



Trusted Computing

Grundlagen – wichtige Eckpunkte

Praxis – Probleme

Dr. Christoph Wegener
wecon.it-consulting

E-Mail: wegener@wecon.net – Web: www.wecon.net

Der Referent stellt sich vor...

- Aktuelle Funktionen:
 - Horst Görtz Institut für IT-Sicherheit
 - **wecon.it**-consulting
- Interessens-Schwerpunkte:
 - IT-Sicherheit / Hochverfügbarkeit
 - Linux / Open Source Software
- Sonstige Aktivitäten ;)
 - Arbeitsgruppe "a-i3"
 - Fachgutachter / Fachlektor (Addison-Wesley, d.punkt-Verlag)
 - Fachartikel / Konferenzbeiträge (iX, c't, DuD, ...)
 - Diverse Programmkomitees (GUUG FFG, LinuxTag, Linux-Kongress, ...)



Was werde ich heute vorstellen?

- Einführung
 - Trusted Computing Konzepte der TCG / TCPA
 - Funktionen des Trusted Platform Module (TPM)
- Status Quo der Unterstützung
 - Hardware
 - Software
- Aktuelle Praxisbeispiele
- Chancen und Risiken
- Fazit

Digital Rights Management (DRM)

- Ziel von DRM-Systemen:
 - Abgrenzung zum reinen Kopierschutz (Verhindern illegaler Vervielfältigung)
 - DRMS erlauben dem Rechteinhaber digitaler Inhalte, zusätzlich ein Rechtemodell gegenüber dem Rechteinhaber durchzusetzen
- Rechtemodelle können beinhalten
 - Wiedergaberechte (Anhören, Ansehen, Ausdrucken, ...)
 - Transportrechte (Kopieren, Vermieten, Weitergeben, ...)
 - Derivativrechte (Extrahieren, Editieren, Einbinden, ...)
 - Dienstrechte (Sicherung, Caching, Integritätssicherung, ...)
- Geschäftsmodelle (Kombinationen möglich)
 - Zeitlich befristete Nutzung
 - Mengenabhängige Nutzung (n Wiedergaben / Aufrufe)
 - Geräteabhängige Nutzung
 - Gebrauchsorientierte Nutzung (n Minuten wiedergegeben)

Trusted Computing Platform Alliance (TCPA)



- Herstellerkonsortium – 1999 von Microsoft, Intel, IBM, Compaq und HP gegründet
- 2003 über 200 Mitglieder (u.a. Infineon, Siemens, RSA, Nokia)
- Einstimmige Entscheidungsfindung
- 07/2000 erste Veröffentlichung der Spezifikationen (v0.9)
- Zielsetzung: Hard- und Softwarestandards zu spezifizieren, um Vertrauen (Trust) in Computerplattformen zu erhöhen
 - Plattformen: Motherboard, CPU, weitere Geräte und Chipsätze
 - Vertrauen: Komponenten agieren so wie erwartet

Trusted Computing Group (TCG) I

- Gründung durch AMD, IBM, Intel und Microsoft
- Seit April 2003 Rechtsnachfolger der TCPA
- Nicht ganz so "basisdemokratisch" wie die TCPA
 - Kein Veto für Mitglieder mehr (2/3 Mehrheit)
- Verschiedene Mitgliedsstufen:
 - Promotor (50.000 \$/Jahr)
 - Contributor (15.000 \$/Jahr)
 - Adopter (7.500 bzw. für kleine Firmen 1.000 \$/Jahr, kein Stimmrecht)
 - Seit Mitte 2004 "Industry Liaison Program" (keine Kosten, kein Stimmrecht, NDA erforderlich)
- Mitglieder Juli 2006: 140 (8 Promotor, 80 Contributor, 52 Adopter)

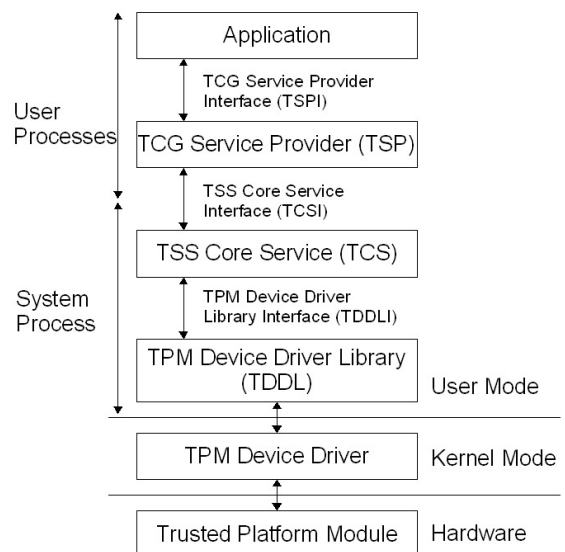


Trusted Computing Group (TCG) II

- Ziel: Entwicklung und Support von offenen Industriestandards für "Trusted Computing" aus verschiedenen Plattformen (PCs, Server, Laptops, Mobiles, Storage, Netzwerk, ...)
- Arbeitsgruppen der TCG:
 - Infrastructure Group
 - PC Client Group
 - Software Stack Group
 - Trusted Network Group (TNG)
 - Trusted Platform Module (TPM) Group
 - Mobile Group
 - Server Group
 - *Storage Group*
 - Trusted Send (In) und Trusted Receive (Out) Kommandos
 - Whitepaper unter: www.trustedcomputinggroup.org/groups/storage

TCG – Architektur und Spezifikationen

- Hardware
Trusted Platform Module (TPM)
- Software
Trusted Software Stack (TSS)
- Wichtige Spezifikationen:
 - TCG TPM Main Specification (alte Version – v1.1b)
 - *TCG TPM Specification v1.2* (November 2003, aktuelle Revision 94 März 2006, 696 Seiten)
 - TCG Software Stack Specification v1.1 (September 2003)
 - *TCG Software Stack Specification v1.2* (Januar 2006, 742 Seiten)



"Roots of Trust" des TCG Subsystems

- *Root of Trust for Measurement (RTM)*
 - Bestimmt beim Booten die Integrität einer Konfiguration
 - Wird vom TPM (Hash-Funktion, PCRs) und Teilen des BIOS erledigt
- *Root of Trust for Storage (RTS)*
 - Schützt Schlüssel und Daten, denen das TPM vertrauen muss
 - Auch für Daten ausserhalb des TPM
 - Wird vom TPM erledigt
- *Root of Trust for Reporting (RTR)*
 - Erstellt auf vertrauenswürdige Weise Bescheinigungen über die Integrität von Daten, die RTS verwaltet (beispielsweise Attestierungen)
 - Wird vom TPM erledigt
- TCG Subsystem bestehend aus TPM und TSS stellt Betriebssystemen vertrauenswürdige Dienste und Mechanismen zur Verfügung

