Research & Vehicle Technology
"Infotainment Systems Product Development"

Feature – IP Pass Through Client

Infotainment Subsystem Part Specific Specification (SPSS)

Version 1.1

UNCONTROLLED COPY IF PRINTED

Version Date: September 4, 2018

FORD CONFIDENTIAL

Revision History

Date	Ver	Notes	
April 3, 2018	**1.0**	**Initial Release**	
September 4, 2018	**1.1**	**Updated Release**	
	IPPT-SD-REQ-304274/B-Application IPPT Request		MBORREL4: Updated diagram
	STR-523925/B-Requirements		MBORREL4: Added REQ-322262, REQ-322263
	IPPT-REQ-304250/B-IPPT FTCP response		MBORREL4: Updated content (added Error codes table)
	IPPT-REQ-304251/B-IPPT FTCP response parameters		MBORREL4: Updated table
	IPPT-REQ-322262/A-Token revocation request		MBORREL4: New req.
	IPPT-REQ-322263/A-Token revocation response		MBORREL4: New req.

Table of Contents

1 Overview

IP Based Pass through works with Wireless Interface Router in the ECG, TCU, SYNC and with an authorization/policy management application in the Ford IPPT Cloud. These will be the key components to provide session authorization, management, and authentication for this service. Once implemented, this service can be used by any Ethernet connected service within Cluster, ADAS, VDS etc. that requires an off board connection to a 1st, 2nd, or third party destination address.

1.1 Terminology and Abbreviations

The following table lists terminologies that are used in this document along with a brief description.

Term	Description
CAN	Controller Area Network
ECG	Enhanced Central Gateway
ECU	Electronic Control Unit
FNV	Fully Networked Vehicle
FTCP	Ford Telematics Control Protocol
GRE	Generic Routing and Encapsulation
HMI	Human Machine Interface
IPC	Inter Process Communication
IPPT	IP Pass Through
SDN	Software Delivery Network
SoA	Service Oriented Architecture
TCU	Telematics Control Unit
WIR	Wireless Interface Router
WLAN	Wireless Local Area Network

2 Architectural Design

2.1 IPPT-CLD-REQ-304022/A-IP Pass Through Server

The IP Pass Through Server (IPPTServer) is responsible for the tasks listed below:

- Interfaces with IPPTClient to receive application requests
- Interfaces with IPPTOffBoardClient to request and receive tokens
- Performs transaction logging for diagnostics purposes

Please review the implementation guide/ block diagram to locate the IPPTServer class.

2.2 IPPT-CLD-REQ-304020/A-IP Pass Through Client

The IP Pass Through Client (IPPTClient) is responsible for the tasks listed below:

- Interfaces with applications via WIR to receive requests and provide responses
- Interfaces with IPPTServer to receive token response from IPPTOffBoardClient

Please review the implementation guide/ block diagram to locate the IPPTClient class.

2.3 IPPT-CLD-REQ-304238/A-IP Pass Through OffBoard Client

The IP Pass Through OffBoard Client (IPPTOffBoardClient) is responsible for the tasks listed below:

- Interfaces with IPPTServer to receive token requests
- Performs policy validations
- Interfaces with 3rd party to receive the authentication tokens

Please review the implementation guide/ block diagram to locate the IPPTOffBoardClient class.

2.4 Physical Mapping of Classes

The table below shows an example of how the logical classes that make up the IPPT feature may be mapped into physical modules. This mapping example is specific to the FNV2 architecture and does not necessarily carryover to other carlines or vehicle architectures.

Logical Class	Physical Module (ECU)
IPPTServer	ECG
IPPTClient	ECG, SYNC, TCU
IPPTOffBoardClient	Ford Cloud or 3rd Party

2.5 IPPTClient Interface

2.5.1 IPPT-IIR-REQ-304023/A-IPPTClientInterface_Tx

The IPPTClientInterface_Tx represents all the IPPT feature related signals sent by the IPPTClientInterface object. The below table represents the mapping of the logical signal names (as described in this specification) to the global GSDB signal names.

