

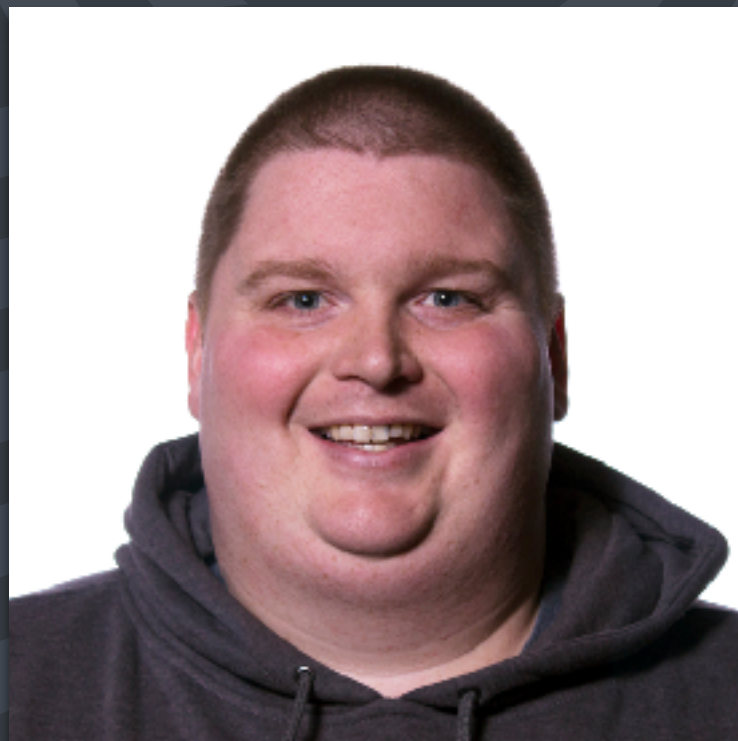


NATION

User Conference

OCTOBER 24-26

2017



Josh Roskos

Professional Services Engineer

 github.com/kc9wwh



Bryson Tyrrell

Systems Administrator

 github.com/brysontryrrell



API Scripting Lab

Tips & Tricks

Loopity Loops & Arrays

Modifying Scope

Let's Script Something



House Keeping Items...

Computers

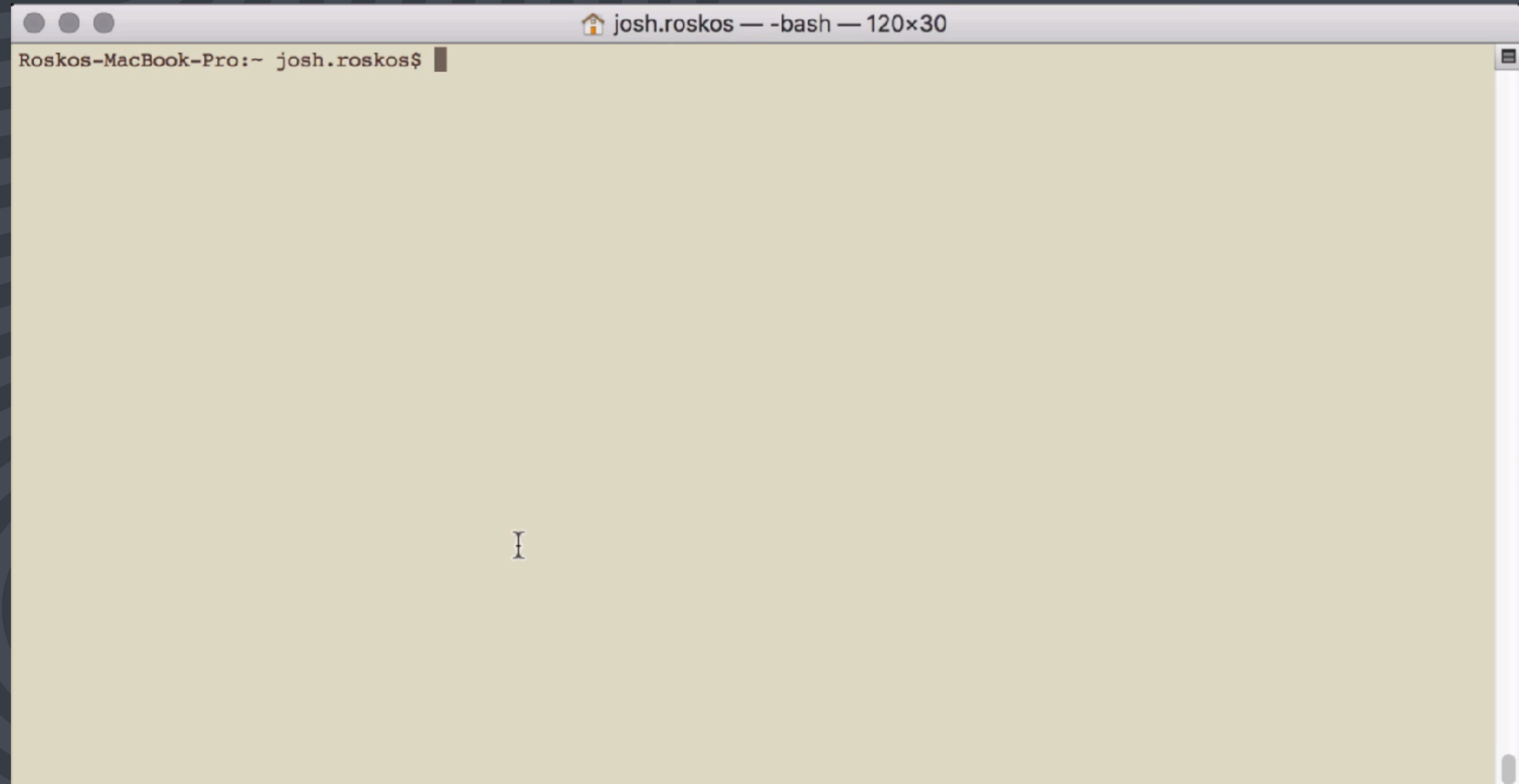
- Logon: localadmin / jamf1234
- /Users/Shared/Resources/...
 - Code Editors
 - Sample API Script

