Outlook for Mac Exchange Setup 5.0 Administrator’s Guide

William Smith

bill@officeformachelp.com

© Copyright 2008-2015 William Smith

[By_plain300.png](http://creativecommons.org/licenses/by/3.0/)

1. Introduction

Adding an Exchange account to Microsoft Outlook 2016 for Mac is not always easy for the typical end-user. A properly configured network will help Outlook connect using Exchange’s Autodiscover feature but that’s not always available.

Outlook *Exchange Setup 5.0* is a set of scripts and plist files to help Mac administrators configure Exchange accounts in Outlook when the Autodiscover service doesn’t work. The entire user experience should be as simple as launching Outlook for the first time and entering a password. Just a few seconds.

Every company's network is different but the primary script should provide enough ideas for a novice scripter to adapt it for use in his or her network in just a few minutes.

1. Quickstart
2. Locate the Outlook Exchange Setup 5.0.scpt AppleScript found in Outlook Exchange 5.0 > Package > ROOT > Library > Talking Moose Industries > Scripts.
3. Open the script in Script Editor and modify each property in the Begin network, server and preferences section of the script. Save and close the script when done.
4. Locate the Package for Deployment app found in Outlook Exchange 5.0 > Package.
5. Double-click the app and follow the prompts to create an Apple Installer package with all the necessary components for deployment.
6. Distribute the package to Mac workstations using any tool that supports deploying Apple Installer packages such as Apple Remote Desktop, Casper or munki.
7. How it works

*Outlook Exchange Setup 5.0* uses four files: a configuration profile, a launchd agent, a shell script and an AppleScript.

When the user logs in to his or her OS X account, the launchd agent checks for the existence of Office 2016 preferences in the current user's folder. If nothing exists, it calls the shell script to pre-create a path and create a second user-level launchd agent to watch the path.

When the user opens Outlook for the first time, the pre-created path changes. When the user-level launchd agent notices the change, it triggers the AppleScript, which does the account configuration. On successful completion, the AppleScript removes the user-level launchd agent.

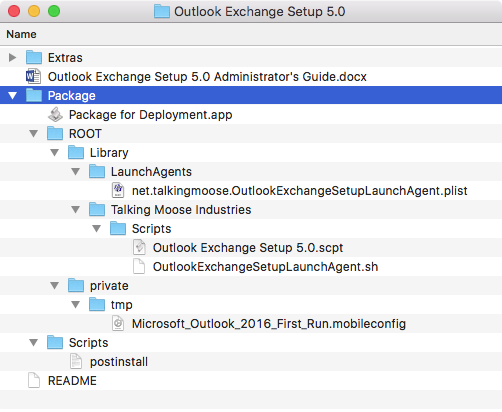
The configuration profile (.mobileconfig file) hides any first run windows and dialogs that would hinder the AppleScript.

From the user’s point-of-view he or she simply launches Outlook for Mac and enters a password to access email.

1. Prepare the Script

This section guides you through configuring the Outlook *Exchange Setup 5.0* files.

Note: Be sure to preserve the Package folder's structure inside the Outlook Exchange Setup 5.0 folder you downloaded. It's designed to facilitate creating an Apple Installer package for deployment.



Preserve the folder and file structure inside the Package folder.

Server, Preferences and Custom Message Properties

Locate the Outlook Exchange Setup 5.0.scpt AppleScript found in Outlook Exchange Setup 5.0 > Package > ROOT > Library > Talking Moose Industries > Scripts.

Double-click the AppleScript to open in Script Editor found in Applications > Utilities.

The beginning of the script contains comments for the reader between a set of (\* and \*) markers. Everything between these markers is ignored when the script is executed.

Throughout the script are additional comments beginning with -- (double-hyphens or multiple hyphens). These are also comments and are ignored when the script runs.

The only section you need to customize is between:

--------------------------------------------

-- Begin network, server and preferences

--------------------------------------------

and

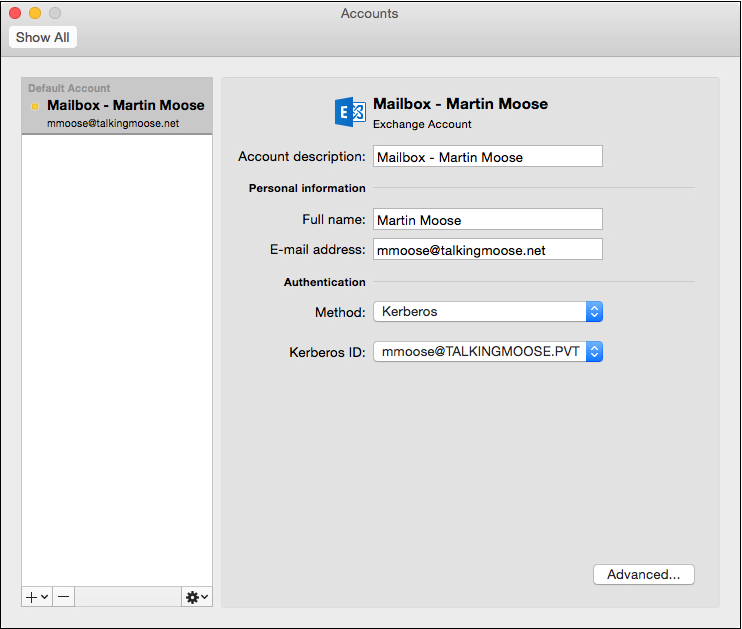
--------------------------------------------

-- End network, server and preferences

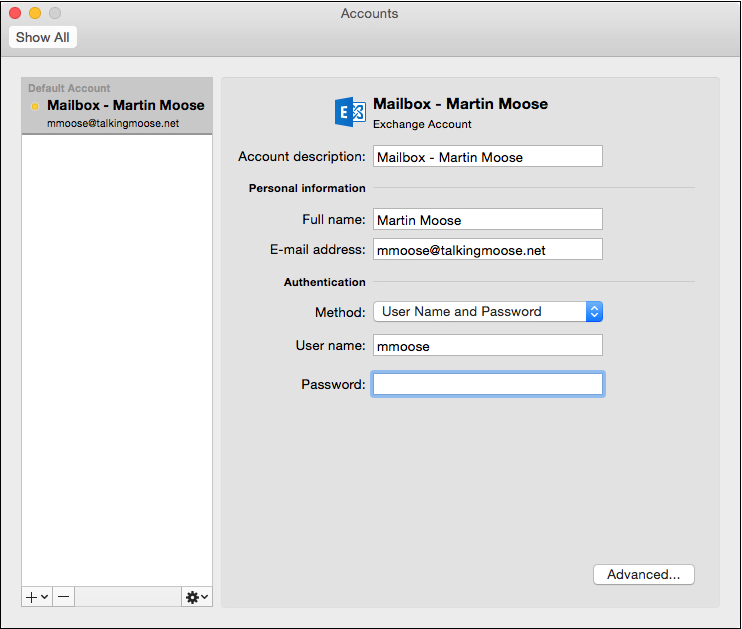
--------------------------------------------

You can customize 23 properties in this section. Each property is a setting or preference the script uses when run. The property names reflect their purposes and include descriptions below each one.

