Keywords: digital certificates, SHA-1, mac, challenge, response, monetary, ecash, e-cash, monetary, certificate, certificates, eCertificate, authentication, vending, fare, tollbooths, meters, bill changers, DS1963S

APPLICATION NOTE 151

Maxim Digital Monetary Certificates

Mar 08, 2002

Abstract: This application note describes how monetary value can be represented and stored in iButton® data carriers (tokens). The document defines a digital certificate called a "Maxim Digital Monetary Certificate" (a.k.a., "eCertificate"), and describes the certificate's various components. It suggests use of the eCertificate for electronic monetary transaction systems, such as vending, fare collection, tollbooths, meter reading, bill changer systems, etc.

Introduction

In order that monetary value may be represented in an electronic form and stored in iButton data carriers and similar portable devices (or tokens), a construct is defined herein called a Maxim Digital Monetary Certificate (hereinafter referred to simply as an "eCertificate").

It is important for the reader to know and understand that an eCertificate consists of, and is defined by, not only the data represented in it but also by the specific, unique device in which it is stored, the physical location of the data in the device, and its "instance". This is to say that the same data written into a different device is not the same eCertificate, and that the same data written elsewhere in the same device is not the same eCertificate, and even that the same data written again into the same place in the same device is not the same eCertificate.

Note: The term "transactor" is used within this document to refer to any valid system that has the ability to add or deduct value from an iButton eCash carrier. This generic term shall apply to vending machines, fare boxes, tollbooths, parking meters, bill changers, or whatever device adds or deducts monetary value from an iButton device.

The Dynamic Nature of the eCash Certificate

In the case of physical currency, the value of each note (bill) is fixed, and so notes of varying denominations are exchanged until the appropriate value has been transacted. In other words, a note (bill) with a larger value than is required is offered for a product, and the difference between the actual cost and the amount tendered is returned to the buyer as change.

In eCash schemes, value is exchanged through the destruction and creation of eCertificates with appropriate values for the transaction being performed. When an eCertificate of value "X" is presented by a customer to purchase and item or service of cost "Y", the original eCertificate with value "X" is destroyed and replaced by one with value "X" minus "Y" in the customer token. Likewise, inside the transactor there may exist an eCertificate that represents the total value of the collected funds, "C". This

eCertificate is destroyed and replaced by one that represents "C" + "Y", adding the funds collected from the customer to the till. In the process of a single purchase, two eCertificates have been destroyed and two new ones created, each representing the appropriate changes in value required to pay for the product or service purchased. These destroyed eCertificates cannot ever be recreated, and they can never exist again.

The nature and design of iButton tokens (and the mechanism by which they are written into) is such that the creation of a new eCertificate occurs at the same moment and by the same action as the destruction of the previous one, so the accidental loss or duplication of monetary value is always prevented. (This is referred to as making the transaction atomic.)

The Definition of an eCertificate

The secure electronic monetary token has several features that allow the creation of valid and secure eCertificates. These include the unique device serial number and the unique physical address of data within the device (called the page number). In some devices, the unique instance of the data in a device assured by an integral, inalterable write-cycle counter.

In keeping with the standard memory page size used in the iButton devices, and with the format required for compatibility with popular iButton space allocation schemes, the data record for an eCertificate consists of 32 data bytes. In addition, the eCertificate (but not the data record) includes the offset address where it resides, the serial number of the iButton device, and the associated write-cycle counter value (if applicable).

The eCertificate data also includes a transaction ID used to differentiate similar transactions, codes representing the monetary value of the certificate, and a cryptographic MAC (Message Authentication Code) used to assure the validity of the eCertificate.

The eCertificate 32-byte data record is formed as follows:

Offset	Length	Contents	Defined By
0	1	Length (Constant = 29)	Operating System
1	1	Certificate Type & Algorithm	Credit/Debit Authority
2	20	MAC	Credit/Debit Authority
22	2	Monetary Units Code & Multiplier	ISO4217
24	3	Monetary Balance	Credit/Debit Authority
27	2	Transaction ID	Credit/Debit Authority
29	1	Continuation Pointer	Operating System
30	2	Inverted CRC-16	Operating System

eCertificate Embedded Data Components

The 32-byte data record, which defines the value of an eCertificate, contains fields with data elements defined as follows:

The Length is used by the operating system to specify the length of the useable data area in a record, less the error checking bytes. Since an eCertificate requires every useable byte of the record, this length is fixed at 29 bytes.

The Certificate Type/Algorithm field indicates the certificate type and a pre-defined code that indicates which algorithm was used to generate the MAC.

