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. How it works

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

When the user opens Outlook for the first time, the application creates a new set of default files. When the launchd agent notices the new files, it triggers the AppleScript, which does the account configuration. The configuration profile (.mobileconfig file) hides any first run windows and dialogs that would hinder the script.

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.

Server, Preferences and Custom Message Properties

Locate the OutlookExchangeSetup5.0.scpt file and double-click it 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

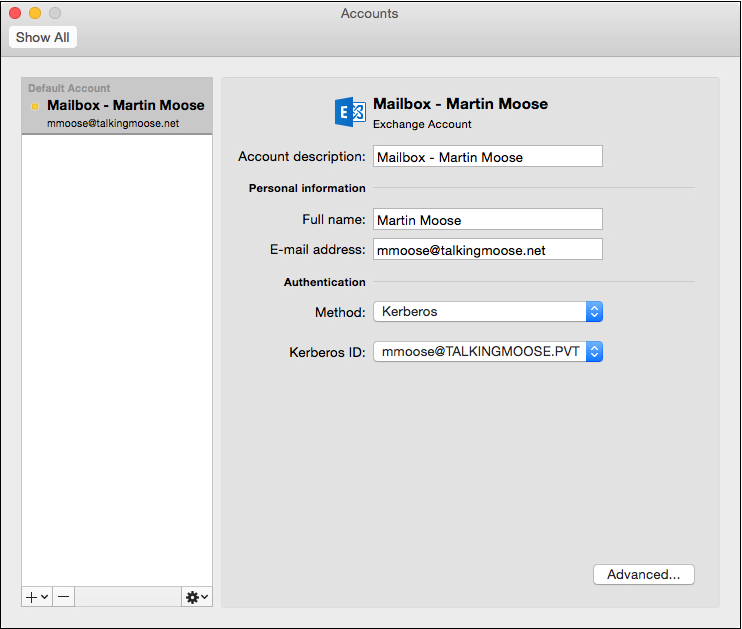
--------------------------------------------

-- End network, server and preferences

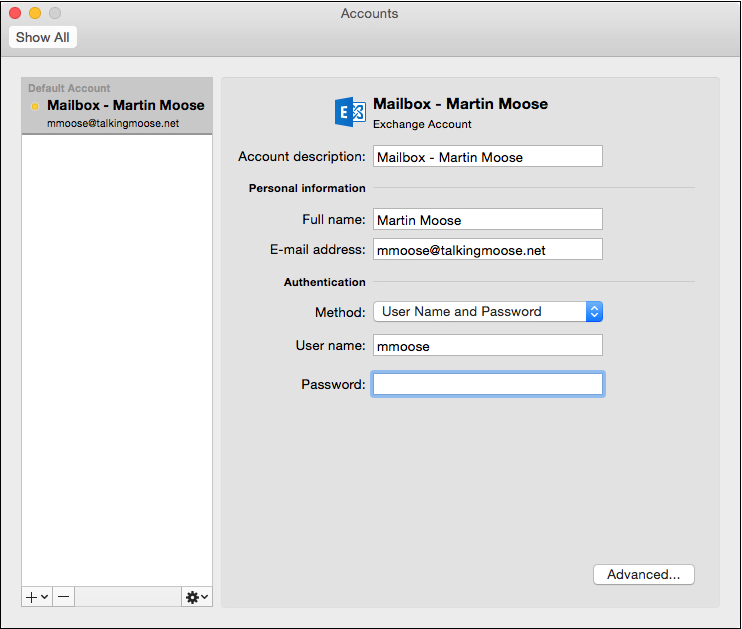
--------------------------------------------

You can customize 21 properties in this section. Each property is a setting or preference the script uses when run. The property names reflect their purposes and include comments below each one.

