CENDI 2009-1

Frequently Asked Questions about Copyright and Computer Software

Issues Affecting the U.S. Government with Special Emphasis on Open Source Software

Prepared by
CENDI Copyright Working Group

Compiled and edited by
Vicki Allums
Defense Information Systems Agency
and
Nancy Kremers
Defense Advanced Research Projects Agency

Published by
CENDI Secretariat
c/o Information International Associates, Inc.
Oak Ridge, TN

November 1, 2009

October 1, 2010 Revised

FREQUENTLY ASKED QUESTIONS ABOUT COPYRIGHT AND COMPUTER SOFTWARE: ISSUES AFFECTING THE U.S. GOVERNMENT WITH SPECIAL EMPHASIS ON OPEN SOURCE SOFTWARE

Prepared by
CENDI Copyright Working Group

Published by
CENDI Secretariat
c/o Information International Associates, Inc.
Oak Ridge, TN

DISCLAIMER

THIS DOCUMENT DOES NOT CONSTITUTE LEGAL ADVICE AND SHOULD NOT BE CONSTRUED OR USED AS SUCH. For specific questions related to the use of proprietary and open source computer software, licensing, and copyright issues, consult the appropriate program office and your agency's Office of the General Counsel. Consult CENDI's "Frequently Asked Questions about Copyright" for general information on copyrighted and U.S. Government works.

Purpose and Use of This Document

This document provides general guidance on a special category of copyrighted works— computer software—and includes a detailed discussion of open source software. Federal agencies are increasingly supporting the use and acquisition of open source software as an alternative to proprietary software in their information technology programs. It is hoped that this Frequently Asked Questions (FAQ) document will serve as a useful resource for contracting officers, program managers, librarians, information center staff, and attorneys.

Copyright Notice

Notice of Change

The information presented in this FAQ is subject to changes enacted by U.S. Government policies, legislation and case law. Please direct comments about this document to copyright@dtic.mil.

CENDI is an interagency cooperative organization composed of the scientific and technical information (STI) managers from the Departments of Agriculture, Commerce, Energy, Education, Defense, the Environmental Protection Agency, Health and Human Services, Interior, the National Aeronautics and Space Administration, the Government Printing Office, the National Archives and Records Administration, the National Science Foundation, and the Library of Congress. CENDI's mission is to help improve the productivity of federal science- and technology-based programs through the development and management of effective scientific and technical information support systems. In fulfilling its mission, CENDI member agencies play an important role in helping to strengthen U.S. competitiveness and address science- and technology-based national priorities.

Table of Contents

CENDI COPYRIGHT WORKING GROUP

CONTRIBUTING MEMBERS:
Bill Adams (Army); Vicki Allums (DISA); Jane Barrow (NAVSEA); Dale Berkley (NIH); Gary Borda (NASA); Cindy Clark (NIH); Christopher Cole (NAL); Linda Field (DOE); Courtney Graham (NASA); Richard Gray (DoD); Phil Greene (DoC); Gail Hodge (CENDI); Laura Jennings (NGA); Rob Kasunic (LoC); Flayo Kirk (MDA); Bonnie Klein (DTIC); Nancy Kremers (DARPA); Richard Lambert (NIH); Jeffrey Landou (NARA); Jan McNutt (NASA); Jeffrey Moore (AFRL); Hope O'Keeffe (LoC); Vinit Patel (DoE); John Raubitschek (Army), Vakare Valaitis (DTIC)

1.0 Glossary of Terms

Abbreviations and Acronyms

BSD	Berkley Software Distribution
CC	Creative Commons
COTS	Commercial-off-the-shelf
DFARS	Defense Federal Acquisition Regulation Supplement
DISA	Defense Information Systems Agency
DOC	Department of Commerce
DoD	Department of Defense
DOE	Department of Energy
DoN	Department of the Navy
EPA	Environmental Protection Agency
FAR	Federal Acquisition Regulation
FISMA	Federal Information Security Management Act
FOIA	Freedom of Information Act
FSF	Free Software Foundation
GPO	Government Printing Office
GNU GPL	General Public License
LGPL	Lesser General Public License
HHS	Department of Health and Human Services
NARA	National Archives and Records Administration
NASA	National Aeronautics and Space Administration
NOSA	NASA Open Source Agreement
NSF	National Science Foundation
OMB	Office of Management and Budget
OSI	Open Source Initiative
OSS	Open Source Software

Definitions

While the following terms may have more than one generally accepted meaning, as used in these FAQs (Frequently Asked Questions), they are defined as follows.

Computer Program means a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result. (See 17 USC § 101.)

Computer Software or ***software*** means one or more computer programs.

Commercial Computer Software, as defined in the DFARS and FAR, means software developed or regularly used for non-governmental purposes, which has been sold, licensed or leased to the public or is a commercial item. (See DFARS 252. 227.7014 (a) (i) and FAR 2.101.) Open source software is commercial computer software licensed under a licensing scheme that provides broad rights to modify and redistribute the original source code and, sometimes, any distributed modified versions (e.g., derivative works). (See FAQ Section 3.1.)

Derivative Work means a work that is based upon one or more preexisting works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which a work may be recast, transformed, or adapted. A work consisting of editorial revisions, annotations, elaborations, or other modifications, which, as a whole, represent an original work of authorship, is a "derivative work." In the computer industry, a second version of a software program is generally considered a derivative work based upon the earlier version.[1] (See 17 USC § 101.) The term "derived work" is often used in commercial parlance to mean "derivative work."

Defense Federal Acquisition Regulation Supplement (DFARS) means the supplement to the Federal Acquisition Regulations used by the Department of Defense to purchase goods and services.

Executable Code means a subroutine, method, procedure, or subprogram of a larger program that performs a specific task and can operate relatively independent of the remaining code. It can be self-contained or call upon other code to execute (take a specific action).

Federal Acquisition Regulation (FAR) means the regulation established to codify uniform policies for acquisition of supplies and services by federal executive agencies. It is issued and maintained jointly, pursuant to the Office of Federal Procurement Policy (OFPP) Reauthorization Act, under statutory authorities granted to the Secretary of Defense (DoD), Administrator of General Services (GSA), and the Administrator, National Aeronautics and Space Administration (NASA). The official FAR appears in the Code of Federal Regulations at 48 CFR Chapter 1.[2] The FAR applies to procurement contracting only; i.e., contracts to procure goods and services primarily for the benefit of the federal government. Other, very different laws and regulations apply to non-procurement award instruments, such as grants, cooperative agreements, "other transactions" agreements, CRADAs, and international agreements. Note also that a number of government agencies use an agency-specific version, or supplement, to the FAR; the DFARS, defined above, is one example. Even within government procurement contracting, there may be important substantive differences between the FAR and an agency supplement, so it is important to identify which acquisition regime is applicable to any particular transaction.

