Manage your account information and publish settings

Your Azure subscription information is used by the tool to connect to your account. This information can be obtained from the Azure portal in a publish settings file as described here. You can import the publish settings file as a persistent local configuration setting that the tool will use for subsequent operations. You only need to import your publish settings once.

**account download [options]**

This command launches a browser to download your .publishsettings file from the Azure portal.

~$ azure account download

info: Executing command account download

info: Launching browser to https://windows.azure.com/download/publishprofile.aspx

help: Save the downloaded file, then execute the command

help: account import <file>

info: account download command OK

**account import [options] <file>**

This command imports a publishsettings file or certificate so that it can be used by the tool going forward.

~$ azure account import publishsettings.publishsettings

info: Importing publish settings file publishsettings.publishsettings

info: Found subscription: 3-Month Free Trial

info: Found subscription: Pay-As-You-Go

info: Setting default subscription to: 3-Month Free Trial

warn: The 'publishsettings.publishsettings' file contains sensitive information.

warn: Remember to delete it now that it has been imported.

info: Account publish settings imported successfully

NOTE:

The publishsettings file can contain details (that is, subscription name and ID) about more than one subscription. When you import the publishsettings file, the first subscription is used as the default description. To use a different subscription, run the following command.~$ azure config set subscription <other-subscription-id>

**account clear [options]**

This command removes the stored publish settings that have been imported. Use this command if you're finished using the tool on this machine and want to assure that the tool cannot be used with your account going forward.

~$ azure account clear

Clearing account info.

info: OK

**account list [options]**

List the imported subscriptions

~$ azure account list

info: Executing command account list

data: Name Id

Current

data: -------------------------------------- -------------------------------

----- -------

data: Forums Subscription 8679c8be-3b05-49d9-b8fb true

data: Evangelism Team Subscription 9e672699-1055-41ae-9c36 false

data: MSOpenTech-Prod c13e6a92-706e-4cf5-94b6 false

**account set [options] <subscription>**

Set the current subscription

Commands to manage your affinity groups

**account affinity-group list [options]**

This command lists your Azure affinity groups.

Affinity groups can be set when a group of virtual machines spans multiple physical machines. The affinity group specifies that the physical machines should be as close to each other as possible, to reduce network latency.

~$ azure account affinity-group list

+ Fetching affinity groups

data: Name Label Location

data: ------------------------------------ ------ --------

data: 535EBAED-BF8B-4B18-A2E9-8755FB9D733F opentec West US

info: account affinity-group list command OK

**account affinity-group create [options] <name>**

This command creates a new affinity group

~$ azure account affinity-group create opentec -l "West US"

info: Executing command account affinity-group create

+ Creating affinity group

info: account affinity-group create command OK

**account affinity-group show [options] <name>**

This command shows the details of the affinity group

~$ azure account affinity-group show opentec

info: Executing command account affinity-group show

+ Getting affinity groups

data: $ xmlns "http://schemas.microsoft.com/windowsazure"

data: $ xmlns:i "http://www.w3.org/2001/XMLSchema-instance"

data: Name "opentec"

data: Label "b3BlbnRlYw=="

data: Description $ i:nil "true"

data: Location "West US"

data: HostedServices ""

data: StorageServices ""

data: Capabilities Capability 0 "PersistentVMRole"

data: Capabilities Capability 1 "HighMemory"

info: account affinity-group show command OK

**account affinity-group delete [options] <name>**

This command deletes the specified affinity group

~$ azure account affinity-group delete opentec

info: Executing command account affinity-group delete

Delete affinity group opentec? [y/n] y

+ Deleting affinity group

info: account affinity-group delete command OK

Commands to manage your account environment

**account env list [options]**

List of the account environments

C:\windows\system32>azure account env list

info: Executing command account env list

data: Name

data: ---------------

data: AzureCloud

data: AzureChinaCloud

info: account env list command OK

**account env show [options] [environment]**

Show account environment details

~$ azure account env show

info: Executing command account env show

Environment name: AzureCloud

data: Environment publishingProfile http://go.microsoft.com/fwlink/?LinkId=2544

data: Environment portal http://go.microsoft.com/fwlink/?LinkId=2544

info: account env show command OK

**account env add [options] [environment]**

This command adds an environment to the account

**account env set [options] [environment]**

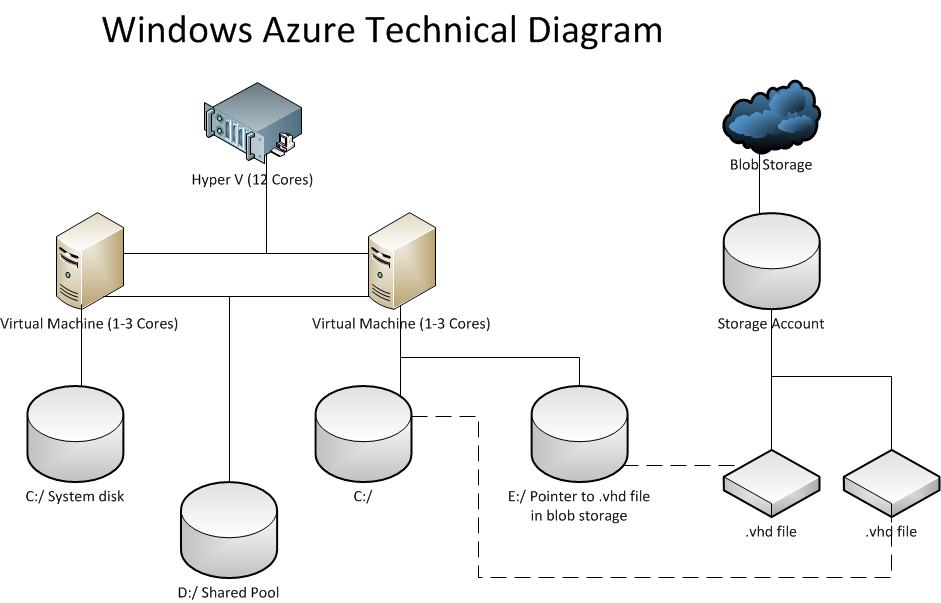
This command sets the account environment

**account env delete [options] [environment]**

This command deletes the specified environment from the account

Commands to manage your Azure virtual machines

The following diagram shows how Azure virtual machines are hosted in the production deployment environment of an Azure cloud service.



**create-new** creates the drive in blob storage (that is, e:\ in the diagram); **attach** attaches an already created but unattached disk to a virtual machine.

**vm create [options] <dns-name> <image> <userName> [password]**

This command creates a new Azure virtual machine. By default, each virtual machine (vm) is created in its own cloud service; however, you can specify that a virtual machine should be added to an existing cloud service through use of the -c option as documented here.

The vm create command, like the Azure portal, only creates virtual machines in the production deployment environment. There is no option to create a virtual machine in the staging deployment environment of a cloud service. If your subscription does not have an existing Azure storage account, the command creates one.

You can specify a location through the --location parameter, or you can specify an affinity group through the --affinity-group parameter. If neither is provided, you are prompted to provide one from a list of valid locations.

The supplied password must be 8-123 characters long and meet the password complexity requirements of the operating system that you are using for this virtual machine.

If you anticipate the need to use SSH to manage a deployed Linux virtual machine (as is usually the case), you must enable SSH via the -e option when you create the virtual machine. It is not possible enable SSH after the virtual machine has been created.

Windows virtual machines can enable RDP later by adding port 3389 as an endpoint.

The following optional parameters are supported for this command:

**-c, --connect** create the virtual machine inside an already created deployment in a hosting service. If -vmname is not used with this option, the name of the new virtual machine will be generated automatically.  
**-n, --vm-name** Specify the name of the virtual machine. This parameter takes hosting service name by default. If -vmname is not specified, the name for the new virtual machine is generated as <service-name><id>, where <id> is the number of existing virtual machines in the service plus 1 For example, if you use this command to add a new virtual machine to a hosting service MyService that has one existing virtual machine, the new virtual machine is named MyService2.  
**-u, --blob-url** Specify the target blob storage URL at which to create the virtual machine system disk.   
**-z, --vm-size** Specify the size of the virtual machine. Valid values are "extrasmall", "small", "medium", "large", "extralarge". The default value is "small".   
**-r** Adds RDP connectivity to a Windows virtual machine.   
**-e, --ssh** Adds SSH connectivity to a Windows virtual machine.   
**-t, --ssh-cert** Specifies the SSH certificate.   
**-s** The subscription   
**-o, --community** The specified image is a community image   
**-w** The virtual network name   
**-l, --location** specifies the location (for example, "North Central US").   
**-a, --affinity-group** specifies the affinity group.  
**-w, --virtual-network-name** Specify the virtual network on which to add the new virtual machine. Virtual networks can be set up and managed from the Azure portal.  
**-b, --subnet-names** Specifies the subnet names to assign the virtual machine.

In this example, MSFT\_\_Win2K8R2SP1-120514-1520-141205-01-en-us-30GB is an image provided by the platform. For more information about operating system images, see vm image list.

~$ azure vm create my-vm-name MSFT\_\_Windows-Server-2008-R2-SP1.11-29-2011 username --location "Western US" -r

info: Executing command vm create

Enter VM 'my-vm-name' password: \*\*\*\*\*\*\*\*\*\*\*\*

info: vm create command OK

**vm create-from <dns-name> <role-file>**

This command creates a new Azure virtual machine from a JSON role file.

~$ azure vm create-from my-vm example.json

info: OK

**vm list [options]**

This command lists Azure virtual machines. The -json option specifies that the results are returned in raw JSON format.

~$ azure vm list

info: Executing command vm list

data: DNS Name VM Name Status

data: -------------------------------- ----------- ---------

data: my-vm-name.cloudapp-preview.net my-vm ReadyRole

info: vm list command OK

**vm location list [options]**

This command lists all available Azure account locations.

~$ azure vm location list

info: Executing command vm location list

data: Name Display Name

data: --------------------- ------------

data: Azure Preview West US

info: account location list command OK

**vm show [options] <name>**

This command shows details about an Azure virtual machine. The --json option specifies that the results are returned in raw JSON format.

