Bechtold, Stefan

Working Paper

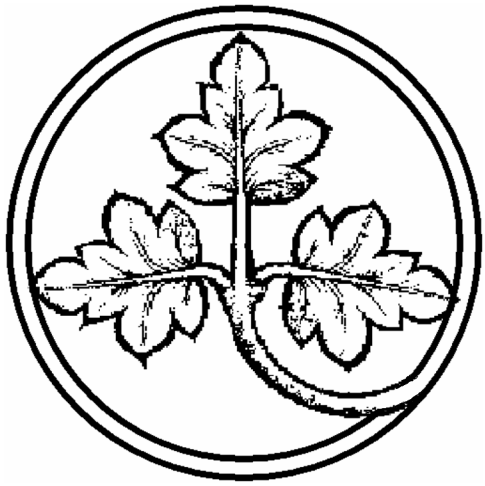
Trusted Computing: rechtliche Probleme einer entstehenden Technologie

Preprints of the Max Planck Institute for Research on Collective Goods, No. 2005,20

Provided in Cooperation with:
Max Planck Institute for Research on Collective Goods

Suggested Citation: Bechtold, Stefan (2005) : Trusted Computing: rechtliche Probleme einer entstehenden Technologie, Preprints of the Max Planck Institute for Research on Collective Goods, No. 2005,20, Max Planck Institute for Research on Collective Goods, Bonn

This Version is available at:
https://hdl.handle.net/10419/26879



Preprints of the
Max Planck Institute
for Research on Collective Goods
Bonn
2005/20

Trusted Computing: rechtliche Probleme einer entstehenden Technologie

Stefan Bechtold

Preprints of the
Max Planck Institute
for Research on Collective Goods

Bonn 2005/20

Trusted Computing: rechtliche Probleme einer entstehenden Technologie

Stefan Bechtold

October 2005

Max Planck Institute for Research on Collective Goods, Kurt-Schumacher-Str. 10, D-53113 Bonn
http://www.mpp-rdg.mpg.de

Trusted Computing: rechtliche Probleme einer entstehenden Technologie[†]

Stefan Bechtold[*]

Abstract

Trusted Computing will das Vertrauen in unsere IT-Umgebung erhöhen. Durch die Einfügung spezieller Hardware-Komponenten in die herkömmliche Computerarchitektur soll die Grundlage für ein Vertrauen in Softwareprogramme gelegt werden. Die zugrunde liegenden Technologien, die sich noch im Entwicklungsstadium befinden, können zu vielfältigen rechtspolitischen und rechtlichen Problemen führen. Dieser Beitrag gibt einen Überblick über spezifische Probleme des Trusted Computing. Nach einer Einführung in die technischen Grundlagen behandelt der Beitrag Fragen des Kartell-, Datenschutz- und Urheberrechts. Viele der aufgeworfenen Fragen haben sich noch nicht derart konkretisiert, dass sie zu aktuellen Rechtsproblemen führen. Es ist das Ziel des Beitrags, Bereiche aufzuzeigen, die zukünftig zu rechtlichen Problemen führen können. Dabei zeigt sich, dass viele der potentiellen Probleme durch ein geschicktes Design der technischen Trusted-Computing-Architektur oder des institutionellen Arrangements, das die Architektur umgibt, gelöst werden können. Es geht also um Fragen im Schnittfeld von Technik und Recht. Es geht auch um Möglichkeiten und Grenzen einer rechtlichen Technikgestaltung.

---

† Dieser Beitrag ist die leicht überarbeitete Fassung eines Manuskripts, das in Computer und Recht 2005, S. 393-404, erschienen ist.

* Dr. *Stefan Bechtold*, J.S.M. (Stanford), ist wissenschaftlicher Mitarbeiter am Max-Planck-Institut zur Erforschung von Gemeinschaftsgütern in Bonn, Habilitand bei Prof. Dr. *Wernhard Möschel*, Juristische Fakultät der Universität Tübingen, und Non-residential Fellow am Center for Internet and Society der Stanford Law School, USA. Er ist Verfasser des „Trusted Computing Blog“ unter *http://cyberlaw.stanford.edu/blogs/bechtold/tcblog.shtml*. Der Verfasser dankt *Boris Brandhoff, Christoph Engel, Hendrik Hakenes* und *Martin Hellwig* für wertvolle Kommentare zu früheren Versionen dieses Beitrags. Er dankt weiterhin *Ross Anderson, Dirk Kuhlmann, Roy Pfitzner, Graeme Proudler, Ahmad-Reza Sadeghi, David Safford, Seth Schoen* und *Hovav Shacham* für äußerst hilfreiche Diskussionen zum Thema „Trusted Computing“.

Inhaltsverzeichnis

I. Einführung

„No issue is currently of greater importance to *Microsoft* and our industry than trustworthy computing.“[1] Diese Aussage von *Microsoft* mag ein Gefühl dafür vermitteln, welche Bedeutung die IT-Industrie dem Thema Trusted Computing inzwischen beimisst. Trusted-Computing-Architekturen könnten in einigen Jahren die IT-Infrastruktur, wie wir sie heute kennen, in bedeutendem Maß verändern. Schon heute geht die Zahl der Computer, die weltweit über Trusted-Computing-Komponenten auf Hardwareebene verfügen, in die zweistelligen Millionen. Marktschätzungen gehen davon aus, dass bis Ende 2007 über 55% aller weltweit installierten PCs und Laptops mit Trusted-Computing-Komponenten ausgestattet sein werden.[2]

In den letzten beiden Jahren hat das Thema Trusted Computing in Deutschland einige Aufmerksamkeit erregt. Das *Bundeswirtschaftsministerium* veranstaltete im Juli 2003 ein zweitägiges Symposium zu technischen, wirtschaftspolitischen und rechtlichen Aspekten des Trusted Computing. Nach weiteren Konsultationen veröffentlichte die *Bundesregierung* – als einzige Regierung weltweit – einige Monate später eine offizielle Stellungnahme.[3] Im *Bundesamt für Sicherheit in der Informationstechnik* wurde eine Projektgruppe gebildet, an der zeitweise bis zu 17 Mitarbeiter teilnahmen.[4] Auf europäischer Ebene veröffentlichte die Art. 29-Datenschutzgruppe – ein Gremium, das nach Art. 29 der europäischen Datenschutzrichtlinie[5] eingerichtet wurde und in dem nationale Datenschutzbeauftragte vertreten sind[6] – ein Dokument, in dem sie die Vereinbarkeit des Trusted Computing mit europäischen Datenschutzbestimmungen problematisierte.[7] In den wirtschafts- und rechtspolitischen Diskussionen um das Trusted Computing nahm Europa und insbesondere Deutschland lange Zeit eine internationale Vorreiterrolle ein. In den USA hat sich erst in jüngerer Zeit auf Betreiben der „*Electronic Frontier Foundation*“[8] eine breitere Debatte über die Implikationen des Trusted Computing entwickelt.

---

1 *Microsoft Corp*., University Relations: Trustworthy Computing, 2004, *http://research.microsoft.com/ur/us/twc*. Auch wenn *Microsoft* von Trustworthy Computing spricht, wird im Folgenden der Begriff Trusted Computing verwendet.

2 *Trusted Computing Group*, Trusted Platform Modules Strengthen User and Platform Authenticity, Januar 2005, S. 5, *https://www.trustedcomputinggroup.org/downloads/whitepapers/TPMs_Strengthen_User_and_Platform_Authenticity_113004_Final.pdf*.

3 Stellungnahme der *Bundesregierung* zu den Sicherheitsinitiativen TCG und NGSCB im Bereich Trusted Computing, 2003, *http://www.bsi.bund.de/sichere_plattformen/trustcomp/stellung/StellungnahmeTCG1_2a.pdf*; s.a. *Sandl*, DuD 2004, 521; *Koenig/Neumann*, DuD 2004, 555, 557.

4 Antwort der *Bundesregierung* auf eine Kleine Anfrage zum Thema „Auswirkungen des ‚Trusted Platform Module' und der Software ‚Palladium'“, BT-Drs. 15/795 vom 7. 4. 2003, S. 2.

5 Richtlinie 95/46/EG des Europäischen Parlaments und des Rates vom 24. 10. 1995 zum Schutz natürlicher Personen bei der Verarbeitung personenbezogener Daten und zum freien Datenverkehr, ABl. EG Nr. L 281 vom 23. 11. 1995, S. 31.

6 Dazu *Di Martino*, Datenschutz im europäischen Recht, Baden-Baden 2005, S. 55 ff.

7 *Artikel 29-Datenschutzgruppe*, Arbeitspapier über vertrauenswürdige Rechnerplattformen und insbesondere die Tätigkeit der *Trusted Computing Group* (*TCG*), WP 86, 23. 1. 2004, *http://europa.eu.int/comm/internal_market/privacy/docs/wpdocs/2004/wp86_de.pdf*.

8 *http://www.eff.org*. Die „*Electronic Frontier Foundation*“ (EFF) ist eine Bürgerrechtsorganisation mit Sitz in San Francisco, die sich mit den Auswirkungen von Computertechnologien auf „civil liberties“ beschäftigt.

Trotz all dieser Debatten haben die spezifischen Möglichkeiten und Probleme des Trusted Computing in der juristischen Fachwelt noch relativ wenig Beachtung gefunden.[9] Dies liegt einerseits daran, dass sich Trusted-Computing-Architekturen noch im Entwicklungsstadium befinden und tatsächliche Rechtsprobleme derzeit nur skizziert werden können. Andererseits sind die technischen Grundlagen derart komplex, dass auch ein gegenüber der Technik aufgeschlossener Jurist viel Zeit benötigt, um sich den Hintergrund anzueignen, der für eine seriöse Debatte über rechtliche Probleme des Trusted Computing sowie – allgemeiner – das Verhältnis zwischen Recht und Technik als Regulierungsmechanismen notwendig ist.

II. Trusted-Computing-Initiativen

Trusted-Computing-Architekturen, deren grundlegende Gedanken in den 1990er Jahren entwickelt wurden,[10] versuchen, durch technische Lösungen das allgemeine Vertrauen in unsere IT-Umgebung zu erhöhen.[11] Nach dem Trusted-Computing-Ansatz kann einer Systemkomponente nur vertraut werden, wenn sie sich stets in der erwarteten Weise hinsichtlich des verfolgten Zwecks verhält. Teilt ein bestimmtes Softwareprogramm einem Anwender mit, was es gerade macht, so soll der Anwender durch Trusted Computing dieser Aussage auch tatsächlich vertrauen können. Um dieses Vertrauen in Softwareprogramme zu ermöglichen, fügen Trusted-Computing-Architekturen spezielle Hardware-Bausteine in die herkömmliche Computerarchitektur ein, die die Grundlage für ein solches Vertrauen legen sollen. Seit einigen Jahren entwickeln mehrere Initiativen unterschiedliche Komponenten für eine umfassende Trusted-Computing-Infrastruktur. Drei Initiativen sind hervorzuheben.

1. Trusted Computing Group (TCG)

Die „*Trusted Computing Group*“ (*TCG*)[12] ist ein Standardisierungsgremium der Computerindustrie, das im Frühjahr 2003 aus der 1999 gegründeten „*Trusted Computing Platform Alliance*“ (*TCPA*) hervorgegangen ist.[13] Die *TCG* zählt inzwischen über 100 Mitglieder. Die wichtigsten Unternehmen der Hardware- und IT-Sicherheitsbranche sind vertreten.[14] Die *TCG* entwickelt zahlreiche Spezifikationen für Trusted-Computing-Plattformen. Die grundlegende Spezifikation, die seit November 2003 in der Version 1.2 vorliegt,[15] spezifiziert eine Hardware-Plattform, die

9 Die weltweit einzigen ausführlichen Beiträge zu Problemen des Trusted Computing, die von Juristen verfasst wurden, stammen von deutschen Autoren: *Bechtold* in Becker u.a. (Hrsg.), Digital Rights Management, Berlin 2003, S. 597, 632 ff., sowie die in Fn. 105 aufgeführten Beiträge von *Koenig/Neumann*.

10 *Arbaugh* u.a. in Proceedings of the IEEE Symposium on Security and Privacy 1997, S. 65.

11 *Pearson* (Hrsg.), Trusted Computing Platforms – TCPA Technology in Context, Upper Saddle River 2003, S. 31, 41. Zu weiteren Informationen über die Ziele des „Trusted Computing“ s. *ebda.*, S. 4 ff.

12 *http://www.trustedcomputinggroup.org*.

13 Zur Entstehung der *TCPA* und den Gründen für die Neugründung der *TCG* s. *Bechtold*, a.a.O. Fn. 9, S. 633.

14 Zu den Mitgliedern zählen u.a. *Hewlett-Packard*, *IBM*, *Intel*, *Microsoft*, *Sony*, *Sun*, *Dell*, *Giesecke & Devrient*, *Infineon*, *Motorola*, *Nokia*, *RSA Security* und *Verisign*. Eine vollständige Mitgliederliste findet sich unter *https://www.trustedcomputinggroup.org/about/members*.