TCG Architektur – Designziele

- Authentifizierung der Systemkomponenten und deren Konfiguration (**sicheres Booten**)
- Sicheres Generieren und **Schutz kryptographischer Schlüssel** (Hardwarespeicher)
- *Remote Platform Attestation*
 - Plattformbenutzer fragt eine Dienstleistung bei einem Anbieter an
 - Dienstleister stellt eine Attestierungsanfrage an das TPM
 - **Root of Trust for Reporting (RTR)** als Teil des TPM erstellt eine entsprechende Attestierung, die den Integritätszustand der Konfiguration beschreibt und signiert diese (mit dem passenden AIK)
 - Diensteanbieter erfragt die Gültigkeit der Signatur bei einer CA
 - Diensteanbieter überprüft die Konfiguration
- *Sealing*
 - Systemkonfiguration wird beim Booten bestimmt
 - Über einen Hash-Wert aus der Systemkonfiguration werden Daten und Applikationen an diese Konfiguration "gebunden"
 - Ver- und Entschlüsselung funktioniert nur anhand dieser Konfiguration

Funktion des Trusted Platform Module (TPM)

- Nach US-Senator Fritz Hollings (DRM-Verfechter) auch "Fritz Chip" genannt
- Chip auf Motherboard (geplant ist aber auch eine Integration in CPUs und andere Bausteine)
- Kryptographische Funktionseinheiten
 - Random Number Generator (RNG)
 - Hash-Einheit (SHA-1)
 - HMAC (Keyed Hashing for Message Authentication)
 - Generator für RSA-Schlüssel mit bis zu 2.048 Bit
 - RSA-Einheit zum Erzeugen von Signaturen (nicht prüfen), sowie zum Ver- und Entschlüsseln
- Enthält einen vertrauenswürdigen Zeitgeber ("timer")
- Führt beim Start einen Selbsttest auf Manipulation durch und vermerkt das Ergebnis (meldet es aber nicht aktiv und deaktiviert sich auch nicht selbst)

TPM – Ein Blick ins Innere

Funktionale Einheit	Nicht flüchtiger Speicher	Flüchtiger Speicher
Zufallszahlengenerator	Endorsement Key (2048 Bit)	RSA Key Slot-0 ... RSA Key Slot-9
Hash	Storage Root Key (2048 Bit)	PCR-0 ... PCR-15
HMAC	Owner Auth Secret (160 Bit)	Key Handles
RSA-Schlüsselgenerator		Auth Session Handles
RSA-Ver- und -Entschlüsselung		

TPM – Nicht flüchtiger Speicher I

- *Endorsement Key (EK)*
 - 2.048 Bit RSA-Schlüsselpaar
 - Vom Hersteller im TPM generiert oder hinein geschrieben und signiert
 - Nicht löscht- oder änderbar
(ab TPM v1.2 ist ein Löschen möglich, wenn EK als "löschtbar" gekennzeichnet)
 - Privater Teil des EK verlässt das TPM nie
 - Öffentlicher Teil des EK Basis der "Attestation", aus Sicht der Privatsphäre kritisch (vgl. Seriennummern von Intel-Prozessoren) -> AIKs
 - Herausgabe des öffentlichen Teils des EK kann abgeschaltet werden
 - Öffentlicher Teil zur Verschlüsselung von sensiblen Daten, die an den Chip gesendet werden (zum Beispiel beim "Besitz übernehmen")
- *Attestation Identity Keys (AIK)*
 - Mit EK signierte pseudonyme Schlüssel (beliebig viele)
 - Bestätigt Vorhandensein und Konfiguration des TPM (zum Beispiel PCRs) ohne den öffentlichen Teil des EK selbst herauszugeben
 - Signierte AIKs werden mit EK-Zertifikat nur an vertrauenswürdige Zertifizierungsstellen (Privacy CA) herausgegeben
 - AIKs können auch verschlüsselt außerhalb des TPM aufbewahrt werden

TPM – Nicht flüchtiger Speicher II

- *Storage Root Key (SRK)*
 - 2.048 Bit RSA Schlüsselpaar
 - Initial ist der Speicherplatz leer
 - Wird beim "Besitz übernehmen" generiert
 - Privater Teil verlässt den Chip nie
 - Kann vom Systembesitzer gelöscht werden
 - Bildet die Wurzel einer Schlüsselhierarchie
 - Dient zum Verschlüsseln ("wrap") von anderen Schlüsseln der ersten Hierarchiestufe, die außerhalb des Chips gespeichert werden, sowie zum Entschlüsseln dieser Schlüssel, wenn sie wieder in den Chip geladen werden
- *Owner Authorization*
 - 160 Bit Schlüssel den der Besitzer mit dem Chip teilt
 - SHA-1 Hash des angegebenen Passworts mit EK verschlüsselt
 - Das Passwort selber kann 256 Byte lang sein
 - Wird beim "Besitz übernehmen" im Chip erzeugt
 - Autorisierung von sensiblen Benutzerbefehlen

TPM – Flüchtiger Speicher I

- 10 Slots für RSA-Schlüssel
 - Extern gespeicherte Schlüssel können hier, nach Eingabe des Passworts, in den Chip geladen und genutzt werden
 - Können aus dem Slot geworfen ("evicted") werden, um den Platz frei zu geben
- 16 Slots für Platform Configuration Register (PCRs)
 - 160 Bit für ermittelte Hash-Werte der Integritätsmessungen
 - Folge von Integritätswerten möglich:
 $PCR(i) = HASH(PCR(i-1), \text{Wert})$
 - Zugriff nur im Rahmen von Sicherheitsdiensten
 - Beim Booten können zum Beispiel Messungen vom BIOS, erweitertem BIOS, MBR und anderen Daten (z.B. Kernel), aber auch von Hardware, die dies unterstützt, erzeugt und hier gespeichert werden
 - Ab TPM v1.2 sind für die PC-Plattform 24 Register vorgesehen

Linux – Inhalt eines TPM Chips (PCRs)

```
root@T42p:/home/wd/TPM/bin
[ root@T42p bin ]# ./tpm_demo
TPM successfully reset
TPM version 1.1.0.6
16 PCR registers are available
PCR-00: 0B 9F 96 F0 AF 4B 9B 6D 01 1A 94 F0 21 AB 61 7B C1 8F DD 66
PCR-01: F3 FF 4E 59 CA 32 50 51 E4 56 3A 48 8E EA 3D 4F ED 56 0B 7B
PCR-02: EB B3 BA AE E7 57 4B B6 37 AA AB 67 0F 9A C1 BC EB 6F 80 F3
PCR-03: 04 FD EC DD 50 1D AF 0F 62 4C 1F 99 60 12 CF 30 44 FF 46 10
PCR-04: 50 11 E7 E6 24 64 4B AB F1 A4 00 FB 34 1C 91 6E 52 2B F7 98
PCR-05: 04 FD EC DD 50 1D AF 0F 62 4C 1F 99 60 12 CF 30 44 FF 46 10
PCR-06: 04 FD EC DD 50 1D AF 0F 62 4C 1F 99 60 12 CF 30 44 FF 46 10
PCR-07: 04 FD EC DD 50 1D AF 0F 62 4C 1F 99 60 12 CF 30 44 FF 46 10
PCR-08: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-09: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-10: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-11: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-12: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-13: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-14: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-15: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10 Key slots are available
Key Handle 711200 loaded
Pubek keylength 256
```

TPM – Flüchtiger Speicher II

- Key Handles
 - Um temporär geladenen Schlüsseln Namen zur weiteren Verwendung zuzuweisen
 - Werden gelöscht, wenn der Schlüssel aus dem Chip geworfen wird
- Authorization Session Handle
 - Wird genutzt, um den Status der Authorisation für mehrere hintereinanderabfolgende Befehle beizubehalten
- Data Integrity Register
 - Ab TPM v1.2 zusätzliche, mindestens 160 Bit große Speicherplätze

