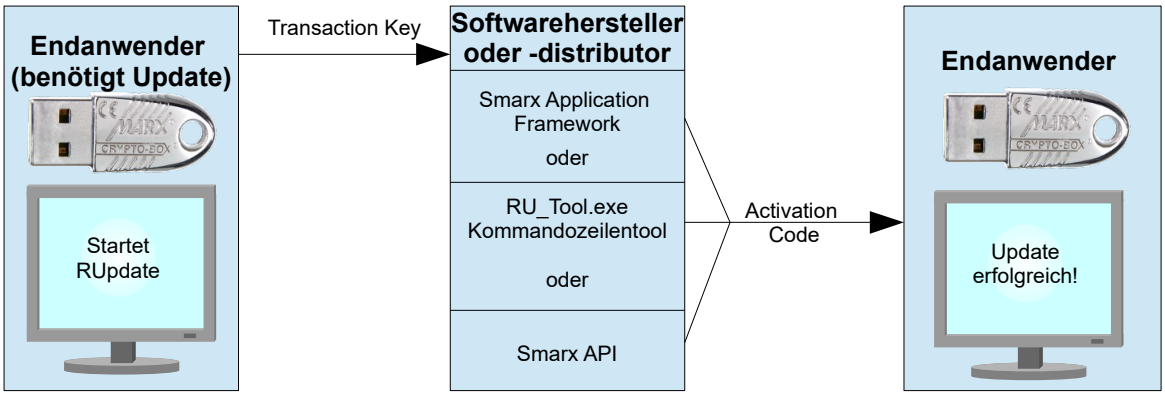
Remote Update (RUMS) AN-4

Inhalt: Fernupdate der CRYPTO-BOX mit dem Remote Update Management System (RUMS)
Version: Smarx®OS PPK 5.75 oder höher
Zuletzt geändert: 15 November 2016 von Steffen Kaetsch
Zielbetriebssystem: Windows
Zielplattform: x86
Unterstützte Produkte: CRYPTO-BOX® SC / XS / Versa

Fern-Aktualisierung der CRYPTO-BOX

Das Remote Update Management System (RUMS) bietet Softwareherstellern oder -distributoren einen komfortablen Weg, die CRYPTO-BOX direkt beim Endanwender zu aktualisieren. So können zum Beispiel Ablauffristen für Test- oder Demoversionen verlängert, Lizenzierungsoptionen modifiziert oder bestimmte Funktionen aktiviert bzw. deaktiviert werden. Der Aktualisierungsvorgang findet hier über das Versenden verschlüsselter Konfigurationsdateien (z.B. per Email) statt.



Remote Update bietet folgende Möglichkeiten zum Update der CRYPTO-BOX:

- Es können bestehende Lizenzen aktualisiert und erweitert, sowie der Rijndael Private und der Rijndael Session Key der CRYPTO-BOX umprogrammiert werden
- Ein Update ist ohne Programmierkenntnisse über die grafische Oberfläche des Smarx Application Framework möglich
- Alternativ sind Kommandozeilentools verfügbar, die von anderen Anwendungen oder über Batchdateien gesteuert werden können.
- Darüber hinaus haben Entwickler mit dem RFP API die Möglichkeit, die Update-Funktionalität direkt in ihre Anwendungen zu integrieren

Table of Contents

1. Remote Update Management System (RUMS): Überblick

Das Remote Update Management System (RUMS) ist Bestandteil des Smarx OS Protection Kits für die CRYPTO-BOX. Es bietet Softwareherstellern oder -distributoren einen komfortablen Weg, eine bereits beim Endanwender installierte CRYPTO-BOX zu aktualisieren. Ein Hin- und Herschicken der CRYPTO-BOX entfällt somit. Beispielsweise können Ablauffristen für Test- oder Demoversionen verlängert, Lizenzierungsoptionen modifiziert oder bestimmte Funktionen aktiviert bzw. deaktiviert werden. Die Updates werden dabei durch das Versenden verschlüsselter Konfigurationsdateien (z.B. per Email) abgewickelt, eine direkte Internetverbindung ist nicht erforderlich.

Remote Updates lassen sich auf verschiedene Arten durchführen:

- Über das Remote Update Management System (RUMS), einen Bestandteil des Smarx Application Framework (siehe Kapitel 2);
- Über RU_Tool.exe, ein kommandozeilenbasiertes Tool (siehe Kapitel 3);
- Über SxFormat.exe, ein kommandozeilenbasiertes Tool (siehe Kapitel 4), bei welchem im Gegensatz zu RU_Tool.exe keine Übersendung von Daten durch den Endanwender notwendig ist;
- Über das RFP API (Remote Update API, siehe Kapitel 5).

Die jeweiligen Vorteile der o.g. Update-Arten werden am Beginn der einzelnen Kapitel erläutert, um Ihnen eine einfache Entscheidung für das passende Verfahren zu ermöglichen.

Remote Update bietet folgende Funktionen (ab Smarx OS Protection Kit 5.75):

- Update bestehender oder Erstellen neuer Datenobjekte innerhalb vorhandener oder neuer Partitionen, dazu zählen z.B. Ablaufdatum, Zähler, Netzwerklizenzen oder individuelle Speicherobjekte.
- Erstellung neuer Partitionen im Speicher der CRYPTO-BOX oder Erweiterung existierender Partitionen.
- Update des Rijndael Session und Rijndael Private Key der CRYPTO-BOX.

Remote Update ist optional erhältlich (einmalige Freischaltung erforderlich). Bitte wenden Sie sich an Ihren MARX-Distributor oder bestellen Sie direkt auf unserer Webseite.

2. Remote Update mit dem Smarx® Application Framework

2.1 Überblick

Das Smarx Application Framework (SxAF) ermöglicht den einfachen Schutz von Programmen oder Dokumenten über eine einheitliche Oberfläche. Software und Dokumente können über die Projekttypen "AutoCrypt" oder "Document Protection" ohne Programmieraufwand geschützt werden. Aber auch bei der manuellen Einbindung über API ist das SxAF eine große Hilfe: die CRYPTO-BOXen lassen sich einfach und schnell passend für die einzelnen Kunden mit den gewünschten Lizenzinformationen konfigurieren. Weitere zum SxAF finden Sie im Smarx Compendium, Kapitel 4.

Vorteile von Remote Update mit dem Smarx Application Framework:

- Einfache Erstellung von Projekten und Durchführung von Remote Updates über eine grafische Oberfläche ohne Programmieraufwand.
- Integrierte Kundendatenbank.
- Einfache Übersicht der bestehenden Projekte und der bisher durchgeführten Updates.

