Dirk Fox

Datenverschlüsselung mit TrueCrypt

In den vergangenen Jahren wurden zahlreiche Schutzprogramme als kostenlose Open Source Software veröffentlicht. Einige dieser Lösungen haben inzwischen einen hohen Reifegrad erreicht. Der Beitrag stellt das Programm TrueCrypt vor, das heute bereits in vielen Unternehmen zum Schutz vor unberechtigtem Datenzugriff eingesetzt wird.

Hintergrund

TrueCrypt ist ein freies, Open Source-Verschlüsselungsprogramm für Daten und Speichermedien. Das Tool wurde 2004 aus dem Quellcode von „Encryption for the Masses" (E4M) weiter entwickelt, dessen Autor Paul le Roux die Pflege von E4M im Jahr 2000 eingestellt hatte. Die Open Source-Lizenz von TrueCrypt besteht daher aus unterschiedlichen, z. T. autorenspezifischen Teillizenzen, die in einer „TrueCrypt Collective License" zusammengefasst wurden. Die Nutzung des Source-Codes unterliegt daher teilweise einzelnen Beschränkungen.

TrueCrypt wurde ursprünglich für das Betriebssystem Microsoft Windows (2000, Server 2003, XP, Vista) entwickelt. Jüngere Programmversionen gibt es auch als Portierungen auf die Betriebssysteme Linux (ab Kernel 2.4, seit Version 4.0) und Mac OS X (ab v10.4, seit Version 5.0); sie werden seitdem für alle drei Betriebssysteme gepflegt.

Über sechs Millionen Mal wurden allein die Windows-Versionen von TrueCrypt von der Original-Webseite heruntergeladen. Da TrueCrypt inzwischen in zahlreichen Unternehmen eingesetzt wird, dürfte die tatsächliche Verbreitung von TrueCrypt um ein Vielfaches größer sein.

Gegenstand des vorliegenden Beitrags ist die Programmversion 6.0a, veröffentlicht am 08.07.2008. Betriebssystemspezifische Elemente der Darstellung beziehen sich auf die Windows-Version von TrueCrypt. Die Linux- und die OS-X-Version weichen teilweise im Detail von der Windows-Version ab; die Funktionsweise ist jedoch überwiegend identisch.

Dirk Fox

Geschäftsführer der Secorvo Security Consulting GmbH und Herausgeber der DuD.

E-Mail: dirk.fox@secorvo.de

1 Leistungsumfang

TrueCrypt unterstützt drei verschiedene Betriebsarten:

- die Erstellung und Nutzung eines verschlüsselten „Datencontainers", also einer (in der Regel großen) Datei, die wie ein logisches Laufwerk angesprochen wird,
- die Verschlüsselung einer kompletten Partition der Festplatte oder eines externen Speichers (z. B. einer USB-Platte) und
- die Verschlüsselung des Gesamtsystems (Systempartition), sodass das Betriebssystem erst nach Eingabe des Entschlüsselungspassworts startet (auch „Pre Boot Authentication" genannt).

Als Verschlüsselungsverfahren kommen wahlweise der vom amerikanischen NIST standardisierte AES[1] [NIST_01] sowie zwei weitere der fünf „Finalisten" des AES-Auswahlprozesses, Serpent[2] und Twofish[3] zum Einsatz – oder Kombinationen der drei Verfahren. Die Schlüssellänge ist einheitlich auf 256 bit festgelegt – und liegt damit Größenordnungen über der heute empfohlenen Mindestlänge von 80 bis 90 bit[4].

Alle drei Verfahren wurden in Assembler implementiert und sind daher sehr schnell. Mit der Option „Benchmark Test" kann die Geschwindigkeit einer RAM-Verschlüsselung auf dem eigenen System bestimmt werden. Die mit Abstand schnellste Implementierung ist der AES.

Der Verschlüsselungsschlüssel wird von einem Pseudozufallszahlengenerator erzeugt – HMAC mit einer Hashfunktion nach PKCS #5 v2.0 [Kali_00]. Als Hashfunktionen stehen der amerikanische NIST-Standard SHA-512 (FIPS PUB 180-2, [NIST_02]) sowie die europäischen Algorithmen Whirlpool und RIPEMD-160 (ISO/IEC 10118-3:2004, [ISO_04]) zur Auswahl.

