NHSE Distribution of HPCC Software: Legal Issues and Technological Options

Alice Armintor, Shirley Browne, Paul McMahan, and Danny Powell

September 28, 1997

1.0 Introduction

The goal of the NHSE is to distribute HPCC software to as broad a U.S. audience as possible, so as to maximize the return on HPCC agency investment in developing this software by promoting further research and increasing U.S. market competitiveness. Where possible, HPCC software should also be made available to foreign researchers who are collaborating with U.S. scientists. However, the distribution mechanisms must provide reasonable assurances that intellectual property rights are protected and that export regulations are abided by.

The approach of the NHSE is to encourage the development of HPCC repositories that are maintained by experts and that provide access to software and documents within their specific domains. Examples of such repositories are the Netlib mathematical software repository, the Parallel Tools Library, and the Chemistry Software and Information Repository. The NHSE will then link together these discipline-oriented repositories and provide access for HPCC users in a convenient way. The NHSE has developed a Repository in a Box (RIB) toolkit that includes tools for carrying out basic repository setup and maintenance tasks.

A repository can provide access just to software catalog records containing pointers to the actual software that is available elsewhere (i.e., be a ``virtual repository''), or it can distribute the actual software itself. Similarly, software may be shared between interoperating repositories at two levels: 1) at the level of catalog information that describes the software, 2) at the level of actual software files. Advantages of the direct distribution approach may be provision of faster and more reliable service to users, as well as a single point of contact for administrative procedures such as license agreements. Problems with the direct approach include liability for enforcing legal restrictions and proper crediting of download and usage statistics to the originating site.

The NHSE is currently developing a set of tools to assist organizations in their software distribution efforts. Before development began, the NHSE composed a list of requirements. The following were the guidelines by which the toolbox is being developed:

- prevent unauthorized users from downloading restricted software

- collect and maintain information on the users of the software and how they use it, and provide an administrative interface to view this information
- be easy to set up, maintain, and administer
- be easy to use
- be compatible with the existing RIB architecture
- run initially in most common forms of UNIX, with possible port to Windows NT
- be flexible, to run with any common Web server or database package
- offer good performance
- be inexpensive
- be customizable, so that each site can incorporate its own identity

Each technology transfer organization will have unique requirements and conditions for distribution, but some intellectual property issues will be universal. This report will begin with the basic intellectual property issues of software. When developing a software distribution system, an evaluation of the types of software that will be included is the best starting point. Section 2 gives an overview of those legal issues to consider in an intellectual property assessment of software. Section 3 discusses different options for distributing software on-line. Section 4 discusses liability issues with section 5 as a conclusion.

2.0 IP Issues to Consider in Software Assessment

A generic software rights management process that might apply to any software is described in the following paragraphs. An example of such a process is given in the NASA draft software release policy dated June 3, 1996 [1].

A request to release a piece of software is typically made by the author to the organization for which he/she works. The author fills out a form that captures the information required to resolve rights issues and detect possible infringement of patents, trademarks, or copyrights. Then an intellectual property assessment is carried out by the organization's intellectual property counsel. This assessment determines the following:

- whether any rights infringements have been made by the author
- whether the organization should seek to protect intellectual property rights embodied in the asset (e.g., by filing a patent application)
- whether the asset is eligible for foreign release and, if so, whether or not an export license will be required
- whether there are any special considerations required by the funding agency (if funding was provided by an outside source)
- whether the software has financial value above its academic value

It is interesting to note that carrying out this assessment, particularly if it involves withholding publication until patent claims can be made, may disqualify the technology from the “basic research” category, thus increasing its export restrictions.

After the intellectual property assessment has been carried out, the software is labeled with a copyright legend and, if applicable, patent and/or trademark notices. Finally, the software is categorized. Organizations typically have several release categories such as public domain, freeware, shareware, free for non-commercial use, beta-testing, and licensed. For categories other than public domain, a license is usually required to obtain the software.

