DATA+AI SUMMIT
BY databricks

Lakehouse Security

Security Best Practices and Tools to build a secure Lakehouse

Anindita Mahapatra
Lead Solutions Architect, Databricks

Arun Pamulapati
Sr. Staff Security Field Engineer, Databricks

Databricks
2023

Product safe harbor statement

This information is provided to outline Databricks' general product direction and is **for informational purposes only.** Customers who purchase Databricks services should make their purchase decisions relying solely upon services, features, and functions that are currently available. Unreleased features or functionality described in forward-looking statements are subject to change at Databricks discretion and may not be delivered as planned or at all.

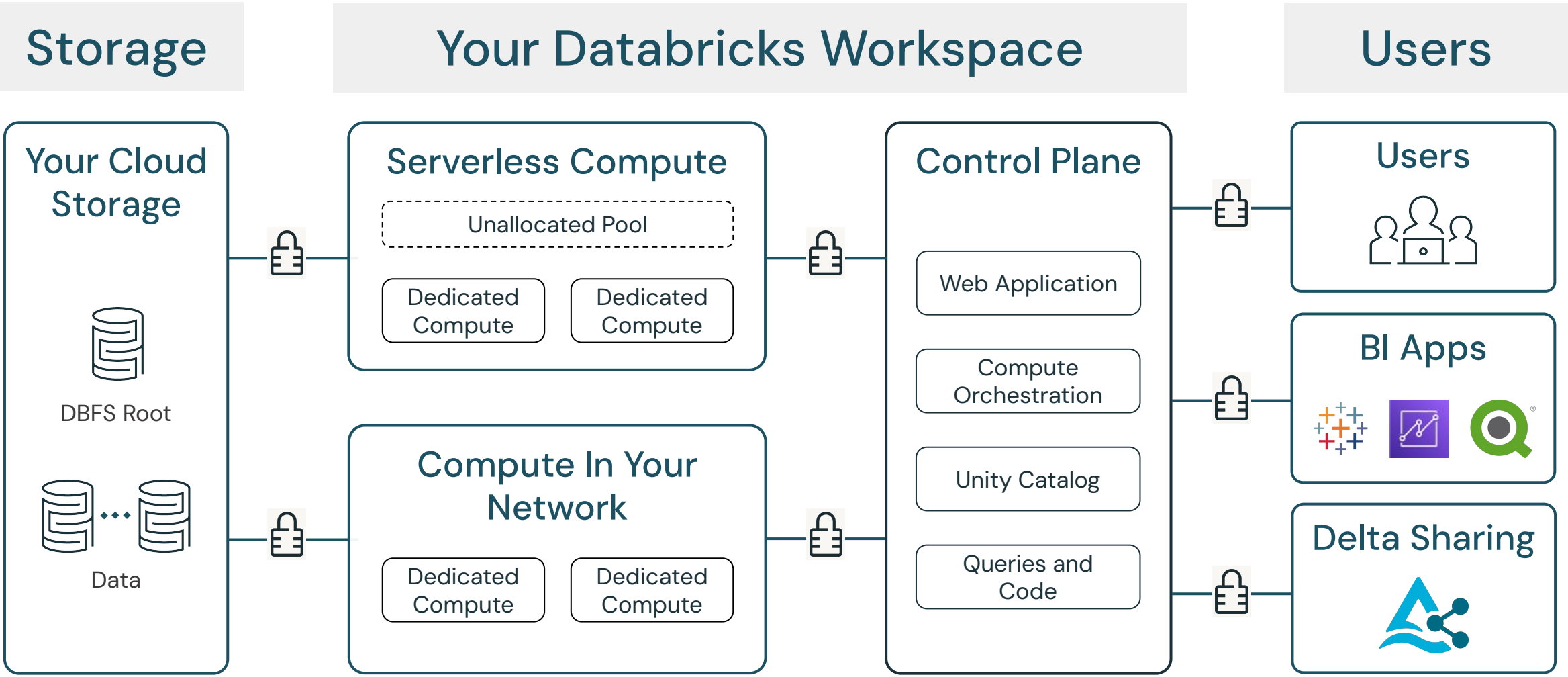
"Databricks Security Analysis Tool (SAT) enables us to automatically detect deviations from our security protocols across all our Databricks workspaces from a single pane of glass. The automated, centralized approach to security has helped harden our Databricks deployments and resulted in productivity improvement for our operations team."
~ Grizel Lopez, Sr. Director of Engineering, Barracuda

"We find SAT very helpful, especially for new Databricks users. It not only enables us to automate checking our workspaces, but most importantly it guides us to understand what are the best practices and security health that we need to apply on our deployment."
- Haitao Helen Du, Lead Advisor, Laurentian Bank

What you will learn today

- Databricks Lakehouse Architecture
- Top Threats impacting deployments
  - Mitigation controls
- **Define:** Security Best Practices
- **Deploy:** How to automate deployment of a secure workspace
  - We are announcing a new tool today!!! 🎉🎉🎉
- **Monitor:** Security Analysis Tool to automate analysis of an existing deployment

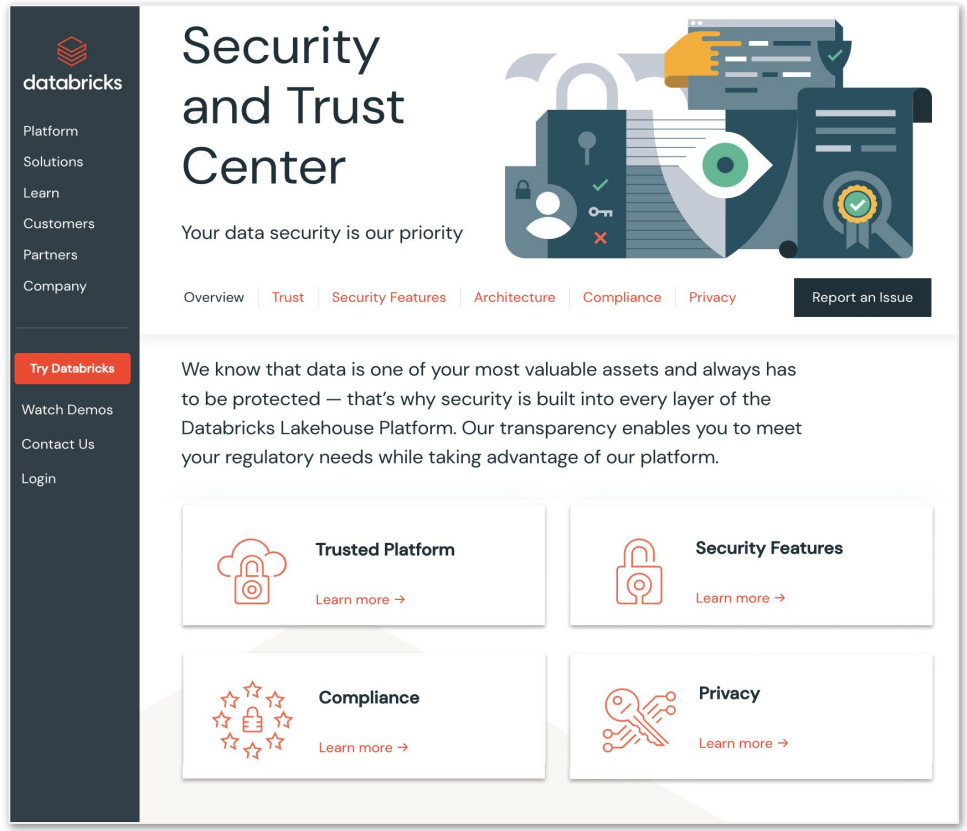
Databricks Lakehouse Architecture



Databricks Security Summary

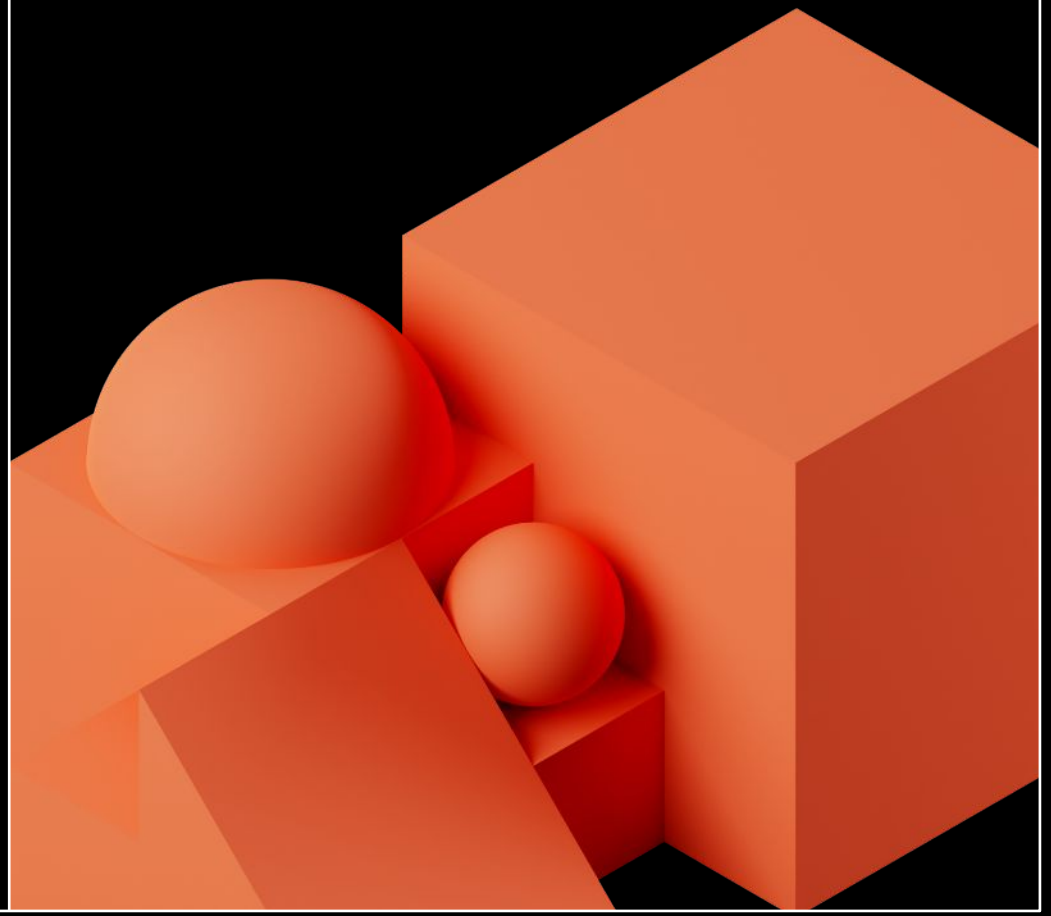
Databricks Security & Trust Center provides an overview of:

