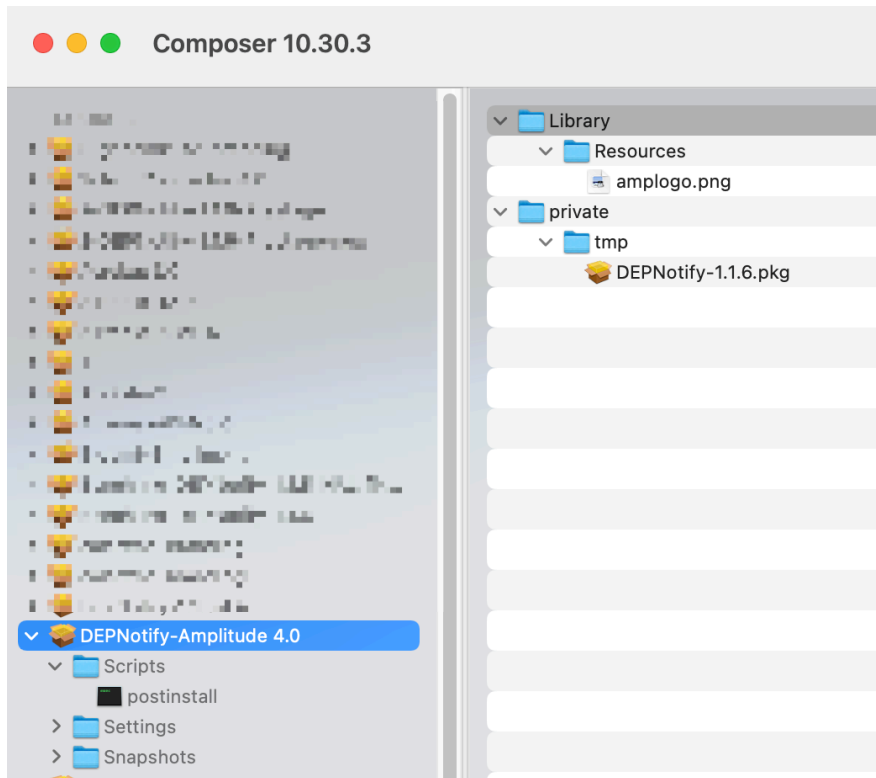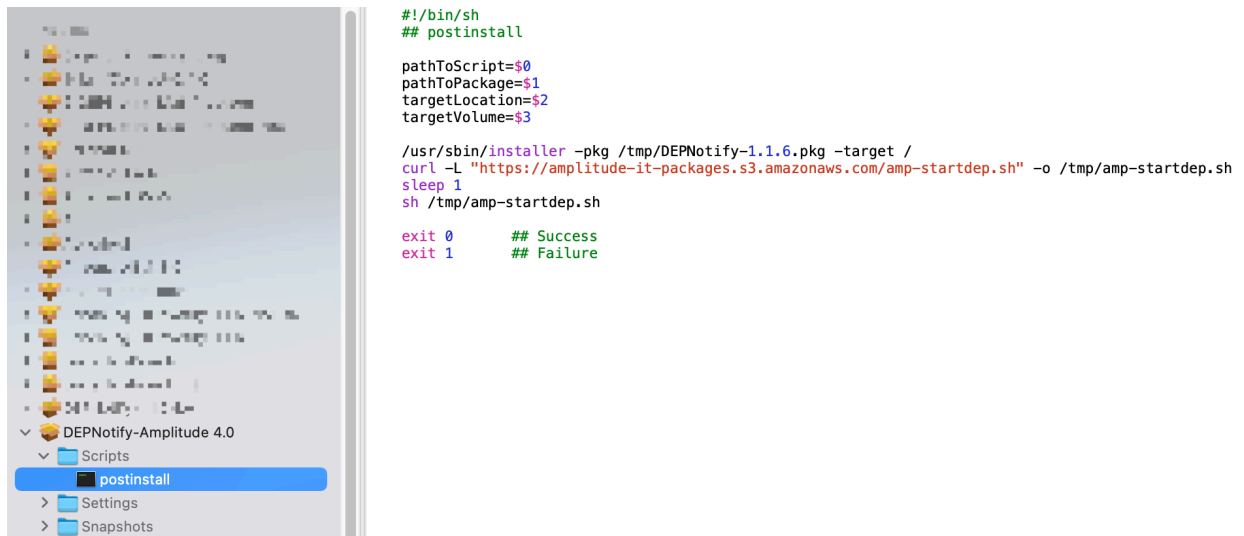# DEPNotify + Kandji

This is the custom PKG with the DEPNotify and your org logo.

