Research & Vehicle Technology
"Infotainment Systems Product Development"

Feature – Lincoln Backup Ignition

APIM Infotainment Subsystem Part Specific Specification (SPSS)

Version 2.3

UNCONTROLLED COPY IF PRINTED

Version Date: October 10, 2019

FORD CONFIDENTIAL

Revision History

Date	Version	Notes	
March 30, 2017	**0.1**	**Draft**	
Octobre 19, 2017	**1.0**	**Initial Release**	
November 9, 2017	**1.1**		
	MD-REQ-246273/C-EmbeddedModemReset_Rq	MBORREL4: Changed SYNC Connect to Brand Connect	
	IIR-REQ-258549/B-LBIClient_Rx	cwu3: Revised to add MD-REQ-241972/D-PaakESN_St	
	STR-459086/B-LBIClient General Requirements	cwu3: Changed format. No real design changed	
	PaaK-REQ-270465/C-Payload Parameters Definitions	rpaquet2 - Updated requirement	
	LBIv1-REQ-281410/A-SYNCP Payload Parameters	rpaquet2 - New	
	LBIv1-REQ-281412/A-LBI Notification Payload	rpaquet2 - New	
	LBIv1-FUN-REQ-269571/B-Creating Keypad Code for PaaK Device	cwu3: Deleted LBIv1-REQ-267614/A-PaaK Keypad Code Rules per Feature owner Aaron DeLong request	
	LBIv1-REQ-271249/B-Conditions for Suspend Center Stack HMI	cwu3: Fixed typo. No design changes	
	LBIv1-REQ-260084/B-HMI Display of Valid Entry and Start Engine Instruction	cwu3: Fixed typo for IgnitionStatus_St . No design changes.	
	STR-455084/B-Requirements	cwu3: revised to add LBIv1-REQ-283992/A-HMI Display of Valid Password and Exit Park Instruction	
	LBIv1-REQ-283992/A-HMI Display of Valid Password and Exit Park Instruction	cwu3: New to cover the use case of exiting Secure Idle State when a valid password is accepted.	
June 1, 2018	**2.0**		
	LBI-FRD-REQ-280168/B-Lincoln Backup Ignition	cwu3: Updated to incorporate FNV2 Architecture. Ver 2.0 shall be used as basline for FNV2 programs	
	STR-467819/B-Feature Description	cwu3: Added LBI market name "Backup Starting Passcode "	
	STR-454710/B-Architectural Design	cwu3: Added FNV2 information to Lincoln Backup Ignition System	
	LBI-DOC-417923/B-Lincoln Backup Ignition Physical Mapping of Classes	cwu3: Added FNV2 information	
	LBI-DOC-524814/A- Lincoln Backup Ignition System	cwu3: New to include ECG for FNV2	
	LBI-DOC-417997/B-Lincoln Backup Ignition Logic Method to Physical Signal Translation Table+	cwu3: Revised to add PwPckTq_St	
	LBI-DOC-417997/C-Lincoln Backup Ignition Logic Method to Physical Signal Translation Table+	cwu3: (1)Added PaakCtrlActionCode and PwPckTq_St (2) Added Duplicate and NotUsed_1 for KeyPadCodeProg_St	
	LBI-DOC-417997/D-Lincoln Backup Ignition Logic Method to Physical Signal Translation Table	cwu3: Correct errors for PaakCtrlSearchRslt's encoding names and values	
	IIR-REQ-258549/C-LBIClient_Rx	cwu3:Revised to add MD-REQ-293501/A-PwPckTq_St	
	MD-REQ-241972/E-PaakESN_St	rpaquet2 - Updated BLEMSyncP definition and removed hardware number and Software part number as they are part of the SyncP package	
	STR-459093/B-Lincoln Backup Ignition Logic Method Requirement	cwu3: Revised to delete LBIv1-TMR-REQ-276027/A-T_ReturnToNull	
	LBIv1-REQ-275627/B-Request/Response Return to Null State	cwu3: Changed display time from T_ReturnToNull to at least as long as signal sampling interval	
	LBIv1-REQ-304588/A-Usages of PaakCtrlRKEData in LBI Keypad Code Programming	cwu3: New	
	PaaK-REQ-307226/A-Remote Start and RKE return to Null	rpaquet2 - New	
	STR-458487/B-LBIServer General Requirements	cwu3: Added 8 new requirements: (1)PaaK-REQ-269555/A-BCM - BLEM Communication AES Encryption (2)PaaK-REQ-269556/A-AES Key (3)PaaK-REQ-269557/A-AES Input (4)PaaK-REQ-269558/A-AES Output (5)PaaK-REQ-270046/B-Remote Start and RKE Challenge Response (6)PaaK-REQ-242454/B-BLEM-BCM Interactions (7)PaaK-REQ-270047/A-RKE Queue (8)LBIv1-REQ-312524/A-Preconditions for Response to LBI Requests	

LBIv1-REQ-271440/B-CAK Asignmenet Restriction	cwu3: Fixed typo in Title. No design change.
LBIv1-REQ-271443/B-Keypad Code Transmit Methods	cwu3: (1) Replaced PaaKTrgtActvData_No_Rq (CAN signal name) with PaakTargetRKEData (logic name) (2)Replaced PaaKTrgtActvData_No_Rq (CAN signal name) with PaakTargetRKEData (logic name) (3) Per [LBI.R288.02] added 6th and 7th data parameters for market calls out 7-digit Keypad code
PaaK-REQ-269555/B-BCM - BLEM Communication AES Encryption	rpaquet2 - Updated picture per feature owner
PaaK-REQ-269557/B-AES Input	rpaquet2 - Corrected typo from 0x0E to 0xE0
PaaK-REQ-270046/B-Remote Start and RKE Challenge Response+	rpaquet2 - Removed section number reference and replaced with requirement number
PaaK-REQ-270046/C-Remote Start and RKE Challenge Response	rpaquet2 - Updated 0x0E to 0xE0 typo fix.
PaaK-REQ-242454/C-BLEM-BCM Interactions	rpaquet2 - Replaced reference to section with reference to actual requirement number
STR-459086/C-LBIClient General Requirements	cwu3: Revised to add LBIv1-REQ-271253 and LBIv1-REQ-283990
LBIv1-TMR-REQ-275603/B-T_LBI Wait Popup Display Timer	cwu3: Fixed typo. No real change.
LBIv1-REQ-276915/B-Restrictions Imposed by LBI Enhanced Valet Mode	cwu3: Added exact requirements VS-FUR-REQ-104343 and H31a for clarification. No design changes.
LBIv1-REQ-271253/A-Keypad Code Transmitting Format	cwu3: per [LBI.R379.02], new to provide Keypad code format for TP BackupIngnition_Rq and instruction for 5-digit code implementation.
PaaK-REQ-270465/D-Payload Parameters Definitions+	cwu3: Replaced BLEM with the LBIServer. No design changed
PaaK-REQ-270465/E-Payload Parameters Definitions	rpaquet2 - Updated BLEMProvDID,BPEKHash and ESN fields
LBIv1-REQ-281410/B-SYNCP Payload Parameters	cwu3: Replaced BLEM with the LBIServer. No design changed
STR-459087/B-Security Requirements	cwu3: Revised to update LBIv1-REQ-260159-Generating Condition and Storage Location for Salt
LBIv1-REQ-260159/B-Generating Condition and Storage Location for Salt	cwu3: Clarified the requirement to generate a random 128-bit salt, not randomly generate a 128-bit salt.
LBIv1-REQ-260165/B-No Direct Display of Backup Password on HMI	cwu3: Fixed title. No real change.
LBIv1-REQ-275186/B-SyncP Message Process for SDN	cwu3: Replaced BLEM with the LBIServer. No design changed
STR-454355/B-Requirements	cwu3: (1) Deleted BCM requirement LBIv1-REQ-268182/A (2) Deleted BIv1-REQ-267236/B-Request of Interior Registry KeyFob Search (3) Deleted BIv1-REQ-275622/A-Monitor Interior Registry KeyFob Search Result
LBIv1-REQ-276039/A-Number of Backup Password per PaaK Device	cwu3: New to add an explicit requirement for number of Backup Password permitted per PaaK device. The requirement had been already implemented through TP BackupIgnition_Rsp. No new work required.
LBIv1-REQ-260055/B-Initiate Password Creation after CAK Created	cwu3: Fixed typo at title. No real change.
LBIv1-REQ-264576/B-Trigger Interior Registry KeyFob Search for Paak without Password	cwu3: (1) Deleted BLEM from Title (2)Replaced LBIv1-REQ-267236-Interior Registry KeyFob Search with LBIv1-FUN-REQ-302285- LBI KeyFob Search
LBIv1-REQ-260061/B-Response of Paak w/o Password	cwu3: (1)per [LBI.R006.02], updated Note to clarify PhoneName (2) Correct the KeyFob status definition
LBIv1-REQ-260164/B-Backup Password Minimum Length Requirement	cwu3: Added new limit for FNV2 programs
LBIv1-REQ-260067/B-Trigger Interior Registry KeyFob Search for Password Response	cwu3:Replaced with REQ-267236-Interior Registry KeyFob Search with LBIv1-FUN-REQ-302285- LBI KeyFob Search
LBIv1-REQ-263584/B-Response of No Device for Password Transmit	cwu3: Added LBIv1-REQ-302288-Authentication of KeyFob Search Response for KeyFob search status
LBIv1-REQ-275718/B-Password Response	cwu3:Replaced LBIv1-REQ-267236-Interior Registry KeyFob Search with LBIv1-FUN-REQ-302285- LBI KeyFob Search
LBIv1-FUN-REQ-269571/C-Creating Keypad Code for PaaK Device	cwu3: Deleted LBIv1-REQ-267613/A due to the fact that REQ-267614 was deleted
STR-456425/C-Requirements	cwu3: Revised to add LBIv1-REQ-271253/A-Keypad Code Transmitting Format
LBIv1-REQ-264859/B-Conditions of Triggering Keypad Code Creation Request	cwu3: Deleted REQ-267614 from the requirement per Aaron DeLong request
LBIv1-REQ-271445/B-Response of Duplicate LBI Keypad Code	cwu3: Deleted 2nd portion of requirement (moved to LBIv1-REQ-304593)
LBIv1-REQ-264864/B-HMI Display of Keypad Code Creation	cwu3: Added response of duplicate keypad code

LBIv1-SD-REQ-274272/B-Creating Keypad Code for PaaK Device	cwu3: Revised to correct SD-319644
STR-454361/B-Requirements	cwu3: Deleted BCM requirement LBIv1-REQ-268182/A
LBIv1-REQ-271264/B-Trigger Interior Registry KeyFob Search of Paak with Password to Delete Reset	cwu3:Replaced LBIv1-REQ-267236-Interior Registry KeyFob Search with LBIv1-FUN-REQ-302285-LBI KeyFob Search
LBIv1-REQ-267239/B-Status for PaaKs with Passwords	cwu3: Added "by executing LBIv1-FUN-REQ-302285" for KeyFob status
STR-455236/B-Requirements	cwu3: (1)Deleted BCM requirement LBIv1-REQ-268182/A (2) Deleted LBIv1-REQ-267236-Interior Registry KeyFob Search
LBIv1-REQ-270384/B-Challenge Request	cwu3: Replaced BackupIgnition_Rsp with BackupIgnition_Rq
LBIv1-REQ-267241/B-Reset Password Response	cwu3: Replaced LBIv1-REQ-267236-Interior Registry KeyFob Search with LBIv1-FUN-REQ-302285-LBI KeyFob Search
STR-454364/B-Requirements	cwu3 : (1) Deleted BCM requirement LBIv1-REQ-268182/A (2) Deleted LBIv1-FUN-REQ-302285-LBI KeyFob Search
LBIv1-REQ-271447/B-Trigger Registry KeyFob Search for Enter Enhanced Valet Mode	cwu3: Replaced LBIv1-REQ-267236-Interior Registry KeyFob Search with LBIv1-FUN-REQ-302285-LBI KeyFob Search
LBIv1-REQ-276997/B-Criteria of Reporting Failed Enhanced Valet Password Creation	cwu3: Fixed missing word. No real change.
LBIv1-REQ-304593/A-Actions after Duplicate LBI Enhanced Valet Keypad Code	cwu3: New. Derived from Valet use case of LBIv1-REQ-271445
STR-454358/C-Requirements	cwu3: Revised to (1)add LBIv1-REQ-293516/A-HMI Display of Valid Entry and Out of Park Instruction (2)delete BCM requirement LBIv1-REQ-260075 (3)add PaaK-REQ-270039-Passive Challenge Data Response
LBIv1-REQ-260084/C-HMI Display of Valid Entry and Start Engine Instruction	cwu3: per [LBI.R383.01] changed HMI display condition and dismissed conditions. Also added Notes to explain the applicable use cases
LBIv1-REQ-271254/B-Positive Response to Crypto Searches	cwu3: Added information about the calculation of PaakCtrlPassiveData ([Response])
PaaK-REQ-270039/B-Passive Challenge Data Response	rpaquet2 - Removed reference to setion number and replaced with actual requirement number
LBIv1-REQ-268502/B-Response of Lockout	cwu3: (1) Per [LBI.R407.01], added requirement for the LBIServer to ignore LBI requests during lockout period (2) Fixed missing word "lockout".
LBIv1-SD-REQ-262294/B-Starting Vehicle with LBI Password-Happy Path	cwu3: Updated diagram to include "Index=0" for all other PaaKCtrlFoundIndex's that are not the one in use
STR-454370/B-Requirements	cwu3: Deleted BCM requirement LBIv1-REQ-268182/A
LBIv1-REQ-267214/B-Trigger Interior Registry KeyFob Search for Exit Valet Mode	cwu3: Replaced LBIv1-REQ-267236-Interior Registry KeyFob Search with LBIv1-FUN-REQ-302285-LBI KeyFob Search
LBIv1-REQ-275757/B-HMI Display of Successful Valet Password Deletion	cwu3: Fixed format. No real change.
STR-454373/B-Requirements	cwu3: Revised to add LBIv1-REQ-276040 and LBIv1-REQ-305614
LBIv1-REQ-276040/A-No LBI Passwords Deletion Notification after Master Reset or Paak Reset	cwu3: New per [LBI.382.01].
LBIv1-REQ-271433/B-Initiate Delete Keypad Code when CAK Revoked	cwu3: Fixed missing word. No real change.
LBIv1-FUN-REQ-258456/B-Transitioning Vehicle from Remote Start State with Backup Password	cwu3:Changed title from "Non Motive to Motive State" to "Remote Start Stage"
STR-454380/B-Use Cases	cwu3: Revised to add LBIv1-UC-REQ-294899
LBIv1-UC-REQ-260140/B-Transitioning Vehicle from Remote Start State with Backup Password	cwu3: (1) Changed title from "Non Motive to Motive State" to "Remote Start State" (2) replace 30 seconds and 20 seconds with timers T_Password Entry Screen Inactive Timer and T_Push to Start Timer respectively (3) Corrected HMI display content (4) Added 4th item for Exception Lists
LBIv1-UC-REQ-294899/A-Dismissing HMI Exit Park Instruction when Engine Shuts Down	cwu3: New to handle the use case where the use shuts down engine
STR-454379/B-Requirements	cwu3: Deleted BCM requirement LBIv1-REQ-260138/A
LBIv1-REQ-283992/B-HMI Display of Valid Password and Exit Park Instruction	cwu3: Per [LBI.R325.04] changed HMI display condition and dismissed conditions. Also added Notes to explain the applicable use cases
LBIv1-UC-REQ-276157/B-Failing to Enter Programming Mode	cwu3: Added Note for clarification . No real change.
LBIv1-UC-REQ-275728/B-Failing to Add or Delete Keypad Code	cwu3: Added Note for clarification . No real change.
LBIv1-UC-REQ-276158/B-Adding or Deleting Keypad Code	cwu3: Added Note for clarification . No real change.
STR-457495/B-Requirements	cwu3: Deleted BCM requirements (1)LBIv1-REQ-264878/A

		(2)LBIv1-REQ-270516/A (3)LBIv1-REQ-274284/A (4)LBIv1-REQ-275753/A (5)LBIv1-REQ-270059/A (6)LBIv1-TMR-REQ-270057/A (7)LBIv1-REQ-264863 (8)LBIv1-TMR-REQ-270519/A
	LBIv1-REQ-304585/A-PaaK RKE Interface for LBI Keypad Code Programming	cwu3: New
	LBIv1-REQ-270265/B-Request to Enter Keypad Code Programming Mode	cwu3: Fixed format and added clarification. No design change
	LBIv1-REQ-304565/A-Designation of RKE Challenge Response in AES Output	cwu3: New per [LBI.R385.001]
	LBIv1-REQ-304562/A-Compute an AES Output for RKE Challenge Response in Entering Programming Mode	cwu3: New per[LBI.R385.001]
	LBIv1-REQ-304563/A-AES Key and AES Input for Computing an AES Output for RKE Challenge Response	cwu3: New per[LBI.R385.001]
	LBIv1-REQ-304560/A-Add Request of Entering Programming Mode to PaaK RKE Queue	cwu3: New per[LBI.R396.001]
	LBIv1-REQ-304561/A-Precondition of Sending Request to Enter Programming Mode	cwu3: New per[LBI.R396.001]
	LBIv1-REQ-264862/B-Request to Store Keypad Code	cwu3: Added reference of LBIv1-REQ-271443 to provide keypad code transmit format details
	LBIv1-REQ-275752/B-Request to Delete Keypad Code	cwu3: Correct description error of KeyPadCodeProg_Rq(Delete), from the store request to the delete request. No real design change.
	LBIv1-REQ-277531/B-Resend Keypad Code Request Requirement	cwu3: Added details about where to start the resend process
	LBIv1-REQ-275827/B-N_LBINumberOfRetries	cwu3: Fixed typo. No real change.
	LBIv1-FUN-REQ-264539/B-Exiting Secure Idle State with Backup Password	cwu3: Revised to add new Use Case
	STR-454383/B-Use Cases	cwu3: Revised to add LBIv1-REQ-260084/C-HMI Display of Valid Entry and Start Engine Instruction
	LBIv1-UC-REQ-260141/B-Exiting Secure Idle State with Backup Password when Engine Off	cwu3: (1) Changed ignition Run to Secure Idle active stage with engine off (2) Replaced 20 seconds and 30 seconds with T_Push to Start Timer and T_Password Entry Screen Inactive Timer, respectively (3) Added LBI notification and HMI Popup dismissing to Post-Conditions (4) Added three List of Exception Use Cases
	LBIv1-UC-REQ-294907/A-Exiting Secure Idle State with Backup Password when Engine Running	cwu3: New
	LBIv1-UC-REQ-294906/A-Dismissing HMI Start Engine Instruction when Entering NonMotive Mode	cwu3:New
	STR-455084/C-Requirements	cwu3: Deleted BCM requirement LBIv1-REQ-264540/A
July 24, 2018	**2.1**	
	LBI-FRD-REQ-280168/B-Lincoln Backup Ignition+	cwu3: Updated to incorporate FNV2 Architecture. Ver 2.0 shall be used as basline for FNV2 programs
	LBI-FRD-REQ-280168/C-Lincoln Backup Ignition	cwu3: Revised for ver 2.1 release
	STR-458487/C-LBIServer General Requirements	cwu3: Added new requirements
	LBIv1-REQ-318311/A-Configurable Parameter for Keypad Code	cwu3: New per[LBI.R288.03] and [LBI.R379.02]
	LBIv1-REQ-271443/C-LBIServer Keypad Code Transmit Methods	cwu3: (1) Added requirement to check Keypad code configuration parameter before arrange keypad code to data string (2) Added LBIServer to title for clarification
	LBIv1-REQ-275544/B-Valet Password Uniqueness	cwu3: Replaced 8-digit with random-numeric per [LBI.R289.03]
	STR-459086/D-LBIClient General Requirements	cwu3: Added new requiremet
	LBIv1-REQ-264925/B-Operation Modes for LBI Functions and Menus	cwu3: Added CCS as one of conditions per [LBI.R381.02]
	LBIv1-REQ-271253/B-LBIClient LBI Keypad Code Transmitting Format	cwu3: (1)Added “LBIClient LBI” to title to distinguish from the similar requirement for the LBIServer (2)Added two notes to Example 2 for clarifications. No real content changed
	STR-458488/B-Notification Requirements	cwu3: Added new and revised requirements
	LBIv1-REQ-321853/A-KeyID Parameter Definitions and Transmit Method in LBI Alerts	cwu3: New for KeyID

PaaK-REQ-270465/F-Payload Parameters Definitions (PaaK/LBI)	rpaquet2 - Added clarification about Etype and service type
LBIv1-REQ-281412/B-LBI Notification Payload for BLEMSyncPPacket	cwu3: (1) Added BLEMSyncPPacket to title (2) Added Valet examples
STR-459087/C-Security Requirements	cwu3: Added new requirements for Keypad code
LBIv1-REQ-260168/B-Password Transmission Mechanism	cwu3: Added note and calculation example per[LBI.R151.02]
LBIv1-REQ-260176/B-LBIClient Programmed Hash Requirement	cwu3: Added note per [LBI.R158.02]
LBIv1-REQ-260177/B-LBIClient Authentication Hash Requirement	cwu3: Added note and calculation example per[LBI.R159.02]
LBIv1-REQ-322144/A-LBI Keypad Code Length Definition	cwu3: New per[LBI.R409.01]
LBIv1-REQ-322204/A-LBI Keypad Code Composition Restriction	cwu3: New per[LBI.R409.01]
STR-459425/B-Functional Definition	cwu3: Revised to add BIv1-FUN-REQ-318631/A-Suspending LBI when CCS is off
LBIv1-REQ-260164/C-Backup Password Minimum Length Requirement	cwu3: per [LBI.146.03] Added minimum length for FNV2 program when only numbers are used
LBIv1-REQ-264928/B-Backup Password Maximum Length Requirement	cwu3: Rephrased for clarification. No real change.
LBIv1-REQ-260065/B-Check Entered Password against Password Rules	cwu3: Deleted unnecessary text "-R0146.01". No real change.
LBIv1-REQ-265042/B-Backup Passwords Strength Definitions	cwu3: Added more rules per [LBI.R178.02]
LBIv1-REQ-267620/B-Check Password Uniqueness	cwu3: Revered Reset1 and Reset2 required devices
LBIv1-REQ-270515/B-Notification of Successful Password Creation to LBIClient2	cwu3:Deleted ItemIndex= 0x01 and KeyID = [Key ID] per [LBI.R231.03]
STR-456424/B-Use Cases	cwu3: New per [LBI.R379.02] and [LBI.R409.01]
LBIv1-UC-REQ-323248/A-Entering Too Little Digits for Keypad Code	cwu3: New per [LBI.R379.02]
LBIv1-UC-REQ-323249/A-Entering Too Many Digits for Keypad Code	cwu3: New per [LBI.R379.02]
LBIv1-UC-REQ-323257/A-Entering Unacceptable Keypad Code	cwu3: New per [LBI.R409.01]
STR-456425/D-Requirements	cwu3: Added new requirements
LBIv1-REQ-260073/B-HMI Display for LBI Keypad Code Entry Screen	wu3: Replaced PaaK with LBI at title. No real content changed.
LBIv1-REQ-318317/A-HMI Functions for LBI Keypad Code Entry Screen	cwu3: New per [LBI.R379.02] and [LBI.R409.01]
LBIv1-REQ-318683/A-LBIClient Error Handling Strategies for Mismatched LBI Keypad Code	cwu3: New per [LBI.R379.02]
LBIv1-REQ-264858/B-HMI Solicit Keypad Code Twice	cwu3: (1)Changed rule-compliant to length-compliant and added REQ number (2)Added composition compliant with REQ number
LBIv1-REQ-267283/B-Request Keypad Code Creation	cwu3: Added LBIv1-REQ-271253 for Keypad Code data string instruction. No real content changed.
LBIv1-REQ-271265/B-Notification of Successful Password Deletion to LBIClient2	cwu3: per[LBI.206.04] (1)Deleted ItemIndex= 0x01 and KeyID = [Key ID] (2)Clarified KeyID
LBIv1-REQ-264866/B-Track Invalid Password Entering Attempts	cwu3: Fixe typo from "Valet Delete Challenge Response" to "Valet Start Challenge Response" in the last paragraph
LBIv1-UC-REQ-260115/B-Generating Valet Backup Password and Keypad Code with PaaK	cwu3: Deleted 8-digit per[LBI.R237.03]. No real change.
LBIv1-UC-REQ-276994/B-Generating Valet Backup Password and Keypad Code without PaaK	cwu3: Deleted 8-digit per [LBI.R237.03]. No real change.
STR-454364/C-Requirements	cwu3: Revised to : (1) Added LBIv1-REQ-317825/A-Length of Enhanced Valet Password
LBIv1-REQ-264929/B-Conditions to Generate Enhanced Valet Code with Paak	cwu3: New per [LBI.237.03].
LBIv1-REQ-264911/B-HMI Display with No Device	cwu3: Fixed Format. No real change
LBIv1-REQ-260112/B-Condition to Generate Enhanced Valet Code with No Device	cwu3:Replaced 8-digit with LBIv1-REQ-317825 per [LBI.R237.03].
LBIv1-REQ-317825/A-Length of Enhanced Valet Password	cwu3: New per [LBI.R237.03].
LBIv1-REQ-268183/B-Valid Enhanced Valet Password Rules	cwu3: Deleted 8-digit per [LBI.R237.03]
LBIv1-REQ-317844/A-Enhanced Valet Password Usages	cwu3: New per [LBI.R237.03].

