Pexip Infinity and Microsoft Azure

Deployment Guide

Software Version 20.1

Document Version 20.1.a

February 2019

pexip

Contents

Introduction 2
Deployment guidelines 3
Configuring your Azure subscription 5
Preparing your Azure environment 6
Configuring Azure Network Security Groups 10
Creating VM instances in Azure for your Pexip nodes 13
Obtaining and preparing disk images for Azure deployments 22
Deploying a Management Node in Azure 25
Deploying a Conferencing Node in Azure 27
Configuring dynamic bursting to the Microsoft Azure cloud 28
Managing Azure instances 33

Introduction

The Microsoft Azure Virtual Machines (VMs) service provides scalable computing capacity in the Microsoft Azure cloud. Using Azure eliminates your need to invest in hardware up front, so you can deploy Pexip Infinity even faster.

You can use Azure to launch as many or as few virtual servers as you need, and use those virtual servers to host a Pexip Infinity Management Node and as many Conferencing Nodes as required for your Pexip Infinity platform.

Azure enables you to scale up or down to handle changes in requirements or spikes in conferencing requirements. You can also use the Azure APIs and the Pexip Infinity management API to monitor usage and bring up / tear down Conferencing Nodes as required to meet conferencing demand, or allow Pexip Infinity to handle this automatically for you via its dynamic bursting capabilities.

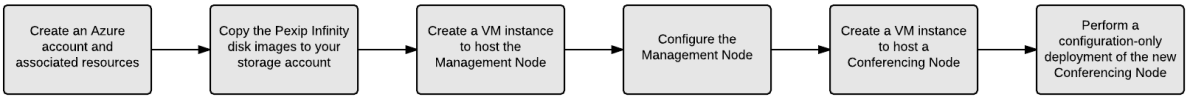
Pexip publishes disk images for the Pexip Infinity Management Node and Conferencing Nodes. These images may be used to launch instances of each node type as required.

© 2019 Pexip AS

Deployment guidelines

This section summarizes the Azure deployment options and limitations, and provides guidance on our recommended Azure instance types, security groups and IP addressing options.

This flowchart provides an overview of the basic steps involved in deploying the Pexip Infinity platform on Azure:



Deployment models

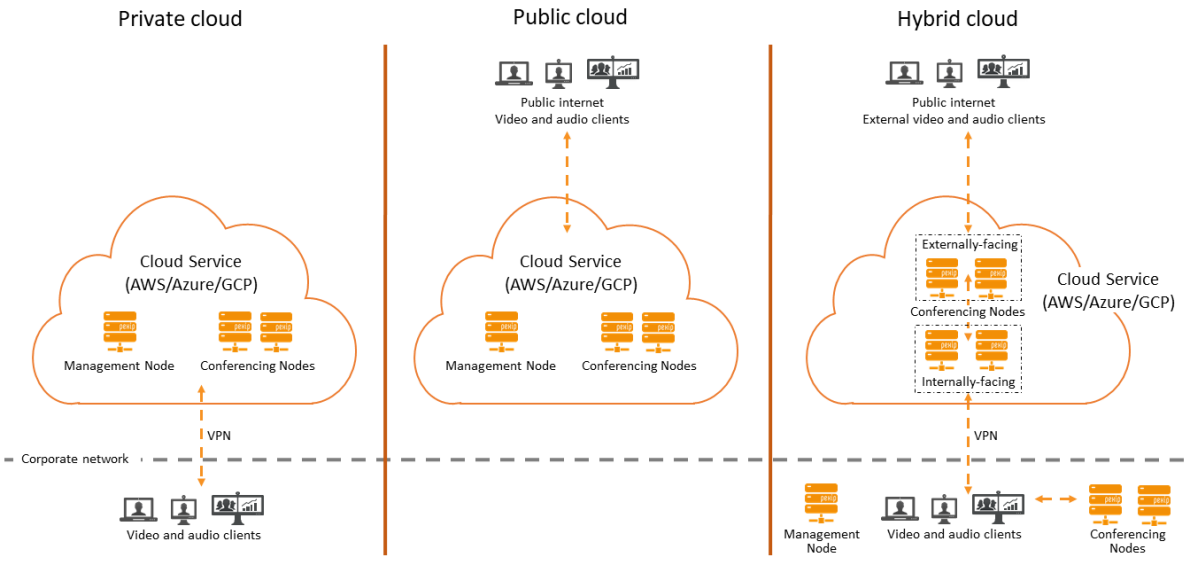
Azure has two deployment models: Classic and Resource Manager.

Resource Manager is the recommended deployment model for new workloads and is the only model supported by Pexip Infinity.

Deployment options

There are three main deployment options for your Pexip Infinity platform when using the Azure cloud:

- **Private cloud**: all nodes are deployed within Azure. Private addressing is used for all nodes and connectivity is achieved by configuring a VPN tunnel between the corporate network and Azure. As all nodes are private, this is equivalent to an on-premises deployment which is only available to users internal to the organization.
- **Public cloud**: all nodes are deployed within Azure. All nodes have a private address but, in addition, public IP addresses are allocated to each node. The node's private addresses are only used for inter-node communications. Each node's public address is then configured on the relevant node as a static NAT address. Access to the nodes is permitted from the public internet, or a restricted subset of networks, as required. Any systems or endpoints that will send signaling and media traffic to those Pexip Infinity nodes must send that traffic to the public address of those nodes. If you have internal systems or endpoints communicating with those nodes, you must ensure that your local network allows such routing.
- **Hybrid cloud**: the Management Node, and optionally some Conferencing Nodes, are deployed in the corporate network. A VPN tunnel is created between the corporate network and Azure. Additional Conferencing Nodes are deployed in Azure and are managed from the on-premises Management Node. The Azure-hosted Conferencing Nodes can be either internally-facing, privately-addressed (private cloud) nodes; or externally-facing, publicly-addressed (public cloud) nodes; or a combination of private and public nodes (where the private nodes are in a different Pexip Infinity system location to the public nodes). You may also want to consider dynamic bursting, where the Azure-hosted Conferencing Nodes are only started up and used when you have reached capacity on your on-premises nodes.



All of the Pexip nodes that you deploy in the cloud are completely dedicated to running the Pexip Infinity platform— you maintain full data ownership and control of those nodes.

Limitations

The following limitations currently apply:

- The OS username is always **admin**, regardless of any other username configured through the Azure Portal.
- SSH keys are the preferred authentication mechanism for Pexip Infinity instances hosted in the Azure Cloud. Password-based authentication also works, however, and will use the password provisioned at instance deployment time.

  Note that:
  - Pexip Infinity node instances only support a single SSH key pair.
  - If you are using a Linux or Mac SSH client to access your instance you must use the **chmod** command to make sure that your private key file on your local client (SSH private keys are never uploaded) is not publicly viewable. For example, if the name of your private key file is my-key-pair.pem, use the following command: `chmod 400 /path/my-key-pair.pem`

  See https://azure.microsoft.com/en-gb/documentation/articles/virtual-machines-linux-ssh-from-windows/ for more information about using SSH on Azure.
- We do not support Azure deployments in China or Germany.

Recommended instance types and call capacity guidelines

Azure instances come in many different sizes. In general, Pexip Infinity Conferencing Nodes should be considered compute intensive and Management Nodes reflect a more general-purpose workload. Our Server design recommendations also apply to cloud-based deployments.

For deployments of up to 20 Conferencing Nodes, we recommend using:

- **Management Node**: a **Standard A3** instance (or Standard A4 v2 if the Standard A3 is not available).
- **Transcoding Conferencing Nodes**: either an **F8 v2** instance (for smaller deployments) or an **F16 v2** instance (for larger deployments).
- **Proxying Edge Nodes**: an **F4** (F-series) instance.

An **F8 v2** instance should provide capacity for approximately 15 HD / 37 SD / 230 audio-only calls per Transcoding Conferencing Node, and an **F16 v2** instance should provide capacity for approximately 33 HD / 75 SD / 400 audio-only calls per Transcoding Conferencing Node.

If the Fv2 series is not available in your region, then you can use an F8 instance (F-series) for your Transcoding Conferencing Nodes.

Capacity planning

By default, Azure Resource Manager virtual machine cores have a regional total limit **and** a regional per series (F, Fv2, etc.) limit, that are enforced per subscription. Typically, for each subscription, the default quota allows up to 10-20 CPU cores per region and 10-20 cores per series. An F8 v2 instance uses 8 CPU cores. Thus, with the default limits in place, only two F8 v2 instances may be deployed (as only 4 CPU cores will remain in either quota pool, which is insufficient for another F8 v2 instance).

The allocated quota may be increased by opening a support ticket with Microsoft via the Azure Portal. Ensure that you request a sufficient number of CPU cores. For example, if 10 Transcoding Conferencing Nodes are required, then the quota must be increased to 8 cores x 10 F8 v2 instances = 80 CPU cores of type F8 v2. It may take a number days for the quota increase request to be processed. For more information see https://docs.microsoft.com/en-us/azure/azure-subscription-service-limits.

IP addressing

Within a Virtual Network, an instance's private IP addresses can initially be allocated dynamically (using DHCP) or statically. However, after the private IP address has been assigned to the instance it remains fixed and associated with that instance until the instance is terminated. The allocated IP address is displayed in the Azure portal.

Public IP addresses may be associated with an instance. Public IPs may be dynamic (allocated at launch/start time) or statically configured. Dynamic public IP addresses do not remain associated with an instance if it is stopped — and thus it will receive a new public IP address when it is next started.

Pexip Infinity nodes must always be configured with the private IP address associated with its instance, as it is used for all internal communication between nodes. To associate an instance's public IP address with the node, configure that public IP address as the node's **Static NAT address** (via **Platform Configuration > Conferencing Nodes**).

Assumptions and prerequisites

The Pexip Infinity deployment instructions assume that within Azure you have already:

- signed up for Azure and created a user account, administrator groups etc.
- decided in which Azure region/location to deploy your Pexip Infinity platform (one Management Node and one or more associated Conferencing Nodes)
- created a Resource Group, Virtual Network, and Storage Account in the chosen Azure region (see Preparing your Azure environment)
- (if necessary) configured a VPN tunnel from the corporate/management network to the Azure Virtual Network
- set your subscription's Microsoft partner ID to Pexip (see Configuring your Azure subscription)
- created a Network Security Group (see Configuring Azure Network Security Groups for port requirements)

For more information on setting up your Azure Virtual Machine environment, see https://azure.microsoft.com/en-gb/documentation/articles/virtual-machines-linux-about/.

Multiple Virtual Networks

Pexip Infinity node instances that are hosted on Azure may be deployed across multiple Azure Virtual Networks (VNets), where each Azure VNet (and the Conferencing Nodes within it) maps onto a Pexip Infinity system location.

You must use a VNet-to-VNet VPN gateway connection; do not use VNet peering.

See https://azure.microsoft.com/en-gb/documentation/articles/vpn-gateway-howto-vnet-vnet-resource-manager-portal/ for information about how to create a connection between Azure VNets.

Configuring your Azure subscription

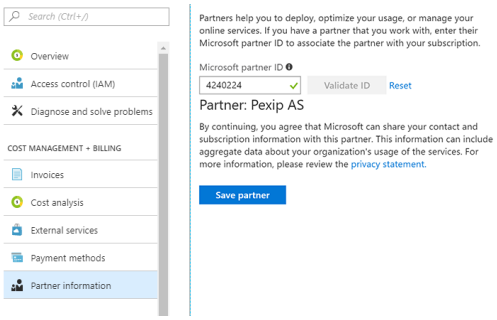
We recommend setting up a separate subscription for your Pexip deployment.

If you have an Enterprise Agreement (EA) or Pay-As-You-Go subscription you should link the Azure subscription you use for your Pexip Infinity platform to Pexip. This allows Microsoft and Pexip to provide full support to your deployment.

This does not apply if you are in the Azure Cloud Solution Provider (CSP) program.

