# Trusted Computing: A universal security infrastructure?

Chris Mitchell
Information Security Group
Royal Holloway, University of London
(Visiting Erskine Fellow, University of Canterbury)

## Contents

- What is trusted computing?
- The TCG
- Using trusted computing functionality
- Security infrastructures
- Using the TPM to support a universal security infrastructure
- Conclusions

## Contents

- What is trusted computing?
- The TCG
- Using trusted computing functionality
- Security infrastructures
- Using the TPM to support a universal security infrastructure
- Conclusions



## Trusted Computing

- We start by introducing the notion of Trusted Computing.
- The notion originates from the Trusted Computing Group (TCG) – in fact from its predecessor body, the TCPA.
- The first fruits of what has been a large scale research and development effort are now visible in the form of a secure chip on the motherboards of many new PCs.
- Microsoft Vista incorporates support for these chips, and uses them as the basis of certain novel security functions.
- Open source software also exists capable of exploiting this hardware.
- However, the full potential remains to be exploited.

## A trusted system

- A trusted system or component is one that behaves in the expected manner for a particular purpose.

  [Trusted Computing Group – www.trustedcomputinggroup.org]
- This is difficult to achieve this for a PC – where typically there is no way of telling whether the ‘real’ (uncorrupted) Windows is running.
- As a result there is no way of getting any confidence in the correct running of applications. [Even if the operating system says that everything is OK, then this does not help because it cannot be believed].
- It is even more difficult to prove to a third party that the state of a PC is as claimed.



## Fundamental requirements

- First we need a way of achieving assurance that the operating system has booted correctly.
- This requires assuming that the PC hardware has not been modified; this is made difficult, but not impossible, for the attacker by embedding key functions in a dedicated chip – the Trusted Platform Module (TPM).
- Need a way of checking the boot process.
- The component that checks the initial boot must be trusted – the ‘Core Root of Trust’ – this is hardware-based.
- If the loaded software has been checked (and hence is reliable), it can check the next software to be loaded, and again there is a solid basis for trust; this process is iterated.

## Monitoring the checking

- As well as performing checks during the boot process, there needs to be a reliable way of recording the results of each of these checks.
- The trusted hardware incorporates hardware registers which store hash-codes of software that has been loaded – these registers provide a reliable record of all the software that has been executed on the trusted platform.
- Anyone wishing to check the state of the platform only needs to be given the contents of these registers (as long as they know what the values 'ought to be').



## Building on the trusted base

- This base of trust can be used to support two fundamental trusted computing functions:
  - **Attestation**, where a PC can reliably attest to its software state to a third party (by describing the contents of the registers which store hashes of software state);
  - **Secure storage**, where a PC can store data in such a way that only if the PC is in a specific trusted state will the data be decrypted and available to an application (by linking the decryption keys to specific register contents).
- We now look in a little more detail at the set of technical functions provided by trusted computing (as needed to support the fundamentals we have outlined).

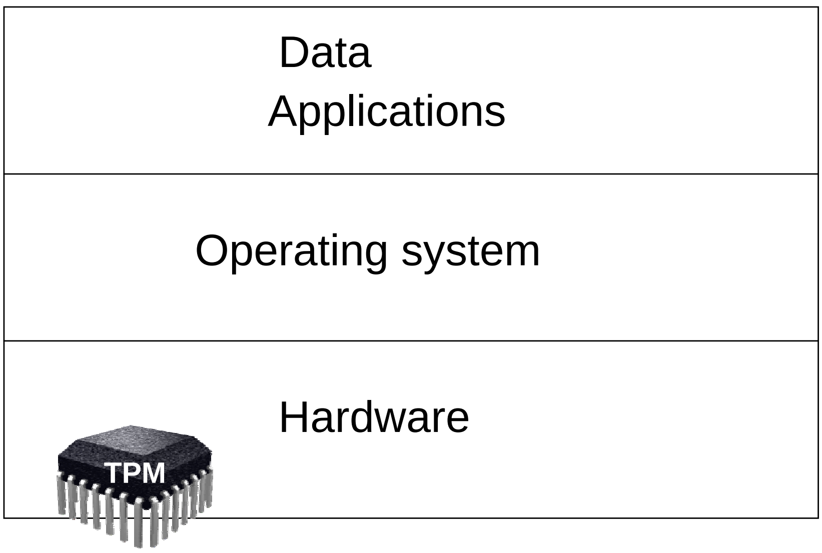
## Components of a trusted computing framework

- Shielded locations and protected capabilities:
  - Protected capabilities are those capabilities whose correct operation is necessary for the platform to be trusted;
  - Shielded locations are areas in which data is protected against interference or snooping;
  - Only protected capabilities have access to shielded locations.
- Attestation:
  - Attestation by the TPM;
  - Attestation to a trusted platform (incorporating a TPM);
  - Attestation of a trusted platform;
  - Authentication of a trusted platform.
- Integrity measurement, storage and reporting.

