

Standardizing Privileged Access Architecture for Multi-Cloud

v1.0

Table of Contents

| | |
|--|-----------|
| Abstract | 4 |
| Introduction | 5 |
| Background of the Tier Model | 5 |
| Mapping the Tier Model to the Cloud | 7 |
| Defining Credential Tiering in the Cloud | 10 |
| Risks Related to Governing Multi-Cloud Infrastructure | 11 |
| Attack Scenarios | 13 |
| Scenario #1: Total Cloud Domain Compromise | 13 |
| Scenario #2: Compromise Multi-Cloud Workloads | 17 |
| Analyzing the Cloud Compromise Scenarios | 21 |
| Pass-the-Cookie Attack | 22 |
| Pass-the-Token Attack | 23 |
| Golden SAML Attack | 23 |
| Proposed Architecture Model to Secure Cloud Resources | 24 |
| Tier Model in Amazon Web Services (AWS) | 26 |
| Resource Segregation in AWS | 26 |
| Overview | 26 |
| AWS Implementation | 27 |
| Credential Tiering | 28 |
| Restrict Inheritance of Privileges | 29 |
| Restrict Inheritance of Privileges in AWS | 30 |
| Use of Managed Administrative Workstations with Restrictive Controls | 30 |
| Apply Network Segmentation and Network Security Controls | 32 |
| Identify and Classify Assets | 33 |
| Design the Segmentation Strategy | 33 |
| Implement Security Controls | 33 |
| Apply Scalable Security Configurations and Governance | 34 |
| Scalable Security Configurations in AWS Cloud | 35 |
| Monitoring and Detection | 37 |

- Tier Model in Microsoft Azure..... 40**
 - Resource Segregation in Microsoft Azure..... 40
 - Overview 40
 - Microsoft Azure Implementation..... 40
 - Credential Tiering 42
 - Restrict Inheritance of Privileges 43
 - Restrict Inheritance of Privileges in Microsoft Azure..... 44
 - Use of Managed Administrative Workstations with Restrictive Controls..... 44
 - Apply Network Segmentation and Network Security Controls 47
 - Identify and Classify Assets..... 47
 - Design the Segmentation Strategy..... 47
 - Implement Security Controls 48
 - Apply Scalable Security Configurations and Governance 48
 - Scalable Security Configurations in Microsoft Azure 50
 - Monitoring and Detection 51

- Tier Model in Google Cloud Platform (GCP)..... 54**
 - Resource Segregation in GCP 54
 - Overview 54
 - GCP Implementation 54
 - Credential Tiering 56
 - Restrict Inheritance of Privileges 57
 - Restrict Inheritance of Privileges in GCP 58
 - Use of Managed Administrative Workstations with Restrictive Controls..... 58
 - Apply Network Segmentation and Network Security Controls 60
 - Identify and Classify Assets..... 61
 - Design the Segmentation Strategy..... 61
 - Implement Security Controls 62
 - Apply Scalable Security Configurations and Governance 62
 - Scalable Security Configuration in GCP 63
 - Monitoring and Detection 65

- Protecting From the Attacks by Applying Tiering Model Practices67**
 - Scenario #1: Protect Against Total Domain Compromise..... 67
 - Scenario #2: Protect Against Multi-Cloud Compromise 76

- Conclusion 84**

- References..... 85**

Abstract

This white paper examines the risks and attack vectors inherent in hybrid multi-cloud infrastructures, where organizations utilize multiple public cloud providers. The lack of robust governance practices often exposes vulnerabilities that threat actors can exploit to steal privileged user credentials, move laterally across networks, and compromise environments.

It will analyze various attack paths observed by Mandiant in real-world multi-cloud scenarios, followed by an in-depth exploration of a proposed architecture designed to enhance the security of resource hosting and management across diverse cloud platforms.

This paper examines the core security principles of the tier model and their practical application across diverse cloud platforms. It does not offer an in-depth analysis of specific cloud vendor capabilities, as this lies beyond the paper's primary focus.

Multi-cloud environments, while offering flexibility and scalability, introduce unique security challenges due to their distributed nature and potential for lateral movement between interconnected resources. Attackers often exploit overly permissive accounts and unprotected logins to compromise entire cloud environments, including identity providers and applications.

To mitigate these risks, this paper proposes adopting a tiered security model in the cloud, aligning with the traditional on-premises approach. This model involves segregating cloud resources into three distinct tiers:

- **Tier 0 or Trusted Services Infrastructure (TSI):** Critical infrastructure, including identity servers (Domain Controllers, OAuth servers), PAM servers, and Certificate Authorities.
- **Tier 1:** Business logic layer, housing applications and sensitive business data.
- **Tier 2:** End-user workstations, help desk, and support functions.

Each tier is managed by dedicated administrators using hardened privileged access workstations (PAWs), restricting access to their respective tier's resources. This minimizes the risk of credential exposure and prevents privilege escalation attacks.

Policy and security configurations, network segregation, and continuous monitoring are implemented across all tiers to ensure consistent security and visibility. This approach aims to reduce attack surfaces, limit lateral movement, and enhance overall security posture in multi-cloud environments.

By applying the well-established tiered model to the cloud, organizations can effectively mitigate risks associated with multi-cloud deployments, such as token-based attacks and unauthorized access. This strategy provides a robust framework for securing critical assets and ensuring business continuity in today's dynamic cloud landscape.

Introduction

Modern organizations increasingly rely on diverse IT infrastructures, encompassing on-premises data centers, public cloud environments, and numerous SaaS applications. This complex landscape often leads to users holding multiple roles, both administrative and productivity-focused, exposing their credentials to various environments. This scenario presents a significant risk, as attackers can exploit these credentials to move laterally across cloud environments, escalating access and potentially compromising sensitive resources.

Organizations frequently deploy a wide range of workloads, including identity servers, domain controllers, and internet-facing applications, within cloud environments. This often results in an unintended overlap of roles and permissions, creating multiple accounts with excessive privileges across cloud platforms. Such vulnerabilities provide attackers with numerous pathways to traverse and escalate their access, ultimately jeopardizing the security of entire cloud environments.

Additionally, cloud platforms have their own individual role-based access control (RBAC) models and the configuration of the RBAC is largely driven by the respective vendor of the cloud platform and their best practices. When organizations have on-premises infrastructure and more than one cloud, it can lead to disjoint and inconsistent configurations across the platforms. By adopting a standard model based on tiering principles it brings consistency and predictability in a multi-cloud environment, by cutting down potential attacker movement paths and reducing governance overhead.

In the following sections, this paper will delve into the background of the tier model, and apply the tiering concepts to create a consistent security model for the cloud platforms. It will also explore the various risks in managing multi-cloud environments, explore the common cloud-based credential theft techniques and attacker tactics as observed by Mandiant, and finally propose an architectural model designed to mitigate these risks and enhance the security of cloud resources.

Background of the Tier Model

The concept of tiering is not new and has been widely researched and implemented by various organizations and software vendors. The following organizations and industry bodies have explicitly recommended the use of the Tier Model for server infrastructure or privileged access management, or have frameworks and guidance that implicitly align with a tiered approach:

Government & Regulatory:

- **National Institute of Standards and Technology (NIST):** NIST's cybersecurity framework¹, particularly its guidelines on risk management and access control, encourage the use of a tiered approach to prioritize assets and apply appropriate security controls.
- **The Center for Internet Security (CIS):** CIS benchmarks and controls often advocate for the segmentation of networks and data², which aligns with the concept of tiering.
- **Payment Card Industry Data Security Standard (PCI DSS):** PCI DSS³ requires the implementation of strong access control measures, including restricting access to cardholder data based on need-to-know and business justification - a principle that can be effectively achieved through tiering.

Industry Bodies:

- **Uptime Institute:** The Uptime Institute's⁴ data center site infrastructure tier standard is a well-known framework for classifying data centers based on their resiliency and redundancy levels, directly relating to the concept of tiering for server infrastructure.
- **Information Systems Audit and Control Association (ISACA):** ISACA's COBIT framework⁵ promotes a risk-based approach to IT governance, which includes the identification and classification of IT assets, another key element of tiering.

Technology Vendors:

- **Major Cloud Providers (AWS, Azure, GCP):** While not explicitly promoting a "tier model," they offer various services and features that enable tiering, such as virtual private clouds (VPCs), security groups, and identity and access management (IAM) tools. Microsoft recommends the Enterprise Access Model⁶, which aligns with the tiering approach to categorize and protect resources based on their sensitivity and criticality.
- **Cybersecurity Solution Providers:** Many security vendors, particularly those offering privileged access management (PAM) solutions, incorporate tiering into their product design and implementation guidelines.

Key Points to consider for tiering:

- **Widespread Adoption:** While not universally mandated, the tier model is a widely recognized and adopted best practice in the IT industry.
- **Flexibility:** Organizations can adapt the tier model to their specific needs and risk tolerance levels.
- **Continuous Improvement:** Tiering should be part of an ongoing process of security assessment and improvement, ensuring that access controls remain effective in the face of evolving threats.

As the concept and use of tiering is not new, some overlap with existing vendor organizations adoption of the model is expected. This paper attempts to harmonize the approach of tiering and apply them to secure a hybrid multi-cloud environment.

Mapping the Tier Model to the Cloud

To begin mapping workloads, identities and endpoints to a tier, we need to understand what the various tiers are:

- **Tier 0:** also known as the Control and Management plane or Trusted Services Infrastructure, as it hosts resources that support the access and authentication of all other entities within the environment. This would constitute identity and access management (IAM) systems, certification authorities, and any other systems that directly affect and can manage configurations of these systems.
 - This means, IAM solutions such as Google Identity, Microsoft Entra Identity, Active Directory Domain Services are all Tier 0. PKI servers issue certificates that can enable authentication for servers, workstations and users and hence PKI is Tier 0.
 - Networking solutions supporting Tier 0 services, such as SD-WAN, Security Services solutions should be managed as Tier 0 services, as their administrators will have direct access to alter the Tier 0 IAM resources hosted within those networks. Note that, for solutions such as SD-WAN depending on the vendor provided role-based access control, the management can be broken down to ensure components of the WAN that host or have direct impact on Tier 0 resources will be categorized as Tier 0. Other networks that host applications that are Tier 1 can be categorized as that tier.
 - Similarly, any Mobile Device Management (MDM) Platforms that enrolls Tier 0 devices will have direct access to make IAM configuration changes, and hence is categorized as Tier 0.
 - Any monitoring SIEM solutions collect logs and use log correlation logic to detect potential misconfigurations within all systems in the environment. As SIEM solutions will store IAM logs and can make changes to IAM systems to collect logs, managing SIEM solutions should be considered a Tier 0 operation.
- **Tier 1:** is the layer that comprises the business applications and data i.e., workloads aligned with the business. This constitutes any productivity applications, line-of-business applications and their data. Similarly, their supporting components and intermediaries follow the same tiering classifications.
- **Tier 2:** comprises the user access layer, which constitutes end user accounts, devices, helpdesk users and their assets, any virtual workstations assigned to end users and staff.

Mapping the Tier Model to the Cloud

| Tier | Also, can be referenced as | Description | Comments |
|---------------|--|--|--|
| Tier 0 | Privileged Access, Control and Management plane | <p>This tier comprises services and assets responsible for secure access. This includes the following:</p> <ul style="list-style-type: none"> • IAM services (Active Directory, SAML, OAUTH servers) • PIM/PAM solutions • Administrative Workstations and accounts • Networking • Configuration Manager and MDM • Tier 0 Monitoring resources • Virtualization infrastructure hosting any Tier 0 workloads | <p>Why are these services Tier 0/highest criticality?</p> <p>If an attacker breaches any of these services, they will be able to take over the authentication mechanism and issuing of tokens for any users and services. This would mean that an attacker can perform complete domain takeover with a Tier 0 breach.</p> |
| Tier 1 | Data or Workload plane | <p>This tier consists of resources that host applications and intellectual property of the organization</p> | <p>Why are these services Tier 1?</p> <p>If an attacker breaches a Tier 1 service, they will be able to steal business data and user data specific to that application. This could potentially lead to data theft, loss of business, and more.</p> <p>If Tier 1 and Tier 0 workloads have cross-over paths to aid lateral movement, attackers can elevate privileges and move from a Tier 1 to a Tier 0 breach, and completely take over the domain. This is what we try to avoid by restricting administrators to specific tiers, in addition to enforcing security controls to harden the tier.</p> |
| Tier 2 | Access plane | <p>This tier consists of resources that enable user and service access to applications hosted by the organization. This can include Virtual desktop environments, productivity workstations, user, customer and partner accounts</p> | <p>Why are these services Tier 2?</p> <p>If an attacker is able to compromise a Tier 2 entity, that would mean the attacker is in control of the user's credentials and can control that user account or their device.</p> <p>The rationale behind separating out Tier 2 is to ensure we don't mix productivity operations and administrative operations, as they fall within separate tiers, and could provide an attacker an opportunity to elevate privileges.</p> |

Table 1: Tier model definitions

Mapping the Tier Model to the Cloud

The primary reason towards segregating the services into tiers, is to ensure specific user identities and endpoints to access those tiers are defined correctly. This is aimed at limiting the privileged user logins into non-critical endpoints and services, to decrease the credential exposure perimeter.

Figure 1 shows how resources, identities and endpoint devices segregated into their various tiers are allowed to only retain access within their respective tier. No privilege elevation or de-elevation to log in to higher or lower tiers is permitted. This ensures that the attacker's scope is minimized to elevate access and restrict valuable credentials to get exposed through lesser restrictive environments. This is the fundamental intent of enforcing tiering.

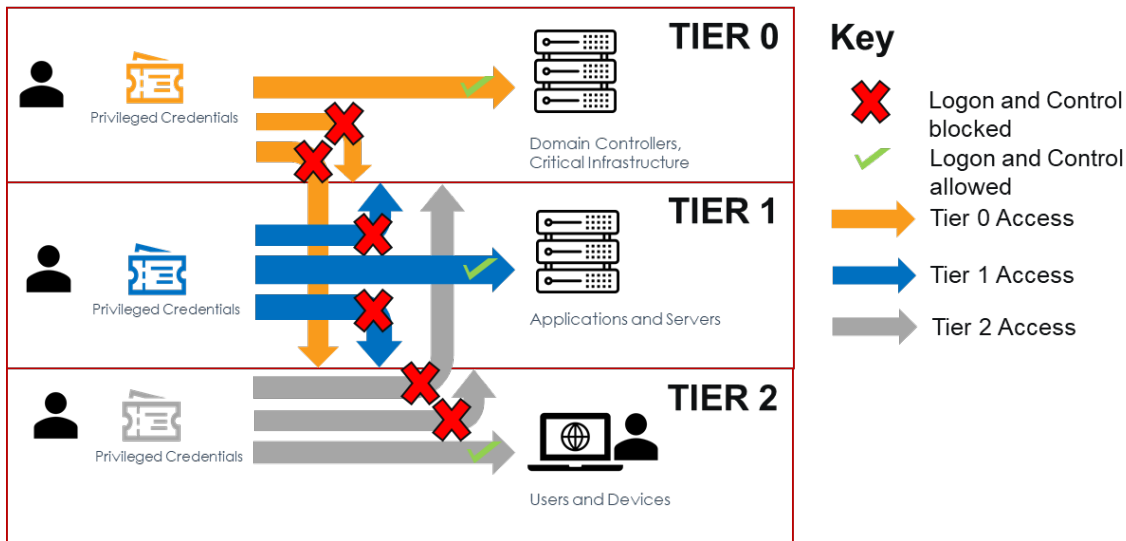


Figure 1: Tier Model visualization and rules

Subsequent sections will describe the tiers for a cloud or hybrid environment constituting one or multiple cloud platforms and on-premises server infrastructure.

Defining Credential Tiering in the Cloud

Table 2 maps different cloud operations by tier and persona.

| Tier | Persona | Action | Source |
|--------|------------------------|--|----------------------------|
| Tier 0 | IAM Administrator | I would like to update federation settings for my IAM solution | From a trusted workstation |
| Tier 0 | IAM Administrator | I would like to update IAM configurations to enable strong-factor authentications for all my users | From a trusted workstation |
| Tier 0 | IAM Administrator | I would like to create, update, delete users/groups and their respective attributes | From a trusted workstation |
| Tier 0 | IAM Administrator | I would like to reset passwords and session tokens for any identity within the IAM system. | From a trusted workstation |
| Tier 0 | MDM Administrator | I would like to define MDM policies that enroll, encrypt and secure the devices. | From a trusted workstation |
| Tier 1 | Cloud Administrator | I would like to manage Container instances that hosts an ecommerce application | From a trusted workstation |
| Tier 1 | SaaS App Administrator | Manage productivity applications that host data | From a trusted workstation |
| Tier 2 | Helpdesk Administrator | I would like to troubleshoot user access to devices/apps | From a trusted workstation |

Table 2: Tier model credentials in the cloud

If the approach from Table 2 is adopted then Security Architects can identify the actions that administrators perform and categorize those administrators and their endpoint devices to the relevant tier.

Risks Related to Governing Multi-Cloud Infrastructure

Mandiant has observed the following risks to managing multi-cloud environments, which gives rise to attackers gaining access to environments and elevating the compromise further:

- **Lack of resource segregation:** Most cloud environments often lack logical segregation to separate business critical resources, production workloads from being deployed within the same resource boundary as non-production workloads. Mandiant has observed customers deploy critical Identity servers such as Active Directory Domain Controllers, Privileged Access Management solutions within the same resource boundary as Internet facing applications, non-Production applications, and so on. This gives attackers the opportunity to compromise non-production resources and then move laterally to compromise critical resources.
- **Identity and Access Management (IAM) misconfigurations:** Most organizations have one or multiple IAM solutions to support authentication for various applications. A misconfiguration in an IAM solution can have a cascading effect in propagating an attack in a connected environment with multiple clouds. For instance, one credential compromise within a cloud, can provide an attacker foothold in the environment to then move laterally to another.

Misconfigurations can include various factors such as assigning excessive permissions to identities, not enabling multi-factor authentication for all users, unused or stale credentials, misconfigured cloud service settings and so on.

- **Credential Theft attacks:** Various types of credential theft attacks can allow attackers to steal access tokens, cookies, password hashes, and more to impersonate a user and access cloud environments. A lot of organizations will set up federation between multiple IAM solutions, such as an integrating Active Directory with Entra ID, or integrating Entra ID and Amazon Web Services IAM are common examples. This is how an attacker in possession of a user credential from AD, can reuse the same credential to access Azure and AWS.
- **Exploiting vulnerabilities in the data and management plane:** Since cloud platforms host Infrastructure-as-a Service (IaaS) and Platform-as-a Service (PaaS) resources to house applications and services, it is crucial to ensure these resources are using strong encryption methods, are up-to-date on applying patches, and use the latest frameworks. Most often this is a huge administrative overhead for organizations to apply all the necessary controls to the cloud infrastructure. This leads to attackers leveraging known CVEs associated with the platform to compromise cloud environments.
- **Abuse of trust relationships between clouds:** If an organization has more than one cloud environment, chances are those environments will share a trust relationship. These trust relationships are meant to host applications with ease and support business needs. Such trust across clouds could be through network connectivity established between the clouds through site-to-site VPN; or it could be through IAM platforms of the various clouds federated with one another.

Risks Related to Governing Multi-Cloud Infrastructure

– For example, an organization may choose to integrate their Microsoft Entra ID tenant with an OAUTH identity provider such as Google Cloud Platform. This enables users to reuse the same credentials to access both clouds. While having a single identity for a user is less operational overhead for administrators and provides ease of provisioning access to applications; this could also be misused by attackers to move across cloud environments and further the compromise if the user account is not properly secured.

Attackers can also use the network channels such as VPNs to move laterally across clouds and compromise services. This is especially common in ransomware attacks where threat actors can use various known tools to scan the resources hosted across multiple clouds and gain access, encrypt the drives and leave ransom notes.

- **Privilege escalation between multiple environments:** Consider a scenario where a user has a single domain user account provisioned in Active Directory (AD). If Active Directory is being federated with Microsoft Entra ID, the user can access Azure with the same credential as AD. The user holds elevated privileges across multiple subscriptions in the cloud. If this user account gets compromised, a threat actor can move between the environments and gain elevated permissions in Azure to compromise any critical workloads and elevate further.

Attack Scenarios

This section provides examples of attackers gaining initial access to an organization’s environment, and moving laterally between various components to elevate access and compromise cloud environments. These scenarios are based on Mandiant’s experience with threat actors and helping customers through containment, eradication and recovery for their various services.

Scenario #1: Total Cloud Domain Compromise

The first scenario explores the case of a hybrid infrastructure within an organization named Moonbeam Energy. Figure 2 shows the organization’s architecture. It uses Active Directory (AD) to manage users and computers. The Active Directory servers are hosted in on-premises data centers and Virtual Machines in Azure. The organization has a single Azure Subscription, which hosts AD Domain Controllers as Virtual Machines (VMs) and some Development/ Testing container instances.

The Active Directory serves as the identity provisioning source for all Moonbeam Energy users, and is integrated with Microsoft Entra ID to synchronize identities and password hashes to the Azure platform.

Microsoft Entra ID has a SAML integration with Google Cloud Platform (GCP) identity provider, which ensures users can use the same AD synchronized Microsoft Entra credentials to authenticate to the GCP platform. GCP hosts certain applications for this organization.

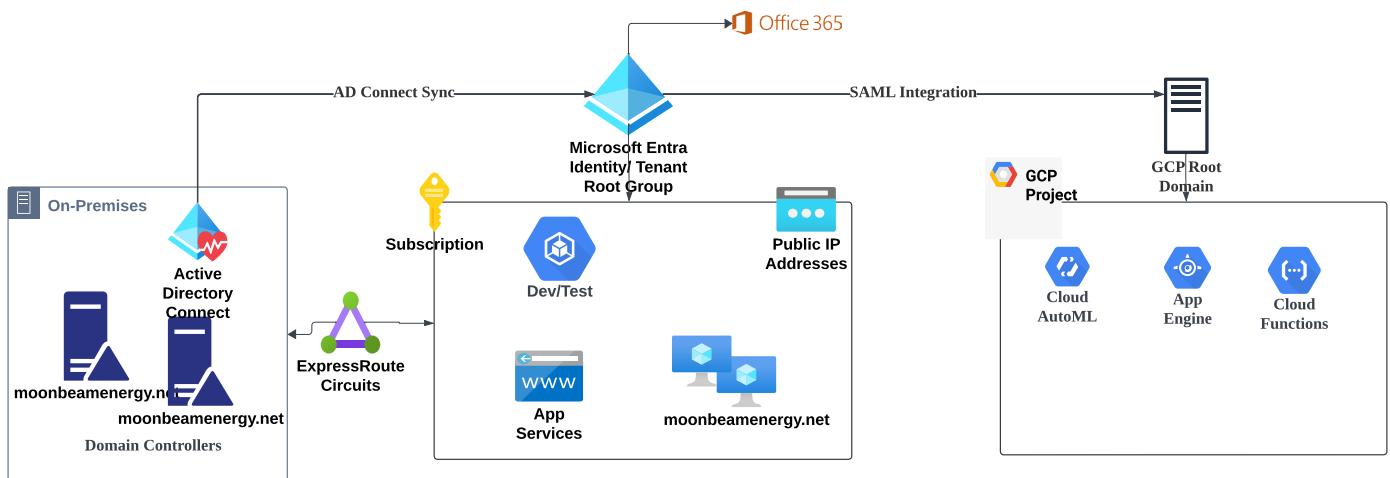


Figure 2: Moonbeam Energy environment with hybrid infrastructure deployed in Microsoft Azure, GCP, and on-premises

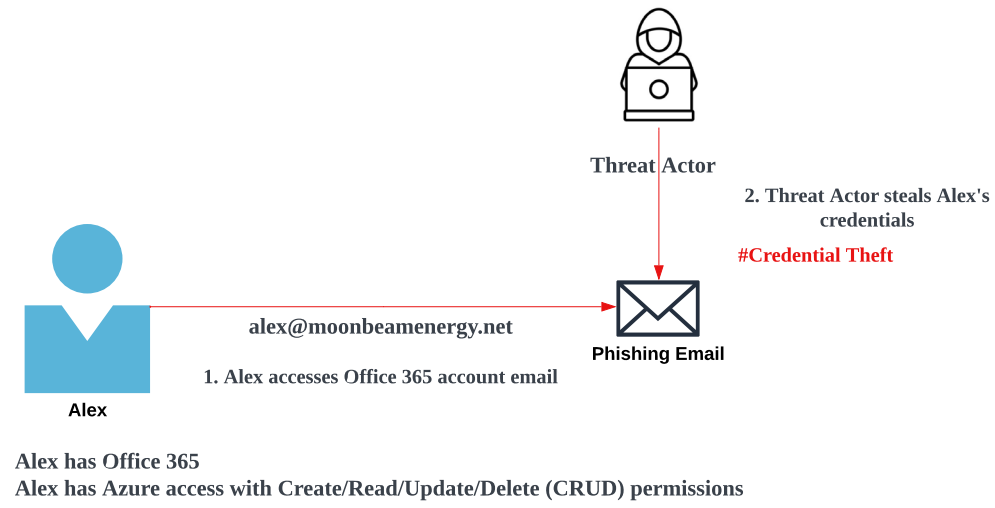
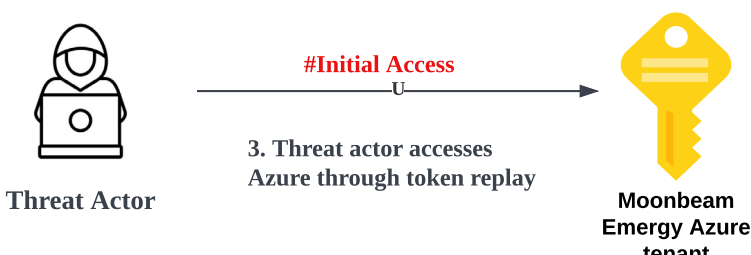
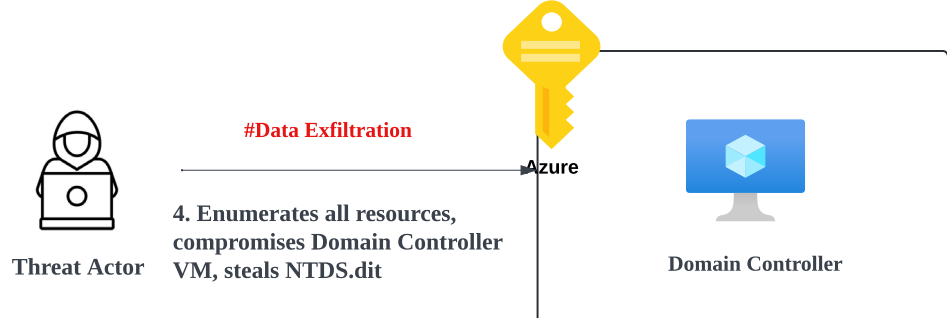
A user called Alex who uses their AD credentials to log in to their Hybrid AD joined workstation (i.e., the workstation is joined to the AD domain and registered with Microsoft Entra ID), uses Outlook to connect to Office 365 email service and check emails. Alex’s credential also holds Subscription Contributor privilege in Azure, which gives their account

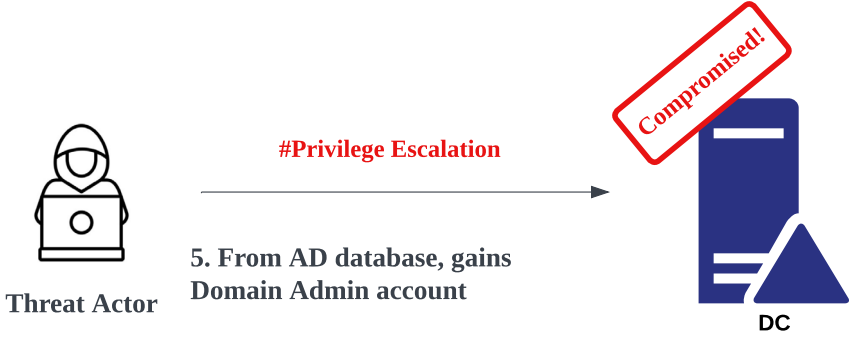
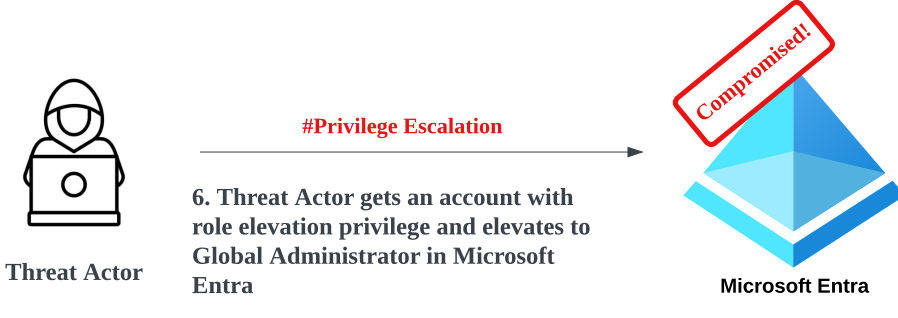
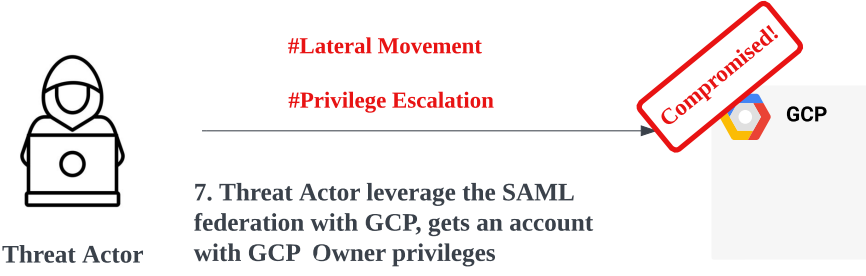
Attack Scenarios

create/read/update/delete permissions in the Azure Subscription. This gives Alex's identity read/write/delete privileges over Domain Controller Virtual Machines and other resources in Azure.

