

**MobiSec 2009**

# **Open Trusted Computing**

## **an overview**

**Gianluca Ramunno**

ramunno @ polito.it

Politecnico di Torino

Computer and Network Security Group (TORSEC)

Turin (Italy), June 4<sup>th</sup>, 2009

This work was prepared by:

- Richard Brown (HPlabs, Bristol, UK)
- Dirk Kuhlmann (HPlabs, Bristol, UK)
- Konrad Eriksson (IBM Research, Zurich, CH)
- Stephane Lo Presti (formerly Royal Holloway, University of London, UK)
- Gianluca Ramunno (Politecnico di Torino, IT)
- Suen Chun Hui (Lehrstuhl für Datenverarbeitung, Technische Universität München, DE)

during 2007-2009, as part of part of the Open Trusted Computing project,  
<http://www.opentc.net>

and released under the following license:

- Creative Commons Attribution-Share Alike 3.0 Unported License.

This work is licensed under the

**Creative Commons Attribution-Share Alike 3.0 Unported License.**

You are free:

- **to Share** - to copy, distribute and transmit the work
- **to Remix** - to adapt the work

Under the following conditions:

- **Attribution.** You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).
- **Share Alike.** If you alter, transform, or build upon this work, you may distribute the resulting work only under the same, similar or a compatible license.



## License – 2

To view a copy of this license, visit:

**<http://creativecommons.org/licenses/by-sa/3.0/>**

or send a letter to Creative Commons, 171 Second Street, Suite 300,  
San Francisco, California, 94105, USA.

To view a copy of the Legal Code (the full license), visit:

**<http://creativecommons.org/licenses/by-sa/3.0/legalcode>**

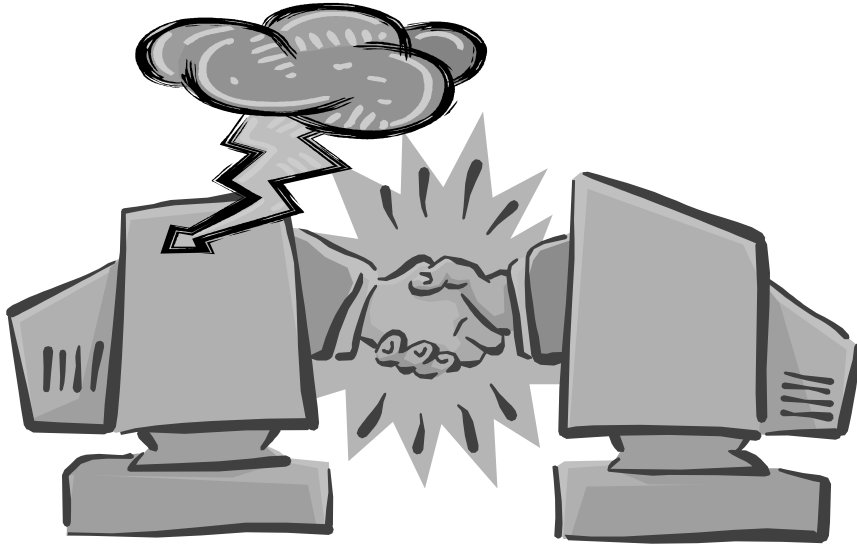


## In the beginning (2004) ...

- Trusted Computing was controversial, so much so ...
  - *"Will lock down computers of end users"*
  - *"Will lock customers into specific OSs and software"*
  - *"TC's purpose is to enforce DRM on consumer machines"*
  - *"TC considered harmful for Open Source Software"*
- ... that academia typically did not dare to touch it.
  - No R&D ecosystem, in particular in Europe
  - No fact base to address Fear, Uncertainty & Doubt
  - No experimental implementations

But underlying problems are real and need investigation!

# Problem Statement: IT is an *Actor*. But whose script does it follow?



## Current Situation

- IT *can* change a YES into a NO
- IT can distort expressions of will and intentions and we have no indication mechanisms for this
- We can't qualify the technical risk of subverted end platforms from the user's authorization



## Future scenario

- Attest platform's "fitness for purpose"
- Use it as admission criterion for network or transactions
- Allows for multiple purposes, roles, and identities



## OpenTC Motivations: Technology that's useful

- Exploit existing capabilities of Virtualization and Trusted Computing technologies beyond proprietary solutions, in a collaborative/cross industry solution
- EC wanted to ensure that the Trusted Computing value proposition was also delivered in an OSS setting
- Counter critical response of OSS spectrum to TC
- Extend corporate options (IBM, HP, Novell/SuSE)
  - Support corporate strategies: TC in managed infrastructures
- Extend market (Infineon, AMD)
  - Wider deployment of Hardware (TPMs, CPUs)



# OpenTC: Working Hypotheses

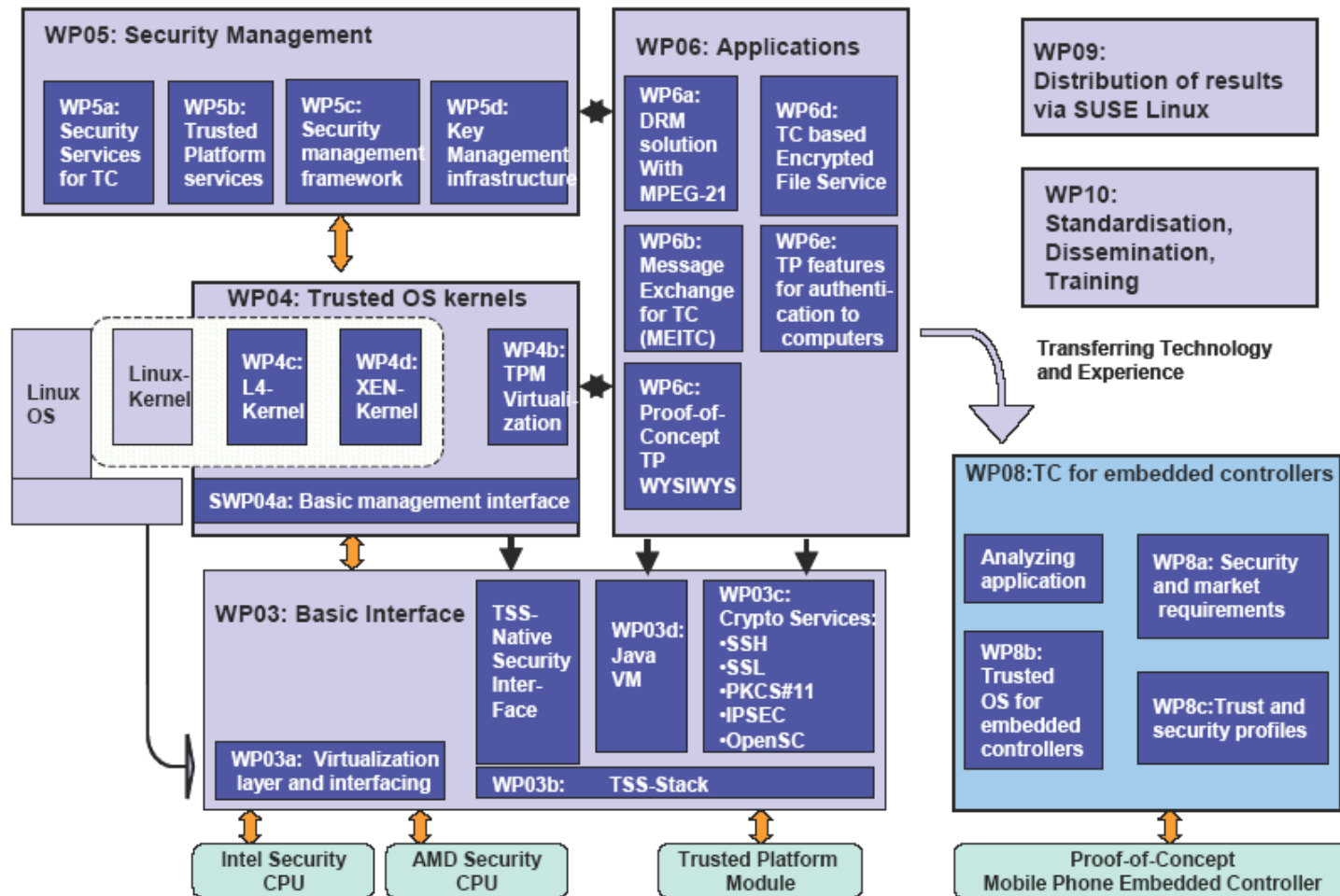
- Trust requires transparency (actual degree tbd.)
  - Inherent advantage of Open Source (can be inspected)?
  - Inherent problem of proprietary hardware and software?
- Flexible Trusted Computing is possible
  - Avoid platform lockdown and consumer lock-in
  - Core idea: Trusted Virtual Platforms
- Trusted Computing and Open Source are compatible
  - Technically, legally, and philosophically



# OpenTC: Project Goals

- Produce a security enhanced OS base
  - Integrating TC and Virtualization technologies
  - Based on Open Source components
- Management framework and protocol software
  - For trusted virtualized systems
- Application prototypes / use examples
  
- Some of the challenges in building such a system:
  - Robust and secure virtualization layer isolation
  - Trusted sharing of hardware I/O devices
  - Manageability and attestability

Open Trusted Computing: Functional Diagram



Technikon Forschungs- und Planungsges.m.b.H, A		Commissariat à l'énergie atomique, FR	
Infineon Technologies AG, D		Ruhr Universität Bochum, D	
Hewlett Packard, UK		Technische Universität Dresden, D	
Technische Universität Graz, A		University of Cambridge Computer Laboratory, UK	
Technische Universität München, D		IBM Research GmbH, CH	
SUSE Linux Products GmbH, D		Institute for Security and Open Methodologies, ESP	
Royal Holloway, University of London, UK		AMD, D	
Forschungszentrum Karlsruhe, D		Portakal Teknoloji, Turkey	
Tubitak, Turkey		Intek, RU	
Politecnico di Torino, I		Technical University of Sofia, BG	
Budapest University of Technology and Economics, HU		Katholieke Universiteit Leuven, B	
		COMNEON, D	