	LBIv1-REQ-268184/B-Track PaaKs for Enhanced Valet Password Alerts	cwu3: Deleted 8-digit per [LBI.R237.03]
	LBIv1-REQ-264904/B-Initiate Keypad Code Storing for Enhanced Valet Password	cwu3: Deleted 8-digit per [LBI.R237.03]
	LBIv1-REQ-264908/B-Store Enhanced Valet Password	cwu3: Added missing requirement number 270053
	LBIv1-REQ-271426/B-Notification of Successful Enhanced Valet Password Creation to LBIClient2	cwu3:(1) Deleted ItemIndex= 0x01-0x04 and KeyID = [Key ID] per [LBI.R241.03]
	LBIv1-REQ-271428/B-Bluetooth Notification of Successful Enhanced Valet Password Creation	cwu3: Deleted 8-digit per [LBI.R237.03]
	LBIv1-REQ-260114/B-HMI Display of Active Enhanced Valet Mode	cwu3: (1) Replaced REQ-267195 with REQ-317844 (2) Added LBIv1-REQ-317825 and note to emphasize that Enhanced Valet password could have different lengths for different markets
	LBIv1-REQ-260076/B-Conditions to Display Password Entry Screen for Starting Vehicle	cwu3: Added CCS as one of conditions per [LBI.R322.03]
	LBIv1-REQ-270547/B-Notification of Valid Backup Password Usage to LBIClient2	cwu3: per [LBI.R200.03] and [R252.04] (1) Deleted ItemIndex= 0x01-0x04 and KeyID = [Key ID] (2) Clarified what KeyID is
	LBIv1-REQ-271255/B-Notification of Lockout to LBIClient2	cwu3: per [LBI.R308.02] and [R315.02] (1) Deleted ItemIndex= 0x01-0x04 and KeyID = [Key ID] (2) Clarified what KeyID is
	LBIv1-REQ-264922/B-Notification of Valet Password Deletion to LBIClient2	cwu3: per [LBI.R255.03] and [R269.04] (1) Deleted ItemIndex and KeyID = [Key ID] (2) Clarified what KeyID is
	LBIv1-UC-REQ-318652/A-HMI Display of CCS Impacts on LBI	cwu3: New per [LBI.R404.01]
	LBIv1-UC-REQ-318653/A-Starting Vehicle without KeyFob/PaaK when CCS off	cwu3: New per [LBI.R381.02]
	LBIv1-REQ-318628/A-LBI HMI Display for CCS off Impacts on LBI	cwu3: New per [LBI.R404.01]
	LBIv1-REQ-318630/A-Suspend LBI when CCS off	cwu3: New per [LBI.R381.02]
	STR-457495/C-Requirements	cwu3: Added new DTC requirement
	LBIv1-REQ-270265/C-Request to Enter Keypad Code Programming Mode	cwu3: Deleted 8-digit per [LBI.237.03]
	LBIv1-REQ-264862/C-Request to Store Keypad Code	cwu3: (1) Replaced 5-digit with LBIv1-REQ-317844 (2) Replaced 8-digit with random number per [LBI.237.03]
	LBIv1-REQ-318296/A-DTC for Mismatch Keypad Code and Configuration Parameter	cwu3: New DTC for mismatch Kaypad code and configuration parameter per [LBI.288.03].
September 14, 2018	**2.2**	
	LBI-FRD-REQ-280168/C-Lincoln Backup Ignition	cwu3: Revised for ver 2.1 release
	STR-458487/D-LBIServer General Requirements	rpaquet2 - Removed 271443 as it already exists in function 275727
	LBIv1-REQ-318311/B-Configurable Parameter for Keypad Code	rpaquet2 - Updated text for clarification
	LBIv1-REQ-264925/C-Operation Modes for LBI Functions and Menus	rpaquet2 - Updated text for clarification
	PaaK-REQ-270465/G-Payload Parameters Definitions (PaaK/LBI)	rpaquet2 - added Brand Reset values
	LBIv1-REQ-322204/B-LBI Keypad Code Composition Restriction	rpaquet2 - Updated the constrictions
	LBIv1-REQ-318630/B-Suspend LBI when CCS off	rpaquet2 - Updated text to clarify ignoring CAN signals when CCS is off
Octobre 10, 2019	**2.3**	
	MD-REQ-241972/F-PaakESN_St+	rpaquet2 - Updated the description to clarify operations
	MD-REQ-241972/G-PaakESN_St+	rpaquet2 - Update description per feature owner
	MD-REQ-241972/H-PaakESN_St	rpaquet2 - Clarified description
	LBIv1-REQ-260142/B-LBI Menu Configurable Parameter	ndecia: Updated requirement to capture when the PaaK BSP menu should and should not be made available.
	PaaK-REQ-270465/H-Payload Parameters Definitions (PaaK/LBI)	rpaquet2- Update table
	STR-459087/D-Security Requirements	ndecia: Revised structure to include two new requirements 351820 & 351821 for the challenge response time out
	LBIv1-REQ-351821/A-Time Out After Waiting for a Challenge Response	ndecia: New requirement to capture the time out period when waiting for a challenge response
	LBIv1-TMR-REQ-351820/A-T_Challenge Timeout Expiration Timer	ndecia - new timing requirement to accompany LBIv1-REQ-351821
	STR-454355/C-Requirements+	ndecia - added new requirements 33138 & 335312

STR-454355/D-Requirements	ndecia: Revised structure to include new requirement 360808 for FNV2 password strength rules
LBIv1-REQ-260056/B-HMI Display of Paak Backup Password Creation Option	ndecia: clarified handling when connectivity is disabled from CCS setting
LBIv1-REQ-265042/C-Backup Passwords Strength Definitions for CGEA1.3C	ndecia: clarified password strength requirements per architecture and updated URL links
LBIv1-REQ-360808/A-Backup Passwords Strength Definitions for FNV2	ndecia: clarified password strength requirements per architecture and updated URL links
LBIv1-REQ-333138/A-De-bounce Timer When Localizing PaaK Device in Vehicle	ndecia: new functional requirement
LBIv1-TMR-REQ-335312/A-T_Phone Localization Timer	ndecia - new timer requirement to accompany LBIv1-REQ-333138
LBIv1-UC-REQ-323248/B-Entering Too Little Digits for Keypad Code	ndecia: Updated use case to match actual HMI implementation.
LBIv1-UC-REQ-323249/B-Entering Too Many Digits for Keypad Code	ndecia: Updated use case to match actual HMI implementation.
LBIv1-UC-REQ-323257/B-Entering Unacceptable Keypad Code (For EU Markets only)	ndecia: Updated use case to match actual HMI implementation.
STR-454364/D-Requirements	ndecia - added new requirements 332837 & 335313
LBIv1-REQ-332837/A-Authorization Timer When Generating Enhanced Valet Password With No Devices Detected in Vehicle	ndecia: new functional requirement
LBIv1-TMR-REQ-335313/A-T_Valet Authorization Timer	ndecia - new timing requirement to accompany LBIv1-REQ-332837
STR-459878/B-Sequence Diagrams	ndecia - added new sequence diagram 332841
LBIv1-SD-REQ-332841/A-Authorization Timer when Generating Enhanced Valet Password - No Devices Detected	ndecia - new Sequence Diagram to accompany new functional requirement "Authorization Timer When Generating Enhanced Valet Password With No Devices Detected in Vehicle."
PaaK-REQ-270039/C-Passive Challenge Data Response	Rpaquet2 - updated to transmit signal PaakLoclzd_D_Stat to transmit BLEM Power mode state to BCM per feature owner
LBIv1-REQ-270547/C-Notification of Valid Backup Password Usage to LBIClient2	ndecia: Updated Key ID assignment requirements
STR-457495/D-Requirements	ndecia: Revised structure to include new requirement 351819 for Keypad Programming Interface
LBIv1-REQ-351819/A-Keypad Programming Interface	ndecia: New requirement to describe keypad programming signal handling

Table of Contents

1 Feature Description

Lincoln Backup Ignition (LBI) is also known as PaaK Backup with the Market name of “Backup Starting Passcode “.

LBI serves as a backup method for starting Lincoln vehicles that are equipped with Phone-as-a-Key (PaaK). This feature utilizes the existing keypad entry system as well as a new, password-based starting system. It allows the customer to start and drive away their vehicle even if their phone is not functional (e.g. drained battery) or if their phone is lost, stolen or destroyed. If any of these situations occur, customers can gain entry to the vehicle via the keypad and then to enter a password at the CenterStack Display Device to prime the vehicle for starting. PaaK Backup also allows customers to generate a temporary password and keypad code to give to valet attendants through an Enhanced Valet Mode .

2 Architectural Design

2.1 LBIv1-CLD-REQ-258558/A-LBI Client

The LBIClient is responsible for displaying password functionality to the user. It allows the user to create, modify, and delete passwords as well as enter the password to authorize vehicle start

2.2 LBIv1-CLD-REQ-258559/B-LBI Server

The LBIServer, which is the main controller of the PaaK system, is responsible for password storage and verification. LBI Server communicates with the LBI Server 2 to request FOB information and will provide a response to the LBI Client.

2.3 LBIv1-CLD-REQ-258560/A-LBI Server2

LBIServer2 shall provide FOB information to the LBIServer. The LBIServer2 shall communicate with the LBIServer to setup Keypad information.

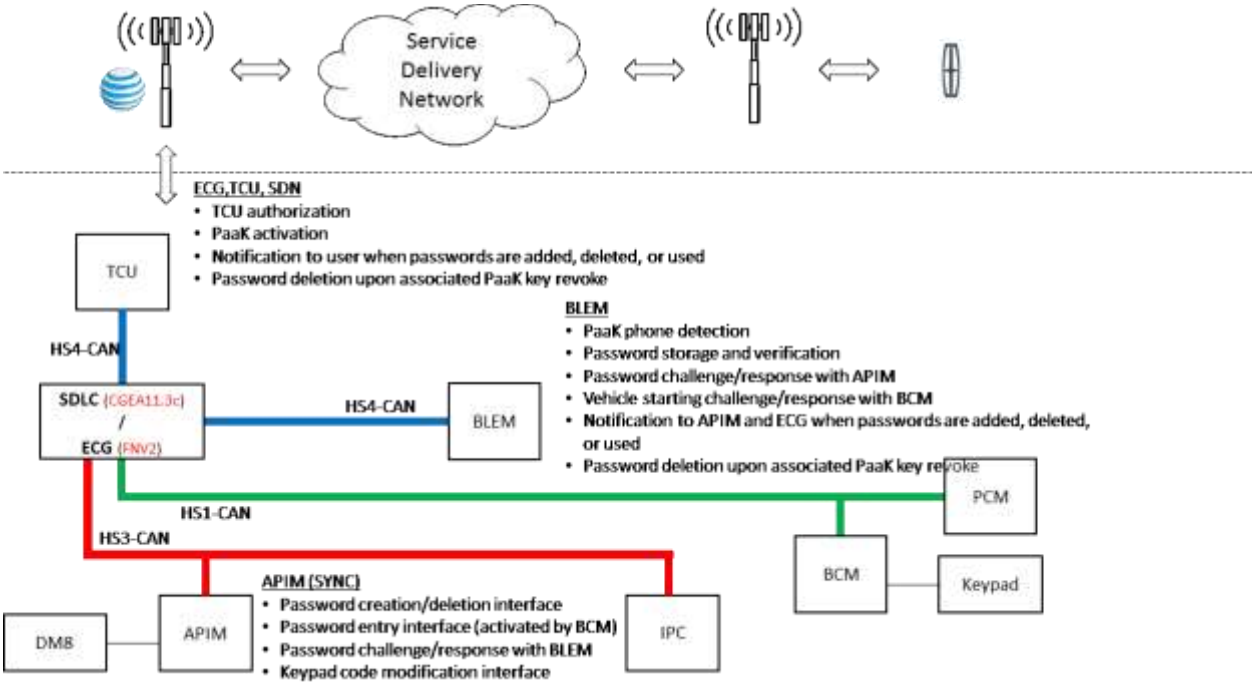
2.4 Lincoln Backup Ignition Physical Mapping of Classes

The table below shows an example of how the logical classes may be mapped into physical modules. This mapping example is specific to the MY20 U611 program under CGEA1.3c architecture and MY21 CX727 under FNV2 architecture and does not necessarily carryover to other carlines or vehicle architectures.

Logical Class	Physical Module (ECU)
LBIClient	APIM
LBIClient2	TCU for CGEA1.3c, ECG for FNV2
LBIServer	BLEM
LBIServer2	BCM
Embedded Modem Reset OnBoardClient	APIM
PaaK Server	BLEM

2.5 Lincoln Backup Ignition System

The diagram below shows an example of how the logical classes may be mapped into physical modules as Lincoln Backup Ignition System. This mapping example is specific to the MY20 U611 program under CGEA1.3c architecture and MY21 CX727 under FNV2 architecture and does not necessarily carryover to other carlines or vehicle architectures.



Lincoln Backup Ignition System Diagram

2.6 Lincoln Backup Ignition Logic Method to Physical Signal Translation Table

Logic Method names are translated into GSDB signal names in this table. The Global Signal Database (GSDB) is the master for all signals. GSDB signal names listed here are for reference only. Readers are advised to refer to Global Signal Database for up to date information. Tx and Rx in this table only reflect MY 2020 U611 program and do not necessarily carryover to other carlines or other model years for U611.

Logic Method Name	GSDB Signal Name	Encoding Value	Encoding Name	Tx	Rx
PasswordReady_St	IgnPsswrdActv_B_Stat	0x0	Inactive	BLEM	APIM/BCM
		0x1	Active		
PasswordEntryScreen_Rq	IgnPsswrdDsply_B_Rq	0x0	Inactive	BCM	APIM
		0x1	Active		
FobTrgtPassiveData	FobTrgtPssvData_No_Rq	0x0 – 0xFFFFFFFFFF	Challenge	BLEM	BCM
FobTargetSearchType	FobTrgtType_D_Rq	0x0	NULL	BLEM	BCM
		0x1	CRYPTO		
		0x2	REGISTRY		
		0x3	POLLING		
		0x4 – 0x7	*Reserved*		
FobCtrlSearchRslt	FobCtlType_D_Stat	0x0	NULL	BCM	BLEM
		0x1	INVALID		
		0x2	VALID		
		0x3	*Reserved*		
FobTargetSearchZone	FobTrgtZone_D_Rq	0x0	NULL	BLEM	BCM
		0x1	INTERIOR		
		0x2	DRIVER		
		0x3	PASSENGER		
		0x4	REAR_EXTERIOR		
		0x5	REAR_INTERIOR		
		0x6	APPROACH		
		0x7 - 0xF	*Reserved*		
FobTargetRollCode	FobTrgtRollCode_No_Rq	0x0 - 0xF	Rolling Code	BLEM	BCM
FobCtrlRollCode	FobCtlRollCode_No_Stat	0x0 - 0xF	Rolling Code	BCM	BLEM
LBISetup_Rq	IgnPsswrdSetup_B_Rq	0x0	Inactive	BLEM	APIM
		0x1	Active		
LBILockout_St	IgnPsswrdLckout _B_Stat	0x0	Inactive	BLEM	APIM
		0x1	Active		
KeyPadCodeProg_Rq	KeyPadCodeProg_D_Rq	0x0	Null	BLEM	BCM
		0x1	ProgrammingMode		
		0x2	Add		
		0x3	Delete		
		0x4	DeleteAll		
KeyPadCodeProg_St	KeyPadCodeProg_D_Stat	0x0	NormalMode	BCM	BLEM

		0x1	LearningMode		
		0x2	Add		
		0x3	Delete		
		0x4	DeleteAll		
		0x5	ProgrammingFailure		
		0x6	Duplicate		
		0x7	NotUsed_1		
PaakTargetRKEData	PaakTrgtActvData_No_Rq	0x0 – 0xFFFFFFFFFF	Challenge	BCM	BLEM
PaakCtrl_RKESubID	PaaKCtlActv_No_Actl	0x00 – 0x3F	PhoneSubID	BLEM	BCM
PaakCtrlRKEData	PaakCtlActvData_No_Actl	0x0 – 0xFFFFFFFFFF	Response	BLEM	BCM
PaaKTargetSearchType	PaakTrgtType_D_Rq	0x0	NULL	BCM	BLEM
		0x1	CRYPTO		
		0x2	REGISTRY		
		0x3	POLLING		
		0x4	*Reserved*		
		0x5	*Reserved*		
		0x6	*Reserved*		
		0x7	*Reserved*		
FobCtrlPassiveData	FobCtlPssvData_No_Actl	0x0 – 0xFFFFFFFFFF	Response	BCM	BLEM
GearLvrPos_D_Actl	GearLvrPos_D_Actl	0x0	Park	TCM	APIM
		0x1	Reverse		
		0x2	Neutral		
		0x3	Drive		
		0x4	Sport_DriveSport		
		0x5	Low		
		0x6	First		
		0x7	Second		
		0x8	Third		
		0x9	Fourth		
		0xA	Fifth		
		0xB	Sixth		
		0xC	Undefined_Treat_as_Fault		
		0xD	Undefined_Treat_as_Fault1		
		0xE	Unknown_Position		
		0xF	Fault		
IgnitionStatus_St	IgnitionStatus	0x0	Unknown	BCM	APIM
		0x1	Off		
		0x2	Accessory		
		0x4	Run		
		0x8	Start		
		0xF	Invalid		
PaakTargetSearchZone	*PaakTrgtZone_D_Rq*	0x0	Null	BCM	BLEM
		0x1	Interior		
		0x2	Driver		

		0x3	Passenger		
		0x4	Rear_Exterior		
		0x5	Rear_Interior		
		0x6	Approach		
		0x7~0xF	Reserved		
PaakTargetPassiveData	PaakTrgtPssvData_No_Rq	0x0 – 0xFFFFFFFFFF	Challenge	BCM	BLEM
PaakTargetRollCode	PaakTrgtRollCode_No_Rq	0x0 - 0xF	Rolling Code	BCM	BLEM
PaakCtrlPassiveData	PaakCtlPssvData_No_Actl	0x0 – 0xFFFFFFFFFF	Response	BLEM	BCM
PaakCtrlSearchRslt	PaakCtlType_D_Stat	0x0	Null	BLEM	BCM
		0x1	Invaild		
		0x2	Valid		
		0x3	NotUsed_1		
PaakCtrlRollCode	PaakCtlRollCode_No_Stat	0x0 – 0xF	Rolling Code	BLEM	BCM
PaakCtrlFoundRollCode	PaakCtlRollCode_No_Copy	0x0 – 0xF	Rolling Code Copy	BLEM	BCM
PaakCtrlFoundIndex[1-8]	*PaakCtlIndx[1-8]_No_Actl*	0x0 – 0x3F	PhoneIndex(0-63)	BLEM	BCM
FactoryReset_Rq	FactoryReset_Rq	0x0	Inactive	APIM	BLEM
		0x1	ResetFactoryDefaults		
EmbeddedModemReset_Rq	ModemReset_D_Rq	0x0	Null	APIM	BLEM
		0x1	WifiHotspot_Reset		
		0x2	PaaK_Reset		
		0x3	OnlineTraffic_Reset		
		0x4	CCS_Reset		
		0x5 – 0xF	Reserved		
DelayAccy_St	Delay_Accy	0x0	Off	BCM	APIM
		0x1	On		
PaakCtrlActionCode	*PaakCtlActv_D_Rq*	0x00	Null	BLEM	BCM
		0x01	UnlockDriverDoor		
		0x02	UnlockAllDoors		
		0x03	UnlockCargoDoor		
		0x04	UnlockAllDoorsPlusCargoDoor		
		0x05	LockAll		
		0x06	DoubleLock		
		0x07	DecklidRelease		
		0x08	LiftgateRelease		
		0x09	LiftgateGlassRelease		
		0x0A	PanicOn		
		0x0B	PanicOff		
		0x0C	GlobalOpen		
		0x0D	GlobalClose		
		0x0E	PowerLiftgate		

		0x0F	LeftPowerSlidingDoor		
		0x10	RightPowerSlidingDoor		
		0x11	PowerDecklid		
		0x12	RemoteStart		
		0x13	RemoteStop		
		0x14..0x1F	*Reserved*		
PwPckTq_St	PwPckTq_D_Stat	0x00	PwPckOff_TqNotAvailable	PCM	APIM
		0x01	PwPckOn_TqNotAvailable		
		0x02	StartInPrgrss_TqNotAvailabl		
		0x03	PwPckOn_TqAvailable		

2.7 LBI Client Interface

2.7.1 IIR-REQ-258548/A-LBIClient_Tx

2.7.1.1 MD-REQ-258536/B-BackupIgnition_Rq

Message Type: Request

This signal is used to request information from the LBIServer.

Name	Literals	Value	Description
OpCode	-	-	Signifies what is being requested or transmitted
	Reserved	0x00	
	Challenge Request	0x01	
	Challenge Response	0x02	
	Salt and Check for PaaK with Passwords	0x03	
	Salt and Check for PaaK without Passwords	0x04	
	Check for Keys to Enter Valet Mode	0x05	
	Check for Keys to Exit Valet Mode	0x06	
	Password Transmit	0x07	
	Keypad Code Create Request	0x08	
	Password Delete Request	0x09	
	Valet Create Challenge Response	0x0A	
	Valet Delete Challenge Response	0x0B	
	Reset Challenge Response	0x0C	
	Reset 1 Password Transmit	0x0D	
	Reset 2 Password Transmit	0x0E	
	Valet Start Challenge Response	0x0F	
	Not Used	0x10 – 0xFF	
KeyIndex	-	-	Index assigned to a PaaK
	Reserved	0x00	
	KeyIndex 1	0x01	
	KeyIndex 2	0x02	
	…		
	KeyIndex 255	0xFF	
Password	-	-	32 byte SHA256 Hash contains password hash or challenge password hash

Keypad Code	-	-	4 byte code

2.7.1.2 MD-REQ-213361/C-FactoryReset_Rq

Message Type: Request

Signal sent by the Master Reset Client to initiate a Master Reset

Logical Signal Name	Literals	Value	Description
FactoryReset_Rq	Inactive	0x0	
	ResetFactoryDefaults	0x1	

2.7.1.3 MD-REQ-246273/C-EmbeddedModemReset_Rq

Message Type: Request

This signal is used to perform a factory reset for the specified Embedded Modem feature.

Name	Literals	Value	Description
Type	-	-	Embedded Modem feature to be reset to factory defaults.
	Null	0x0	
	WifiHotspot_Reset	0x1	
	PaaK_Reset	0x2	
	OnlineTraffic_Reset	0x3	
	CCS_Reset	0x4	
	BrandConnect_Reset1	0x5	
	BrandConnect_Reset2	0x6	
	Reserved	0x7 – 0xF	

2.7.2 IIR-REQ-258549/C-LBIClient_Rx

2.7.2.1 MD-REQ-258535/A-PasswordEntryScreen_Rq

Message Type: Request

Used to request a transition to the Backup Ignition password display .

Name	Literals	Value	Description
Type	-	-	
	Inactive	0x0	
	Active	0x1	

2.7.2.2 MD-REQ-262111/A-PasswordReady_St

Message Type: Status

Signal for when PaaK user creates at least one LBI password.

Name	Literals	Value	Description
Type	-	-	
	Inactive	0x0	
	Active	0x1	

2.7.2.3 MD-REQ-258544/C-BackupIgnition_Rsp

Message Type: Response

This signal is used to respond to the BackupIgnition_Rq.

Name	Literals	Value	Description
Rsp Code	-	-	Response code being sent
	Reserved	0x00	
	Issue Challenge	0x01	
	Challenge Response Acknowledge	0x02	
	Salt and Check for PaaK with Passwords Response	0x03	
	Salt and Check for Paak without Passwords Response	0x04	
	Check for Keys to Enter Valet Mode	0x05	
	Check for Keys to Exit Valet Mode	0x06	
	Password Response	0x07	
	Keypad Code Create Response	0x08	
	Password Delete Response	0x09	
	Valet Create Challenge Response Acknowledge	0x0A	
	Valet Delete Challenge Response Acknowledge	0x0B	
	Reset Challenge Response Acknowledge	0x0C	
	Reset 1 Password Response	0x0D	
	Reset 2 Password Response	0x0E	
	Valet Start Challenge Response Acknowledge	0x0F	
	Reserved	0x10-0xFF	
Rsp Status	-	-	Status of response
	Reserved	0x00	
	One PaaK w/o Password and Fob In Vehicle	0x01	
	One PaaK w/o Password and No Fob In Vehicle	0x02	
	Fob in Vehicle and No PaaK w/o Password	0x03	
	Two+ PaaK w/o Password and Fob In Vehicle	0x04	
	Two+ PaaK w/o Password and No Fob In Vehicle	0x05	
	No PaaK w/o Password and No Fob In Vehicle	0x06	
	PaaK No Longer Detected	0x07	
	Fob No Longer Detected	0x08	
	PaaK and Fob No Longer Detected	0x09	
	Password Already Used	0x0A	
	Password Created Successfully	0x0B	
	Password Created Failed	0x0C	
	Keypad Code Created Successfully	0x0D	
	Keypad Code Created Failed	0x0E	
	Valid Password	0x0F	
	Invalid Password	0x10	

	One PaaK w/ Password and Fob In Vehicle	0x11	
	One PaaK w/ Password and No Fob In Vehicle	0x12	
	Fob in Vehicle and No PaaK w/ Password	0x13	
	Two+ PaaK w/ Password and Fob In Vehicle	0x14	
	Two+ PaaK w/ Password and No Fob In Vehicle	0x15	
	No PaaK w/ Password and No Fob In Vehicle	0x16	
	Password Deleted Successfully	0x17	
	Password Deleted Failed	0x18	
	Lockout	0x19	
	Keypad Code Duplicate	0x1A	
	Fob In Vehicle	0x1B	
	No PaaK and No Fob In Vehicle	0x1C	
	Password Created Successfully and Delivered to PaaK	0x1D	
	Password Deleted Successfully, but Keypad Code Deleted Failed	0x1E	
	Reserved	0x1F-0xFF	
NumOfItem	-	-	Indicates how many items are sent over when responding with multiple KeyID's.
	Reserved	0x00	
	1	0x01	
	…	…	
	4	0x04	
	No Entry	0xFF	
ItemIndex	-	-	
	Reserved	0x00	
	…	…	
	ItemIndex 254	0xFF	
KeyIndex	-	-	Indicates unique value for each PaaK
	Reserved	0x00	
	KeyIndex 1		
	KeyIndex 2		
	…	…	
	KeyIndex 254	0xFF	
Valet Password	-	-	4 byte 0x00000000 to 0x05F5E0FF (0 to 99999999)
Challenge Nonce	-	-	32 byte random number
Salt	-	-	16 byte random number
Phone Name	-	-	Data array that consists of textual information up to 40 characters in length, plus end of string

2.7.2.4 MD-REQ-262113/A-LBISetup_Rq

Message Type: Request

Signal used to trigger LBI setup.

Name	Literals	Value	Description
Type	-	-	
	Inactive	0x0	
	Active	0x1	

2.7.2.5 MD-REQ-262115/A-LBILockout_St

Message Type: Status

Signal used to enable lockout.

Name	Literals	Value	Description
Type	-	-	
	Inactive	0x0	
	Active	0x1	

2.7.2.6 MD-REQ-199809/A-IgnitionStatus_St

Message Type: Status

Signal used to indicate ignition state.

Name	Literals	Value	Description
Type	-	-	Indicates ignition state
	Unknown	0x0	
	Off	0x1	
	Accessory	0x2	
	Run	0x4	
	Start	0x8	
	Invalid	0xF	

2.7.2.7 MD-REQ-199808/A-GearLvrPos_D_Actl

Message Type: Status

Vehicle status signal for the Gear Lever Position on an automatic transmission vehicle.

Name	Literals	Value	Description
Type	-	-	-
	Park	0x0	
	Reverse	0x1	
	Neutral	0x2	
	Drive	0x3	
	Sport_DriveSport	0x4	
	Low	0x5	
	First	0x6	
	Second	0x7	
	Third	0x8	
	Fourth	0x9	
	Fifth	0xA	

	Sixth	0xB	
	Undefined_Treat_as_Fault	0xC	
	Undefined_Treat_as_Fault1	0xD	
	Unknown_Position	0xE	
	Fault	0xF	

2.7.2.8 MD-REQ-277622/A-DelayAccy_St

Message Type: Status

Signal used to Indicates the status of the delayed accessories.

Name	Literals	Value	Description
Type	-	-	
	Off	0x0	
	On	0x1	

2.7.2.9 MD-REQ-241972/H-PaakESN_St

Message Type: Status

"PaakESN_St" is a TP CAN signal used to indicate the provisioning state, ESN and BPEK (One way hashed) "PasKESN_St" contains the BLEMProvDID (Actual name in GMRDB "Bluetooth Low Energy Module (BLEM) Provisioning Status") and ProvOnBoardClient4's metadata. It shall include: "TP header" + "SyncP Header" + Payload as shown on requirement PMPR-REQ-331617.

PaakESN_St" is a periodic TP message that will be transmitted through CAN from ProvOnBOardlcient4 to ProvServer.

The table below denotes the data that is required in the PaakESN_St TP message for the ProvOnBoardClient4..

Peripheral ECU	Transport Protocol Message	FTCP Logic
BLEM	PaakESN_ST	BLEMProvisioningAlert

BLEMProvDID represents the Provisioning State of ProvOnBoardClient4 within itself and stored in ECU memory (DID 0x021). Please refer to requirement PMPR-REQ-354871 for further details.