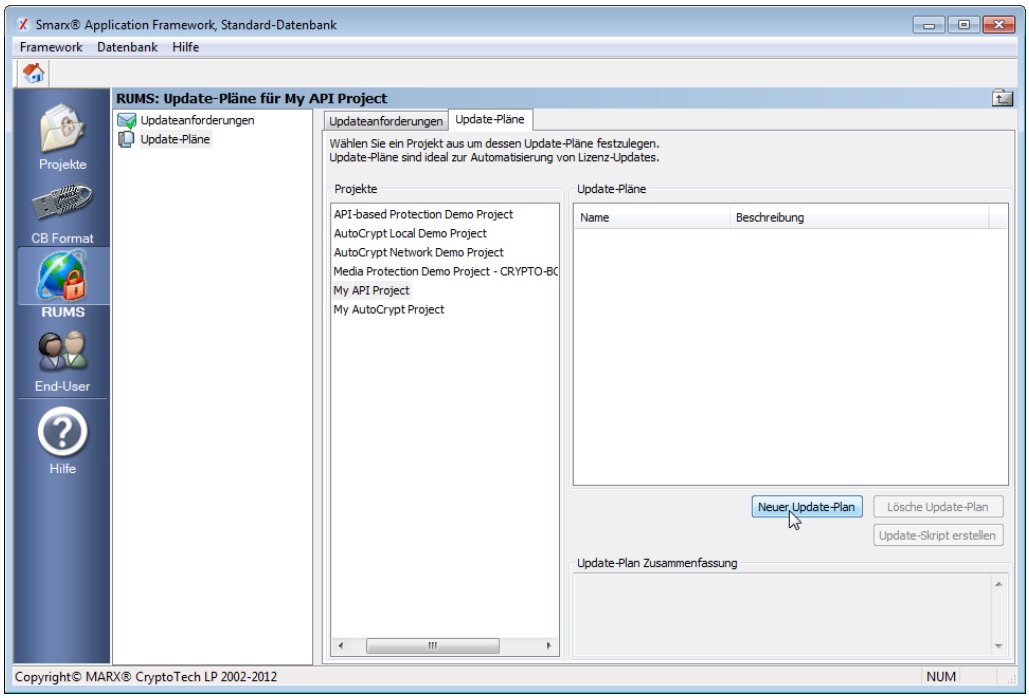
Besonderheit:

- Eigene Datenbank, keine Anbindung an bereits bestehende Kundendatenbanken möglich, falls schon vorhanden.
- Update von CRYPTO-BOXen nur möglich, wenn diese bereits mit einem im SxAF vorhandenen Projekt formatiert wurden.

2.2 Update-Pläne

Mit RUMS können Sie passend zu Ihrer Lizenzierungsstrategie Update-Pläne für Ihr Projekt erstellen. Das erleichtert Ihnen die Arbeit insbesondere bei wiederkehrenden Update-Vorgängen: Sie brauchen nicht jedes Mal die Update-Parameter neu zu spezifizieren. Sie können beliebig viele Update-Pläne erstellen. Zusammen mit dem Produkt-Edition Feature können Sie so eine Vereinfachung und Automatisierung Ihrer Lizenzstrategie erreichen.

Die Erstellung von Update-Plänen ist nicht unbedingt notwendig, sie können Updates natürlich auch manuell durchführen. Allerdings sind sie eine große Hilfe bei wiederkehrenden Updatevorgängen.



Um einen Update-Plan zu erstellen, starten Sie das Smarx Application Framework (SxAF Client im Startmenü) und wählen Sie in der vertikalen Menüleiste links "RUMS" aus. Klicken Sie dann auf den Reiter „Update-Pläne“. Wählen Sie anschließend links das Projekt aus, für welches Sie den Update-Plan erstellen

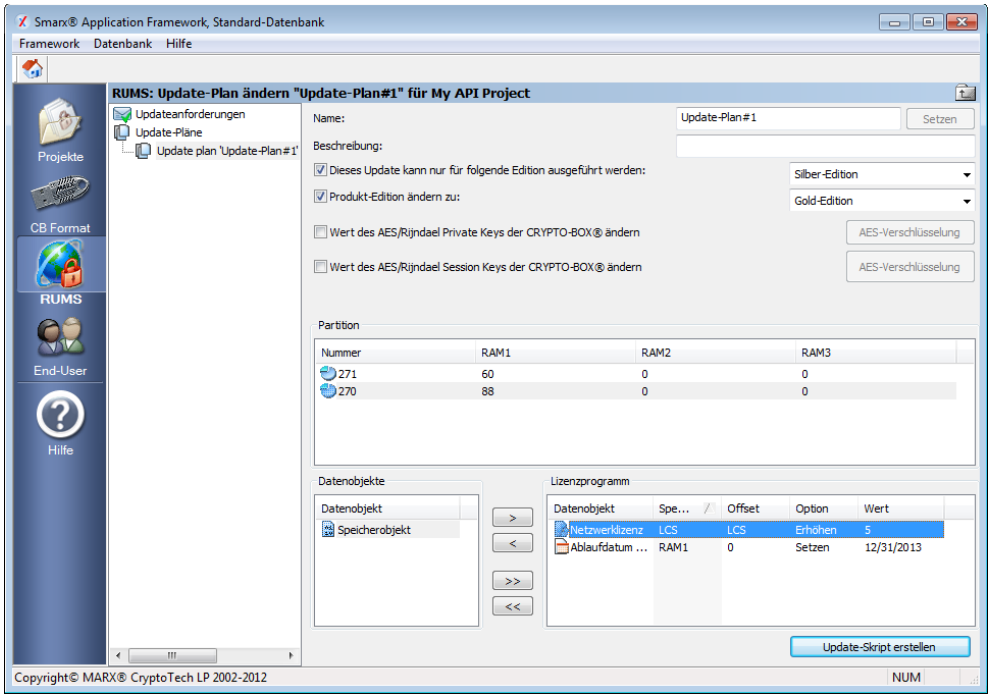
möchten. Klicken Sie dann rechts auf „Neuer Update-Plan“.

Auf der folgenden Seite können Sie die gewünschten Update-Einstellungen für jede Anwendung (bei AutoCrypt-Projekten) oder Partition (bei API-Projekten) vornehmen.

Wählen Sie dazu in der Mitte die gewünschte Anwendung (bei AutoCrypt) oder Partition (bei einem API-Projekt) aus und wählen Sie in unten links Bereich die Datenobjekte aus, die aktualisiert werden sollen.