2.1 Copyright

Authors of original works fixed in any tangible medium of expression can obtain limited protection for their intellectual property through the copyright laws of the United States. Copyright protection is in effect as soon as the work is fixed in a tangible medium of expression, but the copyright owner cannot bring an infringement action until the copyright has been registered with the U.S. Copyright Office. The copyright usually belongs to either the author, authors, or the author(s)’s assignee. The exception is a “work for hire”, which can either be a work authored by an employee within the scope of his or her employment or a commissioned work. The latter requires a written agreement declaring that the task is a work for hire. In the case of a work for hire, the employer owns the copyright. If owned by the author, copyright protection remains in effect for the author's lifetime plus fifty years. For a work for hire, the duration of the copyright is 75 years from publication or 100 years from creation, whichever is earlier.

A copyright owner may assign or license the rights to copyrighted works. The owner of a copy of a copyrighted work may loan, sell, or lease the copy without restriction. The owner of a copy of a computer program can install and execute the program on a single computer. Additionally, the owner can make a copy of the program for archival purposes. The copyright owner may assign users the right to make and distribute copies of the program, but unless the program has been explicitly placed in the public domain, such permission may subsequently be withdrawn.

Works of the U.S. government are public domain and cannot be copyrighted. However, although the U.S. government cannot get copyright for its own works, it can have an existing copyright assigned to it. For example, an independent contractor working for the government owns the copyright to the work it produces but may assign the copyright back to the government.[2]

2.2 Patents

A patent protects an idea and gives an inventor the right to exclude others from making, using, or selling his or her invention for a period of twenty years after the filing date. The governing law for patents in the United States is Title 35 of the United States Code, or 35 USC. In order to be patentable, the invention must fall into one of the following five statutory classes of things that are patentable: 1) processes, 2) machines, 3) manufactures, 4) compositions of matter, and 5) new uses of any of the preceding. Most software patents fall under the category of processes. In addition, to be patentable, an invention must be useful, novel, and nonobvious.

Patents are awarded by the U.S. Patent and Trademark Office. Under U.S. patent law, a patent will not be granted to an applicant unless the application is filed less than one year from the date that the invention was first sold or offered for sale within the United States. The patent will also be denied unless the application is filed within one year of the date the invention was described in a printed publication anywhere in the world. Under 35 USC section 287, a patent owner is required to mark goods embodying the invention with the patent number.

2.3 Trademarks

A trademark is any word, slogan, or symbol which is used in trade with goods and services to indicate their source of origin and to distinguish them from the goods and services of others . Trademark rights may be used to prevent others from using a confusingly similar mark, but not to prevent others from offering the same goods and services under a non-confusing mark. In the United States, trademark rights are created when use of a trademark begins. However, these rights are often limited. Greater rights are available by registering the trademark with the state or federal government. Trademarks used in interstate or foreign commerce may be registered in the U.S. Patent and Trademark Office (PTO). A trademark application can be submitted to the PTO based upon actual use in commerce or a bona fide intent to use the mark in commerce. A trademark Examiner is assigned to each application and considers the registrability of the mark in light of the statutory guidelines.

2.4 Export Restrictions

Export Restrictions are complex and dynamic, requiring the utmost consideration. For simplicity, the first step in understanding export restrictions is understanding the definition of an export. Simply an export is something being shipped, or in the case of the Internet, being downloaded, outside of the United States or its territories. A release to a foreign national who is not protected under the Immigration and Naturalization Act or not admitted for permanent residence in the United States is also considered an export. Such a release is seen as a release to the home country of the foreign national.

Exports are regulated by the Bureau of Export Administration (BXA) with the intent to serve short supply interest of the United States, foreign policy, national security, and occasionally international obligations. Other government agencies such as the Department of State and the Nuclear Regulatory Commission also regulate exports. Questions of who regulates a good of military nature should be directed to Office of Defense Trade Controls.[3]

The BXA of the Department of Commerce issues the Export Administration Regulations (EAR). In March 1996, the term "general license" was dropped, and the term "exception" is now used to indicate a technology exemption from a BXA issued license. To determine whether a technology needs a license, the Commerce Control List (CCL) is provided in the regulations. Technologies are grouped according to Export Control Classification Number (ECCN). Each ECCN listing includes a list of Country Groups which indicates to which countries a license is required and the reason for restricting exportation. The EAR also specifies a list of foreign countries to which the U.S. government prohibits the export of any non-humanitarian good. Currently the list includes Cuba, North Korea, Iraq, Croatia, and Bosnia-Herzegovina. Other countries like China, Syria, and Iran are not currently prohibited, but exports are not generally granted.