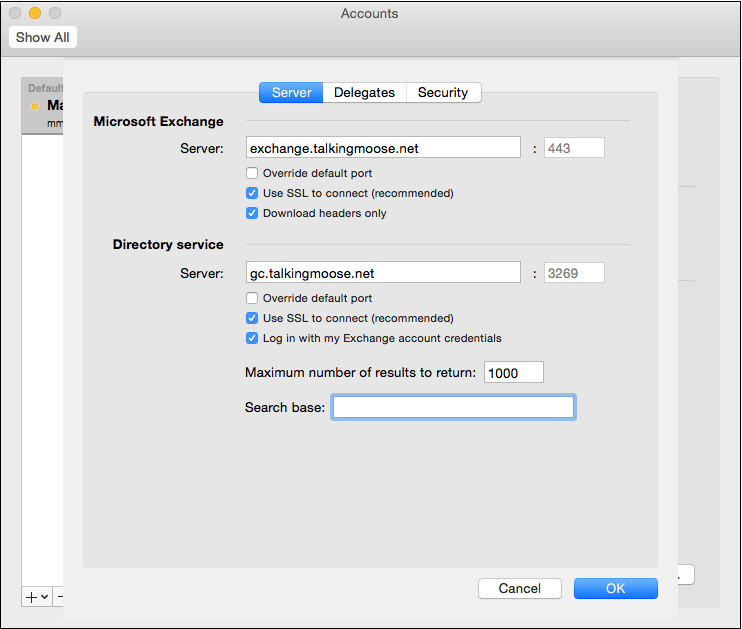
Most every property corresponds to an Exchange account authentication method or preference. You’ll find the Accounts pane under Outlook > Preferences (See Figures 1, 2 and 3).

 Figure . Exchange account using Kerberos authentication.

|  |  |
| --- | --- |
| **Method** | **property** useKerberos : *true*  -- Set this to true only if Macs in your environment are bound  -- to Active Directory and your network is properly configured. |
| **Kerberos ID** | The script queries Active Directory for the Kerberos ID. |

 Figure . Exchange account using standard authentication.

|  |  |
| --- | --- |
| **Method** | **property** useKerberos : *false*  -- Set this to true only if Macs in your environment are bound  -- to Active Directory and your network is properly configured. |
| **User name** | The script queries Active Directory for the Kerberos ID. |
| **Password** | Outlook prompts the user for a password when connecting to Exchange. |

 Figure . Exchange and Directory server settings.

|  |  |
| --- | --- |
| **Microsoft Exchange Server** | **property** ExchangeServer : "exchange.example.com"  -- Address of your organization's Exchange server. |
| **Microsoft Exchange Server** | **property** ExchangeServerRequiresSSL : *true*  -- True for most servers. |
| **Microsoft Exchange Server** | **property** ExchangeServerSSLPort : 443  -- If ExchangeServerRequiresSSL is true set the port to 443.  -- If ExchangeServerRequiresSSL is false set the port to 80.  -- Use a different port number only if your administrator instructs you. |

|  |  |
| --- | --- |
| **Directory Service Server** | **property** DirectoryServer : "gc.example.com"  -- Address of an internal Global Catalog server (a type of Windows domain controller).  -- The LDAP server in a Windows network will be a Global Catalog server,  -- which is separate from the Exchange Server. |
| **Directory Service Server** | **property** DirectoryServerRequiresAuthentication : *true*  -- This will almost always be true. |
| **Directory Service Server** | **property** DirectoryServerRequiresSSL : *true*  -- This will almost always be true. |
| **Directory Service Server** | **property** DirectoryServerSSLPort : 3269  -- If DirectoryServerRequiresSSL is true set the port to 3269.  -- If DirectoryServerRequiresSSL is false set the port to 3268.  -- Use a different port number only if your Exchange administrator instructs you. |
| **Directory Service Server** | **property** DirectoryServerMaximumResults : 1000  -- When searching the Global Catalog server, this number determines  -- the maximum number of entries to display. |
| **Directory Service Server** | **property** DirectoryServerSearchBase : ""  -- example: "cn=users,dc=domain,dc=com"  -- Usually, this is empty. |
| **For Active Directory Users** | **property** getUserInformationFromActiveDirectory : *true*  -- If Macs are connected to Active Directory they can probably use  -- dscl to return the current user's Email address, phone number, title, etc.  -- Use Active Directory when possible. Otherwise, set this to false  -- and set the next two properties to the appropriate numbers. |
| **For Non Active Directory Users** | **property** domainName : "example.com"  -- Complete this only if not using Active Directory to get user information.  -- The part of your organization's email address following the @ symbol. |
| **For Non Active Directory Users** | **property** emailFormat : 1  -- When Active Directory is unavailable to determine a user's Email address,  -- this script will attempt to parse it from the display name of the user's login.  -- Describe your organization's email format:  -- 1: Email format is first.last@domain.com  -- 2: Email format is first@domain.com  -- 3: Email format is flast@domain.com (first name initial plus last name)  -- 4: Email format is shortName@domain.com |
| **For Non Active Directory Users** | **property** displayName : 2  -- Describe how the user's display name appears at the bottom of the menu  -- when clicking the Apple menu (Log Out Joe Cool... or Log Out Cool, Joe...).  -- 1: Display name appears as "Last, First"  -- 2: Display name appears as "First Last" |
| **For Non Active Directory Users** | **property** domainPrefix : ""  -- Append a NetBIOS domain name to the beginning of the user's short name.  -- Be sure to use two backslashes when adding a name.  -- Example: Use "TALKINGMOOSE\\" to set user name "TALKINGMOOSE\username". |

By default, the script requires no user interaction. However, if users need to specify a different email server from the standard server or they need to enter a custom email address, then set these properties to true to prompt them to verify their settings.

|  |  |
| --- | --- |
| **User Experience** | **property** verifyEMailAddress : *false*  -- If set to "true", a dialog asks the user to confirm his Email address. |
| **User Experience** | **property** verifyServerAddress : *false*  -- If set to "true", a dialog asks the user to confirm his Exchange server address. |
| **User Experience** | **property** displayDomainPrefix : *false*  -- If set to "true", the username appears as "DOMAIN\username".  -- Otherwise, the username appears as "username". |

