TR-140
TR-069 Data Model for Storage Service Enabled Devices

Issue: 1 Amendment 1
Issue Date: April 2010

Notice

The Broadband Forum is a non-profit corporation organized to create guidelines for broadband network system development and deployment. This Broadband Forum Technical Report has been approved by members of the Forum. This Broadband Forum Technical Report is not binding on the Broadband Forum, any of its members, or any developer or service provider. This Broadband Forum Technical Report is subject to change, but only with approval of members of the Forum. This Technical Report is copyrighted by the Broadband Forum, and all rights are reserved. Portions of this Technical Report may be copyrighted by Broadband Forum members.

This Broadband Forum Technical Report is provided AS IS, WITH ALL FAULTS. ANY PERSON HOLDING A COPYRIGHT IN THIS BROADBAND FORUM TECHNICAL REPORT, OR ANY PORTION THEREOF, DISCLAIMS TO THE FULLEST EXTENT PERMITTED BY LAW ANY REPRESENTATION OR WARRANTY, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY WARRANTY:

(A)   OF ACCURACY, COMPLETENESS, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE;
(B)   THAT THE CONTENTS OF THIS BROADBAND FORUM TECHNICAL REPORT ARE SUITABLE FOR ANY PURPOSE, EVEN IF THAT PURPOSE IS KNOWN TO THE COPYRIGHT HOLDER;
(C)   THAT THE IMPLEMENTATION OF THE CONTENTS OF THE TECHNICAL REPORT WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

By using this Broadband Forum Technical Report, users acknowledge that implementation may require licenses to patents. The Broadband Forum encourages but does not require its members to identify such patents. For a list of declarations made by Broadband Forum member companies, please see http://www.broadband-forum.org. No assurance is given that licenses to patents necessary to implement this Technical Report will be available for license at all or on reasonable and non-discriminatory terms.

ANY PERSON HOLDING A COPYRIGHT IN THIS BROADBAND FORUM TECHNICAL REPORT, OR ANY PORTION THEREOF, DISCLAIMS TO THE FULLEST EXTENT PERMITTED BY LAW (A) ANY LIABILITY (INCLUDING DIRECT, INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES UNDER ANY LEGAL THEORY) ARISING FROM OR RELATED TO THE USE OF OR RELIANCE UPON THIS TECHNICAL REPORT; AND (B) ANY OBLIGATION TO UPDATE OR CORRECT THIS TECHNICAL REPORT.

Broadband Forum Technical Reports may be copied, downloaded, stored on a server or otherwise re-distributed in their entirety only, and may not be modified without the advance written permission of the Broadband Forum.

The text of this notice must be included in all copies of this Broadband Forum Technical Report.

Issue History

Issue Number	Issue Date	Issue Editor	Changes
Issue 1.0	August 2007	Tim Spets, Westell John Blackford, 2Wire	Original
Issue 1.1	December 2007	John Blackford, 2Wire	Minor corrections to Section 5 "Notification Requirements". Changed UpTime to Uptime and Quota.{i}. to Quota.
Issue 1 Amendment 1		Heather Kirksey Alcatel-Lucent	Added following three parameters: StorageServiceNumberOfEntries, StorageService.{i}.LogicalVolume.{i}.Folder.{i}.UserAccess.{i}.Enable, StorageService.{i}.LogicalVolume.{i}.Folder.{i}.GroupAccess.{i}.Enable

Comments or questions about this Broadband Forum Technical Report should be directed to info@broadband-forum.org.

Editor	Heather Kirskey	Alcatel-Lucent

BroadbandHome™	Greg Bathrick	PMC Sierra
WG Chairs	Heather Kirksey	Alcatel-Lucent

Table of Contents

Executive Summary

TR-140, *TR-069 Data Model for Storage Service Enabled Devices*, permits the remote management of Storage Service devices via CWMP as defined in TR-069 and TR-106. It covers the data model for describing a Storage Service device as well as rules regarding notifications on parameter value change. General use cases are also described including standard data model profiles that would typically be seen while remotely managing a device of this nature.

1   Purpose and Scope

1.1   Purpose

TR-140, *TR-069 Data Model for Storage Service Enabled Devices*, permits the remote management of Storage Service devices via CWMP as defined in [2] and [3]. It covers the data model for describing a Storage Service device as well as rules regarding notifications on parameter value change. General use cases are also described including standard data model profiles that would typically be seen while remotely managing a device of this nature.

TR-140 defines StorageService as the container associated with the provisioning of objects for Storage Service devices. CPE making use of a StorageService object MUST adhere to all of the data-hierarchy requirements defined in [3]. In the context of [3], the StorageService object is a service object. As such, individual CPE devices can contain one or more of these objects within their Services object alongside the generic data objects defined in [3]. The presence of more than one StorageService object would be appropriate primarily where a CPE device serves as a management proxy for other non-TR-069 capable Storage Service devices. For example, an internet gateway device might serve as a management proxy for one or more non-TR-069 capable Network Attached Storage (NAS) Devices.

1.2   Scope

TR-140 defines the data model for provisioning a CPE device that maintains a storage service such as a NAS device by an Auto-Configuration Server (ACS) using the mechanism defined in [2].

The goals of this specification are as follows:

- Enable troubleshooting and remote configuration of Storage Service devices from an ACS.