Jamf Pro server

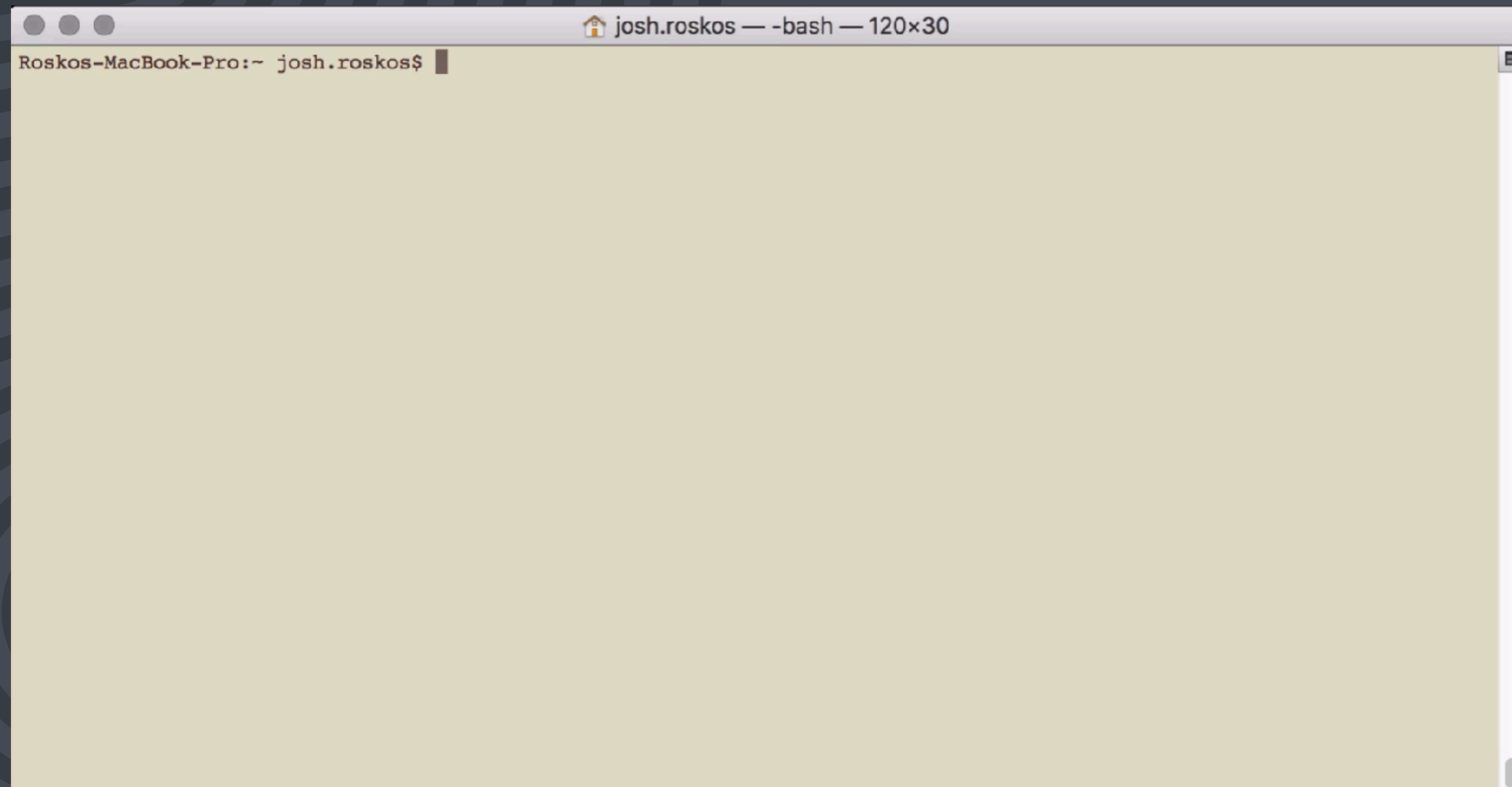
- <https://pro.jamf.training:8443/jnuc##>
- Logon: jamfadmin / jamf1234