A threat actor performs a phishing campaign on Moonbeam Energy organization and sends emails with phishing links to all the employees.

Now let's explore the following attack path:

| Attack steps | Diagram |
|---|---|
| <p>Alex accesses email using an Office 365 account. Alex mistakenly clicks on a phishing email link, which leaks Alex's Microsoft Entra ID session token to a threat actor.</p> |  <p>1. Alex accesses Office 365 account email</p> <p>Alex has Office 365 Alex has Azure access with Create/Read/Update/Delete (CRUD) permissions</p> <p>2. Threat Actor steals Alex's credentials #Credential Theft</p> |
| <p>The threat actor now replays the token in the browser and accesses Microsoft Azure using Alex's privileges.</p> |  <p>3. Threat actor accesses Azure through token replay</p> <p>#Initial Access</p> |
| <p>The threat actor uses Alex's privileges and performs resource and configuration enumeration, accesses DC VM disks, copies AD database information from *NTDS.dit file.</p> |  <p>4. Enumerates all resources, compromises Domain Controller VM, steals NTDS.dit</p> <p>#Data Exfiltration</p> |

| Attack steps | Diagram |
|---|--|
| <p>The threat actor now gets password hashes of AD administrators such as Domain Administrator accounts, leading to AD domain takeover.</p> |  <p>Threat Actor → #Privilege Escalation → DC</p> <p>5. From AD database, gains Domain Admin account</p> |
| <p>From the AD database, the threat actor gets credentials of an Azure synchronized identity, holding role elevation privileges. Threat actor elevates to Global Administrator role and takes control of Azure environment.</p> |  <p>Threat Actor → #Privilege Escalation → Microsoft Entra</p> <p>6. Threat Actor gets an account with role elevation privilege and elevates to Global Administrator in Microsoft Entra</p> |
| <p>The Azure tenant is federated with GCP IAM. This means compromised accounts in Azure can be used to log in to the GCP environment. Threat actor uses one of the compromised accounts to gain GCP Project Owner permission. This leads to GCP environment compromise.</p> |  <p>Threat Actor → #Lateral Movement #Privilege Escalation → GCP</p> <p>7. Threat Actor leverage the SAML federation with GCP, gets an account with GCP Owner privileges</p> |

***Note:** The NTDS.Dit file is a database in AD that stores password hashes of all user accounts.

Figure 3 depicts the complete attack flow.

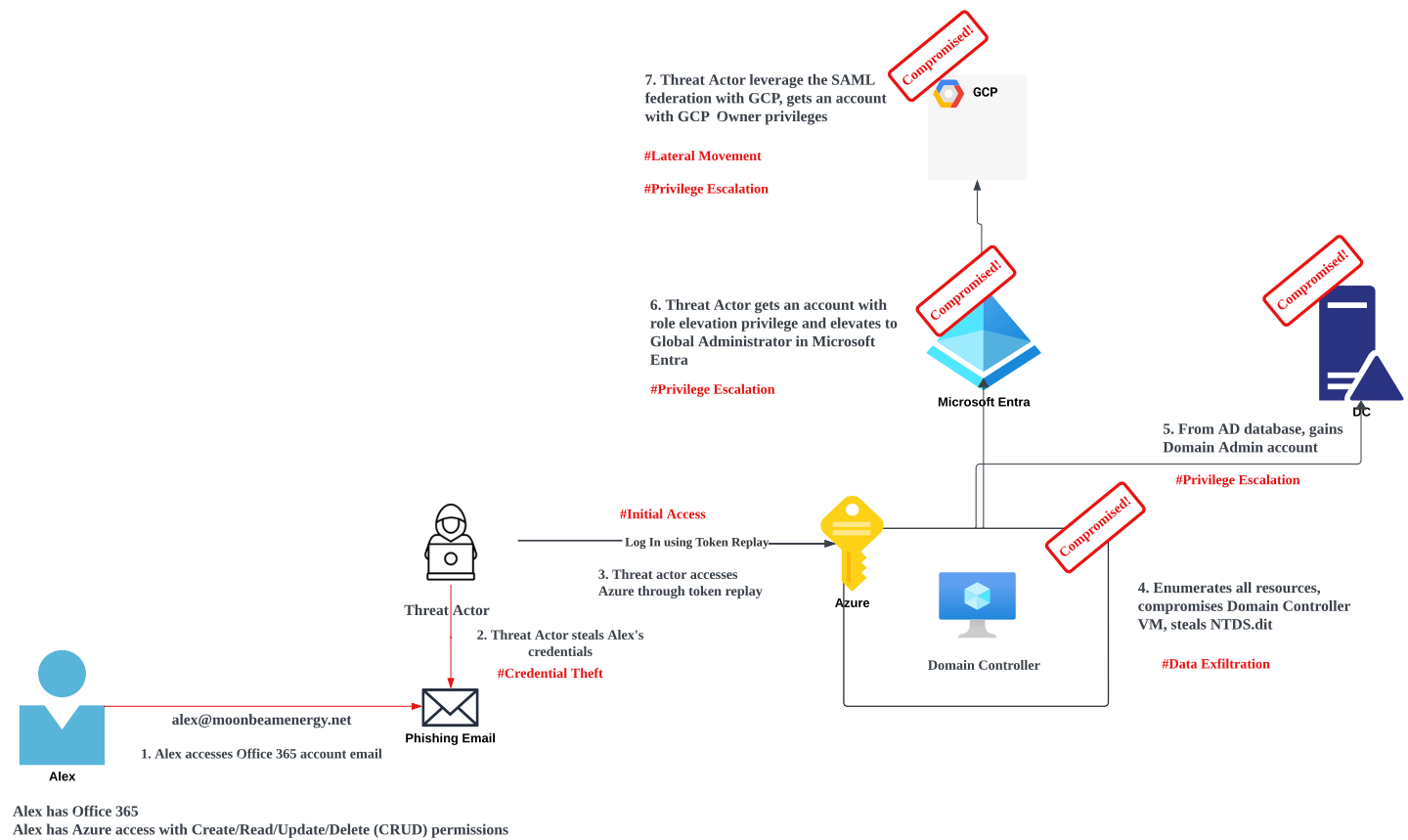


Figure 3: Total Domain Compromise through token theft in Microsoft Azure — an environment with no tiering

This paper illustrates in its last [section how Moon Beam Energy organization applies tiering](#) across on-premises servers and cloud, to protect against the previously detailed attack scenario, and other cloud based credential theft attacks.

Scenario #2: Compromise Multi-Cloud Workloads

In this scenario an organization named Cyber Coffee Co. Ltd. has deployed workloads across Azure, AWS and GCP clouds. The organization uses Okta as their identity provider and Okta is integrated with all the three cloud platforms as the identity and access solution.

Figure 4 depicts the high-level architecture for Cyber Coffee Co. Ltd.

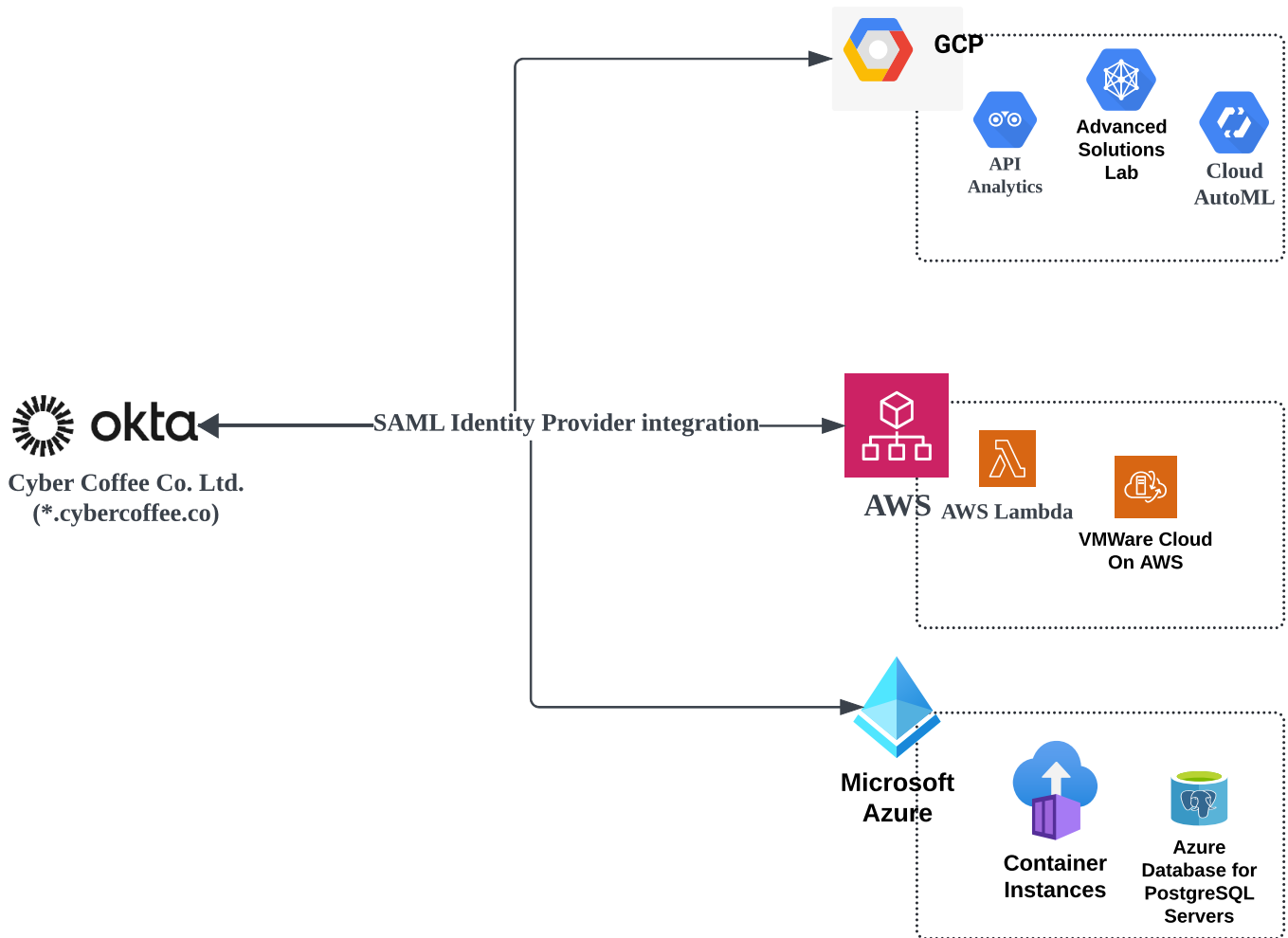
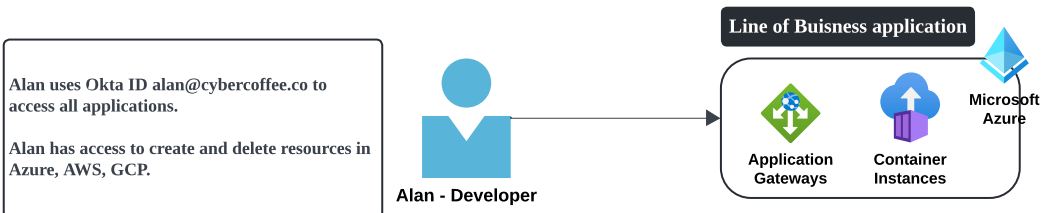
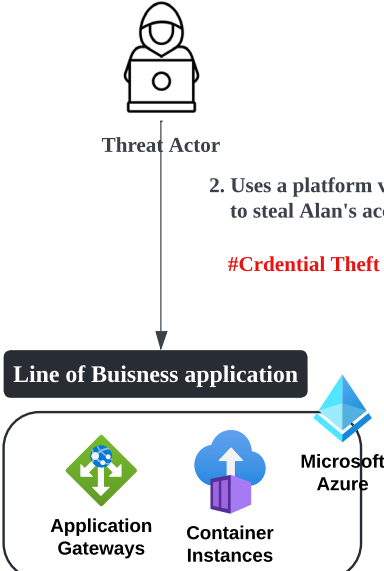
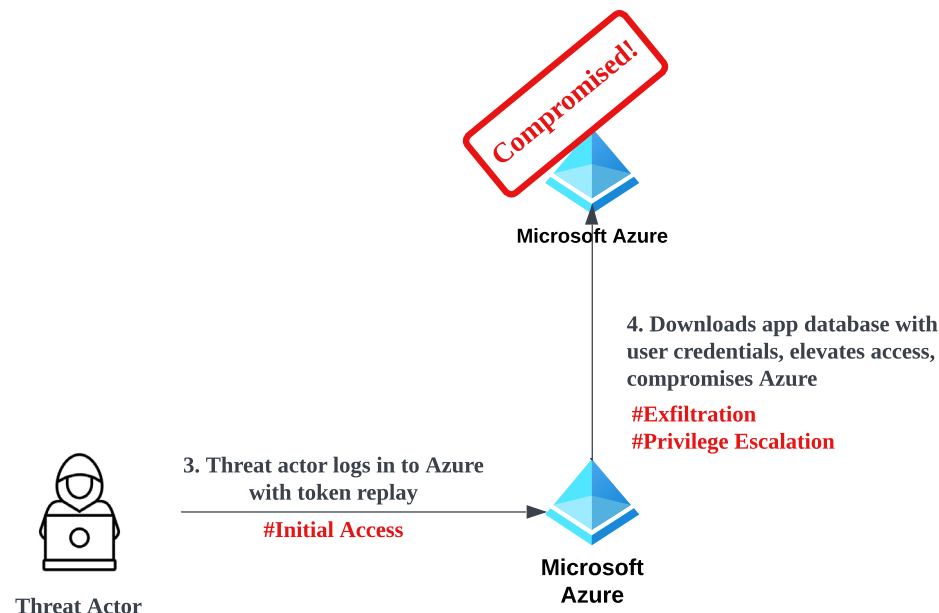




Figure 4: High-level architecture for Cyber Coffee Co. Ltd. with workloads deployed in AWS, GCP, and Azure

Attack Scenarios

This scenario explores the case of a developer named Alan, whose credential holds permissions over all the three cloud environments to create and manage IaaS and PaaS resources.

| Attack Steps | Diagram |
|--|---|
| <p>Alan uses one Okta credential to authenticate into an Internet facing business application, and perform developer responsibilities.</p> <p>Alan accesses the Internet facing business application using Okta credentials.</p> |  <p>Alan uses Okta ID alan@cybercoffee.co to access all applications.</p> <p>Alan has access to create and delete resources in Azure, AWS, GCP.</p> <p>Alan - Developer</p> <p>Line of Buisness application</p> <p>Application Gateways</p> <p>Container Instances</p> <p>Microsoft Azure</p> <p>1. Alan authenticates to LoB app</p> |
| <p>The business application is hosted in Azure container instances. The container instance has a known platform related vulnerability (common vulnerability enumeration or CVE).</p> <p>The threat actor is able to leverage the CVE to gain access to the application container, and steal Alan's session tokens.</p> |  <p>Threat Actor</p> <p>2. Uses a platform vulnerability to steal Alan's access token</p> <p>#Crdential Theft</p> <p>Line of Buisness application</p> <p>Application Gateways</p> <p>Container Instances</p> <p>Microsoft Azure</p> |
| <p>The threat actor accesses Microsoft Azure portal by replaying Alan's token.</p> <p>Then the threat actor proceeds to enumerate resources that Alan has access to within Azure.</p> <p>With Alan's developer access, the threat actor is able to copy data from the application database.</p> <p>The application database contains user credentials, profile data.</p> |  <p>Threat Actor</p> <p>3. Threat actor logs in to Azure with token replay</p> <p>#Initial Access</p> <p>Microsoft Azure</p> <p>4. Downloads app database with user credentials, elevates access, compromises Azure</p> <p>#Exfiltration</p> <p>#Privilege Escalation</p> <p>Compromised!</p> <p>Microsoft Azure</p> |

Attack Scenarios

| | |
|---|---|
| <p>With access to more user credentials, the threat actor cracks them offline, and gets multiple Okta credentials with access to other cloud platforms.</p> <p>With a compromised identity having federated access to AWS IAM platform, the threat actor accesses the AWS environment and carries out the compromise further.</p> |  <p>5. Use stolen credential to access and compromise</p> <p>#Lateral Movement #Privilege Escalation</p> <p>Threat Actor → AWS</p> |
| <p>The threat actor accesses a GCP cloud environment with compromised credentials and elevates access to the GCP Organization, accesses and compromises more resources.</p> |  <p>6. Take over GCP</p> <p>#Lateral Movement #Privilege Escalation</p> <p>Threat Actor → GCP</p> |
| <p>With the compromised identities, the threat actor accesses Okta identity platform with a privileged account.</p> <p>The series of steps results in total domain compromise for the organization.</p> |  <p>7. IdP Compromised</p> <p>#Total Compromise</p> <p>Threat Actor → okta</p> |

Attack Scenarios

This series of attack steps can take weeks, months and sometimes years for an adversary from the time they accessed an environment, performed reconnaissance, to the time they exfiltrate data, escalate privileges and carry out the compromise more aggressively. A lot of times the adversary can go undetected for years in the environment; this is more common with Advanced Persistent Threat (APT) actors. Threat actors will often install backdoors to maintain persistence, while being undetected by the organization's security tools.

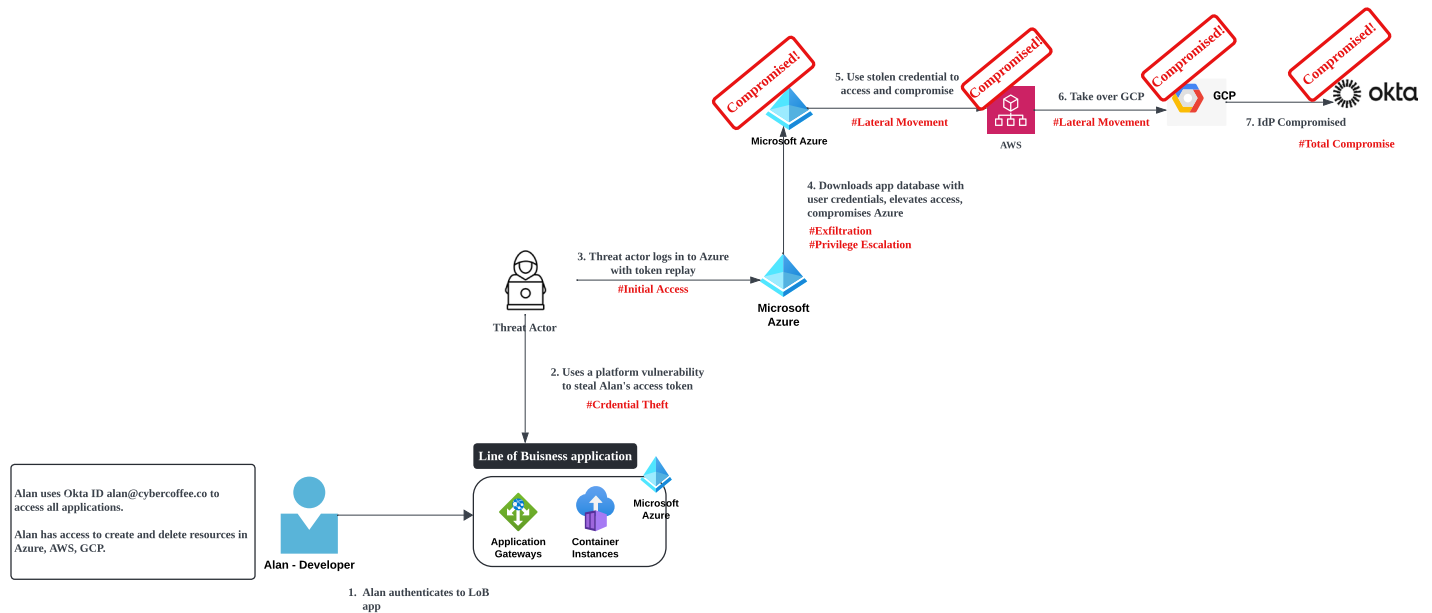


Figure 5: End-to-end attack path for the scenario

This paper illustrates in its last [section how Cyber Coffee Co. Ltd. applies tiering](#) across on-premises servers and cloud, to protect against the previously detailed attack scenario, and other cloud based credential theft attacks.

Analyzing the Cloud Compromise Scenarios

The scenarios focused on various attack paths where the threat actor was able to elevate and compromise the cloud environment. Most of the attacks occur due to the following underlying reasons:

| Why the compromises occurred | How to evade such attacks (by platform) |
|---|---|
| <p>Lack of credential and endpoint separation to conduct administrative tasks vs. productivity tasks: Productivity applications are most often the attacker’s target for phishing attacks, other social engineering attacks. Mandiant has observed attackers deploy malware on endpoints, steal login credentials and/or tokens of users using such techniques. The attackers can then use those credentials to access cloud resources and elevate access.</p> | <p>Consider credential separation or tiering control by cloud:</p> <ul style="list-style-type: none"> • AWS • Microsoft Azure • GCP <p>Endpoint Separation through Cloud Privileged Access Workstations control by cloud:</p> <ul style="list-style-type: none"> • AWS • Microsoft Azure • GCP |
| <p>Critical and non-critical resources deployed in the same cloud environment: Mandiant has observed many customers get compromised due to having lack of proper boundaries to host cloud infrastructure. Active Directory Domain Controllers (ADDC), PAM servers, SAML infrastructure and other identity related infrastructure are often present in the same cloud environment having non-production and often Internet facing applications. Such non-production resources are often present with lesser security controls and a broad set of users having access to those resources. This gives rise to a compromise in such a resource giving rise to paths for lateral movement for an attacker to compromise ADDC, PAM and other identity servers. This leads to stolen administrator and high-value credentials, and total domain takeover.</p> | <p>To mitigate this risk, explore the controls describing segregation of resources into various tiers for each cloud platform:</p> <ul style="list-style-type: none"> • AWS • Microsoft Azure • GCP <p>Define network segmentation strategy for each cloud platform:</p> <ul style="list-style-type: none"> • AWS • Microsoft Azure • GCP |
| <p>Lack of monitoring and detection mechanisms: Oftentimes environments lack proper monitoring and alerting mechanisms to detect potential threats and anomalous patterns. Real-time monitoring can identify suspicious activity, such as unauthorized access attempts, unusual data transfers, or misconfigurations, before they escalate into full-blown breaches. Incident response teams can respond faster when detections are enabled, minimizing the damage caused by attacks and preventing further data loss or system compromise.</p> <p>Many industry regulations and security standards mandate logging and monitoring of critical events to ensure accountability and traceability.</p> | <p>Setting up monitoring and detection use cases for each cloud platform:</p> <ul style="list-style-type: none"> • AWS • Microsoft Azure • GCP |

| | |
|---|--|
| <p>Security misconfigurations: Cloud compromises can also occur due to IAM system misconfigurations, which can give an attacker unauthorized access to perform reconnaissance, elevate access and compromise data. Such common misconfigurations can include overly permissive permissions assigned to identities, poorly managed secrets, API keys, credentials, unpatched systems, and misconfigured network security.</p> | <p>Apply security configurations at scale to all tiers and resources within each tier:</p> <ul style="list-style-type: none">• AWS• Microsoft Azure• GCP |
|---|--|

Common Cloud-Based Credential Theft Techniques

Most cloud environments use modern authentication protocols to enable user access. While the risks associated with credential theft involving legacy protocols such as Kerberos, NTLM, basic authentication is significantly reduced with modern authentication; yet attackers use other mechanisms such as phishing, token theft and replay, attacker in the middle (AiTM), golden SAML techniques and so on to gain access to cloud environments. Due to poor operational practices and misconfigurations, attackers can then elevate access further to steal data, take control of environments.

This section details common attack techniques for cloud workloads that Mandiant has observed.

Pass-the-Cookie Attack

Pass-the-Cookie (PtC) is a cyber-attack technique where an attacker steals a victim's web session cookie. This cookie, typically stored in the victim's browser, contains authentication information that allows the attacker to impersonate the victim and gain unauthorized access to websites or web applications without needing to know the victim's credentials.

This attack is often used to bypass multi-factor authentication (MFA), as the stolen cookie can provide access even after the victim has logged out. PtC attacks can be carried out through various methods, such as malware, phishing, or exploiting vulnerabilities in web applications.