Object Code means computer program code that is written in machine-readable language.

Open Source License is a license to use software that provides the licensee the freedom to use the software for any purpose, to study and modify the software, and to redistribute copies of the original or modified software without payment of royalties. In order to provide the user these freedoms, open source licenses require that the user have access and use the software source code.

Open Source Software refers to software provided to users under an open source license.

Permissive Open Source Licenses allow distribution of the original and derivative works of the open source software under different terms than the original open source license. Thus, derivative works can become proprietary and the original open source software can be incorporated into proprietary software.

Proprietary software means software in which an owner has a legally protectable property interest allowing the owner to limit the way in which the software is treated by others. Many proprietary software products are considered commercial computer software.

Source Code means any sequence of computer programming statements or declarations written in some human-readable computer or programming language.

Strongly Protective Open Source Licenses require that the original open source licensed software, derivative works based of the licensed software, and any software that dynamically links to the licensed software be distributed under the same terms as the original open source license. This prevents the open source software and any derivative works from becoming proprietary or being incorporated into any proprietary software. The GNU General Public License (GPL) is an example of a strongly protective open source license. In the open source community, "strongly protective" open source licenses are also known as "c*opyleft strong*" licenses. ***Copyleft Strong*** licenses require that derivative works be distributed under the same terms as the original license.[1]

Unlimited Rights License means the license of the same name as defined at FAR 52.227-14, or the license of the same name as defined in an applicable agency-specific supplement to the FAR. The DFARS unlimited rights license is defined at DFARS 252.227-7014 (a) (15).

Weakly Protective Open Source Licenses allow derivative works to be distributed under terms different than the original license. This prevents the open source software component (often a software library) from becoming proprietary, yet permits it to be part of a larger proprietary program. Examples weakly protective open source license include the GNU Lesser General Public License (LGPL) and the Mozilla Public License. In the open source community, "weakly protective" open source licenses are also known as "c*opyleft weak*" licenses. ***Copyleft Weak*** licenses allow derivative works to be distributed under terms different than the copyleft provisions of the original license.

2.0 Computer Software Copyright Basics

2.1 General Information Regarding Copyright and Computer Software

2.1.1 Is computer software subject to copyright protection under Section 102 of the Copyright Act?

Yes. Computer programs are protected as "literary works" under Section 102 (a) (1) of the U.S. Copyright Act. Literary works are "works" other than audiovisual works, expressed in

[1] ***Copyleft*** is a general method for making a computer program or other work available free, and requiring all modified and extended versions of the program or work to be free as well, in an effort to include others in improving the program or as a continuing process. Copyleft licenses are referred to as "strong copyleft" or "weak copyleft," licenses depending on the extent to which they impose copyleft provisions on derivative works.

words, numbers, or other verbal or numerical symbols or indicia, regardless of the nature of the material objects, such as books, periodicals, manuscripts, phono-records, film, tapes, disks, or cards, in which they are embodied.

Congress and the courts have interpreted “literary works” to include computer programs because they are original works of authorship expressed in words, numbers, or other verbal or numerical symbols or indicia. (See 17 USC §§ 101 and 102 (a) (1) and FAQ Section 2.1.3.[3]) Copyright protection in computer software extends to both the source code and the object code.

However, not all of the features of a computer program are copyrightable. For example, the First Circuit has ruled that a “menu command hierarchy” is considered a method of operation and uncopyrightable subject matter. (See *Lotus Development Corporation v. Borland International, Inc.*, 516 U.S. 233 (*more*) 49 F.3d 807 (1st Cir. 1995), aff'd, 516 U.S. 233, 116 S. Ct. 804; 133 L. Ed. 2d 610 (1996).) In the Lotus case, the First Circuit reversed a district court decision holding that the Lotus menu command hierarchy was expressed in a particular way and was copyrightable. The Supreme Court affirmed the First Circuit’s ruling without a decision. Therefore, the question of whether all menu command hierarchies are methods of operation and uncopyrightable remains an unsettled area of copyright law. “A “method of operation”[4] refers to the means by which a person operates something, whether it’s a car, a food processor, or a computer. By definition (See Section 101 of the Copyright Act), computer programs are a set of statements or instructions that bring about a certain result, which is not very different from a method of operation. The question that the courts need to resolve is whether some methods of operation may include a particular expression of expression that may be copyrightable.

Methods of operation, including methods implemented by software, may be protected by patent if the method satisfies the requirements for patentability. (See FAQ 2.1.3.2.)

For further discussion and information on registering computer programs with the U.S. Copyright Office, see Copyright Office Circular No. 61, Copyright Registration for Computer Programs.

2.1.2 What rights are granted to owners of copyrights to computer software under Section 106 of the Copyright Act?

Under 17 USC § 106, owners of copyrights to computer software acquire the exclusive right to: (a) reproduce the software; (b) prepare derivative works based upon the original software; (c) distribute the software; (d) publicly perform; and (e) publicly display the software.

Although copyright owners of computer programs generally license their software for use by others, they typically restrict a licensee’s rights to modify, prepare derivative works, and distribute the computer program, and the owner(s) thereby retains these rights. Copyright owners commonly implement these restrictions by giving licensees access only to the object code and not the source code for the software. This a key difference between a proprietary and an open source licensing model. Under open source licenses, copyright owners allow others to exercise their exclusive rights with few, if any, limitations by allowing users to modify the source code, and to prepare and distribute derivative applications, provided that if the modifications are distributed, the source code is shared with the community of users. (See FAQ 4.1 and 4.10.)

While most computer programs are licensed and not sold, whether or not a particular "copy" of a computer program is owned or licensed is a separate question. Courts have begun looking at the ownership versus licensing question in relation to Section 117 of the Copyright Act. Two appellate cases out of the Second Circuit, "Krause v. Titleserv" and "Aymes v. Bonelli (Amyes II) have begun to look at the question in a more nuanced manner than the Ninth Circuit. In Krause, the judge examined the "incidents of ownership" rather than relying on mere title. In both Second Circuit cases, however, the computer programs involved were one-to-one transactions for relatively expensive software created for the particular end-user. How this relates to mass market software is unclear.