- Accommodate Storage Service devices that are either embedded as part of an internet gateway device, as defined in [2], or a standalone device.
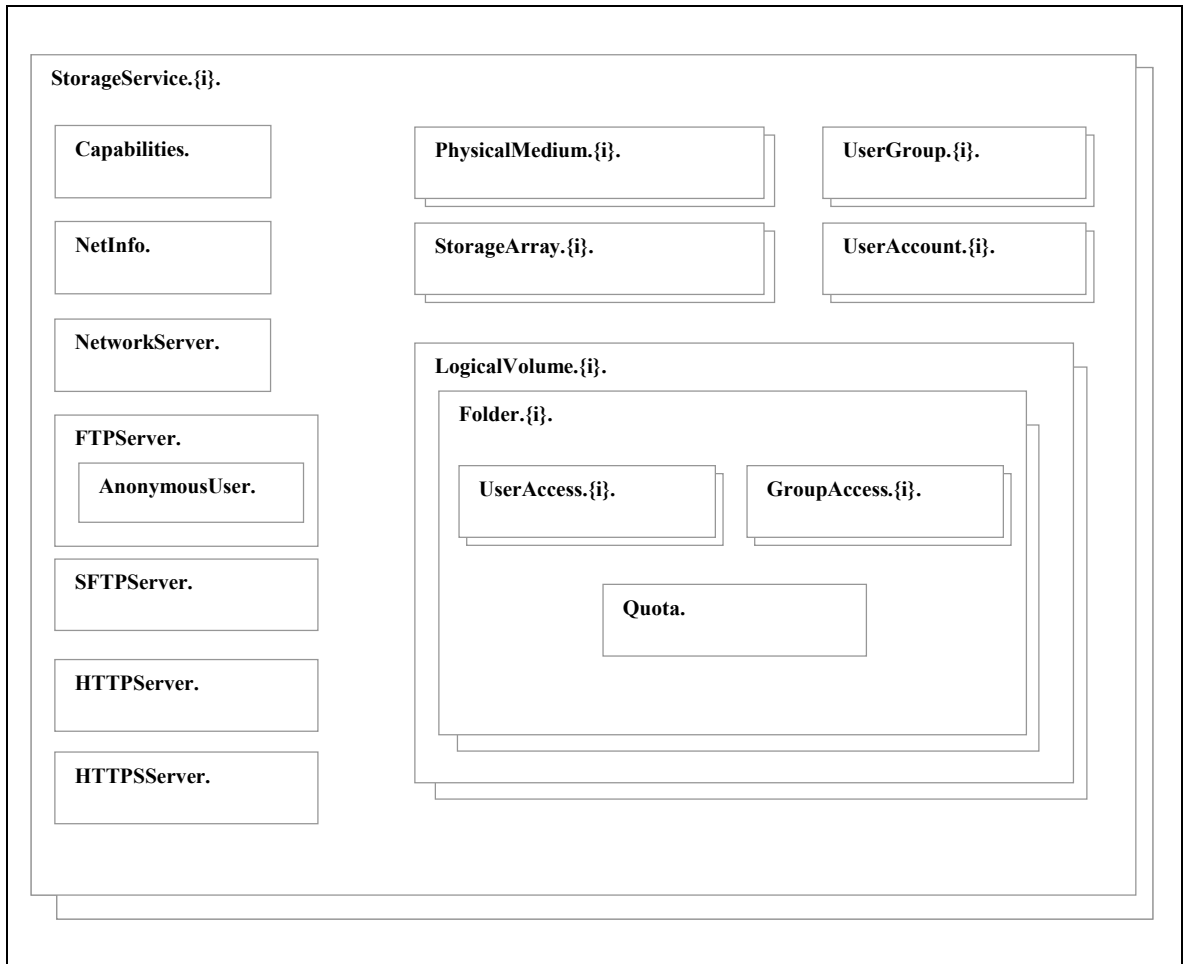
Figure 1 – StorageService Object Structure

Figure 1 depicts the StorageService object structure as seen in the Parameter Definitions section. This image provides a high-level overview of the different objects that exist in this data model and how they are associated.

2   References and Terminology

2.1   Conventions

In this Technical Report, several words are used to signify the requirements of the specification. These words are always capitalized. More information can be found be in RFC 2119 [1].

MUST	This word, or the term "REQUIRED", means that the definition is an absolute requirement of the specification.
MUST NOT	This phrase means that the definition is an absolute prohibition of the specification.
SHOULD	This word, or the adjective "RECOMMENDED", means that there may exist valid reasons in particular circumstances to ignore this item, but the full implications need to be understood and carefully weighed before choosing a different course.
SHOULD NOT	This phrase, or the phrase "NOT RECOMMENDED" means that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications need to be understood and the case carefully weighed before implementing any behavior described with this label.
MAY	This word, or the adjective "OPTIONAL", means that this item is one of an allowed set of alternatives. An implementation that does not include this option MUST be prepared to inter-operate with another implementation that does include the option.

2.2   References

The following references are of relevance to this Technical Report. At the time of publication, the editions indicated were valid. All references are subject to revision; users of this Technical Report are therefore encouraged to investigate the possibility of applying the most recent edition of the references listed below.

A list of currently valid Broadband Forum Technical Reports is published at www.broadband-forum.org

.

[1]	RFC 2119	*Key words for use in RFCs to Indicate Requirement Levels*	IETF	1997
[2]	TR-069 Amendment 1	*CPE WAN Management Protocol*	Broadband Forum	2006
[3]	TR-106 Amendment 1	*Data Model Template for TR-069-Enabled Devices*	Broadband Forum	2008

2.3   Definitions

The following terminology is used throughout this Technical Report.

AFP: The Apple Filing Protocol is a network protocol used to exchange files.

NAT: When a NAS device resides behind a residential gateway then typically Network Address Translation (NAT) will be used to allow the NAS device to communicate to the Internet through the residential gateway while protecting the IP address known to the NAS.

NFS: The Network File System is a network protocol used to exchange files.

RAID: A Redundant Array of Inexpensive Drives allows multiple physical media to be combined into a larger (in either size or number) storage concept.

SMB: Server Message Block is a network protocol used to exchange files.

WebDAV: Web-based Distributed Authoring and Versioning is used to enhance HTTP (and HTTPS) for publishing files.

2.4   Abbreviations

This Technical Report uses the following abbreviations:

ACS   Auto-Configuration Server
CPE   Customer Premises Equipment
CWMP   CPE WAN Management Protocol
FTP   File Transfer Protocol
HTTP   Hyper-Text Transfer Protocol
HTTPS   HTTP over SSL or TLS
NAS   Network Attached Storage
SMART   Self-Monitoring, Analysis, and Reporting Technology
SFTP   SSH FTP
SSH   Secure Shell
UPnP   Universal Plug and Play

3   Technical Report Impact

3.1   Energy Efficiency

TR-140 has no impact on Energy Efficiency.