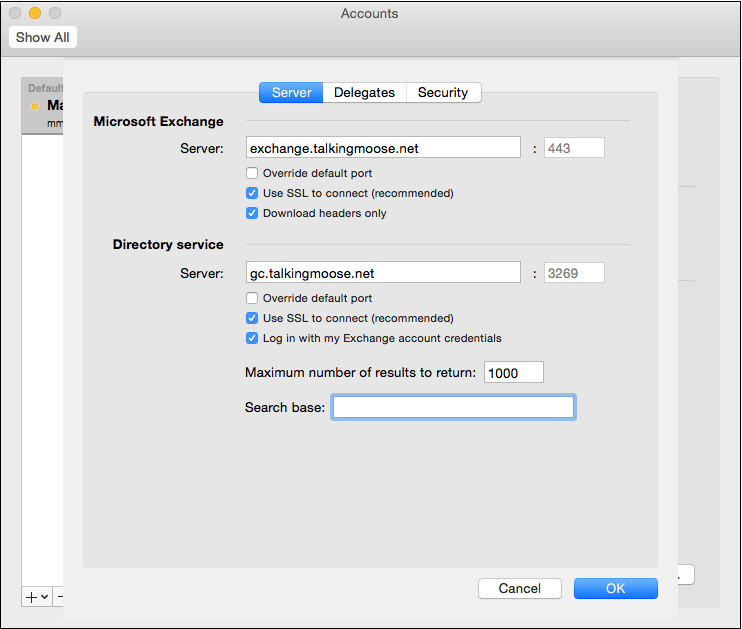
Most every property corresponds to an Exchange account setting or preference. You’ll find the Accounts pane under Outlook > Preferences (See Figures 1, 2 and 3).

 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** 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" |

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. |

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** 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 whether setup is complete or an error occurs.

Saving the Script

Save the script as File Format: Script using *Outlook Exchange Setup 5.0* or another more appropriate name. Users will not see this script. (See Figure 5)

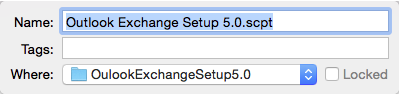


Figure 5. Save the script as Outlook Exchange Setup 5.0.scpt.

1. Test the setup

Testing should include the script and configuration profile(s) to disable Outlook’s first run windows.

Copy the prepared script to a location on your test Mac such as /Users/Shared for easy access.

Install the configuration profile(s)

Locate the Configuration Profiles and Plists folder inside the Oulook 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 the Main Window and lets the script 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)*, JAMF Software’s Casper Suite or Meraki, 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 for Excel, PowerPoint and Word to disable their first run windows.

Deploy the configuration profile to a test Mac over the air using Apple’s Push Notification service or download and install it manually. It should appear in the Profiles pane of System Preferences under Device Profiles *(See Figure 8)*.

Feel free to use the included configuration profiles for disabling first run windows included with the script. 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**.