In the Ninth Circuit, a district court in Vernor v. Autodesk held that it was required to apply a case that preceded MAI v. Peak and its progeny. That case is currently going to the Ninth Circuit and it is being joined with another software/video game case, MDY v. Blizzard. In addition, a case involving the ownership or license of a promotional CD (UMG) is also going up to the Ninth Circuit. Both Section 109 (first sale) and Section 117 (limitations on computer programs) are limited to the owner of a copy. These cases will be addressing how to determine ownership. It may be that the Ninth Circuit will revisit MAI v. Peak, since that case and a few that followed, looked at the question somewhat superficially.

2.1.3 Is it possible to protect computer software under other types of intellectual property law?

Yes, in addition to copyright protection, computer software may also be protectable under trademark, patent and trade secret law. More than one type of protection may apply.

2.1.3.1 May computer software be protected by trademark law?

Yes, under certain conditions. Trademark law can provide protection for source indicators identifying or forming part of a computer program. Source indicators that may be eligible for trademark registration may include names, slogans, designs, graphics, sounds, or other devices by which a person or entity identifies itself as the source of a computer program.

2.1.3.2 May computer software be protected by patent law?

Yes, under certain conditions. Protection of software as a patent is the result of a recent interpretation of the scope of patentable subject matter by the courts.[5] In the late 1990s, the U.S. Patent and Trademark Office began issuing patents for software applications involving methods of operation or processes (aka "business method patents"), a practice which expanded rapidly following the Court of Appeals for the Federal Circuit's decision in *State Street Bank & Trust Co. v. Signature Financial Group Inc.*, 149 F.3d 1368 (Fed. Cir. 1998).

Business method patents have generated controversy in the U.S. and can be difficult and expensive to obtain. Under a recent court decision, *In re Bilski*, 545 F. 3d 943, 88 U.S.P.Q 2d 1385 (Fed. Cir. 2008), the future of business method patents in the U.S. is unclear. The *Bilski* court ruled that the "useful, concrete, and tangible result" test used in *State Street* should no longer be relied upon. The court also reiterated the "machine-or-transformation test" as the applicable test for patent-eligible subject matter. Whether and to what extent business method patents (both new applications and patents already issued) may successfully meet this test remains to be seen.

The controversy surrounding the patentability of software applications remains unresolved after the U.S. Supreme Court's decision in Bilski v. Kappos, 561 U.S. (2010) issued on June 28, 2010. The court affirmed the Federal Circuit court's decision invalidating Bilski's patent. In addition, it also held that the "machine-or-transformation" test was not the exclusive test for determining whether a claimed process is patentable under Section 101 of Title 35. However, it avoided the larger question of the patentability of business methods software. Thus, the patentability of this type of software will continue to be reviewed on a case-by-case basis. The U.S. Patent and Trademark Office revised its instructions for determining subject eligibility and noted that the guidelines were "for the interim, pending the U.S. Supreme Court's decision in Bilski." It remains unclear how the instructions will be revised in light of the Supreme Court's decision. Patent rights in software created under a government contract are generally addressed under FAR 52.227-11 and 52.227-13. The DFARS addresses this issue at 252.227-7038.

2.1.3.3. May computer software be protected as a trade secret?

Yes, computer programs may be protected as trade secrets under both state and federal law and various licensing arrangements. While trade secrets law developed in the U.S. through the common law among the various States, a large majority of States have now adopted some variant of the Uniform Trade Secrets Act (UTSA). The UTSA defines "trade secret" to mean information, including a formula, pattern, compilation, program device, method, technique, or process, that: (i) derives independent economic value, actual or potential, from not being generally known to, and not being readily ascertainable by proper means by, other persons who can obtain economic value from its disclosure or use, and (ii) is the subject of efforts that are reasonable under the circumstances to maintain its secrecy.

There is also a federal criminal statute providing for prosecution of theft of trade secrets, which contains a lengthier, but similar, definition. (See 18 USC 1839, et seq.) (There is also another federal criminal statute, 18 USC 1905, commonly referred to as the "Trade Secrets Act," but this statute does not contain a definition of "trade secret" and primarily addresses the confidentiality obligations of federal employees in performance of their official duties.)

Computer program trade secret claimants often employ protective measures such as licensing agreements containing confidentiality provisos, non-disclosure agreements for third-party code developers, distribution of the software only in executable form, and physical security for source code copies. The U.S. Copyright Office recognizes trade secret claims in computer programs and provides several registration options for the deposit of only a portion of the code. (See U.S. Copyright Office Circular No. 61, Registration for Computer Programs.)

3.0 Open Source Software--The Basics

3.1 What is open source software (OSS) and how does it differ from proprietary software?

OSS is software distributed under a license that provides broad rights to use, modify, and redistribute the original source code and, sometimes, any distributed modified versions also (i.e., derivative works). Many different OSS license types exist; each imposes certain obligations that have various legal implications. For example, many open source licenses do not require payment of royalties or place redistribution limits typically associated with proprietary software licenses. Most open source licenses automatically terminate if the licensee violates the license. Thus, once a licensee violates the terms of an open source license, the licensee has breached the contract and has infringed any copyright in the open source software.

Most open source licenses impose a share-alike clause that requires any redistribution of the original open source code and its derived works to be under the same or similar open terms as the original license. Open source licenses ensure that the rights granted cannot later be revoked and that derivative works must be provided in a form that facilitates modification. For software, this requires that the source code of the derivative work be made available with the software itself. Open source proponents claim that the share-alike clause fosters the free and open development and improvement of the software by a broad open source software development community and equal participation by all users, while opponents claim that share-alike creates undesirable licensing complications and restrictions.

The three major categories of OSS licenses are strongly protective, weakly protective, and permissive licenses.

Open source licenses are typically referred to as being “strongly protective” (aka “strongly copyleft”) or “weakly protective” (aka “weakly copyleft”), based on the extent to which open source provisions can be imposed on derived works. Strongly protective licenses require that any derivative works and any software that dynamically links to the licensed work be treated as a derived work and distributed under the same open source terms. Thus, strongly protective licenses are sometimes referred to as viral licenses because any software that links to the licensed work (even if that software is originally proprietary code) must be released under the same open source terms license.