3.2   IPv6

TR-140 has no impact on IPv6.

3.3   Security

TR-140 has no impact on Security

4   Parameter Definitions

The normative definition of the StorageService:1 data model is contained in a DM Instance document (see TR-069[2] Annex A). Table 1 lists the data model versions and DM Instances that had been defined at the time of writing. It also indicates the corresponding Technical Reports and gives links to the associated XML and HTML files.

Note that, because new minor versions of the StorageService:1 data model can be defined without re-publishing this Technical Report, the table is not necessarily up-to-date. An up-to-date version of the table can always be found at http://www.broadband-forum.org/cwmp.

Table 1 – StorageService:1 Data Model Versions

Version	DM Instance	Technical Report	XML and HTML[1]
1.0	tr-140-1-0.xml	TR-140	http://broadband-forum.org/cwmp/tr-140-1-0.xml
			http://broadband-forum.org/cwmp/tr-140-1-0.html
1.1	tr-140-1-1.xml	TR-106 Amendment 1	http://broadband-forum.org/cwmp/tr-140-1-1.xml
			http://broadband-forum.org/cwmp/tr-140-1-1.html
			http://broadband-forum.org/cwmp/tr-140-1-1-last.html

---

[1]   The HTML with a name of the form tr-xxx-i-a.html, e.g. tr-181-1-0.html, lists the entire data model. The HTML with a name of the form tr-xxx-i-a-dev.html, e.g. tr-157-1-0-dev.html, lists only the Device Root Object (not the InternetGatewayDevice Root Object). The HTML with a name of the form tr-xxx-i-a-last.html, e.g. tr-181-1-0-last.html, lists only the changes since the previous version. "dev" and "last" can be combined, e.g. tr-157-1-0-dev-last.html.

5   Use Cases

Below are some use case scenarios; user experience is a critical driver when introducing a new feature or device in the home network. This is an important reason to allow Service Providers a way to access complex devices by preventing trouble calls (auto-configuration) and resolving trouble calls in a timely fashion by directly accessing the device(s).

5.1   Basic Managed Storage Service

A basic managed storage service offers a Service Provider the option to assist the customer as soon as the customer's StorageService-enabled device is activated and being managed by the ACS.   The following is a sample list of support capabilities an ACS can provide using CWMP (NOTE: Not all of these capabilities are handled with this data model; some are handled from a protocol perspective and some are handled via other data models):

- Basic configuration and setup during device activation [addressed by TR-140 (configuration parameters) & TR-106 (configuration parameters)]
- User credentials setup and file privilege access [addressed by TR-140 (Folder Access)]
- Firmware upgrade [addressed by TR-069 (Download command)]
- Retrieval of device status [addressed by TR-140 (parameters) and TR-106]
- Wireless setup (e.g., WEP security) for a Storage Service device with Wi-Fi access [addressed by a future version of TR-106]
- Configuration and log file retrieval for root cause analysis of problems [addressed by a future version of TR-106]
- Monitoring active/passive notification events, e.g., volume capacity reached, and potential physical media failures [addressed by TR-140 (parameters) and TR-069 (notification mechanism)]
- Network diagnostics and troubleshooting, e.g., network connectivity to the Internet gateway device, and to the Internet [addressed by TR-106 (connection parameters)]

With such support capabilities from the ACS, a technical support agent will be better equipped to help a customer with Storage Service issues during trouble calls. This is in addition to the ACS performing basic configuration/setup tasks during device registration/activation.   Overall, the goal is to provide a true *"plug-and-play"* solution for the customers.

5.2   Remote Storage Back-up Service

A service provider can offer a file back-up service for selected volumes or folders on a Storage Service device. A Storage Service device allows the ability to upload files residing locally, initiated by a centralized network storage server via standard file transfer mechanisms such as SFTP and HTTPS. The file back-up operation can be initiated

automatically on a regular basis using a time schedule selected by the customer or based on a default schedule established during service activation.

This is a service option that customers can use to back-up their *"precious content"* remotely on a network server in case of potential device disk failures. Since a majority of Storage Service devices currently do not have RAID support for disk redundancy, the remote back-up service can be a viable alternative for a certain segment of customers.

5.3   Remote access of Storage Service Content from a Remote Location

This service provides a simple and secure solution for a customer to remotely access (over the internet) content on a Storage Service device installed in his/her home network. Access implies that the customer will be able to retrieve or load files to the Storage Service device.   Since the Storage Service device is typically sitting behind a NAT-enabled Internet Gateway Device, this service would need to consider how the customer could access it through the NAT-enabled device.

Annex A: Theory of Operations

The addition of the Storage Service device provides a data model for devices that have storage capabilities and are used for purposes such as file backup services, media server, and private storage for users or devices. These devices are usually connected within the home network, addressable behind the Gateway. These devices are typically accessible on the home network similarly to a PC through file sharing protocols such as Windows Networking (NetBIOS) and AppleTalk. The device providing the storage service is not usually a PC (although a PC is not excluded from providing the Storage Service) and thus the only interface to their functionality (file access) and setup (physical and logical drive formatting, wireless setup, DHCP server & client configuration, security provided by user access accounts) is through a Web based GUI, possibly UPnP, and (with TR-140) via TR-069.

A.1 Physical Storage Theory of Operations

Physical media are attached to a Storage Service through various methods. Some allow for hot-swapping and some are classified as removable media. The Storage Service device is responsible for discovery of all physical media attached to the device, and the removal of physical media that are no longer attached.   The parameter PhysicalMedium.{i}.Removable indicates that this physical medium can be removed during runtime operations, and the parameter PhysicalMedium.{i}.HotSwappable indicates that this physical medium can be swapped with another drive during runtime operations.

If a Physical Medium is implemented to be removable from the Storage Service device, some additional operational factors need to be considered.

