Azure Platform Image Verification

Cloud Volumes ONTAP

NetApp
June 27, 2024

Table of Contents

Azure Platform Image Verification

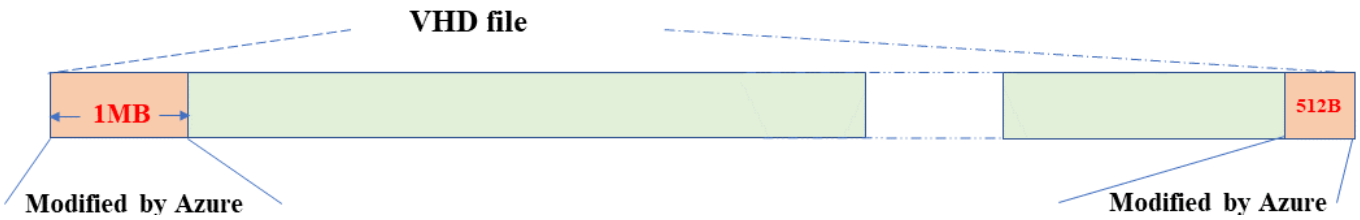
Azure image verification overview

Azure image verification complies with enhanced NetApp security requirements. While verifying an image file is a straightforward process, Azure image signature verification does require special handlings to the well-known Azure VHD image file due to an alternation made by the Azure marketplace.

Azure image verification is supported on Cloud Volumes ONTAP software version 9.15.0 or greater.

Azure's alteration of published VHD files

The leading 1MB(1048576 bytes) and ending 512 bytes of VHD file is modified by Azure. NetApp image signing skips the leading 1MB and ending 512 Bytes and signs the remaining VHD image portion.



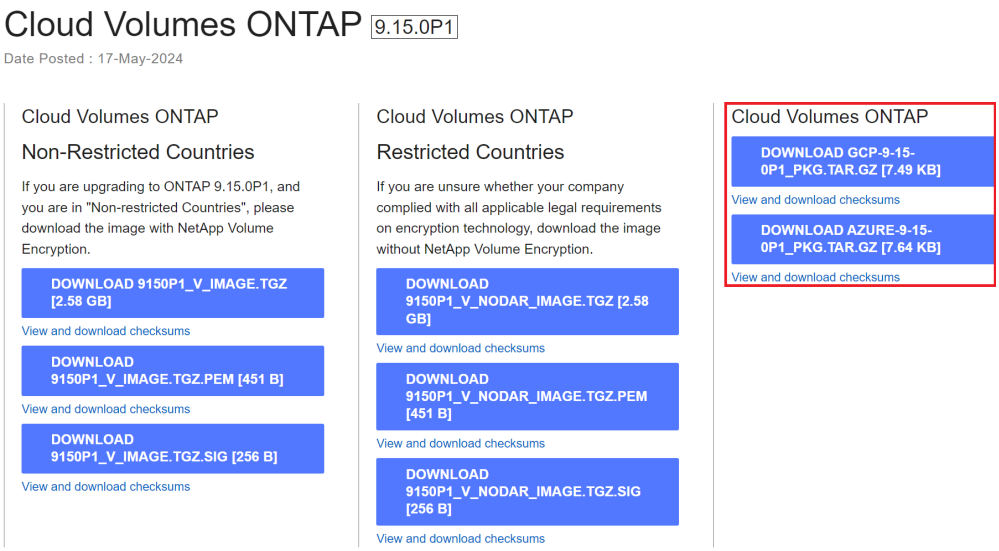
As an example, the above diagram shows a VHD file sized 10GB. But the NetApp signed portion is marked in green with size of 10GB - 1MB - 512B.

Download the Azure Image Digest File

The Azure Image Digest File can be downloaded from the NetApp Support Site. The download is in tar.gz format and contains files for image signature verification.

Steps

1. Go to the Cloud Volumes ONTAP product page on the NetApp Support Site and download the desired software version under the Downloads section.
2. Under the Cloud Volumes ONTAP download page, click the **download button** for the Azure Image Digest File to download the TAR.GZ file.



3. For Linux and MacOS, you must perform the following to get the md5sum and sha256sum for the downloaded Azure Image Digest file.
   i. For md5sum, enter the `md5sum` command.
   ii. For sha256sum, enter the `sha256sum` command.
4. Verify the `md5sum` and `sha256sum` values match the Azure Image Digest File download.
5. On Linux and Mac OS, perform the `tar -xzf` command to extract the tar.gz file.

