Workflow

1. API Management accepts API calls in the form of HTTP requests.
2. API Management securely routes the HTTP requests to Logic Apps.
3. Each HTTP request triggers a run in Logic Apps:
    1. Logic Apps uses secured template parameters to retrieve database credentials from Azure Key Vault.
    2. Logic Apps uses Transport Layer Security (TLS) to send the database credentials and a database statement to the on-premises data gateway.
4. The on-premises data gateway connects to a SQL Server database to run the statement.
5. SQL Server stores the data and makes it available to apps that users access.
6. Azure Monitor collects information on Logic Apps events and performance.

Components

This architecture uses the following components:

- Azure API Management creates consistent, modern API gateways for back-end services. Besides accepting API calls and routing them to back ends, this platform also verifies keys, tokens, certificates, and other credentials. API Management also enforces usage quotas and rate limits and logs call metadata.
- Azure Logic Apps automates workflows by connecting apps and data across clouds. This service provides a way to securely access and process data in real time. Its serverless solutions take care of building, hosting, scaling, managing, maintaining, and monitoring apps.
- An on-premises data gateway acts as a bridge that connects on-premises data with cloud services like Logic

Apps. Typically, you install the gateway on a dedicated on-premises virtual machine. The cloud services can then securely use on-premises data.

- Azure Key Vault stores and controls access to secrets such as tokens, passwords, and API keys. Key Vault also creates and controls encryption keys and manages security certificates.
- SQL Server provides a solution for storing and querying structured and unstructured data. This database engine features industry-leading performance and security.
- Azure Monitor collects data on environments and Azure resources. This information is helpful for maintaining availability and performance. Other Azure services, such as Azure Storage and Azure Event Hubs, can also use this diagnostics data. Two data platforms make up Monitor:
  - Azure Monitor Logs records and stores log and performance data. For Logic Apps, this data includes information on trigger events, run events, and action events.
  - Azure Monitor Metrics collects numerical values at regular intervals. For Logic Apps, this data includes the run latency, rate, and success percentage.

Alternatives

A few alternatives exist for this solution:

- Instead of using an on-premises instance of SQL Server, consider migrating to an up-to-date, fully managed Azure database service. The SQL Server connector that Logic Apps uses also works for Azure SQL Database and Azure SQL Managed Instance. For more information, see Automate workflows for a SQL database by using Azure Logic Apps. To get started with migration, see Azure Database Migration Service.
- For complex automation tasks, consider using Azure Functions instead of Logic Apps. For more information, see Compare Azure Functions and Azure Logic Apps.

- For simple integrations, consider using Power Automate instead of Logic Apps. For more information, see Compare Microsoft Power Automate and Azure Logic Apps.
- Power Apps also provides solutions for automating workflows that involve connecting to on-premises data sources.

Scenario details

A logic app can store HTTP request data in a SQL Server database. Because Logic Apps functions as a secure Azure API Management endpoint, calls to your API can trigger various data-related tasks. Besides updating on-premises databases, you can also send Teams or email messages.

Potential use cases

Use this solution to automate data integration tasks that you perform in response to API calls.

Considerations

Keep these points in mind when considering this architecture.

Availability

For high availability, add the on-premises gateway to a cluster instead of installing a standalone gateway.

Scalability

With the serverless model that Logic Apps uses, the service automatically scales to meet demand. But be aware of limits on read and write operations with the on-premises data gateway.

Security

- The on-premises data gateway uses credential encryption and user authentication to protect data during transfers between on-premises and Azure systems.
- API Management helps to ensure that only authorized clients call your logic app. You can also take these steps:
  - Since API Management is the only client that should call your logic app, consider restricting your app's inbound IP addresses. You can configure your logic app to only accept requests from the IP address of your API Management service instance.
  - You can also use one of these authorization schemes to limit access to your logic app:
    - Shared access signatures (SAS).
    - Azure Active Directory Open Authentication (Azure AD OAuth).
- Consider using Azure role-based access control (Azure RBAC) to only permit specific users or groups to manage, edit, and view your logic apps.
- Information is available on each logic app run, such as the status, duration, inputs, and outputs for each action. Use one of these methods to control who can access the inputs and outputs in the run history:
  - Restrict access by IP address range.
  - Use obfuscation to secure run history data.