

Considerations for Surface and Microsoft Endpoint Configuration Manager

Article • 01/03/2023 • 4 minutes to read • Applies to: Windows 10, Windows 11

Fundamentally, management and deployment of Surface devices with Microsoft Endpoint Configuration Manager is the same as the management and deployment of any other PC. Like any other PC, a deployment of Surface devices includes importing drivers, importing a Windows image, preparing a deployment task sequence, and then deploying the task sequence to a collection. After deployment, Surface devices are like any other Windows client; to publish apps, settings, and policies, you use the same process as you would use for any other device.

To learn more, see [Microsoft Endpoint Configuration Manager documentation](#).

Although the deployment and management of Surface devices is fundamentally similar to other PCs, some scenarios may require extra IT tasks, as described below.

Tip

Use the **Current Branch of Microsoft Endpoint Configuration Manager** to manage Surface devices.

Update Surface device drivers and firmware

To deploy updates device drivers and firmware using Configuration Manager or Windows Server Update Services (WSUS), see [Manage Surface driver and firmware updates](#).

Surface Ethernet adapters and Configuration Manager deployment

The default mechanism that Configuration Manager uses to identify devices during deployment is the Media Access Control (MAC) address. Because the MAC address is associated with the Ethernet controller, an Ethernet adapter shared among multiple devices will cause Configuration Manager to identify each of the devices as only a single device, resulting in a failed deployment.

To ensure that Surface devices using the same Ethernet adapter are identified as unique devices during deployment, you can instruct Configuration Manager to identify devices using another method. You can specify that Configuration Manager use other identification methods, as documented in [Manage duplicate hardware identifiers](#):

- Add an exclusion for the MAC addresses of Surface Ethernet adapters, which forces Configuration Manager to overlook the MAC address in preference of the System UUID.
- Use a script to identify a newly deployed Surface device by the MAC address of its wireless adapter.

Surface Ethernet Driver

Since 2016, the driver for the Surface Ethernet adapter has been included by default in Windows and requires no another IT configuration. The driver is no longer available for download from the Microsoft Download Center. If you need to deploy earlier versions of Windows 10 Pro, you can download the latest driver from the [Microsoft Update Catalog](#).

Deploy Surface app with Configuration Manager

With the release of Microsoft Store for Business, Surface app is no longer available as a driver and firmware download. Organizations deploying Surface app to managed Surface devices or during deployment via Configuration Manager, must first acquire Surface app through Microsoft Store for Business. To learn more, see [Deploy Surface app with Microsoft Store for Business](#).

Use prestaged media with Surface clients

If your organization uses prestaged media to load deployment resources onto machines prior to deployment with Configuration Manager, you may need to take extra steps. Specifically, a native UEFI environment requires that you create multiple partitions on the boot disk of the system. If you're following along with the [documentation for prestaged media](#), the instructions provide for only single partition boot disks and therefore will fail when applied to Surface devices.

To learn more, see [How to apply Task Sequence Prestaged Media on multi-partitioned disks for BIOS or UEFI PCs in System](#) blog post.

Licensing conflicts with OEM Activation 3.0

Surface devices come preinstalled with a licensed copy of Windows. The license key for this preinstalled copy of Windows is embedded in the firmware of the device with [OEM Activation 3.0](#) (OA 3.0). When you run Windows installation media on a device with an OA 3.0 key, Windows setup automatically reads the license key and uses it to install and activate Windows. In most situations, users don't have to find or enter a license key.

When you reimage a device by using Windows Enterprise, the embedded license key doesn't cause a conflict. This is because the installation media for Windows Enterprise is configured to install only an Enterprise edition of Windows and is incompatible with the license key embedded in the system firmware. If a product key isn't specified--such as when you intend to activate with Key Management Services [KMS] or Active Directory Based Activation--a Generic Volume License Key (GVLK) is used until Windows is activated by one of those technologies.

However, issues may arise when organizations intend to use versions of Windows that are compatible with the firmware embedded key. For example, an organization that wants to install Windows 10 Pro on a device that originally shipped with Windows 10 Home may encounter difficulty when Windows setup automatically reads the Home edition key during installation and installs as Windows 10 Home instead of Windows 10 Pro. To avoid this conflict, use the Ei.cfg or Pid.txt file to explicitly instruct Windows setup to prompt for a product key, or enter a specific product key in the deployment task sequence. For more information, see [Windows Setup Edition Configuration and Product ID Files](#). If you don't have a specific key, you can use the default product keys for Windows. For more information, see [Deploy Windows 10](#).

Apply an asset tag during deployment

With the [Surface Asset tag tool](#), you can identify devices from UEFI even if the operating system fails. To learn more about managing assets with Configuration Manager, see [Introduction to asset intelligence in Configuration Manager](#).

Configure push-button reset

When you deploy Windows to a Surface device, the push-button reset functionality of Windows is configured by default to revert the system back to a state where the environment isn't yet configured. When the reset function is used, the system discards any

installed applications and settings. Although in some situations it can be beneficial to restore the system to a state without applications and settings, in a professional environment, this effectively renders the system unusable to the end user.

Push-button reset can be configured, however, to restore the system configuration to a state where it's ready for use by the end user. Follow the process outlined in [Deploy push-button reset features](#) to customize the push-button reset experience for your devices.