   The extracted TAR.GZ file contains the digest file(.sig), public key certificate file(.pem), and chain certificate file(.pem).

List result of untar tar.gz file

```
$ ls cert/ -l
-rw-r----- 1 netapp netapp  384 May  13 13:00 9.15.0P1_azure_digest.sig
-rw-r----- 1 netapp netapp 2365 May  13 13:00 Certificate-
9.15.0P1_azure.pem
-rw-r----- 1 netapp netapp 8537 May  13 13:00 Certificate-Chain-
9.15.0P1_azure.pem
-rw-r----- 1 netapp netapp 8537 May  13 13:00 version_readme
```

Image export from Azure Marketplace

Once the VHD image is published to Azure cloud, the image is no longer managed by NetApp. Instead, the published image is placed on the Azure marketplace. Azure's alteration to the leading 1MB and ending 512B of the VHD occurs when the image is staged and published on the Azure marketplace. To verify the signature of the VHD file, the VHD image modified by Azure needs to be exported from the Azure marketplace first.

What you'll need

You must install the required programs on your system.

- Azure CLI is installed or Azure Cloud Shell through the Azure portal is readily available.

> For more information on how to install Azure CLI, see Azure documentation: How to install Azure CLI.

Steps

1. Map the ONTAP version to the Azure marketplace image version using the content of version_readme file.

   For each version mapping listed in the version_readme file, the ONTAP version is represented by "buildname", and Azure marketplace image version is represented by "version".

   For example, in the following version_readme file, ONTAP version "9.15.0P1" is mapped to Azure marketplace image version "9150.01000024.05090105". This Azure marketplace image version is later used to set the image URN.

   ```
   [
       {
           "buildname": "9.15.0P1",
           "publisher": "netapp",
           "version": "9150.01000024.05090105"
       }
   ]
   ```

2. Identify the region name where you intend to create VMs.

   This region name is used as the value for the "locName" variable when setting the URN of the marketplace image.

   a. To receive a list of available regions, enter the `az account list-locations -o table` command.

      In the table below, the region name is referred to as the "Name" field.

      ```
      $ az account list-locations -o table
      DisplayName               Name                 RegionalDisplayName
      ------------------------  -------------------
      ------------------------------------
      East US                   eastus               (US) East US
      East US 2                 eastus2              (US) East US 2
      South Central US          southcentralus       (US) South Central US
      ...
      ```

3. Review the SKU name for the corresponding VM deployment type from the table below.

   The SKU name is used as the value for the "skuName" variable when setting the URN of the marketplace image.

   For example, Single-Node deployments should use the "ontap_cloud_byol" SKU name.

VM Deployment Type	SKU Name
Single Node	ontap_cloud_byol
High Availability	ontap_cloud_byol_ha

4. Once the ONTAP version and Azure marketplace image are mapped, export the VHD file from Azure marketplace through Azure Cloud Shell or Azure CLI.

Export VHD file through Azure Cloud Shell on Azure portal

1. From Azure Cloud Shell, export the marketplace image to a vhd (image2, e.g. 9150.01000024.05090105.vhd), and download to your local machine (for example, a Linux machine, or a windows PC.)