TPM – v1.2 Spezifikations-Änderungen I

- Motivation
 - Anpassung an neue (technische) Entwicklungen und Voraussetzungen
 - Forderungen von Kritikern (insbesondere zur Verbesserung der "privacy")
- "Neue" Features
 - *Direct Anonymous Attestation (optional)*
 - Variable Anonymity ("Base" der Attestation ist zufällig gewählt)
 - Named Based Solutions ("Base" zufällig, aber konstant für Zeit t)
 - *Locality (optional)*
 - 4 unterschiedliche Stufen der Vertraulichkeit
 - > Vertrauenskontexte von HW und SW
 - *Delegation (mandatory)*
 - Feingranulierte, objektabhängige Rechte bei TPM-Kommandos

TPM - v1.2

Spezifikations-Änderungen II

- *NV Storage (mandatory)*
 - Übergang von fester Größe zu einer "storage facility"
- *Transport Protection (mandatory)*
 - Kommunikationskanäle (nach extern) können verschlüsselt werden
 - "Checksummen" für TPM-Kommandos
- *Monotonic Counters (mandatory)*
 - (Verbesserter) Schutz vor Replay-Attacken
- *Tick Counters (mandatory)*
 - Sicherer Zeitstempel
- *Context Save and Restore (mandatory)*
 - Unterstützung Multi-User-/Multi-Software-Environment

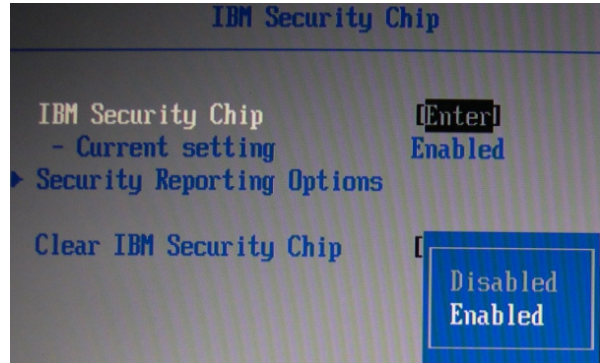
TPM - v1.2

Spezifikations-Änderungen III

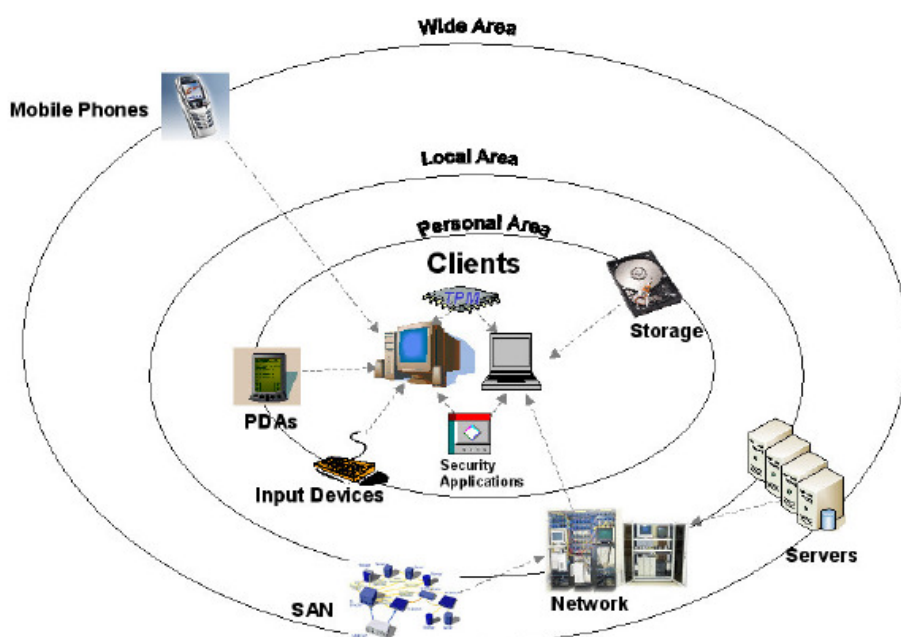
- *Clear Endorsement Key (optional)*
 - EK jetzt löschar / ersetzbar, wenn entsprechend markiert
- *General Purpose I/O Functions (optional)*
 - "Geschützter", physikalischer Kommunikationskanal mit anderer Hardware
- *Certified Migration (optional)*
 - "Certified Migratable Keys" (kombinieren "Certification" und "Migration")
 - Migration Selection Authority: Kontrolle der Migration der Keys
 - Migration Authority: Eigentliche Migration der Keys
- *Maintenance (herstellerspezifisch und optional)*
 - Kann auch als nicht-migrierbar markierte Keys "verschieben"
 - Aber "optional": Was passiert, wenn TPM verloren oder defekt?
- Optionale Funktionen werden (zur Zeit noch) nicht implementiert!

BIOS-Funktionalität: Aktivieren und Löschen des TPM

- BIOS gibt TPM beim Einschalten des Rechners ein "Startkommando"
- Dabei gibt es drei Möglichkeiten:
 - TPM deaktivieren
(kann bis zum (Wieder-)Einschalten nicht mehr aktiviert werden)
 - TPM starten und Reset der PCRs, dann Inhalte der PCRs beim Booten neu berechnen
 - TPM starten und PCRs wieder herstellen, falls sie vorher gespeichert wurden (Resume-Modus)
- BIOS kann TPM "komplett" resettet (*ForceClear*)
 - Benötigt Beweis der physikalischen Präsenz (z.B. bei IBM Thinkpad: Fn beim Systemstart gedrückt halten und dann mit F1 ins BIOS wechseln)
 - Wirft alle geladenen Schlüssel und Handles raus und löscht den SRK sowie das Owner Authorization Secret



TPM – Zentrum der Sicherheit

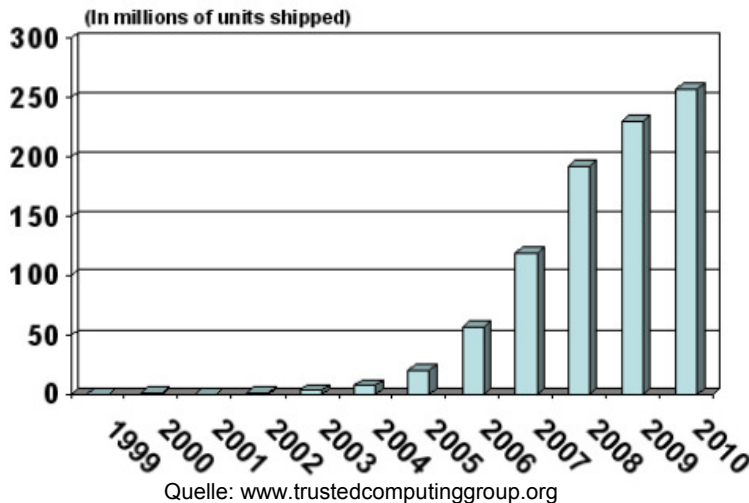


Quelle: www.trustedcomputinggroup.org

Status Quo – TPM Hardware

- Über 16 Millionen Motherboards mit TPM (v1.1b) sind ausgeliefert
 - STMicroelectronics, Atmel und Infineon fertigen bereits seit 2005 v1.2er TPMs

TPM Module Forecast



... und was ist mit den Spezifikationen?

