**Databricks Landing and Registration Issues with Mitigations**

**Databricks Landing & Registration Issues Comparison Table with Mitigations**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Landing & Registration Issue | One-Liner Description | Where It Typically Arises | Landing & Registration Areas Most Affected | Mitigation Strategies |
| 1 | Inconsistent Workspace Provisioning | Workspaces are set up manually with different configurations, creating operational drift. | Initial workspace onboarding | Workspaces, Clusters, Repos | Use Terraform or CLI automation for consistent provisioning across environments. |
| 2 | Unclear Onboarding Processes | New teams lack clear guidance on how to get started, delaying productivity. | Team onboarding, project setup | Workspaces, Notebooks | Document and publish onboarding guides, workspace policies, and naming conventions. |
| 3 | Insufficient Network Configuration | Private link, VNet, or firewall settings are incomplete, blocking access. | Networking setup | Clusters, Jobs, Storage | Validate network connectivity during provisioning; test all endpoints before go-live. |
| 4 | Misaligned Resource Naming | Naming conventions are inconsistent across workspaces, complicating management and discovery. | Workspace setup, cluster creation | Clusters, Jobs, Storage | Define standard naming conventions; enforce with automation templates and validation scripts. |
| 5 | Missing Tags and Metadata | Resources are created without tags, reducing cost tracking and accountability. | Workspace and resource creation | Clusters, Jobs, SQL Warehouses | Require tagging policies for environment, owner, and purpose; validate in provisioning pipelines. |
| 6 | Insufficient Identity Federation | Federation to Azure AD or SAML is incomplete, resulting in manual user management. | Initial workspace configuration | Identity & Access Management | Integrate SCIM for automatic user and group provisioning; validate permissions regularly. |
| 7 | Ineffective Cluster Policies | No cluster policies are defined, allowing users to create inefficient or non-compliant clusters. | Cluster creation | Compute Resources | Implement cluster policies to enforce security and cost standards; require compliance for new clusters. |
| 8 | Incomplete Documentation | Landing zone setup lacks clear documentation for future updates or audits. | Initial workspace deployment | Workspaces, Networking, IAM | Document all configurations, decisions, and procedures in a shared knowledge base. |
| 9 | Lack of Validation Steps | Workspaces are not tested end-to-end before onboarding new teams. | Provisioning workflows | Clusters, Storage, Repos | Create validation checklists and pipelines to test clusters, storage access, and notebooks. |
| 10 | No Central Registry of Workspaces | Teams lack visibility into all existing workspaces, creating confusion and duplication. | Multi-team, multi-region deployments | Governance, Management | Maintain a central registry or inventory of all workspaces with owner and purpose metadata. |

**Quick Reference**

* **Landing Zone:** The foundational Databricks workspace configuration including networking, identity, and policies.
* **Provisioning:** Creating and configuring new workspaces and resources.
* **SCIM:** Standard protocol for automated user and group management.
* **Cluster Policies:** Templates restricting cluster configuration options.
* **Resource Registry:** A catalog of workspaces for visibility and governance.

**Example Mitigation Actions and Configurations**

**Automate Workspace Creation with Terraform:**

hcl

CopyEdit

resource "databricks\_mws\_workspaces" "prod\_workspace" {

account\_id = var.account\_id

aws\_region = "us-west-2"

workspace\_name = "prod-data-platform"

deployment\_name = "prod-data-platform"

}

**Define Naming Conventions:**

* Example pattern:

php-template

CopyEdit

<environment>-<project>-<resource>

e.g., prod-sales-cluster

**Tag Resources on Creation:**

json

CopyEdit

"custom\_tags": {

"Environment": "Production",

"Owner": "data-team",

"Project": "MarketingAnalytics"

}

**Validate Network Configuration:**

* Confirm connectivity:

bash

CopyEdit

curl -I https://<workspace-url>/

* Test private link endpoints and firewall rules.

**Enable SCIM Integration:**

* Azure example:
  + In *Workspace Admin Settings*, enable SCIM.
  + Sync groups and users automatically.

**Enforce Cluster Policies:**

* Example policy JSON:

json

CopyEdit

{

"policy\_family\_definition\_overrides": {

"spark\_version": {

"type": "fixed",

"value": "13.3.x-scala2.12"

},

"node\_type\_id": {

"type": "fixed",

"value": "Standard\_DS3\_v2"

}

}

}

**Maintain a Workspace Registry:**

* Store in a central Delta table:

sql

CopyEdit

CREATE TABLE governance.workspace\_registry (

workspace\_name STRING,

owner STRING,

region STRING,

environment STRING,

created\_at TIMESTAMP

);