A SELECTIVELY RELIABLE TRANSPORT PROTOCOL FOR DISTRIBUTED INTERACTIVE SIMULATION

J. Mark Pullen Vincent P. Laviano
Center for Excellence in
Command, Control, Communications and Intelligence
George Mason University

Keywords: Collision, Multicast, Reliable Service, Transport Layer (Layer 4)

ABSTRACT

In the real-time multicast environment of Distributed Interactive Simulation, fully reliable transport for all data is impractical. Multicasting of acknowledgements from all receivers does not scale, and reliable unicasting for non-multicast events cannot achieve real-time transmission. Presently, DIS applications transmit data using the unreliable transport protocol UDP, forcing the DIS application to perform network tasks such as the retransmission of reference positions, acknowledgement of collisions, and fragmentation or bundling of packets for optimum transmission. This paper addresses the need for a transport protocol that will perform these tasks in a standardized way, relieving DIS applications of the burden of solving network problems, and that can meet the demands of operating in a real-time multicast environment.

We propose a Selectively Reliable Transport Protocol (SRTP) which operates in three distinct modes of reliability: transmission of transient data that does not need reliable transport; transmission of reference entity-state protocol data units that must be received reliably; and timely, reliable transport of data to a single, dynamically determined receiver within the multicast group.

1. INTRODUCTION

DIS applications have special communication needs which are not met by presently available transport protocols. While the majority of the protocol data units (PDUs) generated by a DIS application are transient and do not require reliable transport, some of the PDUs represent important events in the shared virtual world of the distributed simulation (such as collisions) and must be received reliably in order to maintain the consistency of the simulation among the various participating nodes. Also, in order to run large distributed simulations with minimal disruption to real-time service, it has become necessary to employ a number of algorithms and techniques for managing network traffic between sites. Some of these algorithms (such as the Protocol Independent Compression Algorithm) reduce network traffic between sites by compressing entity state information using a reference/offset scheme. In such a scheme, reference PDUs containing entity state data which changes infrequently are multicast periodically, while smaller offset PDUs containing variations from the reference data are sent more frequently. The reference PDUs must be received reliably in order to maintain the consistency of the distributed simulation[CSTVH95][COHEN].

We see that DIS applications sometimes require the reliable, real-time transmission of data in a multicast environment. A fully-reliable protocol such as TCP is not sufficient due to connection overhead and acknowledgement delay. Also, neither positive nor negative acknowledgements scale well in a multicast environment. An unreliable protocol such as UDP is clearly not sufficient because, although it provides real-time transmission, it does not provide reliable service. Present work on log-based receiver-reliable multicast with a variable heartbeat scheme is making the reliability of multicast higher for real-time applications such as DIS[HSC95], but it is still the case that any fully reliable protocol will not operate in real-time.

Since no suitable transport protocol exists today, DIS applications are forced to compromise and use an unreliable transport protocol in order to achieve real-time transmission. The applications must then assume certain tasks that more properly belong to the network in order to ensure the reliable transmission of PDUs when necessary. This increases the complexity of DIS applications, making them more costly to design, implement and maintain. A standardized transport protocol that could meet the communication needs of DIS applications would remove this complexity (and the extra cost associated with it) and would contribute toward the DIS Steering Committee's stated goal of standardizing DIS protocols in order to increase interoperability among DIS applications.[DIS94]To fill this need, we propose a Selectively Reliable Transport Protocol (SRTP) which forms a sublayer between DIS applications and UDP/IP.

2. OVERVIEW OF SRTP

SRTP operates in three distinct transmission modes in order to transport efficiently data belonging to three classes:

- Mode 0 for data that does not require reliable transmission,
- Mode 1 for data that must be received reliably by all members of a multicast group, and
- Mode 2 for data that must be received reliably by a single dynamically-determined member of a multicast group.

DIS PDUs are classified into one of these three groups by individual DIS applications rather than by SRTP. This ensures that reliability needs will be met for each PDU, because the transport protocol need not make assumptions about these needs. In making this classification, DIS applications need only ask two questions about each type of PDU that is generated:

1. Must the PDU be received reliably? and
2. If so, must more than a single member of the multicast group receive the PDU reliably?

These questions can usually be answered by a DIS application before runtime. Therefore, this responsibility will not significantly increase the per-PDU processing time of a distributed simulation.