The consequences of a PtC attack can be severe, as attackers can gain access to sensitive information, perform unauthorized actions, and even compromise entire systems or networks.

While secure browsers offer various security measures, they cannot completely prevent pass-the-cookie (PtC) attacks. Secure browsers can implement features like:

- **HttpOnly cookies:** These cookies are inaccessible to client-side scripts, making them harder to steal through cross-site scripting (XSS) attacks.
- **Secure cookies:** These cookies are transmitted only over HTTPS, preventing interception over unsecured networks.
- **SameSite cookies:** These cookies restrict how cookies are sent in cross-site requests, reducing the risk of certain PtC attacks.
- **Anti-malware and anti-phishing protection:** These features can help detect and block malicious software that attempts to steal cookies.

However, these features are not foolproof. Some advanced PtC attacks can bypass these protections, especially if the attacker has already gained access to the victim's device through malware or social engineering. Additionally, misconfigurations in web applications or browser extensions can also create vulnerabilities that attackers can exploit.

Pass-the-Token Attack

Pass-the-Token (PtT) is a cyberattack technique where an attacker steals a user's security token, which is a digital representation of their identity and privileges within a system. By reusing this stolen token, the attacker can impersonate the user and gain unauthorized access to resources or perform actions on their behalf.

PtT attacks are particularly relevant in cloud environments, where identity and access management rely heavily on tokens for authentication and authorization. The attack typically targets active sessions, exploiting vulnerabilities in token management or storage mechanisms.

Attackers can obtain tokens through various methods, including phishing scams, malware infections, or exploiting vulnerabilities in web applications. Once a token is compromised, the attacker can use it to move laterally across the network, escalate privileges, and potentially gain access to sensitive data or critical systems.

The impact of a PtT attack can be severe, leading to data breaches, service disruptions, and financial losses. Organizations can mitigate this threat by implementing robust security measures, such as strong authentication protocols, token expiration policies, and regular monitoring of network activity.

Golden SAML Attack

The Golden SAML attack is a sophisticated cyber threat targeting cloud-based identity systems that utilize the Security Assertion Markup Language (SAML) protocol for authentication. Attackers exploit vulnerabilities in SAML infrastructure or compromise identity providers to forge SAML assertions, granting them unauthorized access to cloud resources.

The Golden SAML attack is particularly dangerous due to its ability to bypass traditional security measures, such as multi-factor authentication (MFA), and its potential to provide persistent access to cloud environments. It can be used to impersonate legitimate users, escalate privileges, and exfiltrate sensitive data.

Organizations that rely on SAML-based identity systems are encouraged to review their security posture and implement appropriate measures to protect against Golden SAML attacks. In the subsequent sections, we will cover the security controls that will protect against Golden SAML attacks.

In the subsequent section we will go through the proposed architecture to secure a multi-cloud environment by protecting critical assets, users, and applying baseline configurations to protect against breaches.

Proposed Architecture Model to Secure Cloud Resources

From the various [previously detailed](#) attack techniques, it is evident that attackers use multiple vectors to compromise users and gain access to environments. This section defines a baseline architecture to harden and secure access to resources hosted across multiple cloud environments.

The controls mentioned in this section will be focusing on securing the following to establish a baseline/Tier model architecture for each cloud:

- Resource segregation through logical containment of privileged workloads
- Separation of credentials
- Restrict inheritance of privileges
- Use of managed administrative workstations with restrictive controls
- Apply network segmentation and network security controls
- Apply security configurations to resources
- Monitoring and Detection

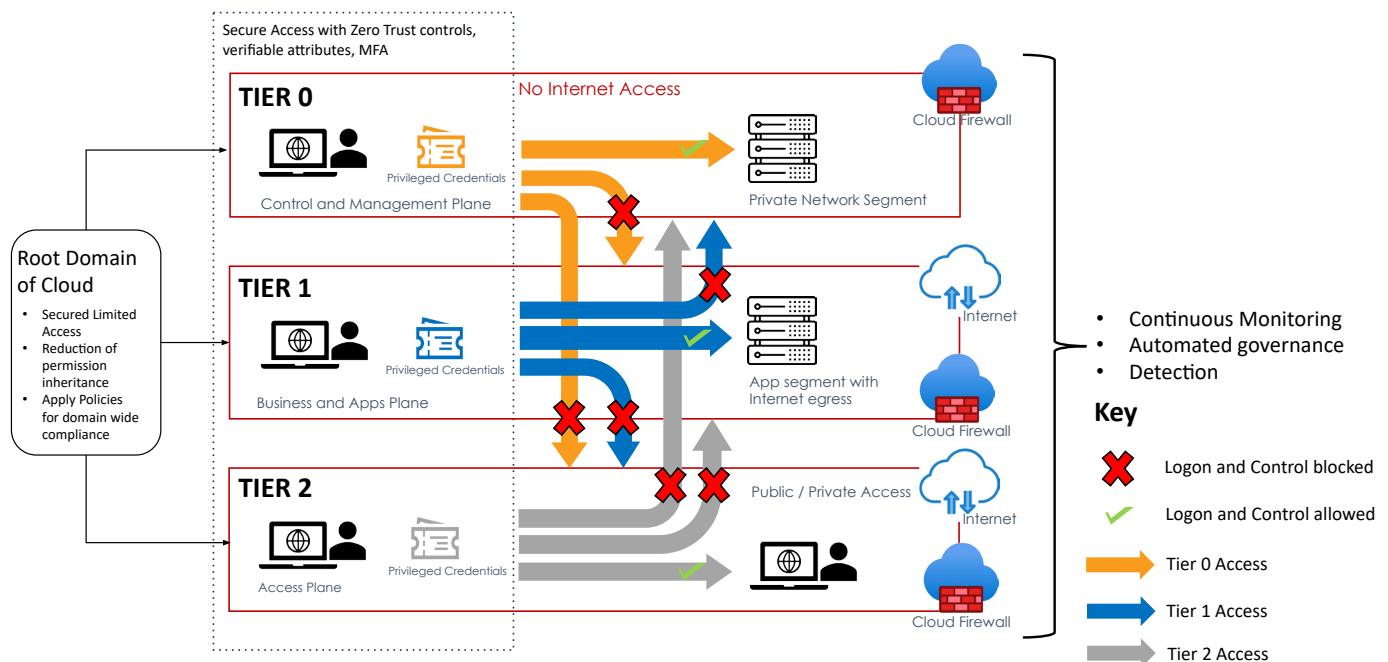


Figure 6: High Level Architecture of tiering in the cloud environment

Proposed Architecture Model to Secure Cloud Resources

Figure 6 shows tiering controls are applied to segregate resources deployed within various tier-driven boundaries, irrespective of the resource hosting across cloud and on-premises platforms. The tiering controls help in successfully achieving the following objectives to applying baseline security and compliance, and protecting against various cloud based attacks:

| Objective accomplished through tiering | Description of applied controls |
|---|---|
| Restricting Logins based on Tiers | Identities and endpoints are locked-down based on the tiers they are allowed to access, and any scope of privilege elevation to log in to higher tiers, or conversely higher tier credentials or endpoints accessing lower tier environments, are blocked using various policies and security controls. |
| Zero Trust Security approach for access | Identities and endpoint devices follow a zero-trust approach to access resources in each tier, with dynamic access control evaluation in place, checking for presence of factors such as multi-factor authentication, device compliance state, type of network segment to access, and device patch status before granting access to the necessary tier and its resources. |
| Resource Segregation | Resources are segregated and deployed based on their aligned tiers. Tier 0 resources are placed within the same resource container boundary, Tier 1 resources are placed within Tier 1 resource container and similarly for Tier 2. This is done to ensure all cross-tier access is blocked and any unintended access is eliminated. |
| Network Segmentation | Network address spaces hosting the tiered infrastructure is segmented to meet the tiering goals, and a zero-trust approach is adopted to ensure all network segments are protected with their respective firewall policies and rules, proxy solutions and additional network security solutions. |
| Consistent Security configuration enforcement | Cloud Security Posture Management tools and various other policy configuration tools are present to audit and enforce desired configuration to all cloud and on-premises hosted resources at scale. |
| Monitoring and Detections | Logging and monitoring is enabled throughout the environment consistently, and logs are forwarded to the central Security Information and Events Management (SIEM) solution. The logs are utilized to construct detection rules to alert security operations teams for any potential threats in the environment. |

The next sections focus on the three most commonly used public clouds: Amazon Web Services (AWS) cloud platform, Microsoft Azure, and Google Cloud Platform (GCP), and describe each security recommendation in greater detail for each cloud.

Tier Model in Amazon Web Services (AWS)

The controls detailed in this section establish a baseline/Tier model architecture for AWS:

- Resource segregation through logical containment of privileged workloads
- Separation of credentials
- Restrict inheritance of privileges
- Use of managed administrative workstations with restrictive controls
- Apply network segmentation and network security controls
- Apply security configurations to resources
- Monitoring and Detection

Resource Segregation in AWS

Overview

Tiering concepts focus on defining segregated zones to place resources based on their functionality. Traditionally the first tier (Tier 0) is reserved for critical resources responsible for privileged access, managing and storing credentials or performing authentication. This means Identity servers, PAM servers, Certification servers and any resources having a direct privilege to make changes to these services should fall under this category.

The second tier (Tier 1) is the business applications or data plane layer. Any resources in the cloud supporting business applications should be categorized under Tier 1.

Tier 2 is for hosting resources responsible for user access. This includes Virtual Desktop interfaces, productivity workstations, and customer and partner identity solutions.

Segregating cloud resources into their individual tiers helps to enhance security posture by enforcing the following:

- It helps limit role-based access control (RBAC) permissions assigned on those resources to specific identities. By doing this we are able to cut down access to critical resources and scope the access to only the identities that are responsible for managing them.
- Applying baseline security configurations in a scalable manner to resources that fall in the same tier.
- Limit paths of lateral movement and privilege escalation between resources aligned with different tiers.

AWS Implementation

The fundamental principle of segregation in AWS is that resources falling within a tier are completely isolated from resources in other tiers. This segregation can be enforced through respective Organizational Units for each tier, having AWS Accounts underneath its hierarchy to host resources within that tier. This minimizes risk of accidental cross-account access between higher and lower tiers, and helps contain the impact of any security breaches. AWS provides the capability to segregate resources starting with the root organization account as follows:

- **Root Organization:** This is the top-level container in AWS Organizations. It represents the entire AWS organization and is the starting point for the AWS environment. By default, the user identity that is used to create the AWS organization becomes Management Account or root Organization administrators for the entire AWS environment.
- **Organizational Units (OUs):** These are groups within your root organization that helps manage accounts and resources more efficiently. We can create multiple OUs to represent different tiers, environments (e.g., production, development, testing), or any other logical grouping that suits the business needs.
- **Accounts:** These are the basic building blocks of AWS. Each account is a separate entity with its own set of resources, billing, and permissions.

The recommended architecture to ensure tiering for AWS, would consider placing all the Tier 0 critical infrastructure within a separate OU, and further breakdown the OU structure based on the type of the Tier 0 workload. This mode of segregation through OU/AWS Accounts is to ensure least privilege controls, through granularity of assigned permissions for users at an assigned scope.

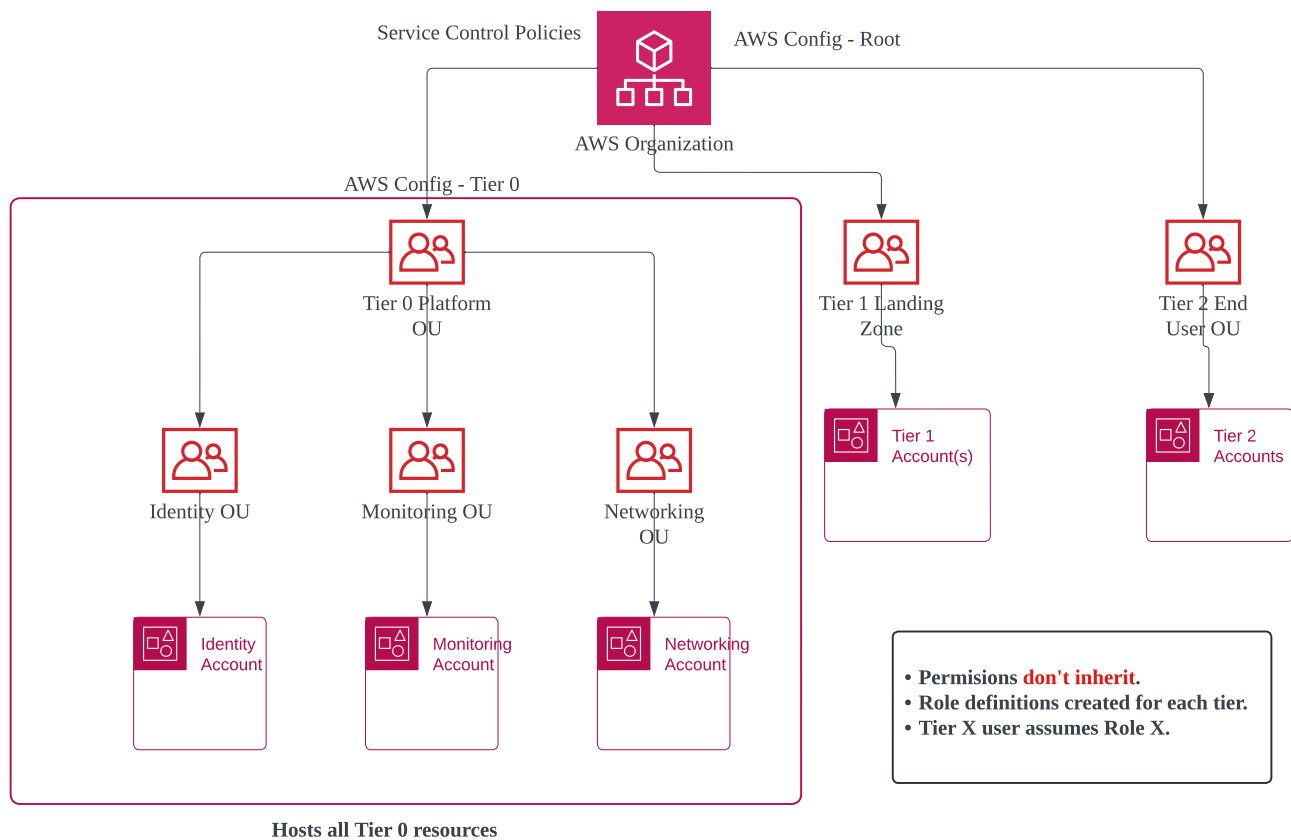


Figure 7: AWS resource segregation to host workloads in tiers

Figure 7 represents how resource segregation can be used to host critical resources within separate logical containers. Tier 0/1/2 resources are placed within separate AWS Accounts under the respective tier OUs. In this figure, we have segregated Tier 0 OU into Identity, Monitoring and Networking segments respectively, to house the respective resources within the individual Accounts. However, the sub-OUs for each tier can be created to host the required Accounts and resources, based on the individual organization's business needs.

Credential Tiering

Credential tiering is a fundamental security practice that significantly brings down the risk of various identity-based attacks. In the cloud, using unique sets of identities for different environments and levels of access, reduces the blast radius of an attacker in the event of a credential compromise, as the attacker will only be limited within that scope or environment. Following are the advantages of using separation of credentials as a mechanism to enforce tier model:

- **Minimize blast radius** in the event of a compromise, by limiting the attacker's access to specific scope of an identity and its permissions.
- **Limits lateral movement** paths between non-critical productivity environments to critical Tier 0 environments, by using separate sets of credentials for administrative and productivity tasks.
- **Enforce least privilege** by ensuring that accounts are provided permissions for only the operations they perform within each tier.
- **Enforce dynamic verifiable authentication controls** for the respective tier users, by segregating identities as Tier 0, 1 or 2, ensuring adequate multi-factor authentication and other verifiable controls (such as compliant device, and trusted location) are applied to every authentication by identities for each tier. The recommendation is to ensure all users are secured with multi-factor authentication, for accessing any application or workload. But having separate credentials for each tier will enable applying granular access policies relevant for each tier, such as adequate multi-factor authentication methods, and device compliance checks.
- **Facilitates Incident Response** in the event of a compromise, by identifying the credentials that would have certain permissions to perform an operation in each tier, and revoking the access without impacting other parts of the business.
- **Helps strengthen accountability** by ensuring that users are only provided credentials that aligns with their responsibilities and limits access that is no longer needed.

Separation of credentials for accessing and managing AWS accounts would involve creation of separate identities aligned with performing activities for each tier. Tier 0 identities will be aimed at managing AWS IAM Identity Center, and managing any Tier 0 servers hosted in AWS such as Active Directory, SAML federation servers, and so on. Tier 1 identities will be provisioned to manage any application workloads hosted in AWS or applications integrated with AWS platform.

The following controls should be considered to ensure credential tiering and protecting critical access to resources:

Limit exposure for Tier 0 administrator credentials

One of the fundamental guidelines of tiering, is using separate sets of credentials for performing administrative and operational tasks in each tier. To ensure total separation of access, ideally administrators in each cloud should use

Tier Model in Amazon Web Services (AWS)

separate cloud-only credentials that are not federated with or synchronized from any other environment. This is to ensure access paths between environments is reduced, and a compromise in one environment does not give an attacker administrator access in AWS.

AWS IAM Identity Center supports federation with Active Directory environment, and with External identity providers such as Okta, and Microsoft Entra ID. This allows organizations to set up the identity platform that AWS cloud can use for its users. It is recommended that any privileged credentials in the AWS cloud environment holding read/write/update/delete permissions over Tier 0 resources in AWS, should not have mailbox assigned or access to any productivity applications. This means if the identity is provisioned from AD or an external identity provider, the identity should not have any workload access in those environments and be restricted from holding privileged roles in the other environments. This will reduce surface area of exposure to attacks for administrator accounts.

Use Roles and Policies to enforce access based on tiers

Administrative operations across distinct cloud tiers (Tier 0, 1, and 2) necessitate strict access segregation. In AWS cloud, IAM user identities directly don't have permissions assigned to them; but instead, there are roles and policies which the users can assume for specific tasks. Role and policy definitions can be used as a mechanism to define specific Tier 0/1/2 based operations, and have specific users assume those roles to perform the operations.

To ensure there is separation of credentials in AWS, separate sets of AWS identities should be created to perform Tier 0, 1 and 2 activities. This will ensure no unintended users access critical resources hosted in AWS. Examples of Tier 0 AWS identities can include but not limited to the following:

- The **root organization user** identity that has privileges on all OUs and Accounts within an AWS Organization. This root user account should not be used for day-to-day operations, and be protected as a Tier 0 identity.
- Any identity holding privileges to make configuration changes in **AWS IAM**
- Identities having **read/write/update/delete** privileges over Active Directory servers in AWS, or **any Tier 0 resources hosted in AWS**.

For managing any Tier 0 resources hosted in AWS, Tier 0 roles and policies should be defined to specify Tier 0 privileged user identities allowed to assume the respective roles, and manage the resources. The policy definition will restrict only specific users from the respective tier to access the resources within AWS Accounts in that tier. This means a Tier 1 user cannot access a Tier 0 resource and vice-versa, as the specific roles assigned would prevent that access.

Similarly, the same approach can be taken to restrict specific Tier 1 users to access Tier 1 resources.

Restrict Inheritance of Privileges

Limiting inheritance of permissions across cloud resources is essential for maintaining a least privilege security model and preventing unauthorized access. Each cloud provider has its own inheritance model. Typically, permissions granted at a higher level (like an organization or project) are inherited by resources at lower levels (folders, services). Instead of granting overly broad permissions at higher levels we need to explicitly define permissions for each resource or group of resources at the appropriate level. Most cloud platforms allow creation of 'Deny' rules. Deny rules can override inherited permissions when applied at a certain resource hierarchy level. Strategically placing deny rules can block specific actions or permissions where inheritance is not desired.

Administrators also need to set up regular reviews to audit permission structures and ensure they align with the least privilege model for managing resources in the cloud and identify any unintended inheritance.

Restrict Inheritance of Privileges in AWS

In AWS cloud, permissions don't inherit in the same way as in other providers such as Microsoft Azure or GCP. Instead, AWS Organizations use Service Control Policies (SCPs) to manage permissions at the organization, organizational unit (OU), and account levels. SCPs define the maximum permissions for all IAM entities (users, groups, and roles) within the affected accounts or OUs. They act as guardrails, preventing actions that are not explicitly allowed by the SCP, even if those actions are allowed by IAM policies attached to the entity.

However, to ensure least privilege access is enforced in AWS, we can leverage the following controls:

- **Use of SCPs:** SCPs are inherited from organization root, to child OUs and accounts. However, SCPs at lower levels i.e., at account or OU levels can override SCPs inherited from the root organization. SCPs can be used to explicitly deny actions or services, restricting what can be done within the organization or OU.
- **Deny Statements in IAM Policies:** While IAM policies cannot override SCPs, we can use deny statements in IAM policies to further restrict permissions within an account or OU.
- **Permission boundaries:** Permission boundaries are an advanced feature within AWS that allows administrators to set the maximum permissions that an IAM entity can have, even if they are granted more permissive permissions through IAM policies or roles.

Use of Managed Administrative Workstations with Restrictive Controls

A Privileged Access Workstation (PAW) is defined as a dedicated, restrictive workstation that is used to perform sensitive tasks that require high privileges. It's designed to isolate privileged activities from regular user activities, reducing the risk of compromise from malware, phishing attacks, or other threats.

The use of PAWs to manage on-premises workloads aligned with tier model architecture, is a well-known mechanism to prevent attacks related to LDAP, Kerberos protocols such as Pass-the-hash, and Over-Pass-the-Hash. The concept of PAWs was introduced as organizations started adopting more structured approaches to privileged access management (PAM). This is when industry frameworks and guidelines, like Microsoft's Enhanced Security Administrative Environment (ESAE) and the National Institute of Standards and Technology (NIST) guidelines, began emphasizing the importance of using dedicated workstations for privileged tasks.

To manage cloud workloads, use of a dedicated administrative workstation such as a PAW, is still very much relevant since it reduces risk of credential theft for administrators, and provides restricted access to public Internet facing endpoints. Some of the controls that are implemented to build a Cloud PAW device for cloud workload management include the following:

- **Reduce risk of token-based attacks:** The Cloud PAW device is hardened through security controls such as device management, group policies, firewall policies, antivirus, and app locker. This ensures that the administrators can only connect to specific endpoints in the Internet (such as the cloud consoles, cloud shell) and run specific tools (command line tools) on the device to carry out administrative tasks. Most token-based attacks originate from productivity applications such as email, Drive and other applications. Such applications are blocked to be run within the Cloud PAW, thereby reducing the risk of token-based thefts.

- **Reduces privilege escalation paths:** A Cloud PAW is built to only allow an administrator to log in using specific credentials meant to perform operations aligned to the respective tiers. This means a Tier 0 Cloud administrator having roles assigned to manage multiple cloud IAM planes, gets a Tier 0 workstation and account, which is used to manage the IAM environment. This administrator credential issued to be used within the PAW device, will be separate from the credential used to perform day to day productivity operations such as access emails, Drive or similar applications. The PAW device enforces device segregation to perform administrator versus productivity tasks, thereby reducing surface area for attacks, and any risk of privilege escalation to compromise Tier 0 workloads.

The Cloud PAW or Cloud enrolled PAW device to manage multi-cloud environments, can be built using solutions such as a SaaS Device Management provider, SaaS Identity and Access Management platform. Depending on the solutions an organization has in place, the following **configurations** should be in place to build a Cloud PAW device:

- **Device Management solution:** Administrative workstations should be enrolled to a device management solution that enforces organization compliance policies, pushes enrollment certificates for encryption and applies necessary security controls. To enroll the Cloud PAW device for critical resource management, specific device management profiles for Tier 0,1 and 2 need to be created within the device management solution. The profile is applied to the users and groups of the respective tiers. The physical device to be enrolled as the Cloud PAW, will install the necessary security configurations based on the tier it is meant to manage. For example, a PAW device enrolled to the device management solution, to manage Tier 0 Cloud workloads will have the required certificates and password policies pushed, endpoint firewall and proxy settings installed, local administrator access disabled, any antivirus and endpoint detection and response solution installed, and device hardened.
- **Dynamic Access control policies:** Access to critical Tier 0 services in the cloud should only be provided under specific verifiable conditions. Such conditions can include additional checks on user access, such as the authentication request is coming from a compliant device, known IP addresses, multi-factor authentication, and so on. Access to privileged resources should be blocked if an authentication attempt does not meet such necessary criteria.
- **Security Configurations:** A managed administrator workstation should have baseline security controls applied. Such security controls can be applied through solutions such as Microsoft Intune configuration policies, and ADMX settings. Such settings are meant to apply to the Tier 0 users and groups to enforce necessary security controls such as Bit locker drive encryption, password policies, Firewall and proxy settings, Antivirus settings and so on. It is also important to block unnecessary capabilities within the OS such as telemetry, location sense services, voice assistant, any data analytics services such as the 'recall' feature in Windows 11, which can increase attack surface area within a highly restricted administrator workstation.
- **Firewall and Proxy solutions:** It is crucial to enforce Firewall at the endpoint device level for privileged access workstations, in addition to having traffic routed via an Enterprise firewall and perimeter network services. The firewall and proxy settings on the PAW device should be enforced to ensure any inbound traffic to the device is blocked, and outbound traffic is only allowed for specific allow-listed endpoint URLs/Ip addresses. The allow-listing of Internet facing addresses is done based on which environments the PAW is aimed to administer. For example, if an organization has Tier 0 resources in Azure and GCP, the specific administrator relevant URLs for both the Cloud platforms can be allow-listed at the device firewall.

- **Strong phishing-resistant multi-factor authentication for critical tier access:** Phishing-resistant MFA should be enforced on every authentication attempt for privileged users managing critical resources in the cloud through a PAW.

Phishing-resistant MFA is a type of multi-factor authentication designed to be immune to phishing attacks. It achieves this by eliminating the use of shared secrets or codes that can be intercepted and replayed by attackers. Most IAM solutions provide multiple MFA options. A comparison of various MFA methods reveals that certain methods are more prone to compromise than others. Examples of weaker methods include SMS One-Time Passcode, Phone authentication, and Push notification on an authenticator app, which have been known to be compromised by threat actors such as UNC3944. The most resilient MFA methods are the ones that satisfy the NIST Authenticator Assurance Level 3 requirements:

1. verifier impersonation resistant
2. verifier compromise resistant
3. verifies authentication intent

These requirements are the foundation for phishing-resistant methods that use a hardware or verification factor that fulfills the following criteria:

1. **No Shared Secrets:** Relies on public-key cryptography and cryptographic challenges, eliminating the vulnerability of shared codes or one-time passwords (OTPs) that can be phished.
2. **Verification of Origin:** Ensures the authentication request comes from the legitimate website or service and not a malicious imposter.
3. **User Presence and Intent:** Often requires a physical action or biometric verification from the user, adding another layer of protection against automated attacks.

Examples of phishing resistant methods include FIDO 2.0 security keys, certificate-based authentication, biometric based authentication.

