
IPv6 Header Compression

North American IPv6 Summit, June 2004

Emre Ertekin
Christos Christou
Booz Allen Hamilton
{ertekin_emre, christou_chris}@bah.com

Outline

➤ Introduction

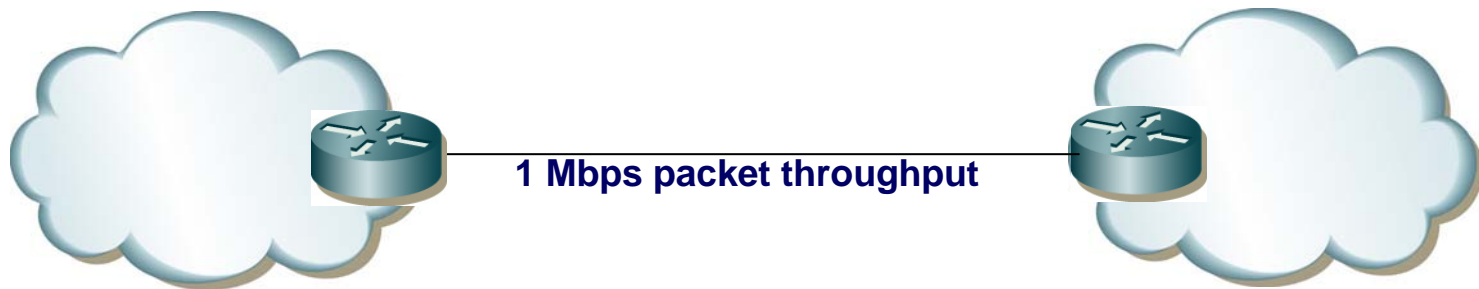
- **Background on IETF Hop-by-Hop Header Compression Mechanisms**
 - Fundamentals of Header Compression
 - Header Compression Algorithms
 - Commercial Applications
 - General Header Compression Considerations
- **Extension of Header Compression to Backbones and Mobile Ad Hoc Networks**
 - Motivation and Challenges of Applying Traditional HC Algorithms to Backbones
 - Backbone-Applicable Adaptations to Header Compression Techniques
 - Motivation and Application of HC Algorithms to MANETs
 - Considerations with the Application of HC Algorithms to MANETs
- **Summary**

Introduction

- **Internet Protocol version 6 (IPv6) provides extended features compared to IPv4**
 - Extended address space
 - Enhanced mobility
- **With the DoD's move to IPv6, migration plans from legacy IPv4 networks to IPv6 networks are beginning to take shape**
- **Implementation of IPv6 introduces concerns related to expanded packet headers**
 - Packet header size doubled from 20 Bytes (IPv4) to at least 40 Bytes (IPv6)
 - Incorporation of network-layer encryption mechanism (i.e., Internet Protocol Security, or IPsec) nearly doubles IP operational overhead
 - Undesirable for many wireless and SATCOM networks, which are already bandwidth constrained
 - IPv6 in the low bandwidth/mobile environment identified as a “transition challenge” by Marilyn Kraus at the December 2003 NAIPv6 Summit
- ***Methods that reduce this expanded overhead will increase user throughput and/or the number of users a network can support***

Benefits of Reducing Header Size

- **Given the need to reduce overhead, simplified calculations can highlight the benefits of header compression**

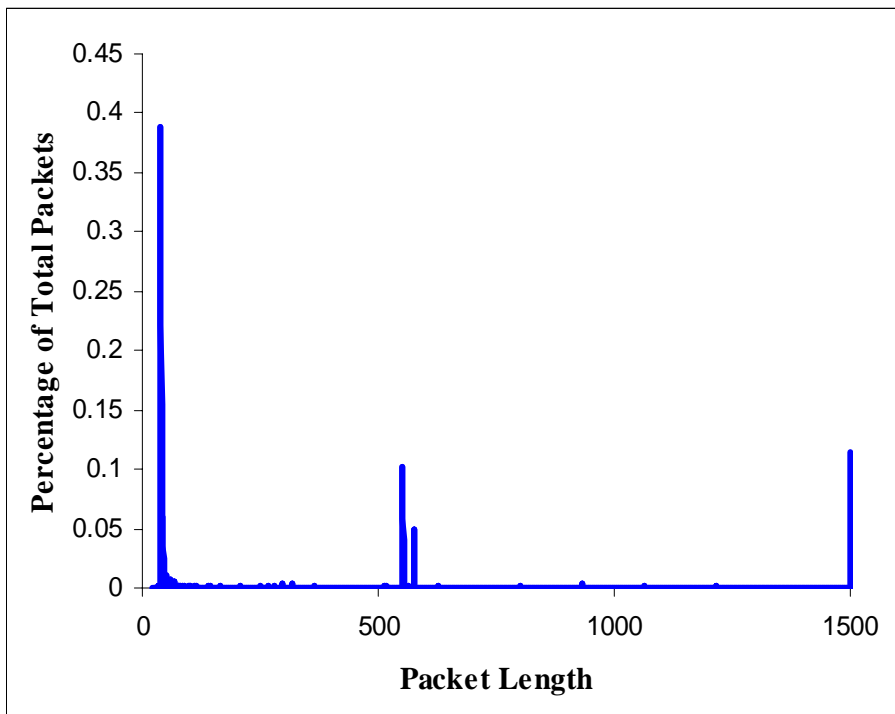


- **Scenario A: *40 Byte IPv6 Header***
 - Constant 20 Byte packet payloads
 - ~2083 packets sent per second transmitted across link
 - Over a 1 second period, ~666 kb transmitted is IPv6 overhead, ~333 kb transmitted is actual user data (i.e., 66% of data transmitted is overhead)
- **Scenario B: *2 Byte Compressed Header***
 - Constant 20 Byte packet payloads
 - ~5680 packets sent per second
 - Over a 1 second period, ~90 kb transmitted is IPv6 overhead, ~910 kb transmitted is actual user data (i.e., 9% of data transmitted is overhead)