SRTP operates under the expectation that only a small percentage of PDUs generated by a DIS application require reliable transport. SRTP is capable of providing selective reliability with minimal disruption to real-time performance in this case.

There will be occasions when DIS applications will require that SRTP differentiate between different types of PDUs which fall into the same mode (as defined above). This mechanism takes the form of an SRTP category code that allows a DIS application to label classes of PDUs using a unique integer. This allows an application to associate a certain PDU with other PDUs (either previously sent or to be sent in the future). This is especially useful when using a compressed state transmission scheme in which entity state information is relayed using both reference and offset PDUs.

SRTP will also perform PDU fragmentation (when the length of a PDU exceeds the maximum transfer unit associated with the network protocol being used) and PDU bundling (to increase cost efficiency in applications in which there is high per-packet overhead).

To achieve these functions, we propose an SRTP header twelve bytes long and consisting of nine fields. The position, length and purpose of each of the header fields is summarized in Figure 1.

3. MECHANISMS OF SRTP

SRTP Mode 0 Transmission

Mode 0 transmission is for transient data that does not need reliable transport. There are two different types of Mode 0 data which SRTP has been designed to handle. The first type is transient data which is independent of any Mode 1 data. PDUs of this type are denoted Mode 0 Category 0. The SRTP sender passes Mode 0 Category 0 packets directly to UDP after encapsulating them within an SRTP header, and the SRTP receiver passes all Mode 0 Category 0 packets directly to the application program after removing the SRTP header.

All other Mode 0 data are associated with reliable Mode 1 transmissions. For example, PDUs containing offset information in a compressed state transmission scheme. These packets are handled in much the same way as Mode 0 Category 0 data, except that the SRTP sender attaches the sequence number of the most recent Mode 1 PDU of the same category (the associated reference PDU). If the SRTP receiver receives a Mode 0 packet with a sequence number that is higher than that of the most recently received Mode 1 PDU of the same category, a NACK (with the offending SN attached) is multicast to all members of the multicast group unless another NACK has been received for the same PDU within a specified period of time, chosen to reflect delay across the network being used. The intention here is to avoid "NACK storms" from adjacent hosts.

The majority of PDUs sent using SRTP are expected to be Mode 0.

SRTP Mode 1 Transmission

Mode 1 transmission is for data that must be received reliably by all members of the multicast group. It is expected that Mode 1 data will be reference information in a compressed state transmission scheme and will bear the same SRTP category numbers as the associated Mode 0 offset information. Mode 1 Category 0 is reserved for data that must be received reliably by all members of the multicast group and is not associated with any Mode 0 data. This is implemented using a

Header Field	Bit Position	Length (in bits)	Description
Type	0-1	2	Type of PDU (data, ACK, or NACK)
Mode	2-3	2	SRTP transmission mode (0, 1, or 2)
Cat	4-13	10	Category of data within transmission mode
F Flag	14	1	Indicates fragmented PDU
B Flag	15	1	Indicates bundled PDUs
SN	16-31	16	Sequence number
Dest Addr	32-63	32	(Mode 2 only) Address of intended receiver
FN	64-71	8	(Fragments only) Fragment number
FSN	72-95	24	(Fragments only) Fragmented PDU Sequence Number

Figure 1: Description of the fields of the SRTP header.

heartbeat scheme.

To achieve reliable transport, SRTP buffers the most recently sent Mode 1 PDU of each category and the sequence number of the most recently received Mode 1 PDU of each category. When the SRTP sender receives Mode 1 data, it increments the sequence number associated with the appropriate category, attaches the new SN to the PDU via the SRTP header, buffers the PDU, and passes it to UDP. When the SRTP receiver receives Mode 1 data, it buffers the sequence number attached to the PDU, removes the SRTP header, and passes the data to the application program. If the SRTP receiver receives a NACK for a Mode 1 PDU, it buffers all relevant information about the NACK such as source node, sequence number, SRTP category, and the time at which the NACK was received. In this fashion, the most recent NACK received for each category is buffered by SRTP. In addition, if the source node of the NACKed PDU is the same as the node receiving the NACK, then the SRTP sender is scheduled to retransmit the most recently buffered Mode 1 PDU of the same category as the NACKed PDU.

