RICHARD R. WIEBE (SBN 121156)
425 California Street, Suite 2025
San Francisco, CA 94104
Telephone: (415) 433-3200
Facsimile: (415) 433-6382

THOMAS E. MOORE III (SBN 115107)
TOMLINSON ZISKO MOROSOLI & MASER LLP
200 Page Mill Road, Second Floor
Palo Alto, CA 94306
Telephone: (650) 325-8666
Facsimile: (650) 324-1808

ALLONN E. LEVY (SBN 187251)
HS LAW GROUP
210 N. Fourth St., Suite 201
San Jose, CA 95112
Telephone: (408) 295-7034
Facsimile: (408) 295-5799

ROBIN D. GROSS (SBN 200701)
ELECTRONIC FRONTIER FOUNDATION
454 Shotwell Street
San Francisco CA 94110
Telephone: (415) 436-9333
Facsimile: (415) 436-9993

Attorneys for Defendant ANDREW BUNNER

SUPERIOR COURT OF THE STATE OF CALIFORNIA

COUNTY OF SANTA CLARA

DVD COPY CONTROL ASSOCIATION, INC., Plaintiff, v. ANDREW THOMAS MCLAUGHLIN; ANDREW BUNNER; et al., Defendants.	Case No. CV - 786804 **DECLARATION OF COMPUTER SCIENTIST DAVID S. TOURETZKY** **IN SUPPPORT OF DEFENDANT ANDREW BUNNER'S MOTION FOR SUMMARY JUDGMENT**

I, DAVID S. TOURETZKY, declare:

1. I am currently a Principal Scientist in the Computer Science Department and the Center for the Neural Basis of Cognition at Carnegie Mellon University, in Pittsburgh, Pennsylvania. I earned both my M.S. and Ph.D. degrees in Computer Science from Carnegie Mellon University. I lecture regularly around the world on such topics as cognitive science, artificial intelligence, robotics, and neural networks. I have authored three books, edited or co-edited nine collections of scholarly works, and authored or co-authored dozens of articles for scholarly journals, conference presentations, and the like. Over the past 25 years I have taught computer science material in a variety of formats, including brief tutorials at national conferences, week-long seminars for industrial clients, and semester-length university courses.

2. I have been interested in the issues surrounding DVD encryption since first hearing about this case in December 1999. At that time, I learned of two DVD decryption programs. The first is DECSS.EXE, a decryption program written for the Microsoft Windows family of operating systems. The second, known as css-auth, is written for Linux, a version of the Unix operating system. Both programs allow users to access a DVD drive and decrypt a DVD movie. The term "DeCSS" originally referred to DECSS.EXE, but has since been used as a generic term for any piece of software that defeats CSS encryption. Therefore, in this declaration I will avoid using "DeCSS" and instead refer explicitly to various DVD decryption programs by name (e.g., DECSS.EXE or css-auth).

EXPLANATION OF CSS ENCRYPTION TECHNOLOGY

3. The sounds and images of movies are translated into digital form for storage and playback by computers and other electronic devices. The information is stored in a publicly-disclosed file format called MPEG, which contains no encryption or access limitation technology. Software for recording and playing MPEG files is widely available.

4. In order to control access to the content distributed on DVD movie disks, motion picture studios encrypt their MPEG movie audiovisual data using a scheme called CSS (Content Scrambling System). The CSS-encrypted MPEG movie data is divided into numerous separate files when it is stored on a DVD disk.

5. CSS is based on a type of encryption algorithm known as a "stream cipher," in which a message is encrypted by combining it mathematically with a stream of seemingly random bits (ones and zeros). The stream is generated by a mathematical formula, or algorithm, based on a numerical password called a "key." The stream is not truly random because the algorithm will always produce the same result when given the same key as input; this is what allows the message to be decrypted later. CSS uses a 5 byte key (or equivalently, a 40 bit key, since a byte is a group of eight bits.) To recover the original message from a stream of encrypted bytes, one merely needs to know the 5 byte key that was used to initialize the stream generator; one can then recreate the stream of pseudo-random bits and subtract them from the encrypted data to obtain the unencrypted message.

