Security Architecture and Design

en.wikibooks.org

January 25, 2014

Contents

0.1 **Computer Systems Architecture** 1
0.2 **Systems Security Architecture** 7
0.3 **Security Models** 11
0.4 **Security Product Evaluation Methods and Criteria** 15

1 Contributors **23**

List of Figures **25**

2 Licenses **29**
2.1 GNU GENERAL PUBLIC LICENSE 29
2.2 GNU Free Documentation License 30
2.3 GNU Lesser General Public License 31

0.1 Computer Systems Architecture

A computer system consists of different types of components like hardware, software,operating systems and firmware.

The most important and common components being

- Hardware Components like: CPU, Storage Devices, I/O Devices, Communication Devices
- Software Components like: Operating Systems, Application Programs
- Firmware

0.1.1 Central Processing Unit (CPU)

Overview

- CPU is the brain of the computer.
- It fetches the instructions from memory and executes them
- Each CPU type has its own instruction set and architecture

CPU Components

- *Registers*: are temporary storage locations that can store references to memory locations, next instruction to be executed etc and also enable the CPU to keep its status information.
- *Arithmetic Logic Unit(ALU)*: performs the actual execution of complex mathematical functions and logical operations on data.
- *Control Unit*: manages and synchronizes the system while different applications code and OS instructions are being executed. It fetches the code,interprets the code and oversees

the execution of the different instruction sets.It determines what application instructions get processed and in what priority and time slice. It controls when instructions are executed, and this execution enables applications to process data.
- *General registers* are used to hold variables and temporary results as the ALU works through its execution steps.
- *Special registers* (dedicated registers) hold information such as the program counter, stack pointer, and program status word (PSW)
  - *Program counter (PC)* register contains the memory address of the next instruction that needs to be fetched
  - *Program status word (PSW)* holds different condition bits. One of the bits indicates whether the CPU should be working in user mode (also called problem state) or privileged mode (also called kernel or supervisor mode).
  - *Stack pointer*directs the CPU where the next piece of data is located.
- *Address bus*, is a hardwired connection to the RAM chips in the system and the individual (I/O) devices. It is used by the CPU to indicate the location of the instructions that need to be processed.
- *Data bus* is used by the memory or I/O device in response to CPU request for sending the data that resides at the requested location.

0.1.2 Storage

Overview

- A storage device is a hardware device capable of storing data.
- Storage devices can be classified into 3 categories
  - Primary Storage or Memory: which are directly accessible to the CPU like Cache Memory (L1,L2,L3),Main Memory (RAM)
  - Secondary Storage: Which are permanent storage devices like Hard Disks, Floppy Disks, CDs, DVDs, flash memory, ZIP drives etc.
  - Tertiary Storage : Tapes

Random Access Memory

- RAM stores data and program instructions temporarily by the operating system and application programs.
- It is described as volatile because if the computer's power supply is terminated, then all information within this type of memory is lost.
  - RAM Types
    - Dynamic RAM- uses capacitors which have to be refreshed periodically to hold the data, slow.
      - Synchronous DRAM (SDRAM)- timing of the CPU and the timing of the memory activities are synchronized. can access only one block at a time.
      - Extended data out DRAM (EDO DRAM)- can capture the next block of data while the first block is being sent to the CPU for processing, faster than SDRAM.
      - Burst EDO DRAM (BEDO DRAM)- Build on top of EDO DRAM.It reads and sends up to four memory addresses in a small number of clock cycles.
      - Double data rate SDRAM (DDR SDRAM)- Carries out read operations on the rising and falling cycles of a clock pulse. speed is twice of SDRAM

- Static RAM- uses more transistors, faster than DRAM,expensive, used in cache.

Read Only Memory

- ROMs are non-volatile memories.the instructions stored in these memories are called as firmware.
- ROM Types
  - Programmable read-only memory (PROM)-can be programmed only one time after the manufacturing.
  - Erasable and programmable read-only memory (EPROM)- can be erased, modified,and upgraded.
  - Flash memory- a special type of memory that is used in digital cameras, BIOS chips, memory cards for laptops, and video game consoles. It is used more as a type of hard drive than memory.

Cache Memory

- The cache memory is a smaller, faster memory which stores copies of the data from the most frequently used main memory locations.
- Cache Levels
  - L1 Cache- located within the CPU
  - L2 Cache- located between the CPU and the main memory (RAM)
  - L3 Cache- can be an externally managed memory that has faster access time than RAM.
- Cache Types
  - Prefetch Cache(P-Cache)- used to store data that has been brought in as a result of a prefetch operation
  - W-Cache - acts as a holding station for stored data.
  - Instruction Cache- used to store instructions.
  - Data Cache- stores data
  - Translation Look aside Buffer (TLB)- stores the translated addresses of virtual page address to a valid physical address.
- Cache Organization- describes the organization of lines and the replacement policy
  - Direct Mapping- The (virtual or physical) memory address of the incoming cache line controls which cache location is going to be used.Suffers from thrashing.
  - Fully Associative- based on LRU policy where the LRU line is replaced.
  - Set Associative- uses several direct-mapped caches or set. A cache controller decides which set the line will go into. Within the set, a direct-mapped scheme is used to allocate a slot in the cache.

