Device account PowerShell scripts for Surface Hub

# Introduction

To help with setup, the Surface Hub team is providing some PowerShell scripts that will make device account setup smoother. To make sure the scripts are useful and make setup easier, we'd like folks to test the scripts and provide some feedback.

**What do the scripts do?**

* **Create device accounts** for setups using pure single-forest on-prem (Exchange and Skype 2013+ only), or pure online (Office 365) that are configured in the recommended manner.
* **Validate existing device accounts** for any setup (on-prem, online, or hybrid using Exchange/Lync 2010+) to see if they're ready and compatible with a Surface Hub.
* Provide a base template for anyone wanting to create their own device account creation/validation scripts.

**What do I need to have so I can use the scripts?**

* **Remote PowerShell access** to your organization's domain/tenant, Exchange servers, and Skype for Business servers
* **Admin credentials** for your organization's domain/tenant, Exchange servers, and Skype for Business servers

**What feedback are we looking for?**

* **Run the scripts.**
  + If you can create an account in your environment, use the creation scripts to make a new device account.
    - Test out a device account on a Surface Hub. (If you're willing to switch out an account on a device.)
  + Validate any existing account you're using on Surface Hubs currently, or any accounts that are created via the scripts
* **Let us know about your setup.**
  + Exchange and Skype versions?
  + On-prem, online, or hybrid?
  + If on-prem or hybrid, single or multi-forest?
* **Answer some questions about usage.**
  + How easy to use were the scripts?
  + Was there any input that was required that was confusing? Did you have trouble finding the right info for the input?
  + Did any of the scripts have errors while running (e.g. a component broke or the script exited with an error before completion)
  + Did you have to modify the scripts to get them to work?
  + Any other general feedback (including on documentation)

Instructions for the scripts and the scripts themselves will be provided in a zip file.

## What is a device account?

A “device account” is an account that the Microsoft Surface Hub uses to sync its meetings calendar, send mail, and enable Skype for Business compatibility. Users can book this account by scheduling a meeting with it, and the Surface Hub will be able join that meeting and provide various features to the meeting attendees. Without a device account, none of the aforementioned features will work.

Every device account should be unique to a single Surface Hub device, and requires some setup, so you need to properly configure one device account for each Surface Hub your organization owns.

## Purpose of scripts

The scripts included for device account setup for Surface Hub serve two purposes:

1. To create device accounts that are compatible with a Surface Hub in either a completely on-premises infrastructure (i.e. directory, Exchange, and Skype for business servers are hosted within your organization) or a completely online infrastructure (Office 365).
2. To validating existing accounts in any infrastructure to check if they are compatible with a Surface Hub.

## Contents

|  |  |  |
| --- | --- | --- |
| Script | Purpose | Infrastructure requirements |
| SHAccountCreateOnPrem.ps1 | Creates a device account on an on-premises infrastructure | Single-forest on-prem, using Exchange 2013 or later and Skype for Business 2013 or later |
| SHAccountCreateO365.ps1 | Creates a device account on an online (Office 365) infrastructure | Pure Office 365 online (no hybrid or Office 365 on-prem) |
| SHAccountValidate.ps1 | Validates a device account on any infrastructure | Any on-prem, online, or hybrid setup |
| EnableSfb.ps1 | Enables a specified account for Skype for Business on either online or on-prem topologies | Single-forest on-prem, using Exchange 2013 or later and Skype for Business 2013 or later OR Pure Office 365 online (no hybrid or Office 365 on-prem) |

## What scripts should I use?

If you plan to create device accounts from scratch, you should use the account creation scripts if you have a compatible environment setup (i.e. SHAccountCreateOnPrem.ps1 or SHAccountCreateO365.ps1).

The account creation scripts cannot modify already existing accounts, but can be used as a basis for what cmdlets need to be run to configure the existing accounts correctly.

If you’re planning to modify already-existing accounts to become device accounts, run the validation script to check if any properties of the existing account are misconfigured, and then use the troubleshooting document to run the appropriate cmdlets to fix them.

Whether you’re creating a new account or modifying an already-existing account, the validation script can inform you if your device account is ready to be added to a Surface Hub. You should always run the validation script before adding a device account to your Surface Hub.