To set your subscription's Microsoft partner ID to Pexip:

1. From the Azure portal, select **Subscriptions**.
2. Select the subscription you are using for your Pexip deployment.
3. Select **Partner Information**.
4. In the **Microsoft partner ID** field , enter **4240224** (Pexip's reference ID) and select **Validate ID**.
   Partner: Pexip AS should be displayed.

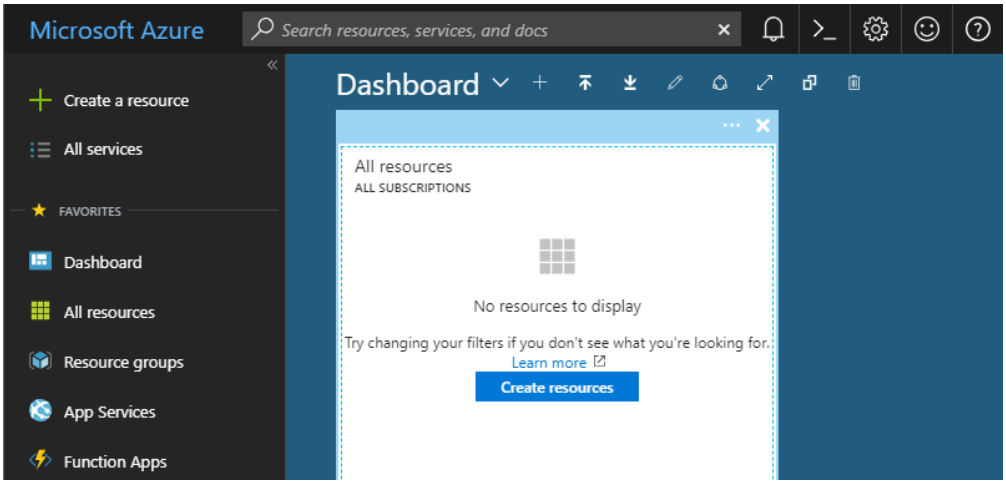


5. Select **Save partner**.

Preparing your Azure environment

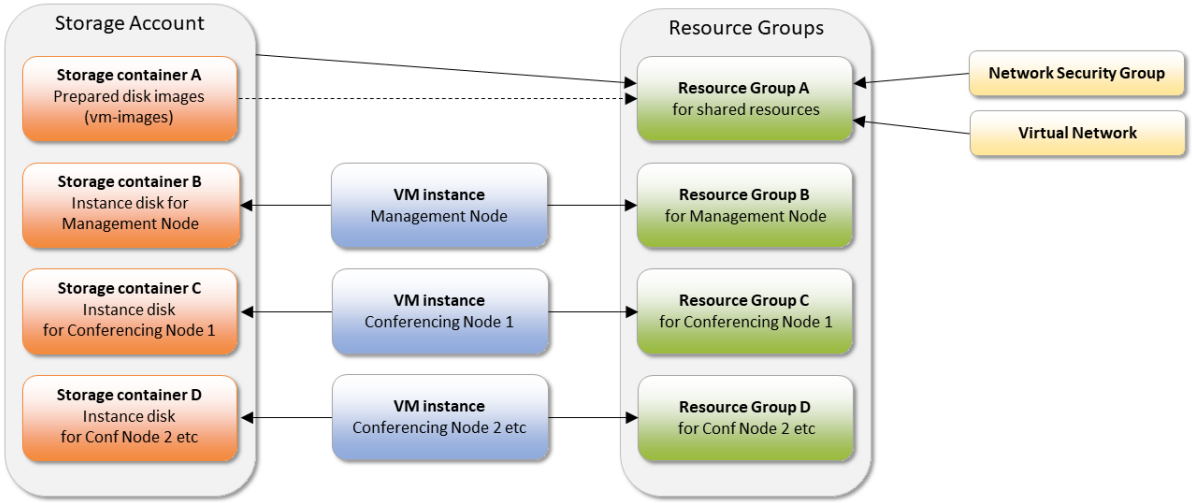
This topic explains how to prepare your Azure environment before installing Pexip Infinity. It describes how to set up your Resource Group, Virtual Network and Storage Account in your chosen Azure region.

This guide assumes you are starting with a fresh Azure portal.



Note that the images shown in this guide depict an example deployment. Unless explicitly stated, you should use your own resource names and address spaces etc. as appropriate for your own environment. Also, as the Azure portal is regularly updated, some images and labels shown here may not directly match the currently available options.

The diagram below shows how the various resource elements that you need to create within the Azure environment are related to each other. This topic explains how to create the shared Resource Group, Virtual Network and Storage Account. Subsequent topics in this guide explain how the storage container for the prepared disk image is created, and how the storage containers and Resource Groups are created for your VM instances for each Pexip node.



Creating a Resource Group

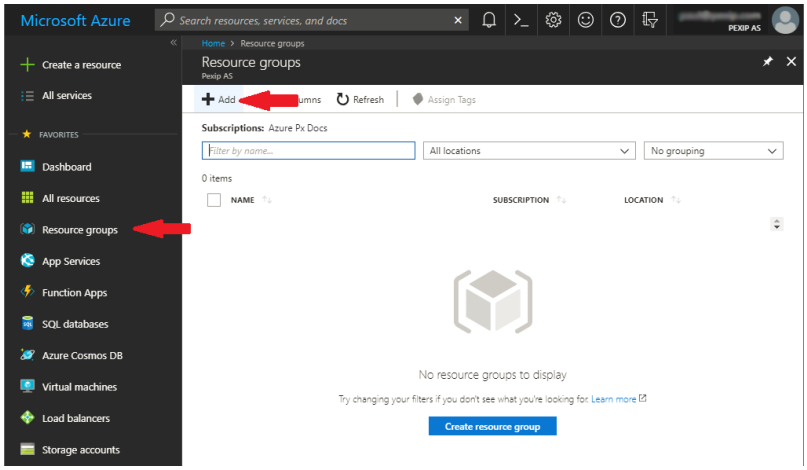
Resource Groups are logical containers for a collection of resources used for your Pexip deployment.

We recommend that all of your shared resources use the same Resource Group. The shared resources that you should place in this Resource Group include:

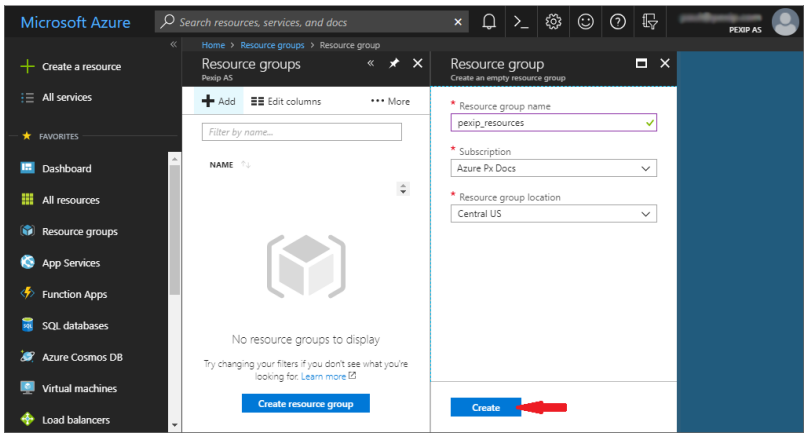
- Network Security Group
- Virtual Network
- Storage Account
- Prepared disk images (these are used to create the individual VM instances for your Conferencing Nodes and Management Node)

Note that when you subsequently deploy your Pexip Infinity nodes, each VM instance that you create for your Conferencing Nodes or Management Node must be deployed in its own storage container and its own Resource Group.

1. Decide in which region you are going to deploy your Pexip system. You can see the regions and their physical geographic locations at https://azure.microsoft.com/en-gb/global-infrastructure/regions/.
2. Configure a Resource Group for your shared resources.
   a. To create a new Resource Group, from the Azure portal select **Resource Groups** and select **+ Add**.



   b. Enter your **Resource group name**, and select your **Subscription** and your **Resource group location**, and then click **Create**.

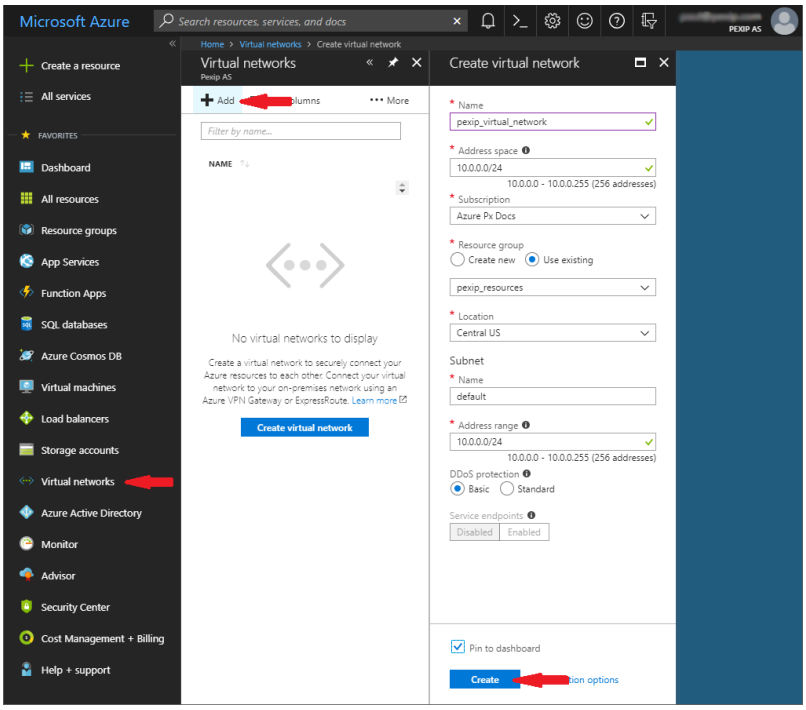


You can then, optionally, select to **Pin to dashboard**.

Creating a Virtual Network

A Virtual Network allows Azure resources to securely communicate with each other, and is where your VM instances will be deployed. See https://docs.microsoft.com/en-us/azure/virtual-network/virtual-networks-overview for more information.

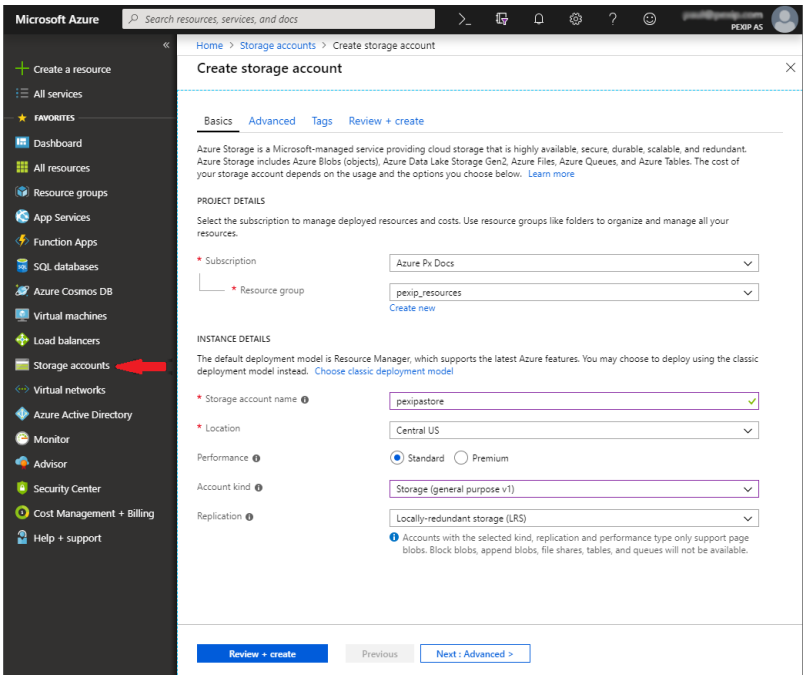
1. From the Azure portal, select **Virtual Networks**, and select **+ Add**.
2. Configure the Virtual Network's properties:
   a. Give it a **Name**, choose the IP range (**Address space**, **Subnet name** and **range**) and then ensure you select your **Subscription**, **Resource Group** and your **Location**.
   b. You can also choose to pin it to the dashboard.
   c. Select **Create**.



