Firmware upgrading protocol through the Ethernet

This protocol is used to perform Gateway firmware upgrading through the Ethernet. Here we define:

Host: Mobile Phone, PC, or other remote server.

Client: The Gateway.

The file we used for firmware upgrading is binary file.

1. Data Frame (Host -> Client)



* Header: “[UPGRADE”, 8bytes.
* Type: Frame type, 2bytes.
  + 0x0800 Authenticated frame
  + 0x0801 Data frame
  + 0x0802 Command frame
  + 0x0803 ACK frame
* Sequence: The sequences of the frame, starting from 0 and each frame coming later should be plus 1 based on the former frame sequence value, 2bytes.
* Length: The encrypted data length. The length should always be 1024 except the last frame, 2bytes.
* Encrypted Data: The actual encrypted data of the upgrading file.
* CRC: 32bit CRC data of the frame (Header, Type, Sequence, Length, Encrypted Data), 4bytes. (IEEE 802.3, Polynom 0x04C11DB7).
* Tail: “]”, 1byte.

Note: Using the Network Byte Order for the Type, Sequence, Length and CRC value.

1. Authentication frame (Host -> Client)



* Header: “[UPGRADE”, 8bytes.
* Type: Frame type, 2bytes.
  + 0x0800 Authenticated frame
  + 0x0801 Data frame
  + 0x0802 Command frame
  + 0x0803 ACK frame
* IV: Initialization vector, 16bytes
* TAG: Authenticated data, 16bytes.
* Tail: “]”, 1byte.

Note: Using the Network Byte Order for the Type value.

1. Command frame (Host -> Client)



* Header: “[UPGRADE”, 8bytes.
* Type: Frame type, 2bytes.
  + 0x0800 Authenticated frame
  + 0x0801 Data frame
  + 0x0802 Command frame
  + 0x0803 ACK frame
* Command
  + 0x55AA0400 Get the version of the current code
  + 0x55AA0401 Get the 32bit CRC value of the code that has just been written into flash

The 32bit CRC of the original unencrypted upgrading file. Host can use this value to check whether the upgrading file is flashed correctly or not.

* + 0x55AA0402 Switch to the new firmware
* Tail: “]”, 1byte.

Note: Using the Network Byte Order for the command value.

1. ACK frame



* Header: “[UPGRADE”, 8bytes.
* Type: Frame type, 2bytes.
  + 0x0800 Authenticated frame
  + 0x0801 Data frame
  + 0x0802 Verifying frame
  + 0x0803 ACK frame
* ACK, 4bytes
  + ACK of the Authentication frame
    - First 2bytes, 0xffff
    - Last 2bytes contain the error code
      * 0x0, pass
      * 0x1, fail
      * 0x2, packet invalid
      * 0x3, packet type error
  + ACK of the Data frame
    - No error
      * Sequence of the current Data frame, first 2bytes
      * Length of the current Data frame, last 2bytes
    - Error occurs
      * First 2bytes, 0xffff
      * Last 2bytes contain the error code
        + 0x1, not be authenticated
        + 0x2, packet invalid
        + 0x3, packet type error
        + 0x4, sequence error
        + 0x5, length error
        + 0x6, CRC error
        + 0x7, decrypt error
        + 0x8, flash write error
  + ACK of the command frame
    - No error
      * Return the value that the host needs except the “Switch to new firmware” command
      * For the “Switch to new firmware” command
        + Return 0xffff0000 to ACK host
    - Error occurs
      * First 2bytes, 0xffff
      * Last 2bytes contain the error code
        + 0x1, not be authenticated
        + 0x2, packet invalid
        + 0x3, command error
* Tail: “]”, 1byte.

Note: Using the Network Byte Order for the Type and ACK value.

1. Error handling

* Authentication frame
  + Resend the authentication frame
* Data frame
  + Error occurs except the sequence error(0x4) and flash write error(0x8)
    - Resend the last frame with the same sequence number
    - Or resend the authentication frame to start the upgrading process from the beginning.
  + Flash write error(0x08) and sequence error(0x4)
    - Resend the authentication frame to start the upgrading process from the beginning.
* Command frame
  + Resend the command frame or the authentication frame

1. Work flow



1. Example

Supposing we have an upgrading file gateway.bin which size is 0xB61 bytes, and now we want to perform the firmware upgrading.

Step 1: Sending authenticated data (Host -> Client)



ACK from the Client (Client -> Host)



Step2: Sending Data frame

Data frame0 (Host -> Client)



ACK from the Client (Client -> Host)



Data frame1 (Host -> Client)



ACK from the Client (Client -> Host)



Data frame2 (Host -> Client, last frame)



ACK from the Client (Client -> Host)



Step3: Sending Command frame (Host -> Client, Get CRC)



ACK from the Client (Client -> Host)



Step4: Sending Command frame (Host -> Client, EXEC)



ACK from the Client (Client -> Host)



Step5: Firmware upgrading process is finished.