~$ azure vm show my-vm

info: Executing command vm show

data: {

data: InstanceSize: 'Small',

data: InstanceStatus: 'ReadyRole',

data: DataDisks: [],

data: IPAddress: '10.26.192.206',

data: DNSName: 'my-vm.cloudapp.net',

data: InstanceStateDetails: {},

data: VMName: 'my-vm',

data: Network: {

data: Endpoints: [

data: {

data: Protocol: 'tcp',

data: Vip: '65.52.250.250',

data: Port: '63238' ,

data: LocalPort: '3389',

data: Name: 'RemoteDesktop'

data: }

data: ]

data: },

data: Image: 'MSFT\_\_Windows-Server-2008-R2-SP1.11-29-2011',

data: OSVersion: 'WA-GUEST-OS-1.18\_201203-01'

data: }

info: vm show command OK

**vm delete [options] <name>**

This command deletes an Azure virtual machine. By default, this command does not delete the Azure blob from which the operating system disk and the data disk are created. To delete the blob as well as the virtual machine on which it is based, specify the -b option.

~$ azure vm delete my-vm

info: Executing command vm delete

info: vm delete command OK

**vm start [options] <name>**

This command starts an Azure virtual machine.

~$ azure vm start my-vm

info: Executing command vm start

info: vm start command OK

**vm restart [options] <name>**

This command restarts an Azure virtual machine.

~$ azure vm restart my-vm

info: Executing command vm restart

info: vm restart command OK

**vm shutdown [options] <name>**

This command shuts down an Azure virtual machine. You may use the -p option to specify that the compute resource not be released on shutdown.

~$ azure vm shutdown my-vm

info: Executing command vm shutdown

info: vm shutdown command OK

**vm capture <vm-name> <target-image-name>**

This command captures an Azure virtual machine image.

A virtual machine image cannot be captured while the virtual machine state unless the virtual machine state is **Stopped**.

~$ azure.cmd vm capture my-vm mycaptureimagename --delete

info: Executing command vm capture

+ Fetching VMs

+ Capturing VM

info: vm capture command OK

**vm export [options] <vm-name> <file-path>**

This command exports an Azure virtual machine image to a file

~$ azure vm export "myvm" "C:\"

info: Executing command vm export

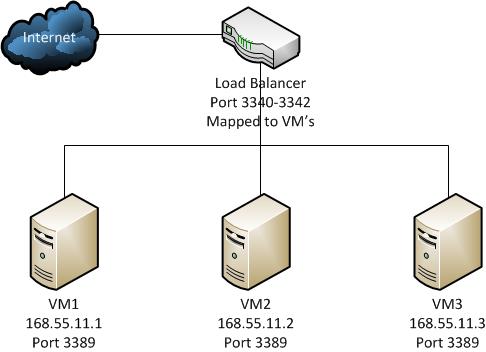
+ Getting virtual machines

+ Exporting the VM

info: vm export command OK

Commands to manage your Azure virtual machine endpoints

The following diagram shows the architecture of a typical deployment of multiple instances of a virtual machine. Note that in this example port 3389 is open on each virtual machine (for RDP access), and there is also an internal IP address (for example, 168.55.11.1) on each virtual machine that is used by the load balancer to route traffic to the virtual machine. This internal IP address can also be used for communication between virtual machines.



External requests to virtual machines go through a load balancer. Because of this, requests cannot be specified against a particular virtual machine on deployments with multiple virtual machines. For deployments with multiple virtual machines, port mapping must be configured between the virtual machines (vm-port) and the load balancer (lb-port).

**vm endpoint create <vm-name> <lb-port> [vm-port]**

This command creates a virtual machine endpoint. You may also use -u or --enable-direct-server-return to specify whether to enable direct server return on this endpoint, disabled by default.

~$ azure vm endpoint create my-vm 8888 8888

azure vm endpoint create my-vm 8888 8888

info: Executing command vm endpoint create

+ Fetching VM

+ Reading network configuration

+ Updating network configuration

info: vm endpoint create command OK

**vm endpoint create-multiple [options] <vm-name> <lb-port>[:<vm-port>[:<protocol>[:<lb-set-name>[:<prob-protocol>:<lb-prob-port>[:<prob-path>]]]]] ]{1-\*}**

Create multiple vm endpoints. You may also use -u or --enable-direct-server-return to specify whether to enable direct server return on this endpoint, disabled by default.

**vm endpoint delete <vm-name> <lb-port>**

This command deletes a virtual machine endpoint.

~$ azure vm endpoint delete my-vm 8888

azure vm endpoint delete my-vm 8888

info: Executing command vm endpoint delete

+ Fetching VM

+ Reading network configuration

+ Updating network configuration

info: vm endpoint delete command OK

**vm endpoint list <vm-name>**

This command lists all virtual machine endpoints. The -json option specifies that the results are returned in raw JSON format.

~$ azure vm endpoint list my-linux-vm

data: Name External Port Local Port

data: ---- ------------- ----------

data: ssh 22 22

**vm endpoint update [options] <vm-name> <endpoint-name>**

This command updates a vm endpoint to new values using these options.

-n, --endpoint-name <name> the new endpoint name

-lo, --lb-port <port> the new load balancer port

-t, --vm-port <port> the new local port

-o, --endpoint-protocol <protocol> the new transport layer protocol for port (tcp or udp)

**vm endpoint show [options] <vm-name>**

This command shows the details of the endpoints on a vm

~$ azure vm endpoint show "mycouchvm"

info: Executing command vm endpoint show

+ Getting virtual machines

data: Network Endpoints 0 LoadBalancedEndpointSetName "CouchDB\_EP-5984"

data: Network Endpoints 0 LocalPort "5984"

data: Network Endpoints 0 Name "CouchDB\_EP"

data: Network Endpoints 0 Port "5984"

data: Network Endpoints 0 Protocol "tcp"

data: Network Endpoints 0 Vip "168.61.9.97"

data: Network Endpoints 1 LoadBalancedEndpointSetName "CouchEP\_2-2020"

data: Network Endpoints 1 LocalPort "2020"

data: Network Endpoints 1 Name "CouchEP\_2"

data: Network Endpoints 1 Port "2020"

data: Network Endpoints 1 Protocol "tcp"

data: Network Endpoints 1 Vip "168.61.9.97"

data: Network Endpoints 2 LocalPort "3389"

data: Network Endpoints 2 Name "RemoteDesktop"

data: Network Endpoints 2 Port "3389"

data: Network Endpoints 2 Protocol "tcp"

data: Network Endpoints 2 Vip "168.61.9.97"

info: vm endpoint show command OK

Commands to manage your Azure virtual machine images

Virtual machine images are captures of already configured virtual machines that can be replicated as required.

**vm image list [options]**

This command gets a list of virtual machine images. There are three types of images: images created by Microsoft, which are prefixed with "MSFT", images created by third parties, which are usually prefixed with the name of the vendor, and images you create. To create images, you can either capture an existing virtual machine or create an image from a custom .vhd uploaded to blob storage. For more information about using a custom .vhd, see vm image create. The -json option specifies that the results are returned in raw JSON format.

~$ azure vm image list

data: Name Category OS

data: --------------------------------------------------------------------- --------- -------

data: CANONICAL\_\_Canonical-Ubuntu-12-04-20120519-2012-05-19-en-us-30GB.vhd Canonical Linux

data: MSFT\_\_Windows-Server-2008-R2-SP1.11-29-2011 Microsoft Windows

data: MSFT\_\_Windows-Server-2008-R2-SP1-with-SQL-Server-2012-Eval.11-29-2011 Microsoft Windows

data: MSFT\_\_Windows-Server-8-Beta.en-us.30GB.2012-03-22 Microsoft Windows

data: MSFT\_\_Windows-Server-8-Beta.2-17-2012 Microsoft Windows

data: MSFT\_\_Windows-Server-2008-R2-SP1.en-us.30GB.2012-3-22 Microsoft Windows

data: OpenLogic\_\_OpenLogic-CentOS-62-20120509-en-us-30GB.vhd OpenLogic Linux

data: SUSE\_\_SUSE-Linux-Enterprise-Server-11SP2-20120521-en-us-30GB.vhd SUSE Linux

data: SUSE\_\_OpenSUSE64121-03192012-en-us-15GB.vhd SUSE Linux

data: WIN2K8-R2-WINRM User Windows

info: vm image list command OK

**vm image show [options] <name>**

This command shows the details of a virtual machine image.

~$ azure vm image show MSFT\_\_Windows-Server-2008-R2-SP1.11-29-2011

+ Fetching VM image

info: Executing command vm image show

data: {

data: Label: 'Windows Server 2008 R2 SP1, Nov 2011',

data: Name: 'MSFT\_\_Windows-Server-2008-R2-SP1.11-29-2011',

data: Description: 'Microsoft Windows Server 2008 R2 SP1',

data: @: { xmlns: 'http://schemas.microsoft.com/windowsazure', xmlns:i: 'http://www.w3.org/2001/XMLSchema-instance' },

data: Category: 'Microsoft',

data: OS: 'Windows',

data: Eula: 'http://www.microsoft.com',

data: LogicalSizeInGB: '30'

data: }

info: vm image show command OK

**vm image delete [options] <name>**

This command deletes a virtual machine image.

~$ azure vm image delete my-vm-image

info: Executing command vm image delete

info: VM image deleted: my-vm-image

info: vm image delete command OK

**vm image create <name> [source-path]**

This command creates a virtual machine image. Your custom .vhd files are uploaded to blob storage, and then the virtual machine image is created from there. You then use this virtual machine image to create a virtual machine. The Location and OS parameters are required.

Some systems impose per-process file descriptor limits. If this limit is exceeded, the tool displays a file descriptor limit error. You can run the command again using the -p <number> parameter to reduce the maximum number of parallel uploads. The default maximum number of parallel uploads is 96.

~$ azure vm image create mytestimage ./Sample.vhd -o windows -l "West US"

info: Executing command vm image create

+ Retrieving storage accounts

info: VHD size : 13 MB

info: Uploading 13312.5 KB

Requested:100.0% Completed:100.0% Running: 105 Time: 8s Speed: 1721 KB/s

info: http://myaccount.blob.core.azure.com/vm-images/Sample.vhd is uploaded successfully

info: vm image create command OK

Commands to manage your Azure virtual machine data disks

Data disks are .vhd files in blob storage that can be used by a virtual machine. For more information about how data disks are deployed to blob storage, see the Azure technical diagram shown earlier.

