April 2022

MINIMAL ENCRYPTION TO ACHIEVE CONFIDENTIALITY AND POWER SAVING

Thomas Vegas

Anirban Karmakar

MINIMAL ENCRYPTION TO ACHIEVE CONFIDENTIALITY AND POWER SAVING

AUTHORS:
Thomas Vegas
Anirban Karmakar

ABSTRACT

Techniques described herein leverage compression schemes to provide partial targeted encryption of a stream of data. Key aspects of the transferred data are still protected, and low end devices benefit from less power consumption.

DETAILED DESCRIPTION

Internet of Things (IoT) devices are very low powered. They are intended to run for months on their battery, and periodically stream gathered data to a remote entity. The data often has to be sent authenticated and encrypted, which can constrain the IoT power budget. This is due in large part to the requirement of encrypting the entire payload in an encryption tunnel.

Accordingly, techniques are described herein for partially encrypting a data stream.

Differential Representation

In practice there may be a great deal of redundancy; as a result, the data may be sent in a differential format. For instance, a compression scheme may send data either as:

- Plain-text data:
    - length
    - actual data
- Compressed:
    - offset: in the previous data
    - length: count to take at data

Then, a fixed-sized window which contains the latest, previously decompressed data. The compressed data can be referred to using length+offset.

Sparse Encryption

Consider the sending of a stream of data (e.g., a very short stream of data). In accordance with techniques described herein, only a few key data elements to be sent may be encrypted. This may render full decoding of the received stream impossible. An example method is provided as follows:

1. Use a longer window to make sure that the sent flow can be better compressed. For example, the window may be 512kB instead of the usual 32kB of previously decoded data.
2. Encrypt plain-text data, or send it in a clear format if that portion is not confidential. These elements may be shared in an encrypted format (general), or shared in clear format (in case no confidentiality is needed).
3. Infrequently randomize 16-bit values by (1) obtaining a mapping of real_16bits -> rand_16bits, which is sent encrypted; and (2) performing the randomization by the remote entity and sending the randomized data, encrypted, from the remote entity to the IoT device.
4. Send the compressed data (equals sending one offset value, and one length value) in a clear format. The offset+length pairs may be sent in the clear format but randomized using the mapping table above.
5. In case of highly compressible data, compressed data may be sent fully encrypted.
6. A Media Access Control (MAC) address may also be sent infrequently to ensure the overall stream has not been tampered with.

From an attacker point of view, the reconstructed stream may contain missing/unknown data (e.g., the one that had been sent in an encrypted format).

Benefits

In a stream compression scheme where only new data is actually sent, this data can be sent encrypted, or in clear format if no confidentiality is needed. As a result, only the actual data sent is generally encrypted. Any reference to existing decompressed data is sent shuffled and refers to data that was not accessible by any eavesdropper. For classical textual data, the corresponding ratio may be 1:4, meaning that only 25% of the overall stream is

sent compressed. This may lead to battery savings. For example, using only 25% of the original energy consumption may lead to four times longer battery duration.

These techniques may be general (e.g., not restricted to header compression on a Low-Power Wide Area Network (LPWAN)). The mechanisms described herein may operate on a stream of data (e.g., a short stream of data) to minimize its encryption. Known deduplication/compression techniques may be re-used to enable sparse encryption

If a data chunk is shared on both sides, there may be at least two options: (1) only a data reference is sent, and the lengths/offsets may be shuffled; and (2) the full data may be re-sent in an encrypted form, or in a clear format if that portion is not confidential. If the data chunk is new, the full data may be sent encrypted, or in clear format if that portion is not confidential. From an attacker's point of view, if only the data reference is sent, the data reference might have been first sent as (1) encrypted, in which case the stream is obliterated and is still missing the critical part; or (2) clear-text, which case the stream may be reconstructed.