Weak protective licenses require derivative works of the open source software to be redistributed under the same or similar open source terms, but specifically allow other software to link to the original open source software or derivative work without imposing the open source license requirements on the linked software. Only changes to the open source software itself become subject to the open source provisions, not the software that links to it. This allows software distributed under other licenses (including proprietary licenses) to be linked to the weakly protected software, and then be redistributed under its own terms.

Permissive licenses do not impose the share-alike clause. Permissive licenses place limited restrictions such as crediting the original author and stating that the original author makes no warranties on the work. Permissive licenses permit redistribution of the original and derivative works of the open source software under their own terms and conditions, which can differ for those of the original work. Therefore, permissive licenses offer many of the same freedoms as releasing a work to the public domain. Thus, derivative works can become proprietary and the original open source software can be incorporated into proprietary software.

3.2 What are some of the common open source licenses and their distribution terms?

The GNU General Public License (GPL) and the GNU Lesser Public License (LGPL) are the most popular and widely-used open source licenses. However, they are by no means the only open source licenses. Many other open source licenses are certified by the Open Source Initiative (OSI). The GPL licenses and other open source licenses certified by OSI must satisfy the Open Source Definition (http://www.opensource.org/docs/osd) that requires open source licenses to meet the following distribution terms:

(a) The software must be freely distributed;
(b) The software must be distributed in in source code as well as compiled form and a publicized means of obtaining the source code;
(c) The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software;
(d) Software built from modified source code must be distributed;
(e) The licensing terms must not discriminate against persons or groups of persons;
(f) The license must not restrict anyone from using the program in a specific field or endeavor;
(g) The licensing rights must apply to all to whom the program is redistributed;
(h) The licensing rights attached to the program must not depend on the program's being part of a specific program;
(i) The license must not place restrictions on other software distributed along with the program; and
(j) The license must be technology neutral.

The most widely used open source license is the GPL, which is an example of a strongly protective license. The Linux Operating System is distributed under the GNU GPL license. Examples of weakly protective licenses include the GNU (LGPL) and the Mozilla Public License. Popular, permissive OSS licenses include Apache licenses (all except v1.0), the BSD (Berkeley Software Distribution) License and the MIT (Massachusetts Institute of Technology) License. Some federal agencies, such as NASA, have created their own licenses. (See Section 4.3.) The Open Source Initiative (OSI) has certified the NASA Open Source Agreement (NOSA).

See www.opensource.org/licenses and http://www.gnu.org/licenses/license-list.html for copies of the licenses classified by the Open Source Initiative.

3.3 How does an open source licensing model impact the exclusive rights granted to copyright owners under Section 106 of the Copyright Act?

Section 106 of the Copyright Act provides copyright owners with five exclusive rights: (1) the right to reproduce; (2) the right to prepare derivative works based upon the original copyright work; (3) the right to distribute the copyrighted work; (4) the right to publicly perform; and (5) the right to publicly display the copyrighted work.

Copyright owners' licensing software using a proprietary license typically restrict a licensee's rights to modify, prepare derivative works, and distribute the software program. By allowing licensees to freely modify, create derivative works, and redistribute the original and modified source code, the copyright owner is providing a broad, but revocable, license of its exclusive rights. If the licensee violates the terms of the open source license, the open source licensor can terminate the license (actually, most open source licenses automatically terminate upon violation of their terms) or enforce the license under both copyright and contract.

The copyright owner(s) may provide the broad license of his/her rights for a variety of motivations (e.g., to encourage others to contribute improvements, to gain recognition/reputation, to gain a competitive advantage, for which the company may sell service/support, or simply to provide a service to the world).

4.0 Computer Software and the U.S. Government

4.1 Have U.S. Government agencies issued policy guidance regarding the use of open source software?

Yes. The Office of Management and Budget guidance requires all software acquisitions, whether open source or proprietary, to be as technology- and vendor-neutral as possible. Prior to purchasing or licensing software of any type, agencies must consider total cost of ownership, including lifecycle maintenance costs, risk-associated costs (including security and data privacy), and licensing limitations. (See Office of Management and Budget Memorandum M-03-14 (June 2, 2003).)[6]

Consistent, agency-specific guidance concerning use of open source software may also exist. In 2003, for example, the Department of Defense issued a policy memo on OSS use (see Open Source Software (OSS) in the Department of Defense (DoD) at http://iase.disa.mil/policy-guidance/oss-in-dodmemo.pdf). DoD recently published clarifying guidance at `http://www.defenselink.mil/cio-nii/sites/oss/index.shtml` and `http://cio-nii.defense.gov/sites/oss/Open_Source_Software_(OSS)_FAQ.htm`. In 2007, the U.S. Department of the Navy (DoN) issued a policy memo recognizing open source software as commercial-off-the-shelf (COTS) when it meets the definition of a commercial item pursuant to Section 403 of Title 41, and encouraging its use in IT acquisitions when it complies with Federal, DoD, and DoN policies. The U.S. Army issued a regulation, US Army Regulation 25-2 Information Assurance, paragraph 4-6.h, providing guidance on software security controls that specifically addresses open source software.

Civilian agencies such as NASA, which releases OSS under the NASA Open Source Agreement (NOSA) that has been certified as an open source license by the Open Source Initiative (OSI) (See NASA Open Source Software and NASA Procedural Requirements 2210.A-External Release of NASA Software) and the Federal Deposit Insurance Corporation (FDIC), have also issued guidance on using open source software. The Library of Congress recently expanded its policy on the distribution of government-created open source software to clarify that the Library will participate in individual open source projects as well as repositories. As part of its ongoing effort to develop an open platform for "WhiteHouse.gov," the White House has released some of the code that it developed for anyone to review, use, or modify. (See http://www.whitehouse.gov/tech.)

4.2 How do the FAR and DFARS address the use of open source software?

Both the FAR and DFARS treat open source software as "commercial software," which would be licensed to the government under the same terms as licensed to the general public. 41 USC § 403[7] defines a commercial item for purposes of both the FAR and DFARS as: (1) …any item, other than real property, that is of a type customarily used by the general public or by non-governmental entities for purposes other than governmental purposes, and (i) has been sold, leased, or licensed to the general public…or (ii) has been offered for sale, lease, or licensed to the general public.