The commands for attaching data disks (azure vm disk attach and azure vm disk attach-new) assign a Logical Unit Number (LUN) to the attached data disk, as required by the SCSI protocol. The first data disk attached to a virtual machine is assigned LUN 0, the next is assigned LUN 1, and so on.

When you detach a data disk with the azure vm disk detach command, use the <lun> parameter to indicate which disk to detach.

*[AZURE>NOTE] Note that you should always detach data disks in reverse order, starting with the highest-numbered LUN that has been assigned. The Linux SCSI layer does not support detaching a lower-numbered LUN while a higher-numbered LUN is still attached. For example, you should not detach LUN 0 if LUN 1 is still attached.*

**vm disk show [options] <name>**

This command shows details about an Azure disk.

~$ azure vm disk show anucentos-anucentos-0-20120524070008

info: Executing command vm disk show

data: AttachedTo DeploymentName "mycentos"

data: AttachedTo HostedServiceName "myanucentos"

data: AttachedTo RoleName "myanucentos"

data: OS "Linux"

data: Location "Azure Preview"

data: LogicalDiskSizeInGB "30"

data: MediaLink "http://mystorageaccount.blob.core.azure-preview.com/vhd-store/mycentos-cb39b8223b01f95c.vhd"

data: Name "mycentos-mycentos-0-20120524070008"

data: SourceImageName "OpenLogic\_\_OpenLogic-CentOS-62-20120509-en-us-30GB.vhd"

info: vm disk show command OK

**vm disk list [options] [vm-name]**

This command lists Azure disks, or disks attached to a specified virtual machine. If it is run with a virtual machine name parameter, it returns all disks attached to the virtual machine. Lun 1 is created with the virtual machine, and any other listed disks are attached separately.

~$ azure vm disk list mycentos

info: Executing command vm disk list

data: Lun Size(GB) Blob-Name

data: --- -------- --------------------------------

data: 1 30 mycentos-cb39b8223b01f95c.vhd

data: 2 10 mycentos-e3f0d717950bb78d.vhd

info: vm disk list command OK

Executing this command without a virtual machine name parameter returns all disks.

~$ azure vm disk list

data: Name OS

data: ------------------------------------------ -------

data: mycentos-mycentos-0-20120524070008 Linux

data: mycentos-mycentos-2-20120525055052

data: mywindows-winvm-20120522223119 Windows

info: vm disk list command OK

**vm disk delete [options] <name>**

This command deletes an Azure disk from a personal repository. The disk must be detached from the virtual machine before it is deleted.

~$ azure vm disk delete mycentos-mycentos-2-20120525055052

info: Executing command vm disk delete

info: Disk deleted: mycentos-mycentos-2-20120525055052

info: vm disk delete command OK

**vm disk create <name> [source-path]**

This command uploads and registers an Azure disk. --blob-url, --location, or --affinity-group must be specified. If you use this command with [source-path], the .vhd file specified is uploaded and a new image is created. You can then attach this image to a virtual machine by using vm disk attach.

Some systems impose per-process file descriptor limits. If this limit is exceeded, the tool displays a file descriptor limit error. You can run the command again using the -p <number> parameter to reduce the maximum number of parallel uploads. The default maximum number of parallel uploads is 96.

~$ azure vm disk create my-data-disk ~/test.vhd --location "Western US"

info: Executing command vm disk create

info: VHD size : 10 MB

info: Uploading 10240.5 KB

Requested:100.0% Completed:100.0% Running: 81 Time: 11s Speed: 952 KB/s

info: http://account.blob.core.azure.com/disks/test.vhd is uploaded successfully

info: vm disk create command OK

**vm disk upload [options] <source-path> <blob-url> <storage-account-key>**

This command allows you to upload a vm disk

~$ azure vm disk upload "http://sourcestorage.blob.core.windows.net/vhds/sample.vhd" "http://destinationstorage.blob.core.windows.net/vhds/sample.vhd" "DESTINATIONSTORAGEACCOUNTKEY"

info: Executing command vm disk upload

info: Uploading 12351.5 KB

info: vm disk upload command OK

**vm disk attach <vm-name> <disk-image-name>**

This command attaches an existing disk in blob storage to an existing virtual machine deployed in a cloud service.

~$ azure vm disk attach my-vm my-vm-my-vm-2-201242418259

info: Executing command vm disk attach

info: vm disk attach command OK

**vm disk attach-new <vm-name> <size-in-gb> [blob-url]**

This command attaches a data disk to an Azure virtual machine. In this example, 20 is the size of the new disk, in gigabytes, to be attached. You can optionally use a blob URL as the last argument to explicitly specify the target blob to create. If you do not specify a blob URL, a blob object will be automatically generated.

~$ azure vm disk attach-new nick-test36 20 http://nghinazz.blob.core.azure-preview.com/vhds/vmdisk1.vhd

info: Executing command vm disk attach-new

info: vm disk attach-new command OK

**vm disk detach <vm-name> <lun>**

This command detaches a data disk attached to an Azure virtual machine. <lun> identifies the disk to be detached. To get a list of disks associated with a disk before you detach it, use vm disk-list <vm-name>.

~$ azure vm disk detach my-vm 2

info: Executing command vm disk detach

info: vm disk detach command OK

Commands to manage your Azure cloud services

Azure cloud services are applications and services hosted on web roles and worker roles. The following commands can be used to manage Azure cloud services.

**service create [options] <serviceName>**

This command creates a new cloud service

~$ azure service create newservicemsopentech

info: Executing command service create

+ Getting locations

help: Location:

1) East Asia

2) Southeast Asia

3) North Europe

4) West Europe

5) East US

6) West US

: 6

+ Creating cloud service

data: Cloud service name newservicemsopentech

info: service create command OK

**service show [options] <serviceName>**

This command shows the details of an Azure cloud service

~$ azure service show newservicemsopentech

info: Executing command service show

+ Getting cloud service

data: Name newservicemsopentech

data: Url https://management.core.windows.net/9e672699-1055-41ae-9c36-e85152f2e352/services/hostedservices/newservicemsopentech

data: Properties location West US

data: Properties label newservicemsopentech

data: Properties status Created

data: Properties dateCreated

data: Properties dateLastModified

info: service show command OK

**service list [options]**

This command lists Azure cloud services.

~$ azure service list

info: Executing command service list

data: Name Status

data: ----------- -------

data: service1 Created

data: service2 Created

info: service list command OK

**service delete [options] <name>**

This command deletes an Azure cloud service.

~$ azure service delete myservice

info: Executing command service delete myservice

info: cloud-service delete command OK

To force the deletion, use the -q parameter.

Commands to manage your Azure certificates

Azure service certificates are SSL certificates connected to your Azure account. For more information about Azure certificates, see [Manage Certificates](http://msdn.microsoft.com/en-us/library/azure/gg981929.aspx).

**service cert list [options]**

This command lists Azure certificates.

~$ azure service cert list

info: Executing command service cert list

+ Fetching cloud services

+ Fetching certificates

data: Service Thumbprint Algorithm

data: -------- ---------------------------------------- ---------

data: myservice 262DBF95B5E61375FA27F1E74AC7D9EAE842916C sha1

info: service cert list command OK

**service cert create <dns-prefix> <file> [password]**

This command uploads a certificate. Leave the password prompt blank for certificates that are not password protected.

~$ azure service cert create nghinazz ~/publishSet.pfx

info: Executing command service cert create

Cert password:

+ Creating certificate

info: service cert create command OK

**service cert delete [options] <thumbprint>**

This command deletes a certificate.

~$ azure service cert delete 262DBF95B5E61375FA27F1E74AC7D9EAE842916C

info: Executing command service cert delete

+ Deleting certificate

info: nghinazz : cert deleted

info: service cert delete command OK

Commands to manage your websites

An Azure website is a web configuration accessible by URI. Websites are hosted in virtual machines, but you do not need to think about the details of creating and deploying the virtual machine yourself. Those details are handled for you by Azure.

**site list [options]**

This command lists your websites.

~$ azure site list

info: Executing command site list

data: Name State Host names

data: -------------- ------- --------------------------------------------------

data: mongosite Running mongosite.antdf0.antares.windows.net

data: myphpsite Running myphpsite.antdf0.antares.windows.net

data: mydrupalsite36 Running mydrupalsite36.antdf0.antares.windows.net

info: site list command OK

**site set [options] [name]**

This command will set configuration options for your website [name]

~$ azure site set

info: Executing command site set

Web site name: mydemosite

+ Getting sites

+ Updating site config information

info: site set command OK

**site deploymentscript [options]**

This command will generate a custom deployment script

~$ azure site deploymentscript --node

info: Executing command site deploymentscript

info: Generating deployment script for node.js Web Site

info: Generated deployment script files

info: site deploymentscript command OK

**site create [options] [name]**

This command creates a new website and local directory.

~$ azure site create mysite

info: Executing command site create

info: Using location northeuropewebspace

info: Creating a new web site

info: Created web site at mysite.antdf0.antares.windows.net

info: Initializing repository

info: Repository initialized

info: site create command OK

NOTE:

The site name must be unique. You cannot create a site with the same DNS name as an existing site.

**site browse [options] [name]**

This command opens your website in a browser.

~$ azure site browse mysite

info: Executing command site browse

info: Launching browser to http://mysite.antdf0.antares-test.windows-int.net

info: site browse command OK

**site show [options] [name]**

This command shows details for a website.

~$ azure site show mysite

info: Executing command site show

info: Showing details for site

data: Site AdminEnabled true

data: Site HostNames mysite.antdf0.antares-test.windows-int.net

data: Site Name mysite

data: Site Owner 00060000814EDDEE

data: Site RepositorySiteName mysite

data: Site SelfLink https://s1.api.antdf0.antares.windows.net:454/subscriptions/444e62ff-4c5f-4116-a695-5c803ed584a5/webspaces/northeuropewebspace/sites/mysite

data: Site State Running

data: Site UsageState Normal

data: Site WebSpace northeuropewebspace

data: Config AppSettings

data: Config ConnectionStrings

data: Config DefaultDocuments 0=Default.htm, 1=Default.asp, 2=index.htm, 3=index.html, 4=iisstart.htm, 5=default.aspx, 6=index.php, 7=hostingstart.aspx

data: Config DetailedErrorLoggingEnabled false

data: Config HttpLoggingEnabled false

data: Config Metadata

data: Config NetFrameworkVersion v4.0

data: Config NumberOfWorkers 1

data: Config PhpVersion 5.3

data: Config PublishingPassword rJ}[Er2v[Y]q16B6vTD]n$[C2z}Z.pvgLfRcLnAp%ax]xstiLny};o@vmMAote@d

data: Config RequestTracingEnabled false

data: Repository https://mysite.scm.antdf0.antares-test.windows-int.net/

info: site show command OK

**site delete [options] [name]**

This command deletes a website.