Click to display

```
#Azure Cloud Shell on Azure portal to get VHD image from Azure
Marketplace
a) Set the URN and other parameters of the marketplace image. URN is
with format "<publisher>:<offer>:<sku>:<version>". Optionally, a
user can list NetApp marketplace images to confirm the proper image
version.
PS /home/user1> $urn="netapp:netapp-ontap-
cloud:ontap_cloud_byol:9150.01000024.05090105"
PS /home/user1> $locName="eastus2"
PS /home/user1> $pubName="netapp"
PS /home/user1> $offerName="netapp-ontap-cloud"
PS /home/user1> $skuName="ontap_cloud_byol"
PS /home/user1> Get-AzVMImage -Location $locName -PublisherName
$pubName -Offer $offerName -Sku $skuName |select version
...
141.20231128
9.141.20240131
9.150.20240213
9150.01000024.05090105
...

b) Create a new managed disk from the Marketplace image with the
matching image version
PS /home/user1> $diskName = “9150.01000024.05090105-managed-disk"
PS /home/user1> $diskRG = “fnf1”
PS /home/user1> az disk create -g $diskRG -n $diskName --image
-reference $urn
PS /home/user1> $sas = az disk grant-access --duration-in-seconds
3600 --access-level Read --name $diskName --resource-group $diskRG
PS /home/user1> $diskAccessSAS = ($sas | ConvertFrom-
Json)[0].accessSas

c) Export a VHD from the managed disk to Azure Storage
Create a container with proper access level. As an example, a
container named 'vm-images' with 'Container' access level is used
here.
Get storage account access key, on Azure portal, 'Storage
Accounts'/'examplesaname'/'Access Key'/'key1'/'key'/'show'/<copy>.
PS /home/user1> $storageAccountName = “examplesaname”
PS /home/user1> $containerName = “vm-images”
PS /home/user1> $storageAccountKey = "<replace with the above access
key>"
PS /home/user1> $destBlobName = “9150.01000024.05090105.vhd”
PS /home/user1> $destContext = New-AzureStorageContext
```

```
-StorageAccountName $storageAccountName -StorageAccountKey
$storageAccountKey
PS /home/user1> Start-AzureStorageBlobCopy -AbsoluteUri
$diskAccessSAS -DestContainer $containerName -DestContext
$destContext -DestBlob $destBlobName
PS /home/user1> Get-AzureStorageBlobCopyState –Container
$containerName –Context $destContext -Blob $destBlobName

d) Download the generated image to your server, e.g., a Linux
machine.
Use "wget <URL of file examplesaname/Containers/vm-
images/9150.01000024.05090105.vhd>".
The URL is organized in a formatted way. For automation tasks, the
following example could be used to derive the URL string. Otherwise,
Azure CLI 'az' command could be issued to get the URL, which is not
covered in this guide. URL Example:
https://examplesaname.blob.core.windows.net/vm-
images/9150.01000024.05090105.vhd

e) Clean up the managed disk
PS /home/user1> Revoke-AzDiskAccess -ResourceGroupName $diskRG
-DiskName $diskName
PS /home/user1> Remove-AzDisk -ResourceGroupName $diskRG -DiskName
$diskName
```

Export VHD file through Azure CLI from local Linux machine

1. Export the marketplace image to a vhd through the Azure CLI from a local Linux machine.

Click to display

```
#Azure CLI on local Linux machine to get VHD image from Azure
Marketplace
a) Login Azure CLI and list marketplace images
% az login --use-device-code
To sign in, use a web browser to open the page
https://microsoft.com/devicelogin and enter the code XXXXXXXXX to
authenticate.

% az vm image list --all --publisher netapp --offer netapp-ontap-
cloud --sku ontap_cloud_byol
...
{
"architecture": "x64",
"offer": "netapp-ontap-cloud",
"publisher": "netapp",
"sku": "ontap_cloud_byol",
"urn": "netapp:netapp-ontap-
cloud:ontap_cloud_byol:9150.01000024.05090105",
"version": "9150.01000024.05090105"
},
...

b) Create a new managed disk from the Marketplace image with the
matching image version
% export urn="netapp:netapp-ontap-
cloud:ontap_cloud_byol:9150.01000024.05090105"
% export diskName="9150.01000024.05090105-managed-disk"
% export diskRG="new_rg_your_rg"
% az disk create -g $diskRG -n $diskName --image-reference $urn
% az disk grant-access --duration-in-seconds 3600 --access-level
Read --name $diskName --resource-group $diskRG
{
  "accessSas": "https://md-
xxxxxx.blob.core.windows.net/xxxxxxx/abcd?sv=2018-03-
28&sr=b&si=xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxx&sigxxxxxxxxxxxxxxxxxxxxxxx"
}

% export diskAccessSAS="https://md-
xxxxxx.blob.core.windows.net/xxxxxxx/abcd?sv=2018-03-
28&sr=b&si=xxxxxxxx-xxxx-xx-xx-xx&sigxxxxxxxxxxxxxxxxxxxxxxx"
#To automate the process, the SAS needs to be extracted from the
standard output. This is not included in this guide.
```

```
c) export vhd from managed disk
Create a container with proper access level. As an example, a
container named 'vm-images' with 'Container' access level is used
here.
Get storage account access key, on Azure portal, 'Storage
Accounts'/'examplesaname'/'Access Key'/'key1'/'key'/'show'/<copy>.
There should be az command that can achieve the same, but this is
not included in this guide.
% export storageAccountName="examplesaname"
% export containerName="vm-images"
% export storageAccountKey="xxxxxxxxxx"
% export destBlobName="9150.01000024.05090105.vhd"

% az storage blob copy start --source-uri $diskAccessSAS
--destination-container $containerName --account-name
$storageAccountName --account-key $storageAccountKey --destination
-blob $destBlobName

{
  "client_request_id": "xxxx-xxxx-xxxx-xxxx-xxxx",
  "copy_id": "xxxx-xxxx-xxxx-xxxx-xxxx",
  "copy_status": "pending",
  "date": "2022-11-02T22:02:38+00:00",
  "etag": "\"0xXXXXXXXXXXXXXXXXX\"",
  "last_modified": "2022-11-02T22:02:39+00:00",
  "request_id": "xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxx",
  "version": "2020-06-12",
  "version_id": null
}

#to check the status of the blob copying
% az storage blob show --name $destBlobName --container-name
$containerName --account-name $storageAccountName

....
    "copy": {
      "completionTime": null,
      "destinationSnapshot": null,
      "id": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxx",
      "incrementalCopy": null,
      "progress": "10737418752/10737418752",
      "source": "https://md-
xxxxxx.blob.core.windows.net/xxxxx/abcd?sv=2018-03-
28&sr=b&si=xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxx",
      "status": "success",
      "statusDescription": null
```

```
    },
....

d) Download the generated image to your server, e.g., a Linux
machine.
Use "wget <URL of file examplesaname/Containers/vm-
images/9150.01000024.05090105.vhd>".
The URL is organized in a formatted way. For automation tasks, the
following example could be used to derive the URL string. Otherwise,
Azure CLI 'az' command could be issued to get the URL, which is not
covered in this guide. URL Example:
https://examplesaname.blob.core.windows.net/vm-
images/9150.01000024.05090105.vhd

e) Clean up the managed disk
az disk revoke-access --name $diskName --resource-group $diskRG
az disk delete --name $diskName --resource-group $diskRG --yes
```

