

MS-ASCMD Test Suite Specification

**Abstract:** This document provides information about how to configure the test suite and how the MS-ASCMD test suite is designed to test the MS-ASCMD Open Specification usability and accuracy. It describes test assumptions, scope and constraints of the test suite. It also specifies test scenarios, detailed test cases, test suite architecture and adapter design.

Contents

[1 Configuring the test suite 3](#_Toc387411343)

[1.1 Configuring the test suite client 3](#_Toc387411344)

[1.1.1 Configuring the test suite client manually 3](#_Toc387411345)

[1.1.2 Configuring the test suite client by scripts 4](#_Toc387411346)

[1.2 Configuring the system under test (SUT) 4](#_Toc387411347)

[1.2.1 Configuring the SUT manually 4](#_Toc387411348)

[1.2.2 Configuring the SUT by scripts 4](#_Toc387411349)

[1.3 Configuring the SHOULD/MAY requirements 4](#_Toc387411350)

[2 Test suite design 5](#_Toc387411351)

[2.1 Assumptions, scope and constraints 5](#_Toc387411352)

[2.2 Test suite architecture 5](#_Toc387411353)

[2.3 Technical dependencies and considerations 7](#_Toc387411354)

[2.4 Adapter design 7](#_Toc387411355)

[2.4.1 Adapter overview 7](#_Toc387411356)

[2.4.2 Technical feasibility of adapter approach 7](#_Toc387411357)

[2.4.3 Adapter abstract layer 8](#_Toc387411358)

[2.4.4 Adapter details 9](#_Toc387411359)

[2.5 Test scenarios 11](#_Toc387411360)

[2.5.1 S01\_Autodiscover 12](#_Toc387411361)

[2.5.2 S02\_FolderCreate 12](#_Toc387411362)

[2.5.3 S03\_FolderDelete 12](#_Toc387411363)

[2.5.4 S04\_FolderSync 13](#_Toc387411364)

[2.5.5 S05\_FolderUpdate 13](#_Toc387411365)

[2.5.6 S06\_GetAttachment 13](#_Toc387411366)

[2.5.7 S07\_GetItemEstimate 13](#_Toc387411367)

[2.5.8 S08\_ItemOperations 14](#_Toc387411368)

[2.5.9 S09\_MeetingResponse 14](#_Toc387411369)

[2.5.10 S10\_MoveItems 14](#_Toc387411370)

[2.5.11 S11\_Ping 15](#_Toc387411371)

[2.5.12 S12\_Provision 15](#_Toc387411372)

[2.5.13 S13\_ResolveRecipients 15](#_Toc387411373)

[2.5.14 S14\_Search 15](#_Toc387411374)

[2.5.15 S15\_SendMail 15](#_Toc387411375)

[2.5.16 S16\_Settings 16](#_Toc387411376)

[2.5.17 S17\_SmartForward 16](#_Toc387411377)

[2.5.18 S18\_SmartReply 16](#_Toc387411378)

[2.5.19 S19\_Sync 16](#_Toc387411379)

[2.5.20 S20\_ValidateCert 17](#_Toc387411380)

[2.5.21 S21\_CommonStatusCode 17](#_Toc387411381)

[2.6 Test case design 17](#_Toc387411382)

[2.6.1 Traditional test case design 17](#_Toc387411383)

[2.6.2 Test case description 23](#_Toc387411384)

# Configuring the test suite

## Configuring the test suite client

### Configuring the test suite client manually

Before you run the test suite, update the values in the MS-ASCMD\_TestSuite.deployment.ptfconfig file. The MS-ASCMD\_TestSuite.deployment.ptfconfig file can also be configured by running the client setup script.

1. Open MS-ASCMD\TestSuite\MS-ASCMD\_TestSuite.deployment.ptfconfig.
2. Update the following value to specify the common configuration file.

Property name="CommonConfigurationFileName" value="ExchangeCommonConfiguration.deployment.ptfconfig"

**Note** This property can be removed or set to empty if the required properties are copied to the test suite specific configuration file. Any other changes to this property will cause all test cases in the test suite to fail during execution. The test suite searches through its specific configuration file and uses those properties, if they are defined, before looking for them in the common configuration file (if specified).

1. Update the following properties' values to match SUT settings and configuration.

* Property name="User1Name" value="MSASCMD\_User01"
* Property name="User1Password" value="Password01!"
* Property name="User2Name" value="MSASCMD\_User02"
* Property name="User2Password" value="Password01!"
* Property name="User3Name" value="MSASCMD\_User03"
* Property name="User3Password" value="Password01!"
* Property name="User7Name" value="MSASCMD\_User07"
* Property name="User7Password" value="Password01!"
* Property name="User8Name" value="MSASCMD\_User08"
* Property name="User8Password" value="Password01!"
* Property name="User9Name" value="MSASCMD\_User09"
* Property name="User9Password" value="Password01!"
* Property name="SharedFolder" value="\\[SutComputerName]\MSASCMD\_SharedFolder"
* Property name="AmbiguousSearchName" value="MSASCMD\_SearchUser"
* Property name="GroupDisplayName" value="MSASCMD\_TestGroup"
* Property name="LargeGroupDisplayName" value="MSASCMD\_LargeGroup"
* Property name="AcceptableResponseSchema" value="http://schemas.microsoft.com/exchange/autodiscover/mobilesync/responseschema/2006"
* Property name="SharedDocument1" value="[SharedFolder]\MSASCMD\_Non-emptyDocument.txt"
* Property name="SharedDocument2" value="[SharedFolder]\MSASCMD\_EmptyDocument.txt"
* Property name="MIMEMailSubject" value="MSASCMD\_SecureEmailForTest"

### Configuring the test suite client by scripts

To configure the test suite client using scripts, see section 5.2.1 of the [ExchangeEASTestSuiteDeploymentGuide.docx](../ExchangeEASTestSuiteDeploymentGuide.docx).

## Configuring the system under test (SUT)

### Configuring the SUT manually

To manually configure the SUT, see section 5.1.2 of the [ExchangeEASTestSuiteDeploymentGuide.docx](../ExchangeEASTestSuiteDeploymentGuide.docx).

### Configuring the SUT by scripts

To configure the SUT using scripts, see section 5.1.1 of the [ExchangeEASTestSuiteDeploymentGuide.docx](../ExchangeEASTestSuiteDeploymentGuide.docx).

## Configuring the SHOULD/MAY requirements

Implementation of the SHOULD/MAY and endnote-related requirements are pre-configured in the format "<Property name="RXXXEnabled" value="XXXX"/>" for the product versions in the following config files:

* MS-ASCMD\_ExchangeServer2007\_SHOULDMAY.deployment.ptfconfig
* MS-ASCMD\_ExchangeServer2010\_SHOULDMAY.deployment.ptfconfig
* MS-ASCMD\_ExchangeServer2013\_SHOULDMAY.deployment.ptfconfig

If RXXXEnabled is set to true, the requirement must be checked. If false, the requirement must not be checked. For Microsoft product versions, all values should not be changed. For third-party products, the closest Microsoft product version should be chosen, and the value of RXXXEnabled should be updated according to the real product behavior. For example, if Exchange Server 2010 is chosen,user can open **MS-ASCMD\_ExchangeServer2010\_SHOULDMAY.deployment.ptfconfig** and update the RXXXEnabled accordingly.

# Test suite design

## Assumptions, scope and constraints

Assumptions

None.

Scope

In scope

* This test suite will verify the accuracy and integrity of the technical content in the Open Specification against the results returned from the protocol server by using 20 commands: Autodiscover, FolderCreate, FolderDelete, FolderSync, FolderUpdate, GetAttachment, GetItemEstimate, ItemOperations, MeetingResponse, MoveItems, Ping, Provision, ResolveRecipients, Search, SendMail, Settings, SmartForward, SmartReply, Sync, and ValidateCert.
* This test suite will verify the server-side and testable requirements by running all test cases on both HTTP and HTTPS.
* This test suite will verify all the XML schema elements embedded inside of the command response.
* This test suite will verify requirements from Exchange ActiveSync: WAP Binary XML (WBXML) Algorithm ([MS-ASWBXML]) and Exchange ActiveSync: Data Types ([MS-ASDTYPE]).

Out of scope

* This test suite will not verify the requirements related to client behaviors.
* This test suite will not verify the requirements related to server internal behaviors.
* This test suite will not verify the internal implementations of its transport protocol stack.

Constraints

None.

## Test suite architecture

This test suite verifies the server-side and testable requirements obtained from the Open Specification. The following figure shows the architecture of this test suite.



The architecture of the test suite

The details of the MS-ASCMD test suite architecture

* SUT hosts the data synchronization and exchanging service, which this test suite runs against.
* From third-party user’s point of view, the SUT is the protocol server implementation.
* The following products have been tested with the MS-ASCMD test suite on the Windows platform.
* Microsoft Exchange Server 2007 Service Pack 3 (SP3)
* Microsoft Exchange Server 2010 Service Pack 3 (SP3)
* Microsoft Exchange Server 2013 Service Pack 1 (SP1)
* The test suite acts as the client to communicate with the SUT and validates the requirements gathered from MS-ASCMD Open Specification.
* Test cases use the MS-ASCMD protocol adapter to call and get the results of the MS-ASCMD commands. Test cases also use the SUT control adapter to configure and retrieve information from SUT.
* MS-ASCMD protocol adapter is used in the test cases. The test cases call the methods in the interface to invoke the MS-ASCMD protocol adapter’s commands.
* MS-ASCMD protocol adapter uses ActiveSyncClient to send command request and retrieve command response.
* ActiveSyncClient encodes and decodes ActiveSync commands defined in [MS-ASCMD] by using MS-ASWBXML and communicates with the SUT via MS-ASHTTP.
* SUT control adapter is used in the test cases. The test cases call the method in the interface to configure the SUT.

## Technical dependencies and considerations

Dependencies

* This test suite depends on HTTP protocol or HTTPS protocol to transmit the messages.
* This test suite depends on Protocol Test Framework (PTF) to derive managed adapters.
* This test suite depends on MS-ASWBXML, which encodes request bodies and decodes response bodies into WBXML for transmission to an ActiveSync server.
* This test suite depends on the xsd.exe tool in the .NET Framework SDK to generate structures used in the MS-ASCMD request and response.
* This test suite depends on MS-ASHTTP to synchronize data which is stored on the server.

Encryption consideration

* Transportation of MS-ASCMD includes HTTP and HTTPS, and encryption will be handled by HTTPS.

## Adapter design

### Adapter overview

One protocol adapter and one SUT control adapter are used in this test suite.

Protocol adapter

* MS-ASCMD protocol adapter
* The MS-ASCMD adapter is a managed adapter, which is derived from the ManagedAdapterBase class in PTF.
* The MS-ASCMD adapter has the following functionalities
* Choose HTTP or HTTPS for transport;
* Construct requests of 20 MS-ASCMD commands;
* Communicate with the SUT by sending command requests and receiving the corresponding responses from the SUT;
* Parse the response messages and validate the messages according to the XSD schema.

SUT control adapter

* SUT control adapter
* The SUT control adapter will be a PowerShell scripted adapter.
* The SUT control adapter has the following functionality
* Delete a user’s ActiveSync device.
* The SUT control adapter is invoked by the test cases.

### Technical feasibility of adapter approach

Message generation

The MS-ASCMD adapter gets the parameter values and calls the corresponding commands in ActiveSyncClient. The ActiveSyncClient serializes the parameter values to XML elements, and then encodes the request bodies into WBXML for transmission to an ActiveSync server.

Message consumption

The messages received from the SUT will be parsed in the ActiveSyncClient and be passed upon to the MS-ASCMD adapter. Then these messages are consumed in the MS-ASCMD adapter to validate the message format and to validate the logic-related requirements in the test cases.

SUT control adapter

The SUT control adapter is designed to remotely control the SUT to delete a user’s ActiveSync device.

### Adapter abstract layer

Protocol adapter

MS-ASCMD adapter interface

There are 26 methods declared in the MS-ASCMD adapter interface IMS\_ASCMDAdapter.

* 20 methods correspond to the 20 MS-ASCMD commands: Autodiscover, FolderCreate, FolderDelete, FolderSync, FolderUpdate, GetAttachment, GetItemEstimate, ItemOperations, MeetingResponse, MoveItems, Ping, Provision, ResolveRecipients, Search, SendMail, Settings, SmartForward, SmartReply, Sync, and ValidateCert. Other six methods ChangePolicyKey, SwitchUser, ChangeHeaderEncodingType, ChangeDeviceID, ChangeDeviceType and SendStringRequest are customized methods.

The methods are described in the following table.

|  |  |  |
| --- | --- | --- |
| No. | Method | Description |
| 1 | Autodiscover | This method is used to facilitate the discovery of core account configuration information by using the user's Simple Mail Transfer Protocol (SMTP) address as the primary input. |
| 2 | FolderCreate | This method is used to create a new folder as a child folder of the specified parent folder. |
| 3 | FolderDelete | This method is used to delete a folder from the server. |
| 4 | FolderSync | This method is used to synchronize the collection hierarchy. |
| 5 | FolderUpdate | This method is used to move a folder from one location to another on the server. |
| 6 | GetAttachment | This method is used to retrieve an email attachment from the server. |
| 7 | GetItemEstimate | This method is used to get an estimate of the number of items in a collection or folder on the server that have to be synchronized. |
| 8 | ItemOperations | This method is used to act as a container for the Fetch element, the EmptyFolderContents element, and the Move element to provide batched online handling of these operations against the server. |
| 9 | MeetingResponse | This method is used to accept, tentatively accept, or decline a meeting request in the user's Inbox folder or Calendar folder. |
| 10 | MoveItems | This method is used to move an item or items from one folder on the server to another. |
| 11 | Ping | This method is used to request that the server monitor specified folders for changes that would require the client to resynchronize. |
| 12 | Provision | This method is used to enable client devices to request from the server the security policy settings that the administrator sets. |
| 13 | ResolveRecipients | This method is used to resolve a list of supplied recipients, retrieves their free/busy information, or retrieves their S/MIME certificates so that clients can send encrypted S/MIME email messages. |
| 14 | Search | This method is used to find entries in an address book, mailbox, or document library (UNC or Windows® SharePoint® Services). |
| 15 | SendMail | This method is used to send MIME-formatted email messages to the server. |
| 16 | Settings | This method is used to support, get and set operations on global properties and Out of Office (OOF) settings for the user. Sends device information to the server, implements the device password/personal identification number (PIN) recovery, or retrieves a list of the user's email addresses. |
| 17 | SmartForward | This method is used to forward messages without retrieving the full, original message from the server. |
| 18 | SmartReply | This method is used to reply to messages without retrieving the full, original message from the server. |
| 19 | Sync | This method is used to synchronize changes in a collection between the client and the server. |
| 20 | ValidateCert | This method is used to validate a certificate that has been received via an S/MIME mail. |
| 21 | ChangeDeviceID | This method is used to change device id. |
| 22 | ChangeDeviceType | This method is used to change device type. |
| 23 | ChangeHeaderEncodingType | This method is used to change http request header encoding type. |
| 24 | ChangePolicyKey | This method is used to change the specified PolicyKey. |
| 25 | SendStringRequest | This method is used to send a plain text request. |
| 26 | SwitchUser | This method is used to change user to call ActiveSync operation. |

MS-ASCMD adapter interface methods

SUT control adapter

SUT control adapter interface

There are three methods declared in the SUT control adapter interface IMS\_ASCMDSUTControlAdapter.

The methods are described in the following table.

|  |  |  |
| --- | --- | --- |
| No. | Method | Description |
| 1 | DeleteDevice | This method is used to delete a user’s ActiveSync device. |
| 2 | GetMailboxFolderPermission | This method is used to get the value of SUT’s AccessRights property. |
| 3 | SetMailboxFolderPermission | This method is used to set SUT’s AccessRights property to a specified value. |

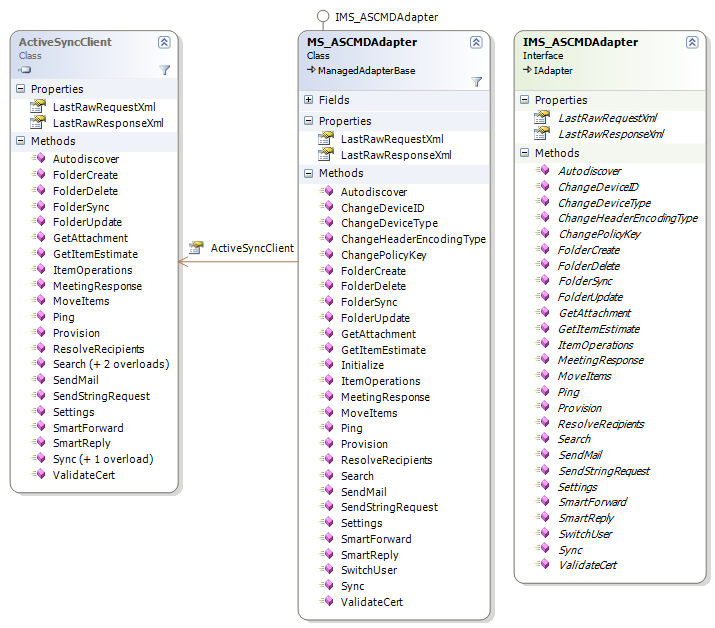
SUT control adapter interface method

### Adapter details

#### Protocol adapter

##### MS-ASCMD protocol adapter

The following figure shows the class diagram of the MS-ASCMD protocol adapter and the relationship between MS\_ASCMDAdapter and ActiveSyncClient.



MS-ASCMD adapter and ActiveSyncClient class diagram

The following outlines details of the class diagram:

Adapter interface

* IMS\_ASCMDAdapter is the interface of the protocol adapter.
* IMS\_ASCMDAdapter defines the methods invoked by test cases, including Autodiscover, FolderCreate, FolderDelete, FolderSync, FolderUpdate, GetAttachment, GetItemEstimate, ItemOperations, MeetingResponse, MoveItems, Ping, Provision, ResolveRecipients, Search, SendMail, SendStringRequest, Settings, SmartForward, SmartReply, Sync, ChangePolicyKey, SwitchUser, ChangeHeaderEncodingType, ChangeDeviceID, ChangeDeviceType, SendStringRequest and ValidateCert methods.
* The two properties LastRawRequestXml and LastRawResponseXml are used to get the raw request and raw response.

Adapter implementation

* MS\_ASCMD is the protocol adapter class of the test suite. It is used to implement IMS\_ASCMDAdapter.
* The initialize method is used to initialize an instance of ActiveSyncClient.
* The two properties LastRawRequestXml and LastRawResponseXml are used to get the raw request and raw response.

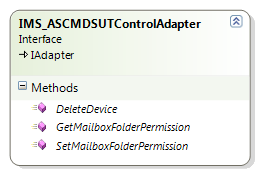
Other class

* ActiveSyncClient is used by MS\_ASCMDAdapter to send (Autodiscover, FolderCreate, FolderDelete, FolderSync, FolderUpdate, GetAttachment, GetItemEstimate, ItemOperations, MeetingResponse, MoveItems, Ping, Provision, ResolveRecipients, Search, SendMail, Settings, SmartForward, SmartReply, Sync, ValidateCert, and SendStringRequest) command requests and retrieve command responses.

#### SUT control adapter

##### SUT control adapter

The following figure shows the class diagram of the SUT control adapter.



SUT control adapter class diagram

The following outlines details of the class diagram:

* The IMS\_ASCMDSUTControlAdapter is the interface of the SUT control adapter which is implemented by Microsoft PowerShell script. The implementation can be substituted by other implementation for the third party’s need.

## Test scenarios

21 scenarios are designed to cover the server-side, testable requirements in the MS-ASCMD test suite. The details of the scenarios are as follows.

|  |  |
| --- | --- |
| Scenario | Description |
| S01\_Autodiscover | Discover the account settings by using the Autodiscover command. |
| S02\_FolderCreate | Create the folders by using the FolderCreate command. |
| S03\_FolderDelete | Delete the folders by using the FolderDelete command. |
| S04\_FolderSync | Synchronize the folders by using the FolderSync command. |
| S05\_FolderUpdate | Update the folders by using the FolderUpdate command. |
| S06\_GetAttachment | Retrieve an email attachment from the server by using GetAttachment command. |
| S07\_GetItemEstimate | Get an estimate of the number of items in a collection or folder on the server that have to be synchronized by using GetItemEstimate command. |
| S08\_ItemOperations | Provide the batched online handling of Fetch, empty the folder contents and move the operations against the server by using the ItemOperations command. |
| S09\_MeetingResponse | Accept, tentatively accept, or decline a meeting request in the user's Inbox folder or Calendar folder by using the MeetingResponse command. |
| S10\_MoveItems | Move an item or items from one folder on the server to another by using the MoveItems command. |
| S11\_Ping | Ping the server for updating the folder by using the Ping command. |
| S12\_Provision | Download and acknowledge policies of Provision by using the Provision command. |
| S13\_ResolveRecipients | Resolve the recipients and retrieve the free or busy data by using the ResolveRecipients command. |
| S14\_Search | Find the entries in an address book, mailbox, or document library by using the Search command. |
| S15\_SendMail | Send the email message by using the SendMail command. |
| S16\_Settings | Set or change the device information, OOF setting and user information by using the Settings command. |
| S17\_SmartForward | Forward the email message by using the SmartForward command. |
| S18\_SmartReply | Reply the email message by using the SmartReply command. |
| S19\_Sync | Synchronize the data by using the Sync command. |
| S20\_ValidateCert | Validate a certificate that has been received via an S/MIME mail by using ValidateCert command. |
| S21\_CommonStatusCode | Test the common negative status for the commands. |

MS-ASCMD scenarios

### S01\_Autodiscover

Description

Discover the account settings by using the Autodiscover command.

