

# Surface Asset Tag Tool

Article • 01/03/2023 • 2 minutes to read • Applies to: Windows 10, Windows 11

Surface Asset Tag is a command line interface (CLI) utility that allows you to view, assign, and modify an assigned asset tag value for Surface devices.

## System requirements

- Surface Pro 3 or later and all newer Surface devices.
- UEFI firmware version 3.9.150.0 or later

## Using Surface Asset Tag

To run Surface Asset Tag:

1. On the Surface device, download **Surface Asset Tag.zip** from the [Microsoft Download Center](#), extract the zip file, and save AssetTag.exe in desired folder (in this example, C:\assets).

### ⓘ Note

For Surface Pro X, use the application named **AssetTag\_x86** in the ZIP file.

2. Open a command console as an Administrator and run AssetTag.exe, entering the full path to the tool.
3. Restart Surface.

### ⓘ Note

After setting the asset tag, a second reboot is required before it appears in WMI.

## Asset Tag tool commands

In the following examples, AssetTag.exe is saved in a directory on a local machine (C:\assets).

To get the proposed asset tag, run **AssetTag -g**:

Console

```
C:\assets\AssetTag.exe -g
```

To clear the proposed asset tag, run **AssetTag -s**:

Console

```
C:\assets\AssetTag.exe -s
```

To set the proposed asset tag, run **AssetTag -s testassettag12**:

```
C:\assets\AssetTag.exe -s testassettag12
```

### Tip

The asset tag value must contain between 1 and 36 characters. Valid characters include A-Z, a-z, 0-9, period (.) and hyphen (-).

## Managing asset tags

You can view the existing asset tag in the UEFI settings under Device Information (**Control Panel > Recovery > Advanced Startup > Restart now.**)

The figure below shows the results of running the Asset Tag Tool on Surface Go.



**Figure 1.** Results of running Surface Asset Tag tool on Surface Go

Alternately, you can use WMI to query the existing asset tag on a device:

```
(Get-WmiObject -query "Select * from Win32_SystemEnclosure")
```

## Example

Console

```
C:\Windows\System32> (Get-WmiObject -query "Select * from  
Win32_SystemEnclosure")
```

## Using PowerShell

You can use the script below as a way of getting the proposed value and interpreting any errors.

PowerShell

```
AssetTag -g \> $asset\_tag 2\> $error\_message  
$asset\_tag\_return\_code = $LASTEXITCODE  
$asset\_tag = $asset\_tag.Trim("`r`n")  
  
if ($asset\_tag\_return\_code -eq 0) {  
Write-Output ("Good Tag = " + $asset\_tag)  
} else {  
Write-Output (  
"Failure: Code = " + $asset\_tag\_return\_code +  
"Tag = " + $asset\_tag +  
"Message = " + $error\_message)
```

}