15 *Trusted Computing Group*, TPM Specification Version 1.2,

eine sichere Vertrauensgrundlage für Softwareprozesse schaffen soll. Zu diesem Zweck wird die herkömmliche PC-Architektur durch die Einfügung zweier spezieller Hardware-Bausteine geringfügig modifiziert.[16] Die Spezifikationen der *TCG* sollen in Zukunft nicht nur auf PCs, sondern auch auf Servern und mobilen Endgeräten wie Handys, Musik-Abspielgeräten und PDAs implementiert werden. Daneben entwickelt die *TCG* weitere Spezifikationen im IT-Sicherheitsbereich. So wurde Anfang Mai 2005 die „Trusted Network Connect (TNC)"-Spezifikation veröffentlicht, die sicherheitsrelevante Fragen des Netzwerkzugangs regelt.[17]

2. Next Generation Secure Computing Base (NGSCB)

Die zweite wichtige Initiative im Bereich des Trusted Computing ist das „Next Generation Secure Computing Base (NGSCB)"-Projekt[18] von *Microsoft*. Ursprünglich unter dem Namen „Palladium" bekannt,[19] ist die Zukunft des Projekts seit längerer Zeit unklar. Inzwischen ist bekannt, dass *Microsoft* plant, einige wenige Komponenten von NGSCB in die nächste Version von Microsoft Windows (Codename Longhorn) zu integrieren.[20] Viele andere Komponenten werden erst später Eingang in Microsoft Windows finden.[21] Ein Zeitplan ist nicht bekannt. NGSCB befindet sich noch im Entwicklungsstadium. Von *Microsoft* sind bis heute nur spärlich Informationen über das Projekt erhältlich.[22] Obwohl dies eine deutliche Vereinfachung ist,[23] kann man sich NGSCB als den Komplex von Systemsoftware vorstellen, der auf TCG-Hardware aufbaut und deren Funktionalität nutzt.

---

https://www.trustedcomputinggroup.org/downloads/specifications. Die besten technischen Einführungen finden sich in *Pearson* (Hrsg.), a.a.O. Fn. 11, und in *Trusted Computing Group*, TCG Specification Architecture Overview, 28. 4. 2004, *https://www.trustedcomputinggroup.org/downloads/TCG_1_0_Architecture_Overview.pdf*.

16 *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 5.

17 Grob gesagt erweitert die TNC-Spezifikation herkömmliche Netzwerkzugangstechnologien um die Möglichkeit, Integritätsmetriken zu überprüfen und Plattformen zu identifizieren. Dabei baut TNC auf den Funktionalitäten der allgemeinen TCG-Spezifikation auf. Näher zu TNC s. *Trusted Computing Group*, TNC Architecture for Interoperability, 3. 5. 2005, S. 11 ff., *https://www.trustedcomputinggroup.org/downloads/specifications/TNC_Architecture_v1_0_r4.pdf*.

18 Das Akronym „NGSCB" wird von Insidern regelmäßig als „enscub" ausgesprochen.

19 Dazu *Bechtold*, a.a.O. Fn. 9, S. 638.

20 Dabei geht es insbesondere um die Funktionalität des sog. „Secure Startup"; s. dazu *Microsoft Corp.*, Secure Startup – Full Volume Encryption, 21. 4. 2005, *http://www.microsoft.com/whdc/system/platform/pcdesign/secure-start_tech.mspx*; *Microsoft Corp.*, Trusted Platform Module Services in Windows Longhorn, 25. 4. 2005, *http://www.microsoft.com/whdc/system/platform/pcdesign/TPM_secure.mspx*. Allgemein zum „Secure Booting" s. *Bechtold*, a.a.O. Fn. 9, S. 635 f.

21 Die aktuellsten und verlässlichsten Informationen zum Gesamtprojekt NGSCB finden sich derzeit in *Abadi/Wobber* in Frutos/Escrig (Hrsg.), Formal Techniques for Networked and Distributed Systems 2004, Berlin 2004, S. 1 ff., und in einem 2004 in Köln gehaltenen Vortrag von *John Manferdelli*, *http://research.microsoft.com/collaboration/university/europe/Events/Workshop/Sec2004/DVDPreview/Claims_files/intro.htm*.

22 S. aber die Nachweise in Fn. 21.

23 NGSCB behandelt nicht nur Software-, sondern auch Hardware-Fragen. Grundsätzlich kann NGSCB auch auf einer anderen Trusted-Computing-Hardware-Architektur als TCG aufbauen. NGSCB erfordert auf Hardware-Seite bestimmte Funktionalitäten (im Bereich der Abschottung von Speicherbereichen), die derzeitige Prozessorarchitekturen ohne LaGrande nicht unterstützen. S. a. *Bechtold*, a.a.O. Fn. 9, S. 638 Fn. 1988.

3. LaGrande

Die dritte wichtige Initiative im Bereich des Trusted Computing ist das „LaGrande"-Projekt von *Intel*.[24] Durch dieses Projekt soll die Architektur zukünftiger Prozessoren und anderer Komponenten derart modifiziert werden, dass mit ihnen Trusted-Computing-Umgebungen betrieben werden können. Insbesondere soll die physikalische Abschottung von Speicherbereichen („Memory Curtaining") ermöglicht werden.[25]

4. Ausgewählte technische Eigenschaften

Im Rahmen dieses Beitrags ist es unmöglich, einen detaillierten technischen Überblick über diese drei Trusted-Computing-Initiativen zu geben. Vielmehr sollen vier Funktionen des Trusted Computing erwähnt werden, die durch die Initiativen ermöglicht werden und aus rechtspolitischer Perspektive von besonderem Interesse sind:

1. *„Remote Attestation"*: Mit Hilfe des Trusted Computing ist es möglich, vertrauenswürdige Informationen über den Systemzustand eines anderen Rechner in einem Netzwerk zu erhalten.[26] Bevor der Rechner A mit Rechner B Daten austauscht, kann der Rechner A vertrauenswürdige Informationen darüber erhalten, ob sich der Rechner B in einem sicheren Systemzustand befindet, der für den Rechner A keine Gefahr darstellt. In diesem Zusammenhang meint der Begriff „vertrauenswürdig", dass Rechner A sicher sein kann, dass die Informationen über den Systemzustand des Rechners B, die er erhält, den wahren Systemzustand dieses Rechners wiedergeben.

2. *„System Compartmentalization"*: Mit Hilfe des Trusted Computing ist es möglich, Speicherbereiche für Softwareprogramme von der Hardwareebene an gegeneinander abzuschotten. Dadurch kann auf einem Computer das Programm A (z.B. ein trojanisches Pferd) nicht mehr den Speicherbereich des Programms B (z.B. eine Online-Banking-Anwendung) auslesen. Auch kann das Betriebssystem nicht die Daten eines Programms auslesen, das unter diesem Betriebssystem läuft. Auf Hardwareebene wird das sog. „Memory Curtaining" durch LaGrande in *Intels* nächste Prozessorgeneration integriert. Auf Betriebssystemebene ermöglicht NGSCB eine ähnliche Art der Abschottung von Speicherbereichen (sog. „System Compartmentalization").[27]

---

24 AMD verfolgt mit seinen Projekten „Presidio" und „Pacifica" ähnliche Ziele. S. dazu *Strongin*, Information Security Technical Report (2005) 10, 120.

25 Näher s. *Intel Corp.*, LaGrande Technology Architectural Overview, September 2003, *http://www.intel.com/technology/security/downloads/LT_Arch_Overview.pdf*.

26 Diese Eigenschaft wird durch TCG ermöglicht.

27 Nach der neuen Architektur von NGSCB soll dabei ein sog. „Inner Nexus" (ein relativ kleiner Betriebssystem-Kernel) bestimmte sicherheitsrelevante Funktionen anbieten. Auf diesem „Inner Nexus" können dann andere Betriebssysteme laufen. Neben dem herkömmlichen Windows kann dies auch ein „Outer Nexus" sein, der auf die spezifischen Sicherheitseigenschaften des „Inner Nexus" zugreifen kann; s. *Manferdelli*, a.a.O. Fn. 21.

3. *„Sealed Storage“*: Durch diese Funktionalität ist es möglich, Daten kryptographisch an bestimmte Systeme oder auch Systemzustände zu binden.[28] Dadurch können die Daten nur von einem bestimmten Computer entschlüsselt werden. Oder sie können von diesem Computer sogar nur entschlüsselt werden, wenn auf dem Computer bestimmte Hardware- oder Softwarekomponenten installiert sind, der Computer sich also in einem festgelegten Systemzustand befindet.

4. *„Secure Input/Output“*: Trusted-Computing-Architekturen bieten einen sicheren Kommunikationsweg zwischen der Tastatur, der Maus und dem Grafiksystem eines Computers auf der einen Seite und dem Prozessor auf der anderen Seite.[29] Durch die eingesetzte Verschlüsselung ist es einem Angreifer unmöglich, die Kommunikation zwischen diesen Komponenten abzuhören.

III. Rechtspolitische und rechtliche Probleme

In der rechtspolitischen Debatte über das Trusted Computing darf man nicht der Versuchung erliegen, die dargestellten Initiativen und Technologien unvorsichtig in einen Topf zu werfen. Obwohl Überlappungen zwischen TCG, NGSCB und LaGrande bestehen, ermöglichen die drei Initiativen auch unterschiedliche Funktionalitäten. Sie bauen aber teilweise aufeinander auf, so dass erst eine Gesamtschau ein Bild dessen ergibt, was Trusted-Computing-Architekturen ermöglichen. Im Folgenden sind Generalisierungen unvermeidbar, werden aber so weit wie möglich minimiert.

Auch muss bei der Debatte unterschieden werden, ob Trusted-Computing-Architekturen im Unternehmens- oder im Haushaltssektor eingesetzt werden. Viele der im Folgenden dargestellten Probleme tauchen entweder ausschließlich oder jedenfalls verstärkt im Haushaltssektor auf. Trusted-Computing-Architekturen sind zunächst für den Unternehmenssektor konzipiert und werden dort auch als erstes Verbreitung finden. Zunehmend werden Trusted-Computing-Architekturen jedoch auch im Haushaltssektor Fuß fassen. Dieser Beitrag wird zeigen, dass es sich schon jetzt lohnt, sich mit den Problemen des Trusted Computing zu befassen.

1. „Attestation“

a) „Remote Attestation“

Die TCG-Spezifikation ermöglicht durch die sog. „Remote Attestation“, vertrauenswürdige Informationen über den Systemzustand eines anderen Rechners in einem Netzwerk zu erhalten.[30]

---

28 Diese Eigenschaft wird durch TCG ermöglicht.

29 Diese Eigenschaft wird durch NGSCB und LaGrande ermöglicht.

30 Diese Funktionalität kann auch dazu verwendet werden, um Informationen über den Zustand des eigenen Rechners zu erhalten. Dennoch ist einer der Kritikpunkte an Trusted-Computing-Architekturen, dass sie sich zu stark mit der Frage beschäftigen, wie Vertrauen zwischen Computern in einem Netzwerk hergestellt wer-

Um die wettbewerbsrechtlichen Probleme dieser Funktionalität zu veranschaulichen, soll zunächst im Überblick erläutert werden, wie die „Remote Attestation“ funktioniert.

aa) Funktionsweise

Will der Rechner A (der „Remote Challenger“) Informationen darüber erhalten, ob sich der Rechner B in einem sicheren Zustand befindet, sendet er eine entsprechende Anfrage an den Rechner B (s. Abbildung 1).

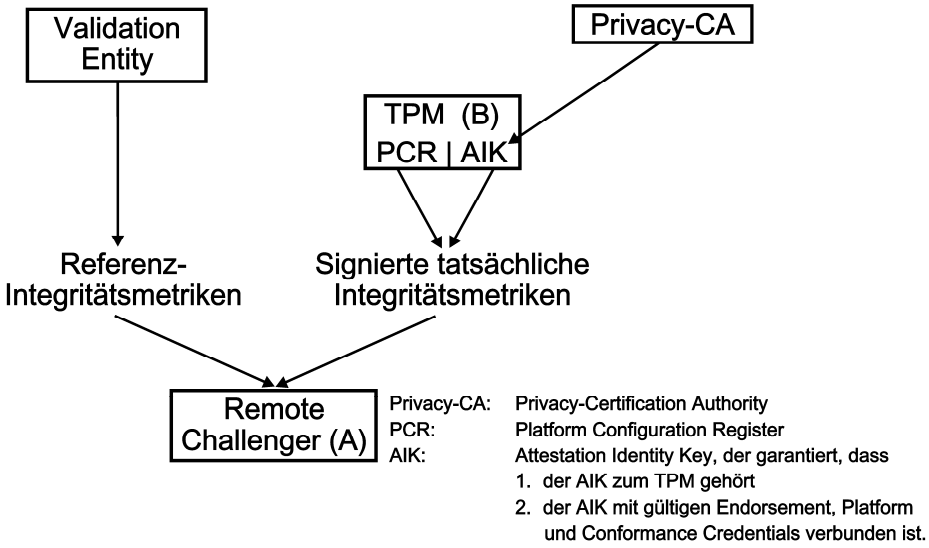


Abbildung 1: „Remote Attestation“

Auf dem Motherboard des Rechners B befindet sich ein Hardwarebaustein namens „Trusted Platform Module“ (TPM).[31] In dem TPM sind in sog. „Platform Configuration Registers“ (PCRs) Informationen über den Systemzustand des Rechners B (sog. Integritätsmetriken) gespeichert.[32] Das TPM ist gegen Softwareangriffe vollständig und gegen Hardwareangriffe relativ gut, wenn auch nicht vollständig, geschützt. Aufgrund dieses Schutzes kann Rechner A mit ziemlich hoher Wahrscheinlichkeit darauf vertrauen, dass die Integritätsmetriken, die in den PCRs des Rechners B gespeichert sind, den wahren Systemzustand dieses Rechners wiedergeben.