When dealing with potentially restricted software, legal counsel or the advice of BXA should be sought to insure proper handling. Export assistance offices are located in most major cities, and BXA has an export assistance hotline for any export questions.

2.5 Categorization

A rights assessment of a piece of software determines how it fits in a software distribution scheme. Rarely is the distribution mechanism designed specifically for each individual piece of software. A general scheme of how each category is distributed simplifies the installation of new additions to the software catalog by stipulating what security measures should be taken and what IP terms should be included in the license.

Value and restriction generally dictate the classifications. The first decision should be whether the software is public domain. Caution should be taken when making this decision since once the software has been declared public domain, it can not be retracted. If it is copyrighted, is it free? For what purposes or users can the software be licensed? Is it free for research purposes? Is the access restricted either by export regulations or by the organization's restrictions? Is a commercial developer the targeted user? These basic questions are the foundation for a classification scheme. Software to be distributed via the NHSE often falls into one or more of several categories:

- Public Domain software technologies and documentation,
- Software requiring limited licensing restrictions, but still freely distributable within those restrictions,

- Commercial codes or other codes requiring fees, royalties or other types of payments. These codes typically require licenses stipulating stronger restrictions in copying, use, and redistribution of the software,
- Software technologies or documentation that fall under the control of Federal Export Regulations.

Once the scheme has been drafted, distribution tools can be assigned to categories according to the authorization and authentication needs.

3.0 Options for Distributing Restricted NHSE Software

3.1 Licensing Options

Mass market software instigated the use of shrink-wrap to replace the necessity of having an end-user sign a license. This practice has been challenged in several court cases. However in the most recent decision, Pro CD v. Zeidenberg, the Seventh Circuit Court ruled that shrink-wraps are enforceable contracts, overturning a previous ruling. [4]

Barring a ruling from the Supreme Court, the Seventh Circuit Court ruling validates the use of shrink-wrap and electronic shrink-wrap, whose validity has not been explicitly questioned in court. Electronic shrink-wrap, whose new buzzword name is “clickwrap”, provides an extra comfort for the courts by creating an interaction with the user. The action of the user accepting the terms of the license by “pressing” a button and the record of that action provide an indication of commitment on the part of the user. A record of this transaction can be strong evidence against an end user’s misuse of downloaded software.[5]

To further computerize contracts, 39 states have pending or passed legislation validating electronic and/or digital signatures. The language of some legislation confuses electronic and digital signatures by not making the distinction that electronic signature is simply keystrokes with the intent to simulate a signature while digital signatures consist of encrypted keys. Nonetheless this legislative activity increases the legal binding of on-line contracts and allows a wider range of uses than clickwraps for software licenses.

3.2 Authorization and Authentication

Authorization means making a judgment as to whether or not a user should be permitted access. Authentication means verifying that the party attempting access is who he/she claims to be. The initial authorization and authentication process may require manual intervention to check the user's credentials and determine access privileges. Pre-authorization for one or more software packages or entire classes of software may be

carried out once and for all, with subsequent access requiring only authentication which may be done automatically. Most methods of authorization and authentication have not been tested in court. However new legislation recognizing encryption methods such as digital signatures provides substantial legal backing.

3.2.1 Digital Signatures

To attach a digital signature to a document, an author uses a secure one-way hash function to compute a digest, or ``digital fingerprint'' of the document. A secure one-way hash function ensures that it is impossible to create a second, different document with the same fingerprint, or to derive the original document from the fingerprint. The author then encrypts the fingerprint with his private key. The encrypted fingerprint is the digital signature for the document. Given a digitally signed document and the author's public key, one can verify both that the document was actually signed by the author and that the document has not been altered since it was signed. In the case of a contract signed by two parties, both may attach their digital signatures to the same document so that it may later be verified that both agreed to it.