Virtual Memory

- Virtual memory is a logical memory that gives an application program the impression that it has a contiguous working memory.
- A VM logically extends the capabilities of RAM by allocation a separate portion of the hard disk space called swap space.
-

0.1.3 Operating Systems

Overview

- An operating system provides an environment for applications and users to work within.
- It is responsible for managing the underlying hardware components,memory management, I/O operations, file system, process management, and providing system services
- OS Architectures
  - Monolithic operating system architecture- mainly made up of various procedures that can call upon each other in a haphazard manner, provides single layer security only. For example, DOS.
  - Layered operating system- separates system functionality into hierarchical layers,provide data hiding, provides multilayer security. For example, Unix.
    - Layer 0 controlled access to the processor and provided multiprogramming functionality
    - Layer 1 carried out memory management
    - Layer 2 provided interprocess communication
    - Layer 3 dealt with I/O devices and
    - Layer 4 was where the applications resided
  - Microkernel Based- The OS functions are divided into several different processes that run in user mode, instead of kernel mode.The processes communicate in a C/S model.The server processes are called as subsystems and provide services to user process and other OS processes. For example, Windows 2000.

Process Management

- A process is a program in execution that is loaded and actuated by the OS.It contains a set of the instructions and the assigned resources.
- When a process is created, the operating system assigns resources to it, such as a memory segment, CPU time slot (interrupt), access to system application programming interfaces (APIs), and files to interact with.
- Process States: A process can run in running state (CPU is executing its instructions and data),ready state (waiting to send instructions to the CPU), or blocked state (waiting for input data, such as keystrokes from a user).
- Process Table: A data structure that contains each individual process's state, stack pointer, memory allocation,program counter, and status of open files in use.
- Threads: A thread is a unit of program execution. conversely a process is a single thread of execution.A thread is made up of individual instruction set and the data that needs to be worked on by the CPU.All the threads of a process share the resources of the processes that created them.
- Process Scheduling: governs the way different processes communication(or synchronize) between each other in order to overcome deadlock conditions.

Memory Management

- The main objectives of memory management is to
  - Provide an abstraction level for programmers
  - Maximize performance with the limited amount of memory available
  - Protect the operating system and applications loaded into memory
- The memory manager has five basic responsibilities:

- Relocation
  - Swap contents from RAM to the hard drive as needed
  - Provide pointers for applications if their instructions and memory segment have been moved to a different location in main memory
- Protection
  - Limit processes to interact only with the memory segments that are assigned to them
  - Provide access control to memory segments
- Sharing
  - Use complex controls to ensure integrity and confidentiality when processes need to use the same shared memory segments
  - Allow many users with different levels of access to interact with the same application running in one memory segment
- Logical organization
  - Allow for the sharing of specific software modules, such as dynamic link library (DLL) procedures
- Physical organization
  - Segment the physical memory space for application and operating system processes

I/O Device Management

- I/O devices are usually considered block or character devices.
- A block device works with data in fixed-size blocks, each block with its own unique address. For example, disk drive .
- A character device, such as a printer, network interface card, or mouse, works with streams of characters, without using any fixed sizes. This type of data is not addressable
- There are different ways that operating systems can manage software I/O procedures.
  - Programmed I/O- the CPU sends data to an I/O device and polls the device to see if it is ready to accept more data. If the device is not ready to accept more data, the CPU wastes time by waiting for the device to become ready. very slow
  - Interrupt-driven I/O- the CPU sends a character/block over to the device and then goes and works on another process's request. When the device is done with its job it sends an interrupt to the CPU. The CPU stops what it is doing, sends the next block/character and this continues until the entire job is processes by the device.lot of interrupt handling overhead.
  - I/O using DMA- A DMA controller feeds the characters from the memory to the device without bothering the CPU. This method is sometimes referred to as unmapped I/O.

OS Protection Mechanisms

- CPU Modes
  - CPU modes also called processor modes or CPU privilege levels,are operating modes for the central processing unit of some computer architectures that place restrictions on the operations that can be performed by the process currently running in the CPU. This design allows the operating system to run at different privilege levels like kernel mode, user mode, master mode etc.
- Protection Rings
  - Protection Rings are concentric rings that provide strict boundaries and definitions for what the processes that work within each ring can access and what operations they can successfully execute.

  - The processes that operate within the inner rings have more privileges than the processes operating in the outer rings, because the inner rings only permit the most trusted components and processes to operate within them.
  - Protection rings support the availability, integrity, and confidentiality requirements of multitasking operating systems.
  - The most commonly used architecture provides four protection rings:
    - Ring 0 Operating system kernel
    - Ring 1 Remaining parts of the operating system
    - Ring 2 I/O drivers and utilities
    - Ring 3 Applications and user activity
- Memory Protection
  - Limit processes to interact only with the memory segments that are assigned to them
  - Provide access control to memory segments
- Process Isolation
  - Ensures that processes do not “step on each other’s toes," negatively affect each other’s productivity and thus communicate in an insecure manner.
  - Methods for process isolation:
    - Encapsulation of objects- no other process understands or interacts with the internal programming code of a process.
    - Time multiplexing of shared resources- allows processes to use the same resources on a time sharing basis.
    - Naming distinctions- different processes have their own name or identification value called as PID
    - Virtual mapping- Every process has its own virtual memory address space.
- Domains
  - A domain is defined as a set of objects that a subject is able to access.
  - The domain can be all the resources a user can access, all the files available to a program, the memory segments available to a process, or the services and processes available to an application.
  - A process that resides in a privileged domain needs to be able to execute its instructions and process its data with the assurance that programs in a different domain cannot negatively affect its environment. This is referred to as an execution domain
  - The higher the level of trust, the larger the number of available resources or objects

0.1.4 Firmware

Overview

- Firmware is a computer program that is embedded in a hardware device
- Like software, it is a computer program which is executed by a microprocessor or a microcontroller. But it is also tightly linked to a piece of hardware, and has little meaning outside of it.

