Interested in learning more about security?

# SANS Institute InfoSec Reading Room



## A Survey Of Trusted Computing Specifications And Related Technologies

2003 is the year that the much talked about technologies of Microsoft's Next Generation Secure Computing Base and Intel's LeGrande will be deployed to provide consumers with Trusted Computing. The Trusted Computing Platform Alliance has handed over the batton to the Trusted Computing Group, and new specifications are to be announced soon. But what do these companies and industry groups mean by Trusted Computing, and how will it work in practice? This paper seeks to survey the key points of these technologies and provid...

# A SURVEY OF TRUSTED COMPUTING SPECIFICATIONS AND RELATED TECHNOLOGIES

Ricard Kelly
28 April 2003

## ABSTRACT

2003 is the year that the much talked about technologies of Microsoft's *Next Generation Secure Computing Base* and Intel's *LeGrande* will be deployed to provide consumers with *Trusted Computing*. The *Trusted Computing Platform Alliance* has handed over the batton to the *Trusted Computing Group*, and new specifications are to be announced soon. But what do these companies and industry groups mean by Trusted Computing, and how will it work in practice?

This paper seeks to survey the key points of these technologies and provide a framework for suggesting whether a TCPA/TCG or NGSCB architecture will improve security in an environment and where it may reduce security.

## INTRODUCTION TO TRUSTED COMPUTING

Trusted Computing is a term being used to describe a platform with two properties:

- The ability to assert a metric for integrity

  In order to allow an application to verify that the platform it is running on fulfils a baseline for acceptability to the application; and to allow the platform to incorporate the application into the platform's calculation of its own integrity.

- The ability to seal some storage area to a particular application and platform state

  In order to provide a location for an application to store information that it wishes to remain privileged to itself or another specified component. The integrity of the platform must meet certain conditions for the information to be retrieved, and these conditions are set at the time of storage.

In order to provide for these two properties, and to extend them as far through the system as possible, vendors also add other components such as short-to-medium term secure memory and secure (encrypted or separated) channels through the system.

The following sections explain these concepts in greater detail.

## ASSERTING A METRIC FOR INTEGRITY

Integrity proofs deal in:

- proving the platform's integrity to running applications; and
- proving an application's integrity to the platform.

The first proof convinces the application to trust the platform and to store encryption keys for protected data with the platform. The second proof convinces the platform to provide the encryption keys back to the application.

In order to convince an application that the platform it runs on meets some criteria for integrity, the platform must have a method for evaluating that integrity.

By having a trusted certifier sign components the platform can check whether the components are certified for a given level of integrity. The certifier may be the system administrator, the platform provider or an external certifier depending on the implementation.

This brings us to some of the concerns critics have with Trusted Computing proposals:

- verification will focus on expected behaviour rather than correctness, and platform integrity may be incorrectly asserted

  In the average operating system or application, bugs are found regularly in one component or another. Provable correctness of components is highly unlikely and where a certificate of integrity is issued this is more likely to be based on whether the component's stated operations are in line with the certifier's concept of integrity.

  Different implementations of the same basic component (e.g. a video card) will operate with potentially subtle differences. The certifier may sign a component that is not securely implemented despite appearing correct in specification. A compromise may be performed of the component while it is trusted by the platform or application. This could reveal data, storage encryption keys or platform keys.

  While bugs in components that are trusted by the platform are unlikely to be more damaging that bugs in today's operating systems, this does negate benefits of trusting the components in the first place.

- verification will only consider components that are understood to affect integrity

  When an application requests information on the integrity of the platform, it can not consider components that it is unaware of. One of the biggest challenges faced by a secured application will be

establishing whether all components that are relevant to the operation of the application are being verified, and that no components which do not affect the application are being required to be trusted.

An application can not always be sure that the platform is not being subverted by a component that it does not know about. A user can not always be sure that a new hardware component is not going to stop existing applications trusting the platform.

An example much touted by opponents is Digital Rights Management (DRM) being used to protect artistic content.