Commands

* Autodiscover

### S02\_FolderCreate

Description

Create the folders by using the FolderCreate command.

Commands

* FolderCreate
* FolderSync
* FolderDelete

### S03\_FolderDelete

Description

Delete the folders by using the FolderDelete command.

Commands

* FolderSync
* FolderCreate
* FolderDelete

### S04\_FolderSync

Description

Synchronize the folders by using the FolderSync command.

Commands

* FolderSync
* FolderCreate
* FolderDelete
* FolderUpdate

### S05\_FolderUpdate

Description

Update the folders by using the FolderUpdate command.

Commands

* FolderSync
* FolderCreate
* FolderUpdate
* FolderDelete

### S06\_GetAttachment

Description

Retrieve an email attachment from the server by using GetAttachment command.

Commands

* Sync
* GetAttachment

### S07\_GetItemEstimate

Description

Get an estimate of the number of items in a collection or folder on the server that have to be synchronized by using GetItemEstimate command.

Commands

* FolderSync
* Sync
* GetItemEstimate

### S08\_ItemOperations

Description

Provide the batched online handling of Fetch, empty the folder contents and move the operations against the server by using the ItemOperations command.

Commands

* SendMail
* FolderSync
* FolderCreate
* Sync
* ItemOperations
* SmartReply

### S09\_MeetingResponse

Description

Accept, tentatively accept, or decline a meeting request in the user's Inbox folder or Calendar folder by using the MeetingResponse command.

Commands

* FolderSync
* Sync
* FolderCreate
* MeetingResponse
* MoveItems
* FolderDelete

### S10\_MoveItems

Description

Move an item or items from one folder on the server to another by using the MoveItems command.

Commands

* FolderSync
* Sync
* FolderDelete
* FolderCreate
* MoveItems

### S11\_Ping

Description

Ping the server for updating the folder by using the Ping command.

Commands

* FolderSync
* Sync
* Ping

### S12\_Provision

Description

Download and acknowledge policies of Provision by using the Provision command.

Commands

* FolderSync
* Provision

### S13\_ResolveRecipients

Description

Resolve the recipients and retrieve the free or busy data by using the ResolveRecipients command.

Commands

* ResolveRecipients

### S14\_Search

Description

Find the entries in an address book, mailbox, or document library by using the Search command.

Commands

* FolderSync
* Search
* SendMail

### S15\_SendMail

Description

Send the email message by using the SendMail command.

Commands

* FolderSync
* Sync
* SendMail

### S16\_Settings

Description

Set or change the device information, OOF setting and user information by using the Settings command.

Commands

* Provision
* Settings

### S17\_SmartForward

Description

Forward the email message by using the SmartForward command.

Commands

* FolderSync
* Sync
* SendMail
* SmartForward
* SmartReply

### S18\_SmartReply

Description

Reply the email message by using the SmartReply command.

Commands

* FolderSync
* Sync
* SendMail
* SmartForward
* SmartReply

### S19\_Sync

Description

Synchronize the data by using the Sync command.

Commands

* FolderSync
* Sync
* SendMail
* SmartReply
* GetItemEstimate

### S20\_ValidateCert

Description

Validate a certificate that has been received via an S/MIME mail by using ValidateCert command.

Commands

* ValidateCert

### S21\_CommonStatusCode

Description

Test the common negative status for the commands.

Commands

* SendMail
* FolderSync
* Sync
* SmartForward
* ResolveRecipients
* Ping
* Provision

## Test case design

### Traditional test case design

Traditional testing is adopted as the test approach for this test suite. The test cases are designed to cover the server-side and testable requirements.

There are 242 traditional test cases are designed to cover the 21 scenarios mentioned in section [2.6 Test scenarios](#_Test_scenarios). Details of the traditional test cases are specified in [section 2.7.2 Test case description](#_Test_Cases_Description). The scenarios distributions of the test cases are listed in the following table.

|  |  |
| --- | --- |
| Scenario ID | Test case name |
| S01\_Autodiscover | MSASCMD\_S01\_TC01\_Autodiscover\_TypeIsMobileSync |
| MSASCMD\_S01\_TC02\_Autodiscover\_Fail |
| S02\_FolderCreate | MSASCMD\_S02\_TC01\_FolderCreate\_Success |
| MSASCMD\_S02\_TC02\_FolderCreate\_Fail |
| MSASCMD\_S02\_TC03\_FolderCreate\_Status2 |
| MSASCMD\_S02\_TC04\_FolderCreate\_Status5 |
| MSASCMD\_S02\_TC05\_FolderCreate\_Status9 |
| MSASCMD\_S02\_TC06\_FolderCreate\_Status10 |
| MSASCMD\_S02\_TC07\_FolderCreate\_Status3 |
| MSASCMD\_S02\_TC08\_FolderCreate\_RecipientInformationCache\_Status3 |
| S03\_FolderDelete | MSASCMD\_S03\_TC01\_FolderDelete\_Success |
| MSASCMD\_S03\_TC02\_FolderDelete\_Status3 |
| MSASCMD\_S03\_TC03\_FolderDelete\_Status4 |
| MSASCMD\_S03\_TC04\_FolderDelete\_Status9 |
| MSASCMD\_S03\_TC05\_FolderDelete\_Status10 |
| S04\_FolderSync | MSASCMD\_S04\_TC01\_FolderSync\_SyncKey0 |
| MSASCMD\_S04\_TC02\_FolderSync\_NoChanged |
| MSASCMD\_S04\_TC03\_FolderSync\_Changed |
| MSASCMD\_S04\_TC04\_FolderSync\_Resynchronizes |
| MSASCMD\_S04\_TC05\_FolderSync\_Status9 |
| MSASCMD\_S04\_TC06\_FolderSync\_Status10 |
| MSASCMD\_S04\_TC07\_FolderSync\_AddFolder |
| MSASCMD\_S04\_TC08\_FolderSync\_UpdateFolder |
| MSASCMD\_S04\_TC09\_FolderSync\_DeleteFolder |
| MSASCMD\_S04\_TC10\_FolderSync\_NoSynchronizeItems |
| S05\_FolderUpdate | MSASCMD\_S05\_TC01\_FolderUpdate\_Success |
| MSASCMD\_S05\_TC02\_FolderUpdate\_Status2 |
| MSASCMD\_S05\_TC03\_FolderUpdate\_Status4 |
| MSASCMD\_S05\_TC04\_FolderUpdate\_Status5 |
| MSASCMD\_S05\_TC05\_FolderUpdate\_Status9 |
| MSASCMD\_S05\_TC06\_FolderUpdate\_Status10 |
| MSASCMD\_S05\_TC07\_FolderUpdate\_Moves |
| MSASCMD\_S05\_TC08\_FolderUpdate\_Status3 |
| S06\_GetAttachment | MSASCMD\_S06\_TC01\_GetAttachment\_Success |
| MSASCMD\_S06\_TC02\_GetAttachment\_Status500 |
| S07\_GetItemEstimate | MSASCMD\_S07\_TC01\_GetItemEstimate\_Success |
| MSASCMD\_S07\_TC02\_GetItemEstimate\_WithoutOptions |
| MSASCMD\_S07\_TC03\_GetItemEstimate\_Contacts\_FilterType |
| MSASCMD\_S07\_TC04\_GetItemEstimate\_Email\_FilterType |
| MSASCMD\_S07\_TC05\_GetItemEstimate\_Calendar\_FilterType |
| MSASCMD\_S07\_TC06\_GetItemEstimate\_Tasks\_FilterType |
| MSASCMD\_S07\_TC07\_GetItemEstimate\_InvalidFilterType |
| MSASCMD\_S07\_TC08\_GetItemEstimate\_MoreThanOneMaxItems |
| MSASCMD\_S07\_TC09\_GetItemEstimate\_Status2 |
| MSASCMD\_S07\_TC10\_GetItemEstimate\_Status3 |
| MSASCMD\_S07\_TC11\_GetItemEstimate\_Status4\_InvalidSyncKey |
| MSASCMD\_S07\_TC12\_GetItemEstimate\_Status4\_MismatchedSyncKey |
| MSASCMD\_S07\_TC13\_GetItemEstimate\_ConversationMode\_Status103 |
| MSASCMD\_S07\_TC14\_GetItemEstimate\_WithCombinationClasses |
| MSASCMD\_S07\_TC15\_GetItemEstimate\_Collection\_LimitValue |
| S08\_ItemOperations | MSASCMD\_S08\_TC01\_ItemOperations\_DeleteSubFolders |
| MSASCMD\_S08\_TC02\_ItemOperations\_Status2 |
| MSASCMD\_S08\_TC03\_ItemOperations\_Status9 |
| MSASCMD\_S08\_TC04\_ItemOperations\_Status6 |
| MSASCMD\_S08\_TC05\_ItemOperations\_Fetch |
| MSASCMD\_S08\_TC06\_ItemOperations\_Move |
| MSASCMD\_S08\_TC07\_ItemOperations\_MultiplePartResponse |
| MSASCMD\_S08\_TC08\_ItemOperations\_Fetch\_MIMESupport |
| MSASCMD\_S08\_TC09\_ItemOperations\_RecipientInformationCache |
| MSASCMD\_S08\_TC10\_ItemOperations\_EmptyFolderContents |
| MSASCMD\_S08\_TC11\_ItemOperations\_Fetch\_DocumentLibrary |
| MSASCMD\_S08\_TC12\_ItemOperations\_Status4 |
| MSASCMD\_S08\_TC13\_ItemOperations\_Status5 |
| MSASCMD\_S08\_TC14\_ItemOperations\_Status8 |
| MSASCMD\_S08\_TC15\_ItemOperations\_Status10 |
| MSASCMD\_S08\_TC16\_ItemOperations\_Status11 |
| MSASCMD\_S08\_TC17\_ItemOperations\_Status15 |
| MSASCMD\_S08\_TC18\_ItemOperations\_MoreThanOneFileReference\_Status2 |
| MSASCMD\_S08\_TC19\_ItemOperations\_Status156 |
| MSASCMD\_S08\_TC20\_ItemOperations\_Status16 |
| S09\_MeetingResponse | MSASCMD\_S09\_TC01\_MeetingResponse\_AcceptMeeting |
| MSASCMD\_S09\_TC02\_MeetingResponse\_DeclineMeeting |
| MSASCMD\_S09\_TC03\_MeetingResponse\_InvalidMeeting |
| MSASCMD\_S09\_TC04\_MeetingResponse\_RequestID |
| MSASCMD\_S09\_TC05\_MeetingResponse\_ResponseNonMeeting |
| MSASCMD\_S09\_TC06\_MeetingResponse\_NonExistInstanceId |
| MSASCMD\_S09\_TC07\_MeetingResponse\_Status103 |
| MSASCMD\_S09\_TC08\_MeetingResponse\_RecurringMeetingInstanceIDInvalid |
| MSASCMD\_S09\_TC09\_MeetingResponse\_Status146 |
| MSASCMD\_S09\_TC10\_MeetingResponse\_TentativeAcceptMeeting |
| MSASCMD\_S09\_TC11\_MeetingResponse\_AcceptMeetingInCalendar |
| MSASCMD\_S09\_TC12\_MeetingResponse\_Status104 |
| MSASCMD\_S09\_TC13\_MeetingResponse\_SubstituteMeetingInvitationEmail |
| MSASCMD\_S09\_TC14\_MeetingResponse\_ForwardSubstituteMeetingInvitationEmail |
| S10\_MoveItems | MSASCMD\_S10\_TC01\_MoveItems\_Success |
| MSASCMD\_S10\_TC02\_MoveItems\_Status1\_InvalidSrcFldId |
| MSASCMD\_S10\_TC03\_MoveItems\_Status1\_InvalidSrcMsgId |
| MSASCMD\_S10\_TC04\_MoveItems\_Status2 |
| MSASCMD\_S10\_TC05\_MoveItems\_Status4 |
| MSASCMD\_S10\_TC06\_MoveItems\_Status5 |
| S11\_Ping | MSASCMD\_S11\_TC01\_Ping\_Status3 |
| MSASCMD\_S11\_TC02\_Ping\_Status4 |
| MSASCMD\_S11\_TC03\_Ping\_Status5 |
| MSASCMD\_S11\_TC04\_Ping\_Status7\_SyncRequired |
| MSASCMD\_S11\_TC05\_Ping\_Status7\_Outdate |
| MSASCMD\_S11\_TC06\_Ping\_ChangesOccurred |
| MSASCMD\_S11\_TC07\_Ping\_HeartbeatIntervalSetTooLarge |
| MSASCMD\_S11\_TC08\_Ping\_WithoutRequestBody |
| S12\_Provision | MSASCMD\_S12\_TC01\_Provision\_DownloadPolicy |
| MSASCMD\_S12\_TC02\_Provision\_WithoutPolicyTypeElement |
| MSASCMD\_S12\_TC03\_Provision\_Status144 |
| S13\_ResolveRecipients | MSASCMD\_S13\_TC01\_ResolveRecipients\_CertificateRetrieval |
| MSASCMD\_S13\_TC02\_ResolveRecipients\_EndTimeInvalid |
| MSASCMD\_S13\_TC03\_ResolveRecipients\_StartTimeInvalid |
| MSASCMD\_S13\_TC04\_ResolveRecipients\_EndTimeSmallThanStartTime |
| MSASCMD\_S13\_TC05\_ResolveRecipients\_DurationIsTooLarge |
| MSASCMD\_S13\_TC06\_ResolveRecipients\_Over256 |
| MSASCMD\_S13\_TC07\_ResolveRecipients\_NoPictureExist |
| MSASCMD\_S13\_TC08\_ResolveRecipients\_AmbiguousUser |
| MSASCMD\_S13\_TC09\_ResolveRecipients\_AmbiguousUserAvailability |
| MSASCMD\_S13\_TC10\_ResolveRecipients\_Group\_Availability |
| MSASCMD\_S13\_TC11\_ResolveRecipients\_Status161 |
| MSASCMD\_S13\_TC12\_ResolveRecipients\_To |
| MSASCMD\_S13\_TC13\_ResolveRecipients\_MergedFreeBusy |
| MSASCMD\_S13\_TC14\_ResolveRecipients\_NoCertificates |
| MSASCMD\_S13\_TC15\_ResolveRecipients\_MergedFreeBusyIntervalTime |
| MSASCMD\_S13\_TC16\_ResolveRecipients\_Picture\_Status173 |
| MSASCMD\_S13\_TC17\_ResolveRecipients\_Picture\_Success |
| MSASCMD\_S13\_TC18\_ResolveRecipients\_Picture\_Status174 |
| MSASCMD\_S13\_TC19\_ResolveRecipients\_Picture\_Status175 |
| MSASCMD\_S13\_TC20\_ResolveRecipients\_Status5 |
| MSASCMD\_S13\_TC21\_ResolveRecipients\_Status163 |
| S14\_Search | MSASCMD\_S14\_TC01\_Search\_Success |
| MSASCMD\_S14\_TC02\_Search\_MultipleAndElements |
| MSASCMD\_S14\_TC03\_Search\_TooComplex\_Status8\_Class |
| MSASCMD\_S14\_TC04\_Search\_TooComplex\_Status8\_CollectionId |
| MSASCMD\_S14\_TC05\_Search\_TooComplex\_Status8\_FreeText |
| MSASCMD\_S14\_TC06\_Search\_MultipleDeepTraversal |
| MSASCMD\_S14\_TC07\_Search\_MoreThanOneMIMESupport |
| MSASCMD\_S14\_TC08\_Search\_MultipleRange |
| MSASCMD\_S14\_TC09\_Search\_MoreThanOneRebuildResults |
| MSASCMD\_S14\_TC10\_Search\_MoreThanOneStores |
| MSASCMD\_S14\_TC11\_Search\_Invalid\_Status2 |
| MSASCMD\_S14\_TC12\_Search\_Result\_NoItemMatched |
| MSASCMD\_S14\_TC13\_Search\_Result\_MultipleMatchedItems |
| MSASCMD\_S14\_TC14\_Search\_GAL\_Success |
| MSASCMD\_S14\_TC15\_Search\_WithAllClass |
| MSASCMD\_S14\_TC16\_Search\_WithoutNoteAndSmsClass |
| MSASCMD\_S14\_TC17\_Search\_WithoutClass |
| MSASCMD\_S14\_TC18\_Search\_DocumentLibraryFolder |
| MSASCMD\_S14\_TC19\_Search\_DocumentLibraryItem |
| MSASCMD\_S14\_TC20\_Search\_GAL\_Status174 |
| MSASCMD\_S14\_TC21\_Search\_Status175 |
| MSASCMD\_S14\_TC22\_Search\_WithDeepTraversal |
| MSASCMD\_S14\_TC23\_Search\_WithoutDeepTraversal |
| MSASCMD\_S14\_TC24\_Search\_DocumentLibraryStatus5 |
| MSASCMD\_S14\_TC25\_Search\_DocumentLibraryStatus6 |
| MSASCMD\_S14\_TC26\_Search\_GAL\_Status173 |
| MSASCMD\_S14\_TC27\_Search\_TooComplex\_Status8\_ConversationId |
| MSASCMD\_S14\_TC28\_Search\_CaseInsensitiveMatch |
| MSASCMD\_S14\_TC29\_Search\_MoreThanOneConversationId |
| MSASCMD\_S14\_TC30\_Search\_TooComplex\_Status8\_Or |
| MSASCMD\_S14\_TC31\_Search\_TooComplex\_Status8\_GreaterThan |
| MSASCMD\_S14\_TC32\_Search\_TooComplex\_Status8\_LessThan |
| MSASCMD\_S14\_TC33\_Search\_MIMESupport |
| S15\_SendMail | MSASCMD\_S15\_TC01\_SendMail\_Success |
| MSASCMD\_S15\_TC02\_SendMail\_AccountIdInvalid |
| S16\_Settings | MSASCMD\_S16\_TC01\_Settings\_Oof\_Success |
| MSASCMD\_S16\_TC02\_Settings\_DevicePassword\_Status2 |
| MSASCMD\_S16\_TC03\_Settings\_OofState |
| MSASCMD\_S16\_TC04\_Settings\_UserInformation |
| MSASCMD\_S16\_TC05\_Settings\_OofSetToExternalKnown |
| MSASCMD\_S16\_TC06\_Settings\_OofSetToExternalUnknown |
| MSASCMD\_S16\_TC07\_Settings\_OofSetToInternal |
| MSASCMD\_S16\_TC08\_Settings\_OofSetBodyTypeIsRequiredWithReplyMessage |
| MSASCMD\_S16\_TC09\_Settings\_OofSetBodyTypeOptional |
| MSASCMD\_S16\_TC10\_Settings\_OofSetForThreeTypeAudiences |
| MSASCMD\_S16\_TC11\_Settings\_Status5 |
| MSASCMD\_S16\_TC12\_Settings\_Status6 |
| MSASCMD\_S16\_TC13\_Settings\_ResponseInOrder |
| MSASCMD\_S16\_TC14\_Settings\_GetRightManagementInformation |
| MSASCMD\_S16\_TC15\_Settings\_SetDeviceInformationPhoneNumber |
| MSASCMD\_S16\_TC16\_Settings\_DevicePassword\_LongPassword |
| S17\_SmartForward | MSASCMD\_S17\_TC01\_SmartForward\_Success |
| MSASCMD\_S17\_TC02\_SmartForward\_Fail |
| MSASCMD\_S17\_TC03\_SmartForwardAppointment |
| MSASCMD\_S17\_TC04\_SmartForwardWithInvalidInstanceId |
| MSASCMD\_S17\_TC05\_SmartForwardWithInstanceIdSuccess |
| MSASCMD\_S17\_TC06\_SmartForwardRecurringMeetingWithoutInstanceId |
| S18\_SmartReply | MSASCMD\_S18\_TC01\_SmartReply\_ContainOriginalMessage |
| MSASCMD\_S18\_TC02\_SmartReply\_Success |
| MSASCMD\_S18\_TC03\_SmartReply\_Status104 |
| S19\_Sync | MSASCMD\_S19\_TC01\_Sync\_Success |
| MSASCMD\_S19\_TC02\_Sync\_MoreAvailable |
| MSASCMD\_S19\_TC03\_Sync\_Email\_Sequence |
| MSASCMD\_S19\_TC04\_Sync\_Status3 |
| MSASCMD\_S19\_TC05\_Sync\_Status4 |
| MSASCMD\_S19\_TC06\_Sync\_Status13 |
| MSASCMD\_S19\_TC07\_Sync\_HeartbeatInterval\_Status14 |
| MSASCMD\_S19\_TC08\_Sync\_ConversationMode\_Status4 |
| MSASCMD\_S19\_TC09\_Sync\_Class\_Redefined |
| MSASCMD\_S19\_TC10\_Sync\_NoChildElementsForCommands |
| MSASCMD\_S19\_TC11\_Sync\_Email\_FilterType |
| MSASCMD\_S19\_TC12\_Sync\_Calendar\_FilterType |
| MSASCMD\_S19\_TC13\_Sync\_Tasks\_FilterType |
| MSASCMD\_S19\_TC14\_Sync\_MoreThanOneFilterTypes |
| MSASCMD\_S19\_TC15\_Sync\_IncludeFilterTypeOrNot |
| MSASCMD\_S19\_TC16\_Sync\_WindowSize\_MoreAvailable\_Omitted |
| MSASCMD\_S19\_TC17\_Sync\_WindowSize\_MoreAvailable |
| MSASCMD\_S19\_TC18\_Sync\_SMIME |
| MSASCMD\_S19\_TC19\_Sync\_NoResponsesElement |
| MSASCMD\_S19\_TC20\_Sync\_MoreThanOneClasses |
| MSASCMD\_S19\_TC21\_Sync\_MoreThanOneMaxItems |
| MSASCMD\_S19\_TC22\_Sync\_MoreThanOneMIMESupport |
| MSASCMD\_S19\_TC23\_Sync\_MoreThanOneMIMETruncation |
| MSASCMD\_S19\_TC24\_Sync\_Status6\_EndTimeMissing |
| MSASCMD\_S19\_TC25\_Sync\_Status6\_AddEmail |
| MSASCMD\_S19\_TC26\_Sync\_GetChanges |
| MSASCMD\_S19\_TC27\_Sync\_Change |
| MSASCMD\_S19\_TC28\_Sync\_Change\_InvalidServerId |
| MSASCMD\_S19\_TC29\_Sync\_HeartbeatIntervalAndWait\_StatusIs4 |
| MSASCMD\_S19\_TC30\_Sync\_Limit\_WaitIsLessThan1 |
| MSASCMD\_S19\_TC31\_Sync\_Limit\_WaitIsGreaterThan59 |
| MSASCMD\_S19\_TC32\_Sync\_Limit\_HeartbeatIntervalIsLessThan60 |
| MSASCMD\_S19\_TC33\_Sync\_Limit\_HeartbeatIntervalIsLargerThan3540 |
| MSASCMD\_S19\_TC34\_Sync\_DeletesAsMovesIsTrue |
| MSASCMD\_S19\_TC35\_Sync\_DeletesAsMovesIsFalse |
| MSASCMD\_S19\_TC36\_Sync\_Fetch |
| MSASCMD\_S19\_TC37\_Sync\_WindowSize\_512 |
| MSASCMD\_S19\_TC38\_Sync\_Delete |
| MSASCMD\_S19\_TC39\_Sync\_Supported |
| MSASCMD\_S19\_TC40\_Sync\_Conflict |
| MSASCMD\_S19\_TC41\_Sync\_MoreThanOneConflicts |
| MSASCMD\_S19\_TC42\_Sync\_WithCombinationClasses |
| MSASCMD\_S19\_TC43\_Sync\_MIMETruncation |
| MSASCMD\_S19\_TC44\_Sync\_MIMESupport |
| MSASCMD\_S19\_TC45\_Sync\_ConversationMode |
| MSASCMD\_S19\_TC46\_Sync\_Supported\_Status4 |
| MSASCMD\_S19\_TC47\_Sync\_Wait\_Status14 |
| MSASCMD\_S19\_TC48\_Sync\_Collection\_LimitValue |
| MSASCMD\_S19\_TC49\_Sync\_SoftDelete |
| MSASCMD\_S19\_TC50\_Sync\_Class |
| MSASCMD\_S19\_TC51\_Sync\_Change\_Exceptions |
| S20\_ValidateCert | MSASCMD\_S20\_TC01\_ValidateCert\_Success |
| MSASCMD\_S20\_TC02\_ValidateCert\_InvalidSignature |
| MSASCMD\_S20\_TC03\_ValidateCert\_LimitingSize |
| S21\_CommonStatusCode | MSASCMD\_S21\_TC01\_CommonStatusCode\_166 |
| MSASCMD\_S21\_TC02\_CommonStatusCode\_173 |
| MSASCMD\_S21\_TC03\_CommonStatusCode\_165 |
| MSASCMD\_S21\_TC04\_CommonStatusCode\_105 |
| MSASCMD\_S21\_TC05\_CommonStatusCode\_164 |
| MSASCMD\_S21\_TC06\_CommonStatusCode\_118 |