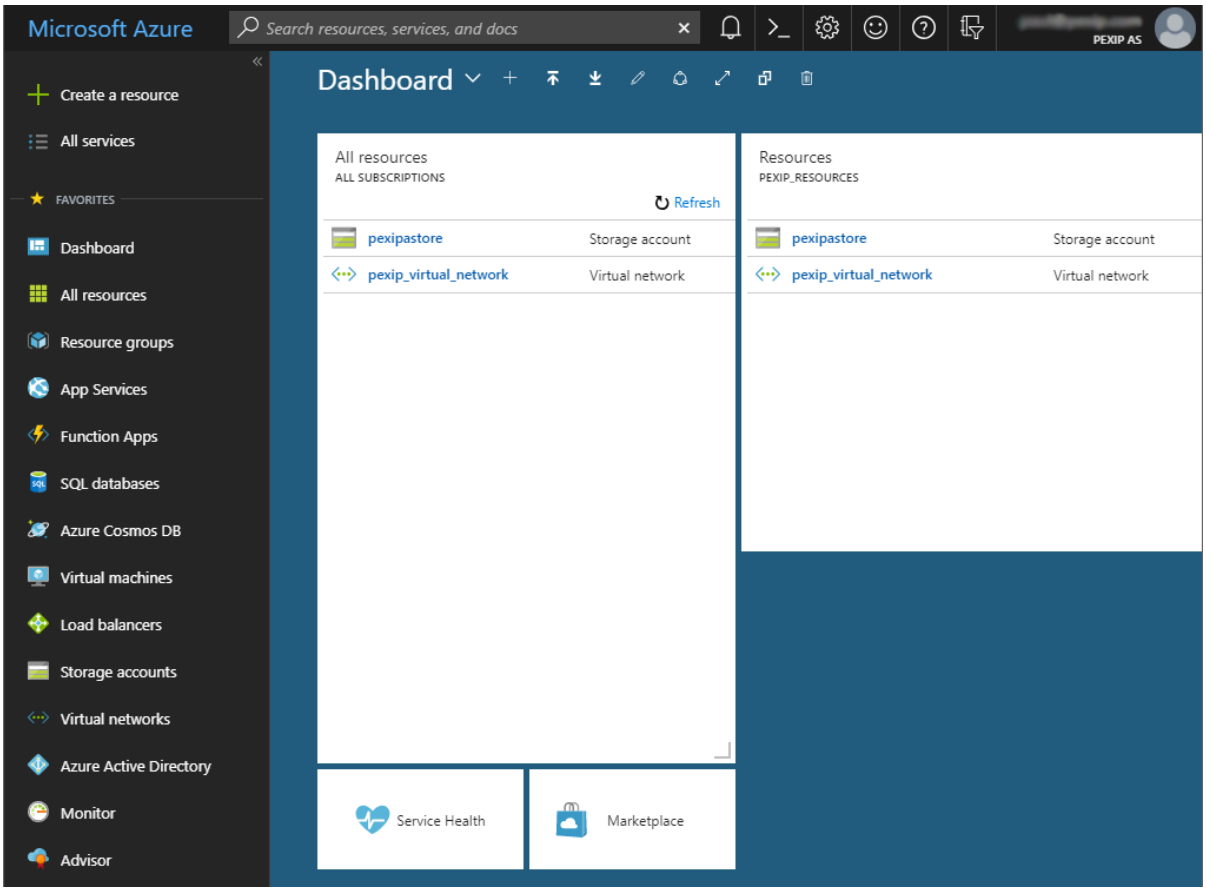
Creating a Storage Account

A general-purpose storage account provides access to all of the Azure storage services, and is used for storing Pexip Infinity disk images and VM instances.

1. From the Azure portal, select **Storage Accounts**, and select **Create storage account**.
2. Configure the Storage Account's Basic properties:
   a. Select your **Subscription** and **Resource Group**.
   b. Leave the deployment model as ***Resource Manager***.
   c. Give it a **Storage account name**.
   d. Select your **Location**.
   e. Select an **Account kind** of ***Storage (general purpose v1)***.
   f. Set **Replication** to ***Locally redundant storage (LRS)***.
   g. Select **Review + create**.
   h. Review your inputs and select **Create**.



You now have a Resource Group, Storage Account and Virtual Network for your Pexip deployment.



Configuring Azure Network Security Groups

Access to Azure instances is restricted by the Azure firewall. This may be configured by associating a subnet or instance with a Network Security Group which specifies the permitted inbound and outbound traffic from the group.

A minimal security group that permits access to a public-cloud style Pexip Infinity deployment would look similar to this:

Inbound security rules

Priority	Name	Source	Destination	Service	Action
105	HTTP	Any	Any	TCP/80	Allow
110	HTTPS	Any	Any	TCP/443	Allow
115	H.323 CS	Any	Any	TCP/1720	Allow
120	SIP TCP	Any	Any	TCP/5060	Allow
125	SIP TLS	Any	Any	TCP/5061	Allow
130	TCP call signaling	Any	Any	TCP/33000-39999	Allow
135	TCP call media	Any	Any	TCP/40000-49999	Allow
140	H.323 LS	Any	Any	UDP/1719	Allow
145	SIP UDP *	Any	Any	UDP/5060	Allow
150	UDP call signaling	Any	Any	UDP/33000-39999	Allow
155	UDP call media	Any	Any	UDP/40000-49999	Allow
160	Management traffic	CIDR block: <management station IP address/subnet>	Any	Any/Any	Allow

* only required if you intend to enable SIP over UDP

Where **Any** implies any source/destination and **<management station IP address/subnet>** should be restricted to a single IP address or subnet for management access only.

Outbound security rules

The default network security group rules suffice. These permit outbound traffic to the same Virtual Network, or to the Internet.

A single security group can be applied to the Management Node and all Conferencing Nodes. However, if you want to apply further restrictions to your Management Node (for example, to exclude the TCP/UDP signaling and media ports), then you can configure additional security groups and use them as appropriate for each Azure instance.

Remember that the Management Node and all Conferencing Nodes must be able to communicate with each other. If your instances only have private addresses, ensure that the necessary external systems such as NTP and DNS servers are routable from those nodes.

For further information on the ports and protocols specified here, see Pexip Infinity port usage guide.

Azure Resource Manager (ARM) templates for deploying a security group

Pexip provides two ARM templates — one with, and one without, SIP UDP access enabled — which may be used to deploy a security group containing the above rules. These templates may be used from PowerShell or the Azure CLI. Alternatively, you may use the Azure Portal to deploy a security group using the relevant template (see below).

The details of each template are as follows. You should pick the one most suitable for your needs.

Name	SIP UDP access	Template URL	Resources created
security-group	Disabled	https://pexipas.blob.core.windows.net/templates/20171026/security-group.json (launch in Azure Portal)	Network security group
security-group-with-sip-udp	Enabled	https://pexipas.blob.core.windows.net/templates/20171026/security-group-with-sip-udp.json (launch in Azure Portal)	Network security group

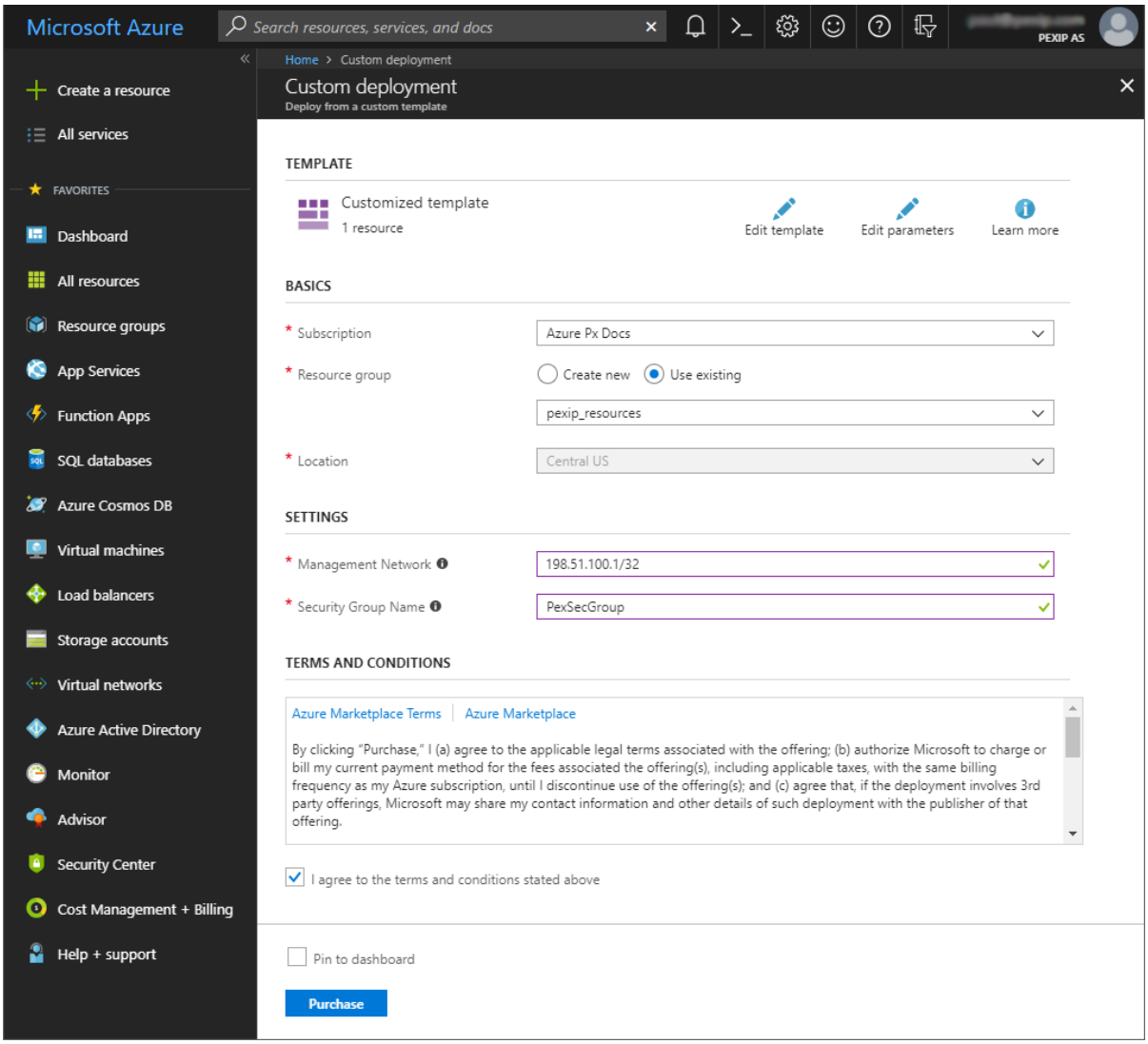
Both templates contain the following parameters:

Name	Description
managementNetwork	Network from which to permit management traffic (CIDR notation e.g. 198.51.100.1/32).
securityGroupName	Name of the security group to create.

Creating a Network Security Group via the Azure portal

To set up a Network Security Group via the Azure portal:

1. Select the appropriate ARM Template URL link from the table above and sign in to the Azure portal if required.
2. Select your **Subscription** and **Resource group**.
3. Supply the template parameters (Settings):
   - The **Management Network** from which to permit management traffic (CIDR notation e.g. 198.51.100.1/32).
   - The **Security Group Name**.
4. Agree to the legal terms.
5. Select **Purchase**.



Creating VM instances in Azure for your Pexip nodes

To deploy a Pexip Infinity Management Node or a Conferencing Node within Azure you must create an Azure VM instance and then use that instance to host that Pexip node.

This section describes how to create a VM instance from a prepared disk image. You must create a separate VM instance for each Pexip node.