Most external storage devices SHOULD be treated as removable, for example: USB hard-drives, FireWire hard-drives, eSATA hard-drives, USB memory drives, portable media players, PDAs, memory cards, etc.

Whether a Physical Medium is removable MUST be indicated by the StorageService.-{i}.PhysicalMedium.{i}.Removable parameter.

A hot-swappable Physical Medium is not considered removable storage in this discussion; removable storage implies that the removal action is part of normal operation and is expected to occur leaving the data on the removed storage intact; whereas, hot-swap capability is intended for removal in the case of disk error or failure.

It is expected that removable storage will typically be used for data transfer to and from the Storage Service, and not permanently participate in the Storage Service (although it is allowed to do so).   It is generally bad practice to have removable storage participate in a Storage Array, as removal will cause degradation or failure of the Storage Array

A.1.1   Physical Medium Discovery

The Storage Service device MUST discover any changes to the Physical Media in the system, including the addition or removal of a removable storage device during runtime.

The ACS can set up active notification on PhysicalMediumNumberOfEntries if it wishes to track addition or removal of a storage device.

Since a storage device could be removed and replaced with a different storage device, the ACS could check vendor/model/serial number before configuration of each removable Physical Medium to determine whether it is the same device or a different one from the last time it was configured.

A.1.2   Logical Volume Discovery

There is no requirement for a Storage Service device to recognize specific file systems that might have been externally formatted on a Physical Medium; however, the Storage Service device will report any recognized Logical Volume on the new Physical Medium.

A.1.3   Folder Discovery

The Storage Service device will report any recognized top-level Folder on the storage device.

A.1.4   Access Permissions

User Account and User Group access permissions MUST be re-configured each and every time a removable storage device is (re-)added to the system. Previous access permissions are not expected to be restored automatically. Access permissions for all newly discovered folders are UserAccountAccess = 0 (no Authentication required for Network Access or Remote Access Protocols).

A.1.5   Data Operations

Like any other Physical Media in the system, the Storage Service device does not define any data operations involving removable storage. Operations like backup to/from the removable storage would be up to other services, applications, or the user.

Also, the Storage Services does not define any methods for ensuring safe removal as required by some removable storage devices; it is left to the user to perform any steps necessary to ensure safe removal.

A.2   File Structure Management Theory of Operations

Devices capable of supporting a Storage Service device can have one or more physical media and or storage arrays formatted with a hierarchical file system. Many physical media and storage arrays support partitioning of the usable capacity into smaller units,

which represent logical partitions or logical volumes, each of which is represented by an instance of the LogicalVolume object. The hierarchical file system can be structured further by creating and removing folders. These folders are modeled as instances of the Folder object residing under the LogicalVolume instance.

A.2.1   File Structure Management

A Storage Service provides one or more Folders that can be configured by the administrator to allow remote file access by specified User Accounts and/or User Groups.

It is likely that the Folders exist within a hierarchical directory structure. However the Storage Service device only defines and administers one level (i.e. top-level) of Folder. Users can further create, administer and delete subfolders and subfolder hierarchies (according to the user permissions on the Folder).

If the administrator wants to manage user-created subfolder hierarchies within a Folder, the administrator cannot use the Storage Service mechanisms; rather the administrator would use a file manager facility (such as Windows Explorer) to access the Folder and its contents. This is possible because the administrator will have remote file access to all Folders created by the Storage Service.

A.2.2   FTP Root File Directory

Storage Services devices that provide FTP protocol for remote file access will have special folders related to the FTP server.

FTP servers have a concept of a root directory. In the Storage Service, this is administered by the StorageService.{i}.FTPServer.AnonymousUser.StartingFolder. The FTPServer.AnonymousUser cannot be enabled until StorageService.{i}.FTPServer.-AnonymousUser.StartingFolder references an existing Folder object. This reference overrides any existing security on the referenced Folder for the Anonymous user only.

A.3   Access and Security Theory of operation

Access to the Storage Service can be via Network Protocols (typically found on the LAN) and Remote Access Protocols (typically used for access over the WAN). Each supported protocol can be enabled or disabled entirely from access to the Storage Service.

A.3.1   Security through UserAccess and GroupAccess profiles

Security for device access and User access is optionally provided through the UserAccess and GroupAccess profiles. Both utilize the same username and password on a per user basis. The username and password are stored in the UserAccount object, which allows multiple users to be defined on the system. With GroupAccess profile support each UserAccount can be associated with a UserGroup, which allows multiple Users the same folder permissions within a UserGroup. When a user logs in, they might be required to enter a username and password. This entry MUST match at least one of the UserAccount objects. Once the user has passed the authentication, they will be able to manipulate files within the folders that they have permissions to do so. Permissions to view and manipulate files and folders are assigned on a per folder basis (Folder.{i}.Name), each folder containing a list of Users and Groups with permission to access them in Folder.{i}.UserAccess.{i} and Folder.{i}.GroupAccess.{i}.

With this type of access control, users logging in via a Network protocol such as HTTP or FTP will be compared to the same set of User Account objects for authentication (when required) and authorization purposes, in essence using the same set of credentials for all access methods.

A.3.2   Levels of Security

With this Model there are two levels of access. One is access to the device with a Storage Service, and the other is to the actual folders (and files) on the device.

Device Access
In a Storage Service, users can connect through a variety of protocols, including but not limited to Remote Access protocols such as HTTP/FTP, and Network protocols such as NFS/SMB/AFP.

When the user connects via an enabled Network/Remote Access protocol server to the Storage Service with no User Access or Group Access profile support, no authentication is required.

If User Access or Group Access profile is supported and the user connects via an enabled Network protocol, access is granted if any of the following are true:
- The account is valid and enabled
- The account is non-existent and .StorageService.{i}.NetworkServer.NetworkProtocolAuthReq = False

If User Access or Group Access profile is supported and the user connects via an enabled Remote Access protocol access is only granted:

- If the account is valid[2] AND the protocol is allowed for this user (AllowHTTPAccess or AllowFTPAccess).