These settings may provide a better user experience. Be sure to customize the error message to include a Help Desk phone number or contact name.

|  |  |
| --- | --- |
| **User Experience** | **property** downloadHeadersOnly : *false*  -- If set to "true", only email headers are downloaded into Outlook.  -- This takes much less time to sync but a user must be online  -- to download and view messages. |
| **User Experience** | **property** hideOnMyComputerFolders : *false*  -- If set to "true", hides local folders.  -- A single Exchange account should do this by default. |
| **User Experience** | **property** unifiedInbox : *false*  -- If set to "true", turns on the Group Similar Folders feature  -- in Outlook menu > Preferences > General. |
| **User Experience** | **property** disableAutodiscover : *false*  -- If set to "true", disables Autodiscover functionality  -- for the Exchange account. Not recommended for mobile devices  -- that may connect to an internal Exchange server addresss and  -- connect to a different external Exchange server address. |
| **User Experience** | **property** errorMessage : "Outlook's setup for your Exchange account failed. Please contact the Help Desk for assistance."  -- Customize this error message for your users in case their account setup fails |

The Rest of the Script

The rest of the script requires no modifications unless the Mac administrator chooses to augment its function. If left as-is then it will proceed to:

1. Determine the user name and display name of the current user.
2. Determine the current user’s email address by using dscl or parsing it from login information.
3. Ask the user to verify his email address and short name if those options are enabled.
4. Configure the Exchange account.
5. Enable Kerberos, if that option is enabled, and set the principal name as short@kerberosRealm.
6. Populate the Me Contact record with the user’s data gathered from Active Directory.
7. Alert the user when an error occurs.

Saving the Script

Save the script as File Format: Script with name *Outlook Exchange Setup 5.0*.scpt. Users do not directly interact with this script. (See Figure 5)

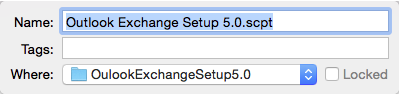


Figure 5. Save the script as Outlook Exchange Setup 5.0.scpt.

Note: If you change the name of this script file, be sure to update the name in the OutlookExchangeSetupLaunchAgent.sh shell script.

1. Test the Setup

Copy these files to a location such as Users > Shared on your test Mac for easy access.

* Microsoft\_Outlook\_2016\_First\_Run.mobileconfig
* net.talkingmoose.OutlookExchangeSetupLaunchAgent.plist
* OutlookExchangeSetupLaunchAgent.sh
* Outlook Exchange Setup 5.0.scpt

Install the configuration profile

Locate the Extras folder inside the Outlook Exchange Setup 5.0 folder. It contains two configuration profiles (.mobileconfig) and four property list (.plist) files for your reference.

Use a configuration profile to disable Outlook’s first run windows. This takes the user directly to Outlook's Main Window and lets the AppleScript proceed with configuring the Exchange account without interference.

If using a Mobile Device Management (MDM) server such as Apple’s Profile Manager *(See Figures 6 and 7)* or JAMF Software’s Casper Suite, create a new configuration profile with a custom payload. Upload the com.microsoft.Outlook.plist file to the custom payload. (Optionally, upload any of the other plist files from the Extras folder for Excel, PowerPoint and Word to disable their first run windows. These aren't required but provided as a convenience.)

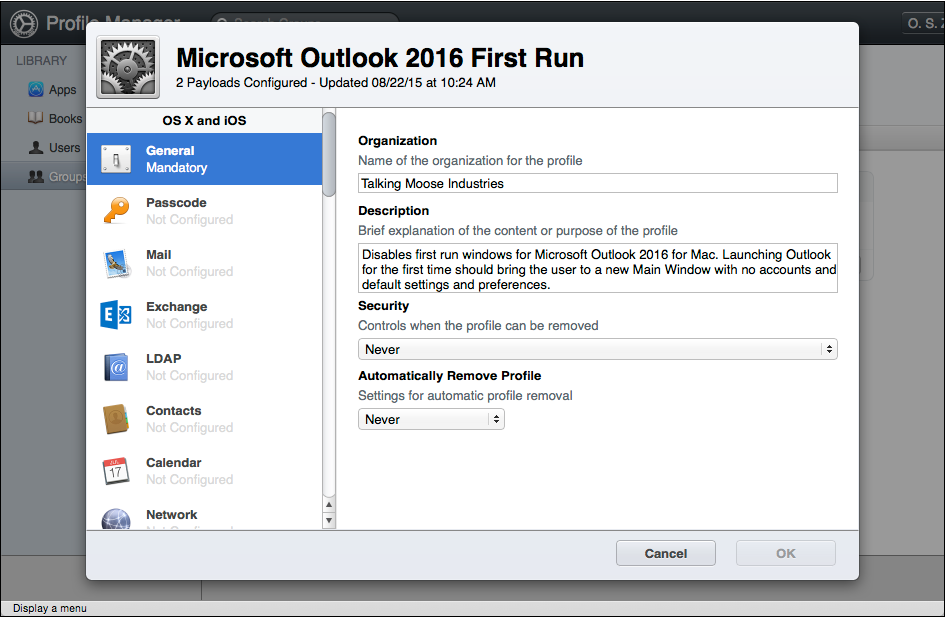


Figure 6. Profile Manager payload – General.

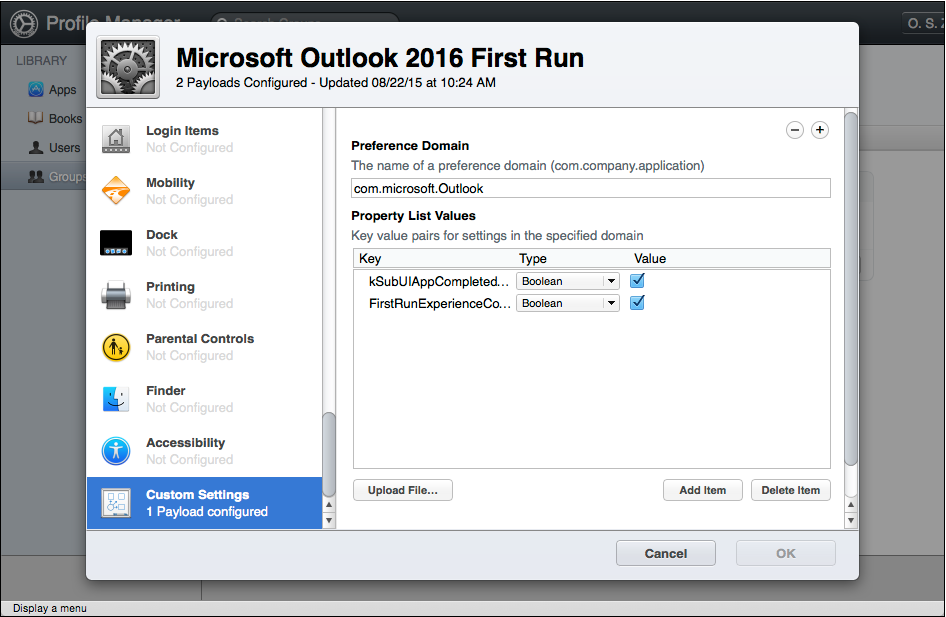


Figure 7. Profile Manager payload – Custom Settings.