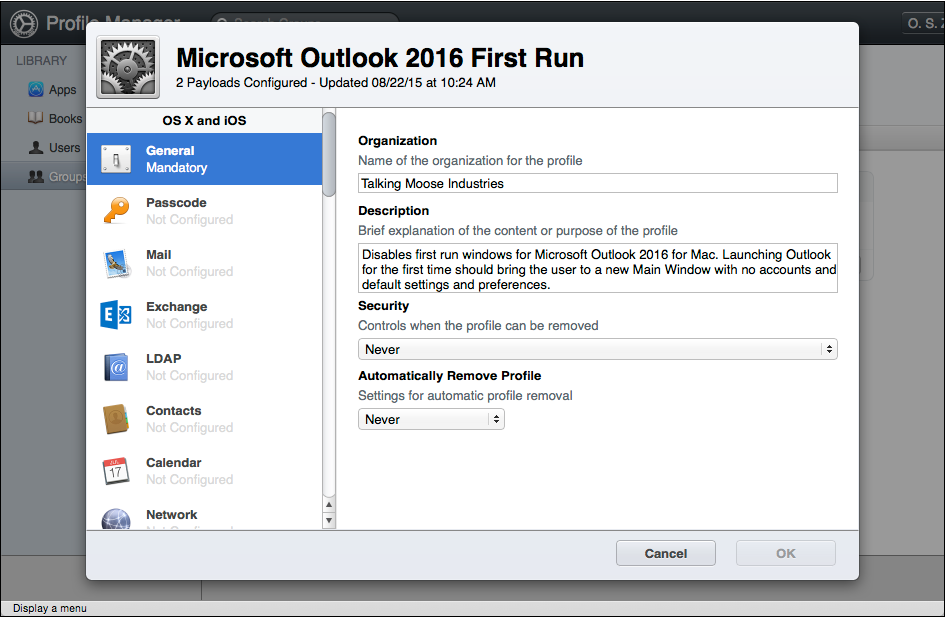


Figure 6. Profile Manager payload – General.

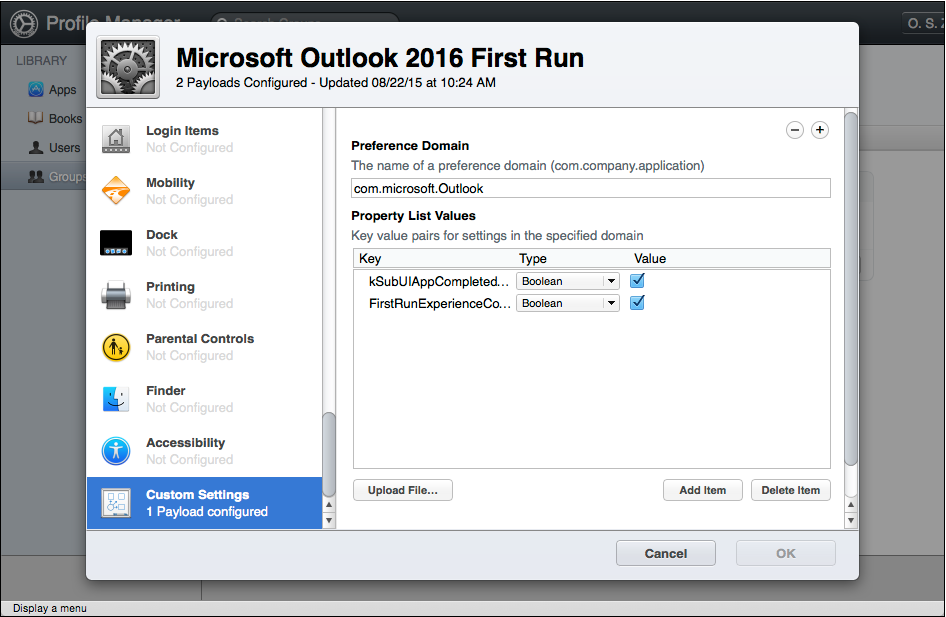


Figure 7. Profile Manager payload – Custom Settings.

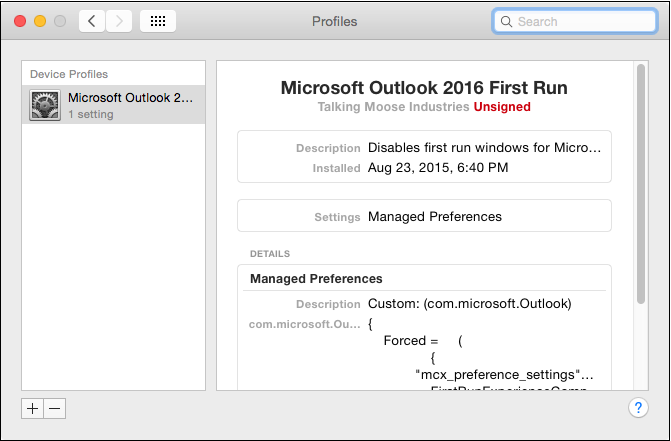


Figure 8. Installed configuration profile to disable Outlook’s first run windows.