# OpenTC Main Phases

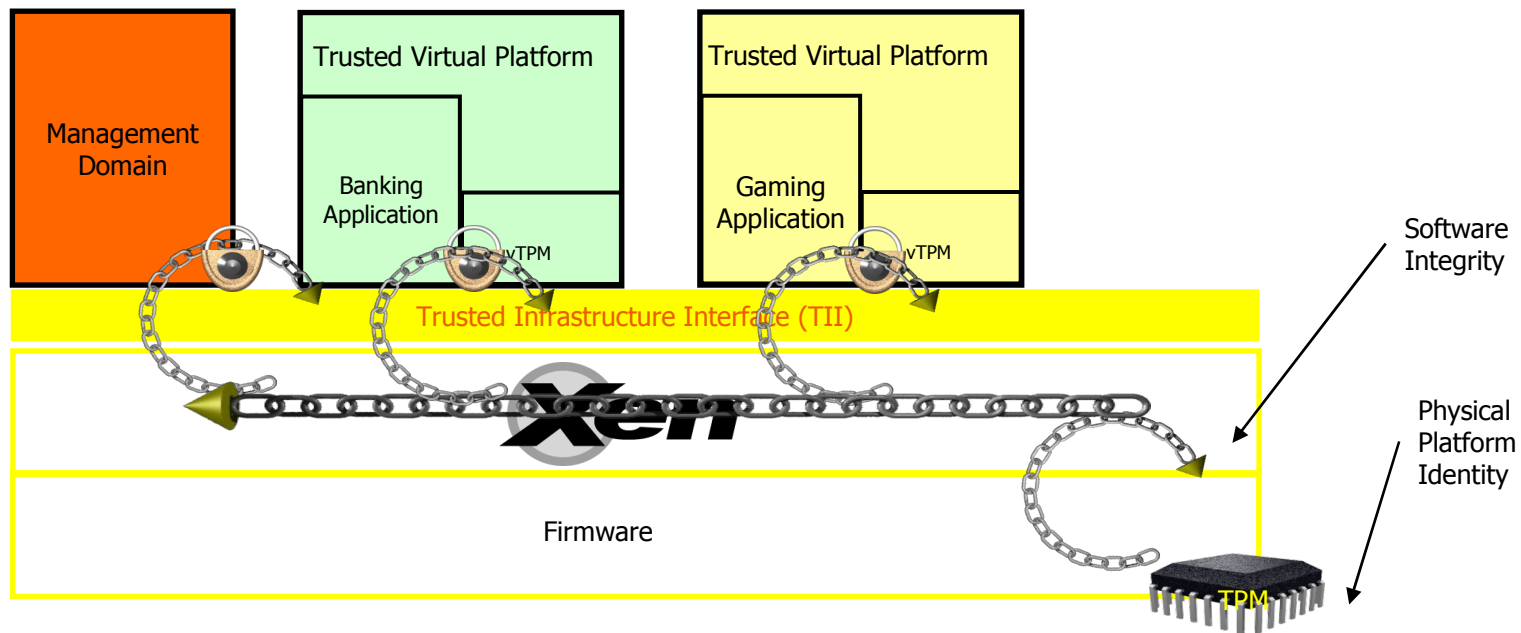
- Phase 1
  - Working together
  - Integrating existing technologies and understanding their potential
  - Client measurements for backend services
  - UC: Privacy Enhanced Electronic Transactions (secure browsing)
- Phase 2
  - Architecture (e.g. disaggregation of VMM, graphics), measurement framework, richer interaction models, validation and verification
  - UC: CC@Home (separating environments for work and private use)
- Phase 3
  - Server side virtualization, using virtual networking and Trusted Virtual Domains concept for customer separation
  - UC: Virtual Data Center



# Chain of Trust

## Manageable and attestable virtualization layer

- Integrity Measurement Architecture
- Enabling remote attestation of invariant security properties implemented in the trusted virtualization layer





# Trusted Computing and virtualization

- Trusted virtualization
  - Use of virtualization for security on PC class platforms
    - Ideally a possible replacement for the traditional security kernels
  - Combined use of virtualization and Trusted Computing
- Used to build general purpose security frameworks
  - Suitable for many broad application scenarios
  - Ideally comprising all application purposes
- Taken into account at least in some R&D projects
  - OpenTC (Open Trusted Computing)
  - EMSCB (European Multilateral Security Computing Base)

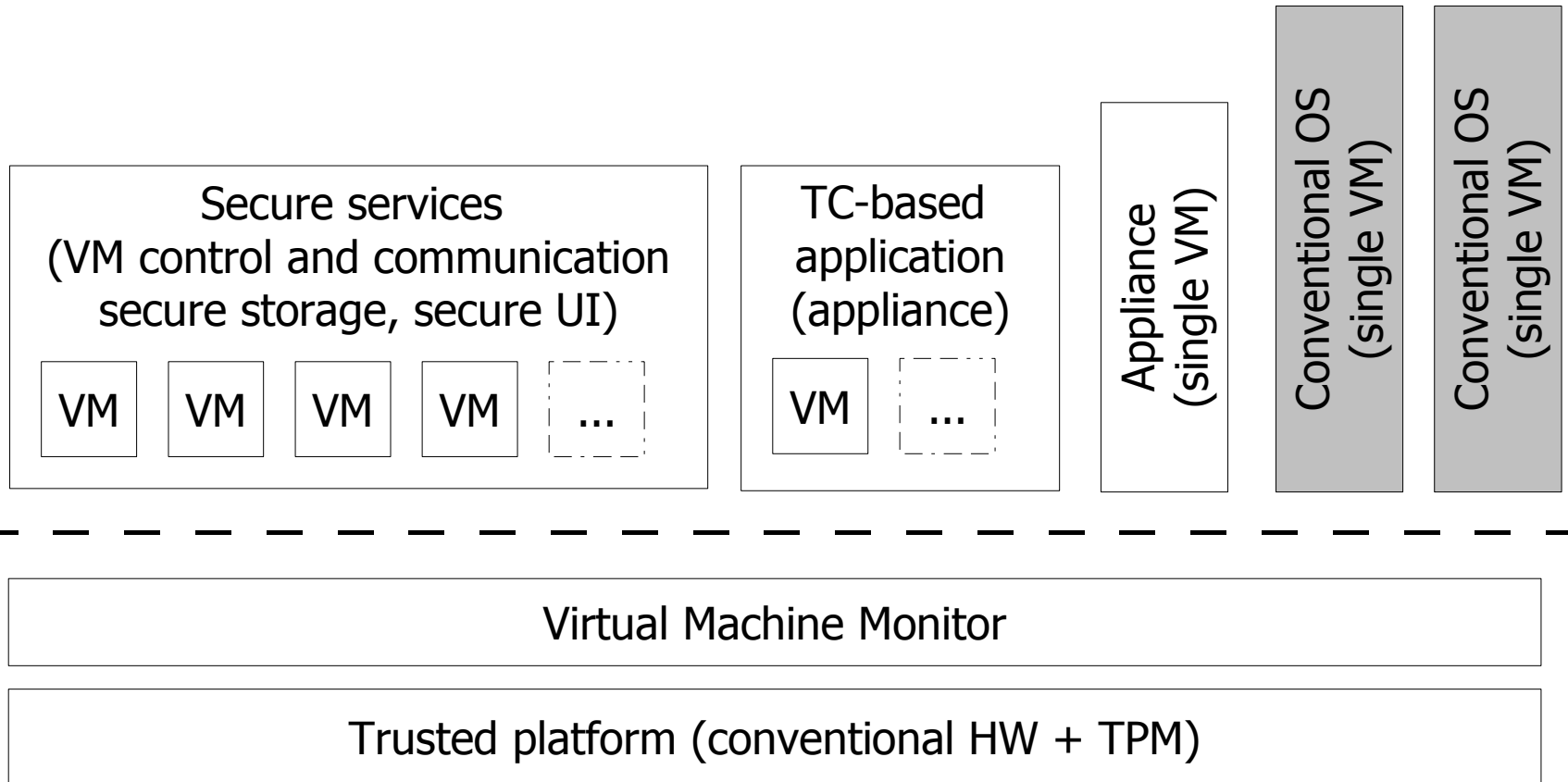


## Trusted Computing and virtualization (2)

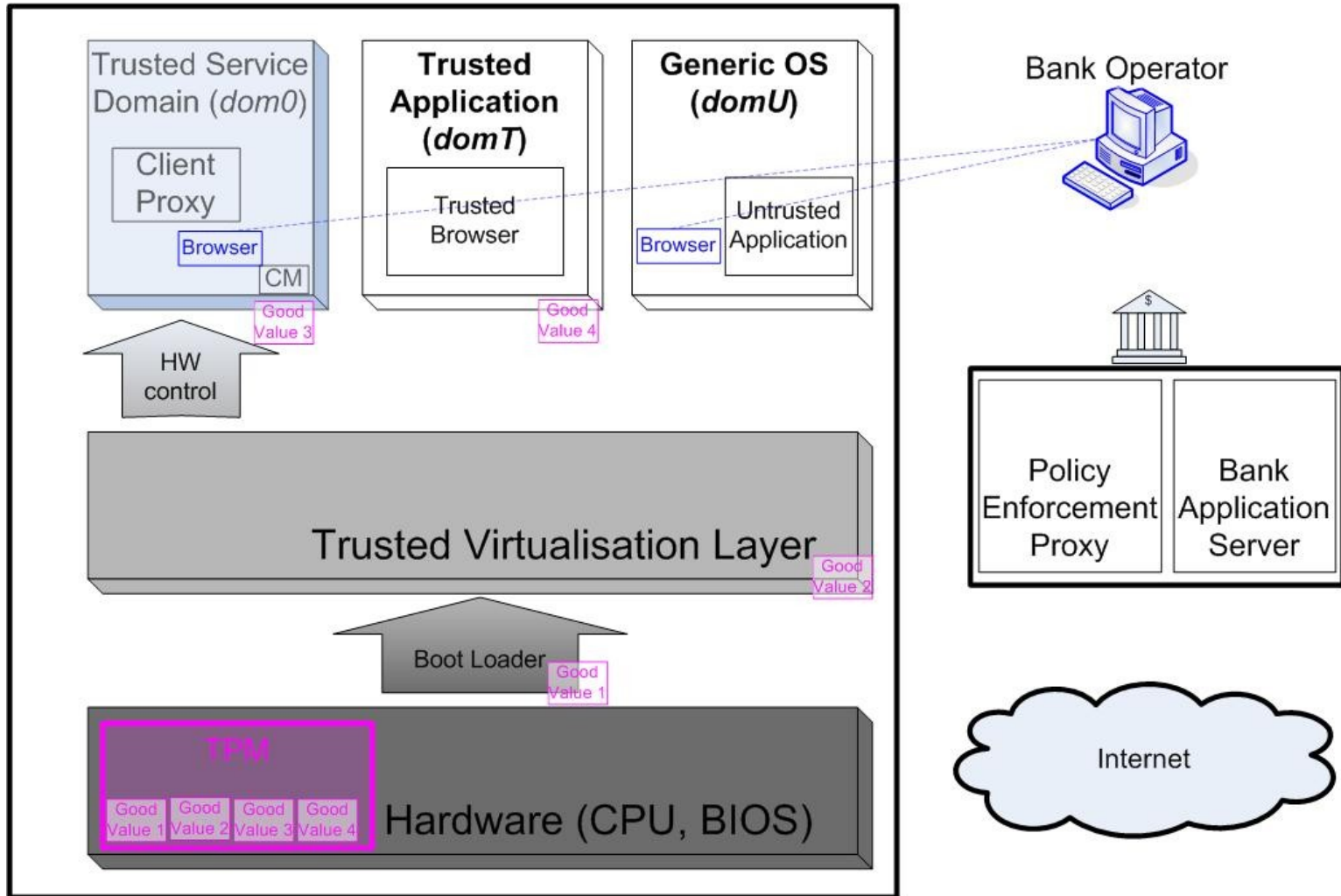
- Key points
  - Much more than TCG-aware OSES and applications
  - TPM used as hardware root of trust
    - many features/services implemented in software
  - Virtualization used to achieve memory isolation
  - Security services running in small virtual machines (compartments)
    - Secure GUI, Secure Storage, Virtual TPMs, ...
    - They can be measured (trusted)
      - Together the Virtual Machine Monitor (VMM) they form the Trusted Computing Base (TCB)
  - Also user security critical applications can be isolated in different virtual machines (compartments) and measured
  - Communications between compartments is subject to flow control policies enforced by VMM

