IPVS StoreAgent API

IPVS StoreAgent API

IPVS StoreAgent API

Table of Contents

List of Examples

Glossary

AppController   AppController running on managementserver. Controller will manage all modules of the system.

StoreManager   StoreManager Running on Appserver. Manager will do policy check of Store agent requests and send the requests to StoreAgent

StoreAgent   DeviceAdminAgent Running on Device. Handles the Request from StoreManager or Internal StoreServer.

StoreServer   StoreServer Running on Device. Which will handle the storage mount or unmount commands and broadcast storage updates

Introduction

This document contains the IPVS Store Agent API. Agent will accept requests from AppController or internal StoreServer

Chapter 1. Startup API

API Calls which used on start up store agent

External API

Requests Received from StoreManager or Clients

Internal API

Requests Received from DeviceClient

GetPublishInfo

Returns the Directory information Which needs to be published to appserver

Example 1.1. GetPublishInfoRequest

```
<GetPublishInfo/>
```

Example 1.2. GetPublishInfoResponse

```
   <GetPublishInfo>
 <Dir uuid='' revision='' serviceVersion=''/>
 <Dir uuid='' revision='' serviceVersion=''/>
</GetPublishInfo>
```

Chapter 2. Sync API

API Calls used on Store sync

External API

Requests Received from StoreManager

SyncDBWithStoreDir

API call used to sync the database with store. Which is used to remove the files from db if the blob does not have store file. Store give the response with the list of files which does not exist

Example 2.1. SyncDBWithStoreDirRequest

```
<SyncDBWithStoreDir byNID='' byUUID=''>
<Media byUUID='' byNID=''/>
</SyncDBWithStoreDir>
```

Example 2.2. SyncDBWithStoreDirResponse

```
 <SyncDBWithStoreDir NID='' UUID=''>
<Media UUID='' NID=''/>
 </SyncDBWithStoreDir>
```

Internal API

Requests Received from StoreServer/DeviceClient

SyncDir

API call used when StoreAgent received Status update from StoreServer (example Mount/umounted of NAS)

Example 2.3. SyncDirRequest

```
 <SyncDir>
  <Dir byNID='' byUUID='' revision='' serviceVersion='' state='Ready/
Offline/Error' />
  </SyncDir>
```

Example 2.4. SyncDirResponse

```
  <SyncDir>
 <Dir NID='' dbRevision='' />
  </SyncDir>
```

SyncFile

API Call is used in response to SyncDir call if the Dir out of synch

Example 2.5. SyncFileRequest

```
  <SyncFile from='' total='' countToFetch='25'>
 <File byNID='' byUUID=''>
  <MediaStoreFile serviceVer="" Ver="1.6" Rev="1">
   <MetaData>
    <Tag Name=""></Tag>
   </MetaData>

   <Type>MediaClip</Type>
   <MediaGroupID ClipID=""></MediaGroupID>
   <Title></Title>
   <Description></Description>
   <StreamType></StreamType>
   <State></State>
   <DateCreated></DateCreated>
   <DateModified></DateModified>

   <MediaClip>
    <PlayLength></PlayLength>

    <MediaStreamProfileInfo>
     <!-- Profile Info -->
    </MediaStreamProfileInfo>
   </MediaClip>
  </MediaStoreFile>
 </File>
 </SyncFile>
```

Example 2.6. SyncFileResponse

```
   <SyncFile  from='' total='' countToFetch=''/>
```

SetDirState

API call used when StoreAgent received Status update from StoreServer (example Mount/umounted of NAS)

Example 2.7. SetDirStateRequest

```
 <SetDirState>
<Dir byNID='' byUUID='' state='Ready/Offline/Error' />
 </SetDirState>
```

Example 2.8. SetDirStateResponse

```
<SyncDir/>
```

Chapter 3. Dir API

API Calls used to Upgrade/Register/UnRegister Directory

External API

Requests Received from StoreManager

UpgardeDir

Upgrade the media store dir to current version

Example 3.1. UpgradeDirRequest

```
<UpgradeDir byNID='' byUUID='' upgradeFiles='true/false'/>
```

Example 3.2. UpgradeDirResponse

```
<UpgradeDir/>
```

UnRegisterDir

UnRegister the directory from Management Server

Example 3.3. UnRegisterDirRequest

```
<UnRegisterDir byNID='' byUUID=''/>
```

Example 3.4. UnRegisterDirResponse

```
<UnRegisterDir/>
```

RegisterDir

Register the unregister directory to Management Server

Example 3.5. RegisterDirRequest

```
<RegisterDir byNID='' byUUID=''/>
```

Example 3.6. RegisterDirResponse

```
<RegisterDir/>
```

Internal API

Requests Received from DeviceClient

Chapter 4. File API

API Calls used to Add/Delete/Set/Upgrade/Import/Export of File

External API

Requests Received from StoreManager

AddFile

API call used to add mediastore file in blob

Example 4.1. AddFileRequest

```
    <AddFile dirNID='' dirUUID='' NID='' type='' uuid='' title=''
 description='' streamType=''>
    <MetaData>
       <Tag Name=''/>
    </MetaData>
    <MediaStreamProfileInfo/>
 </AddFile>
```

Example 4.2. AddFileResponse

```
    <AddFile />
```

SetFile

API call used to update mediastore file information in blob

Example 4.3. SetFileRequest

```
  <SetFile dirNID='' dirUUID='' byUUID='' NID='' title=''
 description=''>
    <MetaData>
       <Tag Name=''/>
    </MetaData>
    <MediaStreamProfileInfo/>
 </SetFile>
```

Example 4.4. SetFileResponse

```
<SetFile />
```

DeleteFile

API call used to update mediastore file information in blob

Example 4.5. DeleteFileRequest

```
<DeleteFile dirNID=''  byUUID='' nid='' />
```

Example 4.6. DeleteFileResponse

```
<DeleteFile />
```

UpgradeFile

API call used to upgrade media store file

Example 4.7. UpgradeFileRequest

```
<UpgradeFile dirNID=''  byUUID='' nid='' />
```

Example 4.8. UpgradeFileResponse

```
<UpgradeFile />
```

ImportFile

API call used to upgrade media store file

Example 4.9. ImportFileRequest

```
<ImportFile dirNID=''  byUUID='' nid='' />
```

Example 4.10. ImportFileResponse

```
<ImportFile />
```

ExportFile

API call used to upgrade media store file

Example 4.11. ExportFileRequest

```
<ExportFile dirNID=''  byUUID='' nid='' />
```

Example 4.12. ExportFileResponse

```
<ExportFile />
```

Internal API

Requests Received from DeviceAdminServer