High-Level API Test Plan

Important notices, IPR statement, disclaimer and Copyright

This chapter contains important information about PRPL and this document (hereinafter 'This PRPL Document').

ABOUT PRPL

The prpl Foundation (PRPL) is a not-for-profit organization which publishes documents including, but not limited to, Requirements, Specifications, Recommendations, API application programming interfaces, and Test Plans.

THIS MAY NOT BE THE LATEST VERSION OF THIS PRPL DOCUMENT

This PRPL Document is the output of the Working Groups of the PRPL and its members as of the date of publication. Readers of This PRPL Document should be aware that it can be revised, edited or have its status changed according to the PRPL working procedures.

THERE IS NO WARRANTY PROVIDED WITH THIS PRPL DOCUMENT

The services, the content and the information in This PRPL Document are provided on an "as is" basis. PRPL, to the fullest extent permitted by law, disclaims all warranties, whether express, implied, statutory or otherwise, including but not limited to the implied warranties of merchantability, non-infringement of third-parties rights and fitness for a particular purpose. PRPL, its affiliates and licensors make no representations or warranties about the accuracy, completeness, security or timeliness of the content or information provided in the PRPL Document. No information obtained via the PRPL Document shall create any warranty not expressly stated by PRPL in these terms and conditions.

EXCLUSION OF LIABILITY

Any person holding a copyright in This PRPL Document, 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 PRPL Document; and (b) any obligation to update or correct this technical report.

THIS PRPL DOCUMENT IS NOT BINDING ON PRPL NOR ITS MEMBER COMPANIES

This PRPL Document, though formally approved by the PRPL member companies, is not binding in any way upon the PRPL members.

Patents essential or potentially essential to the implementation of features described in This PRPL

INTELLECTUAL PROPERTY RIGHTS

COPYRIGHT PROVISIONS

INCORPORATING PRPL DOCUMENTS IN WHOLE OR PART WITHIN DOCUMENTS RELATED TO COMMERCIAL TENDERS

COPYING THIS PRPL DOCUMENT IN ITS ENTIRETY

Executive Summary

This test plan details the comprehensive validation strategy for the High-Level API (HL-API) implementation within the prplWare environment. Our testing prioritizes verifying HL-API component compliance, ensuring adherence to specifications, and proactively identifying defects. This rigorous approach is crucial for delivering a robust, high-quality product that meets the needs of prpl ecosystem stakeholders.

The HL-API is designed as a platform-agnostic interface, primarily for prpl-based platforms but adaptable to others like RDK or third-party systems. To maintain this platform independence and ensure broad applicability, the specific test plans for the HL-API are detailed in a separate document from those covering other network functionalities.

This particular test plan is one component of a broader suite of test plans for formally Certifying implementations of "prplWare and High-Level API".

There are separate test plans for the non-platform-agnostic aspects of the "prplWare and High-Level API" certification. For such certification tests, the goal is to discourage forking by verifying the authenticity and full realization of prplWare components underlying the High-Level API *in the specific implementation under test.*

Revision History

Version	Date	Notes
1.3.0	December 2025	• Added prplWare 4.1 Data Model.
1.2.0	August 2025	• Added prplWare 4.0 Data Model.
1.1.0	April 2025	• Added CWMP Group 2 as an optional feature. • Added support for prplOS 3.2 • Added Executive Summary
1.0	Sept 2023	Initial release from Certification Technical Working Group

Test Organization

This document organizes tests by group based on related test methodology or goals. Each group begins with a brief set of comments pertaining to all tests within that group. This is followed by a series of description blocks; each block describes a single test. The format of the description block is as follows:

Test Label	The Test Label and Title comprise the first line of the test block. The Test Label is composed of the short test suite name, the group number, and the test number within the group, separated by periods.
Purpose	The Purpose is a short statement describing what the test attempts to achieve. It is usually phrased as a simple assertion of the feature or capability to be tested.
Reference	The References section lists cross-references to the specifications and documentation that might be helpful in understanding and evaluating the test and results.
Functionality Tag	The functionality tag indicates whether the test is mandatory or optional as defined in prpl Certification Guide.
Procedure	The procedure indicates the steps, in order, taken to perform the test. Actions are steps taken by the test tool. Expected Behavior is what a device under test is required for passing the test.

Table of Contents

Table of Contents

Results

Each test in the High-Level API (HL-API) Test Plan must have a pass and fail metric, per Section 5.2 of the prpl Certification Program Guide. Test cases require that each test metric pass for all the required parameters. If one parameter in the data model fails a metric in the test case, the entire test case fails.

The HL-API Data Model profile(s) and version(s) used for testing will be recorded and published as part of test results.

