

# MBS Real Studio Dongle Plugin Documentation

Christian Schmitz

May 15, 2012

## 0.1 Introduction

This is the PDF version of the documentation for the Real Studio Plug-in from Monkeybread Software Germany. Plugin part: MBS Real Studio Dongle Plugin

## 0.2 Content

- 1 List of all topics 3
- 2 All items in this plugin 13
- 4 List of all classes 115
- 5 List of all modules 117
- 6 List of all global methods 119

# Chapter 1

## List of Topics

• 2 Dongle	13
– 2.2 class Rockey2MBS	16
* 2.2.1 Close	16
* 2.2.1 Find as Int32	17
* 2.2.1 GenUID(byref uid as UInt32, seed as string, isProtect as boolean)	17
* 2.2.1 Open(mode as Int32, uid as UInt32, byref hid as UInt32)	17
* 2.2.1 Read(BlockIndex as Int32) as string	18
* 2.2.1 Transform(data as string)	18
* 2.2.1 Write(BlockIndex as Int32, data as string)	18
* 2.2.2 Handle as Integer	18
* 2.2.2 Lasterror as Integer	19
* 2.2.3 AUTO_ MODE = 0	19
* 2.2.3 Available = true	19
* 2.2.3 HID_ MODE = -1	19
* 2.2.3 R2_ MINOR = 16	19
* 2.2.3 ROCKEY2_ DISABLE_ WRITE_ PROTECT = false	20
* 2.2.3 ROCKEY2_ ENSABLE_ WRITE_ PROTECT = true	20
* 2.2.3 RY2ERR_ FLUSH_ QUEUE = & hA010000F	20
* 2.2.3 RY2ERR_ FREE_ PREPARSED_ DATA = & hA010000E	20
* 2.2.3 RY2ERR_ GETCAPS = & hA010000D	20
* 2.2.3 RY2ERR_ GET_ ATTRIBUTES = & hA010000B	20
* 2.2.3 RY2ERR_ GET_ PREPARSED_ DATA = & hA010000C	21
* 2.2.3 RY2ERR_ GET_ SERIAL = & hA0100011	21
* 2.2.3 RY2ERR_ NOT_ OPENED_ DEVICE = & hA0100002	21
* 2.2.3 RY2ERR_ NO_ SUCH_ DEVICE = & hA0100001	21

* 2.2.3 RY2ERR_OPEN_DEVICE = & hA0100007	21
* 2.2.3 RY2ERR_READ_REPORT = & hA0100008	22
* 2.2.3 RY2ERR_SETUP_DL_CLASS_DEVS = & hA0100010	22
* 2.2.3 RY2ERR_SETUP_DL_GET_DEVICE_INTERFACE_DETAIL = & hA010000A	22
* 2.2.3 RY2ERR_SUCCESS = 0	22
* 2.2.3 RY2ERR_TOO_LONG_DEVICE_DETAIL = & hA0100012	22
* 2.2.3 RY2ERR_TOO_LONG_SEED = & hA0100005	23
* 2.2.3 RY2ERR_UNKNOWN_DEVICE = & hA0100020	23
* 2.2.3 RY2ERR_UNKNOWN_ERROR = & hA010FFFF	23
* 2.2.3 RY2ERR_VERIFY = & hA0100014	23
* 2.2.3 RY2ERR_WRITE_PROTECT = & hA0100006	23
* 2.2.3 RY2ERR_WRITE_REPORT = & hA0100009	24
* 2.2.3 RY2ERR_WRONG_INDEX = & hA0100004	24
* 2.2.3 RY2ERR_WRONG_REPORT_LENGTH = & hA0100013	24
* 2.2.3 RY2ERR_WRONG_UID = & hA0100003	24
– 2.1 class Rockey4NDMBS	13
* 2.1.1 Rockey(FunctionCode as integer) as integer	13
* 2.1.1 RockeyCall(FunctionCode as integer) as integer	14
* 2.1.2 Buffer as MemoryBlock	14
* 2.1.2 Handle as Integer	15
* 2.1.2 LP1 as Integer	15
* 2.1.2 LP2 as Integer	15
* 2.1.2 P1 as Integer	15
* 2.1.2 P2 as Integer	15
* 2.1.2 P3 as Integer	16
* 2.1.2 P4 as Integer	16
– 2.9 module MatrixDongleMBS	88
* 2.9.1 DongleCount(PortNr as integer) as integer	89
* 2.9.1 DongleDecryptData(UserCode as integer, Data as memoryblock, DongleNr as integer, PortNr as integer) as integer	89
* 2.9.1 DongleEncryptData(UserCode as integer, Data as memoryblock, DongleNr as integer, PortNr as integer) as integer	89
* 2.9.1 DongleExit as integer	89
* 2.9.1 DongleFind as integer	90
* 2.9.1 DongleFindEx(byref LPTNr1 as integer, byref LPTAdr1 as integer, byref DNGCnt1 as integer, byref LPTNr2 as integer, byref LPTAdr2 as integer, byref DNGCnt2 as integer, byref LPTNr3 as integer, byref LPTAdr3 as integer, byref DNGCnt3 as integer) as integer	90
* 2.9.1 DongleGetKeyFlag(UserCode as integer, DongleNr as integer, PortNr as integer) as integer	90
* 2.9.1 DongleMemSize(DongleNr as integer, PortNr as integer) as integer	90
* 2.9.1 DongleModel(DongleNr as integer, PortNr as integer) as integer	90

* 2.9.1 DongleReadData(UserCode as integer, Data as memoryblock, count as integer, DongleNr as integer, PortNr as integer) as integer	91
* 2.9.1 DongleReadDataEx(UserCode as integer, Data as memoryblock, FPos as integer, count as integer, DongleNr as integer, PortNr as integer) as integer	91
* 2.9.1 DongleReadSerNr(UserCode as integer, DongleNr as integer, PortNr as integer) as integer	91
* 2.9.1 DongleSetLedFlag(a as integer, b as integer, c as integer, d as integer) as integer	91
* 2.9.1 DongleVersion(DongleNr as integer, PortNr as integer) as integer	91
* 2.9.1 DongleWriteData(UserCode as integer, Data as memoryblock, count as integer, DongleNr as integer, PortNr as integer) as integer	92
* 2.9.1 DongleWriteDataEx(UserCode as integer, Data as memoryblock, FPos as integer, count as integer, DongleNr as integer, PortNr as integer) as integer	92
* 2.9.1 DongleWriteKey(UserCode as integer, KeyData as memoryblock, DongleNr as integer, PortNr as integer) as integer	92
* 2.9.1 GetConfigMatrixNet(Category as integer) as integer	92
* 2.9.1 GetDriverFlag(UserCode as integer, DongleNr as integer, PortNr as integer) as integer	92
* 2.9.1 GetPortAdr(LptNr as integer) as integer	93
* 2.9.1 GetVersionAPI as integer	93
* 2.9.1 GetVersionDRV as integer	93
* 2.9.1 GetVersionDRV_ USB as integer	93
* 2.9.1 InitMatrixAPI as integer	93
* 2.9.1 LogInMatrixNet(UserCode as integer, AppSlot as integer, DongleNr as integer) as integer	94
* 2.9.1 LogOutMatrixNet(UserCode as integer, AppSlot as integer, DongleNr as integer) as integer	94
* 2.9.1 PausePrinterActivity as integer	94
* 2.9.1 ReleaseMatrixAPI as integer	94
* 2.9.1 ResumePrinterActivity as integer	94
* 2.9.1 SetConfigMatrixNet(Access as integer, File as string) as integer	95
* 2.9.1 SetDriverFlag(UserCode as integer, DriverFlag as integer, DongleNr as integer, PortNr as integer) as integer	95
* 2.9.1 SetW95Access(mode as integer)	95
– 2.4 module SecureDongleXMBS	27
* 2.4.1 Close(handle as Integer)	27
* 2.4.1 Find as Integer	27
* 2.4.1 GetVersion(handle as Integer) as Integer	27
* 2.4.1 Open(mode as integer, uid as Integer) as Integer	28
* 2.4.1 Open(mode as integer, uid as Integer, byref hid as integer) as Integer	28
* 2.4.1 Read(handle as Integer, BlockIndex as integer, byref data as string) as Integer	28
* 2.4.1 RSADecrypt(handle as Integer, startIndex as integer, byref buf as string, byref key as string) as Integer	29

* 2.4.1 RSAEncrypt(handle as Integer, startIndex as integer, byref buf as string, byref key as string) as Integer	29
* 2.4.1 Transform(handle as Integer, byref data as string) as Integer	29
* 2.4.1 Write(handle as Integer, BlockIndex as integer, data as string) as Integer	30
* 2.4.2 HID_ MODE = -1	30
* 2.4.2 SDXERR_ FAILED_ DECRYPTION = & hA010000E	30
* 2.4.2 SDXERR_ FAILED_ ENCRYPTION = & hA010000C	30
* 2.4.2 SDXERR_ FAILED_ WRITE_ KEY = & hA010000D	31
* 2.4.2 SDXERR_ FLUSH_ QUEUE = & hA0100017	31
* 2.4.2 SDXERR_ FREE_ PREPARSED_ DATA = & hA0100016	31
* 2.4.2 SDXERR_ GENERATE_ KEY = & hA010000A	31
* 2.4.2 SDXERR_ GETCAPS = & hA0100015	31
* 2.4.2 SDXERR_ GET_ ATTRIBUTES = & hA0100013	32
* 2.4.2 SDXERR_ GET_ PREPARSED_ DATA = & hA0100014	32
* 2.4.2 SDXERR_ GET_ SERIAL = & hA0100019	32
* 2.4.2 SDXERR_ INVALID_ KEY = & hA010000B	32
* 2.4.2 SDXERR_ INVALID_ LEN = & hA0100008	32
* 2.4.2 SDXERR_ NOT_ OPENED_ DEVICE = & hA0100002	33
* 2.4.2 SDXERR_ NO_ SUCH_ DEVICE = & hA0100001	33
* 2.4.2 SDXERR_ OPEN_ DEVICE = & hA010000F	33
* 2.4.2 SDXERR_ READ_ REPORT = & hA0100010	33
* 2.4.2 SDXERR_ SETUP_ DI_ CLASS_ DEVS = & hA0100018	33
* 2.4.2 SDXERR_ SETUP_ DI_ GET_ DEVICE_ INTERFACE_ DETAIL = & hA0100012	34
* 2.4.2 SDXERR_ SUCCESS = 0	34
* 2.4.2 SDXERR_ TOO_ LONG_ DEVICE_ DETAIL = & hA010001B	34
* 2.4.2 SDXERR_ TOO_ LONG_ ENCRYPTION_ DATA = & hA0100009	34
* 2.4.2 SDXERR_ TOO_ LONG_ SEED = & hA0100005	34
* 2.4.2 SDXERR_ UNKNOWN_ DEVICE = & hA0100020	35
* 2.4.2 SDXERR_ UNKNOWN_ ERROR = & hA010FFFF	35
* 2.4.2 SDXERR_ VERIFY = & hA0100021	35
* 2.4.2 SDXERR_ WRITE_ PROTECT = & hA0100006	35
* 2.4.2 SDXERR_ WRITE_ REPORT = & hA0100011	35
* 2.4.2 SDXERR_ WRONG_ INDEX = & hA0100004	36
* 2.4.2 SDXERR_ WRONG_ REPORT_ LENGTH = & hA010001A	36
* 2.4.2 SDXERR_ WRONG_ START_ INDEX = & hA0100007	36
* 2.4.2 SDXERR_ WRONG_ UID = & hA0100003	36
* 2.4.2 SDX_ MINOR = 16	36
- 2.3 class UnikeyMBS	24
* 2.3.1 Unikey(FunctionCode as integer) as integer	25
* 2.3.2 Buffer as MemoryBlock	25
* 2.3.2 Handle as Integer	25

* 2.3.2 LP1 as Integer	25
* 2.3.2 LP2 as Integer	26
* 2.3.2 P1 as Integer	26
* 2.3.2 P2 as Integer	26
* 2.3.2 P3 as Integer	26
* 2.3.2 P4 as Integer	26
– 2.5 class RockeyMBS	37
* 2.5.1 Rockey(FunctionCode as integer) as integer	37
* 2.5.1 RockeyCall(FunctionCode as integer) as integer	38
* 2.5.2 Buffer as memoryblock	38
* 2.5.2 Handle as Integer	38
* 2.5.2 LP1 as Integer	38
* 2.5.2 LP2 as Integer	39
* 2.5.2 P1 as Integer	39
* 2.5.2 P2 as Integer	39
* 2.5.2 P3 as Integer	39
* 2.5.2 P4 as Integer	39
– 2.7 class HASPHLDMBS	41
* 2.7.1 Close	42
* 2.7.1 Constructor(FeatureID as integer, scope as string, VendorCode as string)	42
* 2.7.1 Constructor(FeatureID as integer, VendorCode as string)	43
* 2.7.1 DateTimeToHaspTime(day as integer, month as integer, year as integer, hour as integer, minute as integer, second as integer) as memoryblock	44
* 2.7.1 DecryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)	44
* 2.7.1 DecryptString(Data as string) as string	45
* 2.7.1 Detach(detachAction as string, scope as string, VendorCode as string, recipient as string, byref info as string) as integer	45
* 2.7.1 EncryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)	45
* 2.7.1 EncryptString(Data as string) as string	46
* 2.7.1 GetInfo(scope as string, format as string, VendorCode as string, byref info as string) as integer	47
* 2.7.1 GetRTC as memoryblock	48
* 2.7.1 GetSessionInfo(format as string) as string	48
* 2.7.1 GetSize(FileID as integer) as integer	49
* 2.7.1 GetVersion(byref MajorVersion as integer, byref MinorVersion as integer, byref BuildServer as integer, byref BuildNumber as integer, VendorCode as string) as integer	49
* 2.7.1 HaspTimeToDateTime(time as memoryblock, byref day as integer, byref month as integer, byref year as integer, byref hour as integer, byref minute as integer, byref second as integer)	49
* 2.7.1 LegacyDecryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)	50
* 2.7.1 LegacyDecryptString(Data as string) as string	50