## What do the scripts do?

The account creation scripts will:

* Ask for administrator credentials
* Create device accounts in your domain/tenant
* Create and/or assign a Surface Hub-compatible ActiveSync policy to the device account(s)
* Set various Exchange and Skype for Business attributes for the created account(s)
* Assign licenses and permissions to the created account(s)

In detail, these are the properties that are set:

|  |  |  |
| --- | --- | --- |
| Cmdlet | Attribute | Value |
| Set-Mailbox | RoomMailboxPassword | User-provided |
|  | EnableRoomMailboxAccount | True |
|  | Type | Room |
| Set-CalendarProcessing | AutomateProcessing | True |
|  | RemovePrivateProperty | False |
|  | DeleteComments | False |
|  | AddOrganizerToSubject | False |
|  | AddAdditionalResponse | True |
| Set-MobileDeviceMailboxPolicy (Set-ActiveSyncMailboxPolicy for Exchange 2010) | PasswordEnabled (DevicePasswordEnabled for Exchange 2010) | False |
|  | AllowNonProvisionableDevices | True |
| Set-CSMeetingRoom | Enabled (This is set by using the Enable-CsMeetingRoom cmdlet) | True |
|  | SipAddress (This is set by using the Enable-CsMeetingRoom cmdlet) | Set to the UPN of the device account (If set to something else, this can be changed in the Settings app on the Surface Hub. |
| Set-MsolUserLicense (O365 only) | AddLicenses | User-provided |
| Set-AdUser (On-prem only) | Enabled | True |
| Set-MsolUserLicense (O365 only)/ Set-AdUser (On-prem only) | PasswordNeverExpires | True |

# Requirements

## Pre-requisites

You must have a proper network or internet connection from your PC that can contact your AD or AAD controller, and also reach your Exchange and Lync servers.

In order to use these scripts and the included cmdlets, your infrastructure must be compatible with the [infrastructure requirements for the scripts](#_Contents).

Make sure that you have the right access to run the PowerShell scripts.

1. Start a PowerShell session as Administrator.
2. Run the cmdlet Set-ExecutionPolicy Unrestricted

This will enable the use of these scripts.

### Online pre-requisites

In order to run cmdlets used by these PowerShell scripts, the following must be installed for the admin PowerShell console if you’re using an online or hybrid setup:

1. [Microsoft Online Services Sign-In Assistant for IT Professionals BETA](http://www.microsoft.com/en-ca/download/details.aspx?id=39267)
2. [Windows Azure Active Directory Module for Windows PowerShell (64-bit version)](http://go.microsoft.com/fwlink/p/?linkid=236297) or [(32-bit version)](http://go.microsoft.com/fwlink/p/?linkid=236298)
3. [Windows PowerShell Module for Lync Online](http://www.microsoft.com/en-us/download/details.aspx?id=39366)

### On-prem pre-requisites

If you are using an on-premises or hybrid setup where you are connecting to an on-premises server, you must make two adjustments to your Exchange administrator account:

1. The account must be remote-PowerShell-enabled. This allows the admin to use the PowerShell cmdlets that are needed by the script. (This permission can be set using **set-user $admin -RemotePowerShellEnabled $true**)
2. The account must have the “Reset Password” role if you plan to run the creation scripts. This allows the admin to change the password of the account, which is needed for the script. The Reset Password Role can be enabled via the Exchange Admin Center.

## On-prem/Active Directory requirements

You should have the following information ready before running any of the scripts.

|  |  |  |
| --- | --- | --- |
| Property | Description | Property can be found by: |
| Room UPN | The UPN for the device account | User-provided |
| Room display name | The display name for the device account. | User-provided |
| Device account password | The password for the device account. Note that it must meet the password requirements of your domain/tenant. | User-provided |
| AD domain controller | The domain controller for device account’s linked master account. (This is only relevant for device accounts that are part of a multi-forest setup) | To get the linked master account:  Get-Mailbox -Identity $strUpn  $strLinkedAccount = $mailbox.LinkedMasterAccount  To get the domain controller, set $domain to the domain found in linked master account. For creds, provide creds that have access to the domain controller:  $dc = Get-ADDomainController -Server $domain -Credential (Get-Credential)  $dc = $dc.HostName |
| AD Admin username and password | The credentials for the admin account that has access to the AD domain controller | N/A |
| Skype for Business server | The Skype for Business server that the device account is using. | User-provided |
| Skype for Business Admin username and password | The credentials for the admin account that has permissions to enable and modify account properties for Skype for Business. This can be the same credentials as the Active Directory and Exchange admin. | User-provided |
| Exchange server | The Exchange server that the device account is using. | For a mailbox on your Exchange server, you can run:  $mailbox = Get-Mailbox –Identity $strUpn  $exchServer = $mailbox.PSComputerName |
| Exchange Admin username and password | The credentials for the admin account that has permissions to enable and modify account properties for Exchange. This can be the same credentials as the Skype for Business and Active Directory admin. | User-provided |
| Surface Hub EAS policy name | The name of an ActiveSync policy to be created that will be configured with the recommended policies | User-provided |
| Registrar pool (SIP pool) | The registrar pool for the Lync/Skype for Business deployment. | User-provided (usually the FQDN of the Lync server) |