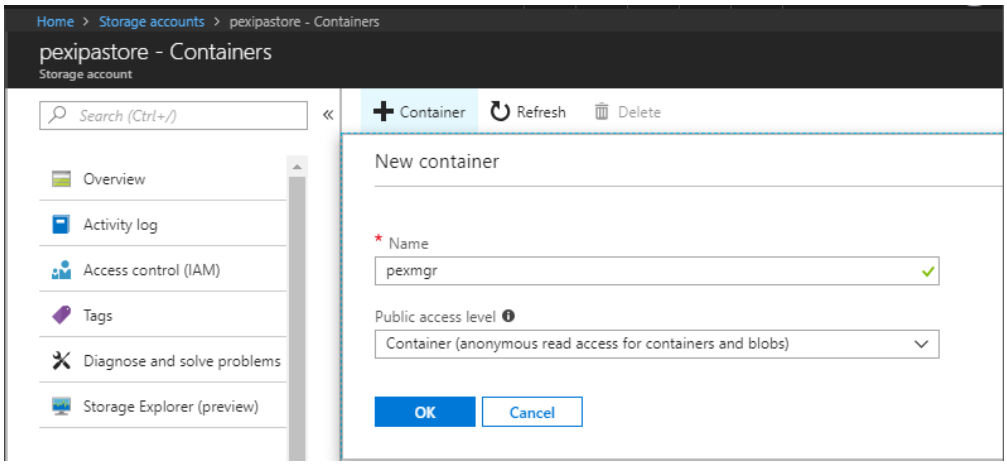
Note that you must deploy the Management Node in your Pexip Infinity platform (either on-premises or in Azure) before deploying any Conferencing Nodes. You must also have prepared your Azure environment.

Creating a VM instance

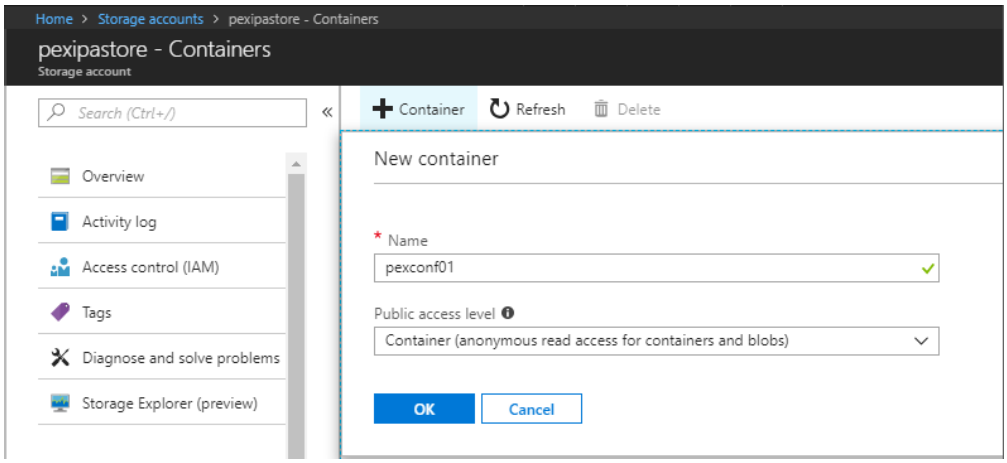
You follow the same basic procedure for creating a VM instance in Azure regardless of whether you are creating the Management Node or a Conferencing Node. The main differences are the VM Image URI and VM size that you need to use.

To create the VM instance:

1. In the **same storage account** that contains the prepared disk images, create a new **storage container** to hold the instance disk:
   a. Go to **Storage Accounts**, select your existing storage account, select **Containers**, then select **+ Container**.
   b. Enter a **Name** such as pexmgr (for the Management Node container) or pexconf01, pexconf02 etc (for a Conferencing Node container).
   c. Specify a **Public access level** of ***Container***.
   d. Select **OK**.



Container for the Management Node

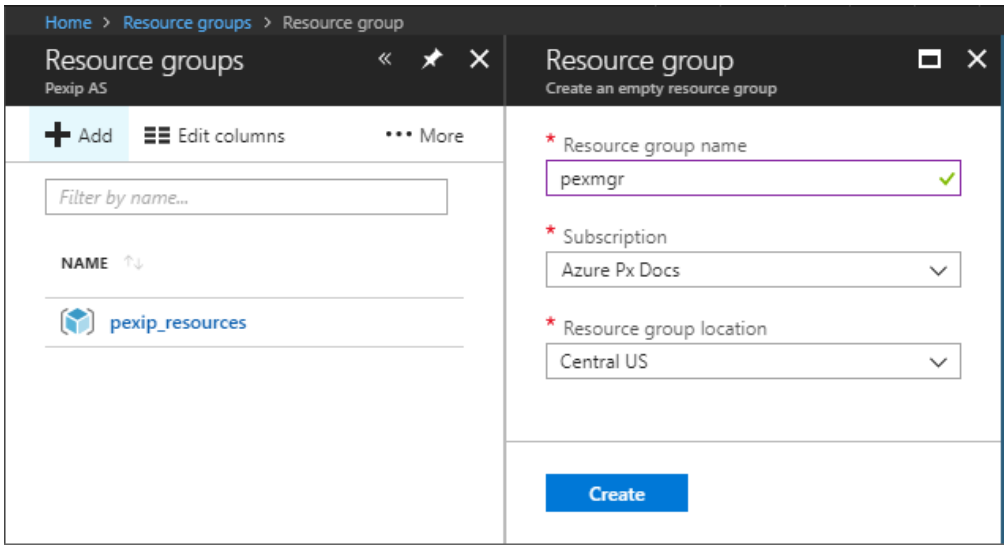


Container for a Conferencing Node

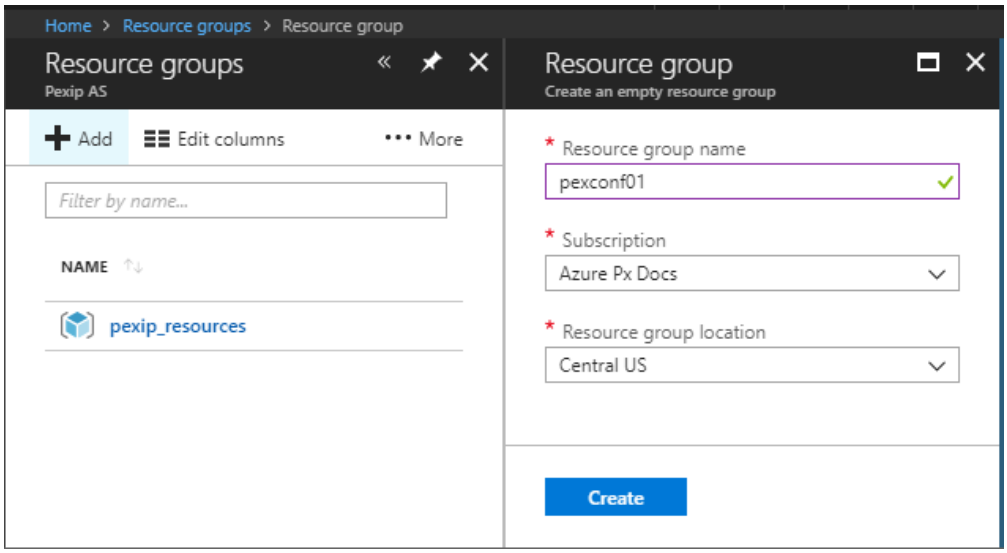
2. Create a new **resource group** to hold the instance.

   Each VM instance (i.e. the Management Node and each Conferencing Node) must be deployed in its own storage container and its own resource group.

   a. Go to **Resource Groups** and select **+ Add**.
   b. Enter your **Resource group name**, such as pexmgr (for the Management Node container) or pexconf01, pexconf02 etc (for a Conferencing Node container).
   c. Select your **Subscription** and your **Resource group location**, and then select **Create**.



Resource Group for the Management Node

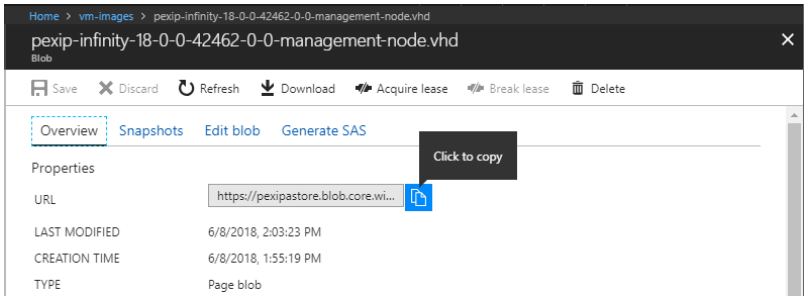
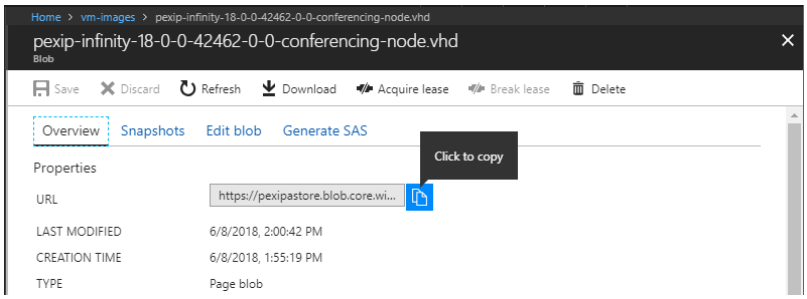


Resource Group for a Conferencing Node

3. Deploy the instance into the new storage container and resource group. This procedure describes how to do this via the Azure portal using an ARM template provided by Pexip:
   a. Decide which ARM template to use from the table below, and then use the "launch in Azure Portal" link to launch the template (after signing in to Azure if necessary).

b. Enter the **Basics** by selecting your **Subscription** and **Resource group**.

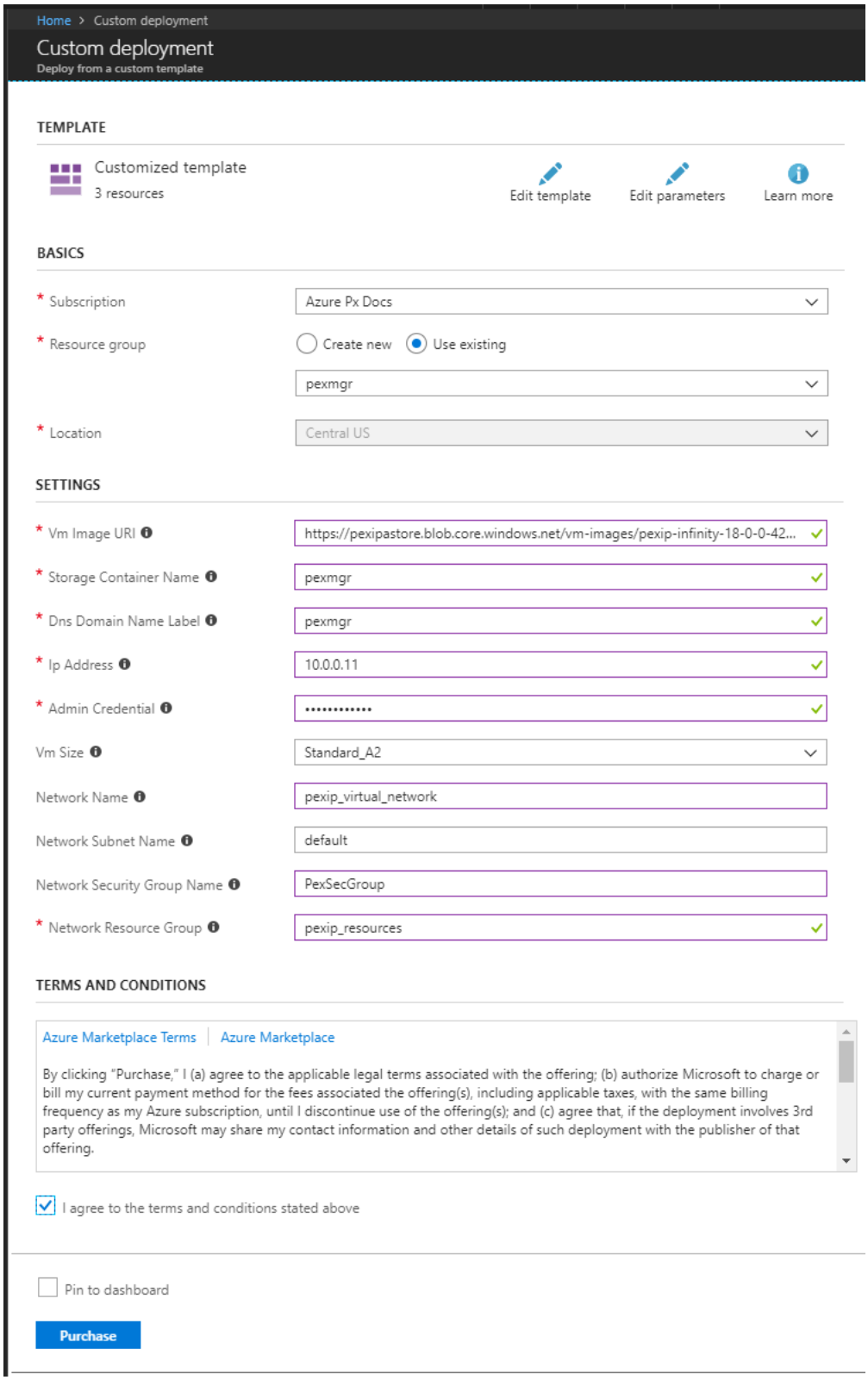
c. Complete the **Settings** (these are based on the template's parameters):