* 2.7.1 LegacyEncryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)	
50	
* 2.7.1 LegacyEncryptString(Data as string) as string	51
* 2.7.1 LegacySetIdleTime(idletime as integer)	51
* 2.7.1 LegacySetRTC(time as memoryblock)	51
* 2.7.1 LoadLibrary(file as folderitem) as boolean	52
* 2.7.1 LoadLibrary(path as string) as boolean	52
* 2.7.1 ReadMemory(FileID as integer, Offset as integer, Size as integer) as Memoryblock	53
* 2.7.1 ReadString(FileID as integer, Offset as integer, Size as integer) as string	53
* 2.7.1 Update(data as string) as string	54
* 2.7.1 WriteMemory(FileID as integer, FileOffset as integer, Data as Memoryblock, DataOffset as integer, Size as integer)	54
* 2.7.1 WriteString(FileID as integer, FileOffset as integer, Data as String)	55
* 2.7.2 Handle as Integer	55
* 2.7.2 Lasterror as Integer	56
* 2.7.3 HASP_ACCESS_DENIED = 5	56
* 2.7.3 HASP_ALREADY_LOGGED_IN = 502	56
* 2.7.3 HASP_ALREADY_LOGGED_OUT = 503	56
* 2.7.3 HASP_BROKEN_SESSION = 39	56
* 2.7.3 HASP_CLONE_DETECTED = 64	57
* 2.7.3 HASP_CONTAINER_NOT_FOUND = 7	57
* 2.7.3 HASP_DEFAULT_FID = 0	57
* 2.7.3 HASP_DEVICE_ERR = 43	57
* 2.7.3 HASP_ENC_NOT_SUPP = 23	57
* 2.7.3 HASP_FEATURETYPE_MASK = & hfff0000	58
* 2.7.3 HASP_FEATURE_EXPIRED = 41	58
* 2.7.3 HASP_FEATURE_NOT_FOUND = 31	58
* 2.7.3 HASP_FEATURE_TYPE_NOT_IMPL = 28	58
* 2.7.3 HASP_FILEID_LICENSE = & hfff2	58
* 2.7.3 HASP_FILEID_MAIN = & hfff0	59
* 2.7.3 HASP_FILEID_RO = & hfff5	59
* 2.7.3 HASP_FILEID_RW = & hfff4	59
* 2.7.3 HASP_FIRST_HASP_ACT = 3001	59
* 2.7.3 HASP_FIRST_HELPER = 2001	60
* 2.7.3 HASP_HARDWARE_MODIFIED = 52	60
* 2.7.3 HASP_HASP_NOT_FOUND = 7	60
* 2.7.3 HASP_INCOMPAT_FEATURE = 6	60
* 2.7.3 HASP_INSUF_MEM = 3	60
* 2.7.3 HASP_INT_ERR = 699	61
* 2.7.3 HASP_INVALID_HANDLE_VALUE = 0	61
* 2.7.3 HASP_INVALID_OBJECT = 500	61
* 2.7.3 HASP_INVALID_PARAMETER = 501	61

* 2.7.3 HASP_INV_API_DYLIB = 401	61
* 2.7.3 HASP_INV_DETACH_ACTION = 59	62
* 2.7.3 HASP_INV_DURATION = 63	62
* 2.7.3 HASP_INV_FILEID = 10	62
* 2.7.3 HASP_INV_FORMAT = 15	62
* 2.7.3 HASP_INV_HND = 9	62
* 2.7.3 HASP_INV_PORT = 651	63
* 2.7.3 HASP_INV_PORT_TYPE = 650	63
* 2.7.3 HASP_INV_PRODUCT = 61	63
* 2.7.3 HASP_INV_PROGNUM_OPT = 2	63
* 2.7.3 HASP_INV_RECIPIENT = 58	63
* 2.7.3 HASP_INV_SCOPE = 36	64
* 2.7.3 HASP_INV_SIG = 30	64
* 2.7.3 HASP_INV_SPEC = 35	64
* 2.7.3 HASP_INV_TIME = 24	64
* 2.7.3 HASP_INV_UPDATE_CNTR = 21	64
* 2.7.3 HASP_INV_UPDATE_DATA = 19	65
* 2.7.3 HASP_INV_UPDATE_NOTSUPP = 20	65
* 2.7.3 HASP_INV_UPDATE_OBJ = 17	65
* 2.7.3 HASP_INV_VCODE = 22	65
* 2.7.3 HASP_INV_VLIB = 49	65
* 2.7.3 HASP_KEYID_NOT_FOUND = 18	66
* 2.7.3 HASP_LOCAL_COMM_ERR = 33	66
* 2.7.3 HASP_MEM_RANGE = 1	66
* 2.7.3 HASP_MIN_BLOCK_SIZE = 16	66
* 2.7.3 HASP_MIN_BLOCK_SIZE_LEGACY = 8	66
* 2.7.3 HASP_NEXT_FREE_VALUES = 5001	66
* 2.7.3 HASP_NOT_IMPL = 698	67
* 2.7.3 HASP_NO_ACK_SPACE = 26	67
* 2.7.3 HASP_NO_API_DYLIB = 400	67
* 2.7.3 HASP_NO_BATTERY_POWER = 25	67
* 2.7.3 HASP_NO_DRIVER = 14	67
* 2.7.3 HASP_NO_EXTBLOCK = 600	68
* 2.7.3 HASP_NO_LOG = 32	68
* 2.7.3 HASP_NO_TIME = 12	68
* 2.7.3 HASP_NO_VLIB = 48	68
* 2.7.3 HASP_OLD_DRIVER = 11	68
* 2.7.3 HASP_OLD_LM = 42	69
* 2.7.3 HASP_OLD_VLIB = 56	69
* 2.7.3 HASP_OPERATION_FAILED = 525	69
* 2.7.3 HASP_PROGNUM_DEFAULT_FID = & hfff0000	69
* 2.7.3 HASP_PROGNUM_FEATURETYPE = & hfff0000	69

* 2.7.3 HASP_PROGNUM_MASK = & h000000ff	70
* 2.7.3 HASP_PROGNUM_OPT_CLASSIC = & h00001000	70
* 2.7.3 HASP_PROGNUM_OPT_MASK = & h0000ff00	70
* 2.7.3 HASP_PROGNUM_OPT_NO_LOCAL = & h00008000	70
* 2.7.3 HASP_PROGNUM_OPT_NO_REMOTE = & h00004000	71
* 2.7.3 HASP_PROGNUM_OPT_PROCESS = & h00002000	71
* 2.7.3 HASP_PROGNUM_OPT_TS = & h00000800	71
* 2.7.3 HASP_REMOTE_COMM_ERR = 40	71
* 2.7.3 HASP_REQ_NOT_SUPP = 16	71
* 2.7.3 HASP_SCHAN_ERR = 46	72
* 2.7.3 HASP_SCOPE_RESULTS_EMPTY = 50	72
* 2.7.3 HASP_STATUS_OK = 0	72
* 2.7.3 HASP_STORAGE_CORRUPT = 47	72
* 2.7.3 HASP_SYS_ERR = 13	72
* 2.7.3 HASP_TIME_ERR = 45	73
* 2.7.3 HASP_TMOF = 4	73
* 2.7.3 HASP_TOO_MANY_KEYS = 37	73
* 2.7.3 HASP_TOO_MANY_PRODUCTS = 60	73
* 2.7.3 HASP_TOO_MANY_USERS = 38	73
* 2.7.3 HASP_TOO_SHORT = 8	74
* 2.7.3 HASP_TS_DETECTED = 27	74
* 2.7.3 HASP_UNKNOWN_ALG = 29	74
* 2.7.3 HASP_UNKNOWN_RECIPIENT = 62	74
* 2.7.3 HASP_UNKNOWN_VCODE = 34	74
* 2.7.3 HASP_UPDATE_BLOCKED = 44	75
* 2.7.3 HASP_UPDATE_TOO_NEW = 55	75
* 2.7.3 HASP_UPDATE_TOO_OLD = 54	75
* 2.7.3 HASP_UPLOAD_ERROR = 57	75
* 2.7.3 HASP_USER_DENIED = 53	75
* 2.7.3 HASP_VM_DETECTED = 51	76
– 2.6 Globals	40
* 2.6 CallHASPMBs(service as integer, seed as integer, lptnum as integer, pass1 as integer, pass2 as integer, byref p1 as integer, byref p2 as integer, byref p3 as integer, byref p4 as integer)	40
* 2.6 CallHASPMemMBS(service as integer, seed as integer, lptnum as integer, pass1 as integer, pass2 as integer, byref p1 as integer, byref p2 as integer, byref p3 as integer, byref p4 as integer, mem as memoryblock)	40
* 2.6 GetHASPErrStrMBS(error as integer) as string	41
* 2.6 GetNetHaspWarningStrMBS(error as integer) as string	41
– 2.8 class HASPHLMBS	76
* 2.8.1 Close	76
* 2.8.1 Constructor(FeatureID as integer, VendorCode as string)	76

* 2.8.1 DateTimeToHaspTime(day as integer, month as integer, year as integer, hour as integer, minute as integer, second as integer) as memoryblock	77
* 2.8.1 DecryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)	77
* 2.8.1 DecryptString(Data as string) as string	78
* 2.8.1 EncryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)	78
* 2.8.1 EncryptString(Data as string) as string	79
* 2.8.1 GetRTC as memoryblock	79
* 2.8.1 GetSessionInfo(format as string) as string	79
* 2.8.1 GetSize(FileID as integer) as integer	82
* 2.8.1 HaspTimeToDateTime(time as memoryblock, byref day as integer, byref month as integer, byref year as integer, byref hour as integer, byref minute as integer, byref second as integer)	83
* 2.8.1 LegacyDecryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)	83
* 2.8.1 LegacyDecryptString(Data as string) as string	83
* 2.8.1 LegacyEncryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)	84
* 2.8.1 LegacyEncryptString(Data as string) as string	84
* 2.8.1 LegacySetIdleTime(idletime as integer)	85
* 2.8.1 LegacySetRTC(time as memoryblock)	85
* 2.8.1 ReadMemory(FileID as integer, Offset as integer, Size as integer) as Memoryblock	85
* 2.8.1 ReadString(FileID as integer, Offset as integer, Size as integer) as string	86
* 2.8.1 Update(data as string) as string	86
* 2.8.1 WriteMemory(FileID as integer, FileOffset as integer, Data as Memoryblock, DataOffset as integer, Size as integer)	87
* 2.8.1 WriteString(FileID as integer, FileOffset as integer, Data as String)	87
* 2.8.2 Handle as Integer	87
* 2.8.2 Lasterror as Integer	88
<b>• 3 Kagi</b>	97
– 3.5 class ZKRMResultMBS	112
* 3.5.1 acgRegCode as String	112
* 3.5.1 acgUserName as String	112
* 3.5.1 KagiReplyXML as String	112
* 3.5.1 KagiTransactionID as String	113
* 3.5.1 ModuleStatus as Integer	113
* 3.5.1 ModuleVersion as Integer	113
* 3.5.1 OrderStatus as Integer	114
* 3.5.1 OrderType as Integer	114
– 3.3 class KRMWindowsMBS	99
* 3.3.1 AddPrice(Currency as string, Price as Double, BaseCurrency as String) as boolean	99
* 3.3.1 CloseWizard as boolean	99

* 3.3.1 Exec as boolean	99
* 3.3.1 ExecModal(win as window) as boolean	100
* 3.3.1 GetLastError as integer	100
* 3.3.1 GetResultInt(ResultID as integer, byref Value as integer) as boolean	100
* 3.3.1 GetResultStr(ResultID as integer, byref Value as String) as Integer	100
* 3.3.1 SetDataInt(DataID as integer, Value as integer) as boolean	101
* 3.3.1 SetDataStr(DataID as integer, Value as String) as boolean	102
* 3.3.2 Available as Boolean	103
* 3.3.2 BuildNumber as Integer	103
* 3.3.2 MajorVersion as Integer	103
* 3.3.2 MinorVersion as Integer	103
* 3.3.3 StatusChanged(Status as integer, Value as integer, Data as string)	103
– 3.1 class ZKRMKeyValuePairMBS	97
* 3.1.1 Key as String	97
* 3.1.1 Value as String	98
– 3.2 Globals	98
* 3.2 EncryptKRMStringMBS(byref data as string) as integer	98
* 3.2 IsKRMAvailableMBS as integer	98
– 3.4 class ZKRMPParametersMBS	104
* 3.4.1 AddKeyValuePair(item as ZKRMKeyValuePairMBS)	105
* 3.4.1 BeginModalKRM(byref result as ZKRMResultMBS) as integer	105
* 3.4.1 BeginModelessKRM as integer	106
* 3.4.1 ClearKeyValuePairs	108
* 3.4.1 GetKeyValuePair(index as integer) as ZKRMKeyValuePairMBS	108
* 3.4.1 KeyValuePairCount as integer	108
* 3.4.1 Version as integer	108
* 3.4.2 ModuleLanguage as Integer	108
* 3.4.2 ModuleOptions as Integer	109
* 3.4.2 ModuleUserEmail as String	110
* 3.4.2 ModuleUserName as String	110
* 3.4.2 ProductAffiliate as String	110
* 3.4.2 ProductInitXML as String	110
* 3.4.2 ProductPO as String	111
* 3.4.2 ProductStoreURL as String	111
* 3.4.2 ProductTFYP as String	111
* 3.4.3 Completed(result as ZKRMResultMBS)	111

# Chapter 2

# Dongle

## 2.1 class Rockey4NDMBS

`class Rockey4NDMBS`

Plugin Version: 8.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** A class to handle calls to the Rockey4ND Dongle API.

### 2.1.1 Methods

`Rockey(FunctionCode as integer) as integer`

Plugin Version: 8.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Calls the Rockey Dongle API.

**Example:**

```
dim r as RockeyMBS
dim e as integer
```

```
r=new RockeyMBS
r.p1=& HC44C
r.p2=& HC8F8
r.p3=0
r.p4=0
```

```
e=r.Rockey(1)

if e=0 then
MsgBox "Found dongle: "+hex(r.lp1)
else
MsgBox "Error: "+str(e)
end if
```

**Notes:**

See the Rockey dongle documentation for more details.

Be aware that no endian correction is done!

Error codes from the plugin:

-1 = memoryblock is nil

-2 = library not present

**RockeyCall(FunctionCode as integer) as integer**

Plugin Version: 8.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Same as Rockey but a different name for compatibility.

**Notes:** The original Rockey dongle calls the function RockeyCall.

### 2.1.2 Properties

**Buffer as MemoryBlock**

Plugin Version: 8.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** A one KB big buffer.

**Notes:** (Read and Write property)

### Handle as Integer

Plugin Version: 8.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The 16bit handle value.

**Notes:** (Read and Write property)

### LP1 as Integer

Plugin Version: 8.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The first 32bit parameter for the rocky function.

**Notes:** (Read and Write property)

### LP2 as Integer

Plugin Version: 8.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The second 32bit parameter for the rocky function.

**Notes:** (Read and Write property)

### P1 as Integer

Plugin Version: 8.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The first 16bit parameter for the rocky function.

**Notes:** (Read and Write property)

### P2 as Integer

Plugin Version: 8.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The second 16bit parameter for the rocky function.

**Notes:** (Read and Write property)

### P3 as Integer

Plugin Version: 8.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The third 16bit parameter for the rocky function.

**Notes:** (Read and Write property)

### P4 as Integer

Plugin Version: 8.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The forth 16bit parameter for the rocky function.

**Notes:** (Read and Write property)

## 2.2 class Rocky2MBS

### class Rocky2MBS

Plugin Version: 9.6 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The class to use Rocky 2 dongles in REALbasic.

