

Bausteine für Secure Embedded Computing



KEM Deggendorf
Product Marketing
Deggendorf, August 2011

Sicherheit im Alltag



Security in your
Everyday-Life

Security in your everyday-life „Mobile Phones“



- » GSM is very secure
- » Based on chip security and modern encryption

- » Chip contains very “secure” access key
- » PIN activates chip which knows key



Security in your everyday-life „Passport“



- » Personal data is stored on chip
- » Chip and “checksum” ensures integrity (prevent from fraud)

- » Chip on chip card is “TPM”

Warum braucht man Sicherheit
bei Embedded Computing?



Why security needs
are arising?

Isolation

- » No network connection
- » System is running for years (w/o updates)
- » Limited group of people gain (physical) access

Is that approach still valid?

Stuxnet

- » Discovered in June 2010
- » Active even longer
- » Targeting Windows/Siemens Systems
- » Made use of 4 zero-day exploits
- » Distribution via **USB stick**

Stars

- » Discovered in April 2011

- » Time-to-market is more critical
- » More and more standard technology is used
- » OSes get more standardized
- » *Consumer exploits* tend to work out on Embedded Systems *out of the box*
- » Devices, also on industrial control level, get more and more connected

Was ist beim Embedded Computing anders?



Adopting behaviors from
consumer market?

Consumer Electronic

- » Hardware and Software replace cycles are short
- » Mass products without customization
- » Antivirus software, firewalls for all common OSes
- » Regular, automatic updates of software

Embedded Industries

- » Highly customized solutions
- » Very small quantities
- » Old OSes, no updates available, no malware protection

Wie kann man das
Problem lösen?



Building up a
secure solution...

Requirements

- » Maximum re-use of existing code
(i.e. automation control routines)
- » Prevent code manipulation
- » Prevent unauthorized access from outside
- » Easy to handle, cheap investment

Building Block Concept

- » Use hardware TPM (TXT, HSM)
- » Use **standardized** methods for secure boot-chain
- » Use **virtualization** for maximized re-use

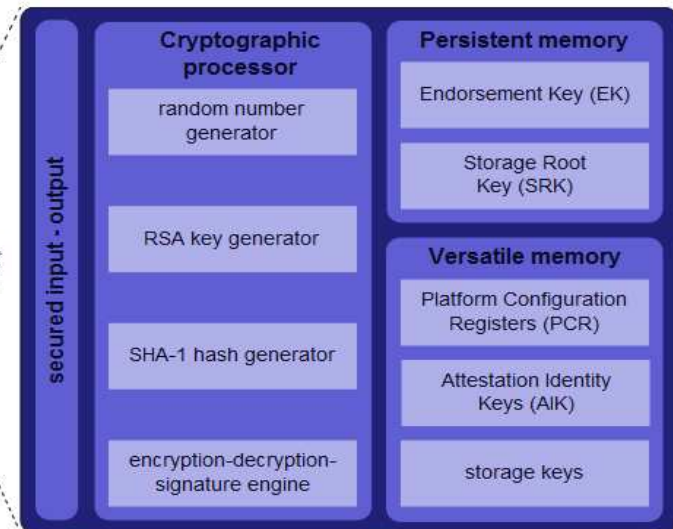
Root of Trust
(TPM)