DFARS 227.7202-1 provides that commercial computer software shall be acquired under licenses customarily provided to the public unless such licenses are inconsistent with federal procurement law or do not otherwise satisfy the government's needs. FAR section 12.212 provides that commercial software is acquired under licenses customarily provided to the public to the extent such licenses are consistent with federal law and otherwise satisfy the government's needs. (See FAR 1.105-2 (c) (3) (iii). Because open source software is licensed to the public and not developed exclusively for government use, it meets the definition of commercial software and would be licensed to the government under the same open source terms as to the general public.

In cases where commercial software, including open source software, is used as part of an application created by a contractor for government use, as noted in Section 4.4 of the "Frequently Asked Questions About Copyright," the contractor should seek the government's permission to use the open source software and provide a copy of the license to the agency for review by Intellectual Property (IP) counsel to ensure that the terms of use do not pose problems.

4.3 Are there issues unique to federal agencies in distributing open source software?

Yes, a civilian or military agency may distribute open source software, depending on its ownership interests or licensing rights in the software. For example, agencies typically use and may want to distribute to other users, within and outside the government, software created by: (1) its employees as part of their official duties; (2) a vendor, acting on the agency's behalf within the context of a procurement or other award instrument; and (3) a vendor who licenses its software using an open source licensing scheme.

An agency seeking to distribute software developed under any of these scenarios must first decide whether it owns or has acquired sufficient licensing rights to make the software available to other users under any licensing model, whether open source or otherwise. First, copyright protection is not available in the U.S. for software created by government employees as part of their official duties (See 17 USC § 105). However, copyright ownership is not a necessary prerequisite to adopting an open source strategy. An open source license is a contract, and even if copyright rights do not subsist in the code, it may still be provided as general property owned by the Government for which the Government has rights. For example, the NASA Open Source Agreement Version 1.3 (NOSA) demonstrates how Government users may adopt an open source licensing strategy for their own benefit. The NOSA makes the Government a third-party beneficiary to the Agreement as key to the open source licensing strategy.[2]

Second, an agency may distribute software created by a vendor to all users under an open source licensing scheme if it acquired sufficient rights from the vendor to do so in the software. For example, an "unlimited rights license" acquired under a DFARS procurement-type contract typically attaches to software generated by the vendor for the government when funded exclusively with government funds. Similarly, DoD agencies generally acquire a "government purposes rights license" in software created for the government by vendors under contracts awarded with mixed funding (i.e., government and private) and the agencies may be able to distribute the software to other agencies under an open source-type model, as long as it complies with any restrictions attached to the software under the original contract. (See DFARS 252.227-7014.) Agencies wishing to disseminate open source software or participate in open source development may wish to include explicit open source requirements in their contracts and grants for software development.

Third, federal agencies may also wish to distribute applications—created by their employees or vendors, acting on their behalf—which include proprietary open source software components. Prior to choosing this option, agencies must carefully evaluate their own licensing rights under the original contract or other award instrument, as well as the requirements of the particular open source licensing scheme under consideration for use.

Given the complex and often confusing issues posed by software acquisitions of all types and the use and licensing of proprietary and open source software, program managers should always consult their agency's acquisition and IP counsel and contracting officers prior to sharing or disclosing software to any government or other user.

Finally, where it has sufficient ownership or licensing rights to do so, a federal agency may wish to coordinate use of open source software licenses and the distribution of the software through already existing open source portals such as SourceForge and the National Institute of Health's Web site. DoD agencies also have the option of distributing software through "Forge.mil," a web site enabling the collaborative development and distribution of open source software and DoD community source software. NASA distributes its open source

[2] See J.T. WESTERMEIER, NEW STRATEGIES AND RISK MANAGEMENT PRACTICES FOR OPEN SOURCE SOFTWARE, American Bar Association Research & Development and Intellectual Property Committee, Luncheon Presentation, June 23, 2005.

software on NASA.gov web sites (see http://opensource.arc.nasa.gov) and through a government-appropriate agreement with SourceForge.

4.4 Is the U.S. Government allowed to use open source software on government computer networks?

Generally yes, when it best fits the needs and mission requirements of the agency involved and it meets applicable information assurance or other security standards for the particular computer network on which it is to be used. Many types of OSS are widely used in the U.S. Government, as one study commissioned by DoD showed. (See Mitre/DISA, *Use of Free and Open-Source Software (FOSS) in the U.S. Department of Defense*, Version 1.2.04, January 2, 2003.[8]) The U.S. Government has also started several OSS projects (e.g., OpenVisa, Expect, EZRO, and DISA/OSSI Open Source CRADA) and contributed funding to improvements for others (e.g., DNSSEC support for Bind, FIPS cryptographic evaluation for OpenSSL, BSD TCP/IP suite implementation, and SELinux).

4.5 Are there any special issues involved in government use of OSS?

The same issues should be considered for OSS as for proprietary software. These issues can be roughly categorized into three main groups, and each of them should be fully assessed prior to any software purchase or use, whether OSS or proprietary: (1) copyright license and contractual terms, (2) acquisition life cycle, and (3) security.

4.6 What are the OSS copyright licensing and contractual considerations of greatest concern to the government?

All copyright licensing and contractual terms should be carefully assessed to ensure that the government can legally agree to them and fully understands the risks involved in accepting them, particularly provisions addressing warranties, indemnifications, distribution and redistribution of code, patent licenses, and applicable law and dispute resolution mechanisms.

4.7 What are the main OSS acquisition life cycle considerations?

All aspects of the acquisition life cycle should also be fully analyzed, including determining the “total cost of ownership” of the software. Low initial purchase price is often a very attractive feature of OSS, but many other costs should also be carefully considered. Characteristics of the software itself should be assessed, including its integrity, reliability, scalability, and flexibility. Transition costs include software configuration and installation, file backups, data file format conversions, and new hardware installation. Training costs include training for help desks and administrators as well as users. Maintenance costs include onsite maintenance and code tracking, as well as patching, adding new functional requirements, etc. Also to consider is the current market share and growth path of the particular OSS within its functional category during projected government use. If continuous public maintenance and upgrade of the particular OSS under consideration is questionable, additional government resources may need to be allocated to replace dwindling public resources.

4.8 What are the main OSS security assessment considerations?

Security assessment for any software, whether OSS or proprietary, is extremely complex and requires special technical expertise for proper assessment. Technical and information technology security personnel should lead this assessment, and acquisition and legal counsel should work closely with them before any decisions are implemented. Among other things, security from the government perspective involves compliance with the National Security Telecommunications and Information Systems Security Policy Number 11, and the Federal Information Security Management Act (FISMA 44 USC 3541 et seq.) as well as meeting applicable agency configuration and information assurance guidelines.

