ucode Encoding Specifications for QR Code Tags

Number: 930-S304-01.A0.00/UID-CO00025-01.A0.00
Title: ucode Encoding Specifications for QR Code Tags
Status: [X] Working Draft, [ ] Final Draft for Voting, [ ] Standard
Date: 2010/05/192006

Table of Contents

ucode Encoding Specifications for QR Code Tags

Introduction

Scope

This document specifies the data representation format for encoding and storing ucodes in 2D barcodes for the QR Code specifications. This document only specifies the QR Code data representation format but does not specify the security level of digital signatures and the system operation policy.

Position of this Document

It is assumed that the contents specified in this document are used in the following:

- Applications which generate data to be stored in QR Code tags
- Applications which read ucodes encoded in QR Code tags

Reference Documents

[1] T-Engine Forum. Ubiquitous ID Center, "ucode Resolution Gateway," 910-S204/UID-00007, 2008.

[2] RFC 2104 "HMAC: Keyed-Hashing for Message Authentication," 1997.

Definition of Terms

- QR Code

QR Codes are a type of printed tags generally called 2D barcodes, and were developed by DENSO WAVE INCORPORATED in 1994. Currently, the QR Code specifications are open to the public, and DENSO WAVE INCORPORATED declared they would not exercise their patent rights. The QR Code was enacted as Japanese Industrial Standards JIS X 0510 in January 1999 and as International Standards by ISO in June 2000 (ISO/IEC18004).

- ucodeRP Gateway

The ucodeRP Gateway is a server software which implements ucode Resolution Gateway specification [1]. It provides the ucode resolution function and signature verification function to clients through an http based interface.

1. ucode Encoding Method

This chapter specifies the ucode encoding method for storing ucodes in the QR Code 2D barcode.   This document specifies the following two encoding methods.

- Standard format
- Gateway address format

1.1. Notation

Here is the notation used in this document.   The following Table 1 shows a list of the notation types.

Table 1: Notation List

<ucode string> = 32HEXDIG[1] ; ex. 0EFFFEC00000000000000000000050123
<sign string>   = 1*HEXDIG   ; ex. 6455FDB217CFE086953A844DABAC0491B05D91D2
<append string> = <key string> "=" <value string>
<key string>   = 1*HEXDIG
<value string>   = 1*OCTET
<algo type> = <mac type> | " PBEWith" <mac type>
<mac type> = "HmacMD5"|"HmacSHA1"|"HmacSHA256"|"HmacSHA384"|"HmacSHA512"
[1] : n<element> represents that a same element is repeated a specific decimal number of times.

<append string> is cache information and additional information stored in ucodeQR. <append string> consists of the key name and its value.

1.2. Parameters

Table 2 shows a list of parameters which is specified for encoding.

Table 2: Parameter List

Parameter name	Value	Meaning
X-UIDC-UCODE	<ucode string>	ucode (hexadecimal/character notation)
X-UIDC-SIGNATURE	<sign string>	Digital signature (hexadecimal/character notation)
X-UIDC-ALGORITHM	<algo type>	Algorithm type

1.3. Standard Format

The standard format is an encoding format which enumerates parameters

separated by commas.

```
<qr code string> = "X-UIDC-UCODE=" <ucode string> ["," "X-UIDC-SIGNATURE=" <sign

string> ["," "X-UIDC-ALGORITHM=" <algo type> [<append values>]]]

<append values> = "," <append string> [<append values>]
```

Figure 1: Standard Format

1.4. Gateway Address Format

The gateway address format is an encoding format to convert a ucode to a URL to send a ucode resolution request to a ucodeRP Gateway. The use of this format can perform ucode resolution by using a web browser if a ucode-specific application does not exist,

```
<qr code string> = "http://" <rhost> "/" <qpath> "?" <query part>

<rhost> = "(Gateway host name with signature verification function)"

<qpath> = "(Gateway software path with signature verification function)"

<query part> = "X-UIDC-UCODE=" <ucode string> ["&" "X-UIDC-SIGNATURE=" <sign

string> ["&" "X-UIDC-ALGORITHM=" <algo type> [<append values>]]]

<append values> = "&" <append string> [<append strings>]
```

Figure 2: Gateway Address Format

2. Encoding Example

2.1. Standard Format Example

Figure 3 shows an example of encoding in the standard format.

X-UIDC-UCODE=0EFFFEC00000000000000000000050123,X-UID
C-SIGNATURE=6455FDB217CFE086953A844DABAC0491B05D
91D2,X-UIDC-ALGORITHM=HmacSHA1

Figure 3: Example of Encoding in the Standard Format

2.2. Gateway Address Format Example

Figure 4 shows an example of encoding in gateway address format.

http://rs.uidcenter.org/search?X-UIDC-UCODE=0EFFFEC0000000
0000000000000050123&X-UIDC-SIGNATURE=6455FDB217CFE08
6953A844DABAC0491B05D91D2&X-UIDC-ALGORITHM=HmacS
HA1

Figure 4: Example of Encoding in the Gateway Address Format

3. Calculation Method of Signatures

Signatures used in the QR Code 2D barcode are calculated based on HMAC defined below [2].

$$HMAC_K(ucode) = h((K \oplus opad), h((K \oplus ipad), ucode))$$

h indicates the hash function. MD5, SHA-1, SHA-256, SHA-384, and SHA-512 can be used for h [2]. K indicates the private key, and if it is shorter than the block size of the hash function, then zero-padding is performed. $ucode$ indicates the ucode to which the signature is provided. $\oplus$ is the logical operation exclusive or. $ipad$ and $opad$ are hexadecimal numbers, in which 0x36 and 0x5c are repeated up to the block length size respectively. Comma (,) represents a concatenation.

The signature recorded in the QR Code 2D barcode is represented by the HMAC value calculated above as a hexadecimal string.