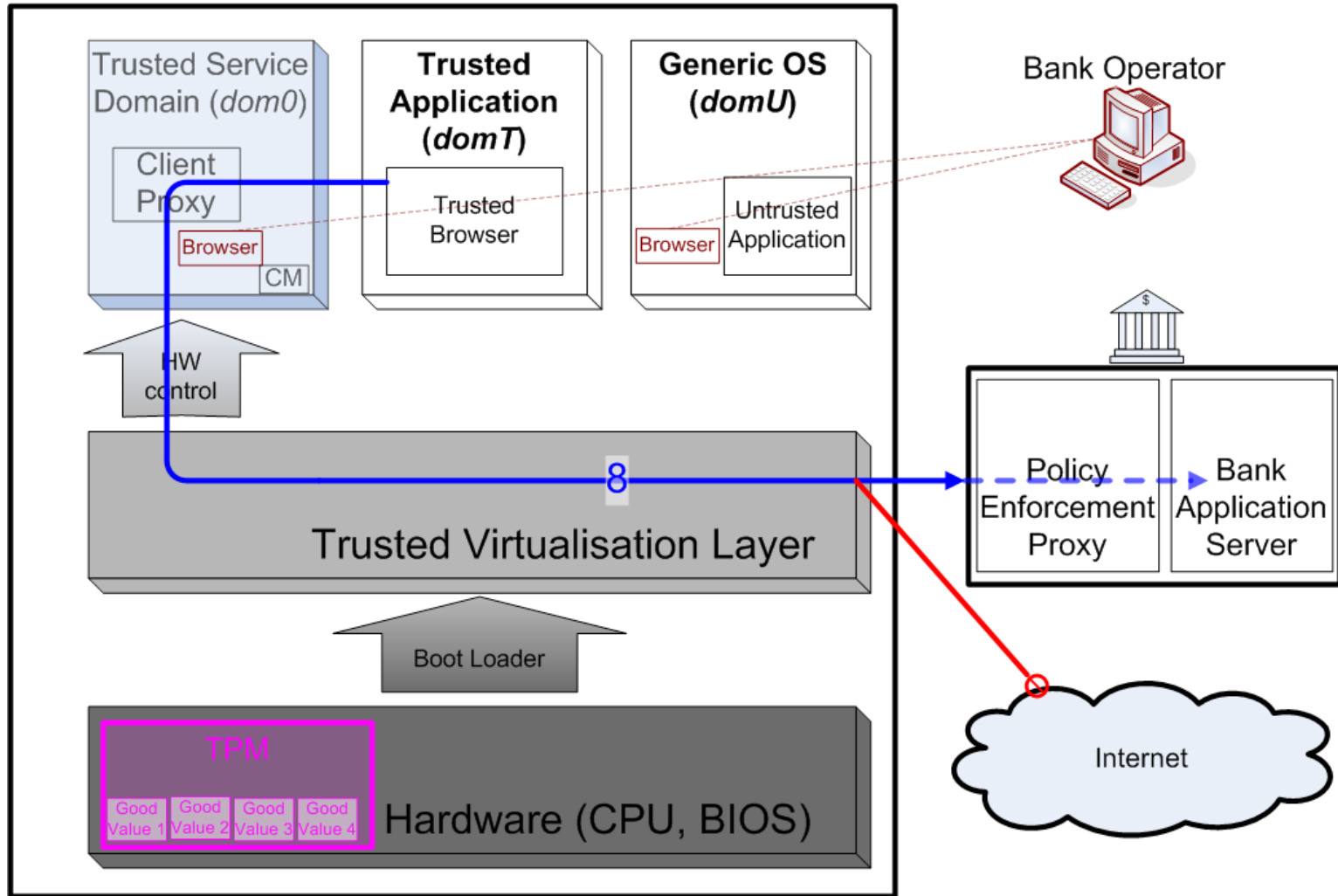


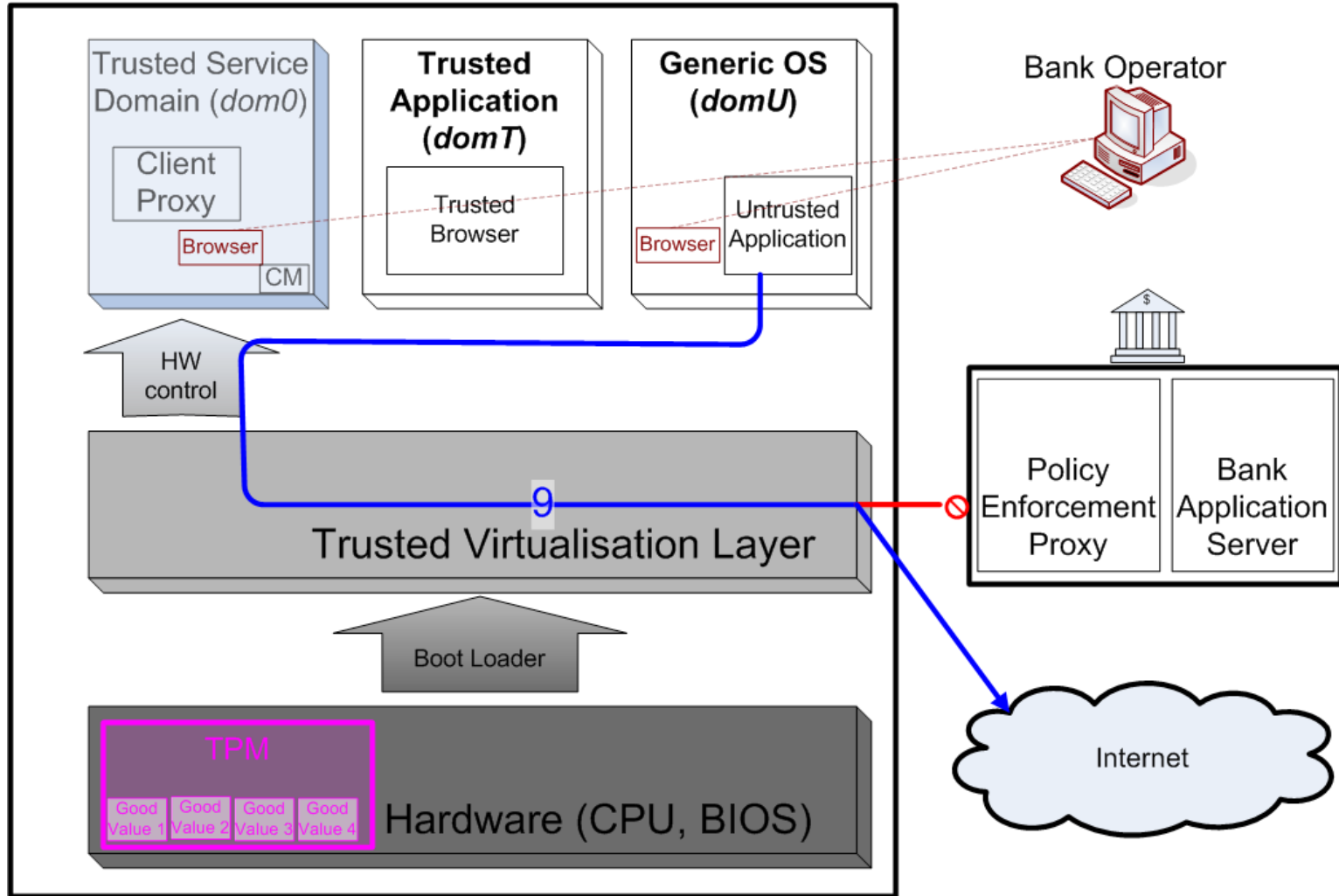
# Comprehensive security framework

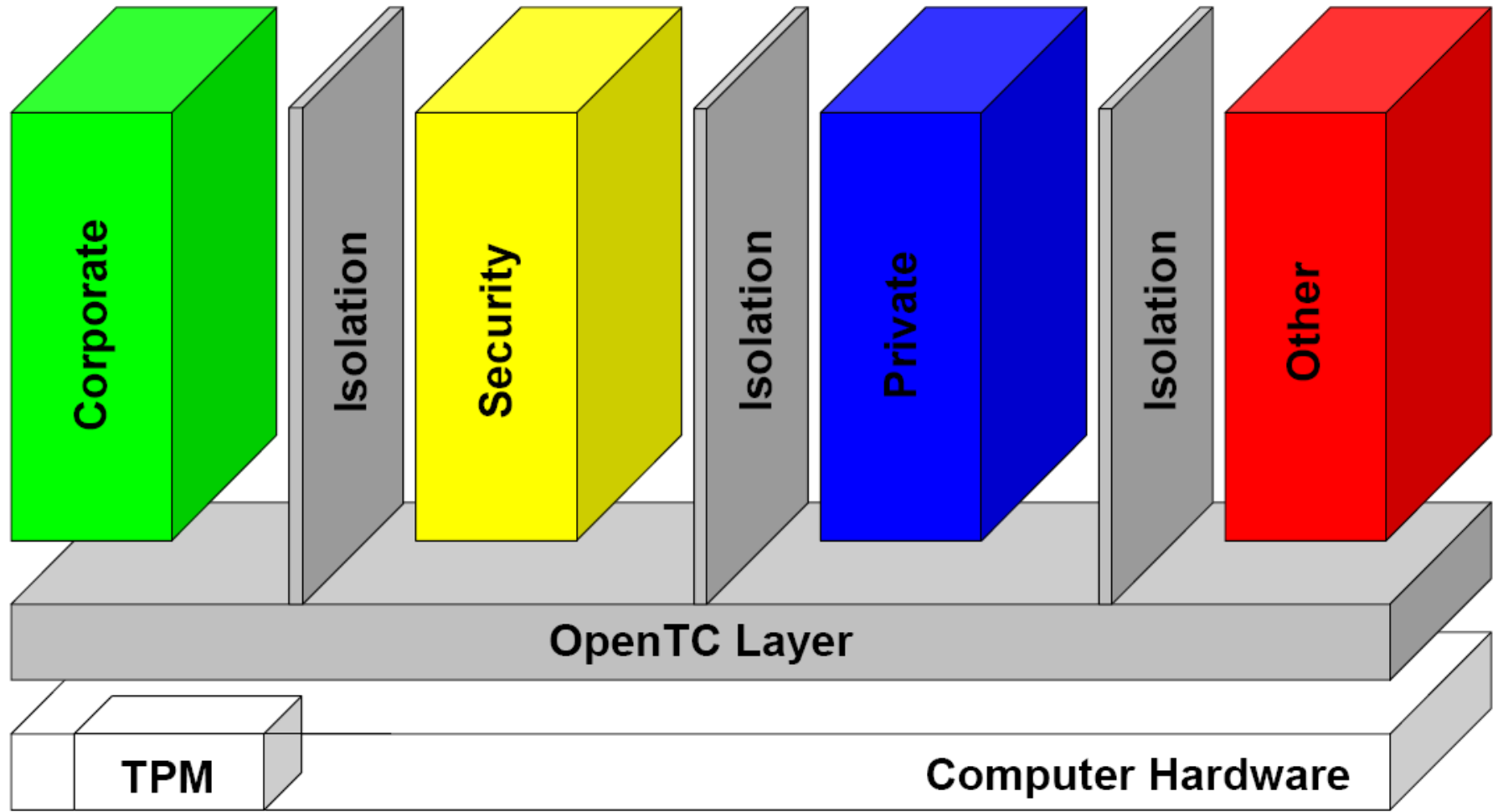


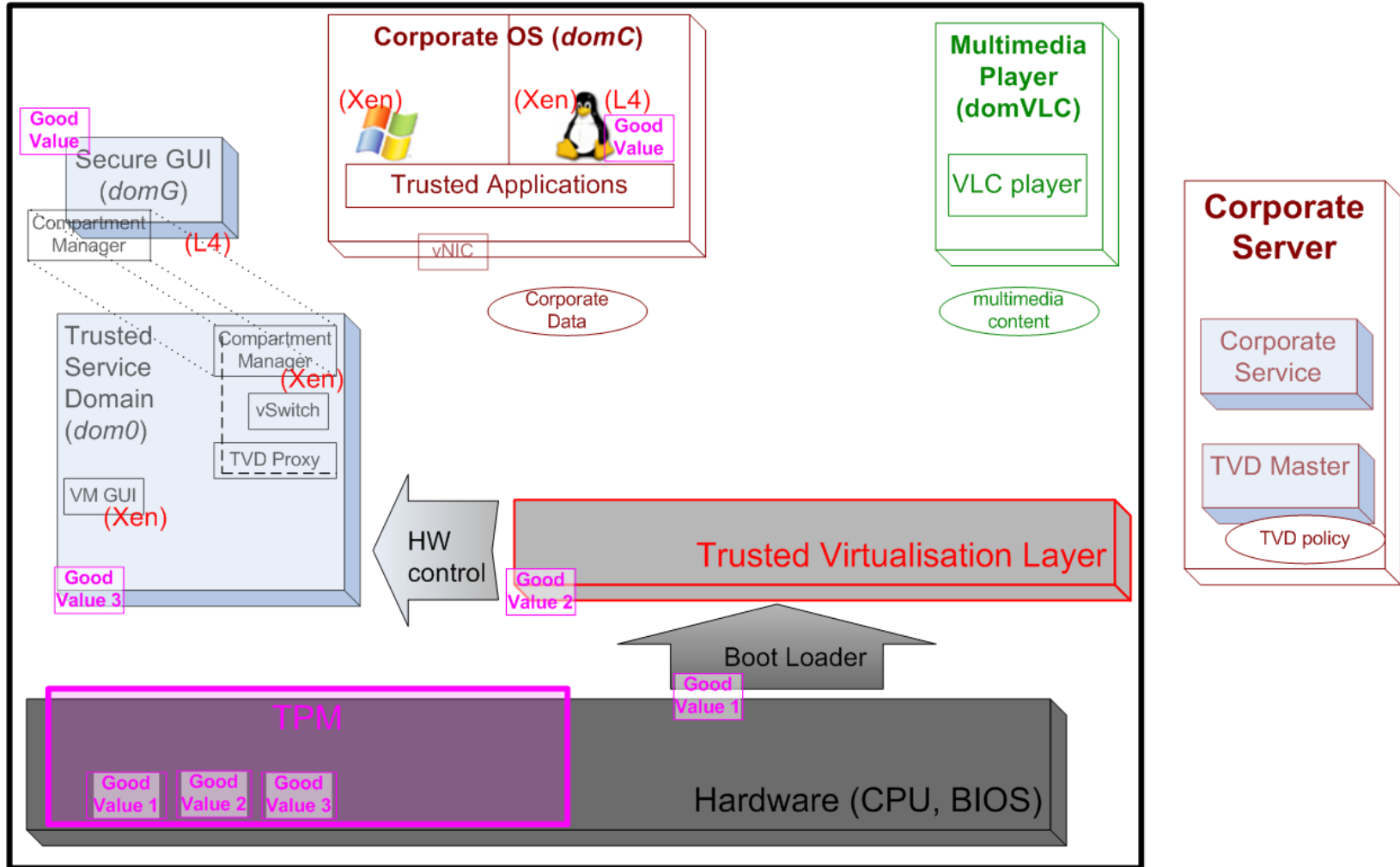
# 1<sup>st</sup> - Private Electronic Transactions (PET)

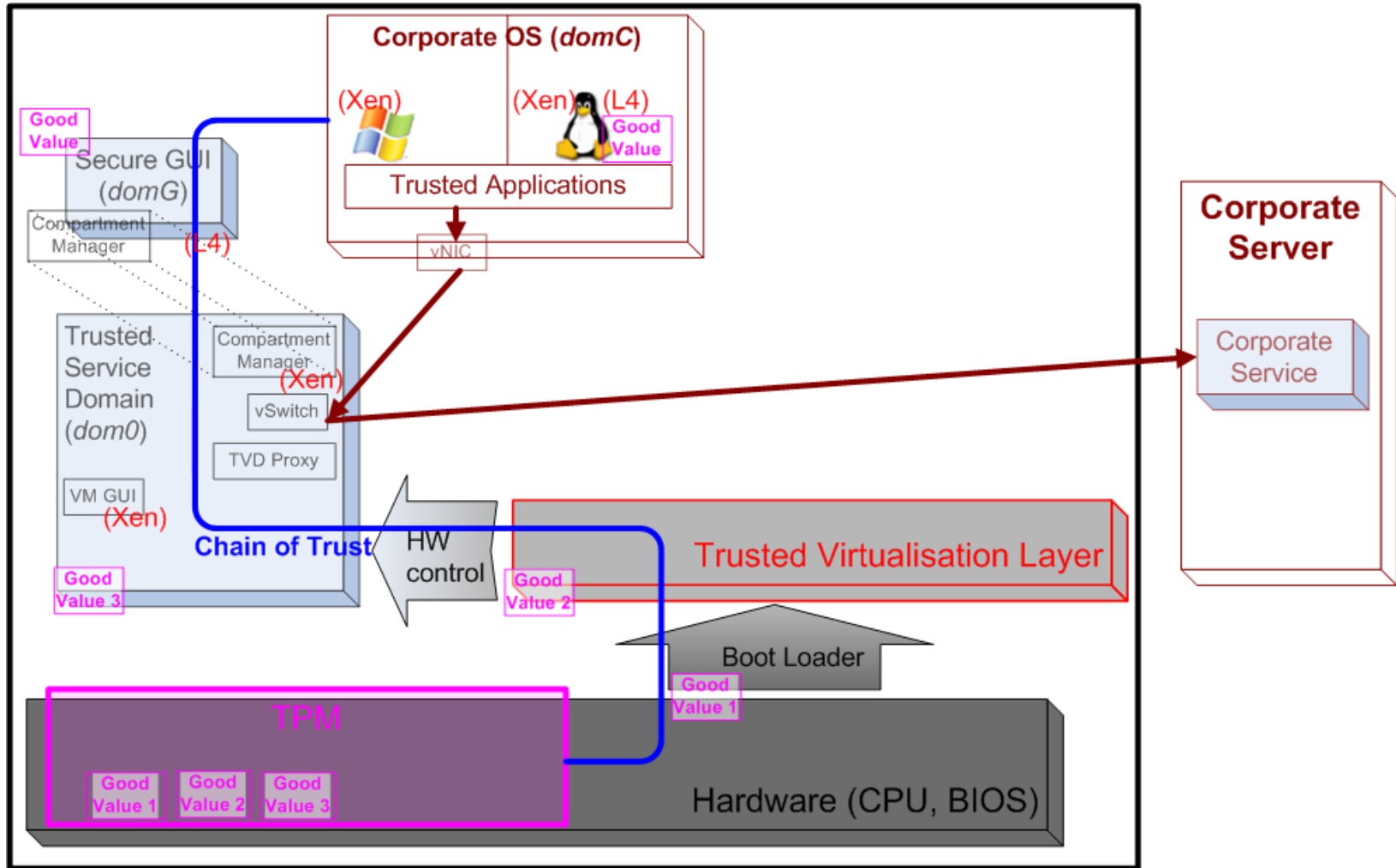












- Control icon
  - only known to the legitimate user
    - not provided with the distribution
  - sealed (i.e. encrypted) against a “good” system configuration
    - if unsealing fails at boot time, this means that the system configuration changed



