

How to Deploy Wave Desktop Clients in Batches - User Guide

Overview

This document introduces how to deploy Wave Desktop clients in batches for enterprise employees' computers and configure parameters to help employees to quickly use the Wave Desktop clients throughout the enterprise.

This document uses the Microsoft Intune tool as an example to describe how to deploy Wave Desktop clients in batches on Windows and MAC OS.

Prerequisites

The administrator needs to register a Microsoft Intune account, and the computers under the enterprise have registered on the Microsoft Intune platform.

Microsoft Intune is a Microsoft tool for managing devices and deploying Windows/MAC OS applications in batches. For more details, please refer to [Microsoft official website](#).

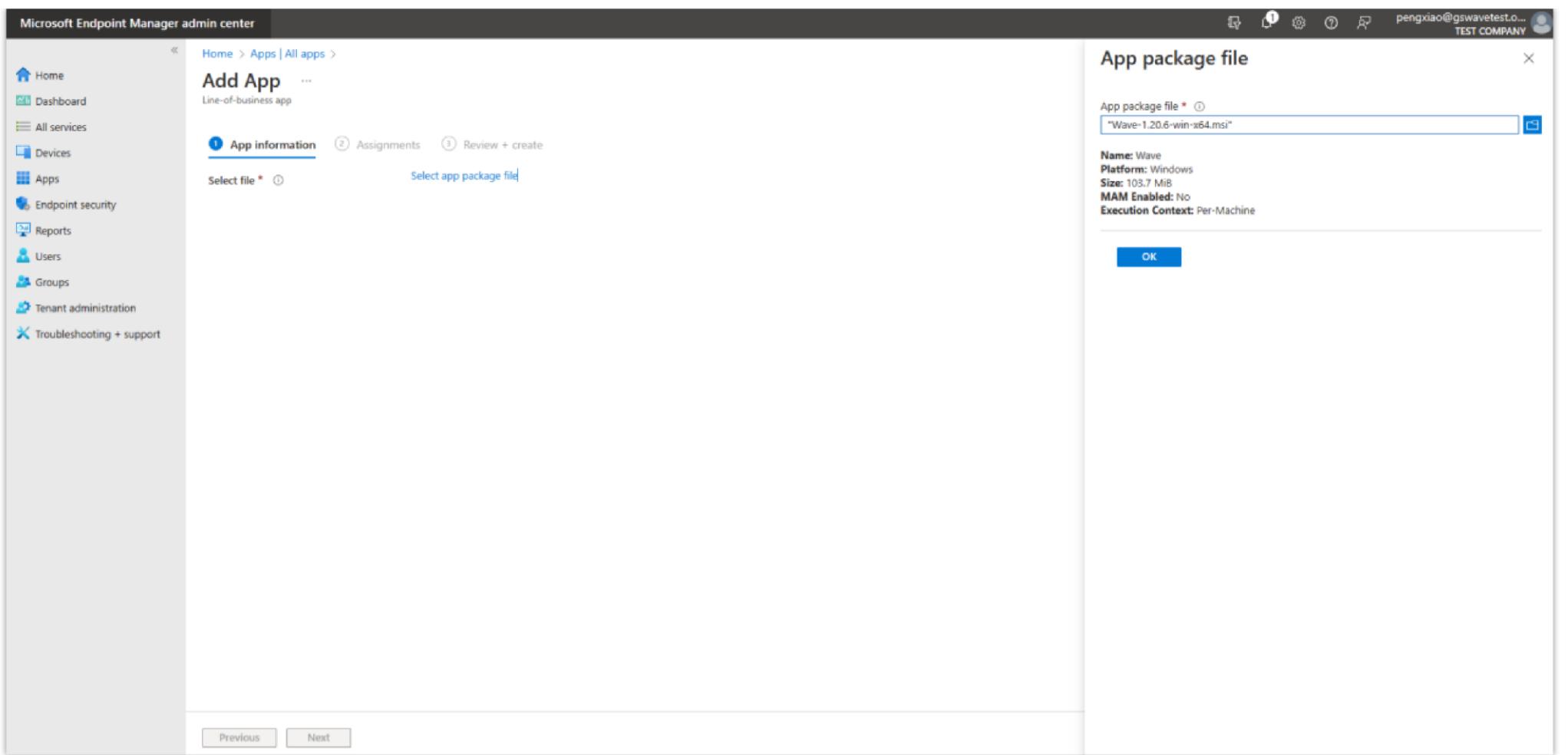
Batch Deploy Wave on Windows

Step 1: Deploy Wave Desktop Clients in Batches

1. Log in to the Endpoint Manager Admin Center of Intune.
2. Click the “Add” option on the “Apps” -> “All apps” page to add the application, select the option “Line-of-business app” for the “App type”, and click the “Select” option as the screenshot shows below:

The screenshot shows the Microsoft Endpoint Manager admin center interface. On the left, there's a navigation sidebar with various options like Home, Dashboard, All services, Devices, Apps, Endpoint security, Reports, Users, Groups, Tenant administration, and Troubleshooting + support. The 'Apps' section is currently selected. In the main content area, the 'All apps' tab is active, showing a table with columns: Name, Type, Status, and Version. There are two entries: 'install init config file' (Windows app (Win32)) and 'Wave' (Windows MSI line-of-business app, version 1.20.6.0). To the right of the table, a modal window titled 'Select app type' is open. It has a dropdown menu set to 'Line-of-business app'. Below the dropdown, there's a section titled 'Line-of-business app' with instructions on how to upload a custom or in-house app. At the bottom of the modal are 'Select' and 'Cancel' buttons.