Deploy the configuration profile to a test Mac over the air using Apple’s Push Notification service (APNs) or download and install the .mobileconfig file manually using Terminal (one line).

sudo profiles -IF /Users/Shared/Microsoft\_Outlook\_2016\_First\_Run.mobileconfig

It should appear in the Profiles pane of System Preferences under Device Profiles *(See Figure 8)*.

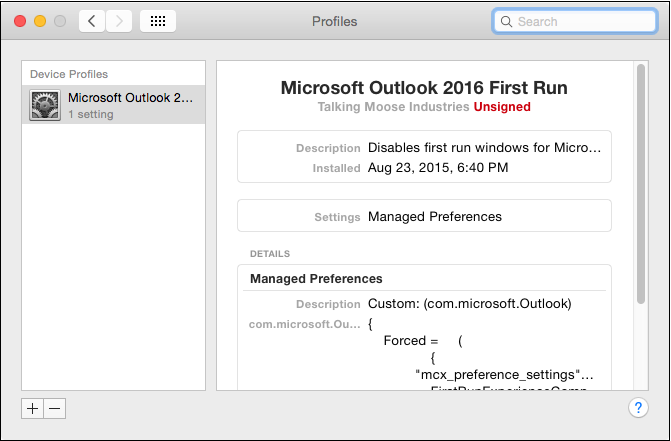


Figure 8. Installed configuration profile to disable Outlook’s first run windows.

Feel free to use the included configuration profiles. Keep in mind they will not reflect your company’s name when anyone views them in System Preferences. Instead, they will display **Talking Moose Industries Unsigned**.

Install the script files

Copy both the OutlookExchangeSetupLaunchAgent.sh shell script and prepared Outlook Exchange Setup 5.0.scpt AppleScript to Library > Talking Moose Industries > Scripts

Note: You may need to use the chmod command in Terminal to make the shell script executable (one line):

/usr/bin/sudo /usr/bin/chmod +x '/Library/Talking Moose Industries/Scripts/OutlookExchangeSetupLaunchAgent.sh'

Install the launch agent

Copy the net.talkingmoose.OutlookExchangeSetupLaunchAgent.plist file to Library > LaunchAgents.

Note: You will need to use the chown and chmod commands in Terminal to correctly set ownership and permissions on the file (two lines):

/usr/bin/sudo /usr/bin/chown root:wheel '/Library/LaunchAgents/net.talkingmoose.OutlookExchangeSetupLaunchAgent.plist'

/usr/bin/sudo /usr/bin/chmod 644 '/Library/LaunchAgents/net.talkingmoose.OutlookExchangeSetupLaunchAgent.plist'

Install Office 2016 for Mac

Deploy Office 2016 for Mac using your preferred deployment tool or manually install it by double-clicking the installer package.

Note: The Outlook Exchange Setup 5.0.scpt AppleScript assumes you’re using a volume license edition of Office and not Office 365. Office 365 not only requires activation but also supports Autodiscover. You don’t need this script where Autodiscover already works.

Create a test user

If your Mac is not bound to Active Directory, then open the Users & Groups pane of System Preferences and create a new local test user with a full name and short name. Be sure to adjust the Outlook Exchange Setup 5.0.scpt AppleScript's properties accordingly. Log in as the test user.

If your Mac is bound to Active Directory, then log in with a test Active Directory account. Be sure to adjust the Outlook Exchange Setup 5.0.scpt AppleScript's properties to use Active Directory for user information.

Test Microsoft Outlook

Launch Microsoft Outlook. Its Main Window should appear without displaying any first run windows. If they do appear then verify your configuration profile is correctly configured and installed in System Preferences.

Outlook should prompt the user for his or her Exchange account password. *(See Figure 9)*.

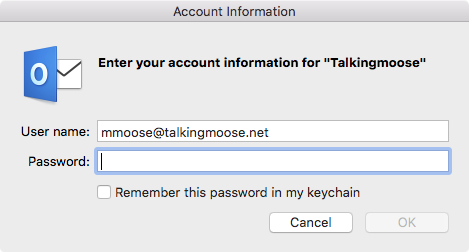


Figure 9. Outlook is ready to use.

Choose Outlook > Preferences > General. Verify whether “Group similar folders, such as Inboxes, from different accounts” matches your setting in the Outlook Exchange Setup 5.0.scpt AppleScript for the unifiedInbox property *(See Figure 10)*. Also verify “Hide On My Computer folders” matches your setting for the hideOnMyComputerFolders property.

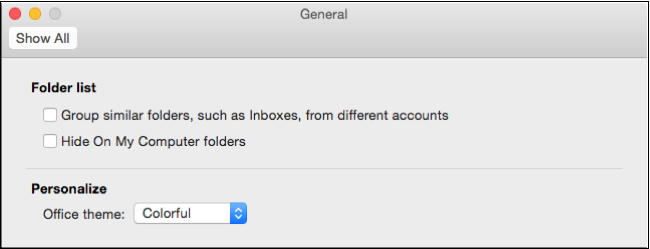


Figure 10. Verify the unifiedInbox property correctly set  
the “Group similar folders, such as Inboxes, from different accounts” setting.

Choose Outlook > Preferences > Accounts. Select your Exchange account and verify its settings *(See Figures 1, 2 and 3)*.

Choose View > Go To > People and then choose View > Me Contact. Verify whether the script correctly populated your user’s contact record. At minimum, it should display first Name, last name and email address. If using Active Directory, it may display more depending on how well your organization populates its user data *(See Figure 11)*.

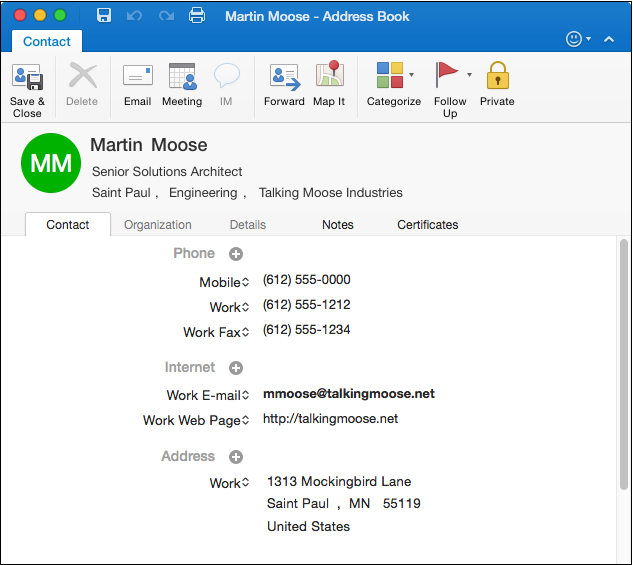


Figure 11. Me Contact record populated from Active Directory.