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 script 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 script’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 script’s properties to use Active Directory for user information.

Test Outlook and the script

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

Open the script in Script Editor and click the Run button at the top of the Script Editor window *(See Figure 9)*.

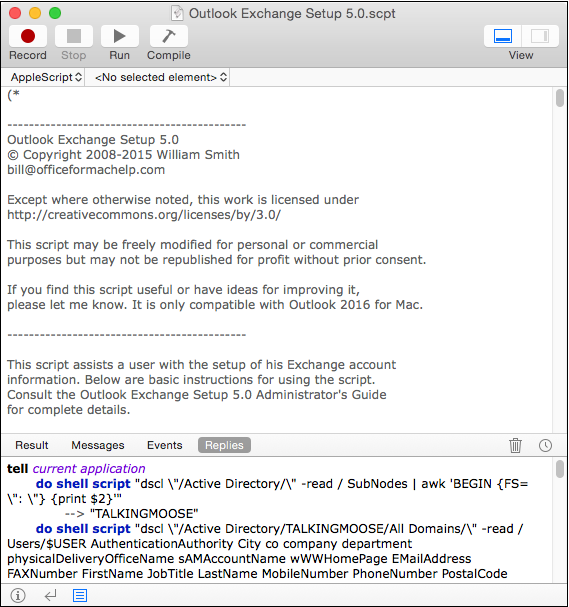


Figure 9. Run the script in Script Editor.

The script should complete with a notification to the user *(See Figure 10)*.

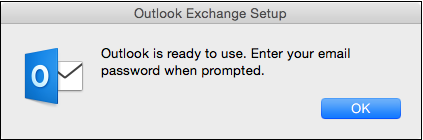


Figure 10. Outlook is ready to use.

Choose Outlook > Preferences > General. Verify whether “Group similar folders, such as Inboxes, from different accounts” matches your setting in the script for the unifiedInbox property *(See Figure 11)*.

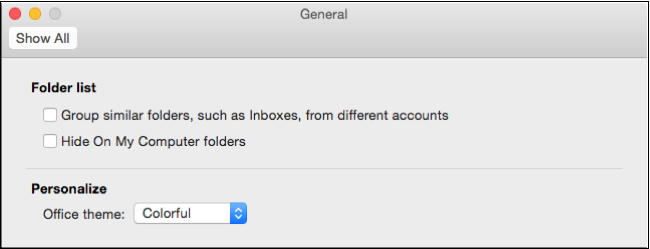


Figure 11. Verify the unifiedInbox property correctly set  
the “Group similar folders, such as Inboxes, from different accounts” setting.

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 12)*.

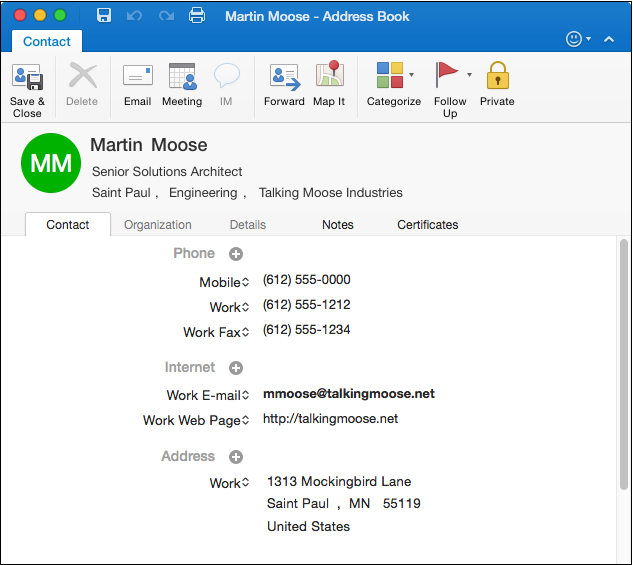


Figure 12. Me Contact record populated from Active Directory.