- Wie überprüft man die Einhaltung der Spezifikation?
 - Testsuite der EMSCB-Gruppe
 - Testet ein Subset aller möglichen Befehle
 - "Functional Tests": Protokoll
 - "Integrity Tests": Manipulation der Eingabewerte
 - "Stress Tests": Reale Bedingungen (Sequenz von Befehlen)
- Tests an insgesamt 5 TPM-Chips von 4 Herstellern
 - v1.1b Chips von Infineon, Atmel und National Semiconductor
 - v1.2 Chips von Infineon und ST Microelectronics
- Ergebnis: 2 von 5 Chips erfüllen Spezifikation "vollständig"
 - Infineon SLB 9635 (TPM v1.2 – Motherboard Intel D865 GLC)
 - National Semiconductor (TPM v1.1b – IBM Thinkpad T43)
- Vollständiges Paper unter:
<http://www.prosec.rub.de/tpmcompliance.html>

Hardware: Verfügbar / angekündigt

- **Verfügbare Komponenten (Beispiele)**
 - Hauptsächlich Komplettrechner oder Motherboards für Unternehmenskunden (u.a. von IBM, HP und Fujitsu-Siemens)
 - IBM verbaut TPM v1.1b innerhalb des "Embedded Security Subsystems"
 - Von Intel bereits Chipsätze und Motherboards mit TPM v1.2 erhältlich
 - Apple Rechner mit Intel-CPU's und TPM
 - Netzwerkkarten mit TPM
- **Angekündigt sind unter anderem**
 - TPM integriert in I/O-Chip von National Semiconductor mit Ports für Tastatur, Maus, Drucker, Floppy, RS-232
 - Trusted Mode Keyboard Controller (Intel, Microsoft)
 - USB-Security-Extension (Intel, Microsoft)
 - TPM in CPU (Intel, AMD)
 - *TPM-Storage Lösungen: HDDs mit TPM*

Status Quo Software

- **Kommerzielle Applikationen mit TPM-Support (Beispiele)**
 - Utimaco SafeGuard (Laufwerksverschlüsselung)
 - Check Point VPN-1 SecureClient
 - Adobe Acrobat
(Verschlüsselung / Zugriffskontrolle von PDF-Dokumenten – DRM)
- **TPM-Unterstützung im Linux Kernel ab v2.6.12**
- **Software von IBM**
 - Windows: verändertes Login, rudimentäre Verschlüsselungswerkzeuge
 - Linux-Testpaket (samt Quellen) mit Kernel-Modul, Bibliothek, API und Beispielprogrammen
 - tcgLinux: TPM-based Linux Run-time Attestation
- **Forschungsprojekte**
 - Enforcer Linux Security Module
 - Trusted GRUB
 - PERSEUS
 - European Multilateral Secure Computing Base (EMSCB)
 - Projekt "Turaya"

Microsofts Paladium

Next Generation Secure Computing Base (NGSCB)

Microsoft Vista – Microsoft BitLocker

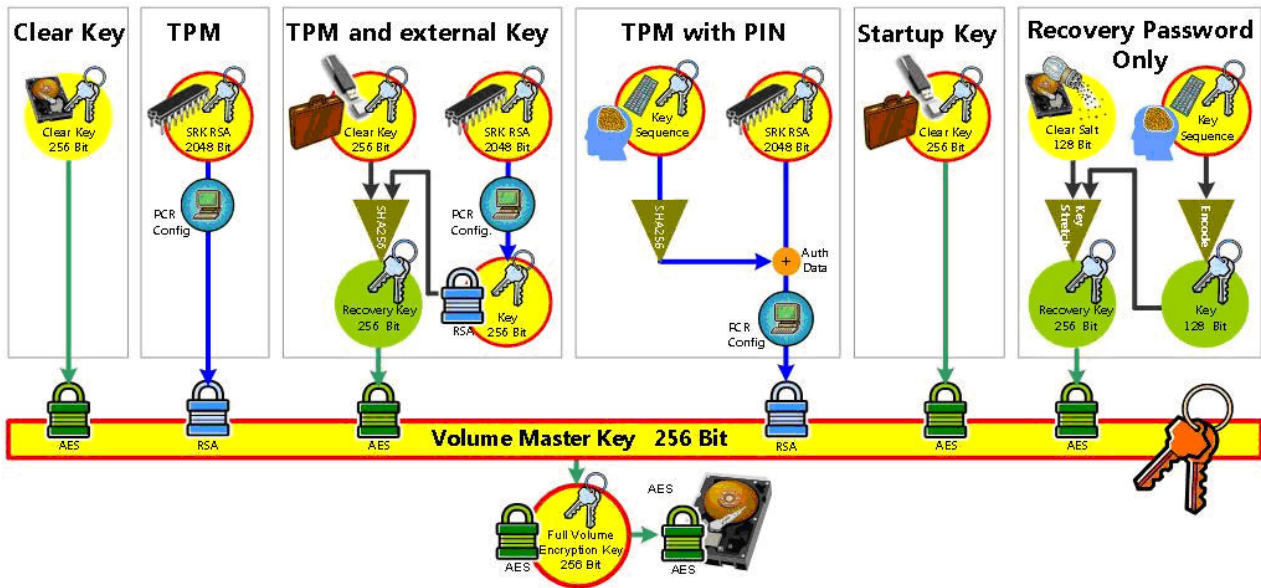


- **Vorschlag 2003**
 - Windows in Quadranten Aufteilen
 - Left Hand Site (LHS) – ungesicherte Windows-Umgebung
 - Right Hand Site (RHS) – Anwendungen im Trusted Mode
 - LHS und RHS haben jeweils Benutzer- und Kernel-Modus
 - Nexus im Kernel-Modus in der RHS
- **Ankündigungen I: WinHEC 2004**
 - LHS nahezu unverändert / RHS wird komplett überarbeitet
 - Compartments – abgeschottete virtuelle Systeme parallel zum Haupt-OS
- **Ankündigungen II: WinHEC 2005**
 - Secure Startup
 - Full Volume Encryption in Longhorn
 - Unterstützung v1.2er TPM
 - Compartments in Longhorn Server erst frühestens 2007
- **Ankündigung "BitLocker"**

Microsoft's BitLocker – Features

- **BitLocker™ Drive Encryption**
 - Verschlüsselt gesamte Platte
 - Nutzt TPM v1.2 um pre-OS Komponenten zu validieren
- **Pre-OS Schutz**
 - USB Startup-Key, PIN und TPM-backed Authentifizierung
- **"Singulärer" Microsoft TPM Driver**
 - Erhöhte Stabilität und Sicherheit ;)
- **TPM Base Services (TBS) für Anwendungen von Dritt-Anbietern**
- **Active Directory Backup**
 - Automatisches Key-Backup zum AD-Server
 - Unterstützung von Gruppenrichtlinien
- **Scriptbares Interfaces**
 - TPM- / BitLocker-Management
 - Kommandozeilen-Tool