Continue testing with other accounts until you’re satisfied the script is working correctly for your environment. Then proceed to packaging the scripts, configuration profile and launch agent for deployment.

1. Package for Deployment

The final step is to package the scripts, configuration profile and launch agent for deployment.

The Package for Deployment app creates an Apple Installer package with the contents of the ROOT and Scripts folders. Use your favorite deployment tool to push your package to your target machines. Below are a few options:

* Apple Remote Desktop
* Casper
* munki

Run the Package for Deployment app

Locate the Package for Deployment app in Outlook Exchange Setup 5.0 > Package. Double-click the app and follow the prompts.

First, name your installer. Click the Next button.

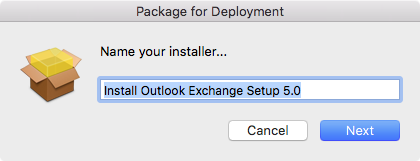


Figure 12. Name your installer.

Enter an identifier for your package. By default, the app takes the installer name and prepends a reverse domain name to the beginning. Follow the same naming convention and replace "net.talkingmoose" with the reverse domain name of your organization. Click the Next button.

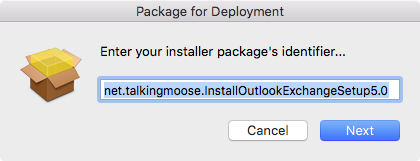


Figure 13. Name your installer.

Finally, provide a version number for your installer. This only appears in the package receipt but may be useful for other tools to help you identify the version of installer you've deployed to systems. Click the Build button.

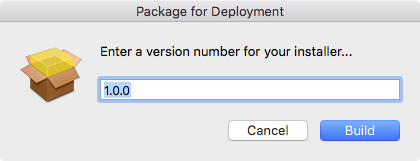


Figure 14. Name your installer.

After packaging, click the Show Package button to reveal your newly created installer. You'll find it in the same folders as the Package for Deployment app.

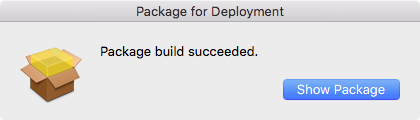


Figure 15. Click Show Package to select the newly created package in the Finder.

1. Appendix A

The Outlook Exchange Setup 5.0.scpt AppleScript:

(\*

--------------------------------------------

Outlook Exchange Setup 5.0

© Copyright 2008-2015 William Smith

bill@officeformachelp.com

Except where otherwise noted, this work is licensed under

http://creativecommons.org/licenses/by/3.0/

This file is one of four files for assisting a user with configuring

an Exchange account in Microsoft Outlook 2016 for Mac:

1. Outlook Exchange Setup 5.0.scpt

2. OutlookExchangeSetupLaunchAgent.sh

3. net.talkingmoose.OutlookExchangeSetupLaunchAgent.plist

4. com.microsoft.Outlook.plist for creating a configuraiton profile

These scripts and files may be freely modified for personal or commercial

purposes but may not be republished for profit without prior consent.

If you find these resources useful or have ideas for improving them,

please let me know. It is only compatible with Outlook 2016 for Mac.

--------------------------------------------

This script assists a user with the setup of his Exchange account

information. Below are basic instructions for using the script.

Consult the Outlook Exchange Setup 5.0 Administrator's Guide

for complete details.

1. Customize the "network and server properties" below with information

appropriate to your network.

2. Deploy this script to a location on your Macs such as

"/Library/CompanyName/OutlookExchangeSetup5.0.scpt".

3. Deploy the recommended "Outlook preferences.mobileconfig"

configuration profile to eliminate Outlook's startup windows.

This assumes you're using the volume license edition

of Office 2016 for Mac.

4. Deploy the OutlookExchangeSetup5.0.plist file to

/Library/LaunchAgents. Update the path to point to the

OutlookExchangeSetup5.0.scpt script.

This script assumes the user's full name is in the form of "Last, First",

but is easily modified if the full name is in the form of "First Last".

It works especially well if the Mac is bound to Active Directory where

the user's short name will match his login name. Optionally, you cans set dscl

to pull the user's EMailAddress from a directory service.

\*)

--------------------------------------------

-- Begin network, server and preferences

--------------------------------------------

--------------- Exchange Server settings ----------------------

**property** useKerberos : *true*

-- Set this to true only if Macs in your environment are bound

-- to Active Directory and your network is properly configured.

**property** ExchangeServer : "exchange.example.com"

-- Address of your organization's Exchange server.

**property** ExchangeServerRequiresSSL : *true*

-- True for most servers.

**property** ExchangeServerSSLPort : 443

-- If ExchangeServerRequiresSSL is true set the port to 443.

-- If ExchangeServerRequiresSSL is false set the port to 80.

-- Use a different port number only if your administrator instructs you.

**property** DirectoryServer : "gc.example.com"

-- Address of an internal Global Catalog server (a type of Windows domain controller).

-- The LDAP server in a Windows network will be a Global Catalog server,

-- which is separate from the Exchange Server.

**property** DirectoryServerRequiresAuthentication : *true*

-- This will almost always be true.

**property** DirectoryServerRequiresSSL : *true*

-- This will almost always be true.

**property** DirectoryServerSSLPort : 3269

-- If DirectoryServerRequiresSSL is true set the port to 3269.

-- If DirectoryServerRequiresSSL is false set the port to 3268.

-- Use a different port number only if your Exchange administrator instructs you.

**property** DirectoryServerMaximumResults : 1000

-- When searching the Global Catalog server, this number determines

-- the maximum number of entries to display.

**property** DirectoryServerSearchBase : ""

-- example: "cn=users,dc=domain,dc=com"

-- Usually, this is empty.

--------------- For Active Directory users ---------------------

**property** getUserInformationFromActiveDirectory : *true*

-- If Macs are bound to Active Directory they can probably use

-- dscl to return the current user's email address, phone number, title, etc.

-- Use Active Directory when possible, otherwise complete the next section.

--------------- For non Active Directory users ---------------

**property** domainName : "example.com"

-- Complete this only if not using Active Directory to get user information.

