

Purpose of Application: CRYPTO-BOX[®] Driver Installation at end-user sites

Version: Smarx OS PPK 5.75 and higher

Last Update: 23 September 2016

Target Operating Systems: Windows 10/8/7/Vista (32 & 64 bit)

Target Processor Platforms: Intel x86

Applicable for Product: CRYPTO-BOX[®] SC / XS / Versa

Convenient Distribution of Protected Applications

After selecting the protection and licensing strategy that best suits your needs; protecting your software and formatting the CRYPTO-BOX hardware, it's now time to send everything to the end user. At this point it is important to include the CRYPTO-BOX drivers with your product. This document describes how to include the CRYPTO-BOX drivers smoothly into your software package.

Quick and efficient hardware based software protection!

Software and information piracy costs billions of dollars in annual losses to software vendors, distributors and content providers worldwide. The internet's role in software and data distribution is growing rapidly and increases the importance of the situation dramatically. Hardware based protection can be used for creating robust and reliable secure demo versions of applications in a straightforward manner. While benefiting from strong and effective application protection provided by a hardware based approach you can create flexible and secure demo versions with ease. The CRYPTO-BOX makes it happen!



CRYPTO-BOX[®]

- Quick and easy protection of Windows and Linux applications with AutoCrypt
- Individual implementations with API for all common programming languages, incl. .NET
- The CRYPTO-BOX system can be customized according to individual requirements
- Multi-platform support: Windows, Linux and macOS
- Unique and stable metal case, optional with customer-specific color and labeling
- Internal secure memory of 4-64 kB
- Network and remote update capability
- AES/Rijndael encryption on-chip
- RSA support on-chip (CRYPTO-BOX SC) or on driver level (CRYPTO-BOX XS/Versa)



Table of Contents

- 1. CRYPTO-BOX driver installation with CBUSetup.....3
- 2. Syntax and Exit Codes.....3
- 3. Installation script sample for InstallShield.....4
- 4. CBUSetup as Windows Installer Merge Module.....4
- 5. CRYPTO-BOX Network Server Installation.....5

1. CRYPTO-BOX driver installation with CBUSetup

The CRYPTO-BOX requires device drivers to work under Windows operating systems. MARX provides a special program called CBUSetup for this task.

CBUSetup performs automatic installation of CRYPTO-BOX-related components, depending on the operating system. The installer is a self-extracting tool containing all files needed to support the CRYPTO-BOX SC, XS or Versa, thus it is self-sufficient. You can use the “Quiet” installation mode. This feature allows you to integrate the setup into other proprietary installation scripts or batch files (see chapter 2 for more details).



We strongly recommend to install the drivers using CBUSetup before plugging the CRYPTO-BOX into the USB port of the computer for the first time.

CBUSetup.exe performs the following tasks:

- Detects the installed Windows version and installs the corresponding CRYPTO-BOX drivers (32 or 64 bit).
- Installs and registers SMRXCOM COM objects (32 or 64 bit) needed by protected applications using COM interface to interact with the CRYPTO-BOX.
- Copies legacy MPI library (mpiw32.dll) to Windows\system32 folder, required by obsolete protected applications still using the legacy MPI system.

The following operating systems are supported by CBUSetup: Windows 10/8/7/Vista/XP



Check our website www.marx.com → Support → Downloads → Driver and Diagnostic Tools to get the latest version of CBUSetup.

2. Syntax and Exit Codes

CBUSetup supports command line switches, allowing you to fully control CBUSetup from within other applications. The following command line options are supported:

CBUSetup.exe /?	Get a help screen
CBUSetup.exe /Q /CRYPTOKEN	Run setup in quiet mode and install CRYPTO-BOX driver
CBUSetup.exe /Q /U /CRYPTOKEN	Perform “quiet” uninstall of CRYPTO-BOX driver
CBUSetup.exe /Q /CRYPTOKEN /DRIVERONLY	Install the driver without any additional files (the SMRXCOM ActiveX COM object will not be installed)

As already mentioned, you can run the setup in “quiet” or in normal mode. The “quiet” setup is useful when you wish to redistribute the setup and include it in your proprietary installations. By analyzing the return value of the setup, the proprietary installation can find out if CBUSetup.exe was executed successfully.

The following is a list of CBUSetup exit codes:

Exit Code	Description
0	No errors occurred
-1	The installer could not detect your operating system
-2	User aborted installation
-3	Wrong version of the operating system
-4	Setup internal error - please, contact your distributor
-5	User has requested the installer to run in the "quiet" mode but the device was specified incorrectly. Run the installer with a help switch (/?) to see the command line syntax.
-6	User does not have enough rights to install the driver. Contact your System Administrator
-7	The installer could not perform registration of device driver(s). Please run the installer again, and if this error code still appears, contact your distributor.