Topology
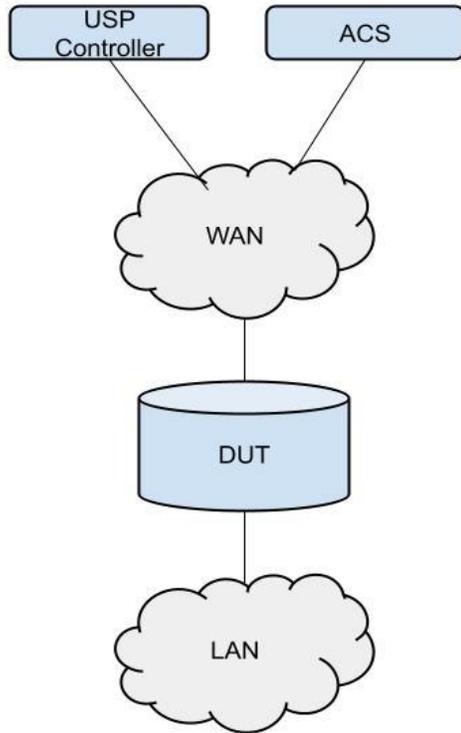
prpl HL-API Data Model Profile

The prpl HL-API data model is required for testing the HL-API.   These profiles are based on the version (A.B) of prplWare (X_PRPLWARE-COM_A_B).

All of the following profiles and associated parameters MUST be supported by the DUT.

- X_PRPL-COM_3_1:1 (prplOS 3.1)
- X_PRPL-COM_3_2:1 (prplOS 3.2)
- X_PRPLWARE-COM_4_0:1 (prplWare 4.0)
- X_PRPLWARE-COM_4_1:1 (prplWare 4.1)

If a device is supporting the Optional Feature of CWMP the following profiles and associated parameters MUST be supported by the DUT.

- X_PRPL-COM_CWMP_3_1:1 (prplOS 3.1)
- X_PRPL-COM_CWMP_3_2:1 (prplOS 3.2)
- X_PRPLWARE-COM_CWMP_4_0:1 (prplWare 4.0)
- X_PRPLWARE-COM_CWMP_4_1:1 (prplWare 4.1)

Additional prpl HL-API Data Model profiles may be added over time, any profile and version listed on the page is considered active and available for us. If a profile or version is removed from the page the HL-API Test Plan MUST be revised.

Optional Features

This is the list of optional features and the associated test cases.

CWMP
- HL-API.2.1 – HL-API.2.5

Group 1: USP

Scope

The following test covers using USP to verify prpl HL-API data model profile.

Overview

The tests in this group verify the DUT properly implements the prpl HL-API and all the associated objects, parameters, events and commands when using USP.

Test HL-API.1.1: Profile GetSupportedDM

Purpose: Verify prpl profile using GetSupportedDM from top level object for the profile.

Reference:
- BBF USP Data Model
- BBF USP TR-369

Functionality Tags: Core Feature

Test Setup: Common Test Setup is used at the beginning of each part.

Procedure:

Step	Action	Expected Behavior
1.	Initiate a GetSupportedDM on the top level object for the profile with the following arguments: ● first _level_only false ● return_commands true ● return_events true ● return_params true	Verify all returned names against the profile definitions. All the required parameters must be present.

Test HL-API.1.2: Profile Parameters Write

Purpose: Verify prpl profile parameters with 'Write' requirement have Writable flag.

Reference:
- BBF USP Data Model
- BBF USP TR-369

Functionality Tags: Core

Test Setup: Common Test Setup is used at the beginning of each part.

Procedure:

Step	Action	Expected Behavior
1.	Initiate a GetSupportedDM on the top level object for the profile with the following arguments: • first_level_only false • return_commands true • return_events true • return_params true	Verify the returned parameter requirements fields match the profile. All the required parameters must be present.
2.		Verify the object is listed as 'W' in the profile definition, make sure the 'Writable' flag is set.

Test HL-API.1.3: Get

Purpose: Verify prpl profile using GET.

Reference:
- BBF USP Data Model
- BBF USP TR-369

Functionality Tags: Core

Test Setup: Common Test Setup is used at the beginning of each part.

Procedure:

Step	Action	Expected Behavior
1.	Initiate a Get on the top level object for the profile.	Verify all returned parameters against the profile definition. All the required parameters must be present.

Test HL-API.1.4: Set

Purpose: Verify prpl profile using SET.

Reference:
- BBF USP Data Model
- BBF USP TR-369

Functionality Tags: Core

Test Setup: Common Test Setup is used at the beginning of each part.

Procedure:

Step	Action	Expected Behavior
1.	Initiate a Get on the top level object for the profile.	
2.	Find all parameters that should be writable based on the prpl profile definition.	
3.	For each parameter, attempt to set its current value using Set.	The DUT must allow all the Sets. Any failed Sets will result in failing the test case.
4.		Report a summary of all Set at the end of the test.