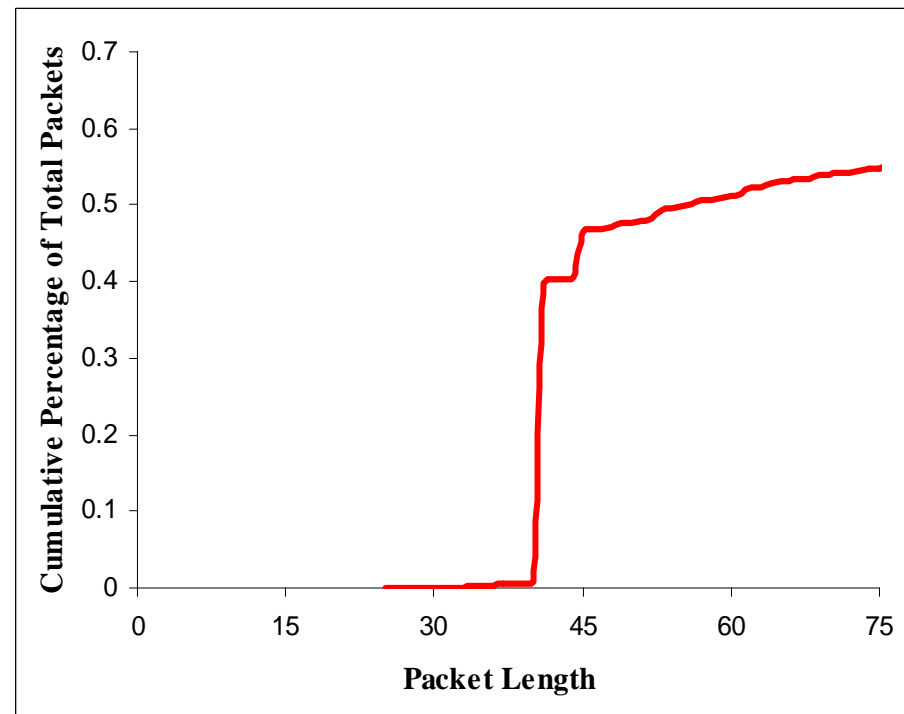
**Under The Best Case Scenario, HC Can Theoretically Decrease Header Overhead
By 95%**

Packet Profiles In the Internet

- Data values below are taken from a 1 minute IPv4 packet trace performed on an OC-3 link on MCI's backbone



(a) Packet length probability distribution function

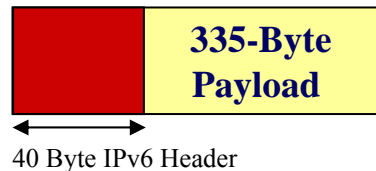


(b) Packet length cumulative distribution function

- As shown, 40% of packets that traverse the Internet are 40 bytes in length
- In IPv6, headers will expand by 20 bytes
 - Significant percentage of the total packet, as a 40 byte header will be used to transport 20 bytes of payload

Numerical Application of Simplified Header Compression to Internet Traffic

- **Distribution of packet lengths from Internet adjusted according to IPv6**
 - Average packet length in previous slide calculated to be ~355 bytes (Header and Payload)
 - Average packet length increases by 20 bytes for IPv6 header



- **Assuming that no header compression is applied, IP header overhead is calculated as follows:**

- Overhead =
$$\frac{\text{IP Header Bytes}}{\text{Total Bytes Transmitted}} = \frac{40 \text{ Bytes}}{375 \text{ Bytes}} = 10.67 \% \text{ overhead}$$

- **Assuming header compression algorithm is applied to the ESP/IP header, overhead calculations can be made:**

- Assuming header size is reduced to 2 bytes per packet
- % Overhead =
$$\frac{\text{Compressed Header Bytes}}{\text{Total Bytes Transmitted}} = \frac{2 \text{ Bytes}}{337 \text{ Bytes}} = 0.59 \% \text{ overhead}$$

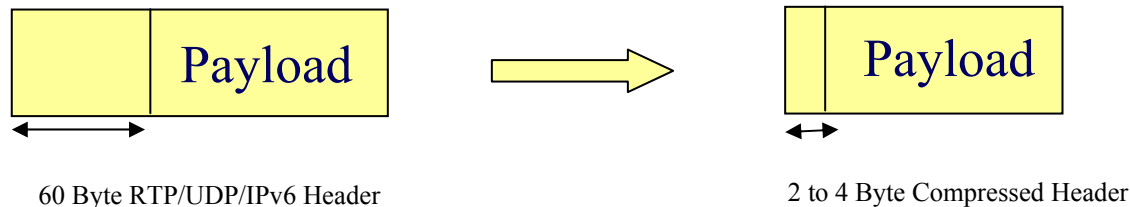
Compression of IP Packet Headers Alone Provides ~10% Increase in System Throughput