### 2.2.1 Methods

#### Close

Plugin Version: 9.6 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Closes the dongle.  
**Notes:**

Lasterror is set.

This is called automatically by the destructor.

**Find as Int32**

Plugin Version: 9.6 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Searches for Rockey 2 dongles on the computer.

**Notes:**

Lasterror is set.

Returns the number of attached Rockey 2 dongles.

**GenUID(byref uid as UInt32, seed as string, isProtect as boolean)**

Plugin Version: 9.6 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Generates an user ID.

**Notes:**

UID: the generated user id.

seed: Seed to generate UID. It is a character string with the maximum length of 64 bytes.

isProtect: Sets write protection. 0=dongle not write protected, 1=dongle write protected.

Lasterror is set.

**Open(mode as Int32, uid as UInt32, byref hid as UInt32)**

Plugin Version: 9.6 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Opens a specific rockey 2 dongle.

**Notes:**

mode:

This parameter indicates the way to open the dongle

mode = 0, open the first found ROCKEY2 dongle

mode >0, open the dongle according to the UID. The mode value is the dongle number, for example: uid=12345678, mode=2, this means it will open the second dongle with UID 12345678

mode = -1, open the dongle according to the HID, and \*hid can not be 0 We defined two constants: AUTO\_MODE=0 and HID\_MODE=-1

uid(UserID): You need to specify the dongle UID and this UID is generated with RY2\_GenUID

hid(HardwareID): Open dongle with HID of hid. The dongle HID will be returned to hid regardless of how the dongle was opened.

Sets the handle and lasterror properties.

### **Read(BlockIndex as Int32) as string**

Plugin Version: 9.6 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Read dongle content.  
**Notes:**

BlockIndex: Block index. Specify the block to write. The value range is 0-4  
Returns 512 bytes on success.  
LastError is set.

### **Transform(data as string)**

Plugin Version: 9.6 Console & Web: Yes Mac: Yes, Win: No, Linux: Yes, . **Function:** Transforms data.

### **Write(BlockIndex as Int32, data as string)**

Plugin Version: 9.6 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Writes data to the dongle.  
**Notes:**

Blockindex: Specify the block to write. The value range is 0-4.  
LastError is set.  
If you pass less than 512 bytes, the remaining space is filled with zeros.

## **2.2.2 Properties**

### **Handle as Integer**

Plugin Version: 9.6 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The internal handle for dongle communication.

**Notes:** (Read and Write property)

### Lasterror as Integer

Plugin Version: 9.6 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The last error code.

**Notes:** (Read and Write property)

### 2.2.3 Constants

#### AUTO\_MODE = 0

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** A constant for use in the Open method.

#### Available = true

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Whether rocky functions are available.

**Notes:** Currently plugin code is working on Linux, Windows and Mac OS X.

#### HID\_MODE = -1

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** A constant for use in the Open method.

#### R2\_MINOR = 16

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The maximum number of dongles the plugin can handle.

**ROCKEY2\_DISABLE\_WRITE\_PROTECT = false**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the constants for the GenUID function.

**ROCKEY2\_ENABLE\_WRITE\_PROTECT = true**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the constants for the GenUID function.

**RY2ERR\_FLUSH\_QUEUE = & hA010000F**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Internal error (Windows error)

**RY2ERR\_FREE\_PREPARED\_DATA = & hA010000E**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Internal error (Windows error)

**RY2ERR\_GETCAPS = & hA010000D**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Internal error (Windows error)

**RY2ERR\_GET\_ATTRIBUTES = & hA010000B**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Internal error (Windows error)

**RY2ERR\_GET\_PREPARED\_DATA = & hA010000C**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Internal error (Windows error)

**RY2ERR\_GET\_SERIAL = & hA0100011**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Internal error (Windows error)

**RY2ERR\_NOT\_OPENED\_DEVICE = & hA0100002**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Need to call RY2\_Open first to open the dongle, then call this function (operation error)

**RY2ERR\_NO\_SUCH\_DEVICE = & hA0100001**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Specified dongle is not found (parameter error)

**RY2ERR\_OPEN\_DEVICE = & hA0100007**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Open device error (Windows error)

**RY2ERR\_READ\_REPORT = & hA0100008**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Read record error(Windows error)

**RY2ERR\_SETUP\_DI\_CLASS\_DEVS = & hA0100010**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Internal error (Windows error)

**RY2ERR\_SETUP\_DI\_GET\_DEVICE\_INTERFACE\_DETAIL = & hA010000A**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Internal error (Windows error)

**RY2ERR\_SUCCESS = 0**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Success

**RY2ERR\_TOO\_LONG\_DEVICE\_DETAIL = & hA0100012**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Internal error

**RY2ERR\_ TOO\_ LONG\_ SEED = & hA0100005**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Seed character string is longer than 64 bytes when calling GenUID (parameter error)

**RY2ERR\_ UNKNOWN\_ DEVICE = & hA0100020**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Unknown device.

**RY2ERR\_ UNKNOWN\_ ERROR = & hA010FFFF**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Unknown error(hardware error)

**RY2ERR\_ VERIFY = & hA0100014**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Verification error(hardware error)

**RY2ERR\_ WRITE\_ PROTECT = & hA0100006**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Tried to write to write-protected dongle(operation error)

**RY2ERR\_ WRITE\_ REPORT = & hA0100009**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Write record error(Windows error)

**RY2ERR\_ WRONG\_ INDEX = & hA0100004**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Block index error (parameter error)

**RY2ERR\_ WRONG\_ REPORT\_ LENGTH = & hA0100013**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Wrong report length.

**RY2ERR\_ WRONG\_ UID = & hA0100003**

Plugin Version: 9.6 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the error constants.

**Notes:** Wrong UID(parameter error)

## 2.3 class UnikeyMBS

### class UnikeyMBS

Plugin Version: 7.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** A class to enable REALbasic to use Unikey dongles.

**Notes:**

Website:

<http://www.esecutech.com>

### 2.3.1 Methods

#### Unikey(FunctionCode as integer) as integer

Plugin Version: 7.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Calls the unikey function with the values from this object.

**Notes:** See the unikey documentation for what values to use.

### 2.3.2 Properties

#### Buffer as MemoryBlock

Plugin Version: 7.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The memoryblock to use when calling the unikey function.

**Notes:** (Read and Write property)

#### Handle as Integer

Plugin Version: 7.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The handle to the dongle.

**Notes:** (Read and Write property)

#### LP1 as Integer

Plugin Version: 7.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The parameter LP1.

**Notes:** (Read and Write property)

**LP2 as Integer**

Plugin Version: 7.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The parameter LP2.

**Notes:** (Read and Write property)

**P1 as Integer**

Plugin Version: 7.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The value for the parameter p1.

**Notes:** (Read and Write property)

**P2 as Integer**

Plugin Version: 7.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The value for the parameter p2.

**Notes:** (Read and Write property)

**P3 as Integer**

Plugin Version: 7.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The value for the parameter p3.

**Notes:** (Read and Write property)

**P4 as Integer**

Plugin Version: 7.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The value for the parameter p4.

**Notes:** (Read and Write property)

## 2.4 module SecureDongleXMBS

### module SecureDongleXMBS

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** This module implements an interface to the SecureDongleX products from SecureMetric Technology Sdn. Bhd.

**Notes:**

For more details see the website:  
<http://www.securemetric.com/>

Please order there your evaluation SDK as well as further dongles.

### 2.4.1 Methods

#### Close(handle as Integer)

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Close specified SecureDongle X

**Notes:** Pass the handle you got returned from Open method.

#### Find as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Find dongles attached to the computer

**Notes:**

See SDK Developer Guide for details.  
Returns number of dongles or negative error code.

#### GetVersion(handle as Integer) as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Get SecureDongle X hardware version

**Notes:**

See SDK Developer Guide for details.  
Returns positive version number or nega

**Open(mode as integer, uid as Integer) as Integer**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Open specified SecureDongle X

**Notes:**

Convenience method where hid parameter is left away.  
The default UID is 715400947 for demo dongles.  
See SDK Developer Guide for details.  
See also:

- 2.4.1 Open(mode as integer, uid as Integer, byref hid as integer) as Integer 28

**Open(mode as integer, uid as Integer, byref hid as integer) as Integer**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Open specified SecureDongle X

**Notes:**

The default UID is 715400947 for demo dongles.  
See SDK Developer Guide for details.  
See also:

- 2.4.1 Open(mode as integer, uid as Integer) as Integer 28

**Read(handle as Integer, BlockIndex as integer, byref data as string) as Integer**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Read SecureDongle X content

**Notes:**

handle: SDX handle. It is the handle returned from Open  
block\_index: The block index. Specify the block to read. The value range is 0-4

data: Read buffer. Here you receive a string with 512 bytes.

Returns an error code.

See SDK Developer Guide for details.

**RSADecrypt(handle as Integer, startIndex as integer, byref buf as string, byref key as string) as Integer**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Read SecureDongle X content and Decrypt with RSA.

**Notes:** See SDK Developer Guide for details.

**RSAEncrypt(handle as Integer, startIndex as integer, byref buf as string, byref key as string) as Integer**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Encrypt with RSA and write to SecureDongle X.

**Notes:** See SDK Developer Guide for details.

**Transform(handle as Integer, byref data as string) as Integer**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Data hashing function.

**Notes:**

On input data contains a string to transform (maximum 32 bytes) and on return the hash value.

See SDK Developer Guide for details.

**Write(handle as Integer, BlockIndex as integer, data as string) as Integer**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Write to SecureDongle X

**Notes:**

handle: SDX handle. It is the handle returned from Open

block\_index: The Block index. Specify the block to write. The value range is 0-4

data: Write buffer. The buffer can be at maximum 512 bytes to accommodate the 512-byte block size. The plugin will fill the rest with zeros.

Returns error code.

See SDK Developer Guide for details.

## 2.4.2 Constants

**HID\_MODE = -1**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The constant for default HID mode.

**SDXERR\_FAILED\_DECRYPTION = & hA010000E**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Failed Decrypt string (Cryptography error)

**SDXERR\_FAILED\_ENCRYPTION = & hA010000C**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Failed encrypt string (cryptography error)

**SDXERR\_FAILED\_WRITE\_KEY = & hA010000D**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Failed write key (cryptography error)

**SDXERR\_FLUSH\_QUEUE = & hA0100017**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Internal error (Windows error)

**SDXERR\_FREE\_PREPARED\_DATA = & hA0100016**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Internal error (Windows error)

**SDXERR\_GENERATE\_KEY = & hA010000A**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Generate key error (cryptography error)

**SDXERR\_GETCAPS = & hA0100015**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Internal error (Windows error)

**SDXERR\_GET\_ATTRIBUTES = & hA0100013**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Internal error (Windows error)

**SDXERR\_GET\_PREPARED\_DATA = & hA0100014**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Internal error (Windows error)

**SDXERR\_GET\_SERIAL = & hA0100019**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Internal error (Windows error)

**SDXERR\_INVALID\_KEY = & hA010000B**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Invalid key (cryptography error)

**SDXERR\_INVALID\_LEN = & hA0100008**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Invalid length (parameter error)

**SDXERR\_NOT\_OPENED\_DEVICE = & hA0100002**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Need to call SDX\_Open first to open the SDX, then call this function (operation error)

**SDXERR\_NO\_SUCH\_DEVICE = & hA0100001**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Specified SDX is not found (parameter error)

**SDXERR\_OPEN\_DEVICE = & hA010000F**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Open device error (Windows error)

**SDXERR\_READ\_REPORT = & hA0100010**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Read record error (Windows error)

**SDXERR\_SETUP\_DI\_CLASS\_DEVS = & hA0100018**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Internal error (Windows error)

**SDXERR\_SETUP\_DI\_GET\_DEVICE\_INTERFACE\_DETAIL = & hA0100012**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Internal error (Windows error)

**SDXERR\_SUCCESS = 0**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Success

**SDXERR\_TOO\_LONG\_DEVICE\_DETAIL = & hA010001B**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Internal error

**SDXERR\_TOO\_LONG\_ENCRYPTION\_DATA = & hA0100009**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Cipher text length is too long (cryptography error)

**SDXERR\_TOO\_LONG\_SEED = & hA0100005**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Seed character string is longer than 64 bytes when calling GenUID (parameter error)

**SDXERR\_ UNKNOWN\_ DEVICE = & hA0100020**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Unknown device.

**SDXERR\_ UNKNOWN\_ ERROR = & hA010FFFF**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Unknown error (hardware error)

**SDXERR\_ VERIFY = & hA0100021**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Verification error (hardware error)

**SDXERR\_ WRITE\_ PROTECT = & hA0100006**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Tried to write to write-protected SDX (operation error)

**SDXERR\_ WRITE\_ REPORT = & hA0100011**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Write record error (Windows error)

**SDXERR\_WRONG\_INDEX = & hA0100004**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Block index error (parameter error)

**SDXERR\_WRONG\_REPORT\_LENGTH = & hA010001A**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Unknown device (hardware error)

**SDXERR\_WRONG\_START\_INDEX = & hA0100007**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Start index error (parameter error)

**SDXERR\_WRONG\_UID = & hA0100003**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the possible error codes for functions in this module.

**Notes:** Wrong UID (parameter error)

**SDX\_MINOR = 16**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The maximum number of sdx devices.

## 2.5 class RockyMBS

**class RockyMBS**

Plugin Version: 3.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** A class to handle calls to the Rocky Dongle API.

### 2.5.1 Methods

**Rocky(FunctionCode as integer) as integer**

Plugin Version: 3.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Calls the Rocky Dongle API.

**Example:**

```
dim r as RockyMBS
dim e as integer

r=new RockyMBS
r.p1=& HC44C
r.p2=& HC8F8
r.p3=0
r.p4=0

e=r.Rockey(1)

if e=0 then
MsgBox "Found dongle: "+hex(r.lp1)
else
MsgBox "Error: "+str(e)
end if
```

**Notes:**

See the Rocky dongle documentation for more details.

Be aware that no endian correction is done!

Error codes from the plugin:

-1 = memoryblock is nil

-2 = library not present (always on Linux)

### **RockeyCall(FunctionCode as integer) as integer**

Plugin Version: 7.0 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Same as Rockey but a different name for compatibility.

**Notes:** The original Rockey dongle calls the function RockeyCall.

## **2.5.2 Properties**

### **Buffer as memoryblock**

Plugin Version: 3.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** A one KB big buffer.

**Notes:** (Read and Write property)

### **Handle as Integer**

Plugin Version: 3.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The 16bit handle value.

**Notes:** (Read and Write property)

### **LP1 as Integer**

Plugin Version: 3.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The first 32bit parameter for the rokey function.

**Notes:** (Read and Write property)

### LP2 as Integer

Plugin Version: 3.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The second 32bit parameter for the rocky function.

**Notes:** (Read and Write property)

### P1 as Integer

Plugin Version: 3.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The first 16bit parameter for the rocky function.

**Notes:** (Read and Write property)

### P2 as Integer

Plugin Version: 3.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The second 16bit parameter for the rocky function.