In certain examples, only critical parts of the data may be encrypted. As a result, an attacker may miss the critical parts of the data. This scheme may cover headers in case of Peer-to-Peer (P2P) communications. Any protocol/stream of data may be partially encrypted using these techniques.

Figure 1 below illustrates a partial encryption mechanism compared to a standard compression mechanism.

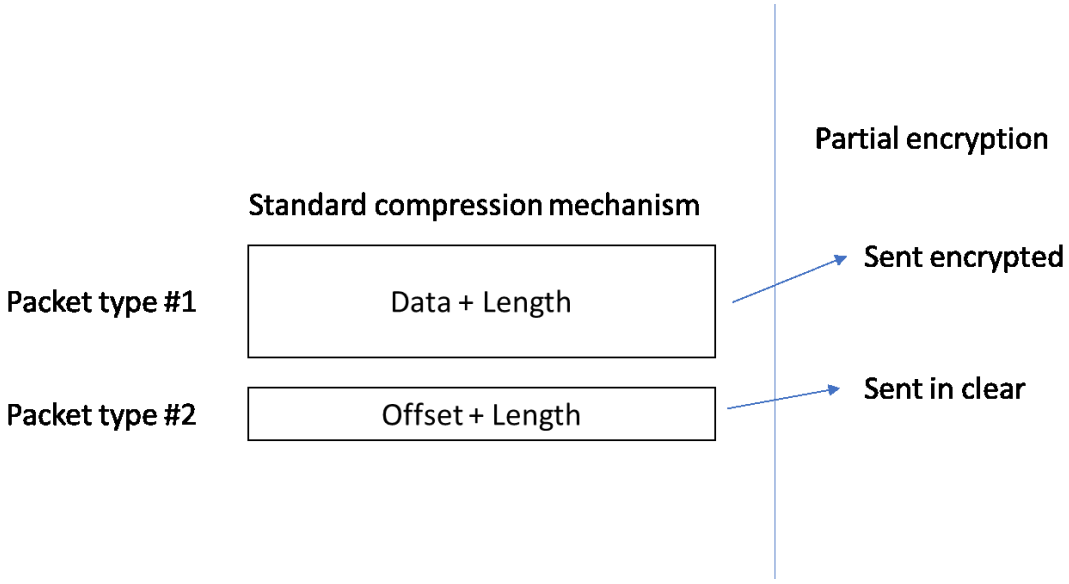

Figure 1

Figure 2 below illustrates an example compression mechanism.

