

5. Connect to and consume Azure services and third-party services

└ 5.2 Develop event-based solutions

└ 5.2.1 Implement solutions that use Azure Event Grid

1. What is Azure Event Grid and what are its primary use cases?
2. What are the key components of Event Grid (events, topics, event subscriptions)?
3. How do you create a custom topic and event subscription via Azure CLI?
4. What event sources can trigger Event Grid?
5. What are the supported event handlers (destinations)?
6. How do you secure Event Grid with authentication and authorization?
7. How do you filter events in Event Grid subscriptions?
8. How do you enable dead-lettering for undeliverable events?
9. How do you validate Event Grid event schema in consumers?
10. How do you troubleshoot failed event deliveries?
11. Is Event Grid push or pull-based?
12. What delivery guarantees does Event Grid provide?
13. What is the default schema of an Event Grid event?
14. How do you choose between Event Grid, Event Hubs, and Service Bus?
15. How are retries and error handling managed in Event Grid?

1. What is Azure Event Grid and what are its primary use cases?

Azure Event Grid is a fully managed event routing service that enables reactive programming using events from Azure services or custom sources.

Use cases: Event-driven architecture, serverless automation, resource provisioning notifications, microservices decoupling.

2. What are the key components of Event Grid?

- Event Sources: Where events originate (e.g., Blob Storage, Resource Groups).
- Topics: Channels where events are sent.
- Event Subscriptions: Define how to handle events (e.g., send to Function, Webhook).
- Event Handlers: Endpoints that process events (e.g., Azure Function, Logic Apps).

3. How do you create a custom topic and event subscription via Azure CLI?

```
az eventgrid topic create --name mytopic --resource-group myrg --location eastus
az eventgrid event-subscription create --name mysub --source-resource-id
/subscriptions/<id>/resourceGroups/myrg/providers/Microsoft.EventGrid/topics/mytopic --endpoint <url>
```

4. What event sources can trigger Event Grid?

- Native: Blob Storage, Resource Groups, Event Hubs, IoT Hub, Media Services, etc.
- Custom sources: Via custom topics.
- Third-party: via webhook-compatible services.

5. What are the supported event handlers (destinations)?

- Azure Function
- Logic Apps
- Event Hubs
- Webhooks (HTTP/S endpoint)
- Service Bus Queue/Topic

6. How do you secure Event Grid with authentication and authorization?

- Event delivery: Signed with validationCode or Azure Active Directory token.
- Inbound auth: Use Azure RBAC for publishing to topics.
- Outbound auth: Use Webhook validation and shared access keys for endpoints.

7. How do you filter events in Event Grid subscriptions?

Use `--included-event-types` or advanced filters:

`--advanced-filter data.subject StringBeginsWith "/blobServices/default/containers/images"`

Filters reduce noise and only forward relevant events.

8. How do you enable dead-lettering for undeliverable events?

Set `--deadletter-destination` when creating a subscription, usually to a Blob Storage container:

`--deadletter-destination blobcontainer:<storage-account>/<container>`

This stores failed events for later review.

9. How do you validate Event Grid event schema in consumers?

Consumers must handle a **validationEvent** initially.

Respond to the `eventType = Microsoft.EventGrid.SubscriptionValidationEvent` by echoing back `data.validationCode`.

10. How do you troubleshoot failed event deliveries?

- Enable dead-lettering.
- Check metrics in Azure Monitor.
- Use `az eventgrid event-subscription show` to inspect subscription state.
- Inspect logs at the handler side (e.g., Azure Function failures).

11. Is Event Grid push or pull-based?

Event Grid is push-only — it pushes events to subscribers via HTTP POST; consumers must expose endpoints to receive events.

12. What delivery guarantees does Event Grid provide?

At-least-once delivery.

Events are retried with exponential backoff for up to 24 hours if the destination is unavailable.

13. What is the default schema of an Event Grid event?

Default schema fields include:

- `id`, `eventType`, `subject`, `data`, `eventTime`, `dataVersion`, `metadataVersion`.
- Custom topics use this schema unless configured for CloudEvents v1.0.

14. How do you choose between Event Grid, Event Hubs, and Service Bus?

- Event Grid: Discrete events, fan-out, serverless, push-only.
- Event Hubs: Telemetry, streaming data, high-throughput ingestion.
- Service Bus: Reliable message delivery with ordering and sessions.

15. How are retries and error handling managed in Event Grid?

- Retries: Exponential backoff for transient failures.
- Permanent failure: After 24 hours or persistent errors, events are dead-lettered if configured.
- Subscriber should return HTTP 2xx for success, otherwise it triggers retries.