~$ azure site delete mysite

info: Executing command site delete

info: Deleting site mysite

info: Site mysite has been deleted

info: site delete command OK

**site swap [options] [name]**

This command swaps two website slots.

This command supports the following additional option:

**-q or \*\*--quiet**: Do not prompt for confirmation. Use this option in automated scripts.

**site start [options] [name]**

This command starts a website.

~$ azure site start mysite

info: Executing command site start

info: Starting site mysite

info: Site mysite has been started

info: site start command OK

**site stop [options] [name]**

This command stops a website.

~$ azure site stop mysite

info: Executing command site stop

info: Stopping site mysite

info: Site mysite has been stopped

info: site stop command OK

\*\*site restart [options] [name]

This command stops and then starts a specified website.

This command supports the following additional option:

**--slot** <slot>: The name of the slot to restart.

**site location list [options]**

This command lists your Website locations.

~$ azure site location list

info: Executing command site location list

+ Getting locations

data: Name

data: ----------------

data: West Europe

data: West US

data: North Central US

data: North Europe

data: East Asia

data: East US

info: site location list command OK

Commands to manage your Website application settings

**site appsetting list [options] [name]**

This command lists the app setting added to the website.

~$ azure site appsetting list

info: Executing command site appsetting list

Web site name: mydemosite

+ Getting sites

+ Getting site config information

data: Name Value

data: ---- -----

data: test value

info: site appsetting list command OK

**site appsetting add [options] <keyvaluepair> [name]**

This command adds an app setting to your website as a key value pair.

~$ azure site appsetting add test=value

info: Executing command site appsetting add

Web site name: mydemosite

+ Getting sites

+ Getting site config information

+ Updating site config information

info: site appsetting add command OK

**site appsetting delete [options] <key> [name]**

This command deletes the specified app setting from the website.

~$ azure site appsetting delete test

info: Executing command site appsetting delete

Web site name: mydemosite

+ Getting sites

+ Getting site config information

Delete application setting test? [y/n] y

+ Updating site config information

info: site appsetting delete command OK

**site appsetting show [options] <key> [name]**

This command displays details of the specified app setting

~$ azure site appsetting show test

info: Executing command site appsetting show

Web site name: mydemosite

+ Getting sites

+ Getting site config information

data: Value: value

info: site appsetting show command OK

Commands to manage your Website certificates

**site cert list [options] [name]**

This command displays a list of the website certs.

~$ azure site cert list

info: Executing command site cert list

Web site name: mydemosite

+ Getting sites

+ Getting site information

data: Subject Expiration Date Thumbprint

data: ---------------------------- -----------------------------------------

---------------- ----------------------------------------

data: \*.msopentech.com Fri Nov 28 2014 09:49:57 GMT-0800 (Pacific Standard Time) A40E82D3DC0286D1F58650E570ECF8224F69A148

data: msopentech.azurewebsites.net Fri Jun 19 2015 11:57:32 GMT-0700 (Pacific Daylight Time) CE1CD6538852BF7A5DC32001C2E26A29B541F0E8

info: site cert list command OK

**site cert add [options] <certificate-path> [name]**

**site cert delete [options] <thumbprint> [name]**

**site cert show [options] <thumbprint> [name]**

This command shows the cert details

~$ azure site cert show CE1CD65852B38DC32001C2E0E8F7A526A29B541F

info: Executing command site cert show

Web site name: mydemosite

+ Getting sites

+ Getting site information

data: Certificate hostNames 0=msopentech.azurewebsites.net

data: Certificate expirationDate

data: Certificate friendlyName msopentech.azurewebsites.net

data: Certificate issueDate

data: Certificate issuer CN=MSIT Machine Auth CA 2, DC=redmond, DC=corp, DC=microsoft, DC=com

data: Certificate subjectName msopentech.azurewebsites.net

data: Certificate thumbprint CE1CD65852B38DC32001C2E0E8F7A526A29B541F

info: site cert show command OK

Commands to manage your Website connection strings

**site connectionstring list [options] [name]**

**site connectionstring add [options] <connectionname> <value> <type> [name]**

**site connectionstring delete [options] <connectionname> [name]**

**site connectionstring show [options] <connectionname> [name]**

Commands to manage your Website default documents

**site defaultdocument list [options] [name]**

**site defaultdocument add [options] <document> [name]**

**site defaultdocument delete [options] <document> [name]**

Commands to manage your Website deployments

**site deployment list [options] [name]**

**site deployment show [options] <commitId> [name]**

**site deployment redeploy [options] <commitId> [name]**

**site deployment github [options] [name]**

**site deployment user set [options] [username] [pass]**

Commands to manage your Website domains

**site domain list [options] [name]**

**site domain add [options] <dn> [name]**

**site domain delete [options] <dn> [name]**

Commands to manage your Website handler mappings

**site handler list [options] [name]**

**site handler add [options] <extension> <processor> [name]**

**site handler delete [options] <extension> [name]**

Commands to manage your Website Web Jobs

**site job list [options] [name]**

This command list all the web jobs under a website.

This command supports the following additional options:

* **--job-type** <job-type>: Optional. The type of the webjob. Valid value is "triggered" or "continuous". By default return webjobs of all types.
* **--slot** <slot>: The name of the slot to restart.

**site job show [options] <jobName> <jobType> [name]**

This command shows the details of a specific web job.

This command supports the following additional options:

* **--job-name** <job-name>: Required. The name of the webjob.
* **--job-type** <job-type>: Required. The type of the webjob. Valid value is "triggered" or "continuous".
* **--slot** <slot>: The name of the slot to restart.

**site job delete [options] <jobName> <jobType> [name]**

This command deletes the specified web job.

This command supports the following additional options:

* **--job-name** <job-name> required. The name of the webjob.
* **--job-type** <job-type> required. The type of the webjob. Valid value is "triggered" or "continuous".
* **-q** or **--quiet**: Do not prompt for confirmation. Use this option in automated scripts.
* **--slot** <slot>: The name of the slot to restart.

**site job upload [options] <jobName> <jobType> [name]**

This command deletes the specified web job.

This command supports the following additional options:

* **--job-name** <job-name>: Required. The name of the webjob.
* **--job-type** <job-type>: Required. The type of the webjob. Valid value is "triggered" or "continuous".
* **--job-file** <job-file>: Required. The job file.
* **--slot** <slot>: The name of the slot to restart.

**site job start [options] <jobName> <jobType> [name]**

This command starts the specified web job.

This command supports the following additional options:

* **--job-name** <job-name>: Required. The name of the webjob.
* **--job-type** <job-type>: Required. The type of the webjob. Valid value is "triggered" or "continuous".
* **--slot** <slot>: The name of the slot to restart.

**site job stop [options] <jobName> <jobType> [name]**

This command stops the specified web job. Only continuous jobs can be stopped.

This command supports the following additional options:

* **--job-name** <job-name>: Required. The name of the webjob.
* **--slot** <slot>: The name of the slot to restart.

Commands to manage your Website Web Jobs History

**site job history list [options] [jobName] [name]**

This command displays a history of the runs of the specified web job.

This command supports the following additional options:

* **--job-name** <job-name>: Required. The name of the webjob.
* **--slot** <slot>: The name of the slot to restart.

**site job history show [options] [jobName] [runId] [name]**

This command gets the details of the job run for the specified web job.

This command supports the following additional options:

* **--job-name** <job-name>: Required. The name of the webjob.
* **--run-id** <run-id>: Optional. The id of the run history. If not specified, show the latest run.
* **--slot** <slot>: The name of the slot to restart.

Commands to manage your Website diagnostics

**site log download [options] [name]**

Download a .zip file that contains your website's diagnostics.

~$ azure site log download

info: Executing command site log download

Web site name: mydemosite

+ Getting sites

+ Getting site information

+ Downloading diagnostic log to diagnostics.zip

info: site log download command OK

**site log tail [options] [name]**

This command connects your terminal to the log-streaming service.

~$ azure site log tail

info: Executing command site log tail

Web site name: mydemosite

+ Getting sites

+ Getting site information

2013-11-19T17:24:17 Welcome, you are now connected to log-streaming service.

**site log set [options] [name]**

This command configures the diagnostic options for your website.

~$ azure site log set -a

info: Executing command site log set

+ Getting output options

help: Output:

1) file

2) storage

: 1

Web site name: mydemosite

+ Getting locations

+ Getting sites

+ Getting site information

+ Getting diagnostic settings

+ Updating diagnostic settings

info: site log set command OK

Commands to manage your Website repositories

**site repository branch [options] <branch> [name]**

**site repository delete [options] [name]**

**site repository sync [options] [name]**

Commands to manage your Website scaling

**site scale mode [options] <mode> [name]**

**site scale instances [options] <instances> [name]**

Commands to manage Azure Mobile Services

Azure Mobile Services brings together a set of Azure services that enable backend capabilities for your apps. Mobile Services commands are divided into the following categories:

* [Commands to manage mobile service instances](http://azure.microsoft.com/en-us/documentation/articles/command-line-tools/#Mobile_Services)
* [Commands to manage mobile service configuration](http://azure.microsoft.com/en-us/documentation/articles/command-line-tools/#Mobile_Configuration)
* [Commands to manage mobile service tables](http://azure.microsoft.com/en-us/documentation/articles/command-line-tools/#Mobile_Tables)
* [Commands to manage mobile service scripts](http://azure.microsoft.com/en-us/documentation/articles/command-line-tools/#Mobile_Scripts)
* [Commands to manage scheduled jobs](http://azure.microsoft.com/en-us/documentation/articles/command-line-tools/#Mobile_Jobs)
* [Commands to scale a mobile service](http://azure.microsoft.com/en-us/documentation/articles/command-line-tools/#Mobile_Scale)

The following options apply to most Mobile Services commands:

* **-h** or **--help**: Display output usage information.
* **-s**<id> or **--subscription**<id>: Use a specific subscription, specified as <id>.
* **-v** or **--verbose**: Write verbose output.
* **--json**: Write JSON output.