Der Startwert der Pseudozufallszahlenerzeugung wird mit Mausbewegungen des Benutzers „gefüttert". Die Blockchiffre-Verschlüsselung erfolgt in der Betriebsart XTS, die auf Phillip Rogaway [Roga_04] zurück geht und inzwischen von IEEE standardisiert wurde [IEEE_07].

2 Installation

Die Installation von TrueCrypt erfordert Administrator-Rechte auf dem lokalen System, da die Verschlüsselung in System-Module eingreift. Die Nutzung des Programms ist jedoch mit einfachen Benutzerrechten möglich.

Die Menus des Programms wurden in englischer Sprache verfasst. Es lassen sich jedoch Sprachmodule für über 35 Sprachen nachladen. Allerdings sind nicht alle Sprachmodule vollständig; darunter auch das deutsche: Will man alle Optionen von TrueCrypt nutzen und jede Mel-

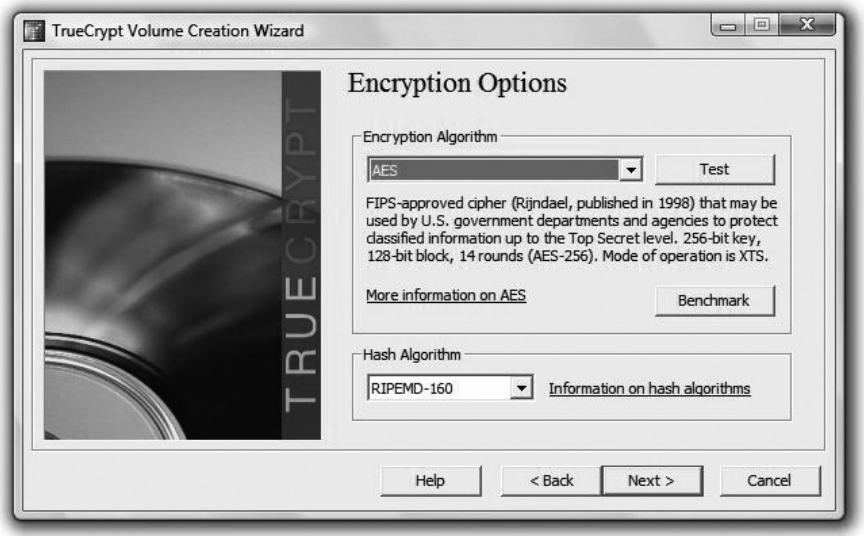
1 Entwickelt von Joan Daemen und Vincent Rijmen.

2 Entwickelt von Ross Anderson, Eli Biham und Lars Knudsen.

3 Entwickelt von Bruce Schneier, John Kelsey, Niels Ferguson, Doug Whiting, David Wagner und Chris Hall.

4 Siehe http://www.keylength.com

Abb. 1 | Auswahl Verschlüsselungsverfahren und Hashfunktion



dung des Systems verstehen, sind englische Sprachkenntnisse unverzichtbar.

3 Verschlüsselte Datencontainer

3.1 Erzeugung

Die Erzeugung eines verschlüsselten Datencontainers erfolgt mit Unterstützung eines „Assistenten", der durch die erforderlichen Schritte führt.

Zunächst wird festgelegt, ob ein neuer Datencontainer oder ein versteckter Container („Hidden Volume") erzeugt werden soll. Ein solches „Hidden Volume" wird in dem freien Speicher eines bereits angelegten Datencontainers platziert und bleibt selbst dann noch verborgen, wenn auf den „Gastgeber"-Container zugegriffen wird.

Anschließend wird der Dateiname gewählt, unter dem der Datencontainer abgelegt wird – mit beliebiger Extension. Damit lässt sich der Container z. B. als Bild oder Office-Dokument tarnen. Ist der Dateiname bereits vergeben, wird die Datei von dem neuen TrueCrypt-Container überschrieben. Der Speicherort des Containers ist nicht bindend – letzterer kann später wie eine normale Datei verschoben werden, auch auf einen USB-Stick, einen anderen mobilen Datenträger oder eine CD/DVD.