-- The part of your organization's email address following the @ symbol.

**property** emailFormat : 1

-- Complete this only if not using Active Directory to get user information.

-- When Active Directory is unavailable to determine a user's email address,

-- this script will attempt to parse it from the display name of the user's login.

-- Describe your organization's email format:

-- 1: Email format is first.last@domain.com

-- 2: Email format is first@domain.com

-- 3: Email format is flast@domain.com (first name initial plus last name)

-- 4: Email format is shortName@domain.com

**property** displayName : 2

-- Complete this only if not using Active Directory to get user information.

-- Describe how the user's display name appears at the bottom of the menu

-- when clicking the Apple menu (Log Out Joe Cool... or Log Out Cool, Joe...).

-- 1: Display name appears as "Last, First"

-- 2: Display name appears as "First Last"

**property** domainPrefix : ""

-- Append a NetBIOS domain name to the beginning of the user's short name.

-- Be sure to use two backslashes when adding a name.

-- Example: Use "TALKINGMOOSE\\" to set user name "TALKINGMOOSE\username".

--------------- User Experience -------------------------------

**property** verifyEMailAddress : *false*

-- If set to "true", a dialog asks the user to confirm his email address.

**property** verifyServerAddress : *false*

-- If set to "true", a dialog asks the user to confirm his Exchange server address.

**property** displayDomainPrefix : *false*

-- If set to "true", the username appears as "DOMAIN\username".

-- Otherwise, the username appears as "username".

**property** downloadHeadersOnly : *false*

-- If set to "true", only email headers are downloaded into Outlook.

-- This takes much less time to sync but a user must be online

-- to download and view messages.

**property** hideOnMyComputerFolders : *false*

-- If set to "true", hides local folders.

-- A single Exchange account should do this by default.

**property** unifiedInbox : *false*

-- If set to "true", turns on the Group Similar Folders feature

-- in Outlook menu > Preferences > General.

**property** disableAutodiscover : *false*

-- If set to "true", disables Autodiscover functionality

-- for the Exchange account. Not recommended for mobile devices

-- that may connect to an internal Exchange server addresss and

-- connect to a different external Exchange server address.

**property** errorMessage : "Outlook's setup for your Exchange account failed. Please contact the Help Desk for assistance."

-- Customize this error message for your users in case their account setup fails

--------------------------------------------

-- End network, server and preferences

--------------------------------------------

--------------------------------------------

-- Begin collecting user information

--------------------------------------------

-- attempt to read information from Active Directory for the Me Contact record

**set** userFirstName **to** ""

**set** userLastName **to** ""

**set** userDepartment **to** ""

**set** userOffice **to** ""

**set** userCompany **to** ""

**set** userWorkPhone **to** ""

**set** userMobile **to** ""

**set** userFax **to** ""

**set** userTitle **to** ""

**set** userStreet **to** ""

**set** userCity **to** ""

**set** userState **to** ""

**set** userPostalCode **to** ""

**set** userCountry **to** ""

**set** userWebPage **to** ""

**if** getUserInformationFromActiveDirectory **is** *true* **then**

-- Get information from Active Directoy

-- get the domain's primary NetBIOS domain name

**try**

**set** netbiosDomain **to** **do shell script** "/usr/bin/dscl \"/Active Directory/\" -read / SubNodes | awk 'BEGIN {FS=\": \"} {print $2}'"

**if** displayDomainPrefix **is** *true* **then**

**set** domainPrefix **to** netbiosDomain & "\\"

**else**

**set** domainPrefix **to** ""

**end** **if**

**on** **error**

-- something went wrong

**display dialog** errorMessage & return & return & "Unable to determine NETBIOS domain name. This computer may not be bound to Active Directory." with icon *stop* buttons {"OK"} default button {"OK"} with title "Outlook Exchange Setup"

**error** number -128

**end** **try**

-- read full user information from Active Directory

**try**

**set** AppleScript's text item delimiters **to** {": "}

**set** userInformation **to** **do shell script** "/usr/bin/dscl \"/Active Directory/" & netbiosDomain & "/All Domains/\" -read /Users/$USER AuthenticationAuthority City co company department physicalDeliveryOfficeName sAMAccountName wWWHomePage EMailAddress FAXNumber FirstName JobTitle LastName MobileNumber PhoneNumber PostalCode RealName State Street"

**on** **error**

-- something went wrong

**display dialog** errorMessage & return & return & "Unable to read user information from network directory." with icon *stop* buttons {"OK"} default button {"OK"} with title "Outlook Exchange Setup"

**error** number -128

**end** **try**

**repeat** **with** i **from** 1 **to** **count** **of** *paragraphs* **in** userInformation

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "EMailAddress:" **then**

**try**

**set** emailAddress **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** emailAddress **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "dsAttrTypeNative:co:" **then**

**try**

**set** userCountry **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userCountry **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "dsAttrTypeNative:company:" **then**

**try**

**set** userCompany **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userCompany **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "dsAttrTypeNative:department:" **then**

**try**

**set** userDepartment **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userDepartment **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "dsAttrTypeNative:physicalDeliveryOfficeName:" **then**

**try**

**set** userOffice **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userOffice **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "dsAttrTypeNative:sAMAccountName:" **then**

**try**

**set** userShortName **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userShortName **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "dsAttrTypeNative:wWWHomePage:" **then**

**try**

**set** userWebPage **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userWebPage **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "City:" **then**

**try**

**set** userCity **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userCity **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "FAXNumber:" **then**

**try**

**set** userFax **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userFax **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "FirstName:" **then**

**try**

**set** userFirstName **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userFirstName **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "JobTitle:" **then**

**try**

**set** userTitle **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userTitle **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "LastName:" **then**

**try**

**set** userLastName **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userLastName **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "MobileNumber:" **then**

**try**

**set** userMobile **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userMobile **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "PhoneNumber:" **then**

**try**

**set** userWorkPhone **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userWorkPhone **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "PostalCode:" **then**

**try**

**set** userPostalCode **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userPostalCode **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "RealName:" **then**

**try**

**set** userFullName **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userFullName **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "State:" **then**

**try**

**set** userState **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userState **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**set** AppleScript's text item delimiters **to** {": "}

**if** *paragraph* i **of** userInformation **begins with** "Street:" **then**

**try**

**set** userStreet **to** *text item* 2 **of** *paragraph* i **of** userInformation

**on** **error**

**set** AppleScript's text item delimiters **to** {""}

**set** userStreet **to** *characters* 2 **through** **end** **of** *paragraph* (i + 1) **of** userInformation **as** *string*