BIOS

- BIOS refers to the firmware code run by a personal computer when first powered on.
- The primary function of the BIOS is to identify and initiate component hardware (such as hard disk, floppy and optical disk drives). This is to prepare the machine so other

software programs stored on various media can load, execute, and assume control of the PC.This process is known as booting, or booting up, which is short for bootstrapping.

ROM image

- A ROM image, or simply ROM, is a computer file which contains a copy of the data from a read-only memory chip, often from a video game cartridge, a computer's firmware, or from an arcade game's main board.
- The term is frequently used in the context of emulation, whereby older games or computer firmware are copied to ROM files on modern computers and can, using a piece of software known as an emulator, be run on the newer computer.

0.1.5 Virtual Machines

Overview

- A virtual machine (VM) is a software implementation of a machine (computer) that executes programs like a real machine
- An essential characteristic of a virtual machine is that the software running inside is limited to the resources and abstractions provided by the virtual machine -- it cannot break out of its virtual world

VM Categories

- Virtual machines are separated in two major categories, based on their use and degree of correspondence to any real machine.
  - *system virtual machine* provides a complete system platform which supports the execution of a complete operating system.
  - *process virtual machine* is designed to run a single program, which means that it supports a single process.

0.2 Systems Security Architecture

The security architecture is one component of a product's overall architecture and is developed to provide guidance during the design of the product. It outlines the level of assurance that is required and potential impacts that this level of security could have during the development stages and on the product overall.

0.2.1 Security Design Principles

Security is a system requirement just like performance, capability, cost, etc.Therefore, it may be necessary to trade off certain security requirements to gain others.

Principles of Secure Design

- Design security in from the start
- Allow for future security enhancements
- Minimize and isolate security controls

- Employ least privilege
- Structure the security relevant features
- Make security friendly
- Don't depend on secrecy for security

Principles for Software Security

- Secure the weakest link
- Practice defense in depth
- Fail securely- If your software has to fail, make sure it does it securely
- Follow the principle of least privilege
- Compartmentalize- Minimize the amount of damage that can be done by breaking the system into units
- Keep it simple- Complex design is never easy to understand
- Promote privacy- Try not to do anything that compromises the privacy of the user
- Remember that hiding secrets is hard
- Be reluctant to trust- Instead of making assumptions that need to hold true, you should be reluctant to extend trust
- Use your community resources- Public scrutiny promotes trust

Design Principles for Protection Mechanisms

- Least privilege- Should only have the rights necessary to complete your task.
- Economy of mechanism- Should be sufficiently small and as simple as to be verified and implemented – e.g., security kernel. Complex mechanisms should be correctly Understood, Modeled, Configured, Implemented and Used
- Complete mediation- Every access to every object must be checked
- Open design- Let the design be open. *Security through obscurity* is a bad idea
- Should be open for scrutiny by the community- Better to have a friend/colleague find an error than a foe
- Separation of privilege- Access to objects should depend on more than one condition being satisfied
- Least common mechanism- Minimize the amount of mechanism common to more than one user and depended on by all users
- Psychological acceptability- User interface must be easy to use, so that users routinely and automatically apply the mechanisms correctly. Otherwise, they will be bypassed
- Fail-safe defaults. Should be lack of access

0.2.2 Trusted Computing Base

Overview

- A Trusted Computing Base (TCB) is the whole combination of protection mechanisms within a computer system.
- The TCB addresses all the security components of the hardware, software, and firmware within the system.
- It does not address the level of security that a system provides, but rather the level of trust that a system provides as because no computer system can be totally secure

- If the TCB is enabled, then the system has a trusted path, a trusted shell, and system integrity–checking capabilities
  - A trusted path is a communication channel between the user, or program, and the kernel. The TCB provides protection resources to ensure that this channel cannot be compromised in any way
  - A trusted shell means that someone who is working in that shell cannot "bust out of it" and other processes cannot "bust into" it.
- The TCB contains components that directly enforce the security policy (is a set of rules and practices that dictates how sensitive information and resources are managed, protected, and distributed.)

Basic Functions of a TCB

- Process Activation- deals with the activities that have to take place when a process is going to have its instructions and data processed by the CPU.
- Execution Domain Switching- takes place when a process needs to call upon a process in a higher protection ring.
- Memory protection and
- I/O operations

Evaluating the TCB

- Evaluating the trust level of a system includes identifying the architecture, security services, and assurance mechanisms that make up the TCB.
- During the evaluation process, the tests must show how the TCB is protected from accidental or intentional tampering and compromising activity.
- For systems to achieve a higher trust level rating, they must meet well-defined TCB requirements,and the details of their operational states, developing stages, testing procedures, and documentation will be reviewed with more granularity than systems that are attempting to achieve a lower trust rating.

Reference Monitor and Security Kernel

- The reference monitor is an abstract machine that mediates all access subjects have to objects, both to ensure that the subjects have the necessary access rights and to protect the objects from unauthorized access and destructive modification.
- The security kernel is made up of hardware, software, and firmware components that fall within the TCB and implements and enforces the reference monitor concept.
- The security kernel mediates all access and functions between subjects and objects. The security kernel is the core of the TCB and is the most commonly used approach to building trusted computing systems. There are three main requirements of the security kernel:
  - It must provide isolation for the processes carrying out the reference monitor concept, and the processes must be tamper-proof.
  - It must be invoked for every access attempt and must be impossible to circumvent. Thus, the security kernel must be implemented in a complete and foolproof way.
  - It must be small enough to be able to be tested and verified in a complete and comprehensive manner.

