

sdi_emv.h

[Go to the documentation of this file.](#)

```
1Â // -*- Mode: C++ -*-      2Â #ifndef SDI_EMV_H_20200406      3Â #define SDI_EMV_H_20200406
<utility>
// for std::pair      12Â      13Â #if defined _WIN32 && defined VFI_SDICLIENT_EXPON
<40,EMVAppFlowCapabilityBits>
EMVAppFlowCapabilities;      25Â      27Â typedef BitMask<
40,EMVKernelAppFlowCapabilityBits>
EMVKernelAppFlowCapabilities;      28Â      30Â typedef BitMask<
80,EMVFlowOptionBits> EMVFlowOptions;      31Â      33Â typedef BitMask<
40,EMVChecksumOptionBits> EMVChecksumOptions;      34Â      36Â typedef BitMask<
<40,EMVTransactionOptionBits>
EMVTransactionOptions;      37Â      39Â typedef BitMask<
24,EMVTransactionStepBits>
EMVTransactionSteps;      40Â      41Â typedef BitMask<
24,EMVDomesticOptionBits>
EMVDomesticOptions;      42Â      43Â      45Â class EMVSpecialTransactions: public
<8> {      46Â public:      48Â      EMVSpecialTransactions():Array<8>
() {}      51Â      EMVSpecialTransactions(const unsigned char val[8]) { set(val); }
<<4); }      58Â      EMVSpecialReversal getReversal() const { return
EMVSpecialReversal(v[0]>
>4 & 0xf); }      59Â      62Â      void setRefund(EMVSpecialRefund r) { v[0]=(v[0]&0xf)|
<<4); }      72Â      EMVSpecialReservation getReservation() const { return
EMVSpecialReservation(v[1]>
>4 & 0xf); }      73Â      76Â      void setTip(EMVSpecialTip r) { v[1]=(v[1]&0xf0)|(u
<<4); }      86Â      EMVSpecialReferral getReferral() const { return
EMVSpecialReferral(v[2]>
>4 & 0xf); }      87Â      90Â      void setVoice(EMVSpecialVoice r) { v[2]=(v[2]&0xf0)|
<<4); }      107Â      EMVSpecialIgnoreCard getIgnoreCard() const { return
EMVSpecialIgnoreCard(v[4]>
>4 & 0xf); }      108Â };      109Â      113Â      116Â struct DllSpec EMVConf: public TLV {
<unsigned char>
bytevector;      120Â      121Â protected:      122Â      bool exists(unsigned tag1) const
<4> &id);      136Â      bool getIssuerID(Array<4>
&id) const;      137Â      140Â      bool setAID(const bytevector &aid);      142Â      templ
<unsigned N>
bool setAID(const unsigned char (&v)[N])      143Â      { return setAID(bytevector(v,v+
<6> &acq);      188Â      bool getAcquirerID(Array<6>
&acq) const;      189Â      192Â      bool setChipAppVersionNumber(const std::vector
<unsigned short> &version);      194Â      template<unsigned N>
bool setChipAppVersionNumber(const unsigned short (&v)[N])      195Â      { return se
<unsigned short>
(v,v+N)); }      198Â      bool getChipAppVersionNumber(std::vector<unsigned
```

```

short>
    &version) const; 199Â      200Â      203Â      bool setMerchantCategory(const Array
<2> &category); 206Â      bool getMerchantCategory(Array<2>
    &category) const; 207Â      210Â      bool setMerchantID(const std::string &mid);
<6>
    &limit); 227Â      bool setFloorLimit(int64_t limit); 230Â      bool getFloorLimit
<6>
    &limit) const; 233Â      bool getFloorLimit(int64_t &limit) const; 234Â      237Â
<8> &id); 240Â      bool getTerminalID(Array<8>
    &id) const; 241Â      244Â      bool setTerminalRiskManagement(const Array<8>
    &trm); 247Â      bool getTerminalRiskManagement(Array<8>
    &trm) const; 248Â      251Â      bool setIFDSerial(const Array<8>
    &serial); 254Â      bool getIFDSerial(Array<8>
    &serial) const; 255Â      258Â      bool setTerminalCapabilities(const Array<