1. Databricks security operations (Vulnerability handling, bug bounty, penetration testing, incident response)
2. Databricks in-product security features
3. Assurance and compliance capabilities
4. Details on our secure software development life cycle
5. Details on how we approach privacy (DPA, Security Addendum, Contracts)



Threats impacting deployments

- Mitigation controls



Threats and Risks

Best Practices

Account Takeover

Attackers gain the credentials/access of your users

- Required two-factor auth on your identity provider
  - Consider FIDO Key
- Use SCIM to deprovision
- IP Access Lists or PrivateLink
- Monitor Audit Log
- Limit token lifetime
- Use Service Principles
- Manage local passwords

Insider Exposure

Resource abuse or copy by an accidental insider

- Use access control lists
- Limit exposure using Unity Catalog and Isolation clusters
- Limit data in DBFS and monitor for large datasets
- Deploy data exfiltration protections
- Backup & automate your data and code
- Monitoring audit logs

Data Exfiltration

Data stolen by an attacker or malicious insider

- Use Customer-managed VPC / VNet Injection (baseline)
- Route traffic through a firewall or proxy to limit destination
- [AWS] VPC Endpoint Policies
- Limit access to sensitive data
- Configure data exfiltration settings in the console
- Monitoring audit Logs

Resource Abuse

Vendor Compromise

Account Takeover

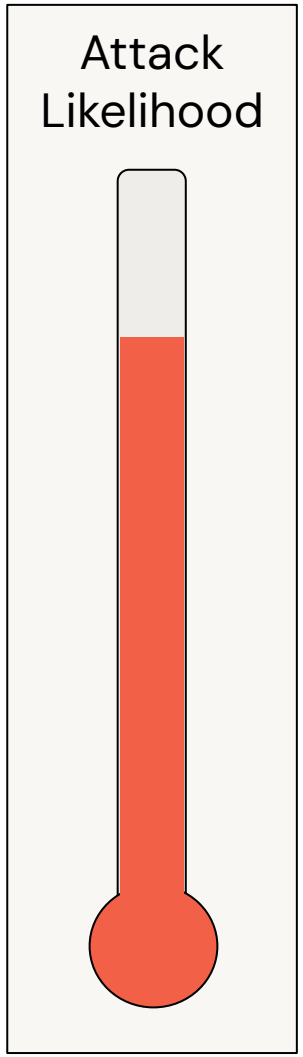
`Attackers gain the credentials/access of your users`

Risk Overview

- Customers often analyze sensitive datasets
- Compromised end-user credentials grant access
  - Phishing, brute force, etc.

Best Practices to Mitigate

- Required two-factor auth on your identity provider
  - Consider FIDO Key
- Use SCIM to deprovision
- IP Access Lists or PrivateLink
- Monitor Audit Log
- Limit token lifetime
- Use Service Principles
- Manage local passwords
-



Accidental Insider Exposure

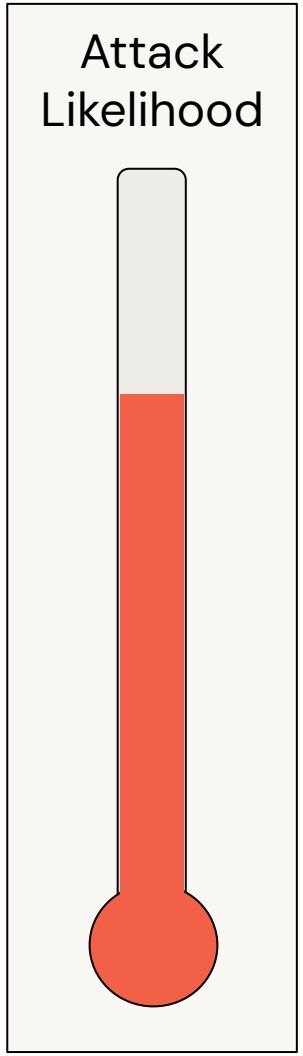
`Resource abuse or accidental copy by insiders`

Risk Overview

- Someone believes their job is easier without pesky security controls
- Data is copied where it shouldn't be, or where ACLs aren't applied

Best Practices to Mitigate

- Use access control lists
- Use Unity Catalog and Isolation clusters to limit exposure
- Limit data in DBFS and monitor for large datasets
- Deploy data exfiltration protections
- Backup & automate your data and code
- Monitoring audit logs



Data Exfiltration

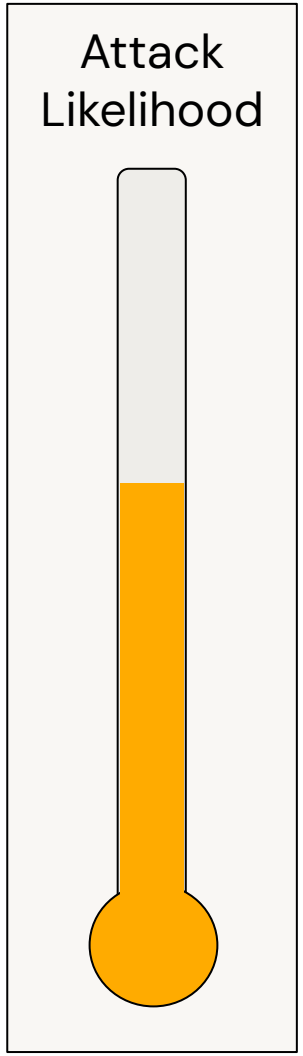
`Data stolen by an attacker or malicious insider`

Risk Overview

- Risk is of a user sending sensitive data out to some external location
- Data plane needs connectivity, by default has full outbound

Best Practices to Mitigate

- Use Customer-managed VPC / VNet Injection (baseline)
- Route traffic through a firewall or proxy to limit destination
- [AWS] VPC Endpoint Policies
- Limit access to sensitive data
- Configure data exfiltration settings in the console
- Monitoring audit Logs



Also covered in the whitepaper

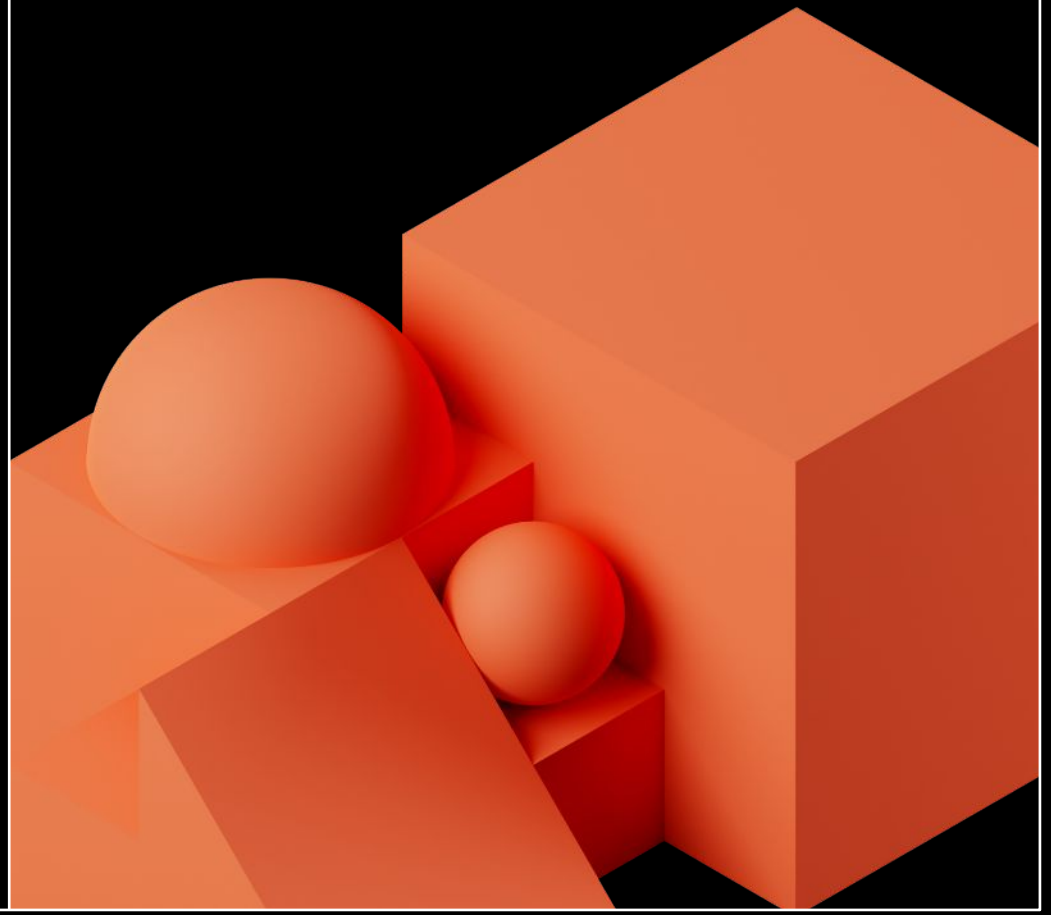
Resource Abuse

- Customer cloud infrastructure hijacking for crypto mining
- Accidental/abusive waste of customer resources