At some point, the content must be decrypted for a person to view or hear. If a component is incorrectly trusted to not reveal content it may be subverted to do so. If the content traverses a component normally trusted in the platform but not specified by the application as requiring trust, the component may be replaced and the platform will still provide the storage key to the application. In neither situation can the platform as a whole be trusted to protect the content.

An additional issue arises where the certifier may have an economic benefit in withholding certification for a component. The certifier may elect to not certify a component as secure where the certifier wishes to prevent the component being used by an application. In any real-world environment, security of a component is likely to be subjectively decided, and a certifier is unlikely to need to prove vulnerability in order to refuse to sign a component.

A methodology removing this issue is permitting the platform to accept any certifier as valid for verifying security where that certifier is trusted by the platform administrator. An application verifying integrity would then also need the certified identity of the integrity certifier. So long as the roots of this certification tree are general purpose certificates rather than special purpose certifiers of hardware, an end-user may install as many certifiers as they feel comfortable with. An application may elect to either trust the end-user administrator to only accept certificates from trusted parties; or trust only parts of the tree.

An example (including the later section of secured storage) could be a 'Media Downloader' provided by a content producer. This application could check that all components of a system are acceptable before downloading encrypted content and providing an encryption key to the platform with the requirement that specified components remain trusted by specified certifiers. A Media Player would then not be able to retrieve the key should the components be trusted by the administrator but not the content producer.

Some anecdotal information from various people 'in the know' suggests this will be a feature of the next specification from the Trusted Computing Group.

This part of Trusted Computing may be enough for an application to prove that it is running on a system that will not subvert the application's operation.

If the application has data that it wishes to protect in long-term storage (such as a DVD or hard drive), sealed storage is required.

**SEALED STORAGE**

In this case, the proofs of integrity are purely to allow components that exist separately to trust each other enough to pass protected data between themselves. Sealed storage is implemented by the specifications as allowing for an application to provide the platform with an encryption key that may be revealed to a requester under certain specified conditions. These conditions are based on integrity metrics similar to those that the platform used to prove its trustworthiness to the application.

For example, an application may provide an encryption key for a video stream to the platform along with the conditions that all components of the system that the video stream will pass through (such as an MPEG2 decoder) are trusted at the time of the key being retrieved.

When the application requests the key to decrypt the video stream, the platform ensures that all components which the application specified meet the conditions stored with the key. If the conditions are met, the key is revealed to the application. The conditions may (and should) also include the integrity of the application itself.

The two primary concerns with sealed storage are:

- changes in the operating environment changing the integrity metrics

  When a component of the system is changed by an administrator (either hardware or other components) the integrity of these components must be checked by the platform. In the case where the application requires a particular certified component that was present at the time of key storage, the changed system may have a trusted component providing the same services but not seen as the same component. In this case the storage key may not be released.

- arbitrarily tight restrictions being placed on data

  Content providers may not permit end-users rights to content that they would normally have in a non-digital environment. Consider the example of an audio recording. Existing copyright laws permit fair use for parody purposes, but in a digital domain controlled by trusted components it may not be feasible to obtain a decrypted audio stream from the content.

  As copyright law only provides for the right to attempt this fair use rather than requiring it to be possible, the content providers would not be required to provide for this use. Also, with laws such as the Digital Millennium Copyright Act (DMCA) being passed, the technical research

that copyright law would seem to permit in attempting fair use is now illegal as copyright circumvention.

To resolve these concerns are more difficult.

In the case of changes to operating environment, access to the content on the platform relies on the original content being reprovided under the new platform description or the system being recertified in a manner permitting retrieval of the key.

In the case of rights limitation, the only method is for the content provider to grant additional permissions or a circumvention technique to be developed (which in the initial case may be as simple as finding somewhere the data is clear-text such as an audio feed to a speaker, but will get progressively harder as further components in systems are secured).