Continue testing with other accounts until you’re satisfied the script is working correctly for your environment. The proceed to packaging and deploying the script, configuration profile and launchd agent.

1. Packaging for deployment

The final step is to collect the Outlook template files to a location where you can package them for distribution.

Packaging the template files is beyond the scope of these scripts. You can use any of several methods for distributing the collected files. Below are a few options:

* Apple Remote Desktop
* Apple’s PackageMaker.app application
* JAMF Software’s Casper Suite
* AFP548.com’s InstaDMG scripts

Collect the Outlook Template files

You will use the application again to collect the files but this time you will specify a folder of your choice.

First, make sure you are logged in to your first new Mac OS X account and that Outlook is quit. Then double-click the application and click the Another folder button. (See Figure 8)



Figure 8

Select a folder or create a new folder and click the Choose button. The same files that were copied to the User Template folder will be copied to your new folder.

Package these files and distribute them to the User Template folders on your Mac workstations. Be sure to include the script with your package. Remember, you are responsible for installing Office 2011 separately.

1. Appendix A

The Outlook for Mac Exchange Setup 4.0 script:

(\*

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

Outlook Exchange Setup 4.0.1

© Copyright 2008-2010 William Smith

bill@officeformachelp.com

Except where otherwise noted, this work is licensed under

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

This script may be freely modified for personal or commercial

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

If you find this script useful or have ideas for improving it,

please let me know. It is only compatible with Outlook for Mac 2011.

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

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 4.0 Administrator's Guide

for complete details.

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

appropriate to your network.

2. a. Launch Microsoft Outlook under a Mac OS X account that has no

Outlook identity. Identities are found in

~/Documents/Microsoft User Data/Office 2011 Identities.

b. Close the Outlook Setup Assistant window and select

Outlook --> Work Offline.

c. In the Finder, create a folder called "Outlook Setup Script"

in /Library/<your company name>/ or any single location

where all users have Read permissions.

d. Copy your customized script to the "Outlook Setup Script" folder.

It should be saved as a compiled script, not an application.

You can rename the script if you wish.

e. In Outlook select Tools --> Schedules... and create a New

Schedule.

Name: Outlook Setup

When: At Startup

Action: Run AppleScript, choose the saved script

in the "Outlook Setup Script" folder.

Click the "OK" button. Your script should be set to run

at "Next Startup".

f. Configure any preferences, schedules, signatures, etc.

This can include Default address format, Default time zone,

fonts, encoding for attachments and more.

g. Quit Outlook.

h. Depending on your admin tools, you can push the script to the

/Library folder of a new computer and the

"~/Documents/Microsoft User Data/Office 2011 Identities/Main

Identity" folder and to the "/System/Library/User

Template/English.lproj/Documents/Microsoft User Data/

Office 2011 Identities" folder.

New users on a machine will have the pre-configured Main Identity

folder copied to their home folders the first time they log in to

a computer.

The first time they launch Outlook, the "Exchange Setup" schedule

will run the script.

The script will disable the Schedule once it has run and will set

Outlook to work online.

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, a dscl command can be used to pull the user's EMailAddress from a directory service.

\*)

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

-- Begin network and server properties

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

**property** useKerberos : false

-- Set this property to true only if Macs in your environment

-- are bound to Active Directory.

-- Kerberos support works only if your network is properly

-- configured.

**property** kerberosRealm : "example.com"

-- For most Kerberos environments the Kerberos realm

-- will be the same as the Email domain name.

-- This property is used only if the userKerberos property above

-- is true.

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

-- example: "domain.com"

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

-- example: "ExchangeServer.domain.com"

**property** ExtendedExchangeServerAddress : false

-- Are you connecting to an Exchange Server 2007 server?

-- If so, you may need to use the extended address:

-- "ExchangeServer.domain.com/exchange/user@example.com".