Nach einer Anfrage des Rechners A versieht der Rechner B die Integritätsmetriken, die in seinen PCRs gespeichert sind, mit einer digitalen Signatur[33] und übermittelt das Resultat an den Rech-

---

den kann, anstatt zunächst die Frage zu lösen, wie Vertrauen zwischen einem Nutzer und seinem lokalen Computer hergestellt werden kann.

31 Zu den Funktionalitäten, die ein TPM bietet, s. Fn. 63.

32 Zur Frage, wie diese Integritätsmetriken zustande kommen, s. bei Fn. 61 ff.

33 Das TPM des Rechners B signiert die Metriken mit dem privaten Schlüssel des „Attestation Identity Key“ (AIK). Der AIK wurde dem TPM von einer „Privacy Certification Authority“ (Privacy-CA) bestätigt. Dies ist zumindest eine der Möglichkeiten im Rahmen der TCG-Spezifikation. Zur „Direct Anonymous Attestation“ (DAA), die auf die Einschaltung einer Privacy-CA verzichtet, s. bei Fn. 118 ff.

ner A.[34] Der Rechner A kann diese Werte mit Referenz-Integritätsmetriken vergleichen. Diese Referenzwerte, die der Rechner A aus unterschiedlichen Quellen (z.B. einer sog. „Validation Entity“) erhalten kann, teilen dem Rechner A mit, welchen Inhalt die Integritätsmetriken des Rechners B haben müssen, wenn sich der Rechner B tatsächlich in dem Hard- und/oder Softwarezustand befindet, den er vorgibt. Durch einen Vergleich der Referenz-Integritätsmetriken mit den tatsächlichen Metriken kann der Rechner A letztlich feststellen, welche Hardware- und Softwarekomponenten auf dem Rechner B installiert sind.[35] Dadurch kann der Rechner A die Sicherheit des Rechners B einschätzen. Dies macht die „Remote Attestation“ aus technischer Perspektive sehr interessant: Bevor der Rechner A mit dem Rechner B Daten austauscht, kann er feststellen, ob dieser Datenaustausch ein Sicherheitsrisiko für ihn selbst darstellt.

bb) Wettbewerbspolitisches Problem

Aus wettbewerbspolitischer Perspektive wirft die „Remote Attestation“ Probleme auf. Wie erwähnt, ermöglicht es die „Remote Attestation“ dem Rechner A, die Sicherheit des Rechners B einzuschätzen. Dabei steht es in der Macht des Rechners A zu entscheiden, was für ihn „Sicherheit“ bedeutet.[36] Dies kann zu Problemen führen. Ein Beispiel mag dies verdeutlichen.

Ein Unternehmen mit marktbeherrschender Stellung, das einen Dienst über das Internet anbietet, könnte die Funktionalität der „Remote Attestation“ verwenden, um nur noch mit bestimmten Client-Applikationen zu interoperieren. Wenn *Microsoft* beispielsweise einen Dienst über das Internet anbietet, könnte das Unternehmen mit Hilfe der „Remote Attestation“ sicherstellen, dass nur Nutzer den Dienst benützen können, die den Microsoft Internet Explorer als Webbrowser verwenden.

Ähnliche Beispiele lassen sich nahezu endlos bilden.[37] Abstrakt betrachtet bedeutet dies, dass ein marktbeherrschendes Unternehmen die Funktionalität der „Remote Attestation“ verwenden

---

34 Diese Übermittlung wird nicht vom TPM selbst, sondern von darüber liegenden Softwareschichten ausgeführt. Das TPM ist ein passives Bauteil; *Brandl/Rosteck*, DuD 2004, 530, 531.

35 Dies ist eine Vereinfachung der tatsächlichen Lage. Das Vertrauen des Rechners A in die übermittelten Integritätsmetriken des Rechners B hängt davon ab, ob der Rechner A den einzelnen Komponenten der Vertrauenskette vertraut, die die Integritätsmetriken erstellen (CRTM, BIOS, Betriebssystem-Loader, Betriebssystem, Anwendungssoftware). Wird die Vertrauenskette unterbrochen, weiß Rechner A ab dem Bruch der Vertrauenskette nicht mehr, ob die übermittelten Integritätsmetriken der Realität entsprechen. Er weiß jedoch, dass die Vertrauenskette bis zu diesem Punkt vertrauenswürdig ist und dass die Integritätsmetriken danach gefälscht sein können. Dieses Wissen um ein mögliches Sicherheitsrisiko reicht für den Rechner A in der Regel aus. Näher zum Konzept der Vertrauenskette s. bei Fn. 61 ff.

36 S. *Bechtold*, a.a.O. Fn. 9, S. 642; *Kuhlmann*, DuD 2004, 545, 547. Zu einem ähnlichen Problem im Rahmen des Vergleichs des U.S.-amerikanischen Kartellrechtsprozesses gegen *Microsoft* im November 2002 s. *Bechtold*, a.a.O. Fn. 9, S. 627 f.

37 *Schoen*, Trusted Computing: Promise and Risk, 2003, S. 8 f., *http://www.eff.org/Infrastructure/trusted_computing/20031001_tc.pdf*. Zwar ist es Unternehmen auch schon bisher möglich, die Interoperabilität zwischen Computerprogrammen bis zu einem gewissen Maß zu kontrollieren. Schon heute senden Webbrowser eine Identifikation an den Webserver, wodurch der Webserver feststellen kann, welchen speziellen Webbrowser der Nutzer verwendet. Jedoch ist es problemlos möglich, diese Identifikation zu verändern (zum sog. „Masquerading“ s. *http://www.ericgiguere.com/articles/masquerading-your-browser.html*). Eine solche Veränderung ist in einer Trusted-Computing-Architektur nicht mehr möglich. Denn die Integritätsmetriken werden in den PCRs in einer manipulationsresistenten Umgebung abgespeichert. Der Nutzer hat keinen Einfluss auf Erhebung und Speicherung der PCR-Werte.

könnte, um seine marktbeherrschende Stellung auf einem Markt auf einen anderen über- oder untergeordneten Markt auszudehnen. Auch kann Innovation auf dem über- oder untergeordneten Markt durch unabhängige Softwareentwickler beeinträchtigt werden.[38] Diese „Foreclosure"-Problematik wurde relativ früh erkannt. Inzwischen wurde eine Vielzahl von Lösungsvorschlägen erarbeitet, die sich in drei Kategorien einordnen lassen.

cc) Lösungsansätze

(1) Lösung durch Technik

Die erste Gruppe der Lösungsvorschläge versucht das Problem auf technischer Ebene zu lösen. Der bekannteste Vorschlag ist der sog. „Owner Override", der von *Seth Schoen* (Techniker bei der „*Electronic Frontier Foundation*" in San Francisco) vorgeschlagen wurde.[39] Nach diesem Vorschlag sollte die TCG-Spezifikation derart modifiziert werden, dass es dem Nutzer des Rechners B erlaubt sein sollte, „falsche" Integritätsmetriken an den Rechner A zu senden. Durch diese Möglichkeit zur Lüge könnte der Nutzer des Rechners B selbst entscheiden, ob er dem Rechner A seinen wahren Systemzustand mitteilt oder nicht. Dadurch könnte der Nutzer des Rechners B ein Produkt verwenden, ohne dem Rechner A mitteilen zu müssen, um welches genaue Produkt es sich handelt. Da der Rechner A nicht feststellen könnte, ob die Integritätsmetriken, die er vom Rechner B erhalten hat, der Realität entsprechen oder nicht, könnte der Rechner A diese Information auch nicht mehr zur Diskriminierung bestimmter Hard- oder Software verwenden, die auf dem Rechner B installiert ist. Dieser Vorschlag ist kritisiert worden, weil er die Funktionalität von Trusted-Computing-Architekturen einschränkt.[40]

Nach anderen Vorschlägen sollte der Rechner B nicht die *Identität* eines Softwareprogramms an den Rechner A übermitteln. Vielmehr sollte er nur sicherheitsrelevante *Eigenschaften* oder das *Verhalten* des Programms übermitteln („Property" und „Semantic Remote Attestation").[41] Dadurch würde der Rechner A nur erfahren, ob das Softwareprogramm auf Rechner B bestimmte Sicherheitseigenschaften erfüllt. Er würde nicht erfahren, um welches spezielle Programm es sich handelt.[42] Eine andere Lösung des Problems könnte darin liegen, die Integritätsmetriken des

---

38 Dazu *Schoen*, Information Security Technical Report (2005) 10, 105, 110. Zur allgemeinen Bedeutung dezentraler Innovationssysteme s. *von Hippel*, Democratizing Innovation, Cambridge 2005.

39 *Schoen*, a.a.O. Fn. 37, S. 12 ff. Ein modifizierter „Owner Override" wird in *Kursawe/Stüble*, Improving End-user Security and Trustworthiness of TCG-Platforms, 29. 9. 2003, *http://www-krypt.cs.uni-sb.de/download/papers/KurStu2003.pdf*, vorgeschlagen.

40 *Bechtold*, *http://cyberlaw.stanford.edu/blogs/bechtold/archives/001607.shtml*; *Kursawe*, DuD 2004, 566.

41 Dies ist stark vereinfacht. Zu den Einzelheiten der „Property Attestation" s. *Sadeghi/Stüble*, Property-based Attestation for Computing Platforms, 2004, *http://www.prosec.rub.de/Publications/SadStu2004.pdf*; *Poritz* u.a., Property Attestation – Scalable and Privacy-friendly Security Assessment of Peer Computers, IBM Research Report RZ 3548, Oktober 2004, *http://domino.watson.ibm.com/library/cyberdig.nsf/Home*. Zu den Einzelheiten der „Semantic Remote Attestation" s. *Haldar* u.a., Proceedings of the 3rd Virtual Machine Research and Technology Symposium 2004, S. 29 ff., erhältlich unter *http://gandalf.ics.uci.edu/~haldar/pubs/trustedvm-tr.pdf*; s. weiterhin *Kursawe*, DuD 2004, 566.

42 Bei diesen Vorschlägen ist fraglich, ob sie zur jetzigen Zeit schon genügend ausgereift sind, um in TCG integriert werden zu können. Dazu *Bechtold*, *http://cyberlaw.stanford.edu/blogs/bechtold/archives/002805.shtml* und *http://cyberlaw.stanford.edu/blogs/bechtold/archives/002809.shtml*. Je kleiner die Anzahl der konkurrierenden Programme ist, desto weniger werden diese Vorschläge helfen.

Rechners B nicht auf ein Mal, sondern in einem gestuften Prozess zu übertragen, wobei der Rechner B mitbestimmen kann, ob die nächste Übertragsstufe initiiert werden soll oder nicht.[43] Wieder andere meinen, dass sich das Problem verringern könnte, wenn im Rahmen der „Remote Attestation" nicht Informationen über den Zustand des gesamten Rechners B übertragen werden, sondern nur über relativ kleine abgeschottete Teile des Rechners[44] – eine Überlegung, die im Zuge der zukünftigen Möglichkeit, Speicherbereiche auf Hard- und Softwarebene sicher voneinander abzuschotten,[45] Einiges für sich hat. Schließlich weisen andere darauf hin, dass sich das Problem im Unternehmensumfeld nicht stellt, wenn das Unternehmen die gesamte Trusted-Computing-Umgebung einschließlich der „Validation Entities" betreibt.[46]

(2) Lösung durch Recht

Die zweite Gruppe der Lösungsvorschläge versucht, das Problem auf rechtlicher Ebene zu lösen, wenn es denn keine überzeugende technische Lösung gibt. Nach einer Ansicht sollte dem Problem durch Instrumentarien der kartellrechtlichen Missbrauchsaufsicht begegnet werden.[47] Auch dieser Ansatz überzeugt nicht vollständig, da die Effektivität kartellrechtlicher Kontrolle in Hochtechnologiemärkten zumindest zweifelhaft ist. Eine andere Lösung wäre die rechtliche oder zumindest faktische Trennung verschiedener Unternehmensbereiche, wie dies im kartellrechtlichen Microsoft-Verfahren in den USA erwogen wurde. Nach einer dritten Ansicht sollten die Unternehmen, die Trusted-Computing-Technologien als Patent oder Know-how lizenzieren, in die Lizenzverträge Klauseln integrieren, die den Missbrauch der lizenzierten Technologien im dargestellten Sinne verbieten. Auch wenn dieses Regelungsinstrument nicht vollständig unbekannt ist,[48] würde es sinnvollerweise die Errichtung eines bisher nicht existierenden Patentpools für Trusted-Computing-Technologien voraussetzen.[49]

(3) Lösung durch Richtlinien

Die dritte Gruppe von Lösungsvorschlägen versucht das Problem durch rechtlich nicht bindende Standardisierungs-Richtlinien zu lösen. Diesen Weg hat die *TCG* beschritten. In einem im Juni 2005 veröffentlichten Dokument stellt das „Best Practices Committee" der *TCG* Richtlinien auf, wonach der Einsatz von TCG-Technologie zu den oben dargestellten Zwecken einen Missbrauch

---

43 In diese Richtung *Trusted Computing Group*, a.a.O. Fn. 17, S. 24; s. a. *Trusted Computing Group*, TNC IF-IMV, 3. 5. 2005, S. 56, *https://www.trustedcomputinggroup.org/downloads/specifications/TNC_IFIMV_v1_0_r3.pdf*.

