Trusted Computing and Digital Rights Management Clearinghouse

A Thesis

Presented in Partial Fulfillment of the Requirements for

the Degree Bachelor of Science with Distinction from

The Ohio State University

by

Adam C. Champion

* * * * *

The Ohio State University

2007

Honors Research Committee:

Professor Bruce W. Weide, Adviser

Professor Dong Xuan

Approved by

Adviser

Department of Computer Science and Engineering

ABSTRACT

Presently, security threats from computer viruses to malicious intrusion imperil computer users. To mitigate and counter these threats, a consortium of hardware and software vendors and consumer-electronics manufacturers are implementing "trusted computing" (TC) standards on present and future computer systems. These standards define security policies that can defend computer systems against attacks. However, in their zeal to protect intellectual property, the content and computer industries plan to use TC technologies in ways that will likely infringe upon users' civil liberties and hamper innovation. Here, details of these technologies and their uses and misuses are explained. An overview of technologies on which TC depends is presented, followed by an explanation of exactly what TC *is*. Interactions between TC-supported computer hardware, operating systems, and software are explored, as are the potential impacts on civil liberties and the economy. Finally, present and future TC computer systems are analyzed and reasonable policy suggestions are offered.

Trusted Computing and Digital Rights Management Clearinghouse

by

Adam C. Champion

The Ohio State University, 2007

Professor Bruce W. Weide, Adviser

Presently, security threats from computer viruses to malicious intrusion imperil computer users. To mitigate and counter these threats, a consortium of hardware and software vendors and consumer-electronics manufacturers are implementing "trusted computing" (TC) standards on present and future computer systems. These standards define security policies that can defend computer systems against attacks. However, in their zeal to protect intellectual property, the content and computer industries plan to use TC technologies in ways that will likely infringe upon users' civil liberties and hamper innovation. Here, details of these technologies and their uses and misuses are explained. An overview of technologies on which TC depends is presented, followed by an explanation of exactly what TC *is*. Interactions between TC-supported computer hardware, operating systems, and

software are explored, as are the potential impacts on civil liberties and the economy. Finally, present and future TC computer systems are analyzed and reasonable policy suggestions are offered.

ACKNOWLEDGMENTS

Thanks to Professors Weide and Xuan for helping me write this paper, the OSU Engineering Honors Committee for making it possible, and my parents for always supporting me along the way.

Thanks to Professor Ross Anderson of England's University of Cambridge, who first "sounded the alarm" about Trusted Computing (TC) in "Security in Open versus Closed Systems: The Dance of Boltzmann, Coase and Moore" and his TC FAQ, which first piqued my interest in the topic. In addition, Bruce Schneier's analyses of TC and other computer security matters were very helpful, as were other, myriad online sources.

Legal Notices: Companies frequently register the names of their products and services as trademarks to differentiate them from competitors' products and services. To the extent that the author is aware of a trademark claim, these names are printed in Initial Caps or ALL CAPS. Interested readers, however, should contact the appropriate companies regarding trademarks and registration.

In addition, while the author has taken great care to ensure the accuracy of all information presented herein, he offers no warranty about its future accuracy as TC technology advances.

VITA

September 11, 1983	Born – Columbus, OH, USA
June 2002	High School Diploma – Worthington Kilbourne High School, W. Worthington, OH, USA

FIELDS OF STUDY

Major Field: Computer Science and Engineering

Studies in:

Computer and Network Security	Dong Xuan, Anish Arora
Computer Graphics	Han-Wei Shen, Naeem Shareef

TABLE OF CONTENTS

Page

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

CHAPTER 1

INTRODUCTION

Today, computer users face a variety of security threats. E-mails contain malicious software programs, or "malware," that can delete important files and e-mail themselves to everyone in address books [154]. "Phishing" scams, which are fraudulent web pages and emails that claim to come from a legitimate business, entice users to enter personal information, which is then stolen. "Spyware" programs surreptitiously change browser settings, display unwanted advertising, and record private information and send it to third parties without authorization. Even mobile phones are not immune from malware: some programs "[send] a stream of text messages, at. . .$5 each, to a [Russian phone number]" or "disable [the] phone entirely" [154]. The situation has grown so dire that computer-security expert Steve Gibson wrote, "It appears that some end-user freedoms will ultimately be diminsihed in order to thwart the abusive interests of commercial entities [that produce and distribute spyware]" [125].

To mitigate and counter these threats, the Trusted Computing Group (TCG), a consortium of hardware and software vendors and consumer-electronics manufacturers, has proposed and implemented "trusted computing" (TC) standards. These standards specify hardware and software policies that can greatly enhance

the security of computer systems. TC hardware has shipped in over 20 million new PCs, and will be ubiquitous in hardware and software by the beginning of 2007 [258]. However, in their zeal to protect intellectual property, the content and computer industries plan to use TC technologies in ways that drastically "[curtail] end-user freedoms" and shackle economic innovation [125]. Compounding the problem is the lack of thorough, technically accurate, and unbiased literature on the topic.

This paper aims to "bridge the gap" between proponents and opponents of TC technology by explaining just what TC *is*, its relationship to intellectual-property law, its uses and possible abuses, and possible social and economic consequences. In addition, suggestions for policymakers and consumers are included. The paper is structured as follows:

- Chapter 2, "TC 'Meta-Technologies,'" explains the concepts of public-key cryptography, which are essential to TC, and intellectual property, whose protection is the primary impetus for the development of TC technology. In particular, the 1998 U.S. Digital Millenium Copyright Act (DMCA) and Sonny Bono Copyright Term Extension Act are examined. The chapter concludes with a description of digital rights management (DRM) technology, which "ties together" the two aforementioned concepts.

- Chapter 3, "TC Overview," precisely defines "trusted computing" and its component technologies: sealed storage, memory curtaining, secure input and output, and remote attestation.

- Chapter 4, "TC Hardware," examines the Trusted Platform Module (TPM), the hardware component of TC. The TPM provides a "root of trust" that provides tamper evidence to a TC operating system and software. The PC basic input/output system (BIOS) allows users to turn TPMs on and off, as does the Extensible Firmware Interface (EFI), Intel's successor to the BIOS. Chapter 4 also explores other "protective" hardware technologies similar to the TPM, such as IBM's SecureBlue "encrypted chip" and Intel's High-Definition Multimedia Interface (HDMI) [135, 158].

- Chapter 5, "TC Operating Systems," analyzes operating systems that support TC technologies, especially Windows Vista on PCs. Microsoft supports TC *very* extensively in Windows Vista, as will be discussed. On Intel Macintosh computers, Mac OS X *does not* use the TPM, though users can enable it via "open source" device drivers [291].[1] GNU/Linux (commonly referred to "Linux") supports user-configurable TPM access and use.

- Chapter 6, "TC Software," examines software that supports TC functionality. At present, not much software does, with the exception of TPM utilities for Windows, restricted-access Microsoft Office and Adobe Portable Document Format (PDF) files, and Microsoft's BitLocker Drive Encryption for Windows Vista. However, it is only a matter of time before software uses TC functionality.

[1]When Apple "switched" to Intel CPUs for the Macintosh, it was reported that the company used TPMs to "lock" OS X to Apple hardware, per the license agreement [17, 22, 156]. However, according to Amit Singh, author of the book *Mac OS X Internals*, Apple did *not* do so [291]. Further details are in Chapter 5.

- Chapter 7, "Other TC Applications," explores the use of TC technologies on other computing platforms such as mobile phones and video-game consoles.

- Chapter 8, "Effects on Civil Liberties and the Economy," considers how uses of TC technology can enhance computer security and how abuses thereof can greatly restrict computer users' civil liberties and stifle economic innovation. Given industry trends, several key abuses of TC and the "post-TC" computing markets are emphasized.

- Chapter 9, "Policy Suggestions,", offers several suggestions to rectify the current imbalance between protecting the rights of intellectual-property holders and consumers. This chapter also analyzes the current "post-TC" market situation and provides tips to help consumers with buying decisions.

- Finally, Chapter 10, "Conclusions," summarizes the arguments above.

Of course, the rapid pace of computing advances may render some of these arguments obsolete after publication. However, the fundamentals of TC and its *probable* applications will likely remain constant even as new products and services enter the marketplace. This guide is designed to help the technically-savvy reader understand profound developments that could make "the open, unrestricted PC. . .a thing of the past," thereby facilitating sensible decision-making in the post-TC computing market [337].

CHAPTER 2

TRUSTED COMPUTING (TC) "META-TECHNOLOGIES"

Implementation of TC requires several other technologies, especially public-key cryptography,[2]and a principal application of TC is enhanced enforcement of digital rights management (DRM) technology. As will be discussed, cryptography secures messages against eavesdroppers, and DRM uses cryptography to restrict the use of files, thereby ensuring that they are used only in accord with their respective intellectual-property (IP) rights-holders' wishes [265]. Since TC relies on public-key cryptography and can tightly enforce DRM, these latter technologies are termed *TC "meta-technologies"* in this paper. This chapter examines these "meta-technologies" in depth, as well as present IP law.

2.1 Public-Key Cryptography

Public-key cryptography lies at the heart of TC, as will be shown in Chapter 4. Before delving into details of cryptography, however, it is instructive to review exactly what cryptography *is*, as well as its purpose—both in general and with

[2]Cryptography is not the *only* technology required for TC implementation. In particular, TC relies on the existence of a special motherboard microcontroller, the Trusted Platform Module (TPM), to securely store cryptographic keys [312]. In addition, developers of operating systems (OSes) and other software must consider how to implement TC at the OS and software level, respectively. A discussion of TC hardware is deferred to Chapter 4, whereas discussions of TC OSes and software are deferred to Chapters 5 and 6, respectively.

respect to TC. This explanation of cryptography, algorithms, and protocols is culled from Bruce Schneier's acclaimed book *Applied Cryptography*. Initially, Schneier explains that cryptography obfuscates a message sent from a sender to a receiver to make it unreadable by eavesdroppers (emphasis in original) [265]:

> A message is *plaintext* (sometimes called cleartext). The process of disguising a message. . .to hide its substance is *encryption*. An encrypted message is *ciphertext*. The process of turning ciphertext into plaintext is *decryption*.. . . The art and science of keeping messages secure is *cryptography*, and it is practiced by *cryptographers*. *Cryptanalysts* are practitioners of *cryptanalysis*, the art and science of breaking ciphertext. . . . The branch of mathematics encompassing both cryptography and cryptanalysis is *cryptology* and its practitioners are *cryptologists*. Modern cryptologists are generally trained in theoretical mathematics—they have to be.
>
> Plaintext is denoted by M, for message, or P, for plaintext. It can be. . .[any sort of] binary data.. . . Ciphertext is denoted by C [and] it is also binary data . . . . The encryption function E. . .operates on M to produce C. Or, in mathematical notation:
>
> $$E(M) = C. \tag{2.1}$$
>
> In the reverse process, the decryption function D operates on C to produce M:
>
> $$D(C) = M. \tag{2.2}$$
>
> Since the whole point of encrypting and then decrypting a message is to recover the original plaintext, the following identity must hold true:
>
> $$D(E(M)) = M. \tag{2.3}$$

Aside from ensuring the message's secrecy, cryptography also has as objectives authentication, integrity, and nonrepudiation (emphasis in original) [265]:

- *Authentication.* It should be possible for the receiver of a message to ascertain its origin; an intruder should not be able to masquerade as someone else.

- *Integrity.* It should be possible for the receiver of a message to verify that it has not been modified in transit; an intruder should not be able to substitute a false message for a legitimate one.
- *Nonrepudiation.* A sender should not be able to falsely deny later that he sent a message.

These [are exactly analogous to similar proofs in] face-to-face interactions. . .that someone is who he says he [is,] that [his] credentials. . .are [valid, and] that a document purporting to come from a person actually came from that person.

Next, Schneier provides a high-level overview of cryptographic algorithms and keys before differentiating between symmetric and asymmetric, *i.e.*, public-key, cryptographic algorithms (emphasis in original) [265].

A *cryptographic algorithm*, also called a *cipher*, is the mathematical function used for encryption and decryption. (Generally, there are two related functions: one for encryption and the other for decryption.)

If the security of an algorithm is based on keeping the way that algorithm works a secret, it is a *restricted* algorithm. [However, restricted algorithms have substantial flaws; in particular, the security of these algorithms depends on their details.]. . .Modern cryptography [works around these flaws] with a *key*, denoted by [K, a parameter to the algorithm that may be] one of a large number of values. The range of possible [key values] is called the *keyspace*. Both the encryption and decryption operations use this key, [which] is denoted by the K subscript. . . , so the functions. . .become:

$$E_K(M) = C \tag{2.4}$$
$$D_K(C) = M, \tag{2.5}$$

[such that]

$$D_K(E_K(M)) = M. \tag{2.6}$$

. . .[In] some algorithms . . . , the encryption key, K_1, is different from the corresponding decryption key, K_2. In this [case, the above equations become]:

$$E_{K_1}(M) = C \tag{2.7}$$
$$D_{K_2}(C) = M \tag{2.8}$$
$$D_{K_2}(E_{K_1}(M)) = M. \tag{2.9}$$

[In contrast to restricted algorithms,] all of the security in [the above] algorithms is based in the key (or keys); none is based in the details of the algorithm.. . .A *cryptosystem* is an algorithm, plus all possible plaintexts, ciphertexts, and keys.. . .

Symmetric algorithms. . .are algorithms where the encryption key can be calculated from the decryption key and vice versa. In most symmetric algorithms, [such as those shown above in Equations 2.4 through 2.6,] the encryption key and decryption key are the same. These algorithms. . .require that the sender and receiver agree on a key before they can communicate securely. The security of a symmetric algorithm rests in the key.. . .Symmetric algorithms can be divided into two categories. Some operate on the plaintext a single bit (or sometimes byte) at a time; these are called *stream algorithms* or *stream ciphers*. Others operate on the plaintext in groups of bits. The groups of bits are called *blocks*, and the algorithms are called *block algorithms* or *block ciphers*.. . .[On the other hand,] *public-key algorithms* (also called asymmetric algorithms), are designed so that the key used for encryption is different from the key used for decryption. Furthermore, the decryption key cannot (at least in any reasonable amount of time) be calculated from the encryption key. The algorithms are called "public-key" because the encryption key can be made public: A complete stranger can use the encryption key to encrypt a message, but only a specific person with the corresponding decryption key can decrypt the message. In these systems, the encryption key is often called the *public key*, and the decryption key is often called the *private key*.. . .Encryption using public key K is denoted by:

$$E_K(M) = C. \tag{2.10}$$

Even though the public key and private key are different, decryption with the corresponding private key is denoted by:

$$D_K(C) = M. \tag{2.11}$$

In addition, Schneier considers cryptanalysis and strategies to extract keys from encrypted messages, which are shown below. These strategies hold for both symmetric and public-key algorithms, and will be explained further with respect to public-key cryptographic protocols (emphasis in original) [265].

> The whole point of cryptography is to keep the plaintext (or the key, or both) secret from eavesdroppers (also called. . .attackers [or] opponents. . .). Eavesdroppers are assumed to have complete access to the communications between the sender and receiver.
>
> Cryptanalysis is the science of recovering the plaintext of a message without access to the key. Successful cryptanalysis may recover the plaintext or the key. . . .An attempted cryptanalysis is called an *attack*. A fundamental assumption in cryptanalysis, first enunciated by the Dutchman A. Kerckhoffs in the [19th] century, is that the secrecy must reside entirely in the key. Kerckhoffs assumes that the cryptanalyst has complete details of the cryptographic algorithm and implementation. . . .[Though this is not always the case, it is a useful assumption:] if others can't break an algorithm, even with knowledge of how it works, than they certainly won't be able to break it without that knowledge.
>
> There are [five basic] types of cryptanalytic [attacks, all of which make the above assumption].
>
> 1. *Ciphertext-only attack.* The cryptanalyst has the ciphertext of several messages, all of which have been encrypted using the same. . . algorithm. [His] job is to recover the plaintext of as many messages as possible, [or, better yet,] deduce the [key(s)] used to encrypt the messages. . . . [That is, given $C_1 = E_K(P_1), C_2 = E_K(P_2), \ldots, C_i = E_K(P_i)$, deduce K from $P_1, P_2, \ldots, P_i$] or an algorithm to infer P_{i+1} from [$C_{i+1} = E_K(P_{i+1})$].
> 2. *Known-plaintext attack.* The cryptanalyst has access not only to the ciphertext of several messages, but also to the plaintext of those messages. His job is to deduce the [key(s)] used to encrypt the messages or an algorithm to decrypt any new messages encrypted with the same [key(s)]. [That is, given $P_1, C_1 = E_K(P_1), P_2, C_2 = E_K(P_2), \ldots, C_i = E_K(P_i)$, deduce K] or an algorithm to infer P_{i+1} from [$C_{i+1} = E_K(P_{i+1})$].
> 3. *Chosen-plaintext attack.* The cryptanalyst not only has access to the ciphertext and associated plaintext for several messages, but he also chooses the plaintext that gets encrypted. . . .His job is to

deduce the [key(s)] used to encrypt the messages or an algorithm to decrypt any new messages encrypted with the same [key(s)]. [That is, given $P_1, C_1 = E_K(P_1), P_2, C_2 = E_K(P_2), \ldots, C_i = E_K(P_i)$], where the cryptanalyst [can] chose $P_1, P_2, \ldots, P_i$, [deduce K] or an algorithm to infer P_{i+1} from [$C_{i+1} = E_K(P_{i+1})$].

4. *Adaptive-chosen-plaintext attack.* This is a special case of a chosen plaintext attack [in which] the cryptanalyst [can] choose the plaintext that is encrypted [and] he can modify his choice based on the results of previous encryption.
5. *Chosen-ciphertext attack.* The cryptanalyst can choose different ciphertexts to be decrypted and [can view] the decrypted plaintext. . . to deduce the key. [That is, given $C_1, P_1 = D_K(C_1), C_2, P_2 = D_K(C_2), \ldots, C_i, P_i = D_K(C_i)$, deduce K].

Having provided an overview of cryptographic algorithms and attacks against them, Schneier explains elementary cryptographic protocols (emphasis in original) [265].

> A *protocol* is a series of steps, involving two or more parties, designed to accomplish a task. [Decomposing this definition yields the following properties:]
>
> - "A series of steps" means that the protocol has a sequence, from start to [finish, just like an algorithm]. Every step must be executed in turn, and no step can be taken before the previous step is finished.
> - "Involving two or more parties" means that least two people are required to complete the protocol; one person alone does not make a protocol. A person alone can perform a series of steps to accomplish a task (like baking a cake), but this is not a protocol. (Someone else must eat the cake to make it a protocol.). . .
> - "[Designed] to accomplish a task" means that the protocol must achieve something. Something that looks like a protocol but [that] does not accomplish a task is not a protocol—it's a waste of time.
>
> Protocols have other characteristics as well:
>
> - Everyone involved in the protocol must know the protocol and all of the steps to follow. . . .
> - Everyone involved in the protocol must agree to follow it.

Alice	First participant in all the protocols
Bob	Second participant in all the protocols
Carol	Participant in the three- and four-party protocols
Dave	Participant in the four-party protocols.
Eve	Eavesdropper
Mallory	Malicious active attacker
Trent	Trusted arbitrator.

Table 2.1: Actors in Protocols. (Excerpted from Table 2.1 in [265].)

- The protocol must be unambiguous; each step must be well defined and there must be no chance of a misunderstanding.
- The protocol must be complete; there must be a specified action for every possible situation.. . .

A *cryptographic protocol* is a protocol that uses cryptography. The parties can be friends. . .or adversaries. . . . A cryptographic protocol involves some cryptographic algorithm [to accomplish a goal] beyond simple [secrecy, such as convincing] one another of [his or her] identity. . .or [simultaneously] signing a contract. The whole point of using cryptography in a protocol is to prevent or detect eavesdropping and cheating. [This leads to the final protocol characteristic:]

- It should not be possible to do more or learn more than what is specified in the protocol.. . .

[The following individuals shown in Table 2.1] help demonstrate protocols. Alice and Bob. . .will perform all general two-person protocols. As a rule, Alice [initiates] all protocols and Bob [responds]. If the protocol requires a third or fourth person, Carol and Dave [perform] those [respective] roles. [As will be explained, Eve is a passive eavesdropper, Mallory is a malicious active attacker, and Trent is a trusted arbitrator.]

[There are three types of protocols: arbitrated protocols, adjudicated protocols, and self-enforcing protocols. Arbitrated protocols rely on an *arbitrator*, a disinterested, trusted third party,] to complete the protocol. Disinterested means that the arbitrator has no vested interest in the protocol and no particular allegiance to any of the parties involved. Trusted means that all people involved in the protocol accept what he says as true, what he does as correct, and that he will complete his part of the protocol.

[A lawyer is a "real-life" example of an arbitrator.] For example, Alice is selling a car to Bob, a stranger. Bob wants to pay by check, but Alice has no way of knowing if the check is good. Alice wants the check to clear before she turns the title over to Bob. Bob, who doesn't trust Alice any more than she trusts him, doesn't want to hand over a check without receiving a title.. . .[If a lawyer trusted by both parties enters the situation], Alice and Bob can use the following protocol to ensure that neither cheats the other.

1. Alice gives the title to the lawyer.
2. Bob gives the check to Alice.
3. Alice deposits the check.
4. After waiting a specified time period for the check to clear, the lawyer gives the title to Bob. If the check does not clear within the specified time period, Alice shows proof of this to the lawyer and the lawyer returns the title to Alice.

In this protocol, Alice trusts the lawyer not to give Bob the title unless the check has cleared, and to give it back to her if the check does not clear. Bob trusts the lawyer to hold the title until the check clears, and to give it to him once it does. The lawyer doesn't care if the check clears. He will do his part of the protocol in either case, because he will be paid in either case.. . .[While computers can be arbitrators, difficulties can arise: people are naturally wary of a non-human arbitrator, network overhead and delay increase, the arbitration of each transaction can degrade protocol performance, and the arbitrator is a target for any malicious attacker. Still, trusted arbitrators play roles in protocols, and in these protocols, Trent plays this role.]

[Alternatively,] arbitrated protocols can be subdivided into two lower-level *subprotocols*. One is a nonarbitrated subprotocol, executed every time parties want to complete the protocol. The other is an arbitrated subprotocol, executed only in exceptional circumstances—when there is a dispute. This. . .type of arbitrator is called an *adjudicator*.. . .[For instance,] judges are professional adjudicators [that are] brought in only if there is a dispute. Alice and Bob can enter into a contract without a judge[, who] never sees the contract until one of them [initiates legal action against the other].. . .

[Adjudicated] computer protocols. . .rely on the [parties' honesty;] but if someone suspects cheating, [a trusted adjudicator can examine available data to] determine if someone cheated. In a good adjudicated protocol, the adjudicator could also determine the cheater's [identity, thus deterring cheating.]

> [The final type of protocol is the] *self-enforcing protocol*, [which guarantees fairness due to its "dispute-proof" construction.[3]

Trent is not the only "dramatis persona" Schneier uses to explain protocols; Eve and Mallory eavesdrop on and actively attack protocols, as is explained below (emphasis in original) [265].

> People can [attack a protocol in various ways]. Someone not involved in the protocol can eavesdrop on some or all of the protocol. This is called a *passive attack*, because the attacker does not affect the protocol. All he can do is observe the protocol and attempt to gain [information, as in a ciphertext-only attack]. . . . Since passive attacks are difficult to detect, protocols try to prevent passive attacks rather than detect them. . . .[Eve plays the role of the eavesdropper.]
>
> Alternatively, an attacker could try to alter the protocol to his own advantage. He could pretend to be someone else, introduce new messages in the protocol, delete existing messages, substitute one message for another, replay old messages, interrupt a communications channel, or alter stored information in a computer. These are called *active attacks*, because they require active intervention, . . .and are much more serious [than passive attacks], especially in protocols in which [parties do not] trust one another. The attacker does not have to be [an outsider:] he could be a legitimate system [user, even] the system [administrator, or there might be] multiple active attackers working together. Mallory plays the role of the malicious active attacker [265].

Public-key cryptography relies heavily on *one-way functions*, mathematical functions that are "easy" to compute "in one direction" and "hard" to compute "in the other direction." Specifically, it relies on *trapdoor one-way functions*, one-way functions with a "secret" means of computing in the latter direction, as Schneier explains (emphasis in original) [265].

> The notion of a *one-way function* is central to public-key cryptography. . . .One-way functions are relatively easy to compute, but significantly harder to reverse. That is, given x it is easy to compute $f(x)$, but

[3] Adjudicated and self-enforcing protocols are not discussed further in this paper, as they are not used in TC implementations.

given $f(x)$ it is hard to compute x, [*i.e.*, it would take millions of years to compute x from $f(x)$.]

Breaking a plate is a good example of a one-way function. It is easy to smash a plate into a thousand tiny pieces. However, it's not easy to put all of those tiny pieces back together into a plate.[4]. . .[Unfortunately,] we can't use [one-way functions to encrypt messages], as no one could decrypt [such a message].. . .A *trapdoor one-way function* is a special type of one-way function, one with a secret trapdoor. [Like a one-way function,] it is easy to compute in one direction and hard to compute in the other [direction, *i.e.*,] it is easy to compute $f(x)$ given x, and hard to compute x given $f(x)$. However, there is some secret [information y such that] given $f(x)$ and y it is easy to compute x.

Taking a watch apart is a good example of a trapdoor one-way [function:] it is easy to disassemble a watch into hundreds of [tiny pieces, but] it is very difficult to put those tiny pieces back together into a working watch. However, with the. . .assembly instructions of the [watch,] it is much easier to put the watch back together. [Another example of a trapdoor one-way function is the following: $C = E(M) = M^e \bmod n$ and $M = D(C) = C^d \bmod n$, where $n = p \cdot q$. M is the numeric representation of a message, C is its corresponding ciphertext, e is a publicly-specified numeric exponent, d is a secret numeric exponent, n is a publicly-specified divisor, and p and q are very large, secret prime numbers [255]. "mod" denotes the modulus operator; for integers u, v, w, and x where $u \times v + w = x$, $w \equiv x \bmod u$. This trapdoor one-way function is used in the RSA public-key cryptosystem, which is used in the TPM [312]. The security of the algorithm is based "in part on the difficulty of factoring the published divisor n" into p and q [255].]

In addition, Schneier details how public-key cryptography also depends on *one-way hash functions*, mathematical functions that map input strings to "almost" unique output strings. One-way hash functions are frequently used with secret keys to form *message authentication codes*, or *MACs* (emphasis in original) [265].

A hash function is a function, mathematical or otherwise, that takes a variable-length input string (called a *pre-image*) and converts it to a fixed-length (generally smaller) output string (called a *hash value*). A simple

[4]Schneier gives another example of a one-way function: $f(x) = x^2$ (and, hence, $f^{-1}(x) = \pm\sqrt{x}$) [265]. It is easy to compute $f(x)$, but it is significantly harder to compute $f^{-1}(x)$ [265]. However, in this example, the computation of $f^{-1}(x)$ is not sufficiently difficult for cryptographic purposes!

hash function [is one that returns a single byte that is the exclusive-or (XOR) of all the pre-image bytes].

[The purpose of hash functions] is to. . .produce a value that indicates whether a candidate pre-image is likely to be the same as the real pre-image. [Though hash functions are not one-to-one, they are used] to get a reasonable assurance [that two strings are equal to each other].

A *one-way hash function* is a hash function that works in one direction: It is easy to compute a hash value from a pre-image, but it is hard to generate a pre-image that hashes to a particular value. [The aforementioned XOR hash function is not one-way, as one can easily generate bytes that yield a given value when XORed together.] A good one-way hash function is also *collision-free*: It is hard to generate two pre-images with the same hash value.

[The one-way hash functions are publicly accessible, and their security is based on their one-wayness:] The output is not [discernibly] dependent on the [input, and] changing a single pre-image bit changes, on. . .average, half the bits in the hash value. [Moreover,] given a hash value, it is computationally [infeasible] to find a pre-image that hashes to that value. [In general, hash functions are used to verify files' authenticity when it is too expensive or inconvenient to send them over a network.][5]

A *message authentication code* (MAC), also known as a data authentication code (DAC), is a one-way hash function with the addition of a secret [key. . .that works just like a "normal" hash function,] except only someone with the key can verify the hash value.[6]

Having discussed one-way functions and hash functions, Schneier describes public-key cryptographic communications, including "hybrid cryptosystems" that combine elements of symmetric and public-key cryptography (emphasis in original) [265].

In 1976, Whitfield Diffie and Martin Hellman. . .described *public-key cryptography*. They used two different keys—one public and the other private. It is computationally hard to deduce the private key from the

[5]TPMs use the National Institute of Science and Technology's (NIST's) Secure Hash Algorithm 1 (SHA-1) [312]. Details are presented in Chapter 4 as well as in [68, 235].

[6]TPMs use keyed-hash message authentication codes (HMAC) to validate TPM commands, as explained in Chapter 4 [312].

public key. Anyone with the public key can encrypt a message but not decrypt [it;] only the person with the private key can decrypt the message.. . .

Mathematically, the process is based on the trap-door one-way functions previously discussed. Encryption is the easy direction. Instructions for encryption are the public key; anyone can encrypt a message. Decryption is the hard direction. It [is] hard enough that people with [supercomputers] and [thousands] of years cannot decrypt the message without. . .[the private key, which is the "secret" or "trapdoor."] With [it], decryption is as easy as encryption.

This is how Alice [sends] a message to Bob using public-key cryptography:

1. Alice and Bob agree on a public-key cryptosystem.
2. Bob sends Alice his public key.
3. Alice encrypts her message using Bob's public key and sends it to Bob.
4. Bob decrypts Alice's message using his private key.. . .

[Remark that] Alice can send a secure message to Bob. Eve, listening in on the entire exchange, has Bob's public key and a message encrypted in that key, but cannot recover either Bob's private key or the message.

More commonly, a network of users agrees on a public-key cryptosystem. Every user has [her] own public key and private key, and the public keys are [stored] in a database. . . . Now the protocol is even easier:

1. Alice gets Bob's public key from the database.
2. Alice encrypts her message using Bob's public key and sends it to Bob.
3. Bob then decrypts Alice's message using his private key.. . .

[However, in practice,] public-key algorithms are not a substitute for symmetric algorithms. [They are used to encrypt keys, not messages, for the following reasons:]

1. Public-key algorithms are [slow, at least 1000 times slower than symmetric algorithms].
2. Public-key cryptosystems are vulnerable to chosen-plaintext attacks. If $C = E(P)$, [where P is one out of n possible plaintexts,] then a cryptanalyst [need only] encrypt all n possible plaintexts and compare the results with C [to determine P].. . .

In most practical implementations public-key cryptography is used to secure and distribute [*session keys*,]. . .which are used with symmetric algorithms to secure message traffic. This is [also] called a [*hybrid cryptosystem*, and is detailed here.]

1. Bob sends Alice his public key.
2. Alice generates a random session key, K, encrypts it using Bob's public key, and sends it to [Bob, *i.e.*,] $E_B(K)$.
3. Bob decrypts Alice's message using his private key to recover the session [key, *i.e.*,] $D_B(E_B(K)) = K$.
4. Both of them encrypt their communications using the same session key.

[Because the session key is destroyed after Alice and Bob use it, the likelihood of Eve accessing it is reduced.]

Public-key cryptography facilitates the use of digital signatures, as Schneier explains below [265].

[A handwritten signature has the following desirable attributes:]

1. The signature is [authentic; it] convinces the document's recipient that the signer deliberately signed the document.
2. The signature is [unforgeable; it] is proof that the signer, and no one else, has deliberately signed the document.
3. The signature is not [reusable; it] is part of the [document and no one can] move [it] to a different document.
4. The signed document is [unalterable; after] the document is signed, it cannot be altered.
5. The signature cannot be [repudiated; as] the signature and the document are physical [things, the] signer cannot later claim that he or she didn't sign it.