Mit dem ">"-Button fügen sie ein Datenobjekt zur Update-Sequenz hinzu, mit "<" können Sie es wieder entfernen. Mit dem ">>"-Button fügen sie alle Datenobjekte hinzu, dabei werden die Standardeinstellungen für die gewählten Datenobjekte übernommen. Die Voreinstellungen hängen außerdem von der gewählten Produkt-Edition ab. Mit "Produkt-Edition ändern zu:" im oberen Bereich kann hier bei Bedarf auf eine andere Produkt-Edition aktualisiert werden.

Durch Doppelklick auf eines der Datenobjekte kann dessen Einstellung individuell geändert werden, die entsprechenden Einstellungen werden in die Update-Sequenz übernommen.

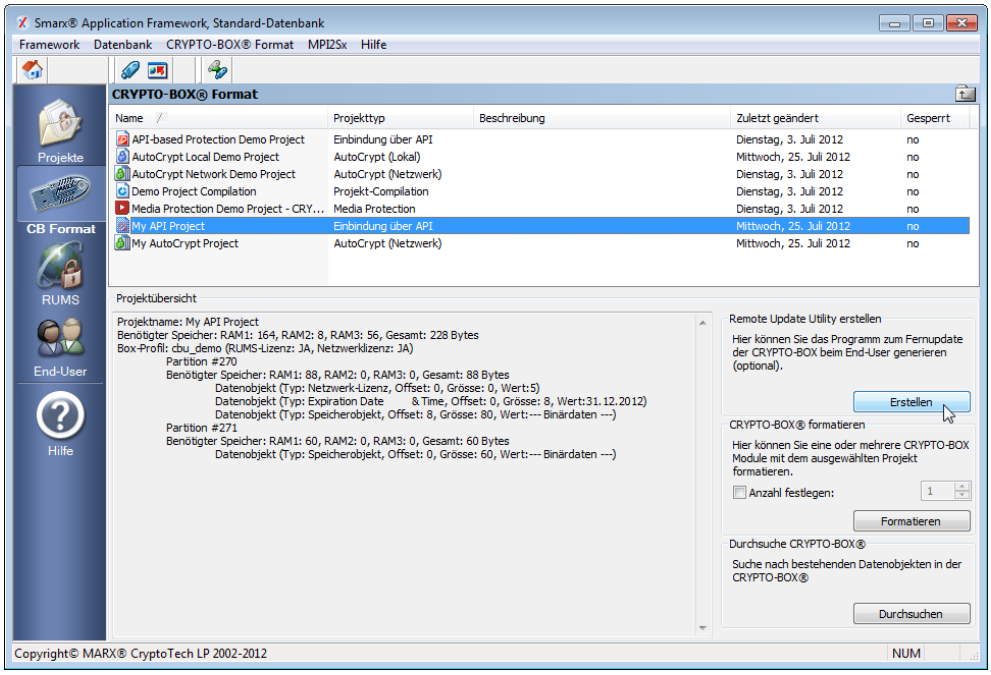


Mit dem Button "Update-Skript erstellen" unten rechts können Sie die Update-Einstellungen in eine XML-Datei exportieren. Diese kann zum Beispiel mit dem Kommandozeilentool "RU_Tool.exe" (siehe Kapitel 3) oder mit dem Online License Management System (OLM) eingesetzt werden. Weitere Details zu OLM finden Sie im Smarx Compendium, Kapitel 6 bzw. im Protection Kit Control Center unter dem Punkt „Online License Management und SOLO Server“.

2.3 Remote Update Utility für den Endanwender erstellen

Voraussetzung für ein Update der CRYPTO-BOX ist ein bestehendes Projekt im Smarx Application Framework (SxAF), mit dem die CRYPTO-BOX formatiert wurde (eine ausführliche Beschreibung des SxAF finden Sie im Smarx Compendium, Kapitel 4).
Der Endanwender, der im Besitz der CRYPTO-BOX ist, muss als erstes eine Updateanforderung erstellen.

Dazu benötigt er das Remote Update Tool. Sie können dieses erstellen, indem Sie das SxAF starten und unter "CB Format" auf den Punkt "Remote Update Tool erstellen" klicken:



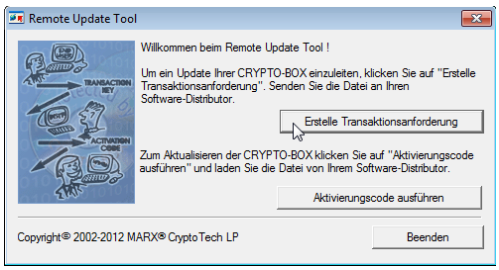
Sie können das RUpdate Tool nur erstellen, wenn Sie eine Lizenz für Remote Update haben. Diese erhalten Sie von MARX oder Ihrem Distributor in Form eines aktualisierten Hardwareprofils (.TRX-Datei), welche Sie in Ihr SxAF-Projekt importieren (Button "Profil importieren" unter "Projekteinstellungen"). Wenn Sie keine Lizenz haben, ist der Button "Erstellen" deaktiviert.

Wählen Sie anschließend einen Ordner, in dem das RUpdate Tool (insgesamt 3 Dateien) gespeichert werden und senden Sie diese Dateien an Ihren Endanwender.

Es empfiehlt sich, das Remote Update Utility zusammen mit Ihrer geschützten Anwendung an Ihre Endanwender auszuliefern.

