Moss UDP Protocol Specification – v. 1.00

Revised 5/23/2007

Overview

MossUDP is a networking protocol that allows efficient and scaleable transmission of data messages in a "one transmitter to many listeners" scenario. MossUDP is a lightweight protocol layer built on top of UDP that provides a mechanism for listeners to detect missed packets. It does not support retransmissions.

In MossUDP, each outbound packet is transmitted only once regardless of the number of listeners. Multiple messages may also be aggregated into a single network packet to reduce network traffic.

This document describes the messages sent between a MossUDP server and its clients. MossUDP transmitters send downstream packets via UDP multicast to transport the normal data stream sent to the listeners.

The MossUDP server will transmit on a well known multicast group for each type of downstream MossUDP datastream on a network. The listeners must subscribe to this multicast group to receive the downstream data.

Assumptions

All number fields in the MossUDP messages specified in this document (i.e. sequence number, message counts and message lengths etc) are binary numbers formatted in Big Endian mode (i.e most significant byte first).

Terms

Message

A message is an atomic piece of information carried by the MossUDP protocol.

MossUDP can theoretically handle individual messages from zero bytes up to 64KB in length although individual messages should be kept small enough so that the UDP underlying network protocol can efficiently carry the resulting MossUDP packets.

The contents of a MossUDP message are defined by the higher level application.

Session

A Session is a sequence of one or more messages.

While a single session can last indefinitely, typically the application will define a session to logically group messages together based on time delimitation.

Once a session is terminated, no more messages can be sent on that session.

A session is considered active if it has started but not yet been terminated.

Downstream Packet

A MossUDP transmitter sends “downstream” packets that are received by MossUDP listeners. A MossUDP packet may contain a payload of 0 or more data stream messages.

Each MossUDP packet consists of a Downstream Packet Header and of a series of Message Blocks. The Message Blocks carry the actual data of the stream.

Header

Downstream Packet

Field Name	Offset	Len	Value	Notes
Packet Length	0	4	NUM	Indicates the length in bytes of the packet including the length of this field (big-endian).
Session	4	10	ANUM	Indicates the current session
Sequence	14	4	NUM	Indicates the sequence of the first message in the packet (big-endian). This value should be treated more like a counter than a sequence, as there is no retransmission capability. It is here purely for keeping track of the fact that one or more packets have been dropped.
Packet Type	18	1	ALPHA	The type of packet U – data packet (carries 0 or more data messages) H – heartbeat E – end of session

Message Block

The first field of a Message Block is the two byte message length. The remainder of the Message Block is the actual message data. Subsequent Message Blocks will begin after the last byte of the previous Message Block.

Downstream Packet Message Block

Field Name	Offset	Len	Value	Notes
Message Length	0	2	NUM	Length of the following message
Message Data	2	*	ANUM	The message data.

* = Variable values

Heartbeat Message

Heartbeat messages are sent periodically by the server so receivers can sense packet loss even during times of low traffic. Typically, these packets are transmitted once per second and contain the next expected Sequence Number. A Heartbeat packet is a MossUDP packet of type H.

End of Session Message

End of session packet is a packet of type E and is typically sent at the end of the session. In case this packet is dropped you should assume the session has rolled over if the next packet you get has a session different from the one in the previous packet.

Receiver Example

A typical MossUDP receiver client would be configured with the following parameter:

- The UDP port to listen on and the Multicast group to join

A typical MossUDP receiver client might obey the following flowchart:

1. Open a UDP socket for the appropriate port and join the desired multicast group.
2. Examine the first received packet to determine the currently active session.
3. If the received session does not match the expected session, a new session has started.