Both the legal and business communities are beginning to recognize digital signatures to be the electronic equivalent to pen and ink signatures for contracts. The National Institute of Standards and Technology has proposed the Digital Signature Standard, or DSS, as an algorithm standard for digital signatures. The NIST standard for generating keys and signatures is compatible with standards in the state legislation. A Government Accounting Office decision requested by NIST opined that digital signatures will meet legal requirements for valid contracts under federal law. The Department of Defense has notified NIST that DSS can be used by the Defense Department to sign unclassified data and, in some cases, classified data. The American Bar Association is drafting guidelines for their use, and twenty one states have passed or proposed legislation stating the equivalence of digital signature to “written” signature. While the plaintiff is responsible for authenticating the signature, most statues provide that the digitally signed message themselves are self-authenticating. The legal presumption is that the name associated with the signature is the name of the person who signed the document. Thus the burden of proof to deny the signature is on the defendant.[6]

3.2.2 Pretty Good Privacy (PGP)

PGP is a public key system for encrypting electronic mail using the RSA public key cypher [7]. It encrypts the message using the Swiss IDEA cypher with a randomly generated key. It then encrypts the key using the recipient's public key. When the recipient receives the message, PGP uses his private RSA key to decrypt the IDEA key and then uses that IDEA key to decrypt the message.

PGP can also be used to sign messages. It does so by first computing a ``hash" of the message using the hash function MD5. It then encrypts this hash output (128 bits or 16 bytes) with the secret RSA key of the sender. Any recipient can calculate that same hash output of the received message and use the sender's public key to decrypt the signature. If the output to this decryption agrees with the recipient's calculated hash output, then the recipient knows both that the sender actually sent that message and that not a single bit of that message has been changed.

The PGP software will generate a public /private key pair for the user. Since it is the user that generates the key pair, one of the problems is that of trust. How do you know that the public key claimed to be from the intended recipient is not that of an enemy instead pretending to be the recipient? PGP uses the idea of a ``Web of Trust". It advises anyone generating a public key to have it signed by a number of other trustworthy people who are in effect affirming that the key belongs to the one it claims to belong to. Thus the hope is that at least one of the signers is someone known to the sender as a trustworthy person, or that he is someone vouched for by a known trustworthy person. MIT maintains a public key server that can be accessed via the Web or by email to retrieve the
public key of a registered party.

An informal Web of Trust would not be sufficient for reliably identifying NHSE users. The following possibilities would improve this situation:

- Have HPCC agencies and organizations sign keys and have users submit them to an existing PGP public key server
- Have the NHSE run a PGP public key server
- Use the Four11 commercial PGP key certification service ($20/certificate)

Because current secure Web servers and browsers are not compatible with PGP, the NHSE would have to write its own applications software to verify PGP signatures. Also, the
X.509 certificates (discussed below) used by current secure Web servers and browsers take advantage of a more scaleable hierarchy of certification authorities for handling the trust problem discussed above. The incompatibility problem may change with future versions of PGP.

3.2.3 Kerberos

Kerberos is a network authentication system for use on physically insecure networks that allows entities communicating over those networks to prove their identity to each other while preventing eavesdropping or replay attacks . Kerberos provides for data stream integrity (detection of modification) and secrecy (prevention of unauthorized reading) using cryptography systems such as DES. Kerberos works by providing principals (users or services) with tickets that they can use to identify themselves to other principals and

with secret cryptographic keys for secure communication with other principals. Kerberos does not provide for authorization or accounting, although applications can use their secret keys to perform these functions securely. The NHSE would need to either incorporate Kerberos into a software distribution application as part of Repository in a Box, or contract with a company that provides commercial support for Kerberos, such as CyberSAFE Corporation, Cygnus Support, or OpenVision Technologies.

In a Kerberos system, there is a designated site on the network, called the Kerberos server, which performs centralized management and administrative functions. The server maintains a database containing the secret keys of all users, generates session keys whenever two users wish to communicate securely, and authenticates the identity of a user who requests certain network services. If the server is compromised, the integrity of the whole system fails. With Kerberos Version 5, multiple Kerberos systems, called ``realms", may interoperate.

Kerberos authentication, but not message content encryption, via HTTP is supported in NCSA HTTPd 1.5 (and 1.6b1) as well as XMosaic 2.7b.

Secret-key authentication systems such as Kerberos were designed to authenticate access to network resources, rather than to authenticate documents, a task which is better achieved via digital signatures. Because authentication of documents is needed for distribution of restricted NHSE software, for example for electronic license agreements, we do not consider Kerberos further in this report.