Government technical assessors should determine, insofar as possible, who has written the code and whether the developers possess appropriate security expertise (and such expertise is reflected in the actual code product being distributed), and whether there is prompt identification and repair of defects for the software under consideration. Under appropriate security and software licensing conditions, some OSS can be used in classified systems; as with all commercial software, however, it is essential that technical and legal personnel consult closely together beforehand so the ramifications of use are well understood in advance.

4.9 Are there any particular advantages to government use of OSS as compared to using proprietary software?

Regardless of whether a particular piece of software is OSS or proprietary, the government should carefully assess its advantages and disadvantages—within the specific context of its intended use on an identified computer network—by examining each of the three issue categories identified in FAQ 4.5.

OSS, however, may have certain inherent advantages for the government that should also be factored into any acquisition or use determination. First, well-established OSS products may be inherently more reliable and more secure than proprietary products available for similar use. This is because OSS is often developed via a public, community-based approach, so it is also continuously subjected to very broad peer review and user assessment. Since the reviewing/user community is generally much wider for OSS than for comparable proprietary software, defects and vulnerabilities in the software may be identified earlier and fixed sooner than is possible with most proprietary products.

Other advantages to the government that are inherent in OSS include: (1) access to source code, allowing government modification to fit particularized needs, rapid response when needs change or new threats are identified, and in-depth security review and audit; (2) reduced dependence on a particular vendor, developer, or product, since OSS can be operated and maintained by multiple entities and many OSS products are easily interoperable with others; (3) potential cost savings resulting from no "per seat" or "per copy" or field-of-use licensing fees and shared (community-based) maintenance/support costs; and (4) applicability of the statutory preference for acquisition of commercial items over noncommercial items. Foreign governments may view reduced dependence on specifically identifiable foreign suppliers (i.e., U.S. or European software sources) as an additional attractive feature of OSS. Some of the same advantages, such as the statutory commercial item preference, may also exist (or at least be negotiable, even if at increased price) for proprietary software.

4.10 Are there any particular disadvantages to government use of OSS as compared to using proprietary software?

Before acquiring or using any software, the government should ensure that the terms of the applicable license are compatible with the government's intended use, users, and identified computer network on which the software will be run. Although open source licenses, particularly the GPL, are popular and widely used by the open source community, a mandatory code distribution requirement may not be appropriate for all government uses. While many government lawyers believe that no public distribution or propagation of modified code occurs so long as the code is used only within the federal government (including federal support contractors operating under nondisclosure agreements), this view has not been reviewed by any court. Particularly where software potentially subject to a mandatory distribution licensing provision will be modified for use on, or linked to, classified or other secure computer systems, or where such software is export-controlled, government managers should include these considerations as part of their risk assessment. Mandatory distribution requirements may also adversely implicate third-party proprietary code or information, depending on computer system and software architecture. Thus, government acquisition planners should ensure they consult closely with appropriate technical and legal advisors and fully understand the effects of such licensing provisions in advance.

Some OSS licenses also contain patent licenses (usually intended to prevent patent infringement that would otherwise necessarily occur through using the OSS) and prohibitions against engaging in patent litigation related to the OSS. The ramifications of these types of provisions must also be well understood beforehand.

Even patented code could effectively lose its protection if mingled with OSS. Further, OSS publication requirements may preclude the later integration of OSS based code with other proprietary software development. This can significantly delay the development of integrated tools, especially with contractor developed proprietary software.

Although many OSS products exist and many entities are available that provide OSS support and maintenance services, some OSS may not have a large enough supporting developer/user community to ensure that sufficient public maintenance and support of the software will remain available during the government's foreseeable use period. In such cases, the government should assess (prior to acquisition or use) whether it can allocate sufficient labor resources for these purposes from its own employee or contractor communities and can justify any additional costs this might entail.

Note also that special licensing terms deviating from the applicable OSS license generally cannot be negotiated by the government, since one or more of the relevant copyright holders may not be available to consent to altered licensing terms. An exception to this general rule is some or all of the OSS licensed by the Free Software Foundation (FSF); since the same organization is the sole copyright holder of all FSF software, its representatives may be able to negotiate special licenses in appropriate cases.

4.11 Is the U.S. Government required to give preference to proprietary software over OSS, or vice versa, in its acquisitions?

The U.S. Government is required to give preference to commercial items over noncommercial items in its acquisitions, in accordance with 41 USC 403. There is no requirement to give preference to either proprietary or open source software over the other. Nearly all OSS is commercial software, as are many proprietary software products.

5.0 Case Law on OSS Licensing: U.S. and International

5.1 Is there any U.S. federal case law addressing OSS licensing?

The Court of Appeals for the Federal Circuit (CAFC) recently found an open source license to constitute an enforceable copyright license—*Jacobsen v. Matthew Katzer and Kamind Associates, Inc. (d/b/a KAM Industries)*, 535 F.3d 1373, C.A. Fed. (Cal.), 2008. Applying the interpretive law of the 9th Circuit, the Court found the terms of the Artistic License were enforceable copyright conditions, potentially allowing for injunctive relief against infringement, rather than merely contractual covenants remediable only by monetary damages. Citing an 11th Circuit opinion from 2001 *(Planetary Motion, Inc. v. Techsplosion, Inc.*,261 F.3d 1188*)*, the Court explained that substantial economic benefits can accrue to copyright holders under open source licensing, despite the fact that traditional copyright royalties are not generated. For example, the Court noted, economic benefits of open source licensing include allowing program creators to generate program market share by providing some components without charge, to increase professional reputation through open source project incubation, and to obtain rapid, free, and expert product improvements. The Court found that the clear restrictions contained in the license, including the requirement to retain reference to the original source files in modified or distributed code, were necessary to accomplish the objectives of the open source collaboration, and might well be rendered meaningless without the ability to enforce them through injunctive relief.

Although there have been a number of other federal lawsuits filed alleging infringement of open source licenses, most of them have been settled prior to judgment, and, apart from *Jacobsen*, none have yet resulted in substantive judicial interpretation of any open source license.

