

Anarion Technologies – Redis

Products > Redis v7.4.2 on Ubuntu v20



Redis v7.4.2 on Ubuntu v20 ♥ Save to my list

Anarion Technologies

Overview [Plans + Pricing](#) [Ratings + reviews](#)

Ready to use VM for Production + Free Support

Redis (Remote Dictionary Server) is a versatile, open-source, in-memory data structure store that can function as a database, cache, and message broker. Known for its high performance and low latency, Redis excels in scenarios requiring rapid data access and real-time processing. It supports a variety of data structures, including strings, hashes, lists, sets, and sorted sets, which provides flexibility for developers to optimize their applications.

One of Redis's standout capabilities is its in-memory storage, which allows for incredibly fast read and write operations. This makes it ideal for use cases such as caching, where quick data retrieval is crucial, or for real-time analytics, where rapid data ingestion and querying are necessary. Redis also supports persistence through mechanisms like snapshots and append-only files, ensuring data durability even though it primarily operates in-memory.

Redis is highly reliable, offering features like replication, which allows data to be copied to multiple servers for high availability and disaster recovery. Its clustering capabilities enable horizontal scaling by distributing data across multiple nodes, making it suitable for applications with large datasets and high throughput requirements.

The platform also includes a powerful publish/subscribe messaging system, enabling real-time communication between clients. This is particularly useful for applications that require live updates, such as chat applications or real-time notifications.

Overall, Redis's combination of speed, flexibility, and robustness makes it a popular choice for developers looking to build efficient and scalable applications. Its rich ecosystem and strong community support further enhance its appeal as a go-to solution for modern data management challenges.

[Get It Now](#)

Categories

[Databases](#)
[Compute](#)

Support

[Support](#)

Legal

[License Agreement](#)
[Privacy Policy](#)

Create this app in Azure



Redis v7.4.2 on Ubuntu v20
By Anarion Technologies

Software plan

Redis

Pricing: Starting at \$0.0019/hour

Details: Redis is a high-performance, in-memory data structure store used as a database, cache, and message broker, known for its speed and flexibility.

This app requires some basic profile information. You have provided the information already so you're good to go! [Edit](#)

By clicking "Continue", I grant Microsoft permission to share my supplied contact information with the provider so that they can contact me regarding this product and related products. The shared information will be handled in accordance with the provider's [terms](#) and [privacy statement](#).

[Continue](#)

Home >

Redis v7.4.2 on Ubuntu v20 (preview) ✨ ...

Anarion Technologies



Redis v7.4.2 on Ubuntu v20 (preview) [Add to Favorites](#)

Anarion Technologies | Virtual Machine

Plan

Redis

Create

Start with a pre-set configuration

Want to deploy programmatically? [Get started](#)

[Overview](#) [Plans + Pricing](#) [Usage Information + Support](#) [Ratings + Reviews](#)

Redis (Remote Dictionary Server) is a versatile, open-source, in-memory data structure store that can function as a database, cache, and message broker. Known for its high performance and low latency, Redis excels in scenarios requiring rapid data access and real-time processing. It supports a variety of data structures, including strings, hashes, lists, sets, and sorted sets, which provides flexibility for developers to optimize their applications.

One of Redis's standout capabilities is its in-memory storage, which allows for incredibly fast read and write operations. This makes it ideal for use cases such as caching, where quick data retrieval is crucial, or for real-time analytics, where rapid data ingestion and querying are necessary. Redis also supports persistence through mechanisms like snapshots and append-only files, ensuring data durability even though it primarily operates in-memory.

Redis is highly reliable, offering features like replication, which allows data to be copied to multiple servers for high availability and disaster recovery. Its clustering capabilities enable horizontal scaling by distributing data across multiple nodes, making it suitable for applications with large datasets and high throughput requirements.

The platform also includes a powerful publish/subscribe messaging system, enabling real-time communication between clients. This is particularly useful for applications that require live updates, such as chat applications or real-time notifications.

Creating a virtual machine, enter or select appropriate values for zone, machine type, resource group and so on as per your choice.

Create a virtual machine ...

[Basics](#) [Disks](#) [Networking](#) [Management](#) [Advanced](#) [Tags](#) [Review + create](#)

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ Azure subscription 1

Resource group * ⓘ Demo [Create new](#)

Instance details

Virtual machine name * ⓘ Demo ✓

Region * ⓘ (US) East US

[Review + create](#) [< Previous](#) [Next : Disks >](#)

Create a virtual machine ...