Folder Access

After the User is granted device access, the user can access folders they are authorized to access. Permissions to view and manipulate files and folders are assigned on a per folder basis (Folder.{i}.Name), each folder containing a list of Users and Groups with permission to access them in Folder.{i}.UserAccess.{i} and Folder.{i}.GroupAccess.{i}. Folder authorization can occur in one or more of the following ways:

1. The folder sets UserAccountAccess to '0' allowing any user from the Network protocols and Remote Access protocols to access this file.
2. The folder sets UserAccountAccess to '3[3]' requiring a user from either the Network protocols or Remote Access protocols to be present and enabled in the Folders.{i}.UserAccess table to access this file.
3. The folder sets UserAccountAccess to '3[4]' requiring a user from either the Network protocols or Remote Access protocols to be present and enabled in the Folders.{i}.GroupAccess table that the UserAccount is a member of through the UserGroupParticipation parameter, to access this file.
4. FTP Anonymous is handled outside of this mechanism and applies to only one top-level folder.

Note : Each UserAccount SHOULD be granted access to at least one account

Figure #1 and Access Table #1 are an example depicting the possible references set by TR-140 and the decision process of the device to determine if the user has access to a folder.

---

[2] If using HTTP the AuthenticationReq = False, the user will not have a valid account
[3] This can be set to '1' requiring only the non Network protocols user to be present in the Folders UserAccess or Group Access tables. This can also be set to '2' requiring only the Network protocols user to be present in the Folders UserAccess or Group Access tables.
[4] This can be set to '1' requiring only the non Network protocols user to be present in the Folders UserAccess or Group Access tables. This can also be set to '2' requiring only the Network protocols user to be present in the Folders UserAccess or Group Access tables.
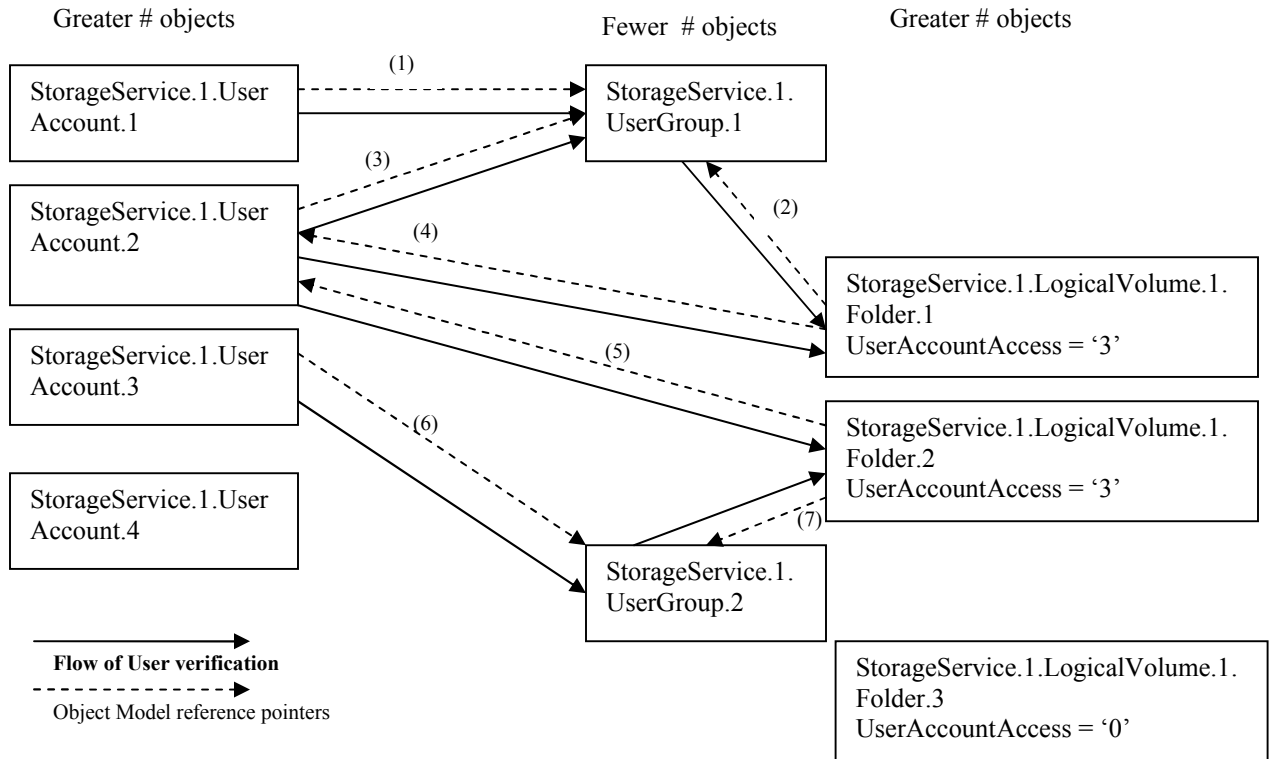
User Access or Group Access profile support

Greater # objects   Fewer # objects   Greater # objects

```
┌─────────────────────┐     (1)      ┌─────────────────────┐
│ StorageService.1.User│- - - - - - →│ StorageService.1.   │
│ Account.1            │─────────────→│ UserGroup.1         │
└─────────────────────┘     (3)      └─────────────────────┘
┌─────────────────────┐   - - - →                    (2)
│ StorageService.1.User│              ┌───────────────────────────────┐
│ Account.2            │     (4)      │ StorageService.1.LogicalVolume.1.│
└─────────────────────┘   - - - →     │ Folder.1                      │
                                      │ UserAccountAccess = '3'        │
┌─────────────────────┐               └───────────────────────────────┘
│ StorageService.1.User│     (5)
│ Account.3            │   - - - →     ┌───────────────────────────────┐
└─────────────────────┘               │ StorageService.1.LogicalVolume.1.│
                            (6)        │ Folder.2                      │
┌─────────────────────┐   - - - →     │ UserAccountAccess = '3'        │
│ StorageService.1.User│               └───────────────────────────────┘
│ Account.4            │     ┌─────────────────────┐   (7)
└─────────────────────┘     │ StorageService.1.   │ - - →
                            │ UserGroup.2         │
                            └─────────────────────┘
                                      ┌───────────────────────────────┐
     ──────────────→                  │ StorageService.1.LogicalVolume.1.│
     Flow of User verification        │ Folder.3                      │
     - - - - - - - →                  │ UserAccountAccess = '0'        │
     Object Model reference pointers  └───────────────────────────────┘
```