Commands to manage mobile service instances

**mobile locations [options]**

This command lists geographic locations supported by Mobile Services.

~$ azure mobile locations

info: Executing command mobile locations

info: East US (default)

info: West US

info: North Europe

**mobile create [options] [servicename] [sqlAdminUsername] [sqlAdminPassword]**

This command creates a mobile service along with a SQL Database and server.

~$ azure mobile create todolist your\_login\_name Secure$Password

info: Executing command mobile create

+ Creating mobile service

info: Overall application state: Healthy

info: Mobile service (todolist) state: ProvisionConfigured

info: SQL database (todolist\_db) state: Provisioned

info: SQL server (e96ean1c6v) state: ProvisionConfigured

info: mobile create command OK

This command supports the following additional options:

* **-r**<sqlServer> or **--sqlServer**<sqlServer>: Use an existing SQL Database server, specified as <sqlServer>.
* **-d**<sqlDb> or **--sqlDb**<sqlDb>: Use existing SQL database, specified as <sqlDb>.
* **-l**<location> or **--location**<location>: Create the service in a specific location, specified as <location>. Run azure mobile locations to get available locations.
* **--sqlLocation**<location>: Create the SQL server in a specific <location>; defaults to the location of the mobile service.

**mobile delete [options] [servicename]**

This command deletes a mobile service along with its SQL Database and server.

~$ azure mobile delete todolist -a -q

info: Executing command mobile delete

data: Mobile service todolist

data: SQL database todolistAwrhcL60azo1C401

data: SQL server fh1kvbc7la

+ Deleting mobile service

info: Deleted mobile service

+ Deleting SQL server

info: Deleted SQL server

+ Deleting mobile application

info: Deleted mobile application

info: mobile delete command OK

This command supports the following additional options:

* **-d** or **--deleteData**: Delete all data from this mobile service from the database.
* **-a** or **--deleteAll**: Delete the SQL Database and server.
* **-q** or **--quiet**: Do not prompt for confirmation. Use this option in automated scripts.

**mobile list [options]**

This command lists your mobile services.

~$ azure mobile list

info: Executing command mobile list

data: Name State URL

data: ------------ ----- --------------------------------------

data: todolist Ready https://todolist.azure-mobile.net/

data: mymobileapp Ready https://mymobileapp.azure-mobile.net/

info: mobile list command OK

**mobile show [options] [servicename]**

This command displays details about a mobile service.

~$ azure mobile show todolist

info: Executing command mobile show

+ Getting information

info: Mobile application

data: status Healthy

data: Mobile service name todolist

data: Mobile service status ProvisionConfigured

data: SQL database name todolistAwrhcL60azo1C401

data: SQL database status Linked

data: SQL server name fh1kvbc7la

data: SQL server status Linked

info: Mobile service

data: name todolist

data: state Ready

data: applicationUrl https://todolist.azure-mobile.net/

data: applicationKey XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

data: masterKey XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

data: webspace WESTUSWEBSPACE

data: region West US

data: tables TodoItem

info: mobile show command OK

**mobile restart [options] [servicename]**

This command restarts a mobile service instance.

~$ azure mobile restart todolist

info: Executing command mobile restart

+ Restarting mobile service

info: Service was restarted.

info: mobile restart command OK

**mobile log [options] [servicename]**

This command returns mobile service logs, filtering out all log types but error.

~$ azure mobile log todolist -t error

info: Executing command mobile log

data:

data: timeCreated 2013-01-07T16:04:43.351Z

data: type error

data: source /scheduler/TestingLogs.js

data: message This is an error.

data:

info: mobile log command OK

This command supports the following additional options:

* **-r**<query> or **--query**<query>: Executes the specified log query.
* **-t**<type> or **--type**<type>: Filter the returned logs by entry <type>, which can be information, warning, or error.
* **-k**<skip> or **--skip**<skip>: Skips the number of rows specified by <skip>.
* **-p**<top> or **--top**<top>: Returns a specific number of rows, specified by <top>.

NOTE:

The **--query** parameter takes precedence over **--type**, **--skip**, and **--top**.

**mobile recover [options] [unhealthyservicename] [healthyservicename]**

This command recovers an unhealthy mobile service by moving it to a healthy mobile service in a different region.

This command supports the following additional option:

**-q** or **--quiet**: Suppress the prompt for confirmation of recovery.

**mobile key regenerate [options] [servicename] [type]**

This command regenerates the mobile service application key.

~$ azure mobile key regenerate todolist application

info: Executing command mobile key regenerate

info: New application key is SmLorAWVfslMcOKWSsuJvuzdJkfUpt40

info: mobile key regenerate command OK

Key types are master and application.

NOTE:

When you regenerate keys, clients that use the old key may be unable to access your mobile service. When you regenerate the application key, you should update your app with the new key value.

**mobile key set [options] [servicename] [type] [value]**

This command sets the mobile service key to a specific value.

Commands to manage mobile service configuration

**mobile config list [options] [servicename]**

This command lists configuration options for a mobile service.

~$ azure mobile config list todolist

info: Executing command mobile config list

+ Getting mobile service configuration

data: dynamicSchemaEnabled true

data: microsoftAccountClientSecret Not configured

data: microsoftAccountClientId Not configured

data: microsoftAccountPackageSID Not configured

data: facebookClientId Not configured

data: facebookClientSecret Not configured

data: twitterClientId Not configured

data: twitterClientSecret Not configured

data: googleClientId Not configured

data: googleClientSecret Not configured

data: apnsMode none

data: apnsPassword Not configured

data: apnsCertifcate Not configured

info: mobile config list command OK

**mobile config get [options] [servicename] [key]**

This command gets a specific configuration option for a mobile service, in this case dynamic schema.

~$ azure mobile config get todolist dynamicSchemaEnabled

info: Executing command mobile config get

data: dynamicSchemaEnabled true

info: mobile config get command OK

**mobile config set [options] [servicename] [key] [value]**

This command sets a specific configuration option for a mobile service, in this case dynamic schema.

~$ azure mobile config set todolist dynamicSchemaEnabled false

info: Executing command mobile config set

info: mobile config set command OK

Commands to manage mobile service tables

**mobile table list [options] [servicename]**

This command lists all tables in your mobile service.

~$azure mobile table list todolist

info: Executing command mobile table list

data: Name Indexes Rows

data: -------- ------- ----

data: Channel 1 0

data: TodoItem 1 0

info: mobile table list command OK

**mobile table show [options] [servicename] [tablename]**

This command shows returns details about a specific table.

~$azure mobile table show todolist

info: Executing command mobile table show

+ Getting table information

info: Table statistics:

data: Number of records 5

info: Table operations:

data: Operation Script Permissions

data: --------- ----------- -----------

data: insert 1900 bytes user

data: read Not defined user

data: update Not defined user

data: delete Not defined user

info: Table columns:

data: Name Type Indexed

data: ---- ------------- -------

data: id bigint(MSSQL) Yes

data: text string

data: complete boolean

info: mobile table show command OK

**mobile table create [options] [servicename] [tablename]**

This command creates a table.

~$azure mobile table create todolist Channels

info: Executing command mobile table create

+ Creating table

info: mobile table create command OK

This command supports the following additional option:

* **-p**&lt;permissions> or **--permissions**&lt;permissions>: Comma-delimited list of <operation>=<permission> pairs, where <operation> isinsert, read, update, or delete and &lt;permissions> is public, application (default), user, or admin.

**mobile data read [options] [servicename] [tablename] [query]**

This command reads data from a table.

~$azure mobile data read todolist TodoItem

info: Executing command mobile data read

data: id text complete

data: -- ------- --------

data: 1 item #1 false

data: 2 item #2 true

data: 3 item #3 false

data: 4 item #4 true

info: mobile data read command OK

This command supports the following additional options:

* **-k**<skip> or **--skip**<skip>: Skips the number of rows specified by <skip>.
* **-t**<top> or **--top**<top>: Returns a specific number of rows, specified by <top>.
* **-l** or **--list**: Returns data in a list format.

**mobile table update [options] [servicename] [tablename]**

This command changes delete permissions on a table to administrators only.

~$azure mobile table update todolist Channels -p delete=admin

info: Executing command mobile table update

+ Updating permissions

info: Updated permissions

info: mobile table update command OK

This command supports the following additional options:

* **-p**&lt;permissions> or **--permissions**&lt;permissions>: Comma-delimited list of <operation>=<permission> pairs, where <operation> isinsert, read, update, or delete and &lt;permissions> is public, application (default), user, or admin.
* **--deleteColumn**<columns>: Comma-delimited list of columns to delete, as <columns>.
* **-q** or **--quiet**: Deletes columns without prompting for confirmation.
* **--addIndex**<columns>: Comma-delimited list of columns to include in the index.
* **--deleteIndex**<columns>: Comma-delimited list of columns to exclude from the index.

**mobile table delete [options] [servicename] [tablename]**

This command deletes a table.

~$azure mobile table delete todolist Channels

info: Executing command mobile table delete

Do you really want to delete the table (yes/no): yes

+ Deleting table

info: mobile table delete command OK

Specify the -q parameter to delete the table without confirmation. Do this to prevent blocking of automation scripts.

**mobile data truncate [options] [servicename] [tablename]**

This commands removes all rows of data from the table.

~$azure mobile data truncate todolist TodoItem

info: Executing command mobile data truncate

info: There are 7 data rows in the table.

Do you really want to delete all data from the table? (y/n): y

info: Deleted 7 rows.

info: mobile data truncate command OK

Commands to manage scripts

Commands in this section are used to manage the server scripts that belong to a mobile service. For more information, see [Work with server scripts in Mobile Services](http://www.windowsazure.com/en-us/develop/mobile/how-to-guides/work-with-server-scripts/).

**mobile script list [options] [servicename]**

This command lists registered scripts, including both table and scheduler scripts.

