Describe the weaknesses of Facebook’s privacy policies and

features. What management, organization, and technology factors have

contributed to those weaknesses

Describe the weaknesses of Facebook’s privacy policies and

features. What management, organization, and technology factors have

contributed to those weaknesses

Describe the weaknesses of Facebook’s privacy policies and

features. What management, organization, and technology factors have

contributed to those weaknesses

Describe the weaknesses of Facebook’s privacy policies and

features. What management, organization, and technology factors have

contributed to those weaknesses

Describe the weaknesses of Facebook’s privacy policies and

features. What management, organization, and technology factors have

contributed to those weaknesses

Describe the weaknesses of Facebook’s privacy policies and

features. What management, organization, and technology factors have

contributed to those weaknesses

**Infrastructure Orchestration Assignment 3**

1. What is virtual machine configuration management? How does it work?

**Virtual machine configuration management:**

Configuration Management is the process of maintaining systems, such as computer hardware and software, in the desired state. Configuration Management (CM) is also a method of ensuring that systems perform in a manner consistent with expectations over time. Originally developed in the US military and now widely used in many different kinds of systems. CM helps identify systems that need to be patched, updated, or reconfigured to meet the desired state.

**Working:**

1. The configuration management process begins with gathering information including configuration data from each application and the network topology.
2. Secrets such as encryption keys and passwords should be identified so, they can be encrypted and stored safely.
3. Once collected, configuration data should be loaded into files that become the central repository of the desired state.
4. Once data has been collected, the organization can establish a baseline configuration, which should be good enough to perform the intended operations without bugs or errors.
5. Typically, this is established by noting the configuration of the working production environment and storing those configuration settings as the baseline.
6. When the baseline has been established, the organization should adopt a version control system. Many organizations utilize Git to create a repository of configuration data for this purpose.
7. Auditing and accounting help to ensure that any changes that are applied to the configuration are reviewed and accepted by stakeholders, ensuring accountability and visibility.
8. Explain different types of load-balancing algorithms.

**Types of load-balancing algorithms:**

1. **Static Algorithm**

They are built for systems with very little variation in load. The entire traffic is divided equally between the servers in the static algorithm. This algorithm requires in-depth knowledge of server resources for better performance of the processor, which is determined at the beginning of the implementation.

However, the decision of load shifting does not depend on the current state of the system. One of the major drawbacks of static algorithms is that load balancing tasks work only after they have been created.

1. **Dynamic Algorithm**

It first finds the lightest server in the entire network and gives it priority for load balancing. This requires real-time communication with the network which can help increase the system's traffic. Here, the current state of the system is used to control the load.

The characteristic of dynamic algorithms is to make load transfer decision in the current system state. In this system, processes can move from a highly used machine to an under-utilized machine in real-time.

1. **Round Robin Algorithm**

It uses the round-robin method to assign jobs. It randomly selects the first node and assigns tasks to other nodes in a round-robin manner. This is one of the easiest methods of load balancing.

Processors assign each process circularly without defining any priority. That’s why all processes have different loading times. Therefore, some nodes may be heavily loaded, while others may remain under-utilized.

1. **Weighted Round Robin Load Balancing Algorithm**

They have been developed to enhance the most challenging issues of Round Robin Algorithms. In this algorithm, there are a specified set of weights and functions, which are distributed according to the weight values.

Processors that have a higher capacity are given a higher value. Therefore, the highest loaded servers will get more tasks. When the full load level is reached, the servers will receive a stable traffic.

1. **Opportunistic Load Balancing Algorithm**

It allows each node to be busy. It never considers the current workload of each system. Regardless of the current workload on each node, it distributes all unfinished tasks to these nodes.

The processing task will be executed slowly and it does not count the implementation time of the node, which causes some bottlenecks even when some nodes are free.

1. **Minimum To Minimum Load Balancing Algorithm**

Under this, those tasks which take minimum time to complete are selected among all the functions. According to that minimum time, the work on the machine is scheduled.

Other tasks are updated on the machine and the task is removed from that list. This process will continue till the final assignment is given. This algorithm works best where many small tasks outweigh large tasks.