- **Continuous monitoring and detection:** The Cloud PAW device should have its device logs forwarded to the SIEM solution. This is to ensure the organization can create necessary alerts to detect any anomalous activities from the logins within the device to flag any potentially malicious activities. This would help organizations quickly identify and remediate potential threat actor activity and protect privileged accounts more effectively.

Apply Network Segmentation and Network Security Controls

Organizations with more than one cloud platform have workloads deployed in multiple cloud resources. Cloud platforms will use virtual network components and cloud native components for its connectivity. There is often interconnectivity set up between different clouds and on-premises to ensure communication over the network. To protect the critical resources from network-based attacks, there needs to be network boundaries defined where we place critical services, applications and enable users to securely connect to them.

In the following sections we will go into the best practices to implement a secure distributed network that will support hosting critical and business resources in a multi-cloud environment.

Identify and Classify Assets

The first step to ensuring resources within various tiers are placed within the right network boundaries, we will first need to define the network segments. The Tier 0 or critical services when hosted on IaaS compute engines, should not have direct communication to the Internet.

Applications on the other hand are Tier 1 resources, and will have separate networking configurations. Tier 1 resources should be placed within a separate network. The exercise to define network boundaries begins with building an inventory of the networking resources and putting classifications for them to define severity of the asset Tier they should be placed in, communication needed to the Internet, and so on.

- **Inventory:** Organizations need to create a comprehensive inventory of all resources across their multi-cloud environment, including virtual machines, databases, storage buckets, and applications.
- **Classification:** The next step will be to classify assets based on their criticality and sensitivity. Consider factors like data classification, regulatory requirements, business impact, and potential attack surface.

Design the Segmentation Strategy

Once the organization has a classification of networking components for the various cloud platforms, the next step is to define the network segments. An example would be:

An organization decides to create two possible segments for the services deployed. The first segment is for Tier 0 trusted services such as Domain Controller resources, or any OAUTH identity services. The second segment is meant for business applications which are used by the organization's internal users.

Network security rules can be built and firewall rules for the first segment to block any traffic inbound from the Internet.

For segment two resources, specific inbound traffic from the Internet may be allowed via the enterprise firewall such as HTTPS.

The following points detail a segmentation strategy:

- **Micro-segmentation:** Divide the multi-cloud network into smaller, isolated segments based on asset classification, function, or business unit. This helps define granular network rules to allow and block specific kinds of traffic and thereby limits the lateral movement of attackers in case of a breach.
- **Zero Trust:** Organizations should adopt a Zero Trust approach, where every access request is verified regardless of its origin. This includes strong user authentication and authorization mechanisms for users based on dynamic access conditions such as device compliance, IP address of origin, and device platform.
- **Network Virtualization:** Leverage network virtualization technologies like virtual private clouds (VPCs), virtual networks (VNETs), and software-defined networking (SDN) to create and manage isolated segments across different cloud providers.

Implement Security Controls

The network communication should be set up to ensure any traffic to the Internet goes through the firewall. Firewall should enable domain name service resolution for any Internet traffic. All data flow within the network should use end-to-end encryption using TLS 1.2 and above. Insecure protocol usage should not be minimized as much as possible. The following sections list the capabilities that will secure all network traffic between components, and users accessing them.

Tier Model in Amazon Web Services (AWS)

- **Firewalls:** Deploy firewalls at various levels, including perimeter, between the network segments, to control traffic flow and enforce access policies.
- **Intrusion Detection and Prevention Systems (IDPS):** Implement IDPS to monitor network traffic for suspicious activity and block potential attacks.
- **Security Groups and Network Access Control Lists (ACLs):** Use security groups and ACLs to filter traffic based on source/destination IP addresses, ports, and protocols.
- **Encryption:** Encrypt data at rest and in transit to protect sensitive information from unauthorized access.
- **Vulnerability Management:** Regularly scan for vulnerabilities and apply patches to keep the environment secure.
- **Zero Trust Network Access (ZTNA):** Implement ZTNA solutions to provide fine-grained access control based on user identity, device posture, and context. ZTNA solutions can replace VPN solutions, which often lack the capability to provide granular access to specific resources. VPN solutions commonly provide access to a wider network range, which if compromised can allow a threat actor to move laterally in the network. ZTNA solutions (also referred as Security Service Edge) acquire traffic from endpoint clients and provide access to specific applications and services on an internal network or the internet. This helps reduce the attack surface and prevent lateral movement. Examples of such solutions include Microsoft Entra Global Secure Access, Zscaler Private and Internet Access, and Palo Alto solutions.
- **Cloud Access Security Broker (CASB):** CASB solutions can help gain visibility into and control over cloud application usage, ensuring compliance and preventing data leakage.
- **Secure Web Gateway (SWG):** SWG solutions are meant to filter web traffic, block malicious content, and protect against web-based threats. Organizations can also use such solutions to enforce blocking usage of sanctioned applications.

Apply Scalable Security Configurations and Governance

Every public cloud platform has their unique sets of capabilities to enforce security configurations and compliance of resources at scale. In this section let's dive into the baseline configurations that need to be enabled for resources classified within each tier.

- **Restrict administrator console/shell access:** Access to the administrator console for AWS and any AWS Cloud Shell access should be restricted to specific Tier 0 identities only. If the organization uses CI/CD deployment approach with Infrastructure-as-Code, the access to the console should be highly restricted only for emergency scenarios.
- **Plan for an emergency access scenario:** Organizations should plan to create identities to be used in the event of an emergency such as cyberattacks leading to a total environment compromise, system failures, and so on. AWS root user account can be treated as an emergency access account, have multi-factor authentication enabled, and password securely stored and governed. In addition to root user identity, additional AWS IAM local identities can be created to have administrator privileges across the AWS Organization, and to be used in the event of an emergency access scenario.

Tier Model in Amazon Web Services (AWS)

- **Encryption at rest:** The resources within the cloud need to have encryption at rest enabled to secure the data hosted within the resources. The encryption can use either platform-managed key or use bring-your-own-key (BYOK) scenarios. The keys should have proper governance to ensure periodic rotation and be stored securely. The storage of such encryption keys should be outside of the cloud platform to ensure they can be recovered in the event of a cloud platform compromise.
- **End-to-End Encryption:** All resources communicating between each other in the cloud networks and resources hosted to communicate through the Internet should be using encryption with the latest transport layer security (TLS) protocols.
- **Protect secrets and certificates:** All secrets, certificates, encryption keys, API keys, and more should be protected and stored within credential storage vaults. The credential vault should use secure storage of the secrets and be recoverable in the event of a compromise or disaster affecting the entire cloud platform. Only authorized users should have access to the vaults.
 - Any code or scripts within the environment should not use secrets in plaintext. Secrets should be referenced within the code to the vaults and handled in a secure manner.
- **Regular patching and vulnerability scanning services:** All servers and endpoint client devices within the environment should receive regular security patches and updates. Solutions should be in place to flag any common vulnerabilities detected in the environment.
- **Restrict Direct Public Access:** In critical environments, resources such as S3 buckets, GCP Cloud Storage, Azure Blob Storage, Key Vaults should not be hosted with public access enabled. This makes the resources discoverable on the Internet. Such resources should have restricted access on the network as much as possible from a range of IP addresses.
- **Restrict resource creation with proper controls:** In critical cloud environments such as Tier 0, and production environments, resource creation within the environment should be restricted by conditions. Such conditions could include regions where creation of resources is allowed, size of Compute Engines or Virtual Machines that are allowed, naming standards to comply, and tags present in the deployment template. This is necessary because Mandiant has observed multiple attackers create attacker controlled VMs in the compromised cloud environment, to perform further malicious activities. This is why such conditions are necessary, in addition to other restricting controls such as RBAC permissions, and network segmentation, as shared in earlier sections.

Scalable Security Configurations in AWS Cloud

AWS Config Rules and Service Control Policies (SCPs) can be used together as effective tools to enforce most of the security configurations described in the previous section, and implement a tiering model across AWS resources.

To apply the security configurations listed in the earlier section to AWS accounts within an organization, we can use AWS Config rules at the root organization level and child Tier 0/1/2 Organization Units (OU), and the inheritance of the rules will apply the configurations desired at the child OU levels. Additionally, Service Control Policies can be used to deny and block specific actions on critical resources such as resource deletion.

The steps for implementation are listed as follows:

- 1. Define Tiered Security Configurations:** For each tier, we will create a list of specific security policies that detail the required configurations for resources in that tier. These policies should align with the organization's security and compliance requirements.
- 2. Implement AWS Config Rules:** Create AWS config rules:
 - **Create Config Rules:** Post defining the security configurations for each tier, we will create AWS Config Rules that correspond to the security policies defined for each tier.
 - **Use of Managed and Custom Rules:** We will leverage AWS Managed Rules for common compliance and security use cases, and create custom rules for specific configurations relevant to the tiering model.
 - **Set Up Notifications:** Configure notifications (e.g., Amazon SNS) to alert when resources become non-compliant.
 - **Consider Automated Remediation:** For certain rules, automated remediation actions are to be implemented using AWS Lambda functions to automatically fix non-compliant configurations.
 - **Apply Config Rules at relevant Organization level:** The Config rules relevant for each tier are to be applied at the root Organization or specific Tier OUs, which will inherit to all resources underneath the OU structure. This is how we will enforce baseline security compliance throughout the AWS environment.
- 3. Implement Service Control Policies (SCPs):** SCPs are to be created at the specific OUs levels to restrict actions that users and roles can perform on AWS resources. SCPs are to be tailored to enforce the security configurations defined for each tier, and prevent actions that would lead to non-compliance. For example, to restrict access to specific services, actions, or resources based on their tier we will need to apply granular SCPs at the desired tier OU.
- 4. Combine AWS Config Rules and SCPs:**
 - **Detect and Prevent:** AWS Config rules are to be utilized to detect non-compliant configurations, and SCPs are to be used to prevent actions that would cause any non-compliance.
 - **Enforce Tiered Model:** SCPs act as a guardrail, ensuring that even privileged users cannot perform actions that violate the security policies for a specific tier.
 - **Layered Security:** Config Rules and SCPs provide a layered approach to security, enhancing the overall protection against misconfigurations and unauthorized access.

Consider the following example:

Tier 0 (Mission-Critical) rules and policies:

- **Policy:** Config Rule will ensure S3 buckets and EC2 OS and data disks containing sensitive data are encrypted with a specific Key Management System (KMS) key.
- **Enforcement:** SCP will prevent users from creating S3 buckets or EC2 instances without encryption or using unapproved KMS keys.

Tier 1 (Sensitive) rules and policies:

- **Policy:** Config Rule will ensure EC2 instances have specific security group configurations.
- **Enforcement:** SCP prevents users from modifying security groups in a way that violates the defined configurations.

Important Considerations to maintain compliance state:

- **Continuous Monitoring:** Organizations are encouraged to regularly review and update Config Rules and SCPs to adapt to changing security requirements and threats.
- **Testing:** Config Rules and SCPs are to be tested in a non-production environment before applying them to production.
- **User Awareness:** Users are to be trained about the tiered security mode of operations and the importance of adhering to the defined policies to create resources in the cloud.

Some of the key advantages to using AWS Config include:

- **Centralized Management:** Security baselines across the entire organization can be managed from a single control plane.
- **Proactive Enforcement:** Config Rules will help prevent misconfigurations and non-compliant deployments.
- **Scalability:** As the organization grows, the rules to ensure compliance are applied by default through the inheritance hierarchy to new accounts and resources. This ensures scalability of the desired security model.
- **Flexibility:** Rules can be customized to meet the specific industry and regional security and compliance requirements.

Organizations are encouraged to regularly review and update the AWS Config Rules to adapt to evolving security threats and compliance needs.

Monitoring and Detection

While tiering limits lateral movement within the environment, determined attackers may still attempt to escalate privileges or move between tiers. Continuous monitoring and detection can help identify such attempts and prevent them from succeeding. When implementing tiering of resources within a cloud environment, we need to ensure proper levels of logging is enabled on all resources. The logs should be collected from the cloud environments and forwarded to a central Security Information and Events Management (SIEM) solution. In the event of a compromise, logs are the primary source of information for security operations teams to investigate the attacker path, and properly remediate and recover the environment. It is recommended that organizations create detection rules within their SIEM solution to proactively detect any anomalies and trigger an incident if needed to ensure expedited recovery from an attack.

The following types of logs should be enabled at minimum in the various tiers:

- Identity logs that include any sign-in and audit events for all types of accounts (users, service accounts, machine identities)
- Security logs

Tier Model in Amazon Web Services (AWS)

- System logs
- Any productivity application logs
- Firewall logs
- Proxy, VPN and any perimeter solution logs
- Endpoint device logs for users of all tiers

The following briefly captures the typical anomalies to alert for while building monitoring and detection scenarios as part of the tiering exercise:

- **Anomalous patterns:** SIEM systems can be tuned to generate alerts on detecting any anomalies from identified patterns. Certain indicators can alert a potential security breach to the Security Operations Center (SOC) team. Anomalies can include consecutive failed login attempts, consecutive logins from different time-zones within a short time frame, disabled accounts getting re-enabled, login events at unusual times, running of malicious binaries, scripts and so on.
- **Credential reset events:** High value user accounts and administrator accounts should be enabled with alerting to trigger notification in the event of password resets, multi-factor authentication method updates and so on.
- **Detection of potential data exfiltration:** Alerting should exist for critical resources to flag any events where large amounts of data is copied to unexpected drive locations or external IP addresses, bulk permission updates on sensitive files, and bulk deletion events.
- **Configuration Drift Detection:** Configurations set in the environments to ensure baseline security and compliance will modify over time depending on how the organization scales, to keep up to date with updated guidelines. This can potentially introduce security vulnerabilities or non-compliance. Monitoring will help identify configuration drift and ensure resources stay aligned with the security policies defined for their tier. Specific unexpected drifts can be alerted to the SOC teams such as updating of federated domains on an Identity provider to add new unknown domains, deletion of existing domains, and disabling encryption and reducing encryption levels in resources. These are common behaviors Mandiant sees attackers perform in compromised environments.

In an environment with multiple cloud platforms in use, along with on-premises infrastructure, organizations are encouraged to explore solutions that support Multi-Cloud Security and Posture Management (CSPM). CSPM solutions can help provide visibility, continuous monitoring, automated remediation to address any misconfigurations, vulnerabilities and compliance issues across different cloud platform resources.

Key features and functionalities of a CSPM solution that can help achieve an elevated level of security for multi-cloud resources include:

- **Visibility and Asset Inventory:** CSPM tools provide a comprehensive view of the cloud assets, including compute instances, storage services, databases, and network configurations, across multiple cloud providers.
- **Configuration Assessment:** It helps to continuously assess the configurations of the cloud resources against security best practices, compliance standards (e.g., PCI DSS, HIPAA, GDPR), and internal security policies.
- **Misconfiguration Detection:** SOC teams can set up alerting for misconfigurations within critical Tier 0 or Tier 1 resources, that could lead to a bigger incident and quickly apply mitigations as needed.

Tier Model in Amazon Web Services (AWS)

- **Vulnerability Management:** CSPM solutions can scan for vulnerabilities and prioritize them based on severity and risk.
- **Remediation and Automation:** Organizations can set up actionable recommendations for the relevant teams and automated workflows as applicable, to fix misconfigurations and mitigate any potential vulnerabilities.
- **Incident Response:** CSPM solutions can support incident response efforts by providing visibility into cloud security events and facilitating forensic investigations.
- **DevSecOps Integration:** Most CSPM solutions integrate with DevOps processes to ensure security is built into the development and deployment lifecycle.

Tier Model in Microsoft Azure

The controls detailed in this section establish a baseline/Tier model architecture for Microsoft Azure:

- Resource segregation through logical containment of privileged workloads
- Separation of credentials
- Restrict inheritance of privileges
- Use of managed administrative workstations with restrictive controls
- Apply network segmentation and network security controls
- Apply security configurations to resources
- Monitoring and Detection

Resource Segregation in Microsoft Azure

Overview

Tiering concepts focus on defining segregated zones to place resources based on their functionality. Traditionally the first tier (Tier 0) is reserved for critical resources responsible for privileged access, managing and storing credentials or performing authentication. This means Identity servers, PAM servers, Certification servers and any resources having a direct privilege to make changes to these services should fall under this category.

The second tier (Tier 1) is the business applications or data plane layer. Any resources in the cloud supporting business applications should be categorized under Tier 1.

Tier 2 is for hosting resources responsible for user access. This includes Virtual Desktop interfaces, productivity workstations, and customer and partner identity solutions.

Segregating cloud resources into their individual tiers helps to enhance security posture by enforcing the following:

- It helps limit role-based access control (RBAC) permissions assigned on those resources to specific identities. By doing this we are able to cut down access to critical resources and scope the access to only the identities that are responsible for managing them.
- Applying baseline security configurations in a scalable manner to resources that fall in the same tier.
- Limit paths of lateral movement and privilege escalation between resources aligned with different tiers.

Microsoft Azure Implementation

Segregation of resources in Azure is performed with the intent of placing critical resources within separate tiers based on their usability and business needs. This helps achieve separation of duties, and apply consistent security configurations to the resources within each tier container, in a scalable manner.

Tier Model in Microsoft Azure

The resource containerization within Azure cloud is achieved through the following:

- **Management Groups:** These are containers that help us manage access, policy, and compliance for multiple subscriptions. They are a key component of large-scale Azure deployments, allowing grouping of subscriptions into a hierarchy that aligns with the organizational structure. The Microsoft Entra tenant identifier is the Root of the Management Group hierarchy, and multiple Management Groups (MG) can be created underneath the tenant root.
- **Subscriptions:** These are the fundamental building blocks of Azure. Each subscription represents a separate billing entity with its own set of resources, permissions, and policies.

The tiered architecture for Azure workloads can be broken down into Tier 0 Platform MG to host IAM resources, networking and monitoring resources, Tier 1 Landing Zone to host application workloads and Tier 2 to host any end-user workloads such as Azure Virtual Desktop environments, B2C or Customer Identity and Access Management (CIAM) tenants. Figure 8 is a simplified example of the MG structure to ensure tiering segregation:

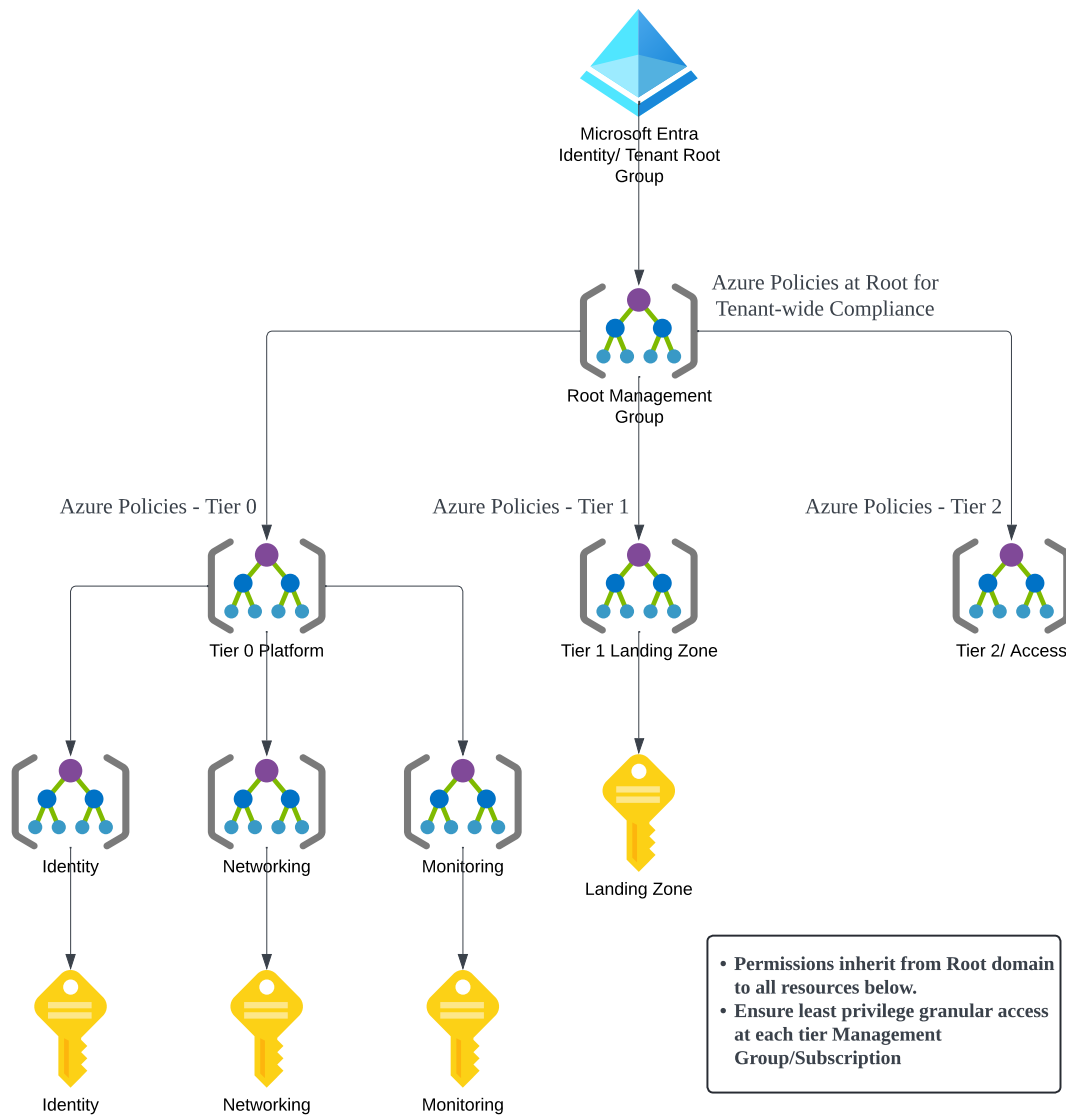


Figure 8: Azure resource segregation to host workloads in tiers

Credential Tiering

Credential tiering is a fundamental security practice that significantly brings down the risk of various identity-based attacks. In the cloud, using unique sets of identities for different environments and levels of access, reduces the blast radius of an attacker in the event of a credential compromise, as the attacker will only be limited within that scope or environment. Following are the advantages of using separation of credentials as a mechanism to enforce tier model:

- **Minimize blast radius** in the event of a compromise, by limiting the attacker's access to specific scope of an identity and its permissions.
- **Limits lateral movement** paths between non-critical productivity environments to critical Tier 0 environments, by using separate sets of credentials for administrative and productivity tasks.
- **Enforce least privilege** by ensuring that accounts are provided permissions for only the operations they perform within each tier.
- **Enforce dynamic verifiable authentication controls** for the respective tier users, by segregating identities as Tier 0, 1 or 2, ensuring adequate multi-factor authentication and other verifiable controls (such as compliant device, and trusted location) are applied to every authentication by identities for each tier. The recommendation is to ensure all users are secured with multi-factor authentication, for accessing any application or workload. But having separate credentials for each tier will enable applying granular access policies relevant for each tier, such as adequate multi-factor authentication methods, and device compliance checks.
- **Facilitates Incident Response** in the event of a compromise, by identifying the credentials that would have certain permissions to perform an operation in each tier, and revoking the access without impacting other parts of the business.
- **Helps strengthen accountability** by ensuring that users are only provided credentials that aligns with their responsibilities and limits access that is no longer needed.

Separation of credentials for accessing and managing Microsoft Azure would involve creation of separate identities aligned with performing activities for each tier. Tier 0 identities will be aimed at managing Microsoft Entra, and managing any Tier 0 servers hosted in Azure Subscriptions such as Active Directory, and SAML federation servers. Tier 1 identities will be provisioned to manage any Tier 1 application workloads hosted in Azure or applications integrated with Microsoft Entra ID.

The following controls should be considered to ensure credential tiering and protecting critical access to resources:

Limit exposure for Tier 0 administrator credentials

One of the fundamental guidelines of tiering, is using separate sets of credentials for performing administrative and operational tasks in each tier. Administrators in the cloud should be separate cloud-only credentials that are not federated with or synchronized from any other environment.

For instance, if there is Active Directory in the environment, which has been synchronized with Microsoft Entra ID using Microsoft Entra ID Connect, it is recommended that any Tier 0 administrator credentials in the Azure environment should not be an Active Directory synchronized account. This helps in limiting paths of lateral movements between environments, and eliminates risk of a compromise in AD to allow attackers to take control of Azure, and vice versa.

Examples of Tier 0 roles in Azure include Microsoft Entra ID Global Administrator, Privileged Role Administrator, Privileged Authentication Administrator, Intune Administrator, Device Administrators and certain other privileged roles that can make configuration changes in Microsoft Entra platform.

Tier 0 roles would also include any Owner role-based-access at the tenant root Management Group level, as this role gives Owner access across all Subscriptions in the tenant. If there are PAM servers, Domain Controller virtual machines and any Tier 0 servers in Azure, any user or service principal identity having direct permission to access or modify the Tier 0 resources, should be treated as a Tier 0 identity.

All the Tier 0 identities mentioned in the earlier example should be Microsoft Entra cloud-only accounts, and not synchronized from other identity provider platforms.

Separate productivity accounts for the tiered identities

Another control to put in place for a Tier 0 or Tier 1 identity, is to ensure the users are provided a separate productivity identity to access applications such as Office 365, and other SaaS productivity applications. This indicates that any Tier 0 cloud-only identity or identities managing Tier 1 resources (i.e., holding high privileges in the cloud platform) should not have assigned mailboxes or be allowed to access any productivity applications.

This measure is to ensure that administrator credentials in each tier are not exposed to endpoint devices running SaaS productivity applications, thereby reducing the surface area of credential compromise for the Tier 0 and 1.

Reduce access to critical resources by limiting permissions through tiering

In Microsoft Azure, permissions to manage resources are assigned directly to the user identities. Since permissions assigned to an identity are cumulative, assigning permissions to manage resources in multiple tiers (such as Tier 0, 1 and 2) to a single user identity, results in that account having cross-tier privileges. This is especially risky as an attacker who compromises that account through a lower tier workload (such as compromising a Tier 1 application having Internet egress), can escalate privileges and gain control of workloads in a higher tier such as Tier 0, leading to more data theft, or total domain takeover.

This is why credential tiering is adopted to ensure separate identities are created to align to manage resources in the respective tiers.

Restrict Inheritance of Privileges

Limiting inheritance of permissions across cloud resources is essential for maintaining a least privilege security model and preventing unauthorized access. Each cloud provider has its own inheritance model. Typically, permissions granted at a higher level (like an organization or project) are inherited by resources at lower levels (folders, services). Instead of granting overly broad permissions at higher levels we need to explicitly define permissions for each resource or group of resources at the appropriate level. Most cloud platforms allow creation of 'Deny' rules. Deny rules can override inherited permissions when applied at a certain resource hierarchy level. Strategically placing deny rules can block specific actions or permissions where inheritance is not desired.

Administrators also need to set up regular reviews to audit permission structures and ensure they align with the least privilege model for managing resources in the cloud and identify any unintended inheritance.

