NRL “ManetMsg” Overview

(a “PacketBB” implementation)

71st IETF - Philadelphia

10 March 2008

Brian Adamson

U.S. Naval Research Laboratory

Overview

- A general purpose “ManetMsg” C++ class created for MANET PacketBB packet/ message building and parsing.
- Based on specification: “`draft-ietf-manet-packetbb-12`”
- Freely available, open source from:

  `http://cs.itd.nrl.navy.mil/work/protolib`

  `http://downloads.pf.itd.nrl.navy.mil/protolib/nightly_builds`
- Part of NRL “Protolib” cross-platform toolkit
- A simple “`msgExample.cpp`” test application is provided to illustrate “ManetMsg” class usage.

ManetMsg C++ Classes

- Interrelated classes:
  - class ManetTlv
  - class ManetTlvBlock
  - class ManetAddrBlock
  - class ManetMsg
  - class ManetPkt
- All are based on “ProtoPkt” class that “wraps” around a buffer for message/packet building and parsing.

Packet Building Example

1. Instantiate a ManetPkt into a “buffer”, add tlv’s, etc:

```
ManetPkt pkt;
pkt.InitIntoBuffer(buffer, bufferSize);
ManetTlv* tlv = pkt.AppendTlv(tlvType);
```

2. Append a message, add tlv’s, etc:

```
ManetMsg* msg = pkt.AppendMsg();
msg->SetType(msgType);
ProtoAddress myAddr;
myAddr.ResolveLocalAddress();
msg->SetOriginator(myAddr);
tlv = msg->AppendTlv(tlvType);
```

3. Add an address block w/ tlv’s, etc:

```
ManetAddrBlock* addrBlk = msg->AppendAddressBlock();
addrBlk->AppendAddress(addr1);
addrBlk->AppendAddress(addr2);
tlv = addrBlk>AppendTlv(tlvType);
tlv->SetIndexRange(0, 1, true);
tlv->SetValue(value, index);
```

4. Finalize and send “pkt”:

```
pkt.Pack();
Send(pkt.GetBuffer(), pkt.GetLength());
```

Packet Parsing Example

1. Initialize a ManetPkt from a "buffer":

```
ManetPkt pkt;
pkt.InitFromBuffer(buffer, bufferSize);;
```

2. Iterate over any pkt-tlv's

```
ManetPkt::TlvIterator tlvIterator(pkt);
while ((tlv = tlvIterator.GetNextTlv()))
{
   switch (tlv->GetType())
   {   …
   }
}
```

3. Iterate over any messages, msg-tlv's, address blocks, etc:

```
ManetPkt::MsgIterator msgIterator(pkt);
while ((msg = msgIterator.GetNextMsg()))
{
   ProtoAddress origAddr;
   msg->GetOriginator(origAddr);
   ManetMsg::TlvIterator tlvIterator(*msg);
   while ((tlv = tlvIterator.GetNextTlv()))
   { …
   }
   ManetMsg::AddrBlockIterator addrBlkIterator(*msg);
   while ((addrBlk = addrBlkIterator.GetNextAddrBlock()))
   {
      addrBlk->GetAddress(addr, index);
      ManetAddrBlock::TlvIterator tlvIterator(*addrBlk);
      while ((tlv = tlvIterator.GetNetTlv()))
      {
            // You get the idea!
```

Next Steps

- Add state to enforce TLV ordering constraint, if applicable.
- Implement NHDP for *nrlsmf* and other use
  - "protolib/manet" source sub-tree also has "ManetGraph" class and `"graphExample.cpp"` that illustrates E-CDS relay set selection algorithm.
- Provide *tcpdump* binary trace files for off-line "interop" testing.
  - Any interested parties?
- Documentation!

“graphExample” Output