[MS-EFSR]: Encrypting File System Remote (EFSRPC) Protocol

This topic lists the Errata found in the MS-EFSR document 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 V23.0 – 2015/10/16.

Errata Published*	Description
2016/04/18	In Section 2.2.2.3, EFSRPC Metadata Version 3, corrected two field names. Changed from: … EncryptedDataOffset (4 bytes): The offset, in bytes, from the beginning of the Preamble field to the EncryptedDataOffset field. … MetaDataOffset (4 bytes): The offset, in bytes, from the beginning of the Preamble field to the EncryptedDataOffset field, formatted as a ULONGLONG (unsigned 64-bit integer as described in section [MS-DTYP] section 2.2.55). … Changed to: … EncryptedDataOffset (4 bytes): The offset, in bytes, from the beginning of the Preamble field to the EncryptedData field. … MetaDataOffset (4 bytes): The offset, in bytes, from the beginning of the Preamble field to the MetaData field, formatted as a ULONGLONG (unsigned 64-bit integer as described in section [MS-DTYP] section 2.2.55). …
2016/01/25	In Section 3.1.4.2.13, Receiving an EfsRpcDuplicateEncryptionInfoFile Message (Opnum 13), clarified how to check whether the objects are of the same type. Changed from: Return Values: The server MUST return 0 if it successfully processes the message received from the client. The server MUST return a nonzero value if processing fails. … If an encrypted object exists with the name specified in the SrcFileName and dwCreationDisposition parameters is equal to CREATE_NEW, then: … ▪ If an object already exists with the name specified in the DestFileName parameter, the server MUST check whether the object referred to by SrcFileName is of the same type; if the object is not of the same type, the server MUST return a nonzero value. In addition, if the object referred to by DestFileName is a container for other objects, and it is not already encrypted, the server MUST return a nonzero value. Otherwise, the server SHOULD overwrite the object, clear its existing attributes, create a new object in its place with the attributes specified, and duplicate the EFSRPC Metadata from the SrcFileName parameter into it.

Errata Published*	Description
	Changed to: Return Values: The server MUST return 0 if it successfully processes the message received from the client. The server MUST return a nonzero value if processing fails. … If an encrypted object exists with the name specified in the SrcFileName and dwCreationDisposition parameters is equal to CREATE_NEW, then: … ▪ If an object already exists with the name specified in the DestFileName parameter, the server MUST check whether the object referred to by SrcFileName is of the same type (either simple object or container for other objects); if the object is not of the same type, the server MUST return a nonzero value. In addition, if the object referred to by DestFileName is a container for other objects, and it is not already encrypted, the server MUST return a nonzero value. Otherwise, the server SHOULD overwrite the object, clear its existing attributes, create a new object in its place with the attributes specified, and duplicate the EFSRPC Metadata from the SrcFileName parameter into it.

* Date format: YYYY/MM/DD