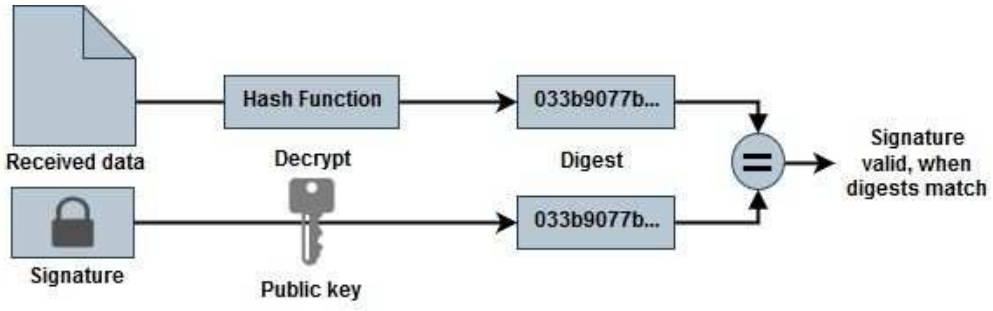
File signature verification

File signature verification

The Azure image verification process will generate a digest from the VHD file with the leading 1MB and ending 512B striped by using hash function. To match the signing procedure, SHA256 is used to hash. You need to remove the leading 1MB and final 512B from the VHD file and then verify the remaining portion of the VHD file.

File signature verification workflow summary

The following is an overview of the file signature verification workflow process.



- Download the Azure Image Digest file from the NetApp Support Site and extract the digest file(.sig), public key certificate file(.pem) and chain certificate file(.pem).

  Refer to the Download the Azure Image Digest File for more information.

- Verify the chain of trust.

- Extract the public key(.pub) from the public key certificate(.pem).
- The extracted public key is used to decrypt the digest file. The result is then compared against a new unencrypted digest of the temporary file created from the image file with leading 1MB and ending 512 bytes removed.

  This step is achieved through the following openssl command.

  - The general CLI statement appears as follows:

    ```
    openssl dgst -verify <public_key> -keyform <form> <hash_function>
    -signature <digest_file> -binary <temporary_file>
    ```

  - OpenSSL CLI tool gives a "Verified OK" message if both the files match and "Verification Failure" if they do not match.

File signature verification on Linux

You can verify an exported VHD file signature for Linux by following the steps below.