Tips & Tricks



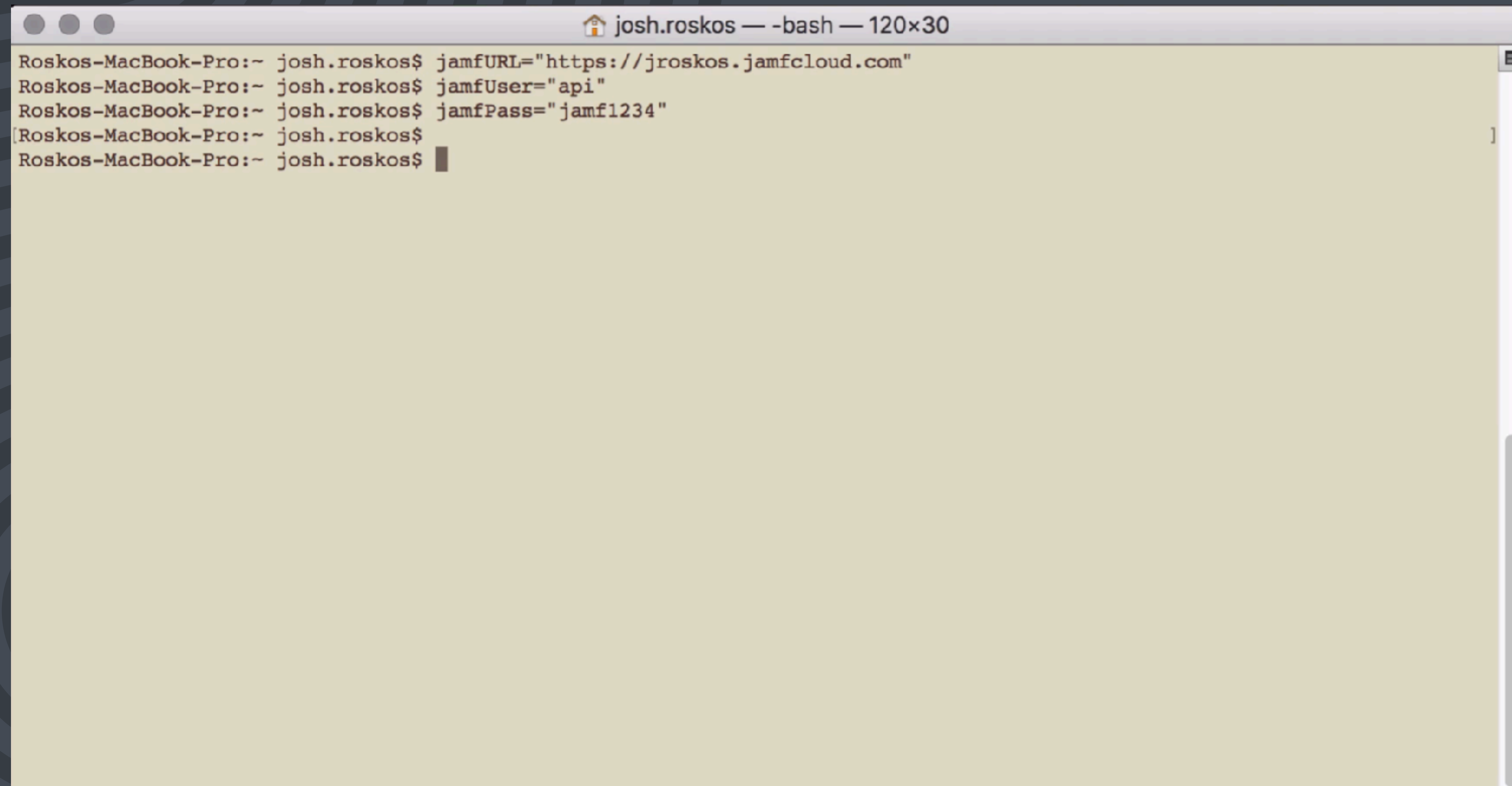
Tips & Tricks - xpath

A screenshot of a macOS terminal window. The title bar at the top reads 'josh.roskos — -bash — 120x30'. The terminal content shows the prompt 'Roskos-MacBook-Pro:~ josh.roskos\$' followed by a cursor. The terminal background is a light beige color.

```
Roskos-MacBook-Pro:~ josh.roskos$
```