## Online/Office 365 requirements

In order to create an account for an online deployment, the following information is needed:

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Property can be found by:** |
| Room UPN | The UPN for the device account | User-provided |
| Room display name | The display name for the device account. | User-provided |
| Device account password | The password for the device account. Note that it must meet the password requirements. | User-provided |
| Admin username and password | The credentials for the admin account that has permissions to create and configure Exchange and Lync accounts in the domain/tenancy. | User-provided |
| Surface Hub EAS policy name | The name of an ActiveSync policy to be created that will be configured with the recommended policies | User-provided |
| License SKU | The O365 license to assign to the device account. | Get-MsolAccountSku |
| Location | The 2-character code indicating the location of the account, needed to apply a license. | User-provided |
| Registrar pool (SIP pool) | The registrar pool for the Lync/Skype for Business deployment. (Note that the UPN of an existing Lync-enabled account may be required in order to find the registrar pool.) | $strRegPool = (Get-CsTenant).TenantPoolExtension  $strRegPool = $strRegPool.Substring($strRegPool[0].IndexOf(':') + 1) |

# Running the scripts

## Creation scripts

### On-prem scripts

#### SHAccountCreateOnPrem

##### Requirements

This script must be run in an Administrator PowerShell session, and used to create accounts in a single-forest pure on-prem setup using only Exchange 2013 or later and Skype for Business 2013 or later.

The following info is required:

* For the account you want to create:
  + UPN
  + Display Name
  + Password
* Admin credentials (the credentials can have permission as an Exchange and Skype for Business admin, otherwise you will need to provide each separately)
* FQDN of Exchange server
* FQDN of Skype for Business server
* EAS policy name (a new policy can be created from the script, or an existing policy can be used)
* Skype for Business registrar pool (may be the same as the FQDN of Skype for Business server)

##### What does the script do?

1. Starts a remote PowerShell session that is connected to the domain, Exchange server, and Skype for Business server via the provided admin credentials.
2. Creates an Exchange mailbox with a password, and create a user in AD.
3. Creates and assign an ActiveSync policy that is compatible with Surface Hub.
4. Configures the account for automatically accepting and declining meetings, and sets various Exchange mailbox settings.
5. Configures the password to not expire and enables the account for authentication.
6. Enables the account for Skype for Business.

### Online scripts

#### SHAccountCreateO365

##### Requirements

This script must be run in an Administrator PowerShell session. This script must be run in an Administrator PowerShell session, and used to create accounts in pure online setup using only Exchange Online or later and Skype for Business Online.

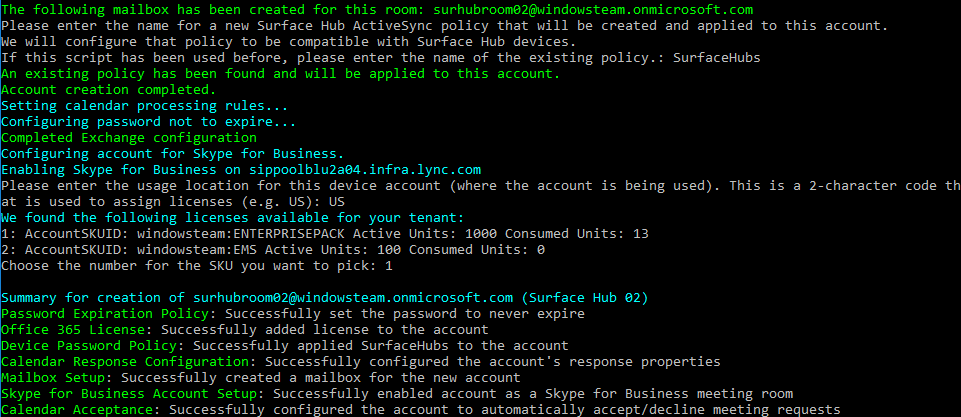
The following info is required:

* For the account you want to create:
  + UPN
  + Display name
  + Password
* Admin credentials (the credentials must have permission as an Exchange and Skype for Business admin)
* EAS policy name
* Region for the O365 license (e.g. “US”)
* O365 license name (all compatible licenses will be shown as a part of running the script)

##### What does the script do?

1. Starts a remote PowerShell session that is connected to an O365 tenant via the provided admin credentials.
2. Creates an account in AAD, and give it an Exchange mailbox with a password.
3. Creates and assign an ActiveSync policy that is compatible with Surface Hub.
4. Configures the account for automatically accepting and declining meetings, and sets various Exchange mailbox settings.
5. Configures the password to not expire.
6. Enables the account for Skype for Business.
7. Assigns an O365 license to the account.

##### Sample output



### Miscellaneous scripts

#### EnableSfb

##### Requirements

This script must be run in an Administrator PowerShell session. Sometimes a delay is required for a device account to be ready to be enabled for Skype for Business. In this case, the creation scripts will fail but EnableSfb.ps1 can be run afterwards within the same PowerShell session.

Note that this script is intended to be run on an existing account.

The following info is required:

* Account environment (i.e. does it live online or on-prem)
* Admin credentials (the credentials must have permission as a Skype for Business admin)
* UPN for the account you want to enable SfB for
* Skype for Business registrar pool

##### What does the script do?

1. Asks if the target account is hosted on-line or on-prem.
   1. If the account is on-prem, the script will ask for the SfB registrar pool
2. Asks for the UPN of the account to enable for SfB
3. Asks for Skype for Business admin credentials.
4. Enables the account for Skype for Business.

## Validation scripts

### SHAccountValidate.ps1

##### Requirements

This script must be run in an Administrator PowerShell session.

Note that this script is intended to be run on an existing account.

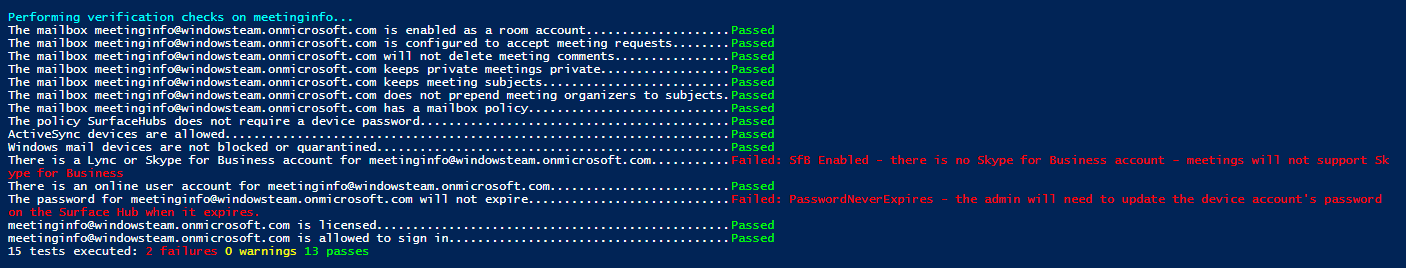
The following info is required:

* Email address of the account you want to validate
* Exchange server (can be discovered for you if the Exchange server is hosted online)
* Skype for Business registrar pool (can be discovered for you if the Skype for Business server is hosted online)
* Admin credentials (the credentials can have permission as an Exchange and Skype for Business admin, otherwise you will need to provide each separately)
* Account environment (i.e. does it live online or on-prem)
* Domain controller that hosts the account (for on-prem only)

##### What does the script do?

1. Ask for the email address of the account you wish to validate.
2. Asks for the Exchange server and an admin for that Exchange server.
3. Asks for the Skype for Business registrar pool, and an admin for the associated Skype for Business Server.
4. Asks if the account has any services hosted online or on-prem.
5. Asks for the domain controller if you picked on-prem for step 5, and if the account is in a multi-forest environment.
   1. When asked for a domain controller, the script will also ask for admin creds to that domain controller.
6. Performs validation of various attributes on the account.

#### Sample output



# Troubleshooting

Each script produces output that indicates if any intended operations fail. For the creation scripts, the script will show if any steps in the creation process were not finished properly. For the validation scripts, the script will show which validation checks pass and which checks fail.