Name	Literals	Value	Description
BLEMProvDID	-	-	Describes the current state Provisioning
	FactoryMode	0x50	BLEM is not Configured
	Unprovisioned	0x51	BLEM Configured, TargetID not Transfer/ BLEM Self-Test not complete
	BLEMProvAlertACK	0x52	BLEM is waiting for Provisioning Alert Ack from PaakOnBoardClient
	ReadyForKeyDelivery	0x53	BLEM is Provisioned and ready for Key Delivery
	KeyDelivered	0x54	Key(s) are delivered to BLEM
BLEMSyncPPacket	-	-	BLEM SyncP Signed (BLEM ESN). BLEM ESN will be in the header of SyncP Signed message. SyncP Payload information found in PaaK-REQ-281398-Provisioning SyncP Payload. Max. 1000 bytes.

2.7.2.10 MD-REQ-293501/A-PwPckTq_St

Message Type: Status

Signal used to indicate if the power pack is a motive (wheel torque producing) or non-motive (non-wheel torque producing) mode. It also indicates to if a transition from a non-motive to a motive mode is in progress.

Name	Literals	Value	Description
PwPckTq_St	-	-	Indicate power pack state
	PwPckOff_TqNotAvailable	0x0	Engine is not running
	PwPckOn_TqNotAvailable	0x1	Engine is running in Remote Start or Secure Idle Mode (NonMotive mode – vehicle cannot be driven)
	StartInPrgrss_TqNotAvailabl	0x2	Engine is cranking(A transition from a NonMotive to a motive mode is in progress)
	PwPckOn_TqAvailable	0x3	Engine is running and in a Motive mode

2.8 Lincoln Backup Ignition Logic Method Requirement

2.8.1 LBIv1-REQ-275627/B-Request/Response Return to Null State

In the context of Lincoln Backup Ignition feature, when updating on event, unless noted otherwise the transmitting modules shall hold their signal encoding values for at least as long as one signal sampling interval as defined in GSDB and then shall transit back to Null:

The receiving modules of these signals shall act upon the event signal and shall not wait for the “Null” to act upon the signal request.

2.8.2 LBIv1-REQ-276010/A-Assumption for Crank Event

In the context of Lincoln Backup Ignition feature, when a Crank event occurs as indicated in IgnitionStatus_St(Crank), IgnitionStatus_St(Crank) is to be considered as a “don’t care” event and the IgnitionStatus_St shall be assumed the last state unless noted otherwise.

For example, if IgnitionStatus_St is in Run and a Crank event happens with IgnitionStatus_St(Crank) and then IgnitionStatus_St goes back to Run, unless noted otherwise it shall be assumed that in the use cases and functional requirements that Ignition remained in Run.

2.8.3 LBIv1-REQ-276035/A-LBI Network WakeUp Signal Designation

The signal IgnPsswrdDsply_B_Rq shall be designated as Network WakeUp Signals as defined in the requirement of EY-0088(HIGH SPEED & MEDIUM SPEED CONTROLLER AREA NETWORK PROTOCOLS), when transmitter and receiver of the signal are not in the same Network.

This requirement is needed to ensure that password entry screen can be displayed in the case when ignition is not in Run in which not all modules are awake at the same time.

2.8.4 LBIv1-REQ-304588/A-Usages of PaakCtrlRKEData in LBI Keypad Code Programming

For LBI Keypad code applications, the LBIServer shall use *PaakCtrlRKEData* for the two following actions:

1. Transmitting RKE challenge response when issue KeyPadCodeProg_Rq(ProgrammingMode) as required by LBIv1-REQ-270265

2. Transmitting Keypad code to be stored when issue KeyPadCodeProg_Rq(Add), as described in LBIv1-REQ-264862

Note: PaakCtrlRKEData alone does not make any significance unless it comes along with meaningful RKE request via PaakCtrlActionCode or Keypad Code request via KeyPadCodeProg_Rq. As the result, t*he same data of PaakCtrlRKEData can be sending out periodically until next RKE events or Keypad programing events (the request of entering programming mode and the request of storing Keypad code) occurs then a new value of PaakCtrlRKEData is needed.*

2.8.5 PaaK-REQ-307226/A-Remote Start and RKE return to Null

The PaakServer shall reset PaakCtrlActionCode to null (0x0) upon receiving the new challenge data in PaakTargetRKEData. If new challenge data is not received within 2000ms after the PaakServer sends a RKE command the PaakServer shall set PaakCtrlActionCode to null (0x0) and fail the RKE command. The PaakServer shall transmit null for at least one transmit period (200ms) before sending a new RKE command.

3 General Requirements

3.1 LBIServer General Requirements

3.1.1 LBIv1-REQ-271444/A-Status of Backup Password Setup

The LBIServer shall send PasswordReady_St(Active) when there is a least one backup password stored in HSM.

3.1.2 LBIv1-REQ-271439/A-TP Functions Execution Time

The LBIServer shall stay in TP heartbeat session no longer than five seconds. If, at the end of five seconds it has not completed its task, it shall respond with CES = 0x10 = Final Result – Failure.

3.1.3 LBIv1-REQ-271440/B-CAK Asignmenet Restriction

The LBIServer shall not assign a CAK to the Valet key index as defined in LBIv1-REQ270053.

3.1.4 LBIv1-REQ-271442/A-Verification of Backup Password without LBIClient2 Authorization

Unless LBIServer has received Auth_St = Remove Keys = 0x2, the LBIServer shall continue to allow verification of backup passwords.

3.1.5 LBIv1-REQ-271441/A-Active Backup Password and Keypad Code Flags

For each key index, the LBIServer shall keep track of whether there is:

- an active backup password associated with it
- an active keypad code associated with it

3.1.6 LBIv1-REQ-318311/B-Configurable Parameter for Keypad Code

The LBIClient and the LBIServer shall each have a configurable parameter to define the LBI Keypad code length for different markets as required by LBIv1-REQ-322144.

The usage of the LBI Keypad code configurable parameter include but not limited to

- For the LBIServer to determine mismatch between received Keypad code length and required length for different markets as required in LBIv1-REQ-318296

- For the LBIServer to determine the length of Enhanced Valet Password as described in LBIv1-REQ-317825

3.1.7 LBIv1-REQ-275544/B-Valet Password Uniqueness

Upon generating a valet password, the LBIServer shall compute its hash and verify its uniqueness by comparing it to existing password hashes. If it is not unique, the LBIServer shall generate a new random-numeric valet password, which the LBIServer shall also verify for uniqueness.

3.1.8 LBIv1-REQ-275626/A-Parameter Memory Storage Requirement for LBIServer

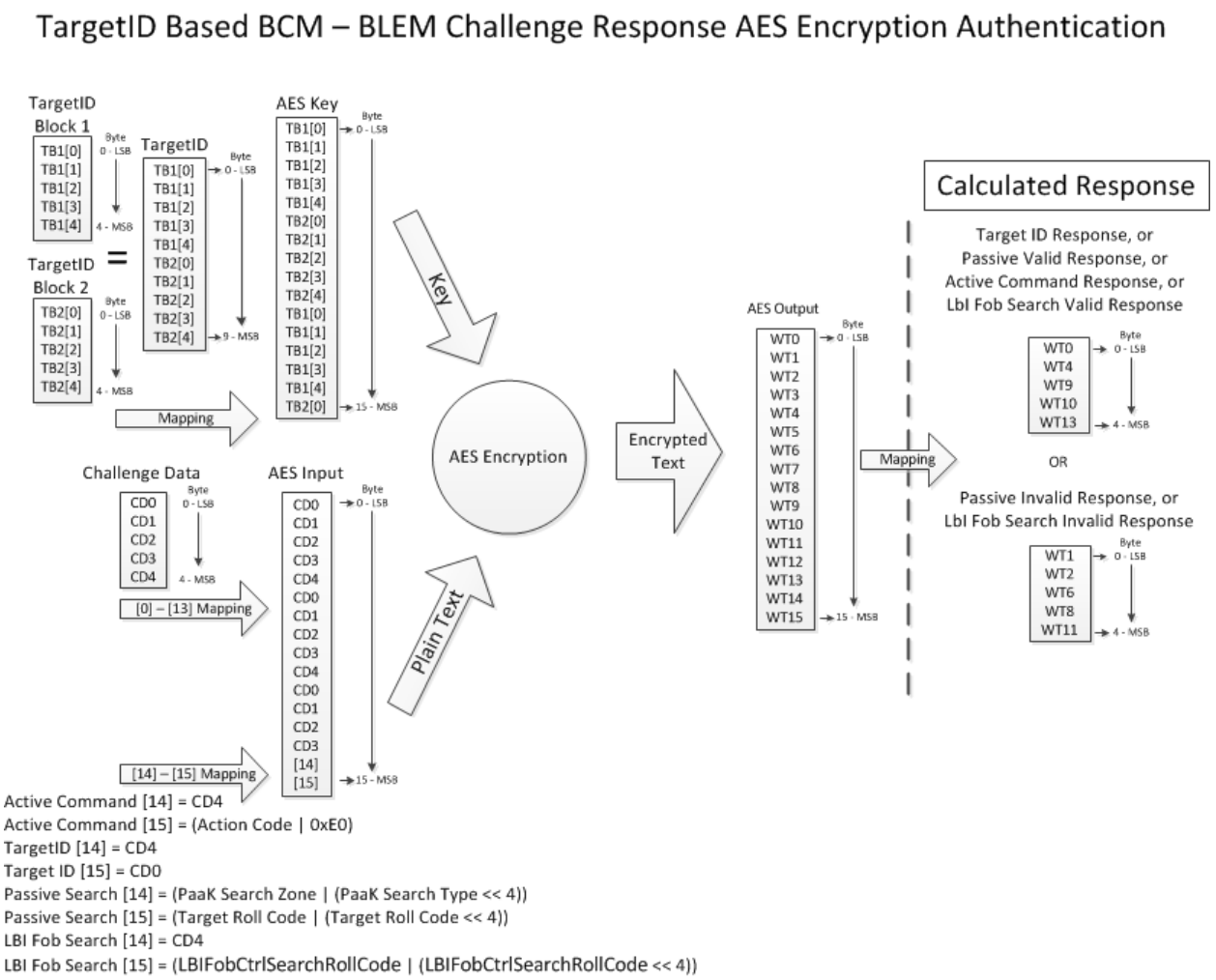
The LBIServer shall maintain the following parameters in non-volatile memory:

- N_NumberOfInvalidBackupAttemp
- N_NumberOfInvalidValetAttemps
- T_LBILockout Timer
- Enhanced Valet password
- Backup passwords with associated key indexes
- Enhanced Valet Mode Status

If the above requirement cannot be met, the LBIServer shall use other mechanism or design to ensure that the values of above signals can be recalled after the following events:

- After a B+ reset
- After a module reset
- Between ignition cycles
- Between network bus sleep and wake-up events
- Any battery charge state

3.1.9 PaaK-REQ-269555/B-BCM - BLEM Communication AES Encryption



The standard for the AES encryption can be found here:
http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf

3.1.10 PaaK-REQ-269556/A-AES Key

The BLEM shall use the TargetID to create an AES Key. To fill the 16 byte Key, the BLEM shall use Block1 and Block2 of the Target ID for the first 10 bytes and then Block1 for the next 5 bytes and byte0 of Block 2 for the byte15 as shown in diagram above

3.1.11 PaaK-REQ-269557/B-AES Input

The BCM will send the BLEM 5 bytes of Challenge Data. The BLEM shall use the Challenge Data to create the byte0 through byte13 of the AES Input. The last 2 bytes will be used for action specific responses.
For active commands the BLEM shall use the byte4 of the Challenge Data for the byte14 of the AES input and (Action Code | 0xE0) for byte15 of the AES input.
ActionCode is the data that will be sent in PaakCtrlActionCode

For passive search the BLEM shall use (PaakSearchZone | (PaakSearchType << 4)) for byte14 of the AES input and (TargetRollCode | (TargetRollCode << 4)) for byte15 of the AES input.
PaakSearchZone is the data that will be sent in PaakTargetSearchZone
PaakSearchType is the data that will be sent in PaakTargetSearchType
TargetRollCode is the data that will be sent in PaakTargetRollCode

For TargetID, the BLEM shall use byte4 of the Challenge Data for byte14 of the AES input and byte0 of the Challenge Data for byte15 of the AES input.

3.1.12 PaaK-REQ-269558/A-AES Output

The AES encryption will result in a 16 byte output. The response to the BCM will consist of 5 specific bytes from this AES output and depend on the desired response.
The BLEM shall use bytes [0, 4, 9, 10, 13] to create the TargetID response, “Authorized Phone in requested zone” Passive Search Response or Active Command Response.
The BLEM shall use bytes [1, 2, 6, 8, 11] to create the “Authorized Phone not in requested zone” Passive Search Response.

3.1.13 PaaK-REQ-270046/C-Remote Start and RKE Challenge Response

The BLEM shall transmit a single RKE command per challenge data in PaakTargetRKEData. The BLEM shall use the action code of the command to create byte15 of the AES input (Action Code | 0xE0).
Action Code is the data the BLEM received from the mobile app and shall be sent in PaakCtrlActionCode.
The BLEM shall perform the AES calculations as described in requirement 269555-BCM-BLEM Communication AES Encryption.
The BLEM shall transmit bytes [0, 4, 9, 10, 13] of the AES calculation output in PaakCtrlRKEData.

3.1.14 PaaK-REQ-242454/C-BLEM-BCM Interactions

The BLEM shall utilize the TargetID, generated and provided by the BCM, which is a shared 10 byte random number, padded to 16 bytes and used as the shared key in AES-128 calculations in order to authenticate BLEM-BCM interactions. The AES calculations are described further in PaaK-REQ-269555-BCM_BLEM Communication AES Encryption.

3.1.15 PaaK-REQ-270047/A-RKE Queue

The PaakServer shall be able to queue up to three RKE commands. The PaakServer shall transmit a queued command when new challenge data in PaakTargetRKEData is received in the order the commands were received. If the queue is filled and a new RKE command is received, the PaakServer shall replace the oldest command with the newest command.

3.1.16 LBIv1-REQ-312524/A-Preconditions for Response to LBI Requests

The LBIServer shall not response the request BackupIgnition_Rq for the functions defined in LBIv1-REQ- 264936 with the exception of staring the vehicle (BackupIgnition_Rq (OpCode = “Challenge Request”, “Challenge Response” or “Valet Start Challenge Response”) unless the conditions below are all met.

- The vehicle ignition is in RUN as indicated in IgnitionStatus_St(*Run*)
- The vehicle transmission must be in the Park position as indicated in *GearLvlPos_D_Actl(Park)*
- The vehicle is not in LBI lockout period as defined by LBIv1-REQ-268242
    - When the vehicle is in the LBI lockout period, the exception of staring the vehicle listed above shall be ignored too. In other words, all LBI requests, including starting the vehicle with LBI passwords, shall be ignored during LBI lockout period. Please refer to LBIv1-REQ-268502 for the lockout actions the LBIServer shall perform

Note: This verification check ensures that the LBI functions (except starting the vehicle with LBI passwords) are only accessible when vehicle ignition is in run, transmission is in park and the vehicle not in lockout period. This check will ensure that the LBIServer and does not get too far ahead if the LBIClient sends an opcode at the same time the user turns the ignition off or shift out of park.

3.2 LBIClient General Requirements

3.2.1 LBIv1-REQ-271434/A-Delete Password Hash from Memory after Transmitted

Once the LBIClient transmits a password hash to the LBIServer, it shall delete this password hash from memory.

3.2.2 LBIv1-REQ-271435/A-Error Handling for Multiple Challenges Nonces

If the LBIClient has received multiple challenges nonces, it shall recognize only the most recent one as valid.

3.2.3 LBIv1-REQ-271436/A-HMI Paak Devices Display Order

Any time the LBIClient HMI displays a list of PaaK devices, the list shall be sorted by key index, with the lowest index at the top.

3.2.4 LBIv1-REQ-264936/A-LBI User Interface

The LBIClient shall provide HMI to support LBU functions listed below:

- Create backup passwords/keypad codes for Paak devices
- Reset (change, not delete) backup passwords/keypad codes for Paak devices
- Delete backup passwords/keypad codes for PaaK devices one password at a time
- Delete all backup passwords/keypad codes for PaaK devices at once
- Activate Enhanced Valet Mode (Creating Enhanced Valet password and entering Enhanced Valet Mode)
- Deactivate Enhanced Valet Mode (Deleting Enhanced Valet password and exiting Enhanced Valet Mode)
- Start the vehicle with backup password or Enhanced Valet password

3.2.5 LBIv1-REQ-260142/B-LBI Menu Configurable Parameter

The LBIClient shall provide LBI specific menu when the vehicle is configured for Phone-as-a-Key as defined in Infotainment Diagnostic Specification. This menu shall be greyed out unless PaaKESN_St parameter ProvDID equals 0x54.

PaaK BSP menu (aka LBI) shall be accessible when the LBIServer has been provisioned and PaaK key(s) delivered to it. If the LBIServer has only been provisioned but PaaK key(s) were not delivered, PaaK BSP menu shall not be accessible.

Note: PaaK BSP Menu should be available if the DE configuration bits are set properly in LBIClient. Menu should be accessible if a Paak device is detected inside vehicle, and menu should be greyed out if a Paak device is not detected inside vehicle but it was fully set up.

BSP Menu option	BSP is setup	BSP not setup
Create	Enabled	Enabled
Reset	Enabled	Greyed out
Delete	Enabled	Greyed out

If PaaK BSP feature is setup and “Create” option selected, then the LBIServer shall search for available devices. If a PaaK device that was previously setup is detected, the LBIClient HMI shall display a pop-up telling the user that it found a PaaK device that already has BSP setup.

3.2.6 LBIv1-REQ-264925/C-Operation Modes for LBI Functions and Menus

The LBIClient shall determine the LBI available functions base on the vehicle inputs ignition(IgnitionStatus_St), transmission(GearLvlPos_D_Actl) and Customer Connectivity Settings (CCS) status to provide LBI functions required by LBIv1-REQ- 264936:

Vehicle Operation Mode			LBI Available Function	
CCS	Ignition	Transmission	Enter Password to Start Vehicle	Functions from Menu
On	Run	Park	N	Y

On	Off	Park		Y	N
Don't Care	Don't Care	Not in Park		N	N
Off	Don't Care	Don't Care		N	N

3.2.7 LBIv1-REQ-276037/A-Abort LBI Process Due to Driving Restriction

While in the middle of LBI process and the ignition changes from Run or the transmission changes from Park, then LBIClient shall display a message with driving restriction information, exit the process and return to the LBI Main Menu for creating, reset and deleting process and shall return to previous screen for other LBI processes.

3.2.8 LBIv1-REQ-275599/A-LBI Wait-Popup Display Duration

The LBIClient shall wait for the fix time defined by T_LBI Wait-Popup Display Timer before transitioning from any wait popup to the screen if in case when LBIClient is expecting a response from the LBIServer.

3.2.9 LBIv1-TMR-REQ-275603/B-T_LBI Wait Popup Display Timer

Name	Description	Units	Range	Resolution	Default
T_LBI Wait Popup Display Timer	The minimum time of displaying LBI Wait-popup before LBI HMI transits to the next screen Note: Use the default value	sec	2-5	1	3

3.2.10 LBIv1-REQ-270557/A-Restrictions Imposed by LBI Lockout

Once the vehicle is locked out as defined by REQ-268242 and is indicated in LBILockout_St(Active), the LBIClient shall display a lockout popup whenever the user attempts to enter a password to do the following functions:

- Reset a backup password
- Activate Enhanced Valet Password
- Deactivate Enhanced Valet Password
- Start the vehicle
- Exit Secure Idle

The LBI lockout shall not affect any operations of Paak feature such as starting a vehicle with a Paak devide while starting a vehicle with a backup password is not allowed

3.2.11 LBIv1-REQ-276915/B-Restrictions Imposed by LBI Enhanced Valet Mode

Once entering to Enhanced Valet Mode, the LBIClient shall apply the Legacy Valet mode HMI restrictions per H31a to Enhanced Valet Mode HMI and shall also meet the Legacy Valet mode requirement VS-FUR-REQ-104343/-Valet Mode Infotainment Operation.

3.2.12 LBIv1-REQ-276038/A-General Exit Enhance Valet Mode HMI Process

When the option to exit Enhanced Valet Mode is selected and the vehicle detects a key in the vehicle, the LBI HMI shall exit Enhanced Valet Mode and return to previous state. If a PaaK or a keyFob is not detected in the vehicle, the backup password entry screen shall be displayed. If the user enters a valid backup password at this screen, the LBI HMI shall exit Enhanced Valet Mode and return to previous state.

3.2.13 LBIv1-REQ-265041/A-Resend Request Requirement for LBIClient

After sending BackupIgnition_Rq, if there is no response or no confirmation within a defined period, T_LBI RetryTimer, or communication data is invalid or corrupted, the LBIClient shall re-send the request BackupIgnition_Rq up to N_LBINumberOfRetries times before quitting or moving to next step.

3.2.14 LBIv1-TMR-REQ-277532/A-T_LBI Retry Timer

Name	Description	Units	Range	Resolution	Default
T_LBI Retry Timer	The minimum time that the LBIClient shall wait before resend BackupIgnition_Rq for no response or communication error cases Note: Use the default value	msec	200-900	100	500

3.2.15 LBIv1-REQ-264938/A-Operation Abort Requirement

When multiple requests of BackupIgnition_Rq, required by REQ-265041, do not yield the correct response, the LBIClient shall abort the process entirely by doing the following actions:

- Set loss communication DTC if no communication can be established
- Provide the user HMI notification about the abort process status and shall instruct the user to recycle ignition before retry

3.2.16 LBIv1-REQ-263017/A-Lincoln Backup Ignition Specific Driving Restriction

The LBIClient shall impose driving restriction, defined by DRIVE-RESv2-FUR-REQ-025157-HMI Driving Restriction – General Applications, on LBI functions.

3.2.17 LBIv1-REQ-318311/B-Configurable Parameter for Keypad Code

The LBIClient and the LBIServer shall each have a configurable parameter to define the LBI Keypad code length for different markets as required by LBIv1-REQ-322144.

The usage of the LBI Keypad code configurable parameter include but not limited to

- For the LBIServer to determine mismatch between received Keypad code length and required length for different markets as required in LBIv1-REQ-318296
- For the LBIServer to determine the length of Enhanced Valet Password as described in LBIv1-REQ-317825

3.2.18 LBIv1-REQ-271253/B-LBIClient LBI Keypad Code Transmitting Format

When the LBIClient transmits a keypad code that is associated with a backup password to the LBIServer, it shall structure the data as a 7-button sequence where each button is represented by three bits.

- Per the Transport Protocol APIM SPSS, this request shall be sent via BackupIgnition_Rq with Byte 4 (Opcode) = “Keypad Code Create Request” (0x08), and Bytes 6-9 (KeypadCode) = the bit encoded button sequence.
- The mapping of button to bit value listed below shall be used to translate button pressed to data string:

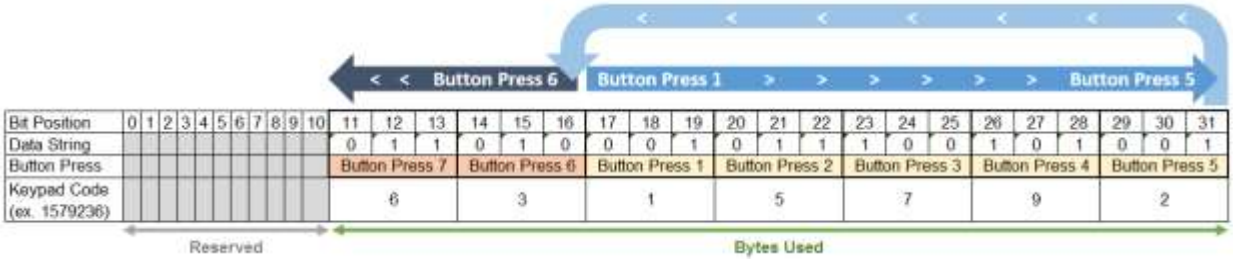
  - 000 = NULL
  - 001 = “1/2” button pressed
  - 010 = “3/4” button pressed
  - 011 = “5/6” button pressed
  - 100 = “7/8” button pressed
  - 101 = “9/0” button pressed

- When the vehicle configuration calls out for 7-digit Keypad codes, the Sixth and Seventh button press parameters shall be populated as requested as shown in Example1

- When the vehicle configuration calls out for 5-digit Keypad codes, the Sixth and Seventh button press parameters shall be set to NULL (000) as shown in Example 2

Example 1: 7-digit Keypad Code

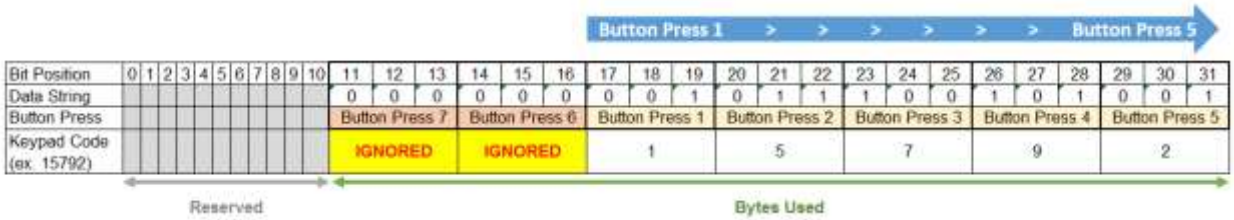
Below is an example of data string for a keypad code of **1579236** transmitted from the LBIClient to the LBIServer:



Example 2: 5-digit Keypad Code

Below is an example of data string for a keypad code of **15792** transmitted from the LBIClient to the LBIServer.

Note:
The Sixth and Seventh button press are ignored per LBIv1-REQ-318317
The bits in the data string for the Sixth and Seventh 6th and 7th button press are all set to Null as required.



3.2.19 LBIv1-REQ-283990/A-LBI HMI Pop-up Requirements

When an active popup screen is dismissed, the LBIClient shall suspend the Center Stack HMI display according to P06-Power Management specification provided below conditions are met:

- *IgnitionStatus_St = Off*
- *DelayAccy _St= Off*
- The Center Stack HMI is *not in Extended Play mode*

When above conditions are not met**,** the LBIClient shall not suspend the Center Stack HMI and shall transition to the previous screen for below use cases:

- *IgnitionStatus_St = Run*
- *DelayAccy_St= On*
- The Center Stack HMI *is in Extended Play mode*

3.3 Notification Requirements

3.3.1 LBIv1-REQ-260144/A-Notification Payload Contents

Whenever a user creates, starts the vehicle with, deletes, or resets a backup password, a notification shall be sent to the user that the backup password is associated with. This notification shall be sent as a push notification to the user's PaaK device and as an email to the user's email account (where available).

3.3.2 LBIv1-REQ-268177/A-Valet Password Creation Notification

Whenever a user authorizes creation of a valet password via the presence of PaaK device(s) in the vehicle, the valet password shall be delivered via Bluetooth as a push notification to all PaaK devices detected in the vehicle. No other notification shall be delivered to the user

3.3.3 LBIv1-REQ-268178/A-Valet Password Creation Notification Formats

Whenever a user authorizes creation of a valet password via a backup password, a notification shall be sent to the user that the backup password is associated with. This notification shall be sent via push notification to user's PaaK device and as an email to the user's email account (where available).

3.3.4 LBIv1-REQ-268179/A-Notification of Starting Vehicle with Valet Password

Whenever a user starts the vehicle with a valet password, a notification shall be sent to all users that received the valet password (if valet password creation was authorized by device(s)) or to the user that the backup password is associated with (if valet password creation was authorized by a backup password). This notification may be sent via push notification to user's PaaK device or via e-mail to e-mail address associated with user's owner account.