3. Upload the Wave Desktop MSI installation package and click the “OK” option:



Upload Wave Desktop .msi package

4. Configure the relevant parameters of the App according to your requirements, and click the “Next” option:

Home > Apps | Windows > Windows | Windows apps > Wave | Properties >

Edit application

Windows MSI line-of-business app

App information Review + save

Select file to update *

Name *

Description *

[Edit Description](#)

Publisher *

App install context

Ignore app version

Command-line arguments

Category [▼](#)

Show this as a featured app in the Company Portal

[Review + save](#) [Cancel](#)

Edit Application Parameters

5. Select users, groups, or devices that you want to install the Wave Desktop client and click the “Next” option:

The screenshot shows the 'Add App' page in the Microsoft Endpoint Manager admin center. The 'Assignments' tab is active. The interface is divided into three main sections: 'Required', 'Available for enrolled devices', and 'Uninstall'. Each section has columns for Group mode, Group, Filter mode, Filter, and Install Context. Under 'Required', there is a note 'No assignments' and a link '+ Add group'. Under 'Available for enrolled devices', there is also a note 'No assignments' and a link '+ Add group'. Under 'Uninstall', there is a note 'No assignments' and a link '+ Add group'. At the bottom, there are 'Previous' and 'Next' buttons.

App Assignments

6. After the above operations are complete, the Wave client will be automatically installed on the user's computer at the next check-in.

Step 2: (Optional) Batch Deploy Wave Initial Configuration Files

When Wave Desktop is initialized, it will read the configuration file in the specific directory (%appdata%/Wave/). You only need to edit the configuration file and copy it to the corresponding user directory before initialization.

1. Create the Wave initialization configuration file "init.cfg" and edit it in the following format (you only need to copy the parameters that you need to configure):

init.cfg:

```

#Server address
ServerAddress=https://xxxxx.a.gdms.cloud

# The default language setting is English. The following languages are supported.
# Prompt: The explanation of the language is in the brackets.
# en-US(English), zh-CN(简体中文), el-GR(Ελληνικά), it-IL(Italiano),
# es-LA(Español (América Latina)), es-ES(Español (Europe)), ar-EG(العربية), fr-FR(Français),
# ru-RU(Русский), de-DE(Deutsch), pl-PL(Polskie), pt-PT(Português),
# vi-VN(Tiếng Việt), th-TH(ภาษาไทย), cs-CZ(Čeština), tr-TR(Türkçe),
# he-IL(עברית)
language=en-US

# 0: No 1: Yes
isLaunchOnSystemStartup=1

# 0: No 1: Yes
isAutoAnswer=0

# 0: disable 1: enable
enableMessageNotificationSound=1

# 0: disable 1: enable
enableIncomingCallAndMeetingReminderSound=1

# 12: 12 Hour Clock      24: 24 Hour Clock
timeFormat=24

# YYYY/MM/DD, DD/MM/YYYY or MM/DD/YYYY
dateFormat=MM/DD/YYYY

# Pre-install Add-in and Related Parameters
#preInstallPlugins=[{"Google Drive": {}}, {"IPVideotalk":{"account":"xxxx", "password":"xxx"} }]

# Call Ringtone , such as "C:\Users\AppData\Wave\CustomRinging\helloWorld.wav"
callRingtonePath=C:\Users\AppData\Wave\CustomRinging\helloWorld.wav

#0: disable 1: enable
enableUpgrade=1

```

Note

For details about add-in configuration parameters, please refer to the document “How to Pre-install Wave Add-ins on UCM63xx”:
<https://documentation.grandstream.com/knowledge-base/pre-install-wave-add-ins-on-the-ucm63xx/>

2. Create PowerShell scripts “install.ps1” and ”uninstall.ps1” by referring to the following contents:

install.ps1 :

```

$TargetFolder = "$env:APPDATA\Wave"
$SourceFolder = $PSScriptRoot
$ProgramDataFolder = "$env:ProgramData\Wave"

If (!(Test-Path $TargetFolder)){
    New-Item -Path $TargetFolder -ItemType Directory -Force
}
Copy-Item -Path "$SourceFolder\init.cfg" -Destination $TargetFolder -Recurse -Force -ErrorAction Stop

If (!(Test-Path $ProgramDataFolder)){
    New-Item -Path $ProgramDataFolder -ItemType Directory -Force
}
Copy-Item -Path "$SourceFolder\init.cfg" -Destination $ProgramDataFolder -Recurse -Force -ErrorAction Stop

```

uninstall.ps1:

```
$TargetFolder = "$env:APPDATA\Wave"
$ProgramDataFolder = "$env:ProgramData\Wave"

If (Test-Path $TargetFolder){
    Remove-Item -Path "$TargetFolder\init.cfg" -Recurse -Force -ErrorAction Stop
}

If (Test-Path $ProgramDataFolder ){
    Remove-Item -Path "$ProgramDataFolder" -Recurse -Force -ErrorAction Stop
}
```

3. Store the three files above ("init.cfg", "install.ps1", "uninstall.ps1") on your computer and use the Microsoft-Win32-Content-Prep-Tool to pack the files to the installation package in .intunewin format.

Official instructions for packaging to .intunewin format

Packing command:

