

Apache Kafka 3

## Working with Kafka topics

**Date of Publish:** 2019-03-08



<https://docs.hortonworks.com>

# Contents

<b>Creating a Kafka Topic.....</b>	<b>3</b>
------------------------------------	----------

## Creating a Kafka Topic

As described in [Apache Kafka Concepts](#), Kafka maintains feeds of messages in categories called topics. Producers write data to topics and consumers read from topics. Since Kafka is a distributed system, topics are partitioned and replicated across multiple nodes. Kafka treats each topic partition as a log (an ordered set of messages). Each message in a partition is assigned a unique offset.

Each topic has a user-defined category (or feed name), to which messages are published.

To create a Kafka topic, run `kafka-topics.sh` and specify topic name, replication factor, and other attributes:

```
/bin/kafka-topics.sh --create \  
  --zookeeper <hostname>:<port> \  
  --topic <topic-name> \  
  --partitions <number-of-partitions> \  
  --replication-factor <number-of-replicating-servers>
```

The following example creates a topic named "test", with one partition and one replica:

```
bin/kafka-topics.sh --create \  
  --zookeeper localhost:2181 \  
  --replication-factor 1 \  
  --partitions 1 \  
  --topic test
```

To view the topic, run the list topic command:

```
> bin/kafka-topics.sh --list --zookeeper localhost:2181  
test
```

To create topics on a cluster with Kerberos enabled, see [Creating A Kafka Topic](#).

The `auto.create.topics.enable` property, when set to true, automatically creates topics when applications attempt to produce, consume, or fetch metadata for a nonexistent topic. For more information, see [Kafka Broker Settings](#).