**end** **try**

**end** **if**

**end** **repeat**

**set** AppleScript's text item delimiters **to** {";Kerberosv5;;", ";"}

**try**

**set** userKerberosRealm **to** *text item* 2 **of** userInformation

**end** **try**

**set** AppleScript's text item delimiters **to** {""}

**if** emailAddress **is** "" **then**

-- something went wrong

**display dialog** errorMessage & return & return & "Unable to read email address from network directory." with icon *stop* buttons {"OK"} default button {"OK"} with title "Outlook Exchange Setup"

**error** number -128

**end** **if**

**else** **if** emailFormat **is** 1 **and** displayName **is** 1 **then**

-- Pull user information from the account settings of the local user account

**set** userShortName **to** short user name **of** (**system info**)

**set** userFullName **to** long user name **of** (**system info**)

-- first.last@domain.com and full name displays as "Last, First"

**set** AppleScript's text item delimiters **to** ", "

**set** userFirstName **to** **last** *text item* **of** userFullName

**set** userLastName **to** *word* 1 **of** *text item* 1 **of** userFullName

**set** AppleScript's text item delimiters **to** ""

**set** emailAddress **to** userFirstName & "." & userLastName & "@" & domainName

**else** **if** emailFormat **is** 1 **and** displayName **is** 2 **then**

-- Pull user information from the account settings of the local user account

**set** userShortName **to** short user name **of** (**system info**)

**set** userFullName **to** long user name **of** (**system info**)

-- first.last@domain.com and full name displays as "First Last"

**set** AppleScript's text item delimiters **to** " "

**set** userFirstName **to** *word* 1 **of** *text item* 1 **of** userFullName

**set** userLastName **to** **last** *text item* **of** userFullName

**set** AppleScript's text item delimiters **to** ""

**set** emailAddress **to** userFirstName & "." & userLastName & "@" & domainName

**else** **if** emailFormat **is** 2 **and** displayName **is** 1 **then**

-- Pull user information from the account settings of the local user account

**set** userShortName **to** short user name **of** (**system info**)

**set** userFullName **to** long user name **of** (**system info**)

-- first@domain.com and full name displays as "Last, First"

**set** AppleScript's text item delimiters **to** ", "

**set** userFirstName **to** **last** *text item* **of** userFullName

**set** userLastName **to** *word* 1 **of** *text item* 1 **of** userFullName

**set** AppleScript's text item delimiters **to** ""

**set** emailAddress **to** userFirstName & "@" & domainName

**else** **if** emailFormat **is** 2 **and** displayName **is** 2 **then**

-- Pull user information from the account settings of the local user account

**set** userShortName **to** short user name **of** (**system info**)

**set** userFullName **to** long user name **of** (**system info**)

-- first@domain.com if full name displays as "First Last"

**set** AppleScript's text item delimiters **to** " "

**set** userFirstName **to** *word* 1 **of** *text item* 1 **of** userFullName

**set** userLastName **to** **last** *text item* **of** userFullName

**set** AppleScript's text item delimiters **to** ""

**set** emailAddress **to** userFirstName & "@" & domainName

**else** **if** emailFormat **is** 3 **and** displayName **is** 1 **then**

-- Pull user information from the account settings of the local user account

**set** userShortName **to** short user name **of** (**system info**)

**set** userFullName **to** long user name **of** (**system info**)

-- flast@domain.com and full name displays as "Last, First"

**set** AppleScript's text item delimiters **to** ", "

**set** userFirstName **to** **last** *text item* **of** userFullName

**set** userLastName **to** *word* 1 **of** *text item* 1 **of** userFullName

**set** AppleScript's text item delimiters **to** ""

**set** emailAddress **to** (*character* 1 **of** userFirstName) & userLastName & "@" & domainName

**else** **if** emailFormat **is** 3 **and** displayName **is** 2 **then**

-- Pull user information from the account settings of the local user account

**set** userShortName **to** short user name **of** (**system info**)

**set** userFullName **to** long user name **of** (**system info**)

-- flast@domain.com and full name displays as "First Last"

**set** AppleScript's text item delimiters **to** " "

**set** userFirstName **to** *word* 1 **of** *text item* 1 **of** userFullName

**set** userLastName **to** **last** *text item* **of** userFullName

**set** AppleScript's text item delimiters **to** ""

**set** emailAddress **to** (*character* 1 **of** userFirstName & userLastName & "@" & domainName)

**else** **if** emailFormat **is** 4 **and** displayName **is** 1 **then**

-- Pull user information from the account settings of the local user account

**set** userShortName **to** short user name **of** (**system info**)

**set** userFullName **to** long user name **of** (**system info**)

-- shortName@domain.com and full name displays as "Last, First"

**set** AppleScript's text item delimiters **to** ", "

**set** userFirstName **to** **last** *text item* **of** userFullName

**set** userLastName **to** *word* 1 **of** *text item* 1 **of** userFullName

**set** AppleScript's text item delimiters **to** ""

**set** emailAddress **to** userShortName & "@" & domainName

**else** **if** emailFormat **is** 4 **and** displayName **is** 2 **then**

-- Pull user information from the account settings of the local user account

**set** userShortName **to** short user name **of** (**system info**)

**set** userFullName **to** long user name **of** (**system info**)

-- shortName@domain.com and full name displays as "First Last"

**set** AppleScript's text item delimiters **to** " "

**set** userFirstName **to** *word* 1 **of** *text item* 1 **of** userFullName

**set** userLastName **to** **last** *text item* **of** userFullName

**set** AppleScript's text item delimiters **to** ""

**set** emailAddress **to** userShortName & "@" & domainName

**else**

-- something went wrong

**display dialog** errorMessage & return & return & "Unable to parse account information from local OS X account." with icon *stop* buttons {"OK"} default button {"OK"} with title "Outlook Exchange Setup"

**error** number -128

**end** **if**

--------------------------------------------

-- End collecting user information

--------------------------------------------

--------------------------------------------

-- Begin account setup

--------------------------------------------

**tell** *application* "Microsoft Outlook"

**activate**

**try**

**set** group similar folders **to** unifiedInbox

**end** **try**

**try**

**set** hide on my computer folders **to** hideOnMyComputerFolders

**end** **try**

**if** verifyEMailAddress **is** *true* **then**

**set** verifyEmail **to** **display dialog** "Please verify your email address is correct." default answer emailAddress with icon 1 with title "Outlook Exchange Setup" buttons {"Cancel", "Verify"} default button {"Verify"}

**set** emailAddress **to** text returned **of** verifyEmail

**end** **if**

**if** verifyServerAddress **is** *true* **then**

**set** verifyServer **to** **display dialog** "Please verify your Exchange Server name is correct." default answer ExchangeServer with icon 1 with title "Outlook Exchange Setup" buttons {"Cancel", "Verify"} default button {"Verify"}