Outline

- **Introduction**
- **Background on IETF Hop-by-Hop Header Compression Mechanisms**
 - Fundamentals of Header Compression
 - Header Compression Algorithms
 - Commercial Application
 - General Header Compression Considerations
- **Extension of Header Compression to Backbones and Mobile Ad Hoc Networks**
 - Motivation and Challenges of Applying Traditional HC Algorithms to Backbones
 - Backbone-Applicable Adaptations to Header Compression Techniques
 - Motivation and Application of HC Algorithms to MANETs
 - Considerations with the Application of HC Algorithms to MANETs
- **Summary**

Background

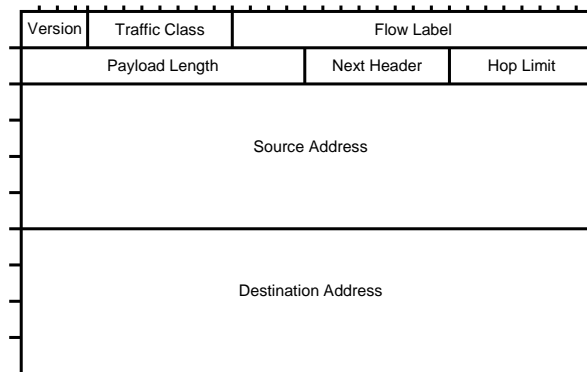
- **Citing the need to reduce the size of IP headers, the IETF has led the development of HC algorithms**
- **In general, HC solutions provide reduction in the operational overhead required by protocols**
 - Compression applied to Layer 3 (IP) and several Layer 4 protocol headers
 - For example, RTP/UDP/IPv6 headers can be compressed from 60 bytes to 2-4 bytes



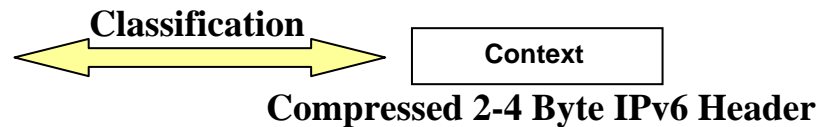
- **HC solutions can reduce the additional overhead introduced by network-layer encryption mechanisms (e.g., IPsec)**
 - Compression algorithms instantiated on encryption/decryption devices have the ability to:
 - Compress inner headers before encryption
 - Compress outer ESP/IP headers after encryption

Header Compression Fundamentals

- Compression is applied over a link between a source node (i.e., compressor), and a destination node (i.e., decompressor)
- HC algorithms exploit protocol inter-packet header field redundancies to improve overall efficiency