44 Dazu *Marchesini* u.a., Open-Source Applications of TCPA Hardware, 2004, S. 6 f., *http://www.cs.dartmouth.edu/~carlo/research/bearapps/bearapps.pdf*; *Smith*, Trusted Computing Platforms, New York 2005, S. 193 f.

45 Dazu bei Fn. 29.

46 Davon scheint die TCG im Rahmen von TNC auszugehen, s. *Trusted Computing Group*, a.a.O. Fn. 17, S. 27 f.

47 *Bechtold*, a.a.O. Fn. 9, S. 642; angedeutet auch in *Bundesregierung*, a.a.O. Fn. 3, S. 7.

48 Die Entwickler von „Digital Rights Management (DRM)"-Technologien verwenden Patent- und Know-how-Lizenzverträge in sehr umfangreichen Maß, um festzulegen, zu welchen Zwecken die lizenzierte Technologie eingesetzt werden darf; s. *Bechtold*, Vom Urheber- zum Informationsrecht, München 2002, S. 178 ff.

49 Dazu bei Fn. 104.

der Technologie darstellt.[50] Der größte Vorteil dieses Lösungsvorschlags ist, dass sich die TCG-Unternehmen auf diesen Vorschlag einigen konnten. Der größte Nachteil ist, dass er weder über rechtliche noch technische Durchsetzungsmechanismen verfügt.[51]

(4) Vergleich der Lösungsansätze

Im Rahmen dieses Beitrags ist es unmöglich, die Vor- und Nachteile dieser Vorschläge im Detail zu untersuchen. Es muss der Hinweis genügen, dass zumindest derzeit keine Patentlösung des Problems in Sicht ist. In der Zukunft muss es darum gehen, die Vor- und Nachteile der einzelnen Vorschläge gegeneinander abzuwägen und eine Lösung zu finden, die im Vergleich am Vorzugswürdigsten scheint. Es wird um die Frage gehen, bis zu welchem Maß Dienstebetreiber Sicherheitsregeln aufstellen dürfen, die erfüllt sein müssen, bevor ihr Dienst benutzt werden darf. Es wird um Fragen der Vergleichbarkeit unterschiedlicher Sicherheits-Regime gehen. Es wird um die Frage gehen, wann hinter dem Schlagwort der Sicherheit tatsächlich wettbewerbsbeschränkende Absichten verborgen werden.[52] Zur Beantwortung solcher Fragen ist eine enge Zusammenarbeit zwischen Technikern und Juristen notwendig. Die Diskussionen um die „Remote Attestation" sind ein gutes Anwendungsbeispiel für das Ineinandergreifen und die teilweise Substituierbarkeit von Technik und Recht.[53]

b) „Validation Entities"

aa) Allgemeines

Die „Remote Attestation" baut auf dem Vergleich von tatsächlichen Integritätsmetriken mit Referenz-Integritätsmetriken auf. Während die tatsächlichen Metriken von dem Rechner B geliefert werden, der untersucht wird, muss der Rechner A die Referenz-Metriken von einer anderen Stelle beziehen. In der TCG-Architektur werden die Referenz-Metriken von sog. „Validation Entities" geliefert.

Es ist eine offene Frage, wer diese „Validation Entities" betreiben soll. Ursprünglich war die Befürchtung geäußert worden, dass in Trusted-Computing-Infrastrukturen nur noch Hard- und Software betrieben werden könne, die von einer zentralen Instanz zertifiziert wurde. Dies hätte der zentralen Instanz eine breite Palette von Missbrauchsmöglichkeiten eröffnet. So wurde unter anderem befürchtet, dass ein Unternehmen wie *Microsoft* oder *Verisign* eine solche Instanz betreiben könnte und den Einsatz von Anwendungssoftware behindern könnte, die von Wettbewerbern entwickelt wurde, indem Zertifikate nicht, verspätet oder nur überteuert erteilt würden.

---

50 *TCG Best Practices Committee*, Design, Implementation, and Usage Principles for TPM-Based Platforms, Version 1.0, Mai 2005, *https://www.trustedcomputinggroup.org/downloads/bestpractices/Best_Practices_Principles_Document_v1.0.pdf*. Ein Kommentar zu diesem Dokument findet sich bei *Schoen*, EFF Comments on TCG Design, Implementation and Usage Principles 0.95, Oktober 2004, *http://www.eff.org/Infrastructure/trusted_computing/20041004_eff_comments_tcg_principles.pdf*.

51 Darauf weist das Dokument der *TCG* selbst hin, s. *TCG Best Practices Committee*, a.a.O. Fn. 50, S. 3, 13; s. a. *Sandl*, DuD 2004, 521, 524.

52 *Bechtold*, a.a.O. Fn. 9, S. 642.

53 Dazu *Lessig*, Code and Other Laws of Cyberspace, New York 1999.

Diese Befürchtung ist zumindest in der Theorie unbegründet.[54] Nach der TCG-Spezifikation und wohl auch der NGSCB-Architektur steht es jedem frei, eine „Validation Entity" zu betreiben.[55] Der Begriff der „Validation Entity" ist funktional zu verstehen. In einer Trusted-Computing-Infrastruktur ist es möglich, dass der Hersteller einer Hardwarekomponente oder eines Computerprogramms selbst die Referenz-Integritätsmetriken zur Verfügung stellt, die von Rechner A im Rahmen einer „Remote Attestation" benötigt werden.[56] Solange die Nutzer in einer Trusted-Computing-Infrastruktur die Referenzwerte dieses Herstellers für vertrauenswürdig halten, kann der Hersteller seine eigene „Validation Entity" sein.

bb) Open-Source-Software

Auch wenn die Gefahr einer Abhängigkeit von externen „Validation Entities" im Allgemeinen nicht groß sein mag, birgt die Architektur besondere Probleme für Open-Source-Programme.[57] Zwar kann in einer Trusted-Computing-Umgebung der Programmierer eines Open-Source-Programms seine eigene „Validation Entity" sein, die selbst Referenz-Integritätsmetriken erstellt. Diese Erstellung kann jedoch kostspielig sein. Auch führt jede Änderung eines Programms führt dazu, dass dessen Referenz-Integritätsmetriken geändert werden müssen.[58] Open-Source-Programme leben von ständiger Veränderung und Weiterentwicklung. Wer soll diese Kosten bei Open-Source-Programmen tragen? Selbst wenn der Programmierer eines Open-Source-Programms die Referenz-Integritätsmetriken selbst erstellt, also seine eigene „Validation Entity" betreibt, ist weiterhin fraglich, wie diese „Validation Entity" das Vertrauen der Nutzer gewinnen will. Auch ist fraglich, ob Netzwerkeffekte nicht zu einer Konzentration in wenige „Validation Entities" führen werden.[59] Dann erscheint eine dezentrale Koexistenz vieler „Validation Entities" unrealistisch. Aus all diesen Gründen ist es derzeit eine völlig ungeklärte Frage, wie Open-Source-Programme mit einer zukünftigen Trusted-Computing-Infrastruktur interoperieren sollen.

Der Verdacht, dass die Unternehmen, die hinter Trusted Computing stehen, diese Technologie einsetzen wollen, um die Open-Source-Bewegung zu schädigen, ist in weiten Teilen nicht zutreffend. Unternehmen wie *IBM* und *HP* arbeiten seit langem an der Integration von Open-Source-Programmen in Trusted-Computing-Architekturen.[60] Selbst *Microsoft* hat sich schon zu diesem Thema geäußert.

---

54 Sie resultiert u.a. aus der unterschiedlichen Verwendung des Begriffs „Zertifikat". Es muss zwischen Zertifikaten im engeren Sinne und Referenz-Integritätsmetriken unterschieden werden; s. *Brandl/Rosteck*, DuD 2004, 529, 535.

55 *Trusted Computing Group*, TCG Specification Architecture Overview, a.a.O. Fn. 15, S. 12, 45 f.

56 Tatsächlich ist dies der Regelfall, der *Microsoft* im Rahmen von NGSCB vorschwebt; s. *Abadi/Wobber*, a.a.O. Fn. 21, S. 5. Zu TCG s. *Trusted Computing Group*, TCG Specification Architecture Overview, a.a.O. Fn. 15, S. 12.

57 S. a. *Kuhlmann* in Koenig/Neumann/Katzschmann (Hrsg.), Trusted Computing, Heidelberg 2003, S. 163, 173 f.; *Bechtold*, a.a.O. Fn. 9, S. 643 f.

58 *Bechtold*, a.a.O. Fn. 9, S. 643; *Kuhlmann* in Wright (Hrsg.), Financial Cryptography 2003, Berlin 2003, S. 255, 267. Vereinfacht handelt es sich bei Integritätsmetriken um einen Hash-Wert. Das dargestellte Problem stellt sich auch bei proprietären Softwareprogrammen, z.B. nach Updates und Patches.

59 Da es um Vertrauen geht, wird die Marktdurchsetzung für „Validation Entities" entscheidend sein.

60 S. z.B. *Sailer/van Doorn/Ward*, DuD 2004, 539, *http://sourceforge.net/projects/tpmdd* und

c) Unterstützende Infrastruktur

Die Probleme der „Remote Attestation“ sind Symptome eines tiefer liegenden Problems. Bisher haben sich Trusted-Computing-Initiativen hauptsächlich mit der Spezifizierung der technischen Komponenten eines Computers beschäftigt, der Trusted-Computing-Funktionalitäten besitzt. Dagegen blieben Fragen, wie die Infrastruktur aussehen muss, die die Kommunikation zwischen solchen Computern ermöglicht, eher am Rande des Interesses. Viele der technischen Eigenschaften, die Trusted Computing bietet, sind für sich genommen aus rechtspolitischer und rechtlicher Perspektive harmlos. Jedoch können sie von darüber liegenden Softwarekomponenten oder der umgebenden Infrastruktur in einer Weise eingesetzt werden, die problematisch sein kann.[61] Daher ist es wichtig, diese Infrastruktur näher zu untersuchen. Zur Erläuterung soll auf einen zentralen Punkt von Trusted-Computing-Architekturen eingegangen werden: die Entstehung einer sog. Vertrauenskette.

aa) Entstehung einer Vertrauenskette

Das Ziel von Trusted-Computing-Architekturen ist, eine sichere Vertrauensgrundlage für Softwareprogramme zu schaffen. Die *TCG* versucht, dies durch die Integration zweier sog. „Roots of Trust“ in die PC-Architektur zu erreichen: den „Core Root of Trust for Measuring Integrity Metrics“ (CRTM) und das „Trusted Platform Module“ (TPM).[62] Das TPM ist ein Chip, das sich zwar vom Hauptprozessor eines PC unterscheidet, aber sicher mit dem PC-Motherboard verbunden ist.[63] Es muss gegen alle Softwareangriffe und gegen eine definierte Gruppe von Hardwareangriffen resistent sein.[64] Das CRTM, das nach der TCG-Spezifikation nicht manipulationsresistent sein muss, wird typischerweise als Teil des PC-BIOS implementiert.[65]

Beide Komponenten werden als „Roots of Trust“ bezeichnet, weil sie die einzigen Komponenten in einer Trusted-Computing-Plattform sind, denen implizit vertraut werden muss.[66] Die grundsätzliche Idee des Trusted Computing ist, das Vertrauen Schritt für Schritt von diesen beiden Komponenten zu anderen Komponenten der Plattform auszudehnen.[67] In einem typischen Boot-Vorgang eines PCs läuft dieser Aufbau einer sog. Vertrauenskette („Chain of Trust“, auch „Transitive Trust“ genannt) wie folgt ab (s. Abbildung 2):

---

http://www.research.ibm.com/secure_systems_department/projects/tcglinux. Zur weitergehenden Vision eines „Open Trusted Computing“ s. *Kuhlmann*, DuD 2004, 545. S. a. *http://enforcer.sourceforge.net*.

61 *Felten*, IEEE Security & Privacy, Mai/Juni 2003, S. 60, 61; *Anderson* in: Camp/Lewis (Hrsg.), Economics of Information Security, Boston 2004, S. 35, 39 f.; *Kuhlmann*, a.a.O. Fn. 57, S. 166.

