

UDP Transfer Protocols

JET Workshop

13 April 2004

Phillip Dykstra

Chief Scientist

WareOnEarth Communications Inc.

phil@sd.wareonearth.com

Why Not TCP?

- Slow adaptation after loss
 - 0 to 500 Mbps takes
 - 36 seconds, mtu=9000, rtt=72msec (DC to San Diego)
 - 15 minutes, mtu=1500, rtt=150msec (DC to Maui)
 - Short flow throughput is determined by slow start
- Loss != congestion
- Large TCP queues increase latency
- End users never tune their systems

NETBLT

- RFC969, 1985
- Block transfer protocol (not streaming)
 - Send a block at predetermined rate
 - Wait for lost packet list
 - Resend those, etc.

RBUDP

- Reliable Blast UDP
- <http://www.evl.uic.edu/cavern/quantum>
- Similar to NETBLT but in active development

Tsunami

- <http://www.anml.iu.edu/anmlresearch.html>
- Last release, Dec 2002
- UDP data, TCP control, rate adaptive
 - Loss rate controls sending rate
- File transfer protocol, no API

SABUL

- Simple Available Bandwidth Utilization Library
- National Center for Data Mining (NCDM) at UIC (University of Illinois at Chicago)
- <http://www.dataspaceweb.net/sabul.htm>
- 2000-2003, now in third generation
- UDP data, TCP control, rate adaptive
- Streaming protocol, window + AIMD rate control (not rtt dependent)
- Includes FTP like application, API

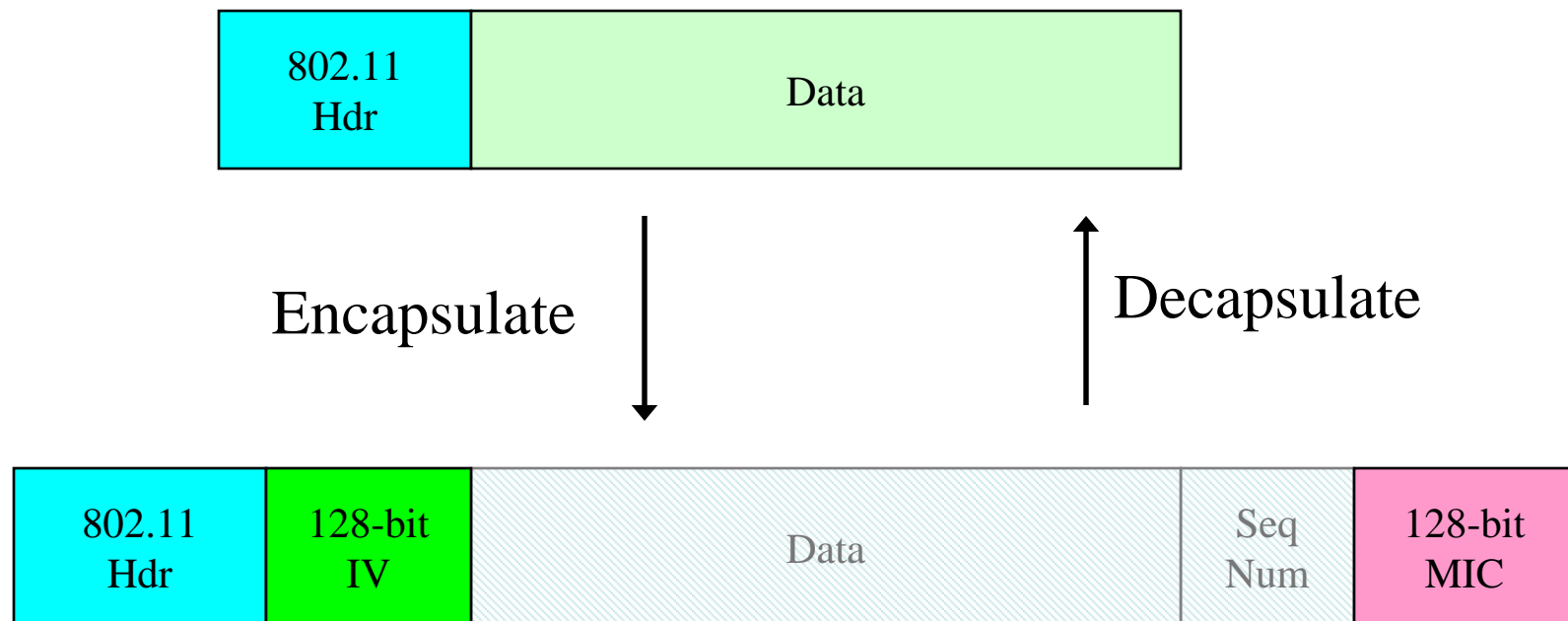
UDT

- UDP-based Data Transport
- <http://sourceforge.net/projects/dataspace>
- UDP for data and control
- Grew out of SABUL work, 2003+

UDP Protocol Security!

- Session hijacking, corruptions, encryption
 - Learn something from WEP+ ?

AES-Based WEP Format



Abstract Storage Layers

- P2P, e.g. BitTorrent
- I2 Logistical Networking LoRS
 - <http://loci.cs.utk.edu/>
- Dataspace
- Decouple LAN/WAN tuning issues
- Should storage be a network resource?