Tips & Tricks - xpath

A terminal window titled 'josh.roskos — -bash — 120x30' is shown. The window has a light yellow background and a dark gray title bar. The terminal text shows a series of commands being entered at the 'Roskos-MacBook-Pro:~ josh.roskos\$' prompt. The commands are: 'jamfURL="https://jroskos.jamfcloud.com"', 'jamfUser="api"', and 'jamfPass="jamf1234"'. The fourth line shows the prompt again without a command, and the fifth line shows the prompt with a cursor. A vertical scrollbar is visible on the right side of the terminal window.

```
Roskos-MacBook-Pro:~ josh.roskos$ jamfURL="https://jroskos.jamfcloud.com"
Roskos-MacBook-Pro:~ josh.roskos$ jamfUser="api"
Roskos-MacBook-Pro:~ josh.roskos$ jamfPass="jamf1234"
Roskos-MacBook-Pro:~ josh.roskos$
Roskos-MacBook-Pro:~ josh.roskos$
```



Loopity Loops & Arrays

```
josh.roskos — -bash — 120x30
Roskos-MacBook-Pro:~ josh.roskos$ jamfURL="https://jroskos.jamfcloud.com"
Roskos-MacBook-Pro:~ josh.roskos$ jamfUser="api"
Roskos-MacBook-Pro:~ josh.roskos$ jamfPass="jamf1234"
Roskos-MacBook-Pro:~ josh.roskos$
```



Modifying Scope

Ways to modify Scope:

- Directly add/remove devices
- Add/remove Smart/Static Groups
- Modify Static Groups
- Modify Extension Attributes for Smart Groups



Modifying Scope

Gotchas! Don't do it this way:

PUT /JSSResource/policies/id/{x}

```
<policy>
  <scope>
    <computers>
      <computer>
        <id>1</id>
      </computer>
    </computers>
  </scope>
</policy>
```



Modifying Scope

Where supported, do this:

PUT /JSSResource/policies/id/{x}

```
<policy>
  <scope>
    <computer_additions>
      <computer>
        <id>1</id>
      </computer>
    </computer_additions>
  </scope>
</policy>
```



Modifying Scope

Where supported, do this:

PUT /JSSResource/policies/id/{x}

```
<policy>
  <scope>
    <computer_deletions>
      <computer>
        <id>1</id>
      </computer>
    </computer_deletions>
  </scope>
</policy>
```



Modifying Scope

Similar gotcha with Groups:

PUT /JSSResource/policies/id/{x}

```
<policy>
  <scope>
    <computer_groups>
      <computer_group>
        <id>1</id>
      </computer_group>
    </computer_groups>
  </scope>
</policy>
```



Modifying Scope

Modify the Group, not the object:

PUT /JSSResource/computergroups/id/{x}

PUT /JSSResource/mobiledevicegroups/id/{x}

```
<mobile_device_group>
  <mobile_device_additions>
    <mobile_device>
      <id>1</id>
    </mobile_device>
  </mobile_device_additions>
</mobile_device_group>
```

```
<computer_group>
  <computer_additions>
    <computer>
      <id>1</id>
    </computer>
  </computer_additions>
</computer_group>
```



Modifying Scope

You don't modify Smart Groups:

PUT /JSSResource/computers/id/{x}

```
<computer>
  <extension_attributes>
    <extension_attribute>
      <id>1</id>
      <value>MyNewValue</value>
    </extension_attribute>
  </extension_attributes>
</computer>
```



Let's Script Something Already!

How can I see what is scoped to my computer groups?



#!/bin/bash

#####

#

What is scoped to my Computer Groups?

#

*# In this script we will utilize the Jamf Pro API to determine what Policies are assigned
to your Computer Groups.*

#

#

OBJECTIVES

- # - Create a list of all Smart Groups*
- # - Provide a list of what is scoped to each Smart Group*

#

#####

#####

USER VARIABLES




```
#####
```

```
# APPLICATION
```

```
#####
```

```
## Retrieve & Extract Computer Group IDs/Names & Build Array
```

```
compGroupData=$( curl -k -s -u "$jamfUser":"$jamfPass" $jamfURL/JSSResource/  
computergroups -H "Accept: application/xml" -X GET )
```

```
compGrpSize=$(echo $compGroupData | xpath "//computer_groups/size/text())"
```

```
index=0
```

```
declare -a compGrpNames
```

```
declare -a compGrpIDs
```