**Notes:** (Read and Write property)

### P3 as Integer

Plugin Version: 3.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The third 16bit parameter for the rocky function.

**Notes:** (Read and Write property)

### P4 as Integer

Plugin Version: 3.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The forth 16bit parameter for the rocky function.

**Notes:** (Read and Write property)

## 2.6 Globals

**CallHASPMB**(service as integer, seed as integer, lptnum as integer, pass1 as integer, pass2 as integer, byref p1 as integer, byref p2 as integer, byref p3 as integer, byref p4 as integer)

Plugin Version: 3.3 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Calls the HASP API.

**Notes:**

You may need drivers from Aladdin Knowledge Systems for your HASP keys, but the plugin should run and compile even without.

Linux added for plugin v7.0

Please check the HASP documentation from Aladdin for the list of error codes.

If you need a plugin for HASP SRM, please contact us.

**CallHASPMemMBS**(service as integer, seed as integer, lptnum as integer, pass1 as integer, pass2 as integer, byref p1 as integer, byref p2 as integer, byref p3 as integer, byref p4 as integer, mem as memoryblock)

Plugin Version: 3.3 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Calls the HASP API.

**Notes:**

This version of the Call will pass in the p3 and p4 parameters the address of the memoryblock to the function.

Linux added for plugin v7.0

Please check the HASP documentation from Aladdin for the list of error codes.

**GetHASPErrStrMBS(error as integer) as string**

Plugin Version: 3.3 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Returns a human readable error string for the given error code.

**Notes:**

May return "" on low memory.

Please check the HASP documentation from Aladdin for the list of error codes.

**GetNetHaspWarningStrMBS(error as integer) as string**

Plugin Version: 3.3 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Returns a human readable string for the given error code.

**Notes:**

May return nil on low memory.

Please check the HASP documentation from Aladdin for the list of error codes.

## 2.7 class HASPHLDMBS

**class HASPHLDMBS**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** A class for accessing HASP HL dongles.

**Notes:** HASPHLMBS uses a linked in library. This is okay for older HASP HL versions. But newer versions require you to create your own signed libraries. For them use the HASPHLDMBS class.

### 2.7.1 Methods

#### Close

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Logs out from a context or session.

**Notes:**

Called automatically for you by the destructor.

Use this function to end a connection to an API session object. Once logged out from a session, all memory allocated for the session is released.

Lasterror is set.

#### Constructor(**FeatureID** as integer, **scope** as string, **VendorCode** as string)

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Logs into a Feature to establish a session, according to predefined search parameters.

**Notes:**

This function is used to specify conditions that describe where login information is to searched for.

The requisite Vendor Codes are stored in a VendorCodes folder in your system. Without the correct Vendor Code, the function call cannot succeed.

You can open up to 512 simultaneous login sessions.

This function does not work with legacy HASP Features.

**FeatureID:** Unique identifier for a specific Feature stored in a Sentinel HASP protection key

**Scope:** Definition of the search parameters for this Feature ID. See the additional HASP API Reference documentation for more information about Scope XML Tags.

**VendorCode:** the vendor code

Lasterror is set.

See also:

- 2.7.1 Constructor(FeatureID as integer, VendorCode as string)

43

### **Constructor(FeatureID as integer, VendorCode as string)**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Logs into a Feature and thereby establishes a session context.

#### **Notes:**

This function establishes a context to a Sentinel HASP protection key containing a license for the requested Feature ID.

The requisite Vendor Codes are stored in a VendorCodes folder in your system. Without the correct Vendor Code, the function call cannot succeed.

You can open up to 512 simultaneous login sessions.

#### Legacy HASP Remarks

For local prognum Features, concurrency is not handled and each login performs a decrement if it is a counting license.

Network "prognum" features continue to use the old HASP LM login logic, with its inherent limitations.

There is only support for concurrent usage of one server (global server address).

With "Program Number" features (see HASP\_ FEATURETYPE\_ MASK), 8 bits are reserved for legacy options (see HASP\_ PROGNUM\_ OPT\_ MASK, currently 5 bits are used):

- \* only local
- only remote
- login is counted per process ID
- disable terminal server check
- enable access to old (HASP3/HASP4) keys

FeatureID: Unique identifier for a specific Feature stored in a Sentinel HASP protection key

VendorCode: the Vendor Code

Lasterror is set.

See also:

- 2.7.1 Constructor(FeatureID as integer, scope as string, VendorCode as string)

42

**DateTimeToHaspTime(day as integer, month as integer, year as integer, hour as integer, minute as integer, second as integer) as memoryblock**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Converts a date and time value to hasptime (the number of elapsed seconds since January 1 1970).

**Notes:**

Time values are in UTC.

Memoryblock has 8 bytes.

**DecryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Decrypts a buffer.

**Notes:**

Decrypts data using the decryption engine in the Sentinel HASP protection key.

The specific session handle determines which Sentinel HASP protection key and which Feature ID decrypts the data buffer. The decryption key remains in the Sentinel HASP protection key. If the decryption fails, the buffer is not modified. To encrypt the data buffer, use the Encrypt function.

Data: The data to decrypt.

DataOffset: Start address in Bytes in the memoryblock.

Size: Data size in Bytes in memoryblock. (16 bytes minimum)

Lasterror is set.

**DecryptString(Data as string) as string**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Decrypts a buffer.  
**Notes:**

Decrypts data using the decryption engine in the Sentinel HASP protection key. The specific session handle determines which Sentinel HASP protection key and which Feature ID decrypts the data buffer. The decryption key remains in the Sentinel HASP protection key. If the decryption fails, the buffer is not modified. To encrypt the data buffer, use the Encrypt function.

Data: The data to decrypt.  
Returns the decrypted data. Lasterror is set.

**Detach(detachAction as string, scope as string, VendorCode as string, recipient as string, byref info as string) as integer**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Detaches or cancels an attached license, according to customizable parameters  
**Notes:**

You do not need to be logged in to a Sentinel HASP Feature in order to use this function. This function is used to detach a license for a Product (i.e. all Sentinel HASP Features and Memory files which belong to this Product) from a HASP SL Protection key. The function returns a H2R file which must then be applied on the recipient machine using hasp\_ update() or the ACC. This function only works with HASP SL Protection Keys; HASP HL Protection Keys are ignored. This function can also be used on the recipient machine to cancel an attached license. In this case, the recipient parameter is ignored and should be set to NULL. For cancelling, the function returns a R2H file which must be applied on the host machine using hasp\_ update() or the ACC. If the detached Product is already expired, no R2H file will be returned. The required Vendor Codes are stored in a VendorCodes folder in your system. Without the correct Vendor Code, the function call cannot succeed.

**EncryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Encrypts a buffer.  
**Notes:**

Encrypts data using the encryption engine in the Sentinel HASP protection key. The specific session handle determines which Sentinel HASP protection key and which Feature ID encrypts

Parameter	Description
detachAction:	Parameters for the operation, in XML format. For more information, see the accompanying Sentinel HASP Run-time API help documentation.
scope:	Search parameters for the Product that is to be detached
VendorCode:	The Vendor Code.
recipient:	Definition in XML format of the recipient computer, on which the detached Product will be installed. This information can be retrieved using either Get-Info or GetSessionInfo together with the format specifier HASP_ RECIPIENT. Set to "" if an attached protection key is cancelled.
info:	The information that is retrieved, in XML format. This information is a V2C, which can then be installed on the recipient computer via Update.

Return code	Description
HASP_ STATUS_ OK	Request was successfully completed
HASP_ INV_ DETACH_ ACTION	Invalid XML "detach_ action" parameter
HASP_ INV_ RECIPIENT	Invalid XML "recipient" parameter
HASP_ TOO_ MANY_ PRODUCTS	Scope for Detach does not specify a unique Parameter
HASP_ TOO_ MANY_ USERS	Too many users currently connected, or: at least one detachable Feature does not have enough network seats available
HASP_ ACCESS_ DENIED	Request cannot be processed due to ACC restrictions
HASP_ FEATURE_ EXPIRED	All detachable Features are expired
HASP_ INV_ PRODUCT	Invalid Product information
HASP_ INV_ DURATION	In the case of a new detachable license, duration exceeds maximum allowed OR, in the case of a detachable license extension, expiration date earlier than original date or too short (if an existing detached Product is extended, and the new expiration date is earlier than the original expiration date)
HASP_ INSUF_ MEM	Out of memory
HASP_ DEVICE_ ERR	Input/Output error in HASP SL secure storage, OR in case of a HASP HL key,USB communication error
HASP_ LOCAL_ COMM_ ERR	Communication error between API and local HASP License Manager
HASP_ REMOTE_ COMM_ ERR	Communication error between local and remote HASP License Manager

the data buffer. The encryption key remains in the Sentinel HASP protection key. If the encryption fails, the buffer is not modified. To decrypt the data buffer, use the Decrypt function.

Data: The data to encrypt.

DataOffset: Start address in Bytes in the memoryblock.

Size: Data size in Bytes in memoryblock. (16 bytes minimum)

### EncryptString(Data as string) as string

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Encrypts a buffer.  
**Notes:**

Encrypts data using the encryption engine in the Sentinel HASP protection key. The specific session handle determines which Sentinel HASP protection key and which Feature ID encrypts the data buffer. The encryption key remains in the Sentinel HASP protection key. If the encryption fails, the buffer is not modified. To decrypt the data buffer, use the Decrypt function.

Data: The data to encrypt. (16 bytes minimum)

Returns encrypted string. Lasterror is set.

**GetInfo(scope as string, format as string, VendorCode as string, byref info as string) as integer**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Retrieves information about system components, according to customizable search parameters, and presents it according to customizable formats.

**Notes:**

You do not need to be logged in to a Sentinel HASP Feature in order to use this function.

This function is used to specify conditions about where to search for information. In addition, it enables you to specify conditions about the format in which the retrieved information is presented. If retrieved information is appropriately formatted, it can be used as a template in the Constructor.

The requisite Vendor Codes are stored in a VendorCodes folder in your system. Without the correct Vendor Code, the function call cannot succeed.

This function cannot be used to retrieve legacy HASP Features.

scope: Definition of the data that is to be searched, in XML format. For more information, see the accompanying Sentinel HASP Run-time API help documentation

format: Definition of the format in which the data is to be displayed, in XML format. For more information, see the accompanying Sentinel HASP Run-time API help documentation.

VendorCode: the Vendor Code

info: The information that is retrieved, in XML format

Return Code	Description
HASP_STATUS_OK	Request was successfully completed
HASP_SCOPE_RESULTS_EMPTY	Unable to locate a Feature matching the scope
HASP_INSUF_MEM	Out of memory
HASP_INV_VCODE	Invalid Vendor Code
HASP_UNKNOWN_VCODE	Vendor Code not recognized
HASP_INVALID_PARAMETER	Scope or format string too long (max. length 32 kb)
HASP_DEVICE_ERR	Input/Output error in HASP SL secure storage, OR in case of a HASP HL key, USB communication error
HASP_LOCAL_COMM_ERR	Communication error between API and local HASP License Manager
HASP_REMOTE_COMM_ERR	Communication error between local and remote HASP License Manager
HASP_INV_FORMAT	Unrecognized format string
HASP_INV_SCOPE	Unrecognized scope string
HASP_BROKEN_SESSION	Session has been interrupted

### GetRTC as memoryblock

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Reads the current time.

#### Notes:

Only HASP HL keys with a Real-time clock (rtc) and HASP SL keys can provide the current time. Primarily used to obtain reliable timestamps that are independent from the system clock. Time values are returned as the number of seconds that have elapsed since Jan-01-1970 0:00:00 UTC. This request is only supported on locally accessed keys. Trying to get the time from a remotely accessed key will return HASP\_NO\_TIME. Returns time as a 8 byte memoryblock. Lasterror is set.

### GetSessionInfo(format as string) as string

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Retrieves information regarding a session context.

#### Notes:

format: Definition for the type of output data structure, in XML format.

There are three format options:

HASP\_KEYINFO For retrieving information on the Sentinel HASP protection key

HASP\_SESSIONINFO For retrieving information on the login session

HASP\_UPDATEINFO For retrieving information on a license update usually contained in a C2V file. The retrieved information includes the current state of the key, including update counters, license availability and memory images

Returns answer. Lasterror is set.

HASP\_KEYINFO is "<haspformat format=""keyinfo""/>"

HASP\_SESSIONINFO is "<haspformat format=""sessioninfo""/>"

HASP\_UPDATEINFO is "<haspformat format=""updateinfo""/>"

### **GetSize(FileID as integer) as integer**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Retrieves the byte size of a memory file from a HASP protection key.

#### **Notes:**

This function is used to determine the file size of a HASP memory file.

FileID: Identifier for the file that is to be queried.

Returns file size. Lasterror is set.

Possible File ID constants: HASP\_FILEID\_LICENSE, HASP\_FILEID\_MAIN, HASP\_FILEID\_RO or HASP\_FILEID\_RW.

### **GetVersion(byref MajorVersion as integer, byref MinorVersion as integer, byref BuildServer as integer, byref BuildNumber as integer, VendorCode as string) as integer**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Retrieves version and build number of the HASP library.

**Notes:** Returns error code.

### **HaspTimeToDateTime(time as memoryblock, byref day as integer, byref month as integer, byref year as integer, byref hour as integer, byref minute as integer, byref second as integer)**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Converts a time value (elapsed seconds since January 1 1970) into a date and time

#### **Notes:**

Memoryblock has 8 bytes.

Time values are in UTC.

**LegacyDecryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP4 compatible decryption function.

**Notes:**

This object must have been created by calling Constructor() with a "prognum" Feature ID.

Data: The memoryblock where bytes are decrypted.

DataOffset is the start position in Bytes in the memoryblock and size the length of the data block to decrypt.

Size: Size of the data in the memoryblock to decrypt.

Lasterror is set.

**LegacyDecryptString(Data as string) as string**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP4 compatible decryption function.

**Notes:**

This object must have been created by calling Constructor() with a "prognum" Feature ID.

Lasterror is set.

**LegacyEncryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP4 compatible encryption function.

**Notes:**

This object must have been created by calling Constructor() with a "prognum" Feature ID.

Data: The memoryblock where bytes are encrypted.

DataOffset is the start position in Bytes in the memoryblock and size the length of the data block to encrypt.

Size: Size of the data in the memoryblock to encrypt.

Lasterror is set.

### **LegacyEncryptString(Data as string) as string**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP4 compatible encryption function.

**Notes:**

This object must have been created by calling Constructor() with a "prognum" Feature ID.

Lasterror is set.

### **LegacySetIdleTime(idletime as integer)**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Set the LM idle time.

**Notes:**

This object must have been created by calling Constructor() with a "prognum" Feature ID.

time: The idle time in minutes.

Lasterror is set.

### **LegacySetRTC(time as memoryblock)**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Writes to HASP4-compatible real-time clock.