Restrict Inheritance of Privileges in Microsoft Azure

In Microsoft Azure cloud, RBAC permissions inherit from management groups to subscriptions and resource groups. This means that if a role is assigned at the management group level, it will apply to all subscriptions and resources within that management group. Role assignments at the root management group level such as Management Group Owner or Contributor is especially risky since these accounts will have create, update or delete permissions to all resources within the Azure tenant. Such permissions should be provided with caution keeping separation boundaries in mind i.e. have Azure Cloud only, non-federated credentials from another IAM solution hold these roles. It is a best practice to always assign permissions at the desired subscription or resource group level.

These are some of the controls we can use to ensure limiting inheritance of permissions in Azure, and operate with principle of least privilege:

- **Explicitly Define Permissions at Lower Levels:** Instead of assigning roles at the management group or subscription level, granular role assignments should be performed at the resource group level. Separate sets of credentials should be used for Tier 0 and Tier 1 operations. This is necessary since permissions are cumulative in Azure, which is why if a user account has access to manage subscriptions hosting both Tier 0 and 1 resource, the account ends up getting both permissions. This results in access paths to escalate from a Tier 1 to a Tier 0, which can be misused by an attacker to compromise the tenant and more workloads.
- **Use of Azure Policy:** Azure Policy is used to enforce specific configurations and restrictions on resources within the management group, subscriptions, or resource groups. Specific security configurations can be enforced through Azure Policy initiatives at the management group or subscription level to ensure resources within the specific tiers conform to baseline security standards.
- **Use of Deny Assignments:** The deny assignments capability can be used at the subscription scope for Tier 0 subscriptions, to ensure that unintended inherited privileges are blocked on Tier 0 resources. However, this capability in Azure has certain limitations; such as, this capability is only available through Deployment Stacks using Azure Resource Manager or Bicep based deployment templates. Resources created outside this deployment mechanism will still have inherited permissions, which cannot be blocked. Refer to the known issues and limitations of this Microsoft capability⁷.

Use of Managed Administrative Workstations with Restrictive Controls

A Privileged Access Workstation (PAW) is defined as a dedicated, restrictive workstation that is used to perform sensitive tasks that require high privileges. It's designed to isolate privileged activities from regular user activities, reducing the risk of compromise from malware, phishing attacks, or other threats.

The use of PAWs to manage on-premises workloads aligned with tier model architecture, is a well-known mechanism to prevent attacks related to LDAP, Kerberos protocols such as Pass-the-hash and Over-Pass-the-Hash. The concept of PAWs was introduced as organizations started adopting more structured approaches to privileged access management (PAM). This is when industry frameworks and guidelines, like Microsoft's Enhanced Security Administrative Environment (ESAE) and the National Institute of Standards and Technology (NIST) guidelines, began emphasizing the importance of using dedicated workstations for privileged tasks.

To manage cloud workloads, use of a dedicated administrative workstation such as a PAW, is still very much relevant since it reduces risk of credential theft for administrators, and provides restricted access to public Internet facing

endpoints. Some of the controls that are implemented to build a Cloud PAW device for cloud workload management include the following:

- **Reduce risk of token-based attacks:** The Cloud PAW device is hardened through security controls such as device management, group policies, firewall policies, antivirus, and app locker. This ensures that the administrators can only connect to specific endpoints in the Internet (such as the cloud consoles, cloud shell) and run specific tools (command line tools) on the device to carry out administrative tasks. Most token-based attacks originate from productivity applications such as email, Drive and other applications. Such applications are blocked to be run within the Cloud PAW, thereby reducing the risk of token-based thefts.
- **Reduces privilege escalation paths:** A Cloud PAW is built to only allow an administrator to log in using specific credentials meant to perform operations aligned to the respective tiers. This means a Tier 0 Cloud administrator having roles assigned to manage multiple cloud IAM planes, gets a Tier 0 workstation and account, which is used to manage the IAM environment. This administrator credential issued to be used within the PAW device, will be separate from the credential used to perform day to day productivity operations such as access emails, Drive or similar applications. The PAW device enforces device segregation to perform administrator versus productivity tasks, thereby reducing surface area for attacks, and any risk of privilege escalation to compromise Tier 0 workloads.

The Cloud PAW or Cloud enrolled PAW device to manage multi-cloud environments, can be built using solutions such as a SaaS Device Management provider, SaaS Identity and Access Management platform. Depending on the solutions an organization has in place, the following configurations should be in place to build a Cloud PAW device:

- **Device Management solution:** Administrative workstations should be enrolled to a device management solution such as Microsoft Intune, that enforces organization compliance policies, pushes enrollment certificates for encryption and applies necessary security controls. To enroll the Cloud PAW device for critical resource management, specific Intune Compliance policies and profiles for Tier 0,1 and 2 need to be created. The profile is applied to the users and groups of the respective tiers. The physical device to be enrolled as the Cloud PAW, will install the necessary Intune configurations based on the tier it is meant to manage. For example, a PAW device enrolled to Intune, to manage Tier 0 Cloud workloads will have the required certificates and password policies pushed, endpoint firewall and proxy settings installed, local administrator access disabled, Microsoft Defender antivirus and endpoint detection and response solution installed, and device hardened.
- **Dynamic Access control policies:** Access to critical Tier 0 services in the cloud should only be provided under specific verifiable conditions. Such conditions can include additional checks on user access, such as the authentication request is coming from a compliant device, known IP addresses, multi-factor authentication, and dynamically calculated risk factors. Access to privileged resources should be blocked if an authentication attempt does not meet such necessary criteria. These controls can be enforced through Microsoft Entra Conditional Access policies. For example, Conditional Access can block a Tier 0 administrator from accessing the Microsoft Azure portal or other administrator portals, unless the user completes a phishing-resistant MFA challenge, and comes from a known privileged access workstation (PAW).
- **Security Configurations:** A managed administrator workstation should have baseline security controls applied. Such security controls can be applied through solutions such as Microsoft Intune configuration policies, and ADMX settings. Such settings are meant to apply to the Tier 0 users and groups to enforce necessary security controls such as Bit locker drive encryption, password policies, Firewall and proxy settings, Antivirus settings and so on. It is also important to block unnecessary capabilities within the OS such as telemetry, location sense services, voice

assistant, any data analytics services such as the 'recall' feature in Windows 11, which can increase attack surface area within a highly restricted administrator workstation.

- **Firewall and Proxy solutions:** It is crucial to enforce Firewall at the endpoint device level for privileged access workstations, in addition to having traffic routed via an Enterprise firewall and perimeter network services. The firewall and proxy settings on the PAW device should be enforced to ensure any inbound traffic to the device is blocked, and outbound traffic is only allowed for specific allow-listed endpoint URLs/Ip addresses. The allow-listing of Internet facing addresses is done based on which environments the PAW is aimed to administer. For example, if an organization has Tier 0 resources in Azure and GCP, the specific administrator relevant URLs for both the Cloud platforms can be allow-listed at the device firewall.
- **Strong phishing-resistant multi-factor authentication:** Phishing-resistant MFA is a type of multi-factor authentication designed to be immune to phishing attacks. It achieves this by eliminating the use of shared secrets or codes that can be intercepted and replayed by attackers. Most IAM solutions provide multiple MFA options. A comparison of various MFA methods reveals that certain methods are more prone to compromise than others. Examples of weaker methods include SMS One-Time Passcode, Phone authentication, and Push notification on an authenticator app, which have been known to be compromised by threat actors such as UNC3944. The most resilient MFA methods are the ones that satisfy the NIST Authenticator Assurance Level 3 requirements:

1. verifier impersonation resistant
2. verifier compromise resistant
3. verifies authentication intent

These requirements are the foundation for phishing-resistant methods that use a hardware or verification factor that fulfills the following criteria:

1. **No Shared Secrets:** Relies on public-key cryptography and cryptographic challenges, eliminating the vulnerability of shared codes or one-time passwords (OTPs) that can be phished.
2. **Verification of Origin:** Ensures the authentication request comes from the legitimate website or service and not a malicious imposter.
3. **User Presence and Intent:** Often requires a physical action or biometric verification from the user, adding another layer of protection against automated attacks.

Examples of phishing resistant methods in Microsoft Entra ID include FIDO 2.0 security keys, certificate-based authentication, Windows Hello for Business authentication.

It is crucial to enforce phishing-resistant MFA methods on every authentication for privileged users managing critical resources in the cloud through a PAW.

- **Continuous monitoring and detection:** The Cloud PAW device should have its device logs forwarded to the SIEM solution. This is to ensure the organization can create necessary alerts to detect any anomalous activities from the logins within the device to flag any potentially malicious activities. This would help organizations quickly identify and remediate potential threat actor activity and protect privileged accounts more effectively.

Apply Network Segmentation and Network Security Controls

Organizations with more than one cloud platform have workloads deployed in multiple cloud resources. Cloud platforms will use virtual network components and cloud native components for its connectivity. There is often interconnectivity set up between different clouds and on-premises to ensure communication over the network. To protect the critical resources from network-based attacks, there needs to be network boundaries defined where we place critical services, applications and enable users to securely connect to them.

In the following sections we will go into the best practices to implement a secure distributed network that will support hosting critical and business resources in a multi-cloud environment.

Identify and Classify Assets

The first step to ensuring resources within various tiers are placed within the right network boundaries, we will first need to define the network segments. The Tier 0 or critical services when hosted on IaaS compute engines, should not have direct communication to the Internet.

Applications on the other hand are Tier 1 resources, and will have separate networking configurations. Tier 1 resources should be placed within a separate network. The exercise to define network boundaries begins with building an inventory of the networking resources and putting classifications for them to define severity of the asset Tier they should be placed in, communication needed to the Internet, and so on.

- **Inventory:** Organizations need to create a comprehensive inventory of all resources across their multi-cloud environment, including virtual machines, databases, storage buckets, and applications.
- **Classification:** The next step will be to classify assets based on their criticality and sensitivity. Consider factors like data classification, regulatory requirements, business impact, and potential attack surface.

Design the Segmentation Strategy

Once the organization has a classification of networking components for the various cloud platforms, the next step is to define the network segments. An example would be:

An organization decides to create two possible segments for the services deployed. The first segment is for Tier 0 trusted services such as Domain Controller resources, or any OAUTH identity services. The second segment is meant for business applications which are used by the organization's internal users.

Network security rules can be built and firewall rules for the first segment to block any traffic inbound from the Internet.

For segment two resources, specific inbound traffic from the Internet may be allowed via the enterprise firewall such as HTTPS.

The following points detail a segmentation strategy:

- **Micro-segmentation:** Divide the multi-cloud network into smaller, isolated segments based on asset classification, function, or business unit. This helps define granular network rules to allow and block specific kinds of traffic and thereby limits the lateral movement of attackers in case of a breach.
- **Zero Trust:** Organizations should adopt a Zero Trust approach, where every access request is verified regardless of its origin. This includes strong user authentication and authorization mechanisms for users based on dynamic access conditions such as device compliance, IP address of origin, and device platform.

- **Network Virtualization:** Leverage network virtualization technologies like virtual private clouds (VPCs), virtual networks (VNETs), and software-defined networking (SDN) to create and manage isolated segments across different cloud providers.

Implement Security Controls

The network communication should be set up to ensure any traffic to the Internet goes through the firewall. Firewall should enable domain name service resolution for any Internet traffic. All data flow within the network should use end-to-end encryption using TLS 1.2 and above. Insecure protocol usage should not be minimized as much as possible. The following sections list the capabilities that will secure all network traffic between components, and users accessing them.

- **Firewalls:** Deploy firewalls at various levels, including perimeter, between the network segments, to control traffic flow and enforce access policies.
- **Intrusion Detection and Prevention Systems (IDPS):** Implement IDPS to monitor network traffic for suspicious activity and block potential attacks.
- **Security Groups and Network Access Control Lists (ACLs):** Use security groups and ACLs to filter traffic based on source/destination IP addresses, ports, and protocols.
- **Encryption:** Encrypt data at rest and in transit to protect sensitive information from unauthorized access.
- **Vulnerability Management:** Regularly scan for vulnerabilities and apply patches to keep the environment secure.
- **Zero Trust Network Access (ZTNA):** Implement ZTNA solutions to provide fine-grained access control based on user identity, device posture, and context. ZTNA solutions can replace VPN solutions, which often lack the capability to provide granular access to specific resources. VPN solutions commonly provide access to a wider network range, which if compromised can allow a threat actor to move laterally in the network. ZTNA solutions (also referred as Security Service Edge) acquire traffic from endpoint clients and provide access to specific applications and services on an internal network or the internet. This helps reduce the attack surface and prevent lateral movement. Examples of such solutions include Microsoft Entra Global Secure Access, Zscaler Private and Internet Access, and Palo Alto solutions.
- **Cloud Access Security Broker (CASB):** CASB solutions can help gain visibility into and control over cloud application usage, ensuring compliance and preventing data leakage.
- **Secure Web Gateway (SWG):** SWG solutions are meant to filter web traffic, block malicious content, and protect against web-based threats. Organizations can also use such solutions to enforce blocking usage of sanctioned applications.

Apply Scalable Security Configurations and Governance

Every public cloud platform has their unique sets of capabilities to enforce security configurations and compliance of resources at scale. In this section let's dive into the baseline configurations that need to be enabled for resources classified within each tier.

- **Restrict administrator console/shell access:** Access to the administrator portals for Microsoft Cloud, any administrator command-line tools, and access to Azure Cloud Shell should be restricted to specific Tier 0

identities only. If the organization uses CI/CD deployment approach with Infrastructure-as-Code, the access to the administrator portal for Azure should be highly restricted.

- **Plan for an emergency access scenario:** Organizations should plan to create identities to be used in the event of an emergency such as cyberattacks leading to a total environment compromise, system failures, and so on. A Microsoft Entra ID cloud-only user account holding Global Administrator role should be treated as an emergency access account, have phishing-resistant FIDO 2.0 based multi-factor authentication enabled. Credentials for emergency access accounts should be securely stored and such accounts should be monitored and governed. In addition to holding Global Administrator role, additional Azure cloud only identities holding root management group Owner role can be created to have Azure subscription administrator privileges across the entire Azure environment. Such accounts should also have FIDO 2.0 based security keys enabled for MFA, and to be used in the event of an emergency access scenario. At least one of the emergency access accounts should be considered to be excluded from all Conditional Access policies, so that the account can be used to recover Azure in the event of a domain-wide compromise.
- **Encryption at rest:** The resources within the cloud need to have encryption at rest enabled to secure the data hosted within the resources. The encryption can use either platform-managed key or use bring-your-own-key (BYOK) scenarios. The keys should have proper governance to ensure periodic rotation and be stored securely. The storage of such encryption keys should be outside of the cloud platform to ensure they can be recovered in the event of a cloud platform compromise.
- **End-to-End Encryption:** All resources communicating between each other in the cloud networks and resources hosted to communicate through the Internet should be using encryption with the latest transport layer security (TLS) protocols.
- **Protect secrets and certificates:** All secrets, certificates, encryption keys, API key, and more. should be protected and stored within credential storage vaults. The credential vault should use secure storage of the secrets and be recoverable in the event of a compromise or disaster affecting the entire cloud platform. Only authorized users should have access to the vaults.
 - Any code or scripts within the environment should not use secrets in plaintext. Secrets should be referenced within the code to the vaults and handled in a secure manner.
- **Regular patching and vulnerability scanning services:** All servers and endpoint client devices within the environment should receive regular security patches and updates. Solutions should be in place to flag any common vulnerabilities detected in the environment.
- **Restrict Direct Public Access:** In critical environments, resources such as S3 buckets, GCP Cloud Storage, Azure Blob Storage, Key Vaults should not be hosted with public access enabled. This makes the resources discoverable on the Internet. Such resources should have restricted access on the network as much as possible from a range of IP addresses.
- **Restrict resource creation with proper controls:** In critical cloud environments such as Tier 0, and production environments, resource creation within the environment should be restricted by conditions. Such conditions could include regions where creation of resources is allowed, size of Compute Engines or Virtual Machines that are allowed, naming standards to comply, and tags present in the deployment template. This is necessary because Mandiant has observed multiple attackers create attacker controlled VMs in the compromised cloud environment, to perform further malicious activities. This is why such conditions are necessary, in addition to other restricting controls such as RBAC permissions, and network segmentation, as shared in earlier sections.

Scalable Security Configurations in Microsoft Azure

Azure Policies can be utilized for implementing and enforcing baseline security across various tiers in the Azure cloud environment.

The steps for implementation are listed as follows:

1. Define Tiered Security Baselines:

- **Identify Tiers:** Resources will be classified into different tiers based on their service criticality and tiering guidelines shared in the earlier sections of this paper.
- **Baseline Security Requirements:** Specific security requirements for each tier are defined, including network configurations, access controls, data protection measures, and logging/monitoring settings.
- **Translate into Azure Policies:** These requirements are converted into Azure Policy definitions that will enforce the necessary configurations.

2. Utilize Azure Policy Features:

- **Built-in Policies:** Azure's built-in policy definitions can be leveraged as much as possible, to create baseline policies related to security and compliance requirements for each tier.
- **Custom Policies:** Custom policies are created using JSON templates to tailor the enforcement to specific requirements that aren't covered by built-in policies.
- **Policy Initiatives:** Policy definitions are then grouped into policy initiatives to apply and manage a set of policies as a single unit. This helps in applying baseline policies for each tier with ease.

3. Implement Tiered Enforcement:

- **Assign Policies at Different Scopes:** Policies are assigned to appropriate management groups such as Tier 0/1/2 to target specific tiers.
- **Enforce Compliance:** Policy actions can be set to "Deny", "Audit" or "DeployIfNotExists" to proactively enforce compliance settings on resources or monitor deviations from the defined baseline.
- **Remediation Tasks:** Azure Policy's remediation feature can be used in certain scenarios to automatically fix non-compliant configurations.

4. Monitor and Adapt:

- **Defender for Cloud:** Microsoft Defender for Cloud to track policy compliance across the multi-cloud environment and protect PaaS resources. Other cloud environments such as AWS, GCP can be integrated with Defender for Cloud to apply the policies and monitor resources uniformly.
- **Continuous Monitoring:** Azure Monitor and Log Analytics workspace or another SIEM solution can be used to track policy events, identify trends, and proactively address potential security risks.
- **Review and Update:** Organizations are encouraged to regularly review the tiered security baselines and adjust policies to accommodate evolving security requirements.

Consider the following example:

Tier 0 (Mission-Critical) policies and enforcement:

- **Policy:** Require encryption at rest for all virtual machine disks
- **Enforcement:** “Deny” creation of virtual machines without encryption capability

Tier 1 (Sensitive) policies and enforcement:

- **Policy:** Restrict network access to specific IP ranges
- **Enforcement:** “Audit” non-compliant network security groups and trigger alerts

Some of the key advantages to using Azure Policies include:

- **Centralized Management:** Security baselines across the entire organization can be managed from a single control plane.
- **Proactive Enforcement:** Policies will help prevent misconfigurations and non-compliant deployments.
- **Scalability:** As the organization grows, the policies to ensure compliance are applied by default through the inheritance hierarchy to new projects and resources. This ensures scalability of the desired security model.
- **Flexibility:** Policies can be customized to meet the specific industry and regional security and compliance requirements.

Organizations are encouraged to regularly review and update the Azure Policies to adapt to evolving security threats and compliance needs.

Monitoring and Detection

While tiering limits lateral movement within the environment, determined attackers may still attempt to escalate privileges or move between tiers. Continuous monitoring and detection can help identify such attempts and prevent them from succeeding. When implementing tiering of resources within a cloud environment, we need to ensure proper levels of logging is enabled on all resources. The logs should be collected from the cloud environments and forwarded to a central Security Information and Events Management (SIEM) solution. In the event of a compromise, logs are the primary source of information for security operations teams to investigate the attacker path, and properly remediate and recover the environment. It is recommended that organizations create detection rules within their SIEM solution to proactively detect any anomalies and trigger an incident if needed to ensure expedited recovery from an attack.

The following types of logs should be enabled at minimum in the various tiers:

- Identity logs that include any sign-in and audit events for all types of accounts (users, service accounts, machine identities)
- Security logs
- System logs
- Any productivity application logs

Tier Model in Microsoft Azure

- Firewall logs
- Proxy, VPN and any perimeter solution logs
- Endpoint device logs for users of all tiers

The following briefly captures the typical anomalies to alert for while building monitoring and detection scenarios as part of the tiering exercise:

- **Anomalous patterns:** SIEM systems can be tuned to generate alerts on detecting any anomalies from identified patterns. Certain indicators can alert a potential security breach to the Security Operations Center (SOC) team. Anomalies can include consecutive failed login attempts, consecutive logins from different time-zones within a short time frame, disabled accounts getting re-enabled, login events at unusual times, running of malicious binaries, scripts and so on.
- **Credential reset events:** High value user accounts and administrator accounts should be enabled with alerting to trigger notification in the event of password resets, multi-factor authentication method updates and so on.
- **Detection of potential data exfiltration:** Alerting should exist for critical resources to flag any events where large amounts of data is copied to unexpected drive locations or external IP addresses, bulk permission updates on sensitive files, and bulk deletion events.
- **Configuration Drift Detection:** Configurations set in the environments to ensure baseline security and compliance will modify over time depending on how the organization scales, to keep up to date with updated guidelines. This can potentially introduce security vulnerabilities or non-compliance. Monitoring will help identify configuration drift and ensure resources stay aligned with the security policies defined for their tier. Specific unexpected drifts can be alerted to the SOC teams such as updating of federated domains on an Identity provider to add new unknown domains, deletion of existing domains, and disabling encryption and reducing encryption levels in resources. These are common behaviors Mandiant sees attackers perform in compromised environments.

In an environment with multiple cloud platforms in use, along with on-premises infrastructure, organizations are encouraged to explore solutions that support Multi-Cloud Security and Posture Management (CSPM). CSPM solutions can help provide visibility, continuous monitoring, automated remediation to address any misconfigurations, vulnerabilities and compliance issues across different cloud platform resources.

Key features and functionalities of a CSPM solution that can help achieve an elevated level of security for multi-cloud resources include:

- **Visibility and Asset Inventory:** CSPM tools provide a comprehensive view of the cloud assets, including compute instances, storage services, databases, and network configurations, across multiple cloud providers.
- **Configuration Assessment:** It helps to continuously assess the configurations of the cloud resources against security best practices, compliance standards (e.g., PCI DSS, HIPAA, GDPR), and internal security policies.
- **Misconfiguration Detection:** SOC teams can set up alerting for misconfigurations within critical Tier 0 or Tier 1 resources, that could lead to a bigger incident and quickly apply mitigations as needed.
- **Vulnerability Management:** CSPM solutions can scan for vulnerabilities and prioritize them based on severity and risk.

Tier Model in Microsoft Azure

- **Remediation and Automation:** Organizations can set up actionable recommendations for the relevant teams and automated workflows as applicable, to fix misconfigurations and mitigate any potential vulnerabilities.
- **Incident Response:** CSPM solutions can support incident response efforts by providing visibility into cloud security events and facilitating forensic investigations.
- **DevSecOps Integration:** Most CSPM solutions integrate with DevOps processes to ensure security is built into the development and deployment lifecycle.

Tier Model in Google Cloud Platform (GCP)

The controls detailed in this section establish a baseline/Tier model architecture for GCP:

- Resource segregation through logical containment of privileged workloads
- Separation of credentials
- Restrict inheritance of privileges
- Use of managed administrative workstations with restrictive controls
- Apply network segmentation and network security controls
- Apply security configurations to resources
- Monitoring and Detection

Resource Segregation in GCP

Overview

Tiering concepts focus on defining segregated zones to place resources based on their functionality. Traditionally the first tier (Tier 0) is reserved for critical resources responsible for privileged access, managing and storing credentials or performing authentication. This means Identity servers, PAM servers, Certification servers and any resources having a direct privilege to make changes to these services should fall under this category.

The second tier (Tier 1) is the business applications or data plane layer. Any resources in the cloud supporting business applications should be categorized under Tier 1.

Tier 2 is for hosting resources responsible for user access. This includes Virtual Desktop interfaces, productivity workstations, and customer and partner identity solutions.

Segregating cloud resources into their individual tiers helps to enhance security posture by enforcing the following:

- It helps limit role-based access control (RBAC) permissions assigned on those resources to specific identities. By doing this we are able to cut down access to critical resources and scope the access to only the identities that are responsible for managing them.
- Applying baseline security configurations in a scalable manner to resources that fall in the same tier.
- Limit paths of lateral movement and privilege escalation between resources aligned with different tiers.

GCP Implementation

Resource segregation in Google Cloud Platform (GCP) is achieved through a hierarchical structure of organizations, folders, and projects. This separation through folder structure limits the blast radius for security incidents, by ensuring consistent security configurations are applied across workloads falling within the same tier, and prevents accidental access for users to critical resources.

Tier Model in Google Cloud Platform (GCP)

The following contains details of the structural elements of resource hierarchy within GCP:

- **Organization:** This represents the root of the resource hierarchy, and usually represents the organization or entity that owns the GCP environment. Organizations help centralize billing and identity management. Policy configurations and permissions inherit from organizations to all resources underneath them.
- **Folders:** Folders are used to group resources together based on departments, type of workload, environments and other criteria. Folders will inherit policies from their parent organization. Targeted policy configurations can also be applied at the folder level to enforce specific configurations for all resources within the same folder.
- **Projects:** Projects are the smallest unit of resource hosting containers within a GCP environment. Projects host compute resources, storage, networking and PaaS services.