3> &caps); 261Â      bool getTerminalCapabilities(Array<3>
    &caps) const; 262Â      265Â      bool setTerminalType(unsigned char type); 268Â
<5> &caps); 282Â      bool getAdditionalCapabilities(Array<5>
    &caps) const; 283Â      286Â      bool setMerchantNameLocation(const std::string &
<3> &icc); 303Â      bool getInterchangeProfile(Array<3>
    &icc) const; 304Â      307Â      bool setMerchantType(unsigned char type); 310Â
<3> &info); 317Â      bool getTerminalTransactionInformation(Array<3>
    &info) const; 318Â      321Â      bool setAppProgramID(const bytevector &id); 323
<unsigned N>
    bool setAppProgramID(const unsigned char (&v)[N]) 324Â      { return setAppProgra
<8> &id); 334Â      bool getDataStoreRequestedOperatorID(Array<8>
    &id) const; 335Â      338Â      bool setReceiptRequiredLimit(const BCD<6>
    &limit); 341Â      bool setReceiptRequiredLimit(int64_t limit); 344Â      bool ge
<6>
    &limit) const; 347Â      bool getReceiptRequiredLimit(int64_t &limit) const; 34
<2> &status); 354Â      bool getOptionStatus(Array<2>
    &status) const; 355Â      358Â      bool setTransactionQualifier(const Array<
4> &q); 361Â      bool getTransactionQualifier(Array<4>
    &q) const; 362Â      365Â      bool setMSRVersionNumber(const std::vector<
unsigned short> &ver); 367Â      template<unsigned N>
    bool setMSRVersionNumber(const unsigned short (&v)[N]) 368Â      { return setMSRV
<unsigned short>(v,v+N)); } 371Â      bool getMSRVersionNumber(std::vector<
unsigned short>
    &ver) const; 372Â      375Â      bool setReaderCapabilities(unsigned char caps);
<4> &caps); 385Â      bool getTransactionCapabilities(Array<4>
    &caps) const; 386Â      389Â      bool setTerminalTransactionData(const bytevector
<unsigned N>
    bool setTerminalTransactionData(const unsigned char (&v)[N]) 392Â      { return
<20> &data); 402Â      bool getMerchantCustomData(Array<20>
    &data) const; 403Â      406Â      bool getLibVersion(std::string &version) const;
<5>
    &cp); 417Â      bool setEMVCOChecksumParameter(const EMVChecksumOptions &cp) 41
<5>
    &cp) const; 424Â      bool getEMVCOChecksumParameter(EMVChecksumOptions &cp) con
<3> &mid); 450Â      bool getFallbackMIDs(Array<3>
    &mid) const; 451Â      454Â      bool setEMVFallback(unsigned char fb); 457Â      b
<bytevector> &apps); 473Â      bool getPriorityApplications(std::vector<
bytevector>
    &apps) const; 474Â      477Â      bool setASI(unsigned char asi); 480Â      bool ge
<5> &tac); 487Â      bool getTACDenial(Array<5>
    &tac) const; 488Â      491Â      bool setTACOnline(const Array<5>
    &tac); 494Â      bool getTACOnline(Array<5>
    &tac) const; 495Â      498Â      bool setTACDefault(const Array<5>

```

```

    &tac); 501Â    bool getTACDefault(Array<5>
    &tac) const; 502Â    505Â    bool setThreshold(unsigned th); 508Â    bool get
<1>
    &p); 515Â    bool setMaxPercentOnline(int p); 518Â    bool getMaxPercentOnline
<1>
    &p) const; 521Â    bool getMaxPercentOnline(int &p) const; 522Â    525Â    bo
<1>
    &p); 528Â    bool setPercentOnline(int p); 531Â    bool getPercentOnline(BCD
<1>
    &p) const; 534Â    bool getPercentOnline(int &p) const; 535Â    538Â    bool
<unsigned N>
    bool setDefaultTDOL(const unsigned char (&v)[N]) 541Â    { return setDefaultTD
<unsigned N>
