DEPARTMENT OF COMPUTER ENGINEERING
UNIVERSITY OF PERADENIYA

CO327: Operating Systems - 2020

E/15/271

Assignment 1:

1. We have stressed the need for an operating system to make efficient use of the computing hardware. When is it appropriate for the operating system to forsake this principle and to "waste" resources? Why is such a system not really wasteful?

   Single-user systems should maximize use of the system for the user. A GUI might “waste” CPU cycles, but it optimizes the user’s interaction with the system.

2. What is the main difficulty that a programmer must overcome in writing an operating system for a real-time environment?

   The main difficulty is keeping the operating system within the fixed time constraints of a real-time system. If the system does not complete a task in a certain time frame, it may cause a breakdown of the entire system it is running. Therefore when writing an operating system for a real-time system, the writer must be sure that his scheduling schemes don’t allow response time to exceed the time constraint.

3. How does the distinction between kernel mode and user mode function as a rudimentary form of protection (security) system?

   The distinction between kernel mode and user mode provides a rudi- mentary form of protection in the following manner. Certain instructions could be executed only when the CPU is in kernel mode. Similarly, hard- ware devices could be accessed only when the program is executing in kernel mode. Control over when interrupts could be enabled or disabled is also possible only when the CPU is in kernel mode. Consequently, the CPU has very limited capability when executing in user mode, thereby enforcing protection of critical resources.

4. Some early computers protected the operating system by placing it in a memory partition that could not be modified by either the user job or the operating system itself. Describe two difficulties that you think could arise with such a scheme.

   The data required by the operating system (passwords, access controls, accounting information, and so on) would have to be stored in or passed through unprotected memory and thus be accessible to unauthorized users.

5. Give two reasons why caches are useful. What problems do they solve? What problems do they cause? If a cache can be made as large as the device for which it is caching (for instance, a cache as large as a disk), why not make it that large and eliminate the device?

   Caches are useful when two or more components need to exchange data, and the components perform transfers at differing speeds. Caches solve the transfer problem by providing a buffer of intermediate speed between the components. If the fast device finds the data it needs in the cache, it need not wait for the slower device. The data in the cache must be kept consistent with the data in the components. If a component has a data value change, and the datum is also in the cache, the cache must also be updated. This is especially a problem on multiprocessor systems where more than one process may be accessing a datum. A component may be eliminated by an equal-sized cache, but only if: (a) the cache and the component have equivalent state-saving capacity (that is, if the component retains its data when electricity is removed, the cache must

6. In a multiprogramming and time-sharing environment, several users share the sys- tem simultaneously. This situation can result in various security problems.

a) What are two such problems?

- accessing (reading/modifying/creating) other users files
- control of other users processes
- fraudulent use of the computer (using others accounts)
- denying others access to their data, processes, or the computer

b) Can we ensure the same degree of security in a time-shared machine as in a dedicated machine? Explain your answer.

A multiuser operating system includes features, often enhanced by hardware in the CPU, that stops regular user processes gaining access to system resources directly. Thus, one user cannot access another user's resources unless the user deliberately allowed them access, or made a mistake. From time to time vulnerabilities are found in the OS which allow a privilege-escalation exploit - a regular user can, using a special program or method, gain privileged or "root" access and then access any users unencrypted resources. These vulnerabilities are usually quickly patched when discovered, so that by keeping the operating system up-to-date with security patches, the multiuser system is kept secure.

In practice, nowadays, a big multiuser machine like at Rackspace or Amazon is likely to be much more secure than a dedicated machine running process control, because no-one ever bothers to do updates on dedicated machines and the requirement for stable uninterrupted operation trumps the security needs for version changes and reboots.

7. Describe the differences between symmetric and asymmetric multiprocessing. What are three advantages and one disadvantage of multiprocessor systems?

Symmetric Multiprocessing system: in this case each processor runs an identical copy of the OS, and hence they can communicate with each other as needed. Example: all modern OS(windows NT, UNIX, LINUX, windows 7,10).

Asymmetric Multiprocessing system: master-slave concept. A master processor controls the system, the other processor either look to the master for instruction or have predefined task assigned . Example SunOS v4. In asymmetric multiprocessing the master processor controls the system. The other processes look to the master for instruction or have predefined tasks. In asymmetric processing, all processors are peers; each processor performs all tasks.

Advantages of multiprocessor systems:

1. Increased throughput
2. economy of scale
3. increased reliability
4. less expensive than multiple single-processor systems

Disadvantages:

1. multiprocessor systems require increased I/O control to ensure that data reaches the right processor
2. common computer bus, clock , memory and peripheral devices.
3. cost is more

8. How are network computers different from traditional personal computers? Describe some usage scenarios in which it is advantageous to use network computers.

A network computer is also called a thin client. These are terminals which can implement web based computing. In order to fulfil its computational needs it is heavily dependent on other server. Example: Remote Desktop Services Tradition computers on the other hand are standalone systems which has its own CPU and all the computational needs can be fulfilled by the system alone. Example: Personal computer used at homes. Network computers are used in following places. 1. Most of the financial firms outsource their I.T operations to other companies. They establish a special area called ODC (Offshore development centers). In ODC thin clients are used to connect to the onsite servers. In this way data security can be maintained as all the data is maintained on the server and user has no way to copy the data on a local computer in an unauthorized manner. Even if the thin client is stolen the data is safe on the onsite server. 2. It also provides hardware resource optimization as the cost of cable, buses and I/O can be minimized by this approach and also the processing power can be utilized by the user session that needs it the most. 3. Software maintenance can be reduced by using network computer as all the software patches and updates and OS migration can be

rolled out for all users in one go.

