Chapter 1

Look Inside

Introduction

1.1 Overview

GAT defines a communications protocol used, between a master and an EGM, to authenticate software and firmware components within the EGM. Typically, a portable PC or a laptop is used for the role of the master. EGMs and other devices can be used for the role of the EGM.

The GAT communication protocol is simple in order to reduce complexity of design, implementation, testing and usage. Due to the simplicity of this protocol, a standard layered approach is not necessary. Only the physical layer and the application layer command set are specified.

The GAT protocol and associated calculations are to be run on a properly functioning EGM. Any attempt to use GAT while an EGM is in an error state, tilted, or otherwise malfunctioning is beyond the scope of this standard.

The GAT protocol and associated calculations are designed for the purposes of verifying software content on an EGM. Any attempt to use GAT for any other purpose, such as verifying jackpots, game history recall, and so forth, is beyond the scope of this standard.

Chapter 3

Look Inside

Application Command Layer

3.1 Overview

At the application layer, the master sends a *query* to the EGM and waits for the *response* before sending another command. The EGM always responds to a query with a response. As a consequence no more than one query / response may be pending at the master / EGM side at any given time.

The EGM MUST validate the length and CRC, and then it MUST validate the command byte. The EGM SHOULD NOT respond to messages with invalid length, CRC, or command bytes.

The Master MUST validate the length and CRC, and then it MUST validate the command byte. The master SHOULD ignore messages with invalid length, CRC, or command bytes.

The following time-outs will be in effect:

1. The EGM MUST respond within 200ms of receiving a complete message from the master.
2. If the master does not receive a response to a request, the master SHOULD wait at least 225ms before sending another request.
3. The recommended inter-byte timeout value is 5ms.
4. If the EGM has determined that the previously received byte was the last byte of a valid message, or 200ms have elapsed since the previously received byte, the EGM SHOULD treat the next byte received as belonging to a new message.
5. The master MUST wait at least 10ms upon receipt of a response before transmitting again.

3.2 Application Layer Format

3.2.1 Byte Order

The GAT protocol uses Big Endian (most significant byte first) byte ordering for all cases where multi-byte, numeric information is conveyed by the GAT protocol unless another format is specifically stated (typically through the use of the Data Format byte).

3.2.2 Bit Order

For bit-field parameters, bit **0** always refers to the least significant bit. Bit **7** always refers to the most significant bit. The following table may be used to determine bit positions:

Table 3.1 Bit Positions (Sheet 1 of 2)

Bit	Bit Mask	Description
0	0x01	Least significant bit.
1	0x02	2nd bit position.
2	0x04	3rd bit position.
3	0x08	4th bit position.
4	0x10	5th bit position.
5	0x20	6th bit position.

Table 3.1 Bit Positions (Sheet 2 of 2)

Bit	Bit Mask	Description
6	0x40	7th bit position.
7	0x80	Most significant bit.

3.2.3 Transmission Order

The bytes of a message are transmitted from left to right—that is, command byte first and CRC bytes last. The order of the bits within a byte follows the RS-232 specification of LSB (bit **0**) first and MSB last. All bits of a byte are transmitted before the next byte is started.

3.2.4 Data Formats

The following data formats are supported by the GAT protocol:

Binary: Each byte represents a binary value between `0x00` through `0xFF` inclusive.

Packed BCD: Each byte represents a decimal value between 00 and 99 inclusive, represented as binary `0x00` through `0x99`.

HEX-ASCII: A hexadecimal string representation of a binary value. Binary values are converted to uppercase ASCII hexadecimal strings that represent the binary values. An even number of nibbles (hexadecimal digits) MUST be included. Only ASCII characters 0-9 (0x30 through 0x39) and A-F (0x41 through 0x46) MUST be used. For example: the binary value 0x0123456789abcdef (or 0x0123456789ABCDEF) is represented as the string 0123456789ABCDEF and is transmitted as the bytes 0x30, 0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 0x37, 0x38, 0x39, 0x41, 0x42, 0x43, 0x44, 0x45, 0x46. See Section 5.4, Example SHA-1 HMAC Authentication, which documents a transmission that includes a 20-byte binary key value.