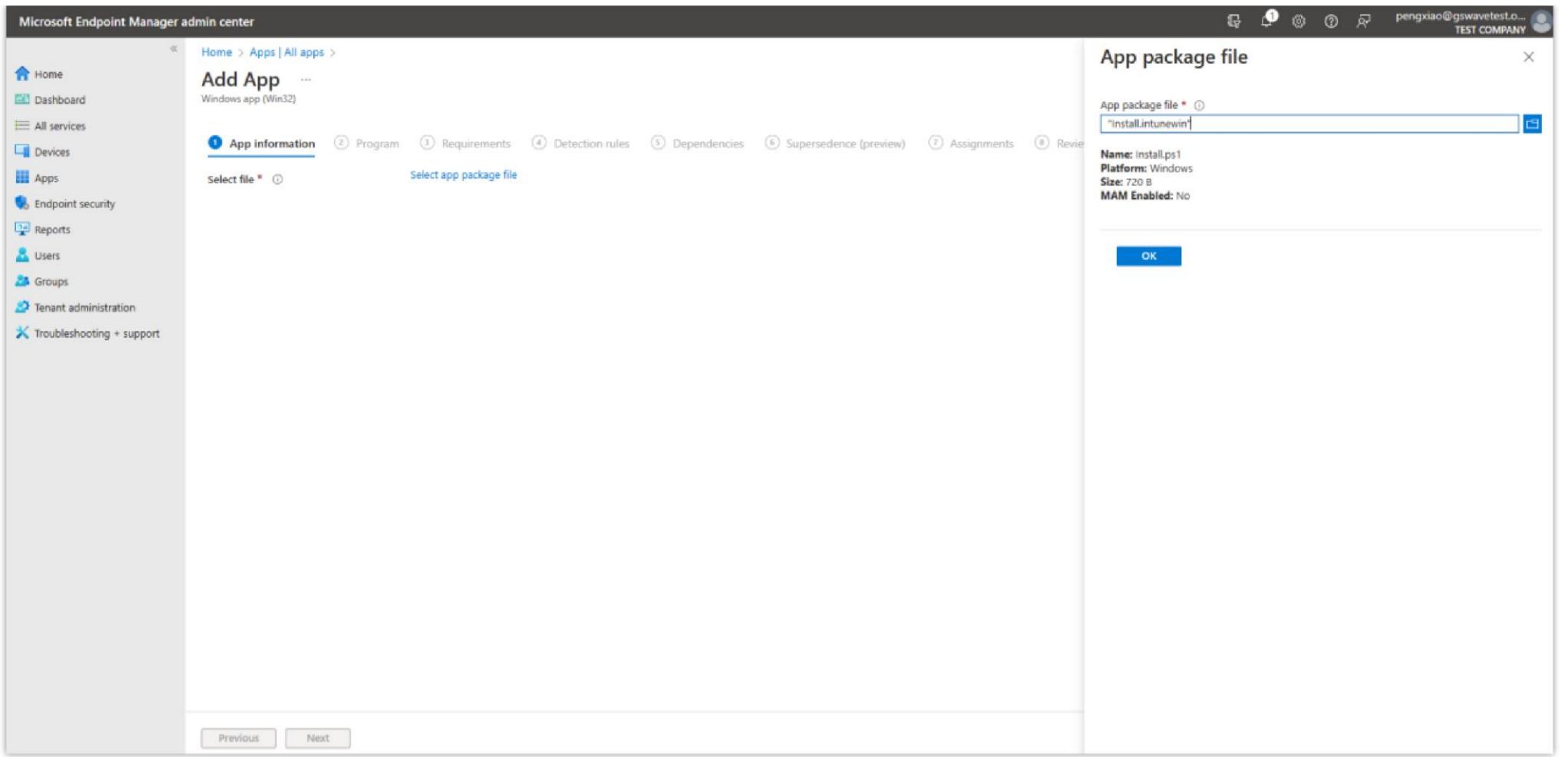
```
IntuneWinAppUtil.exe -c "C:\Users\xxxxxxx\Desktop\WaveScript" -s install.ps1 -o "C:\Users\xxxxxxx\Desktop" -q
```

4. Log in to the Endpoint Manager Admin Center of Intune and deploy the Wave initial configuration file as a Win32 APP.

5. Click the “Add” option on the “Apps” -> “All apps” page to add the application, select the option “Windows app(Win32)” for the “App type”, and click the “Select” option as the screenshot shows below:

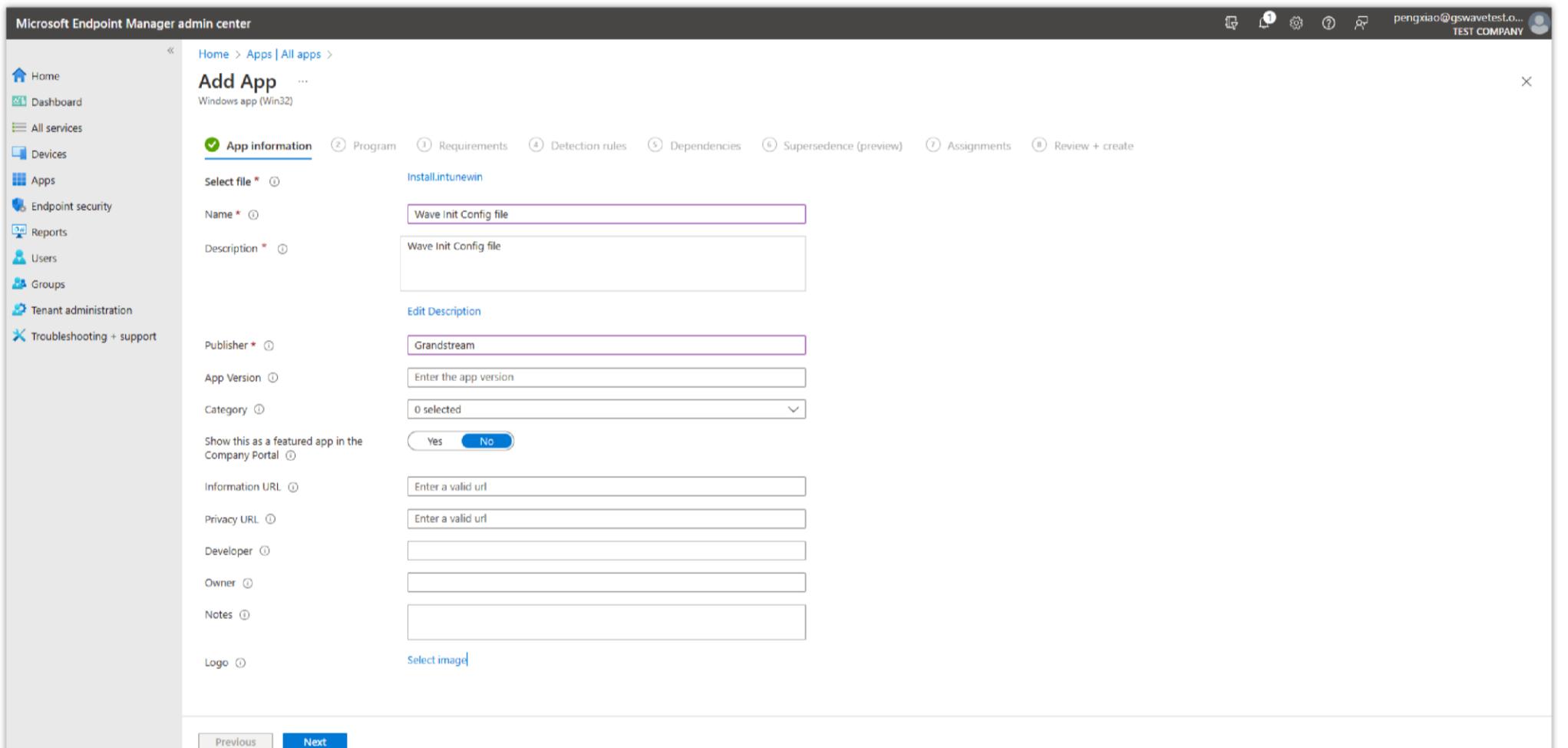
The screenshot shows the Microsoft Endpoint Manager Admin Center interface. On the left, the navigation menu includes Home, Dashboard, All services, Devices, Apps, Endpoint security, Reports, Users, Groups, Tenant administration, and Troubleshooting + support. The main area displays the 'All apps' list under the 'Apps' category. The list shows two items: 'install init config file' (Windows app (Win32)) and 'Wave' (Windows MSI line-of-business app, version 1.20.6.0). To the right of the list, a modal dialog box titled 'Select app type' is open. It has a dropdown menu set to 'Windows app (Win32)'. Below the dropdown, there's a note about Test Base for Microsoft 365, which is a cloud validation service. At the bottom of the dialog are 'Select' and 'Cancel' buttons.