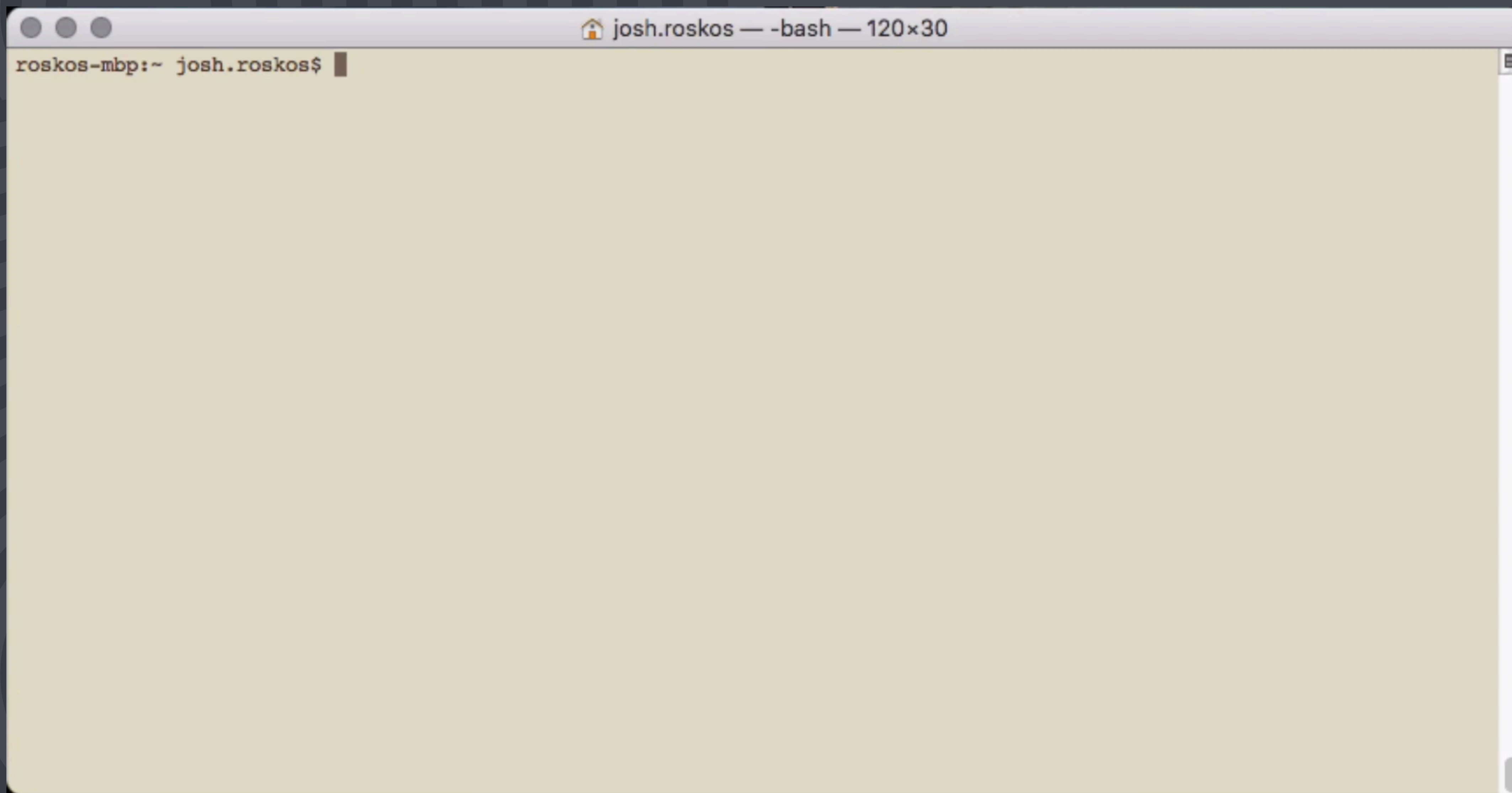
```
while [ $index -lt ${compGrpSize} ]; do
```

```
    element=$((index+1))
```

```
    compGrpName=$( echo $compGroupData | xpath "//computer_groups/  
computer_group[${element}]/name/text()" )
```

```
    compGrpID=$( echo $compGroupData | xpath "//computer_groups/computer_group[${  
{element}]/id/text()" )
```

```
    echo "Computer Group ID: $compGrpID" 
```


A screenshot of a macOS terminal window. The title bar at the top shows a home icon, the text "josh.roskos", and "-bash" followed by the window dimensions "120x30". The terminal area has a light beige background. The first line of text is "roskos-mbp:~ josh.roskos\$". A dark cursor is positioned at the end of this line. There is a small icon in the top right corner of the terminal area and a scrollbar on the right side.

```
roskos-mbp:~ josh.roskos$
```



Retrieve & Filter Policy Data

```
echo "Retrieving List of All Policy IDs..."
```

```
unset policyIDs
```

```
policyIDs=( $( curl -k -s -u "$jamfUser":"$jamfPass" $jamfURL/JSSResource/policies -H  
"Accept: application/xml" -X GET | /usr/bin/perl -lne 'BEGIN{undef $/} while (/<id>(.*?)</id>/sg){print $1}' ) )
```

```
for i in "${policyIDs[@]"; do
```

```
    echo "Retrieving Policy ID ${i}'s' Data..."
```

```
    policyData=$( curl -k -s -u "$jamfUser":"$jamfPass" $jamfURL/JSSResource/policies/id/${i} -H "Accept: application/xml" -X GET )
```

```
    policyName=$( echo $policyData | xpath "//policy/general/name/text()" )
```

```
    ## Check if is a Jamf Remote Policy
```

```
    echo "Checking if this is a Jamf Remote Policy..."
```

```
    if [[ $policyName == $( echo $policyName | egrep -B1 '[0-9]+-[0-9]{2}-[0-9]{2} at [0-9]{1,2}:[0-9]{2,2} [AP]M \. * \. *' ) ]]; then
```

```
        ## This is a Jamf Remote Policy
```