ASCII: An ASCII data string. May include control characters such as CR (`0x0D`) and LF (`0x0A`).

XML: A well-formed XML document conforming to XML version 1.0.

XML version 1.0 requires that XML processors MUST support UTF-8 and UTF-16 encodings of an XML document. Thus, implementations of the GAT protocol MUST support UTF-8 and UTF-16 encodings for the XML data type. However, since UTF-8 tends to create smaller document sizes than UTF-16, implementations of this protocol SHOULD use UTF-8 encodings for XML documents. The GAT protocol does not provide a mechanism for selecting the encoding of an XML document. The default encoding is UTF-8.

3.2.5 Application Layer Frame

Table 3.2 Frame Structure

Command	Length	Message Data	CRC
1 byte binary	1 byte binary	0 to 251 bytes (varies)	2 bytes binary

This frame consists of the following fields:

Table 3.3 Frame Field Descriptions

Field	Description
Command	This is a command byte that indicates the message format and its purpose. Transmitted first.
Length	The total number of bytes in frame (including Command, Length, Message Data, and CRC bytes). Note: The maximum message length is restricted to 255 bytes.
Message Data	This field contains any data relevant to the command. The data format depends on the specific command.
CRC	A CRC-16 checksum of the Command, Length, and Message Data fields. Each frame is protected with a 16-bit Cyclic Redundant Check sequence. The CRC uses the industry standard CRC-16 polynomial generator of x^16 + x^15 + x^2 + 1 starting with a seed of 0xFFFF. See Appendix A for further details on correct implementation of this CRC. Transmitted last.

3.3 Commands - Query / Response Pairs

Each query has one corresponding response. The appropriate matched response should be returned by the EGM when a query is received and processed. The command byte for a response is the same as that of the query, except the high bit is set (i.e. `0x02-0x82`).

3.3.1 Status Query (0x01 SQ)

[Master ⇨ EGM] Request the current status information from the EGM.

Table 3.4 0x01 SQ Structure

Cmd = 0x01 SQ	Length = 0x04	CRC
1 byte binary	1 byte binary	2 bytes binary

3.3.2 Status Response (0x81 SR)

[EGM ⇨ Master] Return the current status information.

Table 3.5 0x81 SR Structure

Cmd = 0x81 SR	Length = 0x08	Version ID	Status Data1	Data Format	CRC
1 byte binary	1 byte binary	2 bytes packed BCD	1 byte binary	1 byte binary	2 bytes binary

Table 3.6 0x81 SR Fields

Field	Description
Version ID	Indicates the version of the GAT protocol supported by the EGM. The version is a 4-digit number, where the first byte is 2-digit major revision number and the second byte is 2-digit minor revision number. The errata revision number is not included. For example, `0x03 0x50` indicates GAT version 3.50.0, 3.50.1, 3.50.2, and so on; `0x03 0x51` indicates GAT version 3.51.0, 3.51.1, 3.51.2, and so on; and, `0x04 0x01` indicates GAT version 4.1.0, 4.1.1, 4.1.2, etc.
Status Data1	General Status: Bit **0**: Calculation Status. `0` = Idle. `1` = Calculating. Bit **1**: Last Authentication Results. `0` = Not Available. `1` = Available. Bit **2** & **3**: See Table 3.7 for Current Calculation. Bit **4** to **7**: Reserved. Always set to `0`.
Data Format	Data formats supported: `0x00` = Reserved, do not use. `0x01` = Plain text format. `0x02` – XML format. `0x03` to `0xFF` – Reserved for future use.

Table 3.7 0x81 Status Data1 Field: Bit 2 & 3, Current Calculation

Bit 3 Value	Bit 2 Value	Description
0	0	Requested.
1	0	Calculating.
0	1	Finished.
1	1	Error, cannot complete or failed.