## BACKGROUND TO SPECIFICATIONS

A subsystem for enabling trust in computing platforms was specified by the Trusted Computing Platform Alliance (TCPA) in a document released in August 2000. The technologies involved bore a resemblance to a 1999 design by alliance member IBM. The latest public update to this document is version 1.1b released in February 2002 [TCPA1]. Microsoft Corporation have been awarded two patents [MS1, MS2] relating to a secure computing base, providing more public information regarding their likely implementation of parts of the TCPA specification.

Finally, Microsoft and Intel have both announced forthcoming implementations of internally-developed technologies that will provide parts of the overall Trusted Computing platform.

## TRUSTED COMPUTING PLATFORM ALLIANCE

The Trusted Computing Platform Alliance was formed to develop a specification for a hardware solution for trusted computing. The specification provides for:

- proof of the integrity of the platform to an application wishing to store information
- secure storage and retrieval of application information through encrypted storage

The founding member companies of the TCPA were:

- Compaq Computer Corporation
- Hewlett-Packard Company
- Intel Corporation
- International Business Machines Corporation
- Microsoft Corporation

By April 2003, the TCPA included 200 members.

## PALLADIUM AND THE NEXT GENERATION SECURE COMPUTING BASE

Microsoft Corporation developed Palladium to be an operating system service that uses some aspects of the TCPA specification but also rewrites some of its details to provide for a broader applicability. In January 2003, the name was changed to the Next Generation Secure Computing Base (NGSCB) and information was published on the architecture of this technology.

Microsoft's stated goal for NGSCB is to protect software requiring a trusted environment from all other software running on the platform (including the operating system providing services to untrusted applications). [MS3]

This is a very broad goal, relying on four concepts (as termed by Microsoft):

- Attestation

  > The ability for a piece of code to digitally sign or otherwise attest to a piece of data and further assure the signature recipient that the data was constructed by an unforgeable, cryptographically identified software stack. [MS3]

  This relates to the concept of integrity proofs. The software stack is proved by signed assertion and the data is signed by the provider (which could conceivably be a software stack running on another system).

- Sealed storage

  > The ability to store information securely so that a nexus-aware application or module can mandate that the information be accessible only to itself or to a set of other trusted components that can be identified in a cryptographically secure manner. [MS3]

  Sealed storage is an interesting component in that it: can not work without the other components; without it the other components can only assert platform security; and with it a platform can enforce data accessibility. Sealed storage is the key component of a Digital Rights Management system.

- Protected memory

  > The ability to wall off and hide pages of main memory so that each nexus-aware application can be assured that it is not modified or observed by any other application or even the operating system. [MS3]

  The key requirement for this is so that an application or system component may hold unencrypted data for a period of time without fear

that even another trusted component may read the data. This obviously requires additional memory management capabilities within the processor and that the base of the secured software stack is trusted to correctly identify the owner of a memory region.

- Secure input and output

> A secure path from the keyboard and mouse to nexus-aware applications, and a secure path from nexus-aware applications to a region of the screen. . [MS3]

This requirement allows for a trusted application to obtain and provide data in user interactions with only trusted components between the application and the hardware interfaces. This is also a hardware requirement and may require encrypted transfers to be implemented completely.

The first two requirements are met by hardware defined in the TCPA specification, but the other two requirements are additional. Microsoft has termed the hardware component providing services to the nexus as the Security Support Component (SSC). The Microsoft description of the SSC suggests it is an implementation of the TCPA. It is likely that Intel's LeGrande technology (to be embedded in forthcoming processors) will implement the requirements of the SSC.

This technology is not designed to provide defences against hardware-based attacks that originate from someone in control of the local machine.

The key component of NGSCB is the *nexus*. The nexus is a new OS component and is described by Microsoft as a "trusted operating root."

> The nexus is essentially the kernel of an isolated software stack that runs alongside the existing software stack. The nexus provides a limited set of APIs and services for applications, including sealed storage and attestation functions. [MS3]

Some key points that Microsoft make are:

- A nexus is not the basis for the operating system

A nexus operates as the kernel-mode component of a new software stack (in the manner that the existing operating system and applications form a software stack).