-- Set this to "true" if you need the extended address.

**property** ExchangeServerRequiresSSL : true

-- If Exchange users will connect to a backend server

-- within the same network then SSL is probably not required.

-- OWA connections will probably require SSL and

-- this setting should be set to true.

**property** ExchangeServerSSLPort : 443

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

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

-- Use a different port number only if specified

-- by your administrator.

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

-- example: "GlobalCatalogServer.domain.com"

-- OWA connections cannot use the OWA address.

**property** DirectoryServerRequiresAuthentication : true

-- This will almost always be true. The LDAP server

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

-- which is separate from the Exchange Server.

-- OWA connections can not use the OWA address

-- with this setting.

**property** DirectoryServerRequiresSSL : false

-- If Exchange users will connect to a backend server

-- within the same network then SSL is probably

-- not required. OWA connections can not use

-- the OWA address with this setting.

**property** DirectoryServerSSLPort : 3268

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

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

-- Use a different port number only if specified

-- by your administrator.

-- OWA connections cannot use the OWA address with this setting.

**property** DirectoryServerMaximumResults : 1000

-- When searching the Global Address list, this number determines

-- the maximum number of entries to display.

**property** DirectoryServerSearchBase : ""

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

-- Search base will be optional in many environments and its

-- format will vary greatly. Experiment first connecting without

-- entering the search base information.

**property** getEMailAddressUsingDSCL : true

-- If the Macs are connected to a directory service such as

-- Active Directory, then they can probably use dscl to return

-- the current user's Email address instead of trying to parse it

-- from the display name.

-- Using dscl is preferred. Otherwise, set this to false

-- and set the next property to the appropriate number.

**property emailFormat : 1**

**-- When dscl is unavailable to determine a user's Email address,**

**-- it can be parsed using the display name of the user's login.**

**-- Set the next two properties to the examples**

**-- that correspond to your organization's setup.**

**-- 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 : 1**

**-- 1: Display name displays as "Last, First"**

**-- 2: Display name displays as "First Last"**

**property** verifyEMailAddress : false

-- If set to "true", a dialog will ask the user to confirm

-- his Email address.

**property** verifyServerAddress : false

-- If set to "true", a dialog will ask the user to confirm

-- his Exchange server address.

**property** scheduled : false

-- Exchange accounts don't require that

-- the "Send & Receive All" schedule be enabled.

-- Change this setting to true if the user

-- will also be connecting to POP or IMAP accounts.

**property** errorMessage : "Setup of your account failed. Please contact your workstation administrator for assistance."

-- Customize this error message for your users

-- if their account setup fails

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

-- End server properties

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

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

-- Begin get user name

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

-- User information is pulled from the account settings of the current

user account

**tell** *application* "System Events"

**set** shortName **to** name **of** current user

**set** fullName **to** full name **of** current user

**end** **tell**

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

-- End get user name

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

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

-- Begin email address properties

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

**if** getEMailAddressUsingDSCL **is** true **then**

**set** emailAddress **to** **do shell script** "dscl . -read /Users/" &

shortName & " EMailAddress | awk 'BEGIN {FS=\": \"} {print $2}'"