3.2.4 X.509 Certificates and Certification Authorities

The following is excerpted from the RSA Security FAQ Version 3.0[8] :

> ITU-T Recommendation X.509 specifies the authentication service for X.500 directories, as well as the widely adopted X.509 certificate syntax. The initial version of X.509 was published in 1988, version 2 was published in 1993, and version 3 was proposed in 1994 and considered for approval in 1995. Version 3 addresses some of the security concerns and limited flexibility that were issues in versions 1 and 2.
>
> Directory authentication in X.509 can be carried out using either secret-key techniques or public key techniques, with the latter based on public key certificates. An X.509 certificate consists of the following fields:
>
> - version
> - serial number
> - signature algorithm ID
> - issuer name
> - validity period

- subject (user) name
- subject public key information
- issuer unique identifier (version 2 and 3 only)
- subject unique identifier (version 2 and 3 only)
- extensions (version 3 only)
- signature on the above fields

This certificate is signed by the issuer to authenticate the binding between the subject (user's) name and the user's public key. The major difference between versions 2 and 3 is the addition of the extensions field. This field grants more flexibility as it can convey additional information beyond just the key and name binding. Standard extensions include subject and issuer attributes, certification policy information, and key usage restrictions, among others.

The X.509 standard is supported by a number of protocols, including PEM, PKCS, S-HTTP, and SSL.

The SSL (Secure Socket Layer) Handshake Protocol was developed by Netscape Communications Corporation to provide security and privacy over the Internet. The protocol supports server and client authentication. The SSL protocol is application independent, allowing protocols like HTTP, FTP (File Transfer Protocol), and Telnet to be layered on top of it transparently. The SSL protocol is able to negotiate encryption keys as well as authenticate the server before data is exchanged by the higher-level application. The SSL protocol maintains the security and integrity of the transmission channel by using encryption, authentication and message authentication codes.

The SSL Handshake Protocol consists of two phases, server authentication and client authentication, with the second phase being optional. In the first phase, the server, in response to a client's request, sends its certificate and its cipher preferences. The client then generates a master key, which it encrypts with the server's public key, and transmits the encrypted master key to the server. The server recovers the master key and authenticates itself to the client by returning a message encrypted with the master key. Subsequent data is encrypted with keys derived from this master key. In the optional second phase, the server sends a challenge to the client. The client authenticates itself to the server by returning the client's digital signature on the challenge, as well as its public-key certificate.

An X.509 certificate binds an identity to a pair of electronic keys that can be used for encrypting and signing digital information. The pair consists of two related keys - a public key and a private key. The public key can be used by anyone to verify a message signed with the private key or to encrypt a message that can only be decrypted using the private key. The private key must be kept secure and protected against unauthorized use.

Certificates are issued by a Certification Authority (CA), which is a trusted party that vouches for the identity of those to whom it issues certificates. To obtain a certificate, an individual generates his own key pair and sends the public key to the CA with proof of his identity. Different CAs may issue certificates with different levels of identification requirements. For example, Verisign is a commercial CA that offers four classes of certificates, with increasing levels of assurance (and cost) . Using the requirements for the particular level applied for, the CA checks the identification and then sends the requester a certificate attesting to the binding between the requester and his public key, along with (possibly) a hierarchy of certificates verifying the CA's public key.

The National Institute of Standards and Technology (NIST) is taking a leadership role in the development of a Federal Public Key Infrastructure that supports digital signatures and other public key-enabled security services [9]. In doing this, NIST is coordinating with industry and technical groups developing PKI technology such as the Federal PKI Steering Committee and its Technical Working Group (TWG), CommerceNet, Internet's PKIX, and the Open Group. NIST chairs the TWG, which is composed of technical representatives from Federal agencies and industry. Active since October 1994, the TWG has developed initial versions of a requirements document, a concept of operations, a technical security policy, an X509 v3 certificate profile, and an interoperability report. Laboratory activities include the development of a Reference Implementation and the initial implementation of a root Certification Authority (CA) for the Federal PKI.