3.3.3 Last Authentication Status Query (0x02 LASQ)

[Master ⇨ EGM] Request the status of the last authentication performed by the EGM. Only the status of the last completed authentication is returned.

Table 3.8 0x02 LASQ Structure

Cmd = 0x02 LASQ	Length = 0x04	CRC
1 byte binary	1 byte binary	2 bytes binary

3.3.4 Last Authentication Status Response (0x82 LASR)

[EGM ⇨ Master] Return the status of the last authentication result calculated by the EGM.

Table 3.9 0x82 LASR Structure

Cmd = 0x82 LASR	Length = 0x09	Authentication Level	Time	CRC
1 byte binary	1 byte binary	1 byte binary	4 bytes binary	2 bytes binary

Table 3.10 0x82 LASR Fields

Field	Description
Authentication Level	Indicates the level or type of authentication that was calculated. A value of `0x01` refers to Level 1 Authentication, `0x02` refers to Level 2 Authentication, and so on. A value of `0x00` indicates no authentication results are available. For this version of the GAT protocol, an EGM MUST support levels 0xBA and 0x00. Other levels MAY be defined in other versions of the GAT protocol and MAY be supported by the EGM.
Time	Time (in seconds) since last results were calculated. If no authentication results are available, then a value of `0x00000000` is returned.

3.3.5 Last Authentication Results Query (0x03 LARQ)

[Master ⇨ EGM] Request the previous/currently available Authentication results.

Table 3.11 0x03 LARQ Structure

Cmd = 0x03 LARQ	Length = 0x07	Data Format	Frame Number	CRC
1 byte binary	1 byte binary	1 byte binary	2 bytes binary	2 bytes binary

Table 3.12 0x03 LARQ Fields

Field	Description
Data Format	The format of the data: `0x00` = Reserved, do not use. `0x01` = Plain text format. `0x02` = XML format. `0x03` to `0xFF` = Reserved for future use.
Frame Number	This number, with the most significant byte first, is used to indicate the Data Frame that should be returned as data in the Last Authentication Results Response (0x83 LARR). The frame number data is indexed from `1`, so a value of `0` is illegal. The range is large enough to handle a file containing up to 65535 frames.

NOTES:

1. It is important to note that this mechanism of accessing the authentication results is linear, not random access. The rule exists in order to reduce any possible load or restrictions on the implementation within the EGM. The implications of this are that for each result, the first frame requested can only be frame 1. After that the master can only request either the *first* frame, frame *n*, or frame *n+1*, where *n* was the previous frame requested. This results in a linear request process, with the ability to reset back to the first frame, or request a retransmit of the current frame, or request that the next frame be transmitted.
2. Prior to reaching the last frame, the master MAY issue another command, such as an SQ or LASQ command. Unless the command nullifies the authentication results, the master MAY resume the LARQ series following the command. The master does not have to restart at frame 1 unless the authentication results have been nullified.

   For example, if the authentication results require 10 frames and the master issues an SQ command after receiving frame 5, the master may resume gathering the authentication results at frame 6. However, if the master issues an IACQ after receiving frame 5, nullifying the previous authentication results, the master must restart at frame 1 once the new authentication results are available.

3.3.6 Last Authentication Results Response (0x83 LARR)

[EGM ⇨ Master] Return a data frame of the previous or currently available Authentication results.

Table 3.13 0x83 LARR Structure

Cmd = 0x83 LARR	Length = 0x07 to 0xFF	Status Data	Frame Number	Data	CRC
1 byte binary	1 byte binary	1 byte binary	2 bytes binary	0 to 248 bytes (varies)	2 bytes binary

Table 3.14 0x83 LARR Fields

Field	Description
Status Data	General Status: Bit **0**:Error Status. `0` = No error. `1` = Error. (Note: Error would usually indicate either no data available, or an invalid frame.) Bit **1**: Frame Status. `0` = Not Last Frame. `1` = Last Frame.
Frame Number	Used to indicate the frame, with the most significant byte first, that is being returned in the Data field. MAY be set to frame 0 (`0x00 0x00`) when an error is being reported (Bit **0** of the Status Data set to `1`).
Data	Contains requested Authentication information (formatted as requested). This response is the mechanism used by the EGM to communicate the result of any special function. See Chapter 4 and Chapter 5 for further discussion of the format for authentication and special function responses.