6. When encrypted movies are distributed on DVDs, the disk must also contain the 5-byte key used to encrypt the movie data, so that the movie can be decrypted and viewed. The protection afforded by CSS is based on the assumptions that (a) consumers don't know how the files are encrypted, and (b) untrusted software running on the consumer's computer will not be able to get at the key, while an authorized DVD player program can. To achieve this, several measures are taken. First, before a program is allowed to access the data on a DVD drive, the DVD player program must "unlock" the drive by going through an authentication sequence with it. This authentication sequence involves an exchange of encrypted messages between the computer and the drive, using one of a set of 32 initial keys. In this way, the DVD player program "proves" to the drive that it knows the secret encryption scheme, and therefore is authorized to access the movie data on the disk.

7. This protection scheme is imperfect. One way around it is to use authorized software to unlock the drive and then switch over to unauthorized software. The drive cannot tell if the computer subsequently switches to a different, unauthorized program; it will continue to honor requests to access movie data files on the disk. Another problem is that the authentication sequence, including the set of 32 initial keys, has become widely known. Code to perform authentication is included as part of the css-auth package (in the file tstdvd.c), and is also included in various DVD player programs for Unix, such as Videolan (from the Ecole Centrale

Paris), Ogle (from Chalmers University of Technology in Sweden), and Xine. These players are "open source" programs, meaning their source code is freely distributed. (The Xine player requires a separate plug-in to unlock and decrypt a DVD. Source code for two different plug-ins with this functionality are available from third parties at the time of this writing.) Anyone interested can learn how to do DVD drive authentication by spending a few minutes reading some of this code. I recommend Videolan's vlc-dvd_css.c file.

8. CSS includes another way to protect DVD content even if the drive is unlocked. The key used to encrypt each movie file (called a "title key") is itself encrypted using a "disk key" that is unique to that disk. And the disk key is itself encrypted using each of 409 "master keys." Given any valid master key, one can decrypt the disk key, then use the disk key to decrypt each title key, and then use the title keys to decrypt the movie. Master keys were kept secret in an attempt to prevent this.

9. As a further precaution, when the disk and title keys are sent to the DVD player program by the DVD drive, they are encrypted using a "session key" exchanged between the drive and the DVD player program as part of the initial authentication process. This prevents the capture of unencrypted disk and title keys by eavesdropping on the computer's input/output bus.

10. Master keys (also called player keys) are not stored on the disk; they are stored either in a chip on a circuit board (in the case of a hardware DVD player) or embedded in an obscured fashion in a piece of executable software (in the case of software DVD players). Different DVD hardware and software player products were assigned different player keys so that if a particular player key were to be disclosed, the studios could simply stop using that key in any future DVD releases. This has in fact already happened. The Xing software DVD player's master key was revealed in 1999. The studios then discontinued use of this key, so players that rely on it are unable to play new movies. Both DECSS.EXE and css-auth employ the Xing key. The key has also been published in the Wall Street Journal, in haiku form ("Banned Code Lives in Poetry and Song", by David P. Hamilton, April 12, 2001, page B1, a copy of which is attached as Exhibit A).

11. More recent DVD decryption programs, such as VobDec, do not rely on player keys. They obtain the title key directly through a type of mathematical analysis known as a cryptographic attack. This is possible because the CSS stream cipher was poorly designed, as documented by Frank Stevenson. Mr. Stevenson's research paper on this topic, entitled "Cryptanalysis of Contents Scrambling System," has been widely circulated on the web, and is archived as part of my Gallery of CSS Descramblers web site, discussed below.

12. What Mr. Stevenson showed was that the mathematical function CSS uses to generate a stream of pseudo-random bits has certain predictable qualities, and as a result, one can make educated guesses about the title key that was used to encrypt a particular sequence of bytes, then test each guess. Due to a flaw in the design, the number of tests required to discover the title key is far less than it should be. In fact, it is small enough that a modern computer can uncover the title key in less than a minute. Mr. Stevenson also showed how the weaknesses in the encryption of the disk key could be used to recover all the player keys, and this was done in 1999. (See the www.free-dvd.org.lu web site, and the file www.free-dvd.org.lu/random-numbers.txt. The file name is an attempt at humor; the numbers are not random.) But as explained earlier, player keys are no longer needed now that the title key cipher's weaknesses are well understood.