[Of course, these properties do not always hold in reality.] Signatures can be forged,. . .lifted from one piece of paper and moved to another, and documents can be altered after signing. However, we are willing to live with these problems because of the difficulty in cheating and the risk of detection.

[It is much more difficult to ensure that computer-based "signatures" have these properties, primarily due to the ease of copying and modifying computer files.. . .However, public-key cryptography provides an elegant solution to this problem with the following protocol:]

1. Alice encrypts the document with her private key, thereby signing the document.
2. Alice sends the signed document to Bob.
3. Bob decrypts the document with Alice's public key, thereby verifying the signature.[7]

This protocol [has the aforementioned attributes:]

1. The signature is authentic; when Bob verifies the message with Alice's public key, he knows that she signed it.
2. The signature is unforgeable; only Alice knows her private key.
3. The signature is not reusable; the signature is a function of the document and cannot be transferred to any other document.
4. The signed document is unalterable; if there is any alteration to the document, the signature can no longer be verified with Alice's public key.
5. The signature cannot be repudiated. Bob doesn't need Alice's help to verify her signature.

[Cheating can occur in this protocol. For instance, Bob] can reuse the document and signature together. This is no problem if Alice signed a contract. . . , but it can [cause serious problems] if Alice signed a digital check.. . .[Suppose] Alice sends Bob a signed digital check for $100. Bob takes the check to the bank, which verifies the signature and moves the money from one account to another. [Bob saves a copy of the digital check and does the same thing the next week.] If Alice never balances her checkbook, [Bob could clean out her account! To prevent this sort of unscrupulous behavior,] digital signatures often include timestamps. The date and time of the signature are attached to the message and signed along with the rest of the message. [In this example, the bank stores this timestamp in a database. If Bob tries to cash Alice's check a second time,] the bank checks the timestamp against its database. Since

[7]In this protocol, Alice signs the document by encrypting it with her private key, and Bob verifies her signature by decrypting the document with her public key [265]. However, the terms "encrypting with a private key" and "decrypting with a public key" are synonymous with "signing the document" and "verifying the signature" *only* in the RSA algorithm (emphasis added) [265]. There are many other methods that use public-key cryptography to sign documents and verify signatures whose protocols and implementations thereof differ from RSA's [265]. Following Schneier, this paper notates "signing a message M with private key K" and "verifying a signature with the *corresponding* public key K" as $S_K(M)$ and $V_K(M)$, respectively (emphasis added) [265]. The former operation is termed "signing a document (or message)," and the latter operation is termed "verifying a document (or message)."

the bank already cashed a check from Alice with the same timestamp, the bank calls the [police, who arrest and imprison him.][8]

[In practice], public-key algorithms are often too inefficient to sign long documents. To save time, digital signature protocols are often implemented with one-way hash functions. Instead of signing a document, Alice signs the hash of the document. In this protocol, [Alice and Bob agree on the one-way hash function and digital signature algorithm] beforehand.

1. Alice produces a one-way hash of a document.
2. Alice. . .[signs] the hash.
3. Alice sends the document and the signed hash to Bob.
4. Bob produces a one-way hash of the document that Alice sent.. . . [Using the digital signature algorithm, he verifies] the hash. If the signed hash matches his generated hash, the signature is valid.

[This technique greatly increases speed] and, since the [odds] of two different documents having the same. . .hash are [very small], anyone can safely equate a signature of the hash with a signature of the document.[9]. . . Moreover, [combining] digital signatures with public-key cryptography [yields a protocol with] the security of encryption [and] the authenticity of digital signatures. [Note that A and B denote Alice and Bob's keys, respectively.]

1. Alice signs the message with her private [key, *i.e.*,] $S_A(M)$.
2. Alice encrypts the signed message with Bob's public key and sends it to [Bob, *i.e.*,] $E_B(S_A(M))$.
3. Bob decrypts the message with his private [key, *i.e.*,] $D_B(E_B(S_A(M))) = S_A(M)$.
4. Bob verifies with Alice's public key and recovers the [message, *i.e.*,] $V_A(S_A(M)) = M$.

[8]Using timestamps also helps mitigate Alice's repudiation of her signature, *i.e.*, Alice signing a message and later claiming she never did so [265]. The corresponding protocol is similar to that shown above, except that Alice "generates a header containing some identifying [information,] concatenates the header with the signed message, signs *that*, and sends it to Trent. Trent verifies the outside signature, confirms [Alice's header,] adds a timestamp to Alice's signed message and [header,] signs [everything,] and sends it to both Alice and Bob. Bob verifies [Trent and Alice's signatures and Alice's header,] and Alice verifies the message Trent sent to Bob. If she did not [send the message, she sounds an alarm]" (emphasis added) [265].

[9]This assumption only holds if the hash is long enough. For instance, SHA-1 produces a 160-bit hash, so if Alice signs such a hash, the odds of two hashes signed with SHA-1 equaling each other is $1/2^{160}$ [68, 265]. Clearly, this assumption cannot hold for, say, a two-bit hash!

> [Alice signs the message before encrypting it to prevent someone from tampering with it. For instance, if] Alice writes a letter, she signs it and then puts it in an envelope. If she put the letter in the envelope unsigned and then signed the envelope, Bob might [think that someone covertly replaced the letter.] If Bob showed [Alice's letter and envelope to Carol,] Carol might accuse Bob of lying about which letter arrived in which envelope.

Schneier explains that, in practice, Trent is not a publicly-accessible and *-writable* database, as "Mallory could substitute any public key for Bob's; after that, Bob could not read messages addressed to him, but Mallory could" (emphasis added) [265]. Mallory could also substitute public keys during transmission [265]. To prevent Mallory from doing this, only Trent has database write-access, and Trent "[signs] each public key with his own private key" [265]. Schneier explains key exchange further below (emphasis in original) [265].

> Trent, when used in this manner, is often known as a *Key Certification Authority* [(CA)] or *Key Distribution Center* (KDC). In practical implementations, the KDC signs a compound message consisting of the user's name, his public key, and any other important information about the user. This signed compound message is stored in the KDC's database. When Alice gets Bob's key, she verifies the KDC's signature to assure herself of the key's validity.. . .[With public-key cryptography, key exchange follows the protocol described on page 17, except, in practice,] Alice and Bob's signed public keys will be available on a database. [The key-exchange protocol is as follows:]
>
> 1. Alice gets Bob's public key from the KDC.
> 2. Alice generates a random session key, encrypts it using Bob's public key, and sends it to Bob.
> 3. Bob then decrypts Alice's message using his private key.
> 4. Both of them encrypt their communications using the same session key.
>
> [This protocol is vulnerable to a "man-in-the-middle" attack from Mallory, in which] Mallory can imitate Bob when talking to Alice and imitate Alice when talking to Bob. [The attack proceeds as follows:]

1. Alice sends Bob her public key. Mallory intercepts this key and sends Bob his own public key.
2. Bob sends Alice his public key. Mallory intercepts this key and sends Alice his own public key.
3. When Alice sends a message to Bob, encrypted in "Bob's" private key, Mallory intercepts it. Since the message is really encrypted with his own public key, he decrypts it with his private key, re-encrypts it with Bob's public key, and sends it on to Bob.
4. When Bob sends a message to Alice, encrypted in "Alice's" public key, Mallory intercepts it. Since the message is really encrypted with his own public key, he decrypts it with his private key, re-encrypts it with Alice's public key, and sends it on to Alice.. . .

[Digital signatures can foil this attack, as Trent has signed both Alice and Bob's public keys. This assures Alice that she is truly communicating with Bob, and vice versa; the previous protocol provides no such assurance. In addition, Ronald Rivest and Adi Shamir's *interlock protocol* can foil this attack.[10] This protocol follows the above key-exchange protocol, except that Alice and Bob only send half their encrypted messages to each other at one time. After the first message halves have been sent, they send the second message halves. This prevents Mallory from decrypting the message halves and forces him to send completely new messages to Alice and Bob, who will likely detect his presence.]

Finally, Schneier discusses authentication, in which a user proves her identity to some computer system (emphasis in original) [265]:

> When Alice logs into a host computer. . .or any other type of [terminal,] how does the host know who she is? How does the host know she is not Eve trying to falsify Alice's identity? Traditionally, passwords solve this [problem:] Alice enters her password, and the host confirms that it is correct. Both Alice and the host know this secret piece of knowledge and the host requests it from Alice every time she tries to log in.
>
> [One-way functions greatly aid the authentication process, as] the host [need not] know the [passwords: it need only distinguish] valid passwords from invalid passwords. [Hence it only stores] one-way

[10] As the astute reader may have noticed, the "R" and "S" in the RSA algorithm correspond to the last names of Rivest and Shamir, two of the algorithm's inventors [255]. The "A" corresponds to the last name of their co-inventor L. Adleman [255].

functions of the [passwords, not the passwords themselves. This protocol is described below:]

1. Alice sends the host her password.
2. The host performs a one-way function on the password.
3. The host compares the result of the one-way function to the value it previously stored.

[With this protocol, any damage resulting from theft of the password list is limited. Since only the one-way function of the passwords are stolen, the thief cannot reverse the function and recover the passwords. However, Mallory can still generate a list of common passwords, apply the one-way function to each of them,] steal an encrypted password [file, and compare his values to those in the file to find matches.] This is a *dictionary attack*, and [it can be very successful]. *Salt* is a way to make it more difficult. Salt is a random string that is concatenated with passwords before [the one-way function is applied to them.] Then, both the salt value and the result of the one-way function are stored in a database on the host. If the number of possible salt values is large enough, this practically eliminates a dictionary attack against commonly-used passwords because Mallory has to generate the one-way hash for each possible salt value. . .

[Unfortunately, salt does not mitigate the threat of Eve listening in on Alice's password before it reaches the host. Alice's password may travel through competing firms and unfriendly countries, and Eve can access the password at any one of these locations. If she] has access to the [host's processor memory,] she can see the password before the host hashes it. Public-key cryptography can solve this problem. The host keeps a file of every user's public key; all users keep their own private keys. . .Secure [public-key authentication] protocols take the following. . .form:[11]

1. Alice performs a computation based on some random numbers and her private key and sends the result to the host.
2. The host sends Alice a different random number.
3. Alice makes some computation based on the random numbers (both the ones she generated and the one she received from the host) and her private key, and sends the result to the host.
4. The host does some computation on the. . .numbers received from Alice and her public key to verify that she knows her public key.
5. If she does, her identity is verified.

2.2 Intellectual Property

Intellectual property (IP) is pervasive in Western society. In general, there are four types of IP: copyrights, trademarks, patents, and trade secrets. The U.S. Patent and Trademark Office defines them as follows [327, 329]:[12]

> [A] copyright is a form of protection provided to the authors of "original works of authorship" including literary, dramatic, musical, artistic, and. . .other intellectual works, both published and unpublished. The 1976 Copyright Act generally gives the owner of copyright the exclusive right to reproduce the copyrighted work, to prepare derivative works, to distribute copies or phonorecords[13] of [it, or] to perform [or display it] publicly.. . .A trademark is a word, name, symbol or device [that is used to indicate the source of a product] and to distinguish [the product from others; a service mark is analogous to a trademark, but applies to services.]. . .A patent for an invention is the grant of a property right to the [inventor, for 20 years from the date of filing,] issued by the Patent and Trademark Office.. . .[A trade secret is a piece of] information that companies keep secret to give them an advantage over their competitors.

Since copyright is the primary type of IP affecting TC, this section focuses on U.S. copyright law and its implications and U.S. treaties that require other countries to adopt IP laws similar to those in the U.S.[14]

U.S. copyright law is largely encoded in the 1976 Copyright Act, though the 1998 Digital Millennium Copyright Act (DMCA) and Sonny Bono Copyright Term

[11]These convoluted steps are necessary to prevent certain protocol attacks, the details of which lie beyond the scope of this paper [265].

[12]Other nations have differing definitions of IP and laws protecting it. This paper primarily discusses U.S. IP law, as many corporate advocates of TC are based in the U.S. Their zealous protection of IP affects laws and treaties around the world, as will be shown.

[13]As the U.S. Copyright Office notes, "a phonorecord is the physical object in which works of authorship are embodied. The word 'phonorecord' includes cassette tapes, CDs, LPs, 45 r.p.m disks, as well as other formats" [332].

[14]Chapters 5 and 8 discusses how other IP regulations affect TC implementations.

Extension Act (CTEA) significantly strengthen the law. The 1976 Act allows anyone who lives or resides in the U.S. or any other nation that has signed *any* treaty to claim copyright on *any* created work, whether or not it is published [332].[15] The "work" in question can be any " 'original [work] of authorship' that [is] fixed in a tangible form of expression" that is one of the following categories [332]:

- [Literary works, including computer programs and "compila tions"];
- [Musical] works, including any accompanying [words;]
- [Dramatic] works, including any accompanying [music;]
- [Pantomimes] and choreographic [works;]
- [Pictorial,] graphic, and sculptural [works;]
- [Motion] pictures and other sculptural [works, including maps and architectural plans;]
- [Sound recordings;]
- [Architectural works.]

However, the following works *cannot* be copyrighted (emphasis added) [332]:

- Works that have not been fixed in a tangible form of expression ([*e.g.*,] choreographic works that have not been notated or recorded, or improvisational speeches or performances that have not been written or recorded)[;]
- Titles, names, short phrases, and slogans; familiar symbols or designs; mere variations of typographic ornamentation, lettering, or coloring; mere listings of ingredients or [contents;]
- Ideas, procedures, methods, systems, processes, concepts, principles, discoveries, or devices, *as distinguished from* a description, explanation, or [illustration;]
- Works consisting entirely of information that is common property and containing no original authorship ([*e.g.*,] standard calendars, height and weight charts, tape measures and rulers, and lists or tables taken from public documents or other common [sources.])

[15] Further restrictions for claiming copyright are detailed in [332]. Most of them are not applicable to this discussion.

The copyright owner has the rights described on page 23—with some limitations. For instance, Section 107 of the Act allows "fair use" of copyrighted work "for purposes such as criticism, comment, news reporting, teaching. . . ,scholarship, or research" [331,332].[16] In particular, the following criteria determine "fair use" [331]:

1. [The] purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes;
2. [The] nature of the copyrighted work;
3. [The] amount and substantiality of the portion used in relation to the copyrighted work as a whole; and
4. [The] effect of the use upon the potential market for or value of the copyrighted work.

However, the passage of the DMCA greatly shifted the balance between owners of copyrighted works and users thereof [339]. Among other things, the law brought the U.S. into compliance with the World Intellectual Property Organization (WIPO) Copyright Treaty of 1996 and criminalized the circumvention of *any* digital copyright-protection technology for any purpose [339, 340].[17] In particular, the law states [177]:

> No person shall circumvent a technological measure that effectively controls access to a [copyrighted work.]. . .No person shall manufacture, import, offer to the public, provide, or otherwise traffic in any technology, product, service, device, component, or part thereof, that. . .is primarily designed or produced for the purpose of circumventing a technological measure that effectively controls access to a [copyrighted work,] has only limited commercially significant purpose or use other

[16]Other sections of the law (§§107–121) further restrict copyright, including clauses for noncommercial reproduction of copyrighted works by libraries [332].

[17]In its own words, WIPO is "an. . .agency of the United Nations [that] is dedicated to developing a balanced and accessible [IP] [system that] rewards creativity, stimulates innovation and contributes to economic development while safeguarding the public interest" [351].

> than to circumvent a technological measure that effectively controls access to a [copyrighted work,] or is marketed by that person or another acting in concert with that person with that person's knowledge for use in circumventing a technological measure that effectively controls access to a [copyrighted work.]. . .To "circumvent a technological measure" means to descramble a scrambled work, to decrypt an encrypted work, or otherwise avoid, bypass, remove, deactivate, or impair a technological measure, without the authority of the copyright owner.. . .A technological measure "effectively controls access to a [copyrighted] work" if the measure, in the ordinary course of its operation, requires the application of information, or a process or a treatment, with the authority of the copyright owner, to gain access to the work.

The law also criminalizes the manufacture, sale, and importation of any analog VCR that does not support automatic gain-control copy-protection technology; this technology forms "the basis of [Macrovision's]" copy-prevention technology [177,340]. Though the law *does* make exemptions for "nonprofit libraries, archives, and educational institutions," reverse-engineering a computer program "for the purpose of enabling interoperability [with] an independently created computer program," encryption research and security testing, the above anti-circumvention provisions have overridden these exemptions in practice. The astute reader has likely ascertained the DMCA's chilling effect on technological innovation, research, and free speech. The most salient example of DMCA abuse is the Dmitry Sklyarov case, discussed in the next section. However, Sklyarov was not the only one silenced by the law, which still has chilling effects today:

- Sony shut down the website `aibopet.com`, which contained posts by users of the company's Aibo robot dog that "taught" it new "tricks" [183]. One of these "tricks" was dancing jazz [183]. Sony, however, alleged that the site's

dancing-jazz program circumvented the dog's "copy protection protocol," and the site owner shut it down [183].

- The Recording Industry Association of America (RIAA) sent Professor Edward Felten of Princeton University a similar letter when he successfully broke the Secure Digital Music Initiative (SDMI) encryption system. The SDMI was an attempt by RIAA record labels to "enable content owners to exercise much better control over their content than the Internet [originally] granted them" [183]. The SDMI encryption system was an early attempt to effect a "trusted" digital rights management (DRM) system, such as Apple and Microsoft's systems today [183].

- At present, cryptographer Niels Ferguson refuses to publish the results of his 2001 examination of flaws in Intel's High-Bandwidth Digital Content Protection (HDCP) system, fearing his arrest in the United States [93, 94]. The HDCP uses encryption to "protect" high-definition content transferred to a high-definition monitor over the Digital Visual Interface (DVI), as will be discussed in Chapter 4 [56, 84, 99, 170]. As Ferguson routinely consults for clients in the United States, his potential arrest for violating the DMCA would place him in serious jeopardy [93, 94].

Several other, high-profile DMCA cases are listed in [339]. In addition, the 1998 Sonny Bono CTEA expanded copyright terms to encompass the creator's lifetime *plus* 95 years thereafter [183].[18] This effectively prevented then-copyrighted

[18] As Lawrence Lessig notes, the CTEA was "enacted in memory of the congressman and former musician Sonny Bono, who, his widow, Mary Bono, says, believed that 'copyrights should be forever' " [183]. Indeed, Congress has expanded copyright terms many times since the original

works from passing into the public domain, which Stanford Law School professor Lawrence Lessig explains in his book *Free Culture: How Big Media Uses Technology and the Law to Lock Down Culture and Stifle Creativity*: "At the end of a copyright term, a work passes into the public domain. No permission is then needed to draw upon or use that work. No permission, and, hence, no lawyers. The public domain is a 'lawyer-free zone' " [183]. The Walt Disney Company (hereafter referred to as *Disney*) lobbied heavily for the law, "contributing more than $800,000 to reelection campaigns in the 1998 [election] cycle" [183]. After all, Disney's 1928 Mickey Mouse cartoon "Steamboat Willie" would have fallen into the public domain if the CTEA were not signed into law [183].[19] The RIAA and Motion Picture Association of America (MPAA), lobbying organizations for their respective industries, "spent over $1.5 million lobbying in [that same election cycle]" [183]. With the passage of the CTEA, Mickey Mouse is still safely copyrighted—at least for 20 more years.

The RIAA and MPAA successfully lobbied—and continue to lobby—for similar IP laws in countries worldwide. In 2001, the European Union (EU) signed the European Union Copyright Directive (EUCD) into law, which contains anti-circumvention provisions similar to those in the DMCA [45,340]. The EU is considering a draconian IP rights enforcement directive that, among other things, restricts software reverse-engineering for the purpose of interoperability and mandates the criminalization of unauthorized street-music performance [12]. Moreover, the aforementioned associations, along with U.S. software companies, rammed a provision into the Central American Free Trade Agreement (CAFTA) that requires

1790 copyright law, which established a 14-year copyright that could be renewed once if a work remained profitable [183].

[19]Just prior to the CTEA's passage, copyright terms lasted *only* 70 years [183].

signatories to ban circumvention of copy-protection technology, just as the DMCA does [202]. Due to CAFTA's signing, Costa Rica, the Dominican Republic, El Salvador, Guatemala, Honduras, and Nicaragua must implement similar copyright laws [202]. Lobbying by the same parties has introduced similar provisions in "free trade" treaties with Jordan, Singapore, Chile, Morocco, Australia, Bahrain, and Oman [70].

2.3 Digital Rights Management

Digital rights management (DRM) technology uses public-key cryptography to restrict the use of files in accord with the wishes of IP rights-holders or corporate policy.[20] Firms widely use it to protect sensitive data and control how employees and other parties use that data. However, DRM is more commonly used to restrict consumers' use of media, and, hence, it is an extremely controversial technology. Enforcement of copyright owners' licenses for media files and "protection" of CDs and DVDs from illegal copying are among DRM's use cases. As noted in the previous section, the DMCA prohibits the circumvention of digital copyright-protection technology; as will be shown, the law provides the "teeth" of DRM enforcement.

[20]The European Standards Committee/Information Society Standardization System (CEN/ISSS) defines DRM as "the management of rights to digital goods and content, including its confinement to authorized use and users and the management of any consequences of that use throughout the entire life cycle of the content" [50, 250]. (The CEN/ISSS also solicited definitions of DRM from many other organizations; these other definitions are not used in this paper.) In general, the above cryptography-based definition is used in this paper, and the CEN/ISSS definition is mentioned for detailed discussion on DRM.

Companies routinely use DRM to protect confidential data (such as trade secrets) and control access to it. As journalist Mike Riley explains in *Dr. Dobb's Journal* [254]:

> In the corporate environment, DRM vendors enable their customers to digitally enforce corporate information distribution and confidentiality policies. Audit trails are captured and can be used to validate business practices and keep company secrets.
>
> While some vendors. . .offer a targeted solution for [Portable Document Format (PDF) files] rendered on multiple operating systems, others. . .supply broader file-type protection on a single [operating system] (Microsoft Windows).. . .For example, [vendor] SealedMedia's document protection technology can be used to enforce the ISO17799 security classification, ranging from level 1 (top secret) to level 5 (public domain) clearance. It also monitors authorized individuals when accessing a document, tracking what they can [and cannot] do with that document (print, cut and paste, forward, and so on).

He notes that Adobe Systems and Microsoft's "enterprise DRM" systems enforce access-control policies on PDF and Microsoft Office files, respectively [254]. In addition, South Korean company Fasoo offers DRM systems for workstations, servers (including file servers) and Web sites [78, 79]. These systems consist of "rights management servers," which establish access controls for employees and other parties, and "client-side" applications that connect to the servers to set or determine document restrictions [254]. However, he found scant information on vendor implementation details [254]:

> "Many of the vendors. . .refused to disclose intimate details about how their products work, adhering to the questionable—but widely practiced—"security through obscurity" policy. While this approach is frowned upon by security experts and perpetuates distrust in DRM . . . products in general, these vendors follow it to protect their own intellectual property as well as that of their customers."[21]

[21]Note that these vendors violate Kerckhoffs' principle articulated above in section 2.1, which increases the "brittleness" of their products' security.

Still, these applications are far less controversial than "protection" of digital and optical media. Since the late 1990s, the content industry has used DRM technology to restrict consumers' use of legally-purchased media files, CDs, and DVDs. Apple's FairPlay DRM and Microsoft's Windows Media DRM (WMDRM) are applied to music and movie files purchased from the iTunes Store and various other online stores, respectively. FairPlay permits users to copy iTunes media files to at most five computers and an unlimited number of iPods, burn the files (without DRM) to an unlimited number of CDs, and "burn playlists up to seven times each" [20, 343].[22] Apple does not describe FairPlay's encryption implementation on its web site. However, Wikipedia describes it as follows (without citing any sources) [343]:

> FairPlay is a fairly simple implementation of DRM techniques. FairPlay-protected files are regular MP4 container files with an encrypted AAC audio stream.[23] The audio stream is encrypted using the Rijndael algorithm in combination with MD5 hashes.[24] The master key required to decrypt the encrypted audio stream is also stored in encrypted form in the MP4 container file. The key required to decrypt the master key is called the "user key."
>
> Each time a customer uses iTunes to buy a [song] a new random user key is generated and used to encrypt the master key. The random user key is stored, together with the account information, on Apple's servers, and also sent to iTunes. iTunes stores these keys in its own encrypted key repository. Using this key repository, iTunes is able to retrieve the user key required to decrypt the master key. Using the master key, iTunes is able to decrypt the AAC audio stream and play it.

[22] As of this writing, Apple has not licensed FairPlay to any other company. Hence, FairPlay-protected files only play on Apple's iPod portable media player (PMP). (Both iPod® and iTunes® are registered trademarks of Apple, Inc. in the U.S. and/or other countries.)

[23] As its name implies, an MP4 file "contains" one or more Motion Picture Experts Group-4 (MPEG-4) "media streams," such as audio and video [290]. MPEG-4 is a media-file standard that scales the "playback" of media files from "cell phones to. . .set-top boxes to broadcast televisions" [18]. Advanced Audio Coding (AAC) is the MPEG-4 audio compression format; songs purchased from the iTunes Store are encoded in AAC [18, 343].

[24] The Rijndael (pronounced "Rhine dahl") algorithm is used in the Advanced Encryption Standard (AES), a symmetric block cipher [234]. Full discussion of AES is beyond the scope of this paper.

> When a user authorizes a new computer, iTunes sends a unique machine identifier to Apple's servers. In return it receives all the user keys that are stored with the account information. This ensures that Apple is able to limit the number of computers that are authorized and makes sure that each authorized computer has all the user keys that are needed to play the tracks that it bought.
>
> When a user deauthorizes a computer, iTunes will instruct Apple's servers to remove the unique machine identifier from their database, and at the same time it will remove all the user keys from its encrypted key repository.
>
> The iPod also has its own encrypted key repository. Every time a FairPlay-protected [song] is copied onto the iPod, iTunes will copy the user key from its own key repository to the key repository on the iPod. This makes sure that iPod has everything it needs to play the encrypted AAC audio stream.
>
> At this time, it looks like the restrictions mentioned above are hard-coded into QuickTime and the iTunes application, and not configurable in the protected files themselves.

Unlike Apple's FairPlay, Microsoft's WMDRM is used by myriad online stores, allows content owners to greatly restrict consumer use of media files, and "protects" them on various PMPs [217]. Content owners can restrict the number of times consumers can "play" the files, how many times they can copy the files to PMPs, and whether or not they can "burn" the files to CD, as well as other limitations. Microsoft describes how its DRM system works [217]:[25]

1. **Packaging.** [Microsoft's rights management server encrypts a digital media file with a key, which is] stored in an encrypted license, which is distributed separately....Other information is added to the digital media file, such as the URL where the license can be acquired. The [encrypted media] file is saved in Windows Media Audio [or Windows Media Video format with a .wma or a .wmv file extension, respectively].
2. **Distribution.** The [encrypted media] file can be placed on a Web site for download, placed on a digital media server for streaming,

[25]Both FairPlay and WMDRM are exclusively available to users of both Mac OS X and Microsoft Windows [217].

distributed on a CD, or e-mailed to consumers. [Depending on the license, consumers can share the WMDRM-protected file with their friends.]

3. **Establishing a license server.** The content [owner] chooses a license [clearinghouse] that stores the specific rights or rules of the license. . .[and authenticates consumer license requests. System management is streamlined by separate distribution and storage of media files and licenses.]
4. **License acquisition.** To play [the encrypted] digital media file, the [consumer—say, Alice—must] first acquire a license key to unlock the file. [The license acquisition process] begins automatically when [she tries] to acquire the [encrypted media] file, acquires a pre-delivered license, or plays the file for the first time. [Microsoft's rights management server] either sends [her] to a registration page where information is requested or payment is required, or "silently" retrieves [the] license from [the respective clearinghouse].
5. **Playing the digital media file.** To play the. . .file, [Alice] needs a [media] player that supports [WMDRM. She] can then play the file according to the rules or rights that are included in the [license, such as limitations on] start times and dates, duration, and counted operations ([*e.g.*, the file may only be played once on one computer during a weekend]). Licenses, however, are not transferable. If [she] sends [the encrypted] media file to [Bob, he] must acquire [his] own license to play [it]. This PC-by-PC licensing scheme ensures that [WMDRM-protected media files] can only be played by [computers that have acquired the respective license keys] for that file.

WMDRM uniquely "ties" each media player to its "host computer" so "any compromised player can be identified and disabled in the licensing process" [217]. It allows license issuers to "blacklist" applications (such as those with faulty DRM components) and "chain together" individual licenses so they can be updated by a single "root license"; this facilitates subscription business models, as only the root license need by updated after Alice renews her subscription [217]. On

Windows Millennium Edition (ME) and XP, WMDRM uses Microsoft's Secure Audio Path technology to encrypt audio "traveling" between the media player and sound card, thereby preventing "sniffing" of a protected media file's audio after it "leaves" the media player [210,217]. In their licenses, content owners can demand that consumers support this "feature," thereby forcing them to use Windows ME or XP [210].[26] In addition, the Windows Media Data Session Toolkit, which is part of WMDRM, can protect media files on CDs and DVDs with the requisite licenses so Alice, Bob, and others can play the files on their computers [211].[27] Like Apple, Microsoft does not divulge its DRM encryption implementation. However, Wikipedia states that an early DRM system for Windows Media Audio used "a combination of elliptic curve cryptography key exchange, [DES and] a custom block cipher, [the] RC4 stream [cipher,] and the SHA-1 hashing function" [217,344].[28]

DRM technologies also restrict consumers' use of optical media, such as CDs and DVDs. Historically, the recording industry has included copy-prevention software on their labels' CDs to hinder copyright infringement.[29] Initially, the labels hired DRM vendors to intentionally introduce errors on CDs that made them uncopyable [141]. Some of these CDs damaged Apple iMac CD-ROM drives or

[26]This restriction presages Microsoft's content-protection measures in Windows Vista, which are discussed in Chapter 5.

[27]The Windows Media Data Session Toolkit is used in combination with the infamous Sony BMG DRM systems, as will be discussed below [140].

[28]Elliptic curve cryptography is a type of public-key cryptography based, as one might imagine, on the algebra of elliptic curves [342]. Further details on it and these algorithms are beyond the scope of this paper.

[29]This software was included on many CDs produced by the four "major" recording *companies*—Universal Music, Sony BMG, EMI Group, and Warner Music Group—who own these record *labels* [150]. Their labels produce 70% of the music sold worldwide [150]. Of course, this software also violates users' fair-use rights to "rip" their legally-purchased music and movies onto their computers and store them on portable devices.

rendered other drives inoperable until the computer was rebooted [141].[30] Still, this "protection" was straightforward to bypass until 2005, when Sony BMG released millions of DRM-protected CDs that, among other things, installed software on users' computers without permission [140]. The CDs contained the Windows DRM software XCP and MediaMax, which were created by the firms First4Internet and SunnComm, respectively [140]. These DRM systems deserve closer examination, as they ultimately led to class-action litigation against Sony BMG, which forced the recall of millions of CDs [140]. Princeton University researchers Ed Felten and J. Alex Halderman examined this software in detail and describe its functionality as follows [140]:

- Both XCP and MediaMax use the Windows "autorun feature" to load "active protection" software to prevent users from copying songs to their computers.[31] XCP launches an "installer" program that displays a license agreement and continually compares the system's executing programs against a "blacklist" of known CD-ripping programs. If one or more matches is found, the installer demands that the user close the programs within 30 seconds, and if any of these programs are still running after that time, "the installer ejects the CD and quits." MediaMax installs the driver `sbcphid.sys` as a service in the Windows registry, and, under some circumstances, launches the service every time Windows boots. MediaMax does this *regardless* of the user's acceptance of its license agreement.