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

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

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

**set firstName to *text item* 2 of fullName**

**set lastName to *text item* 1 of fullName**

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

**set emailAddress to firstName & "." & lastName & "@" & domainName**

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

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

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

**set firstName to *text item* 1 of fullName**

**set lastName to *text item* 2 of fullName**

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

**set emailAddress to firstName & "." & lastName & "@" & domainName**

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

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

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

**set firstName to *text item* 2 of fullName**

**set lastName to *text item* 1 of fullName**

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

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

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

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

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

**set firstName to *text item* 1 of fullName**

**set lastName to *text item* 2 of fullName**

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

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

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

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

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

**set firstName to *text item* 2 of fullName**

**set lastName to *text item* 1 of fullName**

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

**set emailAddress to (*character* 1 of firstName) & lastName & "@" & domainName**

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

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

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

**set firstName to *text item* 1 of fullName**

**set lastName to *text item* 2 of fullName**

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

**set emailAddress to (*character* 1 of firstName & lastName & "@" & domainName)**

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

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

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

**set firstName to *text item* 2 of fullName**

**set lastName to *text item* 1 of fullName**

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

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

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

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

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

**set firstName to *text item* 1 of fullName**

**set lastName to *text item* 2 of fullName**

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

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

**end if**

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

-- End email address properties

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

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

-- Exchange Server 2007 extended address

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

**if** ExtendedExchangeServerAddress **is** true **then**

**set** ExchangeServer **to** ExchangeServer & "/exchange/" & emailAddress

**end** **if**

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

-- End get user name

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

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

-- Begin account setup

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

**try**

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

**set** verifyEmail **to** **display dialog** "Please verify that your email

address is correct." default answer emailAddress with icon 2

buttons {"Cancel", "OK"} default button {"OK"}

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

**end** **if**

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

**set** verifyServer **to** **display dialog** "Please verify that your

Exchange Server name is correct." default answer ExchangeServer

with icon 2 buttons {"Cancel", "OK"} default button {"OK"}

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

**end** **if**

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

**activate**

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

properties ¬

{name:"Mailbox - " & fullName, user name:shortName, full

name:fullName, *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}

-- The following lines enable Kerberos support if the

userKerberos property above is set to true.

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

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

useKerberos

**set** principal **of** newExchangeAccount **to** shortName & "@" &

kerberosRealm

**end** **if**

-- The Me Contact record is now automatically created with the

first account.

-- Set the first and last name of the Me Contact record.

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

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

**set email addresses of me contact to**

**{address:emailAddress, *type*:work}**

-- We're done. Set the schedules back to normal and set Outlook

to work online

**set** enabled **of** *schedule* "Send & Receive All" **to** scheduled

**delete** *schedule* "Exchange Setup"

**set** working offline **to** false

**end** **tell**

**on** **error**

**display dialog** errorMessage with icon 2 buttons {"OK"} default

button {"OK"}

**end** **try**

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

-- End account setup

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

1. Appendix B

The Copy Outlook Files.app script:

(\*

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

Outlook Exchange Setup 4.0

© Copyright 2008-2010 William Smith

bill@officeformachelp.com

Except where otherwise noted, this work is licensed under http://creativecommons.org/licenses/by/3.0/

This script may be freely modified for personal or commercial purposes but may not be republished for profit without prior consent.

If you find this script useful or have ideas for improving it, please let me know. It is only compatible with Outlook 2011 for Mac.

This script is used in conjunction with the OutlookExchangeSetup4.0.scpt script or later.

1. Configure the OutlookExchangeSetup4.0.scpt file with your Exchange Server settings

2. Copy the OutlookExchangeSetup4.0.scpt file to a commonly accessible folder on your computer such as /Library/yourCompanyName. Make sure all users have read permission for the script file..

3. Create a new user account.

4. Launch Microsoft Outlook for Mac and set up and configure all preferences, schedules, rules, etc., that all users should have by default.

5. Create a new schedule using Tools --> Schedules...

Name it exactly "Exchange Setup".

Set it to run "At Startup".

Add Action "Run AppleScript" and select your pre-configured OutlookExchangeSetup4.0.scpt file.

Click the OK button and to save the "Exchange Setup" schedule. It should be enabled.

6. Run this script. It does three things:

a.) It will copy the current Microsoft User Data folder to the User

Template folder.

b.) It will copy all Microsoft preferences to the User Template

folder.

c.) It will set the default mail application to Outlook in the User

Template folder.

7. To test, create a new user account on the same computer and log in. Launch Microsoft Outlook for Mac. Your new user account should run the script and then delete the "Exchange Setup" schedule.

\*)

-- Where do you want to copy your Outlook files?