2.4 Transaktionsanforderung erstellen (beim Endanwender)

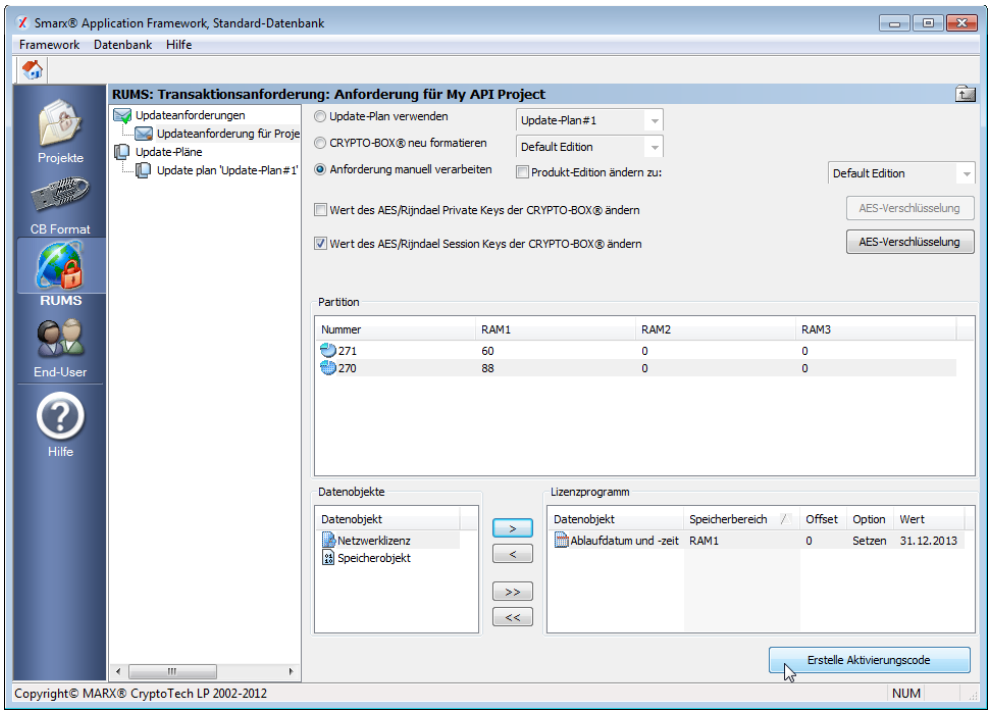
Um einen Remote-Update-Vorgang zu starten, schließt der Endanwender die CRYPTO-BOX an seinen Computer an und startet das RUpdate Tool. Hier klickt er anschließend auf “Erstelle Transaktionsanforderung”:



Es wird eine Datei mit der Endung .rutr erzeugt, diese schickt der Endanwender an Sie zurück (z.B. per Email).
Beim erzeugen der Transaktionsanforderung (*.rutr-Datei) wird wird eine spezielle Transaktions-ID in der CRYPTO-BOX gespeichert. Damit ist sichergestellt, dass nur die CRYPTO-BOX aktualisiert werden kann, mit der die Transaktionsanforderung erzeugt wurde.

2.5 Bearbeitung der Transaktionsanforderung

Sobald Sie die Transaktionsanforderung (*.rutr-Datei) des Endanwenders erhalten haben, laden Sie sie zur weiteren Bearbeitung in den Remote Update Manager. Rufen Sie das Smarx OS Application Framework auf, und wählen Sie aus der Menüleiste links "RUMS" aus. Klicken Sie dann auf "Laden", um die vom Endanwender geschickte Transaktionsanforderung zu öffnen. Anschließend haben Sie die Möglichkeit, die Datenobjekte in der CRYPTO-BOX zu ändern:



- Wenn Sie bereits vorgefertigte Update-Pläne haben (siehe Kapitel 2.2), klicken Sie einfach ganz oben auf die Option "Update-Plan verwenden" aus und wählen den gewünschten Update-Plan aus. Fahren Sie anschließend wie im Kapitel *2.6 Aktivierungscode erstellen* beschrieben fort.
- Mit der Option "CRYPTO-BOX neu formatieren" setzen Sie die CRYPTO-BOX auf den Auslieferungsstand einer bestimmten Produkt-Edition zurück. Fahren Sie anschließend wie im Kapitel *2.6 Aktivierungscode erstellen* beschrieben fort.
- Mit "Anforderung manuell verarbeiten" können Sie die gewünschten Updateeinstellungen manuell vornehmen. Neben einzelnen Datenobjekten können Sie auch die Produkt-Edition und die Werte für den Rijndael Private und den Rijndael Session Key der CRYPTO-BOX ändern. Um einzelne Datenobjekte zu ändern, wählen Sie unten links das gewünschte Datenobjekt und klicken auf ">", um das Datenobjekt in die Liste mit den zu aktualisierenden Objekten (rechtes Fenster) einzufügen. Es öffnet sich nun ein neues

Fenster, in dem Sie die Einstellungen für das Datenobjekt anpassen. Wiederholen Sie die beschriebenen Schritte für die übrigen Anwendungen oder Datenobjekte.

2.6 Aktivierungscode erstellen

Klicken Sie unten rechts auf "Erstelle Aktivierungscode", um den Aktivierungscode zu generieren und als Datei (*.ruac) zu speichern. Diese Datei senden Sie an Ihren Endanwender.

2.7 Ausführen des Aktivierungscodes beim Endanwender

Sobald der Endanwender den Aktivierungscode (*.ruac-Datei) von Ihnen erhalten hat (zum Beispiel per E-Mail), kann er die CRYPTO-BOX aktualisieren. Dazu startet er wieder das Remote Update Tool, schließt die CRYPTO-BOX an und klickt auf "Aktivierungscode ausführen". Die Transaktions-ID, die bei jeder Transaktionsanforderung erstellt und in der CRYPTO-BOX gespeichert wird, garantiert, dass jedes Update nur einmal ausgeführt werden kann und verhindert so unautorisierte (mehrfache) Aktivierungen.

3. Remote Update mit dem kommandozeilenbasierten RU_Tool.exe

3.1 Überblick

Das Remote Update über das Kommandozeilenprogramm RU_Tool.exe bietet mehr Flexibilität als das Update über die grafische Oberfläche des Smarx OS Application Frameworks. So kann es auch von externen Anwendungen heraus aufgerufen werden. Die Update-Informationen liegen in Form von XML-Dateien vor, die sich zum Beispiel auch dynamisch aus eigenen Anwendungen heraus generieren lassen. Somit ist eine Anbindung an nahezu jedes bestehende Distributionssystem und ein hoher Grad an Automatisierung möglich.

Im Gegensatz zu RU_Tool.exe ist beim Kommandozeilentool SxFormat.exe (siehe Kapitel 4) keine vorherige Übersendung von Daten durch den Endanwender (Updateanforderung, siehe Kapitel 3.2) notwendig. Beachten Sie jedoch, dass Sie dadurch ggf. weniger Kontrolle bei der Aktualisierung der CRYPTO-BOX haben.

Sie finden RU_Tool.exe im PPK Control Center unter dem Punkt „Treiber und Tools“ → „Kommandozeilentools“ bzw. unter:
[SmarxOS PPK Installationsordner]\Tools\RU_Tool
In diesem Ordner befindet sich auch eine Datei readme.txt, die alle Funktionen von RU_Tool.exe ausführlich beschreibt.

Der Einsatz von RU_Tool.exe ist in Kombination mit dem anderen Kommandozeilentools AC_Tool.exe (Kommandozeilenversion von AutoCrypt) und SmrxProg.exe (zur Formatierung der CRYPTO-BOX) möglich.