3.3.5 LBIv1-REQ-268180/A-Notification of Deleting Valet Password

Whenever a user deletes a valet password (i.e. exits Enhanced Valet Mode), a notification shall be sent to all users that received the valet password (if valet password creation was authorized by device(s)) or to the user that the backup password is associated with (if valet password creation was authorized by a backup password). This notification may be sent via push notification to user's PaaK device or via e-mail to e-mail address associated with user's owner account.

3.3.6 LBIv1-REQ-268181/A-Lockout Notification

Whenever a user is locked out with PaaK Backup or Valet password, a notification must be sent to all users of PaaK Backup.

3.3.7 LBIv1-REQ-321853/A-KeyID Parameter Definitions and Transmit Method in LBI Alerts

The LBIServer shall sent LBI event notification via the TP message LBIAlert_St with the payload defined per PaaK-REQ-270465, LBIv1-REQ-281410 and LBIv1-REQ-281412.

3.3.8 PaaK-REQ-270465/H-Payload Parameters Definitions (PaaK/LBI)

The following shall be used to populate the Payload data in the corresponding TP messages. The entire SyncP message will be sent to the PaakServer as base64. The Paak/LBI Server shall convert these values into hexadecimal and ASCII. The PaaK/LBI Server shall convert all Hexidecimal and ASCII to base 64 so that the entire SyncP message is base64 when sending to the PaaK/LBI Client.

Specifier	Description	Datatype	Base 64 encoded	Sample Value
keyid	The key id to identify a specific CAK	text	No	CEF95E28
keyname	The name of the CAK	text	No	bobiphone19X24C
muuid	The unique identifier of the mobile app instance	GUID	No	acb24a7d-be67-4eca-8980-de7c565bf120
cct	CAK Creation Time – the time the CAK was created	Timestamp (2017-10-18T:00:00:00:000Z)	Yes	MjAxNy0xMC0xOFQ6MDA6MDA6MDA6MDAwWg==
ced	CAK Expiry Date	Timestamp (9999-12-31T00:00:00:000Z)	Yes	OTk5OS0xMi0zMVQwMDowMDowMDowMDBa
profile	User role definition	Text	No	AD
kslt	CAK Salt	Hexadecimal	Yes	El802A==

		(125f34d8)		
kro	Key Revoke Origination	Hexadecimal	Yes	AA==
kp	Key Progress	Hexadecimal	Yes	AA==
rcode	Reason code	Hexadecimal	Yes	Ag==
event	What kind of events Ex: Key Installed	Hexadecimal	Yes	CQ==
BLEMProvDID	BLEM Prov DID	Hexadecimal	Yes	Ug==
BPEKHash	BPEKHash	Text	Yes	7Ri7wUaH6e3sOWpOi/sd2cTWMGngItgSHUstQSU6QrM=
F111	Hardware Part Number	Text	NO	LU5T-14G628-AA
F188	Software Partnumber	Text	NO	LU5T-14G623-AB
F10A	ECU Configuration	Text	NO	LC5T-14G627-AA
esn	Serial Number	Text	NO	G2E5FCE3
svctype	Service Type	Hexadecimal	Yes	MA==;
subtype	Service Subtype	Hexadecimal	Yes	Aw==
Etype	LBI Passcode Type (Ex: Valet Passcode or LBI Passcode)	Text	No	L

Event shall use the following values.

Code	Base 64 encoded	Decode
0x01	AQ==	LBI Key Created
0x02	Ag==	LBI Key Used
0x03	Aw==	LBI Key Deleted
0x04	BA==	Master Reset
0x05	BQ==	PaaK Reset
0x06	Bg==	LBI Key Reset
0x07	Bw==	LBI Key Lockout
0x08	CA==	Key Installed
0x09	CQ==	Key Revoked
0x0A	Cg==	BLEM Provisioning
0x0B	Cw==	Brand Reset
0x00, 0x0C-0xFF		Reserved

Key Progress shall use the following values.

Code	Base 64 encoded	Decode
0x01	AQ==	Success
0x02	Ag==	Failed
0x03	Aw==	Key Not Found

Key Revoke Origination shall use the following values.

Code	Base 64 encoded	Decode
0x00	AA==	Null
0x04	BA==	Master Reset

0x05	BQ==	PaaK Reset
0x06	Bg==	Brand Reset

Reason Code shall use the following values.

Code	Base 64 encoded	Decode
0x00	AA==	Null
0x01	AQ==	Out of Sequence
0x02	Ag==	Other Error
0x03 – 0xFF		Reserved

Event Type shall use the following values.

Code	Decode
L	LBI Passcode Type
V	Valet Passcode Type

Service Type shall use the following values:

Service Type	Base64 encoding
0x40 (LBI Events)	QA==
0x30 (PaaK Events)	MA==

The following conventions shall be used for the SyncP Payload data.

- Comma (,) = array separator
  - Shall be used when more than one value is present for any given value pair
  - Example: keyid:CEF95E28,AEF95E7,BEF95E6,GEF95E5;
- Semicolon (;) = value pair separator
  - Shall be used to separate different key pair values from the next pair
  - Example: keyid:CEF95E28;kp:0x01;kro:0x00;event:r;
- Colon (:) = name/value separator
  - Shall be used to separate the name from the value portion of the content
  - Example: kp:0x01
- Space (‘ ‘) = not used
  - The payload data shall not contain spaces

3.3.9 LBIv1-REQ-281410/B-SYNCP Payload Parameters

The following table shows the FTCP Messages to TP messages with payload contents and if the payload requires SYNCP Encryption or SYNCP Signing.

Further information about the SyncP message can be found in the LBIServer Security Spec and S13_SyncP specifications.

Service Type	Description	FTCP Message	TP Message	Sub-Service Type for SYNCP	Payload Parameters	Encrypted	Signed
0x40	LBI Event	CAK Status Alert	LBIAlert_St	0x0	KeyID; Key Progress; Key Revoke Origination; Event; Event Type; Reason Code; ESN; Service Type;	No	Yes

					Service Subtype		
0x40	Lockout	CAK Status Alert	LBIAlert_St	0x1	KeyID; Key Progress; Key Revoke Origination; Event; Event Type; Reason Code; ESN; Service Type; Service Subtype	No	Yes

3.3.10 LBIv1-REQ-281412/B-LBI Notification Payload for BLEMSyncPPacket

The LBIServer shall create the following SyncP payload in LBIAlert_St.

Service Type	Service Sub Type	Payload with sample data
0x40 LBI Event	0x0 (Created/Usage/Deletion/Reset)	keyid:CEF95E28;kp:AQ==;kro:AA==;event:AQ==;etype:L;rcode:AA==;esn:A2E5FCE;svctype:QA==;subtype:AA== keyid:CEF95E28,<keyed>,<keyed>[max 4];kp:AQ==;kro:AA==;event:Ag==;etype:L;rcode:AA==;esn:A2E5FCE;svctype:QA==;subtype:AA== keyid:CEF95E28;kp:AQ==;kro:AA==;event:Aw==;etype:L;rcode:AA==;esn:A2E5FCE;svctype:QA==;subtype:AA== keyid:CEF95E28;kp:AQ==;kro:AA==;event:Bg==;etype:L;rcode:AA==;esn:A2E5FCE;svctype:QA==;subtype:AA==
	Failure (Creation Failed) kp:Ag==;	Failure: keyid:CEF95E28;kp:Ag==;kro:AA==;event:AQ==;etype:L;rcode:AA==;esn:A2E5FCE;svctype:QA==;subtype:AA==
0x40 LBI Event	0x1 Lockout	keyid:null;kp:AQ==;kro:AA==;event:Bw==;etype:L;rcode:AA==;esn:A2E5FCE;svctype:QA==;subtype:AQ==
0x40 LBI Event	0x0 (Enhanced Valet Created/ Usage/ Deletion)	keyid:CEF95E28,AEF95E7,BEF95E6,GEF95E5;kp:AQ==;kro:AA==;event:AQ==;etype:V;rcode:AA==;esn:A2E5FCE;svctype:QA==;subtype:AA== keyid:CEF95E28,AEF95E7,BEF95E6,GEF95E5;kp:AQ==;kro:AA==;event:Ag==;etype:V;rcode:AA==;esn:A2E5FCE;svctype:QA==;subtype:AA== keyid:CEF95E28,AEF95E7,BEF95E6,GEF95E5;kp:AQ==;kro:AA==;event:Aw==;etype:V;rcode:AA==;esn:A2E5FCE;svctype:QA==;subtype:AA==

3.4 Security Requirements

3.4.1 LBIv1-REQ-260155/A-Password Cleartext Transmission Restriction

Passwords shall never be transmitted across any interface in cleartext.

Exception: For U611/U554/CX483 implementation, transmission of the valet password to the LBIClient and to the mobile app for display will be in cleartext.

3.4.2 LBIv1-REQ-260156/A-LBI Password Types

The LBIServer shall recognize two different types of passwords: a “backup” passwordand a “temporary” password.

Backup passwords are intended for long-term use, while temporary passwords are short-term passwords intended for usage in valet-type scenarios.

3.4.3 LBIv1-REQ-260157/A-Passwords Storage Location

The LBIServer shall securely store within the HSM all customer passwords created and used to enable vehicle start and drive-away.

3.4.4 LBIv1-REQ-260158/A-Passwords Storage Format

Passwords shall never be stored in the clear, but instead shall be stored in a salted and hashed format, defined as:

$$Programmed\ Hash = SHA256(Salt + Password)$$

Note that the output of SHA256 will always be a 256-bit (32-byte) hash – thus storage requirements are consistent regardless of password length.

3.4.5 LBIv1-REQ-260159/B-Generating Condition and Storage Location for Salt

Upon completing provisioning, the LBIServer shall generate a random a128-bit salt. This salt may be stored in unsecured memory (i.e. outside the HSM).

3.4.6 LBIv1-REQ-260160/A-Conditions of Accepting New Password

The LBIServer shall not accept any new backup passwords for LBI unless all of the following conditions have been met:

- The customer has opted-in to using the feature (by way of LBIClient)
- At least one Phone-as-a-Key device authorized and enabled for the given vehicle (i.e. with a provisioned CAK) is inside the vehicle and is in session (see BLE Interface Security Specification)
- A password does not already exist for the given PaaK device/vehicle pairing
- At least one KeyFob is detected inside the vehicle

3.4.7 LBIv1-REQ-260161/A-Device Check Conditions

PaaK device checks and key fob checks shall be executed when the LBIServer has received a hashed password to be stored.

This mitigates potential time of check/time of use vulnerabilities

3.4.8 LBIv1-REQ-260162/A-Internal Mapping between Password and CAK

All backup passwords MUST be associated internally on the LBIServer to a CAK authorized for the given vehicle.

3.4.9 LBIv1-REQ-260165/B-No Direct Display of Backup Password on HMI

Backup passwords shall not be directly displayed on any HMI.

3.4.10 LBIv1-REQ-260168/B-Password Transmission Mechanism

Backup passwords shall be transmitted from the in-vehicle HMI in a salted and hashed format ("Programmed Hash"), using the following mechanism:

$$\text{Programmed Hash} = \text{SHA256}(\text{Salt} + \text{Password})$$

Note: the "+" operator in above calculations means append, not add. The salt and nonce values must be a lower case hex bytes. There is no space between (salt and password) string, it is one string of hex values. The LBIClient, must convert the password that user is trying to create from asci value into hex string before proceeding with Programmed Hash calculation.

Calculation Example
Programmed hash = SHA256(e7e272cf7bbf00ab9874ad03bde24626717765313233) =
35effe7d0c505913286470ba328f835da9067cd1c9ca2817102db7ec6f95b4c8
Where
Salt = e7e272cf7bbf00ab9874ad03bde24626
Desired password = qwe123 → converted to hex = 717765313233

3.4.11 LBIv1-REQ-260170/A-Select one Device from PaaK List

If more than one PaaK device is discovered within the vehicle without an associated backup password, the user shall be prompted to select a device to associate the password with.

3.4.12 LBIv1-REQ-260171/A-PaaK inside Vehicle Required to Delete Password

Customers shall be required to have a PaaK device inside the vehicle to delete its password via the in vehicle HMI

3.4.13 LBIv1-REQ-260172/A-Password Deletion by CAK Revoking

If a CAK is revoked for any reason and a backup password is associated to that CAK, the backup password shall immediately be deleted and not allowed for further usage.

3.4.14 LBIv1-REQ-260174/A-Password Authentication Mechanism

A challenge/response mechanism shall be used with the hashed password to authenticate and start the vehicle via the in-vehicle HMI.

3.4.15 LBIv1-REQ-260175/A-LBIServer Nonce Requirement

On request, the LBIServer shall generate a random 256-bit nonce using the best available RNG. If possible, a true RNG shall be used. This nonce shall be sent to the LBIClient (the in-vehicle HMI module).

3.4.16 LBIv1-REQ-260176/B-LBIClient Programmed Hash Requirement

When the user enters a password, the in-vehicle HMI module, the LBIClient, shall perform two rounds of hashing before transmitting the response to the LBIServer.

The first shall calculate the Programmed Hash:

$$Programmed\ Hash = SHA256(Salt + Backup\ Password)$$

Note:

- The second round of hash, Authentication Hash, is defined in LBIv1-REQ-260177
- Refer to BIv1-REQ-260168 for "+" operator notation and calculation example

3.4.17 LBIv1-REQ-260177/B-LBIClient Authentication Hash Requirement

When the user enters a password, the in-vehicle HMI module, the LBIClient, shall perform two rounds of hashing before transmitting the response to the LBIServer.

Once the Programmed Hash has been calculated as required by LBIv1-REQ-260176, the in-vehicle HMI module, the LBIClient, shall then calculate and transmit the "Authentication Hash", using the following mechanism:

$$\text{Authentication Hash} = \text{SHA256}(\text{Nonce} + \text{Programmed Hash})$$

Note: the "+" operator in above calculations means append, not add. The nonce value must be a lower case hex byte. There is no space between (nonce and programmed hash) string, it is one string of hex values.

Calculation Example
Authentication hash = SHA256(fb03845b994809d8b265aba4a7c7c64235d125151d7f1eb3fcf42252148a483435effe7d0c505913286470ba328f835da9067cd1c9ca2817102db7ec6f95b4c8) = 7a86e7379848b4106150e3674aaf0f99a1b408588d8b402742782e6db0d16c3e
Where
Nonce = fb03845b994809d8b265aba4a7c7c64235d125151d7f1eb3fcf42252148a4834

Programmed Hash = 35effe7d0c505913286470ba328f835da9067cd1c9ca2817102db7ec6f95b4c8

Refer to Transport Protocol APIM SPSS (BackupIgnition_Rq and BackupIgnition_Rsp) for more details on where to apply Programmed hash and where to apply Authentication hash.
BLEM and Sync both shall validate the output of Programmed hash or Authentication hash depending upon which opcode they are comparing

3.4.18 LBIv1-REQ-260178/A-LBIServer Authentication Hash Requirement

Upon sending the nonce to the in-vehicle HMI module (the LBIClient), the LBIServer shall also calculate the Authentication Hash for all stored passwords within 5ms.

Note: This is done because the LBIServer does not know ahead of time which password to expect, and thus it must compare the received value to all known passwords to find a match.

3.4.19 LBIv1-REQ-260180/A-Master of Password Authenticating System

The LBIServer shall be the master of password authenticating system which determines if a password is valid or invalid and also determines when to lockout LBI feature.

3.4.20 LBIv1-REQ-260181/A-Definitions of Authenticating Lockout

If the user does not provide a valid password within 5 attempts, the password authenticating system shall lock out further attempts for 5 minutes

After the first 5-minute lockout, the user shall be permitted 5 additional attempts to enter a valid password. If the user does not enter a valid password within these 5 attempts, the system shall lock out further attempts for another 5 minutes. This lockout process shall repeat indefinitely.

3.4.21 LBIv1-REQ-260183/A-Conditions of Creating a Temporary Password

The user shall be required to have a PaaK device with or without an associated backup password inside the vehicle or enter a valid backup password to create or delete a temporary password.

3.4.22 LBIv1-REQ-260192/A-Valid Period of Temporary Password Hash

The temporary password hash shall be valid until the system exits Enhanced Valet Mode.

3.4.23 LBIv1-REQ-260193/A-Conditions for Exit Enhanced Valet Mode

The system shall require a valid backup password, PaaK-authenticated device in vehicle, or key fob in vehicle, together with confirmation from the user to exit Enhanced Valet Mode.

3.4.24 LBIv1-REQ-260194/A-LBI Alert

Upon creation, successful usage, deletion, or reset of any backup or valet password, the LBIServer shall send a SyncP signed message (Service Type 0x40/Sub-Service 0x0) to the SDN via the LBIClient2. The payload shall contain:

- The Key ID of the associated CAK (or CAK(s) for valet password creation)
- The timestamp of the action
- An indicator of whether a password was created, successfully used, deleted, or reset

- An indicator of whether the event was for a backup password or an Enhanced Valet password
- An indicator of whether a password, KeyFob, or PaaK device authorized Enhanced valet password creation

3.4.25 LBIv1-REQ-273336/B-LBI Alert Queuing

The LBIClient2 shall queue the LBI event alerts (to be sent per REQ-273337) in case of a connectivity issue with the SDN.

3.4.26 LBIv1-REQ-260195/A-Lockout Notification Payload Contents

Upon lockout of the system, the LBIServer shall send a SyncP signed message (Service Type 0x40/Sub-Service 0x1) to the SDN via LBIClient2. The payload shall contain:

- The timestamp of the action
- An indicator of whether or not valet mode was active at the time

Note: The LBIClient2 *only relays these SyncP messages to the SDN. It does not process these messages.*

3.4.27 LBIv1-REQ-275186/B-SyncP Message Process for SDN

When the SDN receives a SyncP message from the LBIServer via LBIClient2, it shall verify that the SyncP signature is correct (by the LBIServer ESN) and then forward this message to PaaK FI for processing.

3.4.28 LBIv1-REQ-275185/A-Retention of LBI Event History

PaaK FI shall keep a history of LBI events, defined in REQ-260194.for one year.

3.4.29 LBIv1-REQ-322144/A-LBI Keypad Code Length Definition

For security concerns, the vehicle has a different length of Keypad code in different markets as defined in PDL XXXX(TBD). The LBI feature shall comply with the vehicle Keypad code length requirement for LBI Keypad code.

Note: Some markets only require 5-digt Keypad code whereas other markets require 7-digit Keypad code. As LBI feature deploys globally, it is possible that length other than 5-digit and 7-digit code could be added for different regions and markets. It is also possible that security concerns will make Keypad longer at any time. Readers shall refer to the PDL XXX (TBD) for updated information.

3.4.30 LBIv1-REQ-322204/B-LBI Keypad Code Composition Restriction

When a customer is creating a personal keypad code, the SYNC shall verify customer acceptable codes based on the vehicle configuration (e.g. there are markets that require the use of 7-digit codes and 5-digit codes).

If SYNC determines the configuration is for 7-digit codes, then it shall apply the following restrictions:

- Desired door keypad code cannot consist of all the same button presses
  - customer presses button **"1/2"** seven (7) times
  - customer presses button **"3/4"** seven (7) times
  - customer presses button **"5/6"** seven (7) times
  - customer presses button **"7/8"** seven (7) times
  - customer presses button **"9/0"** seven (7) times

When a single button has been pressed six (6) consecutive times, that button shall become grey out for the last digit entry and SYNC screen shall also be populated with instruction stating "Desired 7-digit keypad code must not consist of selecting the same button seven times"

If SYNC determines the configuration is for 5-digit codes, then it shall not apply the above restrictions.

3.4.31 LBIv1-REQ-351821/A-Time Out After Waiting for a Challenge Response

The LBIServer shall implement an expiration timer T_ChallengeTimeout when comparing the password hashes against the received authentication hash value from the LBIClient.

The timer T_ChallengeTimeout shall be initiated when (as a response to a Challenge Request from the LBIClient), the LBIServer sends the **first frame** of the BackupIgnition_Rsp with the RspCode = 0x01 *Issue Challenge* and RspStatus = 0x00 *Reserved*.

The timer shall then be terminated when the LBIServer receives the **last frame** of the BackupIgnition_Rq with the OpCode = 0x02 *Challenge Response*. (Note: If this last frame is received prior to the expiration of the timer T_ChallengeTimeout, and the remaining payload has been successfully validated, then the LBIServer shall respond back to the LBIClient via the BackupIgnition_Rsp with RspCode = 0x02 *Challenge Response Acknowledge* and RspStatus = 0x0F: Valid Password.)

If the timer T_ChallengeTimeout expires prior to receiving the last frame of the BackupIgnition_Rq with the OpCode = 0x02 *Challenge Response*, the LBIServer shall respond back to the LBIClient via the BackupIgnition_Rsp with RspCode = 0x02 *Challenge Response Acknowledge* and RspStatus = 0x10 *Invalid Password*.

3.4.32 LBIv1-TMR-REQ-351820/A-T_Challenge Timeout Expiration Timer

Name	Description	Units	Range	Resolution	Default
T_Challenge Timeout Expiration Timer	The maximum time the LBIServer shall wait during the challenge time out period. Note: Use the default value	sec	1-5	1	3

4 Functional Definition

4.1 LBIv1-FUN-REQ-258448/A-Creating Backup Password for Paak Device

4.1.1 Use Cases

4.1.1.1 LBIv1-UC-REQ-258458/A-Creating Backup Password for Paak Device

Actors	The LBI User.
Pre-conditions	1. The LBI User has previously activated Phone-as-a-Key feature for the vehicle 2. The vehicle ignition Status is in Run 3. The vehicle transmission is in Park 4. The LBI User is inside vehicle 5. One associated phone-as-a-key and one key fob are inside the vehicle
Scenario Description	1. The LBI User selects option to Create Backup Password for Phone-as-a-Key from the LBI Main Menu at the Center Stack HMI 2. The LBI HMI displays a screen with password creation steps 3. The LBI Users is asked to pick a Paak device to create a backup password for 4. The LBI HMI displays alphanumeric password entry screen and instructs the LBI User to enter a backup password 5. The LBI User enters password twice according to password requirements 6. The LBI User selects Enter 7. The LBI HMI displays a message that a backup password has been created successfully 8. The LBI HMI asks the LBI User if he/she would like to create a LBI keypad code 9. The LBI User opts out LBI keypad code creations
Post-conditions	1. PaaK Backup is ready for use 2. A Notification of a backup password has been created is sent to the LBI User via e-mail and via a phone message to the PaaK device which a backup password is just created and associated with
List of Exception Use Cases	1. The LBI User enters password that does not meet requirements 2. The LBI User enters passwords that do not match 3. The LBI User enters passwords that are already in use
Interfaces	The LBIClient The LBIClient2 The LBIServer The LBIServer2 SDN PaakFI

4.1.2 Requirements

4.1.2.1 LBIv1-REQ-276039/A-Number of Backup Password per PaaK Device

The LBIServer shall allow only one LBI Backup Password created for one PaaK device.

4.1.2.2 LBIv1-REQ-260055/B-Initiate Password Creation after CAK Created

The LBIServer shall keep track of whether PaaK devices inside the vehicle have an associated backup password.

If the LBIServer detects one such device while the ignition state is RUN as indicated in IgnitionStatus_St(Run) and the vehicle is in Park as indicated in GearLvrPos_D_Actl(Park), it shall notify LBIClient via LBISetup_Rq(Active) *five times,* once per ignition cycle, per lifetime of the CAK associated with that device.

4.1.2.3 LBIv1-REQ-260056/B-HMI Display of Paak Backup Password Creation Option

Upon receiving PaaK without password existence notification, LBISetup_Rq(Active), or when the use select to create a backup password from the LBI Main Menu, the LBIClient HMI shall display a screen that provides the backup password information and creation steps as well as the option button to create a backup password. The LBIClient HMI shall still allow this in the scenario when the User has disabled vehicle connectivity via the CCS menu, however the password would not become functional until vehicle connectivity resumes.

4.1.2.4 LBIv1-REQ-260057/A-Query for PaaKs without Passwords

Upon receiving the user request of creating a backup password from the option popup as defined inREQ-260056 or from the vehicle Centerstack Setting Menu, LBIClient shall query the LBIServer for PaaK devices without passwords in the vehicle and shall request the cryptographic salt from the LBIServer.

The query and request of salt shall be sent through BackupIgnition_Rq(OpCode= “Salt and Check for PaaK without Passwords”, *KeyIndex= 0x00, Password = EOS, KeypadCode = EOS*).

4.1.2.5 LBIv1-REQ-264576/B-Trigger Interior Registry KeyFob Search for Paak without Password

Upon receiving the query of PaaK devices without passwords in the vehicle and the request the cryptographic salt, BackupIgnition_Rq(OpCode=”Salt and Check for PaaK without Passwords”, KeyIndex=0x00) from the LBIClient, the LBIServer shall also trigger a LBIServer2 Interior Registry search to determine if a Keyfob is found inside the vehicle by executing LBIv1-FUN-REQ-302285-LBI KeyFob Search.

4.1.2.6 LBIv1-REQ-260061/B-Response of Paak w/o Password

In response to BackupIgnition_Rq(OpCode=”Salt and Check for PaaK without Passwords”), the LBIServer shall report to the LBIClient what devices (PaaK w/o passwords and/or key fob) were found in the vehicle, the names and key indexes of all PaaK devices found in the vehicle, as well as the cryptographic salt via BackupIgnition_Rsp with the encoding values set as:
RspCode = “Salt and Check For PaaK without Passwords Response”
RspStatus =

- “One PaaK w/o Password and Fob In Vehicle” or
- “One PaaK w/o Password and No Fob In Vehicle” or
- “Fob in Vehicle and No PaaK w/o Password or
- ”Two+ PaaK w/o Password and Fob In Vehicle or”
- ”Two+ PaaK w/o Password and No Fob In Vehicle or”
- ”No PaaK w/o Password and No Fob In Vehicle”

Salt = Salt
KeyIndex = KeyIndex
PhoneName = PhoneName (*Note: Name here refers to the device name generated during PaaK setup)*
)
The Key fob status shall base on the result of executing LBIv1-REQ-302288/-Authentication of KeyFob Search Response

4.1.2.7 LBIv1-REQ-260058/A-HMI Display for Missing Devices

The LBIClient HMI shall display an error message of missing devices and an instruction of obtaining required devices to continue the creating process when receiving missing device report via BckupIgnition_Rsp with
RspStatus=“Salt and Check for Paak without Passwords Response”
RspStatus=

- “Fob in Vehicle and No PaaK w/o Password” or
- One PaaK w/o Password and No Fob In Vehicle or
- Two+ PaaK w/o Password and No Fob In Vehicle or
- No PaaK w/o Password and No Fob In Vehicle

.

4.1.2.8 LBIv1-REQ-260063/A-HMI Display of PaaKs without Password

If PaaK devices and a Key fob are detected during LBIClient requested key search, indicated in BackupIgnition_Rsp (RspCode = “Salt and Check for Paak without Passwords Response”, RspStatus = “One PaaK w/o Password and Fob In Vehicle” or “Two+ PaaK w/o Password and Fob In Vehicle”), the LBIClient HMI shall display a list of all detected PaaK devices based on information in BackupIgnition_Rsp, with instructions for the user to choose the desired device.

4.1.2.9 LBIv1-REQ-260062/A-HMI Display of Password Entry

The LBIClient HMI shall display the backup password creation screen when a Paak without password is identified.

The identified PaaK could be the only one Paak without Password detected by the LBIServer indicated in Rsp Status= “One PaaK w/o Password and Fob In Vehicle”.

The identified Paak could also be the one Paak chosen by the user from the HMI list of all detected phones.