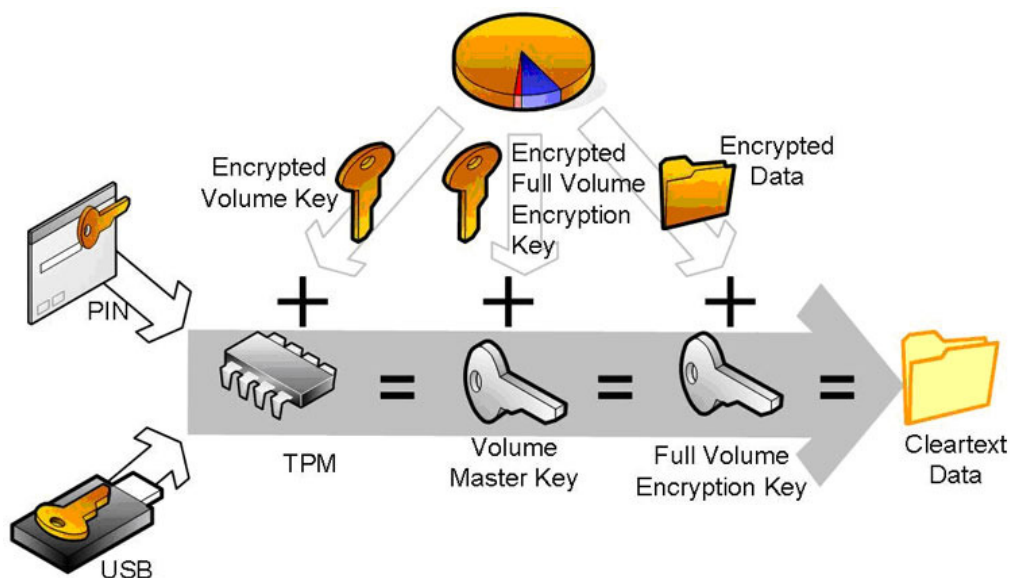
Microsoft's BitLocker – Keys



Quelle: <http://www.microsoft.com/technet/windowsvista/security/bittech.mspx#ERMAE>

Microsoft's BitLocker – HDD-Layout

- "Recovery" durch Backup (Auslagerung oder AD) ;)



Quelle: <http://www.microsoft.com/technet/windowsvista/security/bittech.mspx#ERMAE>

Microsoft's BitLocker – Voraussetzungen

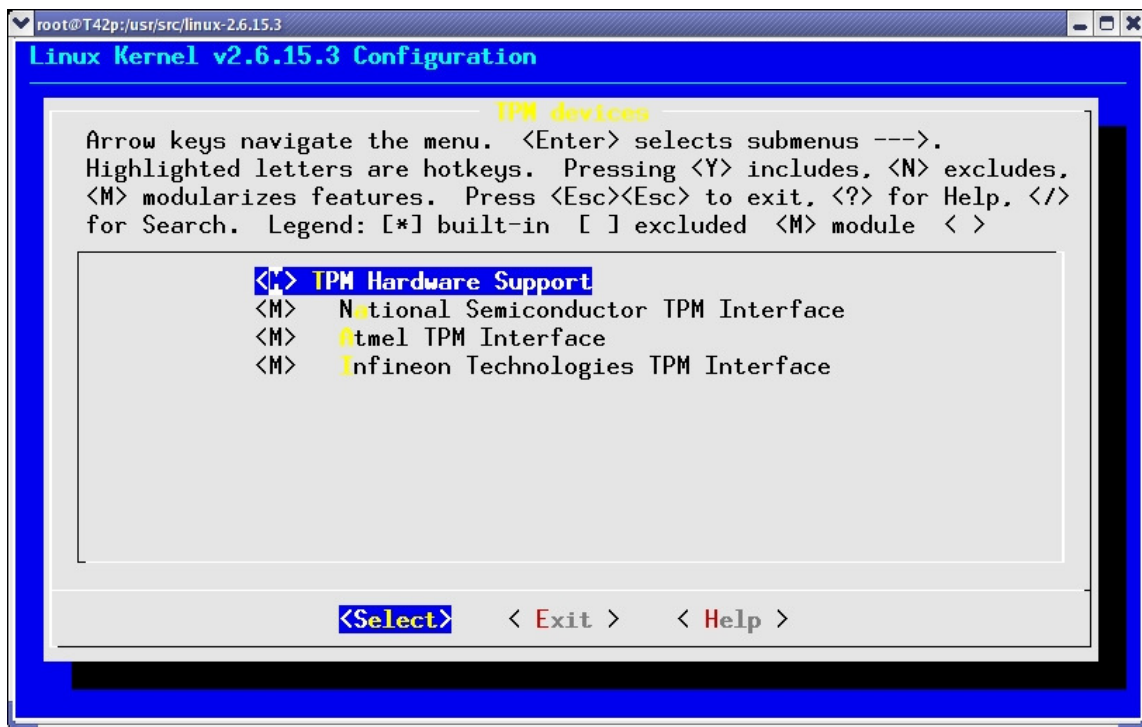
- Trusted Platform Module (TPM) v1.2.
- TCG-compliant (Trusted Computing Group) v1.2 BIOS
- Das BIOS muss das Lesen und Schreiben von "kleinen" Dateien auf einem USB Flash-Drive während vor dem Booten unterstützen.
- Es werden mindestens zwei Partitionen benötigt:
 - Operating System Partition
 - Muss NTFS sein.
 - Enthält Windows OS und die "support files"; die Daten auf dieser Partition sind durch BitLocker geschützt.
 - System Volume Partition
 - Muss NTFS sein, darf nicht mit OS Volume identisch und darf NICHT encryptet sein.
 - Enthält Hardware-spezifische Dateien, die zum Laden von Windows nach dem BIOS-POST gebraucht werden.

"Kommerzielle" Software (Auswahl)

- IBM's "Client Security"
 - Schützt Daten und Passwörter mittels TPM
 - Auf vielen IBM Laptops integriert
 - Kurze Demo
- Wave System's "Embassy Trust Suite"
 - "Suite" von Tools zur Nutzung eines TPM
 - Daten, Passwörter
 - Attestierung von Komponenten
 - Und: Schnittstelle für Programmierer
- Infineon's "TPM Professional Package 3.0"
 - TPM Management Software
 - Und: Schnittstelle für Programmierer
- ...

