NATS
(Beginner)

COURSE OVERVIEW

Prerequisites

- Participants are expected to be developers who have knowledge of microservices development (with either Python/JavaScript/Java or Go) and are familiar with Linux
- Knowledge of Docker, Kubernetes and Istio is a good to have

Course Objectives

- NATS Architecture and Design
- Understanf NATS Clustering Architecture
- NATS based development
- More About Subjects
- Core NATS

Lab Setup

Hardware Requirements

- At least 25GB of free hard-disk space, 4GB (minimum) of RAM and Full Internet Connectivity is required

Software Requirements

- Ubuntu Linux Server or Desktop 24.04 LTS in a VM

- The rest of the software required for the training will be installed during the training itself Lab VMs will be provided if needed

Course Duration

- 2 full days Or 3 half days

Detailed Course Outline

Section 1 – The Big Picture

- Stateless and Stateful components in Cloud Based Distributed System Architectures
- Automating handling of scalability and failures in Stateless/Stateful Components
- Overview of Microservices Architecture
- Inter-service communication, service-discovery issues in Microservices
- Handling scalability and failures in Microservices
- Enforcing traffic management and security policies in Microservices
- Observability of Microservice communication
- RAFT consensus algorithm
- Asynchronous programming model
- Event Driven Architectures
- Role of Messaging in Microservies
- Comparison with
    - Messaging systems
    - NoSQL databases
    - Object storage

Section 2 – Overview of NATS

- What is NATS?
- NATS Architecture and Design
- Comparison of NATS with Kubernetes/Istio based architectures
- Core components: Server, Client, Subjects, and Messages
- Getting Started
  - Installing NATS locally
  - Installing and using NATS via Docker
  - Hello World!
- NATS Clustering Architecture
  - Single server setup
  - Cluster and super cluster with gateways and leaf nodes
  - Cross-cluster communication strategies

Section 3 – NATS based development

- Introduction to Subjects
- Connecting/Disconnecting from a NATS server
- Publish and subscribe to messages
- Request and reply to messages
- Working with structured data

Section 4 – More About Subjects

- Subject based messaging
- Wildcards
- Subject hierarchies
- Subject naming best practices

Section 5 – Core NATS

- Publish/Subscribe
- Request/Reply
- Queue Groups
- Delivery patterns

Section 6 – JetStream

- Overview of Streams
- Stream Configuration
- Stream Replication
- Stream Persistence and Data Retention
- File Based v/s Memory-based Storage
- Key/Value and Object Stores
- Stream Transforms

Section 7 – Revisiting NATS based development

- Detecting and handling server failures
- Dealing with disconnections
- Client reconnection strategies and message durability
- Store and retrieve messages from KV buckets
- Store and retrieve blobs from object store

Section 8 – NATS Tools

- NATS Monitoring
- Use of Prometheus, Prometheus NATS Exporter and Grafana for monitoring

Section 9 – Securing NATS

- Concept of End to End Encryption and Zero Trust architecture
- Authentication with JWT
- Authentication with NKeys
- TLS Enabling NATS

Section 10 – Additional Topics

- Running NATS in Kubernetes via Helm
- Using NATS with Websockets/MQTT
- Profiling NATS