Logical Signal Name	Parameter Name	GSDB Signal Name
AutosarNM	Control	TCU_AutoSarNMControl APIM_AutoSarNMControl
	NodeID	TCU_AutoSarNMNodeId APIM_AutoSarNMNodeId
	NMReserved1	TCU_AutoSarNMReserved1 APIM _AutoSarNMReserved1
	NMReserved2	TCU_AutoSarNMReserved2 APIM _AutoSarNMReserved2
	NMReserved3	TCU_AutoSarNMReserved3 APIM _AutoSarNMReserved3
	NMReserved4	TCU_AutoSarNMReserved4 APIM_AutoSarNMReserved4
	GWNMProxy	TCU_GWNMProxy APIM_GWNMProxy
	GWOnBoardTester	TCU_GWOnBoardTester APIM_GWOnBoardTester

Note: GSDB signal names are reference only. The Global Signal Database (GSDB) is the master for all signals. If there is a conflict, bring to the module D&R’s attention.

2.5.1.1 MD-REQ-304019/A-AutosarNM

Message Type: Status

Autosar signal used to wake up the CAN bus.

Name	Literals	Value	Description
Control	-	-	-
		0x00-0xFF	
NodeID	-	-	-
		0x00-0xFF	
NMReserved1	-	-	-
		0x00-0xFF	
NMReserved2	-	-	-
		0x00-0xFF	
NMReserved3	-	-	-
		0x00-0xFF	
NMReserved4	-	-	-
		0x00-0xFF	
GWNMProxy	-	-	-
		0x00-0xFF	
GWOnBoardTester	-	-	-
		0x00-0xFF	

2.5.2 IPPT-IIR-REQ-304024/A-IPPTClientInterface_Rx

The IPPTClientInterface_Rx represents all the IPPT feature related signals received by the IPPTClientInterface object. The below table represents the mapping of the logical signal names (as described in this specification) to the global GSDB signal names.

Logical Signal Name	Parameter Name	GSDB Signal Name

Note: GSDB signal names are reference only. The Global Signal Database (GSDB) is the master for all signals. If there is a conflict, bring to the module D&R's attention.

3 Functional Definition

3.1 IPPT-FUN-REQ-304265/A-Local Handler

3.1.1 Requirements

3.1.1.1 IPPT-REQ-304266/A-Local Handler in IPPTClient

Each IPPTClient shall contain a Local Handler.

3.1.1.2 IPPT-REQ-304239/A-IPPTClient receive request

The IPPTClient Local Handler shall be able to receive requests from application via WIR

3.1.1.3 IPPT-REQ-304240/A-Token required field

The IPPTClient Local Handler shall process requests only if Token required field is set to True by the requesting application

3.1.1.4 IPPT-REQ-304241/A-IPPTClient passing request to IPPT Server

The IPPTClient Local Handler shall pass Token requests to the IPPTServer Central Handler

3.1.1.5 IPPT-REQ-304242/A-IPPTClient passing response back to application

The IPPTClient Local Handler shall pass Token response received from IPPTServer Central Handler to application via WIR

3.1.1.6 IPPT-REQ-304243/A-IPPTClient request validation - Error

The IPPTClient Local Handler shall validate the application request and shall return an error to the application via WIR if there is any error in the token request parameter

3.1.1.7 IPPT-REQ-304244/A-IPPTClient request validation – Unique ID

The IPPTClient Local Handler shall validate the application request and shall return an unique ID to the application via WIR if there is no error in the token request parameter
The requesting application shall be able to track the request via the unique ID parameter.

3.1.2 Use Cases

3.1.2.1 IPPT-UC-REQ-304269/A-Token Request From Application

Actors	WIR, IPPTServer, IPPTClient, IPPTOffBoardClient
Pre-conditions	1. Application sends a request to WIR with mentioning Token request as Yes 2. WIR passes the request to IPPTClient 3. The IPPTClient generates unique ID and responds to application 4. The IPPTServer receives the request from IPPTClient and sends the request to IPPTOffBoardClient via FTCP 5. The IPPTOffBoardClient sends the request to end point and receives the token details 6. The IPPTOffBoardClient sends the response as a FTCP message 7. IPPTServer receives the FTCP message, process the same 8. IPPTServer with IPPTClient sends the token response information to the requesting application
Scenario Description	Application requests token
Post-conditions	Application uses the token provided by IPPT to access end point URL
Interfaces	
Notes	Exception use cases 1 - Token request invalid parameters 2 - No token received from IPPTOffBoardClient

3.1.2.2 IPPT-UC-REQ-304270/A-Exception 1 – Token Request Invalid Parameters

Actors	WIR, IPPTClient
Pre-conditions	1. Application sends a request to WIR with mentioning Token request as Yes 2. WIR passes the request to IPPTClient 3. IPPTClient returns an error back to the requesting application
Scenario Description	Application sends a token request with invalid parameters
Post-conditions	The application shall process the error message and resend another request
Interfaces	
Notes	