Figure #1

Folder Access	How
StorageService.UserAccount.1	
StorageService.1.LogicalVolume.1. Folder.1	(2)StorageService.1.LogicalVolume.1.Folder.1.GroupAccess.1.UserReference = StorageService.1.UserGroup.1
	(1)StorageService.1.UserAccount.1.UserGroupParticipation [includes] StorageService.1.UserGroup.1
StorageService.1.LogicalVolume.1. Folder.3	No Authentication Required.
StorageService.UserAccount.2	
StorageService.1.LogicalVolume.1. Folder.1	(4)StorageService.1.LogicalVolume.1.Folder.1.UserAccess.1.UserReference = StorageService.1.UserAccount.2
StorageService.1.LogicalVolume.1. Folder.1	(2)StorageService.1.LogicalVolume.1.Folder.1.GroupAccess.1.UserReference = StorageService.1.UserGroup.1
	(3)StorageService.1.UserAccount.2.UserGroupParticipation [includes] StorageService.1.UserGroup.1
StorageService.1.LogicalVolume.1. Folder.2	(5)StorageService.1.LogicalVolume.1.Folder.2.UserAccess.1.UserReference = StorageService.1.UserAccount.2
StorageService.1.LogicalVolume.1. Folder.3	No Authentication Required.
StorageService.UserAccount.3	
StorageService.1.LogicalVolume.1. Folder.2	(7)StorageService.1.LogicalVolume.1.Folder.2.GroupAccess.1.UserReference = StorageService.1.UserGroup.2
	(6)StorageService.1.UserAccount.3.UserGroupParticipation [includes] StorageService.1.UserGroup.2
StorageService.1.LogicalVolume.1. Folder.3	No Authentication Required.
StorageService.UserAccount.4	
StorageService.1.LogicalVolume.1. Folder.3	No Authentication Required.
No Account (Network protocol access only if .StorageService.{i}.Access.NetworkProtocolAuthReq[5] = False.	
StorageService.1.LogicalVolume.1.Folder.3	No Authentication Required

Access Table #1

A.3.3 Recommended Object Creation Pattern:

- UserGroup – Only if prefer to add future UserAccounts
- UserAccount – Can only reference UserGroups that are defined.
- Folder – done after initial Users and/or Groups are added,

As a benefit of this design, If UserGroups are used, adding UserAccounts going forward only require a single Add User.

A.4 Data Model Integrity Theory of Operations

A.4.1 Physical Medium

An ACS is not allowed to add or remove an instance of PhysicalMedium as it is not a writable object.

When the device manages the PhysicalMedium instances, it needs to maintain the following data integrity considerations:

When a PhysicalMedium instance is removed, all references to it MUST be removed.

- If a Physical Medium is removed, the device needs to remove any references to it by StorageService.{i}.StorageArray.{i}.PhysicalMediumReference, and depending on the RAID type and number of disks in the RAID this could the affect the status of the StorageArray instance.
- If a Physical Medium is removed, the device needs to remove any references to it by StorageService.{i}.LogicalVolume.{i}.PhysicalReference

A.4.2 Storage Array

A StorageService can have multiple StorageArray instances, each modeling its own RAID configuration. An ACS can create or remove StorageArray instances by utilizing the AddObject and DeleteObject CWMP RPC methods respectively. To create a new storage array an AddObject will be followed by a SetParameterValues RPC invocation. This is required as the StorageArray object will be created in a disabled state. The device supporting the StorageService will not create another new instance or allow the enabling of the storage array until it is provided a unique name, RAID type, required minimum number of physical medium references, and enabled via the SetParameterValues RPC method. To delete a storage array a DeleteObject RPC invocation for the correct StorageArray instance is required.   A disabled StorageArray instance is not removed until a DeleteObject RPC invocation is carried out for it. Once the StorageArray instance is enabled, the Name and RAIDType parameters are immutable while the StorageArray object exists.

When managing the StorageArray instances the device needs to maintain the following data integrity considerations:

When an object reference parameter is being SET, the object being referenced MUST exist

- The StorageArray.{i}.PhysicalMediumReference parameter needs to refer to instances of PhysicalMedium that currently exist in the data model. A PhysicalMedium MUST NOT be referenced by multiple instances of StorageArray.

Referenced objects that are removed.
- If a PhysicalMedium is removed, the status of the StorageArray might be affected.

When a StorageArray instance is removed, all references to it MUST be removed.
- If a Storage Array is removed, the device needs to remove any references to it by StorageService.{i}.LogicalVolume.{i}.PhysicalReference

A.4.3   Logical Volume

A StorageService can have multiple LogicalVolume instances each modeling a logical partition on a PhysicalMedium or StorageArray. An ACS can create or remove LogicalVolume instances by utilizing the AddObject and DeleteObject CWMP RPC methods respectively. To create a new logical partition an AddObject will be followed by a SetParameterValues RPC invocation. This is required as the LogicalVolume object will be created in a disabled state. The device supporting the StorageService will not create or allow the enabling of the logical partition until it is provided a valid PhysicalReference, Name, Capacity, and enabled via the SetParameterValues RPC method. To delete a logical partition a DeleteObject RPC invocation for the correct LogicalVolume instance is required. A disabled LogicalVolume instance is not removed until a DeleteObject RPC invocation is carried out for it. Once the LogicalVolume is enabled, the PhysicalReference, Name and Capacity parameters are immutable while the LogicalVolume instance exists.