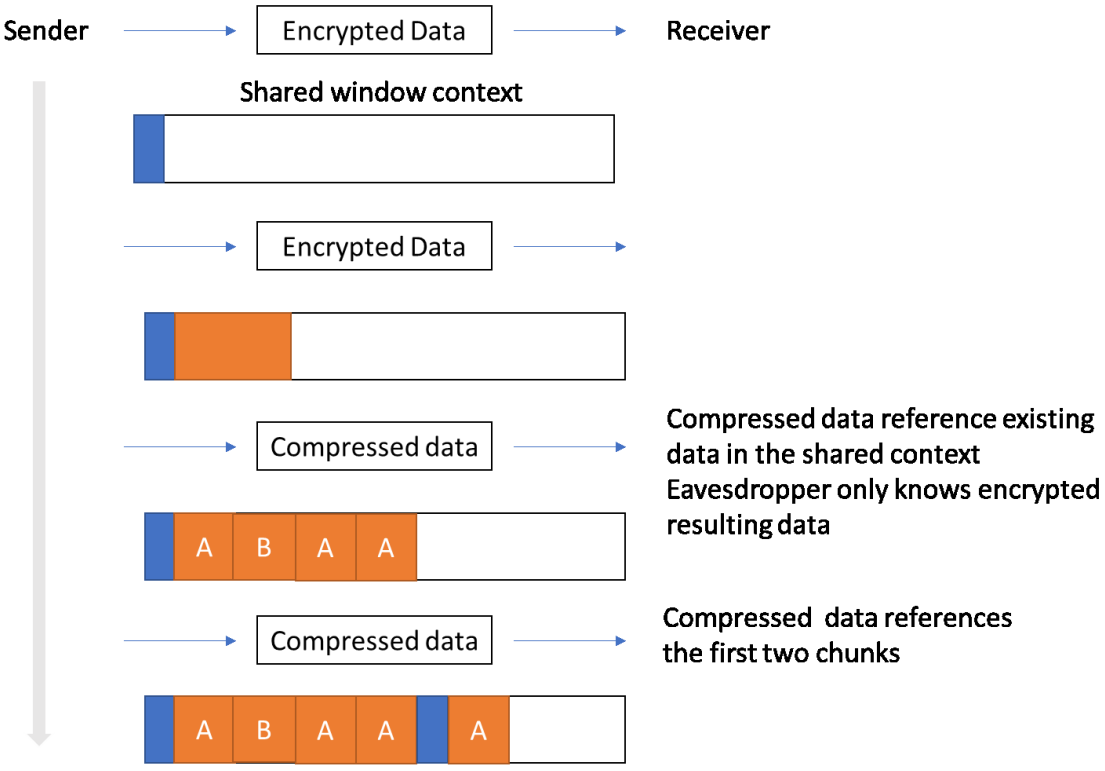

Figure 2

Figure 3 illustrates how the resulting data may be reconstructed.

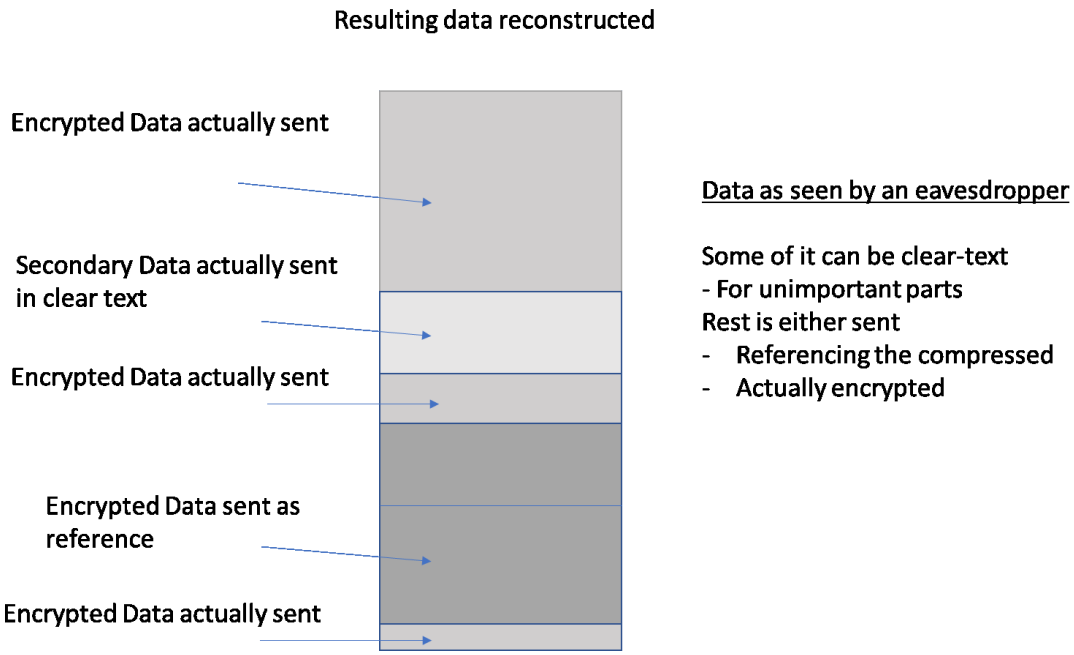

Figure 3

In summary, techniques described herein leverage compression schemes to provide partial targeted encryption of a stream of data. Key aspects of the transferred data are still protected, and low end devices benefit from less power consumption.