LTIMindtree

WHITEPAPER

The Art of Landing Perfect on Cloud - Part II

LTIMindtree

Contents

Executive Summary

Designing identity and access management involves centralizing user identities, using on-premises or cloud-native solutions, and implementing identity federation for seamless access. Single sign-on and third-party apps enhance access management in multi-cloud setups. High-level advantages include simplified identity management, emphasizing robust security controls to mitigate compromised identity risks.

For AWS network connectivity, considerations include VPC connectivity, connectivity between accounts and on-premises networks, and determining network bandwidth. Options like peering connections, Transit Gateway, Direct Connect, and VPNs are discussed, emphasizing trade-offs between cost and high availability.

Security controls for cloud environments involve shared responsibility, with measures like preventing changes to landing zone components, encrypting data, and adopting least-privilege principles. Detective controls, logging, and monitoring make for a strong security posture.

Governance policies cover logging, monitoring, policy compliance, and resource creation controls, ensuring operational consistency in cloud environments.

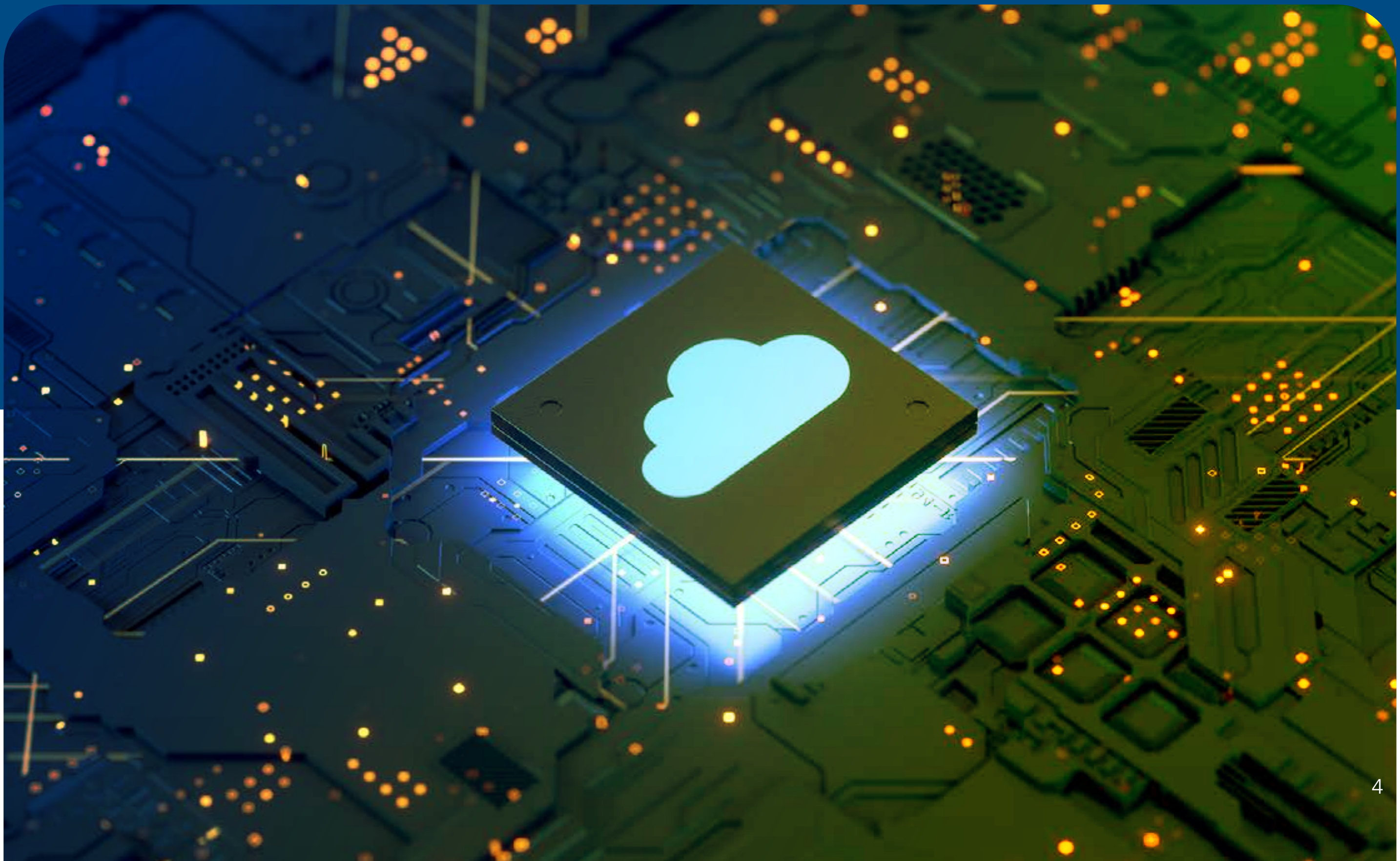
Financial controls, or FinOps, aim at optimizing cloud costs through tagging policies, instance type restrictions, and detective controls.