62 Tatsächlich besteht das TPM aus dem „Root of Trust for Storing Integrity Metrics“ (RTS) und dem „Root of Trust for Reporting Integrity Metrics“ (RTR); s. *Pearson* (Hrsg), a.a.O. Fn. 11, S. 63.

63 Oftmals, aber nicht notwendigerweise, ist das TPM auf das Motherboard gelötet. Auf die einzelnen Komponenten des TPM kann hier nicht eingegangen werden. S. dazu *Bechtold*, a.a.O. Fn. 9, S. 634; *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 30, 36, 180 ff.; *Brandl/Rosteck*, DuD 2004, 529, 531 f.

64 *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 63, 68, 227.

65 *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 63 f.

66 *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 226 ff., 235. Zur Frage, warum diese Komponenten vertrauenswürdig sind, s. bei Fn. 74 ff.

67 *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 226.

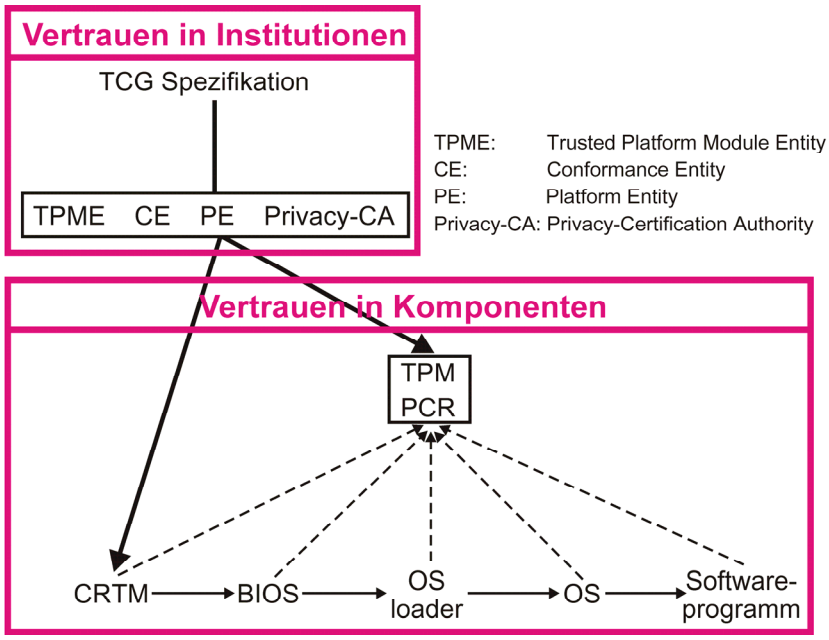


Abbildung 2: Vertrauensverschiebung

Wenn der PC eingeschaltet wird, misst das CRTM innerhalb des BIOS seine eigene Integrität und die Integrität des BIOS. Es speichert eine kondensierte Zusammenfassung dieser Integritätsmetriken in einem manipulationsresistenten PCR innerhalb des TPM.[68] Sobald die Integritätsmetriken in dem PCR gespeichert sind, können sie nicht mehr geändert oder gelöscht werden, solange der PC nicht neu gestartet wird.[69] Das CRTM übergibt dann die Kontrolle an das BIOS, das die Integrität des Betriebssystem-Loaders misst und diese Integritätsmetriken in einem anderen PCR innerhalb des TPM speichert. Das BIOS übergibt dann die Kontrolle an den Betriebssystem-Loader, der die Integrität des Betriebssystems misst, diese Information in einem anderen PCR speichert und die Kontrolle an das Betriebssystem übergibt. Schließlich misst das Betriebssystem die Integrität seiner Komponenten und von Softwareprogrammen, die unter dem Betriebssystem ausgeführt werden sollen, und speichert diese Informationen in nochmals einem anderen PCR.[70]

68 Für mehr Informationen über PCRs s. *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 67 f., 138 ff.
69 *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 36.
70 *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 75, 235; *Brandl/Rosteck*, DuD 2004, 529, 530.

bb) Grundgedanke

Der zentrale Grundgedanke hinter dieser Vorgehensweise ist, dass jede Komponente in einer Trusted-Computing-Plattform die Integrität der nächsten Komponente in der Kette misst und diese Integritätsinformation in einer solchen Weise abspeichert, dass die Information später nicht mehr geändert werden kann.[71] Die Summe dieser Integritätsinformationen gibt im Idealfall einen vollständigen und vertrauenswürdigen Überblick über alle Hardware- und Softwarekomponenten, die auf einer Plattform installiert sind. Dadurch ist es für unsichere Software, Viren und andere gefährliche Programme unmöglich, ihre Existenz auf einer Trusted-Computing-Plattform zu verbergen.[72] Es wird jedoch nicht verhindert, dass ein Virus ausgeführt wird. Auf einer Trusted-Computing-Plattform kann grundsätzlich jede Hard- und Software installiert werden. Durch die Speicherung der Integritätsmetriken wird nur erreicht, dass die Plattform eine vertrauenswürdige Auskunft darüber geben kann, welche Hard- oder Software auf der Plattform installiert ist.[73]

In einer Trusted-Computing-Umgebung muss ein Nutzer also nicht mehr darauf vertrauen, dass ein bestimmtes Computerprogramm auch tatsächlich jene Operationen ausführt, die es auszuführen vorgibt. Er muss nur den „Roots of Trust" vertrauen. Die „Roots of Trust" können dem Nutzer dann versichern, dass das fragliche Computerprogramm tatsächlich vertrauenswürdig ist. Der Nutzer muss nicht mehr einer Unzahl verschiedener Hard- und Softwarekomponenten einer Trusted-Computing-Plattform vertrauen. Er muss nur noch zwei Hardwarekomponenten – den „Roots of Trust" – vertrauen. Die TCG-Architektur baut von diesen Komponenten eine Vertrauenskette zu den anderen Komponenten der Plattform auf.

cc) Folge

Durch die Einführung der „Roots of Trust" wird das Vertrauensproblem letztlich nur verschoben. Denn es ist nicht offensichtlich, warum ein Nutzer den „Roots of Trust" einer Plattform vertrauen sollte. Wurden die „Roots of Trust" ohne Wissen des Nutzers kompromittiert, so ist die gesamte Vertrauenskette wertlos. Um das Vertrauen in die „Roots of Trust" einer bestimmten Plattform herzustellen, müssen nach der TCG-Spezifikation zwei Bedingungen erfüllt sein. *Erstens* müssen die „Roots of Trust" der Plattform mit der TCG-Spezifikation übereinstimmen. Zu diesem Zweck setzt die Spezifikation fünf Zertifikate von vier unterschiedlichen Instanzen ein.[74] Die Zertifikate versichern, dass eine bestimmte Plattform tatsächlich den Vorgaben der TCG-Spezifikation entspricht und damit vertrauenswürdig ist. *Zweitens* muss die TCG-Spezifikation selbst vertrauenswürdig sein. Dies setzt voraus, dass die TCG-Spezifikation genau

71 *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 87.

72 Solche Programme können aber eventuell ihre *Identität* verbergen. Mit Hilfe der dargestellten Mechanismen kann man nur ermitteln, ob eine Plattform vertrauenswürdig ist. Aus welchen Gründen sie nicht vertrauenswürdig ist, erfährt man nicht notwendigerweise. Dazu auch Fn. 35.

73 *Trusted Computing Group*, TCG Specification Architecture Overview, a.a.O. Fn. 15, S. 6; *Poritz* u.a., a.a.O. Fn. 41, S. 6. Zu der Vereinfachung dieser Aussage s. bei Fn. 35.

74 Zu den Einzelheiten der vier Instanzen (TPME, CE, PE, Privacy-CA), s. *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 59 ff., 125 ff., 205 ff., 226 ff.; *Trusted Computing Group*, TCG Architecture Specification Overview, a.a.O. Fn. 15, S. 10; s. a. *Bechtold*, a.a.O. Fn. 9, S. 645, Fn. 2033; *Brandl/Rosteck*, DuD 2004, 529, 534.

und nur das spezifiziert, was sie zu spezifizieren vorgibt. Die *TCG* versucht, dieses Vertrauen in die TCG-Spezifikation herzustellen, indem die Spezifikation öffentlich zugänglich ist und sowohl von Konsumenten als auch von Wissenschaftlern untersucht werden kann.[75]

dd) Innovationsgehalt

Die wirkliche Innovation von Trusted-Computing-Architekturen liegt damit nicht in der Tatsache, dass Nutzer Computerplattformen stärker vertrauen können. Sie liegt darin, dass das Objekt, dem die Nutzer vertrauen müssen, verschoben wird. In einer Trusted-Computing-Umgebung müssen Nutzer nicht mehr einzelnen Komponenten einer Computerplattform oder der gesamten Plattform vertrauen. Sie müssen vielmehr bestimmten Institutionen vertrauen, die für die Sicherheit von Computerplattformen und -komponenten bürgen. Trusted-Computing-Architekturen verwandeln dieses *Vertrauen in Institutionen* in ein *Vertrauen in Komponenten* (s. Abbildung 2).[76] Soziales Vertrauen wird in technisches Vertrauen verwandelt. Zwar lassen sich ähnliche Phänomene auch in anderen Bereichen der Computerentwicklung beobachten.[77] Das Novum von Trusted-Computing-Architekturen ist jedoch, dass die Verwandlung des sozialen Vertrauens in technisches Vertrauen und die dazu notwendigen Vertrauensbeziehungen streng formalisiert werden und dass die sozialen Institutionen, denen letztlich vertraut werden muss, auf ein Minimum reduziert werden.

Bei einer solchen Zentralisierung des Vertrauens in wenige Institutionen ist es umso wichtiger, wie diese Institutionen technisch und rechtlich ausgestaltet sind. Es bedeutende Auswirkungen, ob diese Institutionen monopolistisch ausgestaltet sind oder ob Wettbewerb zwischen verschiedenen gleichrangig agierenden Institutionen besteht. Neutrale Infrastrukturen, die einen unverfälschten Wettbewerb ermöglichen, werden auf dieser Ebene besonders wichtig.[78]

Unglücklicherweise befindet sich die Diskussion über solche Fragen noch in den Anfängen. Das Verhältnis zwischen IT-Sicherheit, Ökonomie und Wettbewerbspolitik hat in der wissenschaftlichen Diskussion erst in letzter Zeit etwas Aufmerksamkeit erregt.[79] Die Entscheidung über die institutionelle Ausgestaltung von Trusted-Computing-Architekturen könnte weitgehende Auswirkungen auf das wettbewerbliche Umfeld haben, in dem sich die IT-Welt in einigen Jahren bewegt. Um so wichtiger ist es, solche Debatten zwischen Technikern, Juristen und Ökonomen voranzutreiben.

---

75 *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 225.
76 S. a. *Pearson* (Hrsg.), a.a.O. Fn. 11, S. 234.
77 Ganz allgemein entspringt ein Vertrauen in eine bestimmte Soft- oder Hardware oft einem Vertrauen in den Entwickler der Soft- oder Hardware.
78 Näher *Bechtold*, a.a.O. Fn. 9, S. 647.
79 S. *Camp/Lewis* (Hrsg.), Economics of Information Security, Boston 2004.

2. Vertrauen

Diese Überlegungen führen zu der abstrakteren Frage, auf welche Weise Trusted-Computing-Architekturen das Vertrauen in die IT-Infrastruktur erhöhen wollen. Indem Trusted-Computing-Architekturen eine Vertrauensverschiebung von Vertrauen in Institutionen zu Vertrauen in Komponenten ermöglichen, reduzieren sie die Anzahl der Elemente, in die Nutzer Vertrauen setzen müssen. Damit geben Trusted-Computing-Architekturen eine Antwort auf unsere heutige IT-Umgebung, die so komplex ist, dass sie von den Nutzern nicht einmal in ihren Grundzügen durchschaut werden kann. Die Einführung von Trusted Computing in eine solche Umgebung kann als Antwort auf die inhärente Komplexität der Umgebung gesehen werden.[80]

Auch in einer Trusted-Computing-Infrastruktur muss der Nutzer aber letztlich jemandem vertrauen. In der TCG-Architektur sind diese „Vertrauensanker" bestimmte Institutionen, die Zertifikate oder Referenz-Integritätsmetriken ausstellen. Nach der Philosophie der *TCG* kann jedermann einen solchen Vertrauensanker betreiben: „*TCG* feels credibility may be found among many organizations ranging from product manufacturers, vendors, product consumers and consultants. The product owner ultimately decides which certifier best contributes to assurance and risk management calculations."[81] Die *TCG* will also nicht festlegen, welchen Institutionen die Nutzer einer Trusted-Computing-Infrastruktur vertrauen müssen. Die Vertrauenswürdigkeit der Institutionen soll sich im Wettbewerb herausbilden. Auch wenn eine solche Philosophie Einiges für sich hat, setzt sie doch voraus, dass ein funktionsfähiger Wettbewerb zwischen verschiedenen Institutionen besteht. Angesichts von Netzwerkeffekten, Pfad-Abhängigkeiten, Informationsasymmetrien, Default-Einstellungen in Hard- und Software sowie anderen Lock-in-Effekten ist eine solche Annahme zumindest begründungsbedürftig.[82] Je größer die Zahl derjenigen ist, die einer bestimmten Institutionen schon vertrauen, umso rationaler wird es für Außenstehende, dies auch zu tun. Ist ein gewisser „Tipping Point" erreicht, so wird es unwahrscheinlich, dass sich andere Institutionen dagegen im Wettbewerb behaupten können.[83]

Unter dem Aspekt einer interdisziplinären Erforschung von Vertrauen ist das Trusted Computing ein interessanter Untersuchungsgegenstand. Es geht um das grundlegende Design von Vertrauensbeziehungen in einem System, in dem alle Komponenten streng formalisiert sind. Dabei kann auch juristischer Sachverstand weiterhelfen. Aus juristischer Perspektive sind die Erstellung, Ausgestaltung und Modularisierung von Vertrauensbeziehungen eigentlich ein altes Thema. Fragen der Ausgestaltung von Vertragsbeziehungen und des Institutionendesigns spielen seit

---

80 *Kuhlmann* in: Büllesbach/Dreier (Hrsg.), Wem gehört die Information im 21. Jahrhundert?, Köln 2003, S. 75, 77; *ders.*, a.a.O. Fn. 58, S. 257 f.; s. a. *Nissenbaum*, 81 Boston University Law Review 635 (2001). Zum generalisierten Vertrauen als Mittel zur Reduzierung von Komplexität s. *Luhmann*, Vertrauen, 4. Aufl., Stuttgart 2000. Zum individuellen Vertrauen aus rationaltheoretischer Perspektive s. *Engel*, Vertrauen: ein Versuch, 1999, *http://www.coll.mpg.de/pdf_dat/9912.pdf*.

