

PESUG21CS596

SINCHANA.S

J.

20/02/2024

chatgpt response points:

1. Mainly focussed on redundancy, load balancing, database replication, failover mechanisms
2. Emphasized on geographic distribution, monitoring and alerting and regular testing and maintenance
3. Load balancers such as Amazon Elastic Load Balancing or Google Cloud Load Balancing used ~~for~~ to distribute incoming traffic across redundant instances.
4. Multi region replication to replicate data across regions
5. Amazon route 53, Google Cloud DNS for DNS failures
6. CDN such as Amazon CloudFront, Google Cloud ~~CDN~~ used to cache static content
7. Regular Testing & Maintenance to ensure the effectiveness
8. Comprehensive monitoring tools like Amazon CloudWatch or Google Cloud Monitoring and also automated alerts enable proactive intervention.

References:

productive and
Referenced URL:

1. Concentrated on creating redundancy across failure domains to ensure higher availability
2. Designing multizone architecture with failure mechanisms for resilience to zonal failures.
3. Advocates for data replication across regions for disaster recovery & resilience against regional outage.
4. Degrading service level gracefully under overload & preventing / mitigating traffic spikes
5. Eliminating scalability bottlenecks by redesigning component to scale horizontally

1. Consideration of security : Measures such as encryption access control and vulnerability management.
 2. Optimizing cost : Consider cost optimization & optimizing architecture for cost effectiveness.
 3. Cache optimization
 4. Continuous development (improvement) : Continuously incorporating practices for improvement such as feedback loops
- Referenced URLs : gfg → architecture of cloud computing
javatpoint, nimplilearn youtube, cloudreco → guide to cost effective design.

1. Ensure redundancy in submission portal to make sure of the availability even during regional outage.
2. Auto scaling mechanism
3. Scalability and ensure enough load & transfer performed
4. Plagiarism checks.
5. Resource utilization & cost effectiveness (instance size)
6. Ensure compliance, data privacy and security.

chatgpt response :

Wst explanation : ~~The purpose~~

Full virtualizatr : Guest OS is completely isolated by the virtual machine from the virtualizatr layer & hardware

para virtualizatr : Guest OS is not completely isolated but is partially isolated

No.

Not required but recommended

The tools mentioned use hypervisor to create VM that runs on linux and host container. Hypervisor allows VM to access hardware of the MAC like CPU, mem disk and network.

1) Chatgpt response.

- Explanation of WSL : Compatibility layer for running linux binary executable natively on windows
- Comparison with containers
- MacOS compatibility with the linux containers (why mac can run linux effectively without additional layers)
- Utilization of the native features (like Docker desktop for mac leveraging macOS's builtin virtualization capacity)
- Cross platform compatibility

Reference paper

1. Comparisons b/n WSL1 and WSL2 covering like integration, boot time, resource footprint
2. Difference in architecture highlighting WSL2 utilization at managing VM & linux kernel
3. Exaptions where WSL1 might be preferred
4. Information on system requirements and updates for WSL2
5. Uses reference in understanding compatibility
6. Detailed explanation about the benefits and improvements introduced by WSL2.

2) Additional points:

The provided comparison covers the technical aspects

It could further drive into the practical use cases and industry adaptations of WSL v/s WSL2. Beneficial to include examples or case studies on how business or developers leverage WSL2's features for enhanced productivity.

③

WSL2:

1. Compatibility layer for running linux ^{binaries} ~~libraries~~ natively on the windows
2. Utilizes lightweight utility VM for enhanced performance
3. Provides linux kernel interface through windows enabling seamless integratr of linux tools and applications
4. Design primarily for developers seeking linux compatibility on windows
5. Focus on ~~light~~ lightweight integration and compatibility within windows

KVM.

1. Full virtualization solution for linux
2. Enables creation and management of multiple VM on host
3. Leverage hardware ~~via~~ virtualization support on modern CPU for near native performance
4. Comprehensive virtualization capabilities including full isolator and resource allocation control.
5. Suitable for system ~~etc~~ administrators and cloud infrastructure for hosting multiple guest O.S