[30] iMac® is a registered trademark of Apple, Inc. in the U.S. and/or other countries.

[31] This "feature," exclusive to the Microsoft Windows operating system, automatically executes the program(s) whose name(s) are listed in the `autorun.inf` file on the CD when the user inserts it in his CD-ROM drive [140].

- Both XCP- and MediaMax-protected CDs use "passive protection" that "exploits subtle differences between the way computers [and ordinary CD players] read CDs" such that the former "see" only the DRM software on the CD but the latter "see" only audio songs. This prevents "ripper applications that use the Windows audio CD driver" from copying the audio.

- The XCP and MediaMax active protection software "install a background process that interposes itself between applications and the [Windows CD audio driver]" to prevent other programs from copying the DRM-protected CD audio. In both DRM systems, this process is one or more kernel-mode drivers: XCP's drivers, `crater.sys` and `cor.sys`, "attach to the CD-ROM and IDE devices", whereas MediaMax's driver, `sbcphid.sys`, *only* attaches itself to the CD-ROM drive. Both sets of drivers examine CDs to determine if they are protected by the respective DRM system, "monitor for attempts to read the audio [tracks, such as] during a playback, rip, or disc copy operation, and corrupt the audio returned by the drive to degrade the listening experience.. . .XCP replaces the audio with random [noise, whereas] MediaMax introduces a large amount of random jitter." It is noteworthy that these "disc recognition" technologies not only detect the "original" disc containing the DRM software, but *any other disc* protected with XCP or MediaMax, respectively. XCP also installs a "rootkit," which "conceals any file, process, or registry key whose name begins with the prefix `$sys$`.. . .The rootkit is a kernel-level driver named `$sys$aries` that [loads whenever the computer boots]" and modifies "system calls" to block the aforementioned files, processes, or registry keys.

- XCP and MediaMax use the Windows Media Data Session Toolkit with proprietary media players on the CD to allow users to listen to the songs. The players also provide "access to 'bonus content,' such as album art, liner notes, song lyrics, and links to artist web sites." The players allow users to burn a certain number of CD copies (usually 3) and rip tracks to their computers in DRM-protected Windows Media files. The files are associated with licenses that *only* allow them to be transferred to WMDRM-supported portable devices or ripped to CD; the files *cannot* be transferred to another computer. Moreover, the players "phoned home" to `connected.sonymusic.com` (for XCP-protected CDs) or `license.sunncomm2.com` (for MediaMax-protected CDs) *without user consent*, identifying the album via a unique code.

Moreover, uninstallers provided by First4Internet and SunnComm executed a proprietary ActiveX control, one of whose parameters was an arbitrary URL "of a DLL file containing code to uninstall" the respective DRM system; this control could be used to execute arbitrary attacker code [140]. Public outcry over XCP and MediaMax ultimately led to the aforementioned litigation against Sony BMG. Finally, *all* "major" record labels abandoned DRM on audio CDs as of January 2007 [238]. Nonetheless, DRM is still widely used on DVDs, both current- and next-generation (HD DVD and Blu-ray). Under the aegis of the DVD Copy Control Association (DVD-CCA), DVDs have been encrypted with the Content Scramble System (CSS) "to deter piracy" [4]. Though CSS was originally broken in 1999 by "an anonymous German hacker," the Norwegian Jon Lech Johansen is usually associated with the break, especially due to his work on DeCSS, a program that removes the CSS encryption [4]. He wrote DeCSS to exercise his fair-use rights to view DVDs

on his GNU/Linux operating system.[32] The Norwegian content industry charged him multiple times for copyright violations, but he was ultimately acquitted [4,25]. His success in cracking CSS was due in part to the cipher's poor design. As Nate Anderson of Ars Technica explains, "DVD players are factory-built with a set of keys. When a DVD is inserted, the player runs through every key it knows until one unlocks the disk. Once this disc key is known, the player uses it to retrieve a title key from the disc. This title key actually allows the player to unscramble the disc's contents" [4]. Though CSS is a 40-bit stream cipher, the maximum allowed under U.S. cryptographic export restrictions at the time, a cryptanalytic attack succeeded in recovering the title key in *18 seconds*—on a *450 MHz Pentium III* processor [4,301]. DVD descramblers rapidly proliferated on the Internet [4]. However, Hollywood—and the DVD-CCA—learned greatly from their mistakes: CSS' successor, the Advanced Access Content System (AACS), is far more robust than its predecessor. Both HD DVD and Blu-ray high-definition discs are "protected" using AACS [4]. AACS makes pervasive use of 128-bit AES, which Steve Gibson calls "our state-of-the-art crypto" [132]. Just as the DVD-CCA issued title keys to DVD players, the AACS Licensing Authority (AACS-LA) issues device keys to HD DVD and Blu-ray players [4]. But AACS uses an advanced cryptographic technique called *subset-difference revocation* to revoke "compromised" device keys once they are made public, which prevents players with those keys from reading successive HD DVD and Blu-ray movies [4,165].[33] As Anderson explains, "AACS-encrypted

[32]The DVD Copy Control Association (DVD-CCA) requires signature of a license agreement and payment of fees before revealing CSS details [4, 67]. The DVD-CCA charges $500 to view technical specifications and annual license fees from $5,200 to $15,400, which many open-source Linux projects cannot afford [67].

[33]In his *Security Now!* podcast with TechTV's Leo Laporte, Gibson also notes that HD DVD and Blu-ray players use AES hashes, digital signatures, and public-key exchange [132]. He adds that

discs will feature a Media Key Block [(MKB)] that all players need to access in order to get the key needed to decrypt the video files on the disc. The MKB can be updated by AACS-LA to prevent certain sets of Device Keys from functioning with future titles—a feature that AACS dubs 'revocation' " [4]. AACS has a "flag," the Image Constraint Token (ICT), that, when set by Hollywood studios, will "degrade" high-definition video on analog displays [4,27]. AACS has a Managed Copy feature that allows users to make copies of high-definition discs without violating the DMCA (though studios may charge for this) [4,97]. Blu-ray discs also support BD+, which allows the Blu-ray encryption algorithm to dynamically update itself if a cipher is broken, and ROM Mark, which uniquely watermarks discs to deter commercial piracy [4]. Anderson adds that "all Blu-ray mastering equipment must be licensed by the [Blu-ray Disc Assocation], [which] will ensure that all of it carries ROM Mark technology" [4]. However, AACS' protection may crumble soon. In January 2007, a hacker known as "Muslix" released BackupHDDVD, a program that decrypts HD DVDs given the disc's *title* key, which software players like PowerDVD and WinDVD supposedly "leave" decrypted in memory [86, 90, 105].[34] "Muslix" has since reported success in decrypting a Blu-ray disc [251]. It is noteworthy that neither format has been "cracked," as a *hardware* player's device key is necessary to compute the title key and published device keys can be revoked [86,105,251]. Moreover, AACS-LA can track down the compromised player, as Halderman explains (emphasis added) [138]:

> the cryptography greatly degrades player performance: first-generation players took *over a minute* "to register the arrival of the disk" after it was inserted, whereas current-generation players *only* take 15 seconds [132].

[34] Many HD DVD title keys are available from websites such as `hdkeys.com` and `aacskeys.com`, and Blu-ray title keys are available online as well [90,251].

> [An attacker can] keep his device keys secret and create a web site where people can upload header information from discs they want to decrypt. Then he would use his device keys to extract the title keys from those headers and post the title keys back to the site—a sophisticated attacker might automate this process. Cryptographers call this kind of site a *decryption oracle*.
>
> As it turns out, the designers of AACS anticipated decryption oracles, so the system includes a way to track down and blacklist the device keys used to operate them. This process is called "*traitor tracing*," and it works roughly like this: [AACS-LA] creates a phony disc header that can be decrypted by about half of the possible devices. ([It] just [needs] the header, so there's no need to press an actual disc.) [AACS-LA uploads] this to the oracle and [sees] whether it can find the title key. The result lets [AACS-LA] narrow down which devices the oracle's key might have come from. [It] repeats the process, creating a new header that will reduce the set of suspects by half again. With a few of these probes, [it] can home in on the oracle's device keys.
>
> ([Of course,] the oracle might know keys from more than one device, [try] to trick [AACS-LA] by pretending it can't decrypt certain headers when it really can, [etc].)

In a game-theoretic analysis of an attacker's publication of title keys, Felten and Halderman argue that an attacker has an incentive to *limit* the number of title keys he publishes *at any one time* to avoid blacklisting [139]. More importantly, "the attacker's best strategy is to withhold any newly discovered [player compromise] until a 'release window'. . .has passed since the last time [a player was blacklisted]," which is not unlike Hollywood's "release window" strategy of releasing movies to maximize revenue [88, 90]. Ultimately, they argue, studios will only collect a small fraction of the revenue they *could* collect based on the "marginal value of [AACS] protection" [88].[35]

[35]However, their model assumes that AACS-LA and the attacker are rational actors, which is unlikely in reality [88]. For instance, if a large number of title keys are leaked online, the movie studios may sue "attackers" *en masse* to "set an example" and deter future attackers from publishing keys [88]. Indeed, AACS-LA is threatening legal action against those who post a new crack online that threatens AACS' security [102, 253].

As astute readers have likely deduced, the creation, uploading, and use of *any* DRM-circumvention software violates the DMCA, as exemplified in the Dmitry Sklyarov case. Sklyarov, an ElcomSoft programmer, was arrested by the FBI in July 2001 after he gave a talk at the DefCon conference "about his company's Advanced eBook Processor, software designed to crack [DRM] protections on Adobe Systems' [electronic books]" [42, 179].[36] After programmers around the world rallied behind him, Adobe declined to prosecute him [42]. The U.S. government also "dropped charges against [him], but only under the condition that he cooperate with [its] case against [ElcomSoft]" [4]. At his trial, he explained that he never meant for the software to be used illegally, and a "jury acquitted ElcomSoft of all counts" on December 17, 2002 [42]. Though he was freed immediately thereafter, the case established a legal precedent for prosecution of those who wrote, distributed, and used software that circumvented DRM *of any kind* [4]. As noted by Colleen Pouliot, Adobe's general counsel, "ElcomSoft's Advanced eBook Processor is no longer available in the United States, and from that perspective the DMCA worked"; indeed, ElcomSoft does not sell the software anymore [4, 69]. The law's "chilling effects" extend beyond the Sklyarov case. In 2000, the MPAA sued the hacker magazine *2600* after the latter published Jon Johansen's DeCSS code, and a federal district court ruled that even *linking to* or publishing the code violated copyright law; the ruling was upheld on appeal [143]. Since the DMCA prohibits "trafficking"

[36] Among other products, Adobe makes electronic book (eBook, e-book) software that allows book publishers to specify if users can copy, print, and "read aloud" from the e-book [183]. ("Read aloud" refers the computer "speaking" the text of the book via text-to-speech conversion [183].) Publishers may also specify if the book may be "lent" or "given" to someone else as well as how many times text may be selected and pages printed [183]. However, these restrictions do not always match the realities of copyright: Lessig points out that Adobe forbids users from copying text from, printing, lending, or reading aloud from Lewis Carroll's book *Alice's Adventures in Wonderland*, which *is in the public domain*!

in tools that circumvent copyright-protection systems, discussing weaknesses in these systems is *verboten*, as illustrated in the Felten and Ferguson examples in Section 2.2.

Unlike DRM proponents, which view the technology as a sensible means of protecting IP, critics allege that:

- DRM stifles competition;
- DRM is bad for business;
- DRM is not standardized;
- DRM (unfairly) strips users of their rights; and
- DRM is simply unworkable [185, 341].

Electronic Frontier Foundation activist Cory Doctorow argues that Apple's hugely successful iTunes Store, whose songs only play on the company's iPod PMP, locks customers in to a single vendor [66]. Due to the DMCA, he claims, it is illegal to reverse-engineer *any* copyright-protection system for *any* reason, even a legal one, and Apple exploits this to "add new restrictions to the songs its customers have already purchased" [66]! Small wonder Anderson proclaims that "DRM's best friend might just be [Apple, Inc]," since DRM, combined with Apple's tight iTunes Store/iTunes/iPod integration, helps the company maintain its *de facto* monopoly in the digital-music sales market [7].[37] In a talk with Microsoft Research, Doctorow explains how DRM hurts business [63]:

[37]Ironically, as of this writing, Apple CEO Steve Jobs claims that Apple would gladly sell DRM-free music through the iTunes Store if the record labels agreed to do so [5, 166]. Yet Apple refuses to sell DRM-free music from Canadian record label Nettwerk "even when Nettwerk artists like the Barenaked Ladies approve" [5]. On the other hand, rival store Yahoo Music has sold several DRM-free songs by EMI artists Norah Jones and Relient K, and hopes to sell all EMI songs without DRM

> This is the worst of all the ideas embodied by DRM: that people who make [record players] should be able to [specify] whose records you can listen to, and that people who make records should have a veto over the design of [record players].
>
> We've never had this principle: in fact, we've always had just the reverse. Think about all the things that can be plugged into a parallel or serial interface, which were never envisioned by their inventors. Our strong economy and rapid innovation are byproducts of the ability of anyone to make anything that plugs into anything else:. . .[they make] billionaires out of nerds.
>
> The courts affirm this again and again. It used to be illegal to plug anything that didn't come from AT&T into your [phone jack]. They claimed that this was for the safety of the network, but [it was really designed to prop up AT&T's "phone rental fee" racket].
>
> When that ban was struck down, it created the market for third-party phone equipment, from talking novelty phones to answering machines to cordless handsets to [headsets—billions] of dollars of economic activity that had been [suppressed] by the closed interface. [AT&T benefited from this from making phone kits.]

Moreover, critics claim, DRM systems limit interoperability and standardization [245]. For example, DVD "region codes" allow DVDs to be played *only* in the part of the world corresponding to the code, and one cannot play FairPlay-encrypted songs in Windows Media Player or WMDRM-encrypted songs in iTunes; those who wish to do so must (illegally) circumvent the DRM [66, 245]. The argument that DRM systems strip users of rights—especially fair-use rights—is self-explanatory.

by the end of 2007 [28, 29]. In a recent Jupiter Research survey of European record executives, over half of the executives "[think] that current DRM systems are too restrictive. . .[and] that dropping DRM and releasing music files that can be enjoyed on any MP3 player would boost [general digital music sales]" [32]. In response to Jobs, Macrovision CEO Fred Amoroso argues that DRM should be applied to *all* media, DRM increases consumer value and electronic distribution of content, and DRM systems "[need] to be interoperable and open"—a point to be discussed shortly [2]. In April 2007, EMI started selling its songs at the iTunes Store in a high-fidelity, DRM-free AAC format for $1.29 per song, *30¢more* than the "standard" low-fidelity DRM-protected AAC format songs that EMI will still sell [91]. Other record companies may soon follow suit, according to one executive [175].

Indeed, Ken Fisher of Ars Technica claims "[DRM is] now a behavioral modification scheme that *permits* this, *prohibits* that, *monitors* you, and *auto-expires* when. Oh, and sometimes you can. . .watch a video or listen to some music.. . .In a nutshell: *DRM's sole purpose is to maximize revenues by minimizing your rights and selling them back to you*" (emphasis in original) [104]. He cites a *BusinessWeek* article that explains why Hollywood studios will not sell their films through the iTunes Store; in that article, an anonymous studio executive said that "[Steve Jobs'] user rules just scare the heck out of us" [104, 136]. The executive was referring to "Apple's video iPod [DRM] scheme, [in which] folks can share their [iTunes-purchased] flicks with as many as three other iPod users" [136]. As Fisher puts it, "[three] devices authorized for playback is too many, and the studios apparently believe this is 'just as bad' as piracy. Hollywood believes that iTunes Store customers will add their buddies' devices to their authorization list, and like evil communists, they'll share what they have purchased" [104]. Due to DRM's erosion of users' rights, it should surprise no one that the Free Software Foundation (FSF) calls the technology "Digital Restrictions Management" [109].[38] ZDNet executive editor David Berlind calls DRM "[Content Restriction, Annulment, and Protection]" [35].[39] In addition, as Doctorow, Schneier, and even Microsoft researchers explain, DRM systems are simply unworkable, digital copy prevention is futile, and DRM is ineffective with respect to ubiquitous file-sharing networks, respectively [37, 63, 266]. In his Microsoft Research talk, Doctorow describes why these systems fail [63] (emphasis added):

[38]The FSF calls TC "Treacherous Computing" for reasons we will examine in Chapters 5 and 8.

[39]One can easily deduce the associated acronym, which is precisely Berlind's point.

> In DRM, the attacker is *also the recipient*. . . .Alice sells Bob a DVD. She sells [him] a DVD player. The DVD has a movie on [it—say *Pirates of the Caribbean*—and] it's enciphered with [the CSS algorithm]. The DVD player has a CSS [descrambler].
>
> Now, let's take stock of what's a secret here: the cipher [(CSS)] is well-known. The ciphertext [(the encrypted DVD)] is most assuredly in enemy hands. . . . So what? As long as the key is secret from the attacker, we're golden.
>
> But there's the rub. Alice wants Bob to buy *Pirates of the Caribbean* from her. [He] will only buy [it] if he can descramble the CSS-encrypted [video] on his DVD player. Otherwise, the disc is only useful to Bob as a drinks-coaster. So Alice has to provide [Bob—the attacker—with] the key, the cipher and the ciphertext. . . .
>
> DRM systems are broken in minutes, sometimes days. . .[They are broken because they] share a common vulnerability: they provide their attackers with ciphertext, the cipher, and the key. At this point, the secret isn't a secret anymore.

Schneier most clearly elucidates the realities behind DRM and why, in his opinion, digital copy prevention is futile (emphasis added) [266]:

> The entertainment industry sees services like [the former, illegal] Napster as the death of its business, and it's using every [possible] technical and legal means. . .to prevail against them. They want to implement widespread copy prevention of digital files, so that people can view or listen to content on their [computers] but can't copy or distribute it.
>
> Abstractly, it is an impossible task. *All entertainment media on the Internet. . .is just bits: ones and zeros. Bits are inherently copyable, easily and repeatedly. If you have a digital [file,] you can make as many copies of that file as you want, do whatever you want with the copies.* This is a natural law of the digital world, and makes copying on the Internet different from copying [brand-name goods].
>
> What the entertainment industry is trying to do is use technology to contradict that natural law. They want a practical way to make copying hard enough to save their existing business. But they are doomed to fail. . .Fifteen years of software copy protection has taught us that, with enough motivation, any copy protection [scheme—even] those based on [hardware—can] be broken. . . .
>
> [The entertainment industry] tries to build [DRM] solutions that work against average users and most hackers. This fails because of

a second natural law of the digital world: *the ability of software to encapsulate skill*. A safe that can keep out 99.9% of all burglars works, because the safe will rarely encounter a burglar with enough skill. But a copy protection scheme with similar characteristics will not, because that one-in-a-thousand hacker can encode his break into software and then distribute it.. . .

The entertainment industry is responding in two ways. *First, it is trying to control the users' computers.* CSS is an encryption scheme, and protects DVDs by encrypting their contents.. . .[It was broken by attacking] the video stream after decryption. This is the Achilles' heel of all [encryption-based] content protection [schemes]: the display device must contain the decryption key in order to work.. . .

The entertainment industry is trying to turn your computer into an Internet Entertainment Console, where they, not you, have control over your hardware and software.. . .

The industry's second response is to enlist the legal system. [Legislation such as the DMCA] made it illegal to reverse-engineer copy protection schemes. Programs such as the one that broke CSS are illegal to write or distribute under the DMCA. This is failing because of a third natural law of the digital world: *the lack of political boundaries*. The DMCA is a U.S. law, and does not affect any of the hundreds of other countries on the Internet. And while similar laws could be passed in many countries, they would never have the global coverage it needs to be successful.

More legal maneuvering is in the works. The entertainment industry is now trying to pin liability on Internet service providers. The next logical step is to require all digital content to be registered, and to make recording and playback equipment without embedded copy protection illegal. All in an attempt to do the impossible: to make digital content uncopyable.

The end result will be failure. All digital copy protection schemes can be broken, and once they are, the breaks will be distributed. . .law or no law. Average users will be able to download these tools from Web sites that the laws have no jurisdiction over. Pirated digital content will be generally available on the Web. Everyone will have access.

The industry's only solution is to accept the inevitable. Unrestricted distribution is a natural law of digital content, and those who figure out how to leverage that natural law will make money.. . .

Digital files cannot be made uncopyable, any more than water can be made not wet. The entertainment industry's two-pronged offensive will have far-reaching [effects—its] enlistment of the legal system erodes fair use and necessitates increased surveillance, and its attempt to turn

> computers into an Internet Entertainment Platform destroys the very thing that makes computers so [useful—but the offensive] will fail in its intent.

In a 2002 ACM DRM Workshop paper, Microsoft researchers Peter Biddle, Paul England, Marcus Peinado, and Bryan Willman argue that DRM is ineffective in light of

> the *darknet*—a collection of networks and technologies used to share digital content. The darknet is not a separate physical network but an application and protocol layer riding on existing networks. Examples of darknets are peer-to-peer [(P2P)] file sharing, CD and DVD copying, and key or password sharing on email and newsgroups.. . .The idea of the darknet is based upon three assumptions:
>
> 1. Any widely distributed [piece of content] will be available to a fraction of [users who will copy content] in a form that permits copying.
> 2. [These users] will copy [content] if it is possible and interesting to do so.
> 3. [These users] are connected by high-bandwidth channels.
>
> (emphasis in original) [37].

They trace the rise of the darknet from "small world" networks of friends sharing content to unauthorized Web and FTP content servers to P2P networks and examine DRM systems, which aim to inhibit darknet content sharing [37]. These systems have the following content-protection responsibilities [37]:

> A [classical DRM] system is one in which a client obtains content in protected (typically encrypted) form, with a license that specifies the uses to which the content may be put.. . .The license and the wrapped content are presented to the DRM system whose responsibility is to ensure that:
>
> 1. The client cannot remove the encryption from the file and send it to a [peer;]

2. The client cannot "clone" its DRM system to make it run on another [host;]
3. The client obeys the rules set out in the DRM [license; and]
4. The client cannot separate the rules from the payload.

Advanced DRM systems may go further.

However, they conclude that DRM systems are mostly ineffective and even impede (legal) commerce in DRM-protected files (strong emphasis added, emphasis in original) [37]:

> DRM systems are limited to protecting the content they contain. Beyond [the first darknet assumption], **it is not impacted by DRM systems**. [With respect to this assumption,] DRM design details, such as properties of [tamper-resistant DRM client software,] may be. . .less relevant than the question [of] whether the current darknet has a global database. In the presence of an infinitely efficient darknet—which allows instantaneous transmission of [content] to all interested users—even sophisticated DRM systems are inherently ineffective. On the other hand, if the darknet [consists] of isolated small worlds, even ["break-once, break-everywhere"] DRM systems are highly effective.
>
> There is evidence that the darknet will continue to exist and provide [low-cost], high-quality service to [many consumers, and, hence,] in many markets, [it] will be a competitor to legal commerce.. . .[Stronger DRM systems] may act as a *disincentive* to legal commerce. Consider an MP3 file sold on a web site: this costs money, but the purchased object is as useful as a version acquired from the darknet. However, a. . .DRM-wrapped song is strictly *less* attractive: although the industry is striving for flexible licensing rules, customers *will* be restricted in their actions if the system is to provide meaningful security. **This means that a vendor will probably make more money by selling unprotected [content] than protected [content]. In short, if you are competing with the darknet, you must compete on the darknet's own terms:. . .convenience and low cost rather than additional security**.

Subsequent chapters describe how TC can be used—and *abused*—to "turn computers into an Internet Entertainment Platform" and make digital files uncopyable, but also greatly enhance computer security under users' control [266].

CHAPTER 3

TC OVERVIEW

This chapter precisely defines the term "trusted computing" and explores its *raisons d'être*, especially with respect to "secure" DRM platforms. In addition, the five primary TC technologies—the endorsement key (EK), sealed storage, memory curtaining, secure input and output (I/O), and remote attestation—are explored.

3.1 Definition of "Trusted Computing"

It is difficult to precisely define the term "trusted computing," as the notion of "trust" embodied in TC diverges from the vernacular meaning thereof. In the case of TC, "trust" refers to a *trusted system*, which "is a system that can be relied on to follow certain rules" [300, 348]. In his *Crypto-Gram* newsletter, Schneier adds that "[the Department of Defense's] definition of a trusted system is one that can break your security policy; *i.e.*, a system that you are forced to trust because you have no choice" (emphasis added) [267]. However, a *trusted* system is not necessarily *trustworthy*. For example, Alice may trust Trent to keep her health care records private, but then Trent sells them online. In this case, Trent is a trusted system, though he is certainly *not* trustworthy! To compound the confusion, there are several terms similar to "trusted computing" that have

very different meanings. For instance, Microsoft's term "Trustworthy Computing" refers to its focus on providing safer and more reliable computing platforms, and the Free Software Foundation's (FSF's) term "Treacherous Computing" reflects the latter's opposition to *trusted* computing technologies [109, 230].[40] In order to understand the *meaning* of TC, it is useful to examine the TCG's definition thereof: "hardware and software behave as intended" [320].[41] One immediately questions *who* wants the hardware and software to behave in accord with his or her intentions. For instance, a software vendor may wish that people *only* use its software in accord with the license agreement and *not* copy or reverse-engineer files; a content owner may wish that people who buy its media files *only* use them in accord with the respective licenses. Hence, "trusted" computers must include mechanisms to facilitate mandatory access and usage controls, such as those DRM systems described in section 2.3. Yet these systems are easily broken, albeit illegally, by software attacks, and hardware-based security is therefore necessary to enforce access and usage controls [4, 147].[42] Based on the TCG definition, the term *trusted computing* may now be defined as "a computing paradigm that uses hardware- and software-based security mechanisms to ensure that hardware and software act in accord with the intentions of a party *who may not be the user of a computer system*." This definition is used in the remainder of the paper.

[40]Throughout the rest of this paper, the acronym *TC* refers to *trusted computing*, *not* any of the "alternative" terms described above.

[41]It is noteworthy that the TCG defines *trust* as "the expectation that a device will behave in a particular manner for a specific purpose" [323]. The TCG defines a *Trusted Computing Platform* as "a computing platform that can be trusted to report its properties" [323].

[42]For instance, hackers "broke" Apple's iTunes DRM and Microsoft's WMDRM several times [147]. As illustrated in section 2.3, they also broke Adobe's e-book DRM and CSS. Furthermore, DRM-"protected" audio and video can be "captured" using programs like Streamripper and Camtasia Studio, respectively [147].

3.2 TC's *Raisons d'Être*

TC primarily came into existence to greatly enhance the security of computer systems, especially in light of massive identity theft, theft of sensitive data, and software attacks [318]. Among TC's many uses—and *ab*uses—are robust DRM enforcement, prevention of cheating in online games and auctions, and mitigation of malware attacks [348]. Robust DRM enforcement is perhaps TC's most likely (ab)use, which is discussed in detail later in this section. The other TC "use cases" are immediately apparent from the discussion of primary TC technologies in the next section. Before describing DRM enforcement, it is useful to *briefly* discuss several threats computer users face today to "provide context" for TC's adoption. Since these threats—and corresponding attacks—continually evolve, the following discussion is by no means exhaustive.

As mentioned in Chapter 1, computer users face a wide variety of security threats. Initially, these threats were confined to viruses, worms, and spyware, but they have since evolved.[43] Miscreants use "social engineering" tactics when they send e-mails purportedly from a financial institution or other company to entice users to enter credit card numbers and other sensitive information. More troublingly, these people entice users to execute purportedly useful, but malicious, e-mail attachments or other programs that give the attackers complete control

[43]Definitions for "virus," "worm," and "spyware" are given in [154]. A *virus* was, "originally, computer code that inserts itself into another program and replicates when the host software runs" [154]. Now, "viruses" encompass worms and "Trojan horse" programs; the term "Trojan horse" is defined on page 52 [154]. A *worm* is "self-replicating code that automatically spreads across a network [154]. *Spyware* is "software that reveals private information about the user or computer system to eavesdroppers" [154].

of users' computers [154]![44] These programs, or *Trojan horses*, can also install "keyloggers," which transmit users' keystrokes over the Internet to a third party; a recent McAfee white paper blames keyloggers in part for the recent surge in identity theft [248]. As one might imagine, businesses face similar threats, too. In particular, attackers (perhaps industrial spies) have launched "targeted attacks" against them since 2006 to steal confidential information [155]. The attack vectors are e-mails that contain Microsoft Office documents; when the documents are opened, Office executes arbitrary programs from the Internet [155].

In this climate, one can easily see why plans to secure computer platforms, including TC, are welcome. Certainly, its proponents—including International Data Corporation, the Enterprise Strategy Group, Endpoint Technologies Associates, and, of course, the TCG—laud TC's capability to protect provide tighter enterprise security and even stop rootkits from spreading across a network [167,256,286,322]. But an equally compelling use of TC is rigorous DRM license enforcement, especially since a purchaser of a DRM-protected media file does *not actually own* the file he purchased! Rather, as researchers Jason Reid and William Caelli of Australia's Queensland University of Technology point out (emphasis in original) [250]:

> The essential premise of DRM is that a rights owner wishes to license digital content ([that] is represented as. . .*bits*) to a licensee or *customer* who agrees to be bound by the terms of the license. Note that the customer is not buying the bits themselves. Rather, they are buying the right to use the bits in a defined and restricted manner, as [authorized] in the terms of the license. Hence the license defines a type of usage policy.

[44]Frequently, these malicious programs download software that allows remote attackers to commandeer the computer, along with many other compromised computers, to attack web sites, among other nefarious purposes [43]. These computer networks are known as *botnets* [43]. [43], which is co-written by the author, explains the botnet threat in much greater detail.

Yet, as Doctorow points out on 44, the DRM-protected media file must be *decrypted* before the purchaser can view or listen to it [250]. This caveat allows "capture" programs like Streamripper and Camtasia Studio to "record" the files' audio and/or video, thereby bypassing the content owner's access controls embodied in the repesctive files' licenses. Reid and Caelli note that "consequently, to reliably enforce typical DRM policies, it must not be possible for the platform user to access the *plaintext* bits that represent the content, *despite the. . .reality that the [computing] platform is under the user's direct control*" (emphasis added) [250]. Then, one can argue, as Mark Stefik did in the *Berkeley Technology Law Journal*, trusted systems are a precondition for robust DRM [300]. Stefik's definition of a trusted system is the same definition thereof presented in Section 3.1, except that "in the context of digital works, a trusted system follows rules governing the terms, conditions, and fees for using [them]" [250, 300]. These terms for using a "digital work" are analogous to the rules in the DRM license for a media file as described by Biddle *et al.* on page 47 and reproduced below for clarity [37]:

1. The client cannot remove the encryption from the file and send it to a [peer;]
2. The client cannot "clone" its DRM system to make it run on another [host;]
3. The client obeys the rules set out in the DRM [license; and]
4. The client cannot separate the rules from the payload.

