Concept and Implementation of 6LoWPAN IPHC Stateful-Compression inside GNU/Linux

Alexander Aring Oder Aring\textsuperscript{1}

Introduction

6LoWPAN (IPv6 over Low-Power Personal Area Network) describes an adaptation from a specific Link-Layer to the IPv6 protocol\textsuperscript{[3]}. The IPHC (IP Header Compression)\textsuperscript{[4]} is specified as an encapsulating header for LoWPAN\textsuperscript{[5]}. This encapsulating Header specifies a compression format for the IPv6 header. To compress IPv6 addresses the IPHC format defines stateless and stateful compression methods. Stateful compression method requires pre-configured context information to restore compressed information from the IPHC header. RFC 6775\textsuperscript{[7]} describes one way to distribute the necessary stateful IPv6 information over a Router Advertisement message (RA message)\textsuperscript{[6]}. These context information will be placed as Options Fields inside the RA message. Such Option Field follows the format of 6LoWPAN Context Option (6CO)\textsuperscript{[7]} and contains necessary information for one IPHC context. This poster represents the work to support IPHC stateful compression and shows an example to handle RA messages inside a GNU/Linux environment.

Description

The IPHC describes stateful compression for general IPv6 addresses. Additionally there is a special stateful compression for "Unicast-Prefix-based IPv6 Multicast Addresses"\textsuperscript{[2]}. The main focus of this work is to support general IPv6 address stateful compression, because it's the common case when using stateful compression. According to RFC 6282\textsuperscript{[4]} the context information stores an IPv6 Prefix and is identifiable by a CID (Context Identifier) value. A CID value has a range from 0 to 15 and is part of the IPHC header to indicate the context information which was used.

\begin{itemize}
\item 00: The UNSPECIFIED address, c.\textsuperscript{[3]}, (SAC). Reserved (OAC)
\item 01: The address is derived using context information and the 64 bits carried in-line.
\item 10: The address is derived using context information and the 16 bits carried in-line. The 16-bit to IID mapping given by 0000:id00:0000:0000:0000:0000:0000:0000, where XXXX are the 16 bits carried inline.
\item 11: The address is fixed and is derived using context information and the encapsulating header (e.g., 802.15.4 or IPv6 source address).
\end{itemize}

Figure 1: DAM/SAM compression values for stateful compression\textsuperscript{[4]}

This work shows a possible solution to deal with stateful compression with 6CO inside a GNU/Linux environment. There following tasks stay open for future work:

\begin{itemize}
\item Safety context distribution
\item Future Work
\end{itemize}

Concept

The stateful compression algorithm\textsuperscript{[4]} will be handled by kernelspace only. A context table will store all necessary information for a context which can be accessed while parsing and creating of IPHC header. While creating the IPHC header, there exists a "lookup" functionality to search the best fitting context information for finding the best stateful IPv6 address compression method. For supporting the context distribution with 6CO, a userspace daemon which handles RA messages will be used. This userspace daemon will create and parse 6CO option fields with handles the right mechanism to filling the in-kernel context table.

Figure 3: 6LoWPAN Context Option Format\textsuperscript{[7]}

Figure 3 shows the 6CO context format according\textsuperscript{[7]}. The following table will explain the important fields of the 6CO:

\begin{itemize}
\item Context Prefix: IPv6 address prefix length. Needs to be a range from 0 to 128.
\item Compression Flag: Set if the context compression can be used for transmitting IPHC headers, otherwise the context information is used for receive handling only.
\item CID: The context identifier. Needs to be a range from 0 to 15.
\item Valid Lifetime: Lifetime for a prefix; if it runs out then the context information isn't valid anymore.
\end{itemize}

Figure 4: Linux/GNU RA 6CO Concept

Figure 4 shows the general concept to handle context information. The "Router Advertisement Daemon" will processing and transmitting RA messages and will manage the in-kernel context table which is necessary for IPHC handling. The context table will store necessary information which is needed for 6LoWPAN IPHC kernel handling only. Lifetime handling need to be handled by userspace software. The compression flag is needed in-kernel to activate or deactivate context handling for transmitting.

References

\begin{itemize}
\end{itemize}