6. Upload the packaged Wave configuration and installation package (in .intunewin format) and click the “OK” option:



Add Configuration File

7. Configure the relevant parameters of the App according to your requirements, and click the “Next” option:



Add Configuration File Parameters

8. Configure the install and uninstall commands. Please refer to the commands below. Then, you can click the “Next” button:

The screenshot shows the 'Add App' wizard in the Microsoft Endpoint Manager admin center. The 'Program' tab is selected. Under 'Install command', the value is 'powershell.exe -ExecutionPolicy Bypass -File install.ps1'. Under 'Uninstall command', the value is 'powershell.exe -ExecutionPolicy Bypass -File uninstall.ps1'. The 'Install behavior' is set to 'System'. The 'Device restart behavior' is set to 'No specific action'. Below these, a table lists return codes with their corresponding code types: 0 (Success), 1707 (Success), 3010 (Soft reboot), 1641 (Hard reboot), and 1618 (Retry). At the bottom, there are 'Previous' and 'Next' buttons.

Program Configuration File

Install command:

```
powershell.exe -ExecutionPolicy Bypass -File install.ps1
```

Uninstall command:

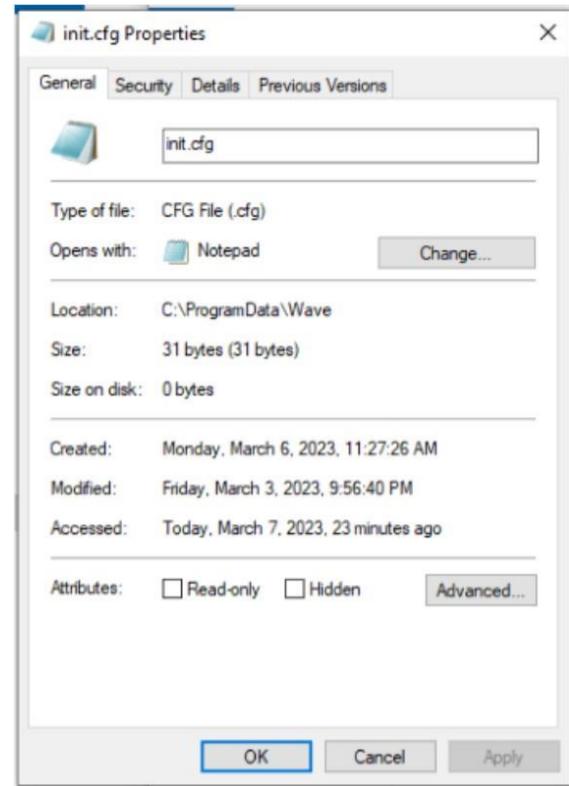
```
powershell.exe -ExecutionPolicy Bypass -File uninstall.ps1
```

9. Configure the configuration required for APP installation and click the “Next” option as the screenshot shows below:

The screenshot shows the 'Add App' wizard in the Microsoft Endpoint Manager admin center. The 'Requirements' tab is selected. Under 'Specify the requirements that devices must meet before the app is installed:', the 'Operating system architecture' is set to '64-bit' and the 'Minimum operating system' is set to 'Windows 10 1607'. There are fields for 'Disk space required (MB)', 'Physical memory required (MB)', 'Minimum number of logical processors required', and 'Minimum CPU speed required (MHz)'. Below these, a table for 'Configure additional requirement rules' shows one row: 'Type' (empty) and 'Path/Script' (empty). At the bottom, there are 'Previous' and 'Next' buttons.

Add Configuration File Requirements

10. Set “Detection rules”, select the option ”Manually configure detection rules“, and click the button “Review+save” as the screenshots show below:



Detection rule

Create a rule that indicates the presence of the app.

