**Unit 10 Seminar - DR Solutions Design and Review**

**Part A**

**1. Vendor Lock-In Issues**

Vendor lock-in is a significant concern in cloud environments when organisations face difficulties switching providers due to proprietary technologies, contractual limitations, or operational dependencies (Opara-Martins et al., 2014; Morrow et al., 2021).

**Main Issues Identified:**

* **Proprietary APIs and Formats:** Vendors use unique APIs or data structures that complicate migration and interoperability (Opara-Martins et al., 2014).
* **Data Portability Challenges:** Exporting data to other vendors often requires custom integrations, making the process time-consuming and expensive (Morrow et al., 2021).
* **Dependence on Vendor-Specific Ecosystems:** Reliance on a vendor’s ecosystem may reduce flexibility in adopting alternative solutions (Opara-Martins et al., 2014).

**Mitigation Strategies:**

* **Standards-Based Solutions:** Utilise open-standard technologies such as OpenStack or Kubernetes, enabling easier migration between platforms (Opara-Martins et al., 2014).
* **Multi-Cloud Strategies:** Distribute workloads across multiple providers to reduce reliance on a single vendor (Morrow et al., 2021).
* **Contractual Safeguards:** Negotiate agreements that ensure data export in a standard format (Morrow et al., 2021).
* **Use of Containers:** Employ containerisation (e.g., Docker) to enhance application portability across cloud environments (Opara-Martins et al., 2014).

**2. Security Concerns in the Modern Cloud**

Modern cloud environments introduce numerous security risks that require mitigation to maintain data integrity, confidentiality, and compliance. Key concerns include data breaches, misconfigurations, insider threats, and regulatory compliance challenges (Opara-Martins et al., 2014; Morrow et al., 2021).

**Main Concerns:**

* **Data Breaches:** Cloud environments consolidate sensitive data, making them attractive targets for cyberattacks (Morrow et al., 2021).
* **Insider Threats:** Unauthorised access by malicious or negligent personnel poses risks (Opara-Martins et al., 2014).
* **Misconfigurations:** Incorrect setups of cloud services can unintentionally expose sensitive data (Morrow et al., 2021).
* **Compliance Issues:** Adhering to regulations such as GDPR and HIPAA can be complex in shared infrastructure environments (Morrow et al., 2021).

**Mitigation Strategies:**

* **Encryption:** Encrypt data at rest and in transit to prevent unauthorized access (Opara-Martins et al., 2014).
* **Access Control Policies:** Implement robust IAM policies, including multi-factor authentication (MFA) and the principle of least privilege (Morrow et al., 2021).
* **Cloud Security Posture Management (CSPM):** Use automated tools to identify and resolve configuration vulnerabilities (Morrow et al., 2021).
* **Continuous Monitoring:** Conduct regular audits and monitor for security anomalies to ensure compliance (Opara-Martins et al., 2014).

**Part B**

**High-Level DR Solution Diagrams**

**1. RPO= 1 hr; RTO= 8 hrs; HA Required**

**Design Overview:**

* This solution uses a primary-secondary cloud architecture with automated failover for disaster recovery.
* **Description:** This scenario requires a highly resilient solution with minimal data loss and rapid recovery.
* **Diagram:**
  + **Data Replication:** Asynchronous replication to a geographically separate secondary site ensures data is up to date within the specified RPO (Opara-Martins et al., 2014).
  + **Active-Passive Cluster:** Utilize an active-passive cluster configuration with real-time replication between the primary and secondary sites.
  + **Components:** Primary site with production servers, secondary site with standby servers, high-speed replication technology (e.g., synchronous replication) using services such as AWS EBS or Azure Site Recovery, network connectivity with low latency and high bandwidth.
  + **Failover Mechanism:** Automated failover mechanism to quickly switch to the secondary site in case of a disaster at the primary site.

**2. RPO= 24 hrs; RTO = 72 hrs; HA Not Required**

**Design Overview:**

* Focus on a backup-oriented approach to store data in cost-efficient solutions like AWS Glacier or Azure Backup.
* **Diagram:**
  + **Backup and Recovery:** Backups are taken daily to ensure the RPO is met, with manual recovery planned for the RTO window (Morrow et al., 2021).
  + **Components:**
    - Full and incremental backups.
    - Manual recovery orchestration using documented procedures.
    - Cold storage to minimise costs.
  + **Recovery Process:** In case of a disaster, restore data and applications from backups at the off-site location.

**3. RPO= 5 mins; RTO= 1 hr; HA Required**

* **Description:** This scenario requires a highly available solution with minimal data loss and rapid recovery, similar to scenario 1 but with more stringent RPO requirements.
  + An active-active multi-region setup ensures zero downtime with synchronous replication between two active regions (Opara-Martins et al., 2014).
  + Real-time monitoring and automated failover ensure rapid recovery within the specified RTO.
* **Diagram:**
  + **Active-Active Cluster:** Utilise an active-active cluster configuration with synchronous replication and load balancing between primary and secondary sites.
  + **Components:**
    - Global load balancers (e.g., AWS Global Accelerator).
    - Real-time replication tools like Amazon Aurora Global Database.
    - Continuous monitoring and automated failback processes.
  + **Failover Mechanism:** Automated failover mechanism to seamlessly switch traffic to the secondary site in case of an outage at the primary site.

A diagram of data storage

Description automatically generated

**References**

Morrow, T., Prakash, S., and Johnson, L. (2021) Modern Cloud Computing: Concepts and Practices. Oxford University Press.

Opara-Martins, J., Sahandi, R., and Tian, F. (2014) ‘Critical Analysis of Vendor Lock-in and Its Impact on Cloud Computing Migration: A Business Perspective,’ Journal of Cloud Computing Advances, 3(1), pp. 35-42.