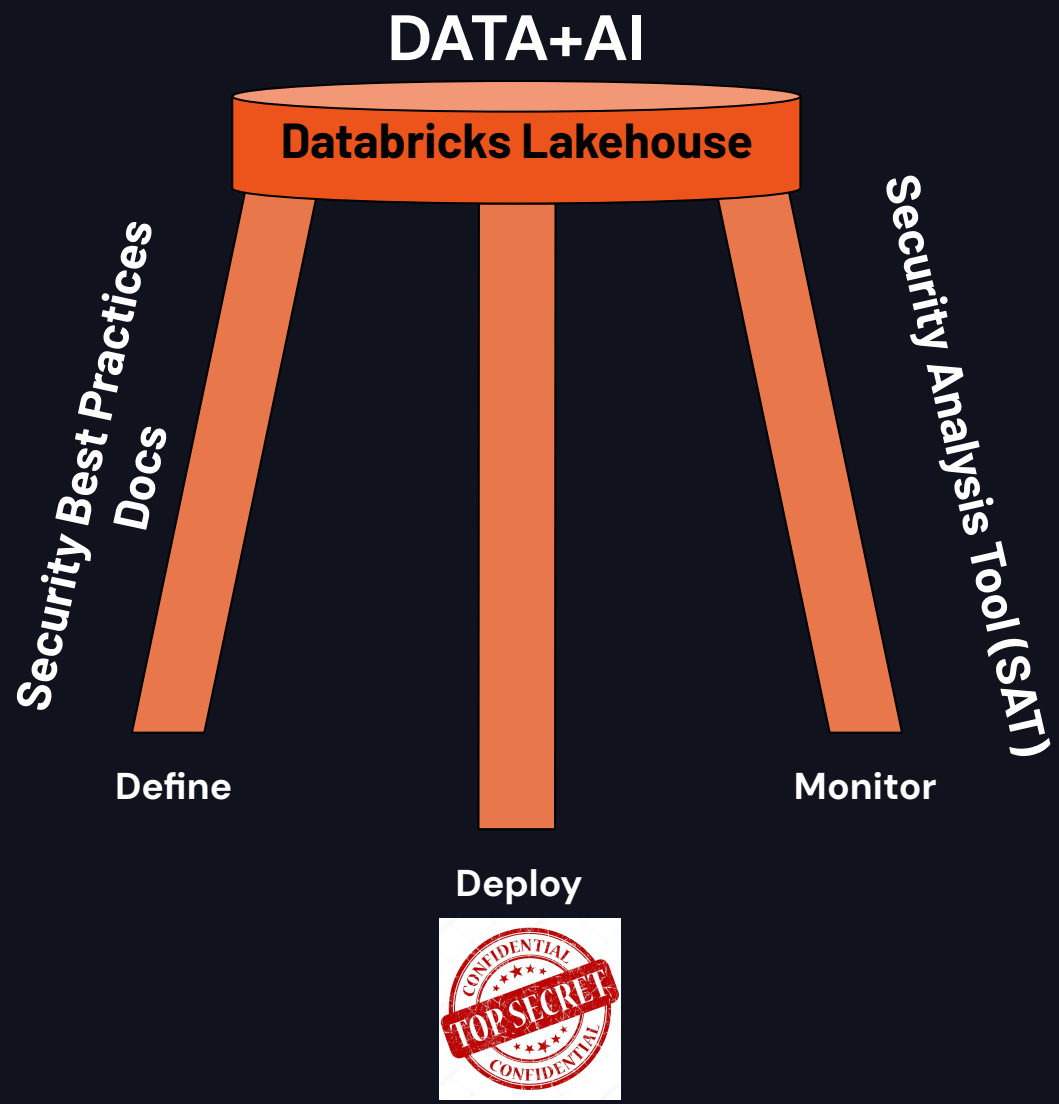
Compromise of SaaS vendor

- Compromise of Databricks Inc user or system could result in compromise of customer environment

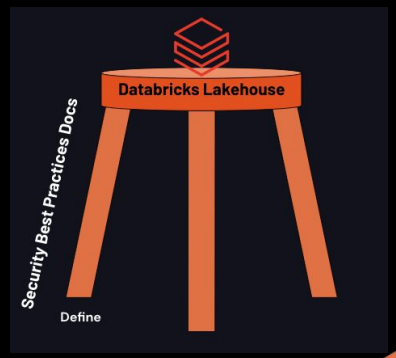
What is a security team to do?



Three legs to your Lakehouse Security



Define: Security Best Practices



There are 35+ best practices in the whitepaper – far more than can be covered in slides

Most deployments

The following typical configurations are part of most enterprise production Databricks deployments. If you are a small data science team of a few people, you may not feel the need to deploy all of these. If Databricks may become a key part of your business or if you are analyzing sensitive data, we recommend that you review these.

- ☐ Evaluate whether multiple workspaces are required for segmentation
- ☐ Check that your S3 buckets are encrypted and that public access is blocked
- ☐ Deploy Databricks into a customer-managed VPC for increased control over the network environment. Even if you do not need this now, this option increases the chances for future success with your initial workspace
- ☐ Authenticate via single sign-on
- ☐ Use multi-factor authentication
- ☐ Separate accounts with admin privileges from day-to-day user accounts
- ☐ Configure Databricks audit log delivery
- ☐ Configure maximum token lifetimes for future tokens using token management
- ☐ Configure admin console settings according to your organization's needs
- ☐ Use Unity Catalog to provide fine grained access control and centralized governance controls
- ☐ Apply bucket policies or other mitigations to avoid storing production datasets in DBFS
- ☐ Backup your notebooks stored in the control plane or store your notebooks in git repos
- ☐ Store and use secrets securely in Databricks or using a third-party service
- ☐ Consider whether to implement network protections for data loss
- ☐ Restart clusters on a regular schedule so that the latest patches are applied.

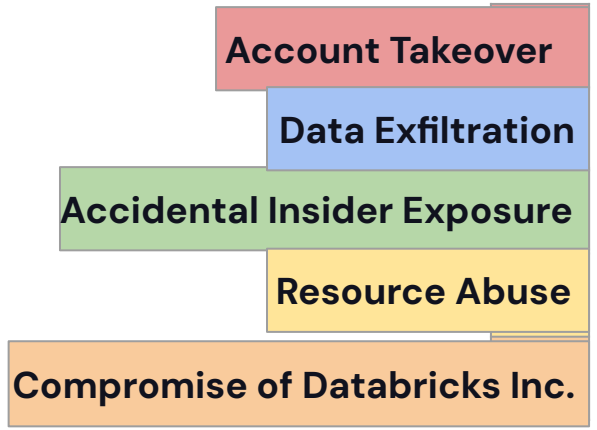
Highly-secure deployments

In addition to the configurations typical to all deployments, the following configurations are often used in highly-secure Databricks deployments. While these are common, not all highly-secure environments use all of these settings. We recommend incorporating these items and the threat model in the following section alongside your existing security practices.

- ☐ Evaluate whether customer-managed encryption keys are needed on the control plane or data plane for control over data at rest (Requires Enterprise tier)
- ☐ Keep an up-to-date user list by using SCIM
- ☐ Set complex local passwords or disable login with local passwords
- ☐ Use either IP access lists or front-end PrivateLink
- ☐ Configure back-end (data plane to control plane) PrivateLink connectivity
- ☐ Implement network protections for data exfiltration
- ☐ Evaluate whether your datasets require bucket versioning
- ☐ Evaluate whether your workflow requires using git repos or CI/CD
- ☐ Plan for and deploy a disaster recovery site if you have strong continuity requirements
- ☐ Consider requiring AWS Nitro instances that provide encryption for ephemeral storage at rest and between instances
- ☐ Encourage the use of clusters that support user isolation
- ☐ Configure cluster policies to enforce data access patterns and control costs
- ☐ Evaluate tagging to monitor and manage chargeback and cost control

We want to have a record of what happens with our data and detect user compromise

Databricks audit log



Audit Logging

- Customers can configure near-real-time logging
  - (AWS/GCP) to a bucket owned by the customer
  - (Azure) to diagnostic logging
- (AWS) Cloudtrail logs also includes provisioning activities

System Logs

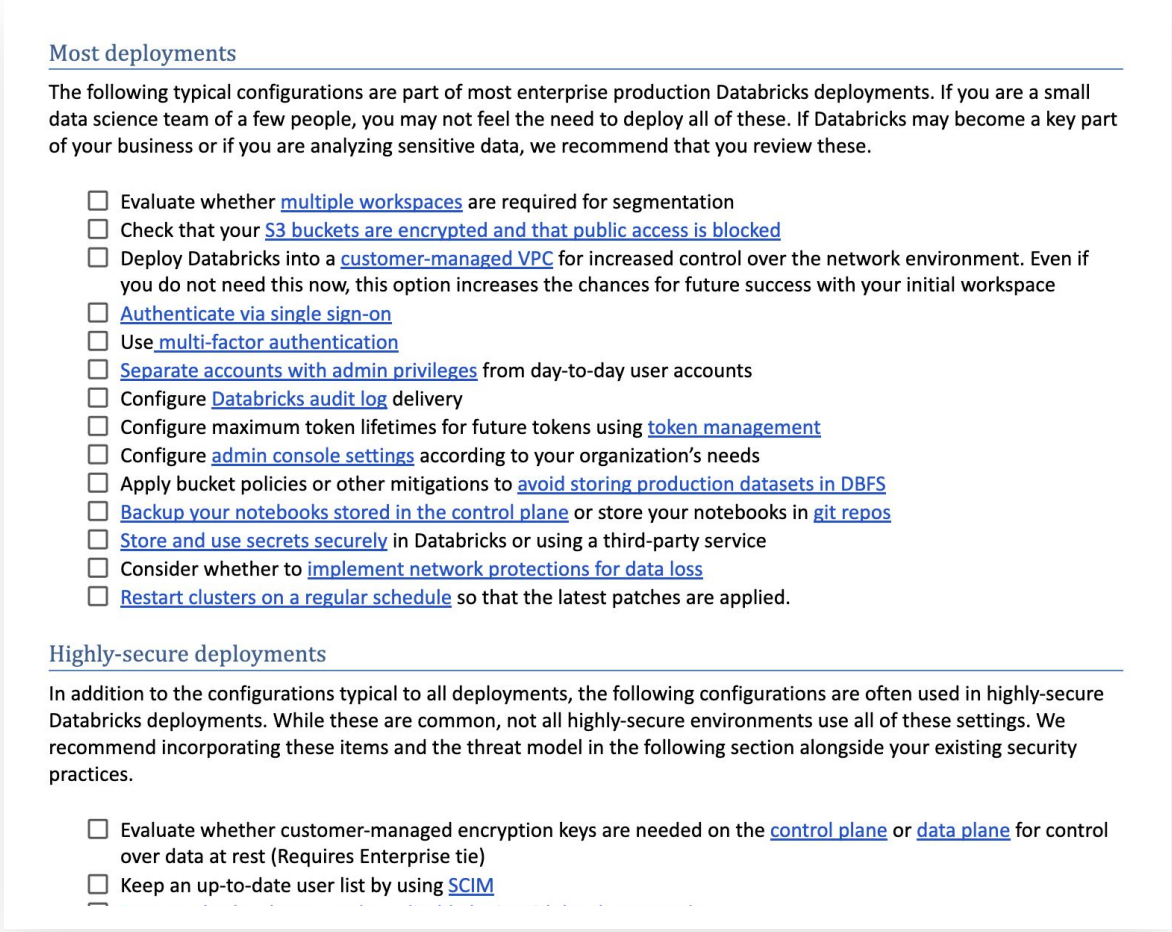
- Understand system activities via system logs, including stdout, stderr, etc.
- Use metrics to understand utilization and health