In either case, LBIClient shall retain the keyindex in order to send it along with entered password for passwords transmit operation defined in R EQ-260066.

4.1.2.10 LBIv1-REQ-267308/A-HMI Display for Backup Password Rules

LBIClient shall make Password Rules, as defined in REQ-260164 and REQ- 264928, available for users upon request.

4.1.2.11 LBIv1-REQ-260164/C-Backup Password Minimum Length Requirement

The LBIClient shall enforce a minimum password length as below:

- For CGEA1.3C architecture vehicle programs, the LBI passwords shall be
  - a minimum of 5 characters in length if a mixture of letters, numbers, and symbols are used
  - a minimum of 8characters in length if only numbers are used

- For FNV2 architecture vehicle programs, the LBI passwords shall be
  - a minimum of 6 characters in length if a mixture of letters, numbers, and symbols are used
  - a minimum of 10characters in length if only numbers are used

4.1.2.12 LBIv1-REQ-264928/B-Backup Password Maximum Length Requirement

The LBIClient shall limit LBI password length up to 64 characters.

4.1.2.13 LBIv1-REQ-260065/B-Check Entered Password against Password Rules

In the Process of creating or resetting a backup password, the LBIClient shall check entered password against password requirements in real time.

The LBIClient hall not allow the user to proceed to the next screen until their password meets the minimum requirements as defined in LBIv1-REQ-260164 and LBIv1-REQ-264928.

4.1.2.14 LBIv1-REQ-260166/A-Backup Password Strength Indicator

An indicator shall be displayed on the HMI indicating the relative strength of the user’s selected password.

4.1.2.15 LBIv1-REQ-260167/A-Contents of Backup Password Strength Meter Indicator

The strength indicator shall have four levels of strengths: weak, fair, good, and strong. These levels shall be indicated using a four-segment fill bar. If minimum requirements are not met, bar shall be empty. If password is weak, bar shall fill ¼ with red color. If password is fair, bar shall fill ½ with orange color. If password is good, bar shall fill ¾ with yellow color. If password is strong, bar shall fill completely with green color.

4.1.2.16 LBIv1-REQ-265042/C-Backup Passwords Strength Definitions for CGEA1.3C

The LBIClient shall use the following rules to determine password strength when deployed on the CGEA 1.3C architecture:

1. Weak: Password must have at least eight (8) characters if password consists only of numbers or at least five (5) characters if password does not consist only of numbers.

2. Fair: Password must have at least eight (8) characters including at least one (1) lower-case letter, one (1) upper-case letter, and one (1) number.

3. Good: Password must have at least ten (10) characters including three of the following four types of characters: lower-case letter, upper-case letter, number, special character (including space). Password also must have no more than two identical characters in a row

4. Strong: Password must have at least twelve (12) characters including three of the following four types of characters: lower-case letter, upper-case letter, number, special character (including space). Password also must have no more than two identical characters in a row.

5. The security guidelines from the document" Authentication General Guidelines" provided from the link listed below:

   https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html

6. The ranking shall be marked not higher than "fair" if the password is found in common password lists provided from the link listed below: https://github.com/danielmiessler/SecLists/tree/master/Passwords

   https://github.com/danielmiessler/SecLists/blob/master/Passwords/xato-net-10-million-passwords-10000.txt

Note: This requirement is cascaded to H31m.R008.

4.1.2.17 LBIv1-REQ-360808/A-Backup Passwords Strength Definitions for FNV2

The LBIClient shall use the following rules to determine password strength when deployed on the FNV2 architecture:

1. Weak: Password must have at least ten (10) characters if password consists only of numbers or at least six (6) characters if password does not consist only of numbers.

2. Fair: Password must have at least ten (10) characters including at least one (1) lower-case letter, one (1) upper-case letter, and one (1) number.

3. Good: Password must have at least twelve (12) characters including three of the following four types of characters: lower-case letter, upper-case letter, number, special character (including space). Password also must have no more than two identical characters in a row

4. Strong: Password must have at least fourteen (14) characters including three of the following four types of characters: lower-case letter, upper-case letter, number, special character (including space). Password also must have no more than two identical characters in a row.

5. The security guidelines from the document" Authentication General Guidelines" provided from the link listed below:

   https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html

6. The ranking shall be marked not higher than "fair" if the password is found in common password lists provided from the link listed below:

   https://github.com/danielmiessler/SecLists/blob/master/Passwords/xato-net-10-million-passwords-10000.txt

Note: This requirement is cascaded to H31m.R008.

4.1.2.18 LBIv1-REQ-260064/A-Check Match for Two Entered Passwords

In the process of creating or resetting a backup password, the LBIClient HMI shall check if two passwords match.

In the cases of passwords do not match, the LBIClient HMI shall notify the user and provide an option to retry.

4.1.2.19 LBIv1-REQ-270524/A-HMI Solicit Password Twice

Once a rule-complied password is entered, the LBIClient HMI shall require the user to enter that password twice, and shall check if both codes match in real time.

4.1.2.20 LBIv1-REQ-260066/A-Transmit Rule-Complied Password

In the creating or resetting backup password process, once a rule-complied password is entered twice, the LBIClient shall compute a hash of entered password and shall send it to the LBIServer with the key index of the selected device via BckupIgnition_Rq as defined in REQ-267240.

4.1.2.21 LBIv1-REQ-267240/A-Transmit Password

Whenever there is a need to transmit entered password to the LBIServer for validation, the LBIClient shall send hashed entered password with the key index of the selected device via BackupIgnition_Rq with encoding value set as

OpCode=

- "Password Transmit", when creating a brand new password (not in the reset process)

or

- "Reset 1 Password Transmit", when resetting password with BackupIgnition_Rsp (RspCode= "Salt and Check for Paak with Passwords Response", RspStatus= "One PaaK w/ Password and No Fob In Vehicle" or "Two+ PaaK w/ Password and No Fob In Vehicle").

or

- "Reset 2 Password Transmit", when resetting password with BackupIgnition_Rsp (RspCode= "Salt and Check for Paak with Passwords Response", RspStatus= "One PaaK w/ Password and Fob In Vehicle" or "Two+ PaaK w/ Password and Fob In Vehicle").
-

KeyIndex = KeyIndex
Password = Password

4.1.2.22 LBIv1-REQ-271305/A-Search Paak for Password Response

In the creating or resetting backup password process, upon receiving the hashed password and keyindex via BackupIgnition_Rq (OpCode= "Password Transmit" or "Reset 1 Password Transmit" or Reset 2 Password Transmit), the LBIServer shall check that the phone associated with the received key index (Reset 1 and 2) and key fob (Reset 2) is still in the vehicle.

4.1.2.23 LBIv1-REQ-260067/B-Trigger Interior Registry KeyFob Search for Password Response

In the creating or resetting backup password process, upon receiving the hashed password and keyindex via BackupIgnition_Rq (OpCode= "Password Transmit" or" Reset 2 Password Transmit"), the LBIServer shall check that the phone associated with the received key index and key fob are still in the vehicle via an Interior Registry KeyFob search by executing LBIv1-FUN-REQ-302285.

4.1.2.24 LBIv1-REQ-263584/B-Response of No Device for Password Transmit

During the creating backup password process, if the LBIServer determines that the phone associated with the received key index or key fob is no longer in the vehicle per LBIv1-REQ-302288 for the response of BackupIgnition_Rq (OpCode= "Password Response"), it shall notify the LBIClient of this via BackupIgnition_Rsp.with the encoding values set as below:

RspCode="Password Response"
RspStatus=

- "PaaK No Longer Detected" or
- "Fob No Longer Detected" or
- "PaaK and Fob No Longer Detected"

VariableData shall set to zero

4.1.2.25 LBIv1-REQ-260068/A-HMI Display of No Device for Password Transmit

Upon receiving the notification that the user selected Paak and or a key fob is no longer in the vehicle via BackupIgnition_Rsp, the LBIClient HMI shall then display a message that the phone and/or key fob is no longer detected in the vehicle and instruct user to bring required devices to vehicle before retry.

The devices no longer detected condition is indicated in BackupIgnition_Rsp with the encoding values of
RspCode=

- ”Password Response” or
- “Reset 2 Password Response”

RspStatus=

- “PaaK No Longer Detected”, or
- ”Fob No Longer Detected” or
- “PaaK and Fob No Longer Detected”.

4.1.2.26 LBIv1-REQ-267620/B-Check Password Uniqueness

In response of BackupIngintion_Rq (OpCode=” Password Transmit” or “Reset 1Password Transmit” or “Reset 2 Password Transmit”), if the required devices are still in the vehicle when the LBIServer receives the password hash, it shall then verify that the entered password is not already being used.

The required devices are defined as

- The selected Paak and a KeyFob are both needed for OpCode=” Password Transmit” or “Reset 2 Password Transmit”
- Only the selected Paak is needed for OpCode=” Reset1 Password Transmit”.

4.1.2.27 LBIv1-REQ-263585/A-Response of Password Already Used

In the response of transmitted password, if the entered password is already being used, the LBIServer shall notify the LBIClient via BackupIgnition_Rsp with the encoding values set as below:

RspCode=

- “Password Response” when OpCode=“Password Response” or
- “Reset 1Password Response” when OpCode=“Reset 1Password Response” or
- “Reset 2 Password Response” when OpCode=“Rest2 Password Response”

RspStatus=”Password Already Used”
VariableData shall set to zero

4.1.2.28 LBIv1-REQ-260069/A-HMI Display for Password Already Used

Upon receiving the notification of Password already being used via BackupIngnition_Rsp(RspStatus=”Password Already Used”), the LBIClient HMI shall display a message that password is already being used and instruct user to enter a different password.

4.1.2.29 LBIv1-REQ-263587/A-Brand New Password Stored Conditions

In the process of creating a brand new backup password, if the entered password is not already being used and both a Paak without password and a Keyfob are still present, the LBIServer shall store the hashed password in its HSM and shall associate it with the key index it received from the LBIClient via BackupIngintion_Rq(OpCode=”Password Transmit”).

4.1.2.30 LBIv1-REQ-270515/B-Notification of Successful Password Creation to LBIClient2

Once a password is stored as defined per LBIv1-REQ-263587(for brand new password creation) or LBIv1-REQ-269569(for reset password), the LBIServer shall report this event, including the corresponding device KeyID, to the LBIClient2 via *LBIAlert_St with encoding values set as below:*

Event =

- *” Backup Password Created” if the Opcode is” Password Transmit”*

or

- *” Backup Password Reset” if the Opcode is” Rest 1 Password Transmit “or ” Rest 2 Password Transmit”*

Source = Reserved

The corresponding device KeyID shall be *the KeyID of PaaK device associated with the backup password that is being created or reset.*

4.1.2.31 LBIv1-REQ-267309/A-Response of Successful Password Creation to LBIClient

Once a password is stored per REQ-263587(for brand new password creation) or REQ-269569(for reset password), the LBIServer shall report to the LBIClient the successful storage of this password via BackupIgnition_Rsp with
RspCode= “Password Response” or “Reset 1 Password Response”, or Rest 2 “Password Response”
RspStatus=”Password Created Successfully
VariableData shall all set to be zero

4.1.2.32 LBIv1-REQ-260070/A-HMI Display for Successful Password Creation

Upon receiving notification of successful password creation via BackupIgnition_Rsp (RspStatus=”Password Created Successfully”), the LBIClient HMI shall notify the user of successful backup password creation.

4.1.2.33 LBIv1-REQ-260071/A-HMI Display of Keypad Code Creation Option

After notifying the user of successful creating or resetting a backup password, the LBIClient HMI shall also present the user the option to set up a personal keypad code.

4.1.2.34 LBIv1-REQ-260072/A-HMI Diaplay of Password Creation Complete

If the user chooses not to create a personal keypad code, the LBIClient HMI shall inform the user that backup password setup is complete and provide instructions on how to use the feature.

4.1.2.35 LBIv1-REQ-269633/A-Initiate Keypad Code Creation

If the user chooses to create a LBI keypad code right after a backup password created or reset, the LBIClient shall initiate the keypad code creation process as described in LBI-FUN-REQ-26957-Creating Keypad Code for Paak Device.

4.1.2.36 LBIv1-REQ-275724/A-Criteria of Reporting Failed Password Creation

The LBIServer shall report a failed password creation if any one of conditions below is met:

- The vehicle operation conditions defined by REQ- 264925 are no longer met before the deletion process completes
- The entered and verified password is unable to be stored by the LBIServer

4.1.2.37 LBIv1-REQ-275725/A-Response of Failed Password Creation to LBIClient

In the creating password process, if the failure criteria described in REQ-275724 is met, the LBIServer shall report to the LBIClient the unsuccessful password creation via BackupIgnition_Rsp with
RspCode= “Password Response”
RspStatus=”Password Created Failed”
VariableData shall all set to be zero

4.1.2.38 LBIv1-REQ-275726/A-HMI Display of Failed Password Creation

Upon receiving *BackupIgnition_Rsp* (*RspCode =”Password Response”, RspStatus =”Password Created Failed”)*, the LBIClient shall notify the user of unsuccessful password creation and shall also offer the user opportunity to retry.

4.1.2.39 LBIv1-REQ-275718/B-Password Response

Upon receiving the hashed password and key index through BackupIgnition_Rq (OpCode =” Password Transmit”), the LBIServer shall check that the phone and key fob are still in the vehicle and the password uniqueness before it can response back the password creation result.

To obtain phone status the LBIServer shall trigger a LBIServer2 Interior Registry KeyFob search to determine if the KeyFob is found inside the vehicle by executing LBIv1-FUN-REQ-302285.

Once determine all necessary devices are still present, the LBIServer shall check the uniqueness of password as defined in LBIv1-REQ-267620.

The LBIServer shall then report phone query result, key fob search result and password uniqueness status to the LBIClient via BackupIgnition_Rsp with the encoding values set as below:

RspCode =" Reset 2 Password Response"
RspStatus =

- "PaaK No Longer Detected" or
- "Fob No Longer Detected" or
- "PaaK and Fob No Longer Detected" or
- "Password Already Used" or
- "Password Created Successfully" or
- "Password Created Failed"

VariableData shall not be transmitted for either RspCode.

4.1.2.40 LBIv1-REQ-333138/A-De-bounce Timer When Localizing PaaK Device in Vehicle

The LBIServer shall de-bounce the phone localization state for the period of T_Phone Localization Timer per below conditions:

- When the phone is detected in vehicle interior zone then this state shall be reflected immediately to LBI function;
- When the phone is not detected in vehicle interior zone then de-bounce for the period of T_Phone Localization Timer before reporting the status to LBI function.

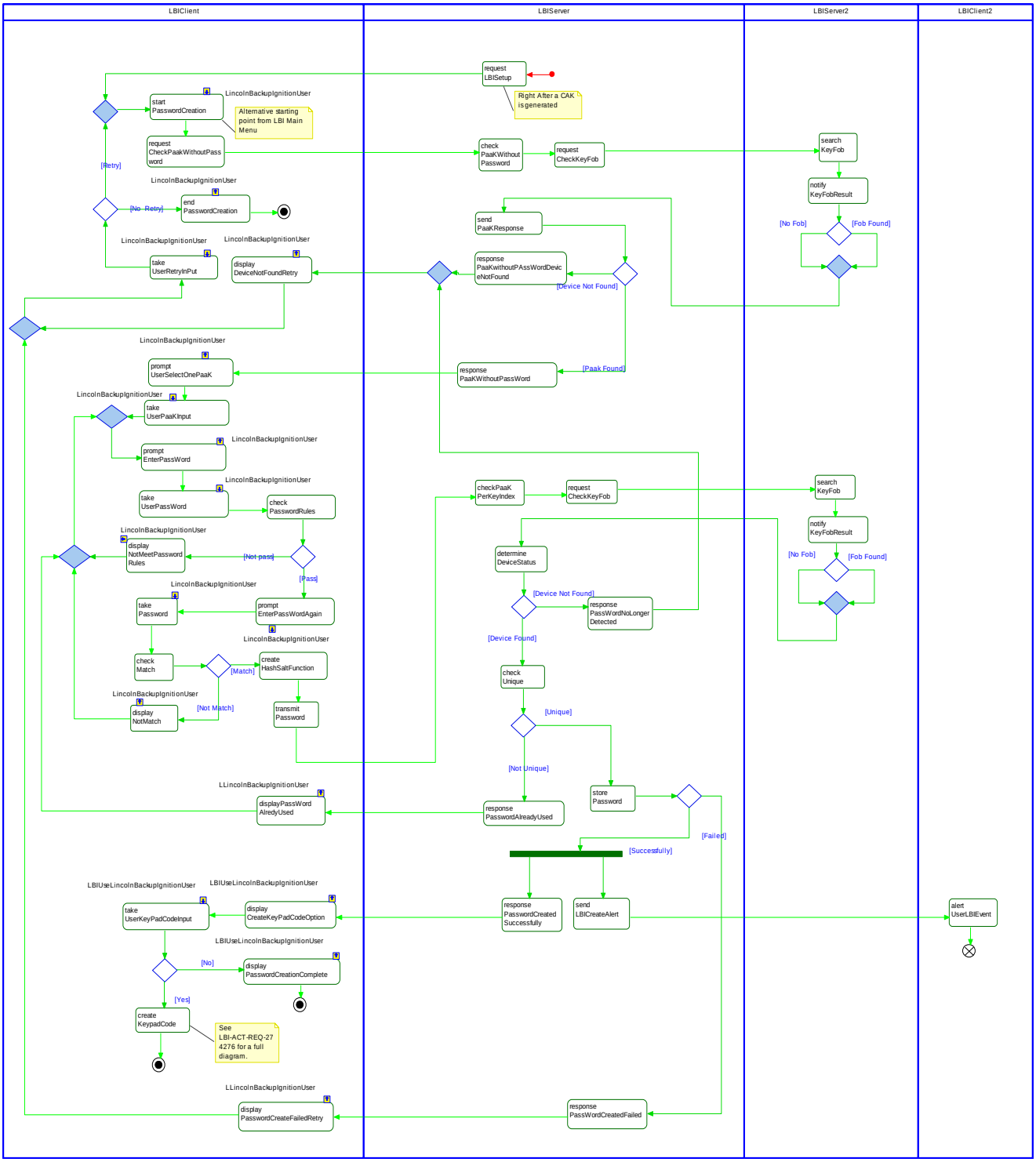
4.1.2.41 LBIv1-TMR-REQ-335312/A-T_Phone Localization Timer

Name	Description	Units	Range	Resolution	Default
T_Phone Localization Timer	Delay timer the LBISeriver must wait before reporting the status of phone localization if the phone is not detected within the vehicle interior zone for the duration of the timer. Note: Use the default value	msec	1000-5000	100	2500

White Box Views

4.1.2.42 Activity Diagrams

4.1.2.42.1 LBIv1-ACT-REQ-274275/A-Creating Backup Password for Paak Device Activity Diagram



4.1.2.43 Sequence Diagrams

4.1.2.43.1 LBIv1-SD-REQ-271726/A-Creating Backup Password - Happy Path

Constraints

Pre-Condition

1. The LBI User has previously activated a PaaK device for their vehicle.
2. The vehicle ignition is in Run
3. The vehicle transmission is in Park
4. The LBI User is inside vehicle.
5. The LBIServer has the vehicle-unique salt

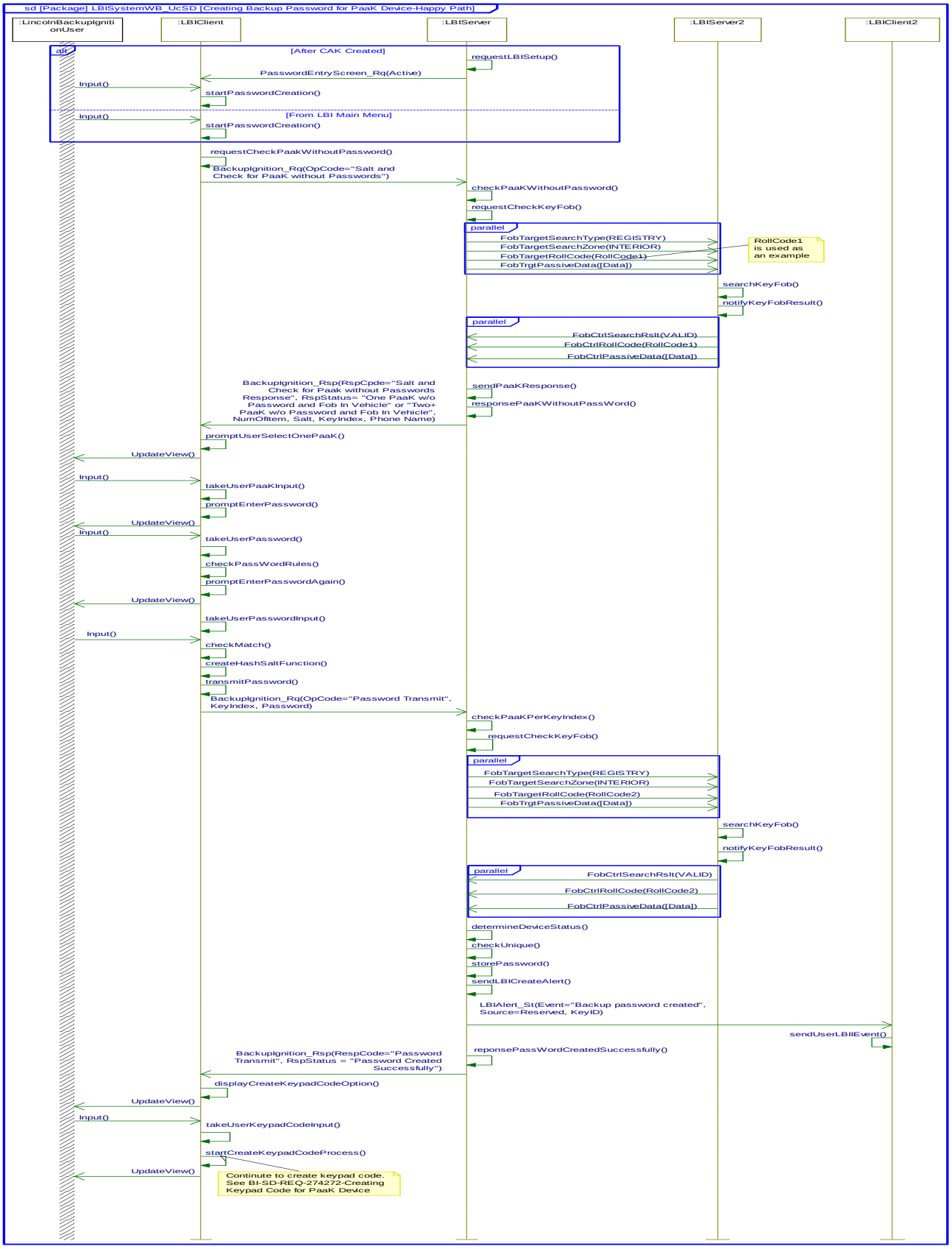
Scenarios

Normal Usage

Post-Condition

1. PaaK Backup is ready for use
2. A Notification of a backup password has been created is sent to the LBI User via e-mail and via a phone message to the PaaK device which a backup password is just created and associated with

Sequence Diagram



4.1.2.43.2 LBIv1-SD-REQ-275717/A-Creating Backup Password - Error Scenarios

Constraints

Pre-Condition

1. The LBI User has previously activated a PaaK device for their vehicle.
2. The vehicle ignition is in Run
3. The vehicle transmission is in Park
4. The LBI User is inside vehicle.
5. The LBIServer has the vehicle-unique salt

Scenarios

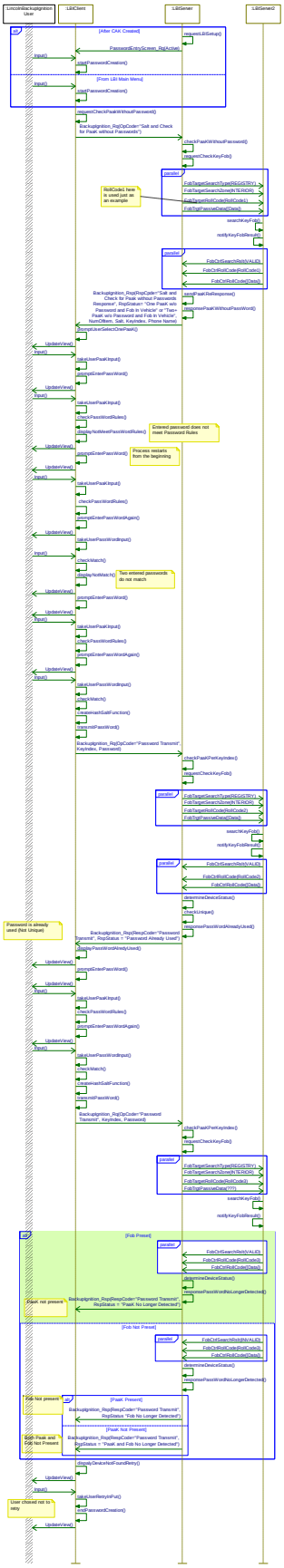
Normal Usage

1. The entered password does not meet Password Rules
2. Two entered passwords do not match
3. The entered password is already used (Not Unique)
4. Devices no longer present

Post-Condition

No backup password created

Sequence Diagram



4.2 LBIv1-FUN-REQ-269571/C-Creating Keypad Code for PaaK Device

4.2.1 Use Cases

4.2.1.1 LBIv1-UC-REQ-269572/A-Creating Keypad Code for PaaK Device

Actors	The LBI User.
Pre-conditions	1. The LBI HMI displays a message that a backup password has been created successfully 2. The LBI HMI asks t The LBI User if he/she would like to create a LBI Keypad code
Scenario Description	1. The LBI User selects option to create a LBI keypad code 2. displays screen for entering personal keypad code 3. The LBI User enters personal keypad code twice 4. The LBI HMI displays message that new LBI keypad code has been created successfully
Post-conditions	A LBI Keypad code is ready for use
List of Exception Use Cases	1. The LBI User enters a keypad code that does not meet requirements 2. The LBI User enters Keypad codes that do not match 3. The LBI User enters a keypad code that is already in use
Interfaces	The LBIClient The LBIServer The LBIServer2

4.2.1.2 LBIv1-UC-REQ-323248/B-Entering Too Little Digits for Keypad Code

Actors	The LBI User.
Pre-conditions	The LBI HMI displays the LBI Keypad code entry screen with length information
Scenario Description	The LBI User enters a keypad code with less digits than the required length
Post-conditions	The LBI HMI doesn't make the "Enter" button available until after the required number of digits have been entered
List of Exception Use Cases	
Interfaces	The LBIClient

4.2.1.3 LBIv1-UC-REQ-323249/B-Entering Too Many Digits for Keypad Code

Actors	The LBI User.
Pre-conditions	The LBI HMI displays LBI Keypad code entry screen with length information
Scenario Description	The LBI User enters a keypad code with more digits than the required length
Post-conditions	The LBI HMI ignores the extra digits and grays out the keypad buttons for entry
List of Exception Use Cases	
Interfaces	The LBIClient

4.2.1.4 LBIv1-UC-REQ-323257/B-Entering Unacceptable Keypad Code (For EU Markets only)

Actors	The LBI User.
Pre-conditions	The LBI HMI displays LBI Keypad code entry screen with length information
Scenario Description	The LBI User enters a keypad code by pressing a Keypad button a 6th consecutive time.
Post-conditions	The LBI HMI grays out the keypad button for that digit to prevent it from being entered a 7th time and displays a message to the user to inform them that the same button cannot be used 7 times.
List of Exception Use Cases	
Interfaces	The LBIClient
Notes	This use case only applies to EU markets.