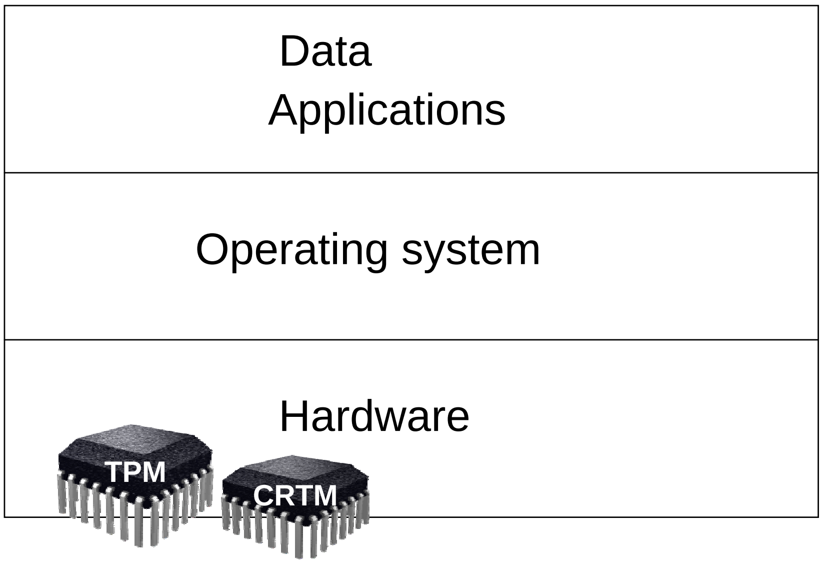
  [TCG specification Architecture Overview]