To protect against long-term factoring attacks, a certificate has a validity period which should be shorter than the expected factoring time. The validity period, together with the need for security and the expected strength of an attacker, determines the appropriate key size which should be chosen when generating the key pair. In the event that someone's private key is compromised before it expires, he must let others know by adding the associated certificate to a Certificate Revocation List (CRL). The CRL is maintained by the Certification Authority that originally certified the key. When verifying a signature, one can check the CRL to make sure the signer's key has not been revoked.

Someone who obtains the private key of a CA could then forge certificates. Thus, a CA must ensure that its private key is extremely secure, for example by keeping it in a tamperproof box called a Certificate Signing Unit, or CSU. Furthermore, to protect against long-term factoring attacks, a CA should use very long keys and change keys regularly.

An individual can use a certificate to identify himself to secure servers such as membership based or access-controlled Web servers. Multiple certificates can be attached to a message or transaction, forming a certificate chain in which each certificate attests to the authenticity of the previous certificate. The top-level CA in the chain must be independently known and trusted by the recipient. When installed in a Web browser, a certificate functions as electronic credentials, eliminating the need for typing in a username and password. Similarly, a secure Web server uses its own certificate to assure clients that

the server is run by the organization claimed and to verify the integrity of the provided documents.

X.509 client certificates are currently supported by Netscape in its Navigator 3.0 browser. Microsoft has also announced support for X.509 certificates in its client applications. X.509 server certificates are currently used in server products from IBM, Microsoft, Netscape, OpenMarket, and Oracle.

NCSA HTTPd 1.6, currently in beta testing, provides support for the Secure-HTTP (S-HTTP) protocol and SSL version 2.0 and 3.0 . A version of XMosaic which supports S-HTTP and SSL is also available.

3.2.5 Simple Distributed Security Infrastructure

Simple Distributed Security Infrastructure (SDSI) has been proposed as a simpler alternative to X.509 [10]. SDSI combines a simple public-key infrastructure design with a means of defining groups and issuing group membership certificates. SDSI's groups provide terminology for defining access-control lists and security policies. SDSI's design relies on linked local name spaces rather than a hierarchical global name space.

3.2.6 Digital Time-stamping Services

After a key expires, everything that was signed with it will no longer be considered valid. If a signed document must remain valid after the key used to sign it expires, then the document should be time-stamped by a digital time-stamping service (DTS). A DTS issues a time-stamp which associates a date and time with a digital document in a cryptographically strong way. In addition to verifying the author's identity and the integrity of the document, the time-stamp also proves that the digital document existed at the time stated. Even if a private key used to sign the document is later compromised, the document remains valid. The contents of the document need not be revealed to the DTS. The author can compute a message digest of the document using a secure hash function and then send the message digest to the DTS, which returns a digital time-stamp consisting of the message digest, the date and time it was received, and the digital signature of the DTS. Strong cryptographic techniques must be used to ensure that time-stamps cannot be forged. One way to satisfy the strong cryptographic requirements is to store the private key of the DTS and an accurate clock inside a tamperproof box. The DTS must also have a long key that will be valid for several decades. A cryptographically strong DTS which avoids the need for tamperproof hardware has been implemented by researchers at Bellcore [11]. A digital time-stamping service is currently offered by Surety Technologies, and Bellcore has plans to offer such a service in the future .

3.3 Options for Restricting Access

3.3.1 By Domain Name

A simple method for enforcing access restrictions for NHSE software would be to test the hostname of the machine from which a request originates. This hostname could be used to determine whether or not the request was made from a host within a certain domain, such as .gov or .edu. Access to software could then either be allowed or denied based on the domain name.

The amount of effort required to enable this type of access control would be minimal. On the file server side, enabling this feature with most mainstream HTTP servers would be as simple as adding a few lines to the configuration file. On the client side, the whole process would be invisible unless, of course, the download request was denied.

Unfortunately, this type of access control presents some problems. One problem is that sometimes the partitioning of hosts created by domain names would not match the restriction criteria. For example, access permission to software is often based on what country the request originates from. However, some domain names, such as .org, .com, and .net, do not indicate where the host is geographically located while others, such as .us, .fr, and .uk do make this distinction.