**Notes:**

This object must have been created by calling Constructor() with a "prognum" Feature ID.

This request is only supported on locally accessed keys. Attempting to set the time on a remotely accessed

key will return `HASP_NO_TIME`.

time: The new time value (8 byte Memoryblock)

Lasterror is set.

### **LoadLibrary(file as folderitem) as boolean**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Loads the HASPHL shared library.

**Notes:**

Returns true on success and false on failure.

Loads a Windows DLL, a Linux shared library, a Mac OS X shared library or a Mac OS X framework from the given path.

See also:

- 2.7.1 LoadLibrary(path as string) as boolean

52

### **LoadLibrary(path as string) as boolean**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Loads the HASPHL shared library.

**Notes:**

Returns true on success and false on failure.

Path can be an absolute, a relative or just a file name.

Loads a Windows DLL, a Linux shared library, a Mac OS X shared library or a Mac OS X framework from the given path.

See also:

- 2.7.1 LoadLibrary(file as folderitem) as boolean

52

**ReadMemory(FileID as integer, Offset as integer, Size as integer) as Memoryblock**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Reads the memory of a Sentinel HASP protection key.

**Notes:**

Valid File IDs are HASP\_ FILEID\_ RW and HASP\_ FILEID\_ RO.

Legacy HASP Remarks

Valid File IDs are HASP\_ FILEID\_ LICENSE and HASP\_ FILEID\_ MAIN.

FileID: Identifier for the file that is to be read

Offset: Byte offset in the file

Size: Number of bytes to read.

Lasterror is set.

**ReadString(FileID as integer, Offset as integer, Size as integer) as string**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Reads the memory of a Sentinel HASP protection key.

**Notes:**

Valid File IDs are HASP\_ FILEID\_ RW and HASP\_ FILEID\_ RO.

Legacy HASP Remarks

Valid File IDs are HASP\_ FILEID\_ LICENSE and HASP\_ FILEID\_ MAIN.

FileID: Identifier for the file that is to be read

Offset: Byte offset in the file

Size: Number of bytes to read.

Lasterror is set.

**Update(data as string) as string**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Updates a Sentinel HASP protection key.

**Notes:**

This function writes update information. Note that the Sentinel HASP protection key must be locally connected.

The update code contains all necessary data to perform the update on a deployed Sentinel HASP protection key including:

- \* The Sentinel HASP protection key on which the updated information is to be written
- The necessary Vendor Code that is required to access the Sentinel HASP key
- The actual update information

Depending on the update data, the function returns an acknowledgement code that is signed/encrypted by the key. The code is evidence that an update has been applied to a license.

data: The update data.  
Returns the acknowledge data.

Lasterror is set.

**WriteMemory(FileID as integer, FileOffset as integer, Data as Memoryblock, DataOffset as integer, Size as integer)**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Writes to the memory of a Sentinel HASP protection key.

**Notes:**

Valid File ID is HASP\_ FILEID\_ RW.

Depending on the provided session handle (either logged into the default Feature or any other Feature), write access to the FAS memory (HASP\_ FILEID\_ LICENSE) is not permitted.

FileID: Identifier for the file that is to be written  
FileOffset: Byte offset in the file  
Data: the data  
DataOffset: Start address in Bytes in the memoryblock.  
Size: Data size in Bytes in memoryblock.

Lasterror is set.

### **WriteString(FileID as integer, FileOffset as integer, Data as String)**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Writes to the memory of a Sentinel HASP protection key.

**Notes:**

alid File ID is HASP\_ FILEID\_ RW.

Depending on the provided session handle (either logged into the default Feature or any other Feature), write access to the FAS memory (HASP\_ FILEID\_ LICENSE) is not permitted.

FileID: Identifier for the file that is to be written  
FileOffset: Byte offset in the file  
Data: the data

Lasterror is set.

## **2.7.2 Properties**

### **Handle as Integer**

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Internal handle for the session.

**Notes:** (Read and Write property)

### Lasterror as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The last error value.

**Notes:** (Read and Write property)

### 2.7.3 Constants

#### HASP\_ACCESS\_DENIED = 5

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Access to Feature, HASP protection key or functionality denied.

#### HASP\_ALREADY\_LOGGED\_IN = 502

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** C++ API: Logging in twice to the same object.

#### HASP\_ALREADY\_LOGGED\_OUT = 503

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** C++ API: Logging out twice of the same object

#### HASP\_BROKEN\_SESSION = 39

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Session been interrupted.

**HASP\_CLONE\_DETECTED = 64**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Cloned HASP SL secure storage detected.

**HASP\_CONTAINER\_NOT\_FOUND = 7**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Deprecated - use HASP\_HASP\_NOT\_FOUND.

**HASP\_DEFAULT\_FID = 0**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** The HASP default Feature ID.

**Notes:** Available in every HASP key.

**HASP\_DEVICE\_ERR = 43**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Input/Output error occurred in secure storage area of HASP SL key OR a USB error occurred when communicating with a HASP HL key

**HASP\_ENC\_NOT\_SUPP = 23**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Sentinel HASP protection key does not support encryption type.

**HASP\_FEATURETYPE\_MASK = & hfff0000**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP HL Run-time API: "Feature Type" mask.

**Notes:** AND-mask used to identify Feature type.

**HASP\_FEATURE\_EXPIRED = 41**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Feature expired.

**HASP\_FEATURE\_NOT\_FOUND = 31**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Requested Feature not available.

**HASP\_FEATURE\_TYPE\_NOT\_IMPL = 28**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Requested Feature type not implemented.

**HASP\_FILEID\_LICENSE = & hfff2**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the Memory File ID constants

**Notes:**

Legacy HASP HL Run-time API: HASP4 FAS memory file  
(Dummy) File ID for the license data area of memory contents.

**HASP\_FILEID\_MAIN = & hfff0**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the Memory File ID constants

**Notes:**

Legacy HASP HL Run-time API: HASP4 memory file  
File ID for HASP4-compatible memory contents w/o FAS.

**HASP\_FILEID\_RO = & hfff5**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the Memory File ID constants

**Notes:**

Sentinel HASP secure read only memory file.  
File ID for HASP Sentinel HASP read only memory.

**HASP\_FILEID\_RW = & hfff4**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the Memory File ID constants

**Notes:**

Sentinel HASP secure writable memory file  
File ID for Sentinel HASP secure writable memory.

**HASP\_FIRST\_HASP\_ACT = 3001**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Reserved for HASP Activation API

**HASP\_FIRST\_HELPER = 2001**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Reserved for HASP helper libraries.

**HASP\_HARDWARE\_MODIFIED = 52**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** HASP SL key incompatible with machine hardware; HASP SL key is locked to different hardware. OR: In the case of a V2C file, conflict between HASP SL key data and machine hardware data; HASP SL key locked to different hardware

**HASP\_HASP\_NOT\_FOUND = 7**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Sentinel HASP protection key not available.

**HASP\_INCOMPAT\_FEATURE = 6**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Legacy decryption function cannot work on Feature.

**HASP\_INSUF\_MEM = 3**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** System is out of memory.

**HASP\_INT\_ERR = 699**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Internal error occurred in API.

**HASP\_INVALID\_HANDLE\_VALUE = 0**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Invalid handle value for Handle property.

**HASP\_INVALID\_OBJECT = 500**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** C++ API: Object incorrectly initialized.

**HASP\_INVALID\_PARAMETER = 501**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Invalid function parameter.

**HASP\_INV\_API\_DYLIB = 401**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** API dispatcher: Unable to load API; DLL possibly corrupt?

**HASP\_INV\_DETACH\_ACTION = 59**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Invalid XML "action" parameter.

**HASP\_INV\_DURATION = 63**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Invalid duration.

**HASP\_INV\_FILEID = 10**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Specified File ID not recognized by API.

**HASP\_INV\_FORMAT = 15**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Invalid XML format.

**HASP\_INV\_HND = 9**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Invalid login handle passed to function.

**HASP\_INV\_PORT = 651**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Internal use: invalid port value.

**HASP\_INV\_PORT\_TYPE = 650**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Internal use: invalid port type.

**HASP\_INV\_PRODUCT = 61**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Invalid Product information.

**HASP\_INV\_PROGNUM\_OPT = 2**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Legacy HASP HL Run-time API: Unknown/Invalid Feature ID option.

**HASP\_INV\_RECIPIENT = 58**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Invalid XML "recipient" parameter.

**HASP\_INV\_SCOPE = 36**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Invalid XML scope.

**HASP\_INV\_SIG = 30**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Signature verification operation failed.

**HASP\_INV\_SPEC = 35**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Invalid XML specification.

**HASP\_INV\_TIME = 24**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Passed time value outside supported value range.

**HASP\_INV\_UPDATE\_CNTR = 21**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Update counter set incorrectly.

**HASP\_INV\_UPDATE\_DATA = 19**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Required XML tags not found; Contents in binary data are missing or invalid

**HASP\_INV\_UPDATE\_NOTSUPP = 20**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Update request not supported by Sentinel HASP protection key.

**HASP\_INV\_UPDATE\_OBJ = 17**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Binary data passed to function does not contain valid update.

**HASP\_INV\_VCODE = 22**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Invalid Vendor Code passed.

**HASP\_INV\_VLIB = 49**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Unable to load Vendor library.

**HASP\_KEYID\_NOT\_FOUND = 18**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** HASP protection key not found.

**HASP\_LOCAL\_COMM\_ERR = 33**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Communication error between API and local Sentinel HASP License Manager.

**HASP\_MEM\_RANGE = 1**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Request exceeds memory range of a HASP file.

**HASP\_MIN\_BLOCK\_SIZE = 16**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Minimum block size for encrypt and decrypt functions.

**HASP\_MIN\_BLOCK\_SIZE\_LEGACY = 8**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Minimum block size for legacy encrypt and legacy decrypt functions.

**HASP\_NEXT\_FREE\_VALUES = 5001**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Next free error code.

#### **HASP\_NOT\_IMPL = 698**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Requested function not implemented.

#### **HASP\_NO\_ACK\_SPACE = 26**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Acknowledge data requested by update, but ack\_ data parameter is NULL.

#### **HASP\_NO\_API\_DYLIB = 400**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** API dispatcher: API for this Vendor Code was not found.

#### **HASP\_NO\_BATTERY\_POWER = 25**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Real-time clock battery out of power.

#### **HASP\_NO\_DRIVER = 14**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Required driver not installed.

**HASP\_NO\_EXTBLOCK = 600**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Internal use: no classic memory extension block available.

**HASP\_NO\_LOG = 32**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Access log not enabled.

**HASP\_NO\_TIME = 12**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Real-time clock (rtc) not available.

**HASP\_NO\_VLIB = 48**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Unable to find Vendor library.

**HASP\_OLD\_DRIVER = 11**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Installed driver or daemon too old to execute function.

**HASP\_OLD\_LM = 42**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Sentinel HASP License Manager version too old.

**HASP\_OLD\_VLIB = 56**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Vendor library version too old.

**HASP\_OPERATION\_FAILED = 525**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** .NET API: Incorrect use of system or platform

**HASP\_PROGNUM\_DEFAULT\_FID = & hfff0000**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP HL Run-time API: The HASP default Feature ID

**Notes:** Available in every legacy HASP hardware key.

**HASP\_PROGNUM\_FEATURETYPE = & hfff0000**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP HL Run-time API: "Program Number Feature" type

**Notes:** After AND-ing with HASP\_FEATURETYPE\_MASK the Feature type contain this value.

**HASP\_PROGNUM\_MASK = & h000000ff**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP HL  
Run-time API: "Program Number Mask"

**Notes:** AND-mask used to extract the Program Number from a "prognum" Feature ID.

**HASP\_PROGNUM\_OPT\_CLASSIC = & h00001000**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP HL  
Run-time API: "Program Number" option

**Notes:** Enables the API to access "classic" (HASP4 or earlier) keys.

**HASP\_PROGNUM\_OPT\_MASK = & h0000ff00**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP HL  
Run-time API: "Program Number Options" mask.

**Notes:**

AND-mask used to identify Program Number options:

HASP\_PROGNUM\_OPT\_NO\_LOCAL

HASP\_PROGNUM\_OPT\_NO\_REMOTE

HASP\_PROGNUM\_OPT\_PROCESS

HASP\_PROGNUM\_OPT\_CLASSIC

HASP\_PROGNUM\_OPT\_TS

3 bits of the mask are reserved for future extensions and currently unused.

Initialize them with zero.

**HASP\_PROGNUM\_OPT\_NO\_LOCAL = & h00008000**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP HL  
Run-time API: "Program Number" option

**Notes:** Disables the search for local licenses.

**HASP\_PROGNUM\_OPT\_NO\_REMOTE = & h00004000**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP HL  
Run-time API: "Program Number" option  
**Notes:** Disables the search for network licenses.

**HASP\_PROGNUM\_OPT\_PROCESS = & h00002000**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP HL  
Run-time API: "Program Number" option  
**Notes:** Sets session count of network licenses to "per-process".

**HASP\_PROGNUM\_OPT\_TS = & h00000800**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Legacy HASP HL  
Run-time API: "Program Number" option  
**Notes:** Ignores the presence of terminal servers.

**HASP\_REMOTE\_COMM\_ERR = 40**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP  
error codes.  
**Notes:** Communication error between local and remote Sentinel HASP License Managers.

**HASP\_REQ\_NOT\_SUPP = 16**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP  
error codes.  
**Notes:** Unable to execute function in this context; the requested functionality is not implemented.

**HASP\_SCHAN\_ERR = 46**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Communication error occurred in secure channel.

**HASP\_SCOPE\_RESULTS\_EMPTY = 50**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Unable to locate any Feature matching scope.

**HASP\_STATUS\_OK = 0**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Request successfully completed

**HASP\_STORAGE\_CORRUPT = 47**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Corrupt data exists in secure storage area of HASP SL protection key.

**HASP\_SYS\_ERR = 13**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Generic error from host system call.

**HASP\_TIME\_ERR = 45**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** System time has been tampered with.

**HASP\_TMOF = 4**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Too many open Features/login sessions.

**HASP\_TOO\_MANY\_KEYS = 37**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Too many Sentinel HASP protection keys currently connected.

**HASP\_TOO\_MANY\_PRODUCTS = 60**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Scope does not specify a unique Product.

**HASP\_TOO\_MANY\_USERS = 38**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Too many concurrent user sessions currently connected.

**HASP\_TOO\_SHORT = 8**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Encrypted/decrypted data length too short to execute function call.

**HASP\_TS\_DETECTED = 27**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Program running on a terminal server.

**HASP\_UNKNOWN\_ALG = 29**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Unknown algorithm used in H2R/V2C file.

**HASP\_UNKNOWN\_RECIPIENT = 62**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Unknown Recipient; update can only be applied to the Recipient specified in `hasp_detach()`, and not to this computer.

**HASP\_UNKNOWN\_VCODE = 34**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Vendor Code not recognized by API.

**HASP\_UPDATE\_BLOCKED = 44**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Update installation not permitted; This update was already applied.