## Current platforms with integrated TPMs

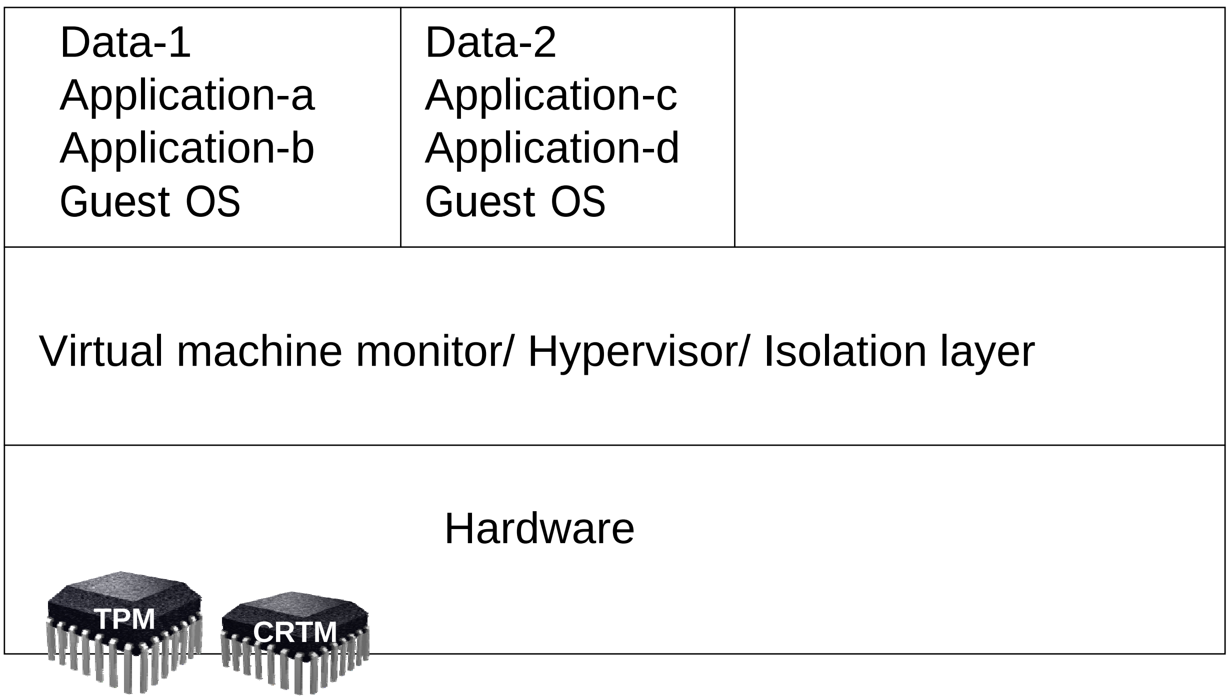


## Envisaged trusted platforms (stage 1)



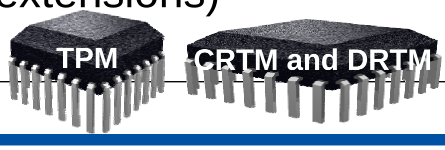



## Envisaged trusted platforms (stage 2)



## Envisaged trusted platforms (stage 3)

| | | |
|---|---|---|
| Data-1<br>Application-a<br>Application-b<br>Guest OS | Data-2<br>Application-c<br>Application-d<br>Guest OS | |
| Virtual machine monitor/ Hypervisor/ Isolation layer | | |
| Hardware (including hardware support for isolation – CPU, chipset, keyboard, mouse, video graphics card extensions) | | |






## Contents

- What is trusted computing?
- The TCG
- Using trusted computing functionality
- Security infrastructures
- Using the TPM to support a universal security infrastructure
- Conclusions

## The TCPA

- TCPA (Trusted Computing Platform Alliance): An industry working group.
- Focus: Enhancing trust and security in computing platforms.
- Originally an alliance of promoter companies (HP, IBM, Intel and Microsoft). Founded in 1999.
- Initial draft standard unveiled in late 1999.
- Invitation then extended to other companies to join the alliance.
- Specification eventually became an open industry standard.
- By 2002 the TCPA had over 150 member companies.



## The TCG

- TCG: announced April 8, 2003.
- TCPA recognised TCG as successor organisation for the development of trusted computing specifications.
- TCG adopted the specifications of the TCPA.
- Aims:
  - To extend the specifications for multiple platform types;
  - To complete software interface specifications to facilitate application development and interoperability;
  - To ensure backward compatibility.

## The TCG main specifications

- TCG TPM main specification (general platform specification) version 1.2:
  - Design principles;
  - Structures of the TPM;
  - TPM commands.
- TCG software stack (TSS) specification version 1.2.
- TCG software stack (TSS) specification header file.
- Specifications available at:
    - www.trustedcomputinggroup.org



## Contents

- What is trusted computing?
- The TCG
- Using trusted computing functionality
- Security infrastructures
- Using the TPM to support a universal security infrastructure
- Conclusions

## Microsoft Vista security features

- BitLocker – secure drive encryption using TPM features.
- BitLocker only available in Enterprise version of Vista.
- Almost certainly because it is very dangerous to users unless a proper backup strategy is also deployed.
- Visa also supports a TPM-based ‘partial’ secure boot.



## Windows Vista BitLocker

- BitLocker Drive Encryption (BDE) designed to protect data from offline viewing – uses a v1.2 TPM:
  - There is a potentially huge threat from offline attacks against PC data, particularly on notebook PCs;
  - Bonus: secure decommissioning by deleting the keys!
- Two volume categories:
  - System volume (unencrypted)
    - MBR, Boot manager and utilities
  - OS volume
    - OS, page/temp/hibernation file, data
- Five default key storage options:
  - TPM, TPM+PIN, TPM+USB, USB, Recovery password

## A crypto chip in every PC

- Putting a TPM on every PC motherboard means that every PC will have a crypto chip, with secure key storage, a random number generator, ...
- Possible security applications for such a chip are almost endless.
- For example, currently there are PC crypto boards available.
- These can be used to make a PC into a secure system, e.g. to:
  - run a Certification Authority as part of a PKI;
  - to perform key management functions for a company network;
  - ...
- In some cases, the TPM may be sufficiently secure to avoid the need for a separate crypto board.