Basics Disks Networking Management Advanced Tags Review + create

Azure VMs have one operating system disk and a temporary disk for short-term storage. You can attach additional data disks. The size of the VM determines the type of storage you can use and the number of data disks allowed. [Learn more](#)

Disk options

OS disk type * ⓘ Premium SSD (locally-redundant storage) ▼

Encryption type * (Default) Encryption at-rest with a platform-managed key ▼

Enable Ultra Disk compatibility ⓘ
Ultra disk is available only for Availability Zones in eastus.

Data disks

You can add and configure additional data disks for your virtual machine or attach existing disks. This VM also comes with a temporary disk.

i Adding unmanaged data disks is currently not supported at the time of VM creation. You can add them after the VM is

[Review + create](#)

[< Previous](#)

[Next : Networking >](#)

Create a virtual machine ...

Basics Disks Networking Management Advanced Tags Review + create

Define network connectivity for your virtual machine by configuring network interface card (NIC) settings. You can control ports, inbound and outbound connectivity with security group rules, or place behind an existing load balancing solution. [Learn more](#)

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network * ⓘ (new) Demo-vnet ▼

[Create new](#)

Subnet * ⓘ (new) default (10.1.0.0/24) ▼

Public IP ⓘ (new) Demo-ip ▼

[Create new](#)

NIC network security group ⓘ None

Basic

Advanced

[Review + create](#)

[< Previous](#)

[Next : Management >](#)

Create a virtual machine ...

Basics Disks Networking Management Advanced Tags Review + create

Define network connectivity for your virtual machine by configuring network interface card (NIC) settings. You can control ports, inbound and outbound connectivity with security group rules, or place behind an existing load balancing solution. [Learn more](#)

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network *	<input type="text" value="(new) Demo-vnet"/>
	Create new
Subnet *	<input type="text" value="(new) default (10.1.0.0/24)"/>
Public IP	<input type="text" value="(new) Demo-ip"/>
	Create new
NIC network security group	<input type="radio"/> None
	<input checked="" type="radio"/> Basic
	<input type="radio"/> Advanced

[Review + create](#)

< Previous

Next : Management >

Create a virtual machine ...

Basics Disks Networking Management Advanced Tags Review + create

Configure monitoring and management options for your VM.

Azure Security Center

Azure Security Center provides unified security management and advanced threat protection across hybrid cloud workloads. [Learn more](#)

Your subscription is protected by Azure Security Center basic plan.

Monitoring

Boot diagnostics On
 Off

Enable OS guest diagnostics

Identity

[Review + create](#)

< Previous

Next : Advanced >

Create a virtual machine ...

Basics Disks Networking Management Advanced Tags Review + create

Tags are name/value pairs that enable you to categorize resources and view consolidated billing by applying the same tag to multiple resources and resource groups. [Learn more about tags](#)

Note that if you create tags and then change resource settings on other tabs, your tags will be automatically updated.

Name ⓘ	Value ⓘ	Resource
<input type="text"/>	: <input type="text"/>	12 selected 

Review + create

< Previous

Next : Review + create >

Create a virtual machine ...

✓ Validation passed

Basics Disks Networking Management Advanced Tags Review + create

PRODUCT DETAILS

Standard B1s
by Microsoft
[Terms of use](#) | [Privacy policy](#)

Subscription credits apply ⓘ
[REDACTED]
[Pricing for other VM sizes](#)

TERMS

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. See the [Azure Marketplace Terms](#) for additional details.

Create

< Previous

Next >

[Download a template for automation](#)

After Process of Create Virtual Machine. You have got an Option **Go to Resource Group** Click **Go to Resource Group**

Add the Network Security Group Inbound Rule Allow tcp Port No.: 8161

Access Redis CLI (Command line interface)

With Redis installed on Ubuntu, the next step is to test it and see whether our server works as intended. To achieve this, connect to the server using the command-line tool

```
$ redis-cli .
```

Upon running the command, your prompt will change to `<ip address>:6379`, signifying that you are now working on the Redis shell.

To test connectivity, run the `ping` command. The output `PONG` will be displayed as shown, a confirmation that Redis is functioning as expected.

```
cherry@ubuntu:~$  
cherry@ubuntu:~$ redis-cli  
127.0.0.1:6379>  
127.0.0.1:6379> ping  
PONG  
127.0.0.1:6379>  
127.0.0.1:6379>
```

Configure authentication for Redis

By default, authentication is not configured, and anyone can easily access stored data without a password. Like in most systems, security is a top priority. Thus, I highly recommend configuring a password that will require clients to authenticate themselves.