~$azure mobile script list todolist

info: Executing command mobile script list

+ Getting script information

info: Table scripts

data: Name Size

data: --------------------- ----

data: table/TodoItem.delete 256

data: table/Devices.insert 1660

error: Unable to get shared scripts

info: Scheduler scripts

data: Name Status Interval Last run Next run

data: ------------------- --------- --------- --------- ---------

data: scheduler/undefined undefined undefined undefined undefined

data: scheduler/undefined undefined undefined undefined undefined

info: mobile script list command OK

**mobile script download [options] [servicename] [scriptname]**

This command downloads the insert script from the TodoItem table to a file named todoitem.insert.js in the table subfolder.

~$azure mobile script download todolist table/todoitem.insert.js

info: Executing command mobile script download

info: Saved script to ./table/todoitem.insert.js

info: mobile script download command OK

This command supports the following additional options:

* **-p**<path> or **--path**<path>: The location in the file in which to save the script, where the current working directory is the default.
* **-f**<file> or **--file**<file>: The name of the file in which to save the script.
* **-o** or **--override**: Overwrite an existing file.
* **-c** or **--console**: Write the script to the console instead of to a file.

**mobile script upload [options] [servicename] [scriptname]**

This command uploads a new script named todoitem.insert.js from the table subfolder.

~$azure mobile script upload todolist table/todoitem.insert.js

info: Executing command mobile script upload

info: mobile script upload command OK

The name of the file must be composed from the table and operation names, and it must be located in the table subfolder relative to the location where the command is executed. You can also use the **-f**<file> or **--file**<file> parameter to specify a different filename and path to the file that contains the script to register.

**mobile script delete [options] [servicename] [scriptname]**

This command removes the existing insert script from the TodoItem table.

~$azure mobile script delete todolist table/todoitem.insert.js

info: Executing command mobile script delete

info: mobile script delete command OK

Commands to manage scheduled jobs

Commands in this section are used to manage scheduled jobs that belong to a mobile service. For more information, see [Schedule jobs](http://msdn.microsoft.com/en-us/library/windowsazure/jj860528.aspx).

**mobile job list [options] [servicename]**

This command lists scheduled jobs.

~$azure mobile job list todolist

info: Executing command mobile job list

info: Scheduled jobs

data: Job name Script name Status Interval Last run Next run

data: ---------- -------------------- -------- ----------- -------------------- --------------------

data: getUpdates scheduler/getUpdates enabled 15 [minute] 2013-01-14T16:15:00Z 2013-01-14T16:30:00Z

info: You can manipulate scheduled job scripts using the 'azure mobile script' command.

info: mobile job list command OK

**mobile job create [options] [servicename] [jobname]**

This command creates a new job named getUpdates that is scheduled to run hourly.

~$azure mobile job create -i 1 -u hour todolist getUpdates

info: Executing command mobile job create

info: Job was created in disabled state. You can enable the job using the 'azure mobile job update' command.

info: You can manipulate the scheduled job script using the 'azure mobile script' command.

info: mobile job create command OK

This command supports the following additional options:

* **-i**<number> or **--interval**<number>: The job interval, as an integer; the default value is 15.
* **-u**<unit> or **--intervalUnit**<unit>: The unit for the *interval*, which can be one of the following values:
  + **minute** (default)
  + **hour**
  + **day**
  + **month**
  + **none** (on-demand jobs)
* **-t**<time> **--startTime**<time> The start time of the first run for the script, in ISO format; the default value is now.

NOTE:

New jobs are created in a disabled state because a script must still be uploaded. Use the **mobile script upload** command to upload a script and the **mobile job update** command to enable the job.

**mobile job update [options] [servicename] [jobname]**

The following command enables the disabled getUpdates job.

~$azure mobile job update -a enabled todolist getUpdates

info: Executing command mobile job update

info: mobile job update command OK

This command supports the following additional options:

* **-i**<number> or **--interval**<number>: The job interval, as an integer; the default value is 15.
* **-u**<unit> or **--intervalUnit**<unit>: The unit for the *interval*, which can be one of the following values:
  + **minute** (default)
  + **hour**
  + **day**
  + **month**
  + **none** (on-demand jobs)
* **-t**<time> **--startTime**<time> The start time of the first run for the script, in ISO format; the default value is now.
* **-a**<status> or **--status**<status>: The job status, which can be either enabled or disabled.

**mobile job delete [options] [servicename] [jobname]**

This command removes the getUpdates scheduled job from the TodoList server.

~$azure mobile job delete todolist getUpdates

info: Executing command mobile job delete

info: mobile job delete command OK

NOTE:

Deleting a job also deletes the uploaded script.

Commands to scale a mobile service

Commands in this section are used to scale a mobile service. For more information, see [Scaling a mobile service](http://msdn.microsoft.com/en-us/library/windowsazure/jj193178.aspx).

**mobile scale show [options] [servicename]**

This command displays scale information, including current compute mode and number of instances.

~$azure mobile scale show todolist

info: Executing command mobile scale show

data: webspace WESTUSWEBSPACE

data: computeMode Free

data: numberOfInstances 1

info: mobile scale show command OK

**mobile scale change [options] [servicename]**

This command changes the scale of the mobile service from free to premium mode.

~$azure mobile scale change -c Reserved -i 1 todolist

info: Executing command mobile scale change

+ Rescaling the mobile service

info: mobile scale change command OK

This command supports the following additional options:

* **-c**<mode> or **--computeMode**<mode>: The compute mode must be either Free or Reserved.
* **-i**<count> or **--numberOfInstances**<count>: The number of instances used when running in reserved mode.

NOTE:

When you set compute mode to Reserved, all of your mobile services in the same region run in premium mode.

Commands to enable preview features for your Mobile Service

**mobile preview list [options] [servicename]**

This command displays the preview features available on the specified service and whether they are enabled.

~$ azure mobile preview list mysite

info: Executing command mobile preview list

+ Getting preview features

data: Preview feature Enabled

data: --------------- -------

data: SourceControl No

data: Users No

info: You can enable preview features using the 'azure mobile preview enable' command.

info: mobile preview list command OK

**mobile preview enable [options] [servicename] [featurename]**

This command enables the specified preview feature for a mobile service. Note that once enabled, preview features cannot be disabled for a mobile service.

Commands to manage your mobile service APIs

**mobile api list [options] [servicename]**

This command displays a list mobile service custom APIs that you have created for your mobile service.

~$ azure mobile api list mysite

info: Executing command mobile api list

+ Retrieving list of APIs

info: APIs

data: Name Get Put Post Patch Delete

data: -------------------- ----------- ----------- ----------- ----------- -----------

data: myCustomRetrieveAPI application application application application application

info: You can manipulate API scripts using the 'azure mobile script' command.

info: mobile api list command OK

**mobile api create [options] [servicename] [apiname]**

Creates a mobile service custom API

~$ azure mobile api create mysite myCustomRetrieveAPI

info: Executing command mobile api create

+ Creating custom API: 'myCustomRetrieveAPI'

info: API was created successfully. You can modify the API using the 'azure mobile script' command.

info: mobile api create command OK

This command supports the following additional option:

**-p** or **--permissions** <permissions>: A comma delimited list of <method>=<permission> pairs.

**mobile api update [options] [servicename] [apiname]**

This command updates the specified mobile service custom API.

This command supports the following additional option:

This command supports the following additional options:

* **-p** or **--permissions** <permissions>: A comma delimited list of <method>=<permission> pairs.
* **-f** or **--force**: Overrides any custom changes to the permissions metadata file.

**mobile api delete [options] [servicename] [apiname]**

~$ azure mobile api delete mysite myCustomRetrieveAPI

info: Executing command mobile api delete

+ Deleting API: 'myCustomRetrieveAPI'

info: mobile api delete command OK

This command deletes the specified mobile service custom API.

Commands to manage your mobile application app settings

**mobile appsetting list [options] [servicename]**

This command displays the mobile application app settings for the specified service.

~$ azure mobile appsetting list mysite

info: Executing command mobile appsetting list

+ Retrieving app settings

data: Name Value

data: ----------------- -----

data: enablebetacontent true

info: mobile appsetting list command OK

**mobile appsetting add [options] [servicename] [name] [value]**

This command adds a custom application setting for your mobile service.

~$ azure mobile appsetting add mysite enablebetacontent true

info: Executing command mobile appsetting add

+ Retrieving app settings

+ Adding app setting

info: mobile appsetting add command OK

**mobile appsetting delete [options] [servicename] [name]**

This command removes the specified application setting for your mobile service.

~$ azure mobile appsetting delete mysite enablebetacontent

info: Executing command mobile appsetting delete

+ Retrieving app settings

+ Removing app setting 'enablebetacontent'

info: mobile appsetting delete command OK

**mobile appsetting show [options] [servicename] [name]**

This command removes the specified application setting for your mobile service.

~$ azure mobile appsetting show mysite enablebetacontent

info: Executing command mobile appsetting show

+ Retrieving app settings

info: enablebetacontent: true

info: mobile appsetting show command OK

Manage tool local settings

Local settings are your subscription ID and Default Storage Account Name.

**config list [options]**

This command displays config settings.

~$ azure config list

info: Displaying config settings

data: Setting Value

data: --------------------- ------------------------------------

data: subscription 32-digit-subscription-key

data: defaultStorageAccount name

**config set [options] <name>,<value>**

This command changes a config setting.

~$ azure config set defaultStorageAccount myname

info: Setting 'defaultStorageAccount' to value 'myname'

info: Changes saved.

Commands to manage Service Bus

Use these commands to manage your Service Bus account

**sb namespace check [options] <name>**

Check that a service bus namespace is legal and available.

**sb namespace create <name> <location>**

Creates a new Service Bus namespace.

~$ azure sb namespace create mysbnamespacea-test "West US"

info: Executing command sb namespace create

+ Creating namespace mysbnamespacea-test in region West US

data: Name: mysbnamespacea-test

data: Region: West US

data: DefaultKey: fBu8nQ9svPIesFfMFVhCFD+/sY0rRbifWMoRpYy0Ynk=

data: Status: Activating

data: CreatedAt: 2013-11-14T16:23:29.32Z

data: AcsManagementEndpoint: https://mysbnamespacea-test-sb.accesscontrol.windows.net/

data: ServiceBusEndpoint: https://mysbnamespacea-test.servicebus.windows.net/

data: ConnectionString: Endpoint=sb://mysbnamespacea-test.servicebus.windows.

net/;SharedSecretIssuer=owner;SharedSecretValue=fBu8nQ9svPIesFfMFVhCFD+/sY0rRbif

WMoRpYy0Ynk=

data: SubscriptionId: 8679c8be3b0549d9b8fb4bd232a48931

data: Enabled: true

data: \_: [object Object]

info: sb namespace create command OK

**sb namespace delete <name>**

Remove a namespace.