Security Perimeter

- A security perimeter is a boundary that divides the trusted from the untrusted.

- For the system to stay in a secure and trusted state, precise communication standards must be developed to ensure that when a component within the TCB needs to communicate with a component outside the TCB, the communication cannot expose the system to unexpected security compromises. This type of communication is handled and controlled through interfaces.

The Relation

- The reference monitor is a concept in which an abstract machine mediates all access to objects by subjects.
- The security kernel is the hardware, firmware, and software of a TCB that implements this concept.
- The TCB is the totality of protection mechanisms within a computer system that work together to enforce a security policy. The TCB contains the security kernel and all other security protection mechanisms

0.2.3 Security Modes of Operation

Overview

- A system can operate in different modes depending on the sensitivity of the data being processed, the clearance level of the users, and what those users are authorized to do.
- The mode of operation describes the security conditions under which the system actually functions.
- Trust vs Assurance
  - A trust is a level of confidence or belief that tells the customer how much protection he can expect out of the system.
  - In a trusted system, all protection mechanisms work together to process sensitive data for many types of uses, and will provide the necessary level of protection per classification level
  - Assurance is an higher level of confidence that looks at the same issue but in more depth and detail where the system is throughly inspected.

Dedicated Security Mode All users must have...

- Proper clearance for all information on the system
- Formal access approval for all information on the system
- Signed NDA for all information on the system
- Valid need to know for all information on the system

All users can access all data.

System High-Security Mode All users must have...

- Proper clearance for all information on the system
- Formal access approval for all information on the system
- Signed NDA for all information on the system
- Valid need to know for some information on the system

All users can access some data, based on their need to know.

Compartmented Security Mode All users must have...

- Proper clearance for the highest level of data classification on the system
- Formal access approval for all information they will access on the system
- Signed NDA for all information they will access on the system
- Valid need to know for some of the information on the system

All users can access some data, based on their need to know and formal access approval.

Multilevel Security Mode All users must have...

- Proper clearance for all information they will access on the system
- Formal access approval for all information they will access on the system
- Signed NDA for all information they will access on the system
- Valid need to know for some of the information on the system

All users can access some data, based on their need to know, clearance, and formal access approval.

0.3 Security Models

- A security policy is a document that expresses clearly and concisely what the protection mechanisms are to achieve. Its a statement of the security we expect the system to enforce.
- A security model is a specification of a security policy:
  - it describes the entities governed by the policy,
  - it states the rules that constitute the policy.
- There are various types of security models:
  - Models can capture policies for confidentiality (Bell-LaPadula) or for integrity (Biba, Clark-Wilson).
  - Some models apply to environments with static policies (Bell-LaPadula), others consider dynamic changes of access rights (Chinese Wall).
  - Security models can be informal (Clark-Wilson), semi-formal, or formal (Bell-LaPadula, Harrison-Ruzzo-Ullman).
- Model vs Policy
  - A security model maps the abstract goals of the policy to information system terms by specifying explicit data structures and techniques that are necessary to enforce the security policy. A security model is usually represented in mathematics and analytical ideas, which are then mapped to system specifications, and then developed by programmers through programming code
  - For Example, if a security policy states that subjects need to be authorized to access objects, the security model would provide the mathematical relationships and formulas explaining how x can access y only through the outlined specific methods
  - A security policy outlines goals without regard to how they will be accomplished. A model is a framework that gives the policy form and solves security access problems for particular situations.

0.3.1 Lattice Models

- A lattice is a mathematical construct that is built upon the notion of a group.

- A lattice is a mathematical construction with:
  - a set of elements
  - a partial ordering relation
  - the property that any two elements must have unique least upper bound and greatest lower bound
- A security lattice model combines multilevel and multilateral security
- Lattice elements are security labels that consist of a security level and set of categories

0.3.2 State Machine Models

- In state machine model, the state of a machine is captured in order to verify the security of a system.
- A given state consists of all current permissions and all current instances of subjects accessing the objects. If the subject can access objects only by means that are concurrent with the security policy, the system is secure.
- The model is used to describe the behavior of a system to different inputs. It provides mathematical constructs that represents sets (subjects, objects) and sequences. When an object accepts an input , this modifies a state variable thus transiting to a different state.
- Implementation tips
  - The developer must define what and where the state variables are.
  - The developer must define a secure state for each state variable.
  - Define and identify the allowable state transition functions.
  - The state transition function should be tested to verify that the overall m/c state will not compromise and the integrity of the system is maintained.

0.3.3 Noninterference Models

- The model ensures that any actions that take place at a higher security level do not affect, or interfere with, actions that take place at a lower level.
- It is not concerned with the flow of data, but rather with what a subject knows about the state of the system. So if an entity at a higher security level performs an action, it can not change the state for the entity at the lower level.
- The model also addresses the inference attack that occurs when some one has access to some type of information and can infer(guess) something that he does not have the clearance level or authority to know.

0.3.4 Bell—LaPadula Confidentiality Model

- It was the first mathematical model with a multilevel security policy that is used to define the concept of a secure state machine and models of access and outlined rules of access.
- It is a state m/c model that enforces the confidentiality aspects of access model.
- The model focuses on ensuring that the subjects with different clearances(top secret, secret, confidential) are properly authenticated by having the necessary security clearance , need to know , and formal access approval-before accessing an object that are under different classification levels (top secret, secret, confidential).