81 *Trusted Computing Group*, TCG Specification Overview, a.a.O. Fn. 15, S. 46.

82 Näher *Bechtold*, a.a.O. Fn. 9, S. 646 f. Auf die potentiellen kartellrechtlichen Probleme in einem Markt von Zertifizierungsinstanzen weisen *Koenig/Neumann*, WuW 2003, 2, 15 f., hin.

83 Zu den zugrunde liegenden ökonomischen Phänomenen s. *Lemley/McGowan*, 86 California Law Review 479 (1998); *Katz/Shapiro*, 75 American Economic Review 424 (1985); *Katz/Shapiro*, 8 Journal of Economic Perspectives 93 (1994); *Liebowitz/Margolis*, 17 Research in Law and Economics 1 (1995); *Bechtold*, a.a.O. Fn. 48, S. 351 ff.; *Elkin-Koren/Salzberger*, Law, Economics and Cyberspace, Cheltenham 2004, S. 44 f.

jeher eine wichtige Rolle. Die Debatte mit Technikern, die im Bereich des Trusted Computing arbeiten, zeigt, dass Anregungen von Juristen und anderen Geisteswissenschaftlern hilfreiche Impulse für die Fortentwicklung von Trusted-Computing-Architekturen geben können.

3. Autonomie und Paternalismus

Einer der Kritikpunkte an Trusted-Computing-Architekturen ist, dass sie dem Inhaber einer Trusted-Computing-Plattform die Kontrolle über seine eigene Plattform entziehen können. Anders als bei herkömmlichen Computern hat der Inhaber einer Trusted-Computing-Plattform keinen vollständigen Zugriff auf alle Komponenten der Plattform. Es kann vorkommen, dass die Plattform Operationen ausführt, die im Einzelfall dem Interesse des Inhabers widersprechen.

Diese Kritik wird oftmals mit dem Argument abgetan, dass eine Trusted-Computing-Plattform unter der vollständigen Kontrolle ihres Eigentümers bleibe. Es sei dessen freie Entscheidung, ob er die Trusted-Computing-Komponenten auf seiner Plattform überhaupt einschalte. Auch sei es seine freie Entscheidung, ob und in welcher Weise er diese Komponenten verwende.[84]

Die Vorstellung, es sei in einer Trusted-Computing-Umgebung die freie Entscheidung des Inhabers einer Plattform, ob er Trusted-Computing-Funktionalitäten verwendet oder nicht, greift zu kurz. Dies gilt zumindest, wenn man dem Begriff der „freien Entscheidung" einen materiellen Bedeutungsgehalt belassen will. Es ist denkbar, dass wir zukünftig in großem Umfang von Computeranwendungen umgeben sein werden, die auf Trusted-Computing-Funktionalitäten aufbauen. In einer solchen Umgebung wird der Inhaber faktisch gezwungen sein, Trusted Computing zu verwenden.[85] Ist es beispielsweise unmöglich, eine E-Mail zu verschicken, im Internet einzukaufen oder Homebanking zu betreiben, ohne Trusted Computing zu verwenden, verliert das Argument, es sei die „freie Entscheidung" des Inhabers, ob er Trusted Computing verwenden wolle oder nicht, an Überzeugungskraft.

Um in der Debatte voranzukommen, scheint es erstens hilfreich zu akzeptieren, dass Trusted Computing in Zukunft eine Technologie sein könnte, die den Inhabern von Computern und anderen Elektronikgeräten in bestimmten Bereichen aufoktroyiert werden wird. Von dieser Einsicht ausgehend, sollte man sich fragen, wie eine solche Technologie im Sinne einer rechtlichen Technikgestaltung beeinflusst werden kann, damit wichtige rechts- und wettbewerbspolitische Anliegen gewahrt bleiben.

Zusätzlich ist zu beachten, dass auch aus der technischen Entwicklergemeinde mitunter vorgebracht wird, Ziel des Trusted Computing sei nicht, die Autonomie des Nutzers von Trusted-Computing-Plattformen zu schützen. Gerade im Haushaltssektor könnten Sicherheitsrisiken auftreten, die der Nutzer aufgrund ihrer Komplexität überhaupt nicht einschätzen kann. In einem

84 So z.B. *Intel Corp.*, LaGrande Technology Policy on Owner/User Choice and Control, Rev. 0.8, September 2003, S. 3, *http://www.intel.com/technology/security/downloads/LT_policy_statement_0_8.pdf.*

85 *Bechtold*, a.a.O. Fn. 9, S. 646 f.

solchen Umfeld müsse der Nutzer vor sich selbst geschützt werden. Dabei könnten die Sicherheitsproblem nicht durch eine Aufklärung des Nutzers gelöst werden. Vielmehr müsse eine Trusted-Computing-Architektur paternalistisch ausgestaltet werden, so dass der Nutzer vor Sicherheitsrisiken geschützt wird, denen er sich ansonsten ungewollt aussetzen würde.[86] Eine Trusted-Computing-Plattform müsse dem Nutzer in manchen Bereichen die Kontrolle über die Plattform entziehen, damit der Nutzer sich nicht selbst schädige.

Eine solche paternalistische Konzeption von IT-Sicherheit basiert auf Phänomenen, die in der „Behavioral Law and Economics"-Literatur unter Stichworten wie „Overconfidence Bias" und „Availability Heuristic" behandelt werden.[87] Zwar ist es im vorliegenden Rahmen unmöglich, dieses Problem genauer zu analysieren. Es soll jedoch auf die Problemdimension hingewiesen werden. Wenn man anerkennt, dass sich Individuen in bestimmten Fällen systematisch selbst schädigen, mag der Einsatz unterschiedlicher Regulierungsinstrumente – z. B. Aufklärung der Nutzer oder Entzug der technischen Kontrolle über eine Trusted-Computing-Plattform – zur Verhinderung solcher Selbstschädigungen angemessen sein.[88] Dabei müssen die Vor- und Nachteile der unterschiedlichen Regulierungsinstrumente sorgfältig gegeneinander abgewogen werden.[89] Hier besteht noch Forschungsbedarf. Auch wenn die ökonomische Analyse von IT-Sicherheit zunehmend an Aufmerksamkeit gewinnt,[90] fehlen bisher Arbeiten, die Probleme der IT-Sicherheit unter einer verhaltenswissenschaftlich informierten Perspektive beleuchten.

4. Wettbewerb

Im Folgenden soll drei weiteren Bereichen nachgegangen werden, die unter wettbewerbspolitischen Aspekten Probleme aufwerfen.

a) „Sealed Storage"

In Trusted-Computing-Plattformen ermöglicht das sog. „Sealed Storage", Daten kryptographisch an bestimmte Systeme oder auch Systemzustände zu binden.[91] Dadurch können die Daten nur von einem bestimmten Computer entschlüsselt werden. Oder sie können sogar von diesem Computer nur entschlüsselt werden, wenn auf dem Computer bestimmte Hardware- oder Softwarekomponenten installiert sind. Ähnlich wie bei der „Remote Attestation" kann auch diese Funkti-

86 Dazu *Schoen*, User Education and Paternalism, 2005, *http://vitanuova.loyalty.org/weblog/nb.cgi/view/vitanuova/2005/09/16/0*; *Bechtold*, The Paternalistic Paternalism of TC, *http://cyberlaw.stanford.edu/blogs/bechtold/archives/003319.shtml*.

87 Zu den Konzepten s. *Rachlinski*, 97 Nw. U. L. Rev. 1165, 1170 ff. (2003). Zur Anwendung dieser Konzepte auf das Vertragsrecht s. *Korobkin*, 70 U. Chi. L. Rev. 1203, 1232 f. (2003).

88 Vor einer übereilten Anwendung paternalistischer Regulierungsinstrumente warnend und differenzierend *Rachlinksi*, a.a.O. Fn. 87.

89 Zum begrenzten Nutzen einer Aufklärung von Individuen in Fällen, in denen diese ihre Entscheidung unter Unsicherheit nur auf dem möglicherweise eintretenden Schaden, nicht aber auf der Wahrscheinlichkeit des Schadenseintritts basieren, s. *Sunstein*, 112 Yale L.J. 61, 91 f. (2002).

90 S. bei Fn. 79.

91 S. bei Fn. 28.

onalität missbraucht werden. Softwarehersteller könnten proprietäre Dateiformate erstellen, die es Wettbewerbern unmöglich machen, dieses Dateiformat zu lesen oder in diesem zu schreiben.[92] Auch mit dem „Sealed Storage" können daher Interoperabilität zwischen konkurrierenden Softwareprogrammen eingeschränkt und Wettbewerb behindert werden.[93] Die Kosten der Konvertierung von Dateien können künstlich erhöht werden. Nutzer können vom Wechsel zu anderen Betriebssystemen, Softwareprogrammen und Hardwareinstallationen abgehalten werden. „Sealed Storage" könnte ein mächtiges Werkzeug sein, um „Lock-in"-Effekte zu erzielen und künstlich Wechselkosten zu erhöhen.[94] Dies wurde schon unter dem Blickwinkel des Art. 82 EG problematisiert.[95]

Ähnlich wie bei den Problemen um die „Remote Attestation" existieren unterschiedliche Lösungsansätze. Die Techniker sehen teilweise Möglichkeiten, bei denen das Problem auf technischer Ebene umgangen wird.[96] Die Juristen argumentieren für das Eingreifen kartellrechtlicher Instrumentarien.[97] Die *TCG* selbst sucht das Problem mit Hilfe nicht bindender Richtlinien ihres „Best Practices Committee" zu lösen.[98]

b) Schnittstellen und Patente

Damit ein Softwareprogramm auf Komponenten einer Trusted-Computing-Architektur zurückgreifen kann, muss es mit diesen Komponenten über Schnittstellen („Application Programming Interfaces", APIs) kommunizieren. Sowohl TCG und NGSCB als auch LaGrande verfügen über eine Vielzahl von APIs. Wären diese APIs proprietär und ihre Funktionsweise nicht näher veröffentlicht, so würde die Entwicklung von Hard- und Software durch unabhängige Unternehmen deutlich beeinträchtigt. Hinsichtlich NGSCB wurde darauf hingewiesen, dass die Kontrolle von *Microsoft* über NGSCB-Schnittstellen zu kartellrechtlichen Bedenken nach Art. 82 EG führen könnte.[99]

Wegen dieser Bedenken wurde schon früh gefordert, dass Trusted-Computing-Architekturen ihre Schnittstellen offen legen müssen.[100] Die wichtigsten Unternehmen im Trusted-Computing-

---

92 Schon bisher haben Softwarehersteller mit Hilfe proprietärer Dateiformate versucht, Wettbewerber zu behindern. Mit Hilfe des „Sealed Storage" sind solche Versuche jedoch weitaus erfolgversprechender, da die Verfahren auf einer hardwarebasierten Verschlüsselung beruhen, was eine Umgehung durch Wettbewerber nahezu unmöglich macht.

93 *Arbaugh*, a.a.O. Fn. 10; *Anderson*, a.a.O. Fn. 61, S. 46 ff.

94 *Anderson*, a.a.O. Fn. 61, S. S. 47 ff.

95 *Koenig/Neumann*, MMR 2003, 695, 700.