In practice, the file's purchaser can easily violate these rules, perhaps for legal reasons (such as converting the DRM-protected file to an "unprotected" file format *for personal, fair use*). Therefore, to enforce the DRM license, the trusted system must

set firm, fine-grained controls that specify how the user may access the file—*and enforce the controls*.[45] These types of controls are called *mandatory access controls* (MACs), as the (trusted) computer system *must* enforce them and the user *cannot* override them [270]. They stand in stark contrast to controls implemented on most "mainstream" operating systems (OSes), such as Windows and most versions of Linux, in which the user *can* change file access privileges at his discretion [250]. Hence, these latter controls are called *discretionary access controls* (DACs). MAC enforcement on a trusted system relies on two key concepts—a *reference monitor* and a *trusted computing base* (TCB)—as Schneier explains in his book *Secrets and Lies* [250, 270]:

> - **Reference monitor**. A piece of software that mediates all accesses to objects by subjects.[46] When some process makes an operating system call, the reference monitor halts the process and figures out whether the call should be allowed or forbidden. . . .[47]
> - **Trusted computing base**. All the protection mechanisms inside the computer—hardware, firmware, operating system, software applications, everything—that are responsible for enforcing the security policy. That is, some administrator somewhere tells the computer what is supposed to be secured from whom in what way (that's the security policy), and the trusted computing base enforces it.

[45] More generally, these controls are part of a "larger" security policy [270]. For instance, the user may allow other users to read "public" files he creates, and he may grant read-and-write access to his files associated with a team project. If the user's computer is owned by an organization, the organization may, in turn, impose its *own* policy on all computer files, such as mandating their confidentiality.

[46] As Schneier explains, access control uses the terminology of "subjects" and "objects" [270]: "There is some 'subject' that has access to some 'object.' Often the subject is a user and the object is a computer file, but not always. The subject could be a computer program or process, and the object [could be] another computer program. . . . The object could be a database record." [270]

[47] For instance, the user may try to play a DRM-protected audio file whose license specifies a maximum of three "plays." If he tried to play it a *fourth* time, the reference monitor would intercept the respective system calls, check the license, and refuse to play the file.

Reid and Caelli add that mainstream OSes do not implement the *principle of least privilege*, which holds that a user or program should have *only* the minimum number of privileges required to complete a job [250, 262].[48] Furthermore, as many OSes have only two classes of users—regular users and "super-users"—and super-users are not bound by any access controls, they can easily access the plaintext of DRM-protected files, thereby "bypassing" the DRM license terms [250]. In the researchers' view, TC is essential to DRM license enforcement, and only trusted systems with hardware-based security mechanisms and the aforementioned OS restrictions can make DRM economically viable [250].[49]

While DRM enforcement is a popular TC use case, it is certainly not the *only* use of TC, as we have seen. The primary TC technologies that realize trusted systems are discussed in the next section.

3.3 Primary TC Technologies

The combination of five technologies—the endorsement key (EK), secure input and output (I/O), memory curtaining, sealed storage, and remote attestation—describe TC in practice. Each one is detailed below.

[48]There are exceptions: both the NSA's Security Enhanced Linux (SELinux) and Windows Vista support the principle of least privilege. However, *previous* versions of Windows and "ordinary" Linux do not [148, 236]. SELinux also supports MAC enforcement [236].

[49]Aspects of TC hardware and OSs are described further in Chapters 4 and 5. Ethical issues surrounding the technology, particularly with respect to civil liberties and the economy, are explored in Chapter 8.

3.3.1 The Endorsement Key (EK)

Every trusted computer has a small chip, the Trusted Platform Module (TPM), soldered to its motherboard.[50] Each TPM has a pair of 2,048-bit RSA encryption keys—one public, one private—called the *Endorsement Key* (EK) [261, 348]. As IBM researcher David Safford explains, the EK "is created randomly on the chip at manufacture time and cannot be changed. The private key never leaves the chip, while the public key is used for...encryption of sensitive data sent to the chip" [261]. He adds that the EK's *public key* can be disabled by the computer's owner for privacy reasons [261]. In TC applications, the EK's *private key* is used to uniquely identify a trusted computer [348]. Not only does the EK prevent TPM software emulators from interacting with trusted computers using TCG protocols, it prevents Alice from masquerading as Bob during a "trusted" transaction (and vice versa) [348]. Therefore, a "challenger" in such a transaction can prove that he is truly "talking" with Alice's computer and *not* some other computer masquerading as Alice's. This can prevent man-in-the-middle attacks in these transactions.

3.3.2 Secure Input and Output (I/O)

Secure I/O ensures that no other program or process can "intercept" the user's input to the or the output from the trusted computer. As Electronic Frontier Foundation (EFF) technologist Seth Schoen explains, secure I/O is effected by the provision of a "secure hardware path from the keyboard [and mouse] to an application...and from the application back to the screen. No other software running on the same [trusted computer] will be able to determine what the user typed, or how

[50] TPMs are discussed in greater detail in Chapter 4.

the application responded" [278]. The use of checksums verifies that no tampering has occurred [348]. This technology aims to foil keyloggers, which are implicated in identity theft, and *screen scrapers*, which are programs that automatically capture an application's output to the screen [278, 347]. It is also called a *trusted path* [348].

3.3.3 Memory Curtaining

Memory curtaining extends traditional operating-system memory protection by enforcing, in hardware, the operating-system security maxim that applications cannot read from or write to each other's memory [278]. Consequently, malware programs cannot read other programs' memory regions or write malicious code into them. Moreover, memory curtaining provides complete hardware-based isolation of selected regions of memory, such as those storing cryptographic keys; even the TC OS cannot access these regions [278, 348]. As a result, the security of sensitive data is preserved even if an attacker gains full control of the OS [278, 348].

3.3.4 Sealed Storage

Sealed storage solves the problem of safely storing private information, including cryptographic keys and passwords [278, 348]. Without it, the information as well as the keys and passwords used to protect it are stored *together* on the hard drive, allowing an attacker to easily access the information [278, 348]. Sealed storage solves this problem by *dynamically* generating keys to encrypt and decrypt information based on the hardware and software used to create it [278, 348]. Schoen explains how the technology can be used to protect private information such as a diary [278]:

> [Suppose] you keep a private diary on your PC today. You want to prevent the diary from being moved off your computer without your permission, much as you might lock a paper diary inside a desk drawer. While existing access control and encryption systems address this goal, they might be bypassed or subverted. If someone compromises your system, or it becomes infected with a worm or virus, local software could be altered, or private documents could be e-mailed or copied to other computers.. . .
>
> You can encrypt your diary using a password, but if your password is short, someone who can copy the encrypted diary will still be able to decrypt it [by a brute force attack]. What's more, if the encryption software you use, or the editor in which you compose the diary, is surreptitiously replaced with a modified version, it might leak the decrypted diary's text (or your password) to a third party.
>
> Sealed storage can work together with memory curtaining and secure I/O to ensure that your diary can only be read on your computer, and only by the particular software with which you created it. Even if a virus or worm. . .leaks your encrypted diary, the recipient will not be able to decrypt it. If an intruder or a virus surreptitiously alters your encryption software, it will not longer be able to decrypt the diary, so the contents of your diary will remain protected.

3.3.5 Remote Attestation

Remote attestation detects "unauthorized" software changes on a trusted computer and, upon user request, informs a remote third party—perhaps the software vendor or system administrator—of these changes [278,348]. Its purpose is to notify an authority if any software tampering has occurred so the tampered computer can be taken offline an fixed, thereby mitigating any damage to other networked computers [278]. Remote attestation works by generating a digital certificate in the TPM that describes, via cryptographic hashes, what software is executing on the user's computer, and sending the certificate to the remote party upon user request [278,348]. The remote party examines the certificate's value and compares it to the (hashed) values of "known good" PC configurations; if they do not match,

the remote party can inform the user of this or take any other action. A common use case is software "validation" to ensure that technological protection measures are not bypassed [348]. As one might imagine, this technology makes many software vendor abuses possible, such as vendor lock-in. The ethical implications of remote attestation are discussed in Chapter 8.

The applications of TC technologies are straightforward. Certainly, they facilitate robust DRM enforcement as described in section 3.2. Using secure I/O, memory curtaining, and sealed storage, a content provider can ensure that *only* purchasers of its media can play it in an "approved" media player in accord with the license terms and *not* record the media player output with another application [348]. Using the EK, memory curtaining, and remote attestation, online auctioneers can require that all buyers and sellers use computers with TPMs and "approved" applications to conduct transactions. Secure I/O and memory curtaining can prevent malware "infections." However, one application *is* used in practice: protection of hard-drive data after theft [348]. The Enterprise and Ultimate versions of Windows Vista include BitLocker Drive Encryption, a feature that automatically encrypts the `C:\` drive [274,348]. BitLocker uses the TPM to perform integrity measurements in the BIOS, master boot record, NTFS boot sector, and so forth until it retrieves hard drive decryption keys; the system can require the user to enter a password and/or insert a USB key drive [220,274,348]. In addition, the Linux program Enforcer uses the TPM to detect and prevent any changes to the Linux file system [348,349].

The next three chapters examine the "nuts and bolts" of trusted computer systems. Chapter 4 discusses the TPM and hardware support for TC. Chapter

5 discusses TC-supporting operating systems such as Windows Vista. Finally, Chapter 6 discusses TC software.

CHAPTER 4

TC HARDWARE

This chapter begins by discussing the core TC hardware component, the Trusted Platform Module (TPM). Next, TC-supporting BIOSes and the Extensible Firmware Interface (EFI)—Intel's successor to the BIOS—are detailed. A brief description of TC-supporting CPUs follows. The chapter concludes with a lengthy description of Intel's High-bandwidth Digital Content Protection (HDCP) system, which "protects" content until it "reaches" the display screen and/or audio device.

4.1 Trusted Platform Module (TPM)

The TPM is a chip that is soldered to a trusted computer's motherboard. It securely stores cryptographic keys and accurately measures and reports the state of a TC platform [312]. Before examining the TPM in detail, it is useful to discuss several basic characteristics of a trusted platform and TPM functionality.

According to the TCG, a trusted platform must implement *protected capabilities*, *integrity measurement*, and *integrity reporting*, as well as the "basic functionality" needed to extend that trust to other entities [312].[51] The TCG states that "protected

[51]The TCG defines *trust* as "the expectation that a device will behave in a particular manner for a particular purpose" [323].

capabilities are a set of commands with exclusive permission to access shielded locations. Shielded locations are places (memory, registers, etc.) where it is safe to operate on sensitive data; [these locations] can be accessed only by protected capabilities" [312]. Both these features are implemented with Platform Configuration Registers (PCRs) on the TPM, which are explained later [312]. The concept of *attestation* also plays a role in a trusted platform [312]. *Attestation* is "the process of vouching for the accuracy of information," and there are several ways to attest that a computer system is "trusted" [312]. *Integrity measurement* is "the process of obtaining metrics of platform characteristics that affect the [trustworthiness] of a platform; storing those metrics; and putting digests of those metrics in PCRs" [312].[52] *Integrity reporting* is "the process of attesting to the contents of [logged integrity metrics and their PCR digests]" [312]. In addition, a trusted platform must provide "basic" components, or *Roots of Trust*, that are *automatically* trusted, since one cannot easily detect their misbehavior [312]. The TCG defines three of these: the *Root of Trust for Measurement* (RTM), the *Root of Trust for Storage* (RTS), and the *Root of Trust for Reporting* (RTR) [312]. The RTM, RTS, and RTR are "low-level" programs that make accurate integrity measurements, maintain accurate summaries of integrity digests, and report accurate integrity digests, respectively [312].[53] Other Roots of Trust that *do not* have shielded locations or protected capabilities are denoted *Trusted Building Blocks* (TBBs), and they are illustrated in Figure 4.1 on page 63. These Roots of Trust are bounded by a "trust boundary" that other entities may

[52] *Metrics* are "[representations] of embedded data or program code," and *digests* are SHA-1 hashes of metrics [312].

[53] These "low-level" firmware programs are frequently called *engines* in specifications and vendor marketing. The "core program" that performs integrity measurement is called the *Core Root of Trust for Measurement* (CRTM) [312]. It will be shown that the CRTM is a TPM component.

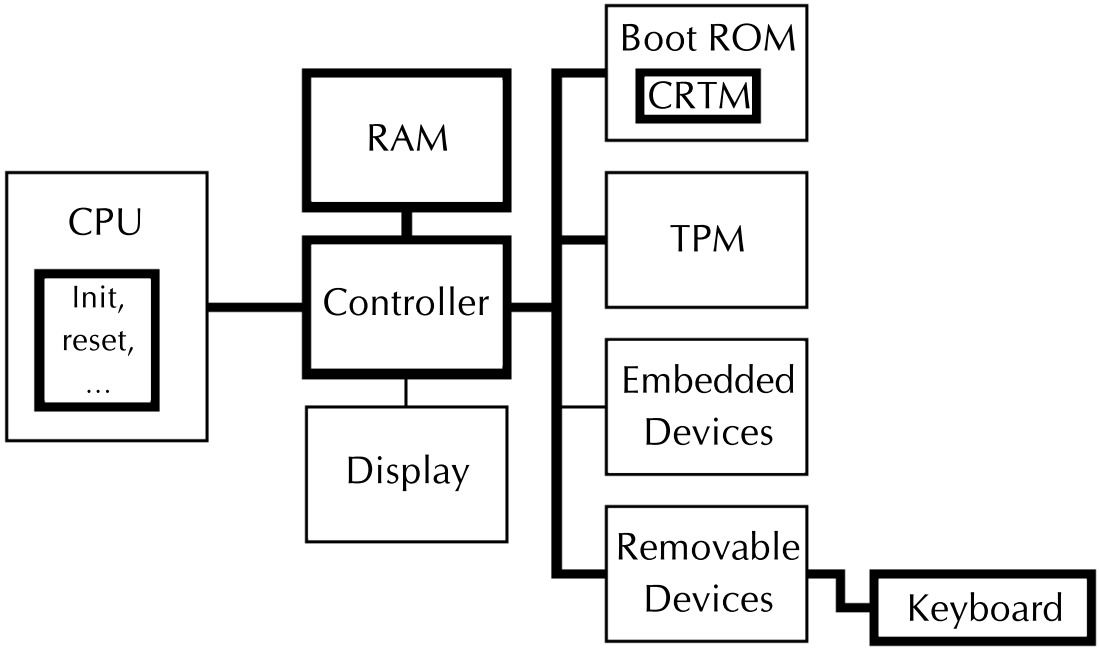

Figure 4.1: Trusted Building Blocks (in Bold) of a Trusted Platform. (Adapted from Figure 4:b in [312].)

wish to expand [312]. For instance, a TC OS may wish to extend the boundary to itself. In order to do so, it must confirm that these Roots of Trust can indeed be trusted, *i.e.*, that there is no malicious code lurking in the BIOS or low-level system code [312]. If the OS determines that they *are* trusted, it extends the trust boundary around itself [312]. Likewise, TC applications may extend the trust boundary around themselves. This process is called *transitive trust* or *inductive trust*, and it is illustrated in Figure 4.2 on page 64.

The TPM stores various *non-migratable* keys: the *endorsement key* (EK), the *storage root key* (SRK), and *attestation identity keys* (AIKs) [312].[54] As described in Section

[54] As their names suggest, *non-migratable keys* cannot be "moved off" the TPM, whereas *migratable keys* can [312].

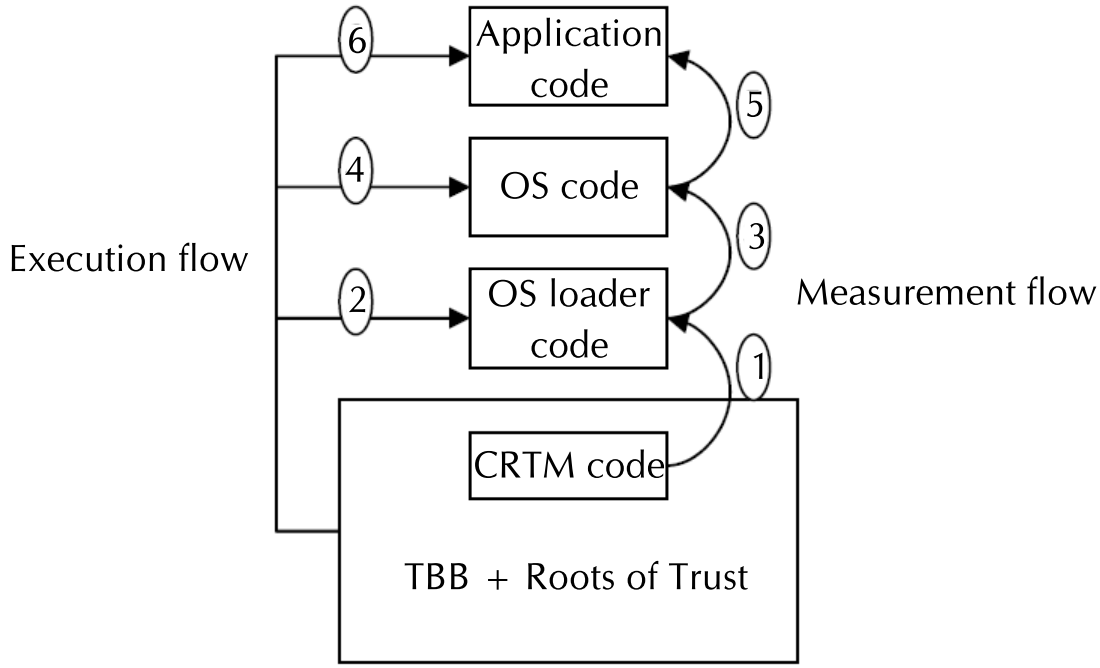

Figure 4.2: Transitive Trust from Hardware to Software. (Adapted from Figure 4:c [312].)

3.3.1, the EK is a pair of RSA public and private keys that uniquely identify a given TPM; both keys must be at least 2,048 bits long [261, 348]. The SRK is a "root key" that manages various "storage keys" in the RTS [312].[55] AIKs are asymmetric cryptographic keys that the TPM generates. In various attestation protocols, they digitally sign PCR values and are associated with a certification authority (Trent) [312]. In this context, Trent's credentials for a given AIK can only be decrypted—and hence verified—by the TPM that "owns" that AIK, and the EK *private* key is required for verification [312].

[55]Further details of the RTS functionality are beyond the scope of this paper.

[55]These attestation protocols use a cryptographic technique called *direct anonymous attestation* [44, 323]. Discussion of these protocols is beyond the scope of this paper.

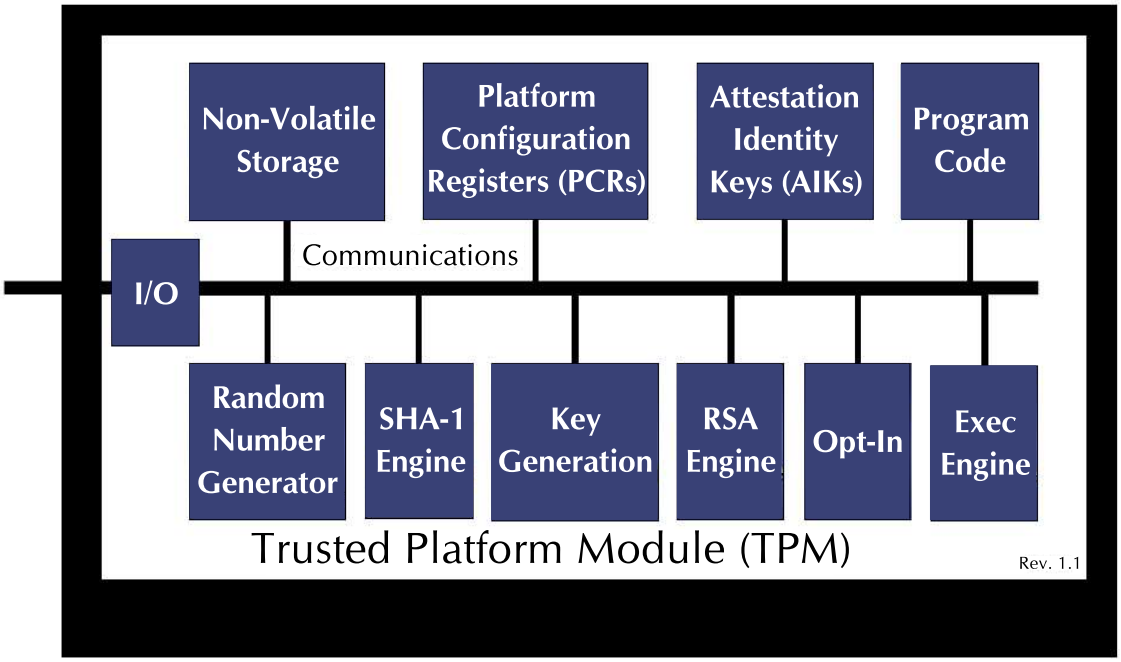

Figure 4.3: Trusted Platform Module. (Adapted from Figure 4:g in [312].)

In addition, the TPM performs the following cryptographic message-exchange operations [312]:

- *Binding*, which is the TCG term for encryption with a public key;
- *Signing* a message (or the hash thereof);
- *Sealing*, which encrypts a message and "ties" it to "a set of [PCR] platform metrics specified by the message sender" such that these metrics must hold before decryption occurs; and
- *Sealed signing*, in which a signed message includes a set of PCR values and the verifier checks that those values are still intact.

We now examine the TPM itself, which is illustrated in Figure 4.3 above. Each TPM component in the figure is examined in turn. Unless otherwise stated, all information herein is from [312].

- *I/O*: Self-explanatory.
- *Non-Volatile Storage*: Persistently stores the EK, SRK, "owner authorization data and persistent flags."
- *Platform Configuration Registers (PCRs)*: These registers store various integrity measurements of the TC system. The TCG mandates that TPM manufacturers have at least 16 registers. The first 8 registers (0–7) are for TPM use; the latter 8 registers are for OS and application use. As the values in PCRs are cleared whenever power is turned off, PCRs may be implemented in *volatile* storage (not shown).
- *Attestation Identity Keys (AIKs)*: AIKs are asymmetric keys as described on page 64. Their associated private keys are non-migratable and protected by the TPM. Their associated public keys are issued by Trent, thereby establishing the TPM's validity [323]. TPMs can store multiple AIKs in encrypted form.
- *Program Code*: This code is executed at the beginning of the boot sequence and is the Core Root of Trust for Measurement (CRTM). Since it is assumed to be "trusted," all subsequent platform measurements are based on the accuracy of its results. The code *should* be stored on the TPM, but this is not necessary.

- *Random Number Generator*: Self-explanatory. It has a "true" random-bit generator to create keys and other cryptographic primitives.[56]

- *SHA-1 Engine*: As its name suggests, this computes the SHA-1 hash of some provided "message," such as the value of a key. The hash value may be used as a digital signature or to generate encrypted keys, among other purposes. Note that SHA-1 is somewhat insecure: in 2005, a Chinese research group only needed 2^{69} computations to find a collision, *i.e.*, two different "messages" that yield the same hash value [271, 350]. Since a brute-force collision attack requires 2^{80} computations, the Chinese group's results decrease the time to find a collision by a factor of $2^{11} = 2048$. However, it would likely require *a few years* to find a collision with today's technology, even after factoring in Moore's Law [271].

- *RSA Key Generation*: TPMs use the RSA one-way function shown on page 14 to generate keys. TPMs support 2,048-bit RSA keys and the TCG *requires* that some keys, including AIKs, have at least 2,048 bits.

- *RSA Engine*: TPMs use the RSA algorithm for encryption and decryption, and this component implements RSA.

- *Opt-In*: Under TCG policy, users may choose to activate or deactivate their TPMs. This component facilitates this functionality.

- *Execution Engine*: As its name suggests, "the execution engine runs program code." Specifically, "it performs TPM initialization and measurement taking."

[56]The TCG does not specify *how* such a "true" random-bit generator is to be implemented.

Finally, TPMs have three mutually-exclusive sets of states, as the TCG describes below (emphasis added) [312]:

- *Enabled/Disabled* — The TPM may be enabled/disabled multiple times within a boot period. When disabled, the TPM restricts all operations except the ability to report TPM capabilities and to accept updates to PCRs.[57] When enabled, all features of the TPM are available.
- *Activated/Deactivated* — Deactivation is similar to disabled, but operational state changes are possible (*i.e.*, change owner or activation with physical presence). When activated all features of the TPM are available.
- *Owned/Un-owned* — A platform is owned when an EK exists and the true owner knows owner authorization data. The owner of a [trusted] platform may perform all operations including operational state changes.

Note that TPMs are shipped *disabled*, *deactivated*, and *un-owned* by default [24, 203]. This ensures that the computer owner must physically enable it, not local or remote software [24].

4.2 BIOS/Extensible Firmware Interface (EFI)

Today, the vast majority of BIOSes can communicate with TPMs. In January 2003, American Megatrends' AMIBIOS contained a TCG-compliant module, and, later that same year, Phoenix Technologies redesigned its BIOS architecture to support TPMs [1,60,352].[58] Phoenix's present BIOS firmware, which the company dubs "TrustedCore," includes a low-level cryptographic "engine" that can authenticate

[57] PCR values are updated as follows: $PCR[n] \leftarrow SHA\text{-}1(PCR[n] + \textit{measured data})$, where $PCR[n]$ is the value of the n^{th} PCR and SHA-1() denotes the SHA-1 hash function. [312].

[58] At that time, the TCG named itself the Trusted Computing Platform Alliance (TCPA) and the TPM was not named as such. Hence, American Megatrends marketed its BIOS' "TCPA-compliant module," not the "TPM support" seen in vendor marketing today [1,246].

the firmware itself, thereby providing CRTM functionality *outside* the TPM [246]. TrustedCore also provides pre-boot user authentication mechanisms [246].

In addition, Intel's Extensible Firmware Interface (EFI) replaces the BIOS altogether. As the company describes it,

> the EFI specification defines a new model for the interface between operating systems and platform firmware. The interface consists of data tables that contain platform-related information, plus boot and runtime service calls that are available to the operating system and its loader. Together, these provide a standard environment for booting an operating system and running pre-boot applications [164].

Intel has designed a modular framework for the EFI that supports architectural interfaces for arbitrary hardware, "protected-mode memory and address space management tailored to. . .preboot environments," and, yes, a trusted platform with its associated protocols [160, 314, 315]. Over one million computers shipped with the framework in 2005 [160].

It is noteworthy that virtually *every* hardware vendor's products support the "trusted platform" concept [203]. Tony McFadden, who maintained an online list of known TPM vendors and hardware vendors with TPM-supporting products, remarked in March 2006 that

> today, every meaningful vendor has [trusted platforms] in [its] roadmap and Microsoft has an OS that requires TPM version 1.2. . .for enterprise users before the end of this year.[59] There are no more surprises – the paradigm has shifted.. . .Adding new platforms is, for the most part, meaningless. It would be easier (and shorter) to maintain a list of vendors/platforms *without* TPM [support] (and some of my correspondents – of the EFF, anti-DRM ilk – would probably prefer that) (emphasis in original) [203].

[59] Here, McFadden refers to Windows Vista, whose TC "features" are discussed in Chapter 5.

4.3 TC-supporting CPUs

Presently, many CPUs support TPM instructions in their chipsets, including Intel's vPro™, AMD's Opteron, and Transmeta's Crusoe processors [55, 142, 162, 163, 227, 309].[60] CPU vendors have various trade names for their TPM-supporting technologies: Intel calls its technology *Trusted Execution Technology* (TXT), AMD calls its technology *Secure Execution Mode*, and HP, IBM, and other vendors have similar names [227].[61] Due to Intel's dominance in the CPU market and a lack of publicly-available information on its competitors' TPM-supporting chipsets, Intel's x86 architecture and enhancements thereto are discussed here in detail.

Recall from Section 3.2 that most "mainstream" OSes only support two major access modes: one for "regular" users and one for super-users [250]. However, OSes must also allow device drivers and OS extensions to access the system with varying degrees of privilege [250]. The x86 architecture provides support for these types of access controls via a series of concentric *rings*, as shown in Figure 4.4 on page 71 [250]. The innermost ring, *ring 0*, has the greatest privilege, whereas the outermost ring, *ring 3*, has the least privilege [250]. The OS kernel and any associated reference monitor execute in ring 0 to provide tamper-resistance for mandatory access controls [250]. OS device drivers, extensions, and applications

[60] Note that TPM support in a chipset *does not* imply that the CPU actively restricts "unauthorized" content, or even that the TPM support is *enabled* for a particular chip! For instance, AMD's Opteron CPU *does not* restrict content, and though previous Intel Pentium 4 CPUs *supported* multithreading, this support was not *enabled* [142, 299].

[61] Intel's TXT was previously known as *LaGrande Technology* (LT), which appears frequently in earlier technical literature [161, 237].

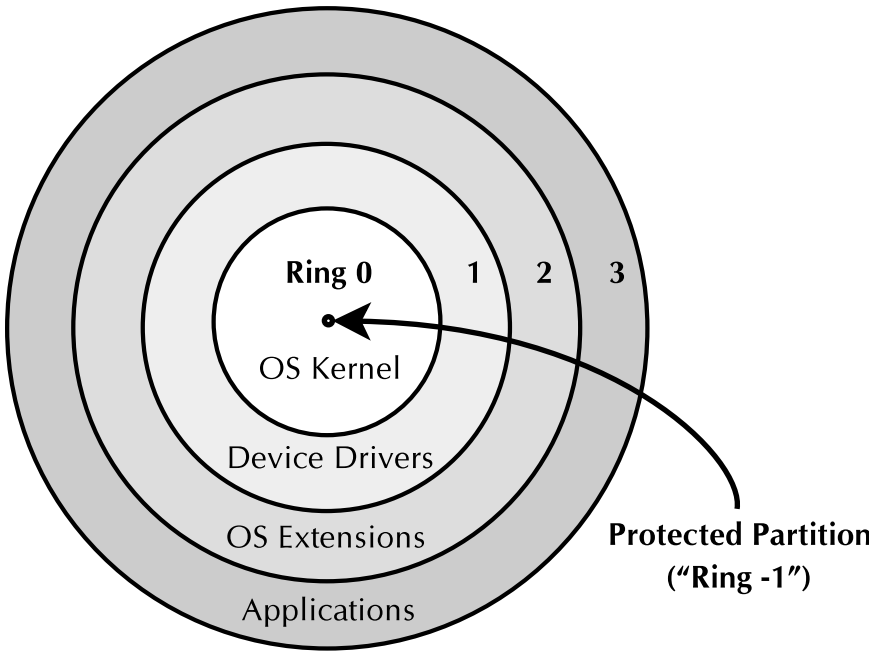

Figure 4.4: Intel x86 Ring Architecture. (Adapted from Figure 1 in [250].)

execute in consecutively higher-numbered rings with respectively lower privileges [250].[62]

Intel's TXT introduces a *protected environment*, or "Ring -1," in which running applications are isolated from the rest of the system "such that no. . .unauthorized software on the platform can observe or compromise the information being operated upon" [162]. This environment implements memory curtaining and sealed storage and encrypts both USB keyboard and mouse input and graphics framebuffer output, thereby preventing other executing programs from determining user input or program output [162]. In addition, TXT attests that the protected environment and/or "protected" OS was correctly loaded and "measures" running

[62]Reid and Caelli argue that, in practice, mainstream operating systems do *not* use rings 1 and 2 very often, just rings 0 and 3 "for system and user [space,] respectively" [250, 299]. They add that, consequently, these OSes cannot robustly enforce DRM policies [250].

software, just as the TPM attests to the validity of the boot sequence and measures system integrity [162,312]. This attestation can be provided to a remote entity upon request [162]. TXT can manage both protected and "standard," *i.e.*, "unprotected," environments called *partitions* via virtualization technology, a description of which follows [162]. A number of virtual machines (VMs) execute on a virtualization-supporting CPU, and each VM executes an OS and programs in a protected or standard partition [162]. Protected partitions use the TPM and TXT to execute applications in isolation as described above; standard partitions run applications just like non-TXT-supporting CPUs [162]. However, to separate partitions from one another, a "partition manager" program must execute in a ring with *higher* privileges than each OS kernel; Intel calls this program a *domain manager* [162].[63] This "ring -1" approach allows backwards compatibility with existing programs for the x86 architecture [162, 250]. After all, few consumers want their brand-new TC-supporting CPUs to be *un*able to run legacy applications.