1. What is ITIL Release and Deployment Management Process? Explain approaches to release and deployment management.

**ITIL (Information Technology Infrastructure Library) Release and Deployment Management Process:**

It is one of the main processes under Service Transition module of the ITIL Framework. This process is sometimes also referred as ITIL Release Management Process.

It is the process of managing, planning, scheduling and controlling the IT services, updates and releases to the production environment. Here, the "Release" means the development of a newer version of a service or component, whereas the "Deployment" means the process of integrating it into the production environment.

The Release Management part is tightly bound with Change Management and Configuration Management to ensure proper evaluation, tracking and record keeping of all releases.

The other part, the Deployment Management is controlled by the Transition Planning, Support and Service Validation as well as Testing processes for ensuring proper implementation of a release.

**Approaches to release and deployment management**

1. **Big Bang Approach:**

* In this approach, the new or changed service is deployed to all users at a single time.
* This approach is used at the time of introducing a very critical application change and when consistency of service across the organization is important.

1. **Phased Approach:**

* In this approach, services are initially deployed to a part of the user base, and if no issue observed, then the deployment is repeated for subsequent parts of the user base via a scheduled plan.
* Some scenarios such as installing new applications take this approach to minimize the impact on current operation.

1. **Push Approach:**

* This approach is used where the service component is deployed from a central location and pushed out to the target locations at a predefined time.
* An example of this is operating system patch deployment through inventory management system within an organization.

1. **Pull Approach:**

* This approach is used for software releases where the software is made available in a central location.
* Users are invited to pull the software down to their own location at their convenience.

1. **Automated Approach:**

* This approach utilizes the technology to automate releases. It helps to ensure repeatability and consistency.
* For Example, Windows update release or antivirus updates are deployed through automated approach.

1. Explain virtual private network solutions.
2. **Networking 101:**

There are two main types of networks: local area networks (LANs) and wide-area networks (WANs). LANs are confined to a single building or site, and use technologies such as Ethernet and Wi-Fi. Although these networks were once very complicated, today the common consumer can buy a simple router and have an LAN up and running in their home in a matter of minutes. WANs connect sites across multiple cities, states and even countries. These networks traditionally used fiber-optic cables and phone circuits like T1 and T3 lines. These networks had the advantage of being very reliable, but also allowed businesses to dedicate part of the connection for phone traffic and part of the connection for data traffic. While WANs can be a very good choice, these networks can be extremely expensive based on the location of the connection and the speed required.

1. **Enter the VPN:**

There are a few key differences between traditional dedicated circuits and the high-speed internet connections used by a VPN. One difference is that traditional dedicated circuits generally rated their speeds bidirectionally, meaning that if you had a 1 Mbps circuit, you could transmit and receive at the same speed. Most high-speed connections through cable and telephone companies have a much higher [download](https://www.techopedia.com/definition/394/download-dl) speed than [upload](https://www.techopedia.com/definition/1883/upload-ul). This is done because the users of these connections are spending most of their time receiving data such as email and video while sending very little data. Businesses generally do an even mix of sending and receiving data. As such, they would need to order a specially classified business connection, where both transmit and receive speeds are rated equally. he second key difference is that these connections are to the internet, as opposed to direct, point-to-point connections. Because of this, businesses that want to use these connections for their WANs would need to use something to secure their traffic and direct the traffic between business locations. This where a virtual private network comes in. VPNs can be implemented by turning on a feature included on a business's [application server](https://www.techopedia.com/definition/432/application-server) or by using a [network appliance](https://www.techopedia.com/definition/16168/virtual-private-network-appliance-vpn-appliance).