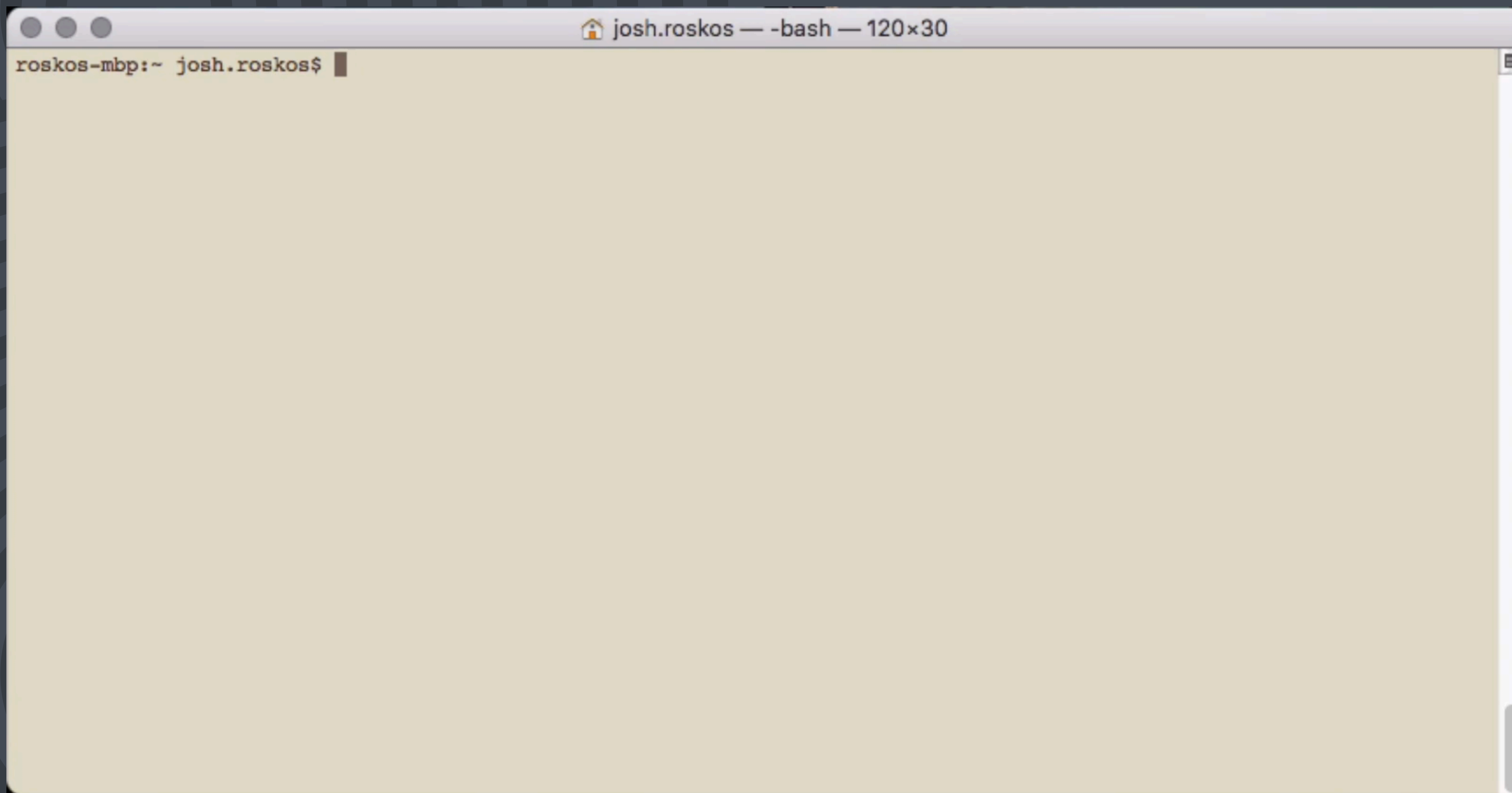
```
        ## Setting policy name in array to "JamfRemotePolicy-Ignore"
```

```
        echo "    This is a Jamf Remote policy"
```

```
        continue
```

```
    else
```