Test case scenarios distribution

### Test case description

The following table describes the common prerequisites and common cleanup for all the test cases.

|  |  |
| --- | --- |
| Common prerequisites | 1. Call TestInitialize method to initialize ActiveSync client service.  2. Call FolderSync command to synchronize the collection hierarchy. |
| Common Cleanup | 1. Call Sync command to delete the items which are created by test case.  2. Call FolderDelete command to delete the folders which are created by test case.  3. Call ChangeDeviceID and ChangeDeviceType to restore the DeviceID and DeiveceType which are changed by test case. |

Test case common prerequisites and common cleanup

The steps in the following test cases use methods and parameters in the adapter interfaces directly.

The following tables describe the traditional test cases.

|  |  |
| --- | --- |
| S01\_Autodiscover | |
| Test case ID | MSASCMD\_S01\_TC01\_Autodiscover\_TypeIsMobileSync |
| Description | This test case is used to verify if the Type element value is set to 'MobileSync', the Name element should be returned. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Autodiscover command to facilitate the discovery of the core account configuration information when the Type element value is set to "MobileSync". |
| Cleanup | Common Cleanup |

MSASCMD\_S01\_TC01\_Autodiscover\_TypeIsMobileSync

|  |  |
| --- | --- |
| S01\_Autodiscover | |
| Test case ID | MSASCMD\_S01\_TC02\_Autodiscover\_Fail |
| Description | This test case is used to verify if Autodiscover failed, the server should return an error child element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Autodiscover command to facilitate the discovery of core account configuration information with an invalid EmailAddress element of a provider. |
| Cleanup | Common Cleanup |

MSASCMD\_S01\_TC02\_Autodiscover\_Fail

|  |  |
| --- | --- |
| S02\_FolderCreate | |
| Test case ID | MSASCMD\_S02\_TC01\_FolderCreate\_Success |
| Description | This test case is used to verify if the FolderCreate command request is successful, ServerId element should be returned. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls FolderSync command to synchronize the new collection hierarchy. |
| Cleanup | Common Cleanup |

MSASCMD\_S02\_TC01\_FolderCreate\_Success

|  |  |
| --- | --- |
| S02\_FolderCreate | |
| Test case ID | MSASCMD\_S02\_TC02\_FolderCreate\_Fail |
| Description | This test case is used to verify if the FolderCreate command request fails, ServerId element should not be returned. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Call method FolderCreate command to create a new folder as a child folder of the specified parent folder without DisplayName element value. |
| Cleanup | Common Cleanup |

MSASCMD\_S02\_TC02\_FolderCreate\_Fail

|  |  |
| --- | --- |
| S02\_FolderCreate | |
| Test case ID | MSASCMD\_S02\_TC03\_FolderCreate\_Status2 |
| Description | This test case is used to verify FolderCreate command, if the folder name already exists, the status should be equal to 2. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls FolderCreate command to create another new folder with same folder name as a child folder of the same parent folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S02\_TC03\_FolderCreate\_Status2

|  |  |
| --- | --- |
| S02\_FolderCreate | |
| Test case ID | MSASCMD\_S02\_TC04\_FolderCreate\_Status5 |
| Description | This test case is used to verify FolderCreate command, if the parentId doesn't exist, the status should be equal to 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder which does not exist in the mailbox. |
| Cleanup | Common Cleanup |

MSASCMD\_S02\_TC04\_FolderCreate\_Status5

|  |  |
| --- | --- |
| S02\_FolderCreate | |
| Test case ID | MSASCMD\_S02\_TC05\_FolderCreate\_Status9 |
| Description | This test case is used to verify FolderCreate command, if the SyncKey is invalid, the status should be equal to 9. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder with an invalid folder SyncKey. |
| Cleanup | Common Cleanup |

MSASCMD\_S02\_TC05\_FolderCreate\_Status9

|  |  |
| --- | --- |
| S02\_FolderCreate | |
| Test case ID | MSASCMD\_S02\_TC06\_FolderCreate\_Status10 |
| Description | This test case is used to verify FolderCreate command, if the request contains a semantic error, the status should be equal to 10. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder without folder SyncKey. |
| Cleanup | Common Cleanup |

MSASCMD\_S02\_TC06\_FolderCreate\_Status10

|  |  |
| --- | --- |
| S02\_FolderCreate | |
| Test case ID | MSASCMD\_S02\_TC07\_FolderCreate\_Status3 |
| Description | This test case is used to verify FolderCreate command, if the specified parent folder is the recipient information cache, the status should be equal to 3. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the recipient information cache. |
| Cleanup | Common Cleanup |

MSASCMD\_S02\_TC07\_FolderCreate\_Status3

|  |  |
| --- | --- |
| S02\_FolderCreate | |
| Test case ID | MSASCMD\_S02\_TC08\_FolderCreate\_RecipientInformationCache\_Status3 |
| Description | This test case is used to verify FolderCreate command, if create a recipient information cache, the status should be equal to 3. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a recipient information cache. |
| Cleanup | Common Cleanup |

**MSASCMD\_S02\_TC08\_FolderCreate\_RecipientInformationCache\_Status3**

|  |  |
| --- | --- |
| S03\_FolderDelete | |
| Test case ID | MSASCMD\_S03\_TC01\_FolderDelete\_Success |
| Description | This test case is used to verify if the FolderDelete command is successful, the status should be equal to 1. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls FolderDelete command to delete the created folder in step 1 from the server.  3. The client calls FolderSync to check if the folder is deleted successfully. |
| Cleanup | Common Cleanup |

MSASCMD\_S03\_TC01\_FolderDelete\_Success

|  |  |
| --- | --- |
| S03\_FolderDelete | |
| Test case ID | MSASCMD\_S03\_TC02\_FolderDelete\_Status3 |
| Description | This test case is used to verify FolderDelete command, if the specified folder is a special system folder, the status in return value is 3. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderDelete command to delete the Calendar folder from the server. |
| Cleanup | Common Cleanup |

MSASCMD\_S03\_TC02\_FolderDelete\_Status3

|  |  |
| --- | --- |
| S03\_FolderDelete | |
| Test case ID | MSASCMD\_S03\_TC03\_FolderDelete\_Status4 |
| Description | This test case is used to verify FolderDelete command, if the specified folder does not exist, the status in return value is 4. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderDelete command to delete the folder that does not exist on the server. |
| Cleanup | Common Cleanup |

MSASCMD\_S03\_TC03\_FolderDelete\_Status4

|  |  |
| --- | --- |
| S03\_FolderDelete | |
| Test case ID | MSASCMD\_S03\_TC04\_FolderDelete\_Status9 |
| Description | This test case is used to verify FolderDelete command, if the SyncKey is an empty string, the status in return value is 9. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls FolderDelete command to delete a folder from the server and set SyncKey value to empty string.  3. The client calls FolderDelete command to delete the created folder in step 2 from the server. |
| Cleanup | Common Cleanup |

MSASCMD\_S03\_TC04\_FolderDelete\_Status9

|  |  |
| --- | --- |
| S03\_FolderDelete | |
| Test case ID | MSASCMD\_S03\_TC05\_FolderDelete\_Status10 |
| Description | This test case is used to verify FolderDelete command, if the request contains a semantic or syntactic error, the status in return value is 10. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls FolderDelete command to delete a folder from the server without folder SyncKey.  3. The client calls FolderDelete command to delete the created folder in step 2 from the server. |
| Cleanup | Common Cleanup |

MSASCMD\_S03\_TC05\_FolderDelete\_Status10

|  |  |
| --- | --- |
| S04\_FolderSync | |
| Test case ID | MSASCMD\_S04\_TC01\_FolderSync\_SyncKey0 |
| Description | This test case is used to verify FolderSync command, all folders MUST be returned to the client when initial folder synchronization is done with a synchronization key of 0(zero). |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderSync command to synchronize the collection hierarchy with synkey value set to “0”. |
| Cleanup | Common Cleanup |

MSASCMD\_S04\_TC01\_FolderSync\_SyncKey0

|  |  |
| --- | --- |
| S04\_FolderSync | |
| Test case ID | MSASCMD\_S04\_TC02\_FolderSync\_NoChanged |
| Description | This test case is used to verify FolderSync command, if there are no changes since the last folders synchronization, a Count element value of 0 (zero) is returned. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderSync command to synchronize the collection hierarchy if no changes occurred for folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S04\_TC02\_FolderSync\_NoChanged

|  |  |
| --- | --- |
| S04\_FolderSync | |
| Test case ID | MSASCMD\_S04\_TC03\_FolderSync\_Changed |
| Description | This test case is used to verify FolderSync command, if any changes have occurred on the server, the count is not equal to 0. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ChangeDeviceID to add a new device id.  2. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  3. The client calls ChangeDeviceID to change the device id to the original device id.  4. The client calls FolderSync command to synchronize the collection hierarchy. |
| Cleanup | Common Cleanup |

MSASCMD\_S04\_TC03\_FolderSync\_Changed

|  |  |
| --- | --- |
| S04\_FolderSync | |
| Test case ID | MSASCMD\_S04\_TC04\_FolderSync\_Resynchronizes |
| Description | This test case is used to verify FolderSync command, if client resynchronizes the existing folder hierarchy, ServerId values do not change. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls FolderSync command to synchronize the collection hierarchy. |
| Cleanup | Common Cleanup |

MSASCMD\_S04\_TC04\_FolderSync\_Resynchronizes

|  |  |
| --- | --- |
| S04\_FolderSync | |
| Test case ID | MSASCMD\_S04\_TC05\_FolderSync\_Status9 |
| Description | This test case is used to verify FolderSync command, if the SyncKey is an empty string, the status is equal to 9. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderSync command to synchronize the collection hierarchy with an empty string of SyncKey. |
| Cleanup | Common Cleanup |

MSASCMD\_S04\_TC05\_FolderSync\_Status9

|  |  |
| --- | --- |
| S04\_FolderSync | |
| Test case ID | MSASCMD\_S04\_TC06\_FolderSync\_Status10 |
| Description | This test case is used to verify FolderSync command, if the SyncKey is invalid, the status is equal to 10. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderSync command to synchronize the collection hierarchy with an invalid SyncKey value. |
| Cleanup | Common Cleanup |

MSASCMD\_S04\_TC06\_FolderSync\_Status10

|  |  |
| --- | --- |
| S04\_FolderSync | |
| Test case ID | MSASCMD\_S04\_TC07\_FolderSync\_AddFolder |
| Description | This test case is used to verify FolderSync command synchronizes the folder hierarchy successfully after adding a folder. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls FolderSync command to synchronize the collection hierarchy. |
| Cleanup | Common Cleanup |

MSASCMD\_S04\_TC07\_FolderSync\_AddFolder

|  |  |
| --- | --- |
| S04\_FolderSync | |
| Test case ID | MSASCMD\_S04\_TC08\_FolderSync\_UpdateFolder |
| Description | This test case is used to verify FolderSync command synchronizes the updated folder successfully. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls ChangeDevideID to add a new device id.  3. The client calls FolderSync command to synchronize the collection hierarchy with new device id, return latest folder SyncKey.  4 The client calls FolderUpdate command to rename the folder created in step 1.  5. The client calls ChangeDeviceID to chage the device id to original one.  6. The client calls FolderSync command to synchronize the collection hierarchy with original device id. |
| Cleanup | Common Cleanup |

MSASCMD\_S04\_TC08\_FolderSync\_UpdateFolder

|  |  |
| --- | --- |
| S04\_FolderSync | |
| Test case ID | MSASCMD\_S04\_TC09\_FolderSync\_DeleteFolder |
| Description | This test case is used to verify FolderSync command synchronizes the folder hierarchy successfully after deleting a folder. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls ChangeDevideID to add a new device id.  3. The client calls FolderSync command to synchronize the collection hierarchy with new device id, return latest folder SyncKey.  4. The client calls FolderDelete command to delete the created folder in step 2 from the server.  5. The client calls ChangeDeviceID to chage the device id to original one.  6. The client calls FolderSync command to synchronize the collection hierarchy with original device id. |
| Cleanup | Common Cleanup |

MSASCMD\_S04\_TC09\_FolderSync\_DeleteFolder

|  |  |
| --- | --- |
| S04\_FolderSync | |
| Test case ID | MSASCMD\_S04\_TC10\_FolderSync\_NoSynchronizeItems |
| Description | This test case is used to verify FolderSync command synchronizes the folder hierarchy successfully, but does not synchronize the items in the collections themselves. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted email to User1.  3. The client calls SwitchUser to switch User2 to User1.  4.The client calls FolderSync command to synchronize the collection hierarchy without synchronizing the items in the collections. |
| Cleanup | Common Cleanup |

**MSASCMD\_S04\_TC10\_FolderSync\_NoSynchronizeItems**

|  |  |
| --- | --- |
| S05\_FolderUpdate | |
| Test case ID | MSASCMD\_S05\_TC01\_FolderUpdate\_Success |
| Description | This test case is used to verify if the FolderUpdate command request was successful, the status should be equal to 1. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls FolderUpdate command to rename a folder, return latest folder SyncKey.  3. The client calls FolderSync command to synchronize the collection hierarchy. |
| Cleanup | Common Cleanup |

MSASCMD\_S05\_TC01\_FolderUpdate\_Success

|  |  |
| --- | --- |
| S05\_FolderUpdate | |
| Test case ID | MSASCMD\_S05\_TC02\_FolderUpdate\_Status2 |
| Description | This test case is used to verify FolderUpdate command, if specified folder is a special folder, the status in return value is 2. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderUpdate command to rename the Calendar folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S05\_TC02\_FolderUpdate\_Status2

|  |  |
| --- | --- |
| S05\_FolderUpdate | |
| Test case ID | MSASCMD\_S05\_TC03\_FolderUpdate\_Status4 |
| Description | This test case is used to verify FolderUpdate command, if specified folder does not exist, the status in return value is 4. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderUpdate command to rename a non existent folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S05\_TC03\_FolderUpdate\_Status4

|  |  |
| --- | --- |
| S05\_FolderUpdate | |
| Test case ID | MSASCMD\_S05\_TC04\_FolderUpdate\_Status5 |
| Description | This test case is used to verify FolderUpdate command, if specified parent folder does not exist, the status in return value is 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls FolderUpdate command to move the created folder in step 1 from Root folder to a folder that does not exist on the server. |
| Cleanup | Common Cleanup |

MSASCMD\_S05\_TC04\_FolderUpdate\_Status5

|  |  |
| --- | --- |
| S05\_FolderUpdate | |
| Test case ID | MSASCMD\_S05\_TC05\_FolderUpdate\_Status9 |
| Description | This test case is used to verify FolderUpdate command, if the request contains invalid synchronization key, the status in return value is 9. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls FolderUpdate command to move a folder from one location to another on the server with an empty value of SyncKey. |
| Cleanup | Common Cleanup |

MSASCMD\_S05\_TC05\_FolderUpdate\_Status9

|  |  |
| --- | --- |
| S05\_FolderUpdate | |
| Test case ID | MSASCMD\_S05\_TC06\_FolderUpdate\_Status10 |
| Description | This test case is used to verify FolderUpdate command, if the request contains a semantic error, the status in return value is 10. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder, return ServerId for FolderCreate command.  2. The client calls FolderUpdate command to rename a folder from one location to another on the server without SyncKey element. |
| Cleanup | Common Cleanup |