Secure
Boot-Chain

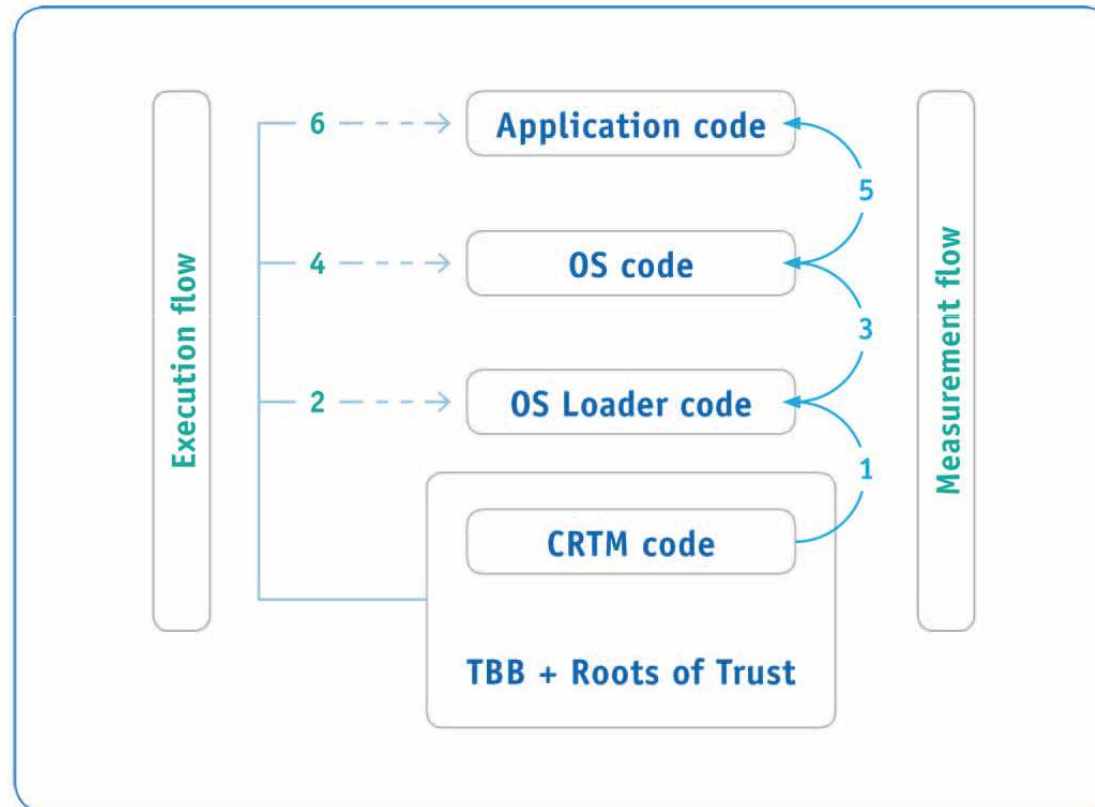
Virtualization

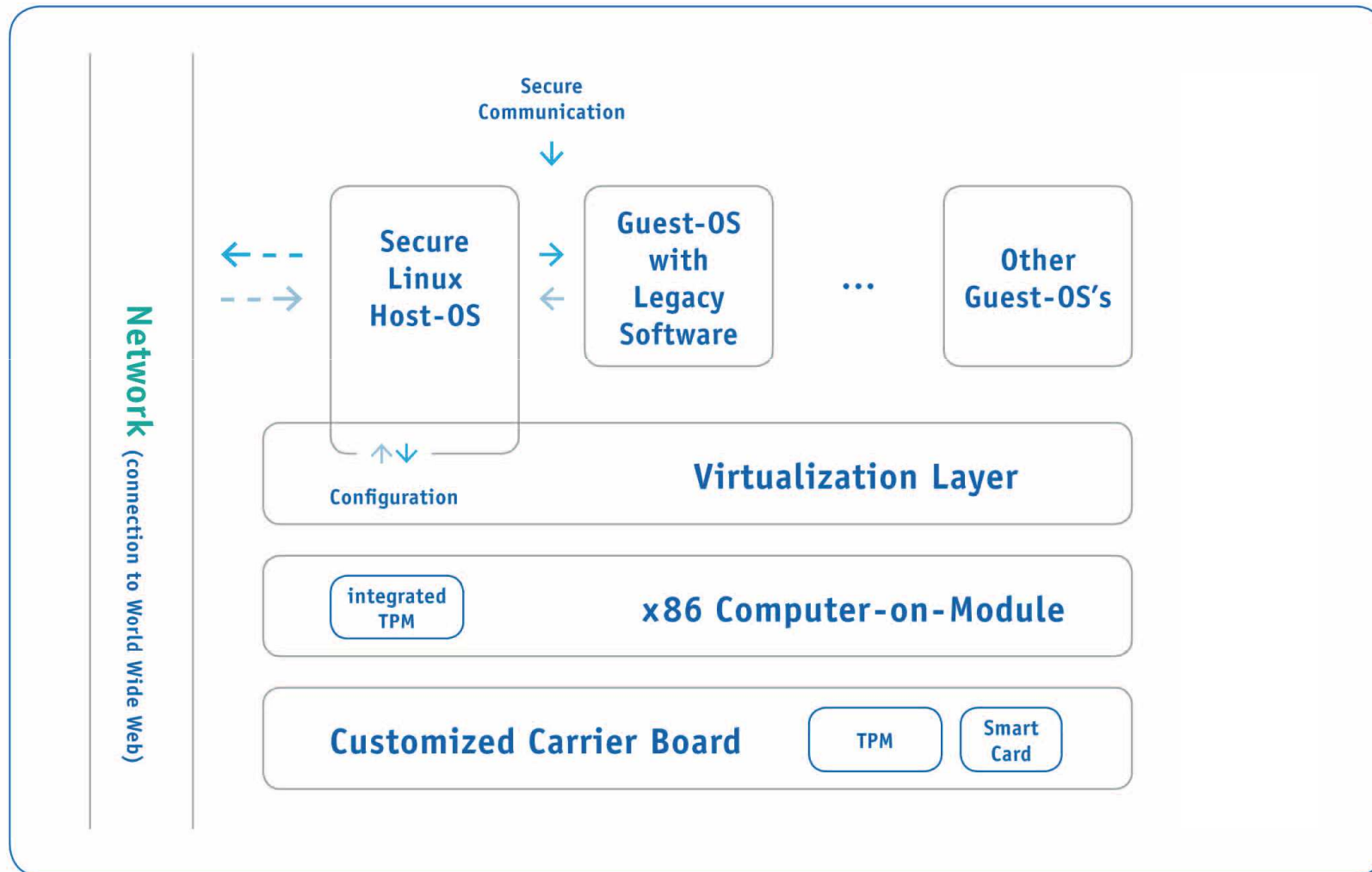
Legacy Code

- » TPM is somehow a System-on-Chip
- » Special logic to do encryption, key generation,...
- » Special logical and mechanical protection from being access unauthorized