4.4 High-bandwidth Digital Content Protection (HDCP) and High Definition Multimedia Interface (HDMI)

High-bandwidth Digital Content Protection (HDCP) is a DRM system developed by Intel subsidiary Digital Content Protection, LLC that encrypts audiovisual data traveling over the Digital Visual Interface (DVI) or High Definition Multimedia Interface (HDMI) [345]. Before examining HDCP, it is useful to understand DVI and HDMI. DVI transmits video from a computer to a display screen in *digital* format—*i.e.*, a stream of bits—instead of an *analog* signal. Though a single DVI

[63]Note that, in our example, the domain manager is a VM monitor [162]. The domain manager is not restricted to the virtualization example explained above; Intel notes that a "future, enhanced OS kernel" could serve as the domain manager [162].

"link" has 3.30 Gbps of bandwidth, *i.e.*, it can easily transfer 60 images per second at 1920 × 1080 progressive-scan resolution (*1080p* resolution at *60 Hz*), HDMI supports 5 Gbps bandwidth, including support for 8-channel audio *along with* 1080p resolution at 60 Hz [170, 288].[64] HDMI "includes" HDCP by default, which is now explained in detail.

As the astute reader has likely deduced, HDCP is a proprietary DRM system; electronics manufacturers who wish to use it must sign a license agreement and pay respective fees [62]. HDCP is a *content* protection system, not a *copy* protection system, though copy protection of high-definition (HD) Hollywood content is *the* impetus behind HDCP's development [99, 189]. At its core, HDCP authenticates a content sender (such as a PC) and a content receiver (such as an HD TV) so each device "knows" the other supports HDCP [85, 189]. Felten describes this authentication protocol in depth [85]:

> Every new HDCP device is given two things: a secret vector, and an addition rule. The secret vector is a sequence of 40 secret numbers that the device is not supposed to reveal to anybody. The addition rule, which is not a secret, describes a way of adding up numbers selected from a vector. Both the secret vector and the addition rule are assigned by HDCP's central authority. . .
>
> An example will help to make this clear. In the example, we'll save space by pretending that the vectors have four secret numbers rather than forty, but the idea will be the same. Let's say the central authority issues the following values:

	Secret Vector	Addition Rule
Alice	(26, 19, 12, 7)	[1] + [2]
Bob	(13, 13, 22, 5)	[2] + [4]
. . .	. . .	. . .

[64] Based on graphics cards' actual DVI interfaces, higher resolutions may be obtained [288]. HDMI *only* supports 5 Gbps bandwidth "over copper interconnects up to 15 feet" [170].

> Suppose Alice and Bob want to do a handshake. Here's how it works. First, Alice and Bob send each other their addition rules. Then, Alice applies Bob's addition rule to her vector. Bob's addition rule is "[2] + [4]," which means that Alice should take the second and fourth elements of her secret vector and add them together. Alice adds 19 + 7, and gets 26. In the same way, Bob applies Alice's addition rule to his secret vector—he adds 13 + 13, and gets 26. (In real life, the numbers are much bigger—about 17 digits.)
>
> There are two things to notice about this process. First, in order to do it, you need to know either Alice or Bob's secret vector. This means that Alice and Bob are the only ones who will know the result. Second, Alice and Bob both got the same answer: 26. This wasn't a coincidence. There's a special mathematical [formula] that the central authority uses in generating the secret vectors to ensure that the two parties to any legitimate handshake will always get the same answer.
>
> Now both Alice and Bob have. . .a secret key. . .that only they know. They can use the key to authenticate each other, and to encrypt messages to each other.

However, it is possible for Alice, Bob, Carol, and Dave to conspire to find the secret vector of a victim, who Felten calls "Ed," by a cryptanalytic attack described by Scott Crosby *et al.* [85].[65] The conspirators use their secret vectors and addition rules to construct and solve a linear system of four equations in four unknowns x_1, x_2, x_3, and x_4, which are precisely the values of Ed's secret vector $[x_1, x_2, x_3, x_4]$ [85]. This *conspiracy attack* easily scales to the 40 x's in an arbitrary secret vector: simply construct a system of 40 equations in 40 unknowns $[x_1, x_2, \ldots, x_{40}]$ and solve for each $x_i, 1 \leq i \leq 40$ [85]. The weakness of this protocol, and HDCP by extension, stems from the *linear* nature of this key exchange; as Felten points out, the cryptography would have been much stronger if HDCP's designers used a *non-linear* key exchange protocol such as Diffie-Hellman [82].[66] However,

[65] Mathematical details of the attack are omitted for brevity. See [56] for a formal description of the attack. "Proof-of-concept" Java code implementing the attack is available online at [201].

[66] Details of the Diffie-Hellman key exchange are beyond the scope of this paper.

HDCP provides a mechanism to revoke known compromised keys so devices using those keys cannot view HD content [189]. It also provides a *constricting* "feature" for Hollywood studios to scale down HD content on those monitors and TVs that do not support HDCP [99]. Already, PlayStation 3 owners have experienced flickering TV screens due to the latter devices' lack of HDCP support [103].

Crosby *et al.* note an HDCP requirement specifying a *maximum* of 10,000 logic gates for the cryptographic hardware implementation; they add that about *30,000 gates* would be necessary to implement secure HDCP cryptography [56,82,83]. Felten argues that HDCP encryption exists "just" as a means to sue reverse engineers under the DMCA (emphasis added) [83]:

> HDCP encryption exists only as a hook on which to hang lawsuits. For example, if somebody makes unlicensed displays or format converters, copyright owners could try to sue them under the DMCA for circumventing the encryption. (Also, converter box vendors who accepted HDCP's license terms might sue vendors who didn't accept those terms.) The price of enabling these lawsuits is to add the cost of 10,000 gates to every high-def TV or video source, and to add another way in which high-def video devices can be incompatible.. . .The bottom line is clear. In HDCP, "security" technologies serve not to disable pirates but to enable lawsuits. *When you buy an HDCP-enabled TV or player, you are paying for this—your device will cost more and do less.*

For his part, Crosby has vowed not to examine any more copyright-protection schemes "as long as the DMCA exists in its current form," especially in light of the Skylarov case and cryptographer Niels Ferguson's refusal to publish his cryptanalysis of HDCP [57]. Recall from Section 2.2 that Ferguson found critical weaknesses in HDCP but refused to publish them out of fear of his arrest in the U.S. for violating the DMCA [93, 94]. These examples clearly illustrate that the

DMCA unjustly silences researchers who perform a public service by exposing flawed encryption algorithms and implementations.

Having thoroughly examined TC hardware and associated protocols, TC-supporting operating systems are examined in the next chapter. In particular, the TC and DRM "features" of Windows Vista, the latest version of Microsoft's flagship OS, are discussed. Vista is then contrasted with Mac OS X and GNU/Linux, which are more "consumer-friendly."

CHAPTER 5

TC OPERATING SYSTEMS

This chapter examines operating systems that support TC, hereafter referred to as *TC operating systems* or *TC OSes*. Initially, the TC and DRM "features" of Microsoft's Windows Vista OS are examined, followed by Microsoft's Vista licensing policies. A brief comparison of Windows XP and Windows Vista is provided. The DRM and *lack* of TC support in Apple's Mac OS X is then discussed. The chapter concludes with a brief examination of TC support under Linux.

5.1 Windows Vista

Windows Vista "features" pervasive DRM support throughout the operating system as well as TC support for high-end versions of Vista. Before delving into details, however, it is useful to review Microsoft's licensing policies for Vista, as they "set the stage" for DRM enforcement and TC support.

5.1.1 License Policies

Microsoft's *End-User License Agreements* (EULAs) for the various versions of Windows Vista are *the* most restrictive licenses the company has issued for its

OSes.[67] The Vista EULAs mandate that users "activate" their copies of the OS with Microsoft within 30 days, after which Vista locks them out of their computers [77, 207, 208, 223, 224].[68] At this point, Vista operates in *Reduced Functionality Mode*, in which users may only use a Web browser and Vista logs them off after an hour without warning [77, 223, 224]. During the time period between Vista installation and activation, users may not access the "glassy" Windows Aero interface, remove "non-critical" malware with Windows Defender, augment their computer's memory with Windows ReadyBoost, or download OS "add-ons" from Windows Update *except* "critical" security updates [77,223,224].[69] The text "This copy of Windows is not genuine" also appears on the "desktop" during this time, and users receive "warning messages" stating that they must validate their copy of Windows as "genuine" [77,223,224]. In addition, the Vista EULAs allow Microsoft to periodically "validate" the OS to ensure that it "has been activated and is properly licensed" as well as download validation software to users' computers and update the validation software [207, 208]. If users upgrade their computers' hardware, attempt to work around the activation process, or use pirated "product keys," Microsoft requires reactivation of Vista within 30 days; as before, if they do not reactivate

[67] While license terms differ slightly for the six Vista versions—Vista Starter, Vista Home Basic, Vista Home Premium, Vista Business, Vista Enterprise, and Vista Ultimate—the respective EULAs are so similar that we collectively refer to them as *Vista EULAs* or *Vista licensing* [100, 207, 208]. We do not discuss Vista Home N and Vista Business N, which are versions for European markets [308].

[68] *Activation* is the process in which Microsoft "associates the use of [its] software with a specific [computer]" by recording "the version, language, and product key of [Windows], the Internet protocol address of the [computer], and information derived from [the computer hardware]" [207, 208].

[69] *Windows Aero* is Vista's semi-transparent user interface that is available in Vista Home Premium, Vista Business, and Vista Ultimate [223, 224]. *Windows Defender* is the company's malware-removal program that is available on XP and Vista [223, 224]. *Windows ReadyBoost* allows Vista users to use free space on USB "flash drives" to effectively "increase" computer memory [223,224,295]. *Windows Update* is a Microsoft service that provides security updates and OS add-ons (such as Microsoft's Web browser and media player) for users to download [224].

Vista, the OS enters Reduced Functionality Mode [207, 208, 224]. Vista's Software Protection Platform encompasses these activation and validation "features," which Microsoft claims are necessary to thwart software piracy [77, 223, 224]. Moreover, the Vista EULAs stipulate that users "must comply with any technical limitations in [the OS] that only allow [it to be used] in certain ways" [207, 208]. As Professor Michael Geist of Canada's University of Ottawa Law School points out, "In the name of shielding consumers from computer viruses and protecting copyright owners from potential infringement, Vista seemingly wrestles control of the 'user experience' from the user" [123]. His point becomes clearer as Vista's DRM and TC support are examined below.

5.1.2 DRM support

Windows Vista is designed to robustly enforce DRM-protected "premium content" from the software application that "plays" the content to the consumer's monitor and speakers.[70] Vista introduces new "protected environments" for video and audio that allow output "ports" to be disabled—or audiovisual output to be "constricted"—if premium content is not encrypted [137, 196–198].[71] For example, video passing through "unprotected" analog ports such as *Video Graphics Array* (VGA) and "component video" is intentionally constricted or even shut off per

[70] In this paper, we follow Microsoft's definition of *premium content* as "valuable [audiovisual] content that needs to be protected from stealing" [197]. This definition encompasses existing commercial content as well as high-definition audio and video [137, 197].

[71] In this context, *constriction* is the process by which a video's resolution is reduced to a level considered acceptable to the content owner and subsequently scaled to its original resolution [197]. As Dave Marsh, Microsoft's Lead Program Manager for Video, describes it, "a high-resolution picture needs to be degraded to make it soft and fuzzy" [197, 199]. In practice, premium content video is constricted to 520,000 pixels—approximately 840×630 resolution—and then upscaled [197]. It is noteworthy that *only* this premium content is constricted, not any other image data on the user's desktop [199].

the content owner's specification [197].[72] Digital Video Interface (DVI) output that is *not* HDCP-protected is also constricted or blocked; only HDCP-encrypted DVI or HDMI output is transmitted in its original resolution [137, 197]. Moreover, both users' graphics cards and monitors must support HDCP for HD premium content to be displayed in full resolution without constriction or signal blockage [99, 137]. Yet few graphics cards and monitors on the market support HDCP at present, meaning users cannot view HD video on their supposedly "HD-ready" hardware [99, 137]. In order to output premium content, Microsoft requires manufacturers of graphics cards to keep some details of their cards' functionality secret in order to perform a *Hardware Functionality Scan* (HFS) [99, 137]. The HFS, which is performed before premium content is sent to users' graphics cards, and, ultimately, their monitors, "exercises" this undocumented functionality to ensure the cards are not "emulators" that hackers could use to copy the content [99, 137].[73] Hollywood studios control the cards' designs: the studios' "hardware robustness rules" dictate acceptable designs to graphics-card manufacturers that preclude the construction of "hacker-friendly cards," and Microsoft must approve the graphics-card drivers before they can be used [184, 196–198]. Manufacturers who allow these cards to be built or whose cards "leak" content face Microsoft's revocation of their drivers' permission to "play" premium content as well as "other remedies" [137, 184, 197]. Peter Gutmann, a cryptographer at New Zealand's University

[72] As Marsh explains, *component video* "was the [consumer electronics] industry's first attempt at an interface to [high-definition] displays," but it provides no content protection except the *Copy Generation Management System for Analog* (CGMS-A) [196, 280]. CGMS-A is a means of "marking" analog signals with copyright holders' permissions to record—or not record—those signals [280].

[73] Marsh claims this undocumented functionality will not prevent the development of open-source drivers for new graphics cards [199]. The actual impact of HFS-exclusive graphics-card functionality on the open-source community remains to be seen.

of Auckland, thoroughly researched Vista's content protection "features," and his industry sources suggested these "remedies" may be multi-million-dollar fines and lawsuits [137]. Manufacturers of *discrete* graphics cards—*i.e.*, those manufacturers whose cards are *not* soldered directly to the motherboard—must also implement 128-bit AES encryption in their cards, which is computationally expensive and prevents manufacturers from adding all possible graphics-processing hardware to their cards [137, 197].[74] Due to the Hollywood-imposed need to encrypt massive amounts of high-definition content, Intel developed the *Cascaded Cipher*, a proprietary version of AES, which manufacturers must license and implement in accord with Intel's specification [197].[75] Furthermore, they must implement "tilt bits" that are set "if at any time the graphics driver determines that something improper has happened. . .for example, if the hash of an output status message doesn't match the message"; if any tilt bits are sent, Vista's graphics subsystem restarts and re-authenticates the graphics card [197].[76] Vista also provides audio

[74]Full details of AES' computational complexity are beyond the scope of this paper. Further computational overhead is incurred whenever the computer sends premium content to the graphics card, as Microsoft's protected environment requires creation of a session key and HFS authentication to communicate with the card [197]. Session-key exchange is discussed in Chapter 2.1. The encryption requirement prevents hackers from "sniffing"—and thereby copying—premium content from the PCI Express bus, on which data travels between the computer and graphics card [197].

[75]Companies *could* design their own ciphers, but Hollywood studios must approve the cipher [197]. As the Windows Vista Output Content specification notes, one acceptance criterion for a new cipher is "Content industry acceptance. . .[Evidence of the cipher's security strength] must be presented to Hollywood and other content owners, and they must agree that it provides the required level of security. Written proof from at least three of the major Hollywood studios is required" [197].

[76]As Gutmann notes, the specification is vague about what conditions "trigger" the setting of a tilt bit [137, 197]. Rather, the specification contains language such as "Adopting [the tilt bit] mechanism is another example of the hardware manufacturer showing [its] intent to properly protect premium conent" [197].

protection measures analogous to those for video, including disability and constriction of audio ports, driver authentication, and audio encryption in transit to the sound card [197].

It readily follows that Vista's content protection may very well degrade system reliability and increase hardware costs, which are passed on to consumers [137,184]. As Gutmann points out, minor voltage fluctuations that occur during normal computer system operation could trigger the setting of a tilt bit, thereby forcing the graphics subsystem to restart and undermining Vista's stability [137]. In a 2005 Windows Hardware Engineering Conference talk, Pete Levinthal, a software-engineering manager at ATI, commented that Vista's content protection necessitated higher design, driver-development, testing, and legal costs, all of which are passed on to the consumer [184, 204]. As Felten notes,

> [In the specification,] performance, cost, and flexibility are sacrificed in a futile effort to prevent [audiovisual] content from leaking to the darknet. And the cost is high. As just one example, nearly all of us will have to discard our PC's monitors and buy ones to take advantage of new features that Microsoft could provide—more easily and at lower cost—on our existing monitors, if Hollywood would only allow it.
>
> There can be little doubt that Microsoft is doing this because Hollywood demands it; and there won't be much doubt among independent security experts that none of these compromises will make a dent in the availability of infringing video online. Law-abiding people will be paying more for PCs, and doing less with them, because of the Hollywood-decreed micromanagement of graphics system design [81].

5.1.3 TC support

Vista supports software TPM interfaces as well as BitLocker Drive Encryption, a TC application in the Enterprise and Ultimate editions [216,308]. BitLocker Drive Encryption, hereafter referred to as *BitLocker*, is part of Microsoft's Next-Generation

Secure Computing Base (NGSCB) technology (formerly known as "Palladium"), which is *not* included in Vista [74]. We discuss BitLocker below, followed by a high-level overview of NGSCB.

BitLocker Drive Encryption

BitLocker Drive Encryption, formerly known as "Secure Startup," is designed to protect users from data theft arising from lost or stolen laptops [95]. To do so, BitLocker uses the TPM to detect tampering during the boot process, encrypts the Windows Vista disk "volume" with 128-bit AES, and stores the encryption key in the TPM, *not* on the hard drive [95, 218, 220].[77] BitLocker uses the TPM's integrity-measurement ability to keep track of executing code during the boot sequence and only unseals the encryption key if the PCR values match a "known good" state [95, 214, 220]. If the PCR values do *not* match this state, which is the same state as at the time of drive encryption, the encryption key is *not* released, Vista will not boot, and the user must type in a "recovery" password to boot Vista and access his data [219].[78] Otherwise, the encryption key is released and Vista boots in a "normal" manner that is transparent to him [218]. The OS simultaneously reads and

[77] In this context, Microsoft defines a *volume* as "an area of storage on a hard disk [that] is formatted [with] a file system, such as NTFS, and has a [Windows] drive letter assigned to it. A volume is different than a [*partition*], which is a portion of a physical disk that functions as though it were a physically separate disk. A volume could exist for each partition on a hard drive, or volumes can span multiple partitions" [220]. In addition to AES, BitLocker uses a *diffuser* that thwarts statistical attacks on the encrypted hard-drive data [95]. Further details of AES and the diffuser are beyond the scope of this paper.

[77] Recall from Section 4.1 that *sealing* encrypts data and "ties" that data to the current TPM and PCR values such that the data can only be decrypted with the same TPM and if the PCR values match those at the time of sealing. *Unsealing* refers to this "data decryption" process.

[78] A system administrator may also require that users type in a *personal information number* (PIN) and/or insert a USB "flash drive" containing a startup key to boot the OS, even if the PCR values match the "known good" state [205, 218, 220]. BitLocker works on computers without TPMs or TCG-compatible BIOSes, but *only* if the latter requirement is specified [220].

decrypts disk data and encrypts files stored to the disk so to minimize performance degradation [95, 214]. However, BitLocker needs two partitions formatted with Microsoft's New Technology File System (NTFS) to operate: one partition, the *system volume*, contains unencrypted boot data, whereas the other partition, the *operating system volume*, contains Vista and user data [219].

The astute reader may surmise that BitLocker prevents the installation of multiple OSes on the same computer. Microsoft does *not* thoroughly address this in its documentation, though the company recommends that users disable BitLocker on all disk partitions before installing multiple OSes [214]. Certainly, BitLocker would detect the replacement of the master boot record (MBR) during installation of other OSes, thereby preventing Vista from booting unless the user "recovered" from it [214]. However, users can work around this to boot multiple operating systems, as Schneier describes [274]:

> If you have Vista running, then set up a dual boot system, [BitLocker] will consider this sort of change to be an attack and refuse to run. But then you can use the recovery [password] to boot into Windows, then tell BitLocker to take the current configuration—with the dual boot code—as correct. After that, your dual boot system will work just fine, or so I've been told. You still won't be able to share any files on your C drive between operating systems, but you will be able to share files on any other drive.
>
> The problem is that it's impossible to distinguish between a legitimate dual boot system and an attacker trying to use another OS—whether Linux or another instance of Vista—to get at the volume.

Next-Generation Secure Computing Base (NGSCB)

Microsoft's *Next-Generation Secure Computing Base* (NGSCB, pronounced EN-*scub*) is an ambitious company initiative to secure PCs against myriad threats

by redesigning PCs with hardware and software security *beyond* TCG specifications [186,244,307]. Originally named "Palladium," its designers claimed its memory curtaining, sealed storage, and remote attestation could stop spam and viruses, protect user privacy, ensure the integrity and confidentiality of information, and enable content publishers to set robust usage policies their content [186,307]. Due to criticism from civil libertarians and software vendors, Microsoft renamed "Palladium" to NGSCB and radically reduced its planned redesign of the PC [74]. Presently, NGSCB, which is *not* included in Vista, "encapsulates" BitLocker and the company's TPM software interfaces in Windows and includes a security kernel that is "isolated" from the rest of the OS [74]. NGSCB's security kernel, termed the *Trusted Operating Root* (TOR) or *Nexus*, executes in CPUs that support the "ring -1" mode discussed in Section 4.3 [107,244]. The Nexus executes "in parallel" with Windows and provides only a few, *provably secure* device drivers (for the keyboard, mouse, and hard disk) that NGSCB applications can access via a handful of application programming interfaces (APIs) [244]. Windows, which executes in ring 0, manages the "rich user experience" and far larger number of device drivers and APIs and only "transfers control" to the Nexus for those secure applications that require NGSCB services [107,244]. Since Microsoft is still developing NGSCB and has renamed its NGSCB team to the *System Integrity Team*, little further information is known at this point [302]. However, a rumor holds that NGSCB will be included in Microsoft's "Vista R2," codenamed "Windows Fiji," to be released in 2008—or perhaps some other future version of Windows [330].

Regardless of NGSCB's release date, it seems certain the technology *will* be released in the future. In 2001, Microsoft received a patent for a "Digital Rights

Management Operating System" (DRMOS) whose "features" encapsulated many of the original Palladium plans [16, 73, 241]. The patent, filed in *1998*, mentions booting, via transitive trust, a DRMOS that "protects rights-managed data, such as downloaded content, from access by untrusted programs while the data is loaded into memory or on a page file as the result of the execution of a trusted application that accesses the memory" [73]. The patent explicitly states that "in a very real sense, the legitimate user of a computer can be an adversary of the data or content provider," and mentions a number of techniques content providers can use to enforce their content's usage policies, even if the "legitimate user of a computer" tries to circumvent them [73]. Given Microsoft's robust DRM "features" in Vista as described in Section 5.1.2, the (future) implementation and release of a DRMOS appear certain.

5.2 Windows XP

Unlike its successor Windows Vista, Windows XP has *much* less support for TC technologies and somewhat fewer usage-policy restrictions. The XP EULA *does* mandate that the user activate the software within 30 days of installation, but XP does not enter a Reduced Functionality Mode if the OS is not activated [206]. Microsoft requires that each XP user install its Windows Genuine Advantage (WGA) software to receive OS updates and add-ons; WGA checks for counterfeit XP copies and prevents users running them from receiving non-security-related OS updates [115].[79] The XP EULA also stipulates that users *not* work around technical

[79]Microsoft also requires users of its Office software to activate it and validate its "genuine" status [30, 221]. Moreover, Microsoft recommends that XP users install its WGA Notifications software, which periodically notifies the company of XP's "genuine" status [76]. However, WGA Notifications remains an "optional" update, unlike Vista's Software Protection Platform [76].

limitations in the OS, which Microsoft uses for activation purposes [206]. Yet XP's DRM support is not nearly as robust as Vista's, and the former OS has *no* TC support. While XP includes support for DRM-protected Windows Media Audio and Video files and its Secure Audio Path technology encrypts DRM-protected audio *en route* to the sound card, XP *does not* impose any of the onerous requirements on graphics cards or display monitors discussed in Section 5.1.2.[80] Furthermore, XP does not include any TC support or related applications.

5.3 Mac OS X

Contrary to popular belief, Mac OS X (hereafter abbreviated as *OS X*) *does not* support TC technologies. When Apple switched its Macintosh computers from IBM's PowerPC CPUs to Intel's Core CPUs, rumors surfaced that Intel Macintosh computers used TPMs to "lock" Mac OS X to Apple hardware [22, 156]. Indeed, Phil Schiller, Apple's Senior Vice President for Worldwide Marketing, stated that the company "will not allow running Mac OS X on anything other than an Apple Mac" [114]. Schiller's statement concurs with the Mac OS X license agreement, which only permits installation and execution of one copy of the OS on one Apple-branded computer at a time [17]. These rumors gained credence in August 2006, when, in a Black Hat conference presentation, security researcher Bruce Potter claimed that Rosetta (Apple's PowerPC-to-Intel intruction-conversion program) used the TPM to ensure that OS X only executed on Apple hardware [180]. Potter added that Apple's "coolness" drove consumer's acceptance of the company's DRM and TC technologies [180, 247]. However, in November 2006, Amit Singh,

[80] Secure Audio Path is discussed in detail on pages 33–34.

author of the book *Mac OS X Internals*, discovered that Apple does *not* use the TPM in any way, and hence OS X does not support TC [291]. While some Macs *do* contain TPMs, Apple's firmware *does not* support the "take TPM ownership" command as required by the TCG [291]! Those OS X users who wish to activate their TPMs may do so with Singh's open-source device drivers [291]. He notes that Apple encrypts some OS executables, such as the Dock, Finder, and user-interface services, with AES to deter pirates from running OS X on non-Apple-branded computers [292]. As Schneier points out, Apple likely encrypted these binaries to thwart attempts at reverse-engineering its programs [326].

5.4 GNU/Linux

Linux supports TC to the extent that its kernel loads TPM device drivers, which savvy users may use if they wish. Open-source APIs such as IBM's *TrouSerS* implement the TCG's "software stack" that enables the Linux OS to communicate with the TPM and vice versa [291, 348]. Again, users may choose whether or not they use these APIs.

Having thoroughly discussed the state of TC support in operating systems, we examine various software packages that use TC functionality in the next chapter.

CHAPTER 6

TC SOFTWARE

At present, very few applications use TC technology. We discuss Windows, Mac OS X, and Linux applications that *do* use TC technology in that order.

6.1 Windows

Windows Vista's BitLocker application is the best-known TC software package.[81] Original equipment manufacturers such as HP and Dell ship TC software with computers that include TPMs [311]. Wave Systems and Infineon also sell TC software. "Endpoint security" software supports TC; the software uniquely identifies corporate and mobile employees' computers via the TPM's Endorsement Key to ensure that sensitive corporate data is not transferred to any external device without authorization [246]. Phoenix Technologies sells one such product, TrustedCore, and other software vendors sell similar products in this fast-growing market. With these exceptions, few other Windows applications support TC.

[81] We discuss BitLocker in Chapter 5 because it is included in high-end versions of Windows Vista and is not a "separate" software package.

6.2 Mac OS X

As noted in Chapter 5, Mac OS X does not use the TPM, and hence no TC software is included with the OS. Amit Singh distributes open-source software that allows Mac users to activate and use their TPMs [291].

6.3 GNU/Linux

Aside from TPM emulators and the previously-discussed Enforcer package, there are few TC Linux software packages. However, the open nature of the platform does not preclude development of such packages.

In the next chapter, we examine applications of TC technologies in other markets, such as cellular phones and video game consoles.

CHAPTER 7

OTHER TC APPLICATIONS

The TCG is developing specifications for trusted servers, network connections, printers, and cellular (mobile) phones [320, 324]. In addition, some video-game consoles use TPMs. We briefly examine TC applications of trusted network connections, mobile phones, and video-game consoles.

7.1 Network Connections

The TCG has released its *Trusted Network Connect* (TNC) specification, which describes integrity-measurement techniques for "endpoint" network-connected computers that allow the computers to ensure each other's integrity and identity [320]. With TNC, each computer "measures" its "health" (or compliance with local IT policy) and checks that only authorized users are logged in before allowing the network connection [320]. TNC does *not* require the use of a TPM, and, according to the TCG, TNC can prevent the propagation of malware on a network [320, 322].

7.2 Mobile Phones

The TCG has also released specifications for mobile phones that require "trusted" phones to have a *Mobile Trusted Module* (MTM)—a phone-optimized TPM—and software that provides roots of trust like those described in Section 4.1 [324]. MTM-enabled phones could robustly enforce policies for DRM-protected content and services, prompting criticism that wireless carriers could control how people use their phones [120, 324]. The TCG maintains that it does *not* produce DRM specifications, though nothing precludes mobile-phone manufacturers from using them to develop robust DRM systems [320].

7.3 Video-Game Consoles

Unique among video-game consoles, Microsoft's Xbox 360 has a TPM [203]. The company uses the TPM to thwart attempts to "hack" the console to "turn it into a media [center], upgrade the hard drive or allow it to play imported games" [106, 149]. Moreover, the Xbox maintains hardware and software security with a *hypervisor*, a small program that executes in privileged mode and intercepts each system call made by programs executing in unprivileged mode [285]. The hypervisor requires that "all executable code. . .be read-only and encrypted" for additional security [285]. Due to a hypervisor bug, hackers were able to successfully execute Linux on the console from October 31, 2006 until February 28, 2007, when Microsoft fixed the bug [285].

Having completed our examination of TC hardware, operating systems, software, and novel applications, we now focus on the effects of TC's (ab)use on civil liberties and the economy. Sensible policy suggestions are listed in Chapter 8.

CHAPTER 8

EFFECTS ON CIVIL LIBERTIES AND THE ECONOMY

TC technology has several positive uses. Companies can use it to protect confidential data from theft, even on employees' laptops, and users can do the same thing with their private data. Both companies and users can drastically reduce the likelihood of malware execution with secure I/O and memory curtaining, as described in Section 3.3.5. In theory, video-game manufacturers and online auctioneers can use TC technology to thwart cheating in multiplayer games and online, respectively.

But TC technology can be readily abused as well. Software vendors can abuse it to "lock" users in to their products. For instance, as Cory Doctorow explains in an *Information Week* editorial, Microsoft could encrypt all Office files with a TPM key by default in a future version of the software [65]. Customers who read and edit TC-protected Office files cannot afford to switch to a competing office suite, such as the free OpenOffice.org, as the files are encrypted [65]. Competitors cannot reverse-engineer Microsoft's file formats without risking a lawsuit under the DMCA, and the company can charge expensive fees to license its technology, thereby imposing barriers to entry to new competitors in the office-suite market [65]. Other software vendors could easily follow suit. In another *Information Week*

editorial, Doctorow describes how Apple used its DMCA-enforced ban on reverse-engineering iTunes songs to update the software and restrict how customers could use their legally-purchased songs [66]. In a similar vein, Schneier argues in a *Forbes* editorial that Microsoft's draconian Vista DRM is intended to "lock" Hollywood studios "into selling content in [the company's] proprietary formats" [276]. Ross Anderson, a professor of security engineering at Cambridge University, illustrates how hardware vendors can facilitate this "lock-in" with TC in [8]. It readily follows that these abuses, combined with pervasive DRM and restrictive software licenses, can deprive computer users of their civil liberties and fair-use rights. Possible abuses and their effects on civil liberties are detailed further in [8, 15, 119, 273] and in our synopsis [51].

We now turn our attention to policy suggestions and consumer behaviors that can help mitigate the above abuses.

CHAPTER 9

POLICY AND CONSUMER SUGGESTIONS

This chapter offers suggested policy changes to prevent the TC abuses described in the previous chapter and to rectify the current imbalance between copyright owners and consumers. These changes focus on copyright and patent law reform. The chapter also offers suggestions to help consumers navigate the present-day computer market, which is saturated with TC technology.

