Thousands of customers trust Databricks to analyze and build data products using machine learning (ML) with their most sensitive data. With significant investment into building a highly secure and scalable platform, Databricks delivers end-to-end platform security for data and users. This document provides an overview of the Databricks platform architecture, design choices and platform security features that enable your data teams to securely access relevant data while enforcing your data governance policies.

**NOTE:** Databricks has multiple deployment options. This document focuses on the E2 multi-tenant deployment on AWS and assumes at least the Premium tier Databricks subscription.

**Architecture**

The Databricks architecture is split to simplify your permissions, avoid data duplication and reduce risk.

### Databricks Platform Security

The Databricks E2 architecture for AWS provides more security for all your data and users.

**DATA PLANE**

- In your AWS account. It processes your data. Includes Databricks Runtime clusters and the networking to support them.

**CONTROL PLANE**

- In the Databricks AWS account. It manages your deployment. Includes web app, jobs, cluster management, configurations and notebooks.

---

**YOUR ACCOUNTS**

**DATA PLANE**

- Cluster
- S3 Bucket
- S3 Bucket

**CONTROL PLANE**

- Web Application
- Jobs
- Hive Metastore
- Cluster Manager
- Notebooks/Query Results
- ACLs/Sessions

**OTHER DATA**

- Data Sources
- S3 Bucket
- S3 Bucket

**USERS**

**INTERACTIVE USERS**

- Icon 1
- Icon 2

**BI APPLICATIONS**

- Icon 3
- Icon 4
END-TO-END EXAMPLE
Suppose you have a data engineer that signs in to Databricks and writes a notebook that transforms raw data in Kinesis to a normalized data set stored in S3.

SIX STEPS TO MAKE THE EXAMPLE HAPPEN
1. Your single sign-on (such as Okta) seamlessly authenticates the data engineer via SAML to the Databricks web UI in the control plane, hosted in the Databricks AWS account. Native authentication is also available.
2. As the data engineer writes code, their web browser sends it to the control plane. JDBC/ODBC requests also follow the same path, authenticating with a personal access token.
3. When ready, the control plane uses AWS APIs to create a Databricks cluster, made of new EC2 instances in the data plane, in your AWS account. Administrators can apply cluster policies to control AWS costs.
4. Once the instances launch, the cluster manager sends the data engineer’s code to the cluster.
5. The cluster pulls from Kinesis in your account, transforms the data in your account, and writes it to an S3 bucket in your account.
6. The cluster reports status and any outputs back to the cluster manager.

The data engineer doesn’t need to worry about many of the details — simply write the code and Databricks runs it. A key benefit of the hybrid PaaS model is that the vast majority of your actual data remains in systems under your control, such as your AWS account. While certain data, such as your notebooks, configurations, Apache Spark™ logs, and user information, is present within the control plane, that information is encrypted at rest within the control plane, and communication to and from the control plane is encrypted in transit. You also have choices for where certain data lives: You can host your own store of metadata about your data tables (Hive metastore), store query results in your AWS account, and you can decide whether to use the Databricks Secrets API.

Network and server security
In this section we’ll review networking, servers and how Databricks interacts with your AWS account.

NETWORKING
Databricks networking is straightforward. By default, Databricks configures networking for you, but you can also get more control of your data plane networking with the customer-managed VPC feature.

All data plane connections are outbound-only. Databricks does not rewrite or change your data structure in your storage, nor does it change or modify any of your security and governance policies. Local firewalls complement security groups to block unexpected inbound connections.

Customers at the Enterprise tier can also use the IP access list feature on the control plane to limit which IP addresses can connect to the web UI or REST API. For example, to only allow VPN or office IPs.
SERVERS

In the data plane, Databricks clusters automatically run the latest hardened base AMI with SSH access disabled by default and both local firewalls and security groups. Users cannot use an older (less secure) AMI. AMIs are typically updated every 2–4 weeks.

Databricks runs scans for every release, including:

1. Base OS scanning for vulnerabilities
2. Container scanning, including container dependencies
3. Static and dynamic code scanning

Databricks code is peer-reviewed by developers with security training. Significant design documents go through comprehensive security reviews. Scans run fully authenticated, with all checks enabled. Issues are tracked against the timeline shown in this table.