MSASCMD\_S05\_TC06\_FolderUpdate\_Status10

|  |  |
| --- | --- |
| S05\_FolderUpdate | |
| Test case ID | MSASCMD\_S05\_TC07\_FolderUpdate\_Moves |
| Description | This test case is used to verify FolderUpdate command, if moves the folder failed, the status in return value is 2. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the mailbox Root folder, return ServerId for FolderCreate command.  2. The client calls FolderUpdate command to move the new created folder from the mailbox Root folder to SentItems folder on the server.  3. The client calls FolderSync command to synchronize the collection hierarchy.  4. The client calls FolderCreate command to create a new folder with same folder name created in step 2 as a child folder of the specified parent folder, return ServerId for FolderCreate command.  5. The client calls FolderUpdate command to move a folder from one location to another on the server. |
| Cleanup | Common Cleanup |

MSASCMD\_S05\_TC07\_FolderUpdate\_Moves

|  |  |
| --- | --- |
| S05\_FolderUpdate | |
| Test case ID | MSASCMD\_S05\_TC08\_FolderUpdate\_Status3 |
| Description | This test case is used to verify FolderUpdate command, if specified folder is the recipient information cache, the status in return value is 3. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderUpdate command to update the recipient information cache. |
| Cleanup | Common Cleanup |

MSASCMD\_S05\_TC08\_FolderUpdate\_Status3

|  |  |
| --- | --- |
| S06\_GetAttachment | |
| Test case ID | MSASCMD\_S06\_TC01\_GetAttachment\_Success |
| Description | This test case is used to verify the requirements related to a successful GetAttachment command. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail to send an email with a normal attachment.  2. The client calls SwitchUser to switch the current user to User 2.  3. The client calls Sync command to synchronize changes in Inbox folder.  4. The client calls GetAttachment command to fetch attachment. |
| Cleanup | Common Cleanup |

MSASCMD\_S06\_TC01\_GetAttachment\_Success

|  |  |
| --- | --- |
| S06\_GetAttachment | |
| Test case ID | MSASCMD\_S06\_TC02\_GetAttachment\_Status500 |
| Description | This test case is used to verify if the GetAttachment command is used to retrieve an attachment that has been deleted on the server, a 500 status code is returned. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail to send an email with a normal attachment.  2. The client calls SwitchUser to switch the current user to User 2.  3. The client calls Sync command to synchronize changes in Inbox folder.  4. The client calls GetAttachment command to fetch the attachment in step 1.  5. The client calls Sync command to delete the email with normal attachment in the Inbox folder.  6. The client calls GetAttachment command to retrieve the attatchment that has been deleted from the server. |
| Cleanup | Common Cleanup |

MSASCMD\_S06\_TC02\_GetAttachment\_Status500

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC01\_GetItemEstimate\_Success |
| Description | This test case is used to verify the requirements related to a successful GetItemEstimate command. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail to send MIME-formatted email to User2.  2. The client calls SwtichUser to switch the current user to User2.  3. The client calls Sync command to synchronize the changes in Inbox folder.  4. The client calls Sync command to get the initial SyncKey.  5. The client calls GetItemEstimate command to get an estimate of the number of items in the Inbox folder on the server. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC01\_GetItemEstimate\_Success

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC02\_GetItemEstimate\_WithoutOptions |
| Description | This test case is used to verify GetItemEstimate command, if Options element is not included in a request, server will enumerate all of the items within the collection, without any filter. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls GetItemEstimate command with filters within the airsync:Options container to Email class to get an estimate of the number of the items in the Inbox folder.  3. The client calls GetItemEstimate command without Options element to get an estimate of the number of all items in the Inbox folder on the server that have to be synchronized. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC02\_GetItemEstimate\_WithoutOptions

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC03\_GetItemEstimate\_Contacts\_FilterType |
| Description | This test case is used to verify GetItemEstimate command for a contact collection, if FilterType element is included, no error is returned. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls GetItemEstimate command with FilterType element to get an estimate of the number of items in the Contacts folder on the server that have to be synchronized. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC03\_GetItemEstimate\_Contacts

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC04\_GetItemEstimate\_Email\_FilterType |
| Description | This test case is used to verify GetItemEstimate command for e-mail, the status should be correspond to the value of FilterType. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SenMail command to send an email from User1 to User2.  2. The client calls SwitchUser to swith User1 to User2.  3. The client calls CheckMail with Sync command to check the email in Inbox folder.  4. The client calls Sync command to get an initial SyncKey.  5. The client calls GetItemEstimate command with FilterType setting to “0” to get an estimate of the number of all items in Inbox folder on the server that have to be synchronized.  6. The client calls GetItemEstimate command without Filtertype.  7. The client calls GetItemEstimate command with FilterType setting to “1” to get estimate number of the items within 1 day in Inbox folder on the server that have to be synchronized.  8. The client calls SwitchUser to switch User2 to User1.  9. The client calls SendMail to send an email from User 1 to User2.  10. The client calls SwitchUser to switch User 1 to User2.  11. The client calls GetItemEstimate command with FilterType setting to “1” to get estimate number of the items within 1 day in Inbox folder on the server that have to be synchronized.  12. The client calls GetItemEstimate command with FilterType setting to “2” to get estimate number of the items within 3 days in Inbox folder on the server that have to be synchronized.  13. The client calls GetItemEstimate command with FilterType setting to “3” to get estimate number of the items within 1 week in Inbox folder on the server that have to be synchronized.  14. The client calls GetItemEstimate command with FilterType setting to “4” to get estimate number of the items within 2 weeks in Inbox folder on the server that have to be synchronized.  15. The client calls GetItemEstimate command with FilterType setting to “5” to get estimate number of the items within 1 month in Inbox folder on the server that have to be synchronized.  16. The client calls GetItemEstimate command with FilterType setting to “6” to get an estimate of the number of emails within 3 months in Inbox folder on the server that have to be synchronized.  17. The client calls GetItemEstimate command with FilterType setting to “7” to get an estimate of the number of emails within 6 months in Inbox folder on the server that have to be synchronized.  18. The client calls GetItemEstimate command with FilterType setting to “8” to get an estimate of the number of all incomplete tasks in Inbox folder on the server that have to be synchronized. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC04\_GetItemEstimate\_Email\_FilterType

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC05\_GetItemEstimate\_Calendar\_FilterType |
| Description | This test case is used to verify GetItemEstimate command for calendar, the value of status should be correspond to the value of FilterType. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls GetItemEstimate command with FilterType setting to “0” to get an estimate of the number of all calendar items in Calendar folder on the server that have to be synchronized.  3. The client calls GetItemEstimate command with FilterType setting to “1” to get an estimate of the number of calendar items within 1 day in Calendar folder on the server that have to be synchronized.  4. The client calls GetItemEstimate command with FilterType setting to “2” to get an estimate of the number of calendar items within 3 days in Calendar folder on the server that have to be synchronized.  5. The client calls GetItemEstimate command with FilterType setting to “3” to get an estimate of the number of calendar items within 1 week in Calendar folder on the server that have to be synchronized.  6. The client calls GetItemEstimate command with FilterType setting to “4” to get an estimate of the number of calendar items within 2 weeks in Calendar folder on the server that have to be synchronized.  7. The client calls Sync command to add a future Calendar item.  8. The client calls Sync command without FilterType to get the Calendar item.  9. The client calls SyncChanges with Sync command to synchronize the changes.  10. The client calls Sync command with FilterType setting to “4” to get the Calendar item.  11. The client calls Sync command to add a recurring Calendar item without EndTime.  12. The client calls Sync command without FilterType to get the Calendar item.  13. The client calls SyncChanges with Sync command to synchronize the changes.  14. The client calls Sync command with FilterType setting to “4” to get the Calendar item.  15. The client calls GetItemEstimate command with FilterType setting to “5” to get an estimate of the number of calendar items within 1 month in Calendar folder on the server that have to be synchronized.  16. The client calls GetItemEstimate command with FilterType setting to “6” to get an estimate of the number of calendar items within 3 months in Calendar folder on the server that have to be synchronized.  17. The client calls GetItemEstimate command with FilterType setting to “7” to get an estimate of the number of calendar items within 6 months in Calendar folder on the server that have to be synchronized.  18. The client calls GetItemEstimate command with FilterType setting to “8” to get an estimate of the number of all incomplete tasks in Calendar folder on the server that have to be synchronized. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC05\_GetItemEstimate\_Calendar\_FilterType

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC06\_GetItemEstimate\_Tasks\_FilterType |
| Description | This test case is used to verify GetItemEstimate command for tasks, the value of status should be correspond to the value of FilterType. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls GetItemEstimate command with FilterType setting to “0” to get an estimate of the number of all task items in Tasks folder on the server that have to be synchronized.  3. The client calls GetItemEstimate command with FilterType setting to “1” to get an estimate of the number of task items within 1 day in Tasks folder on the server that have to be synchronized.  4. The client calls GetItemEstimate command with FilterType setting to “2” to get an estimate of the number of task items within 3 days in Tasks folder on the server that have to be synchronized.  5. The client calls GetItemEstimate command with FilterType setting to “3” to get an estimate of the number of task items within 1 week in Tasks folder on the server that have to be synchronized.  6. The client calls GetItemEstimate command with FilterType setting to “4’ to get an estimate of the number of task items within 2 weeks in Tasks folder on the server that have to be synchronized.  7. The client calls GetItemEstimate command with FilterType setting to “5” to get an estimate of the number of task items within 1 month in Tasks folder on the server that have to be synchronized.  8. The client calls GetItemEstimate command with FilterType setting to “6” to get an estimate of the number of task items within 3 months in Tasks folder on the server that have to be synchronized.  9. The client calls GetItemEstimate command with FilterType setting to “7” to get an estimate of the number of task items within 6 months in Tasks folder on the server that have to be synchronized.  10. The client calls GetItemEstimate command with FilterType setting to “8” to get an estimate of the number of all incomplete tasks in Tasks folder on the server that have to be synchronized. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC06\_GetItemEstimate\_Tasks\_FilterType

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC07\_GetItemEstimate\_InvalidFilterType |
| Description | This test case is used to verify GetItemEstimate command, if FilterType element is invalid, the status should be equal to 103. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls GetItemEstimate command with FilterType setting to “9” to get an estimate of the number of items in Contacts folder on the server that have to be synchronized.  3. The client calls GetItemEstimate command with FilterType setting to “9” to get an estimate of the number of items in Inbox folder on the server that have to be synchronized.  4. The client calls GetItemEstimate command with FilterType setting to “9” to get an estimate of the number of items in Calendar folder on the server that have to be synchronized.  5. The client calls GetItemEstimate command with FilterType setting to “9” to get an estimate of the number of items in Tasks folder on the server that have to be synchronized. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC07\_GetItemEstimate\_InvalidFilterType

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC08\_GetItemEstimate\_MoreThanOneMaxItems |
| Description | This test case is used to verify GetItemEstimate command, if the request includes more than one MaxItem element, the server doesn't return error. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls GetItemEstimate command with two MaxItem elements to get an estimate of the number of items in recipient information cache folder on the server that have to be synchronized. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC08\_GetItemEstimate\_MoreThanOneMaxItems

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC09\_GetItemEstimate\_Status2 |
| Description | This test case is used to verify GetItemEstimate command, if the specified collection id is invalid, server should return a status code 2. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls GetItemEstimate command with two collection ids and one of the ids is an invalid Collection id, to get an estimate of the number of items in Contacts folder on the server that have to be synchronized. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC09\_GetItemEstimate\_Status2

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC10\_GetItemEstimate\_Status3 |
| Description | This test case is used to verify GetItemEstimate command, if the synchronization state has not been primed, server should return a status code 3. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls GetItemEstimate command without issuing a Sync command firstly to get an estimate of the number of items in a collection or folder on the server. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC10\_GetItemEstimate\_Status3

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC11\_GetItemEstimate\_Status4\_InvalidSyncKey |
| Description | This test case is used to verify GetItemEstimate command, if the synchronization key is invalid, server should return a status code 4. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  3. The client calls GetItemEstimate command with an inalid SyncKey to get an estimate of the number of items in Inbox folder on the server that have to be synchronized. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC11\_GetItemEstimate\_Status4\_InvalidSyncKey

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC12\_GetItemEstimate\_Status4\_MismatchedSyncKey |
| Description | This test case is used to verify GetItemEstimate command, if the SyncKey value provided in the GetItemEstimate request does not match those expected within the next Sync command request, server should return a status value of 4. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey in Contacts folder.  2. The client calls Sync command to get an initial SyncKey in Inbox folder.  3. The client calls GetItemEstimate command with the SyncKey returned in step 2 to get an estimate of the number of items in Inbox folder on the server. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC12\_GetItemEstimate\_Status4\_MismatchedSyncKey

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC13\_GetItemEstimate\_ConversationMode\_Status103 |
| Description | This test case is used to verify GetItemEstimate command, if the ConversationMode element for collections that do not store e-mail results in an invalid XML error, server should return a status value of 103. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls GetItemEstimate command with ConversationMode element to get an estimate of the number of items for collection that does not store email. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC13\_GetItemEstimate\_ConversationMode\_Status103

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC14\_GetItemEstimate\_WithCombinationClasses |
| Description | This test case is used to verify only SMS messages and email messages can be synchronized at the same time. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls GetItemEstimate command with a combination of “Email” and “SMS” classes and verify the response status code is 1. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC14\_GetItemEstimate\_WithCombinationClasses

|  |  |
| --- | --- |
| S07\_GetItemEstimate | |
| Test case ID | MSASCMD\_S07\_TC15\_GetItemEstimate\_Collection\_LimitValue |
| Description | This test case is used to verify the limit value of Collection element of GetItemEstimate command. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send a plain email with the information of email sender, receiver and the content.  2. The client calls SwtichUser to switch User1 to User2.  3. The client calls Sync command to get an initial SyncKey.  4. The client calls GetItemEstimate command with 1001 objects of GetItemEstimateCollection type. |
| Cleanup | Common Cleanup |

MSASCMD\_S07\_TC15\_GetItemEstimate\_Collection\_LimitValue

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC01\_ItemOperations\_DeleteSubFolders |
| Description | This test case is used to test when the DeleteSubFolders element is not present in ItemOperations request, the server will not delete subfolders. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder.  2. The client calls FolderSync command to synchronize the collection hierarchy and get the CollcectionId of the created subfolder.  3. The client calls ItemOperations command without DeleteSubFolder element to empty the specified folder.  4. The client calls FolderSync command to synchronize the collection hierarchy and check whether the subfolder exists.  5. The client calls ItemOperations command with DeleteSubFolder element.  6. The client calls FolderSync command to synchronize the collection hierarchy and check whether the subfolder exists. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC01\_ItemOperations\_DeleteSubFolders

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC02\_ItemOperations\_Status2 |
| Description | This test case is used to test when the client specifies an option that is invalid for the parent element in ItemOperations request, the server returns a Status element value of 2. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ItemOperations command which contains the Fetch element, with an invalid option in Options element. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC02\_ItemOperations\_Status2

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC03\_ItemOperations\_Status9 |
| Description | This test case is used to test when the Store element is unknown or unsupported in ItemOperations command request, the server returns a Status element value of 9. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ItemOperations command which contains the Fetch element, with an unsupported value of Store element. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC03\_ItemOperations\_Status9

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC04\_ItemOperations\_Status6 |
| Description | This test case is used to test when the object was not found in ItemOperations request, the server returns a Status element value of 6. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ItemOperations command which contains the Fetch element, with an invalid ServerID. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC04\_ItemOperations\_Status6

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC05\_ItemOperations\_Fetch |
| Description | This test case is used to test Fetch element in ItemOperations command. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted email messages to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User1 calls FolderSync command to synchronize the collection hierarchy.  5. The client using User1 calls Sync command to synchronize changes in Inbox folder between the client and the server.  6. The client using User1 calls ItemOperations command to act as a container for the Fetch element to fech the email item. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC05\_ItemOperations\_Fetch

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC06\_ItemOperations\_Move |
| Description | This test case is used to test Move element in ItemOperations method. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted email messages to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User1 calls FolderSync command to synchronize the collection hierarchy.  5. The client using User1 calls Sync command to synchronize changes in Inbox folder between the client and the server.  6. The client using User1 calls ItemOperations command to act as a container for the Move element to move the received email items to the Deleted Items folder.  7. The client using User1 calls Sync command to synchronize the changes in Deleted Items folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC06\_ItemOperations\_Move

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC07\_ItemOperations\_MultiplePartResponse |
| Description | This test case is used to verify fetching an email item with a multipart response. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls **S**endMail command to send two MIME-formatted email messages to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User1 calls FolderSync command to synchronize the collection hierarchy.  5. The client using User1 calls Sync command to synchronize changes in Inbox folder between the client and the server.  6. The client using User1 calls ItemOperations command to act as a container for the Fetch element to fetch the messages of specified folder with AcceptMultipartResponse in the request header and without Range element. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC07\_ItemOperations\_MultiplePartResponse

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC08\_ItemOperations\_Fetch\_MIMESupport |
| Description | This test case is used to verify if the MIMESupport option is included in the request, the server will return MIME for S/MIME-only messages, all messages, or no messages. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted email messages to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User1 calls FolderSync command to synchronize the collection hierarchy.  5. The client calls Sync command to synchronize the changes of Inbox folder between the client and the server.  6. The client calls ItemOperations command to act as a container for the Fetch element to fetch the specified folder with MIMESupport element setting to “0”.  7. The client calls ItemOperations command to act as a container for the Fetch element to fetch the specified folder with MIMESupport element setting to “1”  8. The client calls ItemOperations command to act as a container for the Fetch element to fetch the specified folder with inline method and MIMESupport element setting to “2”.  9. The client calls ItemOperations command without specifying delivery method and with MIMESupport setting to “2”.  10. The client calls ItemOperations command to act as a container for the Fetch element to fetch the specified folder with Multipart method and MIMESupport element setting to “2” |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC08\_ItemOperations\_Fetch\_MIMESupport

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC09\_ItemOperations\_RecipientInformationCache |
| Description | This test case is used to verify the ItemOperations command cannot perform operations on items in the recipient information cache. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ItemOperations command with the EmptyFolderContents element on recipient information cache, and check the Status element. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC09\_ItemOperations\_RecipientInformationCache

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC10\_ItemOperations\_EmptyFolderContents |
| Description | This test case is used to test EmptyFolderContents element in ItemOperations command. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted email messages to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User1 calls FolderSync command to synchronize the collection hierarchy.  5. The client calls Sync command to synchronize the changes of Inbox folder between the client and the server.  6. The client calls ItemOperations command with EmptyFolderContents element.  7. The client calls Sync command to synchronize the changes in Inbox folder between the client and the server. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC10\_ItemOperations\_EmptyFolderContents

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC11\_ItemOperations\_Fetch\_DocumentLibrary |
| Description | This test case is used to verify fetching the specified document library. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ItemOperations command to get data of a shared document with inline method.  2. The client calls ItemOperations command to get data of a shared document with Multipart method.  3. The client calls ItemOperations command to get data of a shared document with inline method and Range element specified as “0-0” to fetch the first byte. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC11\_ItemOperations\_Fetch\_DocumentLibrary

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC12\_ItemOperations\_Status4 |
| Description | This test case is used to verify the status 4 of ItemOperations command indicated the specified URI of document library is bad. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ItemOperations command to get data of an invalid shared document with inline fetch. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC12\_ItemOperations\_Status4

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC13\_ItemOperations\_Status5 |
| Description | This test case is used to verify the status 5 of ItemOperations command indicated access denied. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ItemOperations command to get data of a shared document by a user with an invalid password. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC13\_ItemOperations\_Status5

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC14\_ItemOperations\_Status8 |
| Description | This test case is used to verify the status 8 of ItemOperations command indicated the byte-range is too large. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ItemOperations command to get data of a shared document with the byte range exceeded the actual length of document contents. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC14\_ItemOperations\_Status8

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC15\_ItemOperations\_Status10 |
| Description | This test case is used to verify the status 10 of ItemOperations command indicated the file is empty. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ItemOperations command to get data of an empty shared document. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC15\_ItemOperations\_Status10

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC16\_ItemOperations\_Status11 |
| Description | This test case is used to verify the status 11 of ItemOperations command indicated the requested data size is too large. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ItemOperations command to get data of a shared document with a too large value in Range element exceeded the maximum range value. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC16\_ItemOperations\_Status11

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC17\_ItemOperations\_Status15 |
| Description | This test case is used to verify the status 15 of ItemOperations command indicated attachment ID is invalid. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ItemOperations command to get data of an attachment with invalid attachment ID. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC17\_ItemOperations\_Status15

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC18\_ItemOperations\_MoreThanOneFileReference\_Status2 |
| Description | This test case is used to verify the status 2 of ItemOperations command when the client specifies an option that is invalid for the parent element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send an email message with normal attachment.  2. The client calls Sync command to synchronize the changes in Inbox folder between the client and the server.  3. The client calls ItemOperations command to get data of an attachment with two FileReference elements. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC18\_ItemOperations\_MoreThanOneFileReference\_Status2