    bool setDefaultDDOL(const unsigned char (&v)[N]) 551Â    { return setDefaultDD
<unsigned N>
    bool setAdditionalTags(const unsigned char (&v)[N]) 561Â    { return setAdditi
<unsigned> &tags); 570Â    template<unsigned N>
    bool setMandatoryTags(const unsigned (&v)[N]) 571Â    { return setMandatoryTags
<unsigned>(v,v+N)); } 574Â    bool getMandatoryTags(std::vector<unsigned>
    &tags) const; 575Â    578Â    bool setAppFlowCapabilities(const Array<5>
    &caps); 581Â    bool setAppFlowCapabilities(const EMVAppFlowCapabilities &caps
<5>
    &caps) const; 588Â    bool getAppFlowCapabilities(EMVAppFlowCapabilities &caps
<unsigned N>
    bool setAdditionalTagsCRD(const unsigned char (&v)[N]) 596Â    { return setAdd
<5> &caps); 627Â    bool getAdditionalTerminalCapabilities(Array<5>
    &caps) const; 628Â    631Â    bool setCDAProcessing(unsigned char cda); 634Â
<3> &caps); 656Â    bool getCapabilitiesBelowLimit(Array<3>
    &caps) const; 657Â    660Â    bool setBeepVolume(unsigned short volume); 663Â
<unsigned N>
    bool setSupportedLanguages(const unsigned char (&v)[N]) 681Â    { return setSup
<unsigned N>
    bool setTagsToRead(const unsigned char (&v)[N]) 729Â    { return setTagsToRead
<6>
    &limit); 795Â    bool setTransactionLimitNoCVMOnDevice(int64_t limit); 798Â
<6>
    &limit) const; 801Â    bool getTransactionLimitNoCVMOnDevice(int64_t &limit) c
<6>
    &limit); 808Â    bool setTransactionLimitCVMOnDevice(int64_t limit); 811Â    l
<6>
    &limit) const; 814Â    bool getTransactionLimitCVMOnDevice(int64_t &limit) cons
<3>
    &time); 835Â    bool setMessageHoldTime(int time); 838Â    bool getMessageHold
<3>
    &time) const; 841Â    bool getMessageHoldTime(int &time) const; 842Â    845Â
<unsigned N>
    bool setPhoneMessageTable(const unsigned char (&v)[N]) 855Â    { return setPhon
<10>
    &opt); 907Â    bool setFlowOptions(const EMVFlowOptions &opt) 908Â    { return
<10>
    &opt) const; 914Â    bool getFlowOptions(EMVFlowOptions &opt) const 915Â    {
<3>
    &id); 929Â    bool setKernelID(unsigned id); 932Â    bool getKernelID(Array
<3>
    &id) const; 935Â    bool getKernelID(unsigned &id) const; 936Â    939Â    boo
<3>
    &id); 957Â    bool setInternalKernelID(unsigned id); 960Â    bool getInternal

```

```

<3>
&id) const; 963Â    bool getInternalKernelID(unsigned &id) const; 964Â    967
<5>
&caps); 977Â    bool setKernelAppFlowCapabilities(const EMVKernelAppFlowCapabi
<5>
&caps) const; 984Â    bool getKernelAppFlowCapabilities(EMVKernelAppFlowCapabi
<6>
&limit); 992Â    bool setCtlsTransactionLimit(int64_t limit); 995Â    bool ge
<6>
&limit) const; 998Â    bool getCtlsTransactionLimit(int64_t &limit) const; 99
<6>
&limit); 1005Â    bool setCVMRequiredLimit(int64_t limit); 1008Â    bool getCVM
<6>
&limit) const; 1011Â    bool getCVMRequiredLimit(int64_t &limit) const; 1012Â
<6>
&t); 1018Â    bool setRiskManagementThreshold(int64_t t); 1021Â    bool getRisk
<6>
&t) const; 1024Â    bool getRiskManagementThreshold(int64_t &t) const; 1025Â
<1>
&p); 1031Â    bool setRiskManagementTargetPercentage(int p); 1034Â    bool getR
<1>
&p) const; 1037Â    bool getRiskManagementTargetPercentage(int &p) const; 1038Â
<1>
&p); 1044Â    bool setRiskManagementMaxTargetPercentage(int p); 1047Â    bool g
<1>
&p) const; 1050Â    bool getRiskManagementMaxTargetPercentage(int &p) const; 10