## Managing distributed systems

- In the long term, one of the key roles envisaged for trusted computing is to enable the secure management of distributed systems (especially in a corporate setting).
- One node in the distributed system can test the level of security offered by another node before deciding what types of task it can safely delegate to that node.
- That is, security policies can be automatically enforced.
- However, there is a long way to go ...

## Other applications

- A huge variety of applications have been suggested for trusted computing functionality.
- Examples include:
  - secure signature generation;
  - digital rights management (DRM);
  - secure identities for peer-to-peer computing;
  - control of personal information;
  - ...
- However, what will actually happen is far from clear!



## Contents

- What is trusted computing?
- The TCG
- Using trusted computing functionality
- Security infrastructures
- Using the TPM to support a universal security infrastructure
- Conclusions

## Security infrastructures

- In order to use cryptography to protect communications, some kind of security infrastructure needs to be in place.
- In its simplest form, this will just be a means to set up shared secret keys between communicating parties.
- Traditionally, e.g. in banking networks, this can be achieved using one or more Trusted Third Parties (TTPs).
- One type of TTP for this purpose is known as a Key Distribution Centre (KDC).
- A KDC shares a secret key with every party, and these keys can be leveraged (using an appropriate protocol) to set up a secret key between any two parties.



## Public Key Infrastructures (PKIs) I

- A PKI is another type of security infrastructure, based on public key cryptography.
- In public key cryptography, each party has a key pair, made up of a matching public key (which can be widely disseminated) and a private key (known only to its owner).
- Such key pairs can support the use of digital signatures.
- The owner of a private key can digitally sign a message using this private key.
- The resulting signature can be verified by anyone with the correct public key (but cannot be forged with just the public key).

## Public Key Infrastructures (PKIs) II

- Whilst the public key does not need to be kept secret, the users need to know that the public key they have is correct.
- This can be achieved by using a special type of TTP known as a Certification Authority (CA).
- The CA takes a public key which it knows to belong to user X, and digitally signs a statement to the effect that X is the owner of this public key.
- This signed statement is known as a certificate, the commonly used standard for which is ITU-T X.509 (=ISO/IEC 9594-8).
- If I know the public key of the CA, I can verify any signatures it produces, and hence I can verify all the certificates it produces.
- This gives me a way of getting a reliable copy of any other entity's public key.
- The CAs and the certificates they produce are collectively known as a Public Key Infrastructure (PKI).



## The promise of a universal PKI

- A few years ago, PKI was the subject of huge amounts of hype.
- Companies producing PKI products (e.g. CA software) or providing PKI services suddenly (and temporarily!) became hugely valuable.
- In many cases the vision sold as part of this hype was of some kind of universal PKI, whereby every PC in the world would have a public key certificate, which could then be used for a huge range of purposes, e.g.:
  - secure e-commerce;
  - universal secure e-government;
  - secure home banking;
  - electronic signatures for all;
  - ...

## PKI – what happens in practice I

- Of course, this has not happened.
- In the main, we have a large number of PKIs, but each one has been set up for a specific purpose.
- For example:
  - companies have their own PKIs, used to support internal secure communications;
  - MasterCard and Visa (and card issuing banks) have PKIs set up to support EMV (used for to support smart card based credit/debit card transactions, e.. in parts of Europe);
  - Internet web sites have certificates used to support SSL/TLS security.
- There are, of course, many explanations for this – one being the fact that the policies under which certificates are issued will depend on the context of use.



## PKI – what happens in practice II

- More generally, PC users do not have the expertise or motivation to generate a signature key pair, and obtain a certificate for their public key.
- This can be seen from the failure of the SET e-commerce secure payment system, one of the major obstacles to the adoption of which was the need for every user to generate a key pair, and take a copy of their public key to their bank.
- End users cannot be expected to understand the operation of public key cryptography.
- Moreover, current PCs typically do not have a means for secure key storage (needed for the private key).

## Contents

- What is trusted computing?
- The TCG
- Using trusted computing functionality
- Security infrastructures
- Using the TPM to support a universal security infrastructure
- Conclusions



## TC – a universal security infrastructure?

- It seems possible that trusted computing may give us a universal security infrastructure ‘by the back door’.
- Every PC owner will have a crypto-capable PC, will have an asymmetric key pair in their TPM, and will have a public key certificate for the public key.
- Moreover, the TPM is capable of generating signature key pairs on demand, of using generated private keys to sign arbitrary data, and of providing secure storage for private keys.

