Amazon ElastiCache – Redis Engine

## Service Overview

Amazon ElastiCache makes it easy to set up, manage, and scale distributed in-memory cache environments in the AWS Cloud. It provides a high performance, resizable, and cost-effective in-memory cache, while removing complexity associated with deploying and managing a distributed cache environment. ElastiCache works with both the Redis and Memcached engines.

The basic building block of ElastiCache for Redis is the [cluster](https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/WhatIs.html#WhatIs.Clusters). A cluster is a collection of one or more cache nodes, all of which run an instance of the Redis cache engine software.

A [node](https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/WhatIs.Components.html#WhatIs.Components.Nodes)is the smallest building block of an ElastiCache deployment. A node can exist in isolation form or in some relationship to other nodes. A node is a fixed-size chunk of secure, network-attached RAM. Each node runs an instance of the engine and version that was chosen when you created your cluster.

[Replication](https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/WhatIs.Components.html#WhatIs.Components.ReplicationGroups) is implemented by grouping from two to six nodes in a [shard](https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/WhatIs.Components.html#WhatIs.Components.Shards) (in the API and CLI, called a node group). One of these nodes is the read/write primary node. All the other nodes are read-only replica nodes.

It’s also important to notice that [ElastiCache is fully integrated with Amazon VPC](https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/VPCs.EC.html), using the same principles, which are used for EC2 VMs.

## Use cases / Considerations

One of the most important choices in working with ElastiCache is whether to use Redis or Memcached engine. You can use [this cheetsheet](https://d0.awsstatic.com/whitepapers/performance-at-scale-with-amazon-elasticache.pdf#page=8&zoom=100,92,493) to decide what suits your needs better.

You also might want to take a look at [reserved nodes](https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/CacheNodes.Reserved.html) as a way to reduce costs, if you can predict what kind of load will you have.

## Governance

As most of AWS services, ElastiCache is fully integrated with [AWS CloudWatch](https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/CacheMetrics.html) for metrics, [AWS CloudTrail](https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/logging-using-cloudtrail.html) for logging and it also supports [AWS SNS](https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/ECEvents.html), sending all important events to a specific topic.

For security, ElastiCache provides [several types of encryption](https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/encryption.html) and [Role Based Access Control](https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/Clusters.RBAC.html).

## Cautions

## Pricing considerations

All info regarding ElastiCache pricing can be found in [AWS docs](https://aws.amazon.com/elasticache/pricing/)

## More details

<https://d0.awsstatic.com/whitepapers/performance-at-scale-with-amazon-elasticache.pdf>