Changes must be made to U.S. copyright law to restore a healthy "balance" between copyright owners and consumers, which is currently skewed heavily toward copyright owners. First, the DMCA must be amended to make provisions that allow consumers to legally bypass copyright-protection technologies while exercising fair-use rights. Presently, people who strip legally-purchased media files of their DRM protection break the law, even if they do so for legal reasons such as storage on a portable media device or to give a copy to their friends [66]. In 2003, representatives Rick Boucher (D-VA) and John Doolittle (D-CA) introduced the *Digital Media Consumers' Rights Act* (DMCRA), which would require copy-protected media to be labeled as such and would grant consumers the right to bypass copyright-protection systems *for legal purposes* [96]. While Congress gave the DMCRA a hearing in 2005, many representatives opposed it due to large-scale

piracy fears, and it was *not* signed into law [26]. Boucher has since introduced the *Freedom And Innovation Revitalizing U.S. Entrepreneurship Act* (FAIR USE Act), which only imposes limited exemptions on circumvention that are nowhere as far-reaching as those proposed in the DMCRA [176].[82] Clearly, a *DMCRA*-like law must be passed to legally restore consumers' fair-use rights *vis-à-vis* the DMCA. In addition, copyright terms are, at present, *far* too long; they persist throughout the creator's lifetime plus 95 years thereafter [183]. As suggested by an editorial in *The Economist*, they should be "rolled back" to a *short* period of time—14 years, renewable once—to ensure copyright's original purpose is maintained: the creator of a work controls its distribution for that time, during which he profits from it, and then the work enters the public domain [303]. By "opening up" very old works whose copyright protection has been repeatedly extended, the public domain will be greatly enriched, thereby bolstering possibilities for innovation [303].[83] Besides, copyright owners will still enjoy the protection for their works granted under current law. Moreover, U.S. patent law must be reformed to prevent the filing of spurious software patents that impose barriers to new entrants in the software market and can result in costly litigation [52]. As U.S. patent law holds that *only* new processes, machines, manufactured goods, or "[compositions] of matter" can be patented, and software is ultimately a set of "pure" abstractions, Congress should, at the very least, revise the law to prevent people from patenting trivial software

[82]See [176] for a full discussion of the Act's nominal—but insubstantial—DMCA "reform."

[83]Of course, even when a work is released into the public domain, a person who uses that work to create his *own* copyrighted work has an ethical responsibility to cite the previous work, even if he need not pay royalties for its use.

"inventions" for the sole purpose of suing "infringers" [52].[84] We discuss software patents further in [52].

Consumers have a variety of options in the TC-saturated computer market. The most obvious option is to do nothing and accept TC and DRM technologies in their present form, an option that is clearly unacceptable, as evidenced by Vista's draconian content protection, Apple's monopoly in digital music, and pervasive DRM support in hardware and software. As Schneier explained in an interview with *TG Daily*, the status quo is that of a market failure (emphasis added) [119]:

> You have the right [to *not* accept AACS-like measures inside your computer]. The question is, do you have the ability? In a capitalist democratic society, you only have the options that are presented to you. The market failure is the monopoly/oligopoly market failure, where the option to turn [them] off, or the option not to have [them], isn't presented to you. Cell phones [are] a good example. . .I don't have the option to go with—even if it's more expensive—a [better-quality] service provider. There isn't one. They all [stink]. They all [stink] because they realize that competing on service doesn't make sense, and they're better off hiring Catherine Zeta-Jones to be in ads. That's the unfortunate truth.
>
> [There is] a monopoly in operating systems. . . . If Microsoft and. . . [Apple] go along with saying, "Only these sorts of things will happen," by cutting out Linux—because who cares?—that's your only option. But the fear is, I have no choice but to buy a DRM-enabled computer, because there isn't anything else available, because it's all the market will give me. That's the fear.
>
> *This is a market failure.* I'm always amazed at people who are big fans of the market, who don't understand what it looks like when a market fails, and what systems don't work in a market. *This is an example of it.* If there were hundreds of operating systems, and you could pick the best one. . .there'd be one that didn't do DRM,. . .we'd all use it, and DRM would die. But all you need is Sony and the big media companies [working with] the two big operating system companies, and. . .[the] choice is no longer there.

[84] As the U.S. Patent and Trademark Office is not nearly as well-funded as, say, the military or "pork-barrel" projects, this scenario is not as farfetched as one might imagine.

This market failure is starkly clear for Windows users buying new PCs preloaded with Vista, as the content-protection measures discussed in Section 5.1.2 effectively transform their computers into Hollywood-approved Internet entertainment devices [81]. Of course, customers could simply boycott computers with TPMs, but this is impractical, as tens of millions of such computers have shipped in the last few years [167]. Besides, technically-savvy, privacy-conscious users can simply disable TPMs in their BIOSes. If there is sufficient consumer demand, companies like GeekSquad can disable TPMs for non-technical users. Seth Schoen of the Electronic Frontier Foundation (EFF) proposed Owner Override, an elegant solution to prevent vendor lock-in via remote attestation [278]. Owner Override allows the owner of a computer system to provide a *false* attestation of her computer configuration, *e.g.*, that she is running Mozilla Firefox on Windows XP instead of Internet Explorer on Windows Vista. For example, if an online auction site required users to attest that they were accessing it with the latter browser and OS configuration, Owner Override allows her to still use the site with *her* browser and OS configuration. Selective disability of TPMs, an Owner Override feature, and conscientious purchasing is, very likely, the best option for consumers. In particular, they should carefully weigh their financial support of RIAA-label recording artists and MPAA-studio movie casts against the anti-consumer behavior of the recording and movie industries, respectively. In particular, the RIAA and MPAA support the DMCA and file infamously spurious lawsuits against their customers, not *just* against copyright infringers. Certainly, purchasing entertainment pales in comparison to providing the necessities of life, caring for one's family, and maintaining one's health!

Finally, consumers *should not* upgrade to Windows Vista at this time due to its draconian content protection and licensing. They should evaluate the Apple Macintosh and Linux platforms as alternatives for production use and run Windows XP if they must use Microsoft technology. While the Mac is a good production machine, due in part to the ability to run Windows via Parallels Desktop, Mac computers cost more than their Windows counterparts and consumers are "locked in" to Apple machines if they wish to run Mac OS X legally. Linux is free, versatile and installs on many computer architectures, but its primary desktop environments are not as polished as OS X. Those working with Windows XP and Vista should *strongly* consider using Dino Nuhagic's nLite and vLite software to remove components, install "service packs" and patches, apply "tweaks," and burn custom install CDs for the respective OS [239, 240]. Ultimately, these "unattended installation" tools can prevent customers from completing the time-consuming process of downloading and installing service packs and patches, since they are "slipstreamed" onto the respective install CD [239, 240]. We have successfully removed unwanted Windows components such as Windows Media Player, Outlook Express, and Windows Messenger using nLite. Of course, Linux users need not use these tools to remove system software, as this "feature" is available with all Linux versions.

CHAPTER 10

CONCLUSIONS

As previously noted, TC technologies offer the promise of increased computer security in a world where security attacks are the norm, *not* the exception. Yet, as Chapter 8 illustrates, TC technologies can be abused in many ways that are detrimental to consumers. Presently, the "powers that be" in the computer and content industries have already eviscerated consumers' fair-use rights by requiring legal online music and video stores to sell content in DRM-encumbered media files.[85] Also, computer companies have *de facto* monopolies in some markets, such as Microsoft's monopoly on operating systems and Apple's monopoly on digital music, which are not necessarily in consumers' best interests. And Microsoft and Hollywood studios have collaborated on Vista's draconian content "protection" measures that may very well threaten the health of the computer industry [81,137]. The only sensible consumer response is to "vote with one's pocketbook," *i.e.* refuse to purchase DRM-encumbered content or operating systems.[86] Those who *must*

[85]Of course, there are exceptions, such as the iTunes Store's sale of EMI songs without DRM [91]. Other record companies may soon follow suit, according to one executive [175]. But iTunes consumers must pay an extra 30¢ for this DRM "freedom" that *should* have been the status quo since iTunes' inception!

[86]If consumers wish to legally buy music, we suggest purchasing CDs, as the tracks thereon can be "ripped" into formats that *any* portable media player can play. For reasons discussed in Section 2.3, the same argument does *not* hold for current- and next-generation DVDs.

use Microsoft technologies are advised *not* to run Windows XP. Finally, political activism and the courts can help restore a sensible balance between copyright owners' protection of their works and consumers' fair-use rights. Class-action and antitrust lawsuits may serve to limit monopoly power and redress any wrongs arising from TC abuses. Ultimately, consumers should educate themselves about TC technology and its potential (ab)uses before blindly adopting it *en masse*. We must strike a reasonable balance between increased computer security and protection from the abuses described above.

BIBLIOGRAPHY

[1] American Megatrends Corporation, "AMIBIOS® enables secure and trusted computing with a TCPA-compliant module," Press Release, American Megatrends Corporation, 6 Jan. 2003, accessed 1 February 2005. [Online]. Available: http://www.ami.com/news/presshow.cfm?PrID=118

[2] F. Amoroso, "Macrovision's Response to Steve Jobs' Open Letter," Macrovision Corporation, 17 Feb. 2007, accessed 18 February 2007. [Online]. Available: http://www.macrovision.com/company/news/drm/response_letter.shtml

[3] J. P. Anderson, "Computer Security Technology Planning Study," Air Force Electronic Systems Division (AFSC), Hanscom AFB, Bedford, MA, Tech. Rep. ESD-TR-73-51, Oct. 1972, accessed 1 March 2007. [Online]. Available: http://seclab.cs.ucdavis.edu/projects/history/papers/ande72.pdf

[4] N. Anderson, "Hacking Digital Rights Management," Ars Technica, 18 July 2006, accessed 20 March 2007. [Online]. Available: http://arstechnica.com/articles/culture/drmhacks.ars, http://arstechnica.com/articles/culture/drmhacks.ars/2, http://arstechnica.com/articles/culture/drmhacks.ars/3, http://arstechnica.com/articles/culture/drmhacks.ars/4

[5] ——, "Apple would "switch to selling only DRM-free music" if labels agree," Ars Technica, 6 Feb. 2007, accessed 7 February 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070206-8782.html

[6] ——, "Coral to Apple: We've got your interoperable DRM right here!" Ars Technica, 13 Feb. 2007, accessed 13 February 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070213-8829.html

[7] ——, "Why DRM's best friend might just be Apple, Inc." Ars Technica, 11 Jan. 2007, accessed 18 January 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070111-8595.html

[8] R. Anderson, "Cryptography and Competition Policy — Issues with 'Trusted Computing'," Cambridge University, accessed 13 January 2007. [Online]. Available: http://www.cl.cam.ac.uk/~rja14/Papers/tcpa.pdf

[9] ——, "Security Engineering - Part 1," Cambridge University, accessed 22 September 2006. [Online]. Available: http://www.cl.cam.ac.uk/~rja14/book/booksec1.html

[10] ——, "Security Engineering - Part 2," Cambridge University, accessed 22 September 2006. [Online]. Available: http://www.cl.cam.ac.uk/~rja14/book/booksec2.html

[11] ——, "Security Engineering - Part 3," Cambridge University, accessed 22 September 2006. [Online]. Available: http://www.cl.cam.ac.uk/~rja14/book/booksec3.html

[12] ——, "The Draft IPR Enforcement Directive — A Threat to Competition and to Liberty," Foundation for Information Policy Research, accessed 29 September 2006. [Online]. Available: http://www.fipr.org/copyright/draft-ipr-enforce.html

[13] ——, *Security Engineering*. Wiley, Jan. 2001, accessed 22 September 2006. [Online]. Available: http://www.cl.cam.ac.uk/~rja14/book.html

[14] ——, "Security in Open versus Closed Systems — The Dance of Boltzmann, Coase, and Moore," in *Open Source Software: Economics, Law, and Policy*, Toulouse, France, 20–21 June 2002, accessed 13 January 2007. [Online]. Available: http://www.ftp.cl.cam.ac.uk/ftp/users/rja14/toulouse.pdf

[15] ——, "'Trusted Computing' Frequently Asked Questions," Cambridge University, Aug. 2003, accessed 17 January 2007. [Online]. Available: http://www.cl.cam.ac.uk/~rja14/tcpa-faq.html

[16] anti-dmca.org, "Microsoft DRMOS Palladium – The Trojan Horse OS," 7 Apr. 2004, accessed 8 January 2007. [Online]. Available: http://anti-dmca.org/DRM-OS.html

[17] Apple Computer, Inc., "Software License Agreement for Mac OS X," Apple Computer, Inc., accessed 5 November 2006. [Online]. Available: http://images.apple.com/legal/sla/macosx1044.pdf

[18] ——, "MPEG-4," Apple Computer, Inc., Oct. 2003, accessed 23 January 2007. [Online]. Available: http://images.apple.com/quicktime/pdf/MPEG4_v3.pdf

[19] Apple, Inc., "About iTunes Store authorization and deauthorization," Apple, Inc., 12 Sept. 2006, accessed 22 January 2007. [Online]. Available: http://docs.info.apple.com/article.html?artnum=93014

[20] ——, "iPod 101: Browse and Buy Music and Books," Apple, Inc., 27 Dec. 2006, accessed 22 January 2007. [Online]. Available: http://docs.info.apple.com/article.html?artnum=304674

[21] Apple/Intel FAQ, "Apple/Intel FAQ," 2007, accessed 10 April 2007. [Online]. Available: http://www.appleintelfaq.com

[22] P. Attivissimo, "Trusted Computing Chips Found in Intel Macs," 18 May 2006, accessed 19 October 2006. [Online]. Available: http://attivissimo.blogspot.com/2006/04/trusted-computing-chips-found-in-intel.html

[23] L. Badger, D. F. Sterne, *et al.*, "Practical Domain and Type Enforcement for UNIX," in *Proceedings of the 1995 IEEE Symposium on Security and Privacy*, Oakland, CA, 8–10 May 1995, pp. 66–77, accessed 1 March 2007. [Online]. Available: http://ieeexplore.ieee.org.proxy.lib.ohio-state.edu/iel2/3181/9013/00398923.pdf?tp=&arnumber=398923&isnumber=9013

[24] S. Bajikar, "Trusted Platform Module (TPM) based Security on Notebook PCs – White Paper," White Paper, Mobile Platforms Group – Intel Corporation, 20 June 2002, accessed 3 October 2006. [Online]. Available: http://www.intel.com/design/mobileplatform/downloads/Trusted_Platform_Module_White_Paper.pdf

[25] E. Bangeman, "DeCSS author cleared again on appeal," Ars Technica, 22 Dec. 2003, accessed 20 March 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20031222-3229.html

[26] ——, "Congress gives Fair Use legislation a hearing," Ars Technica, 16 Nov. 2005, accessed 28 April 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20051116-5589.html

[27] ——, "HD DVD and Blu-ray content to be degraded for analog displays," Ars Technica, 22 Jan. 2006, accessed 1 February 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20060122-6027.html

[28] ——, "Testing DRM-free waters: EMI selling a few MP3s through Yahoo Music," Ars Technica, 6 Dec. 2006, accessed 18 January 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20061206-8368.html

[29] ——, "Yahoo Music: Santa Claus will have DRM-free music in his sleigh," Ars Technica, 13 Feb. 2007, accessed 13 February 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070213-8830.html

[30] C. Barker, "Microsoft gets tough on Office fakers," C|NET News.Com, 30 Oct. 2006, accessed 17 April 2007. [Online]. Available: http://news.com.com/2102-1012_3-6130223.html?tag=st.util.print

[31] M. F. Barrett, "Toward an Open Trusted Computing Framework," Master's thesis, Department of Computer Science, University of Auckland, New Zealand, Feb. 2005, accessed 5 January 2007. [Online]. Available: http://www.cs.auckland.ac.nz/~cthombor/Students/mbarrett/mbarrettThesis.pdf

[32] "Music execs criticise DRM systems," BBC, 15 Feb. 2007, accessed 18 February 2007. [Online]. Available: http://newsvote.bbc.co.uk/mpapps/pagetools/print/news.bbc.co.uk/1/hi/technology/6362069.stm

[33] D. E. Bell, "Looking Back at the Bell-La Padula Model," in *Proceedings of the 21st Annual Computer Security Applications Conference (ACSAC)*, Tucson, AZ, 7–9 Dec. 2005, accessed 28 April 2007. [Online]. Available: http://www.selfless-security.org/presentations/looking-back.pdf

[34] ——, "Looking Back: Addendum," presented at the Proceedings of the 22nd Annual Computer Security Applications Conference (ACSAC), Miami Beach, FL, 11–15 Dec. 2006, accessed 28 April 2007. [Online]. Available: http://www.selfless-security.org/presentations/Bell_LBA.pdf

[35] D. Berlind, "A load of C.R.A.P." At The Whiteboard podcast, accessed 5 January 2007. [Online]. Available: http://news.zdnet.com/html/z/wb/6035707.html

[36] P. Biddle, "Re: dangers of TCPA/Palladium," 5 Aug. 2002, accessed 7 February 2005. [Online]. Available: http://www.cl.cam.ac.uk/%7Erja14/biddle.txt

[37] P. Biddle, P. England, *et al.*, "The Darknet and the Future of Content Distribution," in *Proceedings of 2002 ACM Workshop on Digital Rights Management*, ser. Lecture Notes on Computer Science, vol. 2696. Washington, DC: Springer-Verlag, 18 Nov. 2002, pp. 155–176, accessed 12 January 2007. [Online]. Available: http://crypto.stanford.edu/DRM2002/darknet5.doc

[38] J. Borland, "Microsoft critic dismissed by @Stake," C|NET News.Com, 26 Sept. 2003, accessed 13 January 2007. [Online]. Available: http://news.com.com/2102-1009_3-5082649.html?tag=st.util.print

[39] ——, "Microsoft unveils new antipiracy tools," C|NET News.Com, 3 May 2004, accessed 12 November 2004. [Online]. Available: http://news.com.com/2102-1025_3-5203004.html?tag=st.util.print

[40] ——, "Microsoft's iPod Killer?" C|NET News.Com, 2 Apr. 2004, accessed 12 November 2004. [Online]. Available: http://news.com.com/2102-1027_3-5183692.html?tag=st.util.print

[41] ——, "'Plays for sure' means Microsoft's inside," C|NET News.Com, 25 Aug. 2004, accessed 12 November 2004. [Online]. Available: http://news.com.com/2102-1025_3-5324402.html?tag=st.util.print

[42] L. M. Bowman, "Sklyarov reflects on DMCA travails," C|NET News.Com, 20 Dec. 2002, accessed 13 January 2007. [Online]. Available: http://news.com.com/2102-1023_3-978497.html?tag=st.util.print

[43] M. Brand, A. Champion, and D. Chan, "Combating the Botnet Scourge," Mar. 2007, unpublished. [Online]. Available: http://www.cse.ohio-state.edu/~champion/research/Combating_the_Botnet_Scourge.pdf

[44] E. Brickell, J. Camenisch, and L. Chen, "Direct Anonymous Attestation," in *Proceedings of 11th ACM Conference on Computer and Communications Security*. Washington, DC: ACM Press, 25 – 29 Oct. 2004, accessed 27 March 2007. [Online]. Available: http://eprint.iacr.org/2004/205.pdf

[45] I. Brown, "Implementing the EU Copyright Directive," Foundation for Information Policy Research, accessed 17 September 2007. [Online]. Available: http://www.fipr.org/copyright/guide/eucd-guide.pdf

[46] J. Brown, "Just like being there: Papers from the Fall Processor Forum 2005: Application-customized CPU design," Fall Processor Forum 2005, IBM Corporation, 6 Dec. 2005, accessed 22 October 2006. [Online]. Available: http://www-128.ibm.com/developerworks/power/library/pa_fpfxbox/?ca=dgr-lnxw09XBoxDesign

[47] Business Software Alliance and International Data Corporation, "BSA and IDC Global Software Piracy Study," Business Software Alliance, May 2006, accessed 6 January 2006. [Online]. Available: http://www.bsa.org/globalstudy/upload/2005%20Piracy%20Study%20-%20Official%2-Version.pdf

[48] A. Carroll, M. Juarez, *et al.*, "Microsoft "Palladium": A Business Overview," Microsoft Corporation, 25 Jan. 2003, accessed 7 February 2005. [Online]. Available: http://www.microsoft.com/presspass/features/2002/jul02/0724palladiumwp.asp

[49] P. Cashmere, "Universal Is The Biggest Music Company of 2005," Undercover Media, 5 Jan. 2006, accessed 25 January 2007. [Online]. Available: http://www.undercover.com.au/news/2006/jan06/20060105_universal.html

[50] CEN/ISSS (European Standards Committee/Information Society Standardization System), "Digital Rights Management Final Report," CEN/ISSS, 30 Sept. 2003, accessed 18 January 2007. [Online]. Available: http://europa.eu.int/comm/enterprise/ict/policy/doc/drm.pdf

[51] A. C. Champion, "'Trusted Computing'—A Fool's Bargain," June 2004, unpublished. [Online]. Available: http://www.cse.ohio-state.edu/~champion/research/TC_Fool_Bargain.pdf

[52] ——, "The Scourge of Software Patents," Apr. 2006, unpublished. [Online]. Available: http://www.cse.ohio-state.edu/~champion/articles/cse_601_paper.pdf

[53] J. Cheng, "Standardized DisplayPort 1.1 brings HDCP support," Ars Technica, 4 Apr. 2007, accessed 9 April 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070404-standardized-displayport-1-1-brings-hdcp-support.html

[54] "cheong", "Is this true?" Channel 9 Forums, 28 Dec. 2006, accessed 1 January 2007. [Online]. Available: http://channel9.msdn.com/ShowPost.aspx?POSTID=268997

[55] N. Cochrane, "Bit by bit, digital freedom disappears," The Age, 17 Sept. 2002, accessed 5 January 2007. [Online]. Available: http://www.theage.com.au/cgi-bin/common/popupPrintArticle.pl?path=/articles/2002/09/14/1031608343597.html

[56] S. Crosby, I. Goldberg, *et al.*, "A Cryptanalysis of the High-bandwidth Digital Content Protection System," in *ACM-CSS8 DRM Workshop*, 5 Nov. 2001, accessed 4 November 2006. [Online]. Available: http://apache.dataloss.nl/~fred/www.nunce.org/hdcp/hdcp111901.htm

[57] ——, "A Cryptanalysis of the High-bandwidth Digital Content Protection System," 5 Nov. 2001, accessed 8 April 2007. [Online]. Available: http://cryptome.org/hdcp111901.htm

[58] G. Danezis and R. Anderson, "The Economics of Censorship Resistance," in *Third Annual Workshop on the Economics of Information Security (WEIS 2004)*. Minneapolis, MN: University of Minnesota, 13–14 May 2004, accessed 13 January 2007. [Online]. Available: http://www.cl.cam.ac.uk/users/gd216/redblue.pdf

[59] C. Demerjian, "Intel to cut Linux out of the content market," The Register, 15 July 2005, accessed 19 October 2006. [Online]. Available: http://theinquirer.net/print.aspx?article=24638&print=1

[60] "Phoenix unveils next-gen BIOS firmware roadmap," DeviceForge, 24 Nov. 2003, accessed 5 April 2007. [Online]. Available: http://deviceforge.com/news/NS6888038345.html

[61] W. Diffie and M. E. Hellman, "New Directions in Cryptography," *IEEE Transactions on Information Theory*, vol. IT-22, no. 6, pp. 644–654, Nov. 1976, accessed 25 November 2006. [Online]. Available: http://citeseer.ist.psu.edu/diffie76new.html

[62] "Digital Content Protection LLC: Home," Digital Content Protection LLC, 2007, accessed 8 April 2007. [Online]. Available: http://www.digital-cp.com/home

[63] C. Doctorow, "Microsoft Research DRM talk," Transcript, Microsoft Research Group, Redmond, WA, 17 June 2004, accessed 12 January 2007. [Online]. Available: http://www.patandkat.com/pat/weblog/mirror/cory-drm/doctorow-drm-ms.html

[64] ——, "DRM Talk for Hewlett-Packard Research," Transcript, Hewlett-Packard Research, 28 Sept. 2005, accessed 12 January 2007. [Online]. Available: http://www.xs4all.nl/~collin/test/hpdrm.html

[65] ——, "How Vista lets Microsoft lock users in," EE Times, 5 Dec. 2006, accessed 13 January 2007. [Online]. Available: http://www.eetimes.com/news/latest/showArticle.jhtml;jsessionid=YWZUCY3YGLQ00QSNDLOSKH0CJUNN2JVN?articleID=196602058&printable=true

[66] ——, "Opinion: Apple's Copy Protection Isn't Just Bad For Consumers, It's Bad For Business," InformationWeek, 31 July 2006, accessed 12 January 2007. [Online]. Available: http://www.informationweek.com/shared/printableArticleSrc.jhtml?articleID=191000408

[67] DVD Copy Control Association, "Content Scramble System," DVD Copy Control Association, accessed 1 February 2007. [Online]. Available: http://www.dvdcca.org/css/

[68] D. Eastlake and P. Jones, "US Secure Hash Algorithm 1 (SHA1)," RFC 3174, Internet Engineering Task Force, Sept. 2001, accessed 23 October 2006. [Online]. Available: http://tools.ietf.org/html/rfc3174

[69] ElcomSoft, "Advanced eBook Processing Software," accessed 4 February 2007. [Online]. Available: http://www.elcomsoft.com/aebpr.html

[70] Electronic Frontier Foundation, "FTAA & Bilateral FTA Resources," Electronic Frontier Foundation, accessed 20 October 2006. [Online]. Available: http://www.eff.org/IP/FTAA

[71] R. Enderle, "Trusted Computing: "Maligned by Misrepresentations and Creative Fabrications"," Storage Pipeline, 5 Feb. 2004, accessed 7 February 2005. [Online]. Available: http://subscriber.acumeninfo.com/uploads2/0/A/0AFC7F759F71BF2C4E80FAE3CFD19DCF/1079105225453/SOURCE/4643tsg.html

[72] P. England and M. Peinado, "Authenticated Operation of Open Computing Devices," in *Proceedings of the 7th Australasian Conference on Information Security and Privacy*, ser. Lecture Notes on Computer Science, vol. 2384. Melbourne, Australia: Springer-Verlag, July 3–5 2002, pp. 346–361, accessed 9 January 2007. [Online]. Available: http://cs-people.bu.edu/mpe/acisp.pdf

[73] P. England, J. D. Treville, and B. W. Lampson, "Digital rights management operating system," United States Patent 6 330 670, December 11, 2001, accessed 9 May 2004. [Online]. Available: http://cryptome.org/ms-drm-os.htm

[74] J. Evers, "Microsoft's leaner approach to Vista security," C|NET News.Com, 29 Aug. 2005, accessed 21 March 2007. [Online]. Available: http://m.news.com/Microsofts+leaner+approach+to+Vista+security/2163-7355_3-5843808.html

[75] ——, "Microsoft piracy check comes calling," C|NET News.Com, 25 Apr. 2006, accessed 17 April 2007. [Online]. Available: http://news.com.com/2102-1016_3-6064555.html?tag=st.util.print

[76] ——, "Microsoft set to push out updated antipiracy tool," C|NET News.Com, 29 Nov. 2006, accessed 17 April 2007. [Online]. Available: http://news.com.com/2102-7355_3-6139161.html?tag=st.util.print

[77] ——, "Microsoft to lock pirates out of Vista PCs," C|NET News.com, 4 Oct. 2006, accessed 20 October 2006. [Online]. Available: http://news.com.com/2102-7355_3-6122462.html?tag=st.util.print

[78] Fasoo.com, "Introduction," Fasoo.com, accessed 18 January 2007. [Online]. Available: http://www.fasoo.com/eng/sub_com01.asp

[79] ——, "Total Enterprise DRM Solution," Fasoo.com, accessed 18 January 2007. [Online]. Available: http://www.drmone.com/product01.asp

[80] Federal Deposit Insurance Corporation, Division of Supervision and Consumer Protection, Technology Supervision Branch, "Putting an End to Account-Hijacking Identity Theft Study Supplement," Study Supplement, Federal Deposit Insurance Corporation, 17 June 2005, accessed 4 October 2006. [Online]. Available: http://www.fdic.gov/consumers/consumer/idtheftstudysupp/idtheftsupp.pdf

[81] E. Felten, "Hollywood Controlling Parts of Windows Vista Design," 9 Aug. 2005, accessed 13 January 2007. [Online]. Available: http://www.freedom-to-tinker.com/?p=882

[82] ——, "HDCP Could Have Been Better," 17 Apr. 2006, accessed 4 November 2006. [Online]. Available: http://www.freedom-to-tinker.com/?p=1006!

[83] ——, "HDCP: Why So Weak?" 19 Apr. 2006, accessed 4 November 2006. [Online]. Available: http://www.freedom-to-tinker.com/?p=1007

[84] ——, "HDMI and Output Control," 13 Apr. 2006, accessed 4 November 2006. [Online]. Available: http://www.freedom-to-tinker.com/?p=1004

[85] ——, "Making and Breaking HDCP Handshakes," 14 Apr. 2006, accessed 4 November 2006. [Online]. Available: http://www.freedom-to-tinker.com/?p=1005!