Linux – Kernel 2.6.15

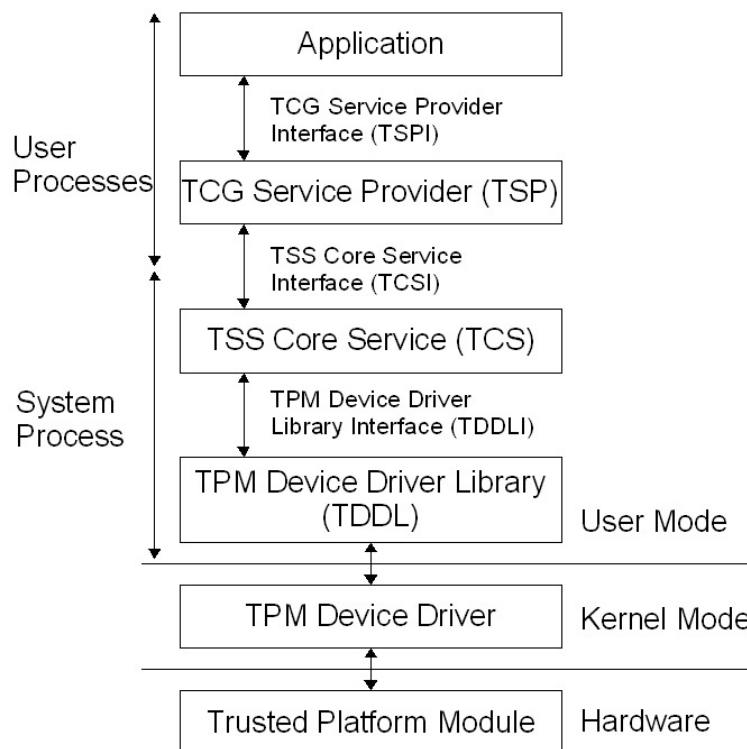
TPM-Unterstützung



tpmdd – TPM Kernel Modul

```
root@T42p:~  
[root@T42p ~]# ls /lib/modules/2.6.15.3/kernel/drivers/char/tpm/  
tpm_atmel.ko tpm_infineon.ko tpm.ko tpm_nsc.ko  
[root@T42p ~]#  
[root@T42p ~]# modprobe -v tpm  
insmod /lib/modules/2.6.15.3/kernel/drivers/char/tpm/tpm.ko  
[root@T42p ~]#  
[root@T42p ~]# modprobe -v tpm_atmel  
insmod /lib/modules/2.6.15.3/kernel/drivers/char/tpm/tpm_atmel.ko  
[root@T42p ~]#  
[root@T42p ~]# ls -l /dev/tpm*  
crw----- 1 root root 10, 224 7. Feb 22:25 /dev/tpm0  
[root@T42p ~]#
```

TCG Software Stack



TSS als Open Source – TrouSerS

- Quelloffener TCG Software Stack (TSS)
- CPL (Common Public License)
- Erfüllt TSS v1.1 Spezifikation
- TCS Daemon (tcsd)
 - User-Space-Dienst (gemäß der Spezifikationen einziger Zugang zum TPM)
 - Wird beim Booten gestartet, dann müssen alle Anfragen an das TPM über diesen Dienst laufen
 - Behandelt lokale und entfernte (z.B. "Remote Attestation") Anfragen
- TSP (TCG Service Provider) shared Library
 - Stellt dem TSCD und Anwendungen Ressourcen zur Verfügung und verwaltet diese
- Persistent Storage Files
 - User Persistent Storage – über die Lebensdauer einer Applikation
 - System Persistent Storage – Lebenszyklus des Systems oder des TSCDs

"Lieferumfang" von TrouSerS

```
root@T42p:/home/wd/TCPA2006/trousers-0.2.5/man/man3
[Root@T42p man3]# ls
Makefile
Makefile.am
Makefile.in
Tspi_ChangeAuth.3
Tspi_ChangeAuthAsym.3
Tspi_Context_Close.3
Tspi_Context_CloseObject.3
Tspi_Context_Connect.3
Tspi_Context_Create.3
Tspi_Context_CreateObject.3
Tspi_Context_FreeMemory.3
Tspi_Context_GetCapability.3
Tspi_Context_GetDefaultPolicy.3
Tspi_Context_GetKeyByPublicInfo.3
Tspi_Context_GetKeyByUUID.3
Tspi_Context_GetRegisteredKeyByUUID.3
Tspi_Context_GetTpmObject.3
Tspi_Context_LoadKeyByBlob.3
Tspi_Context_LoadKeyByUUID.3
Tspi_Context_RegisterKey.3
Tspi_Context_UnregisterKey.3
Tspi_Data_Bind.3
Tspi_Data_Seal.3
Tspi_Data_Unbind.3
Tspi_Data_Unseal.3
Tspi_GetAttribData.3
Tspi_GetAttribUint32.3
Tspi_GetPolicyObject.3
Tspi_Hash_GetHashValue.3
Tspi_Hash_SetHashValue.3
Tspi_Hash_Sign.3
Tspi_Hash_UpdateHashValue.3
Tspi_Hash_VerifySignature.3
Tspi_Key_CertifyKey.3
Tspi_Key_ConvertMigrationBlob.3
Tspi_Key_CreateKey.3
Tspi_Key_CreateMigrationBlob.3
Tspi_Key_GetPubKey.3
Tspi_Key_LoadKey.3
Tspi_Key_UnloadKey.3
Tspi_PcrComposite_GetPcrValue.3
Tspi_PcrComposite_SelectPcrIndex.3
Tspi_PcrComposite_SetPcrValue.3
Tspi_Policy_AssignToObject.3
Tspi_Policy_FlushSecret.3
Tspi_Policy_SetSecret.3
Tspi_SetAttribData.3
Tspi_SetAttribUint32.3
Tspi_TPM_AuthorizeMigrationTicket.3
Tspi_TPM_CertifySelfTest.3
Tspi_TPM_CheckMaintenancePubKey.3
Tspi_TPM_ClearOwner.3
Tspi_TPM_CollateIdentityRequest.3
Tspi_TPM_CreateEndorsementKey.3
Tspi_TPM_CreateMaintenanceArchive.3
Tspi_TPM_DirWrite.3
Tspi_TPM_GetCapability.3
Tspi_TPM_GetEvent.3
Tspi_TPM_GetEventLog.3
Tspi_TPM_GetEvents.3
Tspi_TPM_GetPubEndorsementKey.3
Tspi_TPM_GetRandom.3
Tspi_TPM_GetStatus.3
Tspi_TPM_GetTestResult.3
Tspi_TPM_KillMaintenanceFeature.3
Tspi_TPM_LoadMaintenancePubKey.3
Tspi_TPM_PcrExtend.3
Tspi_TPM_PcrRead.3
Tspi_TPM_Quote.3
Tspi_TPM_SelfTestFull.3
Tspi_TPM_SetStatus.3
Tspi_TPM_StirRandom.3
Tspi_TPM_TakeOwnership.3
[Root@T42p man3]#
```

TrouSerS und TPM-Tools

```
root@T42p:/home/wd/TCPA2006/trousers-0.2.5
[Root@T42p trousers-0.2.5]# find /usr/local/ -user tss
/usr/local/sbin/tcsd
/usr/local/etc/tcsd.conf
/usr/local/var/lib/tpm
[Root@T42p trousers-0.2.5]# ls -l /usr/local/lib/lib*
-rw-r--r-- 1 root root 6242 7. Feb 22:40 /usr/local/lib/libtddl.a
-rwxr-xr-x 1 root root 945 7. Feb 22:40 /usr/local/lib/libtspi.la
lrwxrwxrwx 1 root root 16 7. Feb 22:40 /usr/local/lib/libtspi.so -> libtspi.so.0.0.6
lrwxrwxrwx 1 root root 16 7. Feb 22:40 /usr/local/lib/libtspi.so.0 -> libtspi.so.0.0.6
-rwxr-xr-x 1 root root 220578 7. Feb 22:40 /usr/local/lib/libtspi.so.0.0.6
[Root@T42p trousers-0.2.5]#
```

