

Windows 11 Compatible Devices

Description

Microsoft have provided their list of supported Windows 11 requirements:

<https://www.microsoft.com/en-gb/windows/windows-11-specifications>

Including links to subcategories, for example processor compliance:

<https://docs.microsoft.com/en-gb/windows-hardware/design/minimum/windows-processor-requirements>

The variety of machines that could be either compliant or non-compliant is vast. The recipe here allows for a scripted method to confirm the status of compliance and is based upon Microsoft's [Readiness PowerShell script](#), details of which are highlighted in the following documentation:

<https://techcommunity.microsoft.com/t5/microsoft-endpoint-manager-blog/understanding-readiness-for-windows-11-with-microsoft-endpoint/ba-p/2770866>

Two of the methods provided are edited versions of the original supplied Microsoft script. One is a straight forward Custom Field, whilst the other uses a more advanced method to achieve the same result. The script for both methods will provide an output of Pass or Fail in the Custom Field value. Please choose as desired.

Custom Field values may be added to the Client View:

Search:		Everything	Clients	Mobile	Groups	Clear all filters
Name		Windows 11 Compatible				^
	DESKTOP-JNETSS4	Fail				
	mini	NA				

Unaltered Version

Unaltered version of the Microsoft supplied readiness script. Output will include all text as dictated by Microsoft. As a Custom Field, this information can be lengthy, but inventory Queries may be configured to identify the word 'Fail'.

Ingredients

- Following Custom Field

↓ Windows
Windows 11 Readiness Unaltered.customfields

Directions

1. Download the provide Custom Field: 'Windows 11 Readiness Unaltered'
2. Open the Custom Field Editor: FileWave Admin > Assistants > Custom Fields > Edit Custom Fields
3. Select Import and choose the downloaded Custom Field from step 1
4. Change Name if desired
5. Save

Example failed value:

```
{"returnCode":1,"returnReason":"TPM, Processor, ","logging":"Storage: OSDiskSize=98GB. PASS; Memory: System_Memory=4GB. PASS; TPM: TPMVersion=False. FAIL; Processor: {AddressWidth=64; MaxClockSpeed=2494; NumberOfLogicalCores=4; Manufacturer=GenuineIntel; Caption=Intel64 Family 6 Model 70 Stepping 1; }. FAIL; SecureBoot: Capable. PASS; ","returnResult":"NOT CAPABLE"}
```

Simplified Method

The information output by the default script is lengthy and can be considered as inappropriate as a single Custom Field value. This method alters the script, which when used as a Custom Field will return either Pass or Fail. However the details of why it failed will not be provided.

Ingredients

- Following Custom Field

↓ Windows
Windows 11 Readiness.customfields

Directions

1. Download the provide Custom Field: 'Windows 11 Readiness'
2. Open the Custom Field Editor: FileWave Admin > Assistants > Custom Fields > Edit Custom Fields
3. Select Import and choose the downloaded Custom Field from step 1
4. Change Name if desired
5. Save

Advanced Method

Since the hardware of the device will rarely change, it is unnecessary to have the Custom Field script run on every inventory. Additionally, the information output by the default script is lengthy and can be considered as inappropriate as a single Custom Field value. The following method involves building an Administrator Custom Field and the script will be added as a Fileset instead. This Fileset will update

the Custom Field value when ran, the details will be stored in a local log file on the device, yet the Custom Field will merely show Pass or Fail once the script has ran on a Windows device.

As a Fileset, the script will run only once without intervention, preventing the script from unnecessarily running over and over again.

