Module Testing

Product: BackupBuddies

Team: Free Cloud Backups

SetStoragePath:

Description

Sets where files are downloaded to on recovery. Since the path is set using a GUI file browser, the path is guaranteed to exist and not be an empty string. For this reason, there is only one equivalence class.

Equivalence classes:

File path exists and is not empty

Test Cases:

- Choose ~/Downloads
  - Success: the path exists and is not null

ProcessUserList:

Description

Modify the raw list returned from the network interface and return a list of ListModels better suited for rendering.

Equivalence classes:

Raw user list contains duplicate users

Raw user list is empty

Raw user list is populated

Test Cases:

- list<string> = [‘user1’, ‘user2’, ‘user1’, ‘user3’]
  - Success: returns list<listmodel> with no duplicates
- list<string> = []
  - Success: returns list<listmodel> with no entries
- list<string> = [‘user1’, ‘user2’, ‘user3’]
  - Success: returns list<listmodel> with all users

ProcessFileList:

Description

Modify the raw list returned from the network interface and return a list of ListModels better suited for rendering. Note that there will be no duplicates.

Equivalence classes:

Raw file list is empty

Raw file list is populated

Test Cases:

- list<string> = []
  - Success: returns list<listmodel> with no entries
- list<string> = [‘user1’, ‘user2’, ‘user3’]

        - Success: returns list<listmodel> with all users

UploadFile/Compress/Encrypt:

Description

Uploads files

The file path input is selected with a GUI file picker - as such, we need not worry about it

Equivalence classes

File is too large to store

File is not too large to store

Test cases:

- Upload 1 MB of random bytes to a peer whose limit is set at 0 GB
    - Expect the connection to be closed and reopened by the receiver
- Upload 1 MB of random bytes to a peer whose limit is set at 1 GB
    - Expect the receiver to have the uploaded file stored in their storage path, compressed with zip and encrypted with the uploader's key

Download File/Decompress/Decrypt:

Description

Download a file from some peer

Equivalence classes:

File does not exist

File exists, correct decryption key

File exists, incorrect decryption key

Test Cases:

- Attempt to download a nonexistent file
    - Success: connection is terminated
- Attempt to download a file, but using a different encryption password than the one it was uploaded with
    - Success: Download fails, a message is logged
- Attempt to download a file, using the correct encryption password
    - Success: File is downloaded to the selected folder, and is byte-for-byte identical to the file uploaded

Get/Set Storage Space:

Description

Passes the value of the storage space to the network interface which will allow the user to upload and store larger or smaller quantities of data as they wish.

Equivalence classes:

Setting the storage while logged into the network
Setting the storage while not in a network
Setting the storage below the current number of gigabytes stored

Test Cases:

- Attempt to change the storage slider
  - Success: sends the new storage value to the network interface, and it is stored
- Change the slider while not logged into a network
  - Success: nothing gets sent to the network interface
- Attempt to change the storage slider below the current number of gigabytes stored.
  - Success: sends the new storage value to the network interface, and the new value is stored. However, new files may not be uploaded until the user slides the storage space slider past the current number of gigabytes stored on their device

Get/Set Display Name:

Description

Gets input from the display name field. Sets the name of the device to the input. Other users see this name instead of the device name.

Equivalence classes:

Nothing is entered into the display name field
Something is entered into the display name field

Test Cases:

- DisplayName = “” (nothing entered)
  - The device name is sent to the Network Interface to be used as the display name
- DisplayName = “NewName”
  - NewName is sent to the Network Interface to be set as the new display name

Search:

Description

Searches through the file list and displays only files with the matching string

Equivalence classes:

Nothing is typed into the search bar
String is typed into the search bar that matches at least one file
String is typed into the search bar that does not match with any file

Test Cases:

- Search = “” (empty string)
  - shows every file on the file list
- Search = “fileExists”, shows only fileExists on the file list
  - shows only fileExists on the file list

- Search = “fileNotExists”, shows nothing on the file list
    - shows only fileExists on the file list

Setting Encryption Key:

Description

Sets the key used to encrypt and decrypt files. Note that any string, even anempty string is a valid encryption key.

Equivalence classes:

Key is set before logging into a network after startup
Key is set while a network is active

Test Cases:

- Key = ‘encryptkey1’,
    - Success: error popup appears to warn the user
- Key = ‘encryptkey2’,
    - Success: log notifies key is sent to network interface file

Handshake:

Description

Connecting to a peer accepts only valid handshakes

Equivalence classes:

Handshake has a bad magic constant
Handshake has a bad UUID
Handshake has a bad password hash
Handshake is valid

Test Cases:

- Send a handshake that uses “not correct” as its magic constant
    - Success: Connection is closed
- Send a handshake that uses “not correct” as the UUID it sends
    - Success: Connection is closed
- Send a handshake that uses 16 zero bytes as its password hash
    - Success: Connection is closed
- Send a handshake that is valid
    - Success: Connection remains open and begins listening for commands

Notify of new peer:

Description

When sent, the receiver tries to connect to the new peer

Equivalence classes:

Always

Test Cases:

- Send packet naming a running Backup Buddies server
  - Success: Receiver attempts to connect to that server

List Files:

Description

Contains a list of all files that the peer has

Equivalence classes:

Always

Test Cases:

- Call ListFiles.send()
  - Success: Receiver has a complete list of all files stored by BackupBuddies on the sender

Notify Transfer Failed

Description

If file transfer fails, we send this packet

Equivalence classes:

Always

Test Cases:

- Send this message
  - Success: Receiver logs an error message in its log

Request File List

Description

Sender requests a list of receiver's stored files

Equivalence classes:

Always

Test Cases:

- Send this message
  - Success: Receiver calls listFiles.send()

Request File Restore

Description

Request that someone send us a copy of a file they have stored

Equivalence classes:

File exists

File does not exist

Test Cases:

- RestoreFile.send() with a file that does not exist on the receiver
  - Success: Receiver closes connection
- RestoreFile.send() with a file that exists on the receiver

        - Success: Receiver calls ReplyRestoreFiles.send()

File Restore:

Description

Restore a file to someone who requested it

Equivalence classes:

File was requested

File was not requested

Test Cases:

- Send a file that was not requested
    - Success: Receiver closes connection
- Send a file that was requested
    - Success: Receiver decrypts, decompresses and saves the file to disk

Network Exists:

Description

Checks if the interface has a network yet

Equivalence classes:

Interface has a network

Interface does not have a network

Test Cases:

- With no network in the interface, call the function
    - Success: Returns false
- With a network in the interface, call the function
    - Success: Returns true

Load/Save Network:

Description

Saves the network and reloads it

Equivalence classes:

Always

Test Cases:

- Create a network with some peers and some files. Call save, then call load.
    - Success: The reloaded network has records of all peers and files it had originally. It attempts to reconnect to all peers.