When an object reference parameter is being SET, the object being referenced MUST exist
- The LogicalVolume.{i}.PhysicalReference parameter needs to refer to an existing instance of either PhysicalMedium or StorageArray.

Referenced objects that are disabled
- If the object being referenced by the parameter PhysicalReference is disabled then the LogicalVolume will be disabled.

Referenced objects that are removed.
- If the object being referenced by the parameter PhysicalReference is removed then the LogicalVolume will be removed.

A.4.4   Folder

A StorageService can have multiple Folder instances each modeling a top-level folder on a logical partition. An ACS can create or remove Folder instances by utilizing the AddObject and DeleteObject CWMP RPC methods respectively. To create a new folder on the file system an AddObject will be followed by a SetParameterValues RPC invocation. This is required as the Folder object will be created in a disabled state and the device supporting the StorageService will not create the folder on the file system until

it is provided with a locally unique name and enabled via the SetParameterValues RPC method. To delete a folder from the file system a DeleteObject RPC invocation for the correct Folder instance is required. A disabled Folder object is not removed from the file system until a DeleteObject RPC invocation is carried out for it. Once the Folder is enabled, the Name is immutable while the Folder instance exists.

When an object reference parameter is being SET, the object being referenced MUST exist
- If a Folder references a UserGroup or a UserAccount, it MUST already exist.

When a Folder instance is removed, all references to it MUST be removed
- If the Folder being referenced by the parameter StorageService.{i}.FTPServer.AnonymousUser.StartingFolder is deleted, then the anonymous access to the FTPServer is disabled by the device and the parameter will be set to empty.

When a Folder instance is disabled, all references to it MUST be disabled
- If the Folder being referenced by the parameter StorageService.{i}.FTPServer.AnonymousUser.StartingFolder is disabled, then the anonymous access to the FTPServer is disabled by the device.

Referenced objects that are Disabled
- If a UserAccount or UserGroup is disabled, the device MUST NOT remove references to it by all Folders. The folders MUST verify the account is enabled prior to granting access.

Referenced objects that are removed
- If a UserAccount or UserGroup references is removed, those associated accounts no longer have access to the folder.

A.4.5   UserAccount and UserGroup

A StorageService can have multiple UserAccount and UserGroup instances. An ACS can create or remove these instances by utilizing the AddObject and DeleteObject CWMP RPC methods respectively. To create a new UserAccount or UserGroup an AddObject will be followed by a SetParameterValues RPC invocation. This is required as the UserAccount or UserGroup object will be created in a disabled state and the device supporting the StorageService will not create the UserAccount or UserGroup until it is provided with a locally unique name and enabled via the SetParameterValues RPC method. To delete a UserAccount or UserGroup a DeleteObject RPC invocation for the correct instance is required. A disabled UserAccount or UserGroup object is not removed from the file system until a DeleteObject RPC invocation is carried out for it. Once the UserAccount or UserGroup is enabled, the Name is immutable while the instance exists.

When an object reference parameter is being SET, the object being referenced MUST exist
- If a UserAccount references a UserGroup, it MUST already exist.

When a UserAccount or UserGroup object is removed, the references to it MUST be removed.
- If a UserGroup is deleted, the device MUST remove references to it by all UserAccounts and Folders.
- If a UserAccount is deleted, the device MUST remove references to it by all Folders.

A.5   File Retrieval Theory of Operations

Devices capable of supporting a Storage Service are intended to save files to internal drives, which prompts the question of how to actually get the file onto the device. There are several entities that might want to place a file onto a NAS device, such as a Service Provider, a third-party Content Provider, a network-local STB, or even the owner of the NAS Device. In a typical deployment only the Service Provider would have remote CWMP access to the NAS device, allowing it to use the built-in CWMP Download and Upload commands to actually push files to the device or retrieve files from the device. Third-party Content Providers will need to utilize the remote access capabilities that this device provides; whether that is FTP or HTTP depends on the device and can be identified via the Capabilities object. A network-local STB will need to utilize a LAN based access mechanism like UPnP (in the case that the device is a UPnP Media Server). A network-local STB could also use the device's embedded FTP or HTTP servers, depending on the capabilities of the device. The owner of the device will be able to use either LAN access if inside the LAN or FTP/HTTP if outside the LAN depending on the capabilities of the device.

A.6   Thresholding Theory of Operations

Within a Logical Volume there is a limit to the amount of space configured for storage. In storage applications it is useful to provide a warning when the amount of available storage space is approaching zero. The mechanism provided to the ACS here is in the form of a notification (active or passive) set on the ThresholdReached parameter. The implementation of this parameter requires the Storage Service to monitor the UsedSpace within the Logical Volume as it relates to the ThresholdLimit. When UsedSpace + ThresholdLimit < Capacity, ThresholdReached is false.   When UsedSpace + ThresholdLimit >= Capacity, ThresholdReached is True. Note that if a user is actively adding and deleting files when the UsedSpace + ThresholdLimit is near to Capacity the ThresholdReached parameter might toggle back and forth, causing excessive CWMP traffic in the event that active notification is set for the ThresholdReached parameter.

Annex B: RAID Type Descriptions

B.1 Basic RAID Levels

B.1.1   Linear RAID

Linear RAID is a simple grouping of drives to create a larger virtual drive. In linear RAID, the chunks are allocated sequentially from one member drive, going to the next drive only when the first is completely filled. This grouping provides no performance benefit, as it is unlikely that any I/O operations will be split between member drives. Linear RAID also offers no redundancy, and in fact decreases reliability – if any one drive fails, the entire array cannot be used. The capacity is the total of all member disks.

For Linear RAID Level, the minimum number of disks required is 1.

B.1.2   Level 0

RAID level 0, often called "striping", is a performance-oriented striped data mapping technique. That means the data being written to the array is broken down into strips and written across the member disks of the array. This allows high I/O performance at low inherent cost but provides no redundancy. Storage capacity of the array is equal to the total capacity of the member disks.

For RAID Level 0, the minimum number of disks required is 2.