THE CSS-AUTH SOURCE CODE HAS BEEN CONTINUOUSLY AVAILABLE SINCE THE BEGINNING OF THIS LITIGATION AND REMAINS WIDELY AVAILABLE

13. In December 1999 I established a "mirror" (local copy) of one of the DVD decryption programs, css-auth.tar.gz, on my web site at Carnegie Mellon. The css-auth.tar.gz file contains the software package css-auth. This mirror has remained continuously available on my web site from late December 1999 through today.

14. In March of 2000 I created a web site called the Gallery of CSS Descramblers, at http://www.cs.cmu.edu/~dst/DeCSS/Gallery (incorporated by reference in this declaration). I created this web site as a scholarly publication to illustrate the many forms an algorithm description could take, both in computer code and other forms of speech. My Gallery of CSS Descramblers presented a variety of exhibits, including the original css-auth source code in the C

programming language, a version of the css-auth code translated into a made-up computer language for which there was not yet a compiler (so, technically, it might not even be "code"), and a version of the css-auth code translated line-by-line into plain English.

15. The Gallery of CSS Descramblers has received extensive publicity and media coverage. On July 25, 2000, I testified as an expert witness for the defense in *Universal City Studios, et al. v. Reimerdes, et al.*, 111 F.Supp.2d 294 (S.D.N.Y 2000), commonly known as "the 2600 case." My testimony, which focused on the Gallery and the equivalence of computer code and other forms of speech, was featured in articles in the New York Times, the AP News wire service, the Hollywood Reporter, and several other publications. As a result, people began sending me contributions to the Gallery, in the form of computer code, audio recordings, graphic images, and animations. Each contribution expressed the css-auth source code or the underlying decryption algorithm in a creative way. For example, one person set the English description of the algorithm to music and sang it, with guitar and drum accompaniment. Another sent an image file in which the C program was cleverly encoded as a picture of Jack Valenti, president of the Motion Picture Association of America. And another person sent a 456-stanza haiku that included a complete and technically correct description of the css-auth decryption algorithm in perfect 5-7-5 syllable form.

16. The various exhibits added to the Gallery have resulted in additional media coverage, including articles in the New York Times, the Wall Street Journal, the Washington Post, the San Francisco Chronicle, Le Monde, the Bangkok Post, and Neue Zurcher Zeitung. USA Today named the Gallery a "Hot Site of the Day" for September 21, 2000. The Gallery now includes a collection of some 60 "press clippings," in the form of links to articles that discuss the Gallery or my testimony at trial. I have also made two television appearances to discuss the Gallery and the 2600 case. One was an interview on Tech TV's "Screen Savers;" the other was as a guest on John Dvorak's program, "Silicon Spin."

17. The Gallery has evolved to include not just representations of the css-auth code, but also technical descriptions and lecture notes about the CSS protection scheme and the decryption

algorithm, legal documents relating to the 2600 case, and links to web sites where other DVD decryption software can be found.

18. The Gallery is widely known on the Internet. Google, a popular Internet search engine (www.google.com), ranks its search results, or "hits," by the number of other sites that link to the site found by the search engine. A search for "DeCSS" using the Google Internet search engine on September 14, 2001 brought up the Gallery as the #2 hit out of a total of 77,800 hits returned. A reverse search from Google showed 594 sites with links to the Gallery, including links from Wired Magazine, USA Today, Slashdot, The Register, and the Association for Computing Machinery (the major professional organization for computer scientists.) The Gallery was also the first item listed in Google's human-edited directory on the topic "DVD CSS," which is part of the Cryptography section. See http://directory.google.com/Top/Society/Issues/Human_Rights_and_Liberties/Privacy/Cryptography/DVD_CSS.

19. DVD decryption software remains available from many other sources as well. On September 8, 2001, I used Google to performed a search for the string "css-auth.tar.gz." This is the name usually used for the file containing the source code of the css-auth package. The ".tar" extension denotes Tape ARchive format, which is a Unix convention for encapsulating a collection of files into one large file; the ".gz" extension indicates that the tar file has been compressed with a utility called gzip.