4.2.2 Requirements

4.2.2.1 LBIv1-REQ-260071/A-HMI Display of Keypad Code Creation Option

After notifying the user of successful creating or resetting a backup password, the LBIClient HMI shall also present the user the option to set up a personal keypad code.

4.2.2.2 LBIv1-REQ-260073/B-HMI Display for LBI Keypad Code Entry Screen

If the user chooses to create a personal keypad code right after a backup password created, the LBIClient HMI shall display a screen for entering a new personal keypad code (LBI Keypad code).

This keypad code creation entry display screen is unique for Lincoln Backup Ignition as each keypad code is tied with the newly created backup password. Consequently, backup keypad code creation entry screen shall not be available without backup password and shall only be offered right after a backup password is created.

4.2.2.3 LBIv1-REQ-318317/A-HMI Functions for LBI Keypad Code Entry Screen

The LBIClient HMI shall provide a LBI Keypad code entry screen that includes but not limits to the following functions:

- The LBIClent HMI Keypad code entry screen shall provide the information of the exact number of digits for a keypad code required in a particular vehicle market per the vehicle configuration parameter as required by LBIv1-REQ-318311
- The LBIClient shall monitor the length of entered keypad code against the Keypad code configuration parameter and shall provide feedback to the LBI User when a mismatch is detected per LBIv1-REQ-318683
- The LBIClent HMI Keypad code entry screen shall make the composition restriction, as defined in LBIv1-REQ-322204, available upon the LBI User's request
- The LBIClient shall monitor the composition of entered keypad code (button press sequence) against the Keypad code composition restriction and shall reject the entered Keypad code per LBIv1-REQ-322204, provide feedback and give a retry option to the LBI User when any restriction is violated.

4.2.2.4 LBIv1-REQ-318683/A-LBIClient Error Handling Strategies for Mismatched LBI Keypad Code

The LBIClient HMI shall be able to detect a mismatched Keypad code entry in various types and provide feedback accordingly. The mismatched cases shall include but not limit to the following cases:

- When the LBI User does not enter enough digits (for example, entering 5 or less digits in a 7-digit code market), the LBIClient HMI shall not accept the entry command (Next button press). Instead, the LBIClient shall prompt the LBI User to re-enter correct number of digits until a match between user entered number of digits and vehicle configuration is obtained, and then transition to next step

- Once the entered digits matched the required length, the LBIClient shall ignore the extra entered digits and shall block out any more entries. In addition, the LBIClient shall inform the LBI User about the correct length of Keypad code had been entered and shall instruct the LBI User stop entering any more digits but press Next button to continue

- When the LBI User is allowed to enter too many or too little digits due to LBI feature error (not user error), the LBIClient shall not transition to the next step. Instead it shall set the U2101-00 DTC (*Control Module Configuration Incompatible* DTC) and display an error message to inform the LBI User about the abort of creating process and advise the LBI User seek assistance

  Note: Example of too-many-digit-mismatched case could be described as below:
  In a 5-digit market, the LBIClient mistakenly displays the 7-digit LBI Keypad code entry screen and accepts 7digits. However, internally the LBIClient still expects a 5-digit code per the configuration parameter and thus detects a mismatch. In this case, the LBIClient shall set the U2101-00D DTC, abort the creating process and inform the LBI User about the abort due to system error

4.2.2.5 LBIv1-REQ-264858/B-HMI Solicit Keypad Code Twice

Once a length-compliant LBI Keypad code with correct composition is entered, per LBIv1-REQ-318317 and LBIv1-REQ-322204 respectively, the LBIClient HMI shall require the LBI User to enter that rule-compliant Keypad code twice.

4.2.2.6 LBIv1-REQ-270260/A-Check Match for Two Entered Keypad Codes

The LBIClient HMI shall check if both entered keypad codes match.

In the case when the codes do not match, the LBIClient HMI shall notify the user and offer option to retry.

4.2.2.7 LBIv1-REQ-264859/B-Conditions of Triggering Keypad Code Creation Request

LBIClient shall trigger a keypad code creation request, as defined in REQ-267283, to LBIServer only when the same rule-compiled keypad code is entered twice.

4.2.2.8 LBIv1-REQ-267283/B-Request Keypad Code Creation

To initiate the creation of a LBI Keypad code, the LBIClient shall send the entered Keypad code to the LBIServer via BackupIngnition_Rq (OpCode= “Keypad Code Create Request”, *KeyIndex = KeyIndex, Password = EOS, KeypadCode = KeypadCode*) with *KeyIndex and KeypadCode defined as below:*

- The KeyIndex is the index of the selected Paak device

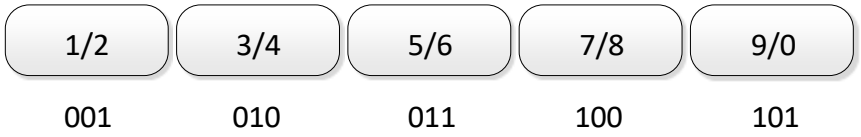
- The KeyPadCode is the user-entered Keypad code mapped into a data string with the structure defined by LBIv1-REQ-271253

4.2.2.9 LBIv1-REQ-271253/B-LBIClient LBI Keypad Code Transmitting Format

When the LBIClient transmits a keypad code that is associated with a backup password to the LBIServer, it shall structure the data as a 7-button sequence where each button is represented by three bits.

- Per the Transport Protocol APIM SPSS, this request shall be sent via BackupIgnition_Rq with Byte 4 (Opcode) = “Keypad Code Create Request” (0x08), and Bytes 6-9 (KeypadCode) = the bit encoded button sequence.

- The mapping of button to bit value listed below shall be used to translate button pressed to data string:



  - 000 = NULL
  - 001 = “1/2” button pressed
  - 010 = “3/4” button pressed
  - 011 = “5/6” button pressed

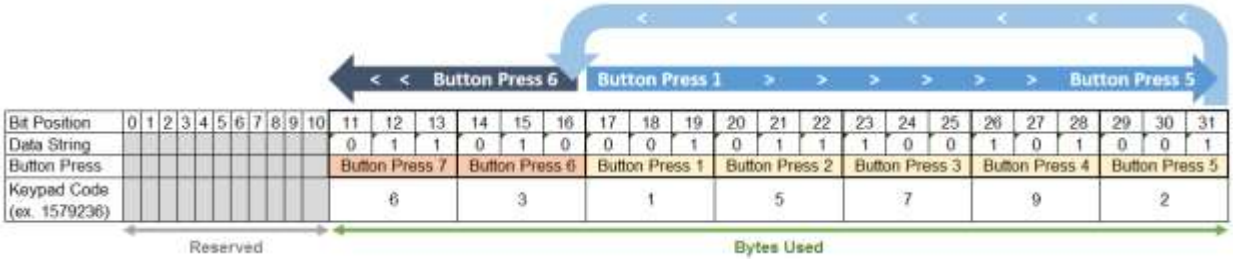
- 100 = “7/8” button pressed
- 101 = “9/0” button pressed

- When the vehicle configuration calls out for 7-digit Keypad codes, the Sixth and Seventh button press parameters shall be populated as requested as shown in Example1

- When the vehicle configuration calls out for 5-digit Keypad codes, the Sixth and Seventh button press parameters shall be set to NULL (000) as shown in Example 2

Example 1: 7-digit Keypad Code

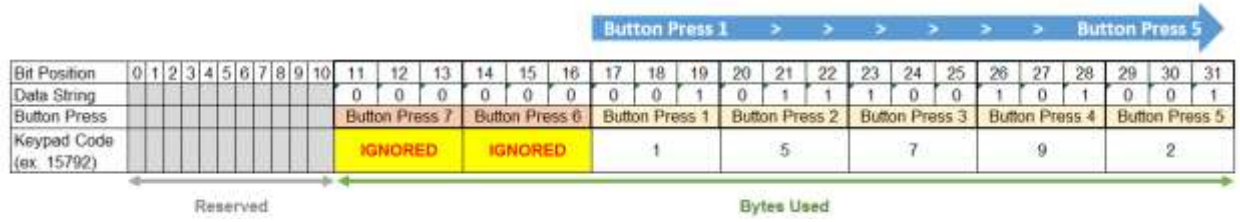
Below is an example of data string for a keypad code of **1579236** transmitted from the LBIClient to the LBIServer:



Example 2: 5-digit Keypad Code

Below is an example of data string for a keypad code of **15792** transmitted from the LBIClient to the LBIServer.

Note:
The Sixth and Seventh button press are ignored per LBIv1-REQ-318317
The bits in the data string for the Sixth and Seventh 6th and 7th button press are all set to Null as required.



4.2.2.10 LBIv1-REQ-264860/A-Initiate Keypad Code Storing for Create Keypad Code

When receiving keypad code create command via BackupIngnition_Rq(OpCode= “Keypad Code Create Request”, KeyIndex, Keypad Code), the LBIServer shall initiate the keypad code storing request by executing LBI-FUN-REQ-275727.

4.2.2.11 LBIv1-REQ-266474/A-Reponse for Keypad Code Creation

Once receiving storage confirmation from the LBIServer2, the LBIServer shall update keypad code association status for selected/detected key index, then shall send Keypad Code Create Response to the LBIClient via BackupIgnition_Rsp with encoding values set as below:

RspCode= “Keypad Code Create Response”
RspStatus=

- “Keypad Code Created Successfully” if KeyPadCodeProg_St (Add) or

- “Keypad Code Created Failed” if KeyPadCodeProg_St(ProgrammingFailure)

VariableData shall set to Zero

4.2.2.12 LBIv1-REQ-271445/B-Response of Duplicate LBI Keypad Code

If the LBIServer receives *KeyPadCodeProg_St(Duplicate)* in the case of storing a keypad code associated with a backup password as indicated in *BackupIgnition_Rq (OpCode = “Keypad Code Create Response)*, the LBIServer shall notify the

LBIClient that keypad code is already in use via *BackupIgnition_Rsp (RspCode = "Keypad Code Create Response", RspStatus =" Keypad Code Duplicate"*)

4.2.2.13 LBIv1-REQ-264864/B-HMI Display of Keypad Code Creation

Once receiving BackupIgnition_Rsp (Rsp Code= "Keypad Code Create Response", RspStatus= "Keypad Code Created Successfully), the LBIClient HMI shall notify the user of successful keypad code creation and shall also inform user that PaaK Backup setup is complete and provide instructions on how to use the feature

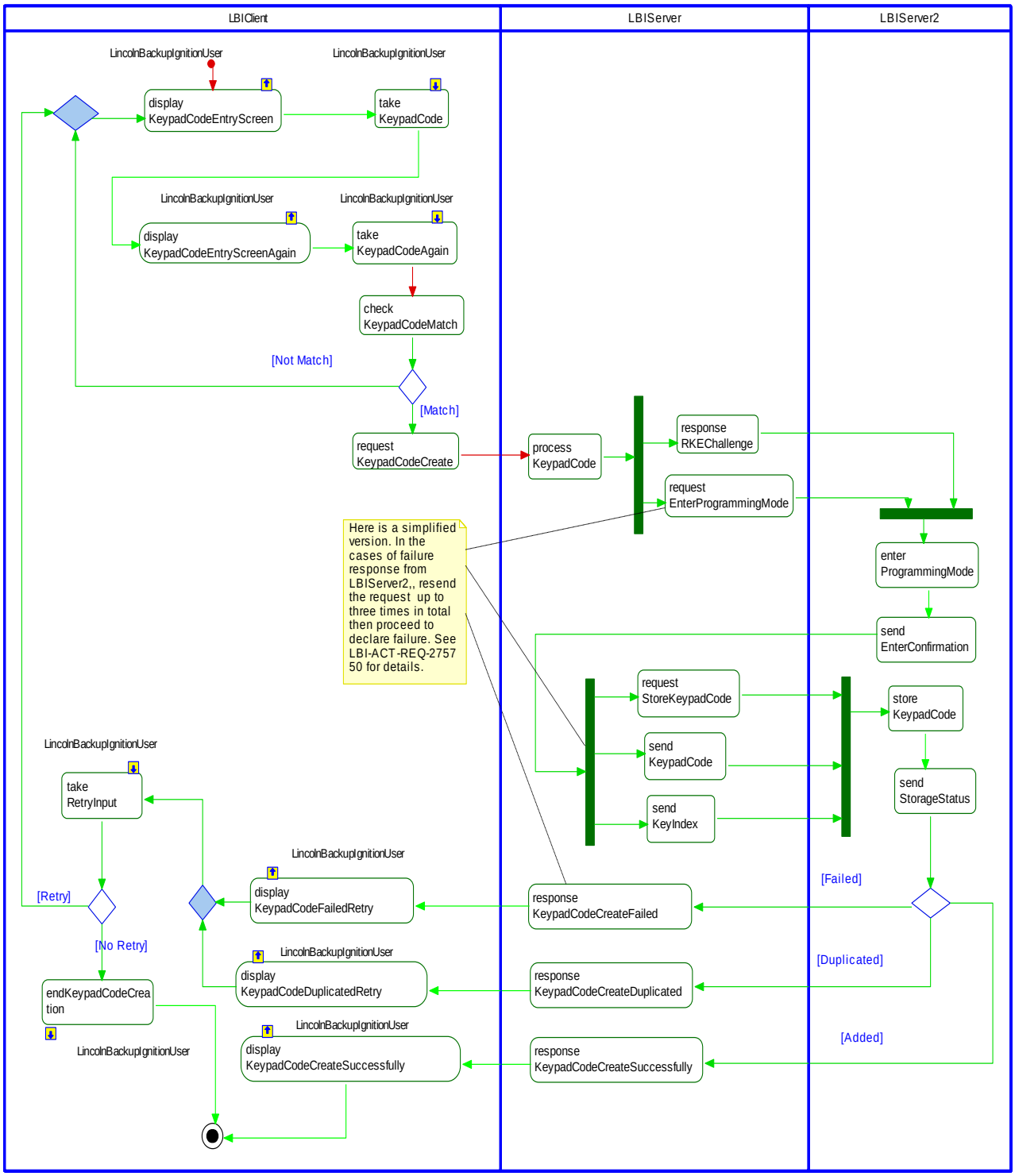
In the case of failure, indicated in RspStatus (Keypad Code Created Failed), the LBIClient HMI shall notify the user about the failure and provide opportunities for retry.

In the case of duplicate, indicated in RspStatus (*Keypad Code Duplicate*), the LBIClient HMI shall notify the user about the keypad code is already in use and provide opportunities for retry

White Box Views

4.2.2.14 Activity Diagrams

4.2.2.14.1 LBIv1-ACT-REQ-274276/A-Creating Keypad Code for PaaK Device Activity Diagram



4.2.2.15 Sequence Diagrams

4.2.2.15.1 LBIv1-SD-REQ-274272/B-Creating Keypad Code for PaaK Device

Constraints

Pre-Condition

1. The LBI HMI displays a message that a backup password has been created successfully
2. The LBI HMI asks The LBI User if he/she would like to create a LBI Keypad
3. PaakCtrlActionCode is set to null per LBIv1-REQ-304561

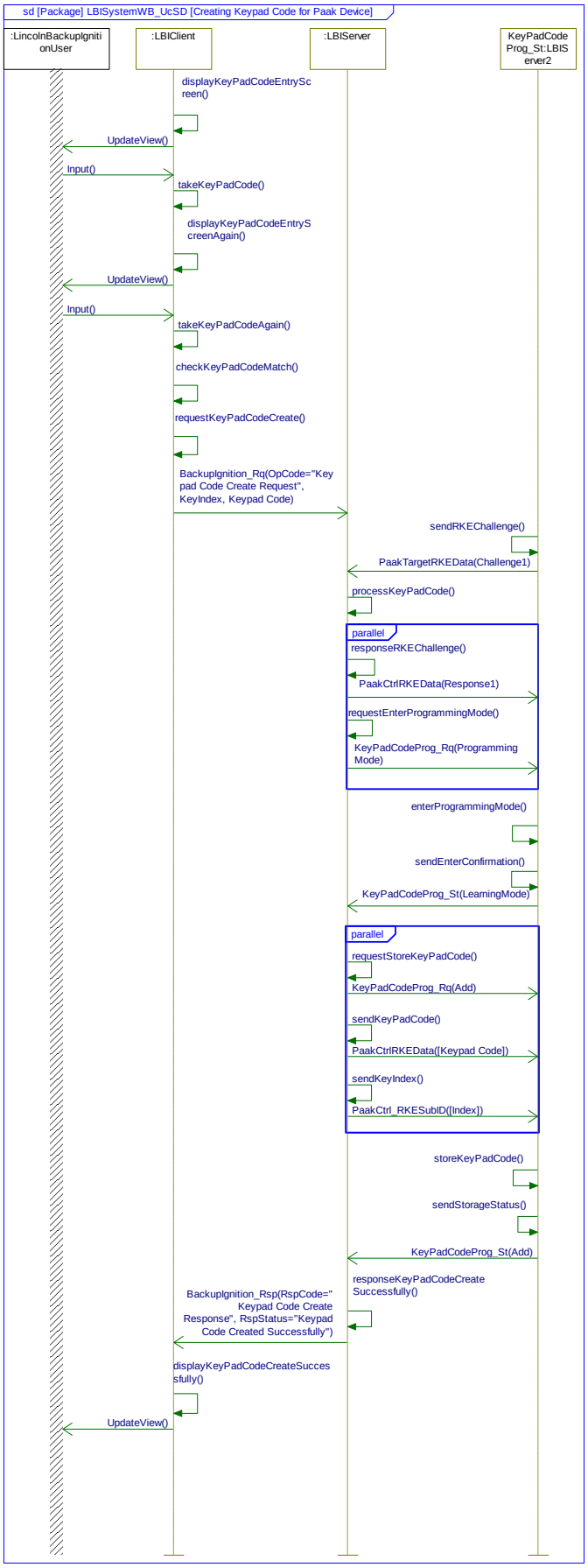
Scenarios

Normal Usage

Post-Condition

A LBI Keypad code is ready for use

Sequence Diagram


sd [Package] LBISystemWB_UcSD [Creating Keypad Code for Paak Device]
:LincolnBackupIgnitionUser
:LBIClient
:LBIServer
KeyPadCodeProg_St:LBIServer2
displayKeyPadCodeEntryScreen()
UpdateView()
Input()
takeKeyPadCode()
displayKeyPadCodeEntryScreenAgain()
UpdateView()
Input()
takeKeyPadCodeAgain()
checkKeyPadCodeMatch()
requestKeyPadCodeCreate()
BackupIgnition_Rq(OpCode="Keypad Code Create Request", KeyIndex, Keypad Code)
sendRKEChallenge()
PaakTargetRKEData(Challenge1)
processKeyPadCode()
parallel
responseRKEChallenge()
PaakCtrlRKEData(Response1)
requestEnterProgrammingMode()
KeyPadCodeProg_Rq(ProgrammingMode)
enterProgrammingMode()
sendEnterConfirmation()
KeyPadCodeProg_St(LearningMode)
parallel
requestStoreKeyPadCode()
KeyPadCodeProg_Rq(Add)
sendKeyPadCode()
PaakCtrlRKEData([Keypad Code])
sendKeyIndex()
PaakCtrl_RKESubID([Index])
storeKeyPadCode()
sendStorageStatus()
KeyPadCodeProg_St(Add)
responseKeyPadCodeCreateSuccessfully()
BackupIgnition_Rsp(RspCode="Keypad Code Create Response", RspStatus="Keypad Code Created Successfully")
displayKeyPadCodeCreateSuccessfully()
UpdateView()

4.3 LBIv1-FUN-REQ-258450/A-Deleting Backup Password and Keypad Code for Paak Device

4.3.1 Use Cases

4.3.1.1 LBIv1-UC-REQ-260103/A-Delete Backup Password

Actors	The LBI User
Pre-conditions	1. The LBI User is inside the vehicle. 2. One associated PaaK with password is inside the vehicle 3. The vehicle ignition is in Run 4. The vehicle transmission is in Park 5. The vehicle is not locked out by LBI Feature 6. The vehicle is not in Enhanced Valet Mode
Scenario Description	1. The LBI User selects option to Delete Backup Password for PaaK from the LBI Main Menu in LBI HMI 2. The LBI HMI displays a message with deletion requirements 3. The LBI User continues 4. The LBI HMI displays message asking the LBI User for deletion confirmation 5. The LBI User confirms 6. The LBI HMI displays message that backup password and personal keypad code have been deleted successfully or backup password is deleted successfully but personal keypad code deletion is unsuccessful
Post-conditions	1. The backup password is deleted 2. A notification of a backup password has been deleted is sent to the LBI User via e-mail and a phone passage to that PaaK whose password is just deleted
List of Exception Use Cases	The vehicle ignition changes to not in Run or the transmission status is changed to out of Park thus the deletion process is aborted
Interfaces	The LBIClient The LBIClient2 The LBIServer The LBIServer2 SDN PaakFI

4.3.2 Requirements

4.3.2.1 LBIv1-REQ-260090/A-Initiate Delete Backup Password

When the user selects the option to delete a backup password within LBIClient settings, LBIClient shall query the LBIServer, for PaaK devices with passwords in the vehicle and shall request the cryptographic salt from the LBIServer as defined in REQ-267238.

4.3.2.2 LBIv1-REQ-267238/A-Query for PaaKs with Passwords

To query if keyfobs and PaaK devices with passwords are in the vehicle, LBIClent shall triger a search via BackupIgnition_Rq(OpCode="Salt and Check for PaaK with Passwords", KeyIndex=0x0, *,Password = EOS, KeypadCode = EOS*).

The usages of this search include but not limit to delete LBIPassword and change LBIPassword.

4.3.2.3 LBIv1-REQ-271264/B-Trigger Interior Registry KeyFob Search of Paak with Password to Delete Reset

In the process of deleting or resetting password, upon receiving the query of PaaK devices with passwords in the vehicle and the request the cryptographic salt via BackupIgnition_Rq (OpCode=" Salt and Check for PaaK with Passwords",

KeyIndex=0x00) from the LBIClient, the LBIServer shall trigger a LBIServer2 Interior Registry KeyFob search by executing LBIv1-FUN-REQ-302285.

4.3.2.4 LBIv1-REQ-264545/A-Response of PaaK with Backup Passwords

In the deleting or resetting password process, upon receiving the request BackupIgnition_Rq(OpCode="Salt and Check for PaaK with Passwords"), the LBIServer shall response to LBIClient the device name and key index for each PaaK device that has a backup password as defined in REQ-267239-Status for PaaKs with Passwords.

4.3.2.5 LBIv1-REQ-264870/A-HMI Display for No Paak with Password

Upon receiving LBIServer report of no PaaK devices with passwords in the vehicle through BackupIgnition_Rsp, the LBIClient HMI shall then display an error message that no phones were detected and shall provide an option to retry.

The No PaaK Device is defined in BackupIgnition_Rsp (RspCode= "Salt and Check for Paak with Passwords Response", RspStatus= "Fob in Vehicle and No PaaK w/ Password" or "No PaaK w/ Password and No Fob In Vehicle").

4.3.2.6 LBIv1-REQ-270058/A-HMI Display of Paak Found

If there were multiple PaaK devices with passwords in the vehicle during LBIClient-requested key search, the LBIClient HMI shall display a list of all devices with instructions for the user to choose the desired device..

The detected PaaK Devices is defined in BackupIgnition_Rsp with encoding values set as:
RspCode= "Salt and Check for Paak with Passwords Response"
RspStatus=

- "*One PaaK w/ Password and Fob In Vehicle"* or
- "*One PaaK w/ Password and No Fob In Vehicle"* or
- "*Two+ PaaK w/ Password and Fob In Vehicle"* or
- *"Two+ PaaK w/ Password and No Fob In Vehicle"*

NumberOfItems = 0x01 - 0x04
Challenge Nonce = EOS
Salt = Salt
Valet Password = EOS
KeyIndex = KeyIndex
PhoneName = PhoneName

4.3.2.7 LBIv1-REQ-264873/A-HMI Display of Delete Intent Comfirmation

When the user chooses their desired device (password) for deletion, the LBIClient shall ask the user to confirm deletion of the associated password.

4.3.2.8 LBIv1-REQ-264874/A-Request Password Deletion

When the user confirms deletion of their password, the LBIClient shall send to the LBIServer the key index of the selected PaaK along with a request to delete password hash associated with this key index via BackupIgnition_Rq(OpCode="Password Delete Request", KeyIndex=Keyindex, *,Password = EOS, KeypadCode = EOS*).

4.3.2.9 LBIv1-REQ-264875/A-Delete Backup Password

Upon receiving BackupIgnition_Rq(Opcode="Password Delete Request", *KeyIndex = KeyIndex*), the LBIServer shall immediately delete the password hash associated with the received key index.

4.3.2.10 LBIv1-REQ-264877/A-Initiate Keypad Code Deletion for Delete Password

In the process of resetting password, if the entered new password is not already being used, then the LBIServer needs to delete the old backup password along with the keypad code if it has one before store the new password.

The LBIServer shall check if a keypad code was created for the password with the received KeyIndex. If a keypad code was created then the LBIServer shall initiate Keypad Code deletion by executing LBI-FUN-REQ-275727.

The received key index is defined in BackupIngnition_Rq(OpCode= "Reset 1 Password Transmit" or" Reset 2 Password Transmit",KeyIndex)

4.3.2.11 LBIv1-REQ-269546/A-Criteria of Reporting Successful Password Deletion

The LBIServer shall report a successful password deletion if the conditions below are both met:

- A successful keypad code deletion confirmation via KeyPadCodeProg_St (Delete) is received
- The requested backup password is deleted from the LBIServer HSM

4.3.2.12 LBIv1-REQ-275351/A-Criteria of Reporting Successful Password Deletion with Keypad Code not Deleted

The LBIServer shall report a successful password deletion if the conditions below are all met:

- A failed Notification is received via KeyPadCodeProg_St(ProgrammingFailure)
- Resend KeyPadCodeProg_Rq(Programming_Mode) or KeyPadCodeProg_Rq(Delete) up to the fined number of times as defined by N_LBINumberOfRetries
- The requested backup password is deleted from the LBIServer HSM

4.3.2.13 LBIv1-REQ-269548/A-Criteria of Reporting Failed Password Deletion

The LBIServer shall report a failed password deletion if any one of conditions below is met:

- The vehicle operation conditions defined by REQ- 264925 are no longer met before the deletion process completes
- The requested backup password is unable to be deleted from the LBIServer HSM

4.3.2.14 LBIv1-REQ-267087/A-Response of Backup Password Deletion

After deleting backup password and receiving keypad deletion result, the LBIServer shall notify LBIClient about the Password and keypad code deletion status via BackupIgnition_Rsp with encoding values set as

RspCode=" Password Delete Response"
RspStatus =

- "Password Deleted Successfully" per REQ-269546 or
- "Password Deleted Successfully but Keypad Code Deleted Failed" per REQ-275351 or
- "Password Deleted Failed "per REQ-269548

NumberOfItems = 0x00,
Challenge Nonce = EOS
Salt = EOS
Valet Password = EOS
KeyIndex = 0x0
PhoneName = EOS

4.3.2.15 LBIv1-REQ-264879/A-HMI Dispaly of Successful Backup Password Deletion

The LBIClient HMI shall notify the user about the successful password and keypad code deletion once it receives deletion confirmation via BackupIgnition_Rsp(RspCode="Password Delete Response",RspStatus="Password Deleted Successfully")

In the case of a failed keypad code deletion as indicated in RspStatus being "Password Deleted Successfully but Keypad Code Deleted Failed", the LBIClient HMI shall notify the use about the successful password deletion and the failed keypad code deletion.