<6>
&limit); 1064Â    bool setTransactionLimitCash(int64_t limit); 1067Â    bool ge
<6>
&limit) const; 1070Â    bool getTransactionLimitCash(int64_t &limit) const; 107
<2> &opt); 1077Â    bool getCombinationOptions(Array<2>
&opt) const; 1078Â    1079Â    1082Â    bool setRemovalTimeout(unsigned short tou
<2> &interval); 1099Â    bool getTornTransactionInterval(Array<2>
&interval) const; 1100Â    1103Â    bool setCtlsKernelCapabilities(const Array
<5> &caps); 1106Â    bool getCtlsKernelCapabilities(Array<5>
&caps) const; 1107Â    1110Â    bool setMTOL(const bytevector &mtol); 1112Â    t
<unsigned N>
bool setMTOL(const unsigned char (&v)[N]) 1113Â    { return setMTOL(bytevector(v
<3> &tcap); 1130Â    bool getTermcapAboveCVMLimit(Array<3>
&tcap) const; 1131Â    1134Â    bool setTermcapBelowCVMLimit(const Array<3>
&tcap); 1137Â    bool getTermcapBelowCVMLimit(Array<3>
&tcap) const; 1138Â    1141Â    bool setTACSwitch(const Array<5>
&sw); 1144Â    bool getTACSwitch(Array<5>
&sw) const; 1145Â    1148Â    bool setIACSwitch(const Array<5>
&sw); 1151Â    bool getIACSwitch(Array<5>
&sw) const; 1152Â    1155Â    bool setDataExchangeTags(const bytevector &tags);
<unsigned N>
bool setDataExchangeTags(const unsigned char (&v)[N]) 1158Â    { return setData
<6>
&limit); 1189Â    bool setsetCtlsTransactionLimitODCVM(int64_t limit); 1192Â
<6>
&limit) const; 1195Â    bool getsetCtlsTransactionLimitODCVM(int64_t &limit) co
<3> &type); 1202Â    bool getTransactionTypesSale(Array<3>
&type) const; 1203Â    1206Â    bool setTransactionTypesCash(const Array<3>
&type); 1209Â    bool getTransactionTypesCash(Array<3>
&type) const; 1210Â    1213Â    bool setTransactionTypesCashback(const Array
<3> &type); 1216Â    bool getTransactionTypesCashback(Array<3>

```

```

&type) const; 1217Â    1220Â    bool setBeepFreqSuccess(unsigned short freq); 1223Â
<3> &type); 1230Â    bool getTransactionTypesRefund(Array<3>
&type) const; 1231Â    1234Â    bool setBeepFreqAlert(unsigned short freq); 1237Â
<unsigned N>
bool setHotlist(const unsigned char (&v)[N]) 1263Â    { return setHotlist(byteve
<6> floorLimit; 1292Â    BCD<6>
transactionLimit; 1293Â    BCD<6>
CVMRequiredLimit; 1294Â    unsigned char options; 1295Â    #if __cplusplus
<6> &floorLimit_, 1298Â    const BCD<6>
&transactionLimit_, 1299Â    const BCD<6>
&CVMRequiredLimit_, 1300Â    unsigned char options_) 1303Â
<DynamicReaderLimits>
&limits); 1315Â    bool getDynamicReaderLimits(std::vector<
DynamicReaderLimits>
&limits) const; 1316Â    1318Â    // issuer specific 1319Â    1322Â    bool setTra
<6>
&limit); 1332Â    bool setFloorLimitIK(int64_t limit); 1335Â    bool getFloorLim
<6>
&limit) const; 1338Â    bool getFloorLimitIK(int64_t &limit) const; 1339Â    1342Â
<6>
&limit); 1345Â    bool setCVMRequiredLimitPK(int64_t limit); 1348Â    bool getCV
<6>
&limit) const; 1351Â    bool getCVMRequiredLimitPK(int64_t &limit) const; 1352Â
<6>
&limit); 1358Â    bool setFloorlimitPK(int64_t limit); 1361Â    bool getFloorlin
<6>
&limit) const; 1364Â    bool getFloorlimitPK(int64_t &limit) const; 1365Â    1368Â
<6>
&limit); 1371Â    bool setFloorLimitMIR(int64_t limit); 1374Â    bool getFloorLI
<6>
&limit) const; 1377Â    bool getFloorLimitMIR(int64_t &limit) const; 1378Â    1381Â
<6>
&limit); 1384Â    bool setTransactionLimitNoCVM MIR(int64_t limit); 1387Â    boo