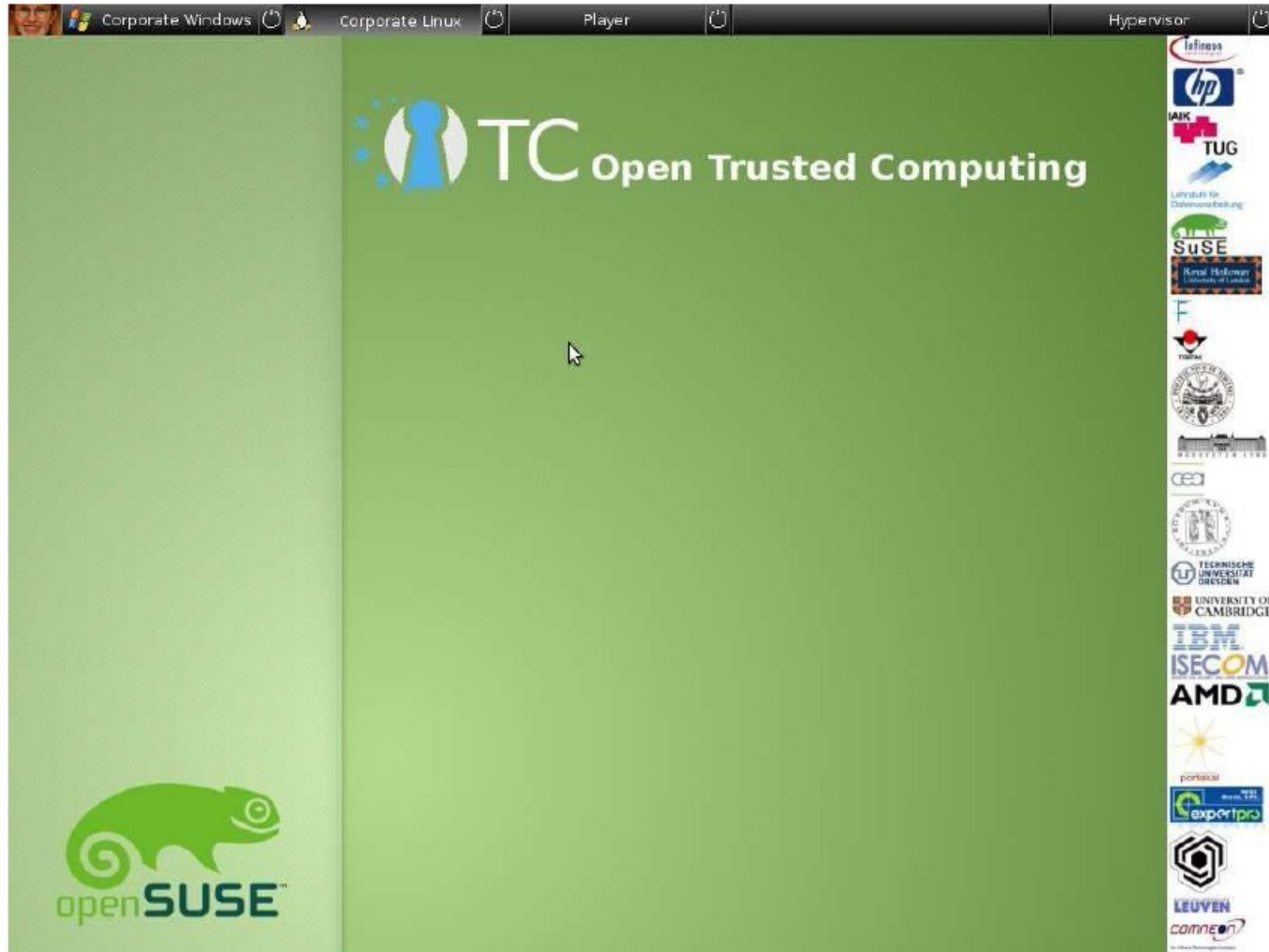


# CC@H: Corporate Compartment with standard VPN software

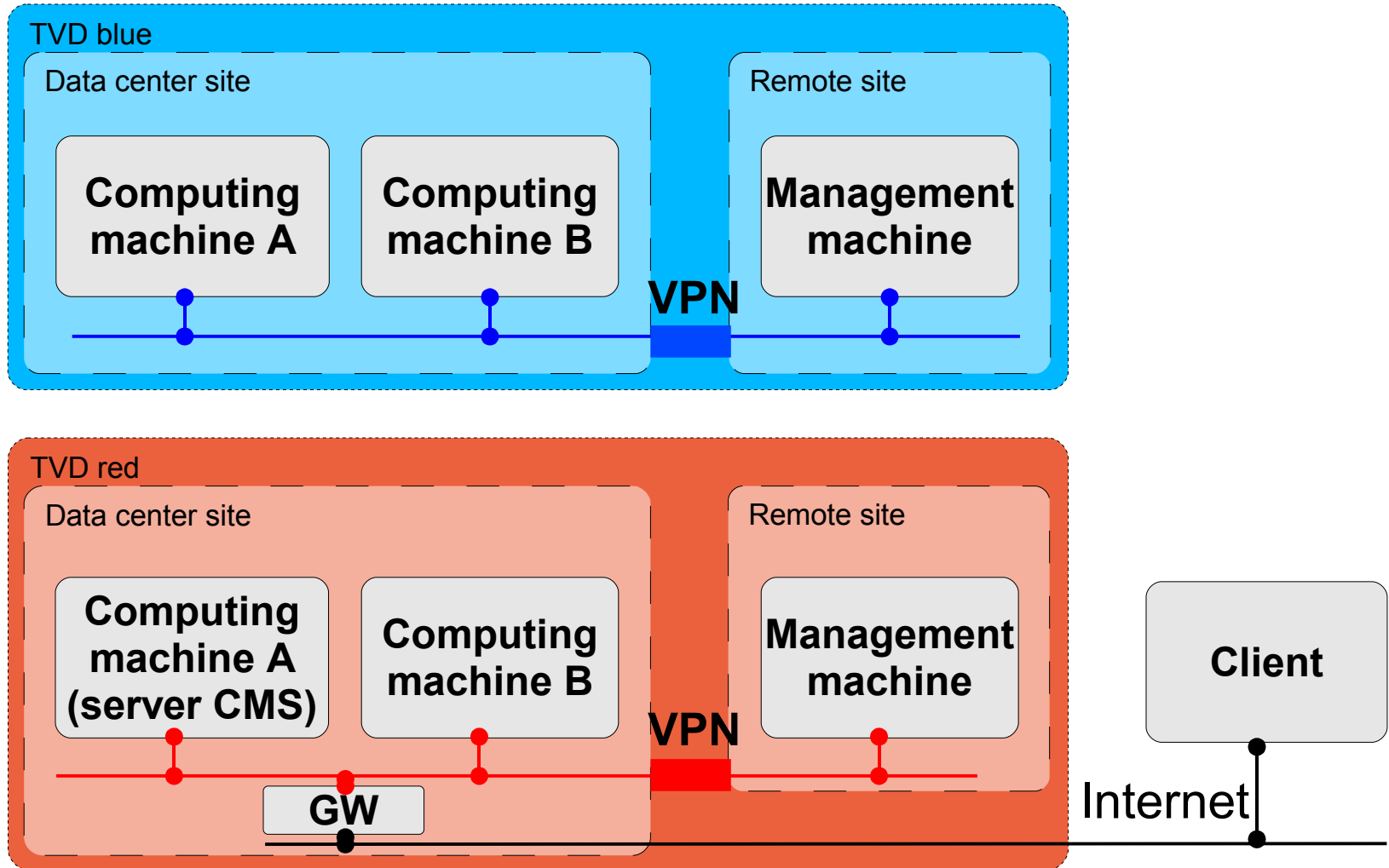




# CC@H: Corporate Compartment with OpenTC VPN bound to integrity



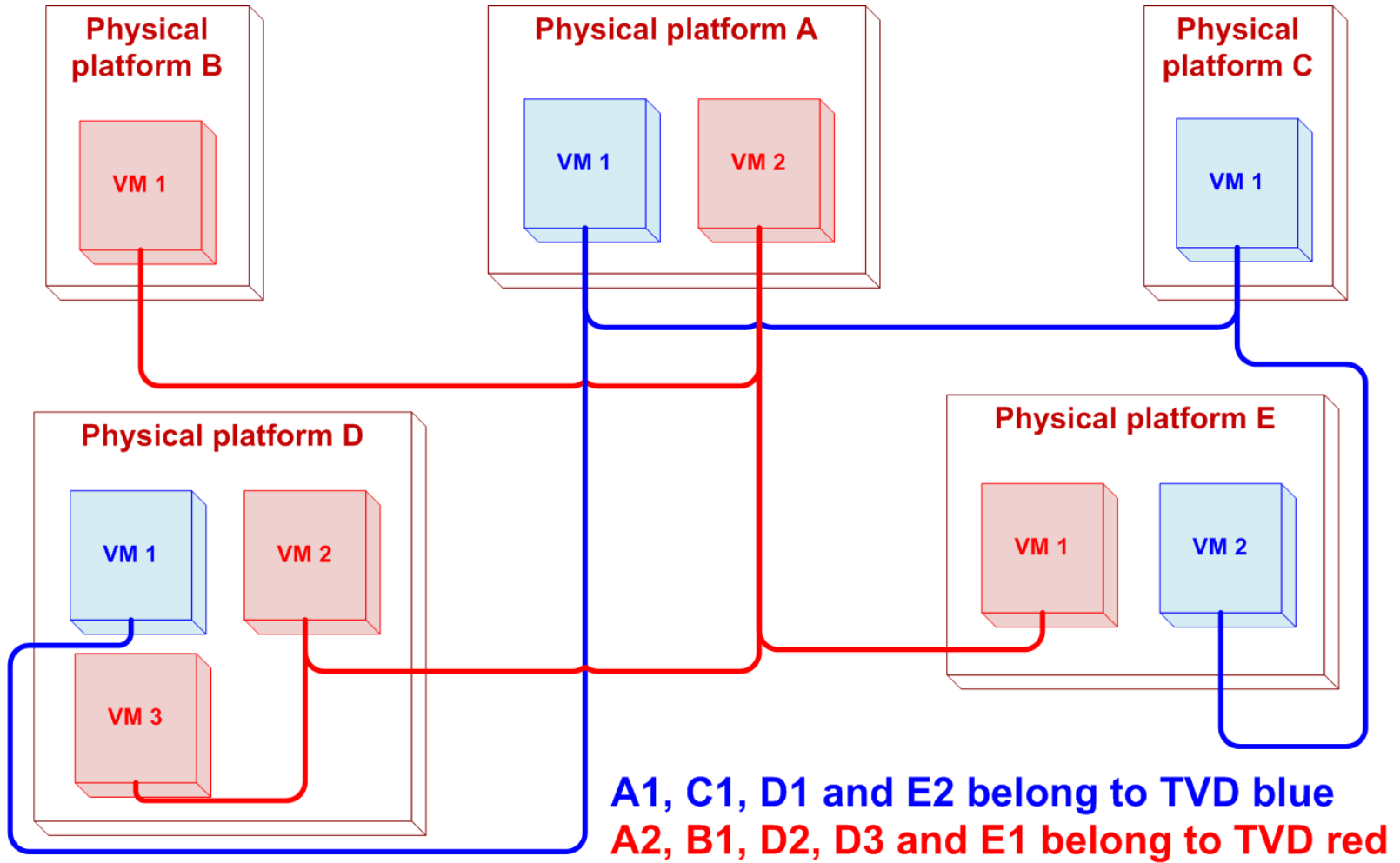
# 3<sup>rd</sup> - Virtual Data Centers (VDC)



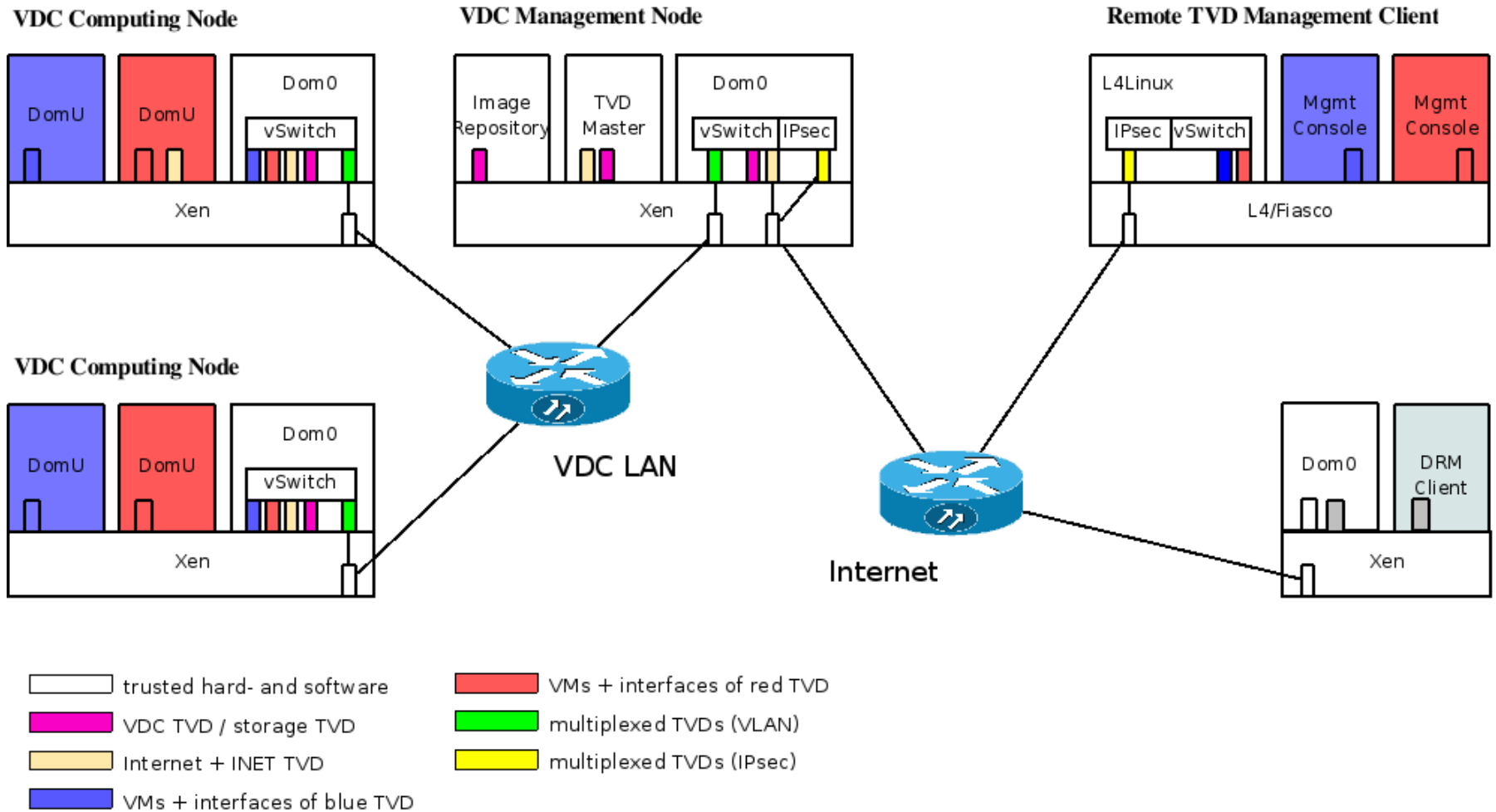


# Trusted Virtual Domain concept

- to implement the access to the corporate network we built upon the TVD concept
  - pushed by IBM research labs (T.J. Watson, Zurich)
- (one possible) definition for Trusted Virtual Domain (TVD)
  - group of virtual resources subject to a (security) policy
    - the resources belonging to a TVD can be logically isolated from the resources belonging to other TVDs
      - resources belonging to the same TVD can run on different physical platforms
        - » they can be interconnected through a trusted network
      - resources belonging to different TVDs can run side-by-side on the same physical platform