3. Installation script sample for InstallShield

The following is a sample of a CRYPTO-BOX installation script for InstallShield (without exit code check):

```
function InstDrivers()
    string szPath,szTargetFile;
begin
    szPath=TARGETDIR^"\Drivers\CBUSetup\CBUSetup.exe";
    szTargetFile=WINSYSDIR^"\CBUSetup.exe";
    CopyFile (szPath, szTargetFile);
    LongPathToQuote (szTargetFile,TRUE);
    LaunchAppAndWait (szTargetFile, "/Q /CRYPTOKEN", WAIT);
end;

function UnInstDrivers()
    string szTargetFile
begin
    szTargetFile=WINSYSDIR^"\CBUSetup.exe";
    LongPathToQuote (szTargetFile,TRUE);
    LaunchAppAndWait (szTargetFile, "/Q /U /CRYPTOKEN", WAIT);
    DeleteFile (szTargetFile);
end;
```

4. CBUSetup as Windows Installer Merge Module

CBUSetup is available as Windows Installer Merge Module (CBUSetup.msm), which can be incorporated into existing Windows Installer or InstallShield setup for your application. This component performs automatic installation of CRYPTO-BOX driver and related components, depending on the operating system found (32 or 64 bit Windows system).



The latest version of CBUSetup.msm can be downloaded from our website:

www.marx.com → Support → Downloads → Driver and Diagnostic Tools

If you want to use another approach for CRYPTO-BOX driver installation and you need the driver as single files (.inf and .sys), you can download them from our website as well.

5. CRYPTO-BOX Network Server Installation

Smarx OS Networking allows Smarx OS based applications to access a CRYPTO-BOX attached to the USB port of any computer in a network. A special program called **Smarx OS Network Server** (or CBIOS Network Server), manages remote connections to this CRYPTO-BOX. More details on Network License Management with the CRYPTO-BOX can be found in the [Smarx Compendium](#) chapter 5.



If your protected application does not support network licensing (CRYPTO-BOX will be accessed on the local computer only), installation of the network server is not required.

The CRYPTO-BOX Network Server is available for different platforms:

- Windows 32 and 64 bit versions (Windows XP and up)
- Linux 32 and 64 bit
- macOS



Check our website www.marx.com → Support → Downloads → Network Utilities to get the latest version of the CRYPTO-BOX Network Server for your platform.

For the Windows platform, the server is provided as Windows Installer setup (.msi) and Windows Installer Merge Module (.msm) for 32 or 64 bit. See section "Network License Control and User Limits" in the PPK Control Center for more information.





The Windows Installer setup (.msi) also installs the CRYPTO-BOX device driver during installation, so it is not required to run CBUSetup separately.

During installation, you can select if you want to start the server as a service under Windows. This can be changed later if required, see [Smarx Compendium](#) chapter 5 for details.



For more information on administration of the network server see [Smarx Compendium](#) chapter 5.

CRYPTO-BOX® Data Sheet

	CRYPTO-BOX SC (CBU SC)	CRYPTO-BOX XS/Versa (CBU XS/Versa)
		
Controller chip	RISC Smart Card Processor with USB Interface	RISC Smart Card Processor with USB Interface
Chip certification	EAL4+	EAL4+
Supported operating systems	Windows, Linux, macOS, iOS, Android	Windows, Linux, macOS, iOS, Android
In hardware implemented algorithms	AES 128 Bit, RSA (up to 2048 Bit key length), others (for example: ECC) on request	AES 128 Bit in hardware, RSA up to 2048 Bit key length on driver level
Memory size (complete)	72KByte, ca. 32KByte free	4, 32 or 64 KByte
Internal memory read/write performance	ca. 80kByte/s	ca. 12kByte/s
Password (PIN/PUK)	up to 16 Byte length	
Case & LED	Designer metal housing, cast zinc, with LED display of operating status, eye for key ring/lanyard	
Connector	USB Type A	
Memory programming	minimum 100,000 write cycles	
Data retention time	minimum 10 years	
Conformity & Certifications	FCC, CE, RoHS, WEEE, USB logo	
Dimensions	14 x 7 x 32,5 mm / 0.55" x 0.28" 1.28"	14 x 7 x 32,5 mm / 0.55" x 0.28" 1.28"
Weight	7,5g	7,5g
Temperature	-10°C to +70°C / 14°F to 158°F	
Humidity	0% to 95% relative humidity	

CRYPTO-BOX Certifications



All brands, trademarks and registered trademarks are the property of their respective owners.

Evaluation Kit

www.marx.com/eval

MARX Software Security GmbH

Vohburger Strasse 68
85104 Wackerstein, Germany
Phone: +49 (0) 8403 / 9295-0
Fax: +49 (0) 8403 / 9295-40
contact-de@marx.com

www.marx.com

MARX CryptoTech LP

489 South Hill Street
Buford, GA 30518 U.S.A.
Phone: (+1) 770 904 0369
Fax: (+1) 678 730 1804
contact@marx.com