Weitere Informationen zu AC_Tool.exe und SmrxProg.exe finden Sie ebenfalls im Protection Kit Control Center unter dem Punkt „Treiber und Tools“ → „Kommandozeilentools“oder unter:

[SmarxOS PPK Installationsordner]\Tools\AC_Tool
bzw.
[SmarxOS PPK Installationsordner]\Tools\SmrxProg

Unsere AutoCrypt Application Notes enthalten eine ausführliche Beschreibung von AC_Tool.exe und SmrxProg.exe.

Vorteile von Remote Update mit RU_Tool.exe:

- Anbindung an nahezu jede Anwendung, Datenbank oder Distributionsszenario möglich.
- Automatisierung möglich, z.B. automatische Erstellung von spezifischen Updateinformationen direkt aus Ihrem Vertriebssystem heraus.
- Komplette Neuaufteilung des CRYPTO-BOX Speichers möglich (Erstellung neuer Partitionen oder Datenobjekte).

Besonderheiten:

- Updateskript (XML-Datei) muss erst erstellt bzw. generiert werden - entweder über das SxAF, manuell oder durch Einbindung in ein bestehendes System (ggf. Programmieraufwand erforderlich).

3.2 Initiieren des Remote Update Vorgangs

Der Endanwender, der im Besitz der CRYPTO-BOX ist, muss als erstes eine Updateanforderung erstellen. Dazu benötigt er das RUpdate Tool.
Dieses erstellen Sie mittels folgendem Aufruf:
RU_Tool.exe -extract <TRX-file> <EXE-file>

Dabei sind:
<TRX-file> - TRX-Datei, Hardwareprofil Ihrer kundenspezifischen CRYPTO-BOX
<EXE-file> - Dateiname für das RUpdate Tool

Ergebnis: <EXE-file> (RUpdate Tool) wird extrahiert, sowie zwei weitere Dateien mit der Endung *.409 und *.407, die Textressourcen in Deutsch und Englisch enthalten. Das Ergebnis wird sowohl am Bildschirm angezeigt als auch in die Datei RU_Tool.LOG geschrieben.

Sie können das RUpdate Tool nur erstellen, wenn Sie eine Lizenz für Remote Update haben. Diese erhalten Sie von MARX oder Ihrem Distributor in Form eines aktualisierten Hardwareprofils (.TRX-Datei). Wenn Sie keine Lizenz haben, wird eine Fehlermeldung angezeigt.

Die erzeugten Dateien senden Sie anschließend an Ihrem Endanwender.
Um einen Remote-Update-Vorgang zu starten, schließt der Endanwender die CRYPTO-BOX an seinen Computer an und startet das Remote Update Tool. Hier klickt er anschließend auf “Erstelle Transaktionsanforderung”:

Es wird eine Datei mit der Endung .rutr erzeugt, diese schickt der Endanwender an Sie zurück (z.B. per Email).
Beim erzeugen der Transaktionsanforderung (*.rutr-Datei) wird wird eine spezielle Transaktions-ID in der CRYPTO-BOX gespeichert. Damit ist sichergestellt, dass nur die CRYPTO-BOX aktualisiert werden kann, mit der die Transaktionsanforderung erzeugt wurde.

Das RUpdate Tool kann ebenso wie RU_Tool.exe alternativ per Kommandozeile gesteuert werden. Eine Übersicht der Parameter finden Sie in der Datei readme.txt, Appendix A im Ordner von RU_Tool.exe (siehe 3.1).

3.3 Erstellen eines Aktivierungscodes

XML-Skriptdatei:

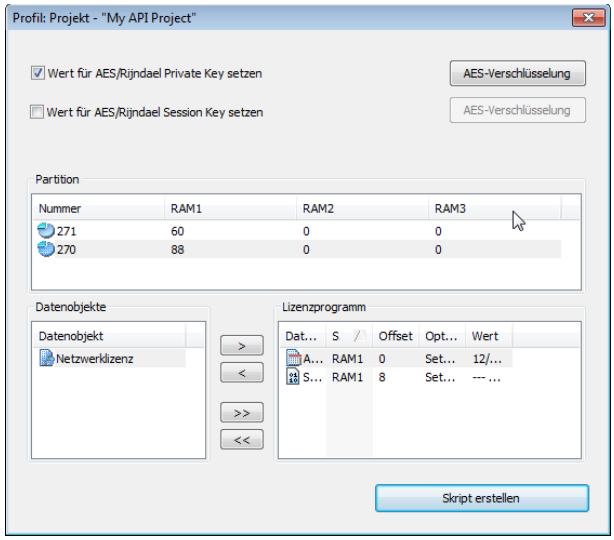
Für die Erstellung des Aktivierungscodes wird neben der Transaktionsanforderung (*.rutr-Datei) des Endanwenders noch eine XML-Datei benötigt, welche Informationen darüber enthält, was in der CRYPTO-BOX aktualisiert werden soll. Ein Beispiel für eine solche XML-Datei ist die Datei *AC_Local_Update.xml* im selben Verzeichnis wie RU_Tool.exe.

XML-Datei automatisch erzeugen:

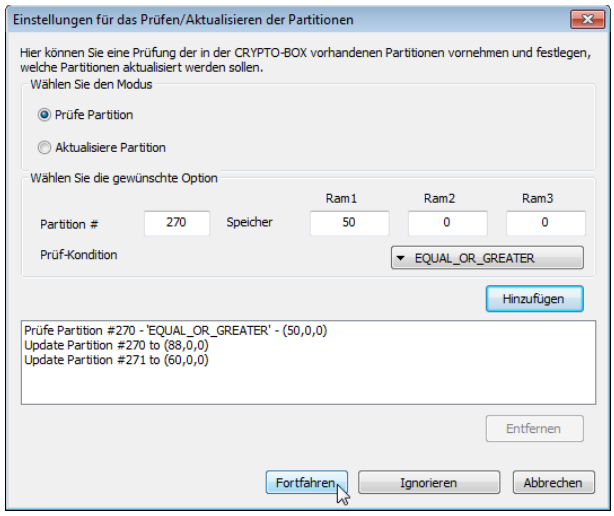
Die Beispieldatei *AC_Local_Update.xml* ist ein guter Ausgangspunkt, um eine angepasste XML-Datei manuell zu erstellen. Es geht aber noch einfacher - mit dem Smarx Application Framework (SxAF) können die XML-Datei automatisch erzeugt werden:

- Starten Sie das SxAF und öffnen Sie ein bestehendes Projekt "Einbindung über API" oder legen Sie ein neues an.
- Importieren Sie unter "Projekteinstellungen" das Hardwareprofil (TRX-Datei) für Ihre CRYPTO-BOX, welches Ihrer ersten CRYPTO-BOX Lieferung beiliegt (CDROM mit der Aufschrift "Confidential"). Weitere Details dazu finden Sie in dem White Paper zur TRX-Datei. (ist auch auf der CD mit der TRX-Datei).
- Legen Sie die Partitionen und Datenobjekte an, die Sie aktualisieren möchten (siehe Smarx Compendium, Kapitel 4.5 für eine detaillierte Anleitung) .
- Klicken Sie im Menü ganz oben auf "Projekt" → "Erstelle Updateskript für RU_Tool"
- Jetzt können Sie festlegen, welche Datenobjekte Sie über Remote Update aktualisieren möchten und welche Operation ausgeführt werden soll (z.B. setzen oder erhöhen bei Zählern oder Ablaufdatum, oder ändern eines Speicherobjekts). Mit dem ">"-Button fügen sie ein Datenobjekt zur Update-Sequenz hinzu, mit "<" können Sie es wieder entfernen. Mit dem ">>"-Button fügen sie alle Datenobjekte hinzu, dabei werden die Standardeinstellungen für die gewählten Datenobjekte übernommen. Außerdem können Sie ganz oben neue Werte für die AES/Rijndael Keys festlegen, falls gewünscht.
- Nachdem Sie alle Einstellungen wie gewünscht vorgenommen haben, klicken Sie unten rechts auf "Skript

erstellen"



- Im folgenden Fenster werden Sie gefragt, ob Sie Prüfbedingungen für die Partitionen erstellen wollen. So können Sie zum Beispiel während des Updatevorgangs überprüfen, ob auf der CRYPTO-BOX bereits eine Partition mit einer bestimmten Größe vorhanden ist - falls nicht, wird das Update nicht ausgeführt. Dies kann sehr nützlich sein, z.B. wenn Sie CRYPTO-BOXen mit unterschiedlichen Lizenzinformationen für verschiedene Programmpakete im Umlauf haben und verhindern wollen, dass die falsche CRYPTO-BOX aktualisiert wird. Klicken Sie dazu auf "Ja", sonst auf "Nein".
- Bei Klick auf "Ja" öffnet sich ein neues Fenster. Im unteren Teil stehen bereits die Updateinformationen, die Sie vorher festgelegt haben. Im oberen Teil können Sie Prüfbedingungen festlegen. Nehmen wir an Sie wollen, dass das Update nur dann ausgeführt wird, wenn die Partition #270 in der CRYPTO-BOX mindestens eine Größe von 50 Bytes in RAM1 hat (zum Beispiel weil Sie vorher hier schon entsprechende Lizenzdaten programmiert haben). Falls nicht, soll das Update nicht ausgeführt werden. Wählen Sie dazu oben die Option "Prüfe Partition" aus und geben bei Partition # "270" ein und bei RAM1 "50". Bei "Prüfkondition" wählen Sie "EQUAL_OR_GREATER" und klicken dann auf "Hinzufügen" – die Einstellung erscheint unten in der Zusammenfassung. Mit "Aktualisiere Partition" können Sie bestehende Partitionen vergrößern oder verkleinern oder neue hinzufügen zur CRYPTO-BOX hinzufügen.



- Klicken Sie auf "Fortfahren", um die Einstellungen zu übernehmen, oder auf "Ignorieren", wenn Sie nichts

ändern wollen.

- Geben Sie im nächsten Fenster den Name für die XML-Datei an und speichern diese.

Machen Sie am besten einen Probelauf und testen Sie die erstellte XML-Datei auf korrekte Funktion, bevor Sie Updates an Ihre Kunden verschicken.
Sie können die XML-Datei natürlich auch nachträglich abändern oder anpassen. Wenn Sie etwas mit den Updateeinstellungen und Prüf-Konditionen wie oben beschrieben spielen, gewinnen Sie einen guten Eindruck wie die XML-Datei funktioniert.

Seriennummer der CRYPTO-BOX auslesen (falls benötigt):
Aus der Transaktionsanforderung des Endkunden kann die Seriennummer der CRYPTO-BOX ausgelesen werden, mit der der Endanwender die Transaktionsanforderung erstellt hat. Dies kann zum Beispiel nützlich sein, wenn man die CRYPTO-BOX anhand der Seriennummer (BoxName) verifizieren möchte. Dazu dient folgender Aufruf:

RU_Tool.exe -getboxname <TRX-file> <RUTR-file> [<LOG-file>]

Dabei sind:
<TRX-file> - TRX-Datei, Hardwareprofil Ihrer kundenspezifischen CRYPTO-BOX
<RUTR-file> - Transaktionsanforderung, aus der die Seriennummer ausgelesen werden soll
<LOG-file> - Log-Datei (optional), in die die Seriennummer geschrieben wird. Ohne Angabe von <LOG-file> wird die Seriennummer auf der Konsole ausgegeben.
Unabhängig davon wird die Seriennummer auch in der Dateibezeichnung der Transaktionskennung übertragen.

Aktivierungscode erstellen:
Zur Erzeugung des Aktivierungscodes dient folgender Aufruf:

RU_Tool.exe -update <TRX-file> <XML-file> <RUTR-file> <RUAC-file>

Dabei sind:
<TRX-file> - TRX-Datei, Hardwareprofil Ihrer kundenspezifischen CRYPTO-BOX
<XML-file> - XML-Datei mit Updateinformationen (weitere Infos siehe weiter vorne)
<RUTR-file> - Transaktionsanforderungsdatei
<RUAC-file> - Datei mit dem Aktivierungscode

Der Aktivierungscode wird in <RUAC-file> gespeichert. Das Ergebnis wird am Bildschirm angezeigt und in die Datei RU_Tool.LOG geschrieben.

3.4 Ausführen des Aktivierungscodes beim Endanwender

Sobald der Endanwender den Aktivierungscode (*.ruac-Datei) von Ihnen erhalten hat (zum Beispiel per E-Mail), kann er die CRYPTO-BOX aktualisieren. Dazu führt er das Remote Update Tool aus, schließt die CRYPTO-BOX an und klickt auf “Aktivierungscode ausführen”. Die Transaktions-ID, die bei jeder Transaktionsanforderung erstellt wird, garantiert, dass jedes Update nur einmal ausgeführt werden kann und verhindert so unautorisierte (mehrfache) Aktivierungen.