Ready-to-use analysis notebooks
on our blog!
(Linked from whitepaper)

Learn more—Security and Trust Center

Define and harden your workspaces with our list of best practices

- Provides best practices collected from our most security-conscious deployments
- Both Databricks configurations and related services like AWS, SSO
- Includes general guidance along with configurations to address specific threats
- Available on databricks.com/trust for AWS, Azure and GCP



Most deployments

The following typical configurations are part of most enterprise production Databricks deployments. If you are a small data science team of a few people, you may not feel the need to deploy all of these. If Databricks may become a key part of your business or if you are analyzing sensitive data, we recommend that you review these.

- ☐ Evaluate whether multiple workspaces are required for segmentation
- ☐ Check that your S3 buckets are encrypted and that public access is blocked
- ☐ Deploy Databricks into a customer-managed VPC for increased control over the network environment. Even if you do not need this now, this option increases the chances for future success with your initial workspace
- ☐ Authenticate via single sign-on
- ☐ Use multi-factor authentication
- ☐ Separate accounts with admin privileges from day-to-day user accounts
- ☐ Configure Databricks audit log delivery
- ☐ Configure maximum token lifetimes for future tokens using token management
- ☐ Configure admin console settings according to your organization's needs
- ☐ Apply bucket policies or other mitigations to avoid storing production datasets in DBFS
- ☐ Backup your notebooks stored in the control plane or store your notebooks in git repos
- ☐ Store and use secrets securely in Databricks or using a third-party service
- ☐ Consider whether to implement network protections for data loss
- ☐ Restart clusters on a regular schedule so that the latest patches are applied.

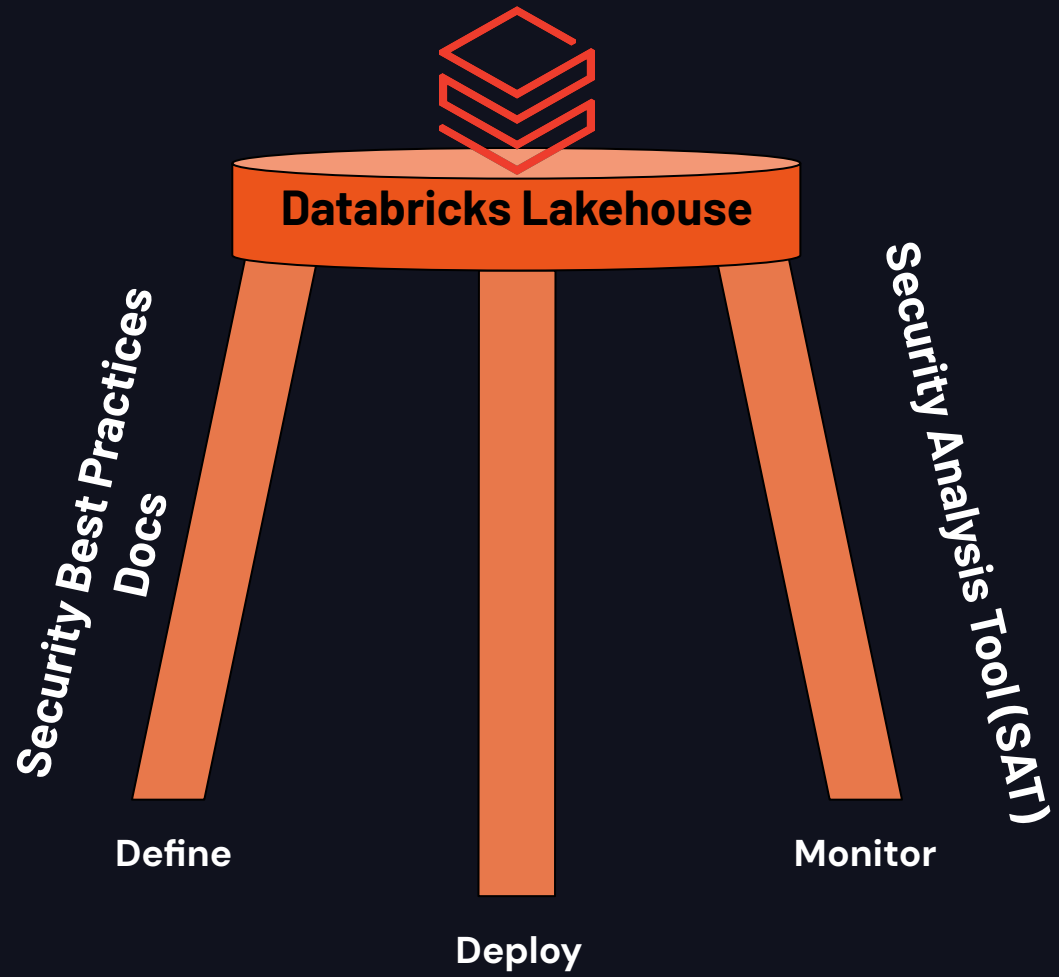
Highly-secure deployments

In addition to the configurations typical to all deployments, the following configurations are often used in highly-secure Databricks deployments. While these are common, not all highly-secure environments use all of these settings. We recommend incorporating these items and the threat model in the following section alongside your existing security practices.

- ☐ Evaluate whether customer-managed encryption keys are needed on the control plane or data plane for control over data at rest (Requires Enterprise tie)
- ☐ Keep an up-to-date user list by using SCIM

https://www.databricks.com/trust/security-features#best-practices

The second leg to your Lakehouse Security



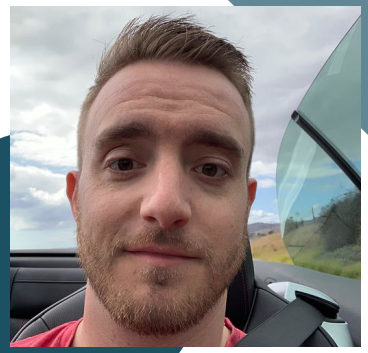
Deploy: Security Reference Architecture (SRAs)

- –automate deploying secure workspaces

SRA Core Team

JD Braun

Specialist Solutions Architect



Nathan Knox

Lead Specialist Solutions Architect

Ricardo Portilla

Principal Solutions Architect

Abhishek Pratap Singh

Specialist Solutions Architect

Special contributors

Alysson Souza

Specialist Solutions Architect

Our customer's ask:

How can we consistently deploy workspaces with Databricks security best practices?

Our answer:

The Security Reference Architecture with Terraform Templates makes deploying workspaces with Security Best Practices easy for customers.