<6>
&limit) const; 1390Â    bool getTransactionLimitNoCVM MIR(int64_t &limit) const;
<6>
&limit); 1397Â    bool setTransactionLimitNoCDCVMMIR(int64_t limit); 1400Â    b
<6>
&limit) const; 1403Â    bool getTransactionLimitNoCDCVMMIR(int64_t &limit) cons
<6>
&limit); 1410Â    bool setTransactionLimitCDCVMMIR(int64_t limit); 1413Â    boo
<6>
&limit) const; 1416Â    bool getTransactionLimitCDCVMMIR(int64_t &limit) const;
<2> &caps); 1423Â    bool getTPMCapabilitiesMIR(Array<2>
&caps) const; 1424Â    1427Â    bool setTransactionRecoveryLimitMIR(unsigned cha
<unsigned N>
bool setAdditionalVersionNumbers(const unsigned char (&v)[N]) 1437Â    { return
<5> &tac); 1450Â    bool getTACDefaultMK(Array<5>
&tac) const; 1451Â    1454Â    bool setTACDenialMK(const Array<5>
&tac); 1457Â    bool getTACDenialMK(Array<5>
&tac) const; 1458Â    1461Â    bool setTACOnlineMK(const Array<5>
&tac); 1464Â    bool getTACOnlineMK(Array<5>
&tac) const; 1465Â    1468Â    bool setFloorLimitMK(const BCD<6>
&limit); 1471Â    bool setFloorLimitMK(int64_t limit); 1474Â    bool getFloorLin
<6>
&limit) const; 1477Â    bool getFloorLimitMK(int64_t &limit) const; 1478Â    1481Â
<6>

```

```

    &limit); 1484Â    bool setCVMRequiredLimitMK(int64_t limit); 1487Â    bool getCV
<6>
    &limit) const; 1490Â    bool getCVMRequiredLimitMK(int64_t &limit) const; 1491Â
<unsigned char>
    bytevector; 1508Â    1509Â    protected: 1510Â    bool exists(unsigned tag1) const
<4>
    &id) const; 1522Â    1525Â    bool setAID(const bytevector &aid); 1527Â    templ
<unsigned N>
    bool setAID(const unsigned char (&v)[N]) 1528Â    { return setAID(bytevector(v,
<10> &pan) const; 1544Â    1547Â    bool getAIP(Array<2>
    &aip) const; 1548Â    1551Â    bool getDFName(bytevector &name) const; 1552Â    1
<6> &code); 1562Â    bool getAuthCode(Array<6>
    &code) const; 1563Â    1566Â    bool setAuthResponseCode(const Array<2>
    &auth); 1569Â    bool getAuthResponseCode(Array<2>
    &auth) const; 1570Â    1573Â    bool getCVMList(Array<28>
    &list) const; 1574Â    1577Â    bool getTVR(Array<5>
    &tvr) const; 1578Â    1581Â    bool setTransactionDate(const Array<3>
    &date); 1584Â    bool setTransactionDate(const std::string &date); 1587Â    boo
<3>
    &date) const; 1590Â    bool getTransactionDate(std::string &date) const; 1591Â
<2>
    &status) const; 1595Â    1598Â    bool setTransactionType(unsigned char type); 1
<3> &date); 1633Â    bool getExpirationDate(Array<3>
    &date) const; 1634Â    1637Â    bool getEffectiveDate(Array<3>
    &date) const; 1640Â    bool getEffectiveDate(std::string &date) const; 1641Â
<2> &code) const; 1670Â    1673Â    bool getPANSequenceNumber(BCD<1>
    &num) const; 1676Â    bool getPANSequenceNumber(int &num) const; 1677Â    1680Â
<6>
    &amount); 1697Â    bool setAmount(int64_t amount); 1700Â    bool getAmount(BCD
<6>
    &amount) const; 1703Â    bool getAmount(int64_t &amount) const; 1704Â    1707Â
<6>
    &amount); 1710Â    bool setCashbackAmount(int64_t amount); 1713Â    bool getCas
<6>
    &amount) const; 1716Â    bool getCashbackAmount(int64_t &amount) const; 1717Â
<2> &version) const; 1725Â    1728Â    bool getChipAppVersionNumber(Array<2>
    &version) const; 1729Â    1732Â    bool getASRPD(bytevector &asrpd) const; 1733Â
<5> &iac) const; 1741Â    1744Â    bool getIACDenial(Array<5>