To configure authentication, open the `redis.conf` file with your preferred editor:

```
$ sudo nano /etc/redis/redis.conf
```

In the configuration file, locate and uncomment the `requirepass` directive. By default, this is set to `Admin@123`, which is merely a placeholder. Be sure to change this to a strong password. This is the password that clients will authenticate with using the `AUTH` command, as we shall see shortly.

Default Password : **Admin@123**

```
785 # IMPORTANT NOTE: starting with Redis 6 "requirepass" is just a compatibility  
786 # layer on top of the new ACL system. The option effect will be just setting  
787 # the password for the default user. Clients will still authenticate using  
788 # AUTH <password> as usually, or more explicitly with AUTH default <password>  
789 # if they follow the new protocol: both will work.  
790 #  
791 requirepass Admin@123  
792  
793 # Command renaming (DEPRECATED).  
794 #  
795 # -----
```

Note: If you want to change the Password replace the `Admin@123` with your secure password

Save the changes and exit the configuration file. For the changes to come into effect, restart the Redis service.

```
$ redis-cli
```

Next, try to perform any `SET` or `GET` operation. For instance, we will try to retrieve the value of the `city` key we previously created.

```
$ get city
```

This time, Redis will throw an error indicating that we need to authenticate. To proceed, authenticate with the `AUTH` command followed by the password you specified in the configuration file.

```
$ AUTH password
```

If the password is correct, you will get `OK` as the output.

Output

```
OK
```

Once authenticated, you can now proceed with running your queries. If you try to retrieve the value of the key, the operation will be successful this time.


```
cherry@ubuntu:~$
cherry@ubuntu:~$ redis-cli
127.0.0.1:6379>
127.0.0.1:6379>
127.0.0.1:6379> get city
(error) NOAUTH Authentication required.
127.0.0.1:6379>
127.0.0.1:6379>
127.0.0.1:6379> AUTH [REDACTED] ←
OK
127.0.0.1:6379>
127.0.0.1:6379> get city
"San Francisco"
127.0.0.1:6379>
```

To exit from the Redis shell, simply run the `exit` command.

```
$ exit
```

```
127.0.0.1:6379>
127.0.0.1:6379> exit
cherry@ubuntu:~$
cherry@ubuntu:~$
cherry@ubuntu:~$
```

Allow remote connections (Optional)

We have now installed Redis on Ubuntu, accessed Redis CLI, and configured authentication. In this last section, we will configure Redis for remote connections. It is necessary if you intend to access your Redis server remotely.

By default, Redis listens to port 6397 and is only accessible from localhost. However, you can configure it to allow remote connections from anywhere.

To allow remote connections, once again, open the default configuration file.

```
$ sudo nano /etc/redis/redis.conf
```

Set the **bind** attribute to **0.0.0.0** to allow remote connections from anywhere.

```
#  
# IF YOU ARE SURE YOU WANT YOUR INSTANCE TO LISTEN TO ALL THE INTERFACES  
# JUST COMMENT OUT THE FOLLOWING LINE.  
# ~~~~~  
bind 0.0.0.0
```

Save the changes and exit the configuration file. To effect the change, restart the Redis service.

```
$ sudo systemctl restart redis.service
```

To confirm that Redis can be accessed remotely, run the **ss** command, as shown.

```
$ sudo ss -an | grep 6379
```

From the output, **0.0.0.0:6379** shows that Redis is listening on port 6379 from all external sources or IP addresses.

```
cherry@ubuntu:~$  
cherry@ubuntu:~$  
cherry@ubuntu:~$ sudo systemctl restart redis-server  
cherry@ubuntu:~$  
cherry@ubuntu:~$  
cherry@ubuntu:~$ sudo ss -an | grep 6379  
tcp LISTEN 0      511                0.0.0.0:6379      0.0.0.0:*  
cherry@ubuntu:~$  
cherry@ubuntu:~$
```

In our case, the command will appear as follows: **<ip address>** is the IP of the Redis server host, and **6379** is the port on which the Redis service listens.

```
$ redis-cli -h 5.199.168.198 -p 6397
```

Once connected, authenticate using the `AUTH` command to start running queries.

```
client@ubuntu:~$  
client@ubuntu:~$ redis-cli -h 5.199.168.198 -p 6379   
5.199.168.198:6379>  
5.199.168.198:6379> AUTH   
OK  
5.199.168.198:6379>  
5.199.168.198:6379>  
5.199.168.198:6379> get city  
"San Francisco"  
5.199.168.198:6379>  
5.199.168.198:6379>
```

[ThankYou...](#)