20. My search returned 830 hits, of which Google's heuristics decided 399 were likely to be unique pages. I examined the first 20 of these by visiting each link. There were 18 unique web sites in the first 20 hits. (Two sites were repeated due to hits on two separate pages on the same site.) Of those 18 unique sites, 9 contained local copies of css-auth.tar.gz, which I verified by downloading the file and either unzipping it or checking the file length in bytes. These sites were located in Austria, Denmark, Norway, the United Kingdom, and the United States. One was my own Gallery of CSS Descramblers. Another 8 of the 18 sites did not contain usable local copies of the file, but had links to other mirror sites where css-auth.tar.gz could be found. I followed some of those links and found additional copies of css-auth.tar.gz in Germany,

Luxembourg, the Netherlands, the United Kingdom, and the United States. The 18th site was down, but by retrieving a copy of the page from the Google cache I was able to determine that it was also a list of mirrors.

21. As a further test, I examined hits number 101 through 110 from the 399 results returned by Google. Each of these hits was a unique site, and none were included in the previous 20 results. 6 of these 10 sites contained local copies of css-auth.tar.gz; the servers were located in Germany, Switzerland, and the United States. Another site had a list of links to mirrors. Two of the sites were down. The tenth site, located in North Carolina, contained a press release and a link to the previously-mentioned Luxembourg site where the file could be found.

22. I also explored hits further down the list and found copies of css-auth.tar.gz on servers in Australia, France, Finland, New Zealand, and Poland.

23. Based on this experiment, I conclude that the css-auth source code remains widely available on the Internet, and can be found in a matter of seconds by anyone who bothers to look for it.

AVAILABILITY OF OTHER UNAUTHORIZED DVD SOFTWARE

24. Unauthorized DVD software falls into several categories: (1) Programs that capture individual frames from the computer's video card while the movie is being played by an authorized player. These were the first programs used to “rip” (capture and store) DVD movies, predating both DECSS.EXE and css-auth. They rely on an authorized player to do the actual decryption; they then intercept the movie’s audiovisual data after it has been decrypted. (2) Programs that decrypt DVD movies and store them on the computer’s hard drive. DECSS.EXE was the first decryption program in this category. The css-auth package also contains a program (css-cat.c) to do this. Many others have since been released, such as SmartRipper, VobDec, cladDVD, and DVD Decrypter. Some programs also compress the movie using a tool called DivX. Compression reduces the amount of disk space the movie takes up. (3) Programs that not only decrypt the movie but also play it on the computer’s monitor and speakers, rather than storing it on the hard drive. Examples include LiViD (available at www.au.linuxvideo.org), Videolan (available at www.videolan.org), Ogle (available at

http://www.dtek.chalmers.se/groups/dvd), and Xine (available at xine.sourceforge.net). (4) Software packages that simply provide drive authentication and/or decryption services. These are components for use in constructing other programs. One example is the css-auth package previously discussed. Another is my Gallery of CSS Descramblers, which contains numerous implementations of the basic decryption algorithm.

25. There are many web sites devoted to the subject of DVD decryption software. Examples include www.flexion.org, www.doom9.net, and www.afterdawn.com, which are all located outside the United States. In addition to offering downloadable copies of the software itself, these pages include tutorials on DVD decryption and reviews of the strengths and weaknesses of different tools.

CSS AND THE COMPUTER SCIENCE ACADEMIC COMMUNITY

26. CSS is of interest to computer scientists for a number of reasons. It's one of the first examples of encryption technology embedded in a home entertainment product. It's also a stellar example of the failure of what experts call the "security through obscurity" approach. "Security through obscurity" refers to concealment of information about how a security mechanism works in the hopes that no attacker will uncover its weaknesses. The alternative is to develop mathematically strong encryption algorithms, publicly disclose them, and allow them to be examined by experts to determine if the algorithms are truly sound. CSS was not designed to withstand such scrutiny. CSS does not provide true security because the scheme is vulnerable to reverse engineering, the stream cipher is much weaker than theoretically possible due to flaws in its design, and in any case, the decryption keys must be present on each DVD sold. So CSS is an object lesson in how not to design a security product.