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC19\_ItemOperations\_Status156 |
| Description | This test case is used to verify the status 156 of ItemOperations command indicated the destination folder must be of type IPF.Note. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted email messages to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User1 calls FolderSync command to synchronize the collection hierarchy.  5. The client calls Sync command to synchronize the changes of Inbox folder between the client and the server.  6. The client calls ItemOperations command with Move element. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC19\_ItemOperations\_Status156

|  |  |
| --- | --- |
| S08\_ItemOperations | |
| Test case ID | MSASCMD\_S08\_TC20\_ItemOperations\_Status16 |
| Description | This test case is used to verify the status 16 of ItemOperations command indicated the access to the resource is denied. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1.The Client calls ItemOperations command to access Recipient Information cache. |
| Cleanup | Common Cleanup |

MSASCMD\_S08\_TC20\_ItemOperations\_Status16

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC01\_MeetingResponse\_AcceptMeeting |
| Description | This test case is used to verify CalendarId is returned, when the meeting request is accepted. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to get the meeting request email.  4. The client using User2 calls MeetingResponse command to accept the meeting request in the user's Inbox folder.  5. The client using User2 calls Sync command to sync Calendar folder change and get accepted calendar item's serverId. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC01\_MeetingResponse\_AcceptMeeting

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC02\_MeetingResponse\_DeclineMeeting |
| Description | This test case is used to verify the response that has no CalendarId, when meeting response is declined. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client calls GetMailItem in ConfirmItemInFolder with Sync command and FolderSync command to synchronize the folder hierarchy.  4. The client using User 2 calls MeetingResponse command to decline the meeting request in the user's Inbox folder.  5. The client calls SyncChanges with Sync command to synchronize the changes. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC02\_MeetingResponse\_DeclineMeeting

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC03\_MeetingResponse\_InvalidMeeting |
| Description | This test case is used to verify returned status value is 2, when userResponse is invalid. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client calls GetMailItem in ConfirmItemInFolder with Sync command and FolderSync command to synchronize the folder hierarchy.  4. The client using User2 calls MeetingResponse command to accept the meeting request in the user's Inbox folder with invalid userResponse value. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC03\_MeetingResponse\_InvalidMeeting

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC04\_MeetingResponse\_RequestID |
| Description | This test case is used to verify RequestId is present in MeetingResponse command response if it was present in the corresponding MeetingResponse command request. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3.  The client calls GetMailItem in ConfirmItemInFolder with Sync command and FolderSync command to synchronize the folder hierarchy.  4. The client using User2 calls MeetingResponse command with RequestId to accept the meeting request in the user's Inbox folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC04\_MeetingResponse\_RequestID

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC05\_MeetingResponse\_ResponseNonMeeting |
| Description | This test case is used to verify the MeetingResponse command response has status equals 2, when the request is referencing an item other than a meeting request, e-mail or calendar item. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to get the meeting request email.  4. The client using User2 calls MeetingResponse command to tentatively accept the meeting request with invalid RequestId. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC05\_MeetingResponse\_ResponseNonMeeting

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC06\_MeetingResponse\_NonExistInstanceId |
| Description | This test case is used to verify if the InstanceId is not a specified meeting request, server should return the status value is 2. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client calls GetMailItem in ConfirmItemInFolder with Sync command and FolderSync command to synchronize the folder hierarchy.  4. The client using User2 calls MeetingResponse command to decline the meeting request in Inbox folder with invalid InstanceId. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC06\_MeetingResponse\_NonExistInstanceId

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC07\_MeetingResponse\_Status103 |
| Description | This test case is used to verify if there are more than 100 Request elements listed in the MeetingResponse command request, the server will return status 103. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client calls GetMailItem in ConfirmItemInFolder with Sync command and FolderSync command to synchronize the folder hierarchy.  4. The client calls MeetingResponse command with 101 Request elements as the child of the MeetingResponse element. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC07\_MeetingResponse\_Status103

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC08\_MeetingResponse\_RecurringMeetingInstanceIDInvalid |
| Description | This test case is used to verify if the InstanceId element specifies an email meeting request item, the server returns status code 2. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one recurring meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to get the meeting request email.  4. The client calls MeetingResponse command with an InstanceId element which specifies an email meeting request item. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC08\_MeetingResponse\_RecurringMeetingInstanceIDInvalid

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC09\_MeetingResponse\_Status146 |
| Description | This test case is used to verify if the InstanceId element value specifies a non-recurring meeting, the server responds with a Status element value of 146. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one single meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to get the meeting request email.  4. The client calls MeetingResponse command with an InstanceId element which specifies a non-recurring meeting. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC09\_MeetingResponse\_Status146

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC10\_MeetingResponse\_TentativeAcceptMeeting |
| Description | This test is used to verify implementation does use the MeetingResponse command to tentatively accept a meeting request in the user's Inbox folder |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls GetMailItem with Sync command and FolderSync command to sync Inbox folder and get the meeting request email.  4. The client using User2 calls GetMailItem with Sync command and FolderSync command to sync Calendar folder.  5. The client calls MeetingResponse command to tentatively accept the meeting in Inbox folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC10\_MeetingResponse\_TentativeAcceptMeeting

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC11\_MeetingResponse\_AcceptMeetingInCalendar |
| Description | This test is used to verify implementation does use the MeetingResponse command to accept a meeting request in the user's Calendar folder. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls GetMailItem with Sync command to get the meeting request email.  4. The client calls GetMailItem in ConfirmItemInFolder with Sync command and FolderSync command to synchronize the folder hierarchy.  5. The client calls MeetingResponse command accept the meeting request in Calendar folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC11\_MeetingResponse\_AcceptMeetingInCalendar

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC12\_MeetingResponse\_Status104 |
| Description | This test case is used to verify if the InstanceId element value specified is not in the proper format, the server responds with a Status value of 104. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one recurring meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to get the meeting request email.  4. The client calls MeetingResponse command with an InstanceId of wrong format. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC12\_MeetingResponse\_Status104

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC13\_MeetingResponse\_SubstituteMeetingInvitationEmail |
| Description | This test case is used to verify that server sends a substitute meeting invitation email. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one meeting request to User7.  2. The client calls SwitchUser to switch User1 to User7.  3. The client using User7 calls Sync command to synchronize the changes of Inbox folder to get the new added meeting request email.  4. The client using User7 calls Sync command to synchronize the changes of Calendar folder.  5. The client calls SwitchUser to switch User7 to User8.  6. The client using User8 calls GetSubstitudeMailItem with FolderSync command to synchronize the changes of Inbox folder to get the substitute invitation email. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC13\_MeetingResponse\_SubstituteMeetingInvitationEmail

|  |  |
| --- | --- |
| S09\_MeetingResponse | |
| Test case ID | MSASCMD\_S09\_TC14\_MeetingResponse\_ForwardSubstituteMeetingInvitationEmail |
| Description | This test case is used to verify if delegate user forward substitute meeting invitation email, server will append the original meeting request to the forwarded message. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one meeting request to User7.  2. The client calls SwitchUser to switch User1 to User7.  3. The client using User7 calls Sync command to synchronize the changes of Inbox folder to get the new added meeting request email.  4. The client using User7 calls Sync command to synchronize the changes of Calendar folder.  5. The client calls SwitchUser to switch User7 to User8.  6. The client using User8 calls SmartForward command to forward email to User2.  7. The client calls SwitchUser to switch User8 to User2.  8. The client using User2 calls Sync command to synchronize the changes of Inbox folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S09\_TC14\_MeetingResponse\_ForwardSubstituteMeetingInvitationEmail

|  |  |
| --- | --- |
| S10\_MoveItems | |
| Test case ID | MSASCMD\_S10\_TC01\_MoveItems\_Success |
| Description | This test case is used to verify if the MoveItems command executes successfully, a new serverID should be assigned by the server. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted email to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User 1 calls FolderSync command to synchronize the collection hierarchy in User1's mailbox.  5. The client using User1 calls Sync command to synchronize changes of Inbox folder.  6. The client using User1 calls Sync command to synchronize changes of Deleted Items folder.  7. The client using User1 calls MoveItems command with the email item's ServerId to move the email item from Inbox folder to Deleted Items folder.  8. The client using User1 calls Sync command to synchronize changes of Inbox folder and DeleteItem folder after Move command success. |
| Cleanup | Common Cleanup |

MSASCMD\_S10\_TC01\_MoveItems\_Success

|  |  |
| --- | --- |
| S10\_MoveItems | |
| Test case ID | MSASCMD\_S10\_TC02\_MoveItems\_Status1\_InvalidSrcFldId |
| Description | This test case is used to verify the MoveItems command, if the request contains invalid source collection ID, the status in response is equal to 1. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted email to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User 1 calls FolderSync command to synchronize the collection hierarchy in User1's mailbox.  5. The client using User1 calls Sync command to synchronize changes of Inbox folder between the client and the server, and get the ServerId of sent email item and the SyncKey.  6. The client useing User1 calls MoveItems command with the email item’s ServerId to move the email item from an invalid source folder to Deleted Items folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S10\_TC02\_MoveItems\_Status1\_InvalidSrcFldId

|  |  |
| --- | --- |
| S10\_MoveItems | |
| Test case ID | MSASCMD\_S10\_TC03\_MoveItems\_Status1\_InvalidSrcMsgId |
| Description | This test case is used to verify the MoveItems command, if the request contains an invalid source Item ID, the status in response is equal to 1. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted email to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User 1 calls FolderSync command to synchronize the collection hierarchy in User1's mailbox.  5. The client calls MoveItems command with the received email item's ServerId to move the email item from Inbox folder to Deleted Items folder.  6. The client calls MoveItems command with the ServerId of the recived email again from Inbox folder to Deleted Items folder after the email item is not exist in Inbox folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S10\_TC03\_MoveItems\_Status1\_InvalidSrcMsgId

|  |  |
| --- | --- |
| S10\_MoveItems | |
| Test case ID | MSASCMD\_S10\_TC04\_MoveItems\_Status2 |
| Description | This test case is used to verify MoveItems command, if the request contains invalid destination collection ID, the status in response is equal to 2. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted email to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User 1 calls FolderSync command to synchronize the collection hierarchy in User1's mailbox.  5. The client using User1 calls Sync command to synchronize changes of Inbox folder between the client and the server.  6. The client using User1 calls MoveItems command with the email item's ServerId to move the email item from Inbox folder to an invalid destination folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S10\_TC04\_MoveItems\_Status2

|  |  |
| --- | --- |
| S10\_MoveItems | |
| Test case ID | MSASCMD\_S10\_TC05\_MoveItems\_Status4 |
| Description | This test case is used to verify the MoveItems command, if the source and destination collection IDs are the same, the status in response is equal to 4. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted email to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User 1 calls FolderSync command to synchronize the collection hierarchy in User1's mailbox.  5. The client using User1 calls Sync command to synchronize changes of Inbox folder between the client and the server.  6. The client using User1 calls MoveItems command with the email item's ServerId to move the email item from Inbox folder to Inbox folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S10\_TC05\_MoveItems\_Status4

|  |  |
| --- | --- |
| S10\_MoveItems | |
| Test case ID | MSASCMD\_S10\_TC06\_MoveItems\_Status5 |
| Description | This test case is used to verify the MoveItems command, if the request contains more than one destination collection ID, the status in response is equal to 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted email to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User 1 calls FolderSync command to synchronize the collection hierarchy in User1's mailbox.  5. The client using User1 calls Sync command to synchronize changes of Inbox folder between the client and the server.  6. The client using User1 calls Sync command to synchronize changes Deleted Itmes folder between the client and the server  7. The client using User1 calls Sync command to synchronize changes of Sent Items folder between the client and the server  8. The client using User1 calls MoveItems command with the email item's ServerId and two different destination collection IDs to move the email item from Inbox folder to Deleted Items folder and Sent Items Folder in one MoveItems command request.  9. The client using User1 calls Sync command to synchronize changes of Inbox folder, Sent Items folder and Deleted Items folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S10\_TC06\_MoveItems\_Status5

|  |  |
| --- | --- |
| S11\_Ping | |
| Test case ID | MSASCMD\_S11\_TC01\_Ping\_Status3 |
| Description | This test case is used to verify when the Ping command request omitted required parameters, the Ping command response has status value 3. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User3.  2. The client calls ChangeDeviceID to set a new device id.  3. The client calls FolderSync command to synchronize the collection hierarchy.  4. The client calls Ping command to create a Ping request with a null Folders element.  5. The client calls Ping command to create a Ping request with Folder element and HeartBeatInterval element.  6. The client calls Ping command to create a Ping request with Folder element and without HeartBeatInterval element.  7. The client calls ClearDevice with DeleteDevice method to clear the user’s ActiveSync device. |
| Cleanup | Common Cleanup |

MSASCMD\_S11\_TC01\_Ping\_Status3

|  |  |
| --- | --- |
| S11\_Ping | |
| Test case ID | MSASCMD\_S11\_TC02\_Ping\_Status4 |
| Description | This test case is used to verify when the Ping command request has the syntax error, the Ping command response has status value 4. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Ping command to send an empty ping request. |
| Cleanup | Common Cleanup |

MSASCMD\_S11\_TC02\_Ping\_Status4

|  |  |
| --- | --- |
| S11\_Ping | |
| Test case ID | MSASCMD\_S11\_TC03\_Ping\_Status5 |
| Description | This test case is used to verify when HeartbeatInterval is setting to a negative value, the Ping command response has status value 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Ping command to create a ping request with a negative value of HeartbeatInterval. |
| Cleanup | Common Cleanup |

MSASCMD\_S11\_TC03\_Ping\_Status5

|  |  |
| --- | --- |
| S11\_Ping | |
| Test case ID | MSASCMD\_S11\_TC04\_Ping\_Status7\_SyncRequired |
| Description | This test case is used to verify folder hierarchy sync is required before calling the Ping command by client. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User3.  2. The client calls ChangeDeviceID to set a new device id.  3. The client using User3 calls Ping command to create a Ping request with new device id.  4. The client calls ClearDevice with DeleteDevice method to clear the user’s ActiveSync device. |
| Cleanup | Common Cleanup |

MSASCMD\_S11\_TC04\_Ping\_Status7\_SyncRequired

|  |  |
| --- | --- |
| S11\_Ping | |
| Test case ID | MSASCMD\_S11\_TC05\_Ping\_Status7\_Outdate |
| Description | This test case is used to verify when a folder is out of date, the Ping command response have status value 7 |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User3.  2. The client calls ChangeDeviceID to set the first device id.  3.The client using User3 to calls FolderSync command to synchronize the collection hierarchy with the first device id created in step 2.  4. The client calls ChangeDeviceID to set the second device id.  5. The client using User3 to calls FolderSync command to synchronize the collection hierarchy with the second device id created in step 4.  6. The client calls FolderCreate command to create a new folder under Inbox folder, returns serverId.  7. The client calls ChangeDeviceID to change device id to first device id created in step 2.  8. The client using User3 and the first device id calls Ping command to create a Ping request. |
| Cleanup | Common Cleanup |

MSASCMD\_S11\_TC05\_Ping\_Status7\_Outdate

|  |  |
| --- | --- |
| S11\_Ping | |
| Test case ID | MSASCMD\_S11\_TC06\_Ping\_ChangesOccurred |
| Description | This test case is used to verify the response specifies the changed folders when changes occurred in two of the monitored folders. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls FolderCreate command to create two subfolders in Inbox folder.  3. The client using User2 calls SyncChanges with Sync command to sync the created folders.  4. The client using User2 calls Ping command to create multiple ping requests.  5. The client calls SwitchUser to switch User2 to User1.  6. The client using User1 calls SendMail to send one mail to User2.  7. The client calls SwitchUser to switch User1 to User2.  8. The client using User2 calls Sync command to get the email.  9. The client using User2 calls MoveItems command to move the email item from inbox folder to one of created subfolder.  10. The client calls Ping command to retrive the changed folders information.  11. The client calls Ping command with empty HeartbeatInterval element. |
| Cleanup | Common Cleanup |

MSASCMD\_S11\_TC06\_Ping\_ChangesOccurred

|  |  |
| --- | --- |
| S11\_Ping | |
| Test case ID | MSASCMD\_S11\_TC07\_Ping\_HeartbeatIntervalSetTooLarge |
| Description | This test case is used to verify when heartbeat interval is setting to a too large number, the Ping command response have status value 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Ping command to create a ping request with HeartbeatInterval set to a large number which exceeds 3540. |
| Cleanup | Common Cleanup |

MSASCMD\_S11\_TC07\_Ping\_HeartbeatIntervalSetTooLarge

|  |  |
| --- | --- |
| S11\_Ping | |
| Test case ID | MSASCMD\_S11\_TC08\_Ping\_WithoutRequestBody |
| Description | This test case is used to verify a Ping command without request body. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Ping command with a normal value of HeartbeatInterval in the request; 2. The client calls Ping command again without request body. |
| Cleanup | Common Cleanup |

MSASCMD\_S11\_TC08\_Ping\_WithoutRequestBody

|  |  |
| --- | --- |
| S12\_Provision | |
| Test case ID | MSASCMD\_S12\_TC01\_Provision\_DownloadPolicy |
| Description | This test case is used to verify when download policies from server, server should return provision policies and a template policy key, and then acknowledge the policies by using template policy. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Provision command to send a downloading policies command of provision to the server.  2. The client calls Provision command to send acknowledge policies command of provision to the server. |
| Cleanup | Common Cleanup |

MSASCMD\_S12\_TC01\_Provision\_DownloadPolicy

|  |  |
| --- | --- |
| S12\_Provision | |
| Test case ID | MSASCMD\_S12\_TC02\_Provision\_WithoutPolicyTypeElement |
| Description | This test case is used to verify if Provision command request does not include PolicyType element, the server returns status 2. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Provision command to send a downloading policies command of provision to the server without PolicyType element. |
| Cleanup | Common Cleanup |

MSASCMD\_S12\_TC02\_Provision\_WithoutPolicyTypeElement

|  |  |
| --- | --- |
| S12\_Provision | |
| Test case ID | MSASCMD\_S12\_TC03\_Provision\_Status144 |
| Description | This test case is used to verify if the client sends the Provision command with invalid policy key, server will return status value 144. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Provision command to download the policy from the server and receives a temporary policy key through the PolicyKey element.  2. The client calls Provision command to acknowledge the policy download and policy application by using a invalidate PolicyKey which different with the temporary PolicyKey obtained in step 1.  3. The client calls FolderSync command with an invalid PolicyKey which is different from the PolicyKey in step 2. |
| Cleanup | Common Cleanup |

MSASCMD\_S12\_TC03\_Provision\_Status144

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC01\_ResolveRecipients\_CertificateRetrieval |
| Description | This test case is used to verify whether the ResolveRecipients command is responded by server, if the client specifies the CertificateRetrieval element, server will return the corresponding status. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command with the CertificateRetrieval element setting to “1” to resolve a list of supplied recipients, retrieve their free/busy information, or retrieve their S/MIME certificates so that clients can send encrypted S/MIME email messages.  2. The client calls ResolveRecipients command with the CertificateRetrieval element setting to “2” to resolve a list of supplied recipients, retrieve their free/busy information, or retrieve their S/MIME certificates so that clients can send encrypted S/MIME email messages.  3. The client calls ResolveRecipients command with the CertificateRetrieval element setting to “3” to resolve a list of supplied recipients, retrieve their free/busy information, or retrieve their S/MIME certificates so that clients can send encrypted S/MIME email messages. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC01\_ResolveRecipients\_CertificateRetrieval

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC02\_ResolveRecipients\_EndTimeInvalid |
| Description | This test case is used to verify if EndTime is invalid, server returns status 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command with an invalid value of EndTime element to resolve a list of supplied recipients, retrieve their free/busy information, or retrieve their S/MIME certificates so that clients can send encrypted S/MIME email messages. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC02\_ResolveRecipients\_EndTimeInvalid

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC03\_ResolveRecipients\_StartTimeInvalid |
| Description | This test case is used to verify if StartTime is invalid, server returns status 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command with an invalid value of StartTime element to resolve a list of supplied recipients, retrieve their free/busy information, or retrieve their S/MIME certificates so that clients can send encrypted S/MIME email messages. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC03\_ResolveRecipients\_StartTimeInvalid

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC04\_ResolveRecipients\_EndTimeSmallThanStartTime |
| Description | This test case is used to verify when the EndTime is smaller than StartTime, server should return status 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command with valid value of EndTime and StartTime but the value of EndTime is smaller than StartTime. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC04\_ResolveRecipients\_EndTimeSmallThanStartTime

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC05\_ResolveRecipients\_DurationIsTooLarge |
| Description | This test case is used to verify StartTime and EndTime of the spans over maximum value, then server returns a status element value of 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command with the duration spanned by the StartTime element value and the EndTime element value is greater than a specific number. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC05\_ResolveRecipients\_DurationIsTooLarge