    &iac) const; 1745Â    1748Â    bool getIACOnline(Array<5>
    &iac) const; 1749Â    1752Â    bool getIssuerAppData(bytevector &idata) const; 1
<6>
    &limit); 1775Â    bool setFloorLimit(int64_t limit); 1778Â    bool getFloorLimi
<6>
    &limit) const; 1781Â    bool getFloorLimit(int64_t &limit) const; 1782Â    1785Â
<8> &serial) const; 1786Â    1789Â    bool setTransactionTime(const Array<3>
    &time); 1792Â    bool setTransactionTime(const std::string &time); 1795Â    boo
<3>
    &time) const; 1798Â    bool getTransactionTime(std::string &time) const; 1799Â
<8> &crypt); 1805Â    bool getCryptogram(Array<8>
    &crypt) const; 1806Â    1809Â    bool getCryptogramInfo(unsigned char &info) con
<3> &caps) const; 1814Â    1817Â    bool getCVMResults(Array<3>
    &cvmr) const; 1818Â    1821Â    bool getTerminalType(unsigned char &type) const;
<2>
    &atc) const; 1826Â    1829Â    bool setUnpredictableNumber(unsigned num); 1832Â
<5> &caps) const; 1841Â    1844Â    bool setTransactionCounter(const BCD<4>
    &counter); 1847Â    bool setTransactionCounter(int counter); 1850Â    bool getT
<4>

```

```

    &counter) const; 1853Â    bool getTransactionCounter(int &counter) const; 1854Â
<2>
    &code) const; 1865Â    1868Â    bool getICCDynamicNumber(bytevector &dn) const;
<6>
    &amount) const; 1879Â    bool getAvailableOfflineAmount(int64_t &amount) const;
<4> &q) const; 1884Â    1887Â    bool setVisaCTQ(const Array<2>
    &ctq); 1890Â    bool getVisaCTQ(Array<2>
    &ctq) const; 1891Â    1894Â    bool setMasterAID(const bytevector &aid); 1896Â
<unsigned N>
    bool setMasterAID(const unsigned char (&v)[N]) 1897Â    { return setMasterAID(by
<Array<5>
    > &results) const; 1916Â    1919Â    bool getCriticalScriptResults(std::vector
<Array<5>
    > &results) const; 1920Â    1923Â    bool setFallbackMIDs(const Array<3>
    &mid); 1926Â    bool getFallbackMIDs(Array<3>
    &mid) const; 1927Â    1930Â    bool setEMVFallback(unsigned char fb); 1933Â    b
<5> &tac); 1948Â    bool getTACDenial(Array<5>
    &tac) const; 1949Â    1952Â    bool setTACOnline(const Array<5>
    &tac); 1955Â    bool getTACOnline(Array<5>
    &tac) const; 1956Â    1959Â    bool setTACDefault(const Array<5>
    &tac); 1962Â    bool getTACDefault(Array<5>
    &tac) const; 1963Â    1966Â    bool setThreshold(unsigned th); 1969Â    bool get
<6>
    &amount) const; 1997Â    bool getBalanceBefore(int64_t &amount) const; 1998Â
<5>
    &opt); 2018Â    bool setTransactionOptions(const EMVTransactionOptions &opt) 20
<5>
    &opt) const; 2025Â    bool getTransactionOptions(EMVTransactionOptions &opt) co
<3>
    &step); 2033Â    bool setTransactionSteps(const EMVTransactionSteps &step) 2034
<3>
    &step) const; 2040Â    bool getTransactionSteps(EMVTransactionSteps &step) cons
<6>
    &amount) const; 2048Â    bool getBalanceAfter(int64_t &amount) const; 2049Â    2
<3> &info); 2055Â    bool getDCCInfo(Array<3>
    &info) const; 2056Â    2059Â    bool setFallbackMSROptions(unsigned char opt); 2
<unsigned N>
    bool setAuthData(const unsigned char (&v)[N]) 2151Â    { return setAuthData(byt
<unsigned N>
    bool setCriticalScript(const unsigned char (&v)[N]) 2161Â    { return setCritica
<unsigned N>
    bool setNonCriticalScript(const unsigned char (&v)[N]) 2171Â    { return setNon