Importantly, Databricks clusters are typically short-lived (often terminated after a job completes) and do not persist data after they terminate. Clusters typically share the same permission level (excluding high concurrency clusters, where more robust security controls are in place). Your code is launched in an unprivileged container to maintain system stability. This security design provides protection against persistent attackers and privilege escalation.

DATABRICKS ACCESS

Databricks access to your environment includes a cross-account IAM role and the Genie system.

The cross-account IAM role allows the Databricks control plane to configure resources in your environment using the AWS APIs. It does not grant access to your data sets (see the next section).

The Genie system was built to allow Databricks staff to fix issues or support you, for example when you open a support request and check the box authorizing access to your deployment. Genie access requires either a support ticket or engineering ticket tied expressly to your workspace, and is limited to a subset of employees and for limited time periods. Additionally, if you’ve configured audit logs, they show the initial Genie event and Databricks staff’s actions.

Identity and access

Databricks supports robust ACLs, SAML 2.0 and SCIM. Many customers use built-in SAML integrations with Okta, Ping Identity, OneLogin, AAD, GSuite, ADFS or AWS. Customers can block non-SSO logins.

Databricks supports a variety of ways to enable users to access their data. Examples include:

1. IAM instance profiles enable the cluster to assume an IAM role, so users of that cluster automatically access allowed resources without the need for explicit credentials.
2. AWS IAM passthrough allows user roles rather than resource roles, enabling granular user-level permissions and auditing.
3. The Table ACLs feature uses traditional SQL-based statements to manage access to data as well as enabling fine-grained view-based access.
4. The Secrets API separates credentials from code when accessing external resources.

<table>
<thead>
<tr>
<th>SEVERITY</th>
<th>REMEDIATION TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>&lt; 14 days</td>
</tr>
<tr>
<td>High</td>
<td>&lt; 30 days</td>
</tr>
<tr>
<td>Medium</td>
<td>&lt; 60 days</td>
</tr>
<tr>
<td>Low</td>
<td>When appropriate</td>
</tr>
</tbody>
</table>

3 2 2
Data security

Databricks provides encryption, isolation and auditing.

### Data-at-rest encryption includes:
- Control plane encrypted
- Data plane supports local encryption
- Customers can use encrypted S3 buckets
- Customer-managed AWS KMS encryption keys for control plane notebooks, results and secrets

### Data-in-motion encryption includes:
- Control plane <-> data plane is encrypted
- Optional inter-node Spark traffic encryption
- Customer code can be written to avoid unencrypted services (e.g., FTP)

Customers can isolate users at multiple levels:

- **Workspace level:** Each team or department can use a separate workspace
- **Cluster level:** Cluster ACLs can restrict the users who can attach to a given cluster
- **High-concurrency clusters:** Process isolation, JVM whitelisting and limited languages (SQL, Python) allow for the safe coexistence of users of different privilege levels
- **Single-user cluster:** Users can create a private dedicated cluster

Activities of Databricks users are logged and can be delivered automatically to an S3 bucket. Customers can also monitor AWS provisioning activities by monitoring CloudTrail logs.

Compliance

Databricks supports the following compliance standards on our AWS multi-tenant platform:

- SOC 2 Type II
- ISO 27001
- ISO 27018 (Privacy-focused)

Alternative Databricks deployment types support FedRAMP High, HITRUST, HIPAA and PCI, if required.

Databricks Inc. and the Databricks platform are also GDPR and CCPA ready.

**LEARN MORE**

For more information, see the [full documentation](#) and our detailed Enterprise Security Guide. Databricks provides an enterprise-ready cloud platform that is built on a strong platform security posture for organizations small and large, and across all industries. We’re happy to discuss your specific needs in more detail — please reach out to your Databricks representative or email sales@databricks.com.

**EVALUATE DATABRICKS FOR YOURSELF**

Visit us at [databricks.com](http://databricks.com) and try Databricks free!