SRTP Mode 2 Transmission

Mode 2 transmission is for data that must be received reliably by a single dynamically-determined member of the multicast group. In this case positive acknowledgement is appropriate, so SRTP Mode 2 uses an ACK-based reliability scheme rather than the NACK-based scheme used by SRTP Mode 1. It would be possible to provide this service via unicast, at the cost of maintaining a separate network protocol, but the relative infrequency of Mode 2 data in DIS simulations makes it cheaper to embed Mode 2 transmission within the multicast stream.

As in Mode 1, SRTP buffers the most recently sent PDU of each category and the sequence number of the most recently received PDU of each category. When the SRTP sender receives Mode 2 data, it increments the sequence number for the appropriate category, attaches both the new SN and the address of the node for which the PDU is intended (via the SRTP header), buffers the PDU in the event that it may need to be retransmitted, and passes it to UDP. After a period of time (chosen to reflect the round-trip latency of the network) without receiving a positive acknowledgement for the PDU, the SRTP sender retransmits the PDU. When the SRTP receiver receives a Mode 2 PDU, it is dropped unless the destination address field of the SRTP header matches the address of the receiving node. This indicates that the receiving node is the intended receiver of the PDU, in which case the SRTP header is removed, the SN of the PDU is buffered, and the data is then passed to the application program. Upon receipt of a Mode 2 PDU at the proper node, an ACK is generated with the original PDU's sender's address in the SRTP header. This ACK is then multicast to all nodes. ACKs are ignored by all nodes for which they are not intended.

We emphasize that this mode of operation is only possible because Mode 2 transmission applies to rare events such as collision PDUs.

Registration Of Interest

To decrease the number of PDUs that must be processed by any given application, SRTP provides a mechanism by which DIS applications can register their interest (or lack of interest) in PDUs of a given category. In this way, applications only receive those PDUs of categories they intend to use. A DIS application may register and/or unregister interest in a category dynamically as many times as necessary during the course of a simulation, but many DIS applications will do this only once, statically, at the beginning of a distributed simulation. The SRTP receiver will only process PDUs if those PDUs are of a category in which interest has been registered (or are of category 0).

Distributed Logging

The recently developed Log-Based Receiver-Reliable Multicast (LBRM) Protocol uses a distributed logging mechanism along with a heartbeat scheme in order to provide "scalable, timely dissemination of state updates, meeting the needs of multicast sources"[HSC95]. We believe that these are useful mechanisms that can be incorporated into SRTP in order to provide for the reliable transport of state updates for infrequently changing data to all members of the multicast group.

4. IMPLEMENTATION OF SRTP

The CNE Networking Workbench is a tutorial programming environment created at George Mason University. The intent of the workbench is to simulate the performance of a moderate-sized packet network in such a way that its behavior under specified node and link-performance characteristics can be observed and measured. New software modules can easily be added to the workbench, allowing students and researchers to observe the impact of changing the functionality of the networking software without impacting our campus network. The workbench has two major components: a discrete event simulation module, which drives the simulation, and a library of C++ modules which provide basic versions of the networking software for a layered TCP/IP-style network. We simulated SRTP on the CNE Networking Workbench in order to test the logic of our SRTP design.

5. CONCLUSION

We believe the SRTP design is appropriate to its intended application in DIS. Our plan is now to create a usable implementation of SRTP within the UNIX environment and evaluate its effectiveness using DIS applications such as ModSAF. In preparation for this, we have submitted this paper to seek feedback on our proposal from the DIS community.

References

[CSTVH95] Calvin, J. O., J. Seeger, G. D. Troxel and D. J. Van Hook, STOW Realtime Information Transfer and Networking System Architecture, *12th DIS Workshop on Standards for the Interoperability of Distributed Simulations*, March 1995.

[COHEN] Cohen, D., Perceptronics, *10th DIS Workshop on Standards for the Interoperability of Distributed Simulations*, March 1994.

[DIS94] Dis Steering Committee, The DIS Vision: A Map to the Future of Distributed Simulation, May 1994.

[HSC95] Holbrook, H. W., S. K. Singhal and D. R. Cheriton, Log-Based Receiver-Reliable Multicast for Distributed Interactive Simulation, *Proceedings of SIGCOMM '95*, August 1995.