Schließlich werden das Verschlüsselungsverfahren (Empfehlung: AES) und der Hashalgorithmus (Empfehlung: SHA-512 oder RIPEMD-160) gewählt und die Größe des Containers festgelegt. Achtung: Wer den Container auf einem FAT-formatierten Speichermedium anlegen möchte, ist auf eine Maximalgröße von 4 GB beschränkt; bei NTFS-formatierten Medien ist nur die Mindestgröße von 2,76 MB zu beachten.

Unter NTFS kann die Größe des Containers zudem dynamisch gewählt werden; der Container wächst dann während der Nutzung nach Bedarf bis zur gewählten Maximalgröße. Solche dynamischen Container erfordern jedoch deutlich mehr Rechenleistung bei der Nutzung.

Neben einem guten Passwort (besser: einer „Pass-Phrase", also einem „Passwort-Satz") kann die Verschlüsselung auch (oder zusätzlich) mit einer „Schlüsseldatei" erfolgen. Als Schlüsseldatei kann jedes beliebige File dienen; die ersten 1.024 Bytes der Datei sollten jedoch eine möglichst hohe Entropie besitzen (z. B. eine komprimierte Datei wie .zip, .mp3 oder .jpg). Man kann sich aber auch von TrueCrypt vorab eine Schlüsseldatei mit Hilfe des Pseudozufallszahlengenerators erzeugen lassen.

Vorteil der Verwendung einer Schlüsseldatei: Sie schützt auch vor Keyloggern, die die Eingabe des Passworts mitschneiden können. Nachteil: Ein einziger Bitfehler unter den ersten 1.024 Byte der Schlüsseldatei, und der Container lässt sich nicht mehr entschlüsseln. Schlüsseldateien müssen daher nicht nur vor unberechtigtem Zugriff und Verlust, sondern auch vor jeder Veränderung geschützt werden.

Sobald das Dateisystem (NTFS oder FAT) gewählt ist, kann die Erzeugung (Formatierung) des verschlüsselten Containers gestartet werden. Das gewählte Dateisystem lässt sich später jederzeit durch Neuformatierung ersetzen. Die Initialisierung des Zufallszahlengenerators, der den Verschlüsselungsschlüssel des Containers erzeugt, erfolgt durch Mausbewegungen im Fenster des Assistenten. Danach wird der gesamte Datencontainer mit zufälligen Werten gefüllt, sofern weder die Option „Quick Format" noch ein dynamischer Container gewählt wurde.

3.2 Öffnen eines Containers

Ist der verschlüsselte Container erzeugt, muss er mit einem logischen Laufwerk verbunden werden, damit auf ihn zugegriffen werden kann. Das gelingt leicht aus der Hauptansicht mit drei Mausklicks (Auswahl logisches Laufwerk und Container-Datei) und Eingabe des vergebenen Passworts. Handelt es sich um einen versteckten Container („Hidden Volume"), entscheidet das eingegebene Passwort darüber, ob der „Gastgeber"-Container oder der im unbenutzten Bereich des Containers versteckte Container geöffnet wird. Anschließend kann mit dem Container wie mit einem logischen Laufwerk gearbeitet werden.

In der Konfiguration (unter „Einstellungen"/„Voreinstellung") lässt sich festlegen, ob TrueCrypt beim Hochfahren des Systems nach der Benutzeranmeldung automatisch gestartet und alle oder nur ausgewählte Datencontainer eingebunden werden sollen. Die für den automatischen Start vorgesehenen Datencontainer müssen zuvor als „Favorit" gekennzeichnet worden sein – dazu werden sie alle eingebunden und über den Menupunkt „Volumes" markiert („Momentan eingebundene Volumes als Favoriten speichern").

Eine Alternative zum automatischen Aufruf von TrueCrypt bei Systemstart ist die Anlage je eines „Mount"- und „Dismount"-Icons auf dem Desktop. Da TrueCrypt die Angabe von Kommandozeilenparametern erlaubt, kann das Einbinden eines Datencontainers mit den folgenden Parametern veranlasst werden:

- **`TrueCrypt.exe /v`** *`<Pfad und Name der Containerdatei>`* **`/l`***`<Name des zuzuweisenden logischen Laufwerks>`*
- Beispiel: **`TrueCrypt.exe /v Testvolume.dat /lx`**

Die Trennung eines Datencontainers über Kommandozeilenaufruf erfordert lediglich die Angabe des logischen Laufwerks:

- **`TrueCrypt.exe /d`***`<Name des logischen Laufwerks>`*
- Beispiel: **`TrueCrypt.exe /dx`**

Das sofortige Trennen („Dismounten") aller eingebundenen verschlüsselten Container kann auch mit einer Tastenkombination verknüpft werden, um eine „Notsperrung" auslösen zu können. Dazu muss lediglich unter „Einstellungen"/„Tastenkombinationen" die gewünschte Tastenkombination festgelegt werden. Nach einer „Notsperrung" wird der Passwort-Cache im Speicher sicher gelöscht.