Steps

1. Download the Azure Image Digest file from the NetApp Support Site and extract the digest file(.sig), public key certificate file(.pem) and chain certificate file(.pem).

   Refer to the Download the Azure Image Digest File for more information.

2. Verify the chain of trust.

   ```
   % openssl verify -CAfile Certificate-Chain-9.15.0P1_azure.pem
   Certificate-9.15.0P1_azure.pem
   Certificate-9.15.0P1_azure.pem: OK
   ```

3. Remove the leading 1MB (1048576 Bytes) and ending 512 Bytes of VHD file.

   If 'tail' is used, the option '-c +K' outputs bytes starting with the Kth bytes of the specified file. Hence, 1048577 is passed to 'tail -c'.

   ```
   % tail -c +1048577 ./9150.01000024.05090105.vhd > ./sign.tmp.tail
   % head -c -512 ./sign.tmp.tail > sign.tmp
   % rm ./sign.tmp.tail
   ```

4. Use openssl to extract public key from certificate and verify the striped file(sign.tmp) with the signature file and public key.

   If the input file passes the verification, the command will display "Verification OK". Otherwise, "Verification Failure" will display.

```
% openssl x509 -pubkey -noout -in ./Certificate-9.15.0P1_azure.pem >
./Code-Sign-Cert-Public-key.pub

% openssl dgst -verify Code-Sign-Cert-Public-key.pub -keyform PEM
-sha256 -signature digest.sig -binary ./sign.tmp
Verification OK

% openssl dgst -verify Code-Sign-Cert-Public-key.pub -keyform PEM
-sha256 -signature digest.sig -binary ./another_file_from_nowhere.tmp
Verification Failure
```

5. Clean up the workspace.

```
% rm ./9150.01000024.05090105.vhd ./sign.tmp
% rm *.sig *.pub *.pem
```

File signature verification on Mac OS

You can verify an exported VHD file signature for Mac OS by following the steps below.

Steps

1. Download the Azure Image Digest file from the NetApp Support Site and extract the digest file(.sig), public key certificate file(.pem) and chain certificate file(.pem).

   Refer to the Download the Azure Image Digest File for more information.

2. Verify the chain of trust.

```
% openssl verify -CAfile Certificate-Chain-9.15.0P1_azure.pem
Certificate-9.15.0P1_azure.pem
Certificate-9.15.0P1_azure.pem: OK
```

3. Remove the leading 1MB(1048576 Bytes) and ending 512 Bytes of VHD file.

   If 'tail' is used, the option '-c +K' outputs bytes starting with the Kth bytes of the specified file. Hence, 1048577 is passed to 'tail -c'. It takes around 13m for the tail command to complete on Mac OS.

```
% tail -c +1048577 ./9150.01000024.05090105.vhd > ./sign.tmp.tail
% head -c -512 ./sign.tmp.tail > sign.tmp
% rm ./sign.tmp.tail
```

4. Use openssl to extract public key from certificate and verify the striped file(sign.tmp) with the signature file and public key.

If the input file passes the verification, the command will display "Verification OK". Otherwise, "Verification Failure" will display.

```
% openssl x509 -pubkey -noout -in ./Certificate-9.15.0P1_azure.pem >
./Code-Sign-Cert-Public-key.pub

% openssl dgst -verify Code-Sign-Cert-Public-key.pub -keyform PEM
-sha256 -signature digest.sig -binary ./sign.tmp
Verified OK

% openssl dgst -verify Code-Sign-Cert-Public-key.pub -keyform PEM
-sha256 -signature digest.sig -binary ./another_file_from_nowhere.tmp
Verification Failure
```

5. Clean up the workspace.

```
% rm ./9150.01000024.05090105.vhd ./sign.tmp
% rm *.sig *.pub *.pem
```

Where to find additional information about Azure image verification

Check out the links below for additional information about Azure Image Verification. The links below take you to non-NetApp sites.

References

- Page Fault Blog: How to sign and verify using OpenSSL
- Use Azure Marketplace image to create VM image for your Azure Stack Edge Pro GPU | Microsoft Learn
- Export/Copy a managed disk to a storage account using the Azure CLI | Microsoft Learn
- Azure Cloud Shell Quickstart - Bash | Microsoft Learn
- How to install the Azure CLI | Microsoft Learn
- az storage blob copy | Microsoft Learn
- Sign in with Azure CLI — Login and Authentication | Microsoft Learn

Copyright information

Trademark information