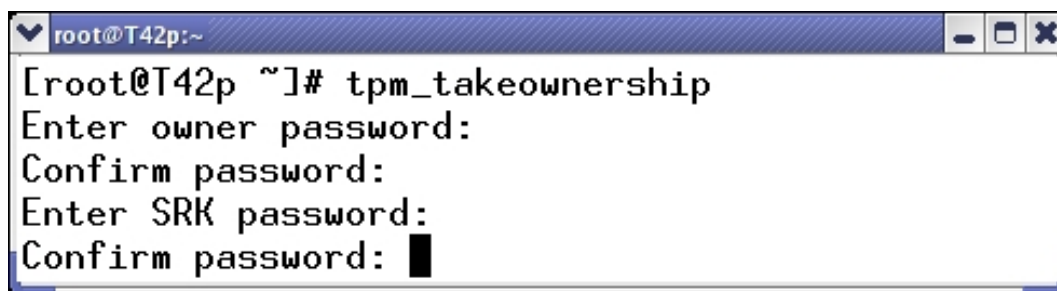
```
root@T42p:/home/wd/TCPA2006/tpm-tools-1.2.2 <2>
[Root@T42p tpm-tools-1.2.2]# ls -l /usr/local/sbin/tpm*
-rwxr-xr-x 1 root root 49121 7. Feb 23:43 /usr/local/sbin/tpm_changeownerauth
-rwxr-xr-x 1 root root 46896 7. Feb 23:43 /usr/local/sbin/tpm_clear
-rwxr-xr-x 1 root root 45641 7. Feb 23:43 /usr/local/sbin/tpm_createek
-rwxr-xr-x 1 root root 45876 7. Feb 23:43 /usr/local/sbin/tpm_getpubek
-rwxr-xr-x 1 root root 46884 7. Feb 23:43 /usr/local/sbin/tpm_restrictpubek
-rwxr-xr-x 1 root root 47189 7. Feb 23:43 /usr/local/sbin/tpm_selftest
-rwxr-xr-x 1 root root 47592 7. Feb 23:43 /usr/local/sbin/tpm_setactive
-rwxr-xr-x 1 root root 47551 7. Feb 23:43 /usr/local/sbin/tpm_setclearable
-rwxr-xr-x 1 root root 47457 7. Feb 23:43 /usr/local/sbin/tpm_setenable
-rwxr-xr-x 1 root root 46961 7. Feb 23:43 /usr/local/sbin/tpm_setownable
-rwxr-xr-x 1 root root 52344 7. Feb 23:43 /usr/local/sbin/tpm_setpresence
-rwxr-xr-x 1 root root 46552 7. Feb 23:43 /usr/local/sbin/tpm_takeownership
-rwxr-xr-x 1 root root 45661 7. Feb 23:43 /usr/local/sbin/tpm_version
[Root@T42p tpm-tools-1.2.2]# ls -l /usr/local/bin/tpm*
-rwxr-xr-x 1 root root 64671 7. Feb 23:43 /usr/local/bin/tpm_sealdata
[Root@T42p tpm-tools-1.2.2]#
```

TPM – "Besitzerkonzept"

- TPM wird immer deaktiviert ausgeliefert
- Vor dem Gebrauch muss der "Besitzer" der TCG Plattform das TPM explizit aktivieren
- Ohne die Aktivierung sind keinerlei Dienste des TPM verfügbar
- Explizite Besitzübernahme notwendig (TPM_TakeOwnership-Kommando)
- Über die Kenntnis eines Passworts (oder physikalischen Zugriff) kann dann der Zugriff kontrolliert werden
- Beim direkten Zugang zur Plattform kann das TPM auch ohne Wissen des "Besitzers" aktiviert bzw. deaktiviert werden (mittels BIOS oder Jumper)

Besitz übernehmen: tpm_takeownership

- tpm_takeownership
 - Installation des angegebenen Owner Authorization Secret im TPM
 - Erzeugen des Storage Root Key (SRK-Schlüsselpaar) im TPM
 - Das SRK Authorization Secret wird auf den Schlüssel angewandt
 - Ausgabe des öffentlichen Teils des SKR Schlüsselpaars



```
root@T42p:~  
[root@T42p ~]# tpm_takeownership  
Enter owner password:  
Confirm password:  
Enter SRK password:  
Confirm password: █
```

... und weitere TPM-Projekte

- IBM tcgLinux – TPM-based Linux Run-time Attestation
 - Erweitert Integritätsprüfung vom Boot-Prozess auf alle geladenen Programme bzw. Konfigurationsdateien
 - Anfragendes System benötigt Hash-Wert-Datenbank
- Enforcer Linux Security Module
 - Linux Security Module (LSM) / Baut auf den IBM-Treibern für Linux auf
 - Gleicht beim Lesen von sensiblen Daten Hash-Werte mit Datenbank ab
 - Modifizierter LILO überprüft Kernel Image und Master Boot Record
- PERSEUS
 - Security-Softwareschicht kontrolliert zu Schutz von sensiblen Anwendungen und Daten kritische Hardware-Ressourcen (auch das TPM)
 - Baut auf L4-Microkernel auf (Codebasis PERSEUS max. 100.000 Zeilen)
- European Multilateral Secure Computing Base (EMSCB)
 - Vorschlag für offene Computing Plattform
 - Soll PERSEUS, TPM und herkömmliche Betriebssysteme kombinieren
 - Projekt "Turaya"

Beispiel 1: Sicheres Booten mit TPM

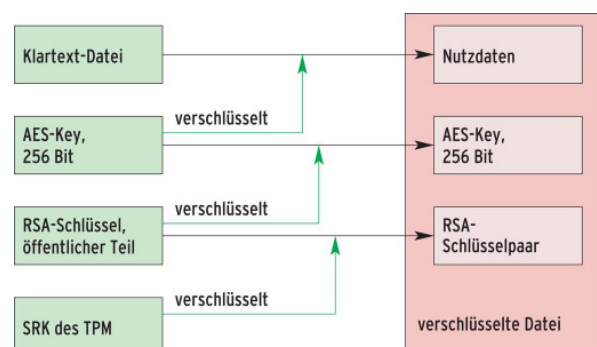
- *Sicheres Booten ist eine Voraussetzung für sicheren Betrieb*
 - BIOS vermisst relevante Teile der Plattform und trägt Werte in TPM ein
 - 16 (v1.1b) / 24 (v1.2) Slots -> Extending möglich
 - Kontrolle wird nur an Boot-Loader übergeben, wenn Messwerte ok
- *Mehrere Szenarien*
 - Trusted Boot -> Reines "Vermessen", keine Action-Line
 - "Enforcing" mit Action-Line
 - Secure Boot -> "Vermessen", Action-Line (z.B. Abbruch des Boot-Vorgangs)
 - Authenticated Boot -> Action-Lines zusätzlich abhängig von Messwerten
- *Trusted GRUB*
 - Quelle: <http://trousers.sourceforge.net/grub.html>
 - Realisiert "Trusted Boot"
- *Offene Fragen*
 - Management der Referenzwerte
 - Initiale Vertrauensbasis?

