requires a router that implements DIFFSERV. The NRSE also performs admission control functions to ensure there is sufficient capacity at the local router for every reservation. Reservations across multiple domains are made using two NRSEs, one at each end, with the assumption that the core network is over-provisioned.

Design

We chose the simple metaphor of an airline booking agent. The agent receives requests from passengers who specify where they want to fly from and to (source and destination IP addresses) and what time they want to fly. The passenger also specifies what class of seat he would like (QoS service level). The agent checks whether there are sufficient seats available on the flight. If there are, he issues the tickets. If the flight is fully booked, but passenger's journey is not urgent (non-realtime), the agent may be able to suggest an alternative flight. If the agent has to contact another agent to arrange the tickets, this is done transparently without involving the customer.

The travel agent is not concerned with actually flying the aeroplanes, just as we are not concerned with how the router meets our QoS specifications.

The diagram shows the NRSE receiving QoS requests from its clients (red), negotiating and scheduling with a remote NRSE (green), and then signalling to the network to implement the requested service at the appointed time (blue).

Conclusion

We successfully implemented a working prototype of the NRSE. There are still a number of rough edges that should be polished off before the code is

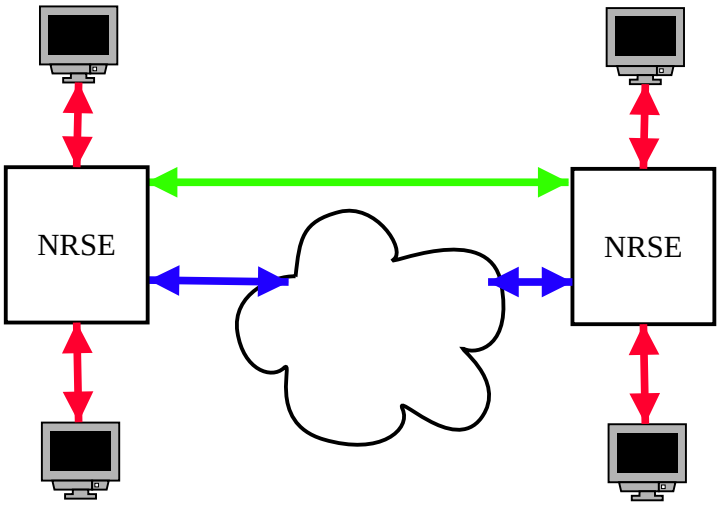

Figure 1: NRSE architecture

'production ready'.

We solved the problem of resource reservation for a single domain for realtime and non-realtime reservations. We also implemented multi-domain reservations for two domains for realtime requests.

Multi-domain non-realtime requests are a simple evolution and would not be difficult to implement given more time.

Most requirements were met. However, we did not have time to do much investigation of how the local NRSE can discover a remote NRSE. At present, remote NRSEs are entered by the administrator. We believe a simple entry in the DNS records should be used to automate the process. (Alternatively, there could be an LDAP database system holding details of NRSEs.) We also recommend that the administration and query features be expanded on.

A more serious problem is that of three or more domains. We ensure there is sufficient bandwidth available for a reservation at the local gateway, and also at the gateway to the destination network. We also mark the packets as EF, to give them priority throughout their journey. However, our packets may be forwarded through several other networks as they traverse the core of Internet. We can not guarantee enough bandwidth is available in all these networks. At present, core networks are usually over provisioned, so this is not a problem. Nevertheless, we suggest this aspect deserves further study.