# Running Redis as a Service

If you installed Redis using the MSI package, then Redis was already installed as a Windows service. Nothing further to do. If you would like to change its settings, you can update the *redis.windows-service.conf* file and then restart the Redis service (Run -> services.msc -> Redis -> Restart).

During installation of the MSI you can either use the installer’s user interface to update the port that Redis listens at and the firewall exception or run it silently without a UI. The following examples show how to install from the command line:

**default install (port 6379 and firewall exception ON):**

 *msiexec /i Redis-Windows-x64.msi*

**set port and turn OFF firewall exception:**

 *msiexec /i Redis-Windows-x64.msi PORT=1234 ADD\_FIREWALL\_RULE=""*

**set port and turn ON firewall exception:**

 *msiexec /i Redis-Windows-x64.msi PORT=1234 ADD\_FIREWALL\_RULE =1*

**install with no user interface:**

 *msiexec /quiet /i Redis-Windows-x64.msi*

If you did *not* install Redis using the MSI package, then you still run Redis as a Windows service by following these instructions:

In order to better integrate with the Windows Services model, new command line arguments have been introduced to Redis. These service arguments require an elevated user context in order to connect to the service control manager. If these commands are invoked from a non-elevated context, Redis will attempt to create an elevated context in which to execute these commands. This will cause a User Account Control dialog to be displayed by Windows and may require Administrative user credentials in order to proceed.

## Installing the Service

*--service-install*

This must be the first argument on the redis-server command line. Arguments after this are passed in the order they occur to Redis when the service is launched. The service will be configured as Autostart and will be launched as "NT AUTHORITY\NetworkService". Upon successful installation a success message will be displayed and Redis will exit.

This command does not start the service.

For instance:

redis-server --service-install redis.windows-service.conf --loglevel verbose

## Uninstalling the Service

*--service-uninstall*

This will remove the Redis service configuration information from the registry. Upon successful uninstallation a success message will be displayed and Redis will exit.

This does command not stop the service.

For instance:

redis-server --service-uninstall

## Starting the Service

*--service-start*

This will start the Redis service. Upon successful start, a success message will be displayed and Redis will begin running.

For instance:

redis-server --service-start

## Stopping the Service

*--service-stop*

This will stop the Redis service. Upon successful termination a success message will be displayed and Redis will exit.

For instance:

redis-server --service-stop

## Naming the Service

*--service-name* ***name***

This optional argument may be used with any of the preceding commands to set the name of the installed service. This argument should follow the service-install, service-start, service-stop or service-uninstall commands, and precede any arguments to be passed to Redis via the service-install command.

The following would install and start three separate instances of Redis as a service:

redis-server --service-install --service-name redisService1 --port 10001

redis-server --service-start --service-name redisService1

redis-server --service-install --service-name redisService2 --port 10002

redis-server --service-start --service-name redisService2

redis-server --service-install --service-name redisService3 --port 10003

redis-server --service-start --service-name redisService3