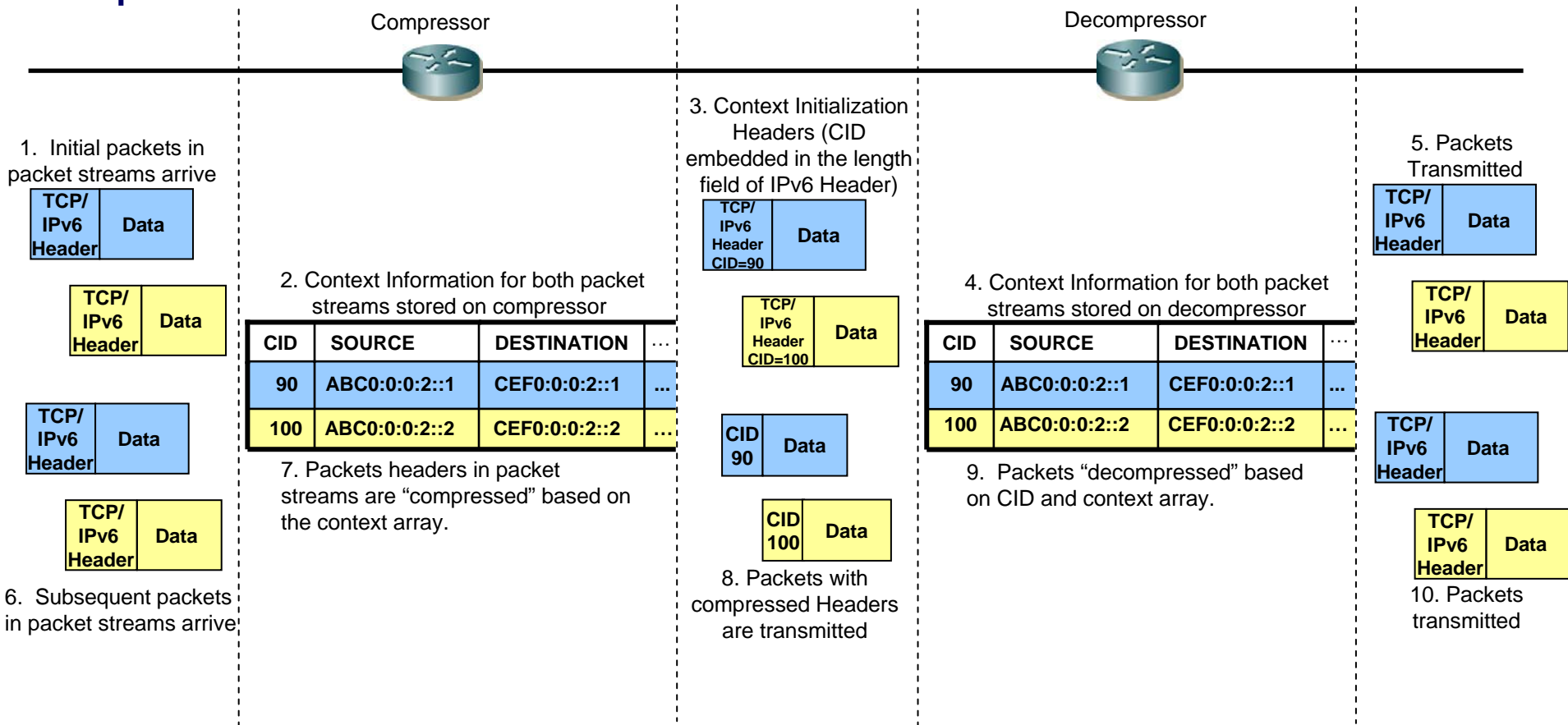
Original 40-Byte IPv6 Header



- Both compressor and decompressor store header fields of each packet stream, and associate each stream with a context identifier (CID)
- Upon reception of a packet with an associated context, the compressor removes the IPv6 header fields from packet header and appends a CID
- Upon reception of a packet with a CID, the decompressor inserts IPv6 header fields back into packet header and transmits packet

Header Compression Operation

- Diagram below illustrates initialization and operation of header compression protocols



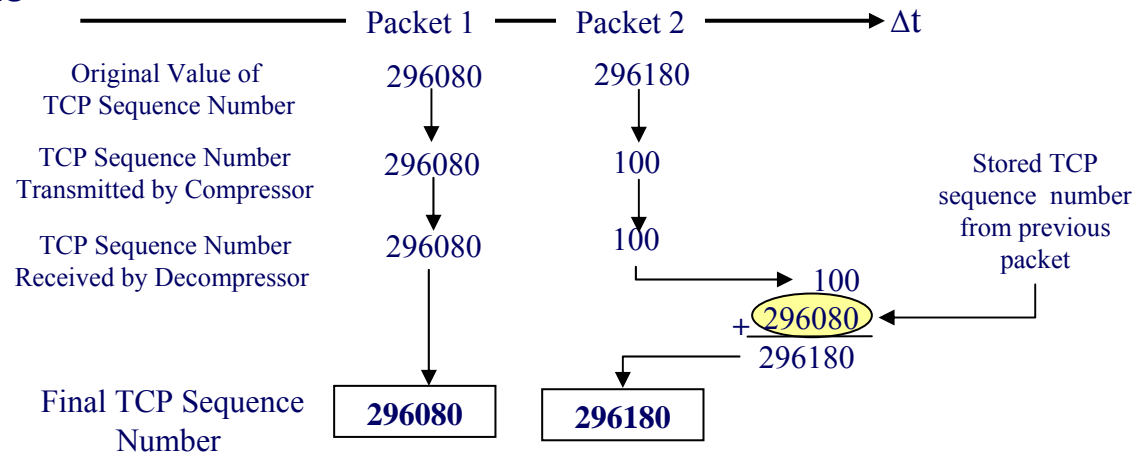
Reduction In The Number Of Full Headers Transmitted Can Result In Overall Decreased Overhead

Header Compression Algorithms

- **Two compression protocols emerged from the IETF**
- **Internet Protocol Header Compression (IPHC)**
 - HC scheme designed for low bit error rate links
 - Compression profiles defined in [RFC 2507] and [RFC 2508]
 - Compression of TCP/IP, UDP/IP, RTP/UDP/IP, and ESP/IP headers
 - “Enhanced” Compression of RTP/UDP/IP (ECRTP) headers defined in [RFC 3545]
- **ROHC Working Group: Robust Header Compression (ROHC)**
 - HC scheme designed for wireless links
 - ***Provides greater compression compared to IPHC at the cost of greater implementation complexity***
 - More suitable for high BER, long RTT links
 - Compression profiles defined in [RFC 3095] and [RFC 3096]
 - Compression of ESP/IP, UDP/IP, RTP/UDP/IP headers
 - Profile for compression of TCP/IP headers currently in IETF internet draft stages

Overview of IPHC

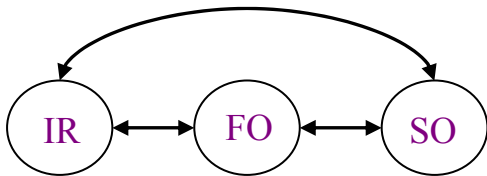
- **Fundamental basics of IPHC algorithm detailed in the previous slide**
- **Mechanisms that facilitate compression of dynamic header fields exist (e.g., TCP sequence numbers)**
 - Delta-based differential encoding mechanism provides compression of incremental header fields