**set** ExchangeServer **to** text returned **of** verifyServer

**end** **if**

-- create the Exchange account

**try**

**set** newExchangeAccount **to** **make** new *exchange account* with properties ¬

{name:"Mailbox - " & userFullName, user name:domainPrefix & userShortName, full name:userFullName, *email address*:emailAddress, server:ExchangeServer, use ssl:ExchangeServerRequiresSSL, port:ExchangeServerSSLPort, ldap server:DirectoryServer, ldap needs authentication:DirectoryServerRequiresAuthentication, ldap use ssl:DirectoryServerRequiresSSL, ldap max entries:DirectoryServerMaximumResults, ldap search base:DirectoryServerSearchBase, receive partial messages:downloadHeadersOnly, background autodiscover:disableAutodiscover}

**on** **error**

-- something went wrong

**display dialog** errorMessage & return & return & "Unable to create Exchange account." with icon *stop* buttons {"OK"} default button {"OK"} with title "Outlook Exchange Setup"

**error** number -128

**end** **try**

-- The following lines enable Kerberos support if the userKerberos property above is set to true.

**if** useKerberos **is** *true* **then**

**try**

**set** use kerberos authentication **of** newExchangeAccount **to** useKerberos

**set** principal **of** newExchangeAccount **to** userKerberosRealm

**on** **error**

-- something went wrong

**display dialog** errorMessage & return & return & "Unable to set Exchange account to use Kerberos." with icon *stop* buttons {"OK"} default button {"OK"} with title "Outlook Exchange Setup"

**error** number -128

**end** **try**

**end** **if**

-- The Me Contact record is automatically created with the first account.

-- Set the first name, last name, email address and other information using Active Directory.

**set** first name **of** me contact **to** userFirstName

**set** last name **of** me contact **to** userLastName

**set** email addresses **of** me contact **to** {address:emailAddress, *type*:*work*}

**set** department **of** me contact **to** userDepartment

**set** office **of** me contact **to** userOffice

**set** company **of** me contact **to** userCompany

**set** business phone number **of** me contact **to** userWorkPhone

**set** mobile number **of** me contact **to** userMobile

**set** business fax number **of** me contact **to** userFax

**set** job title **of** me contact **to** userTitle

**set** business street address **of** me contact **to** userStreet

**set** business city **of** me contact **to** userCity

**set** business state **of** me contact **to** userState

**set** business zip **of** me contact **to** userPostalCode

**set** business country **of** me contact **to** userCountry

**set** business web page **of** me contact **to** userWebPage

-- We're done.

**end** **tell**

**do shell script** "/bin/rm $HOME/Library/LaunchAgents/net.talkingmoose.OutlookExchangeSetup5.0.plist"

**do shell script** "/bin/launchctl remove net.talkingmoose.OutlookExchangeSetup5.0"

--------------------------------------------

-- End account setup

--------------------------------------------

1. Appendix B

The Package for Deployment..app script:

(\*

--------------------------------------------

Outlook Exchange Setup 5.0

© Copyright 2008-2015 William Smith

bill@officeformachelp.com

Except where otherwise noted, this work is licensed under

http://creativecommons.org/licenses/by/3.0/

This file accompanies four files for assisting a user with configuring

an Exchange account in Microsoft Outlook 2016 for Mac:

1. Outlook Exchange Setup 5.0.scpt

2. OutlookExchangeSetupLaunchAgent.sh

3. net.talkingmoose.OutlookExchangeSetupLaunchAgent.plist

4. com.microsoft.Outlook.plist for creating a configuraiton profile

These scripts and files may be freely modified for personal or commercial

purposes but may not be republished for profit without prior consent.

If you find these resources useful or have ideas for improving them,

please let me know. It is only compatible with Outlook 2016 for Mac.

--------------------------------------------

This script assists administrators with creating an Apple Installer

package for deploying the necessary files for Outlook Exchange Setup 5.0.

\*)

-- prompt for package details

**set** packageName **to** text returned **of** (**display dialog** "Name your installer..." default answer "Install Outlook Exchange Setup 5.0" with icon *file* ((**path to** *application* "Installer" as *string*) & "Contents:Resources:package.icns") buttons {"Cancel", "Next"} default button {"Next"} with title "Package for Deployment")

**set** packageIdentifier **to** text returned **of** (**display dialog** "Enter your installer package's identifier..." default answer "net.talkingmoose" & "." & (**every** *word* **of** packageName) with icon *file* ((**path to** *application* "Installer" as *string*) & "Contents:Resources:package.icns") buttons {"Cancel", "Next"} default button {"Next"} with title "Package for Deployment")

**set** packageVersion **to** **every** *word* **of** text returned **of** (**display dialog** "Enter a version number for your installer..." default answer "1.0.0" with icon *file* ((**path to** *application* "Installer" as *string*) & "Contents:Resources:package.icns") buttons {"Cancel", "Build"} default button {"Build"} with title "Package for Deployment")

-- locate the ROOT folder

**set** myLocation **to** **path to** **me**

**tell** *application* "Finder"

**set** rootFolder **to** (*container* **of** myLocation **as** *string*) & "ROOT:"

**if** (**exists** **of** *folder* rootFolder) **is** *false* **then**

**set** rootFolder **to** **choose folder** with prompt "Select the ROOT folder containing files for deployment."

**set** rootFolder **to** rootFolder **as** *string*

**end** **if**

**end** **tell**

**set** rootPath **to** POSIX path **of** rootFolder

**tell** *application* "Finder"

**set** myFolderPath **to** POSIX path **of** (*container* **of** myLocation **as** *string*)

**end** **tell**

-- delete any hidden files

**set** removeDSStores **to** "find '" & rootPath & "' -name .DS\_Store -delete" **as** *string*

**do shell script** removeDSStores

-- build the package

**set** buildPackage **to** "pkgbuild --identifier '" & packageIdentifier & "' --version '" & packageVersion & "' --scripts '" & myFolderPath & "Scripts' --root '" & rootPath & "' '" & myFolderPath & packageName & ".pkg'" **as** *string*

**try**

**set** theInstaller **to** **do shell script** buildPackage

**display dialog** "Package build succeeded." with icon *file* ((**path to** *application* "Installer" as *string*) & "Contents:Resources:package.icns") with title "Package for Deployment" buttons {"Show Package"} default button {"Show Package"}

**set** thePackage **to** *POSIX file* (myFolderPath & packageName & ".pkg")

**tell** *application* "Finder"

**activate**

**select** thePackage

**end** **tell**

**on** **error**

**display dialog** "Package build failed. Verify ROOT and Scripts folders are in the same folder as this Package for Deployment app." with icon *stop* with title "Package for Deployment" buttons {"OK"} default button {"OK"}

**end** **try**