3.1.2.3 IPPT-UC-REQ-304271/A-Exception 2 – No Token Received From IPPTOffBoardClient

Actors	WIR, IPPTServer, IPPTClient, IPPTOffBoardClient
Pre-conditions	1. Application sends a request to WIR with mentioning Token request as Yes 2. WIR passes the request to IPPTClient 3. The IPPTClient generates unique ID and responds to application 4. The IPPTServer receives the request from IPPTClient and sends the request to IPPTOffBoardClient via FTCP 5. The IPPTOffBoardClient sends the request to end point and receives no response after all retries 6. The IPPTOffBoardClient notifies this back to the vehicle via an FTCP message 7. The IPPTClient notifies the application of the lack of response
Scenario Description	Application requests token but end point (IPPTOffBoardClient) fails to respond
Post-conditions	A defect log is created for the end point URL
Interfaces	
Notes	

3.1.3 White Box View

3.1.3.1 Sequence Diagrams

3.1.3.1.1 IPPT-SD-REQ-304274/B-Application IPPT Request

Constraints

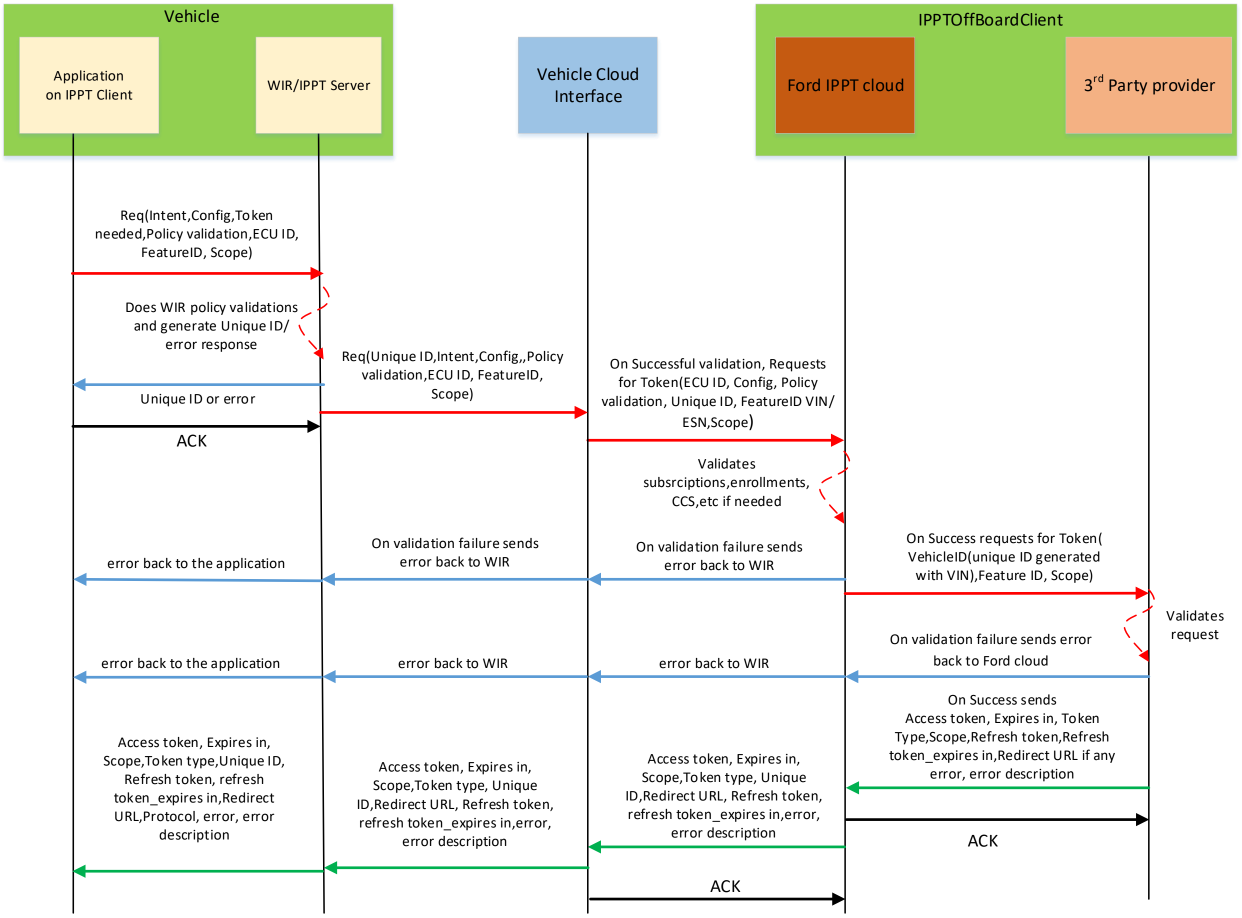
Pre-Condition
Vehicle is ON

Scenarios

Normal Usage
Application requests token

Post-Condition
Application uses the token provided by IPPT to access end point URL