96 So wird – ähnlich den Ansätzen einer „Property Attestation", s. dazu oben Fn. 41 – vorgeschlagen, Daten nicht an bestimmte Hashwerte von Objektcode, sondern an abstraktere Sicherheiteigenschaften des Computerprogramms zu binden; s. *Kühn* u.a. in Proceedings of the Workshop on Cryptographic Hardware and Embedded Systems (CHES), Berlin 2005, S. 324. Daneben hat *Seth Schoen* von der „*Electronic Frontier Foundation*" auf die Möglichkeit der Virtualisierung des „Sealed Storage" hingewiesen, die aber nur funktioniert, wenn eine Trusted-Computing-Architektur keine „Remote Attestation" kennt. Daher ist dieser zweite Vorschlag wenig realistisch.

97 *Bechtold*, a.a.O. Fn. 9, S. 642.

98 Dazu schon bei Fn. 50.

99 *Koenig/Neumann*, MMR 2003, 695, 700.

100 *Bundesregierung*, a.a.O. Fn. 3, S. 2.

Bereich haben dies inzwischen zugesagt. Weiterhin existieren mehrere Projekte, die zentrale Komponenten von Trusted-Computing-Architekturen auf Open-Source-Basis implementieren. So veröffentlichte *IBM* 2004 eine Open-Source-Implementierung des „TCG Software Stack“.[101]

Neben der Schnittstellenproblematik ist zu beachten, dass viele Trusted-Computing-Komponenten durch Patente und als Know-how geschützt sein können. Unter wettbewerbspolitischen Gesichtspunkten kommt den Bedingungen, unter denen die Inhaber dieser Patente außenstehenden Dritten Lizenzen erteilen, eine entscheidende Bedeutung zu. Die *TCG* bietet für ihre Mitglieder Patentlizenzen unter RAND[102]-Bedingungen an.[103] Nichtmitglieder profitieren nicht von dieser RAND-Lizenz. Microsoft hat bis heute noch nicht angekündigt, unter welchen Bedingungen das Unternehmen seine eventuellen Patente an der NGSCB-Technologie lizenzieren will.

Von verschiedener Seite wurde die Einrichtung eines Technologiepools gefordert, der wichtige Trusted-Computing-Technologien erfasst und die Technologien unter RAND-Lizenzen an jedermann lizenziert.[104] Dazu sind die Unternehmen, die Patente in diesem Bereich halten, dem Vernehmen nach aber nicht bereit.

c) Trusted-Computing-Organisationen

In Deutschland wurde von Prof. *Christian Koenig* und *Andreas Neumann* die Organisationsstruktur der *TCG* kritisiert.[105] Neben der dargestellten Beschränkung der RAND-Lizenz auf TCG-Mitglieder wurde der *TCG* vorgeworfen, ihren Mitgliedern einen Vorsprung an technischem Wissen zu verschaffen. Nichtmitglieder hätten keinen Einfluss auf das Ergebnis der Standardisierung und könnten den Standard erst mit zeitlicher Verzögerung übernehmen.[106] Aufgrund des ursprünglichen Mindest-Mitgliedsbeitrags von jährlich 7.500 $, der starren Beitragssätze sowie der zwischen den Mitgliedschafts-Kategorien gestaffelten Mitspracherechte[107] sei zu prüfen, ob die TCG-Standardisierungsinitiative gegen Art. 81 Abs. 1 EG verstoße, wobei eine Freistellung nach Art. 81 Abs. 3 EG denkbar sei.[108]

---

101 *http://sourceforge.net/projects/trousers*.

102 „Reasonable and non-discriminatory“.

103 *Trusted Computing Group*, Bylaws, Section 16.4., 20. 3. 2003, erhältlich unter *https://www.trustedcomputinggroup.org/downloads/org_docs*.

104 *Koenig/Neumann*, DuD 2004, 555, 558 f.; *Bundesregierung*, a.a.O. Fn. 3, S. 6; s. a. *Kuhlmann*, DuD 2004, 545; *Sandl*, DuD 2004, 521, 524.

105 U.a. in *Koenig/Neumann*, MMR 2003, 695; *Koenig/Neumann* in Koenig/Neumann/Katzschmann (Hrsg.), Trusted Computing, Heidelberg 2003, S. 100 ff.; *Koenig/Neumann*, WuW 2003, 2; *Koenig/Neumann*, DuD 2004, 555; s. weiterhin *Koenig/O'Sullivan*, European Competition Law Review 2003, 449.

106 *Koenig/Neumann*, MMR 2003, 695, 698; *Koenig/Neumann* in Koenig/Neumann/Katzschmann (Hrsg.), a.a.O. Fn. 105, S. 100, 120 ff.

107 Die „Adopters“, deren Mitgliedsbeitrag damals mindestens 7.500 $ betrug, haben nur sehr beschränkte Mitwirkungs- und Informationsrechte. Erst die „Contributors“, deren Jahresbeitrag 15.000 $ beträgt, verfügen über weiter gehende Beeinflussungsmöglichkeiten. Zur Übersicht über die unterschiedlichen Mitgliedschafts-Kategorien s. *https://www.trustedcomputinggroup.org/join/levels*.

108 *Koenig/Neumann*, MMR 2003, 695, 699; *Koenig/Neumann* in Koenig/Neumann/Katzschmann (Hrsg.), a.a.O. Fn. 105, S. 100, 120 ff.; 128 ff.; allgemein dazu auch *Sandl*, DuD 2004, 521, 524; *Bundesregierung*, a.a.O. Fn. 3, S. 6.

Diese Kritikpunkte sollen hier nicht überbetont werden. Erstens hat die *TCG* inzwischen teilweise auf die Kritik reagiert. Sie bietet inzwischen den Status eines „Small Adopters“ mit einem reduzierten Jahresbeitrag von 1.000 $ an.[109] Auch existiert inzwischen ein „Industry Liaison Program“, das akademischen Einrichtungen, Standardisierungsiniativen, Behörden und Interessenvertretungen eine kostenlose Mitarbeit in der *TCG* ermöglicht, solange einer Vertraulichkeitsvereinbarung zugestimmt wird. Zweitens sind die dargestellten Probleme hinsichtlich Schnittstellen, Lizenzierung von Immaterialgüterrechten und Organisationsstruktur keine spezifischen Probleme des Trusted Computing. Es sind allgemeine kartellrechtliche Probleme technischer Standardisierung.[110] Dem vorliegenden Beitrag geht es darum, das spezifisch Neue des Trusted Computing aufzuzeigen.

5. Datenschutz

Trusted-Computing-Architekturen können zu datenschutzrechtlichen Problemen führen. In jedem TPM einer Trusted-Computing-Plattform ist der private Schlüssel eines asymmetrischen Schlüsselpaares, das den Namen „Endorsement Key“ trägt, gespeichert. Grundsätzlich könnte der „Endorsement Key“ von Dritten verwendet werden, um die Plattform eindeutig zu identifizieren. Die daraus resultierenden datenschutzrechtlichen Probleme hinsichtlich der Erstellung von Nutzerprofilen hat die *TCG* von Anfang an erkannt. Inzwischen bietet die TCG-Spezifikation zwei Lösungen an.

a) „Privacy Certification Authorities“

Die erste Lösung war schon immer Bestandteil der TCG-Spezifikation. „Privacy Certification Authorities“ (Privacy-CAs) ermöglichen eine pseudonyme Benutzung von Trusted-Computing-Plattformen. Tritt eine Plattform mit einer anderen in Kontakt, wird zur Identifizierung nicht der öffentliche Schlüssel des „Endorsement Keys“ übertragen. Vielmehr verwendet die Plattform einen „Attestation Identity Key“ (AIK) als Pseudonym, der von einer Privacy-CA bestätigt wurde.[111] Dieses Pseudonym enthält ein Zertifikat der Privacy-CA, aber keine Informationen über die Identität der Plattform. Jede Plattform kann beliebig viele Pseudonyme erhalten. Dadurch kann der Nutzer einer Plattform unterschiedliche Pseudonyme für unterschiedliche Zwecke verwenden.

Diese Ausgestaltung hat Nachteile. Einerseits hängt ihre Datenschutzfreundlichkeit von der Vertrauenswürdigkeit der Privacy-CA ab.[112] Da die Privacy-CA die „Attestation Identity Keys“

109 Hinsichtlich der Entwickler von Open-Source-Programmen ist jedoch weiterhin etwas unklar, wie diese an der Standardisierungsarbeit der *TCG* mitwirken sollen. Die Bundesregierung hat die *TCG* aufgefordert, Open-Source-Projekte von immaterialgüterrechtlichen Lizenzgebühren freizustellen; s. *Bundesregierung*, a.a.O. Fn. 3, 6.

110 Allgemein dazu *Heinemann*, Immaterialgüterschutz in der Wettbewerbsordnung, Tübingen 2002, S. 104 ff., 514 ff.

111 *Kursawe*, DuD 2004, 566.

112 *Kursawe*, DuD 2004, 566.

bestätigt, die öffentlichen Schlüssel der „Endorsement Keys“ der einzelnen Plattform kennt und bei jeder Transaktion konsultiert werden muss, kann sie theoretisch mehrere Pseudonyme korrelieren und dadurch einzelne Nutzer identifizieren.[113] Der Privacy-CA-Ansatz baut stark auf der Vertrauenswürdigkeit der Privacy-CAs auf. Auch in diesem Bereich zeigt sich, dass die TCG-Philosophie von einem Wettbewerb zwischen verschiedenen Privacy-CAs ausgeht. Dadurch soll es den Nutzern ermöglicht werden, eine bestimmte Privacy-CA auszuwählen, die ihren persönlichen Präferenzen am Besten entspricht. Leider kann nur eine tatsächliche Implementierung von Trusted-Computing-Architekturen zeigen, ob ein solcher Wettbewerb realistisch ist[114] und ob er ausreicht, um Datenschutzinteressen der Nutzer adäquat zu schützen. Derzeit ist noch völlig unklar, wer die Privacy-CAs betreiben soll.[115]

b) „Direct Anonymous Attestation“

Im Jahr 2003 beschäftigte sich die Art. 29-Datenschutzgruppe[116] wiederholt mit datenschutzrechtlichen Aspekten des Trusted Computing. Dabei wurde unter anderem die starke Datenkonzentration bemängelt, die bei Privacy-CAs auftreten kann.[117] Unter anderem auf diese Kritik hin integrierte die *TCG* in der Spezifikation 1.2, die Ende 2003 fertiggestellt wurde, eine Alternative zu dem Privacy-CA-Ansatz. Mit der sog. „Direct Anonymous Attestation“ (DAA) ist eine weitgehend anonyme Kommunikation zwischen zwei Parteien möglich, ohne dass eine dritte Partei (d.h. eine Privacy-CA) hinzugezogen werden muss.[118] Auch dieser Ansatz hat seine Schwächen. Um im Falle eines erfolgreichen physikalischen Angriffs auf eine bestimmte Trusted-Computing-Plattform die Plattform identifizieren und von der weiteren Kommunikation ausschließen zu können, ist es auch bei der „Direct Anonymous Attestation“ unter gewissen Voraussetzungen möglich, individuelle Plattformen zu identifizieren.[119] Auch in diesem Ansatz können unter gewissen Umständen ein Bezug zur Identität des Nutzers hergestellt oder Benutzerprofile erstellt werden. In einem solchen Fall kann von Anonymität keine Rede mehr sein.[120] Selbst wenn DAA eine vollständige Anonymität garantieren könnte, bestehen immer noch zahlreiche andere Möglichkeiten, um Nutzer in heutigen Computernetzwerken eindeutig zu identifizieren.[121] Positiv ist anzumerken, dass im DAA-Ansatz zumindest keine zentralisierte Instanz

---

113 *Pfitzner* in Koenig/Neumann/Katzschmann (Hrsg.), S. 29, 44 ff.; *Brickell* u.a. in Pfitzmann/Liu (Hrsg.), Proceedings of the 11th ACM Conference on Computer and Communications Security 2004, S. 132, 133.

114 Dazu schon bei Fn. 82.

115 *Kursawe*, DuD 2004, 566.

116 Dazu bei Fn. 5.

117 *Artikel 29-Datenschutzgruppe*, Fn. 7, S. 8.

118 Dazu kurz *Kursawe* in Koenig/Neumann/Katzschmann (Hrsg.), S. 70 f.; etwas länger *Trusted Computing Group*, TPM v1.2 Specification Changes, Oktober 2003, S. 3 ff., *https://www.trustedcomputinggroup.org/downloads/TPM_1_2_Changes_final.pdf*; ausführlich *Brickell* u.a., a.a.O. Fn. 113. Der Ansatz baut auf sog. „Group Signatures“ auf und orientiert sich an sog. „Zero Knowledge“-Beweisen.

119 *Camenisch* in: Samarati u.a. (Hrsg.), 9th European Symposium on Research in Computer Security 2004, Berlin 2004, S. 73, 74 f. Dabei kann der Grad der Anonymität von der Partei beeinflusst werden, die einen „Remote Challenge“ durchführt. Zu den Einzelheiten s. *Trusted Computing Group*, a.a.O. Fn. 118, S. 5 ff.

120 *Artikel 29-Datenschutzgruppe*, a.a.O. Fn. 7, S. 8.

