NAME: **YINSOKNARA**

TP 1: Introduction to Distributed Systems

1. **What's distributed system?**
   A distributed system is a collection of autonomous computer systems that are physically separated but are connected by a centralized computer network that is equipped with distributed system software. The autonomous computers will communicate among each system by sharing resources and files and performing the tasks assigned to them.
   For example:
   When a user enters a web address in their browser, the browser sends a request to the server that hosts the website. The server processes the request and sends back a response, which may include the HTML, CSS, JavaScript, and other files needed to display the web page.
2. **What are the characteristics of distributed systems? Explain.**
   The characteristics of distributed systems are:
   - Concurrency: Distributed systems allow multiple processes to execute simultaneously across different machines. This concurrency makes it possible to achieve high levels of performance and scalability.
   - Lack of a global clock: In a distributed system, each node has its own clock, and there is no global clock that synchronizes all the nodes. This lack of a global clock can lead to challenges in ensuring consistency and accuracy of data across the system.
   - Independent failure: Each node in a distributed system is autonomous and can fail independently of the others. This means that the system must be designed to tolerate failures and continue to function even if some nodes are offline or malfunctioning.
   - Heterogeneity: Distributed systems often involve nodes with different hardware, software, and network configurations. This heterogeneity can add complexity to the system and require additional efforts to ensure interoperability and compatibility.
   - Scalability: Distributed systems are designed to scale horizontally by adding more nodes to the system as needed. This allows the system to handle larger workloads and accommodate growth in users or data.
   - Security: Distributed systems require robust security measures to protect against malicious attacks and ensure the privacy and integrity of data across the system.
3. **List down the 8 challenges of distributed systems? Explain.**
   - Heterogeneity of components: Distributed systems may be composed of different hardware and software platforms, making it difficult to ensure compatibility and interoperability between different nodes.
   - Openness: It is concerned with extensions and improvements in the system (i.e., how openly the software is developed and shared with others).
   - Security: Distributed systems can be vulnerable to attacks, such as distributed denial of service (DDoS) attacks, data breaches, and unauthorized access. The system must be designed to ensure data confidentiality, integrity, and availability
   - Scalability: The ability to grow as the size of the workload increases is an essential feature of distributed systems, accomplished by adding additional processing units or nodes to the network as needed.
   - Failure handling: In a distributed system, nodes can fail due to hardware or software issues, network problems, or other factors. The system must be designed to continue functioning even if some nodes fail, without compromising data consistency and availability.
   - Concurrency of components: It is naturally present in the distributed systems.

- Transparency: It is the ability of the system to hide its internal workings from the users and to present a uniform interface to the users.
- Quality of service: Checking the specific of the service of system.

4. **What are the disadvantages of distributed systems? Explain.**

The disadvantages of distributed systems are:

- Security risks: The distributed system has multiple devices, servers, databases, and connections that can easily raise the possibility of security breaches and issues.
- Difficulty in providing adequate security: It is difficult to provide adequate security in distributed systems because the nodes as well as the connections need to be secured.
- Overloading: The system can become overloaded if too many requests are made at once.
- Complexity: Distributed systems are more complex than centralized systems.
- Network errors: Network errors can cause problems in distributed systems.
- Data integration: Data integration can be difficult in distributed systems.

5. **What are the 8 forms of transparency in distributed systems? Explain.**

The 8 forms of transparency in distributed systems are:

- Access transparency: Enables local and remote resources to be accessed using identical operations.
- Location transparency: Permits access to resources regardless of their physical or network location.
- Concurrency transparency: Enables several processes to operate concurrently on shared resources without interference.
- Replication transparency: Enables multiple instances of resources to be used to increase reliability and performance without the client being aware of the replication.
- Failure transparency: Enables the concealment of faults, allowing the system to continue to operate in the presence of faults.
- Mobility transparency: Enables the movement of resources and clients within the system without affecting the operation of the system.
- Performance transparency: Enables the system to be reconfigured to improve performance without affecting the operation of the system.
- Scaling transparency: Enables the system to be expanded to handle increased load without affecting the operation of the system.

6. **Give 5 types of hardware/software resources that can be shared. Give examples of their sharing as it occurs in practice in distributed systems.**

Five types of hardware resource and five types of data or software resource that can usually be shared are printer, plotter, storage space, CD drive, DVD drive, processing power. For example, a printer takes graphics and texts from the computer and later it gets transferred into a paper which is of standard size. Another example is storage space, which is a shared resource that can be used to store data and files.

7. **What is single-point-of-failure and how can distribution help here?**

In distributed systems, the use of redundancy and replication can help mitigate the risk of SPOF. By replicating data and services across multiple nodes, the system can continue to operate even if one or more nodes fail.

8. **When will we need distributed systems? Giving your ideas.**

In my opinion I think we need distributed system when we need for high availability, scalability and reliability. For example when we need big data processing it can be more efficient to distribute the processing across multiple machines. This allows for parallel processing and can significantly reduce the time required to analyze or process the data.