Figure 9 shows an example of resource segregation to enforce tiering in a GCP environment:

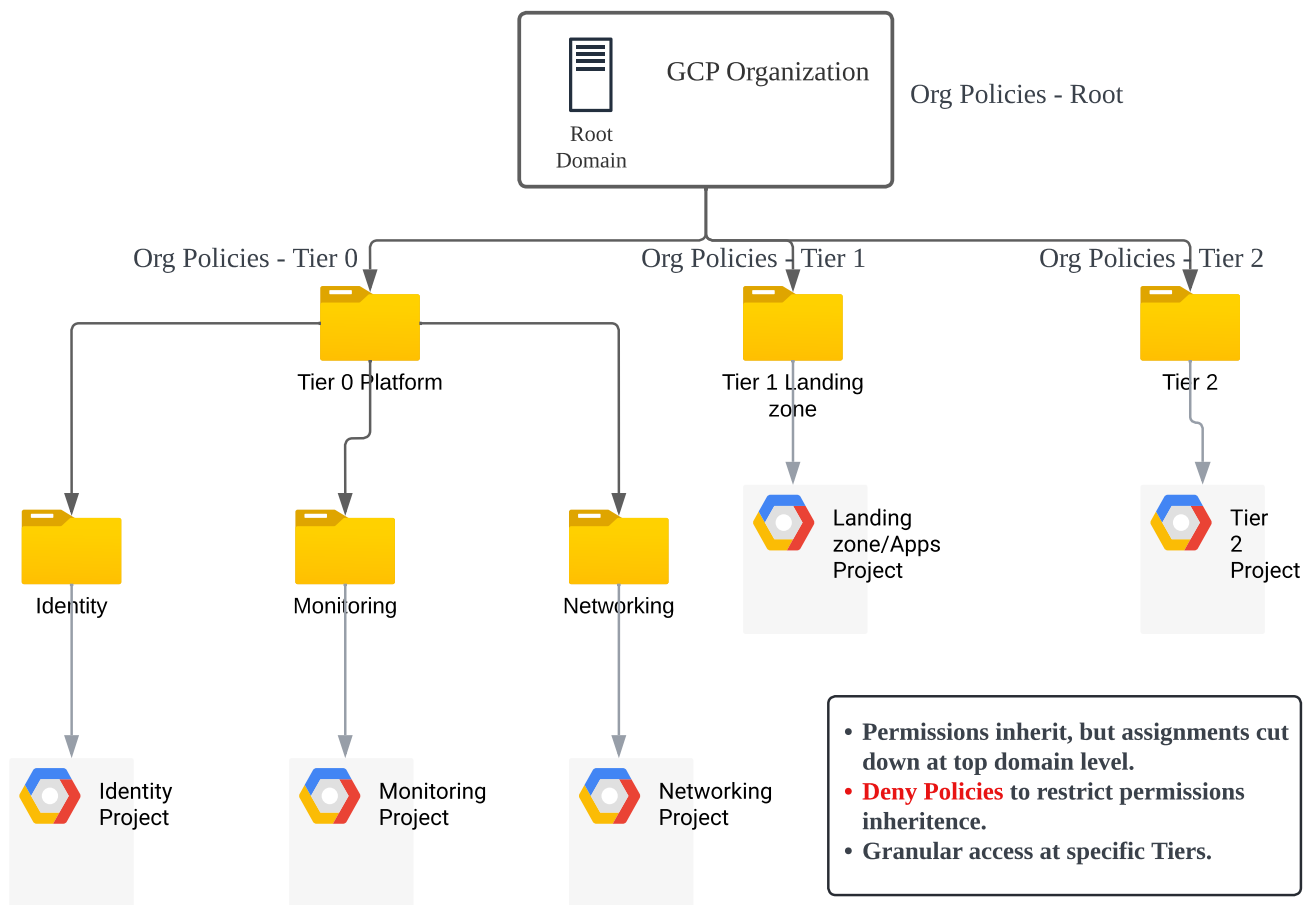


Figure 9: GCP resource segregation to host workloads in tiers

Credential Tiering

Credential tiering is a fundamental security practice that significantly brings down the risk of various identity-based attacks. In the cloud, using unique sets of identities for different environments and levels of access, reduces the blast radius of an attacker in the event of a credential compromise, as the attacker will only be limited within that scope or environment. Following are the advantages of using separation of credentials as a mechanism to enforce tier model:

- **Minimize blast radius** in the event of a compromise, by limiting the attacker's access to specific scope and permissions.
- **Limits lateral movement** paths by using separate sets of credentials for administrative and productivity tasks.
- **Enforce least privilege** by ensuring that accounts are provided permissions for only the operations they perform.
- **Enforce dynamic verifiable authentication controls** for the respective tier users, by segregating identities as Tier 0, 1 or 2, ensuring adequate multi-factor authentication and other verifiable controls (such as compliant device, and trusted location) are applied to every authentication by identities for each tier. The recommendation is to ensure all users are secured with multi-factor authentication, for accessing any application or workload. But having separate credentials for each tier will enable applying granular access policies relevant for each tier, such as adequate multi-factor authentication methods, and device compliance checks.
- **Facilitates Incident Response** in the event of a compromise, by identifying the credentials that would have certain permissions to perform an operation, and revoking the access without impacting other parts of the business.
- **Helps strengthen accountability** by ensuring that users are only provided credentials that aligns with their responsibilities and limits access that is no longer needed.

Separation of credentials for accessing and managing GCP Cloud would involve creation of separate identities aligned with performing activities for each tier. Tier 0 identities will be aimed at managing GCP Cloud Identity, and managing any Tier 0 servers hosted in GCP such as Active Directory, and SAML identity federation servers. Tier 1 identities will be provisioned to manage any Tier 1 application workloads hosted in GCP or applications integrated with Cloud Identity or Google Workspace.

User identities in GCP (Google Cloud Platform) are created primarily through two main methods:

1. Managed Google Accounts (Cloud Identity or Google Workspace): These are accounts managed entirely within Google's ecosystem. These accounts are created through the Cloud Identity or Google Workspace admin consoles. Once created, these accounts can be granted access to various GCP resources based on their assigned roles.

- **Cloud Identity** is a standalone identity management service best suited for organizations that don't need the full suite of Google Workspace collaboration tools.
- **Google Workspace** combines identity management with Gmail, Drive, Docs, and other productivity apps.

2. Federated Identities: These allow organizations to leverage their existing identities from external identity providers (IdPs) like Active Directory, Okta, or Microsoft Entra ID to access GCP resources. GCP supports several federation protocols, including SAML, OIDC, and Workspace Identity Federation. This approach is beneficial for organizations that want to centralize identity management and maintain consistency across various cloud and on-premise environments.

Tier Model in Google Cloud Platform (GCP)

The following controls should be considered to ensure credential tiering and protecting critical access to resources:

Limit exposure for Tier 0 administrator credentials

One of the fundamental guidelines of tiering, is using separate sets of credentials for performing administrative and operational tasks in each tier. Administrators in the cloud should use separate cloud-only credentials which are not federated with or synchronized from any other environment. This approach helps in limiting paths of lateral movements between environments, and eliminates risk of a compromise in another identity provider to allow an attacker to take control of a GCP environment—and vice versa.

GCP allows provisioning of identities through two mechanisms—Cloud Identity or Google Workspace—and through integration with an external identity provider. Creating a user account using Google Cloud Identity or Workspace ensures that the account resides entirely in Google cloud. It is recommended that highly privileged roles in GCP Cloud should be created in Cloud Identity or through Workspace.

The privileged Tier 0 GCP users should not have any access to productivity applications or SaaS applications.

At a minimum, the following roles in GCP should be considered as Tier 0 and be created in GCP Cloud Identity or Workspace:

- **Organization administrator role:** This role should be considered for emergency access scenarios and be restricted from logging in for conducting day to day operations.
- Any users having access at **root organization, Tier 0 Folders or Tier 0 Projects**
- **Cloud Identity or Workspace** administrators

Reduce access to critical resources by limiting permissions through tiering

In GCP Cloud, permissions to manage resources are assigned directly to the user identities. Since permissions assigned to an identity are cumulative, assigning permissions to manage resources in multiple tiers (such as Tier 0, 1 and 2) to a single user identity, results in that account having cross-tier privileges. This is especially risky as an attacker who compromises that account through a lower tier workload (such as compromising a Tier 1 application having Internet egress), can escalate privileges and gain control of workloads in a higher tier such as Tier 0, leading to more data theft, or total domain takeover.

This is why credential tiering is adopted to ensure separate identities are created to align to manage resources in the respective tiers.

Restrict Inheritance of Privileges

Limiting inheritance of permissions across cloud resources is essential for maintaining a least privilege security model and preventing unauthorized access. Each cloud provider has its own inheritance model. Typically, permissions granted at a higher level (like an organization or project) are inherited by resources at lower levels (folders, services). Instead of granting overly broad permissions at higher levels we need to explicitly define permissions for each resource or group of resources at the appropriate level. Most cloud platforms allow creation of 'Deny' rules. Deny rules can override inherited permissions when applied at a certain resource hierarchy level. Strategically placing deny rules can block specific actions or permissions where inheritance is not desired.

Administrators also need to set up regular reviews to audit permission structures and ensure they align with the least privilege model for managing resources in the cloud and identify any unintended inheritance.

Restrict Inheritance of Privileges in GCP

By default, Role-Based Access Control (RBAC) permissions in GCP inherit from the organization level down to folders and projects. This means that roles granted at the organization level will automatically apply to all resources within that organization, including folders and projects. The following controls can help restrict RBAC inheritance and enforce least privilege access model:

- **Explicitly Define Permissions at Lower Levels of resource hierarchy:** Users should be granted specific permissions at the folder or project level based on their responsibilities. Administrators should avoid granting permissions such as Owner or Editor at the organization or folder levels. Instead, granular predefined roles can be assigned to users for specific projects and resources.
- **Use IAM Conditions:** IAM conditions are used to define and enforce conditional, attribute-based access control for Google Cloud resources. It can be used to restrict the scope of inherited permissions by granting additional permissions based on specific criteria. It can be used to set fine-grained access controls based on attributes such as IP addresses, resource types, or time of day.
- **Deny Policies:** Deny policies consist of a set of rules applied at the GCP organization, folder or project level, to deny principals specific permissions on resources. Deny policies can be constructed to restrict users from accessing certain privileged resources, or performing specific actions on resources. However, administrators need to exercise caution while applying deny policies as they can lead to unintended block of access.
- **Organization Policies:** Organization Policies allows GCP administrators to set constraints on allowable configurations and permissions across the entire organization. We can use organization policies to restrict certain actions or resources, effectively limiting the inheritance of permissions.
- **Securing Service Accounts:** Service accounts need to be granted minimum necessary permissions at the project or resource level. This will prevent them from inheriting broader permissions from higher levels. Service accounts need to strictly follow least privilege RBAC as these are used for non-interactive authentication scenarios and hence cannot use multi-factor authentication. Proper access review and governance is needed to ensure service account credentials are rotated at regular intervals, and any stale accounts are stripped off the privileges.

Use of Managed Administrative Workstations with Restrictive Controls

A Privileged Access Workstation (PAW) is defined as a dedicated, restrictive workstation that is used to perform sensitive tasks that require high privileges. It's designed to isolate privileged activities from regular user activities, reducing the risk of compromise from malware, phishing attacks, or other threats.

The use of PAWs to manage on-premises workloads aligned with tier model architecture, is a well-known mechanism to prevent attacks related to LDAP, Kerberos protocols such as Pass-the-hash, and Over-Pass-the-Hash. The concept of PAWs was introduced as organizations started adopting more structured approaches to privileged access management (PAM). This is when industry frameworks and guidelines, like Microsoft's Enhanced Security Administrative Environment (ESAE) and the National Institute of Standards and Technology (NIST) guidelines, began emphasizing the importance of using dedicated workstations for privileged tasks.

To manage cloud workloads, use of a dedicated administrative workstation such as a PAW, is still very much relevant since it reduces risk of credential theft for administrators, and provides restricted access to public Internet facing

endpoints. Some of the controls that are implemented to build a Cloud PAW device for cloud workload management include the following:

- **Reduce risk of token-based attacks:** The Cloud PAW device is hardened through security controls such as device management, group policies, firewall policies, antivirus, and app locker. This ensures that the administrators can only connect to specific endpoints in the Internet (such as the cloud consoles, cloud shell) and run specific tools (command line tools) on the device to carry out administrative tasks. Most token-based attacks originate from productivity applications such as email, Drive and other applications. Such applications are blocked to be run within the Cloud PAW, thereby reducing the risk of token-based thefts.
- **Reduces privilege escalation paths:** A Cloud PAW is built to only allow an administrator to log in using specific credentials meant to perform operations aligned to the respective tiers. This means a Tier 0 Cloud administrator having roles assigned to manage multiple cloud IAM planes, gets a Tier 0 workstation and account, which is used to manage the IAM environment. This administrator credential issued to be used within the PAW device, will be separate from the credential used to perform day to day productivity operations such as access emails, Drive or similar applications. The PAW device enforces device segregation to perform administrator versus productivity tasks, thereby reducing surface area for attacks, and any risk of privilege escalation to compromise Tier 0 workloads.

The Cloud PAW or Cloud enrolled PAW device to manage multi-cloud environments, can be built using solutions such as a SaaS Device Management provider, SaaS Identity and Access Management platform. Depending on the solutions an organization has in place, the following configurations should be in place to build a Cloud PAW device:

- **Device Management solution:** Administrative workstations should be enrolled to a device management solution that enforces organization compliance policies, pushes enrollment certificates for encryption and applies necessary security controls. To enroll the Cloud PAW device for critical resource management, specific device management profiles for Tier 0,1 and 2 need to be created within the device management solution. The profile is applied to the users and groups of the respective tiers. The physical device to be enrolled as the Cloud PAW, will install the necessary security configurations based on the tier it is meant to manage. For example, a PAW device enrolled to the device management solution, to manage Tier 0 Cloud workloads will have the required certificates and password policies pushed, endpoint firewall and proxy settings installed, local administrator access disabled, any antivirus and endpoint detection and response solution installed, and device hardened.
- **Dynamic Access control policies:** Access to critical Tier 0 services in the cloud should only be provided under specific verifiable conditions. Such conditions can include additional checks on user access, such as the authentication request is coming from a compliant device, known IP addresses, multi-factor authentication, and so on. Access to privileged resources should be blocked if an authentication attempt does not meet such necessary criteria.
- **Security Configurations:** A managed administrator workstation should have baseline security controls applied. Such security controls can be applied through solutions such as Microsoft Intune configuration policies, and ADMX settings. Such settings are meant to apply to the Tier 0 users and groups to enforce necessary security controls such as Bit locker drive encryption, password policies, Firewall and proxy settings, Antivirus settings and so on. It is also important to block unnecessary capabilities within the OS such as telemetry, location sense services, voice assistant, any data analytics services such as the 'recall' feature in Windows 11, which can increase attack surface area within a highly restricted administrator workstation.

- **Firewall and Proxy solutions:** It is crucial to enforce Firewall at the endpoint device level for privileged access workstations, in addition to having traffic routed via an Enterprise firewall and perimeter network services. The firewall and proxy settings on the PAW device should be enforced to ensure any inbound traffic to the device is blocked, and outbound traffic is only allowed for specific allow-listed endpoint URLs/Ip addresses. The allow-listing of Internet facing addresses is done based on which environments the PAW is aimed to administer. For example, if an organization has Tier 0 resources in Azure and GCP, the specific administrator relevant URLs for both the Cloud platforms can be allow-listed at the device firewall.
- **Strong phishing-resistant multi-factor authentication:** Phishing-resistant MFA is a type of multi-factor authentication designed to be immune to phishing attacks. It achieves this by eliminating the use of shared secrets or codes that can be intercepted and replayed by attackers. Most IAM solutions provide multiple MFA options. A comparison of various MFA methods reveals that certain methods are more prone to compromise than others. Examples of weaker methods include SMS One-Time Passcode, Phone authentication, and Push notification on an authenticator app, which have been known to be compromised by threat actors such as UNC3944. The most resilient MFA methods are the ones that satisfy the NIST Authenticator Assurance Level 3 requirements:
 1. verifier impersonation resistant
 2. verifier compromise resistant
 3. verifies authentication intent

These requirements are the foundation for phishing-resistant methods that use a hardware or verification factor that fulfills the following criteria::

1. **No Shared Secrets:** Relies on public-key cryptography and cryptographic challenges, eliminating the vulnerability of shared codes or one-time passwords (OTPs) that can be phished.
2. **Verification of Origin:** Ensures the authentication request comes from the legitimate website or service and not a malicious imposter.
3. **User Presence and Intent:** Often requires a physical action or biometric verification from the user, adding another layer of protection against automated attacks.

Examples of phishing resistant methods include FIDO 2.0 security keys, certificate-based authentication, biometric based authentication.

It is crucial to enforce phishing-resistant MFA methods on every authentication for privileged users managing critical resources in the cloud through a PAW.

- **Continuous monitoring and detection:** The Cloud PAW device should have its device logs forwarded to the SIEM solution. This is to ensure the organization can create necessary alerts to detect any anomalous activities from the logins within the device to flag any potentially malicious activities. This would help organizations quickly identify and remediate potential threat actor activity and protect privileged accounts more effectively.

Apply Network Segmentation and Network Security Controls

Organizations with more than one cloud platform have workloads deployed in multiple cloud resources. Cloud platforms will use virtual network components and cloud native components for its connectivity. There is often interconnectivity set up between different clouds and on-premises to ensure communication over the network. To

protect the critical resources from network-based attacks, there needs to be network boundaries defined where we place critical services, applications and enable users to securely connect to them.

In the following sections we will go into the best practices to implement a secure distributed network that will support hosting critical and business resources in a multi-cloud environment.

Identify and Classify Assets

The first step to ensuring resources within various tiers are placed within the right network boundaries, we will first need to define the network segments. The Tier 0 or critical services when hosted on IaaS compute engines, should not have direct communication to the Internet.

Applications on the other hand are Tier 1 resources, and will have separate networking configurations. Tier 1 resources should be placed within a separate network. The exercise to define network boundaries begins with building an inventory of the networking resources and putting classifications for them to define severity of the asset Tier they should be placed in, communication needed to the Internet, and so on.

- **Inventory:** Organizations need to create a comprehensive inventory of all resources across their multi-cloud environment, including virtual machines, databases, storage buckets, and applications.
- **Classification:** The next step will be to classify assets based on their criticality and sensitivity. Consider factors like data classification, regulatory requirements, business impact, and potential attack surface.

Design the Segmentation Strategy

Once the organization has a classification of networking components for the various cloud platforms, the next step is to define the network segments. An example would be:

An organization decides to create two possible segments for the services deployed. The first segment is for Tier 0 trusted services such as Domain Controller resources, or any OAUTH identity services. The second segment is meant for business applications that are used by the organization's internal users.

Network security rules can be built and firewall rules for the first segment to block any traffic inbound from the Internet.

For segment two resources, specific inbound traffic from the Internet may be allowed via the enterprise firewall such as HTTPS.

The following points detail a segmentation strategy:

- **Micro-segmentation:** Divide the multi-cloud network into smaller, isolated segments based on asset classification, function, or business unit. This helps define granular network rules to allow and block specific kinds of traffic and thereby limits the lateral movement of attackers in case of a breach.
- **Zero Trust:** Organizations should adopt a Zero Trust approach, where every access request is verified regardless of its origin. This includes strong user authentication and authorization mechanisms for users based on dynamic access conditions such as device compliance, IP address of origin, and device platform.
- **Network Virtualization:** Leverage network virtualization technologies like virtual private clouds (VPCs), virtual networks (VNETs), and software-defined networking (SDN) to create and manage isolated segments across different cloud providers.

Implement Security Controls

The network communication should be set up to ensure any traffic to the Internet goes through the firewall. Firewall should enable domain name service resolution for any Internet traffic. All data flow within the network should use end-to-end encryption using TLS 1.2 and above. Insecure protocol usage should not be minimized as much as possible. The following sections list the capabilities that will secure all network traffic between components, and users accessing them.

- **Firewalls:** Deploy firewalls at various levels, including perimeter, between the network segments, to control traffic flow and enforce access policies.
- **Intrusion Detection and Prevention Systems (IDPS):** Implement IDPS to monitor network traffic for suspicious activity and block potential attacks.
- **Security Groups and Network Access Control Lists (ACLs):** Use security groups and ACLs to filter traffic based on source/destination IP addresses, ports, and protocols.
- **Encryption:** Encrypt data at rest and in transit to protect sensitive information from unauthorized access.
- **Vulnerability Management:** Regularly scan for vulnerabilities and apply patches to keep the environment secure.
- **Zero Trust Network Access (ZTNA):** Implement ZTNA solutions to provide fine-grained access control based on user identity, device posture, and context. ZTNA solutions can replace VPN solutions, which often lack the capability to provide granular access to specific resources. VPN solutions commonly provide access to a wider network range, which if compromised can allow a threat actor to move laterally in the network. ZTNA solutions (also referred as Security Service Edge) acquire traffic from endpoint clients and provide access to specific applications and services on an internal network or the internet. This helps reduce the attack surface and prevent lateral movement. Examples of such solutions include Microsoft Entra Global Secure Access, Zscaler Private and Internet Access, and Palo Alto solutions.
- **Cloud Access Security Broker (CASB):** CASB solutions can help gain visibility into and control over cloud application usage, ensuring compliance and preventing data leakage.
- **Secure Web Gateway (SWG):** SWG solutions are meant to filter web traffic, block malicious content, and protect against web-based threats. Organizations can also use such solutions to enforce blocking usage of sanctioned applications.

Apply Scalable Security Configurations and Governance

Every public cloud platform has their unique sets of capabilities to enforce security configurations and compliance of resources at scale. In this section let's dive into the baseline configurations that need to be enabled for resources classified within each tier.

- **Restrict administrator console/shell access:** Access to the administrator console for GCP and any Google Cloud Shell access should be restricted to specific Tier 0 identities only. If the organization uses CI/CD deployment approach with Infrastructure-as-Code, the access to the console should be highly restricted only for emergency scenarios.
- **Plan for an emergency access scenario:** Organizations should plan to create identities to be used in the event of an emergency such as cyberattacks leading to a total environment compromise, system failures. GCP Organization

Tier Model in Google Cloud Platform (GCP)

Administrator account can be treated as an emergency access account, have multi-factor authentication enabled, and password securely stored and governed. In addition to the organization administrator, additional GCP identities can be created with Owner permission assigned at the root organization level, to have administrator privileges across the GCP Organization, and to be used in the event of an emergency access scenario.

- **Encryption at rest:** The resources within the cloud need to have encryption at rest enabled to secure the data hosted within the resources. The encryption can use either platform-managed key or use bring-your-own-key (BYOK) scenarios. The keys should have proper governance to ensure periodic rotation and be stored securely. The storage of such encryption keys should be outside of the cloud platform to ensure they can be recovered in the event of a cloud platform compromise.
- **End-to-End Encryption:** All resources communicating between each other in the cloud networks and resources hosted to communicate through the Internet should be using encryption with the latest transport layer security (TLS) protocols.
- **Protect secrets and certificates:** All secrets, certificates, encryption keys, and API keys should be protected and stored within credential storage vaults. The credential vault should use secure storage of the secrets and be recoverable in the event of a compromise or disaster affecting the entire cloud platform. Only authorized users should have access to the vaults.
 - Any code or scripts within the environment should not use secrets in plaintext. Secrets should be referenced within the code to the vaults and handled in a secure manner.
- **Regular patching and vulnerability scanning services:** All servers and endpoint client devices within the environment should receive regular security patches and updates. Solutions should be in place to flag any common vulnerabilities detected in the environment.
- **Restrict Direct Public Access:** In critical environments, resources such as S3 buckets, GCP Cloud Storage, Azure Blob Storage, Key Vaults should not be hosted with public access enabled. This makes the resources discoverable on the Internet. Such resources should have restricted access on the network as much as possible from a range of IP addresses.
- **Restrict resource creation with proper controls:** In critical cloud environments such as Tier 0, and production environments, resource creation within the environment should be restricted by conditions. Such conditions could include regions where creation of resources is allowed, size of Compute Engines or Virtual Machines that are allowed, naming standards to comply, and tags present in the deployment template. This is necessary because Mandiant has observed multiple attackers create attacker controlled VMs in the compromised cloud environment, to perform further malicious activities. This is why such conditions are necessary, in addition to other restricting controls such as RBAC permissions, and network segmentation, as shared in earlier sections.

Scalable Security Configuration in GCP

GCP offers a combination of Organization Policies and other tools to implement baseline security within a tiered infrastructure. The application of security baselines within GCP control plane will include the following considerations:

1. Define Tiered Security Baselines

- **Identify Tiers:** Resources are to be classified into tiers based on sensitivity and criticality (e.g., Tier 0 for identity resources, Tier 1 for applications, Tier 2 for end-user environment).

Tier Model in Google Cloud Platform (GCP)

- **Baseline Requirements:** For each tier, the required security configurations are to be established such as encryption requirements, public Internet access enabled, logging standards and so on.
- **Translation to Policies:** The baseline security configurations are then used to develop specific GCP Organization Policies, IAM roles, and other security configurations.

2. Utilizing Organization Policies

- **Built-in Constraints:** GCP provides several built-in constraints within Organization Policies to restrict resource configurations. These built-in constraints can be used to define the restrictions for resources deployed to the various tiers.
- **Custom Constraints:** Custom constraints can be created using Rego (a policy language) to enforce complex requirements not covered by built-in constraints.
- **Apply Policies at appropriate Levels:** Baseline policies applicable for the whole GCP Organization can be applied to the root organization level. Any granular settings relevant for lower environments such as Tier 0, Tier 1 and so on are to be applied to the specific Folder or Project levels.

3. Enforce and Monitor Compliance

- **Enforcement Modes:** Policy constraints can be set to “Enforced” to block non-compliant configurations or “Audit” to log violations without blocking them. The setting may differ depending on how restrictive we design the various tiers to comply with the desired settings.
- **Policy Inheritance:** Policies applied at higher levels (organization or folder) automatically inherit down to lower levels unless overridden by policies applied at the lower tiers. The inheritance structure is to be kept in mind while applying various settings to avoid conflicts.
- **Monitoring:** Tools such as Policy Intelligence dashboard and Cloud Audit Logs can be utilized to monitor policy compliance and violations across the organization.

Consider the following example:

Tier 0 (Mission-Critical) constraints:

- **Constraint:** `constraints/compute.requireShieldedVm` (Enforced) - All VMs must be shielded.
- **Constraint:** `constraints/iam.allowedPolicyMemberDomains` (Enforced) - Only allow members from organization’s domain to be granted IAM roles.

Tier 1 (Sensitive) constraints:

- **Constraint:** `constraints/compute.vmExternalIpAccess` (Enforced) - Restrict external IP addresses for VMs.
- **Constraint:** `constraints/storage.uniformBucketLevelAccess` (Enforced) - Enforce uniform bucket-level access for Cloud Storage buckets.

Some of the key advantages to using Organization policies include:

- **Centralized Management:** Security baselines across the entire organization can be managed from a single control plane.
- **Proactive Enforcement:** Policies will help prevent misconfigurations and non-compliant deployments.

Tier Model in Google Cloud Platform (GCP)

- **Scalability:** As the organization grows, the policies to ensure compliance are applied by default through the inheritance hierarchy to new projects and resources. This ensures scalability of the desired security model.
- **Flexibility:** Policies can be customized to meet the specific industry and regional security and compliance requirements.

Organizations are encouraged to regularly review and update the Organization Policies to adapt to evolving security threats and compliance needs.

Monitoring and Detection

While tiering limits lateral movement within the environment, determined attackers may still attempt to escalate privileges or move between tiers. Continuous monitoring and detection can help identify such attempts and prevent them from succeeding. When implementing tiering of resources within a cloud environment, we need to ensure proper levels of logging is enabled on all resources. The logs should be collected from the cloud environments and forwarded to a central Security Information and Events Management (SIEM) solution. In the event of a compromise, logs are the primary source of information for security operations teams to investigate the attacker path, and properly remediate and recover the environment. It is recommended that organizations create detection rules within their SIEM solution to proactively detect any anomalies and trigger an incident if needed to ensure expedited recovery from an attack.

The following types of logs should be enabled at minimum in the various tiers:

- Identity logs that include any sign-in and audit events for all types of accounts (users, service accounts, machine identities)
- Security logs
- System logs
- Any productivity application logs
- Firewall logs
- Proxy, VPN and any perimeter solution logs
- Endpoint device logs for users of all tiers

The following briefly captures the typical anomalies to alert for while building monitoring and detection scenarios as part of the tiering exercise:

- **Anomalous patterns:** SIEM systems can be tuned to generate alerts on detecting any anomalies from identified patterns. Certain indicators can alert a potential security breach to the Security Operations Center (SOC) team. Anomalies can include consecutive failed login attempts, consecutive logins from different time-zones within a short time frame, disabled accounts getting re-enabled, login events at unusual times, running of malicious binaries, scripts and so on.
- **Credential reset events:** High value user accounts and administrator accounts should be enabled with alerting to trigger notification in the event of password resets, multi-factor authentication method updates and so on.