Landing zone deployment involves testing against requirements, migrating sample applications, and ensuring functionality, performance, and observability. Best practices include following the least-privilege principle, reserving CIDR ranges, designing network isolation, and leveraging infrastructure as code for consistency. Documenting requirements and understanding organizational needs are crucial for successful implementation.

Introduction

In the first part of this whitepaper series, we learned the importance of landing zones and the challenges encountered while planning and creating them. We also understood the stakeholders involved, which cloud platform to choose, what to deploy, and where to deploy. The paper also highlighted cloud-native and agnostic services and single and multi-account structures.

In this part, while learning more about landing zone planning, we will also look at deployment, validation, and best practices. Identity and access management is probably the first thing that comes to our minds when we think about IT security. It is the first layer of security in an IT environment. If it is accessed to carry out harmful activities, your security controls can be overridden.



How to design identity and access management?

Except for organizations born in the cloud, almost every organization has an active directory setup on-premises. It is where the identities of their employees are managed to allow access to the IT environment and applications. Re-using the same identities in the cloud makes management easy and enables a single pane of glass and centralized access management security. Even in a cloud-native organization, without an on-premises or self-managed active directory setup, a single location to manage all employee identities is preferable.

Almost every cloud provider supports integration with Microsoft Active Directory for identity federation. Identity federation facilitates setting up single sign-on, which allows using active directory identities to access resources in the cloud. You can also use the same identity federation to manage the application and database access. There are third-party applications available in the market that provide the ability to manage identity federation. These applications simplify access management across applications in public and private clouds and can be helpful in a multi-cloud or hybrid-cloud setup.

Some high-level advantages and disadvantages of a single identity setup are:

Advantages

- Only one identity to be managed per employee.
- Only one identity credential is to be remembered by employees.

Disadvantages

- A compromised identity with access to crucial data may leave you vulnerable. Therefore, having proper security controls in place is paramount.

Having different identity sources could help in testing scenarios for determining how the cloud fits into your organization's strategy. It helps create a sandbox environment isolated from the actual IT environment while carrying out the tests.

After you have finished planning for the identity source, consider these aspects of identity and access management as well:

Setting up RBAC controls

- Define a default list of job functions that you think are the bare minimum to start with and who should have those mapped.
- Your entire IAM solution must evolve to support future needs. Adding, removing, or modifying a job function should be the least destructive.

Creating default users, groups, and permissions

- Plan for any break glass users in each account along with the service accounts required.

Multi-factor authentication

- Enable and enforce multi-factor authentication wherever possible.

How to design network connectivity?

The network is another important foundational element of any IT infrastructure. You have to plan for the following to begin with:

- Network connectivity between VPC in the same AWS account
- Network connectivity between VPC in separate AWS accounts
- Network connectivity between AWS accounts and on-premises network, including the fallback option(s)
- Network bandwidth for the connectivity between cloud and on-premises network

Network connectivity between VPCs in the same or different accounts

Virtual Private Clouds (VPC) are isolated private networks designed for you in an AWS account. You may need to establish the connectivity between two VPCs so the resources can communicate with each other. For example, non-production and production connectivity or connecting resources hosted in an application VPC to a domain controller hosted in a shared services VPC and so on.

These VPCs can exist in the same or different accounts. AWS supports connecting two or more VPCs via peering connections (a logical connection between two virtual private clouds that use the cloud provider's global network backbone and are free of charge). However, this peering connection has certain limitations and challenges, such as:

01

It gets difficult to manage a large number of VPCs because of the increased number of peering connections.

02

In a one-to-one connection, there is no support for transitive routing.

03

The VPCs must have non-overlapping Classless Inter Domain Routing (CIDR) ranges.

To address these problems, AWS offers a Transit Gateway.

Network connectivity between AWS accounts and on-premises network

Network connectivity between an on-premises network and AWS is required when migrating workloads from on-premises to the cloud. It is also required for connecting workloads retained on-premises with resources on the cloud and allowing users in on-premises networks to access resources in the cloud.

All of the above tasks can be accomplished by connecting to the resources in the cloud via the public internet. However, it presents several challenges, such as increased latency, inconsistency, and very high security risks.

Therefore, every cloud provider offers services for establishing a private network connection between your on-premises and cloud network. **For example,**

01

AWS has Direct Connect and Site-to-Site VPN

02

Azure has ExpressRoute and Site-to-Site VPN

03

GCP has Cloud Interconnect and Cloud VPN

In addition to the above options, you also have services available for Point-to-Site VPN connections.