121 *Schoen*, a.a.O. Fn. 38, S. 114, weist in diesem Zusammenhang auf die Identifizierung durch Ethernet-Karten, Festplatten und Dateisysteme hin.

notwendig sein soll, um ein kompromittiertes TPM zurückzurufen („certificate revocation agency“).[122] Dadurch werden wenigstens wettbewerbsrechtliche Probleme vermieden, die bei einer Zentralisierung solcher Rückruf-Funktionalitäten auftreten könnten.

c) Keine vollständige Anonymität

Es zeigt sich, dass Trusted-Computing-Umgebungen derzeit keine vollständig anonyme Benutzung in dem Sinne erlauben, dass die Anonymität aus technischen Gründen unter keinerlei Umständen aufgehoben werden kann. Zwar ist auch hier die Entwicklung noch nicht abgeschlossen.[123] Ob man aber jemals zu einer technisch vollständig anonymen Trusted-Computing-Umgebung kommt, darf bezweifelt werden. Dann muss das Datenschutzrecht unterstützend eingreifen.[124]

6. Urheberrecht

In den letzten Jahren bildete das Verhältnis zwischen Trusted Computing und dem Urheberrecht einen der Schwerpunkte der rechtspolitischen Debatte. Kritiker warfen Trusted-Computing-Architekturen vor, diese seien nur entworfen worden, um ein äußerst sicheres „Digital Rights Management“-System (DRM)[125] zu ermöglichen, das berechtigte Interessen von Nutzern und der Allgemeinheit beschneidet.

Trusted-Computing-Architekturen könnten für Entwickler von DRM-Systemen sehr attraktiv sein, da sie eine stabile Grundlage für ein sicheres DRM-System darstellen könnten.[126] DRM-Systeme könnten die hardwarebasierten manipulationsresistenten Trusted-Computing-Mechanismen in den Bereichen der Verschlüsselung, Integritäts- und Authentizitätsprüfung, der Schlüsselverwaltung sowie der Durchsetzung von Zugangsrechten nutzen.

Aus verschiedenen technischen Gründen ist derzeit noch unklar, ob Trusted-Computing-Architekturen in naher Zukunft tatsächlich als Grundlage für DRM-Systeme im Konsumentenbereich eingesetzt werden können.[127] Dennoch soll kurz auf potentielle Gefahren eingegangen werden, die bei DRM-Systemen auftreten können, die auf Trusted-Computing-Architekturen aufbauen. Obwohl solche Gefahren existieren, sind sie in der Regel kein Spezifikum von Trusted-Computing-Architekturen, sondern allgemeine Probleme des Digital Rights Manage-

122 So *Brickell* u.a., a.a.O. Fn. 113, S. 136 f.
123 S. z.B. *Camenisch*, a.a.O. Fn. 119.
124 Es sei darauf hingewiesen, dass noch andere Mechanismen des Trusted Computing datenschutzrechtliche Probleme aufwerfen können. Wenn im Rahmen einer „Remote Attestation“ detaillierte Informationen über den Plattformzustand an außenstehende Dritte übertragen werden, können aus diesen Informationen eventuell Rückschlüsse auf die Identität der Plattform gewonnen werden; s. *Poritz* u.a., a.a.O. Fn. 41, S. 7.
125 Zum DRM allgemein s. *Bechtold*, a.a.O. Fn. 48.
126 *Erickson*, Communications of the ACM, April 2003, S. 34, 38 f.; *Anderson*, a.a.O. Fn. 61, S. 38.
127 Dazu näher *Bechtold*, a.a.O. Fn. 9, S. 639 ff.

ment.[128] Ein Beispiel mag dies belegen. Wie oben dargestellt wurde, können digitale Inhalte durch das „Sealed Storage“ kryptographisch an eine bestimmte Plattform, ja sogar an eine bestimmte Plattformkonfiguration gebunden werden.[129] Wenn eine urheberrechtliche Schrankenbestimmung einem Nutzer erlaubt, den Inhalt von einer Plattform auf eine andere zu kopieren, kann eine Trusted-Computing-Architektur dies verhindern, da der verschlüsselte Inhalt auf der anderen Plattform nicht entschlüsselt werden kann.

Trusted-Computing-Architekturen können dadurch in Konflikt zu urheberrechtlichen Schrankenregelungen geraten. Dieser Konflikt ist kein Spezifikum von Trusted-Computing-Architekturen, sondern ein allgemeiner Konflikt zwischen technischen Schutzmaßnahmen und urheberrechtlichen Schrankenbestimmungen. Die DRM-Standards „Content Protection for Recordable and Pre-recorded Media“ (CPRM und CPPM) können Inhalte ebenfalls kryptographisch an bestimmte Geräte binden.[130] Selbst software-basierte DRM-Systeme können verhindern, dass digitale Inhalte von einem zum anderen Gerät kopiert werden. § 95b UrhG enthält inzwischen eine ausführliche Regelung, durch die der Konflikt zwischen technischen Schutzmaßnahmen und urheberrechtlichen Schrankenbestimmungen allgemein gelöst werden soll.[131]

Trusted-Computing-Architekturen stellen das Urheberrecht vor keine qualitativ neuartigen Herausforderungen.[132] Das Neue an Trusted-Computing-Architekturen ist, dass sie ein sehr viel höheres Sicherheitsniveau bieten. Die Umgehung von DRM-Systemen, die auf Trusted-Computing-Architekturen aufbauen, ist sehr viel schwieriger als die Umgehung herkömmlicher DRM-Systeme. Dadurch treten die potentiellen Konflikte zum Urheberrecht bei DRM-Systemen, die auf Trusted-Computing-Plattformen aufbauen, sehr viel deutlicher zu Tage.

7. Offene Fragen

Auch wenn der Beitrag einen Überblick über unterschiedliche rechtliche Probleme des Trusted Computing geben will, können im vorliegenden Zusammenhang nicht alle Probleme behandelt werden. Dies liegt teilweise daran, dass sich Trusted-Computing-Technologien noch im Entwicklungsstadium befinden, was klare Aussagen über rechtliche Auswirkungen deutlich erschwert. Teilweise liegt die Beschränkung des Beitrags auch daran, dass potentielle Probleme keine Spezifika des Trusted Computing sind, sondern größere Fragestellungen betreffen. Es sollen vier Punkte erwähnt werden, die in Zukunft genauer untersucht werden sollten. *Erstens* stellt sich aus rechtlicher Sicht die Frage, wie Haftungsrisiken in Trusted-Computing-Architekturen verteilt sind. Auch wenn sich ähnliche Fragen wie bei der Haftung von Zertifizie-

128 Eine allgemeine Analyse des Verhältnisses zwischen DRM und Urheberrecht findet sich bei *Bechtold*, a.a.O. Fn. 48.
129 S. bei Fn. 28.
130 *Bechtold*, a.a.O. Fn. 48, S. 113 f.
131 Dazu *Bechtold* in: Hoeren/Siebert (Hrsg.), Handbuch Multimedia-Recht, Losebl.-Slg. München, Teil 7.11; *Peukert* in: Loewenheim (Hrsg.), Handbuch Urheberrecht, München 2003.
132 Weitere Belege für diese These finden sich bei *Bechtold*, a.a.O. Fn. 9, S. 647 ff.

rungsdiensteanbietern unter dem Signaturgesetz stellen,[133] sind konkrete Aussagen zur Haftungsallokation in Trusted-Computing-Architekturen derzeit schwierig, weil zu wenig Informationen über die konkrete Ausgestaltung der Infrastrukturen zur Verfügung stehen. *Zweitens* sind die rechtspolitischen Auswirkungen von Trusted-Computing-Architekturen im Mobilfunksektor noch völlig ungeklärt. *Drittens* soll im nächstens Jahr die „GNU Public License" (GPL) – die wichtigste Open-Source-Lizenz – in einem aufwendigen Verfahren reformiert werden. Zwar sind noch keine Details bekannt. Dem Vernehmen nach wird jedoch erwogen, in die nächste Version der GPL eine Klausel aufzunehmen, die die Interaktion zwischen Open-Source-Software und Trusted-Computing-Architekturen regelt. *Viertens* stellt sich aus rechtsökonomischer Sicht die Frage, ob und in welchem Umfang Kunden wirklich bereit sind, für eine Erhöhung der IT-Sicherheit zu bezahlen.[134]

IV. Vorteile

Zwar hat der Beitrag seinen Schwerpunkt darauf gelegt, potentielle Probleme des Trusted Computing zu behandeln. Es sollen jedoch auch die Vorteile des Trusted Computing nicht verschwiegen werden, die aus rechtlicher Perspektive interessant erscheinen.

1. Materiell-rechtliche Seite

Neben der schon erwähnten Einsatzmöglichkeit des Trusted Computing als Grundlage für DRM-Systeme sowie dessen datenschutzfreundlicher Ausgestaltung durch Privacy-CAs und „Direct Anonymous Attestation" ist Trusted Computing insbesondere aus vertragsrechtlicher Perspektive interessant. Die manipulationsresistenten TPMs können digitale Signaturen erstellen, die sich durch eine besonders hohe Sicherheit und Vertrauenswürdigkeit auszeichnen. Trusted-Computing-Plattformen lassen sich in den rechtlichen Rahmen digitaler Signaturen einfügen, der durch §§ 126a BGB, 130a, 130b, 174 III, 317 V, 371a ZPO und sonstige Vorschriften des neuen Justizkommunikationsgesetzes[135] sowie das Signaturgesetz und die Signaturverordnung errichtet wurde.

2. Prozessuale Seite

Weiterhin ist die eingangs[136] erwähnte Möglichkeit eines sicheren Kommunikationsweges zwischen Prozessor, Maus, Tastatur und Grafiksystem einer Trusted-Computing-Plattform interessant. Stimmt der Nutzer eines heutigen PCs auf seinem Bildschirm einem Vertrag durch das

133 Dazu *Thomale*, MMR 2004, 80.
134 S. dazu die Beiträge in *Camp/Lewis* (Hrsg.), Economics of Information Security, Boston 2004.
135 Gesetz über die Verwendung elektronischer Kommunikationsformen in der Justiz (Justizkommunikationsgesetz – JKomG), BGBl. I vom 29. 3. 2005, S. 837; dazu *Viefhues*, NJW 2005, 1009.
136 S. bei Fn. 29.

Klicken auf einen Button zu, so kann er letztlich nicht sicher sein, dass die Informationen, die ihm am Bildschirm angezeigt wurden, auch tatsächlich mit jenen übereinstimmen, die der Vertragspartner an den PC des Nutzers übermittelt hat. Auch kann der Vertragspartner nicht sicher sein, dass der Nutzer tatsächlich auf den Zustimmungs-Button geklickt hat. Diese Unsicherheit kann durch Trusted-Computing-Plattformen beseitigt werden, da sie über sichere Kommunikationswege zu Ein- und Ausgabegeräten verfügen.[137] Dies könnte auch Auswirkungen auf die Beweiswürdigung solcher Vorgänge im Prozess haben.

V. Schlussbemerkungen

Noch ist vieles im Bereich des Trusted Computing Zukunftsmusik. Die technischen Grundlagen sind im Fluss, die breite Anwendung der Technologie noch mindestens zwei bis drei Jahre entfernt. Manche der beschriebenen Probleme sind keine Spezifika des Trusted Computing. Sie sind auch schon in anderen Kontexten aufgetaucht. Dennoch ist es aus drei Gründen lohnend, sich zum jetzigen Zeitpunkt mit dem Trusted Computing zu beschäftigen.

Erstens bietet Trusted Computing im Vergleich zu früheren Sicherheitstechnologien ein sehr viel höheres Sicherheitsniveau. Dadurch verstärkt sich der Konflikt zwischen Technik und öffentlichen Wertvorstellungen merklich. *Zweitens* ist es ein Ziel des Trusted Computing, eine allumfassende Infrastruktur aufzubauen, die eine möglichst große Verbreitung finden soll. Im Idealfall soll diese Infrastruktur nicht nur PCs, sondern auch andere Geräte wie PDAs, Handys, Spielekonsolen, Netzwerkkomponenten im Internet und in Mobilfunknetzwerken erfassen. Während viele der dargestellten Probleme früher in kleinen, eng umgrenzten Bereichen der IT-Umgebung aufgetreten sein mögen, könnten sie mit Hilfe des Trusted Computing alle Bereiche unserer zukünftigen IT-Umgebung erfassen. Technologien mit dieser Tragweite müssen notwendigerweise kritisch begleitet werden.[138] *Drittens* zeigen die Diskussionen um das Trusted Computing, dass viele der dargestellten Probleme durch ein geschicktes Design der technischen Architektur oder des institutionellen Arrangements gelöst werden können, das Trusted Computing umgibt. Da sich Trusted-Computing-Architekturen noch im Entwicklungsstadium befinden, bestehen auch realistische Chancen, dass die Architekturen derart beeinflusst werden können, dass sie eine neutrale Infrastruktur ermöglichen, die Wettbewerb sichert und Datenschutzinteressen hinreichend beachtet. Die Rechtswissenschaft sollte diese Chance einer rechtlichen Technikgestaltung nicht versäumen.

137 S. *Abadi/Wobber*, a.a.O. Fn. 21, S. 9 f.
138 Zu dieser „Peril of Pervasiveness“ s. *Bechtold*, a.a.O. Fn. 9, S. 650.