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC06\_ResolveRecipients\_Over256 |
| Description | This test case is used to verify the length of To element can be up to 256. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command with To elements value length is 256.  2. The client calls ResolveRecipients command with To elements value length is 257. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC06\_ResolveRecipients\_Over256

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC07\_ResolveRecipients\_NoPictureExist |
| Description | This test case is used to verify if the picture does not exist, sever should return 173. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command with MaxPictures element setting to “3”. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC07\_ResolveRecipients\_NoPictureExist

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC08\_ResolveRecipients\_AmbiguousUser |
| Description | This test case is used to verify one or more Recipient elements are returned to the client in a Response element, when using ambiguous name in the request. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command with an ambiguous name in the command request. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC08\_ResolveRecipients\_AmbiguousUser

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC09\_ResolveRecipients\_AmbiguousUserAvailability |
| Description | The test case is used to verify response will not include free/busy data and MergedFreeBusy element, when To element specifies an ambiguous name and the Availability element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command with an ambiguous name and the Availability element. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC09\_ResolveRecipients\_AmbiguousUserAvailability

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC10\_ResolveRecipients\_Group\_Availability |
| Description | This test case is used to verify if the request contains Availability element and a group name, server will return the merger of the data for the individual members of the specified group. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command with a group name and Availability element. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC10\_ResolveRecipients\_Group\_Availability

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC11\_ResolveRecipients\_Status161 |
| Description | This test case is used to verify if the distribution group contains more than 20 members, a Status element value of 161 is returned in the response. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command to resolve a large group which contains more than 20 members. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC11\_ResolveRecipients\_Status161

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC12\_ResolveRecipients\_To |
| Description | This test case is used to verify To element(s) that are returned in the response correspond directly to the To element(s) that are specified in the request. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command with To element to resolve a list of supplied recipients, retrieve their free/busy information, or retrieve their S/MIME certificates so that client can send encrypted S/MIME email messages. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC12\_ResolveRecipients\_To

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC13\_ResolveRecipients\_MergedFreeBusy |
| Description | This test case is used to verify ResolveRecipients command, if the To element with valid SMTP addresses in the request message, the MergedFreeBusy elements will be in the response. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command by setting “To” element value with invalid SMTP address. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC13\_ResolveRecipients\_MergedFreeBusy

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC14\_ResolveRecipients\_NoCertificates |
| Description | This test case is used to verify whether the ResolveRecipients command is responded by server, if the client specifies a value of 2 in the CertificateRetrieval element, but there is no certificate on the server, the status is equal to 7. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1 The client calls ResolveRecipients command by setting CertificateRetrieval element value to “2” while server has no certificate. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC14\_ResolveRecipients\_NoCertificates

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC15\_ResolveRecipients\_MergedFreeBusyIntervalTime |
| Description | This test case is used to verify ResolveRecipients command, each digit in the MergedFreeBusy element value string indicates the free/busy status for the user or distribution list for every 30 minute interval. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1 The client calls SwitchUser to switch the current user to User2.  2. The client calls ResolveRecipients command to check User2’s MergedFreeBusy status before receiving meeting request.  3. The client calls SwitchUser to switch the User2 to User1.  4. The client using User1 calls SendMail command to send one meeting request to User2.  5. The client calls SwitchUser to switch User1 to User2.  6. The client calls GetMailItem with Sync command and FolderSync command to synchronize the changes of Inbox folder in step 4.  7. The client calls GetMailItem with Sync command and FolderSync command to synchronize the changes of Calendar folder in step 4.  8. The client calls ResolveRecipients command to check User2’s MergedFreeBusy status before sending meeting response. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC15\_ResolveRecipients\_MergedFreeBusyIntervalTime

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC16\_ResolveRecipients\_Picture\_Status173 |
| Description | This test case is used to verify ResolveRecipients command, if user does not have a contact photo, the status is equal to 173. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1 The client calls ResolveRecipients command to resolve a list of supplied recipients, retrieve their free/busy information, or retrieve their S/MIME certificates so that clients can send encrypted S/MIME email messages using User3 the does not have any contact photo. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC16\_ResolveRecipients\_Picture\_Status173

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC17\_ResolveRecipients\_Picture\_Success |
| Description | This test case is used to verify the ResolveRecipients command request is successful, when setting the parameters of the Picture element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1 The client calls ResolveRecipients command with setting Picture element to resolve a list of supplied recipients, retrieve their free/busy information, or retrieve their S/MIME certificates so that clients can send encrypted S/MIME email messages. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC17\_ResolveRecipients\_Picture\_Success

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC18\_ResolveRecipients\_Picture\_Status174 |
| Description | This test case is used to verify ResolveRecipients command, if contact photo exceeded the size limit set by the MaxSize element, the status is equal to 174. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1 The client calls ResolveRecipients command with exceeded size of contact photo to resolve a list of supplied recipients, retrieve their free/busy information, or retrieve their S/MIME certificates so that clients can send encrypted S/MIME email messages. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC18\_ResolveRecipients\_Picture\_Status174

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC19\_ResolveRecipients\_Picture\_Status175 |
| Description | This test case is used to verify ResolveRecipients command, if number of contact photos returned exceeded the size limit set by the MaxPictures element, the status is equal to 175. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1 The client calls ResolveRecipients command with exceeded size of contact photo number to resolve a list of supplied recipients, retrieve their free/busy information, or retrieve their S/MIME certificates so that clients can send encrypted S/MIME email messages. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC19\_ResolveRecipients\_Picture\_Status175

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC20\_ResolveRecipients\_Status5 |
| Description | This test case is used to verify ResolveRecipients command, if the number of To element in the request exceeded the size limit, the status is equal to 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1 The client creates a ResolveRecipients request with 101 recipients.  2. The client calls ResolveRecipients command to resolve the request with 101 To elements. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC20\_ResolveRecipients\_Status5

|  |  |
| --- | --- |
| S13\_ResolveRecipients | |
| Test case ID | MSASCMD\_S13\_TC21\_ResolveRecipients\_Status163 |
| Description | This test case is used to verify ResolveRecipients command, if free/busy data could not be retrieved from the server for a given recipient, the status is equal to 163. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1 The client sets the AccessRights to "None".  2, The client calls ResolveRecipients command to resolve the request with valid value of EndTime and StartTime but the value of EndTime is smaller than StartTime. |
| Cleanup | Common Cleanup |

MSASCMD\_S13\_TC21\_ResolveRecipients\_Status163

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC01\_Search\_Success |
| Description | This test case is used to verify the success status of Search command. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command to find entries in a mailbox. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC01\_Search\_Success

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC02\_Search\_MultipleAndElements |
| Description | This test case is used to verify when the Search command request has more than one And elements, the server should return the response with Status 8. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with two Add elements, and check the Status element of Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC02\_Search\_MultipleAndElements

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC03\_Search\_TooComplex\_Status8\_Class |
| Description | This test case is used to verify when the Class element of Search command isn't a child element of And element, the server should return the response with status value 8. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with Class elements not in And element, and check the Status element of Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC03\_Search\_TooComplex\_Status8\_Class

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC04\_Search\_TooComplex\_Status8\_CollectionId |
| Description | This test case is used to verify when the CollectionId element of Search command isn't a child element for And element, the server should return the response with Status 8. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with CollectionId elements not in And element, and check the Status element of Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC04\_Search\_TooComplex\_Status8\_CollectionId

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC05\_Search\_TooComplex\_Status8\_FreeText |
| Description | This test case is used to verify when the FreeText element of Search command isn't a child element for And element, the server should return the response with Status 8. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with FreeText elements not in And element, and check the Status element of Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC05\_Search\_TooComplex\_Status8\_FreeText

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC06\_Search\_MultipleDeepTraversal |
| Description | This test case is used to verify the Search command with more than one DeepTraversal elements in And element, and the returned status value is equal to 1. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with two DeepTraversal elements, and check the Status element of Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC06\_Search\_MultipleDeepTraversal

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC07\_Search\_MoreThanOneMIMESupport |
| Description | This test case is used to verify the Search command with more than one airSync:MIMESupport elements in And element, and the returned status value is equal to 1. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with two MIMESupport elements, and check the Status element of Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC07\_Search\_MoreThanOneMIMESupport

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC08\_Search\_MultipleRange |
| Description | This test case is used to verify the Search command with more than one Range elements in And element, and the returned status value is equal to 1. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with two Range elements, and check the Status element of Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC08\_Search\_MultipleRange

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC09\_Search\_MoreThanOneRebuildResults |
| Description | This test case is used to verify the Search command with more than one RebuildResults elements in And element, and the returned status value is equal to 1. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with two RebuildResults elements, and check the Status element of Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC09\_Search\_MoreThanOneRebuildResults

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC10\_Search\_MoreThanOneStores |
| Description | This test case is used to verify the Search command with more than one Store elements in And element, and the returned status value is equal to 1. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with two Store elements, and check the Status element of Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC10\_Search\_MoreThanOneStores

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC11\_Search\_Invalid\_Status2 |
| Description | This test case is used to verify the Search command with invalid formatted request to get status 2. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with email class element in And element to create an invalid formatted request, and check the Status element of Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC11\_Search\_Invalid\_Status2

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC12\_Search\_Result\_NoItemMatched |
| Description | This test case is used to verify there is no matched item for the Search request. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send an email message to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client calls GetMailItem with Sync command to synchronize the changes.  4. The client using User2 calls Search command with a non-existing FreeText element, and check the response of Search command. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC12\_Search\_Result\_NoItemMatched

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC13\_Search\_Result\_MultipleMatchedItems |
| Description | This test case is used to verify the Search response when the searched items have the multiple matched items. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send 2 email messages from User1 to User2 with same prefix.  2. The client calls SwitchUser to switch User1 to User2.  3. The client calls GetMailItem with Sync command to synchronize the changes.  4. The client using User2 calls Search command with the prefix as FreeText element, and check the response of Search command. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC13\_Search\_Result\_MultipleMatchedItems

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC14\_Search\_GAL\_Success |
| Description | This test case is used to verify search global address list success |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with options to query the entries in the GAL. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC14\_Search\_GAL\_Success

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC15\_Search\_WithAllClass |
| Description | This test case is used to verify search with Tasks, Email, Calendar, Contacts, Notes, SMS class elements. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command by setting the Class element value to Email/Tasks/Calendar/Contacts/Notes/SMS respectively and to verify if the Result element exists. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC15\_Search\_WithAllClass

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC16\_Search\_WithoutNoteAndSmsClass |
| Description | This test case is used to verify search without Notes and SMS class element when ProtocolVersion set to 12.1 |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command by setting the Class element value to Email/Tasks/Calendar/Contacts respectively. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC16\_Search\_WithoutNoteAndSmsClass

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC17\_Search\_WithoutClass |
| Description | This test case is used to verify search without class element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command to create one item in User2’ Inbox folder, Calendar folder and Contacts folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC17\_Search\_WithoutClass

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC18\_Search\_DocumentLibraryFolder |
| Description | This test case is used to verify if search point to one folder the first search result contains folder metadata and the second search result contains folder items metadata information. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with document library root folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC18\_Search\_DocumentLibraryFolder

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC19\_Search\_DocumentLibraryItem |
| Description | This test case is used to verify that if search document library point to specified folder item, only one search result will be returned. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with document library shared item. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC19\_Search\_DocumentLibraryItem

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC20\_Search\_GAL\_Status174 |
| Description | This test case is used to verify that if the contact photo exceeded the size limit set by MaxSize element , the search result should contain status 174. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with setting MaxSize element to “1\*1024”. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC20\_Search\_GAL\_Status174

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC21\_Search\_Status175 |
| Description | This test case is used to verify that server returns 175 if the number of contact photos returned exceeded the size limit set by the MaxPictures element |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with setting MaxPicture element to 1. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC21\_Search\_Status175

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC22\_Search\_WithDeepTraversal |
| Description | This test case is used to verify if search with DeepTraversal element, server should search subfolders. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send two emails to User2.  2. The client calls SwitchUser to switch the current user to User2.  3. The client calls FoderSync command to synchronize the collection hierarchy.  4. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder.  5. The client calls MoveItems command to move the email with emailSubject1 to new subfolder.  6. The client calls Search command with DeepTraversal element. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC22\_Search\_WithDeepTraversal

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC23\_Search\_WithoutDeepTraversal |
| Description | This test case is used to verify if search without DeepTraversal element, server should not search subfolders. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send two emails to User2.  2. The client calls SwitchUser to switch the current user to User2.  3. The client calls FoderSync command to synchronize the collection hierarchy.  4. The client calls FolderCreate command to create a new folder as a child folder of the specified parent folder.  5. The client calls MoveItems command to move the email with emailSubject1 to new subfolder.  6. The client calls Search command without DeepTraversal element. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC23\_Search\_WithoutDeepTraversal

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC24\_Search\_DocumentLibraryStatus5 |
| Description | This test case is used to verify that if user access to the resource is denied, server return status 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command for document library with unauthorized user. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC24\_Search\_DocumentLibraryStatus5

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC25\_Search\_DocumentLibraryStatus6 |
| Description | This test case is used to verify that if user access the resource is not found, server return status 6. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command for invalid document library. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC25\_Search\_DocumentLibraryStatus6

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC26\_Search\_GAL\_Status173 |
| Description | This test case is used to verify search GAL contacts with photo success. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with options. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC26\_Search\_GAL\_Status173

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC27\_Search\_TooComplex\_Status8\_ConversationId |
| Description | This test case is used to verify if ConversationId element is included outside of And element, server will return status 8. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with conversationId element outside of And element. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC27\_Search\_TooComplex\_Status8\_ConversationId

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC28\_Search\_CaseInsensitiveMatch |
| Description | This test case is used to verify Search comparisons are performed by using case-insensitive matching. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send 2 email messages to User2.  2. The client calls SwitchUser to switch the current user to User2.  3. The client calls GetMailItem with Sync command to synchronize the changes.  4. The client calls Search command with with lowCase search prefix. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC28\_Search\_CaseInsensitiveMatch

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC29\_Search\_MoreThanOneConversationId |
| Description | This test case is used to verify the Search command with more than one ConversationId elements in And element, and the returned status value is equal to 1. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with multiple ConversationId elements. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC29\_Search\_MoreThanOneConversationId

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC30\_Search\_TooComplex\_Status8\_Or |
| Description | This test case is used to verify if Search request includes Or element, server will return status 8. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with Or element. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC30\_Search\_TooComplex\_Status8\_Or

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC31\_Search\_TooComplex\_Status8\_GreaterThan |
| Description | This test case is used to verify when the GreaterThan element of Search command is not a child element for And element, the server should return the response with Status 8 |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with GreaterThan elements not in And element, and check the Status element of Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC31\_Search\_TooComplex\_Status8\_GreaterThan

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC32\_Search\_TooComplex\_Status8\_LessThan |
| Description | This test case is used to verify when the LessThan element of Search command is not a child element for And element, the server should return the response with Status 8. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Search command with LessThan elements not in And element, and check the Status element of Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC32\_Search\_TooComplex\_Status8\_LessThan

|  |  |
| --- | --- |
| S14\_Search | |
| Test case ID | MSASCMD\_S14\_TC33\_Search\_MIMESupport |
| Description | This test case is used to verify the response of searching mail operation with MIMESupport setting that contains the Body element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User2.  2. The client using User2 calls SendMail command to send MIME-formatted e-mail messages to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User1 calls FolderSync command to synchronize the collection hierarchy.  5. The client using User1 calls Sync command to synchronize changes in Inbox folder between the client and the server.  6. The client using User1 calls Search command with the MIMESupport element setting to 1 and FreeText equal to the subject of the sent email to get the expected sent email in step 1, and check the Body element in Search response.  7. The client using User1 calls Search command with the MIMESupport element equal to 2 and FreeText equal to the subject of the sent email to get the expected sent email in step 1, and check the Body element in Search response. |
| Cleanup | Common Cleanup |

MSASCMD\_S14\_TC33\_Search\_MIMESupport

|  |  |
| --- | --- |
| S15\_SendMail | |
| Test case ID | MSASCMD\_S15\_TC01\_SendMail\_Success |
| Description | This test case is used to verify the server returns an empty response, when mail sending successfully. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send MIME-formatted email messages to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to synchronize the changes in Inbox folder,and get the sent email item in step 1. |
| Cleanup | Common Cleanup |

MSASCMD\_S15\_TC01\_SendMail\_Success

|  |  |
| --- | --- |
| S15\_SendMail | |
| Test case ID | MSASCMD\_S15\_TC02\_SendMail\_AccountIdInvalid |
| Description | This test case is used to verify the server returns 166, when AccountId is invalid. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send message with invalid AccountId to the server, check the Status element of SendMail response.  2. The client calls SwitchUser to switch the current user to User2.  3. The client using User2 calls Sync command to synchronize the changes in Inbox folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S15\_TC02\_SendMail\_AccountIdInvalid

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC01\_Settings\_Oof\_Success |
| Description | This test case is used to verify OOF status using Settings command. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command with setting OofState to 2, and check the response of Settings command. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC01\_Settings\_Oof\_Success

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC02\_Settings\_DevicePassword\_Status2 |
| Description | This test case is used to verify the empty DevicePassword using Settings command. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command to send an bad settings request and set element under DevicePassword is present, but empty. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC02\_Settings\_DevicePassword\_Status2

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC03\_Settings\_OofState |
| Description | This test case is used to verify if either the StartTime element or the EndTime element is included in the request without the other, server return status 2. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command with only setting StartTime element.  2. The client calls Settings command with only setting EndTime element. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC03\_Settings\_OofState

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC04\_Settings\_UserInformation |
| Description | This test case is used to verify using Settings command to get user information. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command to get user information. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC04\_Settings\_UserInformation

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC05\_Settings\_OofSetToExternalKnown |
| Description | This test case is used to specify the OOF setting to known external users. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command to enable OOF setting to known external users.  2. The client calls Settings command to get OOF message.  3. The client calls Settings command to set OFF state disabled. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC05\_Settings\_OofSetToExternalKnown

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC06\_Settings\_OofSetToExternalUnknown |
| Description | This test case is used to specify the OOF setting to unknown external users. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command to enable OOF setting to unknown external users.  2. The client calls Settings command to get AppliesToExternalKnown OOF message.  3. The client calls Settings command to set OFF state disabled. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC06\_Settings\_OofSetToExternalUnknown

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC07\_Settings\_OofSetToInternal |
| Description | This test case is used to specify the OOF setting to internal users. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command to enable OOF setting to internal users.  2. The client calls Settings command to get AppliesToInternal OOF message.  3. The client calls Settings command to set OFF state disabled. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC07\_Settings\_OofSetToInternal

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC08\_Settings\_OofSetBodyTypeIsRequiredWithReplyMessage |
| Description | This test case is used to verify that BodyType element is required if the ReplyMessage element is present. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command with ReplyMessage element and BodyType element. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC08\_Settings\_OofSetBodyTypeIsRequiredWithReplyMessage

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC09\_Settings\_OofSetBodyTypeOptional |
| Description | This test case is used to verify the BodyType element is optional if the ReplyMessage element is not present. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command without ReplyMessage element and BodyType element.  2. The client calls Settings command to get AppliesToInternal OOF message.  3. The client calls Settings command to set OFF state disabled. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC09\_Settings\_OofSetBodyTypeOptional

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC10\_Settings\_OofSetForThreeTypeAudiences |
| Description | This test case is used to verify set OOF message for three type audiences which are external known users , external unknown users and internal users. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command with three types of audience.  2. The client calls Settings command to get OOF settings.  3. The client calls SetOofDisabled with Settings command to set OOF state to disabled. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC10\_Settings\_OofSetForThreeTypeAudiences

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC11\_Settings\_Status5 |
| Description | This test case is used to test if set BodyType to invalid data, server should return status 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command with invalid arguments. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC11\_Settings\_Status5

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC12\_Settings\_Status6 |
| Description | This test case is used to test if set different reply message to external known users and external unknown users, server should respond status 6. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command with two types of audience. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC12\_Settings\_Status6

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC13\_Settings\_ResponseInOrder |
| Description | This test case is used to test the implication of batching mechanism, server response should match the order of request. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command to get Oof and user information.  2. The client calls Settings command to set OFF state disabled. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC13\_Settings\_ResponseInOrder

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC14\_Settings\_GetRightManagementInformation |
| Description | This test case is used to test the RightsManagementInformation element in Settings response and request. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User3.  2. The client calls Settings command to get right management information. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC14\_Settings\_GetRightManagementInformation

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC15\_Settings\_SetDeviceInformationPhoneNumber |
| Description | This test case is used to test when the EnableOutboundSMS element is set to 1, the PhoneNumber is required. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command with setting EnableOutboundSMS element to 1. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC15\_Settings\_SetDeviceInformationPhoneNumber

|  |  |
| --- | --- |
| S16\_Settings | |
| Test case ID | MSASCMD\_S16\_TC16\_Settings\_DevicePassword\_LongPassword |
| Description | This test case is used to verify if set password length exceeds 256, server should return status 5. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Settings command with setting DevicePassword element and the password length exceeds 256. |
| Cleanup | Common Cleanup |