- The rules of Bell-Lapadula model
  - Simple security rule (no read up rule): It states that a subject at a given security level can not read data that resides at a higher security level.
  - Star property rule (no write down rule): It states that a subject in a given security level can not write information to a lower security levels.
- Strong star property rule: It states a subject that has read and write capabilities can only perform those functions at the same security level , nothing higher and nothing lower.
- *Tranquility principle* : subjects and objects can not change their security levels once they have been instantiated (created).
- All MAC systems are based on the Bell – Lapadula model because of it multilevel security.
- Designed US govt and mostly adopted by govt agencies

0.3.5 Biba Integrity Model

- It is developed after Bell – Lapadula model.
- It addresses integrity of data unlike Bell – Lapadula which addresses confidentiality.
- It uses a lattice of integrity levels unlike Bell – Lapadula which uses a lattice of security levels.
- It is also an information flow model like the Bell – Lapadula because they are most concerned about data flowing from one level to another.
- The rules of Biba model
  - simple integrity rule(no read down) : it states that a subject can not read data from a lower integrity level.
  - star integrity rule(no write up) : it states that a subject can not write data to an object at a higher integrity level.
  - invocation property : it states that a subject can not invoke(call upon) a subject at a higher integrity level.

0.3.6 Clark—Wilson Integrity Model

- It was developed after Biba and addresses the integrity of information.
- This model separates data into one subject that needs to be highly protected , referred to as a constrained data item(CDI)and another subset that does not require high level of protection , referred to as unconstrained data items(UDI).
- Components
  - Subjects (users): are active agents.
  - Transformation procedures (TPs): the s/w procedures such as read, write, modify that perform the required operation on behalf of the subject (user).
  - Constrained data items (CDI): data that can be modified only by Tp's.
  - Unconstrained data items (UDI): data that can be manipulated by subjects via primitive read/write operations.
  - Integrity verification procedure (IVP): programs that run periodically to check the consistency of CDIs with external reality. These integrity rules are usually defined by vendors.
- Integrity goals of Clark – Wilson model
  - Prevent unauthorized users from making modification (addressed by Biba model).

- Separation of duties prevents authorized users from making improper modifications.
- Well formed transactions: maintain internal and external consistency i.e. it is a series of operations that are carried out to transfer the data from one consistent state to the other.

0.3.7 Access Control Matrix

- This model addressed in access control.
- Commonly used in OS and applications.

0.3.8 Information Flow Models

- In this model, data is thought of as being held in individual discrete compartments.
- Information is compartmentalized based on two factors.
  - Classification and
  - Need to know
- The subjects clearance has to dominate the objects classification and the subjects security profile must contain the one of the categories listed in the object label, which enforces need to know.
- For example:
  - Bell – Lapadula which prevents information flowing from higher source level to lower source level.
  - Biba which prevents information flowing from lower integrity level to higher integrity level

Covert channels

- A covert channel is a way for an entity to receive information in an unauthorized manner.
- It is an information flow that is not controlled by a security mechanism.
- It is an unauthorized communication path that is not protected by the system because it was uncovered while developing the system.
- Types of covert channels
  - Covert timing: in this channel, one process relays information to another by modulating its use of system resources.
  - Covert storage: in this channel, one process writes data to a storage location and another process directly, or indirectly reads it.

0.3.9 Graham—Denning Model

- This model defines a set of basic rights in terms of commands that a specific subject can execute on an object.
- It proposes the eight primitive protection rights, or rules of how these types of functionalities should take place securely.
  - How to securely create an object.
  - How to securely create a subject.
  - How to securely delete an object.
  - How to securely delete a subject.

- How to provide read access rights.
- How to provide grant access rights.
- How to provide delete access rights.
- How to provide transfer access rights.

0.3.10 Harrison—Ruzzo—Ullman Model

- The HRU security model (Harrison, Ruzzo, Ullman model) is an operating system level computer security model which deals with the integrity of access rights in the system. The system is based around the idea of a finite set of procedures being available to edit the access rights of a subject s on an object o.
- The model also discussed the possibilities and limitations of proving safety of a system using an algorithm.

0.3.11 Brewer—Nash (Chinese Wall)

- This model provides access controls that can change dynamically depending upon a user's previous actions.
- The main goal of this model is to protect against conflicts of interests by user's access attempts.
- It is based on the information flow model, where no information can flow between subjects and objects in a way that would result in a conflict of interest.
- The model states that a subject can write to an object if, and only if, the subject can not read another object that is in a different data set.

0.4 Security Product Evaluation Methods and Criteria

- A security evaluation examines the security-relevant parts of a system, meaning the TCB, access control mechanisms, reference monitor, kernel, and protection mechanisms. The relationship and interaction between these components are also evaluated.
- There are different methods of evaluating and assigning assurance levels to systems. Two reasons explain why more than one type of assurance evaluation process exist:
  - methods and ideologies have evolved over time, and
  - various parts of the world look at computer security differently and rate some aspects of security differently
- An evaluation program establishes a trust between the security product vendor and the customer.

