***Practical No. : 1***

***Aim:*** To study introduction to the distributed operating system.

***Theory:*** A distributed system is a network that consists of autonomous computer that are connected of autonomous computer using a distribution moderate. They help in sharing different resources and capabilities to provide users with single and integrated coherent networks.

**Models of distributed system :**

1. **Mini computer model:** The minicomputer model  is a simple extension of the centralized time-sharing system. A distributed computing system based on this model consists of a few minicomputers (they may be large supercomputers as well) interconnected by a communication network. Each minicomputer usually has multiple users simultaneously logged on to it. For this, several interactive terminals are connected to each minicomputer. Each user is logged on to one specific minicomputer, with remote access to other minicomputers. The network allows a user to access remote resources that are available on some machine other than the one on to which the user is currently logged. The minicomputer model may be used when resource sharing (such as sharing of information databases of different types, with each type of database located on a different machine) with remote users is desired. The early ARPAnet is an example of a distributed computing system based on the minicomputer model.
2. **Workstation model:** A distributed computing system based on the workstation model consists of several workstations interconnected by a communication network. An organization may have several workstations located throughout a building or campus, each workstation equipped with its own disk and serving as a single-user computer. It has been often found that in such an environment, at any one time a significant proportion of the workstations are idle (not being used), resulting in the waste of large amounts of CPU time. Therefore, the idea of the workstation model is to interconnect all these workstations by a high-speed LAN so that idle workstations may be used to process jobs of users who are logged onto other workstations and do not have sufficient processing power at their own workstations to get their jobs processed efficiently.  jobs of users who are logged onto other workstations and do not have sufficient processing power at their own workstations to get their jobs processed efficiently.
3. **Workstation server model:** The workstation model is a network of personal workstations, each with its own disk and a local file system. A workstation with its own local disk is usually called a diskful workstation and a workstation without a local disk is called a diskless workstation. With the proliferation of high-speed networks, diskless workstations have become more popular in network environments than diskful workstations, making the workstation-server model more popular than the workstation model for building distributed computing systems.A distributed computing system based on the workstation-server model consists of a few minicomputers and several workstations (most of which are diskless, but a few of which may be diskful) interconnected by a communication network.
4. **Processor pool model:** The processor-pool model is based on the observation that most of the time a user does not need any computing power but once in a while the user may need a very large amount of computing power for a short time (e.g., when recompiling a program consisting of a large number of files after changing a basic shared declaration). Therefore, unlike the workstation-server model in which a processor is allocated to each user, in the processor-pool model the processors are pooled together to be shared by the users as needed. The pool of processors consists of a large number of microcomputers and minicomputers attached to the network. Each processor in the pool has its own memory to load and run a system program or an application program of the distributed computing system
5. **Hybrids model:** Out of the four models described above, the workstation-server model, is the most widely used model for building distributed computing systems. This is because a large number of computer users only perform simple interactive tasks such as editing jobs, sending electronic mails, and executing small programs. The workstation-server model is ideal for such simple usage. However, in a working environment that has groups of users who often perform jobs needing massive computation, the processor-pool model is more attractive and suitable.

***Distributed operating system:***

Distributed operating system is a software over a collection of independent networks communicating and physically separate computational nodes.

A distributed operating system is a program that manages the resources of computers system and provide users an easy and friendly interface to operate the system.

***Issuse of distributed operating system:***

1. **Global knowledge:**
2. **Naming:** A name refers to an object such as computer, printer, file, a service etc. One of the service is naming service which is implemented using look up procedure. In table look up procedure, tables or directories are used, which contains the logical name. Such directories can be implemented in two ways.
3. **Scalability:** Scalability of the system should remain efficient even with a significant increase in the number of users and resources connected.
   1. **Compatibility:** Compatibility refers to inter-operability among resources that is a resource can be used from any computer or it can be used in combination with other resource. There are three levels of compatibility.
   2. **Binary Level:** A system is binary level compatable, if all machines are executing same instruction set that is all of them have same architecture. Most of the operating system functionalities are dependent on underline machine architecture & since architecture is same for all machines so operating system codes will be easy to implement.
   3. **Execution Level:** A system is having execution level compatability if the same code can be compiled & executed on any machine of the system.
4. **Protocol Level: :** A system is having protocol level compatability, if all machines are sharing same set of rules. It is only due to protocol level compatibility, machines running on different operating system is possible.
5. **Process syncronisation:** Processes running on different machines must access the shared resource in a mutually exclusive manner. In such system, a process can request or release resources at any timeand in any order which may lead to deadlock. Such deadlocks must be detected as early as possible otherwise system performance will degrade.
6. **Resource Management:** In such system, a user must access the remote resources with as much ease as it can access the local resources. Resources can be made available in three ways:
   1. **Data Migration:** In this approach data is transferred from its source to the location of computation and if any modification is made in the data then it is also reflected at the source side.
   2. **Computational Migration:** In this approach, computation itself is migrated to the location where required data is present.
   3. **Distributed Scheduling:**
7. **Security:** Security of information system has three components Confidentially, integrity and availability. Encryption protects shared resources, keeps sensitive information secrets when transmitted.

Two main aspect of security are:

1)**Authorization:** it basically involves assigning access rights to the users regarding resources.

**2)Authentication**: it involves the checking of access rights of users while accessing a particular resource.

**8) Structuring:** The structuring of distributed operating system can be performed in three ways:

* 1. **Monolithic Kernel structure:** Distributed operating system is designed in the form of a monolithic kernel where the kernel contains all the services provided by operating system. A copy of huge kernel is running on all the machines of the system. The limitation of this approach is that most of the machines will not require most of the services but the kernel is still providing it.
  2. **Collection kernel structure:** Operating system is designed as a collection of independent processes where each process represents some service such as distributed scheduling, distributed file system etc. The kernel consists of a nucleus of operating system called micro kernel which is installed on all the machines and provides basic functionalities. The micro kernel also provides interaction between services running on different machines example Galaxy, V-Kernel.
  3. **Object oriented operating system:** All services of operating system are implemented in the form of objects. Each object encapsulate a data structure and also a set of operations for those data structure example Amoeba, CLOUDS.

**Advantages of distributed operating systems:-**

* Give more performance than single system
* If one pc in distributed system malfunction or corrupts then other node or pc will take care of
* More resources can be added easily
* Resources like printers can be shared on multiple pc’s

**Disadvantages of distributed operating systems:-**

* Security problem due to sharing
* Some messages can be lost in the network system
* Bandwidth is another problem if there is large data then all network wires to be replaced which tends to become expensive
* Overloading is another problem in distributed operating systems
* If there is a database connected on local system and many users accessing that database through remote or distributed way then performance become slow
* The databases in network operating is difficult to administrate then single user system

***Conclusion:*** We have studied the distributed system, properties of distributed system, model of distributed system, we have also studies issues in distributed operating system with advantages and disadvantage.