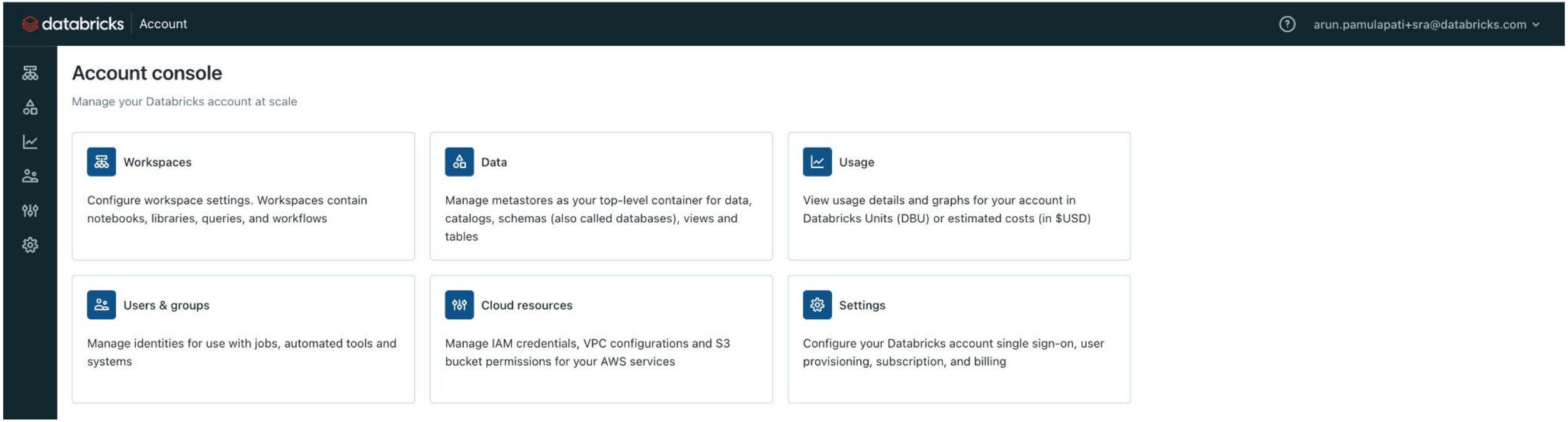
SRA Terraform Template

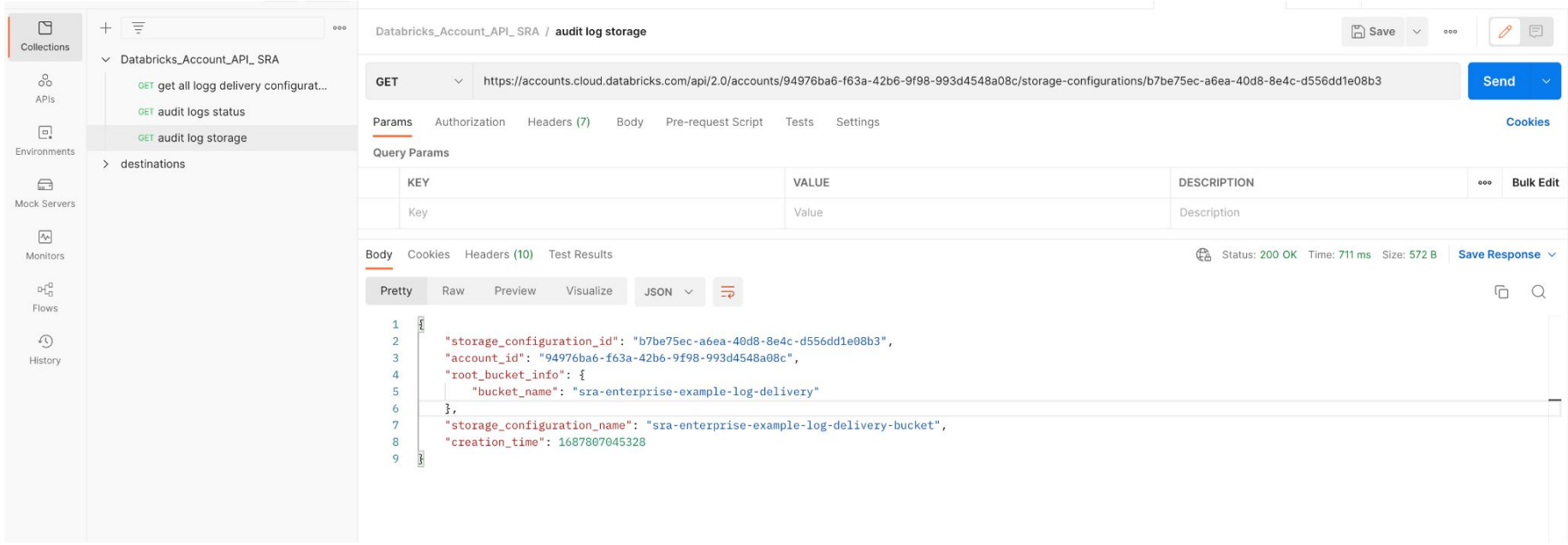
What's included?

- Cloud Infrastructure:
  - The SRA Terraform Templates focus on deploying infrastructure in a secure and scalable way. This includes customer managed network objects, private connectivity, utilizing cloud resource endpoints whenever possible, and integrating Unity Catalog as a priority

- Databricks Resources:
  - Following the deployment of the workspace, SRA Terraform Templates include commonly asked for features like audit and billing logs, creating service principals, setting token maximum lifetimes, and configuring admin configurations

- Additional Security Recommendations and Opportunities:
  - The README contains additional recommendations and opportunities for your customer's to consider

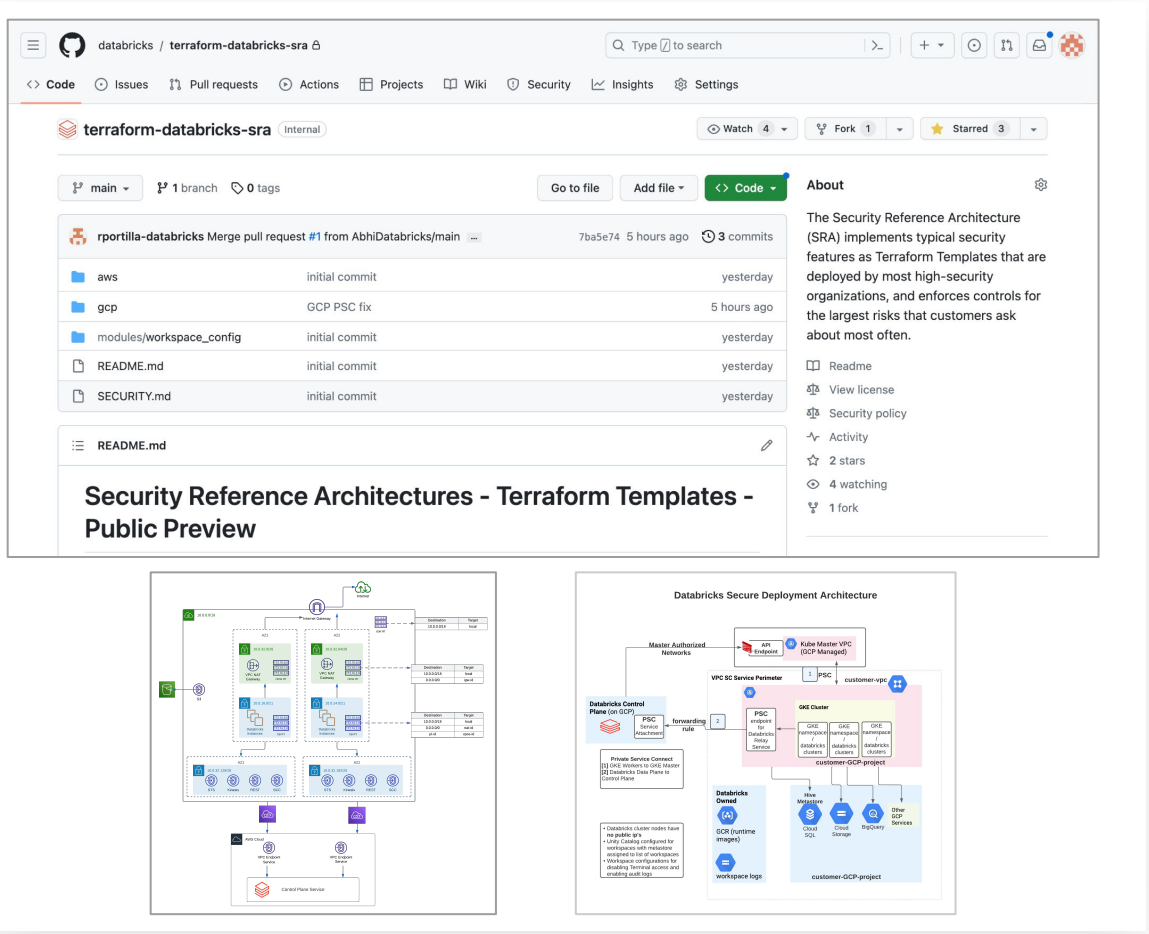

accounts.cloud.databricks.com
databricks | Account
arun.pamulapati+sra@databricks.com
Account console
Manage your Databricks account at scale
Workspaces
Configure workspace settings. Workspaces contain notebooks, libraries, queries, and workflows
Data
Manage metastores as your top-level container for data, catalogs, schemas (also called databases), views and tables
Usage
View usage details and graphs for your account in Databricks Units (DBU) or estimated costs (in $USD)
Users & groups
Manage identities for use with jobs, automated tools and systems
Cloud resources
Manage IAM credentials, VPC configurations and S3 bucket permissions for your AWS services
Settings
Configure your Databricks account single sign-on, user provisioning, subscription, and billing


Collections
APIs
Environments
Mock Servers
Monitors
Flows
History
Databricks_Account_API_ SRA
GET get all logg delivery configurat...
GET audit logs status
GET audit log storage
destinations
Databricks_Account_API_ SRA / audit log storage
Save
GET
https://accounts.cloud.databricks.com/api/2.0/accounts/94976ba6-f63a-42b6-9f98-993d4548a08c/storage-configurations/b7be75ec-a6ea-40d8-8e4c-d556dd1e08b3
Send
Params
Authorization
Headers (7)
Body
Pre-request Script
Tests
Settings
Cookies
Query Params
KEY
VALUE
DESCRIPTION
Bulk Edit
Key
Value
Description
Body
Cookies
Headers (10)
Test Results
Status: 200 OK
Time: 711 ms
Size: 572 B
Save Response
Pretty
Raw
Preview
Visualize
JSON
{
"storage_configuration_id": "b7be75ec-a6ea-40d8-8e4c-d556dd1e08b3",
"account_id": "94976ba6-f63a-42b6-9f98-993d4548a08c",
"root_bucket_info": {
"bucket_name": "sra-enterprise-example-log-delivery"
},
"storage_configuration_name": "sra-enterprise-example-log-delivery-bucket",
"creation_time": 1687807045328
}

Learn more—Security and Trust Center

Run secure-by-default deployments via infrastructure-as-code

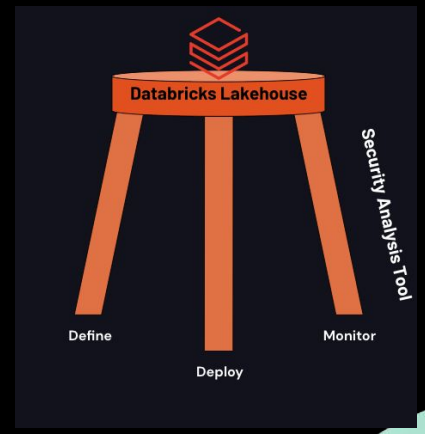
- The most successful Databricks deployments are often managed using Terraform
- Begin with a secure template to default secure configurations
- Initial templates based on Databricks Security Best Practices
- Send us your feedback via git issues and pull requests.



https://www.databricks.com/trust/security-features#best-practices

Monitor: Security Analysis Tool (SAT)

- Automate checking workspaces



SAT Creators

Anindita Mahapatra

Lead Solutions Architect

Ramdas Murali

Lead Solutions Architect

Arun Pamulapati

Sr. Staff Security Field Engineer

Special contributors

Antonio Irizarry

Sr. Specialist SA

Alex Ott

Lead Specialist SA

- *LucasLuigiT*
- *Fseyn*
- *MaartenEvenepoel*
- *roelof-xomnia*

Our customer's ask:

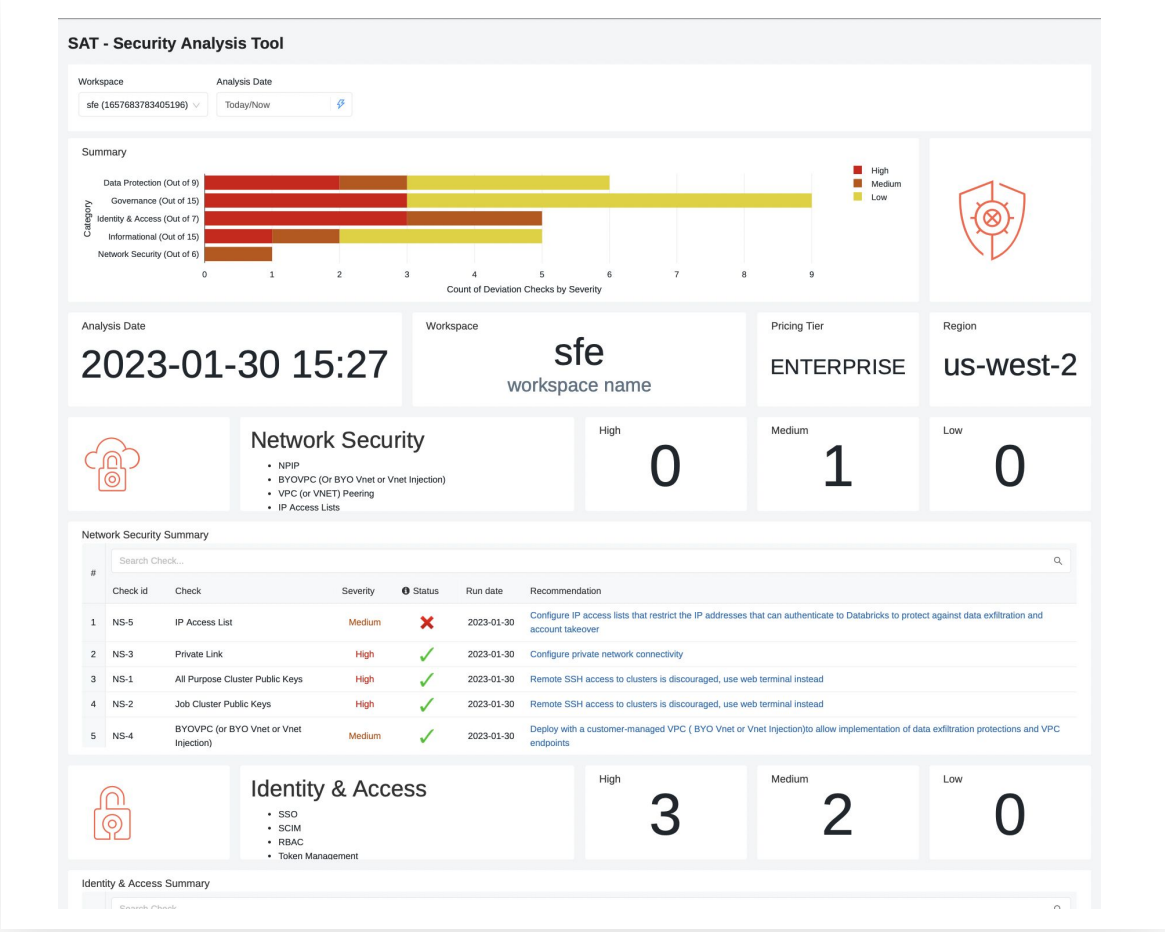
How do I know if I am following Databricks security best practices?

Our answer:

The Security Analytics Tool makes monitoring the security health of Databricks account workspaces easy for customers.

Security Analysis Tool

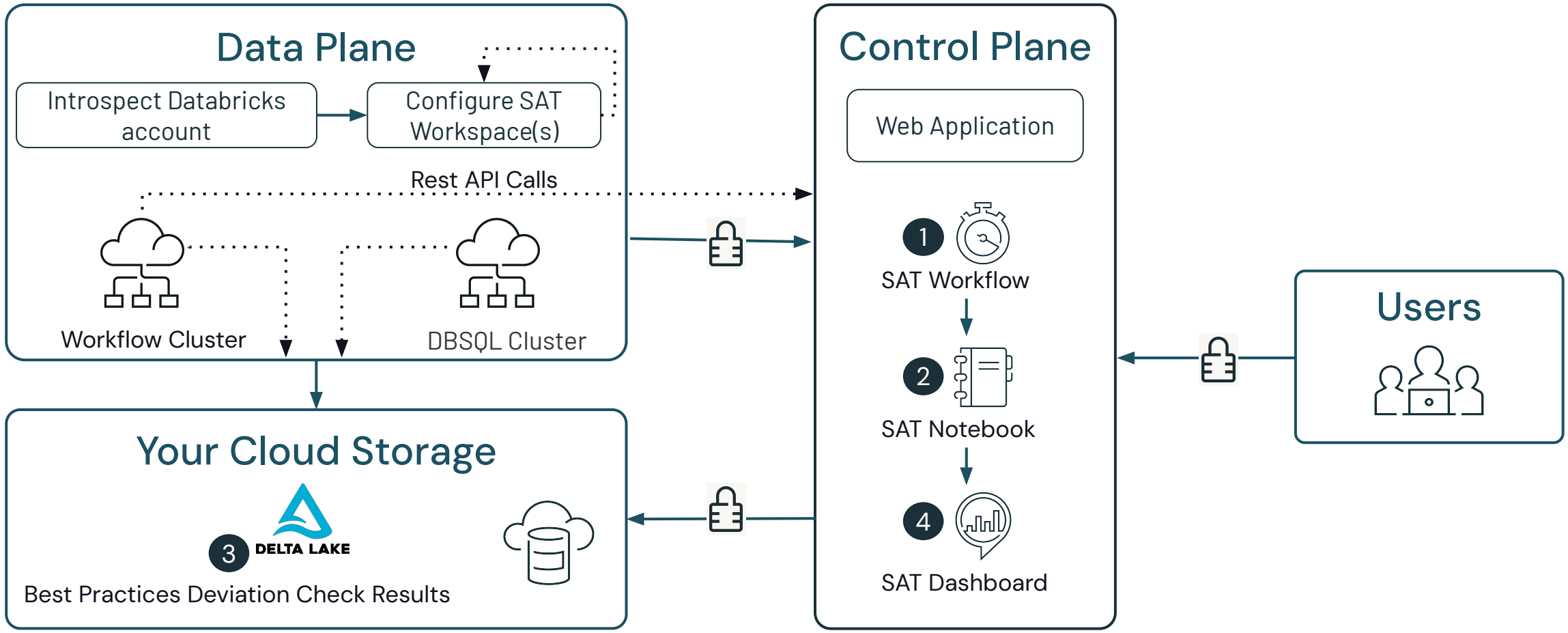
Monitor the security health of your account workspaces over time



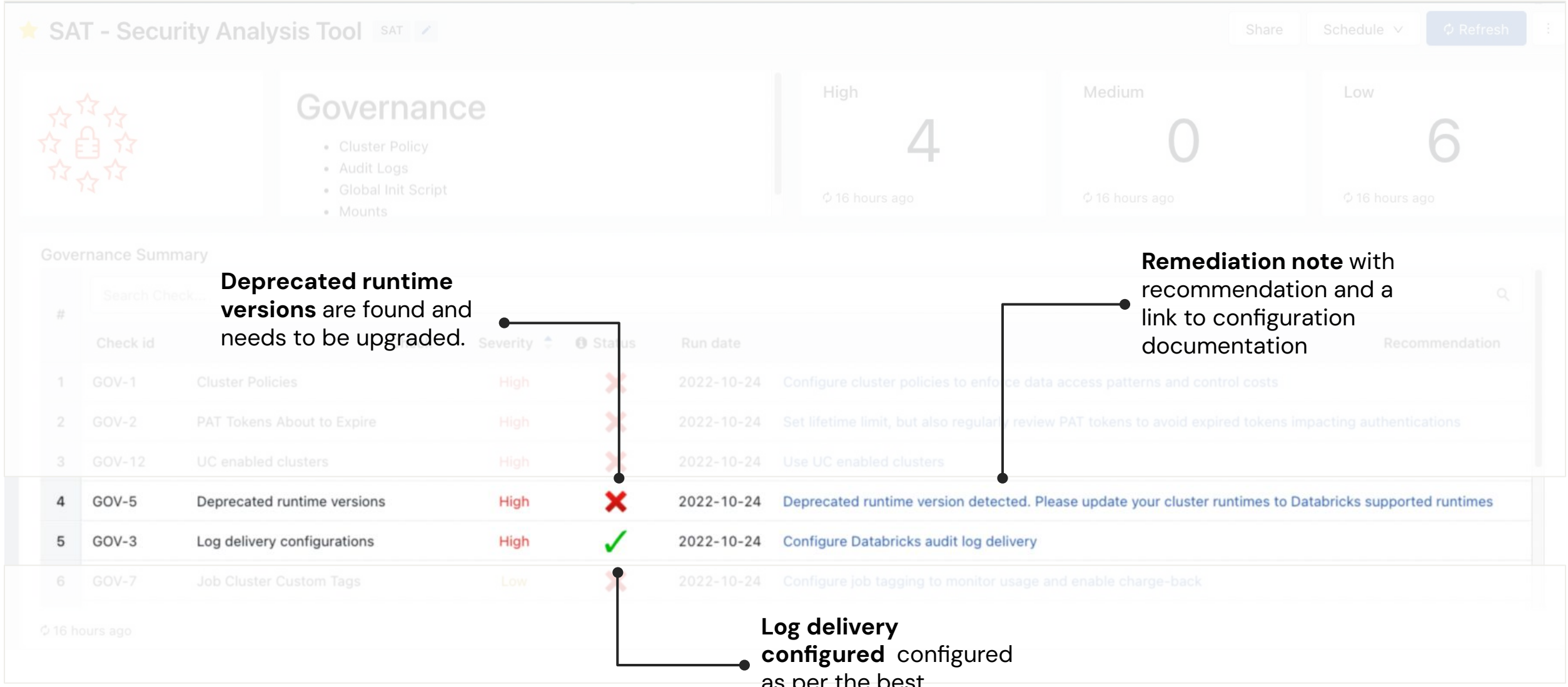
- Compare workspace configurations against specific best practices
- Automatically flag deviations and receive alerts for your account workspaces over a period of time
- Easily identify mitigation references
- Available for AWS, Azure and GCP

SAT helps data teams solve the world's toughest problems *safely*.