Evaluation Standards

- Information Technology Security Evaluation Criteria (ITSEC)- EU
- Trusted Computing Security Evaluation Criteria (TCSEC) -US
- Common Criteria- Hybrid of ITSEC and TCSEC

0.4.1 Rainbow Series

The Rainbow series was a series of books to cover all the areas of security like

- Red Books- Network Security
- Orange Books- Operating System Security
- Yellow- Security Risk Management
- Violet- Database Security

> **Information**
>
> Originally only the orange book existed but the other books have evolved to cover all the areas of security and thus the collection is names as Rainbow series

Orange Book or TCSEC

Overview

- TCSEC was developed by US DoD and was published in an Orange book and hence also called as *Orange Book*
- It mainly addresses the confidentiality, but not integrity and mainly addresses government and military requirements.
- It is used to evaluate whether a product contains the security properties the vendor claims it does and whether the product is appropriate for a specific application or function.
- It is used to review the functionality, effectiveness,and assurance of a product during its evaluation, and it uses classes that were devised to address typical patterns of security requirements.
- TCSEC provides a classification system that is divided into different assurance levels with A representing the highest and D the lowest.
  - A: Verified protection
  - B: Mandatory protection: Variants- B1<B2<B3
  - C: Discretionary protection: Variants-C1<C2
  - D: Minimal security

The levels are concentric.i.e if a product is assured at level B2, that implies it meets D,C1,C2 and B1

The Criteria

- Security policy- The policy must be explicit and well defined and enforced by the mechanisms within the system.
- Identification- Individual subjects must be uniquely identified.
- Labels Access- Control labels must be associated properly with objects.
- Documentation- Documentation must be provided, including test, design, and specification documents, user guides, and manuals.
- Accountability- Audit data must be captured and protected to enforce accountability.
- Life cycle assurance- Software, hardware, and firmware must be able to be tested individually to ensure that each enforces the security policy in an effective manner throughout their lifetimes.

- Continuous protection- The security mechanisms and the system as a whole must perform predictably and acceptably in different situations continuously.

> **Information**
>
> Even though the evaluation is done independently based on the above categories, the rating that is assigned at the end is a sum total of these items.

The Assurance Levels

- Division D: Minimal Protection- It is reserved for systems that have been evaluated but fail to meet the criteria and requirements of the higher divisions.
- Division C: Discretionary Protection- The C rating category has two individual assurance ratings within it.
  - C1: Discretionary Security Protection
    - Based on individuals and/or groups.
    - It requires a separation of users and information, and identification and authentication of individual entities is provided.
    - The type of environment that would require this rating is one in which users are processing information at the same sensitivity level; thus, strict access control and auditing measures are not required.
    - It would be a trusted environment with low security concerns
  - C2: Controlled Access Protection
    - Users need to be identified individually to provide more precise access control and auditing functionality.
    - Logical access control mechanisms are used to enforce authentication and the uniqueness of each individual's identification
    - The type of environment that would require systems with a C2 rating is one in which users are trusted but a certain level of accountability is required.
    - Overall, C2 is regarded to be the most reasonable class for commercial applications, but the level of protection is still relatively weak
- Division B: Mandatory Protection- Mandatory access control is enforced by the use of security labels. The architecture is based on the Bell-LaPadula security model, and evidence of reference monitor enforcement must be available
  - B1: Labeled Security
    - Each data object must contain a classification label and each subject must have a clearance label.
    - When a subject attempts to access an object,the system must compare the subject's and object's security labels to ensure that the requested actions are acceptable.
    - Data leaving the system must also contain an accurate security label.
    - The security policy is based on an informal statement, and the design specifications are reviewed and verified.
    - This security rating is intended for environments that require systems to handle classified data.
  - B2: Structured Protection
    - The security policy is clearly defined and documented,and the system design and implementation are subjected to more thorough review and testing procedures.

    - This class requires more stringent authentication mechanisms and well-defined interfaces among layers.
    - Subjects and devices require labels, and the system must not allow covert channels.
    - The type of environment that would require B2 systems is one that processes sensitive data that requires a higher degree of security and are relatively resistant to penetration and compromise.
  - B3: Security Domains
    - In this class, more granularity is provided in each protection mechanism, and the programming code that is not necessary to support the security policy is excluded.
    - The reference monitor components must be small enough to test properly and be tamper-proof
    - The system must be able to recover from failures without its security level being compromised.
    - The type of environment that requires B3 systems is a highly secured environment that processes very sensitive information and are highly resistant to penetration.
- Division A: Verified Protection- Formal methods are used to ensure that all subjects and objects are controlled with the necessary discretionary and mandatory access controls. The design, development, implementation, and documentation are looked at in a formal and detailed way
  - A1: Verified Design
    - Formal techniques are used to prove the equivalence between the TCB specifications and the security policy model.
    - More stringent change configuration is put in place with the development of an A1 system, and the overall design can be verified
    - The type of environment that would require A1 systems is the most secure of secured environments. This type of environment deals with top-secret information and cannot adequately trust anyone using the systems without strict authentication, restrictions,and auditing.

TCSEC Myths

- It looks specifically at the operating system and not at other issues like networking, databases, and so on.
- It focuses mainly on one attribute of security, confidentiality, and not at integrity and availability.
- It works with government classifications and not the protection classifications that commercial industries use.
- It has a relatively small number of ratings, which means many different aspects of security are not evaluated and rated independently.

Red Book or TNI

Overview

- Red books also called Trusted Network Interpretation (TNI),addresses security evaluation topics for networks and network components.

- Like the Orange Book, the Red Book does not supply specific details about how to implement security mechanisms; instead, it provides a framework for securing different types of networks
- It rates the confidentiality of data and operations that happen within a network and the network components and products.