1. Explain 5-Step Azure High Availability Checklist
2. **Define Availability Requirements**

Identify the cloud workloads that require high availability and their usage patterns. Define your availability metrics. These can include:

* Percentage of Uptime
* Mean Time to Recovery (MTTR)
* Mean Time between Failures (MTBR)
* Recovery Time Objective (RTO)
* Recovery Point Objective (RPO)

1. **Plan your High Availability Infrastructure**

* **Start with a Failure Mode Analysis (FMA)**  
  Identify the types of failure you might experience, the implication of each type of failure, and recovery strategies. Based on your FMA, identify the level of redundancy required for each component.
* **Consider costs**  
  Remember that each redundant layer effectively doubles your [cloud costs](https://cloud.netapp.com/blog/low-cost-storage-options-on-azure). Ensure you have licenses and infrastructure to support the additional instances, including storage, networking and bandwidth.
* **Consider resiliency**  
  Isolate critical resources, use compensating transactions and use asynchronous operations to ensure that if a component fails, business operations can continue and be applied to a redundant component.
* **Replicate data**  
  Ensure application data is replicated in such a way that supports your redundancy strategy. It is not possible to recover if you did not replicate fresh data to the redundant component prior to failure.
* **Document everything**  
  Document the steps that should take place whether automatic or manual and recover the original component. Instructions should be short and clear enough to use in case of emergency.

1. **Perform End-to-End Testing**

To ensure reliability you should test the system under realistic failure conditions. Additional tests you can conduct to increase your level of confidence are:

* **Identify failures under load**—perform realistic load testing until a system fails, and observe how failure mechanisms behave.
* **Run disaster recovery exercises**—conduct a planned or unplanned experiment where systems go down and your team must quickly operate according to your disaster recovery plan.
* **Test health probes**—the Azure load balancer uses [health probes](https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-custom-probe-overview) to identify component failure. Test your probes to ensure they respond correctly in case of failure.
* **Test monitoring systems**—periodically check that data from monitoring systems is accurate, to ensure you can detect failure in time.

1. **Deploy Applications Consistently**

* **Any change can result in failure**  
  Having an automated, consistent deployment process can minimize the chance of errors and failures, and help you recover more easily.
* **Consider availability in your release process**  
  Design your release process to enable updates with minimum disruption of service.
* **Plan for rollback**  
  Design a rollback process that can help you automatically restore systems to a previous working version.

1. **Monitor Application Health**

Detecting failures in time is critical to high availability. Implement Azure health probes and use check functions to get fresh data about service availability.

* **Watch degrading health metrics**  
  Degrading health metrics can provide a warning signal that failure is about to happen. Create an early warning system by identifying key indicators of application health and alerting operators.
* **Watch subscription limits**  
  If you go over the allowed limits of one of your Azure services, you may experience failures. Ensure you are aware of the storage, compute, throughput and other limitations of each Azure service you use.

1. Explain High Availability and the need of high availability.

High availability is a quality of computing infrastructure that allows it to continue functioning, even when some of its components fail. This is important for mission-critical systems that cannot tolerate interruption in service, and any downtime can cause damage or result in financial loss.

Highly available systems guarantee a certain percentage of uptime—for example, a system that has 99.9% uptime will be down only 0.1% of the time—0.365 days or 8.76 hours per year. The number of “nines” is commonly used to indicate the degree of high availability. For example, “five nines” indicates a system that is up 99.999% of the time.

**The basic elements of high availability**  
The following three elements are essential to a highly available system:

* **Redundancy**—ensuring that any elements critical to system operations have an additional, redundant component that can take over in case of failure.
* **Monitoring**—collecting data from a running system and detecting when a component fails or stops responding.
* **Failover**—a mechanism that can switch automatically from the currently active component to a redundant component, if monitoring shows a failure of the active component.

**The Need of High Availability**

Downtime is the amount of time when your device (or connection) is unconscious or inaccessible. Leisure time may cause severe damage for a business, as when their devices are overloaded; all their operations are temporarily suspended. There are two kinds of rest time: scheduled and unscheduled. A planned schedule throughput is an inevitable consequence of service. This involves the application of patches, software updates, and even modifications to the data structure. After all, unplanned downtime is induced by a certain unexpected occurrence, such as system failure. It can occur because of a module's power shortages or failure. Planned schedule breakdowns are largely prohibited from estimations of achievement. The primary goal of introducing the High Availability structure is to ensure that your device or software is designed with limited to no rest time to accommodate multiple pressures and multiple vulnerabilities. There are many factors that influence you to accomplish it, and we will speak about it momentarily.