9. What is the purpose of interrupts? How does an interrupt differ from a trap? Can traps be generated intentionally by a user program? If so, for what purpose?

In real-time computing operating systems are commonly interrupt-driven, which allows computers to execute multiple tasks. Some signals are sent to the processor telling it to interrupt the execution of a current task or activity and execute a new one even if the first task is not finished yet. The CPU would then save the state of the current task being processed, suspending its execution and then deal with the new event by executing a special function named interruption handler. After this new event is dealt with, the CPU will continue executing the suspended task. These particular signals which are sent to the processor and indicate an event needing immediate attention are named interrupts. These interrupts could be caused by

- Software → In this case they are caused by exceptional conditions or special instructions which need to be dealt by immediately from the processor. Invalid memory access, or division by zero are two examples.
- Hardware → In this case the signal could be sent either by external peripheral devices (game controller, mouse, keyboard) or by the parts of the computer itself such as a disk controller or primary memory. Mouse movements or keyboard presses are examples of hardware interrupts.

So interrupts are simply signals sent to the processor by either software or hardware which send the CPU into a high-priority condition where the current code executed is suspended and some new instructions will be executed depending on the interruption type. In other words it transfers program control from some address to another based on some event happening.

10. Direct memory access is used for high-speed I/O devices in order to avoid increasing the CPU's execution load.

a) How does the CPU interface with the device to coordinate the transfer?

The device controller transfers an entire block of data directly to or from its own buffer storage to memory, with no intervention by the CPU.

b) How does the CPU know when the memory operations are complete?

One interrupt is generated per block, to tell the device driver that the operation has completed.

c) The CPU is allowed to execute other programs while the DMA controller is trans- ferring data. Does this process interfere with the execution of the user pro- grams? If so, describe what forms of interference are caused.

Yes. Once DMA processes are done, an interrupt is generated which add so the main processors load. Thus, affecting the sequence of queuing process

11. Some computer systems do not provide a privileged mode of operation in hardware. Is it possible to construct a secure operating system for these computer systems? Give arguments both that it is and that it is not possible.

Theoretically, this can be done by running only one process at a time and carefully monitoring it and inspecting its output. However, this surely affects the efficiency of the system (it will run much more slowly), and we must make sure to know the priority of each process somehow (which is rather hard if we do not have modes).

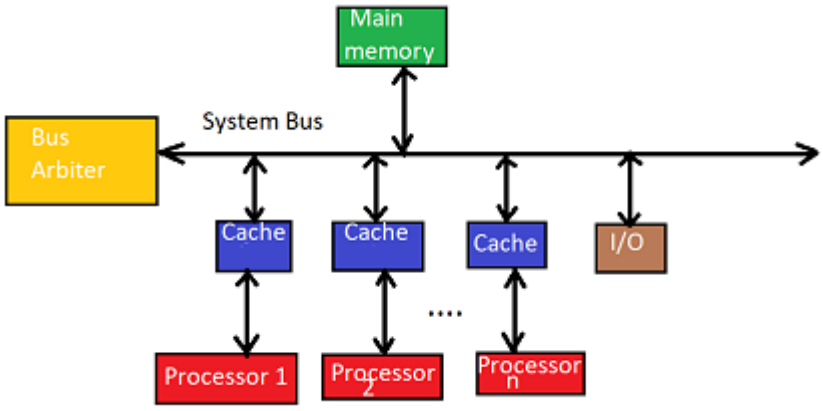
An operating system for a machine of this type would need to remain in control (or monitor mode) at all times. This could be accomplished by two methods: a. Software interpretation of all user programs (like some BASIC, Java, and LISP systems, for example). The software interpreter would provide, in software, what the hardware does not provide. b. Require meant that all programs be written in high-level languages so that all object code is compiler-produced. The compiler would generate (either in-line or by function calls) the protection checks that the hardware is missing.

12. Many SMP systems have different levels of caches; one level is local to each process- ing core, and another level is shared among all processing cores. Why are caching systems designed this way?

SMP System:

The SMP stands for Symmetric Multi-Processing is an architecture of computer hardware and software that is connected to make a single processor with two or more identical processors.

The main reason for the caching systems is designed in a different level of caches because, in different SMP systems, the different levels of caches are implemented on both accessing the speed and the size. Moreover, if the cache is closer to the central processing unit then the access will become faster. Therefore, smaller and faster caches are installed to each CPU to process each and every core. These faster caches are classically much more costly. The shared caches that are larger & still slower, are shared among various distinct processors to process all the cores.



13. Describe a mechanism for enforcing memory protection in order to prevent a pro- gram from modifying the memory associated with other programs.

Security makes sure that a system is well defended from internal and external attacks. It makes sure that access restrictions are well implemented to protect the whole system

The processor could keep track of what locations are associated with each process and limit access to locations that are outside of a program's extent. Information regarding the extent of a program's memory could be maintained by using base and limits registers and by performing a check for every memory access.

14. Identify several advantages and several disadvantages of open-source operating sys- tems. Include the types of people who would find each aspect to be an advantage or a disadvantage.

Advantages

The user enjoys open-source software free of charge, along with an active community of volunteer developers who help maintain it.

Open source software is considered safer because of the availability of the source code for anyone to analyse and debug.

By making their software open source, companies can benefit from increased usage and feedback, as well as an improved public image.

Disadvantages

Companies are likely to lose money from the lack of sales of software licenses.

Some open source communities are underserved, leaving the software dilapidated.