4.3.2.16 LBIv1-REQ-271266/A-HMI Display of Failed Backup Password Deletion

The LBIClient HMI shall notify the user about the failed password deletion and shall also provide the user option to retry once it receives deletion confirmation via BackupIgnition_Rsp(RspCode="Password Delete Response",RspStatus="Password Delete Failed).

4.3.2.17 LBIv1-REQ-271265/B-Notification of Successful Password Deletion to LBIClient2

Once a password hash is deleted as defined in LBIv1-REQ-269546, the LBIServer shall report this event, including the corresponding KeyID, to the LBIClient2 via *LBIAlert_St(Event ="Backup Password Deleted", Source = Reserved)*

The corresponding KeyID shall be the KeyID of the PaaK device associated with the deleted backup password.

4.3.2.18 LBIv1-REQ-267239/B-Status for PaaKs with Passwords

The LBIServer shall report both phone query results with name and key index and the cryptographic salt of Paak devices as well as key fob search results (by executing LBIv1-FUN-REQ-302285) to the LBIClient via BackupIgnition_Rsp with the encoding values set as below:

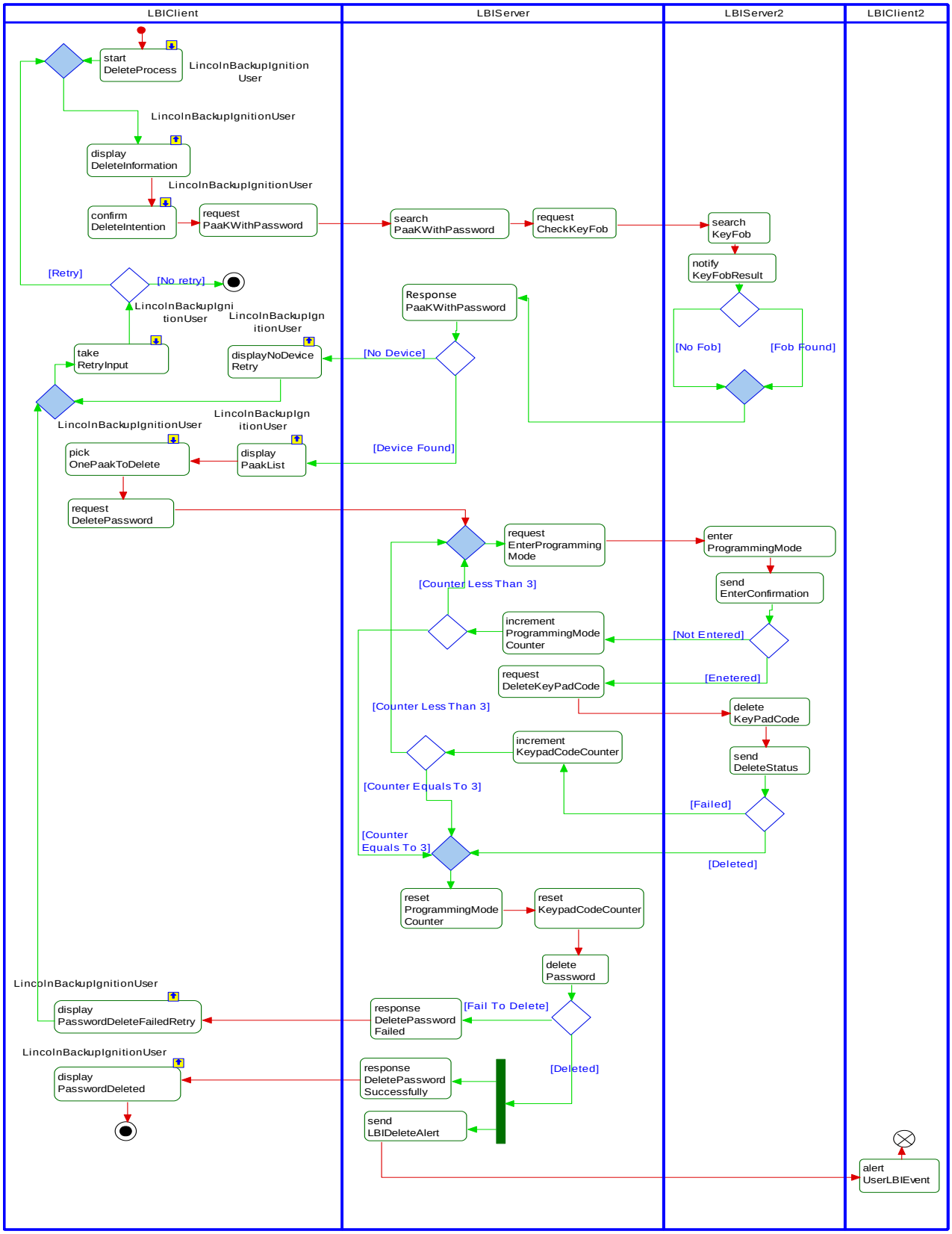
RspCode =" Salt and Check for PaaK with Passwords",
RspStatus =

- "One PaaK w/ Password and Fob In Vehicle" or
- "One PaaK w/ Password and No Fob In Vehicle" or
- "Fob in Vehicle and No PaaK w/ Password" or
- "Two+ PaaK w/ Password and Fob In Vehicle" or
- "Two+ PaaK w/ Password and No Fob In Vehicle" or
- "No PaaK w/ Password and No Fob In Vehicle"

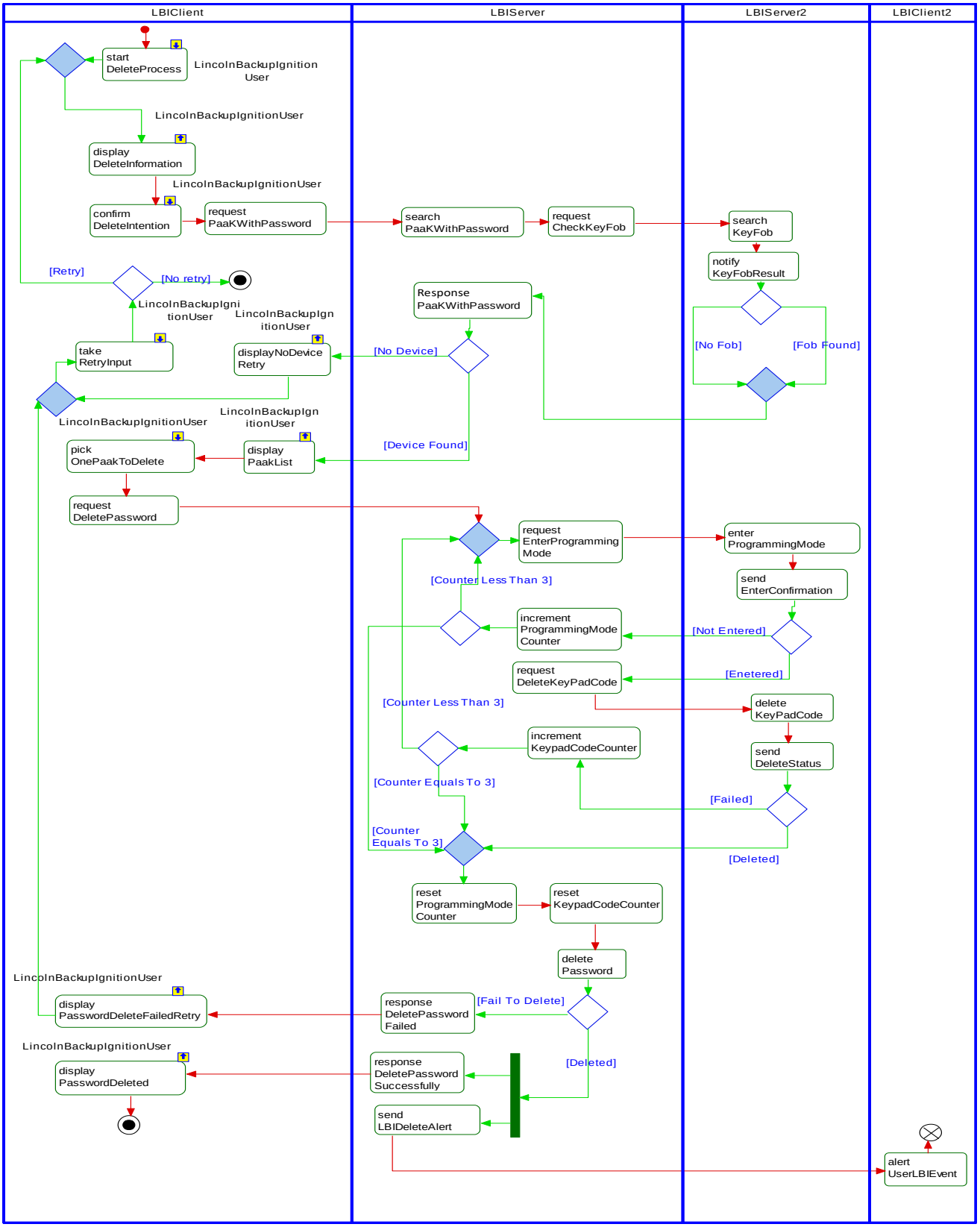
Challenge Nonce = EOS,
Salt = Salt,
Valet Password = EOS,
KeyIndex = KeyIndex,
PhoneName = PhoneName

White Box Views

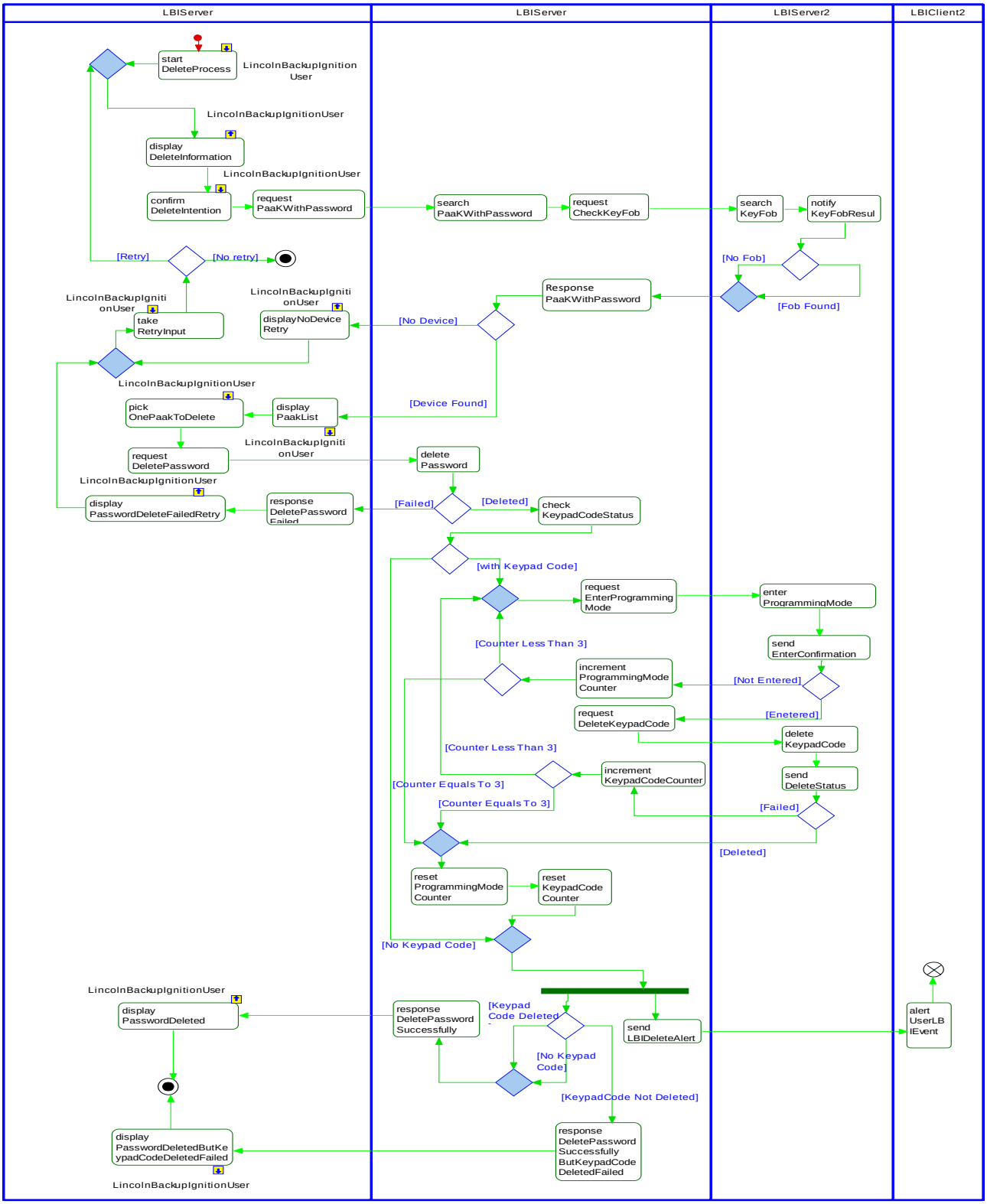
4.3.2.19 Activity Diagrams



4.3.2.19.1 LBIv1-ACT-REQ-274279/A-Deleting Backup Password and Keypad Code for Paak Device



Activity Diagram



4.3.2.20 Sequence Diagrams

4.3.2.20.1 LBIv1-SD-REQ-262299/A-Deleting Backup Password and Keypad Code for Paak Device

Constraints

Pre-Condition

1. The LBI User is inside the vehicle.
2. One associated PaaK with password is inside the vehicle
3. The vehicle ignition is in Run
4. The vehicle transmission is in Park
5. The vehicle is not locked out by LBI Feature
6. The vehicle is not in Enhanced Valet Mode

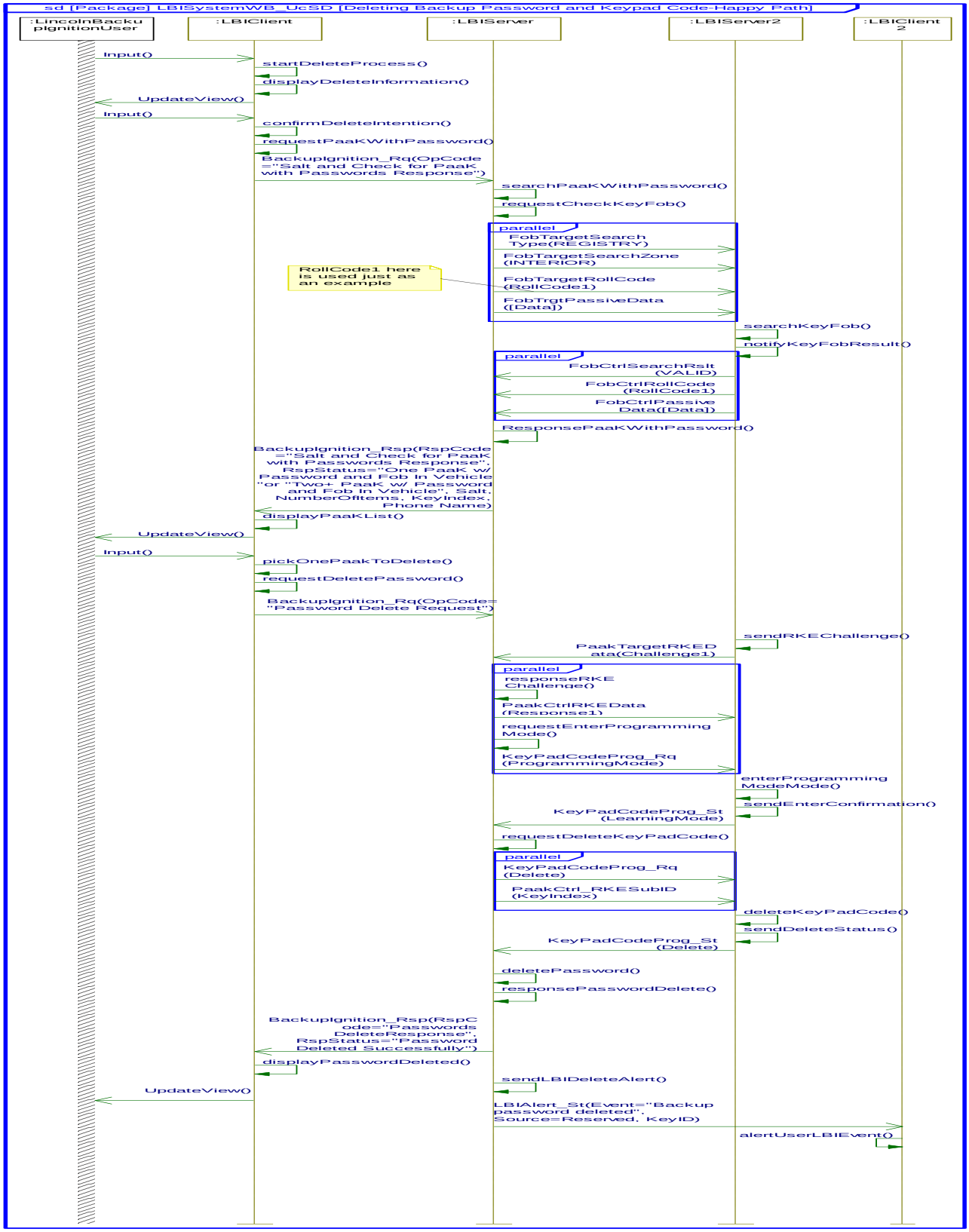
Scenarios

Normal Usage

Post-Condition

1. The backup password is deleted
2. A notification of a backup password has been deleted is sent to the LBI User via e-mail and a phone passage to that PaaK whose password is just deleted

Sequence Diagram



4.4 LBIv1-FUN-REQ-265670/A-Resetting Backup Password and Keypad Code for PaaK Device

4.4.1 Use Cases

4.4.1.1 LBIv1-UC-REQ-265672/A-Resetting backup Password and Keypad Code for Paak Device

Actors	The LBI User
Pre-conditions	1. The LBI User has previously activated PaaK feature for the vehicle 2. The vehicle ignition Status is in Run 3. The vehicle transmission is in Park 4. The LBI User is inside vehicle 5. One associated PaaK and one key fob are inside the vehicle 6. The vehicle is not locked out by LBI feature 7. The vehicle is not in Enhanced Valet Mode
Scenario Description	1. The LBI User selects option to Change Backup Password for Phone-as-a-Key from PaaK Backup Settings in the LBI HM. 2. The LBI HMI displays a message with reset requirements 3. The LBI User continues 4. The LBI HMI displays alphanumeric password entry screen and instructs the LBI User to enter a new backup password 5. The LBI User enters password twice according to password requirements 6. The LBI User selects Enter 7. The LBI HMI displays a message that backup password has been reset successfully 8. The LBI HMI asks the LBI User if he/she would like to create a LBI Keypad code. 9. The LBI User declines options to create a LBI keypad code.
Post-conditions	1. The backup password of a selected Paak is reset 2. A Notification of a backup password has been reset is sent to the LBI User via e-mail and via a phone message to the PaaK device which a backup password is just reset and associated with
List of Exception Use Cases	
Interfaces	The LBIClient The LBIClient2 The LBIServer The LBIServer2 SDN PaakFI I

4.4.2 Requirements

4.4.2.1 LBIv1-REQ-264880/A-Initiate Reset Backup Password

When the user selects the option to reset a backup password In LBI Menu, LBIClient shall query the LBIServer for PaaK devices with passwords in the vehicle and shall request the cryptographic salt from the LBIServer as defined in REQ-267238.

4.4.2.2 LBIv1-REQ-267238/A-Query for PaaKs with Passwords

To query if keyfobs and PaaK devices with passwords are in the vehicle, LBIClent shall triger a search via BackupIgnition_Rq(OpCode="Salt and Check for PaaK with Passwords", KeyIndex=0x0, *,Password = EOS, KeypadCode = EOS*).

The usages of this search include but not limit to delete LBIPassword and change LBIPassword.

4.4.2.3 LBIv1-REQ-264545/A-Response of PaaK with Backup Passwords

In the deleting or resetting password process, upon receiving the request BackupIgnition_Rq(OpCode="Salt and Check for PaaK with Passwords"), the LBIServer shall response to LBIClient the device name and key index for each PaaK device that has a backup password as defined in REQ-267239-Status for PaaKs with Passwords.

4.4.2.4 LBIv1-REQ-267239/B-Status for PaaKs with Passwords

The LBIServer shall report both phone query results with name and key index and the cryptographic salt of Paak devices as well as key fob search results (by executing LBIv1-FUN-REQ-302285) to the LBIClient via BackupIgnition_Rsp with the encoding values set as below:

RspCode =" Salt and Check for PaaK with Passwords",
RspStatus =

- "One PaaK w/ Password and Fob In Vehicle" or
- "One PaaK w/ Password and No Fob In Vehicle" or
- "Fob in Vehicle and No PaaK w/ Password" or
- "Two+ PaaK w/ Password and Fob In Vehicle" or
- "Two+ PaaK w/ Password and No Fob In Vehicle" or
- "No PaaK w/ Password and No Fob In Vehicle"

Challenge Nonce = EOS,
Salt = Salt,
Valet Password = EOS,
KeyIndex = KeyIndex,
PhoneName = PhoneName

4.4.2.5 LBIv1-REQ-264870/A-HMI Display for No Paak with Password

Upon receiving LBIServer report of no PaaK devices with passwords in the vehicle through BackupIgnition_Rsp, the LBIClient HMI shall then display an error message that no phones were detected and shall provide an option to retry.

The No PaaK Device is defined in BackupIgnition_Rsp (RspCode= "Salt and Check for Paak with Passwords Response", RspStatus= "Fob in Vehicle and No PaaK w/ Password" or "No PaaK w/ Password and No Fob In Vehicle").

4.4.2.6 LBIv1-REQ-270058/A-HMI Display of Paak Found

If there were multiple PaaK devices with passwords in the vehicle during LBIClient-requested key search, the LBIClient HMI shall display a list of all devices with instructions for the user to choose the desired device..

The detected PaaK Devices is defined in BackupIgnition_Rsp with encoding values set as:
RspCode= "Salt and Check for Paak with Passwords Response"
RspStatus=

- "*One PaaK w/ Password and Fob In Vehicle"* or
- "*One PaaK w/ Password and No Fob In Vehicle"* or
- "*Two+ PaaK w/ Password and Fob In Vehicle"* or
- *"Two+ PaaK w/ Password and No Fob In Vehicle"*

NumberOfItems = 0x01 - 0x04
Challenge Nonce = EOS
Salt = Salt
Valet Password = EOS
KeyIndex = KeyIndex
PhoneName = PhoneName

4.4.2.7 LBIv1-REQ-270148/A-HMI Display of Reset with No Fob

In the reset process once the user selects the desired device, the LBIClient shall display the backup password entry screen when receiving no fob report as indicated via BackupIgnition_Rsp with

RspCode= "Salt and Check for Paak with Passwords Response"

RspStatus=

- "*One PaaK w/ Password and No Fob In Vehicle*"

or

- "*Two+ PaaK w/ Password and No Fob In Vehicle"*

4.4.2.8 LBIv1-REQ-270149/A-HMI Display of Reset with Paak and Fob

In the reset process once the user selects the desired device, the LBIClient shall initiate the password creation process by first displaying the backup password creation screen when receiving both PaaK and Fob present report as indicated via BackupIgnition_Rsp with

(RspCode= "Salt and Check for Paak with Passwords Response"

RspStatus=

- "*One PaaK w/ Password and Fob In Vehicle*"

or

- "*Two+ PaaK w/ Password and Fob In Vehicle"*

4.4.2.9 LBIv1-REQ-270384/B-Challenge Request

When the user enters a password at the password entry screen, the LBIClient shall request a challenge from LBIServer via BackupIgnition_Rsp(OpCode="Challenge Request", KeyIndex=0, *Password = EOS KeypadCode = EOS*).

The use cases of this Requirement are:

- Starting a vehicle with a backup password or an Enhanced Valet password
- Resetting a backup password
- Generating an Enhanced Valet Password with no device
- Deleting an Enhanced Valet password with no device

4.4.2.10 LBIv1-REQ-260077/A-Issue Cryptographic Nonce and Salt Challenge

After receiving a request for a challenge from LBIClient via BackupIgnition_Rq (OpCode="Challenge Request"), the LBIServer shall issue a challenge to the LBIClient via BackupIgnition_Rsp with cryptographic nonce and salt.

BackupIgnition_Rsp shall be set as
RspCode="Issue challenge"
RspStatus= *Reserved*
NumberOfItems= 0x00
Challenge Nonce = Challenge Nonce
Salt = Salt

4.4.2.11 LBIv1-REQ-270244/A-Compute Hash for All Stored Passwords

After receiving a request for a challenge from the LBIClient via BackupIgnition_Rq (OpCode="Challenge Request"), in addition to issue challenge to the LBIClient as described in REQ-260077, the LBIServer shall also compute, using the cryptographic nonce, another hash of all stored password hashes.

4.4.2.12 LBIv1-REQ-260080/A-Compute Hash for Entered Password

Once receiving BackupIgnition_Rsp(RspCode="Issue challenge"), the LBIClient shall compute a hash of entered password using received salt and then compute a hash of this result using received nonce.

4.4.2.13 LBIv1-REQ-270250/A-Reset Respond to Challenge with Hashed Backup Password

In the Reset process, LBIClient shall respond to the LBIServer challenge with hashed password as required by REQ-260080 via BackupIgnition_Rq(OpCode="Reset Challenge Response", KeyIndex = 0x00, Password = Challenge Password).

4.4.2.14 LBIv1-REQ-271293/A-Compare Hashed and Computed Password for Reset

When LBIServer receives a challenge hash and key index via BackupIgnition_Rq (OpCode="*Reset Challenge Response*", *KeyIndex = 0x00, Password = Challenge Password*), it shall compare the challenge hash with the password hash associated with the received key index.

4.4.2.15 LBIv1-REQ-271294/A-Response of Invalid Password Entry for Reset

In the response for reset challenge (OpCode="Reset Challenge Response"), if LBIServer determines password is invalid per REQ-268538, it shall notify LBIClient of this with BackupIgnition_Rsp(RespCode="Reset Challenge Response *Acknowledge*", RspStatus="Invalid Password").

4.4.2.16 LBIv1-REQ-264866/B-Track Invalid Password Entering Attempts

The LBIServer shall keep track of invalid attempts at entering the backup password and invalid attempts at entering the valet password in separate counters, N_NumberOfInvalidBackupAttempts and N_NumberOfInvalidValetAttempts.

Each time when the LBIServer determines a password is invalid, it shall increment invalid password counter N_NumberOfInvalidBackupAttempts for the followings use cases:

- Resetting a backup password (Option 1) with OpCode=" Reset Challenge Response" in LBIv1-REQ- 271294
- Activating Enhanced Valet Password (Option 2) with OpCode=" Challenge Response"
- Deactivating Enhanced Valet Password (Option 2) with OpCode=" Valet Delete Challenge Response" in LBIv1-REQ-260125
- Starting the vehicle when the vehicle is not in Enhanced Valet Mode with OpCode=" Challenge Response
- Exiting Secure Idle with OpCode="Challenge Response"

Each time when the LBIServer determines a password is invalid (per LBIv1-REQ-268538), when starting vehicle in Enhanced Valet Mode (OpCode=" Valet Start Challenge Response") it shall increment invalid password counter N_NumberOfInvalidValetAttempts

4.4.2.17 LBIv1-REQ-270254/A-Response of Valid Password Entry for Reset

In the response for reset challenge (OpCode="Reset Challenge Response"), if LBIServer determines password is valid per REQ-268538, it shall notify LBIClient of this with BackupIgnition_Rsp(RspCode="Reset Challenge Response Acknowledge", RspStatus="Valid Password").