NOTE:
Authentication Results are not available while an Authentication Calculation is in progress. If a 0x03 LARQ request is received while an Authentication Calculation is in progress, the EGM MUST return an error to the master in the 0x83 LARR response, setting Bit **0** and Bit **1** of the Status Data to `1`.

3.3.7 Initiate Authentication Calculation Query (0x04 IACQ)

[Master ⇨ EGM] Request that the EGM start authentication calculation.

Table 3.15 0x04 IACQ Structure

Cmd = 0x04 IACQ	Length = 0x05 to 0xFF	Authentication Level	Authentication Parameter	CRC
1 byte binary	1 byte binary	1 byte binary	0 to 250 bytes HEX-ASCII	2 bytes binary

Table 3.16 0x04 IACQ Fields

Field	Description
Authentication Level	Indicates the level or type of authentication calculation that should be returned. A value of `0x01` refers to Level 1 Authentication, `0x02` refers to Level 2 Authentication, and so on. A value of `0x00` is illegal. For this version of the GAT protocol, an EGM MUST support level 0xBA. The EGM MUST return error code 0x04 if level 0x00 is requested. Other levels MAY be defined in other versions of the GAT protocol and MAY be supported by the EGM. The special authentication level `0xBA` is used by the master to signal that the Authentication Parameter field contains a special function command. In this case, the Authentication Parameter field MUST have the first byte set to `0x00`. See Chapter 4 and Chapter 5 for further discussion of special functions.
Authentication Parameter	The Authentication Parameter value is used for some Authentication Levels. The same value is used for all modules verified by an Authentication Level. If the value is longer than required by an Authentication Level, it is truncated, the high order bytes discarded. The Authentication Parameter is represented in HEX-ASCII format. If the Authentication Level is set to the special value `0xBA`, the first byte of the Authentication Parameter field MUST be set to `0x00` while the remainder of the field contains the special function. See Chapter 4 for details. The data format is specified with each command.

NOTE:
If an Authentication Calculation is in progress when this command is received by the EGM, the EGM MUST abort the calculation and start the new Authentication Calculation. Issuing a new Authentication Calculation while the EGM is calculating is not recommended. The master can determine the state of the EGM using the 0x01 SQ command.

3.3.8 Initiate Authentication Calculation Response (0x84 IACR)

[EGM ⇨ Master] Indicate that the EGM has received a 0x04 IACQ command. The EGM SHOULD maintain the last 0x04 IACQ result for the master to retrieve for as long as that result is valid, even while the master is disconnected. Whenever a new 0x04 IACQ request is received by the EGM, the EGM MUST overwrite any previous results with the new authentication results. If an error occurred such that the IACQ request did not result in new authentication results, the 0x84 IACR response MUST report the error and the EGM MAY overwrite or otherwise discard the previous authentication results. In addition, the EGM SHOULD discard the last 0x04 IACQ result whenever the EGM is reset or the set of supported calculations changes—for example, due to a change to the set of components on the EGM. If the operator has placed the EGM in a special GAT authentication mode in order to calculate authentication results, the EGM MAY also discard the last result when the operator causes the EGM to exit its GAT authentication mode.

Table 3.17 0x84 IACR Structure

Cmd = 0x84 IACR	Length = 0x05	Status	CRC
1 byte binary	1 byte binary	1 byte binary	2 bytes binary

Table 3.18 0x84 IACR Fields

Field	Description
Status	Bit **0**: ACK/NACK. `0` = Cannot Acknowledge. `1` = Acknowledged. Bit **1**: Calculation Started. `0` = Not started. `1` = Started. Bit **2**: Level Compliance Error. `0` = Valid Level. `1` = Invalid Level requested.