Rule type: File

Path *: %programdata%\Wave\init.cfg

Detection method *: Date modified

Operator *: Equals

Value *: 2023/03/03 9:56:40 PM

Associated with a 32-bit app on 64-bit clients: No

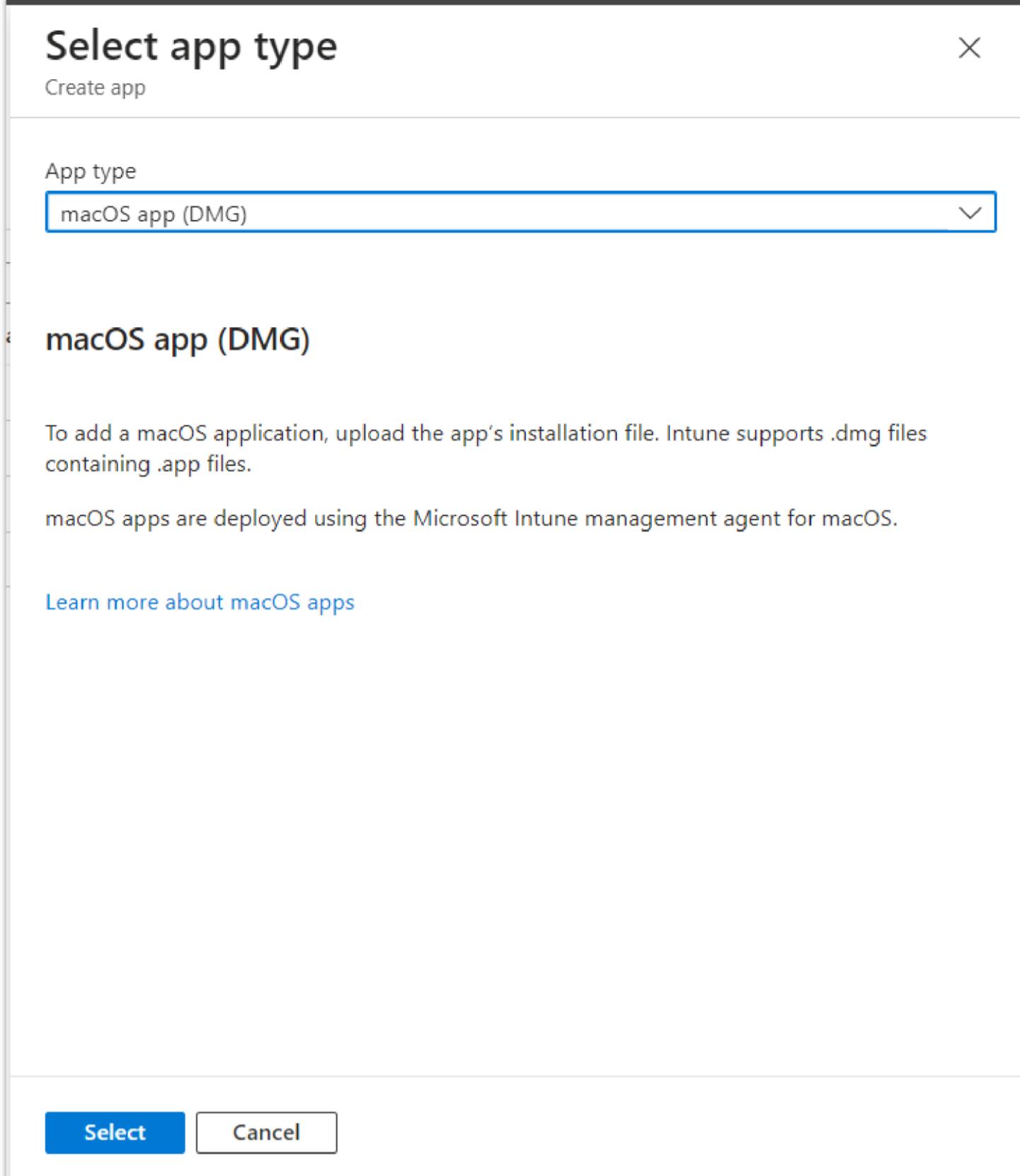
Configuration Files Detection Rules

11. After the operations above are complete, the configuration file will be written to the corresponding Wave directory after the employee's computer check-in. The configuration parameters will take effect after the Wave client starts.

Batch Deploy Wave on MAC

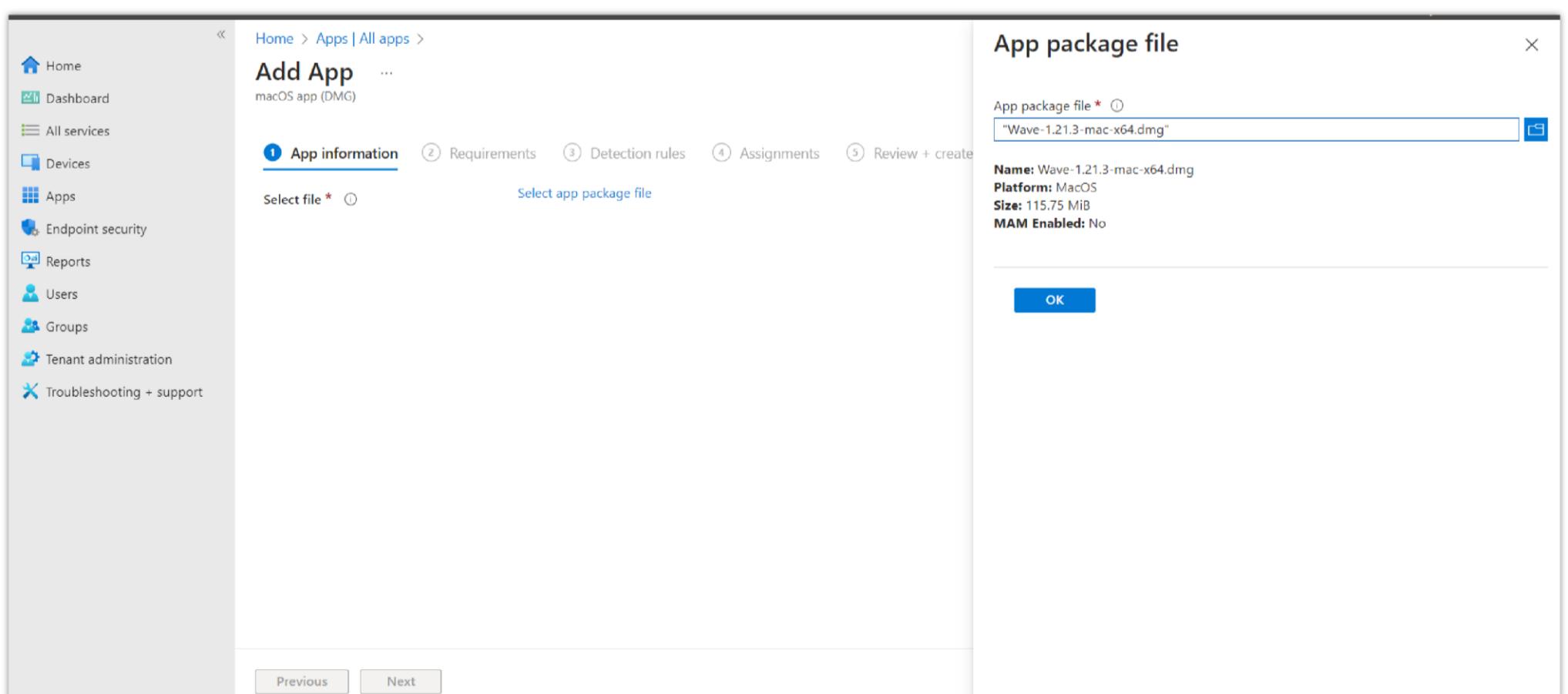
Step 1: Deploy Wave Desktop Client in Batches