Ingredients

- Administrator Custom Field with an internal name of 'windows_11_compatible'

↓ Windows
Windows 11 Compatible.customfields

- Following Fileset:

↓ Windows
Windows 11Compliance.fileset.zip

Directions

Custom Field

1. Download the provided Custom Field: 'Windows 11 Compliance'
2. Open the Custom Field Editor: FileWave Admin > Assistants > Custom Fields > Edit Custom Fields
3. Select Import and choose the downloaded Custom Field from step 1
4. Change Name as desired, but ensure the Internal Name is not altered and association is to all devices
5. Save
6. Once configured, the Fileset may then be associated and pushed to devices

Fileset

1. Download the provided Fileset
2. Edit the Fileset's script Environment Variables (details below)
3. Associate to devices for testing and then once satisfied push to all devices

Fileset Editing

- Open the Fileset, select the script and choose Get Info
- Select the Executable tab and then Environment Variables
- Replace the Values as appropriate

Info - Windows 11 Compliance : HardwareReadiness.ps1

 HardwareReadiness.ps1

Kind: File

Created: Mon Nov 22 2021 12:47 pm

Modified: Mon Nov 22 2021 12:47 pm

Permissions

ACLs

Verification

Executable

Flags

Execution Control

☒ Execute once when activated

☐ Interactive (ignored in non Windows™ clients)
 ☒ Non-interactive (background)

☒ Wait for executable to finish

Wait for: Infinite

Launch Arguments

Environment Variables

Variable	Value
server	REPLACE WITH SERVER FQDN
token	REPLACE WITH FILEWAVE ADMIN TOKEN

+

-

Reset

Reset All

The values of the environment variables are set just before the script execution.
 To use an inventory field value, use the syntax %FIELD_NAME%.
 For instance: MY_VAR: foo-%asset_tag%
 Note: environment variable names are case insensitive in Windows

Note: Log files will be collected for synchronous non-interactive scripts only

Apply

 Click the lock to take control of this Fileset

- The 'value' for the 'server' variable should be replaced with the name of the server as seen in Preferences > Mobile of the Admin console
- The 'value' for the 'token' should be replaced with a chosen Admin token from: Assistants > Manage Administrators > (Chosen Account Name) > Application Tokens. Copy the 'Token (base64)'

Additional Information

The Fileset will use the FileWave API to report back the current status of the device's compatibility during Fileset activation. If devices are addressed to change their compatibility status, it is possible to run a 'Reinstall Fileset' which will cause the API to update the current information, refreshing the Custom Field.

The full output of the script will be available in the script log, accessible from the right click menu item of a Fileset's script status view from Client Info (local network between Admin device and selected machine is required). A failure example:

```
{"returnCode":1,"returnReason":"TPM, Processor, ","logging":"Storage: OSDiskSize=98GB. F
```

Self-Signed Certs

The Fileset Activation Script 'HardwareReadiness.ps1' must be edited to allow for Self-Signed Certificates. The following section should have the mentioned lines updated to remove the leading hashes. After removal it should look like the following:

```
#####  
# Beginning of ammendment for FileWave Custom Field report  
  
# REMOVE HASHES FROM FOLLOWING 12 LINES IF USING A SELF-SIGNED CERTIFICATE  
add-type @"  
using System.Net;  
using System.Security.Cryptography.X509Certificates;  
public class TrustAllCertsPolicy : ICertificatePolicy {  
    public bool CheckValidationResult(  
        ServicePoint srvPoint, X509Certificate certificate,  
        WebRequest request, int certificateProblem) {  
        return true;  
    }  
}  
"@  
[System.Net.ServicePointManager]::CertificatePolicy = New-Object TrustAllCertsPolicy
```

 The client must be able to reach the server on port 443 to be able to post the API update back to the server.

Result

The Custom Field for the Simplified and Advanced methods actually provides 3 possible values:

- NA – Default value
- Fail – One or more items failed the check
- Pass – All items passed the check and the device is ready for Windows 11

Notes

These options are by no means the only options available. The script could be used within an Upgrade Fileset for Windows 11, for example, and the script may run prior to confirm if the device satisfies the requirements. However, requirement scripts should only be used where they will eventually become true, to prevent them from running forever and being a constant draw on the server.

🔄Revision #7

★Created 2023-07-13 08:28:41 UTC by Rommel Navarro

✎Updated 2025-11-03 16:54:32 UTC by Rommel Navarro