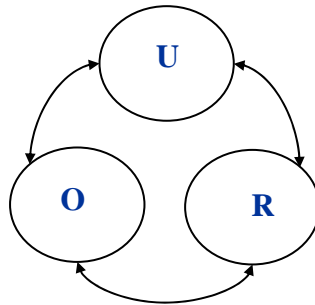
- However, if sequential packets are lost, compressor-decompressor contexts will desynchronize
- Resynchronization depends on how quickly the compressor notices desynchronization
 - For example, by peeking into the TCP sequence number, and noticing a retransmission
- Compressor reestablishes context by transmitting a full header
- **IPHC algorithm performs poorly over high BER, long RTT links [Degermark et. Al]**

Overview of ROHC

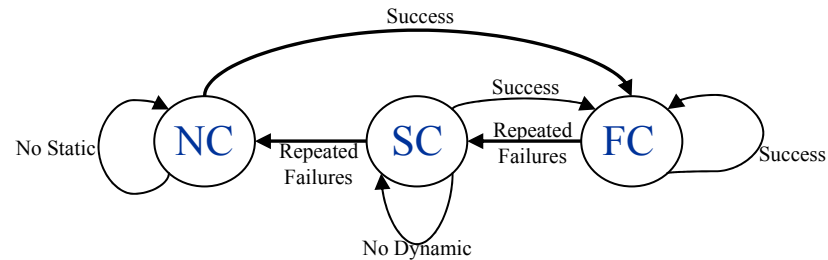
- **Fundamental basics of ROHC detailed in the previous slide**
- **ROHC provides improved header compression over IPHC in high bit error rate (BER), and high round-trip time wireless links [5]**
 - Benefits come at a cost, as ROHC implementation is significantly more complex than IPHC



Compressor States



Modes of Operation

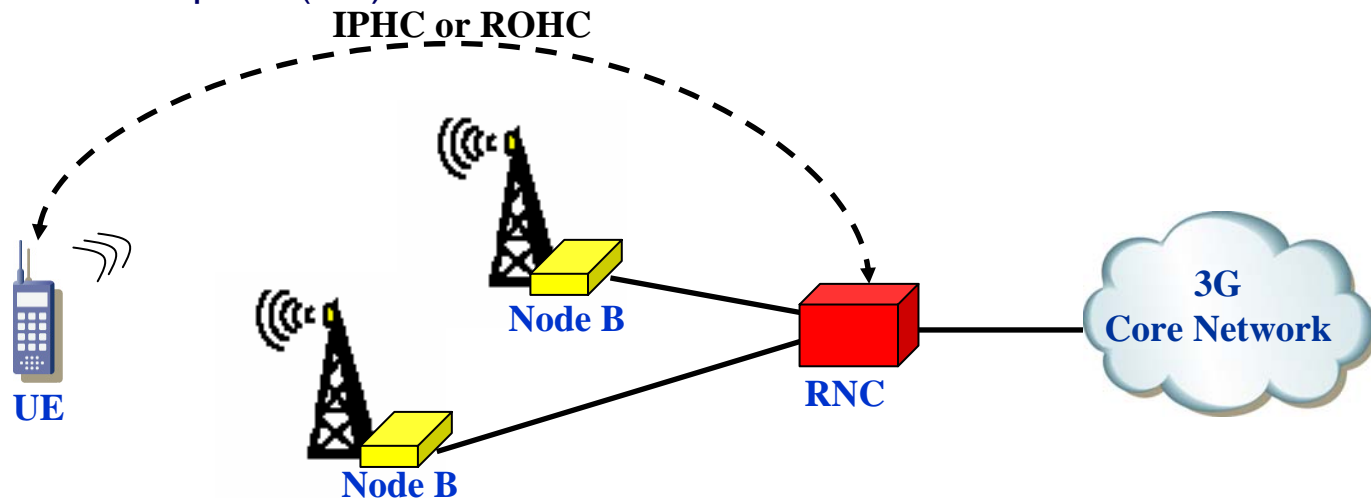


Decompressor States

- Details of state machines (e.g., logic at each state, state transitions, etc.) are left to [RFC 3095]
- ROHC incorporates enhanced mechanisms compared to IPHC
 - Implementation flexibility—capability to compress headers with or without a feedback mechanism
 - Improved compression ratios
 - Equipped with the ability to compress, for example, TCP SYN and FIN messages and timestamps
 - Optimized encoding schemes (e.g., Window-based Least Significant Bit (W-LSB))
- **Provides greater compression compared to IPHC at the cost of greater implementation complexity**

Commercial Applications of Header Compression

- **Application of HC in commercial networks is rare**
- **Cisco Systems router Internetwork Operating System (IOS) provides IPHC implementation**
- **Both IPHC and ROHC are Specified in Release 4 and Release 5 of the 3rd Generation Partnership Project (3GPP)**
 - Function of the Packet Data Convergence Protocol (PDCP)
 - PDCP specification applies HC between the Radio Network Controller (RNC) and the User Endpoint (UE)



- Version 6.0 specifies context relocation during a Service Radio Network Subsystem (SRNS) relocation, between two RNCs (only for ROHC)