[86] ——, "AACS Decryption Code Released," 8 Jan. 2007, accessed 1 February 2007. [Online]. Available: http://www.freedom-to-tinker.com/?p=1104

[87] ——, "AACS: Extracting and Using Keys," 10 Jan. 2007, accessed 1 February 2007. [Online]. Available: http://www.freedom-to-tinker.com/?p=1106

[88] ——, "AACS: Modeling the Battle," 18 Jan. 2007, accessed 1 February 2007. [Online]. Available: http://www.freedom-to-tinker.com/?p=1111

[89] ——, "AACS: Sequence Keys and Tracing," 17 Jan. 2007, accessed 1 February 2007. [Online]. Available: http://www.freedom-to-tinker.com/?p=1110

[90] ——, "AACS: Title Keys Start Leaking," 16 Jan. 2007, accessed 1 February 2007. [Online]. Available: http://www.freedom-to-tinker.com/?p=1109

[91] ——, "EMI To Sell DRM-Free Music," 3 Apr. 2007, accessed 7 April 2007. [Online]. Available: http://www.freedom-to-tinker.com/?p=1141

[92] ——, "Software HD-DVD/Blu-ray Players Updated," 13 Apr. 2007, accessed 19 April 2007. [Online]. Available: http://www.freedom-to-tinker.com/?p=1148

[93] N. Ferguson, "Censorship in action: why I don't publish my HDCP results," 15 Aug. 2001, accessed 20 October 2006. [Online]. Available: http://www.macfergus.com/niels/dmca/cia.htm

[94] ——, "Frequently asked questions about DMCA and my HDCP paper," 4 Apr. 2003, accessed 20 October 2006. [Online]. Available: http://www.macfergus.com/niels/dmca/faq.htm

[95] ——, "AES-CBC + Elephant diffuser: A Disk Encryption Algorithm for Windows Vista," White Paper, Microsoft Corporation, Aug. 2006, accessed 21 March 2007. [Online]. Available: http://download.microsoft.com/download/0/2/3/0238acaf-d3bf-4a6d-b3d6-0a0be4bbb36e/BitLockerCipher200608.pdf

[96] K. Fisher, "DMCA challenge to be considered this week," Ars Technica, 9 May 2004, accessed 28 April 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20040509-3754.html

[97] ——, "Blu-ray adopts mandatory managed copy, but says no to iHD," Ars Technica, 17 Nov. 2005, accessed 1 February 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20051117-5592.html

[98] ——, "BSA disgusted with critiques of their inflammatory piracy loss methodology," Ars Technica, 14 June 2005, accessed 31 December 2006. [Online]. Available: http://arstechnica.com/news.ars/post/20050614-4993.html

[99] ——, "On Windows Vista, DRM, and new monitors," Ars Technica, 21 Aug. 2005, accessed 8 January 2007. [Online]. Available: http://arstechnica.com/articles/paedia/hardware/hdcp-vista.ars

[100] ——, "Microsoft unveils Windows Vista editions," Ars Technica, 27 Feb. 2006, accessed 17 April 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20060227-6269.html

[101] ——, "The Problem with MPAA's shocking piracy numbers," Ars Technica, 5 May 2006, accessed 15 January 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20060505-6761.html

[102] ——, "AACS LA: Internet "revolt" be damned, this fight is not over," Ars Technica, 4 May 2007, accessed 5 May 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070504-aacs-la-internet-revolt-be-damned-this-fight-is-not-over.html

[103] ——, "HDCP: beta testing DRM on the public?" Ars Technica, 21 Jan. 2007, accessed 22 January 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070121-8665.html

[104] ——, "Privately, Hollywood admits DRM isn't about piracy," Ars Technica, 15 Jan. 2007, accessed 15 January 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070115-8616.html

[105] ——, "The AACS crack that wasn't: BackupHDDVD," Ars Technica, 7 Jan. 2007, accessed 3 Febuary 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070107-8564.html

[106] "TPM chip makes XBoX 360 hard to mod," Flexbeta, 12 Sept. 2005, accessed 28 April 2007. [Online]. Available: http://www.flexbeta.net/main/print.php?id=14740

[107] M. J. Foley, "Microsoft: 'Palladium' Is Still Alive and Kicking," Microsoft Watch, 5 May 2004, accessed 24 April 2007. [Online]. Available: http://www.microsoft-watch.com/content/operating_systems/microsoft_palladium_is_still_alive_and_kicking.html

[108] J. Fontana, "Microsoft piracy plan means work, concerns for IT," Network World, 5 Oct. 2006, accessed 8 October 2006. [Online]. Available: http://www.networkworld.com/news/2006/100506-microsoft-antipiracy.html

[109] Free Software Foundation, Inc., "Digital Restrictions Management and Treacherous Computing," Free Software Foundation, Inc., 18 Sept. 2006, accessed 22 January 2007. [Online]. Available: http://www.fsf.org/campaigns/drm.html

[110] ——, "BadVista.org: Stopping Vista adoption by promoting free software," Free Software Foundation, Inc., 10 Jan. 2007, accessed 13 January 2007. [Online]. Available: http://badvista.fsf.org

[111] ——, "DefectiveByDesign.org | The Campaign to Eliminate DRM," Free Software Foundation, Inc., Boston, MA, 12 Jan. 2007, accessed 13 January 2007. [Online]. Available: http://www.defectivebydesign.org/en/node

[112] I. Fried, "Microsoft battles piracy with free software," C|NET News.Com, 28 Oct. 2004, accessed 12 April 2007. [Online]. Available: http://news.com.com/2102-1016_3-5429449.html?tag=st.util.print

[113] ——, "Microsoft: Can we check your software license?" C|NET News.Com, 17 Oct. 2004, accessed 12 April 2007. [Online]. Available: http://news.com.com/2102-1016_3-5371664.html?tag=st.util.print

[114] ——, "Apple throws the switch, aligns with Intel," 7 June 2005, accessed 7 January 2007. [Online]. Available: http://news.com.com/2102-1014_3-5733756.html?tag=st.util.print

[115] ——, "Microsoft: Legit Windows or no updates," C|NET News.Com, 25 Jan. 2005, accessed 12 April 2007. [Online]. Available: http://news.com.com/2102-1016_3-5550205.html?tag=st.util.print

[116] ——, "Microsoft limits Vista transfers," C|NET News.com, 17 Oct. 2006, accessed 17 October 2006. [Online]. Available: http://news.com.com/2102-1016_3-6126379.html?tag=st.util.print

[117] ——, "MSN Music presses mute on downloads," C|NET News.Com, 2 Nov. 2006, accessed 4 November 2006. [Online]. Available: http://news.com.com/2102-1027_3-6132201.html?tag=st.util.print

[118] ——, "Microsoft makes copying Vista a monster task," C|NET News.Com, 30 Mar. 2007, accessed 12 April 2007. [Online]. Available: http://news.com.com/2102-1016_3-6171911.html?tag=st.util.print

[119] I. S. M. Fulton, "Does Trusted Computing provide security for users or from them?" TG Daily, 6 Oct. 2005, accessed 6 April 2007. [Online]. Available: http://www.schneier.com/news-012.html

[120] S. M. Fulton, III., "DRM chip for mobile devices may bind cell phones to service, content providers," TG Daily, 4 Oct. 2005, accessed 19 October 2006. [Online]. Available: http://www.tgdaily.com/2005/10/04/tpm_for_cellphones/print.html

[121] J. F. Gantz, C. A. Christiansen, and A. Gillen, "The Risks of Obtaining and Using Pirated Software," White Paper, International Data Corporation, Oct. 2006, accessed 6 January 2007. [Online]. Available: http://go.microsoft.com/fwlink/?LinkId=73969

[122] D. Geer, R. Bace, *et al.*, "CyberInsecurity: The Cost of Monopoly," Computer & Communications Industry Association, 24 Sept. 2003, accessed 4 November 2006. [Online]. Available: http://www.ccianet.org/papers/cyberinsecurity.pdf

[123] M. Geist, "Vista's Fine Print Raises Red Flags," 29 Jan. 2007, accessed 29 January 2007. [Online]. Available: http://www.michaelgeist.ca/index2.php?option=com_content&task=view&id=1640&Itemid=159&pop=1&page=0

[124] S. Ghosemajumder, "Advanced Peer-Based Technology Business Models," M.B.A. Master's Thesis, Massachusetts Institute of Technology, Sloan School of Management, Cambridge, MA, 2002, accessed 6 April 2007. [Online]. Available: http://shumans.com/p2p-business-models.pdf

[125] S. Gibson, "Spyware was Inevitable," *Communications of the ACM*, vol. 48, no. 8, pp. 37–39, Aug. 2005.

[126] ——, "EVERYONE should read this... Let's discuss!!!" 25 Dec. 2006, accessed 3 January 2007. [Online]. Available: news:/news.grc.com/grc.news.feedback/

[127] ——, "Re: EVERYONE should read this... Let's discuss!!!" 27 Dec. 2006, accessed 3 January 2007. [Online]. Available: news:/news.grc.com/grc.news.feedback/

[128] ——, "Re: EVERYONE should read this... Let's discuss!!!" 31 Dec. 2006, accessed 3 January 2007. [Online]. Available: news:/news.grc.com/grc.news.feedback/

[129] ——, "Re: EVERYONE should read this... Let's discuss!!!" 1 Jan. 2007, accessed 3 January 2007. [Online]. Available: news: /news.grc.com/grc.news.feedback/

[130] ——, "Re: Microsoft's response to Peter Gutmann's paper," 23 Jan. 2007, posted at 10:29:38 -0800, Accessed 7 February 2007. [Online]. Available: news:/news.grc.com/grc/news/feedback

[131] ——, "Re: Microsoft's response to Peter Gutmann's paper," 23 Jan. 2007, posted at 13:47:01 -0800, Accessed 7 February 2007. [Online]. Available: news:/news.grc.com/grc/news/feedback

[132] S. Gibson and L. Laporte, "Transcript of Episode #73: Digital Rights Management," Security Now! podcast, 4 Jan. 2007, accessed 5 January 2007. [Online]. Available: http://www.grc.com/sn/SN-073.pdf

[133] ——, "Transscript of Episode #74: Peter Gutmann on Vista DRM," Security Now! podcast, 11 Jan. 2007, accessed 12 January 2007. [Online]. Available: http://www.grc.com/sn/SN-074.pdf

[134] ——, "Transscript of Episode #75: Vista DRM Wrap-Up and Announcing "SecurAble"," Security Now! podcast, 18 Jan. 2007, accessed 19 January 2007. [Online]. Available: http://www.grc.com/sn/SN-075.pdf

[135] K. Greene, "The Encrypted Chip," Technology Review, 19 Apr. 2006, accessed 5 November 2006. [Online]. Available: http: //www.technologyreview.com/printer_friendly_article.aspx?id=16712

[136] R. Grover, "Why Hollywood Snubbed Jobs at Macworld," BusinessWeek, 12 Jan. 2007, accessed 15 January 2007. [Online]. Available: http://businessweek.com/print/bwdaily/dnflash/content/jan2007/db20070112_399642.htm

[137] P. Gutmann, "A Cost Analysis of Windows Vista Content Protection," 3 Apr. 2007, accessed 16 April 2007. [Online]. Available: http://www.cs.auckland.ac.nz/~pgut001/pubs/vista_cost.html

[138] J. A. Halderman, "AACS: Blacklisting, Oracles, and Traitor Tracing," 11 Jan. 2007, accessed 4 February 2007. [Online]. Available: http://www.freedom-to-tinker.com/?p=1107

[139] ——, "AACS: Game Theory of Blacklisting," 12 Jan. 2007, accessed 1 February 2007. [Online]. Available: http://www.freedom-to-tinker.com/?p=1108

[140] J. A. Halderman and E. W. Felten, "Lessons from the Sony CD DRM Episode," Revision 3, 16 May 2006, Center for Information Technology Policy, Department of Computer Science, Princeton University, 14 Feb. 2006, accessed 5 January 2007. [Online]. Available: http://itpolicy.princeton.edu/pub/sonydrm-ext.pdf

[141] J. A. Halderman, "Evaluating New Copy-Prevention Techniques for Audio CDs," in *Proceedings of 2002 ACM Workshop on Digital Rights Management*, ser. Lecture Notes on Computer Science, vol. 2696. Washington, DC: Springer-Verlag, 18 Nov. 2002, pp. 101–117, accessed 26 January 2007. [Online]. Available: http://www.cs.princeton.edu/~jhalderm/papers/drm2002.pdf

[142] P. Hales, "AMD's Opteron *won't* reject unlicensed content," The Inquirer, 20 Sept. 2002, accessed 7 April 2007. [Online]. Available: http://www.theinquirer.net/print.aspx?article=5489&print=1

[143] E. Hansen, "Ban on DVD-cracking code upheld," C|NET News.Com, 3 Mar. 2002, accessed 22 January 2007. [Online]. Available: http://news.com.com/2102-1023_3-276353.html?tag=st.util.print

[144] ——, "Microsoft prepares reply to iTunes," C|NET News.Com, 23 May 2003, accessed 12 November 2004. [Online]. Available: http://news.com.com/2102-1027_3-1009794.html?tag=st.util.print

[145] ——, "Behind the music: Microsoft?" C|NET News.Com, 1 Sept. 2004, accessed 12 November 2004. [Online]. Available: http://news.com.com/2102-1027_3-5343676.html?tag=st.util.print

[146] ——, "Court: DeCSS ban violated free speech," C|NET News.Com, 27 Feb. 2004, accessed 22 January 2007. [Online]. Available: http://news.com.com/2102-1026_3-5166887.html?tag=st.util.print

[147] T. Hauser and C. Wenz, "DRM Under Attack: Weaknesses in Existing Systems," in *Digital Rights Management: Technological, Economic, Legal*

and Political Aspects, ser. Lecture Notes on Computer Science, E. Becker, W. Buhse, *et al.*, Eds., vol. 2770, 2003, pp. 206–223, accessed 1 March 2007. [Online]. Available: http://www.vis.uky.edu/~cheung/courses/ee639_fall04/readings/DRM_attack.pdf

[148] A. Heaton, "Applying the Principle of Least Privilege to Windows Vista," Microsoft Corporation, 11 Oct. 2006, accessed 23 March 2007. [Online]. Available: http://www.microsoft.com/technet/community/columns/secmgmt/sm1006.mspx?pf=true

[149] A. Hermida, "Microsoft aims for hack-proof 360," BBC News, 9 Sept. 2005, accessed 28 April 2007. [Online]. Available: http://newsvote.bbc.co.uk/mpapps/pagetools/print/news.bbc.co.uk/1/hi/technology/4218670.stm

[150] K. Holton, "Independent record labels sign MySpace deal," Reuters, 22 Jan. 2007, accessed 25 January 2007. [Online]. Available: http://news.yahoo.com/s/mn/20070122/media_nm/myspace_independents_dc&printer=1

[151] R. Hoskins, "Phoenix Technologies Ltd. Introduces Trusted Security Suite to Eliminate Dangerous Endpoint Security Leaks," Broadband Wireless Exchange, 8 June 2006, accessed 5 April 2007. [Online]. Available: http://www.bbwexchange.com/pubs/2006/06/08/page1399-151468.asp

[152] J. Hu, "Microsoft opens MSN Music store," C|NET News.Com, 1 Sept. 2004, accessed 12 November 2004. [Online]. Available: http://news.com.com/2102-1027_3-5342795.html?tag=st.util.print

[153] J. Hu and J. Borland, "MSN Music: It's really about Windows," C|NET News.Com, 30 Aug. 2004, accessed 12 November 2004. [Online]. Available: http://news.com.com/2102-1027_3-5327631.html?tag=st.util.print

[154] M. Hypponen, "Malware Goes Mobile," *Scientific American*, vol. 295, no. 5, pp. 70–77, Nov. 2006.

[155] ——, "Targeted Attacks," F-Secure Security Labs, 16 Mar. 2007. [Online]. Available: http://www.youtube.com/watch?v=nFw9ZHy0V3c

[156] L. Iacolare, "Mac Security: The Evil DRM Chip Is Bolted Inside The New Intel Macs?" Master New Media - IKONOS New Media, 10 Apr. 2006, accessed 7 January 2007. [Online]. Available: http://www.masternewmedia.org/news/2006/04/05/mac_security_the_evil_drm.htm

[157] ——, "Windows Vista And Office 2007: Pros And Cons Of Upgrading – The Becta Report," MasterNewMedia, 12 Feb. 2007, accessed 4 April 2007. [Online]. Available: http://www.masternewmedia.org/microsoft/vista-and-office/pros-and-cons-of-upgrading-to-Vista-or-Office-2007-20070212.htm

[158] IBM Corporation, "IBM Extends Enhanced Data Security to Consumer Electronics Devices," IBM, 10 Apr. 2006, accessed 5 November 2006. [Online]. Available: http://www-03.ibm.com/press/us/en/pressrelease/19527.wss

[159] iGillottResearch Corporation, "The Trusted Computing Group Mobile Specification: Securing Mobile Devices on Converged Networks," White Paper, iGillottResearch Corporation, Sept. 2006, accessed 4 October 2006. [Online]. Available: https://www.trustedcomputinggroup.org/groups/movile/Final_iGR_mobile_security_white_paper_sept_2006.pdf

[160] Intel Corporation, "Intel® Platform Innovation Framework for EFI," Intel Corporation, accessed 7 April 2007. [Online]. Available: http://www.intel.com/technology/framework/overview1.htm, http://www.intel.com/technology/framework/overview2.htm, http://www.intel.com/technology/framework/overview3.htm, http://www.intel.com/technology/framework/overview4.htm, http://www.intel.com/technology/framework/overview5.htm

[161] ——, "Intel® Trusted Execution Technology," Intel Corporation, accessed 5 January 2007. [Online]. Available: http://www.intel.com/technology/security

[162] ——, "Intel® Trusted Execution Technology Architectural Overview," Intel Corporation, accessed 5 January 2007. [Online]. Available: http://www.intel.com/technology/security/downloads/arch-overview.pdf

[163] ——, "Intel® Trusted Execution Technology Overview," Intel Corporation, accessed 5 January 2007. [Online]. Available: http://www.intel.com/technology/security/downloads/trusted_exec_tech_over.pdf

[164] ——, "Extensible Firmware Interface," Intel Corporation, 1 Mar. 2006, accessed 7 April 2007. [Online]. Available: http://www.intel.com/technology/efi/

[165] Intel Corporation, IBM Corporation, *et al.*, "Advanced Access Content System (AACS) Technical Overview (informative)," Advanced Access Content System Licensing Administrator, LLC, 21 July 2004, accessed 19 January 2007. [Online]. Available: http://www.aacsla.com/marketplace/overview/aacs_technical_overview_040721.pdf

[166] S. Jobs, "Thoughts on Music," Apple, Inc., 6 Feb. 2007, accessed 7 February 2007. [Online]. Available: http://www.apple.com/hotnews/thoughtsonmusic/

[167] Jon Oltsik, "Trusted Enterprise Security: How the Trusted Computing Group (TCG) Will Advance Enterprise Security," White Paper, The

Enterprise Strategy Group Corporation, Jan. 2006, accessed 4 October 2006. [Online]. Available: https://www.trustedcomputinggroup.org/news/Industry_Data/ESG_White_Paper.pdf

[168] M. Kotadia, "Vista firewall shackled due to customer demand: Microsoft," ZDNet Australia, 26 Apr. 2006, accessed 8 January 2007. [Online]. Available: http://www.zdnet.com.au/news/security/print.htm?TYPE=story&AT=139252954-130061744t-1100000000c

[169] H. Krawczyk, M. Bellare, and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication," RFC 2104, Internet Engineering Task Force, Feb. 1997, accessed 25 November 2006. [Online]. Available: http://tools.ietf.org/html/rfc2104

[170] K. Kubicki, "Talk about HDMI," AnandTech, 17 Jan. 2005, accessed 18 January 2005. [Online]. Available: http://www.anandtech.com/printarticle.aspx?i=2321

[171] A. Kucer, "Protected Media Path and Driver Interoperability Requirements," Windows Hardware Engineering Conference presentation, Microsoft Corporation, 2005, accessed 28 December 2006. [Online]. Available: http://download.microsoft.com/download/9/8/f/98f3fe47-dfc3-4e74-92a3-088782200fe7/TWEN05005_WinHEC05.ppt

[172] J. Lacy, J. H. Snyder, and D. P. Maher, "Music on the Internet and the Intellectual Property Protection Problem," in *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE '97')*, vol. 1, Guimarães, Portugal, 7–11 July 1997, pp. SS77–SS83, accessed 1 March 2007. [Online]. Available: http://ieeexplore.ieee.org.proxy.lib.ohio-state.edu/iel4/5230/14218/00651739.pdf?tp=&arnumber=651739&isnumber=14218

[173] L. Lamport, *LaTeX : A Document Preparation System*, 2nd ed. Addison-Wesley, 1994.

[174] K. Langer, "Feature: Will your PC run Windows Vista?" PC World, 27 Apr. 2006, accessed 28 December 2006. [Online]. Available: http://www.pcw.co.uk/articles/print/2154785

[175] T. Lee, "Exec: Music labels "about to cave in the next six months" on DRM," Ars Technica, 25 Apr. 2007, accessed 29 April 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070228-8942.html

[176] ——, "FAIR USE Act analysis: DMCA reform left on the cutting room floor," Ars Technica, 28 Feb. 2007, accessed 28 April 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070228-8942.html

[177] Legal Information Institute, "US CODE: Title 17, 1201. Circumvention of copyright protection systems," Cornell Law School, accessed 6 Janury 2007. [Online]. Available: http://www.law.cornell.edu/uscode/html/uscode17/usc_sec_17_00001201----000-.html

[178] S. Lemon, "Doing business in China @ the speed of dumb," Computerworld Hong Kong, 13 July 2004, accessed 27 May 2004. [Online]. Available: http://www.cw.com.hk/Comment/c990713001.htm

[179] R. Lemos, "Russian crypto expert arrested at Def Con," C|NET News.Com, 2 Mar. 2002, accessed 4 February 2007. [Online]. Available: http://news.com.com/2102-1001_3-270082.html?tag=st.util.print

[180] ——, "Apple makes Trusted Computing cool," SecurityFocus, 2 Aug. 2006, accessed 20 March 2007. [Online]. Available: http://www.securityfocus.com/print/brief/270

[181] ——, "Trutsted computing a shield against worst attacks?" SecurityFocus, 31 Aug. 2006, accessed 5 April 2007. [Online]. Available: http://www.securityfocus.com/print/news/11410

[182] ——, "U.S. Army requires trusted computing," SecurityFocus, 28 July 2006, accessed 20 March 2007. [Online]. Available: http://www.securityfocus.com/print/brief/265

[183] L. Lessig, *Free Culture: How Big Media Uses Technology and the Law to Lock Down Culture and Control Creativity*. New York: The Penguin Press, 2004, accessed 6 January 2007. [Online]. Available: http://www.free-culture.cc/freeculture.pdf

[184] P. Levinthal, "Digital Media Content Protection," Windows Hardware Engineering Conference presentation, ATI Technologies Corporation, 2005, accessed 28 December 2006. [Online]. Available: http://download.microsoft.com/download/9/8/f/98f3fe47-dfc3-4e74-92a3-088782200fe7/TWEN05002_WinHEC05.ppt

[185] C. Levy, "Making Money with Streaming Media," Streaming Media Inc., 3 Feb. 2003, accessed 13 February 2007. [Online]. Available: http://www.streamingmedia.com/r/printerfriendly.asp?id=8306

[186] S. Levy, "The Big Secret," Newsweek, 24 June 2002, accessed 1 February 2005. [Online]. Available: http://cryptome.org/palladium-sl.htm

[187] ——, "A Net of Control," Newsweek, 2004, accessed 13 January 2007. [Online]. Available: http://www.msnbc.msn.com/id/3606168/print/1/displaymode/1098/

[188] P. A. Loscocco, S. D. Smalley, *et al.*, "The Inevitability of Failure: The Flawed Assumption of Security in Modern Computing Environments," in *Proceedings of the 21st National Information Systems Security Conference*. Crystal City, Virginia: National Institute of Standards and Technology, 5–8 Oct. 1998, accessed 1 March 2007. [Online]. Available: http://www.nsa.gov/selinux/papers/inevitability.pdf

[189] J. Lyle, "HDCP: what it is and how to use it," EDN, 18 Apr. 2002, accessed 8 April 2007. [Online]. Available: http://www.edn.com/index.asp?layout=articlePrint&articleID=CA209091

[190] J. Marchesini, S. Smith, *et al.*, "Experimenting with TCPA/TCG Hardware, Or: How I Learned to Stop Worrying and Love The Bear," Department of Computer Science/Dartmouth PKI Lab, Dartmouth College, Hanover, NH, Tech. Rep. TR2003-476, 15 Dec. 2003, accessed 5 January 2007. [Online]. Available: http://www.cs.dartmouth.edu/~sws/papers/mswm03.pdf

[191] Mark Russinovich, "Mark's Blog: More on Sony: Dangerous Decloaking Patch, EULAs and Phoning Home," Windows Sysinternals, 4 Nov. 2005, accessed 1 February 2007. [Online]. Available: http://blogs.technet.com/markrussinovich/archive/2005/11/04/more-on-sony-dangerous-decloaking-patch-eulas-and-phoning-home.aspx

[192] ——, "Mark's Blog: Sony, Rootkits, and Digital Rights Management Gone Too Far," Windows Sysinternals, 31 Oct. 2005, accessed 1 February 2007. [Online]. Available: http://blogs.technet.com/markrussinovich/archive/2005/10/31/sony-rootkits-and-digital-rights-management-gone-too-far.aspx

[193] J. Markoff, "Microsoft Moves To Weather Time Of Slow Growth," Late Edition - Final; Section C; Page 1; Column 5; Business/Financial Desk, The New York Times, 25 July 2003, lexisNexis™ Academic Database, accessed 9 May 2004.

[194] ——, "Attack of the Zombie Computers Is Growing Threat," The New York Times, 7 Jan. 2007, accessed 8 January 2007. [Online]. Available: http://www.nytimes.com/2007/01/07/technology/07net.html?pagewanted=print

[195] ——, "Tips for Protecting the Home Computer," The New York Times, 7 Jan. 2007, accessed 8 January 2007. [Online]. Available: http://www.nytimes.com/2007/01/07/technology/07tips.html?pagewanted=print

[196] D. Marsh, "Longhorn Output Content Protection," Windows Hardware Engineering Conference presentation, Microsoft Corporation, 2005, accessed 28 December 2006. [Online]. Available: http://download.microsoft.com/download/9/8/f/98f3fe47-dfc3-4e74-92a3-088782200fe7/TWEN05006_WinHEC05.ppt

[197] ——, "Output Content Protection and Windows Vista," White Paper, Microsoft Corporation, 27 Apr. 2005, accessed 20 October 2006. [Online]. Available: http://download.microsoft.com/download/5/D/6/5D6EAF2B-7DDF-476B-93DC-7CF0072878E6/output_protect.doc

[198] ——, "How To Implement Windows Vista Content Output Protection," Windows Hardware Engineering Conference presentation, Microsoft Corporation, 2006, accessed 28 December 2006. [Online]. Available: http://download.microsoft.com/download/5/b/9/5b97017b-e28a-4bae-ba48-174cf47d23cd/MED038_WH06.ppt

[199] ——, "Windows Vista Content Protection - Twenty Questions (and Answers)," Microsoft Corporation, 20 Jan. 2007, accessed 24 January 2007. [Online]. Available: http://windowsvistablog.com/blogs/windowsvista/archive/2007/01/20/windows-vista-content-protection-twenty-questions-and-answers.aspx

[200] I. Marson, "Trusted Computing comes under attack," ZDNet.co.uk, 27 Jan. 2006, accessed 20 March 2007. [Online]. Available: http://www.zdnet.co.uk/misc/print/0,1000000169,39249368-39001093c,00.htm

[201] M. Mastracci, "HDCP Conspiracy Attack," accessed 4 November 2006. [Online]. Available: http://www.grack.com/programming/misc/HDCPConspiracyAttack.html

[202] D. McCullagh, "Copyright lobbyists strike again," C|NET News.Com, 1 Aug. 2005, accessed 19 October 2006. [Online]. Available: http://news.com.com/2102-1071_3-5811025.html?tag=st.util.print

[203] T. McFadden, "TPM Matrix," 26 Mar. 2006, accessed 7 April 2007. [Online]. Available: http://www.tonymcfadden.net/tpmvendors_arc.html

[204] R. Merritt, "Longhorn: tough trail to PC digital media," EE Times, 2 May 2005, accessed 8 January 2007. [Online]. Available: http://www.google.com/search?q=cache:zABDk0o9pzkJ:www.eetimes.com/issue/fp/showArticle.jhtml%3FarticleID%3D162100180+http://www.eetimes.com/issue/fp/showArticle.jhtml%3FarticleID%3D162100180&hl=en&gl=us&ct=clnk&cd=1&client=opera

[205] Microsoft Corporation, "BitLocker Drive Encryption Frequently Asked Questions," Microsoft Corporation, accessed 30 December 2006. [Online]. Available: http://technet.microsoft.com/en-us/windowsvista/aa906016.aspx

[206] ——, "End-User License Agreement for Microsoft Software Microsoft Windows XP Professional Edition Service Pack 2,"

Microsoft Corporation, accessed 24 April 2007. [Online]. Available: http://download.microsoft.com/documents/useterms/Windows%20XP%20SP2_Professional_English_29e61d64-43e3-4ca3-b201-fe0c62507034.pdf

[207] ——, "Microsoft Software License Terms: Windows Vista Business," Microsoft Corporation, accessed 17 April 2007. [Online]. Available: http://download.microsoft.com/documents/useterms/Windows%20Vista_Business_English_e59f6893-6b14-4262-964c-993ed16d138a.pdf

[208] ——, "Microsoft Software License Terms: Windows Vista Home Basic, Windows Vista Home Premium, Windows Vista Ultimate," Microsoft Corporation, accessed 19 January 2006. [Online]. Available: http://download.microsoft.com/documents/useterms/Windows%20Vista_Ultimate_English_36d0fe99-75e4-4875-8153-889cf5105718.pdf

[209] ——, "Next-Generation Secure Computing Base," Microsoft Corporation, accessed 22 April 2007. [Online]. Available: http://www.microsoft.com/resources/ngscb/default.mspx

[210] ——, "Understanding Secure Audio Path," White Paper, Microsoft Corporation, accessed 18 January 2007. [Online]. Available: http://download.microsoft.com/download/a/1/a/a1a66a2c-f5f1-450a-979b-ddf790756f1d/WMRMsap-bro.pdf

[211] ——, "Microsoft Windows Media Data Session Toolkit," White Paper, Microsoft Corporation, 2002, accessed 25 January 2007. [Online]. Available: http://download.microsoft.com/download/a/1/a/a1a66a2c-f5f1-450a-979b-ddf790756f1d/Data_Session_Datasheet.pdf

[212] ——, "Microsoft Next-Generation Secure Computing Base - Technical FAQ," Microsoft Corporation, July 2003, accessed 9 January 2007. [Online]. Available: http://www.microsoft.com/technet/archive/security/news/ngscb.mspx

[213] ——, "Secure Startup — Full Volume Encryption: Executive Overview," White Paper, Microsoft Corporation, 22 Apr. 2005, accessed 22 April 2007. [Online]. Available: http://download.microsoft.com/download/5/D/6/5D6EAF2B-7DDF-476B-93DC-7CF0072878E6/secure-start_exec.doc

[214] ——, "Secure Startup — Full Volume Encryption: Technical Overview," White Paper, Microsoft Corporation, 22 Apr. 2005, accessed 22 April 2007. [Online]. Available: http://download.microsoft.com/download/5/D/6/5D6EAF2B-7DDF-476B-93DC-7CF0072878E6/secure-start_tech.doc

[215] ——, "Secure Startup – Full Volume Encryption," Microsoft Corporation, 3 May 2005, accessed 22 April 2007. [Online]. Available: http://www.microsoft.com/resources/ngscb/productinfo.mspx

[216] ——, "Trusted Platform Module Services in Windows Longhorn," White Paper, Microsoft Corporation, 25 Apr. 2005, accessed 21 March 2007. [Online]. Available: http://download.microsoft.com/download/5/D/6/5D6EAF2B-7DDF-476B-93DC-7CF0072878E6/TPM.doc

[217] ——, "Windows Media DRM FAQ," Microsoft Corporation, 17 Oct. 2005, accessed 18 January 2007. [Online]. Available: http://www.microsoft.com/windows/windowsmedia/forpros/drm/faq.aspx

[218] ——, "BitLocker Drive Encryption: Executive Overview," Microsoft Corporation, 14 Aug. 2006, accessed 30 December 2006. [Online]. Available: http://technet.microsoft.com/en-us/windowsvista/aa906018.aspx

[219] ——, "BitLocker Drive Encryption Step-by-Step Guide," Microsoft Corporation, 29 Sept. 2006, accessed 30 December 2006. [Online]. Available: http://technet.microsoft.com/en-us/windowsvista/aa905089.aspx

[220] ——, "BitLocker Drive Encryption: Technical Overview," Microsoft Corporation, 4 Apr. 2006, accessed 30 December 2006. [Online]. Available: http://technet.microsoft.com/en-us/windowsvista/aa906017.aspx

[221] ——, "How to troubleshoot problems that you may experience when you try to activate an Office product," Knowledge Base Article, Microsoft Corporation, 13 Nov. 2006, accessed 27 December 2006. [Online]. Available: http://support.microsoft.com/kb/903275

[222] ——, "Microsoft Statement: More Information on Windows Vista Software Protection Policies," Press Release, Microsoft Corporation, 14 Dec. 2006, accessed 31 December 2006. [Online]. Available: http://www.microsoft.com/presspass/press/2006/dec06/12-14ProtectionPR.mspx

[223] ——, "Microsoft's Software Protection Platform," Feature Story, Microsoft Corporation, 4 Oct. 2006, accessed 8 January 2007. [Online]. Available: http://www.microsoft.com/presspass/features/2006/oct06/10-04SoftwareProtection.mspx

[224] ——, "Microsoft's Software Protection Platform: Innovations for Windows Vista™ and Windows Server® "Longhorn"," White Paper, Microsoft Corporation, Oct. 2006, accessed 22 October 2006. [Online]. Available: http://download.microsoft.com/download/c/2/9/c935f83-1a10-4e4a-a137-c1db829637f5/10-03-06SoftwareProtectionWP.doc

[225] ——, "Benefits of Windows Media DRM," Microsoft Corporation, 2007, accessed 18 January 2007. [Online]. Available: http://www.microsoft.com/windows/windowsmedia/forpros/drm/benefits.aspx

[226] ——, "Features of Windows Media DRM," Microsoft Corporation, 2007, accessed 18 January 2007. [Online]. Available: http://www.microsoft.com/windows/windowsmedia/forpros/drm/features.aspx

[227] *Trust the Computer*, vol. 10, no. 7, Military Information Technology, 14 Aug. 2006, accessed 8 April 2007. [Online]. Available: http://www.military-information-technology.com/print_article.cfm?DocID=1593

[228] F. Mittelbach, M. Goossens, *et al.*, *The LaTeX Companion*, 2nd ed. Addison-Wesley, 2004.