1. Log in to the Endpoint Manager Admin Center of Intune.
2. Click the “Add” option on the “Apps” -> “All apps” page to add the application, select the option “macOS app(DMG)” for the “App type”, and click the “Select” option as the screenshot shows below:



Select App Type

3. Upload the dmg installation package for Wave Desktop client and click the “OK” option:



Add .dmg Package File

4. Configure the relevant parameters of the App according to your requirements, and click the “Next” option:

Home > Apps | All apps >

Add App

macOS app (DMG)

Select file *	Wave-1.21.3-mac-x64.dmg
Name *	Wave-1.21.3-mac-x64.dmg
Description *	Wave-1.21.3-mac-x64.dmg
Publisher *	Grandstream
Category	0 selected
Information URL	Enter a valid url
Privacy URL	Enter a valid url
Developer	
Owner	
Notes	
Logo	Select image

[Previous](#) [Next](#)

Add Application Parameters

Home > Apps | All apps >

Add App

macOS app (DMG)

App information Requirements Detection rules Assignments Review + create

Minimum operating system * macOS High Sierra 10.13

[Previous](#) [Next](#)

Add App Minimum Requirements

5. Fill in "Grandstream/DesktopApp.GswaveSoftphone" to "App bundle ID (CFBundleIdentifier)" field under the tab "Detection rules", and fill in the Wave client actual version number (e.g. 1.21.3) to "App Version (CFBundleShortVersionString)" field and click the "Next" option:

The screenshot shows the 'Add App' process in the Intune Admin Center. The current step is 'Detection rules'. It includes fields for 'App bundle ID (CFBundleIdentifier)' (Grandstream.DesktopApp.GwaveSoftphone) and 'App version (CFBundleShortVersionString)' (1.21.3). There are also input fields for 'Enter bundle ID' and 'Enter app version'. A note says: 'Provide the list of apps included in the uploaded file. The app list is case-sensitive. The app listed first is used as the primary app in app reporting. [Learn more about included apps](#)'.

Add Detection Rules

6. Select users, groups, or devices that you want to install the Wave Desktop client and click the “Next” option:

The screenshot shows the 'Add App' process in the Intune Admin Center. The current step is 'Assignments'. It includes sections for 'Required' (Group mode: No assignments) and 'Uninstall' (Group mode: No assignments). Below these are buttons for '+ Add group', '+ Add all users', and '+ Add all devices'. To the right, a modal titled 'Select groups' lists 'Azure AD groups' with 'Wavetest' selected. A 'Selected items' section shows 'Wavetest' with a 'Remove' button. At the bottom are 'Previous' and 'Next' buttons, and a large blue 'Select' button.

Add Assignments

7. After the above operations are complete, the Wave client will be automatically installed on the user's computer at the next check-in.

Step 2: (Optional) Batch Deploy Wave Initial Configuration Files

When Wave Desktop is initialized, it will read the configuration file in the specific directory (~/Library/Application Support/Wave/). You only need to edit the configuration file and copy it to the corresponding user directory before initialization.

1. Log in to the Endpoint Manager Admin Center of Intune.

2. Go to “Devices” -> “Scripts” page, click the “Add” option, and select the “macOS” option:

The screenshot shows the Microsoft Intune Device Scripts page. On the left, there's a navigation sidebar with various service icons. The main area has a search bar at the top, followed by a table listing scripts. One script, "macOS", is selected and highlighted. The table columns are Platform, Script type, Assigned, and Last modified. A "Add" button is located above the table.

	Platform	Script type	Assigned	Last modified
macOS	macOS	Shell script	Yes	3/09/23, 9:46 AM

Add Installation Script

The screenshot shows the "Add script" wizard in the "Basics" step. The left sidebar is identical to the previous screen. The main area has tabs for Basics, Script settings, Assignments, and Review + add. The Basics tab is selected. It has fields for Name (containing "Wave init config for Mac") and Description (containing "Wave init config for Mac"). Below the fields are "Previous" and "Next" buttons.

Write the Script

1. Set parameters in the script and upload the script:

The screenshot shows the "Add script" wizard in the "Script settings" step. The left sidebar is identical. The main area has tabs for Basics, Script settings, Assignments, and Review + add. The Script settings tab is selected. It has a "Upload script" input field containing "InstallWaveCfg.sh" and a code editor window displaying a shell script. Below the code editor are "Previous" and "Next" buttons.

```

2
3
4 function setConfig()
5 {
6     local configKey=$1
7     local configValue=$2
8     local configPath=$3
9     echo $configValue
10    if ! grep '^[:space:]*$configKey'[:space:]*.* "$configPath"
11    then
12        test -s "$configPath" && sed -i '' '$a\\
13 '$configKey'="$configValue"
14 '$configPath' || echo "$configKey=$configValue" >> "$configPath"
15    else
16        sed -i '' 's@^[:space:]*$configKey'[:space:]*.*@$configKey'="$configValue"@
17    fi
18 }
19
20
21 w -h |LC_ALL=C sort -u -t' '-k1,1 | while read user etc
22