Sequence Diagram


Vehicle
Application on IPPT Client
WIR/IPPT Server
Vehicle Cloud Interface
IPPTOffBoardClient
Ford IPPT cloud
3rd Party provider
Req(Intent,Config,Token needed,Policy validation,ECU ID, FeatureID, Scope)
Does WIR policy validations and generate Unique ID/ error response
Unique ID or error
ACK
Req(Unique ID,Intent,Config,,Policy validation,ECU ID, FeatureID, Scope)
On Successful validation, Requests for Token(ECU ID, Config, Policy validation, Unique ID, FeatureID VIN/ ESN,Scope)
Validates subsrciptions,enrollments, CCS,etc if needed
error back to the application
On validation failure sends error back to WIR
On validation failure sends error back to WIR
On Success requests for Token(VehicleID(unique ID generated with VIN),Feature ID, Scope)
Validates request
On validation failure sends error back to Ford cloud
error back to the application
error back to WIR
error back to WIR
On Success sends Access token, Expires in, Token Type,Scope,Refresh token,Refresh token_expires in,Redirect URL if any error, error description
Access token, Expires in, Scope,Token type,Unique ID, Refresh token, refresh token_expires in,Redirect URL,Protocol, error, error description
Access token, Expires in, Scope,Token type, Unique ID,Redirect URL, Refresh token, refresh token_expires in,error, error description
Access token, Expires in, Scope,Token type, Unique ID,Redirect URL, Refresh token, refresh token_expires in,error, error description
ACK
ACK

3.2 IPPT-FUN-REQ-304260/A-Central Handler

3.2.1 Requirements

3.2.1.1 IPPT-REQ-304314/A-Central Handler in IPPTServer

The IPPTServer shall contain the Central Handler.

3.2.1.2 IPPT-REQ-304245/A-IPPTServer/Client interaction – request

The IPPTServer Central Handler shall be able interact with all IPPTClient Local Handlers and shall be able to receive and process token requests.
The IPPTServer Central Handler design shall be flexible to accommodate requests from additional IPPTClient Local Handlers in the future as well.

3.2.1.3 IPPT-REQ-304246/A-IPPTServer/Client interaction – response

The IPPTServer Central Handler shall be able interact with all IPPTClients and shall be able to send token response received from IPPTOffBoardClient to the IPPTClient Local Handlers.
The IPPTServer Central Handler design shall be flexible to accommodate request /responses from additional IPPTClient Local Handlers in the future as well.

3.2.1.4 IPPT-REQ-304247/A-IPPT FTCP request

The IPPTServer Central Handler shall be able to compose an FTCP message and request for token from the IPPTOffBoardClient
For more details please refer the FTCP protofile

3.2.1.5 IPPT-REQ-304248/A-IPPT FTCP request parameters

The IPPTServer Central Handler shall be able to fill FTCP token request based on the information it received from the application

Request parameters	Data type
Feature ID	String
Unique ID	String
ECU ID	Integer
Policy validation	Boolean
Config key value pair URL key value is a string	String
Scope	String / byte array

For more details please refer the FTCP protofile

3.2.1.6 IPPT-REQ-304249/A-IPPT FTCP request ECU ID

The IPPTServer Central Handler shall be able to fill the appropriate ECU ID based on which ECU the request originated

3.2.1.7 IPPT-REQ-304250/B-IPPT FTCP response

The IPPTServer Central Handler shall be able to decode a token response FTCP message and send the token response to the requesting application via WIR. In case of failure, IPPTServer Central Handler shall send an appropriate error code to the application.