- **Detection of potential data exfiltration:** Alerting should exist for critical resources to flag any events where large amounts of data is copied to unexpected drive locations or external IP addresses, bulk permission updates on sensitive files, and bulk deletion events.
- **Configuration Drift Detection:** Configurations set in the environments to ensure baseline security and compliance will modify over time depending on how the organization scales, to keep up to date with updated guidelines. This can potentially introduce security vulnerabilities or non-compliance. Monitoring will help identify configuration drift and ensure resources stay aligned with the security policies defined for their tier. Specific unexpected drifts can be alerted to the SOC teams such as updating of federated domains on an Identity provider to add new unknown domains, deletion of existing domains, and disabling encryption and reducing encryption levels in resources. These are common behaviors Mandiant sees attackers perform in compromised environments.

In an environment with multiple cloud platforms in use, along with on-premises infrastructure, organizations are encouraged to explore solutions that support Multi-Cloud Security and Posture Management (CSPM). CSPM solutions can help provide visibility, continuous monitoring, automated remediation to address any misconfigurations, vulnerabilities and compliance issues across different cloud platform resources.

Key features and functionalities of a CSPM solution that can help achieve an elevated level of security for multi-cloud resources include:

- **Visibility and Asset Inventory:** CSPM tools provide a comprehensive view of the cloud assets, including compute instances, storage services, databases, and network configurations, across multiple cloud providers.
- **Configuration Assessment:** It helps to continuously assess the configurations of the cloud resources against security best practices, compliance standards (e.g., PCI DSS, HIPAA, GDPR), and internal security policies.
- **Misconfiguration Detection:** SOC teams can set up alerting for misconfigurations within critical Tier 0 or Tier 1 resources, that could lead to a bigger incident and quickly apply mitigations as needed.
- **Vulnerability Management:** CSPM solutions can scan for vulnerabilities and prioritize them based on severity and risk.
- **Remediation and Automation:** Organizations can set up actionable recommendations for the relevant teams and automated workflows as applicable, to fix misconfigurations and mitigate any potential vulnerabilities.
- **Incident Response:** CSPM solutions can support incident response efforts by providing visibility into cloud security events and facilitating forensic investigations.
- **DevSecOps Integration:** Most CSPM solutions integrate with DevOps processes to ensure security is built into the development and deployment lifecycle.

Protecting From the Attacks by Applying Tiering Model Practices

Here we will take the examples from the Attack Scenarios section and apply tiering to protect from the threat actor.

Scenario #1: Protect Against Total Domain Compromise

Let's explore how the Moonbeam Energy organization has applied tiering to its infrastructure to protect against the [attack scenario #1](#) described in the earlier section.

Resource Segregation applied to On-premises and Cloud

- The Active Directory organizational unit has Tier 0, 1 and 2 segregation for any servers and accounts in the domain. Group Policy Objects (GPO) are applied to apply hardening controls, firewall policies, user rights assignment controls for the respective tier objects in AD. Domain Controllers in the environment are placed within Tier 0 Organizational Unit. Similarly any on-premises and cloud hosted AD Virtual Machines, servers, workstations and identities having direct permissions on Tier 0 servers are placed within the Tier 0 organization unit (OU). AD Tier 1 OU is reserved for any domain joined application servers, workstations and identities.
- Microsoft Azure cloud is also segregated based on tiers, to have Tier 0, and Tier 1 Management Groups. Any resources associated with Identity and Access Management (IAM) such as DC VMs, Privileged Access Management (PAM) solutions are placed within Tier 0 Subscription under Tier 0 Management Group. All application resources such as App Servers, containers, application gateways are placed within Tier 1 Management Group and Subscriptions. Tier 1 Management Group has separate subscriptions for hosting Production resources and any Development and Test workloads.

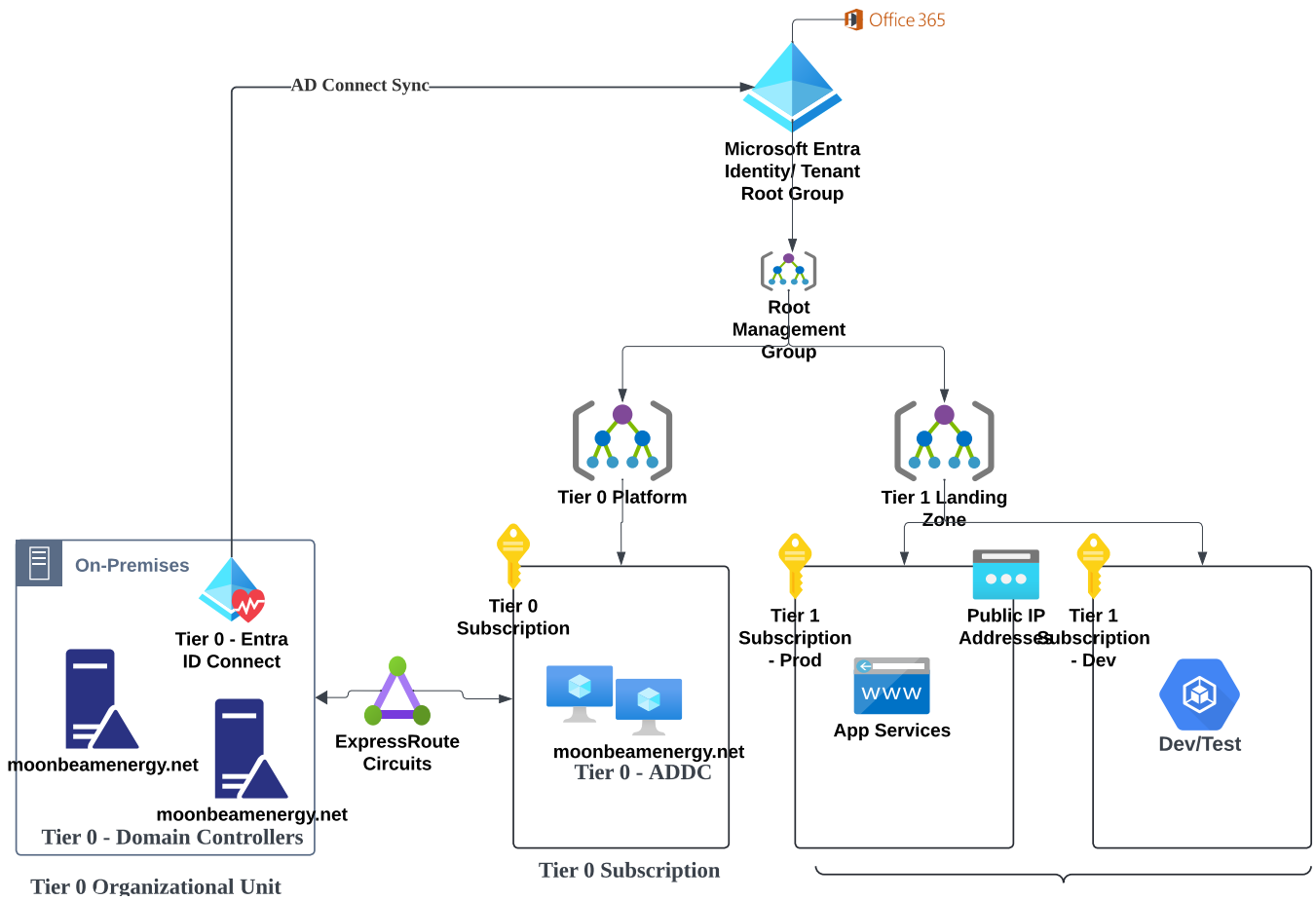


Figure 10. Tiering applied to segregate resources into Tiers 0 and 1 - On-Premises and Azure

- For the Google Cloud Platform (GCP) environment, Moon Beam Energy organization has followed the tiering approach as well to segregate resources and place them into the correct tiers. The GCP organization has folders underneath its resource structure to host Tier 0 and Tier 1 Projects. All the production application related GCP resources are placed within the Tier 1 Landing Zone folder, within the Tier 1 Project. SAML integration for SSO and identity provisioning has been set up between Microsoft Entra ID and GCP Identity, to ensure user accounts are created from Microsoft Entra to GCP.

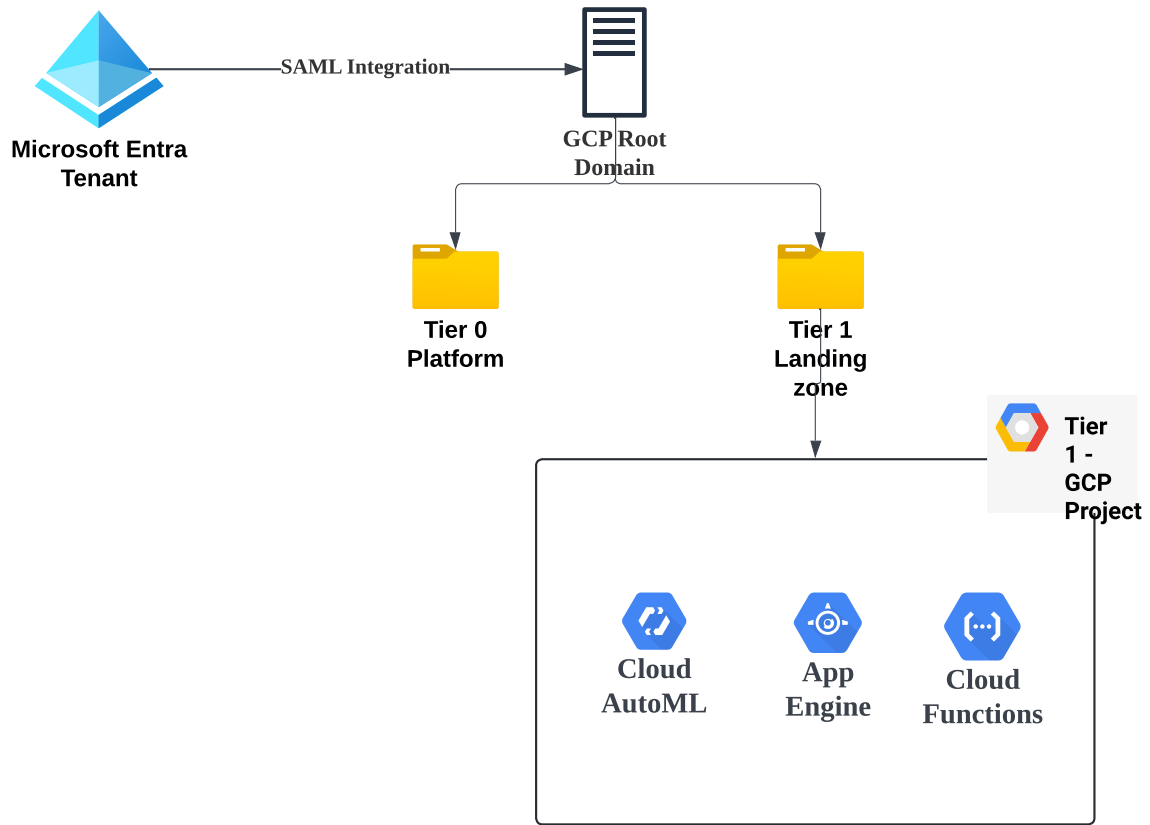


Figure 11: Tiering based resource segregation applied to GCP Cloud

Credential Tiering and Limiting Inheritance

Moon Beam Energy team implemented the following controls to adopt credential tiering.

| Control | Implementation |
|---------------------------------|---|
| <p>Tier 0 Identities</p> | <p>The following accounts in the clouds are Tier 0 and are provisioned in the respective cloud platforms and not a federated identity provider. The Tier 0 user identities have no access provisioned to emails, Drive or SaaS productivity applications.</p> <p>Microsoft Azure</p> <p>Microsoft Entra Cloud only accounts provisioned to users holding Global Administrator, Privileged Role Administrator, Intune Administrator, Privileged Authentication Administrator and other privileged roles.</p> <p>Microsoft Entra Cloud only accounts provisioned to users holding Subscription access to Tier 0 resources.</p> <p>GCP</p> <p>Organization administrator of GCP organization, Workspace and Cloud Identity super admin users hold identity created in GCP Cloud Identity or Workspace. Similarly, any Owner or Editor roles applying to Tier 0 GCP resources are GCP Cloud Identities.</p> <p>Active Directory</p> <p>All built-in administrator accounts in Active Directory are Active Directory based accounts and are not synchronized to Microsoft Entra ID. All Tier 0 users managing Tier 0 servers are excluded from Microsoft Entra ID Connect sync scope as well.</p> |
| <p>Tier 1 Identities</p> | <p>The following accounts in the clouds are Tier 1 and are provisioned separately from day to day productivity user accounts. The Tier 1 user identities have no access provisioned to emails, OneDrive or SaaS productivity applications.</p> <p>Microsoft Azure</p> <p>Users having permissions to manage Tier 1 application resources such as App Services, Container instances, and database services.</p> <p>GCP</p> <p>Users having permissions on Tier 1 application resources hosted in Compute Engines, Google Kubernetes Engine, and Cloud AutoML services.</p> <p>Active Directory</p> <p>Users having permission on Tier 1 application servers such as SharePoint, Exchange, and application database servers.</p> |
| <p>Tier 2 Identities</p> | <p>Active Directory users and groups are synchronized using Password-Hash sync configuration with Microsoft Entra ID, for Single-sign-on (SSO) and provisioning user accounts to Microsoft 365 and Azure. Microsoft Entra ID is also integrated using SAML with GCP Cloud Identity, to provide SSO capabilities in GCP.</p> <p>All productivity user accounts are created in AD and synchronized to the Microsoft Entra ID platform as needed to support various application single-sign-on scenarios, such as log in to emails, OneDrive, Dropbox, Google Meet, and HRMS application.</p> |

Protecting From the Attacks by Applying Tiering Model Practices

| | |
|--|--|
| <p>Limit Permission inheritance</p> | <p>Most cloud platforms inherit permissions applied at the top domain root level to all the resources in the hierarchy below. This is risky as this leads to multiple unintended users gaining privileges on critical resources. To mitigate this Moon Beam Energy team implemented the following:</p> <p>GCP</p> <p>In GCP cloud, users having access to higher level Folders, have access by default to the resources below, due to inheritance. But the Cyber Coffee Co. team was able to use Deny Policies to restrict inherited privileges, at the various projects containing critical Tier 0 and Tier 1 resources.</p> <p>Microsoft Azure</p> <p>In Azure however, there are no ways to deny inherited permissions. So, to mitigate the risks of inherited privileges, the team provided only emergency access accounts permissions at the root management group level.</p> |
| <p>Create Emergency Access Accounts</p> | <p>Emergency Access accounts were created in each cloud platform holding the following super administrator privileges:</p> <ul style="list-style-type: none"> • Microsoft Entra Global Administrator • Microsoft Azure Root Management Group Owner • GCP Organization administrator • AD Administrator accounts for emergency access |
| <p>Multi-Factor authentication (MFA)</p> | <p>Cyber Coffee Co. team implemented the following:</p> <p>Tier 0</p> <p>All Tier 0 users are enforced to use MFA challenge at every authentication attempt using a phishing resistant method of FIDO 2.0 security keys.</p> <p>Tier 1</p> <p>All Tier 1 user accounts are challenged with MFA for every authentication using a phishing resistant method of FIDO 2.0 security keys.</p> <p>Tier 2</p> <p>Tier 2 users are enabled with MFA methods of Software OATH tokens, Phone.</p> |
| <p>Restricted administrator tool access</p> | <p>Administrator portals, command line tools and APIs for Azure and GCP are restricted to be accessed only by authorized Tier 0 and Tier 1 identities from trusted endpoints (Cloud PAW).</p> |

Use of Privileged Access Workstations

The Tier 0 users are provided with separate hardened administrative workstation devices, also referred as Privileged Access Workstations (PAWs) to manage Tier 0 cloud resources in Azure and GCP. Tier 0 administrator users use separate workstations to perform productivity tasks. Similarly, Tier 1 users managing cloud resources for Tier 1 are provided with Tier 1 cloud PAWs.

Note that, on-premises and cloud administration is performed using separate PAW devices; as on-premises PAW devices will be domain joined workstations meant to manage resources within the AD environment. While cloud

Protecting From the Attacks by Applying Tiering Model Practices

management will be performed using Cloud PAW, which is meant to be enrolled with a cloud based mobile device management (MDM) provider. This form of device management is done to ensure that we reduce the possible paths through which a threat actor can move laterally between environments and elevate privileges.

Network Segmentation

Moon Beam Energy organization has applied network segmentation to segregate hosting of Tier 0, 1 and 2 resources. There are dedicated firewalls between each tier network segment that restrict the traffic to other networks by protocol, port, domains, IP addresses.

Secure Web Gateway solution is implemented to ensure all traffic from endpoints are captured and tunneled irrespective of the location i.e. both in the organization network as well as external network. All communication between endpoints and services are encrypted with the latest version of TLS protocol. There are rules in place to ensure any sanctioned countries, web applications and malicious IP addresses are blocked from accessing using Moon Beam Energy managed endpoints.

Security Configurations applied at scale

Moon Beam Energy organization has used the following tools to ensure the baseline [security configurations mentioned in this section](#) are applied to all cloud resources:

- Azure Policies applied to all resources in Azure
- CSPM solution implemented to flag and remediate any inconsistencies in cloud resources from baseline security configurations.
- GCP Organization policies applied to all resources in GCP

Monitoring and Detection

Logs are sent to a central SIEM workspace from the following sources:

- On-premises servers
- Active Directory event logs
- Microsoft Entra logs
- SaaS application logs
- Azure activity logs
- GCP logs
- Network logs from firewall, proxy solutions, IDPS solution, CASB solution

The Moon Beam Energy team has used Google Security Operations SIEM tool to create detections for the following use cases:

- Suspicious administrator actions
- MITRE attack tactics, techniques and procedures⁸
- Anomalous cloud resource modifications for Tier 0 and critical Tier 1 production applications

Protecting From the Attacks by Applying Tiering Model Practices

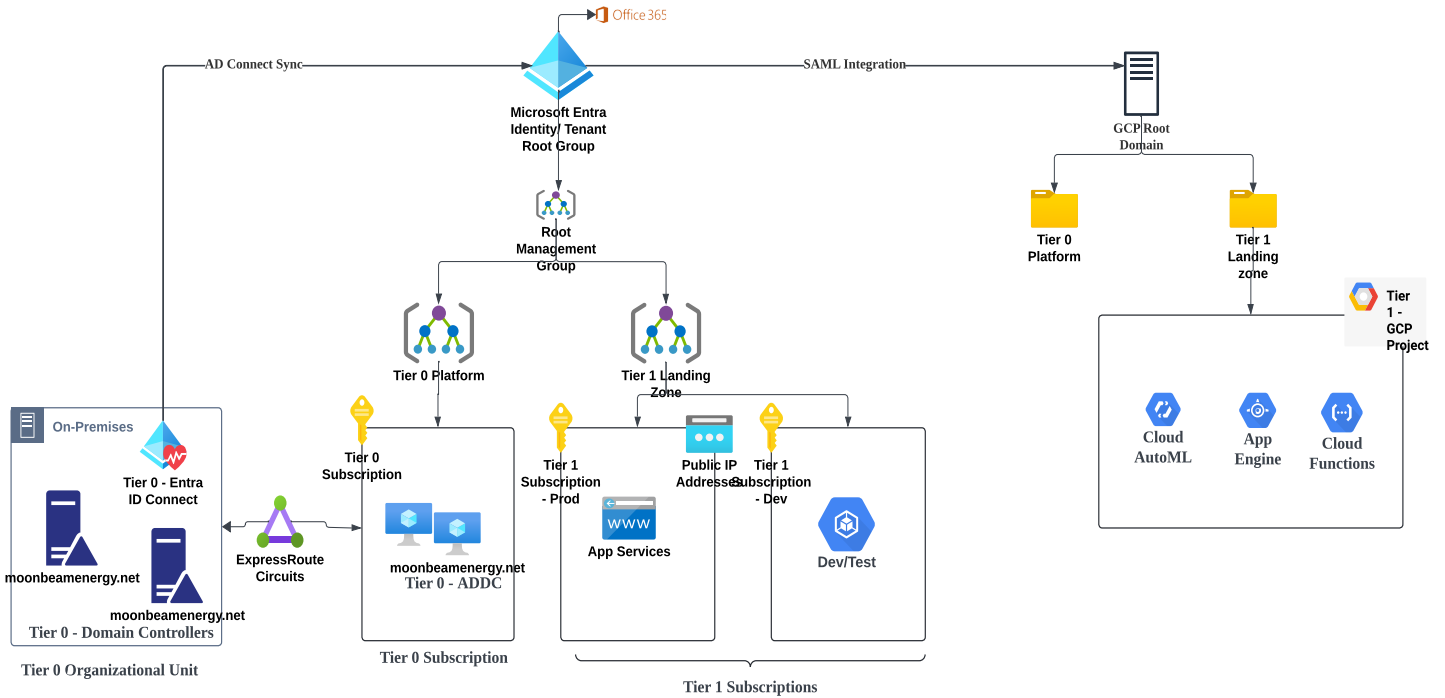


Figure 12. Tiering controls applied organization-wide to on-premises servers, Azure and GCP cloud platforms

Exploring the attack path and how the tiering controls protect

Alex now has two sets of credentials based on Alex being a Tier 0 user and performing productivity functions as well:

- Cloud only Tier 0 Identity
- AD synchronized Productivity account

Protecting From the Attacks by Applying Tiering Model Practices

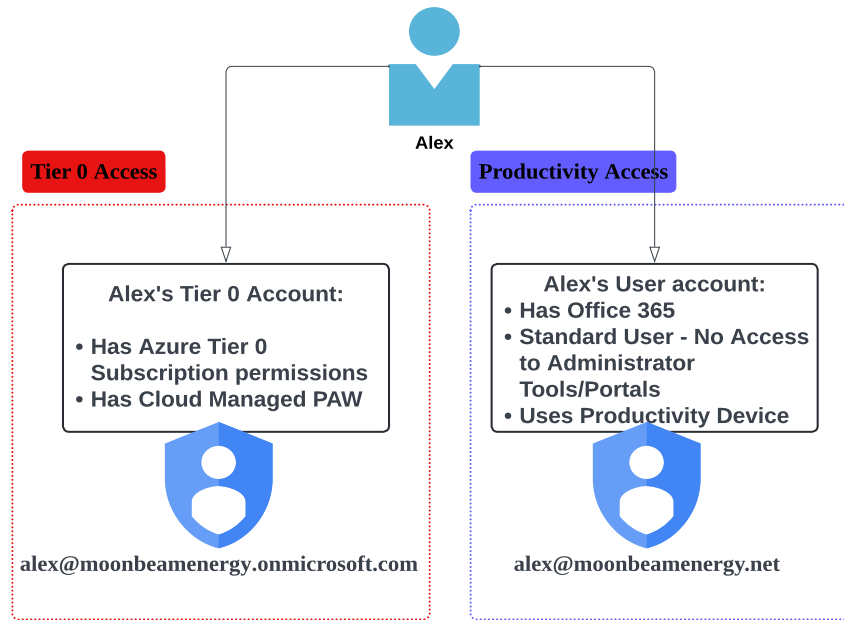


Figure 13: Alex having two credentials: Tier 0 cloud only credential and AD synced productivity credential

| Attack steps | Diagram |
|--|--|
| <p>Alex logs in to their mailbox using the productivity identity alex@moonbeamenergy.net.</p> <p>Alex clicks on a phishing email link. The link sends Alex's productivity account credentials and session token to a threat actor.</p> | <p>The diagram shows a person icon labeled 'Alex' on the left. A red arrow points from Alex to an envelope icon labeled 'Phishing Email' with the address 'alex@moonbeamenergy.net'. Above the envelope is a hooded figure icon representing a 'Threat Actor'. A red arrow points from the threat actor down to the phishing email, with the text 'Threat Actor steals Alex's token' next to it.</p> |

Protecting From the Attacks by Applying Tiering Model Practices

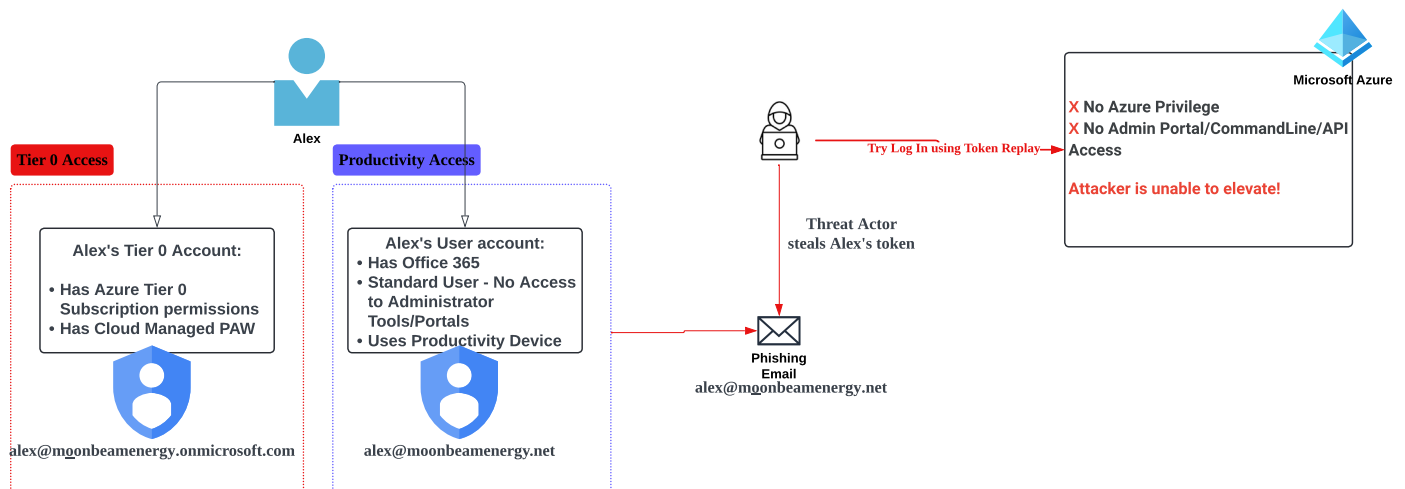
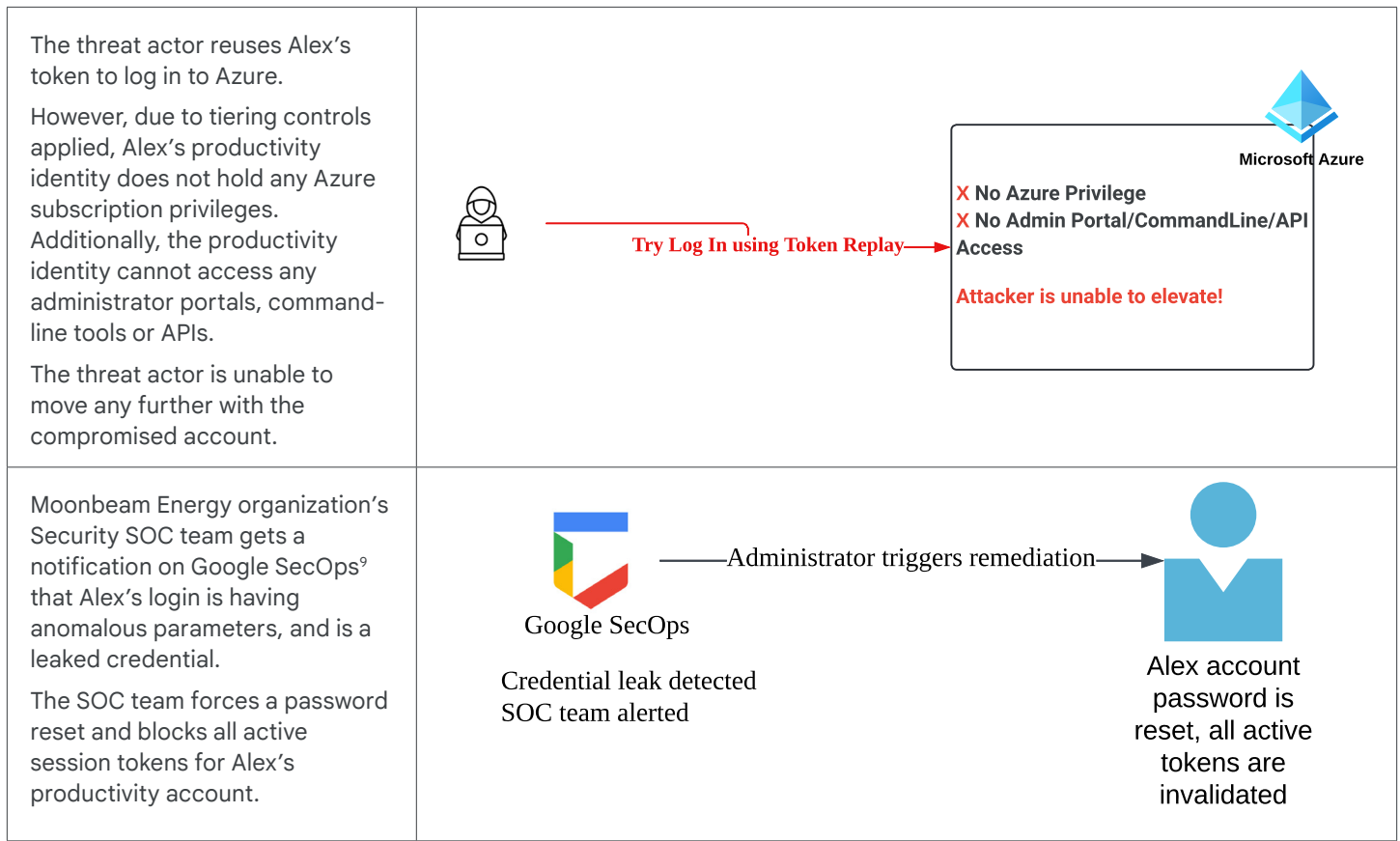


Figure 14: Complete attack path and how tiering prevented the attack from proceeding

Scenario #2: Protect Against Multi-Cloud Compromise

Let's explore how the Cyber Coffee Co. organization has applied tiering to its infrastructure to protect against the [attack scenario #2](#) described in the earlier section.