Nachteil des verschlüsselten Datencontainers ist, dass die maximale Größe des Speicherbereichs bei der Erstellung festgelegt werden muss. Sofern bei der Erzeugung keine „dynamische" Größenentwicklung des Containers ausgewählt wurde, belegt er auf der betroffenen Partition des Speichermediums den vollen Speicherbereich, auch wenn im verschlüsselten Container noch keine einzige Datei abgelegt wurde. Eine Kompression der Daten nimmt TrueCrypt ebenfalls nicht vor. Daher benötigen Backups des Datencontainers immer den maximalen Platz – oder es müssen die Dateien einzeln und unverschlüsselt aus dem Container (also dem zugewiesenen logischen Laufwerk) auf das Backupmedium kopiert werden.

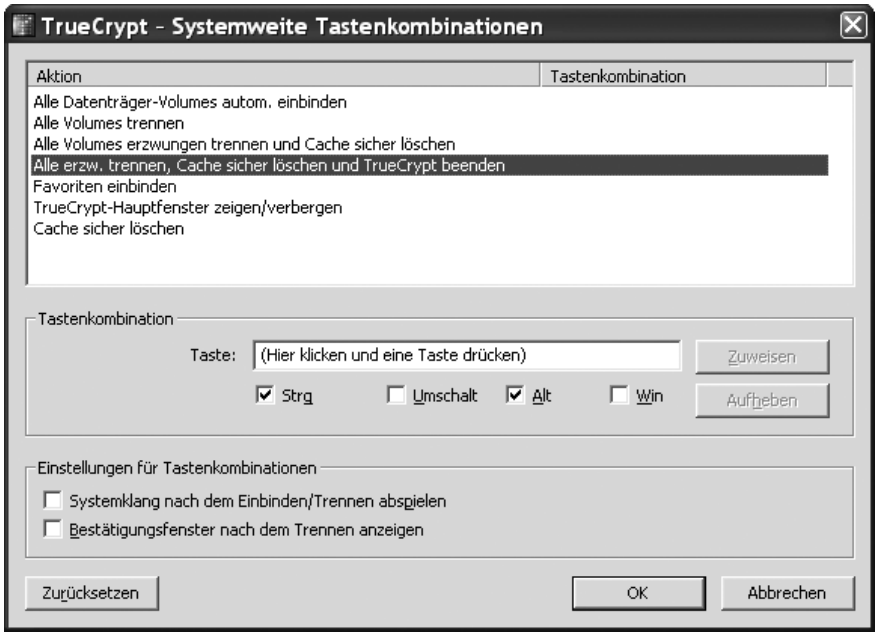
Gegenüber einer Verschlüsselung einzelner Dateien hat die Containerverschlüsselung jedoch – neben dem Vorzug einer größeren Bequemlichkeit – den Vorteil, dass aus den verschlüsselten Daten z. B. durch Dateigröße und Dateiname keine Rückschlüsse auf die Dateiinhalte möglich sind.

3.3 Verschlüsselter USB-Speicher

Auch auf einem USB-Speicher lassen sich verschlüsselte Datencontainer erzeugen. Damit die verschlüsselten Daten auf unterschiedlichen Systemen gelesen werden können, verfügt TrueCrypt über eine „Traveler Disk"-Option. Mit dieser Funktion kann TrueCrypt als aufrufbares Programm auf dem USB-Speicher installiert werden. Für diese Installation auf dem USB-Speicher (Aufruf über „Extras"/„Traveller Disk Setup") sind Administratorrechte erforderlich.

Die „mobile" Version von TrueCrypt kann so konfiguriert werden, dass sie über die Autostart-Funktion des Mediums aufgerufen wird und gleich den Datencontainer des USB-Mediums einbindet. Wurde bei der Verschlüsselung neben dem TrueCrypt-Passwort eine Schlüsseldatei verwendet, muss diese bei jeder Einbindung des Volumes verfügbar sein.