Token Error codes	Error message/ Description
PAYLOAD_ERROR	Mandatory Token fields are missing
INVALID_URL	URL in the Config is not located
SUBSCRIPTION_EXPIRED	Subscription got expired
NOT_ENROLLED	Feature Not enrolled
INVALID_CSS	Invalid CCS

EXPIRED_SUBSCRIPTION_ AND_NOT_ENROLLED	Subscription got expired and Not enrolled
EXPIRED_SUBSCRIPTION_ AND_INVALID_CCS	Subscription expiration and Invalid CCS
NOT_ENROLLED_AND_ INVALID_CCS	Not enrolled and Invalid CCS
EXPIRED_SUBSCRIPTION_ NOT_ENROLLED_AND_ INVALID_CCS	Subscription expiration, Not enrolled and Invalid CCS
POLICY_VALIDATION_SYSTEM_DOWN	CVFMA/SuMo/CCS down – Unable to check subscription, enrollment and/or CCS
PROVIDER_SYSTEM_DOWN	Content Provider system down
IPPT_CLOUD_SYSTEM-DOWN	IPPTOffBoardClient system down
PROVIDER_TOKEN_REFUSAL	Refusal from content provider for a Token
OTHER_ERRORS	All other Errors - will be defined later

For more details about FTCP message please refer the FTCP protofile

3.2.1.8 IPPT-REQ-304251/B-IPPT FTCP response parameters

Response parameters	Data type
Access Token	String
Expires_in	Integer
Token Type	String
Refresh Token	String
Unique ID	String
Error (optional)	String
Error Description (optional)	String
Scope (optional)	String array
Redirect URL(optional)	String
Refresh Token_expires in (optional)	Integer

3.2.1.9 IPPT-REQ-304252/A-IPPTOffBoardClient response correlation with application request

Both IPPTServer Central Handler and IPPTClient Local Handler shall be able to correlate an application request and appropriate IPPTOffBoardClient response by means of using the unique ID which acts as a transaction ID.

3.2.1.10 IPPT-REQ-304253/A-IPPTServer Logging

The IPPTServer shall log all requests and response for a period of IPPT_LOGS_STORAGE_TIME

3.2.1.11 IPPT-REQ-304254/A-On demand diagnostics logs

Upon request from IPPTOffBoardClient the IPPTServer shall push the stored diagnostic logs to the IPPTOffBoardClient via diagnostics framework

3.2.1.12 IPPT-REQ-322262/A-Token revocation request

The IPPTServer shall be able to receive the Token revocation request from IPPTOffBoardClient. Upon receiving the token revocation request command the IPPTServer shall send an ack (command response) with status (in progress / failure) back to the IPPTOffBoardClient. Additionally with help from IPPTClient the IPPTServer shall send the token revocation request to the application based on the app ID present in the token revocation request. The token revocation request shall be per application.

For more information please refer the FTCP protofile.

3.2.1.13 IPPT-REQ-322263/A-Token revocation response

Once application successfully sends the revocation response (success / failure) back to the IPPTClient, the IPPTServer shall convey this information back to the IPPTOffBoardClient via Alert.

Based on the response from the application, the token revocation response Alert shall indicate:

0 – Success (Token successfully revoked for the app ID)
1 – Failure (No response from the application)

If no response is received from the application, the IPPTServer shall retry 5 times and if there is no response shall return “1 – Failure” to the application.

For more information please refer to the FTCP Protofile.

3.2.2 Use Cases

3.2.3 White Box View

3.3 IPPT-FUN-REQ-304267/A-IPPT Performance

3.3.1 Requirements

3.3.1.1 IPPT-REQ-304255/A-IPPT application request processing

The IPPTClient and IPPTServer shall not add any latency with respect to token generation.

The round trip time from receipt of a request from application to composing an FTCP message and sending to IPPTOffBoardClient shall happen within IPPT_REQUEST_PROCESSING_TIME which is 1 millisecond

3.3.1.2 IPPT-REQ-304256/A-IPPT application response processing

The IPPTClient and IPPTServer shall not add any latency with respect to token response.
The round trip time from receipt of a token FTCP message from IPPTOffBoardClient to composing and sending the token information to requesting application in vehicle shall happen within IPPT_RESPONSE_PROCESSING_TIME which is 1 millisecond

3.3.2 Use Cases

3.3.3 White Box View

3.4 IPPT-FUN-REQ-304268/A-IPPT Configuration

3.4.1 Requirements

3.4.1.1 IPPT-REQ-304257/A-Configuration parameters

Parameter	Description	Tuned Value FoE	Metric	Resolution	Tuning Range
IPPT_LOGS_STORAGE_TIME	Time for which logs will be stored at any point on IPPTServer	43200	Minutes	1	0 - 65535

3.4.2 Use Cases

3.4.3 White Box View

4 Appendix: Reference Documents

Reference #	Document Title
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	