TPM is the HSM, “The root of trust”





It is a Research Project with FH Deggendorf and Intel

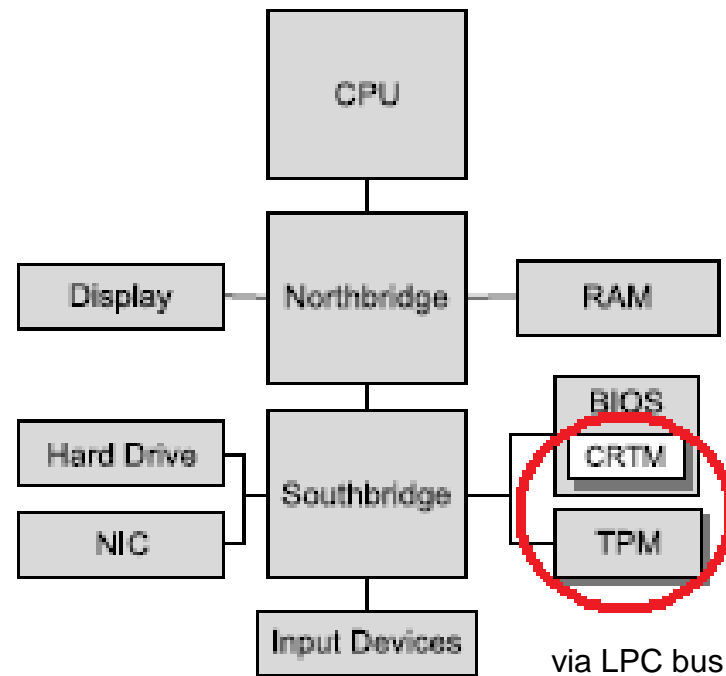
- » Kontron Modules with TPM (TPM is most critical)
- » Standard Ubuntu Linux 10.04 LTS
- » Adapted boot routines *tboot*
- » Encrypted files