<2> &ac); 2181Â    bool getAuthResponseReferral(Array<2>
    &ac) const; 2182Â    2185Â    bool setAuthResponseSwitch(const Array<2>
    &resp); 2188Â    bool getAuthResponseSwitch(Array<2>
    &resp) const; 2189Â    2192Â    bool setAuthResponseWrongPIN(const Array<2>
    &ac); 2195Â    bool getAuthResponseWrongPIN(Array<2>
    &ac) const; 2196Â    2199Â    bool setAuthResponseOnlinePIN(const Array<2>
    &resp); 2202Â    bool getAuthResponseOnlinePIN(Array<2>
    &resp) const; 2203Â    2206Â    bool setAdditionalResponseOK(const Array<2>
    &ac); 2209Â    bool getAdditionalResponseOK(Array<2>
    &ac) const; 2210Â    2213Â    bool setWriteDSData(const bytevector &dsdata); 221
<3> &info); 2244Â    bool getDomesticInfo(Array<3>
    &info) const; 2245Â    2248Â    bool setDomesticOptions(const Array<3>
    &opt); 2251Â    bool setDomesticOptions(const EMVDomesticOptions &opt) 2252Â
<3>
    &opt) const; 2258Â    bool getDomesticOptions(EMVDomesticOptions &opt) const 22

```

<15>

```
&error) const; 2271Â    2274Â    bool getDisplayTextIndex(unsigned char &id) const;
<3> &bm); 2281Â    bool getVelocityProcessingResultBitmap(Array<3>
&bm) const; 2282Â    2285Â    bool getKernelDebug(bytevector &debug) const; 2286
<bytevector> &aid); 2292Â    bool getExcludeAID(std::vector<bytevector>
&aid) const; 2293Â    2301Â    bool setExcludeCombos(const std::vector<
std::pair<bytevector, Array<3>
> > &combos); 2302Â    2305Â    bool setCardholderInfo(unsigned char info); 2308
<6>
&amount); 2336Â    bool setStoredAmount(int64_t amount); 2339Â    bool getStored
<6>
&amount) const; 2342Â    bool getStoredAmount(int64_t &amount) const; 2343Â    23
<unsigned N>
bool setReducedCanditateList(const unsigned char (&v)[N]) 2356Â    { return set
<unsigned N>
bool setPINPublicModulus(const unsigned char (&v)[N]) 2387Â    { return setPINP
<3> &kernel); 2418Â    bool getKernelToUse(Array<3>
&kernel) const; 2419Â    2422Â    bool setCommunicationError(unsigned char err);
<unsigned N>
bool setDataExchangeDataNeeded(const unsigned char (&v)[N]) 2457Â    { return s
<unsigned N>
bool setModifiedCandidates(const unsigned char (&v)[N]) 2467Â    { return setMo
<5>
&result) const; 2475Â    2478Â    bool setCBDDisplayTextIndex(unsigned char idx);
<20> &id) const; 2486Â    2489Â    bool getTrackStatus(Array<3>
&track_status) const; 2490Â    2493Â    bool get0bfuscatedTrack1String(std::stri
<3>
&id); 2504Â    bool setKernelID(unsigned id); 2507Â    bool getKernelID(Array
<3>
&id) const; 2510Â    bool getKernelID(unsigned &id) const; 2511Â    2514Â    boo
<3>
&id); 2524Â    bool setApplicationKernelId(unsigned id); 2527Â    bool getAppli
<3>
&id) const; 2530Â    bool getApplicationKernelId(unsigned &id) const; 2531Â    25
<DomesticApp> &app); 2596Â    bool getDomesticApps(std::vector<DomesticApp>
&app) const; 2597Â    2599Â    struct FallbackMSR { 2600Â    unsigned char mi
<FallbackMSR> &mid); 2621Â    bool getFallbackMSR(std::vector<FallbackMSR>
&mid) const; 2622Â    2624Â    struct Candidate { 2625Â    bytevector aid;
<Candidate> &cand); 2639Â    bool getCandidateList(std::vector<Candidate>
&cand) const; 2640Â    2642Â    struct CandidateExt { 2643Â    bytevector aid
<CandidateExt> &cand); 2671Â    bool getCBCandidateList(std::vector<
CandidateExt>
&cand) const; 2672Â    2675Â    bool setDataExchangeReceivedData(const TLV &data
```