**HASP\_UPDATE\_TOO\_NEW = 55**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Trying to install a V2C file with an update counter that is out of sequence with update counter in the Sentinel HASP protection key. The first value in the V2C file is greater than the value in the Sentinel HASP protection key.

**HASP\_UPDATE\_TOO\_OLD = 54**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Trying to install a V2C file with an update counter that is out of sequence with the update counter on the Sentinel HASP protection key. The update counter value in the V2C file is lower than the value in sentinel HASP protection key.

**HASP\_UPLOAD\_ERROR = 57**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Upload via ACC failed, e.g. because of illegal format.

**HASP\_USER\_DENIED = 53**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Login denied because of user restrictions.

**HASP\_VM\_DETECTED = 51**

Plugin Version: 11.1 Console & Web: No Mac: Yes, Win: Yes, Linux: Yes, . **Function:** One of the HASP error codes.

**Notes:** Program running on a virtual machine

**2.8 class HASPHLMBS****class HASPHLMBS**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** A class for accessing HASP HL dongles.

**Notes:** HASPHLMBS uses a linked in library. This is okay for older HASP HL versions. But newer versions require you to create your own signed libraries. For them use the HASPHLDMBS class.

**2.8.1 Methods****Close**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** The destructor.

**Notes:**

If you logged in successful before, it logs out from a session and frees all allocated resources for the session. Lasterror is set.

**Constructor(FeatureID as integer, VendorCode as string)**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Login into a feature.

**Notes:**

This function establishes a context (logs into a feature).

feature\_id: Unique identifier of the feature. Combinations of constants below.

vendor\_code: string with the vendor code

For local prognum features, concurrency is not handled and each login performs a decrement if it is a counting license.

Network prognum features just use the old HASPLM login logic with all drawbacks. There is only support for concurrent usage of \b one server (global server address).

Lasterror is set.

```
const HASP_PROGNUM_OPT_NO_LOCAL    =& h00008000  Disable local license search
const HASP_PROGNUM_OPT_NO_REMOTE  =& h00004000  Disable network license search
const HASP_PROGNUM_OPT_PROCESS    =& h00002000  Sets session count of network licenses to per-process
const HASP_PROGNUM_OPT_CLASSIC    =& h00001000  Enables the API to access "classic" (HASP4 or earlier) keys
const HASP_PROGNUM_OPT_TS         =& h00000800  Presence of Terminal Services gets ignored
```

**DateTimeToHaspTime(day as integer, month as integer, year as integer, hour as integer, minute as integer, second as integer) as memoryblock**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Convert broken up time into a time type.

**Notes:**

Lasterror is set.  
Times are in UTC.

**DecryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** This function decrypts a buffer.

**Notes:**

This is the reverse operation of the hasp\_encrypt() function.  
Lasterror is set.

Data: Buffer

DataOffset: Start offset in the buffer.

Size: size in bytes of the buffer to be decrypted (16 bytes minimum)

If the decryption fails (e.g. key removed in-between) the data pointed to by buffer is unmodified.

### **DecryptString(Data as string) as string**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** This function decrypts a buffer.

#### **Notes:**

This is the reverse operation of the `hasp_encrypt()` function.  
This function encrypts a buffer.  
Lasterror is set.

If the decryption fails (e.g. key removed in-between) an empty string is returned.

### **EncryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Encrypt a buffer.

#### **Example:**

```
dim h as HASPHLMBS // your hasp object
dim mem as MemoryBlock // your data

h.EncryptMemory(mem,0,mem.size) // Whole block
```

#### **Notes:**

This function encrypts a buffer.  
Lasterror is set.

Data: Buffer

DataOffset: Start offset in the buffer.

Size: size in bytes of the buffer to be encrypted (16 bytes minimum)

If the encryption fails (e.g. key removed in-between) the data pointed to by buffer is unmodified.

**EncryptString(Data as string) as string**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Encrypt a buffer.

**Notes:**

This function encrypts a buffer.  
Lasterror is set.

If the encryption fails (e.g. key removed in-between) an empty string is returned.

**GetRTC as memoryblock**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Read current time from a time key.

**Notes:**

This function reads the current time from a time key.  
The time will be returned in seconds since Jan-01-1970 0:00:00 GMT.

The general purpose of this function is not related to licensing, but to get reliable timestamps which are independent from the system clock.

This request is only supported on locally accessed keys. Trying to get the time from a remotely accessed key will return `HASP_NO_TIME`.

Returns a 64bit integer inside an eight byte memoryblock.

**GetSessionInfo(format as string) as string**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Get information in a session context.

**Notes:**

format: XML definition of the output data structure

```
const HASP_UPDATEINFO = <haspformat format="updateinfo"/> hasp_get_sessioninfo() format to get update info (C2V)
const HASP_SESSIONINFO = <haspformat format="sessioninfo"/> hasp_get_sessioninfo() format to get session info
const HASP_KEYINFO = <haspformat format="keyinfo"/> hasp_get_sessioninfo() format to get key/hardware info
```

Returns XML data with the requested information.

Calling `GetSessionInfo` with `HASP_ UPDATEINFO` format will return something like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<hasp_ info>
<c2v>
YYIBIIADY3R2oQaABEAwulCiCYABaoEBBIIBAKOBxoABAIGBwD2sfFj8UKuDvNWH9
LhfRKDzUbLCAi6E9mN8ea7EclwOl9VeLMDuLvfsEvkor2igmwxg/wWs6HCuyPEFi6
V/FkI4EUmQNmcKSIY302s9CzHP7aCrG7QKvzArVq25Nc7UxIQJ4kZJm1oWiw3zZJq
UY+G0EleETkPZ8n2uDfMauBpdWhW0R35rHIRM4wiYCZzaelpRtDX36HDh1caqfpaL
mUnwWXRz0+tLs+Dvd+kLmvcQ6jWJb4r2rxywG2IW1WTjIWBSI+h0/UgalhG1J+9R
EQ1SrMx3YQ2bpdIK3FluZVDayW9okv7idxKJS4zGG+4UoccpKT4aWJi9cR0vdm4s/
J6fUNbhK522x/gdvR51a6ll46GpVn2HjD0ZpAgCeuxAIwHJ7Kc6tjeRfxYX9YksE
aB9JoV/uaPTHnbu2AgQmd0r09p0zmXgD4Kuk8EtTs1GoBbY7WF3qHJsj1Iz1ZeAdA
rdNOYKsOgA/q1tuLLR7O0dag
</c2v>
</hasp_ info>
```

Calling `GetSessionInfo` with `HASP_ SESSIONINFO` format will return something like this:

```
<hasp_ info>
<feature>
<featureid>4294905856</featureid>
<maxlogins>5</maxlogins>
<currentlogins>1</currentlogins>
<activations>unlimited</activations>
</feature>
</hasp_ info>
```

In case of a expiring license on a time enabled key (prognum  $\leq 8$ ), instead of the remaining activations the expiration date will be returned:

```
<hasp_ info>
<feature>
<featureid>4294905857</featureid>
<expirationdate>1052919239</expirationdate>
</feature>
</hasp_ info>
```

For locally accessed keys there is no maxlogins and currentlogins field.

Calling GetSessionInfo with HASP\_ KEYINFO format will return something like this for a locally accessed key:

```
<?xml version="1.0" encoding="UTF-8"?>
<hasp_ info>
<keyspec>
<keycaps>
<hasp4/>
<aes/>
<newintf/>
</keycaps>
<hasp>
<haspid>12345</haspid>
<nethasptype>0</nethasptype>
<memoryinfo>
<name>"Main" </name>
<fileid>65520</fileid>
<size>48</size>
</memoryinfo>
<memoryinfo>
<name>"FAS" </name>
<fileid>65522</fileid>
<size>80</size>
</memoryinfo>
</hasp>
<port>
<type>"USB" </type>
<address>1</address>
</port>
</keyspec>
</hasp_ info>
```

Calling GetSessionInfo with HASP\_ KEYINFO format will return something like this for a remotely accessed key:

```
<?xml version="1.0" encoding="UTF-8"?>
<hasp_ info>
<keyspec>
<keycaps>
<hasp4/>
</keycaps>
```

```

<hasp>
<haspid>782062012</haspid>
<nethasptype>5</nethasptype>
<memoryinfo>
<name>"Main"</name>
<fileid>65520</fileid>
<size>432</size>
</memoryinfo>
<memoryinfo>
<name>"FAS"</name>
<fileid>65522</fileid>
<size>448</size>
</memoryinfo>
</hasp>
<serveraddress>
<protocol>"IP"</protocol>
<address>"10.20.3.10"</address>
</serveraddress>
</keyspec>
</hasp_info>

```

keycaps flags:

```

hasp4    support HASP4 compatible encryption
aes      support AES encryption
rtc      key has real time clock chip
newintf  supports new access interface

```

### GetSize(FileID as integer) as integer

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Get memory size.  
**Notes:**

This function is used to determine the memory size.  
Returns the number of bytes inside the file.  
LastError is set.

```

const HASP_ FILEID_ MAIN      = & hfff0  File id for HASP4 compatible memory contents w/o FAS
const HASP_ FILEID_ LICENSE  = & hfff2  (Dummy) file id for license data area of memory contents

```

**HaspTimeToDateTime(time as memoryblock, byref day as integer, byref month as integer, byref year as integer, byref hour as integer, byref minute as integer, byref second as integer)**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Convert time type into broken up time

**Notes:**

Lasterror is set.  
Times are in UTC.

**LegacyDecryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** This function decrypts a buffer.

**Notes:**

This is the reverse operation of the hasp\_ encrypt() function.  
Lasterror is set.

Data: Buffer  
DataOffset: Start offset in the buffer.  
Size: size in bytes of the buffer to be decrypted (16 bytes minimum)

If the decryption fails (e.g. key removed in-between) the data pointed to by buffer is unmodified.

Legacy HASP functionality for backward compatibility.

**LegacyDecryptString(Data as string) as string**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** This function decrypts a buffer.

**Notes:**

This is the reverse operation of the hasp\_ encrypt() function.  
This function encrypts a buffer.  
Lasterror is set.

If the decryption fails (e.g. key removed in-between) an empty string is returned.

Legacy HASP functionality for backward compatibility.

### **LegacyEncryptMemory(Data as Memoryblock, DataOffset as integer, Size as integer)**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Encrypt a buffer.

#### **Example:**

```
dim h as HASPHLMBS // your hasp object
dim mem as MemoryBlock // your data

h.LegacyEncryptMemory(mem,0,mem.size) // Whole block
```

#### **Notes:**

This function encrypts a buffer.  
LastError is set.

Data: Buffer

DataOffset: Start offset in the buffer.

Size: size in bytes of the buffer to be encrypted (16 bytes minimum)

If the encryption fails (e.g. key removed in-between) the data pointed to by buffer is unmodified.

Legacy HASP functionality for backward compatibility.

### **LegacyEncryptString(Data as string) as string**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Encrypt a buffer.

#### **Notes:**

This function encrypts a buffer.  
LastError is set.

If the encryption fails (e.g. key removed in-between) an empty string is returned.

Legacy HASP functionality for backward compatibility.

### **LegacySetIdleTime(idletime as integer)**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Set the LM idle time.

#### **Notes:**

idletime: the idle time in minutes

Legacy HASP functionality for backward compatibility.

### **LegacySetRTC(time as memoryblock)**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Write to HASP4 compatible real time clock

#### **Notes:**

This request is only supported on locally accessed keys. Trying to set the time on a remotely accessed key will return HASP\_NO\_TIME.

Legacy HASP functionality for backward compatibility.

### **ReadMemory(FileID as integer, Offset as integer, Size as integer) as Memoryblock**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** This function is used to read from the key memory.

#### **Notes:**

Lasterror is set.

Offset: Byte start in the file.

Size: Number of bytes in the file.

Result: The data read from the file.

```
const HASP_FILEID_MAIN      = & hfff0  File id for HASP4 compatible memory contents w/o FAS
const HASP_FILEID_LICENSE  = & hfff2  (Dummy) file id for license data area of memory contents
```

### **ReadString(FileID as integer, Offset as integer, Size as integer) as string**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** This function is used to read from the key memory.

#### **Notes:**

Lasterror is set.

Offset: Byte start in the file.

Size: Number of bytes in the file.

Result: The data read from the file.

```
const HASP_FILEID_MAIN      = & hfff0  File id for HASP4 compatible memory contents w/o FAS
const HASP_FILEID_LICENSE  = & hfff2  (Dummy) file id for license data area of memory contents
```

### **Update(data as string) as string**

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Write an update.

#### **Notes:**

This function writes update information. The update blob contains all necessary data to perform the update: Where to write (in which "container", e.g. dongle), the necessary access data (vendor code) and of course the update itself.

If the update blob requested it, the function returns in an acknowledge blob, which is signed/encrypted by the updated instance and contains a proof that this update was successfully installed.

data: string with the complete update data.

Update via LM is not supported.

Lasterror is set.

**WriteMemory**(FileID as integer, FileOffset as integer, Data as Memoryblock, DataOffset as integer, Size as integer)

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Write to key memory.  
**Notes:**

Lasterror is set.

FileOffset: Byte start in the file.

Data: The data to be written to the file.

DataOffset: The byte offset inside the memoryblock.

Size: The number of bytes to be written.

```
const HASP_FILEID_MAIN      = & hfff0  File id for HASP4 compatible memory contents w/o FAS
const HASP_FILEID_LICENSE  = & hfff2  (Dummy) file id for license data area of memory contents
```

**WriteString**(FileID as integer, FileOffset as integer, Data as String)

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Write to key memory.  
**Notes:**

Lasterror is set.

FileOffset: Byte start in the file.

Data: The data to be written to the file.