General Header Compression Considerations

- **Incorporation of HC between two nodes places restrictions on underlying link layer**
 - IPv6 payload length must be inferred from the underlying link layer
 - Decompressor must receive the packets in the same order that the compressor sends them
 - Packets are not duplicated by the link layer between the compressor and decompressor
 - RFC 3241, RFC 3544 defines PPP extensions for realizing ROHC and IPHC over PPP, respectively
- **HC requires extra resources on nodes that instantiate the compression algorithms**
 - Additional memory required on nodes for storage of context information
 - Additional processing required on nodes for compression and decompression of packets

Outline

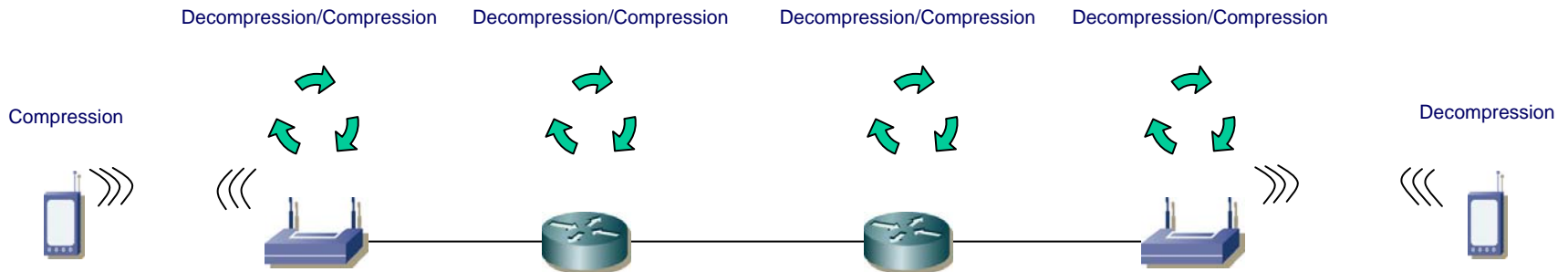
- **Introduction**
- **Background on IETF Hop-by-Hop Header Compression Mechanisms**
 - Fundamentals of Header Compression
 - Header Compression Algorithms
 - Commercial Applications
 - General Header Compression Considerations
- **Extension of Header Compression to Backbones and Mobile Ad Hoc Networks**
 - Motivation and Challenges of Applying Traditional HC Algorithms to Backbones
 - Backbone-Applicable Adaptations to Header Compression Techniques
 - Motivation, and Application of HC Algorithms to MANETs
 - Considerations with the Application of HC Algorithms to MANETs
- **Summary**

Motivation for Header Compression in Network Backbones

- **The amount of traffic that flows through a commercial backbone network provides motivation for HC in Wide Area Networks (WAN)**
- **With the increased utilization of voice and video over IP, a significant amount of backbone resources will be devoted to packet overhead**
 - On a VoIP WAN, 300 million calls per day could consume on the order of about 20-40 Gbps for headers alone [Ash et. Al]
- **Overprovisioning provides an expensive solution for increasing transport network bandwidth**
 - Requires a core transport network bandwidth upgrade
 - Translates to increased end-user cost
- **In most hybrid communication networks, overprovisioning is not an option**

Challenges Associated With Implementing HC in IP Backbones

- **Scalability issues in core network nodes**
 - Large number of contexts hampers performance of core network nodes
 - Significant memory requirements to store contexts
 - Significant processing requirements to compress/decompress packets
- **E2E packet path may include several compression-decompression cycles**
 - Extra compression/decompression cycles in core network may increase packet delay



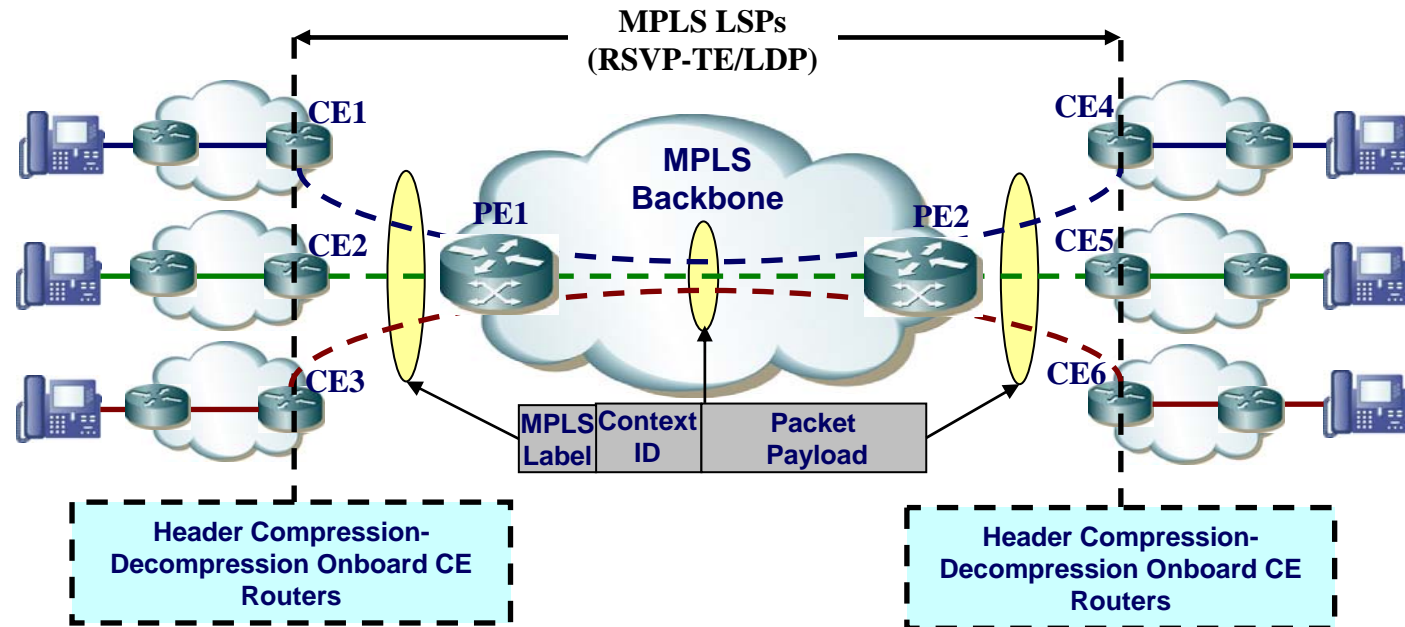
Recent Internet-Drafts In The IETF Attempt To Modify Standard, Per-Hop, HC Techniques To Solve These Challenges