[229] C. Mundie, "Trustworthy Computing - Today and in the Future," Silicon Valley Speaker Series Transcript, Microsoft Corporation, 13 Nov. 2002, accessed 9 January 2007. [Online]. Available: http://www.microsoft.com/presspass/exec/craig/11-13svspeaker.mspx

[230] C. Mundie, P. de Vries, *et al.*, "Trustworthy Computing," White Paper, Microsoft Corporation, 1 Oct. 2002, accessed 9 January 2007. [Online]. Available: http://download.microsoft.com/download/a/f/2/af22fd56-7f19-47aa-8167-4b1d73cd3c57/twc_mundie.doc

[231] MusicAlly, "Microsoft tells music biz to 'back lock-down CD standard'," The Register, 16 Sept. 2004, accessed 14 January 2007. [Online]. Available: http://www.theregister.co.uk/2004/09/16/ms_cd_copy_protection/print.html

[232] S. Nagaraja and R. Anderson, "The topology of covert conflict," University of Cambridge Computer Laboratory, Cambridge, United Kingdom, Tech. Rep. 637, July 2005, accessed 13 January 2007. [Online]. Available: http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-637.pdf

[233] D. Naor, M. Naor, and J. Lotspiech, "Revocation and Tracing Schemes for Stateless Receivers," in *Proceedings of Advances in Cryptology '01 (CRYPTO '01)*, ser. Lecture Notes in Computer Science, vol. 2139. Santa Barbara, CA: Springer-Verlag, 18 Nov. 2001, pp. 41–62, accessed 4 February 2007. [Online]. Available: http://citeseer.ist.psu.edu/rd/20178014%2C502910%2C1%2C0.25%2CDownload/http%3AqSqqSqwww.wisdom.weizmann.ac.ilqSqpeopleqSqhomepagesqSqnaorqSqPAPERSqSq2nl.pdf

[234] *Specification for the Advanced Encryption Standard (AES)*, National Institute of Standards and Technology Federal Information Processing Standard

197, 26 Nov. 2001, accessed 25 January 2007. [Online]. Available: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[235] *Secure Hash Standard*, National Institute of Standards and Technology Federal Information Processing Standard 180-2, 1 Aug. 2002, accessed 25 November 2006. [Online]. Available: http://csrc.nist.gov/publications/fips/fips180-2/fips180-2withchangenotice.pdf

[236] National Security Agency, "SELinux Frequently Asked Questions (FAQ)," National Security Agency, accessed 23 March 2007. [Online]. Available: http://www.nsa.gov/selinux/info/faq.cfm

[237] T. Nguyen, "Intel: Malicious Software and Viruses Can Be a Thing of the Past," DailyTech, 19 Oct. 2006, accessed 5 January 2007. [Online]. Available: http://dailytech.com/article.aspx?newsid=4624

[238] NOS News, "DRM on audio CD's abolished," NOS Journaal, 9 Jan. 2007, english translation, accessed 27 January 2007. [Online]. Available: http://lxer.com/module/newswire/printstory.php?rid=78008

[239] D. Nuhagic, "About – vLite – Windows Vista configuration tool," 18 Mar. 2007, accessed 28 April 2007. [Online]. Available: http://www.vlite.net/about.html

[240] ——, "nLite – Deployment Tool for the bootable Unattended Windows Installation – About," 5 Mar. 2007, accessed 28 April 2007. [Online]. Available: http://www.nliteos.com/nlite.html

[241] A. Orlowski, "The Microsoft Secure PC: MS patents a lock-down OS," The Register, 13 Dec. 2001, accessed 24 April 2007. [Online]. Available: http://www.theregister.co.uk/2001/12/13/the_microsoft_secure_pc_ms/print.html

[242] ——, "The open PC is dead – start praying, says HD guru," The Register, 7 Mar. 2001, accessed 31 December 2006. [Online]. Available: http://www.theregister.co.uk/2001/03/07/the_open_pc_is_dead/print.html

[243] OSx86 Project, "TPM," accessed 19 October 2006. [Online]. Available: http://wiki.osx86project.org/wiki/index.php?title=TPM&printable=yes

[244] M. Peinado, Y. Chen, *et al.*, "NGSCB: A Trusted Open System," in *Proceedings of the 9th Australasian Conference on Information Security and Privacy*, ser. Lecture Notes for Computer Science, vol. 3108, Microsoft Corporation. Sydney, Australia: Springer-Verlag, 13–15 July 2004, accessed 13 January 2007. [Online]. Available: http://research.microsoft.com/~yuqunc/papers/ngscb.pdf

[245] Peter Seebach, "Standards and specs: Digital rights management: When a standard isn't," IBM Corporation, 1 Nov. 2005, accessed 18 January 2007. [Online]. Available: http://www-128.ibm.com/developerworks/power/library/pa-spec11/?ca=dgr-lnxw06StandardDRM

[246] Phoenix Technologies, Ltd., "Phoenix TrustedCore," Phoenix Technologies, Ltd., accessed 7 April 2007. [Online]. Available: http://www.phoenix.com/PhoenixTPLT/tplt/www/en/PagePrint.aspx?PAGE_URL=/en/Products/Core+System+Software/TrustedCore/default.htm&PrinterFriendly=PrinterFriendly

[247] B. Potter, "The Trusted Computing Revolution," Black Hat, 2006, accessed 19 October 2006. [Online]. Available: http://www.blackhat.com/presentations/bh-usa-06/BH-US-06-Potter-trusted.pdf

[248] B. Prince, "Report Shows Spike in Online Identity Theft," eWeek, 16 Jan. 2007, accessed 22 March 2007. [Online]. Available: http://www.eweek.com/print_article2/0,1217,a=198721,00.asp

[249] M. Rasch, "Vista's EULA Product Activation Worries," SecurityFocus, 20 Nov. 2006, accessed 8 January 2007. [Online]. Available: http://www.securityfocus.com/print/columnists/423

[250] J. F. Reid and W. J. Caelli, "DRM, Trusted Computing and Operating System Architecture," in *Proceedings of Australasian Information Security Workshop 2005 (AISW2005)*, ser. Conferences in Research and Practice in Information Technology, P. Montague and R. Safavi-Naini, Eds., vol. 44. Newcastle, Australia: Information Security Research Centre, Queensland University of Technology, 2005, accessed 12 January 2007. [Online]. Available: http://citeseer.ist.psu.edu/rd/94362525%2C721418%2C1%2C0.25%2CDownload/http%3AqSqqSqeprints.qut.edu.auqSqarchiveqSq00000515qSq01qSqReid2005-AISW-DRM-TrustedComputing.pdf

[251] J. Reimer, "Blu-ray copy protection "cracked"," Ars Technica, 24 Jan. 2007, accessed 3 February 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070124-8697.html

[252] ——, "Crack in Blu-ray, HD DVD encryption gets wider," Ars Technica, 13 Feb. 2007, accessed 13 February 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070213-8837.html

[253] ——, "New AACS cracks cannot be revoked, says hacker," Ars Technica, 15 Apr. 2007, accessed 5 May 2007. [Online]. Available: http://arstechnica.com/news.ars/post/20070415-aacs-cracks-cannot-be-revoked-says-hacker.html

[254] M. Riley, "A Special Guide to DRM and Software Activation Tools: Protect Data, Enforce Licenses," Dr. Dobb's Journal, 17 Jan. 2006, accessed 18 January 2007. [Online]. Available: http://www.ddj.com/article/printableArticle.jhtml;jsessionid=IE2WOS5Z3WSPAQSNDLRSKHSCJUNN2JVN?articleID=184415471&dept_url=/dept/security/

[255] R. L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978, accessed 22 October 2006. [Online]. Available: http://theory.lcs.mit.edu/~rivest/rsapaper.pdf

[256] Roger L. Kay, "How to Implement Trusted Computing: A Guide to Tighter Enterprise Security," Endpoint Technologies Associates, accessed 4 October 2006. [Online]. Available: https://www.trustedcomputinggroup.org/news/Industry_Data/Implementing_Trusted_Computing_RK.pdf

[257] ——, "Trusted Computing is Real and it's Here," Endpoint Technologies Associates, 29 Jan. 2007, accessed 23 March 2007. [Online]. Available: https://www.trustedcomputinggroup.org/news/Industry_Data/Endpoint_Technologies_Associates_TCG_report_Jan_29_2007.pdf

[258] M. Rogers, "Let's See Some ID, Please," MSNBC, 13 Dec. 2005, accessed 22 March 2006. [Online]. Available: http://msnbc.msn.com/id/10441443/print/1/displaymode/1098

[259] D. Safford, "Clarifying Misinformation on TCPA," IBM Research, Oct. 2002, accessed 9 January 2007. [Online]. Available: http://researchweb.watson.ibm.com/gsal/tcpa/tcpa_rebuttal.pdf

[260] ——, "The Need for TCPA," IBM Research, Oct. 2002, accessed 13 January 2007. [Online]. Available: http://www.research.ibm.com/gsal/tcpa/why_tcpa.pdf

[261] ——, "Take Control of TCPA," Linux Journal, 27 Oct. 2006, accessed 21 March 2007. [Online]. Available: http://www.linuxjournal.com/node/6633/print

[262] J. H. Saltzer and M. D. Schroeder, "The Protection of Information in Computer Systems," in *Proceedings of the IEEE*, vol. 63, no. 9, Sept. 1975, paper Invited Paper, pp. 1278–1308, accessed 1 March 2007. [Online]. Available: http://www.acsac.org/secshelf/papers/protection_information.pdf

[263] S. E. Schechter, R. A. Greenstadt, and M. D. Smith, "Trusted Computing, Peer-To-Peer Distribution, and the Economics of Pirated Entertainment," in *Second Annual Workshop on Economics and Information Security (WEIS 2003)*. College Park, MD: University of Maryland,

29–30 May 2003, accessed 13 January 2007. [Online]. Available: http://www.eecs.harvard.edu/~stuart/papers/eis03.pdf

[264] P. B. Schneck, "Persistent Access Control to Prevent Piracy of Digital Information," in *Proceedings of the IEEE*, vol. 87, no. 7, July 1999, pp. 1239–1250, accessed 1 March 2007. [Online]. Available: http://ieeexplore.ieee.org.proxy.lib.ohio-state.edu/iel5/5/16709/00771075.pdf?tp=&arnumber=771075&isnumber=16709

[265] B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, 2nd ed. New York, NY: Wiley, 1996.

[266] ——, "Crypto-Gram Newsletter: May 15, 2001," 15 May 2001, accessed 31 December 2006. [Online]. Available: http://www.schneier.com/crypto-gram-0105.html

[267] ——, "Crypto-Gram Newsletter: August 15, 2002," 15 Aug. 2002, accessed 19 October 2006. [Online]. Available: http://www.schneier.com/crypto-gram-0208.html#1

[268] ——, *Secrets and Lies*. New York, NY: Copernicus Books, 2003.

[269] ——, "Cryptanalysis of MD5 and SHA: Time for a New Standard," Computerworld, 19 Aug. 2004, accessed 31 December 2006. [Online]. Available: http://www.schneier.com/essay-074.html

[270] ——, *Secrets and Lies*. Indianapolis, IN: Wiley, 2004.

[271] ——, "Cryptanalysis of SHA-1," 18 Feb. 2005, accessed 31 December 2006. [Online]. Available: http://www.schneier.com/blog/archives/2005/02/cryptanalysis_o.html

[272] ——, "Microsoft's Machiavellian manoeuvring," ZDNet UK, 1 Sept. 2005, accessed 19 October 2006. [Online]. Available: http://www.zdnet.co.uk/print/?TYPE=story&AT=39215921-39020682t-21000005c

[273] ——, "Everyone Wants to 'Own' Your PC," Wired News, 4 May 2006, accessed 3 October 2006. [Online]. Available: http://www.schneier.com/essay-113.html

[274] ——, "Microsoft's BitLocker," 2 May 2006, accessed 23 December 2006. [Online]. Available: http://www.schneier.com/blog/archives/2006/05/bitlocker.html

[275] ——, "Quickest Patch *Ever*," Wired News, 7 Sept. 2006, accessed 18 January 2007. [Online]. Available: http://www.schneier.com/essay-126.html

[276] ——, "Why Vista's DRM Is Bad For You," Forbes, 12 Feb. 2007, accessed 21 March 2007. [Online]. Available: http://www.schneier.com/essay-157.html

[277] S. Schoen, "Palladium Details," 8 July 2002, accessed 7 February 2005. [Online]. Available: http://www.activewin.com/articles/2002/pd.shtml

[278] ——, "Trusted Computing: Promise and Risk," Electronic Frontier Foundation, 1 Oct. 2003, accessed 13 January 2007. [Online]. Available: http://www.eff.org/Infrastructure/trusted_computing/20031001_tc.pdf

[279] ——, "EFF Comments on TCG Design, Implementation and Usage Principles 0.95," Electronic Frontier Foundation, 1 Oct. 2004, accessed 13 January 2007. [Online]. Available: http://www.eff.org/Infrastructure/trusted_computing/20041004_eff_comments_tcg_principles.pdf

[280] ——, "Microsoft Sells Out the Public on CGMS-A," Electronic Frontier Foundation, 27 July 2005, accessed 24 March 2006. [Online]. Available: http://www.eff.org/deeplinks/archives/003807.php

[281] ——, "Microsoft Trusted Computing Updates," Electronic Frontier Foundation, 15 July 2005, accessed 24 March 2006. [Online]. Available: http://www.eff.org/deeplinks/archives/003804.php

[282] ——, "Protected Media Path, Component Revocation, Windows Driver Lockdown," Electronic Frontier Foundation, 25 July 2005, accessed 24 March 2006. [Online]. Available: http://www.eff.org/deeplinks/archives/003806.php

[283] ——, "The Dangers of Device Authentication," Electronic Frontier Foundation, 19 July 2005, accessed 24 March 2006. [Online]. Available: http://www.eff.org/deeplinks/archives/003805.php

[284] S. Schwankert, "Windows Vista, Longhorn to get new antipiracy measures," Network World, 4 Oct. 2006, accessed 8 October 2006. [Online]. Available: http://www.networkworld.com/news/2006/100406-windows-vista-longhorn-to-get.html

[285] SecurityFocus, 27 Feb. 2007, accessed 28 April 2007. [Online]. Available: http://www.securityfocus.com/archive/1/461489/30/0/threaded

[286] Shane Rau, "The Trusted Computing Platform Emerges as Industry's First Comprehensive Approach to IT Security," Executive Brief, International Data Corporation, Feb. 2006, accessed 4 October 2006. [Online]. Available: https://www.trustedcomputinggroup.org/news/Industry_Data/IDC_448_Web.pdf

[287] V. Shannon, "Record labels rethink digital rights management at Midem," International Herald Tribune, 21 Jan. 2007, accessed 22 January 2007. [Online]. Available: http://www.iht.com/bin/print.php?id=4279356

[288] A. L. Shimpi, "Digital Visual Interface (DVI) Explained & Improving GeForce2/3 Image Quality," AnandTech, 17 Jan. 2002, accessed 8 April 200=7. [Online]. Available: http://anandtech.com/printarticle.aspx?i=1577

[289] ——, "Apple's Mac Pro – Upgrading CPUs, Memory & Running XP," AnandTech, 12 Sept. 2006, accessed 22 October 2006. [Online]. Available: http://anandtech.com/printarticle.aspx?i=2832

[290] D. Singer and M. Z. Visharam, *MPEG-4 File Formats white paper*, International Organization for Standardization Proposal ISO/IEC JTC 1/SC 29/WG 11 N7609, Oct. 2005, accessed 24 January 2007. [Online]. Available: http://www.chiariglione.org/mpeg/technologies/mp04-ff/index.htm

[291] A. Singh, "Trusted Computing for Mac OS X," 30 Oct. 2006, accessed 20 March 2007. [Online]. Available: http://www.osxbook.com/book/bonus/chapter10/tpm/

[292] ——, "Understanding Apple's Binary Protection in Mac OS X," 22 Oct. 2006, accessed 7 January 2007. [Online]. Available: http://osxbook.com/book/bonus/chapter7/binaryprotection/

[293] D. Slater, "Your General-Purpose PC –> Hollywood-Approved Entertainment Appliance," Electronic Frontier Foundation, 9 Aug. 2005, accessed 13 January 2007. [Online]. Available: http://www.eff.org/deeplinks/archives/003882.php

[294] P. Smith, "Windows Vista DRM nonsense," 31 Dec. 2006, accessed 1 January 2007. [Online]. Available: http://blogs.dasmirnov.net/paul/2006/12/31/windows_vista_drm_nonsense

[295] R. Smith, "Windows Vista Performance Guide," AnandTech, 1 Feb. 2007, accessed 2 February 2007. [Online]. Available: http://www.anandtech.com/printarticle.aspx?i=2917

[296] T. Smith, "Intel pushes 'East Fork' home PC 'back to Q1 2006'," The Register, 25 July 2005, accessed 19 October 2006. [Online]. Available: http://www.theregister.co.uk/2005/07/25/intel_east_fork_delay/print.html

[297] R. Stallman, "Can you trust your computer?" GNU Project, 18 Dec. 2006, accessed 20 March 2007. [Online]. Available: http://www.gnu.org/philosophy/can-you-trust.html

[298] ——, "The Right to Read," 29 Nov. 2006, accessed 13 January 2007. [Online]. Available: http://www.gnu.org/philosophy/right-to-read.html

[299] N. Stam, "Inside Intel's Secretive 'LaGrande' Project," ExtremeTech, 19 Sept. 2003, accessed 5 January 2007. [Online]. Available: http://www.extremetech.com/print_article2/0,1217,a=107418,00.asp

[300] M. Stefik, "Shifting the Possible: How Trusted Systems and Digital Property Rights Challenge Us to Rethink Digital Publishing," *Berkeley Technology Law Journal*, vol. 12, no. 1, 1997, accessed 20 March 2007. [Online]. Available: http://btlj.boalt.org/data/articles/12-1_spring_1997_symp_6-stefik.pdf

[301] F. A. Stevenson, "Cryptanalysis of Content Scrambling System," 8 Nov. 1999, accessed 1 February 2007. [Online]. Available: http://web.archive.org/web/20021211053158/http://dvd-copy.com/news/cryptanalysis_of_contents_scrambling_system.htm

[302] System Integrity Team, "System Integrity Team Blog," Microsoft Corporation, 26 Oct. 2006, accessed 24 April 2007. [Online]. Available: http://blogs.msdn.com/si_team/default.aspx

[303] The Economist, "A radical rethink," The Economist, 23 Jan. 2003, accessed 19 October 2006. [Online]. Available: http://www.economist.com/opinion/PrinterFriendly.cfm?story_id=1547223

[304] ——, "BSA or just BS?" *The Economist*, vol. 375, no. 8427, p. 69, 21 May 2005, Academic Search Premier database. Accessed 31 December 2006. [Online]. Available: http://web.ebscohost.com.proxy.lib.ohio-state.edu/ehost/delivery?vid=3&hid=105&sid=177d7e05-075e-4c65-a11d-5dc55cfbc932%40sessionmgr102

[305] B. Thompson, "Sign software on the digital line," BBC, 2 Feb. 2006, accessed 19 October 2006. [Online]. Available: http://newsvote.bbc.co.uk/mpapps/pagetools/print/news.bbc.co.uk/1/hi/technology/4677454.stm

[306] ——, "What price for 'trusted PC security'?" BBC, 18 Mar. 2006, accessed 19 October 2006. [Online]. Available: http://newsvote.bbc.co.uk/mpapps/pagetoos/print/news.bbc.co.uk/2/hi/technology/4360793.stm

[307] P. Thurrott, "Microsoft's Secret Plan to Secure the PC," Windows & .NET Magazine, 23 June 2002, accessed 9 January 2007. [Online]. Available: http://www.winnetmag.com/Articles/Print.cfm?ArticleID=25681

[308] ——, "Windows Vista Product Editions," Windows IT Pro, 2 Mar. 2006, accessed 22 April 2007. [Online]. Available: http://www.winsupersite.com/showcase/winvista_editions_final.asp

[309] Transmeta Corporation, "Transmeta Announces First Embedded Security Features for x86 Microprocessors," Transmeta Corporation, 14 Jan. 2003, accessed 7 April 2007. [Online]. Available: http://investor.transmeta.com/news/20030114-99407.cfm

[310] Trusted Computing Group, "About the Trusted Computing Group," Trusted Computing Group, accessed 19 October 2006. [Online]. Available: https://www.trustedcomputinggroup.org/about/

[311] ——, "Trusted Computing: A Solutions Guide for Personal Computers," Trusted Computing Group, 2003.

[312] ——, "TCG Specification Architecture Overview," Specification – Revision 1.2, Trusted Computing Group, 28 Apr. 2004, accessed 27 March 2007. [Online]. Available: https://www.trustedcomputinggroup.org/groups/TCG_1_0_Architecture_Overview.pdf

[313] ——, "Embedded Systems and Trusted Computing Security," Backgrounder, Trusted Computing Group, 14 Sept. 2005, accessed 3 October 2006. [Online]. Available: https://www.trustedcomputinggroup.org/groups/tpm/embedded_bkgdr_final_sept_14_2005.pdf

[314] ——, *TCG EFI Platform*, Trusted Computing Group Specification Version 1.20 Final, Rev. 1.00, 9 June 2006, accessed 7 April 2007. [Online]. Available: https://www.trustedcomputinggroup.org/specs/PCClient/TCG_EFI_Platform_1_20_Final_.pdf

[315] ——, *TCG EFI Protocol*, Trusted Computing Group Specification Version 1.20 Final, Rev. 1.00, 9 June 2006, accessed 7 April 2007. [Online]. Available: https://www.trustedcomputinggroup.org/specs/PCClient/TCG_EFI_Protocol_1_20_Final_.pdf

[316] ——, *TPM Main Part 1 Design Principles*, Trusted Computing Group Specification Version 1.2, Rev. 94, 29 Mar. 2006, accessed 27 March 2007. [Online]. Available: https://www.trustedcomputinggroup.org/specs/TPM/Main_Part1_Rev94.zip

[317] ——, *TPM Main Part 2 TPM Structures*, Trusted Computing Group Specification Version 1.2 Level 2, Rev. 94, 29 Mar. 2006, accessed 27 March 2007. [Online]. Available: https://www.trustedcomputinggroup.org/specs/TPM/Main_Part2_Rev94.zip

[318] ——, "Trusted Computing Group Backgrounder," Trusted Computing Group, Oct. 2006, accessed 3 November 2006. [Online]. Available: https://www.trustedcomputinggroup.org/about/TCGBackgrounder_revised_amp_oct_17_06.pdf

[319] ——, "Trusted Computing Group Fact Sheet," Trusted Computing Group, Oct. 2006, accessed 14 January 2006. [Online]. Available: https://www.trustedcomputinggroup.org/about/FACTSHEET_revised_nov_27_2006.pdf

[320] ——, "Trusted Computing Group Frequently Asked Questions," Trusted Computing Group, Jan. 2006, accessed 31 December 2006. [Online]. Available: https://www.trustedcomputinggroup.org/faq/CompleteFAQ/

[321] ——, "Trusted Network Connect Frequently Asked Questions," Trusted Computing Group, Sept. 2006, accessed 14 January 2007. [Online]. Available: https://www.trustedcomputinggroup.org/groups/network/TNC_FAQ_updated_sept_13_2006.pdf

[322] ——, "Stopping Rootkits at the Network Edge," White Paper, Trusted Computing Group, Jan. 2007, accessed 13 January 2007. [Online]. Available: https://www.trustedcomputinggroup.org/news/Industry_Data/Whitepaper_Rootkit_Strom_v3.pdf

[323] ——, "TCG Glossary of Technical Terms," Trusted Computing Group, 2007, accessed 27 March 2007. [Online]. Available: https://www.trustedcomputinggroup.org/groups/glossary/

[324] ——, "Trusted Computing Group Frequently Asked Questions," Trusted Computing Group, Jan. 2007, accessed 7 April 2007. [Online]. Available: https://www.trustedcomputinggroup.org/faq/

[325] Trusted Computing Group Best Practices Committee, "Design, Implementation, and Usage Principles," Best Practices, Trusted Computing Group, Dec. 2005, accessed 4 October 2006. [Online]. Available: https://www.trustedcomputinggroup.org/specs/bestpractices/Best_Practices_Principles_Document_V2.0.pdf

[326] D. D. Turner, "Apple Places Encrypted Binaries in Mac OS X," eWeek, 3 Nov. 2006, accessed 7 January 2007. [Online]. Available: http://www.eweek.com/print_article2/0,1217,a=193117,00.asp

[327] United States Patent and Trademark Office, "Answers to the most frequently asked kids' questions about patents, trademarks and copyrights and the U.S. Patent and Trademark Office," United States Patent

and Trademark Office, accessed 15 January 2007. [Online]. Available: http://www.uspto.gov/web/offices/ac/ahrpa/opa/kids/kidprimer.html

[328] ——, "Marks Composed, in Whole or in Part, of Domain Names," Examination Guide No. 2-99, United States Patent and Trademark Office, 29 Sept. 1999, accessed 15 January 2007. [Online]. Available: http://www.uspto.gov/web/offices/tac/notices/guide299.htm

[329] ——, "What are Patents, Trademarks, Servicemarks, and Copyrights?" United States Patent and Trademark Office, 12 May 2004, accessed 31 December 2006. [Online]. Available: http://www.uspto.gov/web/offices/pac/doc/general/whatis.htm

[330] "Unnecessary", "Looking at Fiji and Vienna," 29 Dec. 2006, accessed 8 January 2007. [Online]. Available: http://jameskyton.wordpress.com/2006/12/29/beyond-windows-vista-fiji-and-vienna/

[331] U.S. Copyright Office, "Copyright Law of the United States of America and Related Laws Contained in Title 17 of the United States Code," Circular 92, U.S. Copyright Office, July 2006, accessed 15 January 2007. [Online]. Available: http://www.copyright.gov/title17/92chap1.html

[332] ——, "Copyright Office Basics," Circular 1, U.S. Copyright Office, July 2006, accessed 19 October 2006. [Online]. Available: http://www.copyright.gov/circs/circ1.html

[333] H. Varian, "New Chips Can Keep a Tight Rein on Consumers," The New York Times, 4 July 2002, accessed 9 May 2004. [Online]. Available: http://www.nytimes.com/2002/07/04/business/04SCEN.html

[334] E. von Hippel, "Open source software projects as user innovation networks," in *Open Source Software: Economics, Law, and Policy*, Toulouse, France, 20–21 June 2002, accessed 13 January 2007. [Online]. Available: http://www.idei.fr/doc/conf/sic/papers_2002/vonhippel.pdf

[335] F. von Lohmann, "Why Would MS Do Hollywood's Bidding?" Electronic Frontier Foundation, 12 Aug. 2005, accessed 13 January 2007. [Online]. Available: http://www.eff.org/deeplinks/archives/003897.php

[336] J. Walker, "The Digital Imprimatur," 4 Nov. 2003, accessed 13 January 2007. [Online]. Available: http://www.fourmilab.ch/documents/digital-imprimatur

[337] A. Weiss, "Trusted Computing: who will control the PC of the future?" *netWorker*, vol. 10, no. 3, pp. 18–25, Sept. 2006.

[338] F. Wiki, "Documentation," Linux on Xbox 360, accessed 22 October 2006. [Online]. Available: http://wiki.free60.org/Documentation?action=print

[339] Wikipedia, "WIPO Copyright and Performances and Phonograms Treaties Implementation Act," Wikimedia Foundation Corporation, 24 Nov. 2006, accessed 25 January 2007. [Online]. Available: http://en.wikipedia.org/w/index.php?title=WIPO_Copyright_and_Performances_and_Phonograms_Treaties_Implementation_Act&printable=yes

[340] ——, "Digital Millennium Copyright Act," Wikimedia Foundation Corporation, 20 Jan. 2007, accessed 25 January 2007. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Digital_Millennium_Copyright_Act&printable=yes

[341] ——, "Digital Rights Management," Wikimedia Foundation Corporation, 25 Jan. 2007, accessed 25 January 2007. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Digital_Rights_Management&printable=yes

[342] ——, "Elliptic curve cryptography," Wikimedia Foundation Corporation, 5 Jan. 2007, accessed 25 January 2007. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Elliptic_curve_cryptography&printable=yes

[343] ——, "FairPlay," Wikimedia Foundation Corporation, 19 Jan. 2007, accessed 25 January 2007. [Online]. Available: http://en.wikipedia.org/w/index.php?title=FairPlay&printable=yes

[344] ——, "FairPlay," Wikimedia Foundation Corporation, 17 Jan. 2007, accessed 25 January 2007. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Windows_Media_DRM&printable=yes

[345] ——, "High-bandwidth Digital Content Protection," Wikimedia Foundation Corporation, 4 Apr. 2007, accessed 8 April 2007. [Online]. Available: http://en.wikipedia.org/w/index.php?title=High-bandwidth_Digital_Content_Protection&printable=yes

[346] ——, "Next-Generation Secure Computing Base," Wikimedia Foundation Corporation, 17 Apr. 2007, accessed 24 April 2007. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Next-Generation_Secure_Computing_Base&printable=yes

[347] ——, "Screen scraping," Wikimedia Foundation Corporation, 7 Mar. 2007, accessed 21 March 2007. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Screen_scraping&printable=yes

[348] ——, "Trusted Computing," Wikimedia Foundation Corporation, 4 Mar. 2007, accessed 21 March 2007. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Trusted_Computing&printable=yes

[349] O. Wild, "Enforcer Homepage," 9 Apr. 2004, accessed 24 March 2007. [Online]. Available: http://enforcer.sourceforge.net

[350] R. Wobst and J. Schmidt, "Hash cracked," Heise Zeitschriften Verlag, 8 Apr. 2006, accessed 31 December 2006. [Online]. Available: http://www.heise-security.co.uk/articles/75686, http://www.heise-security.co.uk/articles/75686/1, http://www.heise-security.co.uk/articles/75686/2

[351] World Intellectual Property Organization, "What is WIPO?" World Intellectual Property Organization, accessed 15 January 2007. [Online]. Available: http://www.wipo.int/about-wipo/en/what_is_wipo.html

[352] D. Worthington, "Phoenix Sounds Death Knell for BIOS," BetaNews, 27 Nov. 2003, accessed 5 January 2007. [Online]. Available: http://www.betanews.com/article/print/Phoenix_Sounds_Death_Knell_for_BIOS/1069920675

[353] T. Yager, "Apple's OS X x86 Darwin Kernel is Open Source," InfoWorld, 7 Aug. 2006, accessed 22 October 2006. [Online]. Available: http://weblog.infoworld.com/enterprisemac/archives/2006/05/apples_os_x_x86.html