This is the script inside the PKG with the AWS link to download the script.


```
#!/bin/sh
## postinstall

pathToScript=$0
pathToPackage=$1
targetLocation=$2
targetVolume=$3

/usr/sbin/installer -pkg /tmp/DEPNotify-1.1.6.pkg -target /
curl -L "https://amplitude-it-packages.s3.amazonaws.com/amp-startdep.sh" -o /tmp/amp-startdep.sh
sleep 1
sh /tmp/amp-startdep.sh

exit 0          ## Success
exit 1          ## Failure
```

DEPNotify + Kandji

These are the values in the amp-script.sh that may need to be updated at some point.

BID is the blueprint ID of the blueprint we are moving devices to.
This can be found in the url of the blueprint when you open it in Kandji console.
The Bearer is a API password for the API calls.
The path to Liftoff may be updated at some point contact Kandji support for help with this.
I added some commands to ensure the MDM profiles install once they device has been moved to a new blueprint.

```bash
# Popup 2
################################################################################
  # Label for the popup
    REG_POPUP_LABEL_2="Are you an Engineer?"

  # Array of options for the user to select
    REG_POPUP_LABEL_2_OPTIONS=(
      "NO"
      "YES"


    )

  # Help Bubble for Input. If title left blank, this will not appear
    REG_POPUP_LABEL_2_HELP_TITLE="Engineer Question"
    REG_POPUP_LABEL_2_HELP_TEXT="Please answer the question."

  # Logic below was put in this section rather than in core code as folks may
  # want to change what the field does. This is a function that gets called
  # when needed later on. BE VERY CAREFUL IN CHANGING THE FUNCTION!
    REG_POPUP_LABEL_2_LOGIC (){
      REG_POPUP_LABEL_2_VALUE=$(/usr/bin/defaults read "$DEP_NOTIFY_USER_INPUT_PLIST" "$REG_POPUP_LABEL_2")
      echo "Status: $REGISTRATION_BEGIN_WORD $REG_POPUP_LABEL_2 $REGISTRATION_MIDDLE_WORD $REG_POPUP_LABEL_2_VALUE" >> "$DEP_NOTIFY_LC
      if [ "$TESTING_MODE" = true ]; then
          sleep 10
      else

      if [ $REG_POPUP_LABEL_2_VALUE = "YES" ]; then
          echo $REG_POPUP_LABEL_2_VALUE >> "$DEP_NOTIFY_LOG"
          BID="███████████████████████"
          echo "Blueprint ID for YES $BID" >> "$DEP_NOTIFY_LOG"
      fi

      if [ $REG_POPUP_LABEL_2_VALUE = "NO" ]; then
          echo $REG_POPUP_LABEL_2_VALUE >> "$DEP_NOTIFY_LOG"
          BID="███████████████████████"
          echo "Blueprint ID for NO $BID" >> "$DEP_NOTIFY_LOG"
      fi

      echo "Blueprint ID Final is $BID" >> "$DEP_NOTIFY_LOG"
      ID=$(defaults read /Library/Preferences/io.kandji.Kandji.plist ComputerURL)
      ID2=$(echo ${ID##*/})
      echo "Computer URL is $ID and Device ID is $ID2" >> "$DEP_NOTIFY_LOG"
      string="https://████████.clients.us-1.kandji.io/api/v1/devices/"
      string="${string}${ID2}/"
      echo "String is $string" >> "$DEP_NOTIFY_LOG"
      echo "String fixed is $string" >> "$DEP_NOTIFY_LOG"
      curl --location --request PATCH $string \
      --header 'Content-Type: application/json' \
      --header 'Authorization: Bearer ████████████████████████' \
      --data-raw '{
          "blueprint_id": "'"$BID"'"
      }'

      #Update Device
      string="${string}${ID2}/action/updateinventory"
      curl --location --request POST $string \
      --header 'Content-Type: application/json' \
      --header 'Authorization: Bearer ████████████████████████'

      #Send Blank Push
      string="${string}${ID2}/action/blankpush/"
      curl --location --request POST $string \
      --header 'Content-Type: application/json' \
      --header 'Authorization: Bearer ████████████████████████'

      /usr/local/bin/kandji update-mdm

      open -a /Library/Kandji/Kandji\ Agent.app/Contents/Resources/Helpers/Kandji\ Liftoff.app
      sleep 4
      killall DEPNotify
      fi

    }

# Popup 3 - Code is here but currently unused
################################################################################
```