Abb. 2 | Auswahl einer Tastenkombination zur „Notabschaltung"



TrueCrypt-Datencontainer sind durch kein äußerliches Merkmal als Datencontainer zu erkennen: Weder eine TrueCrypt-spezifische Anwendungs-ID noch ein spezieller Header weisen auf das Vorliegens eines TrueCrypt-Containers hin. Solange der gewählte Dateiname keinen Hinweis gibt, lässt sich ein TrueCrypt-Volume damit wirksam sogar vor dem Erkennen durch Dritte schützen.

3.4 Gemeinsame Nutzung verschlüsselter Container

Verschlüsselte Datencontainer, die beispielsweise auf einem Netzlaufwerk abgelegt sind, können mit TrueCrypt auf einem lokalen System eingebunden („gemounted") werden. Dabei erfolgt nicht nur die Speicherung, sondern auch die Übermittlung der Daten über die Netzverbindung automatisch verschlüsselt.

Mehrere Nutzer können allerdings nur gleichzeitig auf diesen Datencontainer zugreifen, wenn er auf den unterschiedlichen Client-Systemen im „Nur-Lese"-Modus gestartet wird. Zwar ist grundsätzlich auch ein gemeinsamer Schreibzugriff möglich. Dazu muss der TrueCrypt-Container auf dem Server geöffnet werden; die Datenübermittlung erfolgt dann allerdings unverschlüsselt.

3.5 Schutzmaßnahmen und Restrisiken

Eine grundsätzliche Gefahr für verschlüsselte Datencontainer sind Bitfehler, die durch eine Fehlfunktion des Betriebssystems oder eines Programms verursacht werden – und nicht grundsätzlich ausgeschlossen werden können. Denn anders als bei einzelnen Dateien sorgt ein einziges fehlerhaftes Bit in einem Datencontainer dafür, dass alle Dateien des Containers unlesbar werden.

Dieser Gefahr kann nur durch regelmäßige Backups vorgebeugt werden. Sofern das Backupmedium nicht durch andere Maßnahmen wirksam vor unberechtigtem Zugriff geschützt ist, sollte auch das Backup verschlüsselt gespeichert werden. Dieser Vorgang lässt sich beschleunigen, indem mit TrueCrypt auf dem Backupmedium ein gleichgroßer Datencontainer angelegt, geöffnet und die Dateien dorthin übertragen werden.

Aber auch die Zwischenspeicherung des TrueCrypt-Schlüssels im Speicher, die per Maus-Click beim Verbinden des Datencontainers oder der Partition gewählt werden kann, ist nicht ungefährlich: Der Schlüssel bleibt im RAM, bis er explizit, z. B. über die Option „Extras"/„Kennwort-Cache sicher löschen" entfernt wird – und damit auch aus dem ausgelagerten RAM-Speicher der Festplatte.

Werden ausschließlich Nutzdaten in einem Datencontainer verschlüsselt, bleiben

zumindest unter dem Betriebssystem Windows zahlreiche Punkte, über die möglicherweise doch unberechtigt auf unverschlüsselte, sensible Daten zugegriffen werden kann. So bietet Windows die Möglichkeit, den Arbeitsspeicher (RAM) durch „virtuellen Speicher" zu erweitern – eine Standardkonfiguration, bei der ein fester Bereich der Festplatte für die zeitweilige Auslagerung von Daten aus dem RAM reserviert wird. Auf diese Weise können die Inhalte geöffneter Dateien ohne Wissen des Nutzers unverschlüsselt auf der Festplatte gespeichert werden, sofern diese Betriebssystemfunktion nicht deaktiviert wurde.

Ähnliches gilt, wenn der Rechner in den Stromspar-Mode (Standby- oder „Schlafmodus") versetzt wird. Dabei schreibt das Betriebssystem den gesamten Betriebszustand – also auch den Inhalt des flüchtigen RAM-Speichers inklusive aller geöffneten Dateien – auf die Festplatte. Zwar lässt sich TrueCrypt so konfigurieren, dass das im RAM gespeicherte TrueCrypt-Passwort gelöscht und alle TrueCrypt-Container heruntergefahren werden; das schützt die Inhalte der geöffneten Dateien jedoch nicht vor einer unverschlüsselten Speicherung auf der Festplatte.