## Possible problems

- The key pair provided in every TPM (when shipped to user) is not suitable for use as a general purpose key pair:
  – although it is an RSA key pair, it is intended for use as an encryption/decryption key pair;
  – the TPM does not enable its use for signing arbitrary data.
- The TPM is capable of generating an RSA key pair designed for signing (known as an AIK – Attestation Identity Key), and can also obtain an X.509 certificate for the public part of the AIK from an entity known as a Privacy-CA.
  – However, the private part of the AIK cannot be used to sign arbitrary data.



## Solutions to problems

- Get the TPM to generate another signature key pair, and use an AIK to sign a ‘certificate’ for the public key.
- The private key of this key pair **can** be used to sign arbitrary data.
- This means that the PC now has a means of generating arbitrary numbers of signature key pairs (essentially automatically) and obtaining certificates for them.
- Only problems are:
  a) There is a need to associate two certificates with each key pair (the Privacy-CA certificate for the public AIK, and the AIK-signed certificate for the public key in use);
  b) The AIK-signed certificate is not in the standard (X.509) format.

## Certificate 'translation'

- A means of addressing these last two problems has been proposed by the TCG.
- They propose a special extension to PKCS#10 (PKCS#10 is a format for submitting public key certification requests to a CA).
- This extension (called SKAE) would allow a PC to submit the PC-generated certificate (signed using an AIK) for the signature public key, along with other evidence, as part of a certification request.
- The CA would verify the certificate and evidence, and would then generate a new certificate for the PC public key.
- All these processes could be performed by a Java applet, which would give the PC user a secure and automatic means of joining a global PKI.



## Example 1 : SSL client side authentication

- Currently, SSL is only used for unilateral authentication i.e. of the server to the client, mainly because client PCs typically do not have key pairs and certificates.
- Precisely the procedure just described could give a means for a PC user to acquire a signature key pair and a public key certificate in order to support SSL client side authentication.
- This is described in greater detail in:

  A. Alsaid and C. J. Mitchell, 'Preventing phishing attacks using trusted computing technology', in *Proceedings of INC 2006, Sixth International Network Conference, Plymouth, UK, July 2006*, pp.221-228.
- Related work, including implementations, is being conducted by the OpenTC project.

## Example 2: Secure PC-based electronic signatures

- A considerable amount of work has gone into developing legislative and commercial frameworks for electronic signatures.
- However, such frameworks typically require a cumbersome registration procedure for users, and also some means of storing private keys securely.
- The possibility exists that, with the aid of the TPM in a PC, the PC itself can become a trusted platform for the implementation of a personal electronic signature capability, since it can provide the secure storage and also automatically perform the registration procedures.



## Portability and privacy issues

- The problem remains that PCs are not typically in one-one correspondence with human users.
- That is, users have multiple PCs (and transferring secrets between TPMs is difficult), and PCs may have multiple users.
- In the latter case. issues may arise in holding a single user accountable for the behaviour of a PC.
- However, TPMs are ‘owned’ by a single user, which typically means that only one individual will be able to use the TPM-protected keys.
- If users wish to have multiple ‘unlinkable’ identities, the TPM can support this, by generating new key pairs as required. (Privacy-preserving certification and use of cryptography is a key feature of the TCG specifications).

## Contents

- What is trusted computing?
- The TCG
- Using trusted computing functionality
- Security infrastructures
- Using the TPM to support a universal security infrastructure
- Conclusions



## Resources

- www.trustedcomputinggroup.org
- http://www.microsoft.com/windowsvista/default.aspx
- http://www.intel.com/technology/security/
- http://os.inf.tu-dresden.de/L4/LinuxOnL4/
- http://www.opentc.net/
- Siani Pearson (editor), *Trusted Computing Platforms – TCPA Technology in Context*, HP Invent.
- Chris Mitchell (editor), *Trusted Computing*, IEE (London), 2005.

# Acknowledgements

- Must first thank Eimear Gallery and Stéphane Lo Presti for preparing many of the slides in this presentation.
- Would also like to thank the University of Canterbury for hosting me as a Visiting Erskine Fellow (in the Department of Computer Science and Software Engineering), and Ray Hunt for organising everything.
- Finally, this talk has been prepared as part of the European OpenTC project.



# OpenTC EC Contract No: IST-027635

The OpenTC project is co-financed by the EC.

If you need further information, please visit our website 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