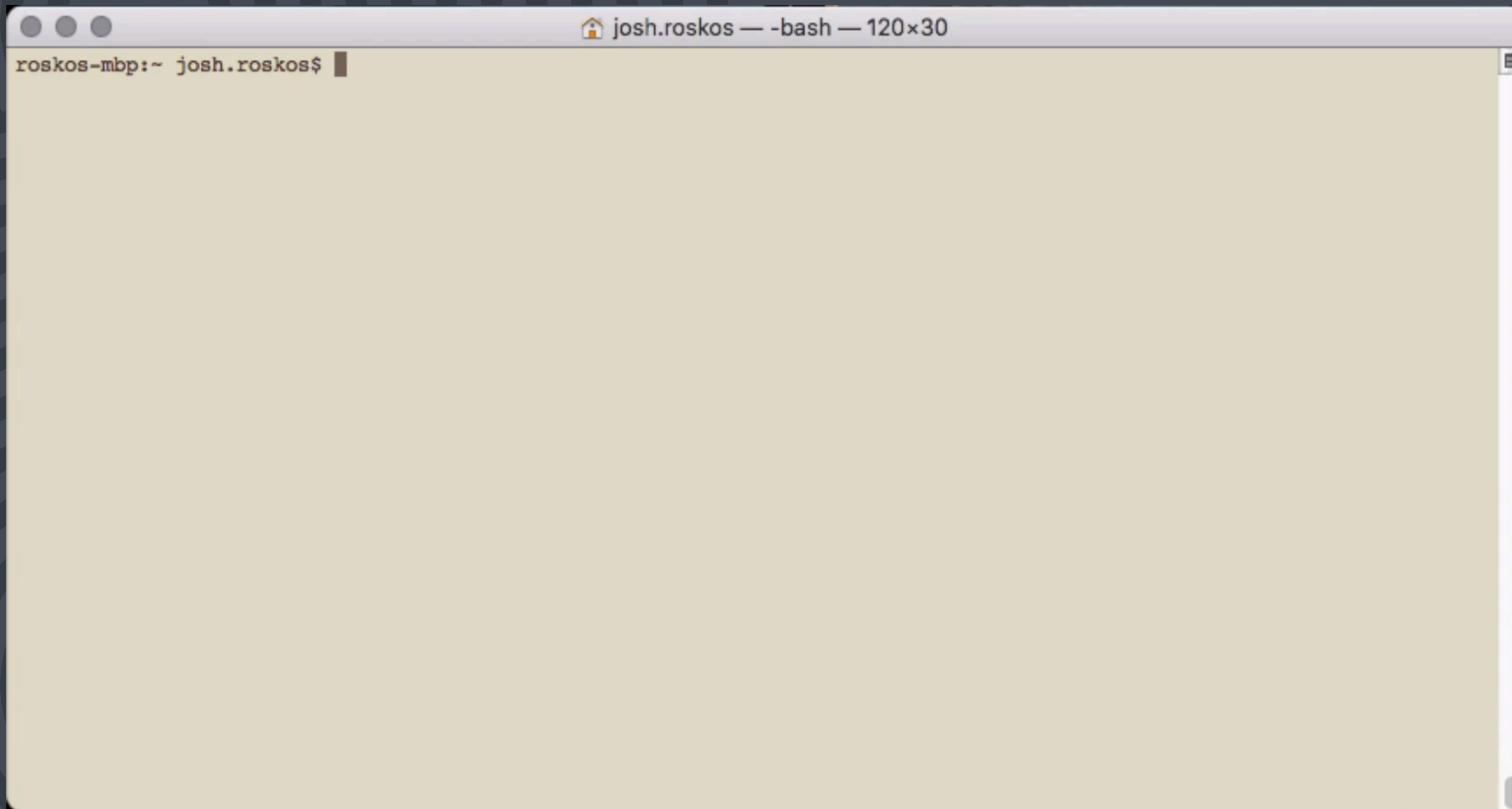
A screenshot of a macOS terminal window. The title bar at the top shows a home icon, the text "josh.roskos", and "-bash" followed by a separator and "120x30". The terminal area has a light beige background. The first line of text is "roskos-mbp:~ josh.roskos\$". A dark cursor is positioned at the end of the second line.

```
roskos-mbp:~ josh.roskos$
```