**set** folderChoice **to** **display dialog** "This script will copy the default set of Outlook files to your computer's User Template folder or to a folder of your choice.

Where would you like to copy the files?" with title "Copy files" with icon 1 buttons {"Cancel", "Another folder", "User Template folder"} default button {"User Template folder"}

**if** button returned **of** folderChoice **is** "Another folder" **then**

copyToAnotherFolder()

**end** **if**

**if** button returned **of** folderChoice **is** "User Template folder" **then**

copyToUserTemplateFolder()

**end** **if**

**on** copyToUserTemplateFolder()

-- Get the path to the current home folder

-- in the form of /Users/homefolder

**set** homeFolder **to** POSIX path **of** (**path to** home folder)

-- copy the Microsoft User Data folder from the current user's home

folder

-- to /System/Library/User Template/English.lproj/Documents

**set** MUD **to** "\"" & homeFolder & "Documents/Microsoft User Data" &

"\""

**set** UTDocuments **to** "\"" & "/System/Library/User

Template/English.lproj/Documents" & "\""

**set** copyMUD **to** "cp -fR " & MUD & " " & UTDocuments **as** *string*

**do shell script** copyMUD **with** administrator privileges

-- copy the Microsoft preferences from the current user's home

folder

-- to /System/Library/User

Template/English.lproj/Library/Preferences

**set** MSPreferences **to** homeFolder &

"Library/Preferences/\*[Mm]icrosoft\*"

**set** UTPreferences **to** "\"" & "/System/Library/User

Template/English.lproj/Library/Preferences" & "\""

**set** copyMSPreferences **to** "cp -fR " & MSPreferences & " " &

UTPreferences **as** *string*

**do shell script** copyMSPreferences **with** administrator privileges

**do shell script** "defaults write

/System/Library/User\\Template/English.lproj/Library/Preferences

com.apple.LaunchServices LSHandlers '({LSHandlerRoleAll =

\"com.microsoft.outlook\"; LSHandlerURLScheme = mailto;})'" **with**

administrator privileges

**end** copyToUserTemplateFolder

**on** copyToAnotherFolder()

**set** theFolder **to** **choose folder** with prompt "Select a folder for your

default Outlook for Mac files..."

**set** theFolder **to** POSIX path **of** theFolder

-- Get the path to the current home folder

-- in the form of /Users/homefolder

**set** homeFolder **to** POSIX path **of** (**path to** home folder)

-- copy the Microsoft User Data folder from the current user's home

folder

-- to /System/Library/User Template/English.lproj/Documents

**set** MUD **to** "\"" & homeFolder & "Documents/Microsoft User Data" &

"\""

**set** UTDocuments **to** "\"" & theFolder & "/Documents/Microsoft User

Data" & "\""

**set** copyMUD **to** "cp -fR " & MUD & " " & UTDocuments **as** *string*

**do shell script** "mkdir -p " & "\"" & theFolder & "Documents" & "\""

**do shell script** copyMUD

-- copy the Microsoft preferences from the current user's home

folder

-- to /System/Library/User

Template/English.lproj/Library/Preferences

**set** MSPreferences **to** homeFolder &

"Library/Preferences/\*[Mm]icrosoft\*"

**set** UTPreferences **to** "\"" & theFolder & "Library/Preferences" & "\""

**set** copyMSPreferences **to** "cp -fR " & MSPreferences & " " &

UTPreferences **as** *string*

**do shell script** "mkdir -p '" & theFolder & "Library/Preferences'"

**do shell script** copyMSPreferences

**set** emailDefaults **to** "defaults write '" & theFolder &

"Library/Preferences/com.apple.LaunchServices' LSHandlers

'({LSHandlerRoleAll = \"com.microsoft.outlook\";

LSHandlerURLScheme = mailto;})'" **as** *string*

**do shell script** emailDefaults

**end** copyToAnotherFolder