Deployment Architecture



Detection example



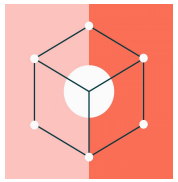
Deprecated runtime versions are found and needs to be upgraded.

Remediation note with recommendation and a link to configuration documentation

Log delivery configured configured as per the best practices

Security Analysis Tool Functionality

Best Practices



Monitors workspace settings of an account

Detects potential security deviations in minutes

Provides guidance & best practices

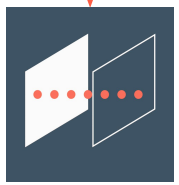
Easy connection from issue to solution documentation

Simplified Management

Centralized view of security configuration state

Scheduled to run automatically

Alerts on policy deviations



Audit Friendly

Tracks metrics over time

Compare configurations

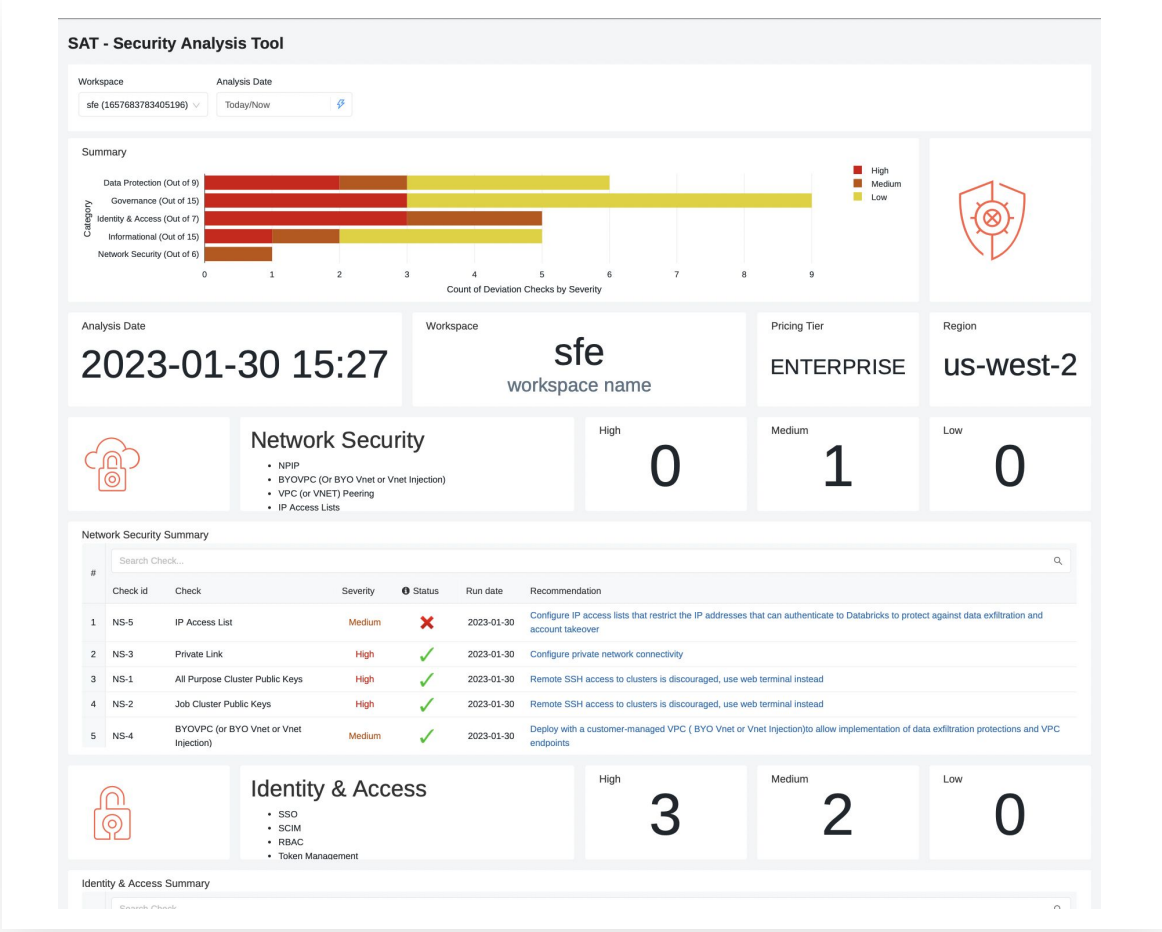
Simplified reporting

Microsoft Azure aws Google Cloud

Learn more—Security and Trust Center

Monitor and harden your workspaces with our list of best practices

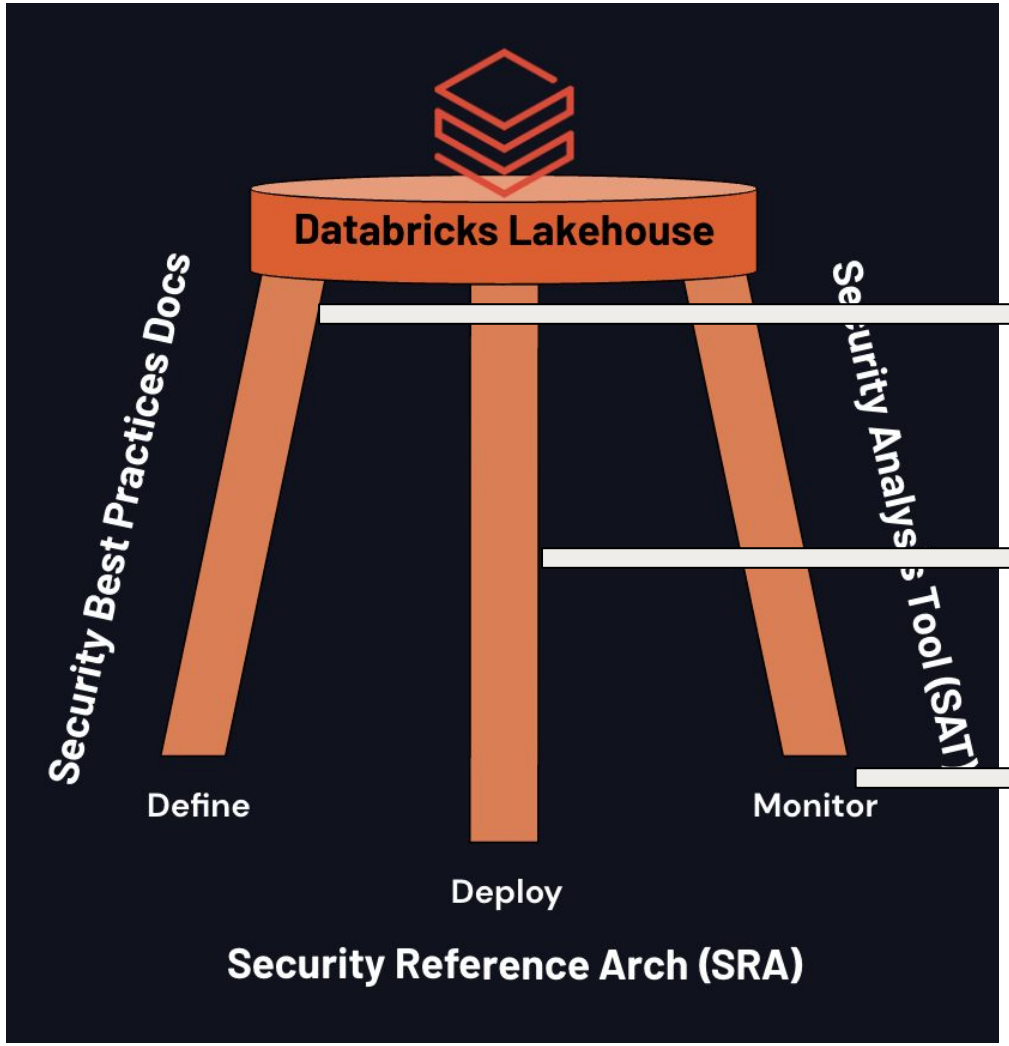
- Compare workspace configurations against specific best practices
- Automatically flag deviations and receive alerts for your account workspaces over a period of time
- Easily identify mitigation references
- Available for AWS, Azure and GCP (including Terraform deployments)



SAT helps data teams solve the world's toughest problems *safely*.

https://www.databricks.com/trust/security-features#best-practices

Closing remarks



Define: Prescriptive Best Practices Guides

Databricks has captured our best practices into a doc with “common” and “high-security” models and checkboxes. *Check those boxes.*

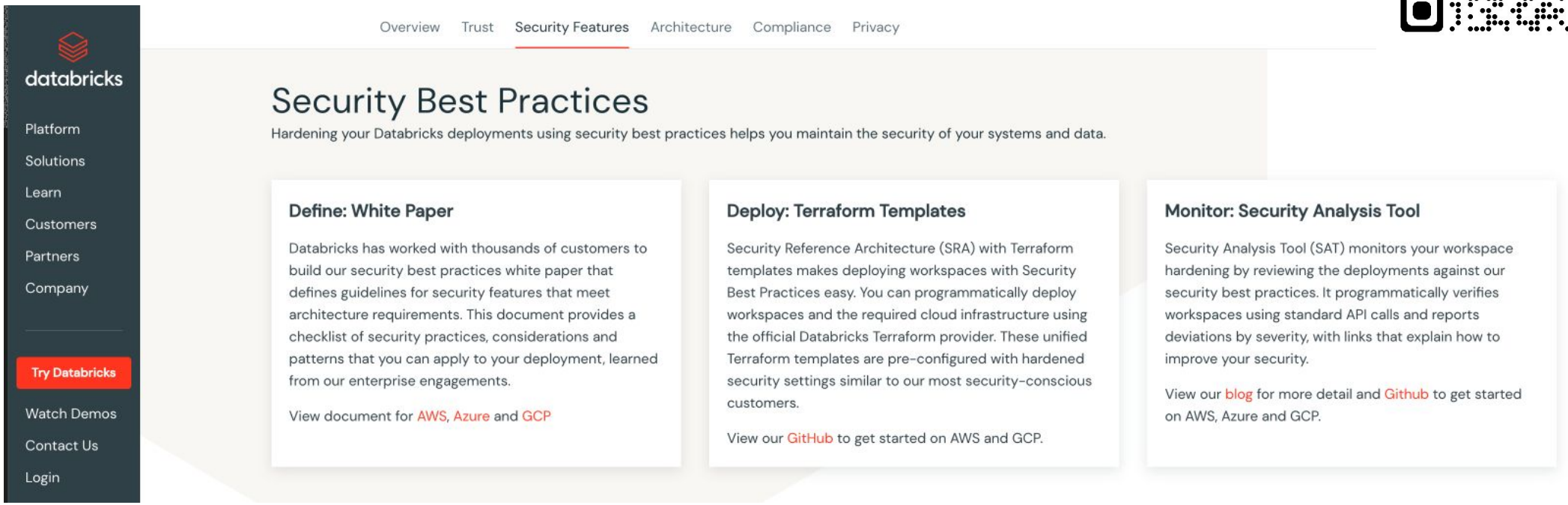
Deploy: Secure workspaces

Security Reference Architecture (SRA): Terraform Templates to deploy security hardened deployments