Security Items addresses in the Orange Books

- Communication integrity
  - Authentication Protects against masquerading and playback attacks. Mechanisms include digital signatures, encryption, timestamp, and passwords.
  - Message integrity Protects the protocol header, routing information, and packet payload from being modified. Mechanisms include message authentication and encryption.
  - Non-repudiation Ensures that a sender cannot deny sending a message.Mechanisms include encryption, digital signatures, and notarization.
- Denial-of-service prevention
  - Continuity of operations Ensures that the network is available even if attacked. Mechanisms include fault tolerant and redundant systems and the capability to reconfigure network parameters in case of an emergency.
  - Network management Monitors network performance and identifies attacks and failures. Mechanisms include components that enable network administrators to monitor and restrict resource access.
- Compromise protection
  - Data confidentiality Protects data from being accessed in an unauthorized method during transmission. Mechanisms include access controls, encryption,and physical protection of cables.
  - Traffic flow confidentiality Ensures that unauthorized entities are not aware of routing information or frequency of communication via traffic analysis. Mechanisms include padding messages, sending noise, or sending false messages.
  - Selective routing Routes messages in a way to avoid specific threats.Mechanisms include network configuration and routing tables.

0.4.2 Information Technology Security Evaluation Criteria (ITSEC)

Overview

- As TCSEC was developed by US, ITSEC was developed by the EU to address all the security evaluation issues.
- The ITSEC had two main evaluation attributes
  - Functionality- When the functionality of a system's protection mechanisms is being evaluated, the services that are provided to the subjects like access control mechanisms, auditing, authentication, and so on are examined and measured.
  - Assurance- Assurance, is the degree of confidence in the protection mechanisms,and their effectiveness and capability to perform consistently. Assurance is generally tested by examining development practices, documentation, configuration management, and testing mechanisms.

Evaluation Criteria

ITSEC had 10 classes F1 to F10 to evaluate the functional requirements and 7 classes E0 to E6 to evaluate the assurance requirements.

- Security functional requirements
  - F00:Identification and authentication
  - F01:Audit
  - F02:Resource utilization
  - F03:Trusted paths/channels
  - F04:User data protection
  - F05:Security management
  - F06:Product access
  - F07:Communications
  - F08:Privacy
  - F09:Protection of the product's security functions
  - F10:Cryptographic support
- Security assurance requirements
  - E00:Guidance documents and manuals
  - E01:Configuration management
  - E02:Vulnerability assessment
  - E03:Delivery and operation
  - E04:Life-cycle support
  - E05:Assurance maintenance
  - E06:Development
  - Testing

ITSEC Ratings

- The ratings are based on effectiveness and correctness.
  - Effectiveness means that the TOE meets the security claims that the vendor has specified. This analysis looks at the construction and operational vulnerabilities and the ease of use, to ensure that the proper security settings do not get in the way of productivity.- rates functionality
  - Correctness deals with how the TOE was built and implementation issues. This type of analysis looks at the architectural design, how the security mechanisms enforce the policy, and the operational documentation and environment.- rates assurance.

TCSEC vs ITSEC

- TCSEC bundles functionality and assurance into one rating, whereas ITSEC evaluates these two attributes separately.
- ITSEC provides more flexibility than TCSEC.
- ITSEC addresses integrity, availability, and confidentiality whereas TCSEC addresses only confidentiality.
- ITSEC also addresses networked systems, whereas TCSEC deals with stand-alone systems.

0.4.3 Common Criteria

Overview

- Common criteria was the result of ISO's attempt to gather several organizations to come together to combine and align existing and emerging evaluation criteria like TCSEC, ITSEC, Canadian Trusted Computer Product Evaluation Criteria [CTCPEC], and the Federal Criteria.
- It was developed through a collaboration among national security standards organizations within the United States, Canada, France, Germany, the United Kingdom, and the Netherlands.
- Under the Common Criteria model, an evaluation is carried out on a product and is assigned an Evaluation Assurance Level (EAL)

Benefits of the CC

- Helps consumers by reducing the complexity of the ratings and eliminating the need to understand the definition and meaning of different ratings within various evaluation schemes.
- Helps manufacturers because now they can build to one specific set of requirements if they want to sell their products internationally, instead of having to meet several different ratings with varying rules and requirements.

CC Assurance Levels

- The Common Criteria has seven assurance levels which ranges from EAL1(lowest), where functionality testing takes place, through EAL7(highest), where thorough testing is performed and the system design is verified.
  - EAL 1 Functionally tested
  - EAL 2 Structurally tested
  - EAL 3 Methodically tested and checked
  - EAL 4 Methodically designed, tested, and reviewed
  - EAL 5 Semi-formally designed and tested
  - EAL 6 Semi-formally verified design and tested
  - EAL 7 Formally verified design and tested

Protection Profile

- A protection profile is a mechanism that is used by CC in its evaluation process to describe a real-world need of a product that is not currently on the market.
- Protection Profile Characteristics
  - Contains the set of security requirements, their meaning and reasoning, and the corresponding EAL rating that the intended product will require.
  - Describes the environmental assumptions, the objectives, and the functional and assurance level expectations. Each relevant threat is listed along with how it is to be controlled by specific objectives.
  - Justifies the assurance level and requirements for the strength of each protection mechanism.
  - Provides a means for a consumer, or others, to identify specific security needs; this is the security problem that is to be conquered
  - Provide the necessary goals and protection mechanisms to achieve the necessary level of security and a list of the things that can go wrong during the type of system development.
- Protection Profile Sections

- *Descriptive elements* Provides the name of the profile and a description of the security problem that is to be solved.
- *Rationale* Justifies the profile and gives a more detailed description of the real-world problem to be solved. The environment, usage assumptions, and threats are illustrated along with guidance on the security policies that can be supported by products and systems that conform to this profile.
- *Functional requirements* Establishes a protection boundary, meaning the threats or compromises that are within this boundary to be countered. The product or system must enforce the boundary established in this section.
- *Development assurance requirements* Identifies the specific requirements that the product or system must meet during the development phases, from design to implementation.
- *Evaluation assurance requirements* Establishes the type and intensity of the evaluation.