In particular, the enforceability of the GNU GPL has not yet been addressed by a U.S. Federal Court, although it is at issue in a complaint filed in the Southern District of New York and has been contested in a number of settled U.S. cases. The Free Software Foundation filed a copyright infringement lawsuit against Cisco Systems, Inc., alleging violation of three GNU open source licenses, including the GPL, for OSS used in hardware devices sold commercially to the public. (*Free Software Foundation, Inc., v. Cisco Systems, Inc.*, 08-cv-10764, S.D.N.Y., *complaint filed December 11, 2008.)*Settled cases include the “BusyBox” litigation in which the Software Freedom Law Center (SFLC) sought to enforce the GNU GPL against various companies that had included GPL-licensed Busybox software into products offered for commercial sale, without releasing modified code back to the public.

(*Andersen v. Monsoon Multimedia, Inc.*, 07-cv-08205-JES, S.D.N.Y., *complaint filed September 19, 2007*; *Andersen v. High Gain Antennas, LLC*, 07-cv-10456, S.D.N.Y., *complaint filed November 19, 2007*; *Andersen v. Xterasys Corporation*, 07-cv-10455, S.D.N.Y., *complaint filed November 19, 2007*; *Andersen v. Verizon Communications Inc.*, 07-cv-11070, S.D.N.Y., *complaint filed December 6, 2007*; *Andersen v. Bell Microproducts, Inc.*, 08-cv-5270, S.D.N.Y., *complaint filed June 9, 2008*; *Andersen v. Super Micro Computer, Inc.*, 08-cv-5269, S.D.N.Y., *complaint filed June 9, 2008*.*)* Other settled cases concerning OSS are *Progress Software Corp. v. MySQL AB*, 195 F. Supp. 2d 328, 329 (D. Mass.*)* and *MySQL v NuSphere*.

In one well-publicized case, the SCO Group sued Novell, claiming IBM had infringed SCO's Unix-related copyrights by allowing copyrighted code to be released into the public domain in support of a Linux open source project, but the court held that since SCO did not own the copyrights, it lacked standing to sue for copyright infringement. *(SCO Group v. Novell, Inc.*, 377 F. Supp. 2dd 1145, N.D. Utah 2007.)

In another case, a software developer unsuccessfully alleged that IBM, Red Hat, and Novell used the GPL to fix software prices (at $0) in a pooling and cross-licensing scheme illegal under antitrust law that prevented the plaintiff from competitively marketing his own software. (*Wallace v. IBM et al*, 467 F.3d 1104, 2006-*2 Trade Cases* P 75, 480, 80 U.S.P.Q. 2d 1956.*)*

5.2 Is there any foreign case law addressing OSS licensing?

Two German courts have enforced the GPL license terms against several foreign vendors that have not made modified source code available after incorporating OSS in their products offered for commercial sale. (*Welte v. Sitecom, Final Judgment of the District Court of Munich I, issued 19 May 2004* – Docket No. 21 O 6123/04; *Welte v. D-Link Germany GmbH, District Court (Landgericht) of Frankfurt Am Main*, Docket No. 2-6 0 224/06; *Welte v. Skype Technologies S.A., District Court (Landgericht) of Munich I*, Docket No. 7 O 5245/07.*)* A variant of the antitrust argument used unsuccessfully in the U.S. in *Wallace v. IBM* was also put forth initially on appeal of *Skype* in Germany, but the appeal was subsequently withdrawn, so the German judicial view of this argument remains unknown.

An assignation was filed before a French court (le Tribunal de Grande Instance de Paris) in late November 2008, against the French telecom company, Iliad, on behalf of the Free Software Foundation, Mr. Harald Welte, and others. The complaint alleges that Iliad incorporated GPL-licensed software into its Freebox products, which were then distributed to the public without making modified source code available. (No citation is provided for this case because only an unauthenticated copy of the assignation document is available via Internet search engine sources.)

6.0 Advising Government Clients on Use of Open Source Software: Tips and Best Practices

6.1 When should I discuss OSS with my clients?

OSS should be discussed in detail whenever your client is considering: (1) the incorporation of Open Source Software into software developed by or for the agency; or (2) the original development by or for the agency of software intended for open source release.

OSS should be discussed more generally whenever your client is considering: (1) acquiring or using (or modifying) a new computer program or computer-based technical data; (2) modifying current computer programs or technical data; or (3) acquiring or using or designing computer networks or hardware that contains embedded software that may ultimately be linked to non-OSS software or technical data.

If a proposed release of software developed by or for the agency includes the release of Open Source Software, care must be taken to ensure that the pertinent license for such Open Source Software is acceptable. Legal or IP Counsel should review the Open Source Software license and assess any special risks that may be involved, and confirm that the agency has obtained clear rights from any third party rights owners (such as through an assignment or license) to make the Open Source Release. For example, at least one widely used Open Source Software license requires that all software distributed with that Open Source Software be distributed under the same license terms.

6.2. What should I advise them they need to understand before making a programmatic or acquisition decision?

At a minimum, they need to understand exactly what types of OSS are proposed for delivery/use or work performance, how each type of OSS will be used (during development, as well as in any delivered software), where each type of OSS will be used (during development, in delivered code, and for use on which computer networks or systems), the terms of each OSS license, and whether the OSS has been or will be modified (during development or after delivery to the government). Noncompliance with OSS license terms could result in litigation or loss of use of the software.

6.3 How can my agency identify the OSS it may already be using if the software doesn't already carry identifying markings that reflect it is OSS?

Other than asking the software developer for the code provenance, or by doing an extensive manual examination of the source code (which would likely not yield conclusive results anyway), experts differ on whether it is presently feasible to determine reliably whether OSS code may be present in computer software already delivered to, or developed by the government, unless the developer has previously labeled all or part of the existing code as an identifiable version of OSS. Several commercial companies offer examination services for this purpose.

6.4 Should the government care whether its contractors identify OSS they may be embedding in or linked to software delivered to the government under procurement contracts, cooperative agreements, and other instruments?

Yes. Some OSS licensing provisions can directly affect whether and when the government may be obliged to provide source code to the public. Agency procurement officials should consider including notices in RFPs and contracts for software development regarding: (1) whether OSS should or may be used in software developed for or delivered to the government; (2) requirements that the contractor identify any OSS that may be incorporated into software developed for or delivered to the government; and (3) requirements that the contractor provide copies of all OSS licenses.

6.5 Should the government care whether its contractors use OSS products to develop software for the government, but don't embed any OSS in any of the delivered code?