Name	Description
Vm Image URI	The URI of the prepared VM disk image (e.g. https://mystorageaccount.blob.core.windows.net/vm-images/my-machine-image.vhd). This should refer to either the Management Node or Conferencing Node prepared disk image, depending on which type of node you want to install. This URI can be obtained from the Azure portal by selecting **Storage account > Containers > vm-images** and then selecting either your Management Node or Conferencing Node image, and then copying the displayed **URL**.  ***Management Node image URI***  ***Conferencing Node image URI***
Storage Container Name	The name of the storage container you created earlier to hold the instance disk.
Dns Domain Name Label	The domain name label (i.e. hostname) for the Virtual Machine. When deploying an instance with a public IP address, this label must be the host part only — the name must not contain any periods.
Ip Address	The statically-assigned private IP address for the Virtual Machine. This must be within the IP range of your Virtual Network. Note that the first and last IP addresses of each subnet are reserved for protocol conformance, along with the x.x.x.1-x.x.x.3 addresses of each subnet, which are used for Azure services.
Admin Credential	For password-based authentication templates, this is the password for logging into the Virtual Machine. Note that Azure requires a strong password (such as a mix of upper case, lower case and numeric characters). For SSH key-based templates, this is the public SSH key for logging into the Virtual Machine (e.g. ssh-rsa AAA.... user@host).

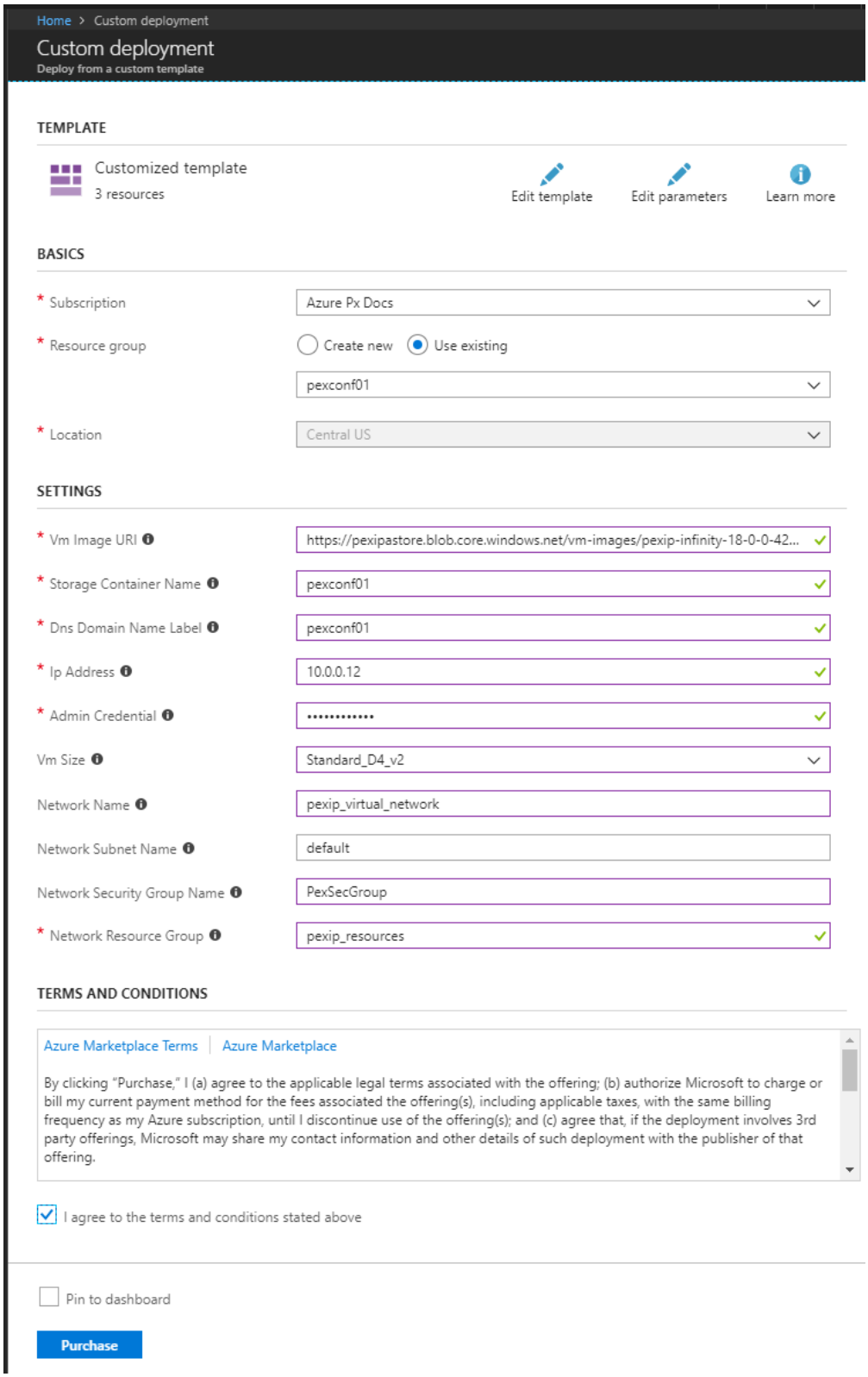
Name	Description
Vm Size	The size of the Virtual Machine (typically Standard A3 for a Management Node or F8 v2 for a Transcoding Conferencing Node). You can change the **Size** later from within the Azure portal (Virtual Machine settings), if required. See Recommended instance types and call capacity guidelines for more information. Azure regularly updates the instance types that can be deployed in each region. Therefore, you may receive a "The requested size for resource <name> is currently not available in location <region> zones" style error when deploying a node. See Troubleshooting instance size errors for help in resolving these errors.
Network Name	The name of the Virtual Network to connect to.
Network Subnet Name	The name of the Virtual Network subnet in which to place the Virtual Machine.
Network Security Group Name	The name of the Network Security Group to apply to the Virtual Machine.
Network Resource Group	The name of the Resource Group containing the Virtual Network and Network Security Group.

d. Review and confirm the Terms and Conditions.

e. Select **Purchase** to deploy the instance.



Management Node VM deployment



Conferencing Node VM deployment

4. It can sometimes take several minutes for your instance to be created and start running. You can use the Azure portal to monitor the status of your new instance.

If the instance deployment fails, review the Azure event diagnostics to help identify the problem.

5. You can now complete the deployment of the Pexip node.
   See either Deploying a Management Node in Azure or Deploying a Conferencing Node in Azure as appropriate.

Azure Resource Manager (ARM) templates for deploying a VM instance

Pexip provides ARM templates which may be used to deploy a VM instance into a resource group (as described above). These templates may be used from PowerShell or the Azure CLI. Alternatively, you can use the templates to deploy an instance via the Azure Portal.

The templates allow you to choose whether to deploy an instance with either password-based or SSH-key based authentication, and with either no public address or a static public address. Every template enables boot diagnostics for the Virtual Machine instance.

Pick the template most suitable for your needs. Every template can be used to launch either a Management Node or a Conferencing Node instance.

Name	Authentication	Public IP address	Template URL	Resources created
virtual-machine-password-static-public-ip	Password based	Yes (static)	https://pexipas.blob.core.windows.net/templates/20180924/virtual-machine-password-static-public-ip.json (launch in Azure Portal)	• Public IP address • Network interface • Virtual Machine
virtual-machine-password-no-public-ip	Password based	No	https://pexipas.blob.core.windows.net/templates/20180924/virtual-machine-password-no-public-ip.json (launch in Azure Portal)	• Network interface • Virtual Machine
virtual-machine-sshkey-static-public-ip	SSH key based	Yes (static)	https://pexipas.blob.core.windows.net/templates/20180924/virtual-machine-sshkey-static-public-ip.json (launch in Azure Portal)	• Public IP address • Network interface • Virtual Machine
virtual-machine-sshkey-no-public-ip	SSH key based	No	https://pexipas.blob.core.windows.net/templates/20180924/virtual-machine-sshkey-no-public-ip.json (launch in Azure Portal)	• Network interface • Virtual Machine

Troubleshooting instance size errors

Azure regularly updates the instance types that can be deployed in each region. Therefore, you may receive a "The requested size for resource <name> is currently not available in location <region> zones" style error when deploying a node.

To resolve these types of errors you can edit the Pexip ARM template to add in a supported instance type for that region:

1. Obtain a list of available compute resources for your region:
   a. Run a PowerShell command in the format `Get-AzureRmComputeResourceSku | where {$_.Locations -icontains "<region name>"}`
      For example: `Get-AzureRmComputeResourceSku | where {$_.Locations -icontains "uksouth"}`
      This will output a list of resources, including VirtualMachines, for example:

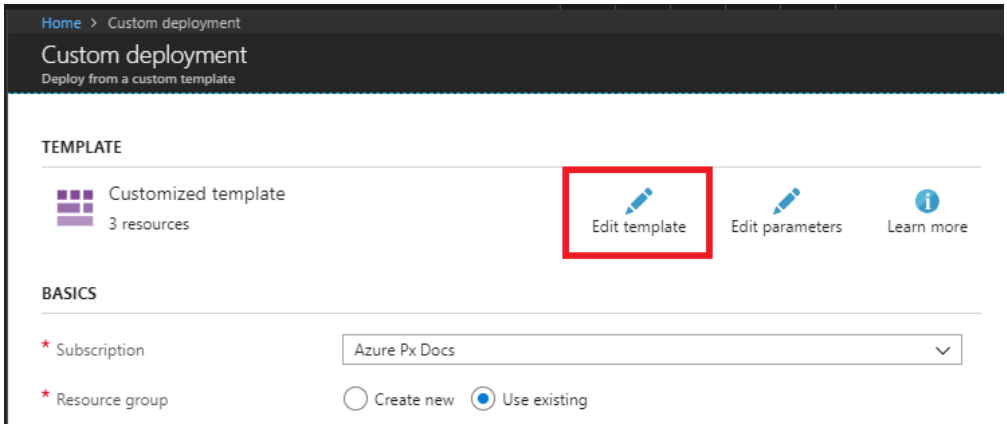
```
virtualMachines        Standard_E32s_v3  uksouth       NotAvailableForSubscription            MaxResourceVolumeMB ...8
virtualMachines     Standard_E64-16s_v3  uksouth       NotAvailableForSubscription            MaxResourceVolumeMB ...6
virtualMachines     Standard_E64-32s_v3  uksouth       NotAvailableForSubscription            MaxResourceVolumeMB ...6
virtualMachines       Standard_E64is_v3  uksouth       NotAvailableForSubscription            MaxResourceVolumeMB ...6
virtualMachines       Standard_E64s_v3   uksouth       NotAvailableForSubscription            MaxResourceVolumeMB ...6
virtualMachines            Standard_H8   uksouth                                              MaxResourceVolumeMB ...0
virtualMachines           Standard_H16   uksouth                                              MaxResourceVolumeMB ...0
virtualMachines           Standard_H8m   uksouth                                              MaxResourceVolumeMB ...0
virtualMachines          Standard_H16m   uksouth                                              MaxResourceVolumeMB ...0
virtualMachines          Standard_H16r   uksouth                                              MaxResourceVolumeMB ...0
virtualMachines         Standard_H16mr   uksouth                                              MaxResourceVolumeMB ...0
virtualMachines        Standard_F2s_v2   uksouth                                              MaxResourceVolumeMB ...4
virtualMachines        Standard_F4s_v2   uksouth                                              MaxResourceVolumeMB ...8
virtualMachines        Standard_F8s_v2   uksouth                                              MaxResourceVolumeMB ...6
```

b. Identify a resource name that is **not** tagged as "NotAvailableForSubscription", such as **Standard_F2s_v2** in this example.