CC Components

- Protection profile- Description of needed security solution.
- Target of evaluation- Product proposed to provide needed security solution.
- Security target- Vendor's written explanation of the security functionality and assurance mechanisms that meet the needed security solution; in other words, "This is what our product does and how it does it."
- Packages—EALs- Functional and assurance requirements are bundled into packages for reuse. This component describes what must be met to achieve specific EAL ratings.

0.4.4 Certification and Accreditation

Certification

- Certification is the comprehensive technical evaluation of the security components and their compliance for the purpose of accreditation.
- A certification process may use safeguard evaluation, risk analysis, verification, testing, and auditing techniques to assess the appropriateness of a specific system.
- The goal of a certification process is to ensure that a system, product, or network is right for the customer's purposes.

Accreditation

- Accreditation is the formal acceptance of the adequacy of a system's overall security and functionality by management.
- Certification is a technical review that assesses the security mechanisms and evaluates their effectiveness. Accreditation is management's official acceptance of the information in the certification process findings.

1 Contributors

Edits	User
1	Adrignola[1]
5	QuiteUnusual[2]

1 http://en.wikibooks.org/wiki/User:Adrignola
2 http://en.wikibooks.org/wiki/User:QuiteUnusual

List of Figures

- GFDL: Gnu Free Documentation License. http://www.gnu.org/licenses/fdl.html
- cc-by-sa-3.0: Creative Commons Attribution ShareAlike 3.0 License. http://creativecommons.org/licenses/by-sa/3.0/
- cc-by-sa-2.5: Creative Commons Attribution ShareAlike 2.5 License. http://creativecommons.org/licenses/by-sa/2.5/
- cc-by-sa-2.0: Creative Commons Attribution ShareAlike 2.0 License. http://creativecommons.org/licenses/by-sa/2.0/
- cc-by-sa-1.0: Creative Commons Attribution ShareAlike 1.0 License. http://creativecommons.org/licenses/by-sa/1.0/
- cc-by-2.0: Creative Commons Attribution 2.0 License. http://creativecommons.org/licenses/by/2.0/
- cc-by-2.0: Creative Commons Attribution 2.0 License. http://creativecommons.org/licenses/by/2.0/deed.en
- cc-by-2.5: Creative Commons Attribution 2.5 License. http://creativecommons.org/licenses/by/2.5/deed.en
- cc-by-3.0: Creative Commons Attribution 3.0 License. http://creativecommons.org/licenses/by/3.0/deed.en
- GPL: GNU General Public License. http://www.gnu.org/licenses/gpl-2.0.txt
- LGPL: GNU Lesser General Public License. http://www.gnu.org/licenses/lgpl.html
- PD: This image is in the public domain.
- ATTR: The copyright holder of this file allows anyone to use it for any purpose, provided that the copyright holder is properly attributed. Redistribution, derivative work, commercial use, and all other use is permitted.
- EURO: This is the common (reverse) face of a euro coin. The copyright on the design of the common face of the euro coins belongs to the European Commission. Authorised is reproduction in a format without relief (drawings, paintings, films) provided they are not detrimental to the image of the euro.
- LFK: Lizenz Freie Kunst. http://artlibre.org/licence/lal/de
- CFR: Copyright free use.

- EPL: Eclipse Public License. `http://www.eclipse.org/org/documents/epl-v10.php`

Copies of the GPL, the LGPL as well as a GFDL are included in chapter Licenses[3]. Please note that images in the public domain do not require attribution. You may click on the image numbers in the following table to open the webpage of the images in your webbrower.

3 Chapter 2 on page 29

2 Licenses

2.1 GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed. Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow. TERMS AND CONDITIONS 0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion. 1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work. 2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary. 3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures. 4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee. 5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

* a) The work must carry prominent notices stating that you modified it, and giving a relevant date. * b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices". * c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it. * d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate. 6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

* a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange. * b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge. * c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b. * d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements. * e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying. 7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

* a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or * b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or * c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or * d) Limiting the use for publicity purposes of names of licensors or authors of the material; or * e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or * f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way. 8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10. 9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so. 10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it. 11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law. 12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy

both those terms and this License would be to refrain entirely from conveying the Program. 13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such. 14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version. 15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION. 16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. 17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author> This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

2.2 GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed. 0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference. 1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License. 2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies. 3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document. 4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

* A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission. * B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement. * C. State on the Title page the name of the publisher of the Modified Version, as the publisher. * D. Preserve all the copyright notices of the Document. * E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices. * F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below. * G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice. * H. Include an unaltered copy of this License. * I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence. * J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission. * K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein. * L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles. * M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version. * N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section. * O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version. 5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements". 6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document. 7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate. 8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title. 9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it. 10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document. 11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing. ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (C) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with ... Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

2.3 GNU Lesser General Public License

GNU LESSER GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below. 0. Additional Definitions.

As used herein, “this License” refers to version 3 of the GNU Lesser General Public License, and the “GNU GPL” refers to version 3 of the GNU General Public License.

“The Library” refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An “Application” is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A “Combined Work” is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the “Linked Version”.

The “Minimal Corresponding Source” for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The “Corresponding Application Code” for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work. 1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL. 2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

* a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or * b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

* a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License. * b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

* a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License. * b) Accompany the Combined Work with a copy of the GNU GPL and this license document. * c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document. * d) Do one of the following: o 0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source. o 1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version. * e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

* a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License. * b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.