~$ azure sb namespace delete mysbnamespacea-test

info: Executing command sb namespace delete

Delete namespace mysbnamespacea-test? [y/n] y

+ Deleting namespace mysbnamespacea-test

info: sb namespace delete command OK

**sb namespace list**

List all namespaces created for your account.

~$ azure sb namespace list

info: Executing command sb namespace list

+ Getting namespaces

data: Name Region Status

data: ------------------- ------- ------

data: mysbnamespacea-test West US Active

info: sb namespace list command OK

**sb namespace location list**

Display a list of all available namespace locations.

~$ azure sb namespace location list

info: Executing command sb namespace location list

+ Getting locations

data: Name Code

data: ---------------- ----------------

data: East Asia East Asia

data: West Europe West Europe

data: North Europe North Europe

data: East US East US

data: Southeast Asia Southeast Asia

data: North Central US North Central US

data: West US West US

data: South Central US South Central US

info: sb namespace location list command OK

**sb namespace show <name>**

Display details about a specific namespace.

~$ azure sb namespace show mysbnamespacea-test

info: Executing command sb namespace show

+ Getting namespace

data: Name: mysbnamespacea-test

data: Region: West US

data: DefaultKey: fBu8nQ9svPIesFfMFVhCFD+/sY0rRbifWMoRpYy0Ynk=

data: Status: Active

data: CreatedAt: 2013-11-14T16:23:29.32Z

data: AcsManagementEndpoint: https://mysbnamespacea-test-sb.accesscontrol.windows.net/

data: ServiceBusEndpoint: https://mysbnamespacea-test.servicebus.windows.net/

data: ConnectionString: Endpoint=sb://mysbnamespacea-test.servicebus.windows.

net/;SharedSecretIssuer=owner;SharedSecretValue=fBu8nQ9svPIesFfMFVhCFD+/sY0rRbif

WMoRpYy0Ynk=

data: SubscriptionId: 8679c8be3b0549d9b8fb4bd232a48931

data: Enabled: true

data: UpdatedAt: 2013-11-14T16:25:37.85Z

info: sb namespace show command OK

**sb namespace verify <name>**

Check whether the namespace is available.

Commands to manage your Storage objects

Commands to manage your Storage accounts

**storage account list [options]**

This command displays the storage accounts on your subscription.

~$ azure storage account list

info: Executing command storage account list

+ Getting storage accounts

data: Name Label Location

data: --------------- ----- --------

data: mybasestorage West US

info: storage account list command OK

**storage account show [options]**

This command displays information about the specified storage account including the URI and account properties.

**storage account create [options]**

This command creates a storage account based on the supplied options.

~$ azure storage account create mybasestorage --label PrimaryStorage --location "West US"

info: Executing command storage account create

+ Creating storage account

info: storage account create command OK

This command supports the following additional options:

* **-e** or **--label** <label>: The label for the storage account.
* **-d** or **--description** <description>: The description storage account.
* **-l** or **--location** <name>: The geographic region in which to create the storage account.
* **-a** or **--affinity-group** <name>: The affinity group with which to associate the storage account.
* **--geoReplication**: Indicates if geo-replication is enabled.
* **--disable-geoReplication**: Indicates if geo-replication is disabled.

**storage account set [options]**

This command updates the specified storage account.

~$ azure storage account set mybasestorage --geoReplication

info: Executing command storage account set

+ Updating storage account

info: storage account set command OK

This command supports the following additional options:

* **-e** or **--label** <label>: The label for the storage account.
* **-d** or **--description** <description>: The description storage account.
* **-l** or **--location** <name>: The geographic region in which to create the storage account.
* **--geoReplication**: Indicates if geo-replication is enabled.
* **--disable-geoReplication**: Indicates if geo-replication is disabled.

**storage account delete [options]**

This command deletes the specified storage account.

This command supports the following additional option:

**-q** or **--quiet**: Do not prompt for confirmation. Use this option in automated scripts.

Commands to manage your Storage account keys

**storage account keys list [options]**

This command lists the primary and secondary keys for the specified storage account.

**storage account keys renew [options]**

Commands to manage your Storage container

**storage container list [options] [prefix]**

This command displays the storage container list for a specified storage account. The storage account is specified by either the connection string or the storage account name and account key.

This command supports the following additional options:

* **-p** or **-prefix** <prefix>: The storage container name prefix.
* **-a** or **--account-name** <accountName>: The storage account name.
* **-k** or **--account-key** <accountKey>: The storage account key.
* **-c** or **--connection-string** <connectionString>: The storage connection string.
* **--debug**: Runs the storage command in debug mode.

**storage container show [options] [container]\*\* \*\*storage container create [options] [container]**

This command creates a storage container for the specified storage account. The storage account is specified by either the connection string or the storage account name and account key.

This command supports the following additional options:

* **--container** <container>: The name of the storage container to create.
* **-p** or **-prefix** <prefix>: The storage container name prefix.
* **-a** or **--account-name** <accountName>: The storage account name
* **-k** or **--account-key** <accountKey>: The storage account key
* **-c** or **--connection-string** <connectionString>: The storage connection string
* **--debug**: Runs the storage command in debug mode.

**storage container delete [options] [container]**

This command deletes the specified storage container. The storage account is specified by either the connection string or the storage account name and account key.

This command supports the following additional options:

* **--container** <container>: The name of the storage container to create.
* **-p** or **-prefix** <prefix>: The storage container name prefix.
* **-a** or **--account-name** <accountName>: The storage account name.
* **-k** or **--account-key** <accountKey>: The storage account key.
* **-c** or **--connection-string** <connectionString>: The storage connection string.
* **--debug**: Runs the storage command in debug mode.

**storage container set [options] [container]**

This command sets access control list for the storage container. The storage account is specified by either the connection string or the storage account name and account key.

This command supports the following additional options:

* **--container** <container>: The name of the storage container to create.
* **-p** or **-prefix** <prefix>: The storage container name prefix.
* **-a** or **--account-name** <accountName>: The storage account name.
* **-k** or **--account-key** <accountKey>: The storage account key.
* **-c** or **--connection-string** <connectionString>: The storage connection string.
* **--debug**: Runs the storage command in debug mode.

Commands to manage your Storage blob

**storage blob list [options] [container] [prefix]**

This command returns a list of the storage blobs in the specified storage container.

This command supports the following additional options:

* **--container** <container>: The name of the storage container to create.
* **-p** or **-prefix** <prefix>: The storage container name prefix.
* **-a** or **--account-name** <accountName>: The storage account name.
* **-k** or **--account-key** <accountKey>: The storage account key.
* **-c** or **--connection-string** <connectionString>: The storage connection string.
* **--debug**: Runs the storage command in debug mode.

**storage blob show [options] [container] [blob]**

This command displays the details of the specified storage blob.

This command supports the following additional options:

* **--container** <container>: The name of the storage container to create.
* **-p** or **-prefix** <prefix>: The storage container name prefix.
* **-a** or **--account-name** <accountName>: The storage account name.
* **-k** or **--account-key** <accountKey>: The storage account key.
* **-c** or **--connection-string** <connectionString>: The storage connection string.
* **--debug**: Runs the storage command in debug.

**storage blob delete [options] [container] [blob]**

This command supports the following additional options:

* **--container** <container>: The name of the storage container to create.
* **-b** or **--blob** <blobName>: The name of the storage blob to delete.
* **-q** or **--quiet**: Remove the specified Storage blob without confirmation.
* **-a** or **--account-name** <accountName>: The storage account name.
* **-k** or **--account-key** <accountKey>: The storage account key.
* **-c** or **--connection-string** <connectionString>: The storage connection string.
* **--debug**: Runs the storage command in debug.

**storage blob upload [options] [file] [container] [blob]**

This command upload the specified file to the specified\ storage blob.

This command supports the following additional options:

* **--container** <container>: The name of the storage container to create.
* **-b** or **--blob** <blobName>: The name of the storage blob to upload.
* **-t** or **--blobtype** <blobtype>: The storage blob type: Page or Block.
* **-p** or **--properties** <properties>: The storage blob properties for uploaded file. Properties are key=value pair s and separated with semicolon(;). Available properties are contentType, contentEncoding, contentLanguage, and cacheControl.
* **-m** or **--metadata** <metadata>: The storage blob metadata for uploaded file. Metadata are key=value pairs an d separated with semicolon (;).
* **--concurrenttaskcount** <concurrenttaskcount>: The maximum number of concurrent upload requests.
* **-q** or **--quiet**: Overwrite the specified Storage blob without confirmation.
* **-a** or **--account-name** <accountName>: The storage account name.
* **-k** or **--account-key** <accountKey>: The storage account key.
* **-c** or **--connection-string** <connectionString>: The storage connection string.
* **--debug**: Runs the storage command in debug.

**storage blob download [options] [container] [blob] [destination]**

This command downloads the specified storage blob.

This command supports the following additional options:

* **--container** <container>: The name of the storage container to create.
* **-b** or **--blob** <blobName>: The storage blob name.
* **-d** or **--destination** [destination]: The download destination file or directory path.
* **-m** or **--checkmd5**: The check md5sum for the downloaded file.
* **--concurrenttaskcount** <concurrenttaskcount> the maximum number of concurrent upload requests
* **-q** or **--quiet**: Overwrite the destination file without confirmation.
* **-a** or **--account-name** <accountName>: The storage account name.
* **-k** or **--account-key** <accountKey>: The storage account key.
* **-c** or **--connection-string** <connectionString>: The storage connection string.
* **--debug**: Runs the storage command in debug.

Commands to manage SQL Databases

Use these commands to manage your Azure SQL Databases

Commands to manage SQL Servers.

Use these commands to manage your SQL Servers

**sql server create <administratorLogin> <administratorPassword> <location>**

Create a new database server

~$ azure sql server create test T3stte$t "West US"

info: Executing command sql server create

+ Creating SQL Server

data: Server Name i1qwc540ts

info: sql server create command OK

**sql server show <name>**

Display server details.