```
const HASP_FILEID_MAIN      = & hfff0  File id for HASP4 compatible memory contents w/o FAS
const HASP_FILEID_LICENSE  = & hfff2  (Dummy) file id for license data area of memory contents
```

## 2.8.2 Properties

### Handle as Integer

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** The handle of the current session.

**Notes:** (Read and Write property)

### Lasterror as Integer

Plugin Version: 6.1 Console & Web: Yes Mac: Yes, Win: Yes, Linux: No, . **Function:** Last error code reported from one of the functions.

#### Notes:

HASP_STATUS_OK = 0	no error occurred
HASP_MEM_RANGE = 1	invalid memory address
HASP_INV_PROGNUM_OPT = 2	unknown/invalid feature id option
HASP_INSUF_MEM = 3	memory allocation failed
HASP_TMOF = 4	too many open features
HASP_ACCESS_DENIED = 5	feature access denied
HASP_INCOMPAT_FEATURE = 6	incompatible feature
HASP_CONTAINER_NOT_FOUND = 7	license container not found
HASP_TOO_SHORT = 8	en-/decryption length too short
HASP_INV_HND = 9	invalid handle
HASP_INV_FILEID = 10	invalid file id / memory descriptor
HASP_OLD_DRIVER = 11	driver or support daemon version too old
HASP_NO_TIME = 12	real time support not available
HASP_SYS_ERR = 13	generic error from host system call
HASP_NO_DRIVER = 14	hardware key driver not found
HASP_INV_FORMAT = 15	unrecognized info format
HASP_REQ_NOT_SUPP = 16	request not supported
HASP_INV_UPDATE_OBJ = 17	invalid update object
HASP_KEYID_NOT_FOUND = 18	key with requested id was not found
HASP_INV_UPDATE_DATA = 19	update data consistency check failed
HASP_INV_UPDATE_NOTSUPP = 20	update not supported by this key
HASP_INV_UPDATE_CNTR = 21	update counter mismatch
HASP_INV_VCODE = 22	invalid vendor code
HASP_ENC_NOT_SUPP = 23	requested encryption algorithm not supported
HASP_INV_TIME = 24	invalid date / time
HASP_NO_BATTERY_POWER = 25	clock has no power
HASP_NO_ACK_SPACE = 26	update requested acknowledgement, but no area to return it
HASP_TS_DETECTED = 27	terminal services (remote terminal) detected
HASP_FEATURE_TYPE_NOT_IMPL = 28	feature type not implemented
HASP_UNKNOWN_ALG = 29	unknown algorithm
HASP_INV_SIG = 30	signature check failed
HASP_FEATURE_NOT_FOUND = 31	feature not found
HASP_NO_LOG = 32	trace log is not enabled

(Read and Write property)

## 2.9 module MatrixDongleMBS

### module MatrixDongleMBS

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** This module implements methods to access the matrix lock dongles from Technodata Interware.

#### Notes:

Note: You will find the newest versions of the API and tools available for download at [www.tdi-matrix.com](http://www.tdi-matrix.com)

The plugin is from MBS. The library code it is using is based on work from tdi, so if you have trouble, ask both of us.

### 2.9.1 Methods

#### **DongleCount(PortNr as integer) as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Returns the number of dongles available at the specified LPT or USB interface.

**Notes:** Please check the documentation of the dongle API for more details.

#### **DongleDecryptData(UserCode as integer, Data as memoryblock, DongleNr as integer, PortNr as integer) as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Transmit a 8 bytes encrypted data block to the dongle. This is returned decrypted as clear data.

**Notes:** Please check the documentation of the dongle API for more details.

#### **DongleEncryptData(UserCode as integer, Data as memoryblock, DongleNr as integer, PortNr as integer) as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Transmit a 8 bytes clear data block to the dongle. This is returned in XTEA-encrypted form.

**Notes:** Please check the documentation of the dongle API for more details.

#### **DongleExit as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Exits the dongle.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleFind as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Searches for the dongle and returns the LPT/USB interface at which it was found.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleFindEx**(byref LPTNr1 as integer, byref LPTAdr1 as integer, byref DNGCnt1 as integer, byref LPTNr2 as integer, byref LPTAdr2 as integer, byref DNGCnt2 as integer, byref LPTNr3 as integer, byref LPTAdr3 as integer, byref DNGCnt3 as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Searches for all LPT ports and dongles and stores this information in a data buffer.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleGetKeyFlag**(UserCode as integer, DongleNr as integer, PortNr as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Checks whether a 128-bit XTEA key different to zero is available in the dongle.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleMemSize**(DongleNr as integer, PortNr as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Returns the memory size of the dongle in Bytes.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleModel**(DongleNr as integer, PortNr as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Reads the model number of the hardware from the dongle.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleReadData**(UserCode as integer, Data as memoryblock, count as integer, DongleNr as integer, PortNr as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Reads the data from the Matrix-dongle beginning from the first memory field.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleReadDataEx**(UserCode as integer, Data as memoryblock, FPos as integer, count as integer, DongleNr as integer, PortNr as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Reads the data from the Matrix-dongle beginning from the specified memory field.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleReadSerNr**(UserCode as integer, DongleNr as integer, PortNr as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Reads the unique serial number which is assigned to each Matrix-dongle.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleSetLedFlag**(a as integer, b as integer, c as integer, d as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: No, Linux: Yes, . **Function:** Sets the LED flag.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleVersion**(DongleNr as integer, PortNr as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Reads the version number of the dongle.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleWriteData**(UserCode as integer, Data as memoryblock, count as integer, DongleNr as integer, PortNr as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Writes data into the Matrix-dongle beginning from the first memory field.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleWriteDataEx**(UserCode as integer, Data as memoryblock, FPos as integer, count as integer, DongleNr as integer, PortNr as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Writes data into the Matrix-dongle beginning from the specified memory field.

**Notes:** Please check the documentation of the dongle API for more details.

**DongleWriteKey**(UserCode as integer, KeyData as memoryblock, DongleNr as integer, PortNr as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Saves the 128-bit XTEA key in the dongle.

**Notes:** Please check the documentation of the dongle API for more details.

**GetConfigMatrixNet**(Category as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Returns from Server-File parameters configured with the MxNet server program.

**Notes:** Please check the documentation of the dongle API for more details.

**GetDriverFlag**(UserCode as integer, DongleNr as integer, PortNr as integer) as integer

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Reads the current USB operating mode of the dongle HID-Mode or Driver-Mode.

**Notes:** Please check the documentation of the dongle API for more details.

**GetPortAdr(LptNr as integer) as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Returns the address of the LPT port.

**Notes:** Please check the documentation of the dongle API for more details.

**GetVersionAPI as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Returns the version number of the Matrix-API.

**Notes:** Please check the documentation of the dongle API for more details.

**GetVersionDRV as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Returns the version number of the LPT driver.

**Notes:** Please check the documentation of the dongle API for more details.

**GetVersionDRV\_ USB as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Returns the version number of the USB driver.

**Notes:** Please check the documentation of the dongle API for more details.

**InitMatrixAPI as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Starts the Matrix API.

**Notes:** Please check the documentation of the dongle API for more details.

**LogInMatrixNet(UserCode as integer, AppSlot as integer, DongleNr as integer) as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Logs on the network client and acquires/ refreshes the user slot in the server file.

**Notes:** Please check the documentation of the dongle API for more details.

**LogOutMatrixNet(UserCode as integer, AppSlot as integer, DongleNr as integer) as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Logs off the network client and releases the UserSlot in the server file again.

**Notes:** Please check the documentation of the dongle API for more details.

**PausePrinterActivity as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Stops the Windows print-spooler. (under Win9.x / Win-NT / 2000 only)

**Notes:** Please check the documentation of the dongle API for more details.

**ReleaseMatrixAPI as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Close the Matrix API.

**Notes:** Please check the documentation of the dongle API for more details.

**ResumePrinterActivity as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Releases the Windows print-spooler again. (under Win9.x / Win-NT / 2000 only)

**Notes:** Please check the documentation of the dongle API for more details.

**SetConfigMatrixNet(Access as integer, File as string) as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Activates or deactivates network access.

**Notes:** Please check the documentation of the dongle API for more details.

**SetDriverFlag(UserCode as integer, DriverFlag as integer, DongleNr as integer, PortNr as integer) as integer**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Sets the dongle to the desired USB operating mode HID-Mode or Driver-Mode.

**Notes:** Please check the documentation of the dongle API for more details.

**SetW95Access(mode as integer)**

Plugin Version: 9.4 Console & Web: Yes Mac: Yes, Win: Yes, Linux: Yes, . **Function:** Win95/98 with or without VXD driver.

**Notes:**

Can have the value 1 or 2 or the predefined values: IW\_ DRIVER / IW\_ NODRIVER

- 1 Communication takes place via the VXD driver.
- 2 Communication takes place without the VXD driver.

Please check the documentation of the dongle API for more details.



# Chapter 3

## Kagi

### 3.1 class ZKRMKeyValuePairMBS

class ZKRMKeyValuePairMBS

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Holds a key-value pair.

**Notes:**

This class is part of the "Kagi Realbasic Plugin" which is distributed by Kagi.  
<http://www.kagi.com/KRMzonic/>

#### 3.1.1 Properties

**Key as String**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The key for the key-value pair.

**Notes:** (Read and Write property)

### Value as String

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The value for the key-value pair.

**Notes:** (Read and Write property)

## 3.2 Globals

### EncryptKRMStringMBS(byref data as string) as integer

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Encrypt a string for the KRM.

**Notes:**

A string to be encrypted. The string is updated to receive the encrypted form.

String data is encrypted with the Blowfish algorithm, and the encrypted form is returned as a series of alphanumeric hex characters.

This function is part of the "Kagi Realbasic Plugin" which is distributed by Kagi:  
<http://www.kagi.com/KRMzonic/>

### IsKRMAvailableMBS as integer

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Checks to see if the KRM can be used on this system.

**Notes:**

Returns kZKRMMModuleNoErr (0) or kZKRMMModuleUnavailable (10) to indicate if the module is available or not. If kZKRMMModuleUnavailable is returned, no further calls should be made to the KRM.

This function returns -1 on the platforms where KRM library is not available.

This function is part of the "Kagi Realbasic Plugin" which is distributed by Kagi.  
<http://www.kagi.com/KRMzonic/>

## 3.3 class KRMWindowsMBS

### class KRMWindowsMBS

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A class to handle the Kagi Registration Module

**Notes:** The krm.dll is included in the plugin.

### 3.3.1 Methods

#### AddPrice(Currency as string, Price as Double, BaseCurrency as String) as boolean

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** AddPrice adds a price for the product in the given currency.

#### CloseWizard as boolean

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:**

You can close the Wizard at any time from your application by calling the CloseWizard function.

**Notes:** However, once you've received the StatusTransactionStart status code via the event it is not recommended that you close the Wizard prematurely, since the Kagi Store may have already charged the user's credit card. Closing the Wizard will probably prevent you from receiving the license key.

#### Exec as boolean

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Displays the KRM Wizard.

**Notes:**

The user can then enter the purchase details and the purchase is processed by the Kagi server.

Returns true on success and false on failure.

**ExecModal(win as window) as boolean**

Plugin Version: 6.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:**

Same as Exec, but makes the KRM Wizard modal to the window with the given Handle.

**Notes:** See Exec for more information.

**GetLastError as integer**

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Returns an integer value which reflects the last error code of Win.

**Notes:** See the `krmErrorXXXX` constants for details.

**GetResultInt(ResultID as integer, byref Value as integer) as boolean**

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Reads an integer value.

**Notes:**

Constants for ResultID:

```
const krmGetStatusCode      & h0000
const krmGetTestOrder      & h0001
const krmGetTransactionKey & h0010
const krmGetTransactionID  & h0011
const krmGetUserName       & h0012
const krmGetLicenseKey     & h0013
const krmGetOrderXML       & h0020
```

**GetResultStr(ResultID as integer, byref Value as String) as Integer**

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Reads a string value.

**Notes:**

Constants for ResultID:

```

const krmGetStatusCode      & h0000
const krmGetTestOrder      & h0001
const krmGetTransactionKey & h0010
const krmGetTransactionID  & h0011
const krmGetUserName       & h0012
const krmGetLicenseKey     & h0013
const krmGetOrderXML       & h0020

```

### SetDataInt(DataID as integer, Value as integer) as boolean

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Sets an integer value.

#### Notes:

Constants for DataID:

```

const krmSetUserName       & h0001
const krmSetUserCountry   & h0006
const krmSetExemptVAT     & h000A
const krmSetCurrencyCode  & h000B
const krmSetACGVar        & h000D
const krmSetOrderPassThroughData & h000E
const krmSetAffiliateTrackingData & h000F
const krmSetPurchaseOrderData & h0010
const krmSetConnectLimit  & h0020
const krmSetInitURLTimeout & h0021
const krmSetInitDocTimeout & h0022
const krmSetTransactionTimeout & h0023
const krmSetSupplierName  & h0100
const krmSetSupplierID    & h0101
const krmSetSupplierURL   & h0102
const krmSetSupplierEMail & h0103
const krmSetProductDbName & h0110
const krmSetProductPartNo & h0111
const krmSetProductSupplierSKU & h0112
const krmSetProductName   & h0113
const krmSetProductVersion & h0114
const krmSetTransactFailURL & h0120
const krmSetConnectFailURL & h0121
const krmSetLanguage      & h1001
const krmSetShowRegCode   & h1004
const krmSetShowUnlockText & h1005
const krmSetTestOrder     & h1010
const krmSetLanguageXML   & h1011
const krmSetLanguageReceipt & h1012
const krmSetInitializationString & h2000

```

**SetDataStr(DataID as integer, Value as String) as boolean**

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Sets a string value.

**Example:**

```
const theName = "Hello World"
const krmSetACGVar = & h000D

dim k as KRMWindowsMBS // your window
// an example on how to call this method:
call k.SetDataStr krmSetACGVar, "UserName="+theName
```

**Notes:**

Constants for DataID:

```
const krmSetUserName           & h0001
const krmSetUserCountry       & h0006
const krmSetExemptVAT         & h000A
const krmSetCurrencyCode      & h000B
const krmSetACGVar            & h000D
const krmSetOrderPassThroughData & h000E
const krmSetAffiliateTrackingData & h000F
const krmSetPurchaseOrderData & h0010
const krmSetConnectLimit      & h0020
const krmSetInitURLTimeout    & h0021
const krmSetInitDocTimeout    & h0022
const krmSetTransactionTimeout & h0023
const krmSetSupplierName      & h0100
const krmSetSupplierID        & h0101
const krmSetSupplierURL       & h0102
const krmSetSupplierEMail     & h0103
const krmSetProductDbName     & h0110
const krmSetProductPartNo     & h0111
const krmSetProductSupplierSKU & h0112
const krmSetProductName       & h0113
const krmSetProductVersion    & h0114
const krmSetTransactFailURL    & h0120
const krmSetConnectFailURL    & h0121
const krmSetLanguage          & h1001
const krmSetShowRegCode       & h1004
const krmSetShowUnlockText    & h1005
const krmSetTestOrder         & h1010
const krmSetLanguageXML       & h1011
const krmSetLanguageReceipt   & h1012
const krmSetInitializationString & h2000
```

### 3.3.2 Properties

#### Available as Boolean

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** True if the krm.dll has been loaded correctly.

**Notes:** (Read and Write property)

#### BuildNumber as Integer

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The Build Number of the DLL.

**Notes:** (Read and Write property)

#### MajorVersion as Integer

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The Major Version number of the DLL.

**Notes:** (Read and Write property)

#### MinorVersion as Integer

Plugin Version: 6.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The Minor Version number of the DLL.

**Notes:** (Read and Write property)

### 3.3.3 Events

#### StatusChanged(Status as integer, Value as integer, Data as string)

Plugin Version: 6.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** An event called whenever the status changed.

**Notes:**

Constants for Status Codes:

```

const krmStatusError           & h0000
const krmStatusCancel         & h0001
const krmStatusInitSuccess    & h0010
const krmStatusDownloadInitDoc & h0020
const krmStatusInitDocSuccess & h0021
const krmStatusDisplayGUI     & h0030
const krmStatusTransactionStart & h0040
const krmStatusTransactionResult & h0041
const krmStatusTransactionID   & h0042
const krmStatusTransactionKey  & h0043
const krmStatusTestOrder      & h0044
const krmStatusUserName       & h0045
const krmStatusLicenseKey     & h0046
const krmStatusKartOrderXML   & h0047

```

Constants for Order Status Codes:

```

const krmOrderNetworkError     -2000
const krmOrderResponseError    -1000
const krmOrderInvalid          0
const krmOrderDuplicate        1000
const krmOrderFailureNoCharge  2000
const krmOrderFailureCharged   3000
const krmOrderFailureUnknownCharge 4000
const krmOrderOrderApprovalNeeded 4500
const krmOrderSuccess          5000

```

## 3.4 class ZKRMPParametersMBS

**class ZKRMPParametersMBS**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** A class for configuration parameters passed to the Zonic KRM.

**Notes:**

This class is part of the "Kagi Realbasic Plugin" which is distributed by Kagi.  
<http://www.kagi.com/KRMzonic/>

### 3.4.1 Methods

#### AddKeyValuePair(item as ZKRMKeyValuePairMBS)

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Adds a key-value pair.

#### BeginModalKRM(byref result as ZKRMResultMBS) as integer

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Begin a modal KRM session.

##### Example:

```
Const kZKRMLanguageSystem = 0
```

```
Const kZKRMOptionOfferWebOrder = 1
```

```
dim r as ZKRMResultMBS
```

```
dim p as ZKRMPParametersMBS
```

```
dim e as integer
```

```
dim s as string
```

```
StaticText2.text=""
```

```
p=new ZKRMPParametersMBS
```

```
p.AddKeyValuePair NewZKRMKeyValuePairMBS("mykey1","myvalue1")
```

```
p.AddKeyValuePair NewZKRMKeyValuePairMBS("mykey2","myvalue2")
```

```
s= "<krmData vendorID=""UC9"">"
```

```
s=s+ "<transactionFailureURL>http://www.mywebsite.invalid/MyTransactionFailurePage.html</transactionFailureURL>"
```

```
s=s+ "<connectionFailureURL>http://www.mywebsite.invalid/MyConnectionFailurePage.html</connectionFailureURL>"
```

```
s=s+ "<products>"
```

```
s=s+ "<product partNo="" dbName=""KART_TEST_COMPLEX_ACG_UC9"" supplierSKU=""n/a"">"
```

```
s=s+ "<displayName lang=""en"">My Product</displayName>"
```

```
s=s+ "<price currency=""AUD"" basePrice=""100.00"" baseCurrency=""USD""/>"
```

```
s=s+ "<price currency=""CAD"" basePrice=""100.00"" baseCurrency=""USD""/>"
```

```
s=s+ "<price currency=""EUR"" basePrice=""100.00"" baseCurrency=""USD""/>"
```

```
s=s+ "<price currency=""GBP"" basePrice=""100.00"" baseCurrency=""USD""/>"
```

```
s=s+ "<price currency=""JPY"" basePrice=""100.00"" baseCurrency=""USD""/>"
```

```
s=s+ "<price currency=""USD"">100.00</price>"
```

```
s=s+ "</product>"
```

```
s=s+ "</products>"
```

```
s=s+ "</krmData>"
```

```
p.ModuleVersion=0
```

```

p.ModuleLanguage=kZKRMLanguageSystem
p.ModuleOptions=kZKRMOptionOfferWebOrder
p.ModuleUserEmail=""
p.ModuleUserName=""

p.ProductInitXML=s
p.ProductStoreURL="http://order.kagi.com/?UC9"
p.ProductTFYP="Custom string to include in the TFYP email"
p.ProductPO="Custom string to use as a Purchase Order number"

e=p.BeginModalKRM(r)

if e=kZKRModuleNoErr then
ResultWindow.ShowResult r
end if

StaticText2.text=str(e)+" "+GetModuleStatusString(e)

```

**Notes:**

Runs the KRM session through a modal dialog. The application will regain control when the transaction is complete.

The results are placed in a new object which will be stored in the passed result reference. (the result can be nil)

**BeginModelessKRM as integer**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Begin a modeless KRM session.

**Example:**

```

Const kZKRMLanguageSystem = 0
Const kZKRMOptionOfferWebOrder = 1

dim r as ZKRMRResultMBS
// dim p as ZKRMPParametersMBS // make your own subclass to handle event
dim e as integer
dim s as string

StaticText1.text=""

p=new MyZKRMPParametersMBS

```

```

p.AddKeyValuePair NewZKRMKeyValuePairMBS("mykey1","myvalue1")
p.AddKeyValuePair NewZKRMKeyValuePairMBS("mykey2","myvalue2")

s= "<krmData vendorID=""UC9"">"
s=s+ "<transactionFailureURL>http://www.mywebsite.invalid/MyTransactionFailurePage.html</transactionFailureURL>"
s=s+ "<connectionFailureURL>http://www.mywebsite.invalid/MyConnectionFailurePage.html</connectionFailureURL>"
s=s+ "<products>"
s=s+ "<product partNo="" dbName=""KART_TEST_COMPLEX_ACG_UC9"" supplierSKU=""n/a"">"
s=s+ "<displayName lang=""en"">My Product</displayName>"
s=s+ "<price currency=""AUD"" basePrice=""100.00"" baseCurrency=""USD"" />"
s=s+ "<price currency=""CAD"" basePrice=""100.00"" baseCurrency=""USD"" />"
s=s+ "<price currency=""EUR"" basePrice=""100.00"" baseCurrency=""USD"" />"
s=s+ "<price currency=""GBP"" basePrice=""100.00"" baseCurrency=""USD"" />"
s=s+ "<price currency=""JPY"" basePrice=""100.00"" baseCurrency=""USD"" />"
s=s+ "<price currency=""USD"">100.00</price>"
s=s+ "</product>"
s=s+ "</products>"
s=s+ "</krmData>"

p.ModuleVersion=0
p.ModuleLanguage=kZKRMLanguageSystem
p.ModuleOptions=kZKRMOptionOfferWebOrder
p.ModuleUserEmail=""
p.ModuleUserName=""

p.ProductInitXML=s
p.ProductStoreURL="http://order.kagi.com/?UC9"
p.ProductTFYP="Custom string to include in the TFYP email"
p.ProductPO="Custom string to use as a Purchase Order number"

e=p.BeginModelessKRM

StaticText1.text=str(e)+" "+GetModuleStatusString(e)

```

**Notes:**

Runs the KRM session through a modeless dialog. The application will regain control immediately, with a preliminary status result.

The result is given with the Complete Event being called.

If kZKRMMModuleNoErr is returned, a Complete Event will be dispatched to the application later.

**ClearKeyValuePair**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Clears the list of key-value pairs.

**GetKeyValuePair(index as integer) as ZKRMKeyValuePairMBS**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Returns a key-value pair.

**Notes:**

Returns nil on any error.  
Index is 0 based.

**KeyValuePairCount as integer**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Counts all key-value pairs.

**Notes:** Returns 0 on any error.

**Version as integer**

Plugin Version: 5.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The version of this structure.

### 3.4.2 Properties

**ModuleLanguage as Integer**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The language to be used by the module user interface.

**Notes:**

useful constants:

kZKRMLanguageSystem	=0	The user interface will attempt to match the user's preferred language for the system. If this language is not supported by the Zonic KRM, kZKRMLanguageEnglishIntl will be used.
kZKRMLanguageEnglishUS	=10	The user interface will be displayed in US English.
kZKRMLanguageEnglishIntl	=20	The user interface will be displayed in International English.
kZKRMLanguageJapanese	=30	The user interface will be displayed in Japanese.
kZKRMLanguageGerman	=40	The user interface will be displayed in German.
kZKRMLanguageFrench	=50	The user interface will be displayed in French.
kZKRMLanguageItalian	=60	The user interface will be displayed in Italian.

(Read and Write property)

### ModuleOptions as Integer

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Options to control the module.

#### Notes:

Should be set to kZKRMOptionDefault if the default behaviour is required.

Useful constants:

kZKRMOptionDefault	=0	The KRM should use the default behaviour for all options.
kZKRMOptionOfferWebOrder	=1	If set, the KRM will offer the user a chance to purchase through the Kagi Web Store before presenting the normal KRM user interface. This option can be used for products that can be purchased in a manner not supported by the KRM (e.g., cash or bulk orders), and where the application has not invoked the KRM through a "Order a single copy now with a credit card" user interface element. If the KRM has been invoked through a generic "Register Now" user interface element, this option ensures that users can still reach the Web Store if required.
kZKRMOptionEncryptedProductXML	=2	If set, the productInitXML field of the ZKRMPParametersMBS structure is decrypted before use. This option can be set if the productInitXML field has been encrypted with a call to EncryptKRMString.

(Read and Write property)

**ModuleUserEmail as String**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Optional user email address to use as the default for the purchase window.

**Notes:**

If this field is "", the user email address will be obtained from the system.  
(Read and Write property)

**ModuleUserName as String**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Optional user name to use as the default for the purchase window.

**Notes:**

If this field is "", the user name will be obtained from the system.  
(Read and Write property)

**ProductAffiliate as String**

Plugin Version: 5.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** A product affiliate code to pass to the Kagi Transaction Server.

**Notes:** (Read and Write property)

**ProductInitXML as String**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Product initialization XML.

**Notes:**

This string is passed to the Kagi Registration Server, which uses it to construct an appropriate order description for the KRM.  
(Read and Write property)

**ProductPO as String**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Optional product-specific purchase order number.

**Notes:**

This string is passed unchanged through the Kagi Registration Server, and included in both the final TFYP email and any printed receipt.

(Read and Write property)

**ProductStoreURL as String**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Product store URL.

**Notes:**

If moduleOptions contains kZKRMOptionOfferWebOrder, the user will be offered the chance to bypass the KRM and visit this URL in their browser.

(Read and Write property)

**ProductTFYP as String**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** Optional product-specific "TFYP" data.

**Notes:**

This string is passed unchanged through the Kagi Registration Server, and included in both the final TFYP email and any printed receipt.

(Read and Write property)

**3.4.3 Events****Completed(result as ZKRMResultMBS)**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The event called on a modeless KRM session.

## 3.5 class ZKRMResultMBS

### class ZKRMResultMBS

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** A class for results returned by the Zonic KRM.

**Notes:**

This class is part of the "Kagi Realbasic Plugin" which is distributed by Kagi.  
<http://www.kagi.com/KRMzonic/>

### 3.5.1 Properties

#### acgRegCode as String

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The registration code returned by the Kagi Automatic Code Generation system.

**Notes:**

If this product does not use an ACG, "" is returned.  
(Read and Write property)

#### acgUserName as String

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The user name returned by the Kagi Automatic Code Generation system.

**Notes:**

If this product does not use an ACG, "" is returned.  
(Read and Write property)

#### KagiReplyXML as String

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The reply XML returned by the Kagi server.

**Notes:** (Read and Write property)

### **KagiTransactionID as String**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The Kagi Transaction ID for the order.

**Notes:** (Read and Write property)

### **ModuleStatus as Integer**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The module result returned by the Zonic KRM.

**Notes:**

If this field is anything other than kZKRMMModuleNoErr, the remaining fields are undefined.

Useful constants:

kZKRMMModuleNoErr	=0	No error.
kZKRMMModuleUnavailable	=10	The KRM is not available on this platform.
kZKRMMModuleInProgress	=20	A transaction is already underway.
kZKRMMModuleBadResponse	=30	The data returned by the Kagi server was malformed.
kZKRMMModuleUserCancelled	=40	The user has cancelled the operation.
kZKRMMModuleUsedWebStore	=50	The user has used the Kagi web store rather than the KRM.
kZKRMMModuleConnectionLost	=60	The connection to the server was lost.
kZKRMMModuleOutOfMemory	=70	The module was unable to allocate memory.

(Read and Write property)

### **ModuleVersion as Integer**

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The module version.

**Notes:**

This value is set by Realbasic internally.

(Read and Write property)

### OrderStatus as Integer

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The order result returned by the Kagi server.

**Notes:**

Constants:

kZKRMOrderInvalid	=0	The transaction could not be completed because there was a protocol disagreement between Zonic KRM and the Kagi server.
kZKRMOrderDuplicate	=1000	The transaction was not processed because it was identified as a duplicate of a previous order. No charge was placed against the credit card for this transaction.
kZKRMOrderFailureNoCharge	=2000	There was a failure while processing the transaction. This status includes the possibility of the sale not being authorized. No charge was placed against the credit card for this transaction.
kZKRMOrderFailureWithCharge	=3000	There was a failure while processing the transaction. The failure was of a nature which did not prevent sale authorization, but which possibly prevents the generation of user name/registration code data. A charge will be placed against the credit card for this transaction, and any user name/registration code data will be e-mailed to the address supplied by the user.
kZKRMOrderFailureChargeStatusUnknown	=4000	There was a failure while processing the transaction. The failure was of a nature which does not make it possible to immediately determine whether or not a charge was placed against the credit card for this transaction.
kZKRMOrderHumanApprovalRequired	=4500	The order has neither been approved or denied. Before the order can be approved or denied, it is necessary for a human to review the order.
kZKRMOrderSuccess	=5000	The transaction was successful. A charge will be placed against the credit card for this transaction, and any relevant user name/registration code data has been generated and returned.

(Read and Write property)

### OrderType as Integer

Plugin Version: 4.3 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** The type of order processed by the Kagi server.

**Notes:**

Some constants:

kZKRMTTypeTest	=0	The order was a test order.
kZKRMTTypeReal	=1	The order was a real order.

(Read and Write property)

# Chapter 4

## List of all classes

• HASPHLDMBS	41
• HASPHLMBS	76
• KRMWindowsMBS	99
• Rockey2MBS	16
• Rockey4NDMBS	13
• RockeyMBS	37
• UnikeyMBS	24
• ZKRMKeyValuePairMBS	97
• ZKRMParametersMBS	104
• ZKRMResultMBS	112



## Chapter 5

### List of all modules

- MatrixDongleMBS 88
- SecureDongleXMBS 27



## Chapter 6

# List of all global methods

- 2.6 CallHASPMBs(service as integer, seed as integer, lptnum as integer, pass1 as integer, pass2 as integer, byref p1 as integer, byref p2 as integer, byref p3 as integer, byref p4 as integer) 40
- 2.6 CallHASPMemMBS(service as integer, seed as integer, lptnum as integer, pass1 as integer, pass2 as integer, byref p1 as integer, byref p2 as integer, byref p3 as integer, byref p4 as integer, mem as memoryblock) 40
- 3.2 EncryptKRMStringMBS(byref data as string) as integer 98
- 2.6 GetHASPErrStrMBS(error as integer) as string 41
- 2.6 GetNetHaspWarningStrMBS(error as integer) as string 41
- 3.2 IsKRMAvailableMBS as integer 98