# VDC: Data Center Physical Layout





# OpenTC: Mobile and Embedded Systems

- Investigation on the use of Trusted Computing technologies and virtualization on mobile and embedded devices
  - a thorough examination of the TCG and OMTP standards
  - definition and security analysis of several use cases that are relevant for mobile scenarios
  - the development and analysis of the Secure Wallet use case as an example scenario
  - the port of basic microkernel-based operating system components from other OpenTC WPs (in particular: the L4 microkernel, L4 environment, L4Linux)
  - the port of the TPM emulator and its modification to use security features of the mobile hardware

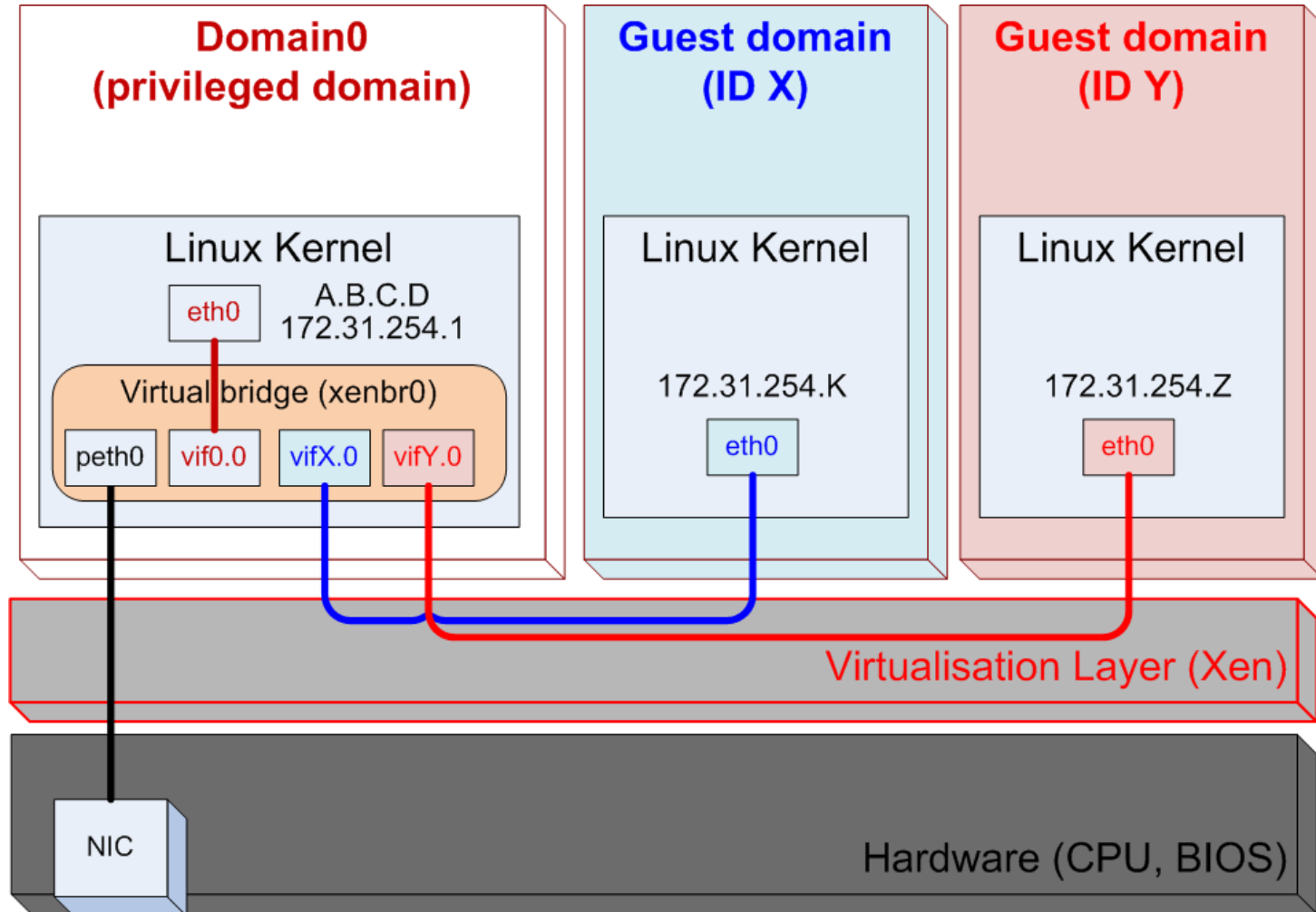


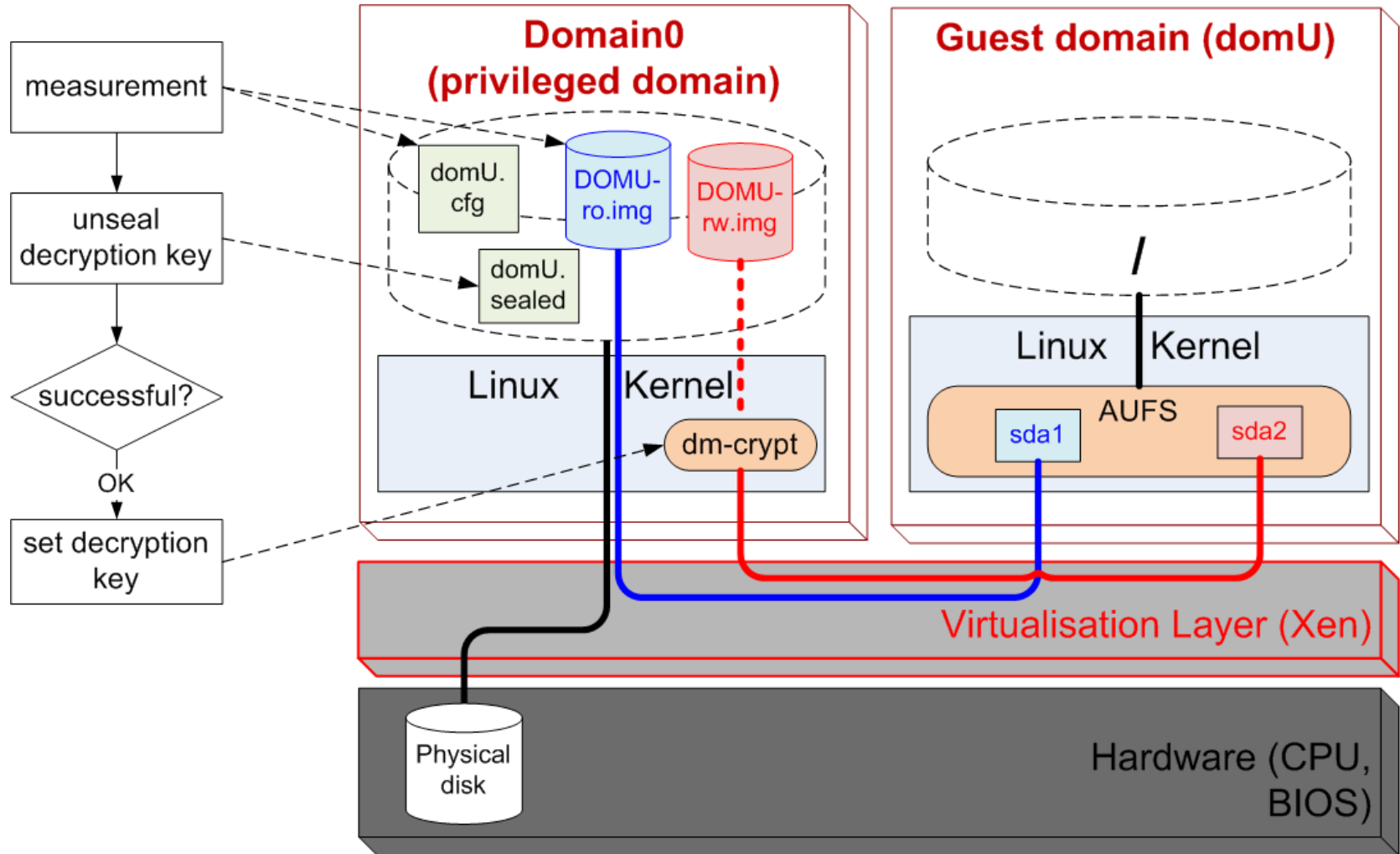
## OpenTC: other relevant aspects

- Architecture for accelerated 3D graphics in VMs via Gallium stack
  - Xen3D
- Standardization effort
  - Trusted Computing Group
  - Java community
- Validation, Verification and Testing
  - White and Black-box testing
    - TSS
  - Static analysis and code review
    - Virtualization layer (Xen and L4)
  - Work on Protection Profile and Trust methodology/metrics

