Managing Client States via the FileWave API

What

This article explains how to manage client device states in FileWave, specifically focusing on how to archive and reinstate clients using the FileWave API. The client state can be defined as Tracked, Archived, Missing, or Untracked, each represented by a numerical value.

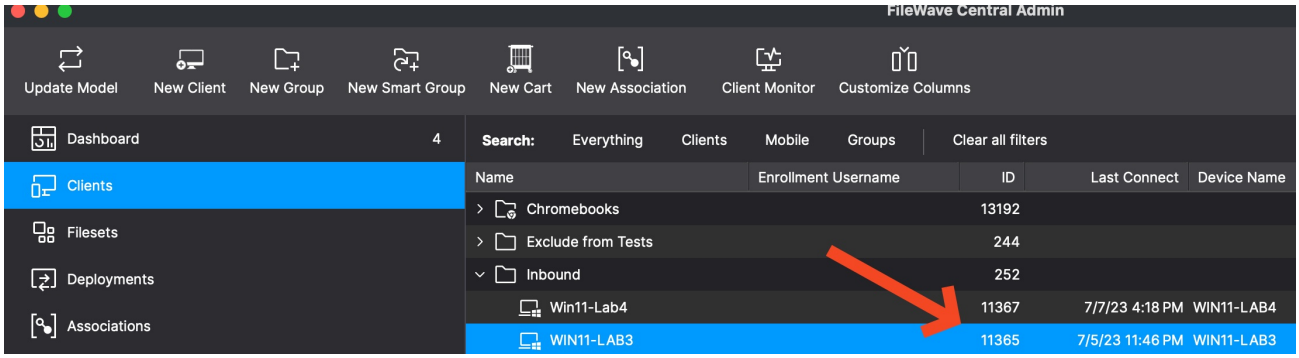
When/Why

Changing the state of a client may be necessary during device management tasks such as inventory control, security audits, or when a device is no longer in active use but needs to be retained in the system for record-keeping. Archiving clients helps in decluttering the active management list without permanently deleting the device record, allowing for easy reinstatement if needed.

> ⚠ Note that Apple MDM enrolled devices will break their MDM enrollment upon being Archived so they can't as easily be reinstated simply by changing their state.

How

To change the state of a device, you can use the FileWave API to send a PATCH request that updates the device state. Below is a script using zsh followed by a PowerShell script to change the state of a device and ensure the model is updated to reflect this change. The DeviceID can be seen in FileWave Central as shown in the below image or in FileWave Anywhere you will see it in the URL when looking at a device.



Shell script:

```
#!/bin/zsh

# Variables
ServerURL="https://fwjoshlab.filewave.net"   # Replace with your server address.
Token="your_token_here"  # Replace 'your_token_here' with your actual token.
DeviceID="11365"          # Specify the device ID.
NewState="0"              # Set the desired state (0: Tracked, 1: Archived, 2: Missing, 3: Untracked).

# Update device state
curl -X PATCH "$ServerURL/filewave/api/devices/v1/devices/$DeviceID" \
     -H "Authorization: Bearer $Token" \
     -H 'Content-Type: application/json' \
     -d '{"state":'$NewState'}'

# Update the model to reflect changes
curl -X POST "$ServerURL/filewave/api/fwserver/update_model" \
     -H "Authorization: Bearer $Token" \
     -H 'Content-Type: application/json'
```

PowerShell:

```
# PowerShell Script to Manage FileWave Client States

# Variables
$ServerURL = "https://fwjoshlab.filewave.net" # Replace with your server address.
```

```
$Token = "your_token_here"  # Replace 'your_token_here' with your actual token.
$DeviceID = "11365"         # Specify the device ID.
$NewState = "0"             # Set the desired state (0: Tracked, 1: Archived, 2: Missing, 3: Untracked).

# Headers for authorization and content type
$headers = @{
    "Authorization" = "Bearer $Token"
    "Content-Type" = "application/json"
}

# Body data for changing the state
$body = @{
    "state" = $NewState
} | ConvertTo-Json

# Update device state
Invoke-RestMethod -Uri "$ServerURL/filewave/api/devices/v1/devices/$DeviceID" -Method Patch -Headers $headers -
Body $body

# Update the model to reflect changes
Invoke-RestMethod -Uri "$ServerURL/filewave/api/fwserver/update_model" -Method Post -Headers $headers

# Output for user confirmation
Write-Host "Device state updated and model refreshed successfully."
```

Tips:

- Ensure that the `Token` variable contains a valid authorization token.
- Replace `DeviceID` and `NewState` with the appropriate values according to your needs.

Related Links

- FileWave API Documentation - Official API documentation.
- CURL Command Line Tool - Learn more about how to use curl.

Digging Deeper

Understanding the model update process is crucial for ensuring that changes made via the API are reflected in the FileWave management interface. The `update_model` API call triggers the FileWave server to reprocess its internal data models, ensuring that any state changes are accurately shown in the admin console. This is especially important after bulk changes to device states to maintain consistency across the system.

Revision #3
★Created 22 April 2024 17:14:26 by Josh Levitsky
Updated 22 April 2024 17:23:04 by Josh Levitsky