AWS Direct Connect allows you to establish a dedicated private connection between your data center and AWS account(s). Setting it up, however, requires more careful planning than the Site-to-Site VPNs and is costlier. It provides a higher bandwidth, lower latency, consistent performance, and greater control over the entire network connectivity.

Site-to-site VPNs are suitable when you are running lesser workload in the cloud and require private connectivity to the cloud. They can also be considered if in-transit data security is a concern. These VPNs allow you to establish an encrypted channel over the public internet for connecting your data center and AWS account(s). The AWS Site-to-Site VPN supports IPSec protocol. It is easier and quicker to set up and manage than Direct Connect but has its own disadvantages, like lower bandwidth, higher latency, inconsistency, and higher security risks.

Site-to-site VPNs are a good option in the following scenarios:

01

You are starting your journey on the cloud and have a limited budget or non-critical workload running in the cloud or need a quick solution for private connectivity.

02

An interim solution until the implementation of AWS Direct Connect is in place.

03

A fallback option for AWS Direct Connect if bandwidth and security risks are not a concern.

Understand the tradeoffs between cost and high availability and design a network architecture that fits your needs.

Network bandwidth for connectivity between cloud and on-premises network

Predicting the correct network bandwidth for an AWS direct connect connection is very important as it can have adverse performance effects, and the bandwidth cannot be changed for an existing connection. The cost for any changes in the existing connection is significant in terms of efforts for re-planning and execution, along with downtime requirements or the cost of running parallel connections.

Here are some parameters to determine how much network bandwidth you will require:

- Number of applications and servers expected to be migrated or hosted in the cloud in the next three to five years
- Number of file servers expected to be migrated or hosted in the cloud in the next three to five years
- Number of users expected to connect from on-premises to the applications running in the cloud
- Type of data traffic expected between cloud and on-premises platforms
- For example, application access, including web traffic and application data transfer, backup transfer and replication, AD sync and authentication, application data migration, OS/DB patches, etc.
- Type of workloads expected to migrate and be hosted in the cloud
  For example, Content servers, SAP, Digital Engineering vault (Commercially off-the-shelf (COTS) and In-house), Infrastructure applications, Databases, Virtual desktops etc.
- Largest application in terms of size to be migrated to the cloud
- Data transfer for application/database migration to the cloud
- Look at the internet bandwidth utilization reports for all the data centers if you are planning to re-use the existing internet connections for site-to-site VPN
- Size of backup and data replication transfer from on-premises to AWS, if applicable, along with a backup window

Which security controls to begin with?

the provider is responsible for the security of the cloud, the consumer is responsible for the security of the data. As we have seen in the recent past, security breaches on the cloud have increased significantly due to the lack of knowledge or enough attention.

Given the dynamics of modern IT, infrastructure security requires an ever-evolving strategy and process. However, there are certain basic guiding principles that every organization must adhere to when starting their cloud security journey.

There are some cloud security controls that every organization must have as a foundation to build on. They are:

01 Prevent changes to the foundational components of your landing zone

02 Adopt the least-privilege method for managing access for users and applications

03 Enable and configure management events logging and storage into a centralized storage

04 Disable changes to log files by unauthorized users or detect changes to those files by monitoring them

05 Enable and configure automated notification upon security events and apply auto-remediation, if possible

06 Enable and configure detective control to identify EC2 security groups allowing unauthorized access from public IP addresses or unrestricted access

07 Enable and enforce data encryption at rest and in transit

08 Deny creation of programmatic access keys for AWS root users

09 Detect users with passwords aging more than x number of days

10 Detect users with multiple programmatic access keys

11 Detect users without multi-factor authentication

12 Detect users with programmatic access keys not rotated in the last x number of days

13 Set up and configure security information, event management, vulnerability management, patch management, antivirus, advanced threat detection, cloud security posture management application and services

Which governance policies to begin with?

You should also have some governance controls for operations and management, along with security and FinOps controls. Additionally, you can also define controls on how resources should be built and operated in the cloud.

There are some cloud governance controls that every organization must have as a foundation on which to build. They are:

01 Enable logging for resources as much as possible, especially in the production environment.

02 Feed resource logs into a monitoring solution and enable alerts based on the requirements.

03 Enable andconfigure automated notification upon policy non-compliance.

04 Disable access to AWS regions where the workload is not expected to be hosted.

05 Enable and configure the AWS Config service to record all configuration items. (Helps you track the history of changes made to a particular resource along with the actual changes made)

06 Monitor activities by AWS root users. (Strongly discourage activities by AWS root users unless meant for any break glass purpose or activities that only root users can do.)