**Thanks for the attention!**

**Any question?**

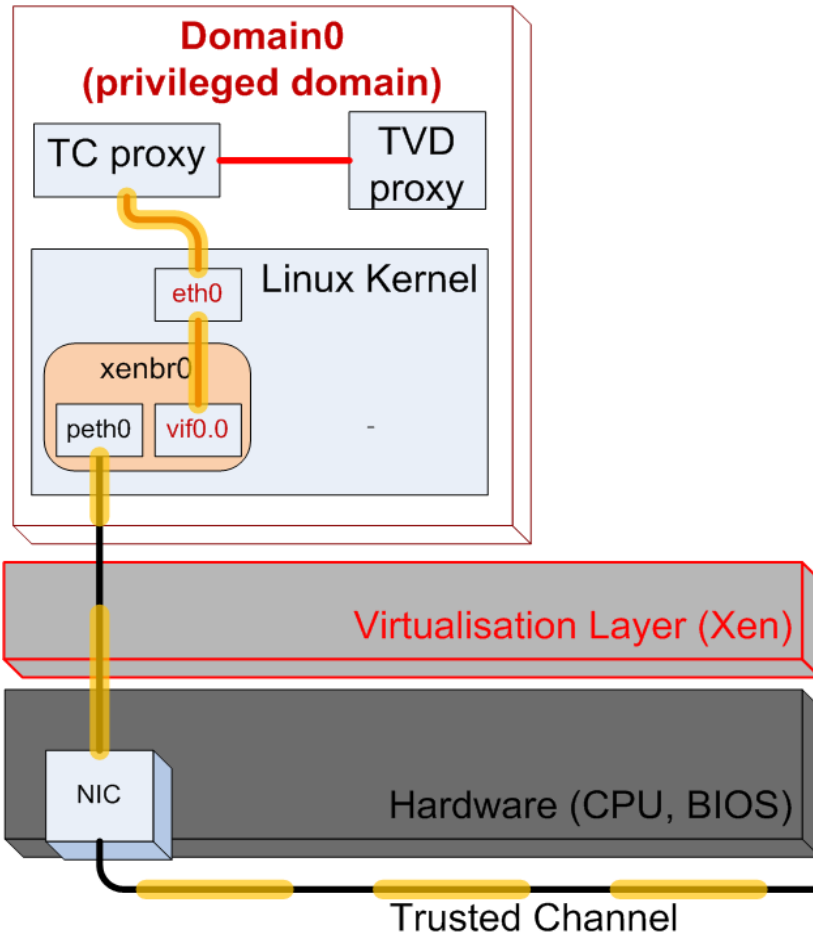




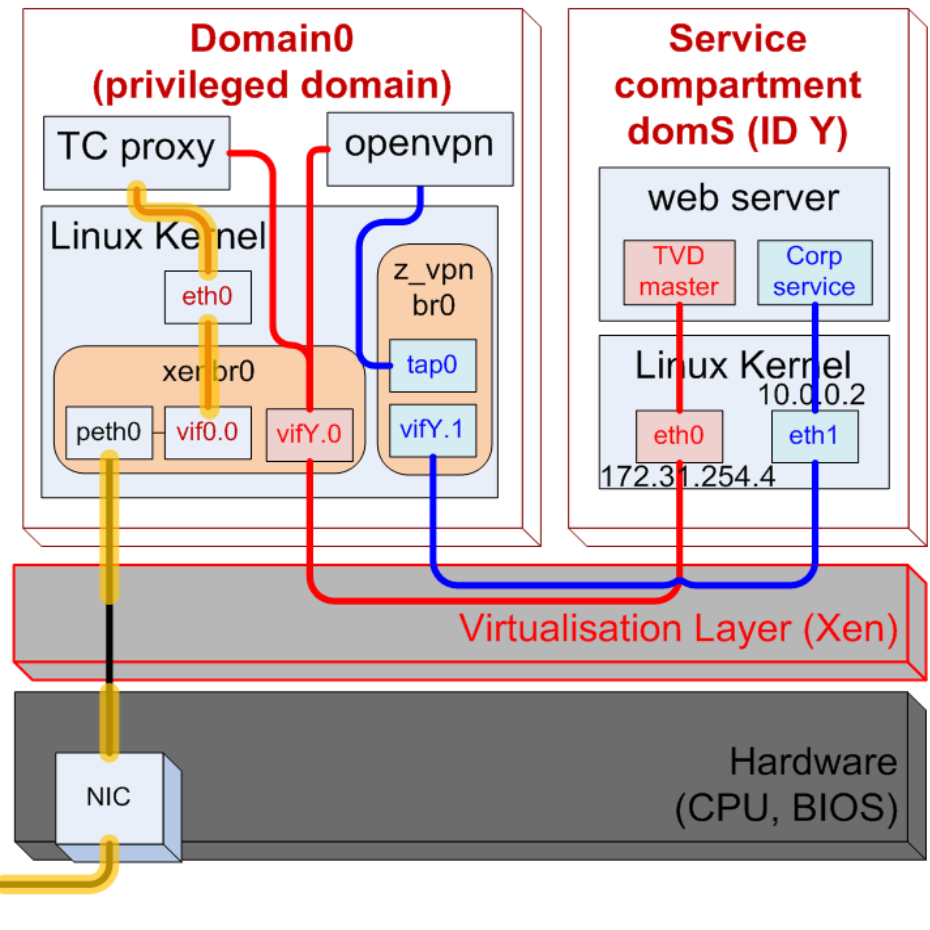


# CC@H: Trusted Channel via proxies for TVD pre-admission phase

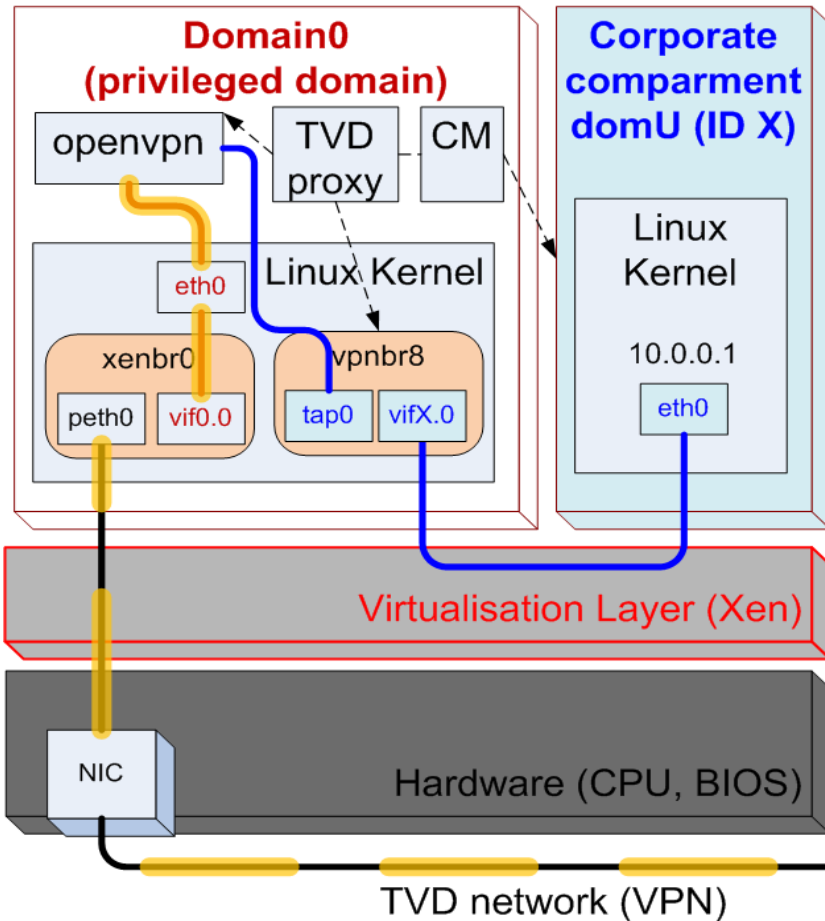
CC@H client (configuration#1)



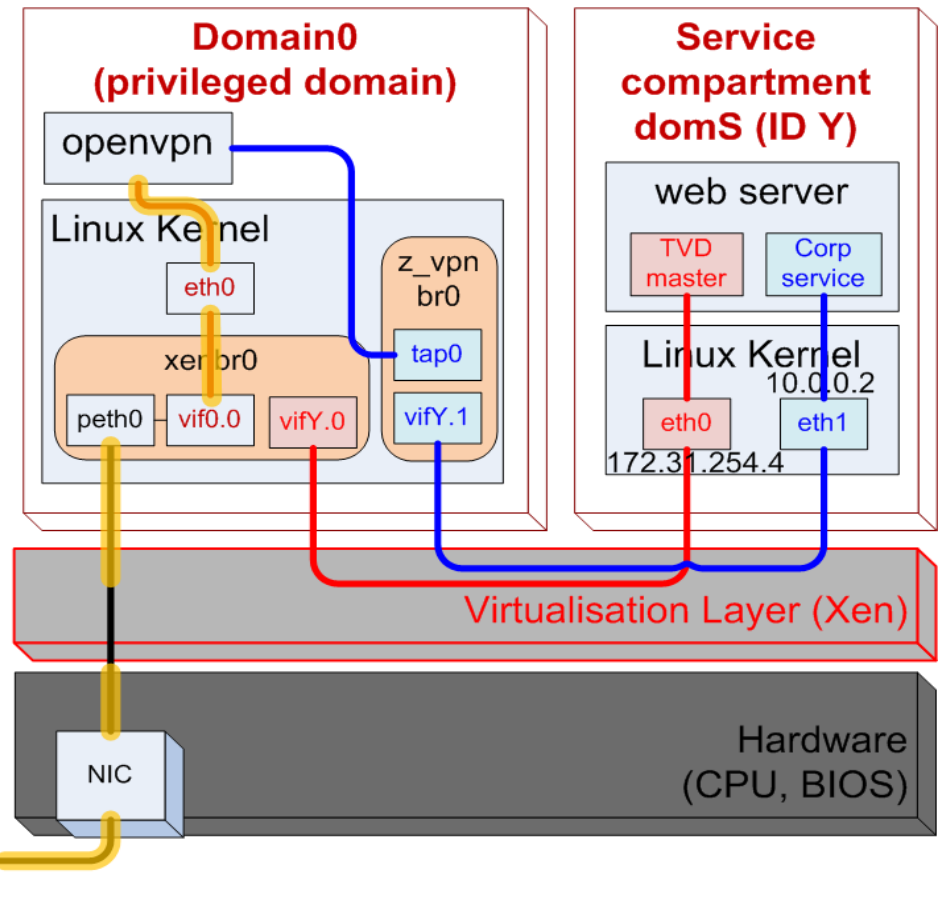
CC@H server (configuration#2)



CC@H client (configuration#1)



CC@H server (configuration#2)





## CC@H: some facts about the prototype

- linux distribution
  - commercial grade openSUSE 10.3
- building the prototype
  - harmonisation of the building tools
    - all source code compiles with the latest GCC/G++
  - OpenTC components packaged as RPMs
    - using the public SUSE build server
  - OpenTC prototype generated using SUSE KIWI imaging system
    - developed KIWI templates and wrapping scripts
    - CC@H completely generated “from scratch” in about 1.5 hour



## CC@H: some facts about the prototype - 2

- first rough support for measuring persistent root file system
  - for dom0 and other guest domains (RO and RW layers)
  - currently working on a Hierarchical Integrity Measurement (HIM) architecture
- integrated Xen/L4 prototype
  - for all domains the same root file system image can be started with both Xen and L4
- support for MS Windows XP (Xen only, HD installation only)



## CC@H: which are the used components?

- VMMs (or hypervisors)
  - XEN hypervisor, L4 Fiasco  $\mu$ -kernel
- boot loaders
  - Trusted GRUB (S-CRTM)
  - OSLO for TPM 1.2 and AMD CPUs (D-CRTM)
- GUI
  - proof of concept of unified secure GUI running under Xen
  - real secure GUI for L4
- security services
  - compartment manager and sealed storage
  - secure virtual network: TVD components
  - TC proxy for remote attestation
- VLC multimedia player (video and audio in a VM)



## CC@H: everything is open source!

- OpenTC CC@H Proof of Concept prototype

- live CD released in mid July 2008:

- [http://ftp.suse.com/pub/projects/opentc/period\\_2-POC\\_CCAH/](http://ftp.suse.com/pub/projects/opentc/period_2-POC_CCAH/)**  
**[ftp://ftp.suse.com/pub/projects/opentc/period\\_2-POC\\_CCAH/](ftp://ftp.suse.com/pub/projects/opentc/period_2-POC_CCAH/)**

- all software components are open source (GPL, CPL)

- the packages are automatically created/updated using the SUSE public build server

- <https://build.opensuse.org/>**

- the source and binary package can be downloaded from (choose openSUSE 10.3)

