[MS-RDPEUDP]: Remote Desktop Protocol: UDP Transport Extension

This topic lists the Errata found in [MS-RDPEUDP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS

Atom

Errata below are for Protocol Document Version V9.0 – 2016/07/14.

Errata Published*	**Description**

2016/08/15	In Section 3.1.1.6.1.3, Logarithms and Exponents, clarified the modulo operation and indented the first line of syntax in the pseudo-code examples
	Changed from:
	...
	Pseudo-code example:
	 reduction = 0x1d;
	    m_ffExp2Poly[0] = 0x01;
	    for (i = 1; i < m_fieldSize - 1; i++)
	...
	Where m_fieldSize is 256 for GF(28)
	Logarithms are the inverse of exponents, and can be easily calculated by reversing the previous operation as shown in the following pseudo-code example:
	 m_ffPoly2Exp[0] = 2 * m_fieldSize; // no exponential representation, doesn't exist
	    for (i = 0; i < m_fieldSize - 1; i++)
	...
	Changed to:
	...
	Pseudo-code example:
	    reduction = 0x1d;
	    m_ffExp2Poly[0] = 0x01;
	    for (i = 1; i < m_fieldSize - 1; i++)
	...
	Where m_fieldSize is 256 for GF(28). Note that m_ffExp2Poly is modulo m_fieldSize – 1. In other words, m_ffExp2Poly[n] = m_ffExp2Poly[n + m_fieldSize – 1]. The
	pseudo-code in this document makes the assumption that m_ffExp2Poly is defined for at least m_fieldSize * 2 elements.
	Logarithms are the inverse of exponents, and can be easily calculated by reversing the previous operation as shown in the following pseudo-code example:
	    m_ffPoly2Exp[0] = 2 * m_fieldSize; // no exponential representation, doesn't exist

Errata Published*	Description

```
    for (i = 0; i < m_fieldSize - 1; i++)
...
```

In Section 3.1.1.6.4, Selecting the Coefficients Matrix, replaced the source sequence number 0xf with 0xff in the Matrix coefficient calculation figure and in the pseudo-code.

Changed from:

If the Source sequence numbers (section 3.1.1.2) for packets S1, S2, S3 … Sn are s1, s2, s3 … sn, the coefficient matrix is calculated as follows.

….

Figure 12: Matrix coefficient calculation

...

Only the last byte of the Source sequence number is used in calculating the coefficient. The fecIndex field described in the following pseudo-code example is

equivalent to the uFecIndex field, as specified in section 2.2.2.2. The value of the fecIndex field is updated using the following code prior to every call for

encoding an FEC Packet:

```
 if ((sn&0xf) >= (s1 &0xf) && ((fecIndex >= (s1 &0xf)) && (fecIndex <= (sn&0xf))) ||
      (sn&0xf) < (s1 &0xf) && ((fecIndex >= (s1 &0xf)) || (fecIndex <= (sn&0xf))))
    fecIndex = (sn+1) & 0xf;
```

Pseudo-code example:

```
...
    for (int i=0; i < cLength; i++, ucOrigStart++)
    {
        BYTE e = Div(1, (*pucFecIndex)^ucOrigStart);
        pbCoEfficientArray[i] = (BYTE)m_ffPoly2Exp[e];
    }
...
    for (int i=0; i < cLength; i++, ucOrigStart++)
    {
        BYTE e = Div(1, fecIndex^ucOrigStart);
        pbCoefficientArray[i] = (BYTE)m_ffPoly2Exp[e];
    }
...
```

Changed to:

If the Source sequence numbers (section 3.1.1.2) for packets S1, S2, S3 … Sn are s1, s2, s3 … sn, the coefficient matrix is calculated as follows.

…

Figure 12: Matrix coefficient calculation

...

Only the last byte of the Source sequence number is used in calculating the coefficient. The fecIndex field described in the following pseudo-code example is

equivalent to the uFecIndex field, as specified in section 2.2.2.2. The value of the fecIndex field is updated using the following code prior to every call for

encoding an FEC Packet:

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td> if ((sn & 0xff) >= (s1 & 0xff) && ((fecIndex >= (s1 & 0xff)) && (fecIndex <= (sn & 0xff))) ||
     (sn & 0xff) < (s1 & 0xff) && ((fecIndex >= (s1 & 0xff)) || (fecIndex <= (sn & 0xff))))
     fecIndex = (sn + 1) & 0xff;

Pseudo-code example:
...
     for (int i=0; i < cLength; i++, ucOrigStart++)
     {
         pbCoEfficientArray[i] = (BYTE)Div(1, (*pucFecIndex)^(ucOrigStart & 0xff));
     }
...
         for (int i=0; i < cLength; i++, ucOrigStart++)
     {
         pbCoefficientArray[i] = (BYTE)Div(1, fecIndex^(ucOrigStart & 0xff));
     }
...

In Section 4.2.2.1, Payload of an FEC Packet, updated the FEC Payload table values and the CoEff Array packet value.

Changed from:
The following is an example of an FEC Packet network payload.

Sequence number Size Value
...
FEC Payload   0 66 208 168 239 37 29 238 180 193 24 58 66 252 233 126 172 211 135 31 206 27

The following are FEC encoding internals; these packets are not transferred on the wire:
§ CoEff Array [0 254 230 253 205]
...

Changed to:
The following is an example of an FEC Packet network payload.

Sequence number Size   Value
...
FEC Payload   0 203 146 55 209 198 69 147 95 141 120 66 86 91 174 141 153 99 169

The following are FEC encoding internals; these packets are not transferred on the wire:
§ CoEff Array [1 142 244 71 167]
...</td></tr>
</table>

*Date format: YYYY/MM/DD