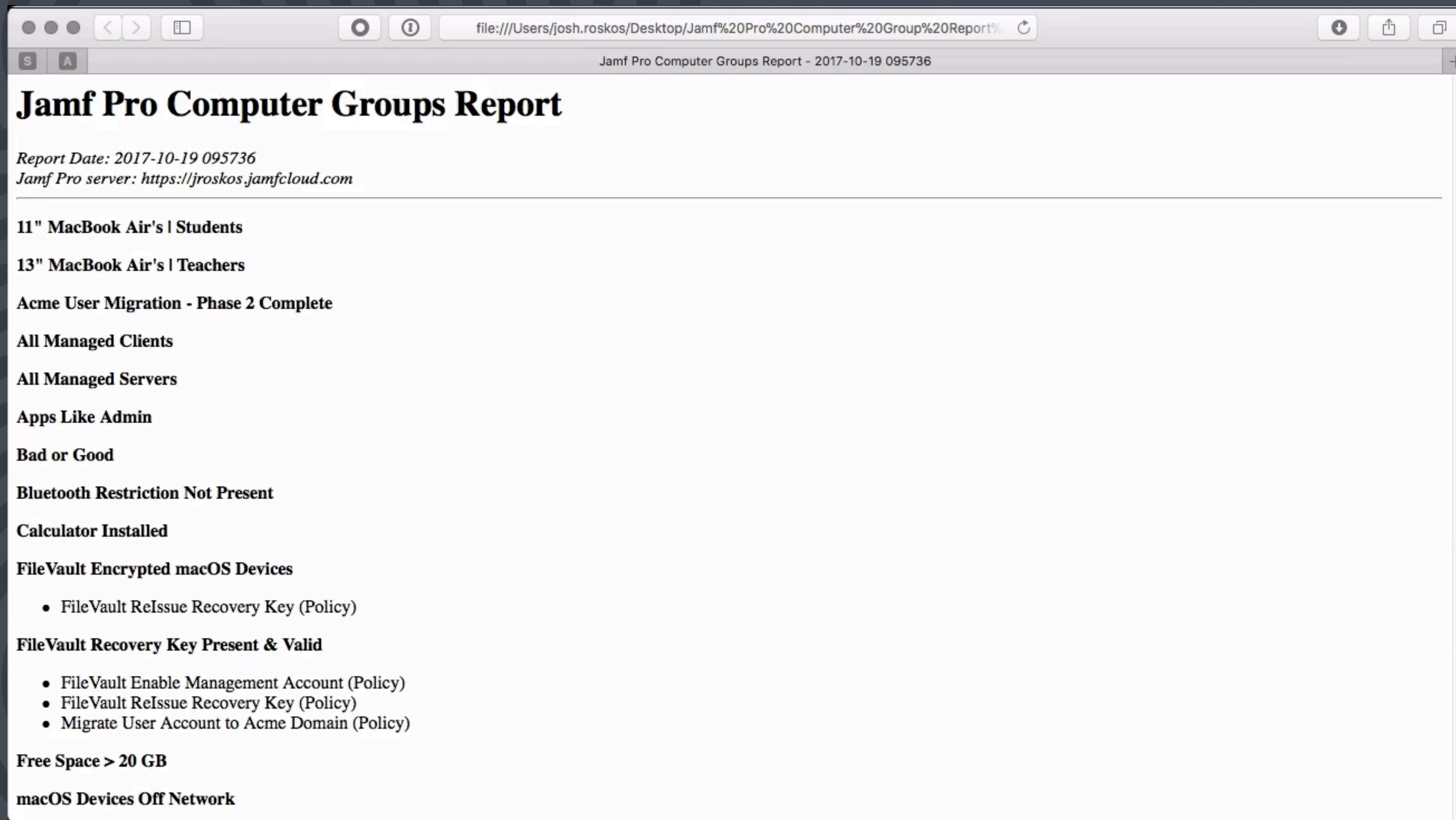

```
#####
# BUILD HTML REPORT
#####
```

```
reportDate=$(/bin/date "+%Y-%m-%d %H%M%S")
reportName="/Users/${currentUser}/Desktop/Jamf Pro Computer Group Report -
${reportDate}.html"
echo "<html>
<head>
<title>Jamf Pro Computer Groups Report - ${reportDate}</title>
</head>
<body>
<h1>Jamf Pro Computer Groups Report</h1>
<i>Report Date: ${reportDate}<br/>Jamf Pro server: ${jamfURL}</i>
<hr/>
<p/>" > "${reportName}"
for (( x = 0 ; x < ${#compGrpNames[@]} ; x++ )); do
    echo "<b>${compGrpNames[$x]}</b><br/><ul>" >> "${reportName}"
    groupID="${compGrpIDs[$x]}"
```

A screenshot of a macOS terminal window. The title bar at the top shows a home icon, the text "josh.roskos", and a separator followed by "-bash" and "120x30". The terminal area has a light yellow background and shows the prompt "roskos-mbp:~ josh.roskos\$" with a dark cursor. There is a small icon in the top right corner of the terminal area and a scrollbar on the right side.

```
roskos-mbp:~ josh.roskos$
```





The screenshot shows a web browser window displaying a Jamf Pro report. The address bar shows a file path: `file:///Users/josh.roskos/Desktop/Jamf%20Pro%20Computer%20Group%20Report%`. The browser tabs show 'S' and 'A'. The report title is 'Jamf Pro Computer Groups Report'. Below the title, the report date is '2017-10-19 095736' and the Jamf Pro server is 'https://jroskos.jamfcloud.com'. The report content is a list of computer groups and their status:

- 11" MacBook Air's | Students
- 13" MacBook Air's | Teachers
- Acme User Migration - Phase 2 Complete
- All Managed Clients
- All Managed Servers
- Apps Like Admin
- Bad or Good
- Bluetooth Restriction Not Present
- Calculator Installed
- FileVault Encrypted macOS Devices
 - FileVault ReIssue Recovery Key (Policy)
- FileVault Recovery Key Present & Valid
 - FileVault Enable Management Account (Policy)
 - FileVault ReIssue Recovery Key (Policy)
 - Migrate User Account to Acme Domain (Policy)
- Free Space > 20 GB
- macOS Devices Off Network



Q&A



Thank you!