Beispiel 2: RootCA mit TPM schützen

- *RootCA abhängig von Integrität der Keys / Certificates*
- *Lösung: Schutz einer RootCA durch TPM*
-> RootCA an Plattform gebunden
- *OpenSSL Engine*
 - Quelle: <http://sf.net/projects/trousers/>
 - Benötigt OpenSSL-0.9.8
- *Vorgehensweise*
 - RSA-Objekt im TPM erzeugen
 - Verschlüsseln mit SRK, Export und persistentes Speichern
-> RootCA-Key!
 - Erzeuge Zertifikat mit den Optionen
"-keyform engine -engine tpm"
- Vorteil: Keine Offline-Angriffe möglich
- Nachteil: TPM sehr langsam (daher nicht für Web-Server geeignet)

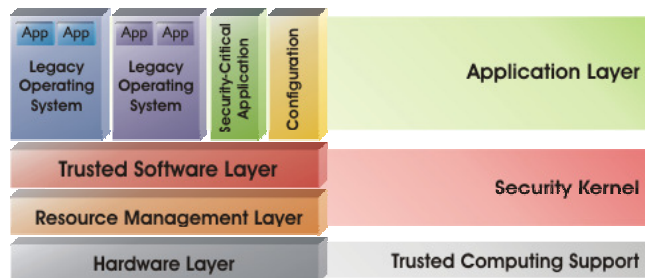
Beispiel 3: Daten mit TPM schützen

- *Data Binding <-> Data Sealing*
 - Binding -> Daten durch Verschlüsselung mit SRK an TPM "gebunden"
 - Sealing -> Hinzunahme von PCR-Werten versiegelt Daten (spezielle Konfiguration)
- *Nutzung des TPM zur (indirekten) Verschlüsselung von Daten*
 - Nutze TPM, um symmetrischen AES Schlüssel (256 Bit) zu erzeugen
 - Verschlüssele Datei unter Nutzung von OpenSSL mit AES
 - Nutze TPM, um RSA Objekt zu erzeugen
 - Schütze AES Schlüssel mit RSA Objekt
 - Schütze RSA Objekt mit SRK
 - Ergebnis:
 - Daten verschlüsselt
 - Daten an Plattform gebunden
- *Probleme und offene Fragen*
 - tpmseal.c
 - Probleme mit bereits gesetztem SRK
 - Vorgehensweise bei "Verlust" des TPMs?
 - Backup von Daten?



EMSCB – Turaya-Projekt

- Projekt der European Multilaterally Secure Computing Base (EMSCB)
- Konzept
 - "Security" L4-Microkernel ("geringer" Codeumfang) als Basis
 - On-Top: Virtualisierung, um sichere Compartments zu schaffen

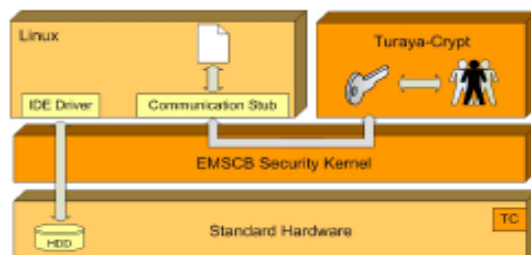


Quelle: www.emscb.org

- *Demo-CD*
 - Quelle: <http://www.emscb.org>
 - Bisher lediglich Nutzung von Trusted GRUB (Secure Boot)
 - Ist (noch) nicht an "Plattform" gebunden

EMSCB – Turaya-Crypt

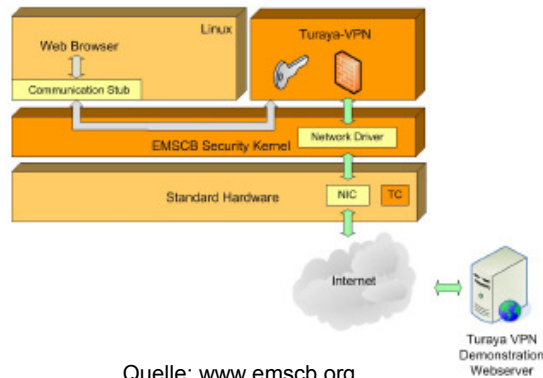
- Anwendung: Secure Device Encryption unter Linux
- Aufsplittung in getrennte Bereiche
 - "Standard"-Linux: Verschlüsselung von Daten/Partition/Festplatte
 - Turaya-Crypt: "Trusted GUI" für "sichere" Passwortabfrage
- Anwendungen bekommen keinerlei Zugang zum "Passwort"
- Schematische Darstellung:



Quelle: www.emscb.org

EMSCB – Turaya-VPN

- Anwendung: VPN unter Linux
 - Kompatibel zum IPSec-Standard, transparent für den Anwender
- Aufsplittung in getrennte Bereiche
 - "Standard"-Linux: Web-Browser
 - Turaya-VPN: Regelt Key-Management, Zertifikatsverwaltung
- Anwendungen bekommen keinerlei Zugang zu sicherheitssensitiven Informationen wie Passwort, Keys, Zertifikate
- Schematische Darstellung:



Trusted Computing: Chancen ...

- Sicherer Hardwarespeicher
 - Verhindert unter anderem off-line-Angriffe auf "Geheimnisse"
- Sicheres Booten
 - TPM als sicherer Hardwareanker (Root of Trust)
 - Chain of Trust beim Boot-Prozess
- Digital Rights Management (DRM)
 - Mit Hardwareanker sicherer und zuverlässiger als ausschließlich durch Software bzw. das Betriebssystem
- Remote Attestation
 - GRID-Computing, Peer-to-Peer Computing
 - Sichere Verbindungen (vgl. auch TNG)

... und Risiken

- Remote Platform Attestation
 - Wirklich (komplette) Hard- und Softwareumgebung preisgeben?
- Sealing - Zensur - DRM
 - TCG ist "policy neutral"
- Open Source Software / Patente
- Gefahr von (nicht entdeckbaren) Hintertüren
 - Ron Rivest: "...renting out a part of your PC to people you may not trust."
- Migrations / Backup
 - der Schlüssel
 - der verschlüsselten Daten ?
- "ungeeignete" Kryptographie?
 - 2006er Empfehlungen des BSI / der RegTP:
Mindestwert 1.536 Bit, empfohlen 2.048 Bit (jeweils bis 2009)

Forderung von Kritikern

- Endorsement Key austauschbar
 - Bereits in TPM Spezifikationen v1.2 enthalten
 - Für kleine Organisationen nicht sinnvoll einsetzbar
- Direct Anonymous Attestation
 - Bereits in TPM Spezifikationen v1.2 enthalten
 - Beliebig viele anonyme Zertifikate
 - Unlinkbarkeit
- *Internationale* und *unabhängige* Kontrolle des TPMs (CPU-Integration?)
- CCC: Vollständige Kontrolle über alle TPM-Schlüssel
- EFF: Owner Override

Fazit – Ausblick

- Trusted Computing wird kommen, daher
 - *offene* Ansätze vorantreiben,
 - (mit)diskutieren und
 - informieren!
- Es lohnt sich definitiv die Entwicklung im Auge zu behalten!
- Der frühe Vogel fängt den Wurm ;)

Literatur

- Wilhelm Dolle, Michael Nerb und Christoph Wegener:
"Anwendungen für das Trusted Plattform Module"
<http://www.linux-magazin.de/Artikel/ausgabe/2006/12/tpmeinsatz/tpmeinsatz.html>
- Wilhelm Dolle und Christoph Wegener:
"Höllenglut – Trusted Computing für Linux: Stand der Dinge"
<http://www.linux-magazin.de/Artikel/ausgabe/2006/04/tcpa/tcpa.html>
- Spezifikationen und viele weitere Infos unter:
<http://www.trustedcomputinggroup.org>

Danke :)

- Wilhelm Dolle, Michael Nerb
- Ihnen für Ihre Aufmerksamkeit :)
- Fragen?!?
- Kontakt:
E-Mail: wegener@wecon.net
Web: www.wecon.net