Yes. Some OSS licensing provisions can directly affect whether and when the government may be obliged to provide source code to the public, even when the OSS is not embedded in the delivered code, but is only used in its development.

6.6 Are government contractors obligated to tell the government about OSS they are using in code that is licensed or delivered to the government under government contracts, or that is used to develop such code?

Probably. But even if government contractors do not volunteer this information, government agencies should always request all contractors to identify fully to the government their intended uses and planned modifications of OSS that are expected during their performance of a government contract, whether or not the OSS used or generated will be delivered to the government. As with all other commercial software code, the government should ask the contractor to clearly identify in writing all of the following items: (1) each type of OSS used/modified and its title and version number; (2) each concomitant OSS license and, if applicable, license version number; (3) identity of the asserting party (contractor/sub/awardee); (4) whether any of the OSS has been or will be modified, and, if so, by whom; and (5) whether such modification occurred or will occur by incorporating it into any third party software (if so, identify). While this information should be provided for all commercial software (including OSS) prior to execution of any contract or award instrument, the parties should also agree that full written identification of all commercial code used, including OSS, must be provided by the contractor and approved by the government before incorporating it into any deliverable, using it to develop a deliverable, or using it to modify or link to preexisting code used in any government computer program or system.

6.7 Is software considered an agency record covered by the Freedom of Information Act (FOIA)?

Generally, no, although in some instances, computer software may have to be treated as an agency record and disclosed under the FOIA. These situations are rare, and should be reviewed on a case-by-case basis to determine whether the data on the software requires that it be treated as an agency record subject to FOIA.

Releasing open source software under FOIA may be problematic if it contains sensitive or critical data. Thus, in addition to the security concerns discussed in FAQ 4.0, agencies should also consider whether software that they use or distribute under an open source licensing arrangement would be considered an agency record.

Specific examples of computer software treated as an agency record that may be processed under the FOIA include software that: (1) contains an embedded database that cannot be extracted and is itself releasable under the FOIA; (2) reveals information about agency policy, functions, decision making, or procedures; or (3) is so related to such an accompanying database that the database itself would be unintelligible or unusable without the software. In the first scenario, both the data and the software must be reviewed for release or denial under the FOIA.

See relevant regulations and cases addressing this issue, such as *32 C.F.R. 518.10 (c), Gilmore v. Department of Energy*, 4 F. Supp 2d 912 (N.D. CA 1998), and *DeLorme Pub. Co. v. NOAA*, 907 F. Supp. 10 (D. Me 1995*).*

6.8 Does the licensing of open source software raise export control issues?

Yes, if open source software that requires an export license under the Export Administration Act (EAA) (50 U.S.C. App. §§ 2401 et seq.) or is on the U.S. Munitions List (22 CFR § 121.1) is released to a foreign person via an open source license to an individual, academic institution, company, government, or nongovernmental organization, violations of the EAA, the Arms Export Control Act (AECA) (22 U.S.C. §§ 2778-2780) and the International Traffic in Arms Regulations (ITAR) (22 C.F.R. Parts 120-130) may occur. Generally, the EAA regulates the export of items that are neither military nor nuclear in nature. The AECA and ITAR regulate defense articles and services and related technical data identified on the U.S. Munitions List. Additional information on licensing requirements, policies and procedures related to export controls may be found on the U.S. Department of Commerce's and Department of State's web sites at http://www.bis.doc.gov and http://www.pmddtc.state.gov. Other U.S. Agencies also regulate the export of certain goods and services. Consult agency counsel with expertise in export control matters for guidance on licensing open source software that may raise export control issues.

7.0 Legislation and Other Resources

The bibliography lists some recent publications, articles, brochures, web sites, and listservs related to computer software copyright that provide information and a variety of perspectives on this issue. This list is not intended to be exhaustive nor does the U.S. Copyright Office necessarily endorse the works listed. Cited web site addresses were all correct and active as of October 2009.

7.1 Web Sites

These web sites contain references, links, and additional informational resources and opinions on copyright as it relates to computer software. Many of these sites have links to other informational materials with related OSS themes.

AF Software Technology Support Center
www.stsc.hill.af.mil

The Data Analysis Center for Software (DACS)
***DoD/DTIC-managed Information Analysis Center**
www.thedacs.com/databases/url/key/4878

Forge.mil
www.disa.mil/forge

Free Software Foundation
www.fsf.org

NASA
http://opensource.arc.nasa.gov/

Open Source Initiative (OSI)
www.opensource.org

Open Source Software Institute (OSSI)
http://www.oss-institute.org

Source Forge
sourceforge.net

U.S. Copyright Office
www.copyright.gov

U.S. Department of the Navy
www.navy.mil/swf/index.asp

United States Patent and Trademark Office
www.uspto.gov

7.2 Other Sources (Publications, Reports etc.)

The American Bar Association
www.abanet.org/intelprop/opensource.html

Mitre/DISA, Use of Free and Open-Source Software (FOSS) in the U.S. Department of Defense, Version 1.2.04, January 2, 2003
www.isd.mel.nist.gov/projects/rtlinux/dod-mitre-report.pdf

7.3 References

[1] See http://www.bitlaw.com/copyright/scope.html.

[2] Federal Acquisition Regulation: http://www.arnet.gov/far.

[3] 17 USC Section 101, http://www.copyright.gov/title17/92chap1.html#101; H. Rep. No. 94-1733, 94th Cong., 2d Sess. (Sept. 29, 1976); Computer Software Copyright Act of 1980, Act of Dec. 12, 1980, Pub L. 96-517, Sec. 10, 94 Stat. 3015; Atari Games Corp. v. Oman, 888 F. 2d 878 (D.C. Cir. 1989); Whelan Associates, Inc. v. Jaslow Dental Library, 797 F. 2d 1222 (3d Cir. 1986); Nimmer on Copyrights, Section 2.04[C].

[4] See 17 USC 102 (b).

[5] "The Rise of the Information Processing Patent," Ben Klemens, Journal of Science and Technology Law, http://www.bu.edu/law/central/jd/organizations/journals/scitech/volume141/documents/Klemens.pdf.

[6] See http://www.whitehouse.gov/omb/memoranda_m03-14/ .

[7] See http://www.law.cornell.edu/uscode/+++html/uscode41/usc_sec_41_00000403----000-.html .

[8] See www.isd.mel.nist.gov/projects/rtlinux/dod-mitre-report.pdf.