04 Set up a configuration management database using third-party or cloud-native services.

05 Deny resource creation with older hardware for better performance, security, and cost optimization.

06 Deny the creation of Internet and Network address translation (NAT) gateway by unauthorized users.

Which financial controls to begin with?

One of the reasons why organizations are adopting the cloud is the financial benefits it offers; you pay for only what you use, and there are so many ways to reduce your total cost. However, with inappropriate controls and monitoring, the cloud cost can shoot up exponentially and defeat the purpose.

It is recommended that you define a Cloud Financial Management governance framework. It must outline how financial controls and processes will be established and monitored regularly for cloud cost optimization.

To learn more about cloud cost optimization, refer to *The Right Way to Approach Cloud Cost Optimization.*

There are some financial management (FinOps) controls that every organization must have as a foundation on which to build. They are:

01

Define a tagging strategy for all resources in cloud and apply tag policies

02

Create a list of approved instance types and deny creation of resources outside of it

03

Configure a detective control to identify S3 buckets with no lifecycle management rules configured

04

Define a process to identify unused resources for cost optimization

Landing zone deployment and validation

By now, you would have completed planning and designing the essential components of a cloud landing zone. It is now time to get your hands dirty and start implementing what you have designed. With proper resources and toolsets in place, landing zone deployment should not take very long for you.

After the landing zone deployment, it is a good practice to test if it meets the actual requirements. Develop some test cases based on what you want to test and the expected result. If possible, migrate a sample non-critical application from on-premises to the cloud and validate its functionality and performance along with required observability.

Best practices for landing zone

Some of the best practices for planning and designing a cloud landing zone have already been covered in the above sections. Here are some additional ones that can be beneficial:

- Follow the least-privilege method for identity and access management.
- Discourage the creation of programmatic access keys unless absolutely necessary.
- Reserve and block a CIDR range for the cloud environment—Provides you with consistency and makes it easier to manage CIDRs and Virtual LANs (VLANs)
- Design a strategy for network isolation in the cloud, for example:

01

How many VPCs should an account have by default?

02

How many subnets, route tables, and network Access control lists (ACLs) should a VPC have by default?

03

Should there be public subnets in all the VPCs or not?

04

Should public subnets be only in a Demilitarized zone (DMZ) VPC?

- Place a network firewall between the cloud environment and the Internet.
- Leverage VPC endpoints for connectivity between resources in the cloud and cloud services endpoints.
- If possible, create golden images as per your organizational requirements and use them to provision any server in the cloud. It helps maintain consistency in deployment and reduces manual efforts.
- Start with small instance sizes and only the essential services.
- Enable and configure ‘deletion prevention’ for critical components of the landing zone.
- Document your requirements, design decisions, and the rationale behind them.

- Define a naming convention for accounts and resources to help with governance and operations in the future.

01

It is good to have the location (cloud + region), application, functionality (web/app/db), and environment in a resource name.

02

Try to keep the resource name to less than 16 characters for the resources that are expected to be joined to an AD domain. Host names longer than 16 characters are normally trimmed down when joined to the domain.

- Leverage Infrastructure as code and DevOps pipelines, if possible, for reusability with fewer chances of errors.

- Understand your requirements and design a landing zone that fits those needs.

- In the resource tagging policy, define the pre-approved values that can be used against a tag key. For the free text fields, define some guidelines to be followed.

Conclusion

Building a cloud landing zone requires balancing seamless user access with robust security and cost efficiency. Users benefit from centralized identity management with single sign-on and cloud-native/on-premises solutions. Secure network connectivity options like VPCs, peering, and Direct Connect offer varying cost-availability trade-offs. Shared responsibility dictates foundational security measures like data encryption and least privilege, while detective controls and logging build a secure posture. Governance policies and FinOps ensure operational consistency and optimized cloud spending. Finally, successful deployment involves thorough testing, sample application migration, and adherence to best practices like network isolation and infrastructure as code. Documenting requirements and understanding organizational needs are key to a successful cloud landing zone.

LTIMindtree’s SmartDeploy is an automation tool that lets you deploy resources in Multi-Cloud/Multi-Subscription environments using infrastructure as code (IaC). It has a user-friendly interface for navigating through different cloud providers and services. SmartDeploy uses customizable and reusable templates to deploy cloud resources and is useful for bulk deployment with minimum inputs, landing zone creation, or improving day-to-day operations.

Author

Gaurav Goel

Gaurav Goel is an IT professional with over 15 years of experience in Cloud Architecture, Cloud Consulting, DevOps, Infrastructure as Code, System Administration, and many other roles. He has worked with clients from various industries, such as Telecom, Retail, Oil and Gas, Utility, Manufacturing, Education, etc. Gaurav is a technology enthusiast who likes learning new technologies and sharing knowledge with others.