Resource Segregation applied to On-premises and Cloud

- Cyber Coffee Co. organization has applied resource tiering in its Microsoft Azure cloud to segregate resources in their respective management groups i.e., Tier 0, and Tier 1 Management Groups. Any resources associated with hosting business applications are Tier 1 category, and placed within Tier 1 Azure subscription under Tier 1 Landing Zone Management Group.
- For the Google Cloud Platform (GCP) environment, Cyber Coffee Co. organization has followed the tiering approach as well to segregate resources and place them into the correct tiers. The GCP organization has folders underneath its resource structure to host Tier 0 and Tier 1 Projects. All the application related GCP resources are placed within the Tier 1 Landing Zone folder, within the Tier 1 Project.
- AWS cloud root organization is also segregated to its child organizational units for Tier 0 and Tier 1. The lambda, and virtualization services are hosted within a separate Tier 1 Account.

Protecting From the Attacks by Applying Tiering Model Practices

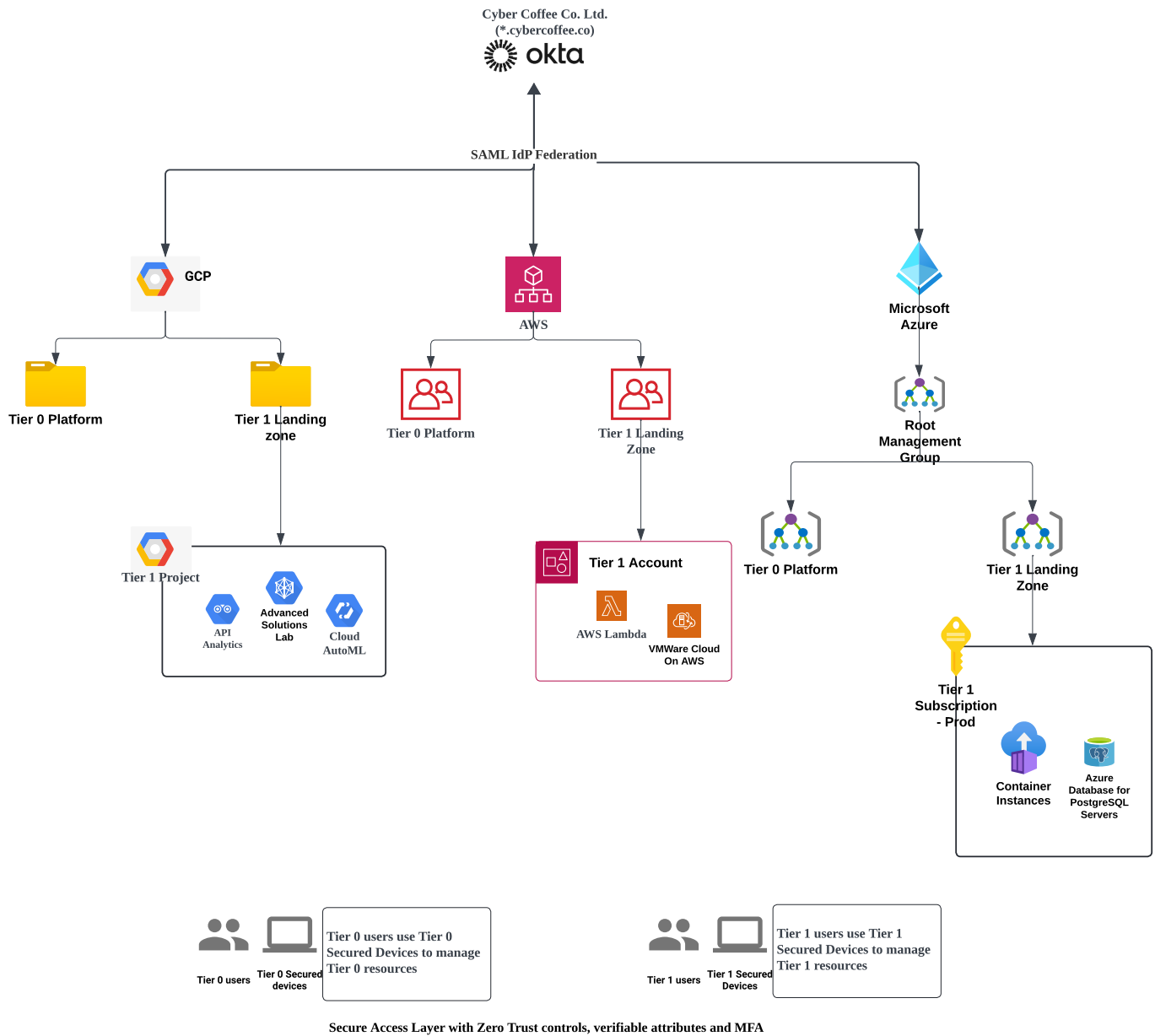


Figure 15: Cyber Coffee Co. multi-cloud tiering of cloud resources

Credential Tiering and Limiting Inheritance

Cyber Coffee Co. implemented the following controls to adopt credential tiering.

| Control | Implementation |
|---------------------------------|--|
| <p>Tier 0 Identities</p> | <p>The following accounts in the clouds are Tier 0 and are provisioned in the respective cloud platforms and not a federated identity provider. The Tier 0 user identities have no access provisioned to emails, Drive or SaaS productivity applications.</p> <p>Microsoft Azure</p> <p>Microsoft Entra Cloud only accounts provisioned to users holding Global Administrator, Privileged Role Administrator, Intune Administrator, Privileged Authentication Administrator and other privileged roles.</p> <p>Microsoft Entra Cloud only accounts provisioned to users holding Subscription access to Tier 0 resources.</p> <p>GCP</p> <p>Organization administrator of GCP organization, Workspace and Cloud Identity super admin users hold identity created in GCP Cloud Identity or Workspace. Similarly, any Owner or Editor roles applying to Tier 0 GCP resources are GCP Cloud Identities.</p> <p>AWS</p> <p>The AWS root organization user, AWS IAM administrator user identities are Tier 0 and are separate identities from productivity identities. Any user having permissions/ assumable roles on Tier 0 AWS resources are also Tier 0 identities.</p> |
| <p>Tier 1 Identities</p> | <p>The following accounts in the clouds are Tier 1 and are provisioned separately from day to day productivity user accounts. The Tier 1 user identities have no access provisioned to emails, Drive or SaaS productivity applications.</p> <p>Microsoft Azure</p> <p>Users having permissions to manage Tier 1 application resources such as App Services, Container instances, and database services.</p> <p>GCP</p> <p>Users having permissions on Tier 1 application resources hosted in Compute Engines, Google Kubernetes Engine, and Cloud AutoML services.</p> <p>AW</p> <p>Users having permission or roles on Tier 1 application resources hosted in AWS such as Lambda, S3 buckets, and VMWare on AWS.</p> |
| <p>Tier 2 Identities</p> | <p>Okta identity provider is federated through SAML for Single-sign-on and provisioning user accounts to all the three cloud platforms: Microsoft Azure, AWS, GCP.</p> <p>All productivity user accounts are created in Okta and synchronized to the three cloud platforms as needed to support various application single-sign-on scenarios, such as log in to emails, Drive, Dropbox, Google Meet, and HRMS application.</p> |

| | |
|--|--|
| <p>Limit Permission inheritance</p> | <p>Most cloud platforms inherit permissions applied at the top domain root level to all the resources in the hierarchy below. This is risky as this leads to multiple unintended users gaining privileges on critical resources. To mitigate this Cyber Coffee Co. implemented the following:</p> <p>AWS</p> <p>Unlike most platforms AWS does not automatically inherit permissions on AWS Accounts from organization units. Users can assume roles in AWS to authenticate and get authorization to access AWS services that they have permissions to manage. Due to this, Cyber Coffee Co. team was able to use this mapping of roles and policies for the separate tiers, to allow users to only access resources they need to manage, and use of permission boundaries to restrict any unintended privileges.</p> <p>GCP</p> <p>In GCP cloud, users having access to higher level Folders, have access by default to the resources below, due to inheritance. But the Cyber Coffee Co. team was able to use Deny Policies to restrict inherited privileges, at the various projects containing critical Tier 0 and Tier 1 resources.</p> <p>Microsoft Azure</p> <p>In Azure however, there are no ways to deny inherited permissions. So, to mitigate the risks of inherited privileges, the team provided only emergency access accounts permissions at the root management group level.</p> |
| <p>Create Emergency Access Accounts</p> | <p>Emergency Access accounts were created in each cloud platform holding the following super administrator privileges:</p> <ul style="list-style-type: none"> • Microsoft Entra Global Administrator • Microsoft Azure Root Management Group Owner • GCP Organization administrator • AWS Root user • Okta break glass accounts |
| <p>Multi-Factor authentication (MFA)</p> | <p>Cyber Coffee Co. team implemented the following:</p> <p>Tier 0</p> <p>All Tier 0 users are enforced to use MFA challenge at every authentication attempt using a phishing resistant method of FIDO 2.0 security keys.</p> <p>Tier 1</p> <p>All Tier 1 user accounts are challenged with MFA for every authentication using a phishing resistant method of FIDO 2.0 security keys.</p> <p>Tier 2</p> <p>Tier 2 users are enabled with MFA methods of Software OATH tokens, Phone.</p> |
| <p>Restricted administrator tool access</p> | <p>Administrator portals, command line tools and APIs for Azure, GCP and AWS are restricted to be accessed only by authorized Tier 0 and Tier 1 identities from trusted endpoints (Cloud PAW).</p> |

Use of Privileged Access Workstations

- The Tier 0 users are provided with separate hardened administrative workstation devices, also referred as Privileged Access Workstations (PAWs) to manage Tier 0 cloud resources in AWS, Azure and GCP. Tier 0 administrator users use separate workstations to perform productivity tasks.
- Similarly, Tier 1 users managing cloud resources for Tier 1 are provided with Tier 1 cloud PAWs.
- To manage any cloud environment, the Cloud PAW is used with no local administrator privileges on the device.
- Policies are in place to restrict all cloud Tier 0 and Tier 1 privileged accounts to only manage the environment through a PAW device.
- The PAW devices issued to administrators are enrolled using Microsoft Intune. Intune compliance and configuration policies are applied on the enrolled devices to ensure endpoint BitLocker, Credential Guard, antivirus, firewall, real-time protection and other device hardening capabilities are enforced.

Network Segmentation

Cyber Coffee Co. organization has applied network segmentation to segregate hosting of Tier 0, 1 and 2 resources. There are dedicated firewalls between each tier network segment that restrict the traffic to other networks by protocol, port, domains, IP addresses.

Secure Web Gateway solution is implemented to ensure all traffic from endpoints are captured and tunneled irrespective of the location i.e. both in the organization network as well as external network. All communication between endpoints and services are encrypted with the latest version of TLS protocol. There are rules in place to ensure any sanctioned countries, web applications and malicious IP addresses are blocked from accessing using Cyber Coffee Co. managed endpoints.

Security Configurations applied at scale

Moon Beam Energy organization has used the following tools to ensure the baseline [security configurations mentioned in this section](#) are applied to all cloud resources:

- Azure Policies applied to all resources in Azure
- AWS Config Rules applied to resources in AWS
- CSPM solution implemented to flag and remediate any inconsistencies in cloud resources from baseline security configurations.
- GCP Organization policies applied to all resources in GCP

Monitoring and Detection

Logs are sent to a central SIEM workspace from the following sources:

- AWS logs
- Microsoft Entra logs
- Okta Sign In logs
- SaaS application logs
- Azure activity logs

Protecting From the Attacks by Applying Tiering Model Practices

- GCP logs
- Network logs from firewall, proxy solutions, IDPS solution, CASB solution

The Moon Beam Energy team has used Google Security Operations SIEM tool to create detections for the following use cases:

- Suspicious administrator actions
- MITRE attack tactics, techniques and procedures
- Anomalous cloud resource modifications for Tier 0 and critical Tier 1 production applications

Exploring the attack path and how the tiering controls protect

Alan now has two sets of credentials based on Alan being a Tier 1 user and performing productivity functions as well:

- Cloud only Tier 1 Identity
- Okta based Productivity account

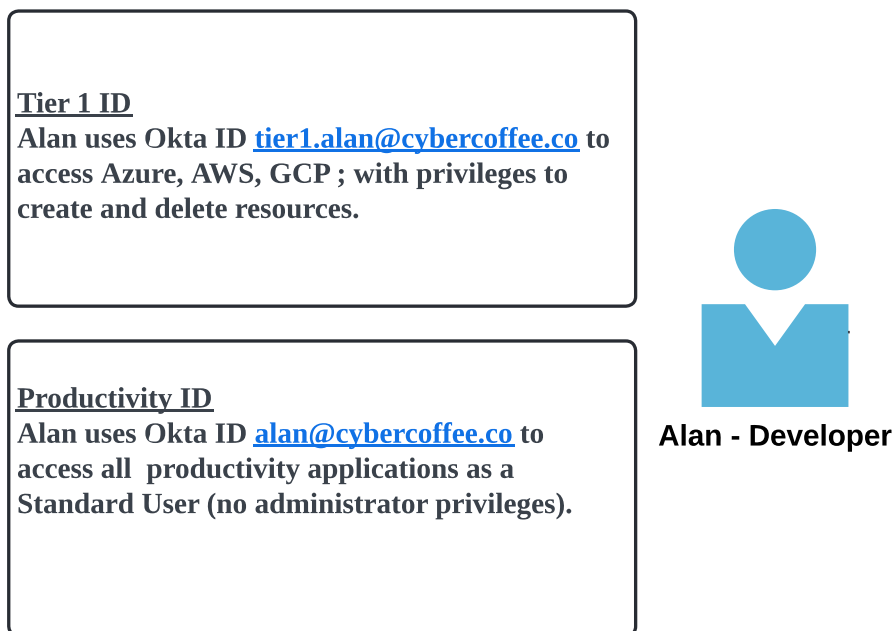
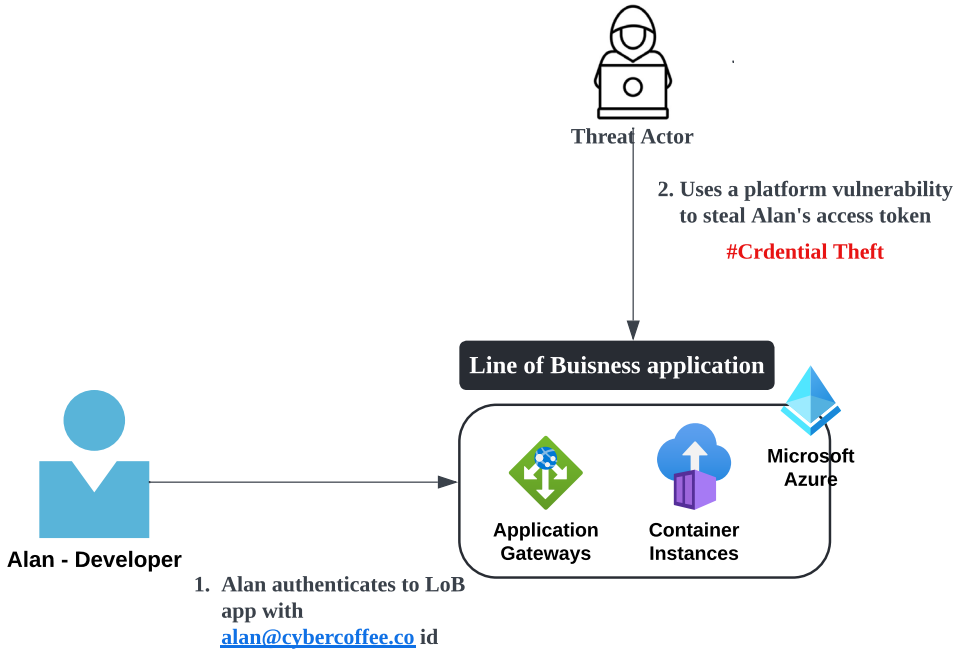
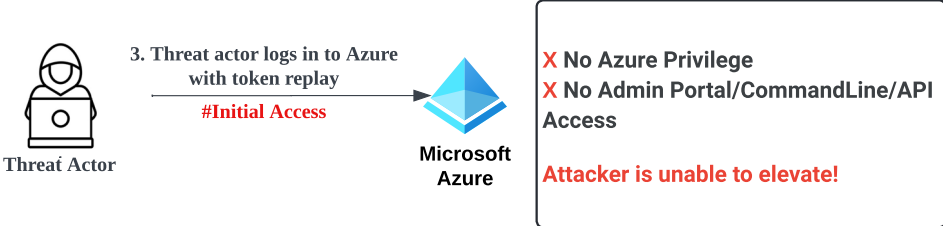
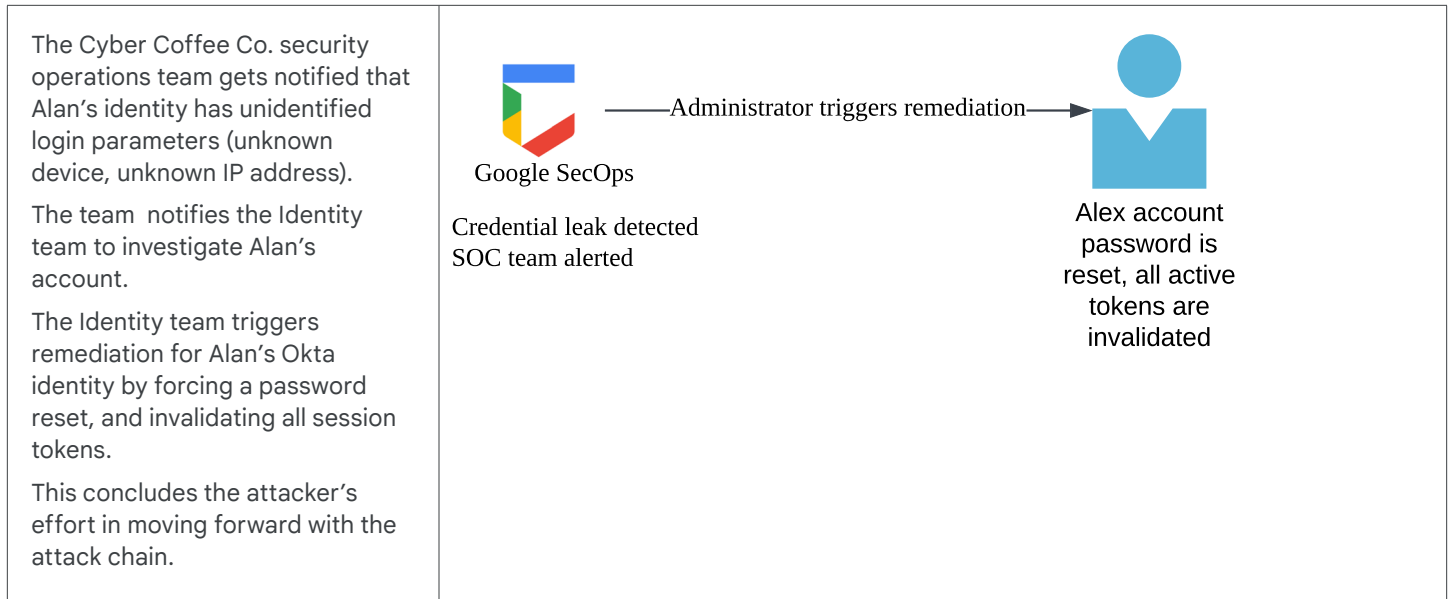


Figure 16: Alan having two credentials: Tier 1 and productivity

Protecting From the Attacks by Applying Tiering Model Practices

| Attack steps | Diagram |
|--|---|
| <p>Alan uses their productivity Okta identity alan@cybercoffee.co to authenticate and access an Internet facing line-of-business application as a user.</p> <p>The application is hosted on Azure Container instances.</p> <p>The threat actor uses a known platform vulnerability to gain access to the application.</p> <p>The threat actor steals Alan's access token and session tokens.</p> |  <p>1. Alan authenticates to LoB app with alan@cybercoffee.co id</p> <p>2. Uses a platform vulnerability to steal Alan's access token #Crdential Theft</p> |
| <p>The threat actor tries to replay the token to access Microsoft Azure Portal.</p> <p>But the threat actor is blocked from moving forward in the attack due to the following:</p> <ol style="list-style-type: none"> 1. Alan's Okta productivity identity has no Azure subscription privilege. Alan has a separate Tier 1 Entra ID credential that has Azure subscription permissions. 2. Accessing any administrator portals, command line or Graph API tools is blocked for all users except Tier 0 and Tier 1 identities. Hence, the attacker cannot access these tools either to move within the Azure environment. |  <p>3. Threat actor logs in to Azure with token replay #Initial Access</p> <p>X No Azure Privilege X No Admin Portal/CommandLine/API Access Attacker is unable to elevate!</p> |

Protecting From the Attacks by Applying Tiering Model Practices



Conclusion

This white paper for standardizing privileged access architecture for multi-cloud environments attempts to establish how the implementation of a cloud agnostic tiered architecture effectively protects against various cloud based attacks. It further establishes how the tiering architecture can be standardized and consistently applied to any cloud platforms. This paper further delves into the security controls for tiering by examining the configurations applied to the three most widely used cloud platforms: Amazon Web Services, Microsoft Azure and Google Cloud Platform.

The controls detailed within this paper attempts to define the baseline configuration for tiering in multi-cloud, using resource segregation, credential isolation, privilege restriction, use of privileged access workstations, network segmentation, security configurations, and robust monitoring and detection, to provide a formidable defense against the multifaceted cloud attack scenarios that frontline security teams witness everyday.

By aligning the security controls with the specific requirements of each tier, organizations can establish a proactive security posture that thwarts unauthorized access, lateral movement and data exfiltration attempts. This paper incorporates a zero-trust security approach, coupled with continuous monitoring and detection, to ensure that potential threats are identified and neutralized before they can cause significant damage.

This white paper has also explored the tiering approach in detail across AWS, Azure, and GCP, demonstrating its effectiveness in mitigating risks and bolstering cloud security. It is evident that embracing this architecture, organizations can confidently leverage the benefits of multi-cloud environments while maintaining the highest levels of security and compliance.

After establishing the tiering model best practices and configurations, this paper has revisited the initial attack scenarios described in earlier sections, and illustrated how the tiering model successfully prevents the exploitation of vulnerabilities and safeguards critical assets. This reinforces the importance of adopting tiering practices as a holistic and proactive security strategy that adapts to the evolving threat landscape.

As cloud technologies continue to advance and attackers become more sophisticated, it is imperative for organizations to remain vigilant and continuously refine their security measures. The cloud agnostic tiering model architecture presented in this paper is aimed at offering a robust framework for achieving this goal, and ensuring the long-term resilience of cloud infrastructure.

References

- 1: NIST Cybersecurity framework, 02/26/2024
<https://www.nist.gov/cyberframework/cybersecurity-framework-components>
- 2: CIS Benchmark for network segmentation, 04/03/2023
https://essentialguide.docs.cisecurity.org/en/latest/bp/network_segmentation.html
- 3: PCI DSS Quick Reference Guide v3.2.1
https://listings.pcisecuritystandards.org/documents/PCI_DSS-QRG-v3_2_1.pdf
- 4: The Uptime Institute Tier classification
<https://uptimeinstitute.com/tiers>
- 5: ISACA COBIT - Using a Risk Based Approach to Prioritize Vulnerability Remediation, 02/07/2023
<https://www.isaca.org/resources/news-and-trends/industry-news/2023/using-a-risk-based-approach-to-prioritize-vulnerability-remediation>
- 6: Microsoft Enterprise Access Model, 01/30/2024
<https://learn.microsoft.com/en-us/security/privileged-access-workstations/privileged-access-access-model>
- 7: Microsoft article of known limitations of deployment stacks for deny assignments 06/24/2024
<https://learn.microsoft.com/en-us/azure/azure-resource-manager/bicep/deployment-stacks?tabs=azure-powershell#known-limitations>
- 8: Mitre Attack Tactics, Techniques and Procedures
<https://attack.mitre.org/>
- 9: Google Security Operations
<https://cloud.google.com/security/products/security-operations>