~$ azure sql server show xclfgcndfg

info: Executing command sql server show

+ Getting SQL server

data: SQL Server Name xclfgcndfg

data: SQL Server AdministratorLogin msopentechforums

data: SQL Server Location West US

data: SQL Server FullyQualifiedDomainName xclfgcndfg.database.windows.net

info: sql server show command OK

**sql server list**

Get the list of servers.

~$ azure sql server list

info: Executing command sql server list

+ Getting SQL server

data: Name Location

data: ---------- --------

data: xclfgcndfg West US

info: sql server list command OK

**sql server delete <name>**

Deletes a server

~$ azure sql server delete i1qwc540ts

info: Executing command sql server delete

Delete server i1qwc540ts? [y/n] y

+ Removing SQL Server

info: sql server delete command OK

Commands to manage SQL Databases

Use these commands to manage your SQL Databases.

**sql db create [options] <serverName> <databaseName> <administratorPassword>**

Creates a new database instance

~$ azure sql db create fr8aelne00 newdb test

info: Executing command sql db create

Administrator password: \*\*\*\*\*\*\*\*

+ Creating SQL Server Database

info: sql db create command OK

**sql db show [options] <serverName> <databaseName> <administratorPassword>**

Display database details.

C:\windows\system32>azure sql db show fr8aelne00 newdb test

info: Executing command sql db show

Administrator password: \*\*\*\*\*\*\*\*

+ Getting SQL server databases

data: Database \_ ContentRootElement=m:properties, id=https://fr8aelne00.datab

ase.windows.net/v1/ManagementService.svc/Server2('fr8aelne00')/Databases(4), ter

m=Microsoft.SqlServer.Management.Server.Domain.Database, scheme=http://schemas.m

icrosoft.com/ado/2007/08/dataservices/scheme, link=[rel=edit, title=Database, hr

ef=Databases(4), rel=http://schemas.microsoft.com/ado/2007/08/dataservices/relat

ed/Server, type=application/atom+xml;type=entry, title=Server, href=Databases(4)

/Server, rel=http://schemas.microsoft.com/ado/2007/08/dataservices/related/Servi

ceObjective, type=application/atom+xml;type=entry, title=ServiceObjective, href=

Databases(4)/ServiceObjective, rel=http://schemas.microsoft.com/ado/2007/08/data

services/related/DatabaseMetrics, type=application/atom+xml;type=entry, title=Da

tabaseMetrics, href=Databases(4)/DatabaseMetrics, rel=http://schemas.microsoft.c

om/ado/2007/08/dataservices/related/DatabaseCopies, type=application/atom+xml;ty

pe=feed, title=DatabaseCopies, href=Databases(4)/DatabaseCopies], title=, update

d=2013-11-18T19:48:27Z, name=

data: Database Id 4

data: Database Name newdb

data: Database ServiceObjectiveId 910b4fcb-8a29-4c3e-958f-f7ba794388b2

data: Database AssignedServiceObjectiveId 910b4fcb-8a29-4c3e-958f-f7ba794388b2

data: Database ServiceObjectiveAssignmentState 1

data: Database ServiceObjectiveAssignmentStateDescription Complete

data: Database ServiceObjectiveAssignmentErrorCode

data: Database ServiceObjectiveAssignmentErrorDescription

data: Database ServiceObjectiveAssignmentSuccessDate

data: Database Edition Web

data: Database MaxSizeGB 1

data: Database MaxSizeBytes 1073741824

data: Database CollationName SQL\_Latin1\_General\_CP1\_CI\_AS

data: Database CreationDate

data: Database RecoveryPeriodStartDate

data: Database IsSystemObject

data: Database Status 1

data: Database IsFederationRoot

data: Database SizeMB -1

data: Database IsRecursiveTriggersOn

data: Database IsReadOnly

data: Database IsFederationMember

data: Database IsQueryStoreOn

data: Database IsQueryStoreReadOnly

data: Database QueryStoreMaxSizeMB

data: Database QueryStoreFlushPeriodSeconds

data: Database QueryStoreIntervalLengthMinutes

data: Database QueryStoreClearAll

data: Database QueryStoreStaleQueryThresholdDays

info: sql db show command OK

**sql db list [options] <serverName> <administratorPassword>**

List the databases.

~$ azure sql db list fr8aelne00 test

info: Executing command sql db list

Administrator password: \*\*\*\*\*\*\*\*

+ Getting SQL server databases

data: Name Edition Collation MaxSizeInGB

data: ------ ------- ---------------------------- -----------

data: master Web SQL\_Latin1\_General\_CP1\_CI\_AS 5

info: sql db list command OK

**sql db delete [options] <serverName> <databaseName> <administratorPassword>**

Deletes a database.

~$ azure sql db delete fr8aelne00 newdb test

info: Executing command sql db delete

Administrator password: \*\*\*\*\*\*\*\*

Delete database newdb? [y/n] y

+ Getting SQL server databases

+ Removing database

info: sql db delete command OK

Commands to manage your SQL Server firewall rules

Use these commands to manage your SQL Server firewall rules

**sql firewallrule create [options] <serverName> <ruleName> <startIPAddress> <endIPAddress>**

Create a new firewall rule for a SQL Server.

~$ azure sql firewallrule create fr8aelne00 allowed 131.107.0.0 131.107.255.255

info: Executing command sql firewallrule create

+ Creating Firewall Rule

info: sql firewallrule create command OK

**sql firewallrule show [options] <serverName> <ruleName>**

Show firewall rule details.

~$ azure sql firewallrule show fr8aelne00 allowed

info: Executing command sql firewallrule show

+ Getting firewall rule

data: Firewall rule Name allowed

data: Firewall rule Type Microsoft.SqlAzure.FirewallRule

data: Firewall rule State Normal

data: Firewall rule SelfLink https://management.core.windows.net/9e672699-105

5-41ae-9c36-e85152f2e352/services/sqlservers/servers/fr8aelne00/firewallrules/allowed

data: Firewall rule ParentLink https://management.core.windows.net/9e672699-1

055-41ae-9c36-e85152f2e352/services/sqlservers/servers/fr8aelne00

data: Firewall rule StartIPAddress 131.107.0.0

data: Firewall rule EndIPAddress 131.107.255.255

info: sql firewallrule show command OK

**sql firewallrule list [options] <serverName>**

List the firewall rules.

~$ azure sql firewallrule list fr8aelne00

info: Executing command sql firewallrule list

\data: Name Start IP address End IP address

data: ------- ---------------- ---------------

data: allowed 131.107.0.0 131.107.255.255

+

info: sql firewallrule list command OK

**sql firewallrule delete [options] <serverName> <ruleName>**

This command will delete a firewall rule.

~$ azure sql firewallrule delete fr8aelne00 allowed

info: Executing command sql firewallrule delete

Delete rule allowed? [y/n] y

+ Removing firewall rule

info: sql firewallrule delete command OK

Commands to manage your Virtual Networks

Use these commands to manage your Virtual Networks

**network vnet create [options] <location>**

Create a new Virtual Network.

~$ azure network vnet create vnet1 --location "West US" -v

info: Executing command network vnet create

info: Using default address space start IP: 10.0.0.0

info: Using default address space cidr: 8

info: Using default subnet start IP: 10.0.0.0

info: Using default subnet cidr: 11

verbose: Address Space [Starting IP/CIDR (Max VM Count)]: 10.0.0.0/8 (16777216)

verbose: Subnet [Starting IP/CIDR (Max VM Count)]: 10.0.0.0/11 (2097152)

verbose: Fetching Network Configuration

verbose: Fetching or creating affinity group

verbose: Fetching Affinity Groups

verbose: Fetching Locations

verbose: Creating new affinity group AG1

info: Using affinity group AG1

verbose: Updating Network Configuration

info: network vnet create command OK

**network vnet show <name>**

Show details of a Virtual Network.

~$ azure network vnet show vnet1

info: Executing command network vnet show

+ Fetching Virtual Networks

data: Name "vnet1"

data: Id "25786fbe-08e8-4e7e-b1de-b98b7e586c7a"

data: AffinityGroup "AG1"

data: State "Created"

data: AddressSpace AddressPrefixes 0 "10.0.0.0/8"

data: Subnets 0 Name "subnet-1"

data: Subnets 0 AddressPrefix "10.0.0.0/11"

info: network vnet show command OK

**vnet list**

List all existing Virtual Networks.

~$ azure network vnet list

info: Executing command network vnet list

+ Fetching Virtual Networks

data: Name Status AffinityGroup

data: ---------- ------- -------------

data: vnet1 Created AG1

data: vnet2 Created AG1

data: vnet3 Created AG1

data: vnet4 Created AG1

info: network vnet list command OK

**network vnet delete <name>**

Deletes the specified Virtual Network.

~$ azure network vnet delete opentechvn1

info: Executing command network vnet delete

+ Fetching Network Configuration

Delete the virtual network opentechvn1 ? (y/n) y

+ Deleting the virtual network opentechvn1

info: network vnet delete command OK

**network export [file-path]**

For advanced network configuration, you can export your network configuration locally. Note that the exported network configuration includes DNS server settings, virtual network settings, local network site settings, and other settings.

**network import [file-path]**

Import a local network configuration.

**network dnsserver register [options] <dnsIP>**

Register a DNS server that you plan to use for name resolution in your network configuration.

~$ azure network dnsserver register 98.138.253.109 --dns-id FrontEndDnsServer

info: Executing command network dnsserver register

+ Fetching Network Configuration

+ Updating Network Configuration

info: network dnsserver register command OK

**network dnsserver list**

List all the DNS servers registered in your network configuration.

~$ azure network dnsserver list

info: Executing command network dnsserver list

+ Fetching Network Configuration

data: DNS Server ID DNS Server IP

data: -------------------- --------------

data: DNS-bb39b4ac34d66a86 44.55.22.11

data: FrontEndDnsServer 98.138.253.109

info: network dnsserver list command OK

**network dnsserver unregister [options] <dnsIP>**

Removes a DNS server entry from the network configuration.

~$ azure network dnsserver unregister 77.88.99.11

info: Executing command network dnsserver unregister

+ Fetching Network Configuration

Delete the DNS server entry dns-4 ( 77.88.99.11 ) %s ? (y/n) y

+ Deleting the DNS server entry dns-4 ( 77.88.99.11 )

info: network dnsserver unregister command OK