> This component and the special processes that the nexus commissions, called nexus computing agents (NCAs) offer a parallel execution environment to the "traditional" Windows kernel. NGSCB creates a new environment that runs alongside the OS, not underneath it. [MS3]

Existing applications that do not bind to the nexus-provided interfaces will continue to operate on top of the existing operating system and its services. Nexus-aware applications will run on top of the nexus and its agents. Whether these applications will have a method for communication either through the nexus or a different mechanism is unknown at this point.

- A nexus can be written by anyone

  Microsoft does not want to appear to be creating a monopoly position by being the only vendor for the kernel component of this system.

  > It will be possible, of course, to write applications that require access to one or more nexus-aware services in order to run. Such applications could implement access policies, enforced by a nexus-aware application, which would allow the application to run only if it has received some type of cryptographically signed license or certificate. [MS3]

  It is most likely that an application will be able to verify that a nexus it trusts is running. In this case it is unlikely that a user-supplied nexus will be trusted by an application (such as a media player) enough to allow data to be accessed.

- Only one nexus will run on a machine at a time

  If the user wishes to run more than one nexus they will need to switch, making it impossible to simultaneously run applications that trust differing nexus.

  It may also be that only the Microsoft-written nexus will be trusted by Microsoft applications critical to operating system functionality.

Microsoft has stated that the first cut of their Next Generation Secure Computing Base is unlikely to provide the benefits outlined by either the published specifications, the patents or the publications of the TCG. However, this is the goal being pursued. Once all components (including widely deployed SSC-compliant hardware) are in place, incremental tightening of component permissions and the structure of the operating system is likely to occur.

**LEGRANDE**

In the second half of 2003, Intel will be releasing processors equipped with LeGrande, a technology that provides the secure hardware component of the TCPA specification and probably the features required to be the SSC of NGSCB. Intel has stated that this is a purely 'opt-in' [INT1] component and the user may disable the functionality at will.

However once all components of the TCPA and NGSCB are in place and data providers are mandating the protocols for transfer, LeGrande (or a similar component supported by NGSCB) will not be optional where this data is required.

There is also no guarantee that an operating system or other platform component will provide for disabling LeGrande while this component is running.

**TRUSTED COMPUTING GROUP**

On April 8, 2003 the Trusted Computing Group (TCG) was launched by several TCPA members. The TCG broadened the scope of trusted computing beyond the carefully limited TCPA specification with plans to develop additional hardware specifications for such devices as PDAs and digital phones. The TCG also announced that they would develop a software specification for trusted computing.

The primary reason stated for the launch of the TCG was this broadening, but at the same time the membership structure was adjusted and a scheme for licensing of member patents was announced. It may be that the removal of certain rights from non-'promoter' members and the formation of a licensable specification will turn out to have as much consequence in the wider picture.

Until the TCG release their own specifications it is not completely known what the final picture will be, but it will probably be very similar to the NGTCB as implemented in Windows 2003 Server and.LeGrande as implemented in Pentium 4 processors shipping late this year. Other companies involved in the TCG may provide input from their technologies.

**THREATS TO STANDARDISATION**

There appears to be many different competing technologies being developed by several vendors despite all of these companies being members of the TCPA and TCG (which have specifications differing from the companies' implementations). If the new TCG specification supports one or a few of the technologies but not others there could be more than one standard published. This would make compatibility between Trusted Computing platforms difficult. Services that run on these platforms, in particular Digital Rights Management, could become fragmented and not be cross-platform compatible.

There is also much debate in the wider security and IT community over the appropriateness of the specifications and what may be built on the top of Trusted Computing. In particular, many believe that the current drive to create the platform is more to do with protecting media and software companies from piracy and some aspects of copyright fair-use provisions than protecting systems from compromise.

The debate will obviously go on for some time until and after the first cut of technologies are released to the general community.