```

Script Settings

Please refer to the script contents below (you only need to copy the parameters you need to configure):

```

#!/bin/bash
function setConfig()
{
    local configKey=$1
    local configValue=$2
    local configPath=$3
    echo $configValue
    if ! grep '^[:space:]*$configKey[:space:]*=.*' "$configPath"
        then
        test -s "$configPath" && sed -i '' '$a\' \
'$configKey'="$configValue"'
        '$configPath' || echo "$configKey=$configValue" >> "$configPath"
    else
        sed -i '' 's@^[:space:]*$configKey[:space:]*=.*@'$configKey'="$configValue"@g' "$configPath"
    fi
}

w -h |LC_ALL=C sort -u -t' ' -k1,1 | while read user etc

do
homedir=$(dscl . -read /Users/$user NFSHomeDirectory | cut -d' ' -f2)
WaveInitCfgFile="$homedir/Library/Application Support/Wave/init.cfg"

if [ ! -f "$WaveInitCfgFile" ]
then
    touch "$WaveInitCfgFile"
fi

# add config here

# Server address
setConfig "ServerAddress" "https://xxxxx.a.gdms.cloud" "$WaveInitCfgFile"

# The default language setting is English. The following languages are supported
# Prompt: The explanation of the language is in the brackets
# en-US(English), zh-CN(简体中文), el-GR(Ελληνικά), it-IL(Italiano),
# es-LA(Español (América Latina)), es-ES(Español (Europe)), ar-EG(العربية), fr-FR(Français),
# ru-RU(Русский), de-DE(Deutsch), pl-PL(Polskie), pt-PT(Português),
# vi-VN(Tiếng Việt), th-TH(ภาษาไทย), cs-CZ(Čeština), tr-TR(Türkçe),
# he-IL(עברית)
setConfig "language" "en-US" "$WaveInitCfgFile"

# 0: No 1: Yes
setConfig "isLaunchOnSystemStartup" "1" "$WaveInitCfgFile"

# 0: No 1: Yes
setConfig "isAutoAnswer" "0" "$WaveInitCfgFile"

# 0: disable 1: enable
setConfig "enableIncomingCallAndMeetingReminderSound" "1" "$WaveInitCfgFile"

# 0: disable 1: enable
setConfig "enableMessageNotificationSound" "1" "$WaveInitCfgFile"

# 12: 12 Hour Clock      24: 24 Hour Clock
setConfig "timeFormat" "24" "$WaveInitCfgFile"

# YYYY/MM/DD, DD/MM/YYYY or MM/DD/YYYY
setConfig "dateFormat" "MM/DD/YYYY" "$WaveInitCfgFile"

# Pre-install Add-in and Related Parameters
#setConfig "preInstallPlugins" "[{\\"Google Drive\\":{}}]" "$WaveInitCfgFile"

# Call Ringtone
setConfig "callRingtonePath" "~/Library/Application Support/Wave/Ring.wav" "$WaveInitCfgFile"

#0: disable 1: enable
setConfig "enableUpgrade" "1" "$WaveInitCfgFile"

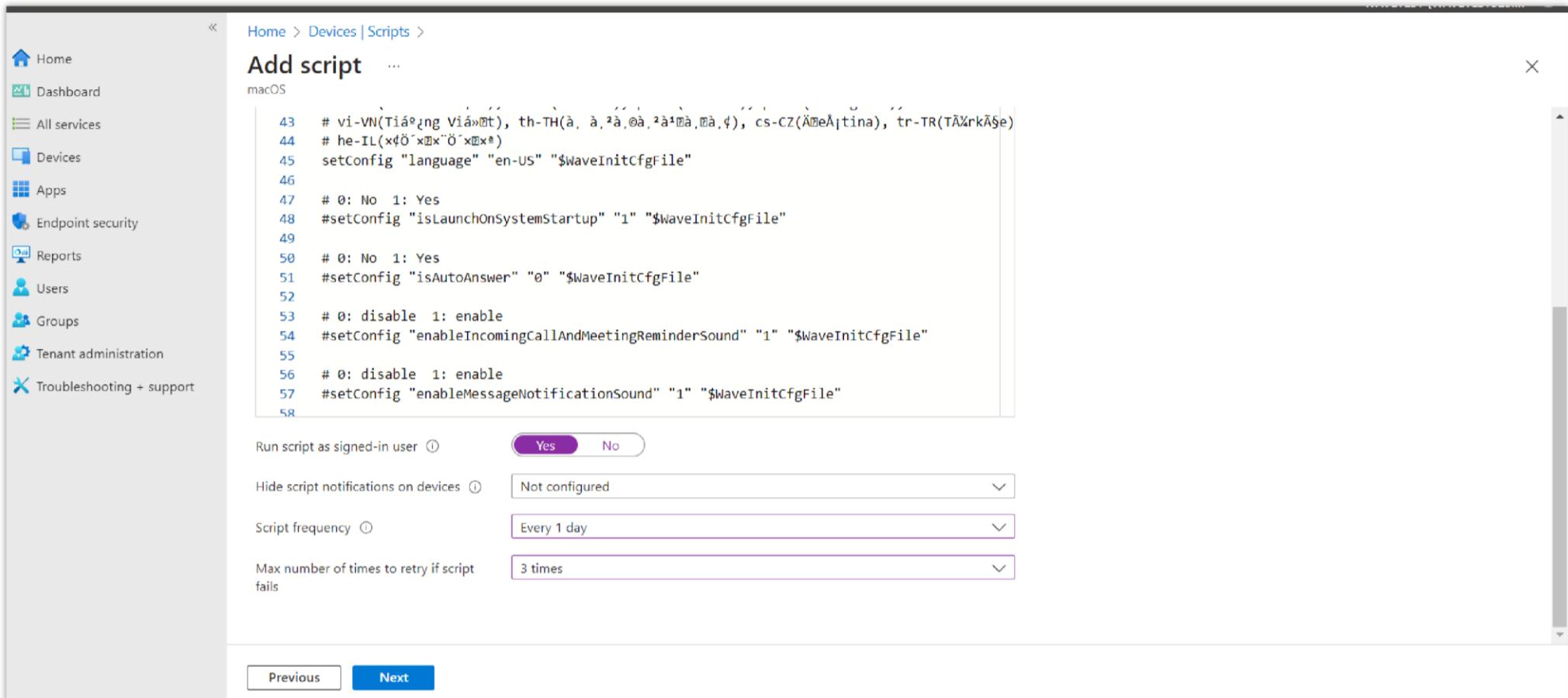
```

done

Note

For details about add-in configuration parameters, please refer to the document “How to Pre-install Wave Add-ins on UCM63xx”:
<https://documentation.grandstream.com/knowledge-base/pre-install-wave-add-ins-on-the-ucm63xx/>