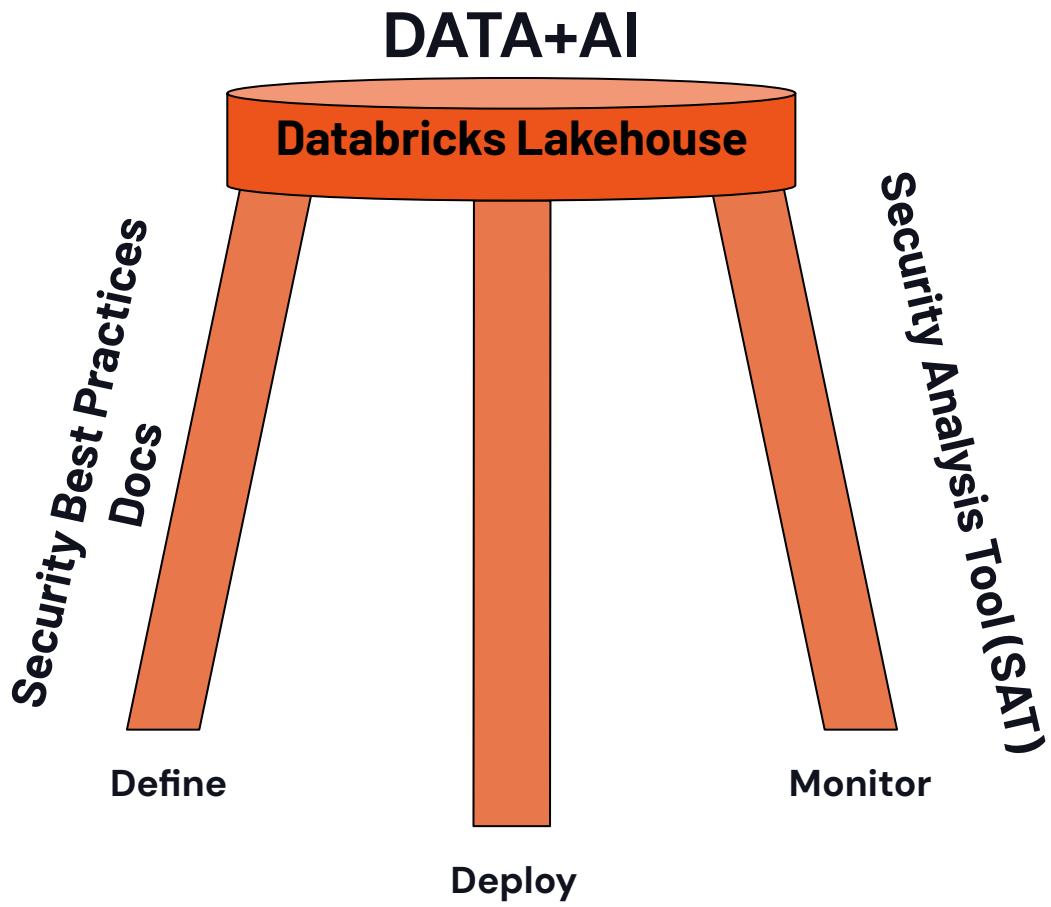
Monitor: Analyze your deployment

Security Analysis Tool (SAT): check your workspace against our most common best practices.

Summary

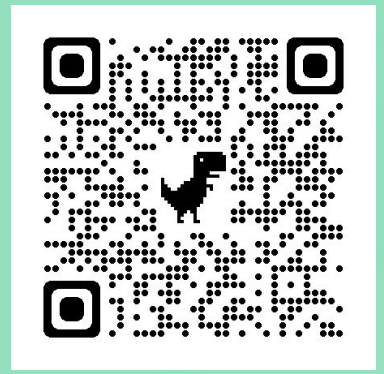


https://www.databricks.com/trust/security-features#best-practices



Security Reference Arch (SRA)
Terraform Templates

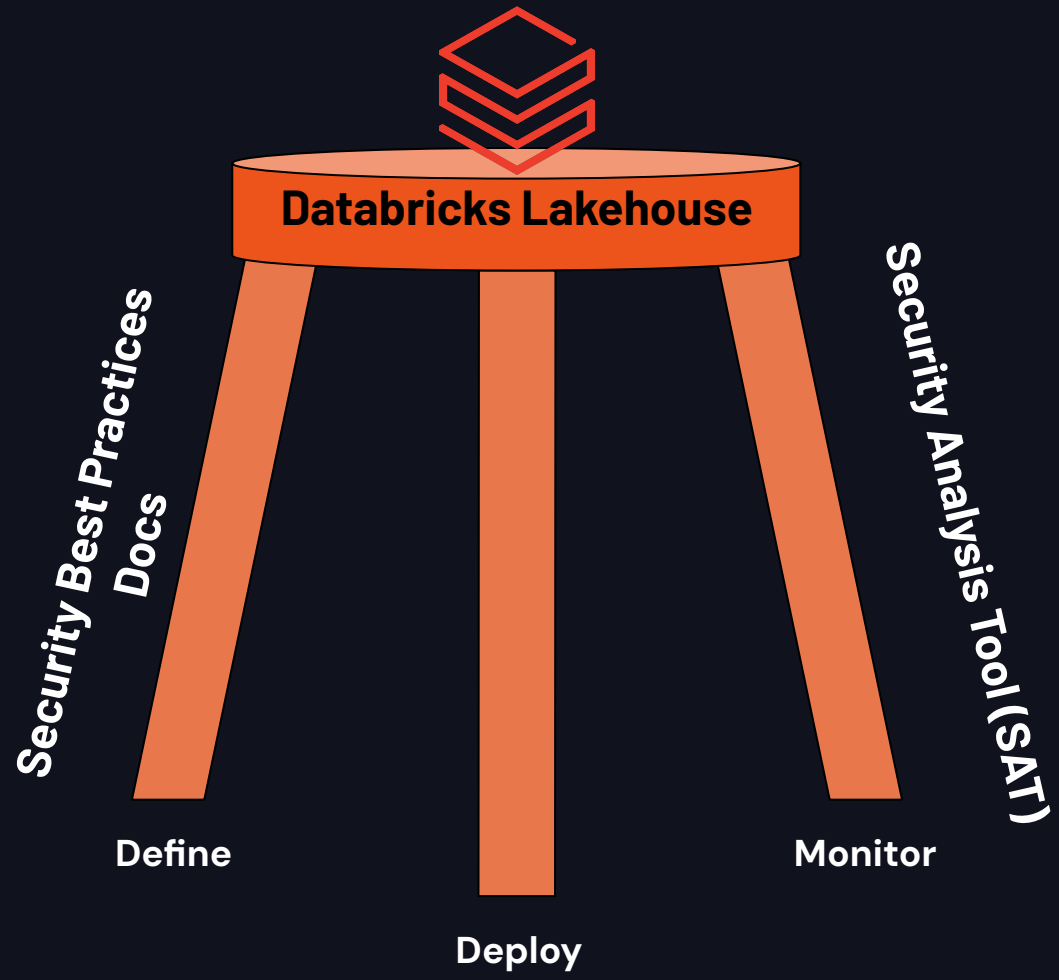
Thank you



It takes a village …

- **David Veuve**
  Head of Security Field Engineering
- **Andrew Weaver**
  Principal Specialist Solutions Architect
- **Silvio Fiorito**
  Principal Security Field Engineer
- **David Wells**
  Staff Security Field Engineer
- **Derek King**
  Senior Staff Security Field Engineer
- **Aliaksandra Nita**
  Sr. Technical Program Manager
- **Andrew Dowdell**
  Security Technical Program Coordinator
- **Abhi Arikapudi**
  Sr. Director, Security Engineering
- **Omar Khawaja**
  VP, Field CISO
- **Fermín Serna**
  Chief Security Officer
- **Kelly Albano**
  Product Marketing Manager
- **Bhavin Kukadia**
  Principal Specialist Solutions Architect
- **Ganesh Rajagopal**
  Lead Specialist Solutions Architect
- **Mohan Mathews**
  Sr. Delivery Solutions Architect
- **Suchi Pahi**
  Sr. Product & Privacy Counsel
- **Grace Chiang**
  Sr. Products Counsel
- **Filippo Seracini**
  Staff Product Manager
- **Greg Wood**
  Lead Product Specialist
- **Lipyeow Lim**
  Technical Director, Cybersecurity
- **Anindita Mahapatra**
  Lead Solutions Architect
- **Ramdas Murali**
  Lead Solutions Architect
- **Arun Pamulapati**
  Sr. Staff Security Field Engineer
- **Antonio Irizarry**
  Sr. Specialist Solutions Architect
- **Alex Ott**
  Lead Specialist Solutions Architect
- **JD Braun**
  Specialist Solutions Architect
- **Nathan Knox**
  Lead Specialist Solutions Architect
- **Ricardo Portilla**
  Principal Solutions Architect
- **Abhishek Pratap Singh**
  Specialist Solutions Architect
- **Alysson Souza**
  Specialist Solutions Architect

The second leg to your Lakehouse Security



Security Reference Arch (SRA)
Terraform Templates