Snowflake Security Controls

August 28, 2023

This conversation may reflect the link creator’s Custom Instructions, which aren’t shared and can meaningfully change how the model responds.

---

we are onboarding snowflake in our firm, and our team will be mainting the product for the firm. please suggest optimum security controls with snowflakes

---

Onboarding a new data warehouse solution like Snowflake requires careful consideration of various security controls to protect both the data and the infrastructure. Here's a guide on implementing optimum security controls with Snowflake:

Data Encryption

- **At-rest Encryption**: Snowflake automatically encrypts data at rest. Make sure it is always enabled.
- **In-transit Encryption**: Use TLS for all data traveling over networks.

Access Control

- **Role-Based Access Control (RBAC)**: Assign roles based on the principle of least privilege.
  - Predefined roles: ACCOUNTADMIN, SECURITYADMIN, USER, etc.
  - Custom roles: Specific to your organizational needs.
- **Multi-Factor Authentication (MFA)**: Always enable MFA for accessing Snowflake.
- **Single Sign-On (SSO)**: Integrate with an existing identity provider for seamless and secure access.

Monitoring and Auditing

- **Query Monitoring**: Use Snowflake's History and Query features to monitor SQL queries.
- **Audit Trails**: Enable audit trails and regularly review them.
- **Alerts**: Set up alerts for unusual activities.

Network Security

VPC Peering: Establish a private connection between your VPC and Snowflake.
IP Whitelisting: Only allow trusted IPs to connect.

Data Masking and Tokenization

Dynamic Data Masking: Mask sensitive fields automatically.
Tokenization: Use tokenized fields for sensitive data where possible.

Operational Security

Patch Management: Always keep the Snowflake connector up to date.
Incident Response Plan: Have an incident response plan in place, customized for Snowflake.

Compliance

Data Governance: Ensure compliance with GDPR, CCPA, or any other regional laws.
Regular Audits: Perform security audits and compliance checks regularly.

Miscellaneous

Resource Monitors: Use Snowflake's resource monitor feature to keep track of consumption.
Time Travel and Fail-safe: Enable features for easy data recovery.

Documentation

Always document every security control, so that the team can easily understand the configurations and why they are necessary.

Taking these steps can considerably lower the risk of a security breach or unauthorized data access in your Snowflake environment

---

can you please list particular snowflake configuration and setting to enable the above configurations

---

Certainly, implementing the recommended security controls in Snowflake usually involves a mix of configuration settings and features that are either activated via the web interface or

SQL commands. Below are the details on how you can enable or enforce some of these configurations:

Data Encryption

At-rest Encryption

Automatically enabled; no action required.

In-transit Encryption

Enabled by default when using Snowflake drivers and connectors.

Access Control

Role-Based Access Control (RBAC)

Create custom roles via SQL:

sql Copy code

```sql
CREATE ROLE custom_role;
```

Grant privileges to the role:

sql Copy code

```sql
GRANT USAGE ON DATABASE database_name TO ROLE custom_role;
```

Multi-Factor Authentication (MFA)

Enable through Snowflake Web UI in the Account settings or via SQL.

Single Sign-On (SSO)

Configure SSO through identity provider settings and Snowflake Account settings.

Monitoring and Auditing

Query Monitoring

Use `QUERY_HISTORY` or `QUERY_HISTORY_BY_*` functions for auditing.

Audit Trails

Use the `ACCOUNT_USAGE` schema for monitoring.

Alerts

Snowflake doesn't natively support alerting, but you can use external tools like SnowAlert.

Network Security

VPC Peering

Enable through your cloud provider's network settings and then link it to Snowflake.

IP Whitelisting

Set network policies in Snowflake to allow specific IPs:

sql Copy code

```sql
CREATE NETWORK POLICY trusted_ips ALLOWED_IP_LIST =
('192.168.1.1','192.168.1.2');
```

Data Masking and Tokenization

Dynamic Data Masking

Use the `MASKING_POLICY` expression when defining a table.

sql Copy code

```sql
CREATE MASKING POLICY ssn_masking_policy RETURNS STRING -> CASE WHEN
CURRENT_ROLE() IN ('FULL_ACCESS_ROLE') THEN val ELSE 'XXX-XX-' ||
RIGHT(val, 4) END;
```

Tokenization

No native support; use third-party services to tokenize data before inserting it into Snowflake.

Operational Security

Patch Management

Check Snowflake documentation for driver and connector updates.

Incident Response Plan

Custom to the organization; no specific Snowflake setting.

Compliance

Data Governance

Mostly organizational but can use Snowflake features like `DATA_RETENTION_TIME_IN_DAYS` to set retention policies.

Regular Audits

Utilize Snowflake's `ACCOUNT_USAGE` schema to conduct audits.

Miscellaneous

Resource Monitors

Create a resource monitor via SQL:

sql Copy code

```sql
CREATE RESOURCE MONITOR "my_monitor" FREQUENCY = '1 HOUR' CREDIT_QUOTA =
100;
```

Time Travel and Fail-safe

Set data retention policies during table creation:

sql Copy code

```sql
CREATE TABLE my_table DATA_RETENTION_TIME_IN_DAYS = 14;
```

---

anything else important that need to be considered

In addition to the technical settings and configurations, there are several other broader considerations that are crucial for maintaining robust security when onboarding Snowflake:

Training and Awareness

Staff Training: Ensure that all staff involved in managing or using Snowflake are trained in best security practices relevant to data warehousing and Snowflake specifically.

Phishing Awareness: Educate users about the risks of phishing attacks, as human error often serves as an entry point for security breaches.

Vendor Security Assessment

Periodic Reviews: Periodically review Snowflake's security features and how they align with your organization's requirements. This could be a formal vendor security assessment.

Data Backup and Recovery

Backup Strategy: Even though Snowflake has its fail-safes, you should still have a data backup strategy in place, which could include exporting data to a secure location.

Disaster Recovery Plan: Document a plan outlining how to recover data and services in case of various types of failures or security incidents.

Data Privacy

Data Classification: Identify the types of data stored in Snowflake and classify them based on sensitivity.
PII and PHI: If you're storing personally identifiable information (PII) or protected health information (PHI), additional protections and compliance requirements like HIPAA may apply.

Performance Considerations

Throttling: Consider setting up resource monitors to prevent any runaway queries that could not only increase costs but also potentially impact performance, causing delays in data accessibility, which can be a security risk in certain scenarios.

Integration with Other Tools

Security Information and Event Management (SIEM): Integrate Snowflake logs with a SIEM solution for real-time analysis of security alerts.
Data Loss Prevention (DLP): Consider using DLP solutions to monitor and control data transferring across the company's network.
API Security: If you're integrating Snowflake with other systems via APIs, make sure those connections are secure.

Documentation and Record-Keeping

Maintain Records: Document any changes in settings, permissions, and other configurations, along with the reasoning and authorization behind the change.
Audit Logs: Maintain logs for a reasonable period to assist in any future investigations, audits, or compliance checks.

Regular Audits and Inspections

Third-Party Audits: Regularly engage with third-party security services for audits to identify vulnerabilities and suggest improvements.
Internal Audits: Conduct periodic internal audits and security assessments.

Taking a comprehensive, layered approach to security, encompassing technical, organizational, and human factors, will provide the most robust protection when onboarding