B.1.3   Level 1

RAID level 1, or "mirroring", provides redundancy by writing identical data to each member disk of the array, leaving a "mirrored" copy on each disk. It has been used longer than any other form of RAID. Level 1 Mirroring remains popular due to its simplicity and high level of data availability. Level 1 operates with two or more disks that can use parallel access for high data-transfer rates when reading, but more commonly operate independently to provide high I/O transaction rates. Level 1 provides very good data reliability and improves performance for read-intensive applications but at a relatively high cost. Array capacity is equal to the capacity of one member disk.

For RAID Level 1, the minimum number of disks required is 2.

B.1.4   Level 2

RAID level 2 uses bit-level striping with parity. Data is split at a bit-level and spread over a number of data disks and redundancy disks. The redundant bits are calculated using Hamming codes (which is a form of ECC). On a write, the codes are calculated and written to dedicated redundancy disks alongside the data, which is written to the data disks. On a read, the data and ECC codes are read and the latter is used to verify the

consistency of the former. The ECC codes can correct single-bit errors "on the fly". The total usable capacity varies by implementation. Due to the overhead of calculating and storing codes, RAID 2 is not so popular today.

For RAID Level 2, the minimum number of disks required is 3.

B.1.5   Level 3

Level 3 uses byte-level striping with dedicated parity. Striping is done at a byte level with the same number of bytes in each stripe. The parity information is stored on a dedicated disk. The latter can be a source of bottlenecks especially during random writes. Further, since every write is striped across all the disks, the array could be writing 1 block of data at a time. Thus RAID3 performance depends on the nature of the writes: it performs well on large sequential writes but poorly on small random writes. Array capacity is equal to the capacity of the member disks, minus the capacity of one member disk.

For RAID Level 3, the minimum number of disks required is 3.

B.1.6   Level 4

Level 4 uses block-level striping with parity concentrated on a single disk drive to protect data. It's better suited to transaction I/O rather than large file transfers. Because the dedicated parity disk represents an inherent bottleneck, level 4 is seldom used without accompanying technologies such as write-back caching. RAID 4 differs from RAID 3 only in the size of the stripes sent to the various disks. Array capacity is equal to the capacity of the member disks, minus the capacity of one member disk.

For RAID Level 4, the minimum number of disks required is 3.

B.1.7   Level 5

This is the most common type of RAID. By distributing parity across some or all of an array's member disk drives, RAID level 5 eliminates the write bottleneck inherent in level 4. The only bottleneck is the parity calculation process. With modern CPUs and software-based RAID, that isn't a very big bottleneck. As with level 4, the result is asymmetrical performance, with reads substantially outperforming writes. Level 5 is often used with write-back caching to reduce the asymmetry. Array capacity is equal to the capacity of the member disks, minus the capacity of one member disk.

For RAID Level 5, the minimum number of disks required is 3.

B.1.8   Level 6

RAID 6 uses block-level striping with dual distributed parity. RAID 6 can be thought as RAID 5, but instead calculates 2 sets of parities on the data. Just as in RAID 5, the

parities are distributed across some or all of an array's member disk drives. Dual-distributed parity helps to improve fault tolerance, by making the subsystem tolerant to 2 disk failures. In terms of performance, RAID 6 is slightly worse than RAID 5 due to the usage of dual parity._Array capacity is equal to the capacity of the member disks, minus the capacity of two member disks.

For RAID Level 6, the minimum number of disks required is 4.

B.2   Combination RAID Levels

In addition to the RAID levels described above, arrays can implement a combination of the above RAID levels, thus leveraging the benefits of multiple RAID levels. The next section describes some of the useful combinations of RAID levels.

B.2.1   Level 10

RAID 10 is a combination of RAID 1 (mirroring) followed by RAID 0 (striping), in other words it is a "stripe of mirrors". For RAID 10, a bunch of disks is grouped in pairs, with mirroring within pairs, and then data is striped across these pairs. In terms of failures, the array can tolerate n/2 disk failures across the group, as long as no 2 disks within the same "mirrored set" fail. Array capacity is equal to the capacity of the member disks divided by 2.

For RAID Level 10, the minimum number of disks required is 3.

B.2.2   Level 01

RAID level 01 is combination of RAID 0 (striping) followed by RAID 1 (mirroring), in other words it is a "mirror of stripes". For an array with n disks using RAID 01, data is striped across n/2 disks and the remaining n/2 disks are used as mirrors. In terms of fault tolerance, the array can continue functioning as long as both mirrors across the stripe-sets don't go down. Array capacity is equal to the capacity of the member disks divided by 2.

For RAID Level 01, the minimum number of disks required is 3.

B.2.3   Level 30

Level 30 is formed by striping across a number of RAID 3 sub-arrays. In general, RAID 30 leverages the performance benefit of RAID 0 due to striping, and the fault tolerance benefits of RAID 3. Array capacity is equal to the capacity of the member disks, minus the capacity of one member disk into the number of RAID 3 sets (used for striping). The number of disks MUST be a multiple of 2 integers (one of them 2 or higher and the other 3 or higher).

For RAID Level 30, the minimum number of disks required is 6.

B.2.4   Level 50

RAID 50 uses block striping with distributed parity (RAID 5) along with block striping (RAID 0). In other words, data is striped across a number of RAID 5 sub-arrays. Just like RAID 30, RAID 50 leverages the performance benefit of RAID 0 due to striping, and the fault tolerance benefits of RAID 5. Array capacity is equal to the capacity of the member disks, minus the capacity of one member disk into the number of RAID 5 sets (used for striping).

For RAID Level 50, the minimum number of disks required is 6.

B.2.5   Level 60

RAID 60 combines stripes (RAID 0) across RAID 6 sets. The dual parity of RAID 6 allows for failure of up to two disks. Striping across RAID 6 sets provides the performance benefits of RAID 0. Array capacity is equal to the capacity of the member disks, minus the capacity of one member disk into the number of RAID 6 sets (used for striping).

For RAID Level 60, the minimum number of disks required is 8.

End of Broadband Forum Technical Report TR-140