2. Edit the Pexip ARM template and add in the resource that you have identified above:

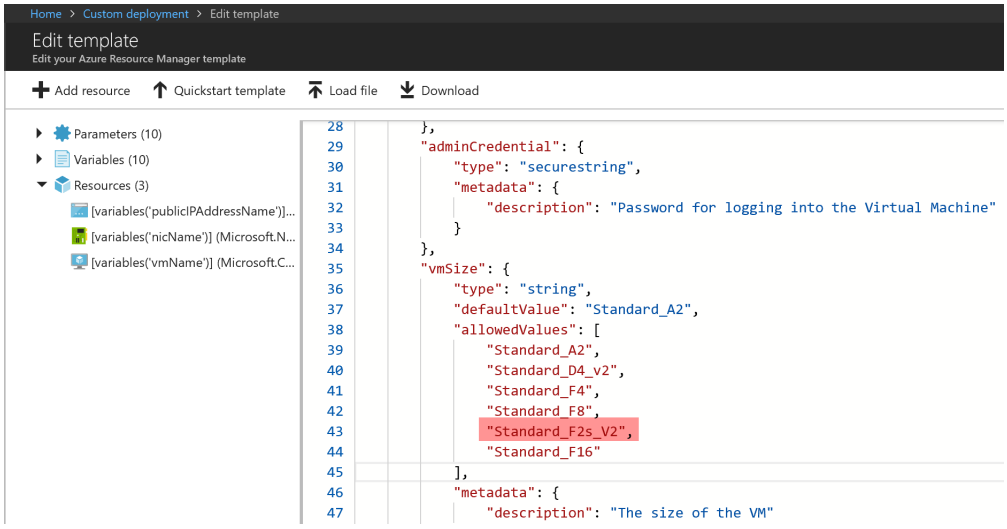
a. Start to deploy the instance into the new storage container and resource group, using the "launch in Azure Portal" link to launch the template as usual.

b. Select **Edit template**.



c. In the **vmSize** section, add in the name of resource that you have identified above, for example **Standard_F2s_v2**.

Ensure that you maintain the correct style and structure of the list, for example:



d. Select **Save**.

You can now continue to deploy your node.

3. Complete the **Settings** as described above, but for **Vm Size** select the name of the resource you added to the template.

You can change the **Size** later from within the Azure portal (Virtual Machine settings), if required.

Obtaining and preparing disk images for Azure deployments

Pexip publishes Azure-optimized disk images for the Management Node and for Conferencing Nodes. This topic includes a PowerShell script that copies and prepares your disk images, and instructions for running the script via a Windows client.

Preparing disk images for use

Before you can use the published Pexip Infinity disk images, you must copy them to your storage account in Azure. This guide refers to a disk image copied to your storage account as a **prepared disk image**. All deployment operations use prepared disk images.

The following PowerShell script copies both of the published disk images (for a Management Node and for a Conferencing Node) to your storage account (into a storage container called **vm-images**) so that they may be used to deploy VM instances. Note that these images can be stored in your shared resource group (whereas each VM instance created from these images must be deployed in its own storage container and its own resource group).

You **must** edit the variables in the script to provide the name of your Azure subscription (`$subscriptionName`), the name of the resource group (`$resourceGroupName`), the name of the storage account (`$storageAccountName`), and the Pexip Infinity version-build number to use (`$version` — currently 20-1-0-45433-0-0 for v20.1 software). If you are running an older version of Pexip Infinity software, see Version information for previous Pexip Infinity releases.

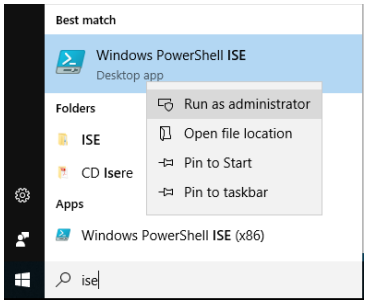
```
# Name of your Azure subscription
$subscriptionName = ""
# Name of the resource group to use
$resourceGroupName = ""
# Name of the storage account to copy the disk images into
$storageAccountName = ""
# Name of the container within the storage account to copy the disk images into
$containerName = "vm-images"
# Version of Pexip Infinity to copy
$version = "20-1-0-45433-0-0"

# Add your Azure account to the PowerShell environment
Add-AzureRmAccount

# Set the current subscription
Get-AzureRmSubscription -SubscriptionName $subscriptionName | Select-AzureRmSubscription

# Obtain the access key for the storage account
$storageAccountKey = Get-AzureRmStorageAccountKey -ResourceGroupName $resourceGroupName -
Name $storageAccountName
If($storageAccountKey.GetType().Name -eq "StorageAccountKeys") {
    # AzureRM.Storage < 1.1.0
    $storageAccountKey = $storageAccountKey.Key1
} Else {
    # AzureRm.Storage 1.1.0
    $storageAccountKey = $storageAccountKey[0].Value
}

# Create the storage access context
$ctx = New-AzureStorageContext -StorageAccountName $storageAccountName -StorageAccountKey
$storageAccountKey

# Ensure that the container exists
New-AzureStorageContainer -Name $containerName -Context $ctx

# Start copying the Management Node image
$mgmt = Start-AzureStorageBlobCopy -AbsoluteUri
"https://pexipas.blob.core.windows.net/infinity/$version/management-node.vhd" -
```

```
DestContainer $containerName -DestBlob "pexip-infinity-$version-management-node.vhd" -
DestContext $ctx

# Start copying the Conferencing Node image
$cnfc = Start-AzureStorageBlobCopy -AbsoluteUri
"https://pexipas.blob.core.windows.net/infinity/$version/conferencing-node.vhd" -
DestContainer $containerName -DestBlob "pexip-infinity-$version-conferencing-node.vhd" -
DestContext $ctx

# Wait for the Management Node image to finish copying
$status = Get-AzureStorageBlobCopyState -Blob $mgmt.Name -Container $containerName -
Context $ctx
While($status.Status -eq "Pending") {
    $status
    $status = Get-AzureStorageBlobCopyState -Blob $mgmt.Name -Container $containerName -
Context $ctx
    Start-Sleep 10
}
$status

# Wait for the Conferencing Node image to finish copying
$status = Get-AzureStorageBlobCopyState -Blob $cnfc.Name -Container $containerName -
Context $ctx
While($status.Status -eq "Pending") {
    $status
    $status = Get-AzureStorageBlobCopyState -Blob $cnfc.Name -Container $containerName -
Context $ctx
    Start-Sleep 10
}
$status

# Print out the prepared disk image URLs for later use
"Prepared Management Node disk image:   " + $mgmt.ICloudBlob.Uri.AbsoluteUri
"Prepared Conferencing Node disk image: " + $cnfc.ICloudBlob.Uri.AbsoluteUri
```

Running the script via your Windows client

To run the PowerShell script from your Windows PC, you need to connect your local machine to Azure:

1. From your PC, run PowerShell ISE as Administrator by right-clicking on it and selecting **Run as Administrator**.



2. When in PowerShell run the following two commands:
   a. `Install-Module AzureRM`
      This installs the Azure Resource Manager modules from the PowerShell Gallery.
   b. `Install-Module Azure`
      This installs the Azure Service Management module from the PowerShell Gallery.

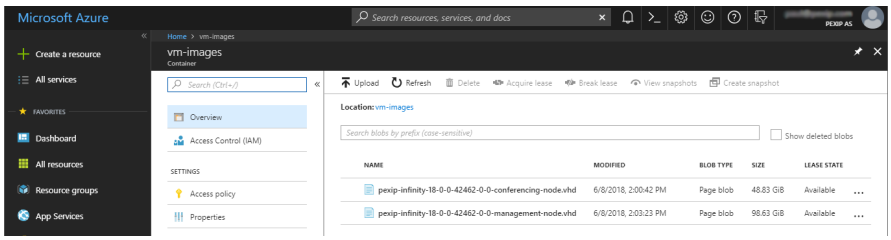
   If you get any prompts while running these commands, select Y for Yes.

See https://docs.microsoft.com/en-gb/powershell/azure/overview?view=azurermps-6.2.0 for more information about Azure PowerShell.

3. Cut and paste the PowerShell script from above into a text editor and add in your Subscription Name, Resource Group Name and Storage Account name (where shown below), and then save the file as a .ps1 file.

```
# Name of your Azure subscription
$subscriptionName = ""
# Name of the resource group to use
$resourceGroupName = ""
# Name of the storage account to copy the disk images into
$storageAccountName = ""
# Name of the container within the storage account to copy the disk images into
$containerName = "vm-images"
```

4. Run the script:
   a. Close the file, right-click on it and select **Run with PowerShell**.
   b. You are prompted to login, so enter your Azure login credentials.
   c. The script will run and copy the images over to your storage account. This takes approximately 10 minutes
5. You can use the Azure portal to confirm when the images have been copied across. From the Azure portal go to your Azure dashboard, select your **Storage Account**, then **Blobs > vm-images**. You will see the Pexip Management Node and Conferencing Node prepared disk images.



Now that you have your prepared disk images in your Azure storage account, you can use them to create the VM instances in Azure in which you can deploy the Pexip Infinity Management Node and Conferencing Nodes.

Version information for previous Pexip Infinity releases

If you are running an older version of Pexip Infinity software, and you want to deploy a new Conferencing Node, you must use a published Pexip Infinity disk image version that corresponds to the software version running on your Management Node. This includes dot releases — so for example, for a v17.3 Management Node you must install a v17.3 Conferencing Node rather than a v17 Conferencing Node. Similarly, if your system has been upgraded since you first installed the Management Node and some Conferencing Nodes, you will need to obtain and prepare the appropriate Conferencing Node image corresponding to the software version you are currently running.

To obtain the published disk images for older software versions (for both the Management Node and for a Conferencing Node) you need to use the appropriate software version number in the PowerShell scripts supplied above.

You must replace the current version number (20-1-0-45433-0-0) with the relevant older version as given in the table below:

Pexip Infinity release	Version number to use in script
v20	20-0-0-45400-0-0
v19	19-0-0-43079-0-0
v18	18-0-0-42462-0-0
v17.3	17-3-0-40234-0-0
v17.2	17-2-0-40212-0-0
v17.1	17-1-0-40171-0-0
v17	17-0-0-40161-0-0

Pexip Infinity release	Version number to use in script
v16.2	16-2-0-37904-0-0
v16.1	16-1-0-37902-0-0
v16	16-0-0-37875-0-0
v15.1	15-1-0-35780-0-0
v15	15-0-0-35724-0-0
v14.2	14-2-0-33750-0-0
v14.1	14-1-0-33745-0-0
v14	14-0-0-33724-0-0
v13	13-0-0-32124-0-0
v12.2	12-2-0-29803-0-0
v12.1	12-1-0-29734-0-0
v12	12-0-0-29682-0-0

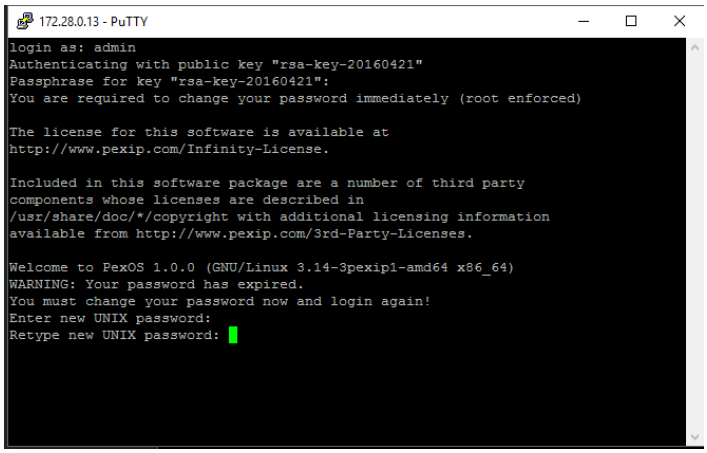
Deploying a Management Node in Azure

As with all Pexip Infinity deployments, you must first deploy the Management Node before deploying any Conferencing Nodes. In a hybrid cloud deployment the Management Node may be deployed in the corporate network or in Azure. This section describes how to deploy the Management Node in Azure.