FH works on Proof-of-Concept for securer system

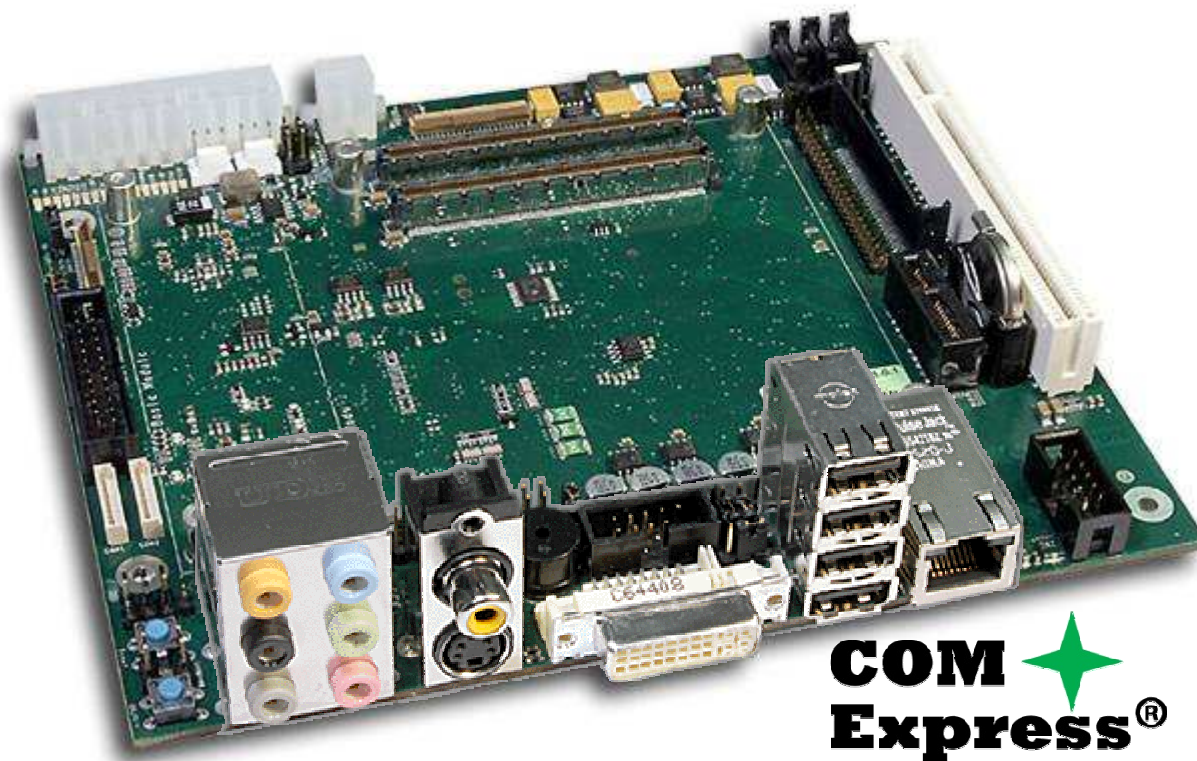
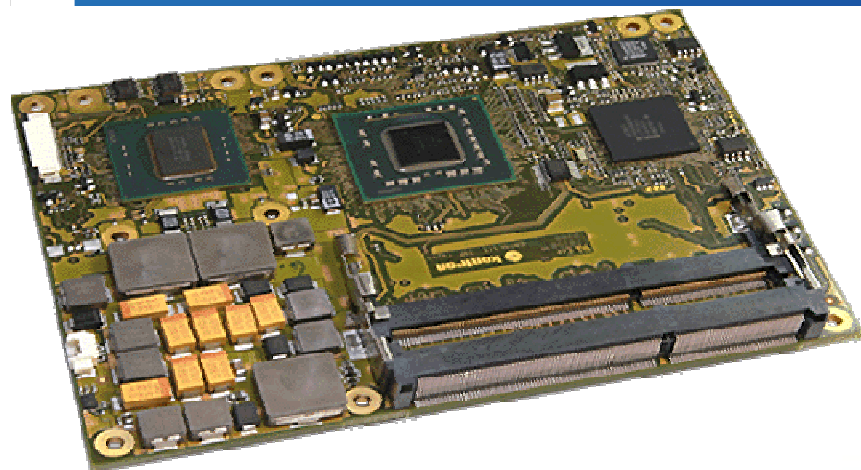
TPM on Kontron Modules

Modules with optional Infineon TPM:

- » nanoETXexpress-SP
- » nanoETXexpress-TT
- » microETXexpress®-SP
- » microETXexpress®-DC
- » microETXexpress®-OH
- » microETXexpress®-PV
- » ETXexpress®-PC
- » ETXexpress®-AI
- » ETXexpress®-SC
- » ETX®-CD
- » ETX®-DC



Example:



COM  **Express**[®]

Live Demonstration...

Whitepaper

» Get Whitepaper

» Visit Ineltro and Kontron Booth!

ineltro





DANKE!

Hubert Hafner, eMail: hubert.hafner@kontron.com

Kontron Embedded Modules GmbH, Deggendorf