MSASCMD\_S16\_TC16\_Settings\_DevicePassword\_LongPassword

|  |  |
| --- | --- |
| S17\_SmartForward | |
| Test case ID | MSASCMD\_S17\_TC01\_SmartForward\_Success |
| Description | This test case is used to verify the server returns an empty response, when mail forward successfully. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send plain text email messages to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to synchronize the changes in Inbox folder,and get the sent email item in step 1.  4. The client using User2 calls SmartForward command to forward message to User3 without retrieving the full, original message from the server.  5. The client calls SwitchUser to switch User2 to User3.  6. The client using User3 calls Sync command to synchronize the changes in Inbox folder,and get the forwarded email item in step 4. |
| Cleanup | Common Cleanup |

MSASCMD\_S17\_TC01\_SmartForward\_Success

|  |  |
| --- | --- |
| S17\_SmartForward | |
| Test case ID | MSASCMD\_S17\_TC02\_SmartForward\_Fail |
| Description | This test case is used to verify server returns status code, when SmartForward is failed. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send plain text email messages to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to synchronize the changes in Inbox folder,and get the sent email item in step 1.  4. The client using User2 calls SmartForward command with an empty Mime element, and check the Status element of SmartForwward response. |
| Cleanup | Common Cleanup |

MSASCMD\_S17\_TC02\_SmartForward\_Fail

|  |  |
| --- | --- |
| S17\_SmartForward | |
| Test case ID | MSASCMD\_S17\_TC03\_SmartForwardAppointment |
| Description | This test case is used to verify when the SmartForward command is used for an appointment, the original message is included as an attachment in the outgoing message. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls Sync command to create one appointment by uploading a calendar item.  2. The client using User1 calls SmartForward command to forward mail to User2.  3. The client calls SwitchUser to switch User1 to User2.  4. The client using User2 calls Sync command to get the forwarded email in step 2. |
| Cleanup | Common Cleanup |

MSASCMD\_S17\_TC03\_SmartForwardAppointment

|  |  |
| --- | --- |
| S17\_SmartForward | |
| Test case ID | MSASCMD\_S17\_TC04\_SmartForwardWithInvalidInstanceId |
| Description | This test case is used to verify if the value of the InstanceId element is invalid, the server returns status value 104. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one recurring meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to synchronize the changes in Inbox folder,and get the sent email item in step 1.  4. The client using User2 calls Sync command to synchronize the changes in Calendar folder.  5. The client using User2 calls SmartForward command to forward the calendar item to User3 with invalid InstanceId. |
| Cleanup | Common Cleanup |

MSASCMD\_S17\_TC04\_SmartForwardWithInvalidInstanceId

|  |  |
| --- | --- |
| S17\_SmartForward | |
| Test case ID | MSASCMD\_S17\_TC05\_SmartForwardWithInstanceIdSuccess |
| Description | This test case is used to verify when SmartForward is applied to a recurring meeting, the InstanceId element specifies the ID of a particular occurrence in the recurring meeting. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one recurring meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to synchronize the changes in Inbox folder,and get the sent email item in step 1.  4. The client using User2 calls MeetingResponse command to accept the meeting.  5. The client using User2 calls Sync command to synchronize the calendar changes in Calendar folder.  6. The client using User2 calls SmartForward command to forward the calendar item to User3 with correct InstanceId.  7. The client calls SwitchUser to switch User2 to User3.  8. The client using User3 calls Sync command to synchronize the changes in Inbox folder.  9. The client using User3 calls Sync command to synchronize the changes in Calendar folder and get the forwarded calendar item in step 5.  10. The client calls SwitchUser to switch User3 to User1.  11. The client calls GetMailItem with FolderSync command to synchronize the change of DeletedItem folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S17\_TC05\_SmartForwardWithInstanceIdSuccess

|  |  |
| --- | --- |
| S17\_SmartForward | |
| Test case ID | MSASCMD\_S17\_TC06\_SmartForwardRecurringMeetingWithoutInstanceId |
| Description | This test case is used to verify when SmartForward request without the InstanceId element, the implementation forward the entire recurring meeting. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one recurring meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to synchronize the changes in Inbox folder,and get the sent email item in step 1.  4. The client using User2 calls Sync command to synchronize the changes in Calendar folder.  5. The client using User2 calls SmartForward command to forward the calendar item to User3 without InstanceId.  6. The client calls SwitchUser to switch User2 to User3.  7. The client using User3 calls Sync command to synchronize the changes in Inbox folder.  8. The client using User3 calls Sync command to synchronize the changes in Calendar folder and get the forwarded calendar item in step 5.  9. The client calls SwitchUser to switch User3 to User1.  10. The client calls GetMailItem with Sync command and FolderSync command to synchronize the collection hierarchy. |
| Cleanup | Common Cleanup |

MSASCMD\_S17\_TC06\_SmartForwardRecurringMeetingWithoutInstanceId

|  |  |
| --- | --- |
| S18\_SmartReply | |
| Test case ID | MSASCMD\_S18\_TC01\_SmartReply\_ContainOriginalMessage |
| Description | This test case is used to verify the replied mail contains original message. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one recurring meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to synchronize the changes in Inbox folder,and get the sent email item in step 1.  4. The client using User2 calls Sync command to synchronize the changes in Calendar folder.  5. The client using User2 calls SmartReply command to reply message without instanceID in SmartReply command request.  6. The client calls SwitchUser to switch User2 to User1.  7. The client using User1 calls GetMailItem with Sync command and FolderSync command to synchronize the changes in Inbox folder,and get the replied email item in step 5. |
| Cleanup | Common Cleanup |

MSASCMD\_S18\_TC01\_SmartReply\_ContainOriginalMessage

|  |  |
| --- | --- |
| S18\_SmartReply | |
| Test case ID | MSASCMD\_S18\_TC02\_SmartReply\_Success |
| Description | This test case is used to verify the server returns an empty response, when mail is replied successfully. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one plain text email to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to synchronize the changes in Inbox folder,and get the sent email item in step 1.  4. The client using User2 calls SmartReply command without ReplaceMime element to reply message and without retrieving the full, original message from the server, and check the response of SmartReply command.  5. The client calls SwitchUser to switch User2 to User1.  6. The client using User1 calls Sync command to synchronize the changes in Inbox folder,and get the replied email item in step 4. |
| Cleanup | Common Cleanup |

MSASCMD\_S18\_TC02\_SmartReply\_Success

|  |  |
| --- | --- |
| S18\_SmartReply | |
| Test case ID | MSASCMD\_S18\_TC03\_SmartReply\_Status104 |
| Description | This test case is used to verify if the value of the InstanceId element is invalid, the server responds with Status value 104. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send one recurring meeting request to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to synchronize the changes in Inbox folder,and get the sent email item in step 1.  4. The client using User2 calls Sync command to synchronize the changes in Calendar folder.  5. The client using User2 calls SmartReply command with invalid InstanceId element. |
| Cleanup | Common Cleanup |

MSASCMD\_S18\_TC03\_SmartReply\_Status104

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC01\_Sync\_Success |
| Description | This test case is used to verify the requirements related to a successful Sync command. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls Sync command to add a new contact item in Contacts folder.  3. The client calls Sync command to synchronize the changes in Contacts folder, and get the new contact item in step 2.  4. The client calls Sync command to send an empty request. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC01\_Sync\_Success

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC02\_Sync\_MoreAvailable |
| Description | This test case is used to verify the MoreAvailable element will be returned in the Sync command response if there are more changes than the number that are requested in the WindowSize element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderSync command to synchronize the collection hierarchy.  2. The client calls FolderCreate command to create a folder under the Contacts folder.  3. The client calls Sync command to get an initial SyncKey in the folder created in step 2.  4. The client calls Sync command to add two contact items into the folder created in step 2.  5. The client calls Sync command with WindowSize element setting to 1 to get the latest SyncKey.  6. The client calls Sync command to synchronize the changes in the folder created in step 2.  7. The client calls Sync command with Collections element to synchronize the changes in the folder created in step 2. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC02\_Sync\_MoreAvailable

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC03\_Sync\_Email\_Sequence |
| Description | This test case is used to verify the Sync command sequence for folder synchronization. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send a MIME-formatted email messages to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client using User2 calls Sync command to synchronize the changes in Inbox folder.  4 The client calls Sync command with the SyncKey returned in step 3 to synchronize the changes in Inbox folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC03\_Sync\_Email\_Sequence

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC04\_Sync\_Status3 |
| Description | This test case is used to verify Sync command, if the synchronization key is invalid, then the status code in the server response will be 3. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with an invalid SyncKey and check the Status elment of Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC04\_Sync\_Status3

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC05\_Sync\_Status4 |
| Description | This test case is used to verify Sync command, if the request does not comply with the specification requirements, then the status code in the server response will be 4. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with a Sync request containing no child elements.  2. The client calls Sync command to add a contact item to the recipient information cache. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC05\_Sync\_Status4

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC06\_Sync\_Status13 |
| Description | This test case is used to verify Sync command, if an empty Sync command request is received and the cached set of notify-able collections is missing, then the status code in the server response will be 13. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls Sync command with an empty Sync command request and check the Status element of Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC06\_Sync\_Status13

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC07\_Sync\_HeartbeatInterval\_Status14 |
| Description | This test case is used to verify Sync command, if the HeartbeatInterval element is outside the range set, or smaller than the minimum allowable value, then the status code in the server response will be 14. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with HeartbeatInterval element setting to “3541”.  2. The client calls Sync command with HeartbeatInterval element setting to “59”. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC07\_Sync\_HeartbeatInterval\_Status14

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC08\_Sync\_ConversationMode\_Status4 |
| Description | This test cases is used to verify the server will return a status value 4 if call Sync command with ConversationMode element for collections that do not store e-mails. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls sync command with ConversationMode set to true and CollectionId set to the id of collection that do not store emails. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC08\_Sync\_ConversationMode\_Status4

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC09\_Sync\_Class\_Redefined |
| Description | This test case is used to verify Sync command, if options for the same Class within the same Collection are redefined, a Status element value of 4 is returned. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls Sync command to add a calendar item with two Options elements defined, the first Options element defines the Class element, and the second Optionselement also defines the Class element with same value, and check the Status element in Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC09\_Sync\_Class\_Redefined

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC10\_Sync\_NoChildElementsForCommands |
| Description | This test case is used to verify Sync command, if no child elements for the Commands element are defined, server does not return a status error. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with Commands element that has no child elements and check the Status elment in Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC10\_Sync\_NoChildElementsForCommands

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC11\_Sync\_Email\_FilterType |
| Description | This test case is used to verify if Sync command for e-mail, the status should be correspond to the value of FilterType. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with FilterType setting to “0” to get all items Inbox folder and check the Status element of Sync response.  2. The client calls Sync command with FilterType setting to “1” to get items that are dated within 1 day in Inbox folder and check the Status element of Sync response.  3. The client calls Sync command with FilterType setting to “2” to get items that are dated within 3 days in Inbox folder and check the Status element of Sync response.  4. The client calls Sync command with FilterType setting to “3” to get items that are dated within 1 week in Inbox folder and check the Status element of Sync response.  5. The client calls Sync command with FilterType setting to “4” to get items that are dated within 2 weeks in Inbox folder and check the Status element of Sync response.  6. The client calls Sync command with FilterType setting to “5” to get items that are dated within 1 month in Inbox folder and check the Status element of Sync response.  7. The client calls Sync command with FilterType setting to 6 to get items that are dated within 3 months in Inbox folder and check the Status element of Sync response.  8. The client calls Sync command with FilterType setting to 7 to get items that are dated within 6 months in Inbox folder and check the Status element of Sync response.  9. The client calls Sync command with FilterType setting to 8 to get all incomplete items in Inbox folder and check the Status element of Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC11\_Sync\_Email\_FilterType

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC12\_Sync\_Calendar\_FilterType |
| Description | This test case is used to verify if Sync command for Calendar, the status should be correspond to the value of FilterType. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to add a new contact in Calendar folder.  2. The client calls Sync command with FilterType setting to 0 to get all items in Calendar folder and check the Status element of Sync response.  3. The client calls Sync command to synchronize the changes and get the latest SyncKey.  4. The client calls Sync command without FilterType element.  5. The client calls Sync command to synchronize the changes and get the latest SyncKey.  6. The client calls Sync command with FilterType setting to 1 to get all items that are dated within 1 day in Calendar folder and check the Status element of Sync response.  7. The client calls Sync command with FilterType setting to 2 to get all items that are dated within 3 days in Calendar folder and check the Status element of Sync response.  8. The client calls Sync command with FilterType setting to 3 to get all items that are dated within 1 week in Calendar folder and check the Status element of Sync response.  9. The client calls Sync command with FilterType setting to 4 to get all items that are dated within 2 weeks in Calendar folder and check the Status element of Sync response.  10. The client calls Sync command to synchronize the changes and get the latest SyncKey.  11. The client calls Sync command to add a future Calendar item.  12. The client calls Sync command without FilterType element.  13. The client calls Sync command with FilterType setting to 4 to get all items that are dated within 2 weeks in Calendar folder and check the Status element of Sync response.  14. The client calls Sync command to synchronize the changes and get the latest SyncKey.  15. The client calls Sync command to add a recurring Calendar item without EndTime.  16. The client calls Sync command without FilterType element.  17. The client calls Sync command to synchronize the changes and get the latest SyncKey.  18. The client calls Sync command with FilterType setting to 4 to get all items that are dated within 2 weeks in Calendar folder and check the Status element of Sync response.  19. The client calls Sync command to synchronize the changes and get the latest SyncKey.  20. The client calls Sync command with FilterType setting to 5 to get all items that are dated within 1 month in Calendar folder and check the Status element of Sync response.  21. The client calls Sync command to synchronize the changes and get the latest SyncKey.  22. The client calls Sync command with FilterType setting to 6 to get all items that are dated within 3 months in Calendar folder and check the Status element of Sync response.  23. The client calls Sync command to synchronize the changes and get the latest SyncKey.  24. The client calls Sync command with FilterType setting to 7 to get all items that are dated within 6 months in Calendar folder and check the Status element of Sync response.  25. The client calls Sync command to synchronize the changes and get the latest SyncKey.  26. The client calls Sync command to create an overdue calendar.  27. The client calls Sync command without FilterType element.  28. The client calls Sync command to synchronize the changes and get the latest SyncKey.  29. The client calls Sync command with FilterType setting to 7 to get all items that are dated within 6 months in Calendar folder and check the Status element of Sync response.  30. The client calls Sync command to synchronize the changes and get the latest SyncKey.  31. The client calls Sync command with FilterType set to 8 to synchronize the changes for Calendar folder and check the Status element of Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC12\_Sync\_Calendar\_FilterType

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC13\_Sync\_Tasks\_FilterType |
| Description | This test case is used to verify if Sync command for Tasks, the status should be correspond to the value of FilterType. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with FilterType setting to 0 to get all items in Tasks folder and check the Status element of Sync response.  2. The client calls Sync command with FilterType setting to 1 to get all items that are dated within 1 day in Tasks folder, and check the Status element of Sync response.  3. The client calls Sync command with FilterType setting to 2 to get all items that are dated within 3 days in Tasks folder, and check the Status element of Sync response.  4. The client calls Sync command with FilterType setting to 3 to get all items that are dated within 1 week in Tasks folder, and check the Status element of Sync response.  5. The client calls Sync command with FilterType setting to 4 to get all items that are dated within 2 weeks in Tasks folder, and check the Status element of Sync response.  6. The client calls Sync command with FilterType setting to 5 to get all items that are dated within 1 month in Tasks folder, and check the Status element of Sync response.  7. The client calls Sync command with FilterType setting to 6 to get all items that are dated within 3 months in Tasks folder, and check the Status element of Sync responsev.  8. The client calls Sync command with FilterType setting to 7 to get all items that are dated within 6 months in Tasks folder, and check the Status element of Sync response.  9. The client calls Sync command with FilterType setting to 8 to get all incomplete items in Tasks folder, and check the Status element of Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC13\_Sync\_Tasks\_FilterType

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC14\_Sync\_MoreThanOneFilterTypes |
| Description | This test case is used to verify the server does not return a protocol status error if including more than one FilterType elements in Sync command request. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send a MIME-formattted email to User2.  2. The client calls SwitchUser to switch the current user to User2.  3. The client calls Sync command with two FilterType elements and check the Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC14\_Sync\_MoreThanOneFilterTypes

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC15\_Sync\_IncludeFilterTypeOrNot |
| Description | This test case is used to verify Sync command for contacts, if FilterType element is not included in a contact Sync request, no error is thrown. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey without FilterType element and check the response.  2. The client calls Sync command to get an initial SyncKey with FilterType element and compare and check the responses. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC15\_Sync\_IncludeFilterTypeOrNot

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC16\_Sync\_WindowSize\_MoreAvailable\_Omitted |
| Description | This test case is used to verify Sync command, if no additional changes are available, MoreAvailable element is omitted. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls Sync command to add a contact item.  3. The client calls Sync command without WindowSize element to synchronize the changes in the Contacts folder, check the Sync response.  4. The client calls Sync command to get the latest SyncKey. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC16\_Sync\_WindowSize\_MoreAvailable\_Omitted

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC17\_Sync\_WindowSize\_MoreAvailable |
| Description | This test case is used to verify Sync command, if the WindowSize element is omitted, the server behaves as if a WindowSize element with a value of 100 were submitted. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderSync command to synchronize the collection hierarchy, return the CollectionId of Calendar folder.  2. The client calls FolderCreate command to create a folder under the Contacts folder.  3. The client calls Sync command to add 101 calendar items into the new created folder.  4. The client calls Sync command to get an initial SyncKey.  5. The client calls Sync command without WindowSize element to synchronize the changes in Contacts folder.  6. The client calls Sync command to get the latest SyncKey. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC17\_Sync\_WindowSize\_MoreAvailable

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC18\_Sync\_SMIME |
| Description | This test case is used to verify the Sync command, when handling S/MIME content in the response, the server MUST include the airsyncbase:Body element, which is a child of the ApplicationData element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send MIME-formatted email messages to User2.  2. The client calls SwitchUser to switch User 1 to User2.  3. The client calls Sync command to synchronize the changes, and find the sent email item, and check the Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC18\_Sync\_SMIME

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC19\_Sync\_NoResponsesElement |
| Description | This test case is used to verify Sync command, if the client requested server changes but had no changes to send to the server, the Response element is omitted. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls Sync command to synchronize the changes in Inbox folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC19\_Sync\_NoResponsesElement

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC20\_Sync\_MoreThanOneClasses |
| Description | This test case is used to verify the server does not return a protocol status error if call Sync command with more than one Class elements. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send a MIME-formattted email to User2.  2. The client calls SwitchUser to switch the current user to User2.  3. The client calls Sync command with two Class elements to get an initial SyncKey and check the Status element in Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC20\_Sync\_MoreThanOneClasses

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC21\_Sync\_MoreThanOneMaxItems |
| Description | This test case is used to verify Sync command, if request includes more than one MaxItem element, the server does not return error. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with two MaxItem elements to get an initial SyncKey and check the Status element in Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC21\_Sync\_MoreThanOneMaxItems

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC22\_Sync\_MoreThanOneMIMESupport |
| Description | This test case is used to verify Sync command, if request includes more than one MIMESupport element, the server does not return error. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send a MIME-formattted email to User2.  2. The client calls SwitchUser to switch the current user to User2.  3. The client calls Sync command with two MIMESupport elements to get an initial SyncKey and check the Status element in Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC22\_Sync\_MoreThanOneMIMESupport

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC23\_Sync\_MoreThanOneMIMETruncation |
| Description | This test case is used to verify Sync command, if request includes more than one MIMETruncation elements, the server does not return error. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send a MIME-formattted email to User2.  2. The client calls SwitchUser to switch the current user to User2.  3. The client calls Sync command with two MIMETruncation elements to get an initial SyncKey and check the Status element in Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC23\_Sync\_MoreThanOneMIMETruncation

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC24\_Sync\_Status6\_EndTimeMissing |
| Description | This test case is used to verify Sync command, when adding a new calendar item, of which the EndTime element is missing, the status should be equal to 6. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to synchronize the changes in a collection between the client and the server.  2. The client calls Sync command to add a calendar with StartTime element but without EndTime element, and check the Status element in Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC24\_Sync\_Status6\_EndTimeMissing

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC25\_Sync\_Status6\_AddEmail |
| Description | This test case is used to verify Sync command, when adding an email item to the server, the status should be equal to 6. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls Sync command with the SyncKey returned in step 1 to add an email item, and check the Status element of Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC25\_Sync\_Status6\_AddEmail