Even if one chose to err on the side of caution and allow access only to those hosts that are partitioned correctly by domain names then the identity of the person who initiated the download would still be in question; a host with access privileges might be used merely as a waystation for software headed towards a host in another domain. Another problem with this type of access control is that there are many ``hacker tools" that could be used to break into or impersonate hosts from trusted domains.

Another problem with domain name based access restriction is that a browser can be set to use a proxy server to fetch documents, and the server doing the access restriction will only know about the domain name of the proxy, not the real user. If the proxy is in an allowed domain, then anyone can use the proxy to access files, unless the proxy does its own restriction.

Despite its weaknesses, access control by domain name is currently in use at MIT for distributing PGP, at Lucent Technologies for distributing Inferno, and at NCSA for distributed cryptographically-enhanced versions of NCSA httpd and Mosaic .

3.3.2 Username/password Access Restriction

When a user attempts to access a file that is protected by username/password access restriction, he or she is asked to enter a correct username and password before being allowed to download the file. In the case of an HTTP server, this type of access restriction is implemented by means of a configuration file which may be either global or directory-specific. In the case of an FTP server, accounts are set up for the allowed users on the file server machine, and access permission bits and ownership of files are set appropriately. With both HTTP Basic Authentication and commonly used FTP applications, passwords are sent over the network unencrypted. In HTTP MD5 Message Digest Authentication, the password is not sent over the network at all. Rather, a ``digest'' that is generated based on the password and other information about the request is hashed using MD5 and sent over the network. Digest Authentication is more secure over the network, but requires more rigorous security on the server machine, because the stored information cannot be encrypted with a one way function, whereas with Basic Authentication the server stores password using a one way encryption function.

3.3.3 Encryption Using Public Key Cryptography

With the public key cryptography method of access control, both the request for the software and the software itself are encrypted so that they cannot be read by anyone but the intended recipient. This method is intended to be combined with one of the public key authentication mechanisms described in section 3.2 - e.g., PGP, X.509, or SDSI. The request would take the form of a license agreement that the user signs using his public key to indicate agreement to the terms and conditions for using the software. The user's public key also identifies him or her so that the software server can check whether or not that user is authorized to obtain the software. The software itself would be best encrypted using a symmetric session key which would be generated for the purpose of this transmission only.

3.3.4 Distribution of Encrypted Software

An alternative to restricting access to the actual software files is to encrypt these files and allow anyone to download them, but require authorization and authentication to obtain the decryption key. This approach is used in the Crytolope technology of IBM Infomarket . In this case, the SSL protocol would be used by the server providing the decryption key to authenticate the user, sign a license agreement, and deliver the decryption key in a secure manner. A separate application would be needed to decrypt the software files.

3.3.5 A Combination of Access Controls for Export Restriction

In the December 30, 1996 Federal Register[12], a three part scheme for the distribution of encryption software over the Internet was outlined by the Bureau of Export Administration. The three parts consisted of a clickwrap declaring the citizenship of the consumer, a form providing personal information about the consumer, and a filter restricting access from IP addresses outside the United States. Using this as a basis, export experts say that any export restricted software, barring those explicitly used by the military, could be distributed using this scheme. Currently Rice University is seeking an advisory opinion directly from BXA on this issue.[13]

4.0 Liability Issues

The intellectual property issues between the organization that runs the repository and the organization that contributed the software can be rather complicated. Who owns the liability is the central question. The simple answer is whomever has possession of the software and whomever owns the software. However recent legal opinion is that a webpage which has a hotlink to a site which is in violation is also liable. For example, if a software has been found in violation of infringing on a registered copyright, the owner of the software, the repository which holds it, and any site which have active links to that software are all liable. However the courts will look favorably on a linking site if it showed a good faith effort not to provide access to an infringing site, does not make a profit from the site, and is only linking, not possessing any part of the software. Though liability most often lies with the organization holding the software, the intellectual property and legal restrictions involved with that responsibility are sometimes vast, vague and dynamic.