## ADVANTAGES

The Trusted Computing initiatives of the TCPA, TCG and their member organisations can be used, when appropriately implemented, to make significant dents in inappropriate code execution and data access. Administrators can use these technologies to prevent users running applications with privileges that are inappropriate. An example would be by storing data within protected storage areas and requiring these applications to not be present for the storage keys to be released.

## UNCERTAINTIES

Where the standards are unclear:

- The precise details of component certification

  If an organisation can obtain the right to be the sole certifier of 'acceptable' components for a particular purpose, that organisation will effectively own the purpose.

- Licensing of the specification and patents

  If these technologies are to be developed independent of the interests of their creators, and prevented from creating an environment where lock-in to a particular platform, provider or way of doing business is avoided, the specifications will need to be freely implementable.

  The patents held by Microsoft (although there is generally seen to be much prior art) would seem to give them an exclusive license to create the software technologies. IBM and Intel would seem to be in the same position with respect to hardware. The RAND patent licensing of the TCG would seem to permit any other organisation represented in the TCG to also implement the patents. This does not guarantee that other platforms competing with members of the TCG (e.g.: Linux) would also be able to implement Trusted Platforms that could be used in the same manner.

## DISADVANTAGES

Where applications are developed to utilise Trusted Computing (in particular, sealed storage) organisations may find that data stored can not be retrieved in the future due to any one of the following issues:

- changes in an application removing integrity

  Upgraded versions of an application may not be certified to the same level. A platform checking the integrity of the application requesting keys may not release these keys.

- changes in a platform removing integrity

  Should the platform be modified in some way, the integrity metrics may become outside the bounds set by the original application and prevent key retrieval.

- time-based integrity constraints

  If time constraints are placed on the keys in use (Microsoft have stated this will be a benefit of the technology) an application could cease to be able to retrieve data unless upgraded to a newer version regularly.

- revocation of integrity certificates

  Should the certificate asserting the integrity of a component be revoked, the platform may not release keys to an application.

**CONCLUSION**

Trusted Computing may herald many benefits for data security and integrity checking, but there are dangers inherent in requiring continuing third-party approval for data access. At any time, any one of many risk factors may remove the ability to access system components and data.

Unless the standards are developed publicly much further, and the issues presented are dealt with to prevent error or abuse, Trusted Computing could in fact reduce the integrity of companies' information through loss of access to the very data most highly prized (and thus most secured).

**REFERENCES**

All URLs checked as valid at 28 April 2003

[AND1] Ross Anderson, *Trusted Computing Frequently Asked Questions*, last update April 2003,
URL http://www.cl.cam.ac.uk/%7Erja14/tcpa-faq.html

[TCPA1] Electronic Privacy Information Center, *Microsoft Palladium*, November 11, 2002,
URL http://www.epic.org/privacy/consumer/microsoft/palladium.html

[INT1] Paul Otellini, Intel Corporation speaking at the Intel Developers Forum

[MS1] Microsoft Corporation (Assignee), *Loading and Identifying a Digital Rights Management Operating System*, US Patent #6,327,652

[MS2] Microsoft Corporation (Assignee), *Digital rights management operating system*, US Patent #6,330,670

[MS3] *Microsoft Next-Generation Secure Computing Base - Technical FAQ*, February 2003,
URL http://www.microsoft.com/technet/security/news/NGSCB.asp

[TCPA1] Trusted Computing Platform Alliance, *Main Specification Version 1.1b*, February 2002,
URL http://www.trustedcomputing.org/docs/main%20v1_1b.pdf

[IBM1] David Safford, IBM Research, *The Need For TCPA*, October 2002,
URL http://www.research.ibm.com/gsal/tcpa/why_tcpa.pdf