Test HL-API.1.5: Add and Delete

Purpose: Verify prpl profile using Add and Delete on all creatable objects

Reference:
- BBF USP Data Model
- BBF USP TR-369

Functionality Tags: Core

Test Setup: Common Test Setup is used at the beginning of each part.

Procedure:

Step	Action	Expected Behavior
1.	Initiate a Get on the top level object for the profile.	
2.	Find all objects that can be created based on the profile definition.	
3.	For each object, attempt to create a new instance using Add.	The DUT must allow all the Adds. If any Add fails, the test fails.
4.	Initiate a Get on new object instance.	Verify all sub parameters have been created based on the prpl profile.
5.	After creating all objects, delete each object using Delete.	The DUT must allow all the Deletes. If any Delete fails, the test fails.

Group 2: CWMP

Scope

The following test covers using CMWP to verify prpl HL-API data model profile.

Overview

The tests in this group verify the DUT properly implements the prpl HL-API and all the associated objects and parameters when using CWMP.

Test HL-API.2.1: CWMP Profile GetParameterNames

Purpose: Verify prpl CWMP profile GetParameterName from top level.

Reference:
- BBF CWMP Data Model
- BBF CWMP TR-069

Functionality Tags: CWMP

Test Setup: Common Test Setup is used at the beginning of each part.

Procedure:

Step	Action	Expected Behavior
1.	Initiate a GetParameterNames on the top level object using the argument of NextLevel=true.	
2.	Continue to send GetParameterNames or every partial path that is returned until all parameter names have been discovered.	Verify all returned names against the profile definitions. All the required parameters must be present.

Test HL-API.2.2: CWMP Profile Parameters Write

Purpose: Verify prpl CWMP profile parameters with `Write` requirement have Writeable flag.

Reference:
- BBF CWMP Data Model
- BBF CWMP TR-069

Functionality Tags: CWMP

Test Setup: Common Test Setup is used at the beginning of each part.

Procedure:

Step	Action	Expected Behavior
1.	Initiate a GetParameterValues on the top level object for the profile.	Verify the returned parameter requirements fields match the profile. All the required parameters must be present.
2.		Verify the object is listed as 'W' in the profile definition, make sure the 'Writable' flag is set.

Test HL-API.2.3: GetParameterValues

Purpose: Verify prpl CWMP profile using GetParameterValues RPC.

Reference:
- BBF CWMP Data Model
- BBF CWMP TR-069

Functionality Tags: CWMP

Test Setup: Common Test Setup is used at the beginning of each part.

Procedure:

Step	Action	Expected Behavior
1.	Initiate a GetParameterValues on the top level object for the profile.	Verify all returned parameters against the profile definition. All the required parameters must be present.

Test HL-API.2.4: SetParameterValues

Purpose: Verify prpl CWMP profile using SetParametersValues RPC.

Reference:
- BBF CWMP Data Model
- BBF CWMP TR-069

Functionality Tags: CWMP

Test Setup: Common Test Setup is used at the beginning of each part.

Procedure:

Step	Action	Expected Behavior
1.	Initiate a GetParameterValues on the top level object for the profile.	
2.	Find all parameters that should be writable based on the prpl CWMP profile definition.	
3.	For each parameter, attempt to set its current value using Set.	The DUT must allow all the Sets. Any failed Sets will result in failing the test case.
4.		Report a summary of all Set at the end of the test.

Test HL-API.2.5: AddObject and DeleteObject

Purpose: Verify prpl CWMP profile using AddObject and DeleteObject on creatable object.

Reference:
- BBF CWMP Data Model
- BBF CWMP TR-069

Functionality Tags: CWMP

Test Setup: Common Test Setup is used at the beginning of each part.

Procedure:

Step	Action	Expected Behavior
1.	Initiate a GetParameterValues on the top level object for the profile.	
2.	Find all objects that can be created based on the profile definition.	
3.	For each object, attempt to create a new instance using AddObject.	The DUT must allow all the AddObject. If any AddObject fails, the test fails.
4.	Initiate a GetParameterValues on new object instance.	Verify all sub parameters have been created based on the prpl CWMP profile.
5.	After creating all objects, delete each object using DeleteObject.	The DUT must allow all the DeleteObject. If any DeleteObject fails, the test fails.

References

- BBF USP Data Model
  - https://usp-data-models.broadband-forum.org
- BBF USP TR-369
  - https://usp.technology/specification/index.html
- BBF CWMP Data Model
  - https://cwmp-data-models.broadband-forum.org
- BBF CWMP TR-069
  - https://www.broadband-forum.org/pdfs/tr-069-1-6-1.pdf