Valid certificate types defined to date are:

00000=Unsigned Certificate (MAC field contains customer data)

00001=SHA-1 Signed Dynamic Certificate

00002=SHA-1 Signed Static Certificate

The MAC (Message Authentication Code) holds a binary code generated from all the eCertificate elements, and some secret elements. It is used to protect the eCertificate from illicit alteration or duplication.

The Monetary Unit Code is a binary value 10 bits in length that represents the currency represented by the certificate per the ISO-4217 currency standard. The upper order six bits of the field hold a multiplier to be applied to the Monetary Balance field.

The Monetary Balance field is a 24-bit binary value that represents the value that the eCertificate represents (after applying the multiplier from the Monetary Unit Code).

The Transaction ID is a value that is written at each transaction so that similar transactions can be differentiated. (This is important to prevent some types of possible attacks on the system.)

The Continuation Pointer is an operating system defined value used to direct the reader to subsequent records (pages) within the iButton device that are associated with the monetary record. (See Application Note 114 for more information on standard file structures.) It should be noted that any subsequent file data beyond the 32-byte monetary record are not a part of the eCertificate, although the single byte Continuation Pointer itself is a part and shares in the security of the eCertificate.

The Inverted CRC16 is a value that is used by the reader to be sure that the data is correct against routine communication errors. This value is an integral part of the eCertificate definition.

Other eCertificate Data Components

The eCertificate also includes several data items that are not contained within the 32-byte record but are nonetheless integral components of the eCertificate. These are:

The Monetary Secret is a 64-bit value that is not stored with the eCertificate or within the iButton device. This value is known to authorized service provider equipment only. It is the secrecy of this value that protects the value of the eCertificate from illicit alteration.

The Write Cycle Counter is a 32-bit value that increments once each time new data is written to the iButton data page holding the eCertificate. Including this counter as an integral part of the eCertificate makes each instance of it unique and protects against the illicit re-use of previous monetary certificates.

The Page Number is a 6-bit value that represents the physical location of the eCertificate data within an iButton device. Including this value as an integral part of the eCertificate prevents illicit movement or duplication of value within the iButton device. (The upper two bits in the byte allocated for the Page Number are used by the system to differentiate some special conditions. A discussion of their application can be found in the specific token device data sheet.)

The iButton Address (the device serial number) is a 56-byte value that represents the identity of a single, unique iButton device. Making this an integral part of the eCertificate precludes illicit duplication of value among iButton devices.

The Constant String fields are data that are included in the eCertificate but are not carried in the 32-byte data record. This data must be known to all the service provider systems that have the ability to perform eCertificate transactions.

The Secret Counter is a 32-bit counter that is incremented any time a secret is altered in the Maxim DS1963S monetary iButton device. All or part of this counter value may optionally be included in the eCertificate by the service provider.

SHA-1 MAC Generation

The security of the eCertificate is based in the use of a mathematically generated MAC. The process that creates the MAC requires as its input a 512-bit record that contains all the data that fully defines the eCertificate. This record includes an image of the eCertificate data as well as other data pertinent to the location and uniqueness of the eCertificate instance that is being protected. The format of the 64-byte SHA-1 input record is defined as follows:

Offset	Length	Contents	Defined By
0	4	Monetary Secret (Left)	Credit/Debit Authority
4	1	Length (Constant = 01Dh)	Operating System
5	1	Certificate Type & Algorithm	Credit/Debit Authority
6	20	Constant String (Note 1)	Constant
26	2	Monetary Units Code & Multiplier	ISO4217
28	3	Monetary Balance	Credit/Debit Authority
31	2	Transaction ID	Credit/Debit Authority
33	1	Continuation Pointer	Operating System
34	2	Inverted CRC-16	Operating System
36	4	Write-Cycle Counter (Note 2)	Token
40	1	Page Number + X,M bits	Token
41	7	iButton Serial Number (w/o CRC)	Token
48	4	Monetary Secret (Right)	Credit/Debit Authority
52	3	Secret Counter (Note 3)	Credit/Debit Authority
55	1	SHA Pad Byte (= 080H)	SHA Constant
56	8	SHA Message Length (440 bits)	SHA Constant

Notes

1. The shaded area between offset 4 and 35 (inclusive) represents an image of the eCertificate payload data, with the exception that constants have been substituted for the MAC and Secret Counter data. When the MAC has been generated, it will be written into the eCertificate replacing the Constant String bytes.
2. The Write-Cycle Counter does not exist in some tokens that provide protection of the data integrity in hardware. In these devices, 0xFFFFFF is used in this field.
3. The lower order three bytes of the Secret Counter may optionally be included by the service

provider. If the Secret Counter is not included, then these three bytes are assumed to be constant data known by all of the system transactors.