4. Remote Update mit dem kommandozeilenbasierten SxFormat.exe

4.1 Überblick

Eine Aktualisierung der CRYPTO-BOX über das Kommandozeilenprogramm SxFormat.exe bietet im Gegensatz zu RU_Tool.exe (siehe Kapitel 3) die Möglichkeit, ein unbedingtes Update durchzuführen, ohne dass der Endanwender Ihnen zuerst die Transaktionsanforderung schicken muss. Der Endanwender muss lediglich die EXE-Datei ausführen, die Sie ihm zusenden. SxFormat wird aus dem Kommandozeilentool "SmrxProg.exe" heraus erzeugt. Die Update-Informationen werden dabei über XML-Dateien spezifiziert, die sich ebenfalls dynamisch aus eigenen Anwendungen heraus generieren lassen. Somit ist eine Anbindung an nahezu jedes bestehende Distributionssystem und ein hoher Grad an Automatisierung möglich.
Sie finden SmrxProg.exe im SmarxOS Protection Kit unter:
[SmarxOS PPK Installationsordner]\Tools\SmrxProg
In diesem Ordner befindet sich auch eine Datei readme.txt, die alle Funktionen von SmrxProg.exe ausführlich beschreibt.

Vorteile von Remote Update mit SxFormat.exe:

- Kommandozeilentool, welches über Skripte und von externen Anwendungen heraus aufgerufen werden kann - dadurch Automatisierung und Anbindung an nahezu jede Anwendung oder Vertriebssystem möglich
- Die CRYPTO-BOX kann komplett neu beschrieben werden (Erstellung neuer Partitionen oder Datenobjekte).
- Eine vorherige Übermittlung von Daten durch den Endwanwender wie bei der RUMS-Funktion des SxAF (siehe Kapitel 2) oder bei RU_Tool.exe (siehe Kapitel 3) ist nicht erforderlich.

Besonderheiten:

- Ggf. weniger Kontrolle bei der Aktualisierung der CRYPTO-BOX, da keine Daten über die zu aktualisierende CRYPTO-BOX vorliegen.
- Updateskript (XML-Datei) muss erst erstellt bzw. generiert werden - entweder über das SxAF, manuell oder durch Einbindung in ein bestehendes System (ggf. Programmieraufwand erforderlich).

4.2 Erzeugung der SxFormat-Package für den Endanwender

Starten Sie das Protection Kit Control Center und gehen Sie auf den Punkt „Treiber und Tools“ → „Kommandozeilentools“. Hier finden Sie den Link zu SmrxProg.exe, der zugehörigen Readme-Datei sowie Beispielskripten.
Sie erstellen das SxFormat-Package mittels folgendem Aufruf:
SmrxProg.exe -extractSx <TRX-file> <XML-file> <EXE-file>

Dabei sind:
<TRX-file> - TRX-Datei, Hardwareprofil Ihrer kundenspezifischen CRYPTO-BOX
<XML-file> - XML-Datei mit Updateinformationen (weitere Infos siehe Kapitel 4.3)
<EXE-file> - Dateiname für das RUpdate Tool

Ergebnis: <EXE-file> (SxFormat-Tool) wird extrahiert, sowie zwei weitere Dateien mit der Endung *.409 und

*.407, die Textressourcen in Deutsch und Englisch enthalten. Das Ergebnis wird sowohl am Bildschirm angezeigt als auch in die Datei SmrxProg.LOG geschrieben.
Diese drei Dateien senden Sie anschließend an Ihren Endanwender.

Sie können das SxFormat Tool nur erstellen, wenn Sie eine Lizenz für Remote Update haben. Diese erhalten Sie von MARX oder Ihrem Distributor in Form eines aktualisierten Hardwareprofils (.TRX-Datei). Wenn Sie keine Lizenz haben, wird eine Fehlermeldung angezeigt.

Beachten Sie, dass Sie mit SxFormat.exe im Gegensatz zu RU_Tool.exe nur beschränkte Kontrolle haben, welche CRYPTO-BOX mit dem Tool umprogrammiert wird! Es besteht zwar über das "Extended Script Format" (siehe SmrxProg Readme-Datei, Appendix A) die Möglichkeit, bestimmte Checks einzubauen. Ein Feedback wie bei RU_Tool.exe oder dem RFP API (es läst sich nur die CRYPTO-BOX aktualisieren, von der die Anforderung kommt) erhalten Sie jedoch nicht.

4.3 Erzeugung der XML-Skriptdatei

Für die Erstellung des SxFormat-Tools zur Aktualisierung der C RYPTO-BOX wird eine XML-Datei benötigt, welche Informationen darüber enthält, was in der CRYPTO-BOX aktualisiert werden soll. Ein Beispiel für eine solche XML-Datei ist die Datei *AC_Local.xml* (für AutoCrypt-Nutzer) oder *SmrxProg_Demo.xml* (für Einbindung mit API) im selben Verzeichnis wie SmrxProg.exe.
Insbesonders bei der Einbindung über API kann es schwierig sein, die Updatesequenz basierend auf dem vorhandenen Beispiel *AC_Local.xml* manuell zu erstellen. Das Smarx Application Framework (SxAF) nimmt Ihnen diese Arbeit ab:

- Starten Sie das SxAF und öffnen Sie ein bestehendes Projekt "AutoCrypt", "AutoCrypt (Netzwerk)" oder "Einbindung über API", oder legen Sie ein neues Projekt an.
- Importieren Sie unter "Projekteinstellungen" das Hardwareprofil (TRX-Datei) für Ihre CRYPTO-BOX, welches Ihrer ersten CRYPTO-BOX Lieferung beiliegt (CD mit der Aufschrift "Confidential"). Weitere Details dazu finden Sie in dem beiliegenden White Paper zur TRX-Datei.
- Legen Sie die Partitionen und Datenobjekte an, die Sie in die CRYPTO-BOX schreiben wollen.
- Klicken Sie im Menü ganz oben auf "Projekt" → "Erstelle XML-Skript für SmrxProg"
- Legen Sie im folgenden Fenster fest, welche (Partitions-) Infos in die XML-Datei geschrieben werden sollen und klicken Sie auf "Exportieren"
- Geben Sie im nächsten Fenster den Name für die XML-Datei an und speichern Sie sie.

Machen Sie am besten einen Probelauf und testen Sie die erstellte XML-Datei auf korrekte Funktion, bevor Sie Updates an Ihre Kunden verschicken.
Sie können die XML-Datei natürlich auch nachträglich abändern oder anpassen. Wenn Sie etwas mit den Partitionseinstellungen und Datenobjekten des Projekts spielen, gewinnen Sie einen guten Eindruck wie die XML-Datei funktioniert.