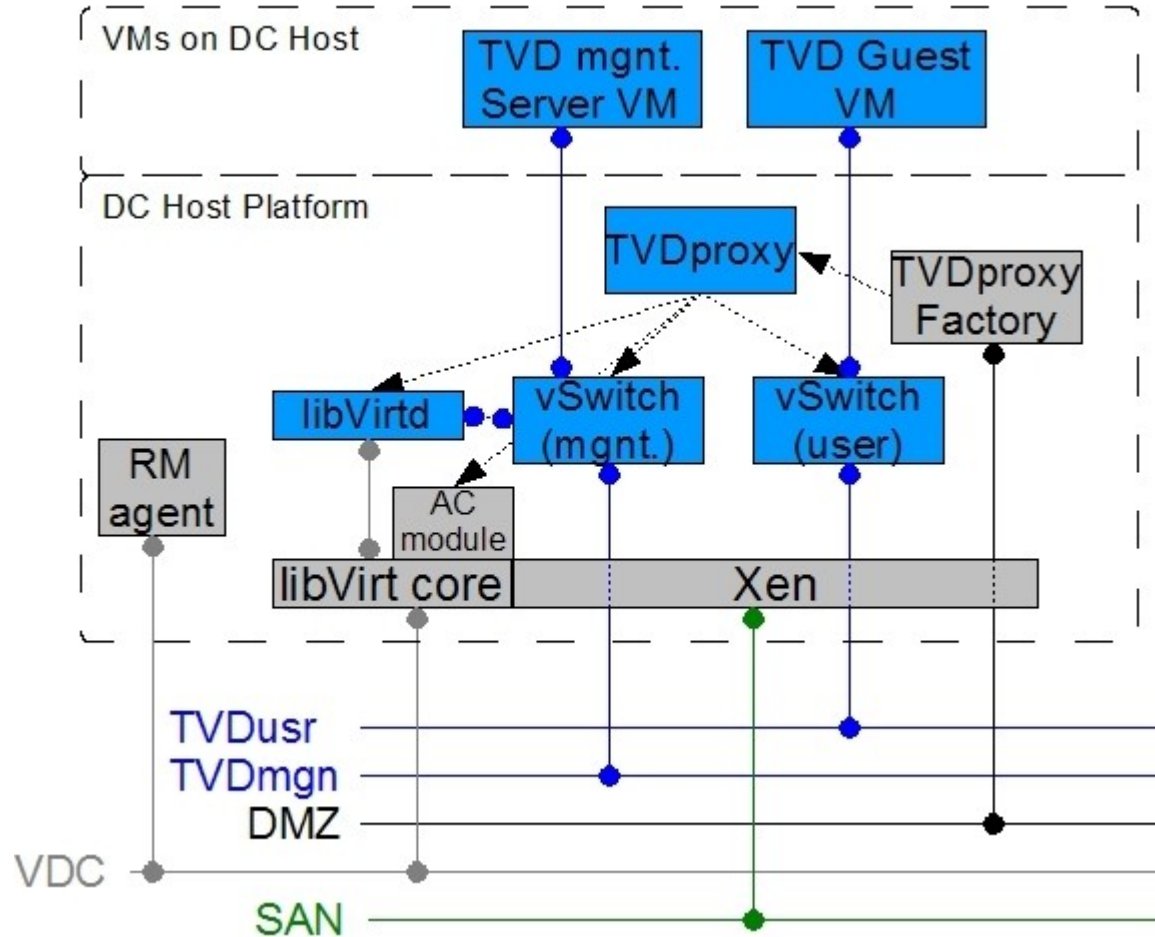
- <http://download.opensuse.org/repositories/security:/OpenTC>**



## CC@H: stay tuned ([www.opentc.net](http://www.opentc.net))!

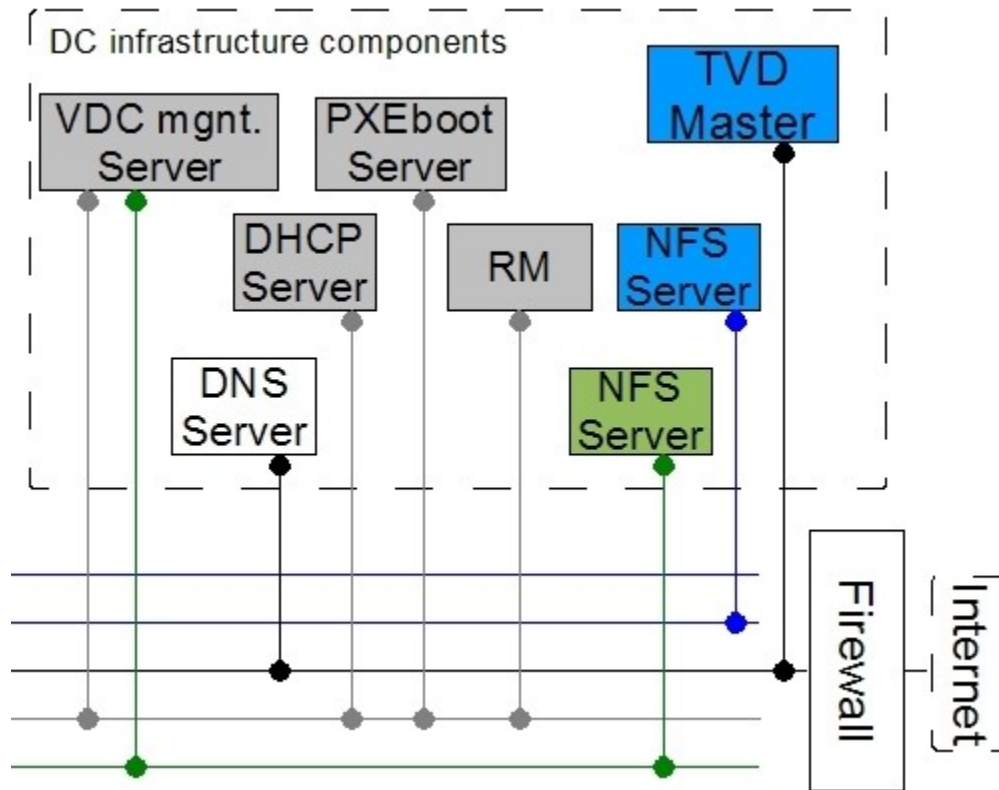
- OpenTC CC@H Proof of Concept prototype
  - KIWI templates and scripts will be released in fall 2008
    - to build your own instance for hard disk installation
    - to re-build the live CD with updated components
    - to play with it (modify for your own purposes)
  - future versions of CC@H in fall 2008 or spring 2009 (also VDC)
- to be informed when templates and future versions will be released
  - check our web site  
**<http://www.opentc.net>**
  - subscribe to the OpenTC newsletter  
**[http://www.opentc.net/index.php?option=com\\_forme&fid=4](http://www.opentc.net/index.php?option=com_forme&fid=4)**
- please send us your feedback about CC@H prototype  
**[http://www.opentc.net/index.php?option=com\\_forme&fid=3](http://www.opentc.net/index.php?option=com_forme&fid=3)**

# VDC: Data Center Host Platform



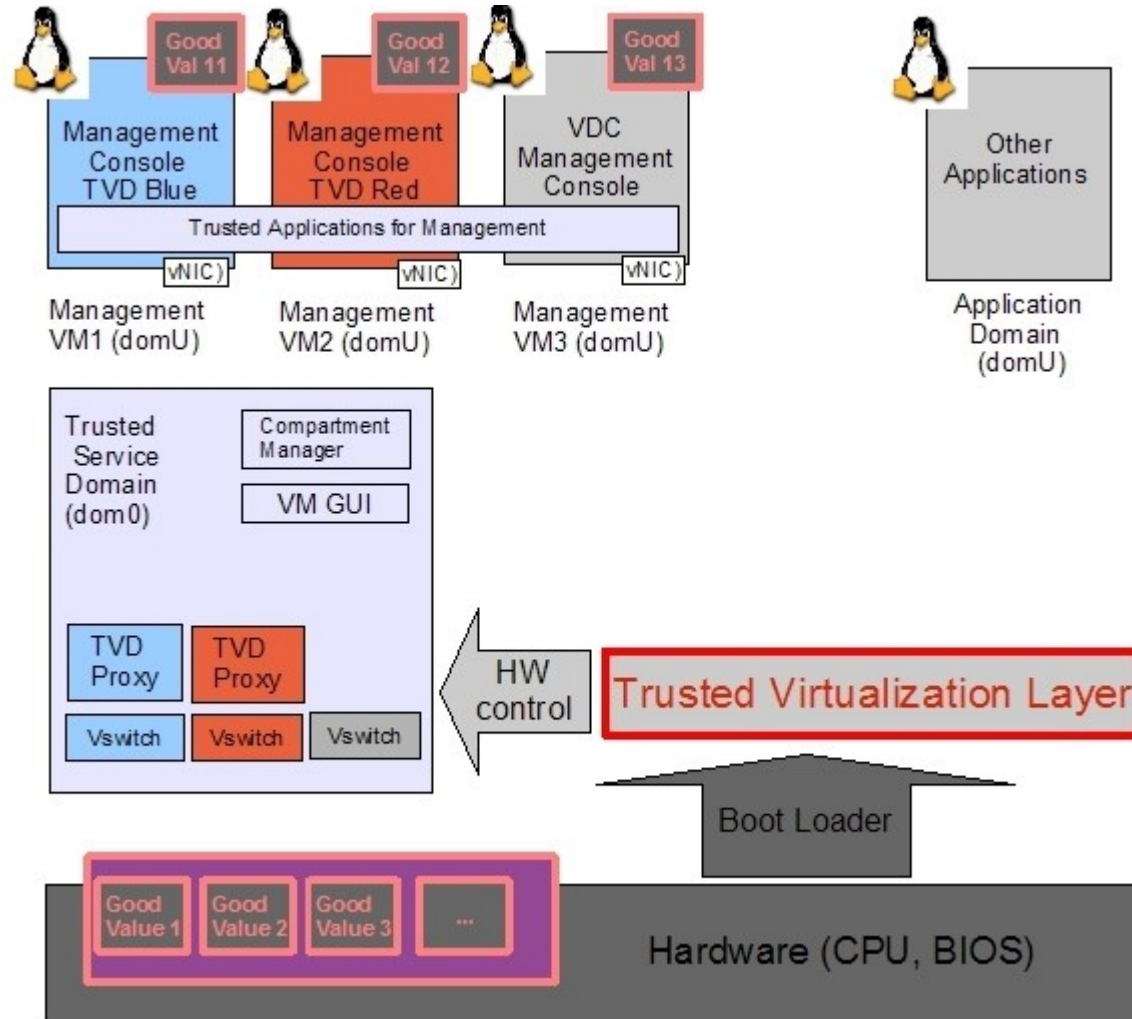
Graphics courtesy of K. Eriksson, IBM

# VDC: Data Center infrastructure services



Graphics courtesy of K. Eriksson, IBM

# VDC: TVD Management Console





## Open\_TC EC Contract No: IST-027635

The Open-TC project is co-financed by the EC.

If you need further information, please visit our website [www.opentc.net](http://www.opentc.net) or contact the coordinator:

Technikon Forschungs- und Planungsgesellschaft mbH  
Richard-Wagner-Strasse 7, 9500 Villach, AUSTRIA  
Tel. +43 4242 23355 – 0  
Fax. +43 4242 23355 – 77  
Email [coordination@opentc.net](mailto:coordination@opentc.net)

The information in this document is provided "as is", and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.