Visit [here](https://microsoft.sharepoint.com/teams/Surface_Hub_Pilot_Programs/Shared%20Documents/Documentation/Troubleshooting/SurfaceHub_Setup_troubleshooting.xlsx?web=1) for troubleshooting. Troubleshooting for the creation scripts is under the “Scripts\_Tshoot\_Create” sheet. Troubleshooting for the validation scripts is under the “Scripts\_Tshoot\_Validate” sheet.

# ~~Appendix~~

~~This contains old information that may only be relevant to old versions of the scripts.~~

### ~~CSV scripts~~

#### ~~SHAccountValidateO365\_CSV~~

##### ~~Requirements~~

~~This script must be run in an Administrator PowerShell session.~~

~~The following info is required:~~

* ~~Admin credentials (the credentials must have permission as an Exchange and Skype for Business admin)~~
* ~~SHAccounts.csv~~

~~SHAccounts.csv must live in the same directory as the validation script. Its first line must be:~~

~~Alias,DisplayName,Tenant,Password,SipPool,Location,Policy,Licenses~~

~~The attributes for each account must be provided appropriately.~~

##### ~~What does the script do?~~

1. ~~Starts a remote PowerShell session that is connected to an O365 tenant via the provided admin credentials.~~

~~Then for every account in the CSV, the script will:~~

1. ~~Check that the account has a room mailbox.~~
2. ~~Check that the account is configured to automatically accept/decline meetings.~~
3. ~~Check that the account is enabled for Skype for Business.~~
4. ~~Check that the account has a SIP address.~~
5. ~~Check that the account’s password will not expire.~~
6. ~~Check that the account has an O365 license.~~
7. ~~Check that the account has a compatible ActiveSync policy.~~
8. ~~Check that the tenant has compatible device access rules.~~
9. ~~Print any checks that do not pass.~~

#### ~~SHAccountCreateO365\_CSV~~

##### ~~Requirements~~

~~This script must be run in an Administrator PowerShell session.~~

~~The following info is required:~~

* ~~Admin credentials (the credentials must have permission as an Exchange and Skype for Business admin)~~
* ~~SHAccounts.csv~~

~~SHAccounts.csv must live in the same directory as the validation script. Its first line must be:~~

~~Alias,DisplayName,Tenant,Password,SipPool,Location,Policy,Licenses~~

~~The attributes for each account must be provided appropriately.~~

##### ~~What does the script do?~~

1. ~~Starts a remote PowerShell session that is connected to an O365 tenant via the provided admin credentials.~~

~~Then for every account in the CSV, the script will:~~

1. ~~Create an account in AAD, and give it an Exchange mailbox with a password.~~
2. ~~Create and assign an ActiveSync policy that is compatible with Surface Hub.~~
3. ~~Configure the account for automatically accepting and declining meetings.~~
4. ~~Configure the password to not expire.~~
5. ~~Enable the account for Skype for Business.~~
6. ~~Assign an O365 license to the account.~~

#### ~~SHAccountValidateOnPrem\_CSV~~

##### ~~Requirements~~

~~This script must be run in an Administrator PowerShell session.~~

~~The following info is required:~~

* ~~Admin credentials (the credentials can have permission as an Exchange and Skype for Business admin, otherwise you will need to provide each separately)~~
* ~~SHAccounts.csv~~

~~SHAccounts.csv must live in the same directory as the validation script. Its first line must be:~~

~~Alias,DisplayName,Tenant,Password,SipPool,Location,Policy~~

~~The attributes for each account must be provided appropriately.~~

##### ~~What does the script do?~~

1. ~~Starts a remote PowerShell session that is connected to the domain, Exchange server, and Skype for Business server via the provided admin credentials.~~

~~Then for every account in the CSV, the script will:~~

1. ~~Check that the account has a room mailbox.~~
2. ~~Check that the account is configured to automatically accept/decline meetings.~~
3. ~~Check that the account is enabled for Skype for Business.~~
4. ~~Check that the account has a SIP address.~~
5. ~~Check that the account’s password will not expire.~~
6. ~~Check that the account is enabled.~~
7. ~~Check that the account has a compatible ActiveSync policy.~~
8. ~~Check that the organization has a compatible device access rule.~~
9. ~~Print any checks that do not pass.~~