5.0 Conclusion

Different electronic tools are being used by university, federal, and industrial organizations to facilitate more efficient distributions of software technologies via the Internet. An organized and documented collection of these tools can provide a starting point to create or modify existing software distribution systems. The following list provides different types of technologies that should be considered when attempting to improve software distribution, authorization or authentication goals:

State the terms of licensing and receive a signed agreement

- Clickwrap
- Electronic Signature
- Digital Signature
- Digital Time-Stamping

Identify the recipient

- Certification Authorities and Encryptions Technologies
  - X.509 certificates

PGP
Simple Distributed Security Infrastructure
Email verification
Password

Verify the integrity of the software technology being distributed

MD5 Checksum

Limit the access to technologies the recipient is legally allowed to receive

By Domain (IP) Filter
User/password Restriction
Certification Authorities including X.509 Certificates and others
Simple Distributed Security Infrastructure
Encryption using Public Key Cryptography
Cookies
User Information Form

Log the transaction for future reference

Digital Time-stamping Services
On-line Form of User Information
Domain Name Log
Certification Authorities including X.509 Certificates and others
Cookies
Integration With System Databases (technology database, user database, log, etc.)
Automatic Email Notification to the developer when software is downloaded with user information about the recipient
Automatic Email Notification to registered users when appropriate new software entries are added to the software catalogue

Current and emerging technologies that facilitate software distribution on the web are receiving increasing support from the legal and business community as well as the government. Confidence can be gained from the recent amount of legal discussion from the ABA and state governments about digital signatures and export regulation changes. Federal officials have been receptive to the three-part export protection scheme (currently used for encryption software), and thirty-nine states have digital and/or electronic signature legislation in place or progress. Internet issues are being researched and discussed before cases are being filed, and lawyers and government officials have confidence that the web's authentication and authorization methods will hold up in court.

However since there is an absence of legal precedent, the outcome of future cases involving technologies such as PGP, X.509 certificates, and digital signatures are uncertain. Courts could come to different conclusions as has been seen in the Pro CD case. Consulting legal counsel about how your distribution system complies with federal, state, and organizational policy is the safest method of violation prevention.

References

1 Uniform methodology for releasing NASA computer software. Commercial Development and Technology Transfer Division, NASA Headquarters, June 3, 1996 draft version.

2 17 USC Section 105

3 D.C. Toedt III. The Law and Business of Computer Software, Chapter 11, July 1997.

4 *Pro CD v. Zeidenberg*, No. 96-1139, (United States Court of Appeals for the Seventh Circuit, June 20, 1996)

5 Jacob C. Reinbolt. "Licenses agreements of the Internet" from "Introduction to Internet Law". California Computer Expo '96. San Diego Convention Center, August 30, 1996.

6 Thomas J. Semdinghoff. "Analyzing State Digital Signature Legislation", Electronic Banking Law and Commerce Report, June 1997.

7 Simon Garfinkel. PGP - Pretty Good Privacy. O'Reilly and Associates, Inc., 1995.

8 CCITT. Recommendation X.509: The Directory - Authentication Framework. 1988.

9 Warwick Ford. A Public Key Infrastructure for U.S. Government Unclassified but Sensitive Applications. National Institute of Standards and Technology, 1995. http://csrc.nist.gov/pki/

10 Ronald Rivest and Butler Lampson. SDSI - a simple distributed security infrastructure. http://theory.lcs.mit.edu/~rivest/sdsi11.html, September 1996.

11 S. Haber and W. W. Stornetta. How to time-stamp a digital document. Journal of Cryptology, 3(2):99-112, 1991.

12 Federal Register. Vol. 61, no. 251.

13 Hugh Kress, *Arnold, White, and Durkee*. Letter to the Office of Exporter Assistance. August 13, 1997.

Useful On-line References

General IP
http://www.lawnotes.com - IP Law FAQs
http://www.ljextra.com/practice/intellectualproperty

Electronic Commerce
http://www.law.miami.edu/~froomlein/articles/trustedno.htm - Role of Trusted Third Parties
http://java.sun.com/products/jdk/lil/docs/guide/security/cert2.html - X.509 Certificates

Export Regulations
http://www.jya.com/eartoc.htm - Updated Regulations
http://www.ita.doc.gov - International Trade Administration
http://www.bxa.doc.gov - Bureau of Export Administration

Copyrights
http://www.patents.com/copyrights.sht - General Information
http://lcweb.loc.gov/copyright - US Copyright Office

Patents
http://www.uspto.gov - US Patent and Trademark Office
http://www.patents.com/patents.sht - General Information about Patents