27. The application of the Digital Millennium Copyright Act to DECSS.EXE and css-auth in the 2600 case has raised the issue of the First Amendment status of computer code, a topic of vital concern to computer scientists and engineers. It has thus generated widespread interest in CSS decryption software among computer scientists and academics, even those, such as myself, who have no desire to watch DVD movies.

28. Here are some examples of how CSS has made its way into the computer science curriculum. Gregory Kesden, who teaches an undergraduate computer science course on Operating Systems at Carnegie Mellon University, now includes a lecture on the CSS encryption scheme. His lecture notes are available on the web at http://www.cs.cmu.edu/~dst/DeCSS/Kesden/index.html. Professor Greg Newby at the University of North Carolina also covers CSS in his course Distributed Systems and Analysis; see http://www.ils.unc.edu/gbnewby/DVD for more information.

29. MIT held a two-session seminar on "Decrypting DVD" in January 2001. The speakers included two undergraduates, Keith Winstein and Marc Horowitz, plus Professor Hal Abelson of the MIT Laboratory for Computer Science, Harvard Law School Professor Jonathan Zittrain of the Berkman Center for Internet & Society at Harvard Law School, and David Barr, lead engineer for C-Cubed Microsystems. As part of this event, Winstein and Horowitz dissected the CSS encryption scheme and presented the world's shortest CSS decryption algorithm: a 7-line program in the Perl computer language (later shortened to 6 lines). They demonstrated the algorithm's correctness for the audience by decrypting and playing a portion of the movie *The Matrix*. Their Perl program has been published in the July/August 2001 issue of the MIT-published journal Technology Review as part of the article "The Net Effect: The DVD Rebellion," by Simson Garfinkel. Technology Review is a print journal, but the article is also available on the web at http://www.technologyreview.com/magazine/jul01/garfinkel.asp. (A copy of the print version of this article is attached as Exhibit B.) Wired Magazine also published the source code in an article on March 7, 2001, available on the web at http://www.wired.com/news/culture/0,1284,42259,00.html. (A copy of the print version of this article is attached as Exhibit C.) The publication of the Winstein and Horowitz work inspired an MIT alumnus, Charles M. Hannum, to devise a 7-line C program to implement the same algorithm. Both these programs attracted considerable media attention, including a March 8, 2001 article in ZDNet News that was picked up by USA Today and MSNBC, plus articles in Slashdot and The Register. Further publicity came when Phil Carmody, a computer scientist in the United Kingdom, found ways to encode these tiny programs as prime numbers. More

information on these programs and their prime number encodings is available at the Gallery of CSS Descramblers.

30. Another indication of the growing familiarity with CSS in the computer science community is the appearance of new video playing software that includes DVD decryption. Videolan (www.videolan.org) was created as an academic project by a group at the Ecole Centrale Paris. A listing of the students involved and their faculty advisors may be found at http://www.videolan.org/team.html. Similarly, Ogle was created by a group at Chalmers University of Technology in Sweden; see http://www.dtek.chalmers.se/groups/dvd/authors.html for their names. Both Videolan and Ogle are distributed under the GNU Public License, allowing anyone to download and read the source code.

SUMMARY AND CONCLUSION

31. The technical details of how CSS works and how it can be defeated are now widely known. Not only are the early decryption programs DECSS.EXE and css-auth still available, but they have been joined by more sophisticated solutions using a cryptographic attack (based on Frank Stevenson's work), and a profusion of more refined DVD descrambling software that is both more reliable and easier to use. The story of how CSS was defeated will almost certainly be included in the next generation of security and cryptography textbooks as a perfect example of why the "security through obscurity" approach does not work.

32. At this point, there is nothing secret about DVD encryption. The cat has been long out of the bag. In fact, she's produced several litters of kittens.

I, DAVID S. TOURETZKY, declare under penalty of perjury under the laws of the State of California that the foregoing is true and correct.

Dated: _____ _____

David S. Touretzky