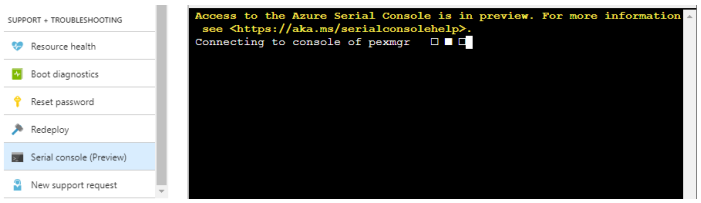
To deploy a Management Node in Azure:

1. Create a VM instance using the prepared Management Node disk image. For more information on this, see:
   - Obtaining and preparing disk images for Azure deployments
   - Creating VM instances in Azure for your Pexip nodes
2. If you are using SSH key-based authentication (instead of password-based authentication), after the Management Node instance has booted, you must SSH into the instance to set the operating system password:
   a. Use an SSH client to access the Management Node by its private IP address, supplying your private key file as appropriate.
      If you cannot access the Management Node, check that you have allowed the appropriate source addresses in your Network Security Group inbound security rules for management traffic.
   b. Follow the login process in the SSH session:
      i. At the login prompt, enter the username ***admin***.
      ii. Supply the key passphrase, if requested.
      iii. At the "Enter new UNIX password:" prompt, enter your desired password, and then when prompted, enter the password again.

This will then log you out and terminate your SSH session.



3. Complete the Pexip Infinity installation wizard:
   a. Connect to the Management Node:
      - If using key-based authentication: reconnect over SSH into the Management Node instance by its private IP address.
      - If using password-based authentication: you can connect via the Azure portal's serial console — go to **Virtual Machines**, select your Management Node VM instance, and then select **Serial console**.



   b. Log in as ***admin***, and — if you are using password-based authentication — enter the **password** you supplied when using the ARM template.

      You are presented with another login prompt:

      ```
      [sudo] password for admin:
      ```

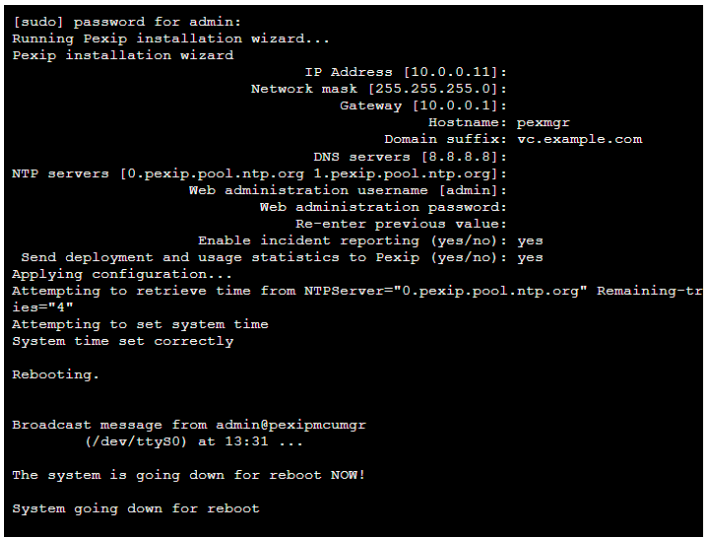
   c. Enter the operating system administrator password (for SSH key-based authentication this is the password you just created in the previous step; for password-based authentication you must enter again the password you supplied when using the ARM template).

      The Pexip installation wizard will begin after a short delay.

   d. Complete the installation wizard to apply basic configuration to the Management Node:
      i. Accept the defaults for the **IP address**, **Network mask** and **Gateway** settings.
      ii. Enter your required **Hostname** and **Domain suffix** for the Management Node.
      iii. Configure one or more **DNS servers** and **NTP servers**. You must override the default values if it is a private deployment.
      iv. Set the **Web administration username** and **password**.
      v. Select whether to **Enable incident reporting** and whether to **Send deployment and usage statistics to Pexip**.

      The DNS and NTP servers at the default addresses are only accessible if your instance has a public IP address. The installation wizard will fail if the NTP server address cannot be resolved and reached.

After successfully completing the wizard, the SSH connection will be lost as the Management Node reboots.



4. After a few minutes you will be able to use the Pexip Infinity Administrator interface to access and configure the Management Node (remember to use https to connect to the node if you have only configured https access rules in your security group).
   You can now configure your Pexip Infinity platform licenses, VMRs, aliases, locations etc, and add Conferencing Nodes.

Deploying a Conferencing Node in Azure

After deploying the Management Node you can deploy one or more Conferencing Nodes in Azure to provide conferencing capacity.

To deploy a Conferencing Node in Azure:

1. Create a VM instance using the prepared Conferencing Node disk image. For more information on this, see:
   - Obtaining and preparing disk images for Azure deployments
   - Creating VM instances in Azure for your Pexip nodes

   Each Conferencing Node VM instance must be deployed in its own storage container and its own resource group.
2. After the instance has booted, perform a configuration-only deployment on the Management Node to inform it of the new Conferencing Node:
   a. Log in to the Pexip Infinity Administrator interface on the Management Node.
   b. Go to **Platform Configuration > Conferencing Nodes**.
   c. Select **Add Conferencing Node**.
   d. For deployment type, choose ***Generic (configuration-only)*** and select **Next**.
   e. Enter the details of the new Conferencing Node, including:

IPv4 address	Enter the Private IP address that you assigned to the VM instance (the ipAddress ARM template parameter).
Network mask	Ensure that the mask matches the one defined for the subnet selected for the instance (the networkSubnetName ARM template parameter). For example a subnet with a /20 prefix size has a network mask of 255.255.240.0.
Gateway IP address	The gateway address is the first usable address in the subnet selected for the instance (e.g. 172.31.0.1 for a 172.31.0.0/20 subnet).
IPv4 static NAT address	Configure the Conferencing Node's static NAT address, if you have a assigned a public/external IP address to the instance. Enter the public address allocated by Azure.

   You must also specify other fields such as the **Name**, **Role**, **Hostname**, **Domain**, **System location** and assign a **TLS certificate**. For a full list of configuration fields, see https://docs.pexip.com/admin/deploy_vm_template.htm#deployment.
   f. Select **Finish**.

g. Select **Download Conferencing Node Configuration** and save the XML configuration file.

A zip file with the name **pexip-<hostname>.<domain>.xml** will be downloaded.

3. You must now upload the XML configuration file to the new Conferencing Node:
   a. Browse to **https://<conferencing-node-ip-address>:8443/** and use the form provided to upload the XML configuration file to the Conferencing Node VM.

   If you cannot access the Conferencing Node, check that you have allowed the appropriate source addresses in your security group inbound rules for management traffic. In public deployments and where there is no virtual private network, you need to use the public address of the node.

      i. Select **Choose File** and select the XML configuration file.

      ii. Select **Upload**.

   b. The Conferencing Node will apply the configuration and then reboot. When it has rebooted, it will connect to the Management Node.

   You can close the browser window used to upload the file.

After deploying a new Conferencing Node, it takes approximately 5 minutes before the node is available for conference hosting and for its status to be updated on the Management Node. Until it is available, the Management Node will report the status of the Conferencing Node as having a last contacted and last updated date of "Never". "Connectivity lost between nodes" alarms relating to that node may appear temporarily.

Configuring dynamic bursting to the Microsoft Azure cloud

Pexip Infinity deployments can burst into the Microsoft Azure cloud when primary conferencing capabilities are reaching their capacity limits, thus providing additional temporary Conferencing Node resources.

This provides the ability to dynamically expand conferencing capacity whenever scheduled or unplanned usage requires it. The Azure cloud Conferencing Nodes instances are only started up when required and are automatically stopped again when capacity demand normalizes, ensuring that Azure costs are minimized.

For complete information about dynamic bursting, see Dynamic bursting to a cloud service.

Configuring your system for dynamic bursting to Microsoft Azure

These instructions assume that you already have a working Pexip Infinity platform, including one or more primary (always on) Conferencing Nodes in one or more system locations. These existing Conferencing Nodes can be deployed using whichever platform or hypervisor you prefer.

Firewall addresses/ports required for access to the Azure APIs for cloud bursting

Access to the Microsoft Azure APIs for cloud bursting is only required from the Management Node.

The Management Node always connects to destination port 443 over HTTPS.

DNS is used to resolve the Azure API addresses. Currently, Pexip Infinity uses the following DNS FQDNs (but these may change in the future):

- login.microsoftonline.com
- management.azure.com

Setting up your bursting nodes in Microsoft Azure and enabling bursting in Pexip Infinity

You must deploy in Azure the Conferencing Nodes that you want to use for dynamic bursting, and then configure those nodes in Pexip Infinity as the overflow destination for your primary (always on) Conferencing Nodes:

1. In Pexip Infinity, configure a new "overflow" system location e.g. "Azure burst", that will contain your bursting Conferencing Nodes.

(Note that system locations are not explicitly configured as "primary" or "overflow" locations. Pexip Infinity automatically detects the purpose of the location according to whether it contains Conferencing Nodes that may be used for dynamic bursting.)

2. In Azure, set up an Active Directory (AD) application and assign the required permissions to it that will allow the Pexip Infinity Management Node to log in to Azure and start and stop the node instances.

   See Configuring an Active Directory (AD) application and permissions for controlling overflow nodes for more information.

3. Deploy in Azure the Conferencing Nodes that you want to use for dynamic bursting. Deploy these nodes in the same manner as you would for "always on" usage (see Deploying a Conferencing Node in Azure), except:

   a. Apply to each cloud VM node instance to be used for conference bursting a tag with a **Key** of **pexip-cloud** and an associated **Value** set to the **Tag value** that is shown in the **Cloud Bursting** section on the **Platform Configuration > Global Settings** page.

      This tag indicates which VM nodes will be started and shut down dynamically by your Pexip system.

   b. When adding the Conferencing Node within Pexip Infinity:

      i. Assign the Conferencing Node to the overflow system location (e.g. "Azure burst").

      ii. Disable (uncheck) the **Enable distributed database** setting (this setting should be disabled for any nodes that are not expected to always be available).

   c. After the Conferencing Node has successfully deployed, manually stop the node instance on Azure.

4. In Pexip Infinity, go to **Platform Configuration > Global Settings**, enable cloud bursting and then configure your bursting threshold, minimum lifetime and other appropriate settings for Azure:

Option	Description
Enable bursting to the cloud	Select this option to instruct Pexip Infinity to monitor the system locations and start up / shut down overflow Conferencing Nodes hosted in your cloud service when in need of extra capacity.
Bursting threshold	The bursting threshold controls when your dynamic overflow nodes in your cloud service are automatically started up so that they can provide additional conferencing capacity. When the number of additional HD calls that can still be hosted in a location reaches or drops below the threshold, it triggers Pexip Infinity into starting up an overflow node in the overflow location. See Configuring the bursting threshold for more information.
Tag name and Tag value	These read-only fields indicate the tag name (always **pexip-cloud**) and associated tag value (the hostname of your Management Node) that you must assign to each of your cloud VM node instances that are to be used for dynamic bursting.
Minimum lifetime	An overflow cloud bursting node is automatically stopped when it becomes idle (no longer hosting any conferences). However, you can configure the **Minimum lifetime** for which the bursting node is kept powered on. By default this is set to 50 minutes, which means that a node is never stopped until it has been running for at least 50 minutes. If your service provider charges by the hour, it is more efficient to leave a node running for 50 minutes — even if it is never used — as that capacity can remain on immediate standby for no extra cost. If your service provider charges by the minute you may want to reduce the **Minimum lifetime**.
Cloud provider	Select ***Azure***.
Azure subscription ID	The ID of your Azure subscription.
Azure client ID	The ID used to identify the client (sometimes referred to as Application ID).
Azure secret key	The Azure secret key that is associated with the Azure client ID.
Azure tenant ID	The Azure tenant ID that is associated with the Azure client ID.