Maxim Electronic Token Authentication

The Maxim iButton tokens contain special hardware features that provide a secure and reliable eCertificate container. These devices are able to internally maintain secrets and to perform the SHA-1 hashing function, and so can provide a MAC for validation of the entire eCertificate. This allows a transactor to verify the authenticity of the iButton device as well as the integrity of all the data received from it before it proceeds to test the monetary value for validity.

Authentication involves a 64-bit Authentication Secret that is known inside the device and is usually generated from the device serial number and a master secret value known only within the service provider system. The Authentication Secret is used in two 32-byte portions referred to as Secret (Left) and Secret (Right).

Authentication of the iButton device also involves a 24-bit Challenge value (randomly generated at the start of each transaction by the transactor) sent to the iButton prior to the authentication of the data in the iButton. This ever-changing value precludes the illicit playback of previous valid authentication responses.

As with the generation of a SHA-1 MAC for monetary validation, a slightly different SHA-1 input record is defined for the authentication of the iButton device and data. The format of the 64-byte SHA-1 input data record for device and data authentication is as follows:

Offset	Length	Contents	Defined By
0	4	Authentication Secret (Left)	Credit/Debit Authority
4	1	Length (Constant = 29)	Operating System
5	1	Certificate Type & Algorithm	Credit/Debit Authority
6	20	MAC	Credit/Debit Authority
26	2	Monetary Units Code & Multiplier	ISO4217
28	3	Monetary Balance	Credit/Debit Authority
31	2	Transaction ID	Credit/Debit Authority
33	1	Continuation Pointer	Operating System
34	2	Inverted CRC-16	Operating System
36	4	Write-Cycle Counter (Note 1)	Token
40	1	Page Number + X,M bits (Note 2)	Token, Authority
41	7	iButton Serial Number (w/o CRC)	Token
48	4	Authentication Secret (Right)	Credit/Debit Authority
52	3	Challenge	Credit/Debit Authority
55	1	SHA Pad Byte (= 080H)	SHA Constant
56	8	SHA Message Length (440 bits)	SHA Constant

Notes

1. In tokens that do not have a Write-Cycle Counter, this field is always 0xFFFFFFFF.
2. In tokens that do not have a Write-Cycle Counter, this field equals the page number (0–15) plus 64. See the token device specification for details.

eCertificates Outside the Token

As defined herein, there does not exist an eCertificate outside of the monetary iButton. The iButton itself, with its unique identity and unique instance of the eCertificate value data (assured by the write-cycle counter or token write protection) is an integral part of the eCertificate itself. It is critical to remember that an eCertificate is not the data that it holds, but is a single instance of that data in a single device and a physical location within that device.

The format of the eCertificate data record may be used as a method to transfer values between systems, but this representation of the eCertificate value does not in and of itself represents a valid eCertificate in any sense. This distinction is an important one. At best, a collection of all the data items involved in an eCertificate is an image of the eCertificate and has no monetary value in and of itself (much like a photocopy of a dollar bill, which represents the bill but has no monetary value).

eCertificates In Non-Secure iButton Tokens

There exist many iButton devices that contain memory (NV RAM or EEPROM) without safeguards such as write cycle counters, SHA hashing facilities, and secure secret storage. Despite this, it is possible to use these devices to hold likenesses of Maxim Digital Monetary Certificates and, indeed, to treat them as if they are legitimate eCertificates. Although this is akin to allowing the use of photocopied currency for purchases, it may be acceptable in some low security or highly controlled applications.

When monetary values are secured using a MAC and the resulting certificate is copied into an insecure token (i.e., a token that cannot be reliably authenticated, and one in which the certificate cannot be associated with any form of instantiation) it is vulnerable to various replay and emulation attacks, but is nonetheless secure against alteration and duplication into other tokens. Because the MAC embedded in the eCertificate includes the token address (serial number), the memory offset and the monetary secret, as well as the other elements in the data record, it remains secure in many respects.

However, without the special mechanisms afforded by the secure iButton devices, a variety of attacks are possible and some are almost trivial. Extreme caution must be exercised when considering any type of monetary system using non-secure tokens.

Related Parts		
DS1961S	1Kb Protected EEPROM iButton with SHA-1 Engine	
DS1963S	SHA iButton	
DS2432	1Kb Protected 1-Wire EEPROM with SHA-1 Engine	Free Samples

More Information
For Technical Support: http://www.maximintegrated.com/support
For Samples: http://www.maximintegrated.com/samples
Other Questions and Comments: http://www.maximintegrated.com/contact

Application Note 151: http://www.maximintegrated.com/an151
APPLICATION NOTE 151, AN151, AN 151, APP151, Appnote151, Appnote 151