Schließlich kann TrueCrypt ein automatisches Trennen verschlüsselter Partitionen oder Datencontainer nach Zeitablauf, bei Abmeldung vom Betriebssystem oder der Aktivierung des Bildschirmschoners erzwingen.

3 Partitions- und Systemverschlüsselung

Seit Version 5 verschlüsselt die Windows-Version von TrueCrypt unter Windows XP, Vista und Server 2003/2008 auch komplette Partitionen. Das können nicht nur Datenpartitionen (z. B. ein kompletter USB-Stick oder eine USB-Platte), sondern auch Systempartitionen sein. Damit wurde ein zentraler Nachteil von TrueCrypt gegenüber Lösungen für eine Vollverschlüsselung des Systems beseitigt: auch temporäre Daten, die Registry und RAM-Auslagerungen können nun ausnahmslos verschlüsselt werden. Anschließend ist ein Neustarten des Rechners erst nach Eingabe eines TrueCrypt-Boot-Passworts möglich.

Zur Sicherheit wird während des Verschlüsselungsprozesses einer Systempartition eine TrueCrypt Rescue Disk (auf CD oder DVD) erstellt, auf der der Boot-Loader und der Header der verschlüsselten System-Partition gesichert werden. Die Systempartition lässt sich später auch jederzeit als reine Datenpartition („Mount Without Pre-Boot Authentication") einbinden. Die Nutzung eines Authentisierungs-Token wie bspw. einer Chipkarte oder eines USB-Sticks sowie die Nutzung des TPM-Chips ist bisher nicht möglich, aber für eine spätere Version von TrueCrypt angekündigt.

4 Fazit

TrueCrypt ist eine interessante Alternative für diejenigen, die den Verschlüsselungsoptionen des Betriebssystems (wie der Laufwerksverschlüsselung Bitlocker unter Windows Vista) nicht trauen und eine transparente und möglichst flexible Lösung suchen. Die Lösung genügt sehr hohen Sicherheitsanforderungen und ist einfach zu bedienen; im praktischen Betrieb fällt sie nur durch die zusätzliche Passworteingabe überhaupt auf.

Sie eignet sich insbesondere für Privatnutzer und kleine Unternehmen. Spätestens bei Unternehmen mit mehr als 100 Nutzern dürfte der Aufwand für eine lokale Konfiguration und Pflege des Tools jedoch unvertretbar ausfallen. Das Fehlen einer zentralen Konfigurationsmöglichkeit und das Fehlen einer PKI-Anbindung macht TrueCrypt schon für mittelgroße Unternehmen als Lösung für eine durchgängige Verschlüsselung sensibler Daten im Normalfall weniger geeignet.

Auch beim „konkurrierenden" Zugriff mehrerer Nutzer auf einen gemeinsam genutzten verschlüsselten Datencontainer haben kommerzielle Produkte die Nase deutlich vorn – hier muss TrueCrypt passen.

Dennoch bleibt es ein hilfreiches Tool, das in vielen Projekten oder Prozessen sinnvoll eingesetzt werden kann, in denen eine zentrale Administration keine Einsatzbedingung ist.

Download

Download-Quelle für TrueCrypt:
http://www.truecrypt.org

Literatur

[IEEE_07] Carl Ellison: *Cryptographic Random Numbers*. Appendix to the P1363 standard; Dezember 2007.

[ISO_04] International Organization for Standardization (ISO): *Information technology – Security techniques – Hash-functions – Part 3: Dedicated hash-functions*, ISO/IEC 10118-3:2004, 24. Februar 2004

[Kali_00] Burt Kaliski: *PKCS #5: Password-Based Cryptography Specification*, RFC 2898, September 2000; http://tools.ietf.org/html/rfc2898

[NIST_01] NIST: *Advanced Encryption Standard (AES)*, FIPS PUB 197, 26.11.2001; http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[NIST_02] NIST: Secure Hash Standard, FIPS PUB 180-2, 01.08.2002; http://csrc.nist.gov/publications/fips/fips180-2/fips180-2withchangenotice.pdf

[Roga_04] Phillip Rogaway: Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC, Asiacrypt 2004, LNCS vol. 3329. Springer, 2004.