IETF Requirements for HC over Multiple Hop Paths in MPLS Backbones

- **Recent drafts in the IETF are defining requirements to adapt header compression schemes over multiple hop paths**
- **<draft-ietf-avt-hc-mpls-reqs-00.txt>, April 2004**
 - Authored by J. Ash (AT&T), B. Goode (AT&T), J. Hand (AT&T) and R. Zhang (Infonet)
 - Describes motivation for header compression over multiple-hop paths
 - A multiple-hop header compression scheme would eliminate the drawbacks associated with multiple compression-decompression cycles in core networks
 - Goal is to resolve core network implementation issues associated with header compression
- **[Ash et. Al] requires that multi-hop header compression techniques must:**
 - Be independent of existing compression algorithms (e.g., IPHC, ROHC, ECRTP)
 - Allow HC over an MPLS LSP, thereby avoiding hop-by-hop compression/decompression cycles
 - Allow use of standard protocols to signal context identification and control information (e.g., [RSVP], [RSVP-TE], [LDP])
 - Minimize incremental performance degradation due to increased delay, packet loss and jitter introduced by HC

[Ash. et Al]: Header Compression Over Multiple Hops in MPLS Backbones

- **Diagram below illustrates the envisioned application of HC at customer edge devices, through MPLS label switching between CE devices**



- RSVP-TE or LDP used to establish LSPs between CE routers with (de) compression being performed at the CE routers
- Packet with compressed header is switched from CE1 to CE2 according to LSP
- **Leverages MPLS LSPs to navigate packet with compressed header through MPLS backbone, alleviating core network from (de)compression processing**
- **Internet-Draft <draft-ash-avt-ecrtp-over-mpls-protocol-01.txt>, May 2004**
 - Defines protocol extensions for ECRTP over MPLS LSPs using RSVP-TE between CE routers

Challenges of Backbone-Applicable Multi-Hop Header Compression

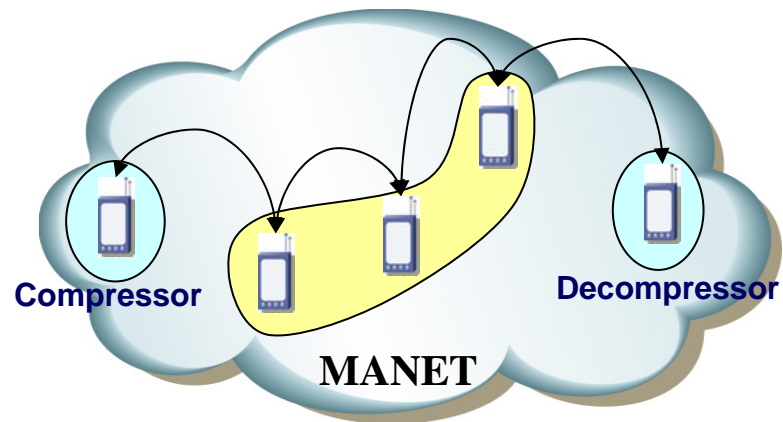
- **[Ash et al.] mentions several issues that need to be resolved to achieve HC over multiple hops**
 - Protocol extensions are required
 - Header compression algorithms (e.g., ROHC, IPHC)
 - Extensions to MPLS signaling to negotiate:
 - HC algorithm used
 - HC protocol parameters
 - Session context ID (SCID) space
 - Resynchronization of compression algorithms over multiple hops, as HC algorithms are designed for application on a hop-by-hop basis
 - Scalability of extending MPLS to CE networks, in terms of LSP signaling mechanisms (i.e., LDP, RSVP-TE, etc.)
 - Packet reordering issues which increase probability of context desynchronization