1. Explain ITIL Life cycle.

The ITIL Framework process contains the following stages: Service Strategy, Service Design, Service Transitions, Service Operations and Continual Service Improvement.

**1. Service Strategy:**

Service Strategy Operations ensure that services such as fulfilling user requests, working on service failures, fixing problems, and carrying out routine operational tasks efficiently and effectively.

Here, are important services come under this stage:

**Finance management:**

Financial Management services provide a means of understanding and controlling costs and opportunities associated with services.

**Service Portfolio Management:**

Service Portfolio Management helps you to organize the process by which services are identified, evaluated, selected, and chartered.

**Demand Management:**

Demand Management is concerned with understanding and influencing customer demand. It also involves User Profiles, which characterize various groups of users for a given service.

**2. Service Design:**

This stage ensures that agreed services are delivered when, where, and at the defined cost.

Here, are important services that come under this stage:

**Service Level Management:**

Service Level Management deals with securing and managing agreements between customers and the service provider irrespective of the level of performance and reliability associated with specific services.

Here, are important services that come under this stage:

**Availability Management:**

Availability Management service is concerned with the agreed-upon availability requirements as established in Service Level Agreements (SLA).

**Capacity Management:**

Capacity Management is focused on ensuring that at all times, the cost-effective capacity exists that meets or exceeds the demands of the business as established in Service Level Agreements

**IT Service Continuity Management:**

IT Service Continuity Management (ITSCM) process ensures that the service provider provides the minimum agreed-upon levels of service. It uses techniques like Business Impact Analysis (BIA) and Management of Risk (MOR).

**Service Catalog Management:**

The Service Catalog is a subset that contains services available to customers and users.

**3.Service Transitions:**

The goal of the Service Transition process is to build and deploy IT services. It also makes sure that changes to services and Service Management ITIL processes are conducted in a coordinated way.

**Change Management:**

Change management activity controls the lifecycle of all the changes with minimum disruption to IT services.

**Service Asset and Configuration Management:**

The goal of this service is to maintain information about configuration items needed to deliver an IT service, including their relationships.

**Release and Deployment Management:**

This process helps you to plan, schedule, and control the movement of releases to conduct testing to live environments. It also ensures that the integrity of the live environment is protected and the correct components are released.

**Transition Planning and Support:**

This ITIL process mainly focuses on planning and coordinating the use of resources to deploy a major release within the expected cost, time, and quality.

**4. Service Validation and Testing:**

This process helps to deploy releases and the resulting services are able to meet the expectations of the customer.

**Evaluation:**

The evaluation process helps you to assess major changes, like the introduction of a new service or a significant change to an existing service

**Knowledge Management:**

The objective of a knowledge management service is to gather, analyze, store, and share knowledge and information within an organization. It helps improve efficiency by reducing the need to rediscover knowledge.

**Service Operations**

This ITIL stage focuses on meeting end-users expectations while balancing costs and discovering any potential problems.

**Service Desk:**

It is the main point of contact between users and the service provider. A service desk handles communication with the users and also manages incidents and service requests.

**Incident Management:**

The objective of Incident Management is to manage the lifecycle of all incidents. It also makes sure that services are returning back to the IT service to users as fast as possible.

**5.Problem Management:**

The objective of problem management is to manage the lifecycle of all problems. It helps IT organizations to prevent incidents from happening and minimize the impact of incidents that cannot be prevented.

**Event Management:**

The object of event management is to make sure configuration items and services are continually monitored and to filter and categorize events to determine the specific actions.

**Request Fulfilment:**

The objective of request management is to fulfil service requests. In many cases, they are minor changes (for example, requests for changing a password).

**Technical Management:**

This function offers technical expertise and support for the management of the IT infrastructure.

**Application Management:**

Application Management is a service that is responsible for managing applications throughout their lifecycle.

**IT Service Operations:**

The goal is to maintain information about configuration items needed to deliver an IT service operations, including their relationships.

**Continual Service Improvement:**

It makes sure that IT services can recover and continue from a service incident. It helps to conduct business simper analysis to prioritize business recovery.