5. Go to **Platform Configuration > Locations** and configure the system locations that contain your "always on" Conferencing Nodes (the nodes/locations that initially receive calls) so that they will overflow to your new "Azure burst" location when necessary.

   When configuring your "always on" locations, you must set the **Primary overflow location** to the bursting location containing your overflow nodes. (Automatic bursting, and the stopping and starting of overflow nodes only applies to the **Primary**

overflow location; the **Secondary overflow location** can only be used for standard overflow i.e. to other "always on" nodes.)
You should configure the **Transcoding location** as normal to the location that will initially handle the call media (before overflow/bursting is required).
We recommend that you do not mix your "always on" Conferencing Nodes and your bursting nodes in the same system location.

Configuring an Active Directory (AD) application and permissions for controlling overflow nodes

Within Azure you must set up an Active Directory (AD) application and permissions to be used by Pexip Infinity to start up and shut down the Conferencing Node overflow instances. You need to ensure that your Azure account has sufficient permissions to register an application with your Active Directory, and assign the application to a role in your Azure subscription.

A summary description of the tasks involved and the required permissions is given below. Full information of how to check your account permissions, create the application and assign a role is available at https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-create-service-principal-portal.

1. Create an Active Directory application of type ***Web app / API***. Enter a name such as "**pexip**" for example.
   a. Use the assigned **Application ID** as the **Azure client ID** in the Pexip Infinity **Global Settings** page.
   b. Generate a key for the application, copy its value and use it as the **Azure secret key** in the **Global Settings** page.
2. Lookup the Directory ID in the properties of your Azure Active Directory and use it as the **Azure tenant ID** in the **Global Settings** page.
3. Assign the Active Directory application to a role. Typically you will assign a role at the Subscriptions level.
   Select **Access Control (IAM) > Add**, select the role you want to assign, and then search for and select your application e.g. "pexip".
   Azure contains many built-in roles; the most appropriate built-in role to use for dynamic bursting is ***DevTest Labs User*** (https://docs.microsoft.com/en-us/azure/active-directory/role-based-access-built-in-roles#devtest-labs-user).
   If you want to create your own custom role (https://docs.microsoft.com/en-us/azure/active-directory/role-based-access-control-custom-roles), the permissions required by Pexip Infinity are:

Actions (permissions)	Allows Pexip Infinity to...
Microsoft.Authorization/*/read	Read roles and role assignments.
Microsoft.Compute/virtualMachines/*/read	Read the properties of a virtual machine (VM sizes, runtime status, VM extensions, etc).
Microsoft.Compute/virtualMachines/deallocate/action	Deallocate virtual machines.
Microsoft.Compute/virtualMachines/read	Read the properties of a virtual machine.
Microsoft.Compute/virtualMachines/start/action	Start virtual machines.

Configuring the bursting threshold

When enabling your platform for cloud bursting the most important decision you must make is the level at which to set the bursting threshold:

- The bursting threshold controls when your dynamic overflow nodes in your cloud service are automatically started up so that they can provide additional conferencing capacity. When the number of additional HD calls that can still be hosted in a location reaches or drops below the threshold, it triggers Pexip Infinity into starting up an overflow node in the overflow location.
  For example, setting the threshold to 5 means that when there are 5 or fewer HD connections still available in a location, an overflow node will be started up.
- When an overflow location reaches the bursting threshold i.e. the number of additional HD calls that can still be hosted on the Conferencing Nodes in the overflow location reaches the threshold, another overflow node in that location is started up, and so on.

Note that the current number of free HD connections in the original location are not taken into account when seeing if the overflow location needs to overflow further — however, new calls will automatically use any available media resource within those original primary locations that has become available.

- The bursting threshold is a global setting — it applies to every system location in your deployment.
- Note that it takes approximately 5 minutes for a dynamic Conferencing Node instance to start up and become available for conference hosting. If your primary deployment reaches full capacity, and the overflow nodes have not completed initiating, any incoming calls during this period will be rejected with "capacity exceeded" messages. You have to balance the need for having standby capacity started up in time to meet the expected demand, against starting up nodes too early and incurring extra unnecessary costs.

Manually starting an overflow node

If you know that your system will need additional capacity at a specific time due to a predictable or scheduled spike in demand, but do not want to wait for the bursting threshold to be triggered before starting up the overflow nodes, you can manually start up any of your overflow nodes.

Do not manually start an overflow node too early. If you manually start up a node more than the **Minimum lifetime** minutes before the node is needed, it will most probably get automatically stopped again before it is used.

You can start overflow nodes via the management API or via the Administrator interface:

- **via the management API**: the cloud_node status resource can be used to list all of the available overflow nodes, the cloud_monitored_location and cloud_overflow_location resources retrieve the current load on the primary locations and any currently active overflow locations respectively, and the start_cloudnode resource can be used to manually start up any overflow node. This means that a third-party scheduling system, for example, could be configured to start up the overflow nodes via the management API approximately 10 minutes before a large conference is due to start.

  For example, let's assume that you have:
  - a regular spike in conferencing capacity demand at 9:00am every morning
  - an even usage of about 20% of that spike level during the rest of the day
  - a 30:70 ratio between your "always on" capacity and your overflow cloud capacity

  we would recommend:
  - configuring a low bursting threshold, such as 10-20% of your "always on" capacity (i.e. if your "always on" capacity is 80 HD calls, then set the bursting threshold to 12)
  - getting your scheduling system to call the API to manually start up all of your overflow cloud nodes at 8:50am on weekdays.
- **via the Pexip Infinity Administrator interface**: go to **Status > Cloud Bursting** and select **Start** for the required nodes (the **Start** option is in the final column of the **Cloud overflow nodes** table).

Converting between overflow and "always on" Microsoft Azure Conferencing Nodes

If you need to convert an existing "always on" Azure Conferencing Node into an overflow node:

1. In Azure:
   a. Apply a tag with a **Key** of **pexip-cloud** and an associated **Value** set to the **Tag value** that is shown at the bottom of the **Platform Configuration > Global Settings** page.
   b. Manually stop the node instance on Azure.
2. In Pexip Infinity:
   a. Change the system location of the Conferencing Node to the overflow system location (e.g. "Azure burst").
   b. Disable the node's **Enable distributed database** setting.

      You should avoid frequent toggling of this setting. When changing this setting on multiple Conferencing Nodes, update one node at a time, waiting a few minutes before updating the next node.

If you need to convert an existing Azure overflow Conferencing Node into an "always on" node:

1. In Azure:
   a. Remove the tag with a **Key** of **pexip-cloud** from the Azure instance.
   b. Manually start the node instance on Azure.
2. In Pexip Infinity:
   a. Change the system location of the Conferencing Node to a location other than the overflow system location.
   b. Enable the node's **Enable distributed database** setting.

      You should avoid frequent toggling of this setting. When changing this setting on multiple Conferencing Nodes, update one node at a time, waiting a few minutes before updating the next node.

Managing Azure instances

This section describes the common maintenance tasks for stopping, restarting and permanently removing Conferencing Node Azure instances, and provides guidelines for backing up your instances.

Temporarily removing (stopping) a Conferencing Node instance

At any time you can temporarily remove a Conferencing Node instance from your Pexip Infinity platform if, for example, you do not need all of your current conferencing capacity.

To temporarily remove a Conferencing Node instance:

1. Put the Conferencing Node into maintenance mode via the Pexip Infinity Administrator interface on the Management Node:
   a. Go to **Platform Configuration > Conferencing Nodes**.
   b. Select the Conferencing Node(s).
   c. From the **Action** menu at the top left of the screen, select **Enable maintenance mode** and then select **Go**.
   While maintenance mode is enabled, this Conferencing Node will not accept any new conference instances.
   d. Wait until any existing conferences on that Conferencing Node have finished. To check, go to **Status > Live View**.
2. Stop the Conferencing Node instance on Azure:
   a. From the Azure portal, select **Virtual Machines** to see the status of all of your instances.
   b. Select the instance you want to shut down.
   c. Select **Stop** to shut down the instance.

Reinstating (restarting) a stopped Conferencing Node instance

You can reinstate a Conferencing Node instance that has already been installed but has been temporarily shut down.

To restart a Conferencing Node instance:

1. Restart the Conferencing Node instance on Azure:
   a. From the Azure portal, select **Virtual Machines** to see the status of all of your instances.
   b. Select the instance you want to restart.
   c. Select **Start** to start the instance.
2. Take the Conferencing Node out of maintenance mode via the Pexip Infinity Administrator interface on the Management Node:
   a. Go to **Platform Configuration > Conferencing Nodes**.
   b. Select the Conferencing Node.
   c. Clear the **Enable maintenance mode** check box and select **Save**.
3. Update the Conferencing Node's static NAT address, if appropriate.
   If your Conferencing Node instance was configured with an auto-assigned public IP address, it will be assigned a new public IP address when the instance is restarted.
   a. Go to **Platform Configuration > Conferencing Nodes** and select the Conferencing Node.
   b. Configure the **Static NAT address** as the instance's new public IP address.

After reinstating a Conferencing Node, it takes approximately 5 minutes for the node to reboot and be available for conference hosting, and for its last contacted status to be updated on the Management Node.

Permanently removing a Conferencing Node instance

If you no longer need a Conferencing Node instance, you can permanently delete it from your Pexip Infinity platform.

To remove a Conferencing Node instance:

1. If you have not already done so, put the Conferencing Node into maintenance mode via the Pexip Infinity Administrator interface on the Management Node:
   a. Go to **Platform Configuration > Conferencing Nodes**.
   b. Select the Conferencing Node(s).
   c. From the **Action** menu at the top left of the screen, select **Enable maintenance mode** and then select **Go**.
      While maintenance mode is enabled, this Conferencing Node will not accept any new conference instances.
   d. Wait until any existing conferences on that Conferencing Node have finished. To check, go to **Status > Live View**.
2. Delete the Conferencing Node from the Management Node:
   a. Go to **Platform Configuration > Conferencing Nodes** and select the Conferencing Node.
   b. Select the check box next to the node you want to delete, and then from the **Action** drop-down menu, select **Delete selected Conferencing Nodes** and then select **Go**.
3. Terminate the Conferencing Node instance on Azure:
   a. Delete the resource group which holds the instance:
      i. From the Azure Portal, select **Resource Groups** to see all of your resource groups.
      ii. Select the resource group you want to delete.
      iii. Select **Delete** to delete the resource group.
   b. Delete the storage container which holds the instance disk:
      i. From the Azure Portal, select **All Resources** to see all of your storage accounts.
      ii. Select the storage account that is being used to store the instance disk.
      iii. Under **Services**, select **Blobs** to see all of the storage containers in the storage account.
      iv. Select the storage container you want to delete.
      v. Select **Delete** to delete the storage container.
   c. If boot diagnostics were enabled on the VM instance, delete the storage container which holds the instance boot diagnostics logs:
      i. From the Azure Portal, select **All Resources** to see all of your storage accounts.
      ii. Select the storage account that is being used to store the boot diagnostics logs.
      iii. Under **Services**, select **Blobs** to see all of the storage containers in the storage account.
      iv. Select the storage container you want to delete.
      v. Select **Delete** to delete the storage container.

Backing up VM instances

When backing up Azure VMs to a Recovery Services vault, you must shut down the VM before performing the backup.