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC26\_Sync\_GetChanges |
| Description | This test case is used to verify the requirements related to GetChanges element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with GetChanges element setting to true, and check the Status element in Sync response.  2. The client calls Sync command with GetChanges element is present and setting to true and check the Status element in Sync response.  3. The client calls Sync command with GetChanges element setting to false, and check the Status element in Sync response.  4. The client calls Sync command with GetChanges element is absent, and check the Status element in Sync response.  5. The calls Sync command to add a new contact.  6. The client calls Sync command with GetChanges element setting to false and check the Status element in Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC26\_Sync\_GetChanges

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC27\_Sync\_Change |
| Description | This test case is used to verify Sync command, if there are changes since the last synchronization, the server response includes a Commands element that contains additions, deletions, and changes. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to synchronize the changes of Contacts folder.  2. The client calls Sync command to add a contact item.  3. The client calls ChangeDeviceId to change DeviceId to “Device2”.  4. The client calls Sync command to synchronize the changes in the Contacts folder.  5. The client calls Sync command to change the value of FileAs element of the contact item added in step 2, and then calls Sync command to synchronize the changes to the server.  6. The client calls ChangeDeviceID to change DeviceId to the original device.  7. The client calls Sync command to synchronize the changes in the Contacts folder, and then checks the Sync response.  8. The client using User1 calls SendMail command to send a MIME-formatted email to User2.  9. The client calls SwitchUser to switch User1 to User2.  10. The client calls Sync command to set a flag to an email.  11. The client calls Sync command to update the email flag.  12. The client calls Sync command to synchronize the changes and get the latest SyncKey.  13. The client using User1 calls SendMail command to send a MIME-formatted email to User2.  14. The client calls SwitchUser to switch User1 to User2.  15. The client calls Sync command to set a Read element in an email.  16. The client calls Sync command to synchronize the changes and get the latest SyncKey.  17. The client calls SwitchUser to switch User2 to User1.  18. The client using User1 calls SendMail command to send a MIME-formatted email to User2.  19. The client calls SwitchUser to switch User1 to User2.  20. The client calls Sync command to set a Categories in an email.  21. The client calls Sync command to synchronize the changes and get the latest SyncKey.  22. The client calls SwitchUser to switch User2 to User1.  23. The client calls Sync command to get an initial SyncKey.  24. The client calls Sync command to add a new contact.  25. The client calls Sync command to synchronize the changes and get the latest SyncKey.  26. The client calls Sync command to update the FileAs element of the contact.  27. The client calls Sync command without FileAs element.  28. The client calls Sync command to synchronize the changes and get the latest SyncKey. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC27\_Sync\_Contact\_Change

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC28\_Sync\_Change\_InvalidServerId |
| Description | This test case is used to verify Sync command, if operation failed, Change element as a child element of Response element should appear. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get the initial SyncKey.  2. The client calls Sync command to change a contact item with an invalid serverId, and check the Change element in Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC28\_Sync\_Change\_InvalidServerId

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC29\_Sync\_HeartbeatIntervalAndWait\_StatusIs4 |
| Description | This test case is used to verify Sync command, if both HeartbeatInterval and Wait elements are included, server should return a status value of 4. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with both HeartbeatInterval and the Wait elements, check the Status element in Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC29\_Sync\_HeartbeatIntervalAndWait\_StatusIs4

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC30\_Sync\_Limit\_WaitIsLessThan1 |
| Description | This test case is used to verify Sync command, if a Wait element value of less than 1 is sent, the server will return a Limit element value of 1. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with a Wait element value of “0” to synchronize the changes in Inbox folder, and check the Limit element in sync reponse. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC30\_Sync\_Limit\_WaitIsLessThan1

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC31\_Sync\_Limit\_WaitIsGreaterThan59 |
| Description | This test case is used to verify Sync command, if a Wait element value of greater than 59 is sent, the server will return a Limit element value of 59. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with a Wait element value of “60” to synchronize the changes in Inbox folder, and check the Limit element in sync reponse. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC31\_Sync\_Limit\_WaitIsGreaterThan59

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC32\_Sync\_Limit\_HeartbeatIntervalIsLessThan60 |
| Description | This test is used to verify Sync command, if a HeartbeatInterval element value of less than 60 is sent, the server will return a Limit element value of 60. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with a HeartbeatInterval element value of “59” to synchronize the changes in Inbox folder, and check the Limit element in sync reponse. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC32\_Sync\_Limit\_HeartbeatIntervalIsIsLessThan60

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC33\_Sync\_Limit\_HeartbeatIntervalIsLargerThan3540 |
| Description | This test is used to verify Sync command, if a HeartbeatInterval element value of greater than 3540 is sent, the server will return a Limit element value of 3540. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with a HeartbeatInterval element value of “3541” to synchronize changes in Inbox folder, and check the Limit element in sync reponse. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC33\_Sync\_Limit\_HeartbeatIntervalIsLargerThan3540

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC34\_Sync\_DeletesAsMovesIsTrue |
| Description | This test case is used to verify Sync command, if DeletesAsMoves is true, the deleted item is moved to the Deleted Items folder. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send a MIME-formatted email to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client calls Sync command to check if the email has been in the Inbox folder and not in the DeletedItems folder.  4. The client using User2 calls Sync command with setting DeleteAsMoves to true to delete the email sent by User1.  5. The client using User2 calls Sync command to check if the email has been deleted from the Inbox folder.  6. The client using User2 calls Sync command to check if the email has been moved to the DeletedItems folder.  7. The client using User1 calls SendMail command to send a MIME-formatted email to User2.  8. The client calls SwitchUser to switch User1 to User2.  9. The client calls Sync command to check if the email has been in the Inbox folder and not in the DeletedItems folder.  10. The client using User2 calls Sync command without DeleteAsMoves to delete the email sent by User1.  11. The client using User2 calls Sync command to check if the email has been deleted from the Inbox folder.  12. The client using User2 calls Sync command to check if the email has been moved to the DeletedItems folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC34\_Sync\_DeletesAsMovesIsTrue

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC35\_Sync\_DeletesAsMovesIsFalse |
| Description | This test case is used to verify Sync command, if DeletesAsMoves is false, the deletion is permanent. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send a MIME-formatted email to User2.  2. The client calls SwitchUser to switch User1 to User2.  3. The client calls Sync command to check if the email has been in the Inbox folder and not in the DeletedItems folder.  4. The client using User2 calls Sync command with setting DeleteAsMoves to false to delete the email sent by User1.  5. The client using User2 calls Sync command to check if the email has been deleted from the Inbox folder.  6. The client using User2 calls Sync command to check if the email has not been moved to the DeletedItems folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC35\_Sync\_DeletesAsMovesIsFalse

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC36\_Sync\_Fetch |
| Description | This test case is used to verify Sync command for Fetch operation. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get the initial SyncKey.  2. The client calls Sync command to add a contact in Contacts folder.  3. The client calls Sync command to get the server id of the new created item.  4. The client calls Sync command with setting server id gotten from step 3 in the Fetch element to fetch the new created item. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC36\_Sync\_Fetch

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC37\_Sync\_WindowSize\_512 |
| Description | This test case is used to verify Sync command, if the WindowSize element is set to 512, the server can send Sync response messages that contain less than 512 updates. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls FolderSync command to synchronize the collection hierarchy, return the CollectionId of Contacts folder.  2. The client calls FolderCreate command to create a user created contacts folder, and gets the CollectionId of the new Contacts folder.  3. The client calls Sync command to add 513 contact items in new Contacts folder.  4. The client calls Sync command without setting WindowSize element.  5. The client calls Sync command to synchronize the changes in new Contacts folder with WindowSize element set to 512, and check both Add and MoreAvailable elements in Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC37\_Sync\_WindowSize\_512

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC38\_Sync\_Delete |
| Description | This test case is used to verify Sync command for Delete operation. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to synchronize the changes of Contacts folder.  2. The client calls Sync command to add a contact item.  3. The client calls Sync command to synchronize the changes in Contacts folder, and get the created contact item.  4. The client calls ChangeDeviceId to set the device id to “Device2”.  5. The client calls Sync to synchronize the changes of Contacts folder.  6. The client calls Sync commands to delete the got contact item in step 3.  7. The client calls ChangeDeivceID to restore the device id to the original device id.  8. The client calls Sync command to synchronize the changes in Contacts folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC38\_Sync\_Delete

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC39\_Sync\_Supported |
| Description | This test case is used to verify the requirements related to Supported element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get the initial SyncKey.  2. The client calls Sync command to add a new contact.  3. The client calls Sync command to set SyncKey to 0 with “Supported” element.  4. The client calls Sync command to get the latest SyncKey.  5. The client calls Sync command to synchronize the changes in Contact folder.  6. The client calls Sync command to change the JobTiltle element.  7. The client calls Sync command to get the latest SyncKey.  8. The client calls Sync command to synchronize the changes.  9. The client calls Sync command to get the latest SyncKey.  10. The client calls Sync command to add a new contact.  11. The client calls Sync command to get the latest SyncKey.  12. The client calls Sync command without Supported element to indicate that elements that can be ghosted are considered not ghosted.  13. The client calls Sync command to change the JobTiltle element.  14 The client calls Sync command to get the latest SyncKey..  15. The client calls Sync command to synchronize the changes.  16. The client calls Sync command to get the latest SyncKey.  17. The client calls Sync command to add a new contact.18. The client calls Sync command to indicate that JobTitle element is not ghosted.  19. The client calls Sync command to synchronize the changes.  20. The client calls Sync command to change FileAs element.  21. The client calls Sync command to get the latest SyncKey.  22. The client calls Sync command to synchronize the changes.  23. The client calls Sync command to get the latest SyncKey.  24. The client calls Sync command to add a new contact.  25. The client calls Sync command with empty Supported element.  26. The client calls Sync command to synchronize the changes.  27. The client calls Sync command to change the JobTiltle element.  28. The client calls Sync command to get the latest SyncKey.  29. The client calls Sync command to synchronize the changes. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC39\_Sync\_Supported

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC40\_Sync\_Conflict |
| Description | This test case is used to verify Sync command, if the client has changed an item for which the conflict policy indicates that the server's changes take precedence, server will return a status value 7. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to synchronize the changes of the Contacts folder.  2. The client calls Sync command to add a contact in Contacts folder.  3. The client calls Sync command to set SyncKey to 0.  4. The client calls Sync command to synchronze the changes and get the latest SyncKey.  5. The client calls ChangeDeviceID to set device id to “Device2”.  6. The client calls FolderSync command with new device id to synchronize the collection hierarchy,return the CollectionId of the Contacts folder.  7.The client calls Sync command to get the latest SyncKey.  8. The client calls Sync command to get the created contact in Contacts folder.  9. The client calls Sync command to change the got contact in step 8.  10. The client calls ChangeDeviceID to restore the devide id to original device id.  11. The client calls Sync command without Conflict element to change the got contact in step 8.  11. The client calls Sync command with Conflict element setting to “1” to change the contact item again.  12. The client calls Sync command with Conflict element setting to “0” to change the contact item again. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC40\_Sync\_Conflict

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC41\_Sync\_MoreThanOneConflicts |
| Description | This test case is used to verify Sync command, if it includes more than one Conflict elements as the child of an Options element, the server does not return any error code. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to sync changes with two Conflicts elements set in Sync request, and check the status of the Sync response. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC41\_Sync\_MoreThanOneConflicts

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC42\_Sync\_WithCombinationClasses |
| Description | This test case is used to verify the combination of the classes in the Sync command. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send a MIME-formatted email to User2.  2. The client calls SwitchUser to switch the current user to User2.  3. The client calls Sync command to get both Email and SMS messages.  4. The client calls Sync command to get the initial SyncKey.  5. The client calls Sync command to synchronize the changes in Task folder.  6. The client calls Sync command to get both Calendar and Tasks items. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC42\_Sync\_WithCombinationClasses

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC43\_Sync\_MIMETruncation |
| Description | This test case is used to verify the requirements related to MIMETruncation element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send a MIME-formatted email to User2.  2. The client calls SwitchUser to switch the current user to User2.  3. The client calls Sync command with MIMETruncation element setting to “8” to get MIME data.  4. The client calls Sync command without MIMETruncation to get MIME data.  5. The client calls Sync command with MIMETruncation element setting to “0” to truncate all body data.  6. The client calls Sync command with MIMETruncation element setting to “1” to truncate text over 4096 characters.  7. The client calls Sync command with MIMETruncation element setting to “2” to truncate text over 5120 characters.  8. The client calls Sync command with MIMETruncation element setting to “3” to truncate text over 7168 characters.  9. The client calls Sync command with MIMETruncation element setting to “4” to truncate text over 10240 characters.  10. The client calls Sync command with MIMETruncation element setting to “5” to truncate text over 20480 characters.  11.The client calls Sync command with MIMETruncation element setting to “6” to truncate text over 51200 characters.  12. The client calls Sync command with MIMETruncation element setting to “7” to truncate text over 102400 characters. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC43\_Sync\_MIMETruncation

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC44\_Sync\_MIMESupport |
| Description | This test case is used to verify the requirements related to MIMESupport element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send a MIME-formatted email to User2.  2. The client calls SwitchUser to switch the current user to User2.  3. The client calls CheckMail with Sync command to check if the sent mail is in User2’s Inbox folder.  4. The client calls Sync command to get the email body and check that Type element, EstimatedDataSize element and Truncated element exist in the email. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC44\_Sync\_MIMESupport

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC45\_Sync\_ConversationMode |
| Description | This test case is designed to verify the ConversationMode element related requirements. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send a MIME-formatted email to User2.  2. The client calls SwitchUser to switch the current user to User2.  3. The client calls Sync command to get an initial SyncKey.  4. The client calls Sync command with ConversationMode setting to true.  5. The client calls Sync command to get an initial SyncKey.  6. The client calls Sync command wit FilterType setting to 1 to get items that are dated within 1 day in Inbox folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC45\_Sync\_ConversationMode

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC46\_Sync\_Supported\_Status4 |
| Description | This test case is used to verify Sync command, if the CollectionId element is not included in the Sync request, then the status code in the response is 4. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get an initial SyncKey.  2. The client calls Sync command to add a new contact.  3. The client calls Sync command without CollectionId element to indicate that the JobTitle element is not ghosted. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC46\_Sync\_Supported\_Status4

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC47\_Sync\_Wait\_Status14 |
| Description | This test case is used to verify Sync command, if the Wait element is outside the range set, or smaller than the minimum allowable value, then the status code in the server response is 14. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with Wait element setting to “60”.  2. The client calls Sync command with Wait element setting to “0”. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC47\_Sync\_Wait\_Status14

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC48\_Sync\_Collection\_LimitValue |
| Description | This test case is used to verify the limit value of the sum of the number of Add, Change, Delete, and Fetch elements of Sync command. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to create 51 Add elements, 50 Change elements, 50 Delete elements and 50 Fetch elements in the Contacts folder.  2. The client calls Sync command to synchronize the changes on Contacts folder. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC48\_Sync\_Collection\_LimitValue

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC49\_Sync\_SoftDelete |
| Description | This test case is used to verify the requirements related to SoftDelete element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get the initial SyncKey.  2. The client calls Sync command to add two new calendars: the first one is out of scope of 14 days; the second one is in this scope.  3. The client calls Sync command with FilterType of 0 to get all items in the Calendar folder.  4. The client calls Sync command with FilterType of 4 to get the number of the filtered-out SoftDelete elements.  5. The client calls Sync command with FilterType of 4 to get the number of the filtered-in Add elements. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC49\_Sync\_SoftDelete

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC50\_Sync\_Class |
| Description | This test case is used to verify the Class element in the Sync Add response. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command to get the initial SyncKey.  2. The client calls Sync command to add a new Note item. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC50\_Sync\_Class

|  |  |
| --- | --- |
| S19\_Sync | |
| Test case ID | MSASCMD\_S19\_TC51\_Sync\_Change\_Exceptions |
| Description | This test case is used to verify Sync command, if Exceptions element is not specified or not present in the sync change request, the original Exceptions element will remain unchanged. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync Add operation to add a new recurrece calendar.  2. The client changes the subject of the added recurrece calendar.  3. The client changes the subject of the added recurrece calendar again. |
| Cleanup | Common Cleanup |

MSASCMD\_S19\_TC51\_Sync\_Change\_Exceptions

|  |  |
| --- | --- |
| S20\_ValidateCert | |
| Test case ID | MSASCMD\_S20\_TC01\_ValidateCert\_Success |
| Description | This test case is used to verify the status value will be 1, when certificate validation is successfully completed for the ValidateCert command. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch the current user to User9.  2. The client calls Sync command to synchronize the changes of Inbox folder in User9’s mailbox between the client and the server and get the body content of the email item. |
| Cleanup | Common Cleanup |

MSASCMD\_S20\_TC01\_ValidateCert\_Success

|  |  |
| --- | --- |
| S20\_ValidateCert | |
| Test case ID | MSASCMD\_S20\_TC02\_ValidateCert\_InvalidSignature |
| Description | This test case is used to verify the status value is 3, when the signature in the certificate is invalid. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ValidateCert command to verify the certificate with invalid signature. |
| Cleanup | Common Cleanup |

MSASCMD\_S20\_TC02\_ValidateCert\_InvalidSignature

|  |  |
| --- | --- |
| S20\_ValidateCert | |
| Test case ID | MSASCMD\_S20\_TC03\_ValidateCert\_LimitingSize |
| Description | This test case is used to verify the status value is 2, when ValidateCert command request contains more than 100 Certificate elements. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ValidateCert command to verify the number of certificate element is more than 100. |
| Cleanup | Common Cleanup |

MSASCMD\_S20\_TC03\_ValidateCert\_LimitingSize

|  |  |
| --- | --- |
| S21\_CommonStatusCode | |
| Test case ID | MSASCMD\_S21\_TC01\_CommonStatusCode\_166 |
| Description | This test case is used to verify the server will return 166, when AccountId is invalid. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SendMail command to send plain text email messages with invalid AccountId to the server.  2. The client calls SwitchUser to switch the current user to User2.  3. The client using User2 calls Sync command to synchronize the changes of User2 mailbox. |
| Cleanup | Common Cleanup |

MSASCMD\_S21\_TC01\_CommonStatusCode\_166

|  |  |
| --- | --- |
| S21\_CommonStatusCode | |
| Test case ID | MSASCMD\_S21\_TC02\_CommonStatusCode\_173 |
| Description | This test case is used to verify server will return 173, when the picture does not exist. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls ResolveRecipients command to resolve a recipient which don’t have a picture. |
| Cleanup | Common Cleanup |

MSASCMD\_S21\_TC02\_CommonStatusCode\_173

|  |  |
| --- | --- |
| S21\_CommonStatusCode | |
| Test case ID | MSASCMD\_S21\_TC03\_CommonStatusCode\_165 |
| Description | This test case is used to verify the server will return 165, when the required DeviceInformation element is missing in the Provision request. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Provision command without the DeviceInformation element. |
| Cleanup | Common Cleanup |

MSASCMD\_S21\_TC03\_CommonStatusCode\_165

|  |  |
| --- | --- |
| S21\_CommonStatusCode | |
| Test case ID | MSASCMD\_S21\_TC04\_CommonStatusCode\_105 |
| Description | This test case is used to verify the server will return 105, when the request contains a combination of parameters that is invalid. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls SwitchUser to switch User1 to User2.  2. The client using User2 calls SendMail to send MIME-formatted email messages to User1.  3. The client calls SwitchUser to switch User2 to User1.  4. The client using User1 calls FolderSync command to synchronize the collection hierarchy.  5. The client using User1 calls Sync command to synchronize the changes of Inbox folder between client and server.  6. The client using User1 calls MoveItems command to move the email item from Inbox folder to recipient information cache. |
| Cleanup | Common Cleanup |

MSASCMD\_S21\_TC04\_CommonStatusCode\_105

|  |  |
| --- | --- |
| S21\_CommonStatusCode | |
| Test case ID | MSASCMD\_S21\_TC05\_CommonStatusCode\_164 |
| Description | This test case is used to verify the server returns 164, when the BodyPartPreference node has an unsupported Type element value. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client calls Sync command with setting an unsupported Type element in the BodyPartPreference node. |
| Cleanup | Common Cleanup |

MSASCMD\_S21\_TC05\_CommonStatusCode\_164

|  |  |
| --- | --- |
| S21\_CommonStatusCode | |
| Test case ID | MSASCMD\_S21\_TC06\_CommonStatusCode\_118 |
| Description | This test case is used to verify the server returns 118, when the message was already sent in a previous request. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. The client using User1 calls SendMail command to send email messages to User2.  2. The client calls SendMail command using the same ClientId as step 1. |
| Cleanup | Common Cleanup |

MSASCMD\_S21\_TC06\_CommonStatusCode\_118