Header Compression and MANETs

- **Mobile Ad Hoc Networking Goal [RFC 2501]**
 - “... to extend mobility into the realm of autonomous, mobile, wireless domains where a set of nodes ... form the network routing infrastructure in an ad-hoc fashion.”
- **MANETs are multi-hop networks that are self-organizing, rapidly-changing, and dynamic**
 - Do not require the support of a fixed infrastructure
 - Eliminates the need for home agents
- **MANET node resource constraints include**
 - Bandwidth
 - Processing
 - Power
- **Use of header compression algorithms can increase throughput in bandwidth-constrained ad-hoc networks**
 - *Reduction in the number of transmitted bits reduces the probability that a packet will be dropped due to a bit error*

Application of HC to MANETs

- **Hop by hop header compression applicable on point to point ad-hoc links**
- **Multiple-hop header compression schemes can be utilized to achieve end-to-end MANET bandwidth efficiency**
 - Reduces the overhead transmitted through intermediate node(s)



- **Suggest applying ROHC, which is designed for high BER links**
- **Operational Mode for ROHC in MANETs**
 - Unidirectional mode (i.e., no feedback) provides a simple solution to achieve HC in MANETs – no new logic needed to be added to the states
 - Bi-directional modes (i.e., feedback) will offer a greater compression, but will introduce signaling into the MANET between compressor/decompressor
 - May require modification to state logic specified in RFC 3095

Considerations with the Application of HC Algorithms to MANETs

- **Hop by hop HC schemes**
 - Undesirable to apply hop-by hop HC schemes to achieve end-to-end HC, as it requires compression/decompression cycles on intermediate nodes
- **Multiple-hop HC schemes**
 - MPLS-based multiple-hop HC scheme unfeasible
 - Possibly identify a “lighter” way to achieve end to end header compression (?)
 - Intermediate nodes frequently entering and leaving MANET should not be included in end to end compression schemes
 - IPv6 Traffic Class header field issues
 - Intermediate routers need to know the Differential Service Code Point (DSCP) values to enforce Quality of Service
 - Intermediate routers marking of Explicit Congestion Notification (ECN) bits
- **Context management in a mobile network**
 - Volatile nature of ad hoc networks requires more advanced schemes for managing contexts between compressor and decompressor

Outline

- **Introduction**
- **Background on IETF Hop-by-Hop Header Compression Mechanisms**
 - Fundamentals of Header Compression
 - Header Compression Algorithms
 - Commercial Applications
 - General Header Compression Considerations
- **Extension of Header Compression to Backbones and Mobile Ad Hoc Networks**
 - Motivation and Challenges of Applying Traditional HC Algorithms to Backbones
 - Backbone-Applicable Adaptations to Header Compression Techniques
 - Motivation, and Application of HC Algorithms to MANETs
 - Considerations with the Application of HC Algorithms to MANETs
- **Summary**

Summary

- **Traditional Header compression techniques provide savings in protocol operational overhead**
 - Applied to a link, on a per-hop basis
- **Application of hop-by-hop header compression techniques to backbones is rare**
 - To achieve compression through Provider network, multiple compression-decompression cycles required
 - Scalability and resource issues on core network nodes
- **New developments in the IETF provide framework for applying header compression over multiple-hops to backbone networks**
 - Internet-Draft applies header compression techniques to MPLS backbones
 - Idea of multiple-hop HC can be applied to MANETs to increase efficiency over bandwidth-constrained links
 - In implementing header compression in a MANET, a trade needs to be made between computational processing, power requirements, and bandwidth savings
- **HC algorithms can reduce the impact of expanded IPv6 packet headers**

Thank you! Questions?

References

- **WAN Packet Size Distribution. "Contribution of Packet Sizes to Packet and Byte Volumes". 18 May 2004.**
<http://www.nlanr.net/NA/Learn/packetsizes.html>
- **Degermark, M., Nordgren, B. and S. Pink, "IP Header Compression", RFC 2507, February 1999.**
- **Casner, S. and V. Jacobson, "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links", RFC 2508, February 1999.**
- **Degermark, M., Hannu, H., Jonsson, L-E. and K. Svanbro, "Evaluation of CRTP Performance over Cellular Radio Networks", IEEE Personal Communications Magazine, Volume 7, number 4, pp. 20-25, August 2000.**
- **Bormann, C., Editor, "Robust Header Compression (ROHC)", RFC 3095, June 2001.**
- **3rd Generation Partnership Project, "Packet Data Convergence Protocol (PDCP) Specification", 3GPP, Tech.Rep., 2002.**
- **M. Egan, S. Casner, C. Bormann, and T. Koren, "IP Header Compression over PPP", RFC 3544, July 2003.**
- **Bormann, C., "Robust Header Compression (ROHC) over PPP", RFC 3241, April 2002.**
- **Ash, Jerry et. al., "Requirements for Header Compression Over MPLS", Internet-Draft <draft-ietf-avt-hc-mpls-reqs-00.txt>, April 2004.**
- **Ash, Jerry et. al., "Protocol Extensions for ECRTTP Over MPLS", Internet-Draft <draft-ash-avt-ecrtp-over-mpls-reqs-01.txt>, January 2004.**
- **Corson, S., and Macker, J., "Mobile Ad Hoc Networking: Routing Protocol Performance Issues and Evaluation Considerations", RFC 2501, January 1999.**
- **Networking components in pictures taken from Cisco Systems Packet Icon® Library.**