4. Adjust other parameters in the script as required, and click the “Next” option as the screenshot shows below:



The screenshot shows the 'Add script' configuration page for macOS. The left sidebar includes Home, Dashboard, All services, Devices, Apps, Endpoint security, Reports, Users, Groups, Tenant administration, and Troubleshooting + support. The main area shows the following script code:

```
# vi-VN(Tiếng Việt), th-TH(ไทย), cs-CZ(Cesky), tr-TR(Türkçe)  
# he-IL(עברית)  
setConfig "language" "en-us" "$WaveInitCfgFile"  
# 0: No 1: Yes  
#setConfig "isLaunchOnSystemStartup" "1" "$WaveInitCfgFile"  
# 0: No 1: Yes  
#setConfig "isAutoAnswer" "0" "$WaveInitCfgFile"  
# 0: disable 1: enable  
#setConfig "enableIncomingCallAndMeetingReminderSound" "1" "$WaveInitCfgFile"  
# 0: disable 1: enable  
#setConfig "enableMessageNotificationSound" "1" "$WaveInitCfgFile"
```

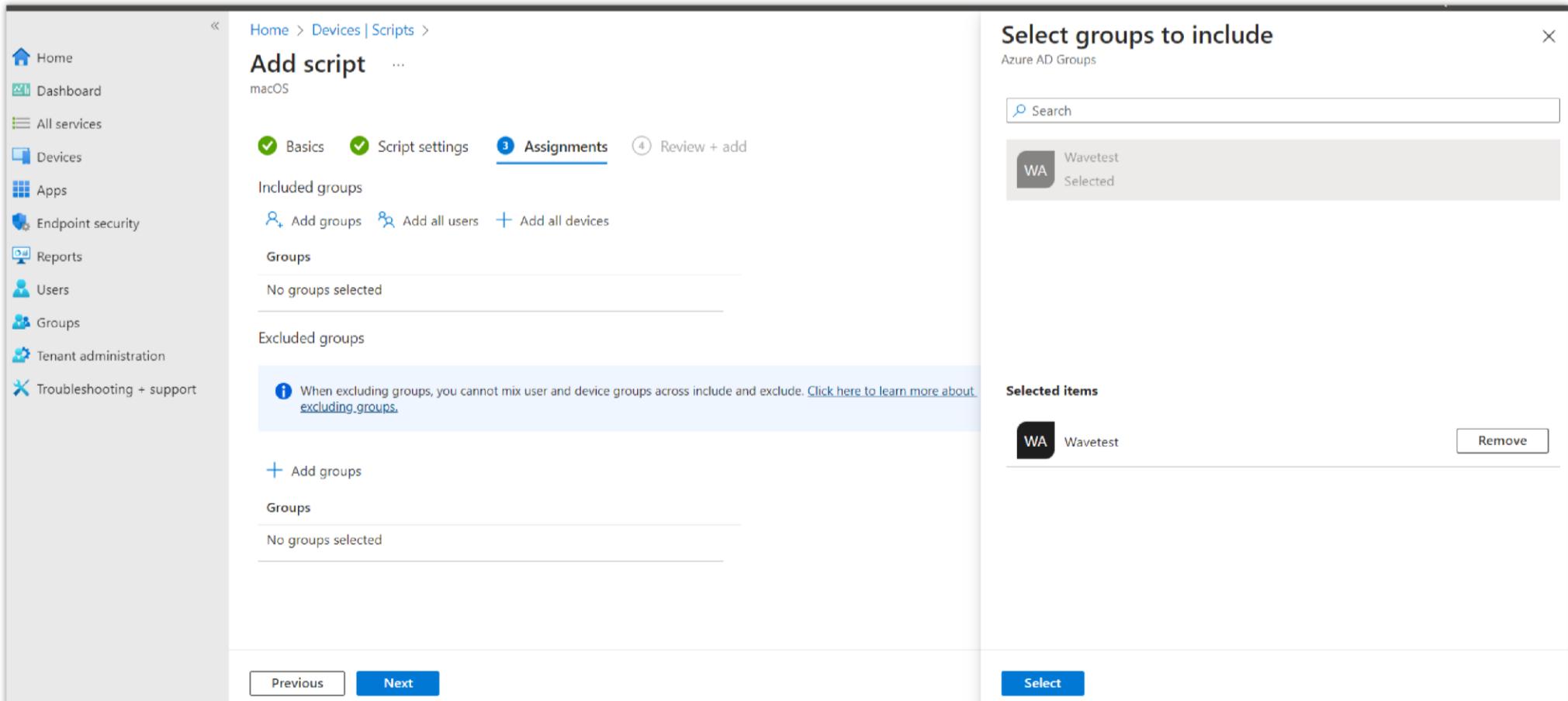
Below the code are several configuration options:

- Run script as signed-in user: Yes (selected)
- Hide script notifications on devices: Not configured
- Script frequency: Every 1 day
- Max number of times to retry if script fails: 3 times

At the bottom are 'Previous' and 'Next' buttons.

Script Settings

5. Select users, groups, or devices that you want to install the Wave Desktop client and click the “Next” option:



The screenshot shows the 'Select groups to include' dialog and the 'Script Assignments' page. The left sidebar is identical to the previous screenshot. The main area shows the 'Assignments' step of the script creation process.

Select groups to include

Azure AD Groups

Search:

WA Wavetest Selected

Script Assignments

Basics Script settings Assignments Review + add

Included groups

- Add groups
- Add all users
- Add all devices

Groups

No groups selected

Excluded groups

When excluding groups, you cannot mix user and device groups across include and exclude. [Click here to learn more about excluding groups.](#)

Add groups

Groups

No groups selected

Previous Next Select

Script Assignments

6. After the operations above are complete, the configuration file will be written to the corresponding Wave directory after the employee's computer check-in. The configuration parameters will take effect after the Wave client starts.