Über das "Extended Script Format" (siehe SmrxProg Readme-Datei, Appendix A) besteht ausserdem die Möglichkeit, bestimmte Checks einzubauen und die Programmierung der CRYPTO-BOX zu verweigern, wenn diese Bedingungen nicht zutreffen. So können Sie zum Beispiel verhindern, dass Endanwender eine CRYPTO-BOX aktualisieren, für die das Update nicht gedacht war. Entsprechende XML-Schablonen im Extended Script Format mit den gewünschten Prüf-Konditionen können Sie auch aus dem SxAF heraus erzeugen lassen – siehe dazu Kapitel 3.3.

5. Remote Update mit dem Remote Update API für Entwickler

Der Einsatz der Remote Update API als Bestandteil des Smarx API bietet Ihnen als Entwickler die maximale Flexibilität. So können Sie den Updatevorgang direkt in Ihr Konzept einbauen. MARX liefert Bibliotheken und Beispielcode für verschiedene Entwicklungsumgebungen, zum Beispiel:

- Als statische Bibliothek für C++ (Visual Studio, 32 und 64Bit)
- CBIOS.RFP als Bestandteil von CBIOS4NET (objektorientierte, komponentenbasierte API für .NET-Entwickler)

Eine ausführliche Dokumentation der API-Befehle finden Sie im CRYPTO-BOX Compendium, bzw. in der CBIOS4NET-Dokumentation. Beide sind im Protection Kit als PDF-Datei enthalten.

Vorteile von Remote Update mit der Remote Update API:

- Vollständige Integration in das eigene Konzept und Anwendungen möglich
- Maximale Flexibilität

Besonderheiten:

- Programmierkenntnisse erforderlich

Eine Beschreibung des Remote Update API finden Sie im Smarx Compendium bzw. in der CBIOS4NET API Referenz, Kapitel 7 (für C#-Entwickler).

6. FAQ - häufige Fragen

1. Wenn ich mit den Kommandozeilentools versuche, das Remote Update Utility oder SxFormat.exe zu erstellen, erhalte ich immer die Fehlermeldung „Error: Decryption of 'C:\...\CBU_XSN-xxxxx.trx' failed - your CRYPTO-BOX(R) firmware 2.2 or higher should be attached“!

Zur Erstellung dieser Tools ist es erforderlich, dass eine zu der TRX-Datei passende CRYPTO-BOX am Computer angeschlossen ist.

2. Remote Update funktioniert bei mir nicht: im SxAF ist die Schaltfläche zum Erstellen des Remote Update Utility deaktiviert, und bei dem Kommandozeilentools bekomme ich immer die Fehlermeldung „ Error: RUMS not licensed”!

Remote Update ist optional erhältlich. Bitte wenden Sie sich an Ihren MARX-Distributor oder bestellen Sie direkt auf unserer Webseite.

3. Warum muss bei Remote Update immer erst der Endanwender den Updatevorgang auslösen, indem er die Transaktionsanfordeung schickt? Das ist mir zu aufwändig!

Mit "SxFormat" können Sie ein Update ohne vorherige Übermittlung einer Transaktionskennung durch den Endanwender durchführen. Für weitere Details siehe Kapitel 4.

4. Ist Remote Update bzw. SxFormat auch für Linux bzw. Mac OS X verfügbar?

Diese Tools bzw. Bibliotheken sind derzeit nur für Windows erhältlich, da wir bisher keine entsprechenden Anfragen für andere Plattformen erhalten haben. Bei Interesse können wir die Update-Funktionen jedoch auch unter Linux und Mac OS X bereit stellen – bitte wenden Sie sich dazu an unseren Technischen Support oder vereinbaren Sie einen Rückruf.

5. Ich verwende zur Aktualisierung der CRYPTO-BOX die RUMS-Funktionalität im SxAF. Ich möchte jetzt noch zusätzliche Daten (Partitionen bzw. Datenobjekte) in die CRYPTO-BOX meiner Kunden schreiben. RUMS erlaubt mir aber nur, bestehende Partitionen zu aktualisieren!

Nutzen Sie in dem Fall das kommandozeilenbasierte RU_Tool.exe. Eine detaillierte Anleitung dazu finden Sie in Kapitel 3 in diesem Dokument.

6. Steht das Remote Update API auch für andere Entwicklugsumgebungen außer C++ und C# zur Verfügung?

Bei Interesse an Bibliotheken für andere Compiler oder Plattformen wenden Sie sich bitte an unseren Technischen Support oder vereinbaren Sie einen Rückruf.

Remote Update (RUMS) AN-4

CRYPTO-BOX® Datenblatt

	CRYPTO-BOX SC (CBU SC)	CRYPTO-BOX XS/Versa (CBU XS/Versa)
		
Controller-Chip	RISC Smartcard Prozessor	RISC Smartcard Prozessor
Chip Zertifizierungen	EAL4+	EAL4+
Unterstützte Betriebssysteme	Windows, Linux, Mac OS X, iOS, Android	Windows, Linux, Mac OS X, iOS, Android
In Hardware integrierte Algorithmen	AES 128 bit, RSA (bis zu 2048 Bit Schlüssellänge), andere auf Anfrage (z.B. ECC)	AES 128 Bit auf Hardwareebene, RSA (bis zu 2048 Bit Schlüssellänge, auf Treiberebene)
Speichergröße (insgesamt)	72KByte, ca. 30KByte frei	4, 32 oder 64 KByte
Lese-/Schreibrate interner Speicher	ca. 80kByte/s	ca. 12kByte/s
Passwort (PIN/PUK)	Bis zu 16 Byte Länge	
Gehäuse & LED	Designer-Metallgehäuse, Zinkguss, LED mit Anzeige des Betriebszustandes, Öse für Schlüsselring	
Steckverbindung	USB Typ A	
Programmierung des Speichers	minimum 100.000 Zyklen	
Datenerhaltszeit	minimum 10 Jahre	
Konformität und Zertifizierungen	FCC, CE, RoHS, USB-Logo	
Abmessung	14 x 7 x 32,5 mm	14 x 7 x 32,5 mm
Gewicht	7,5g	7,5g
Temperaturbereich	-10°C bis zu +70°C	
Luftfeuchtigkeit	0% bis 95% relative Luftfeuchtigkeit	

CRYPTO-BOX Zertifizierungen

Evaluation Kit

www.marx.com/de/store

MARX Software Security GmbH
Vohburger Strasse 68
85104 Wackerstein, Germany
Phone: +49 (0) 8403 / 9295-0
Fax: +49 (0) 8403 / 9295-40

MARX CryptoTech LP
489 South Hill Street
Buford, GA 30518 U.S.A.
Phone: (+1) 770 904 0369
Fax: (+1) 678 730 1804

www.marx.com