Last Updated: February 24th, 2018

# Upcoming SANS Training

**Click Here for a full list of all Upcoming SANS Events by Location**

| | | | |
|---|---|---|---|
| CyberThreat Summit 2018 | London, GB | Feb 27, 2018 - Feb 28, 2018 | Live Event |
| SANS London March 2018 | London, GB | Mar 05, 2018 - Mar 10, 2018 | Live Event |
| SANS Secure Osaka 2018 | Osaka, JP | Mar 12, 2018 - Mar 17, 2018 | Live Event |
| SANS San Francisco Spring 2018 | San Francisco, CAUS | Mar 12, 2018 - Mar 17, 2018 | Live Event |
| SANS Paris March 2018 | Paris, FR | Mar 12, 2018 - Mar 17, 2018 | Live Event |
| SANS Secure Singapore 2018 | Singapore, SG | Mar 12, 2018 - Mar 24, 2018 | Live Event |
| SANS Northern VA Spring - Tysons 2018 | McLean, VAUS | Mar 17, 2018 - Mar 24, 2018 | Live Event |
| ICS Security Summit & Training 2018 | Orlando, FLUS | Mar 18, 2018 - Mar 26, 2018 | Live Event |
| SANS Munich March 2018 | Munich, DE | Mar 19, 2018 - Mar 24, 2018 | Live Event |
| SEC487: Open-Source Intel Beta One | McLean, VAUS | Mar 19, 2018 - Mar 24, 2018 | Live Event |
| SANS Pen Test Austin 2018 | Austin, TXUS | Mar 19, 2018 - Mar 24, 2018 | Live Event |
| SANS Secure Canberra 2018 | Canberra, AU | Mar 19, 2018 - Mar 24, 2018 | Live Event |
| SANS Boston Spring 2018 | Boston, MAUS | Mar 25, 2018 - Mar 30, 2018 | Live Event |
| SANS 2018 | Orlando, FLUS | Apr 03, 2018 - Apr 10, 2018 | Live Event |
| SANS Abu Dhabi 2018 | Abu Dhabi, AE | Apr 07, 2018 - Apr 12, 2018 | Live Event |
| Pre-RSA® Conference Training | San Francisco, CAUS | Apr 11, 2018 - Apr 16, 2018 | Live Event |
| SANS Zurich 2018 | Zurich, CH | Apr 16, 2018 - Apr 21, 2018 | Live Event |
| SANS London April 2018 | London, GB | Apr 16, 2018 - Apr 21, 2018 | Live Event |
| SANS Baltimore Spring 2018 | Baltimore, MDUS | Apr 21, 2018 - Apr 28, 2018 | Live Event |
| SANS Seattle Spring 2018 | Seattle, WAUS | Apr 23, 2018 - Apr 28, 2018 | Live Event |
| Blue Team Summit & Training 2018 | Louisville, KYUS | Apr 23, 2018 - Apr 30, 2018 | Live Event |
| SANS Riyadh April 2018 | Riyadh, SA | Apr 28, 2018 - May 03, 2018 | Live Event |
| SANS Doha 2018 | Doha, QA | Apr 28, 2018 - May 03, 2018 | Live Event |
| SANS SEC460: Enterprise Threat Beta Two | Crystal City, VAUS | Apr 30, 2018 - May 05, 2018 | Live Event |
| Automotive Cybersecurity Summit & Training 2018 | Chicago, ILUS | May 01, 2018 - May 08, 2018 | Live Event |
| SANS SEC504 in Thai 2018 | Bangkok, TH | May 07, 2018 - May 12, 2018 | Live Event |
| SANS Security West 2018 | San Diego, CAUS | May 11, 2018 - May 18, 2018 | Live Event |
| SANS Melbourne 2018 | Melbourne, AU | May 14, 2018 - May 26, 2018 | Live Event |
| SANS Northern VA Reston Spring 2018 | Reston, VAUS | May 20, 2018 - May 25, 2018 | Live Event |
| SANS New York City Winter 2018 | OnlineNYUS | Feb 26, 2018 - Mar 03, 2018 | Live Event |
| SANS OnDemand | Books & MP3s OnlyUS | Anytime | Self Paced |