4.4.2.18 LBIv1-REQ-270259/A-HMI Display of Valid Password Entry for Reset

After LBIClient receives a valid notification, BackupIgnition_Rsp(RespCode="Reset Challenge Response", RspStatus="Valid Password"), from the LBIServer, the LBIClient HMI shall notify the user that the entered password has been accepted and shall also display the password creation screen for user to enter new password.

4.4.2.19 LBIv1-REQ-260065/B-Check Entered Password against Password Rules

In the Process of creating or resetting a backup password, the LBIClient shall check entered password against password requirements in real time.

The LBIClient hall not allow the user to proceed to the next screen until their password meets the minimum requirements as defined in LBIv1-REQ-260164 and LBIv1-REQ-264928.

4.4.2.20 LBIv1-REQ-270524/A-HMI Solicit Password Twice

Once a rule-complied password is entered, the LBIClient HMI shall require the user to enter that password twice, and shall check if both codes match in real time.

4.4.2.21 LBIv1-REQ-260064/A-Check Match for Two Entered Passwords

In the process of creating or resetting a backup password, the LBIClient HMI shall check if two passwords match.

In the cases of passwords do not match, the LBIClient HMI shall notify the user and provide an option to retry.

4.4.2.22 LBIv1-REQ-260066/A-Transmit Rule-Complied Password

In the creating or resetting backup password process, once a rule-complied password is entered twice, the LBIClient shall compute a hash of entered password and shall send it to the LBIServer with the key index of the selected device via BckupIgnition_Rq as defined in REQ-267240.

4.4.2.23 LBIv1-REQ-267240/A-Transmit Password

Whenever there is a need to transmit entered password to the LBIServer for validation, the LBIClient shall send hashed entered password with the key index of the selected device via BackupIgnition_Rq with encoding value set as

OpCode=

- "Password Transmit", when creating a brand new password (not in the reset process)

or

- "Reset 1 Password Transmit", when resetting password with BackupIgnition_Rsp (RspCode= "Salt and Check for Paak with Passwords Response", RspStatus= "One PaaK w/ Password and No Fob In Vehicle" or "Two+ PaaK w/ Password and No Fob In Vehicle").

or

- "Reset 2 Password Transmit", when resetting password with BackupIgnition_Rsp (RspCode= "Salt and Check for Paak with Passwords Response", RspStatus= "One PaaK w/ Password and Fob In Vehicle" or "Two+ PaaK w/ Password and Fob In Vehicle").
-

KeyIndex = KeyIndex
Password = Password

4.4.2.24 LBIv1-REQ-260067/B-Trigger Interior Registry KeyFob Search for Password Response

In the creating or resetting backup password process, upon receiving the hashed password and keyindex via BackupIgnition_Rq (OpCode= "Password Transmit" or" Reset 2 Password Transmit"), the LBIServer shall check that the phone associated with the received key index and key fob are still in the vehicle via an Interior Registry KeyFob search by executing LBIv1-FUN-REQ-302285.

4.4.2.25 LBIv1-REQ-271305/A-Search Paak for Password Response

In the creating or resetting backup password process, upon receiving the hashed password and keyindex via BackupIgnition_Rq (OpCode= "Password Transmit" or "Reset 1 Password Transmit" or Reset 2 Password Transmit), the LBIServer shall check that the phone associated with the received key index (Reset 1 and 2) and key fob (Reset 2) is still in the vehicle.

4.4.2.26 LBIv1-REQ-271306/A-Response of No PaaK for Reset1 Password Transmit

During the Reset1 password resetting process, if the LBIServer determines that the phone is no longer in the vehicle for the response of transmitted password, it shall notify LBIClient of this via BackupIgnition_Rsp.with the encoding values set as below:

RspCode="Reset 1 Password Response"
RspStatus="PaaK No Longer Detected"
VariableData shall all be set to Zero

4.4.2.27 LBIv1-REQ-271307/A-HMI Display of No Device for Reset Password Transmit

Upon receiving the notification that the user selected phone is no longer in the vehicle via BackupIgnition_Rsp, the LBIClient HMI shall then display a message that the user selected phone is no longer detected in the vehicle and instruct user to bring Paak to vehicle before retry.

The PaaK no longer detected status is indicated in BackupIgnition_Rsp(RspCode=" Reset 1 Password Response",RspStatus="PaaK No Longer Detected").

4.4.2.28 LBIv1-REQ-271308/A-Response of No Device for Reset2 Password Transmit

During the resetting backup password process, if the LBIServer determines that the phone or key fob is no longer in the vehicle for the response of BackupIgnition_Rq(OpCode=" Reset 2 Password Response"), it shall notify LBIClient of this via BackupIgnition_Rsp.with the encoding values set as below:

RspCode=" Reset 2 Password Response"
RspStatus=

- "PaaK No Longer Detected" or
- "Fob No Longer Detected" or
- "PaaK and Fob No Longer Detected"

4.4.2.29 LBIv1-REQ-260068/A-HMI Display of No Device for Password Transmit

Upon receiving the notification that the user selected Paak and or a key fob is no longer in the vehicle via BackupIgnition_Rsp, the LBIClient HMI shall then display a message that the phone and/or key fob is no longer detected in the vehicle and instruct user to bring required devices to vehicle before retry.

The devices no longer detected condition is indicated in BackupIgnition_Rsp with the encoding values of
RspCode=

- "Password Response" or
- "Reset 2 Password Response"

RspStatus=

- "PaaK No Longer Detected", or
- "Fob No Longer Detected" or
- "PaaK and Fob No Longer Detected".

4.4.2.30 LBIv1-REQ-267620/B-Check Password Uniqueness

In response of BackupIngintion_Rq (OpCode=" Password Transmit" or "Reset 1Password Transmit" or "Reset 2 Password Transmit"), if the required devices are still in the vehicle when the LBIServer receives the password hash, it shall then verify that the entered password is not already being used.

The required devices are defined as

- The selected Paak and a KeyFob are both needed for OpCode=" Password Transmit" or "Reset 2 Password Transmit"
- Only the selected Paak is needed for OpCode=" Reset1 Password Transmit".

4.4.2.31 LBIv1-REQ-263585/A-Response of Password Already Used

In the response of transmitted password, if the entered password is already being used, the LBIServer shall notify the LBIClient via BackupIgnition_Rsp with the encoding values set as below:

RspCode=

- "Password Response" when OpCode="Password Response" or
- "Reset 1Password Response" when OpCode="Reset 1Password Response" or
- "Reset 2 Password Response" when OpCode="Rest2 Password Response"

RspStatus="Password Already Used"
VariableData shall set to zero

4.4.2.32 LBIv1-REQ-260069/A-HMI Display for Password Already Used

Upon receiving the notification of Password already being used via BackupIngnition_Rsp(RspStatus="Password Already Used"), the LBIClient HMI shall display a message that password is already being used and instruct user to enter a different password.

4.4.2.33 LBIv1-REQ-264889/A-Initiate KeyPad Code Deletion for Reset

In the process of resetting a password, if the entered new password is not already being used, then the LBIServer needs to delete the old password along with the keypad code if it has one before store the new password.

After determine that the entered new password is not already being used, the LBIServer shall check if a keypad code was created for the password with the received KeyIndex via BackupIgnition_Rq(Opcode="Reset 1 Password Transmit" or ="Reset 2 Password Transmit", *KeyIndex = KeyIndex*).

Only when a keypad code was created then the LBIServer shall initiate a keypad code deletion request by executing LBI-FUN-REQ-275727.

4.4.2.34 LBIv1-REQ-275719/A-Conditions of Password Deletion for Reset Process

In the process of resetting password, the LBIServer shall delete the current backup password only if one of the following conditions is met:

- A successful Keypad code deletion notification is received via KeyPadCodeProg_St(Delete)
- There is no Keypad Code associated with the received key index

4.4.2.35 LBIv1-REQ-269569/A-Reset Password Stored Conditions

In the process of resetting backup password, only when the three conditions listed below are all met, the LBIServer then shall store the hashed password in its HSM and shall associate it with the key index it received from BackupIngintion_Rq(OpCode="Reset 1 Password Transmit" or ="Reset 2 Password Transmit").

The three conditions are:

1. The newly entered password is not already being used as required by REQ-260064
2. The old backup password is successfully deleted from the LBIServer HSM as required by REQ-275719
3. 3(a) The Paak with the received key index still exists when OpCode="Reset 1Password Transmit", based on the result of executing REQ- 271305

or

3(b) Both the selected Paak with the received key index and a Keyfob are present when OpCode="Reset 2 Password Transmit" , based on the result of executing REQ- 271305 and REQ-260067

4.4.2.36 LBIv1-REQ-270515/B-Notification of Successful Password Creation to LBIClient2

Once a password is stored as defined per LBIv1-REQ-263587(for brand new password creation) or LBIv1-REQ-269569(for reset password), the LBIServer shall report this event, including the corresponding device KeyID, to the LBIClient2 via *LBIAlert_St with encoding values set as below:*

Event =

- *" Backup Password Created" if the Opcode is" Password Transmit"*

or

- *" Backup Password Reset" if the Opcode is" Rest 1 Password Transmit "or " Rest 2 Password Transmit"*

Source = Reserved

The corresponding device KeyID shall be *the KeyID of PaaK device associated with the backup password that is being created or reset.*

4.4.2.37 LBIv1-REQ-267309/A-Response of Successful Password Creation to LBIClient

Once a password is stored per REQ-263587(for brand new password creation) or REQ-269569(for reset password), the LBIServer shall report to the LBIClient the successful storage of this password via BackupIgnition_Rsp with

RspCode= “Password Response” or “Reset 1 Password Response”, or Rest 2 “Password Response”
RspStatus=”Password Created Successfully
VariableData shall all set to be zero

4.4.2.38 LBIv1-REQ-264893/A-HMI Display of Password Reset Successfully

Upon receiving *BackupIgnition_Rsp* (*RspCode =”Reset 1/2 Password Response”, RspStatus =”Password Created Successfully”)*, the LBIClient shall notify the user of successful password reset.and shall also inform the use that password reset is complete.

In the case of a failed keypad code deletion as indicated in RspStatus being “Password Deleted Successfully but Keypad Code Deleted Failed”, the LBIClient HMI shall notify the use about the successful password rest, the failed keypad code deletion and the complete of password reset.

4.4.2.39 LBIv1-REQ-260071/A-HMI Display of Keypad Code Creation Option

After notifying the user of successful creating or resetting a backup password, the LBIClient HMI shall also present the user the option to set up a personal keypad code.

4.4.2.40 LBIv1-REQ-260072/A-HMI Diaplay of Password Creation Complete

If the user chooses not to create a personal keypad code, the LBIClient HMI shall inform the user that backup password setup is complete and provide instructions on how to use the feature.

4.4.2.41 LBIv1-REQ-269633/A-Initiate Keypad Code Creation

If the user chooses to create a LBI keypad code right after a backup password created or reset, the LBIClient shall initiate the keypad code creation process as described in LBI-FUN-REQ-26957-Creating Keypad Code for Paak Device.

4.4.2.42 LBIv1-REQ-275722/A-Criteria of Reporting Failed Password Reset

The LBIServer shall report a failed password reset if any one of conditions below is met:

- The vehicle operation conditions defined by REQ- 264925 are no longer met before the deletion process completes
- The requested backup password is unable to be deleted from the LBIServer HSM
- The associated keypad code of requested backup password is unable to be deleted after multiple deleting requests as required by REQ-277531

4.4.2.43 LBIv1-REQ-275721/A-Response of Failed Password Reset to LBIClient

In the resetting password process, if the failure criteria described in REQ- 275722 is met, the LBIServer shall report to the LBIClient about the failure of password reset via BackupIgnition_Rsp with
RspCode=“Reset 1 Password Response”, or “Rest 2 “Password Response”
RspStatus=”Password Created Failed
VariableData shall all set to be Zero.

4.4.2.44 LBIv1-REQ-275723/A-HMI Display of Failed Password Reset

Upon receiving *BackupIgnition_Rsp* (*RspCode =”Reset 1/2 Password Response”, RspStatus =”Password Created Failed”)*, the LBIClient shall notify the user of unsuccessful password reset and shall also offer the user opportunity to retry.

4.4.2.45 LBIv1-REQ-267241/B-Reset Password Response

Upon receiving the hashed password and key index through BackupIgnition_Rq (OpCode= “Reset 1 Password Transmit”or “Reset 2 Password Transmit”), the LBIServer shall check that the phone and key fob (only if OpCode is “Reset 2 Password Transmit “) are still in the vehicle and the password uniqueness before it can response back the password creation result.

To obtain phone status for OpCode of “Reset 2 Password Transmit”, the LBIServer shall trigger a LBIServer2 Interior Registry Keyfob search to determine if a Keyfob is found inside the vehicle by executing LBIv1-FUN-REQ-302285.

.

Once determine all necessary devices are still present, the LBIServer shall check the uniqueness of password as defined in REQ-267620.

The LBIServer shall then report phone query result, key fob search result and password uniqueness status to the LBIClient via BackupIgnition_Rsp with the encoding values set as below:
If Opcode="Reset 1 Password Transmit"
then

RspCode="Reset 1 Password Response"
RspStatus =

- "PaaK No Longer Detected" or
- "Password Already Used" or
- "Password Created Successfully" or
- "Password Created Failed

If Opcode="Reset 2 Password Transmit"
then

RspCode="Reset 2 Password Response"
RspStatus =

- "PaaK No Longer Detected" or
- "Fob No Longer Detected" or
- "PaaK and Fob No Longer Detected" or
- "Password Already Used" or
- "Password Created Successfully" or
- "Password Created Failed"

VariableData shall not be transmitted for either RspCode.

4.4.2.46 LBIv1-REQ-266971/A-Error Handling Strategy for Password Entry Screen Challenge Request

Upon receiving Password entry request, PasswordEntryScreen_Rq(Active), if valid LBILockout_St is not available at this time, the LBIClient shall assume LBILockout_St (inactive) and shall display password entry screen as described in REQ- 260076.

4.4.2.47 LBIv1-REQ-260081/A-Response to Challenge with Hashed Password

The LBIClient shall respond to the LBIServer challenge of *BackupIgnition_Rsp(RspCode="Issue Challenge"*) with computed password hash per REQ-260080 via BackupIgnition_Rq(OpCode="Challenge Response"or "Valet Start Challenge Response" or "Reset Challenge response", KeyIndex = EOS, Password = Challenge Password, KeypadCode = EOS).

4.4.2.48 LBIv1-REQ-260082/A-Compare Hashed and Computed Password

When the LBIServer receives a challenge hash via BackupIgnition_Rq(OpCode="Challenge Response" or "Valet Start Challenge Response" or "Reset Challenge Response"), it shall compare it with the hashes that it computed for the stored passwords as required by REQ-270244 in order to determine if a password is valid.

4.4.2.49 LBIv1-REQ-268538/A-Criteria of Valid Password

A valid password is defined when both conditions are met:

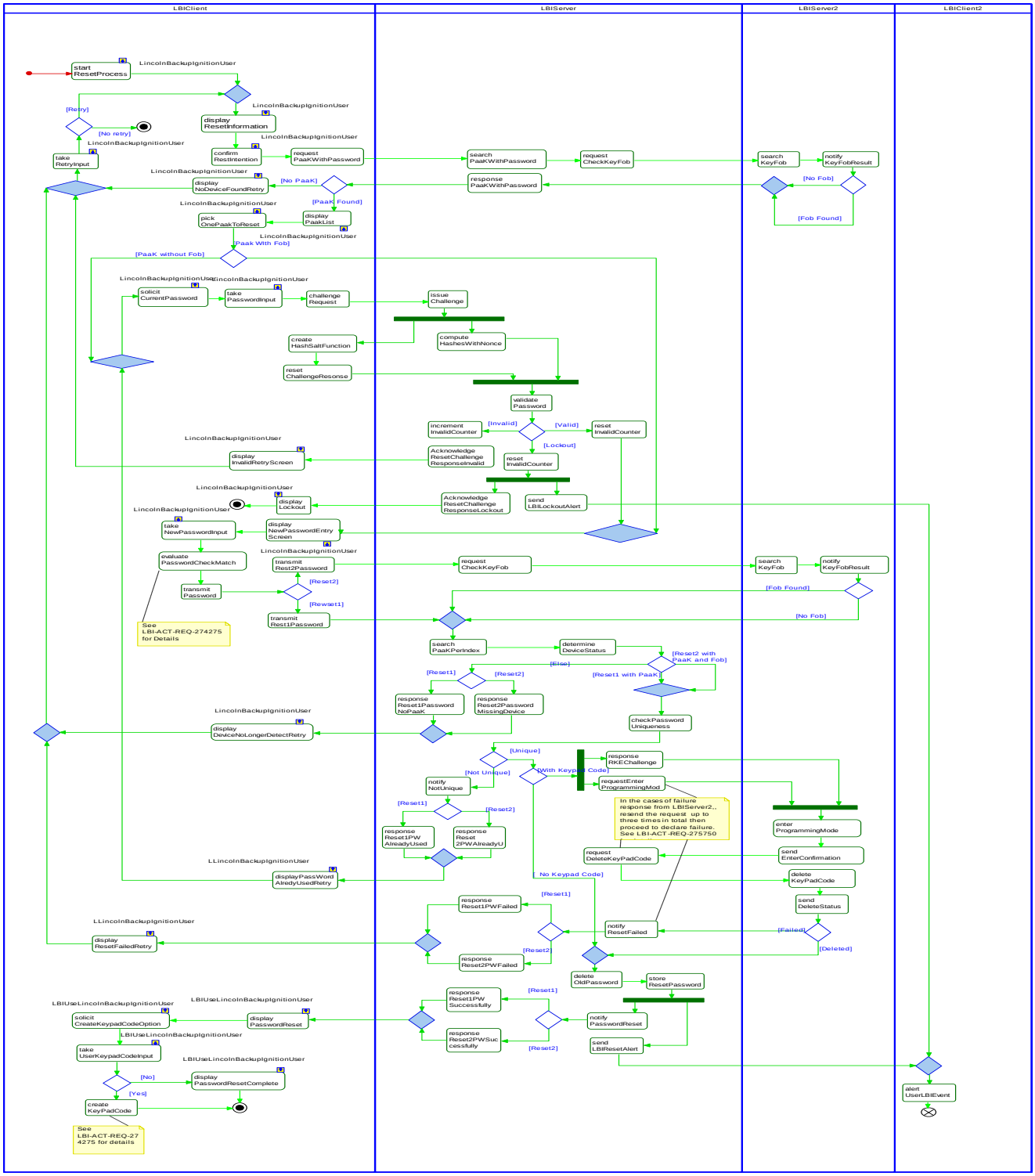
- The computed hashes, as required by REQ-270244, matches with that of received hashed password via BackupIgnition_Rq(OpCode="Challenge Response" or "Valet Start Challenge Response" or "Reset Challenge Response")

- The correct type of password is used in the correct mode. A backup password shall be treated as invalid when OpCode is "Valet Start Challenge Response" per REQ-267202. Likewise, an Enhanced Valet password, as defined by REQ-264929, shall be treated as invalid when OpCode is either "Valet Start Challenge Response" or "Reset Challenge Response")

White Box Views

4.4.2.50 Activity Diagrams

4.4.2.50.1 LBI-ACT-REQ-274280/A-Resetting Backup Password and Keypad Code for PaaK Device Activity Diagram



4.4.2.51 Sequence Diagrams

4.4.2.51.1 LBIv1-SD-REQ-271728/A-Resetting Backup Password

Constraints

Pre-Condition

1. The LBI User is inside the vehicle.
2. One associated PaaK with password is inside the vehicle
3. The vehicle ignition is in Run
4. The vehicle transmission is in Park
5. The vehicle is not locked out by LBI Feature
6. The vehicle is not in Enhanced Valet Mode

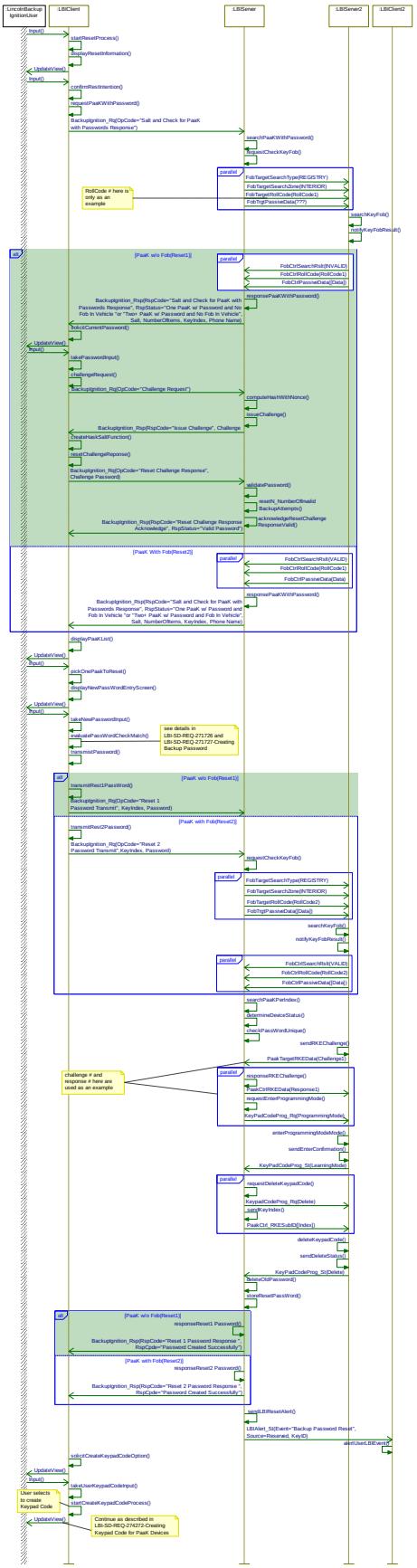
Scenarios

Normal Usage

Post-Condition

1. The backup password of a selected Paak is reset
2. A Notification of a backup password has been reset is sent to the LBI User via e-mail and via a phone message to the PaaK device which a backup password is just reset and associated with

Sequence Diagram



4.4.2.51.2 LBIv1-SD-REQ-271729/A-Resetting Backup Password - Paak and Fob

Constraints

Pre-Condition

1. The LBI User is inside the vehicle.
2. One associated PaaK with password is inside the vehicle
3. The vehicle ignition is in Run
4. The vehicle transmission is in Park
5. The vehicle is not locked out by LBI Feature
6. The vehicle is not in Enhanced Valet Mode

Scenarios

Normal Usage

Post-Condition

1. The backup password of a selected Paak is reset
2. A Notification of a backup password has been reset is sent to the LBI User via e-mail and via a phone message to the PaaK device which a backup password is just reset and associated with

4.5 LBIv1-FUN-REQ-258451/A-Generating Valet Password and Keypad Code

4.5.1 Use Cases

4.5.1.1 LBIv1-UC-REQ-260115/B-Generating Valet Backup Password and Keypad Code with PaaK

Actors	The LBI User
Pre-conditions	1. The LBI User has previously activated PaaK feature for the vehicle 2. The vehicle ignition is in Run 3. The vehicle transmission is in Park 4. The LBI User is inside vehicle 5. At least one PaaK and no KeyFob is inside the vehicle 6. The vehicle is not locked out by LBI feature 7. The vehicle is not in Enhanced Valet Mode
Scenario Description	The LBI User presses Valet button from Settings Menu at Center Stack Menu
Post-conditions	1. One Enhanced Valet password is generated 2. Connected PaaK devices receive a notification with the Enhanced Valet password 3. The LBI HMI displays a message that says an Enhanced Valet password is created and has been sent to connected device 4. The Vehicle enters Enhanced Valet Mode 5. The LBI HMI displays a screen that contains the valet password, Active Enhanced Valet Mode status and an explanation of how to use Enhanced Valet password 6. Enhanced Valet password will be displayed until the vehicle is shutdown
List of Exception Use Cases	
Interfaces	The LBIClient The LBIClient2 The LBIServer The LBIServe2 SDN PaaK

4.5.1.2 LBIv1-UC-REQ-276994/B-Generating Valet Backup Password and Keypad Code without PaaK

Actors	The LBI User

Pre-conditions	1. The LBI User has previously activated PaaK feature for the vehicle 2. The vehicle ignition is in Run 3. The vehicle transmission is in Park 4. The LBI User is inside vehicle 5. No PaaK and no KeyFob is inside the vehicle 6. The vehicle is not locked out by LBI feature 7. The vehicle is not in Enhanced Valet Mode
Scenario Description	1. The LBI User presses Valet button from Settings Menu at Center Stack Menu 2. The LBI HMI displays a backup entry screen 3. The LBI User enters a valid backup password
Post-conditions	1. One Enhanced Valet password is generated 2. The Vehicle enters Enhanced Valet Mode 3. The LBI HMI displays a screen that contains the valet password, Active Enhanced Valet Mode status and an explanation of how to use Enhanced Valet password. 4. Valet password will be displayed in Lincoln Backup Ignition HMI Menu until vehicle is shutdown
List of Exception Use Cases	
Interfaces	The LBIClient The LBICerver2 The LBICerve The LBIClient2 SDN Paak

4.5.2 Requirements

4.5.2.1 LBIv1-REQ-267973/A-Configurable Parameter to Enable Enhanced Valet Mode HMI

The LBIClient shall use Paak configurable parameter along with other conditions as defined in Req-260104 to determine whether to allow access to Enhanced Valet Mode.

Please refer to Latest Infotainment Diagnostic Specification for exact Paak configuration Bit information.

If the Paak configurable parameter indicates that the vehicle does not support Paak then the LBIClient shall not allow access to Enhanced Valet Mode

4.5.2.2 LBIv1-REQ-260104/A-Condition to Query Devices for Access Enhanced Valet Mode

The LBIClient shall initiate Paak query as defined in REQ-264934 in order to determine whether to allow the access to Enhanced Valet Mode if all the following conditions are all met:

- The vehicle is equipped with Paak feature as described in REQ-267973
- At least a backup password is stored in the LBIServer as indicated PasswordReady_St(Active)
- All conditions required by REQ-264925

4.5.2.3 LBIv1-REQ-264934/A-Query for PaaKs to Enter Valet Mode

To determine whether to allow access to Enhanced Valet Mode, the LBIClient shall query the LBIServer for PaaK and KeyFob devices in the vehicle via BackupIgnition_Rq(OpCode="Check for Keys to Enter Valet Mode", VariableData=0x0).

4.5.2.4 LBIv1-REQ-271447/B-Trigger Registry KeyFob Search for Enter Enhanced Valet Mode

Upon receiving BackupIgnition_Rq(OpCode=" *Check for Keys to Enter Valet Mode*", KeyIndex=0x00), the LBIServer shall trigger a LBIServer2 Interior Registry KeyFob searchto determine if a Keyfob is found inside the vehicle by executing LBIv1-FUN-REQ-302285.

4.5.2.5 LBIv1-REQ-267088/A-Response of Missing Devices or Fob Detections

In response to BackupIgnition_Rq(OpCode="Check for Keys to Enter Valet Mode"), if both PaaK and Fob are not detected or any KeyFob is detected, the LBIServer shall report to the LBIClient via BackupIgnition_Rsp with encoding values set as below

RspCode="Check for Keys to Enter Valet Mode") and different results as below:
RspStatus =

- "Fob in Vehicle", or
- "*No PaaK and No Fob In Vehicle"*

VariableData shall set to 0x0

4.5.2.6 LBIv1-REQ-264935/A-HMI Display of Legacy Valet Mode Password Entry for Fob Present

If a key fob is detected during "Check for Keys to Enter Valet Mode" search, the LBIClient HMI shall display Legacy Valet Mode (PIN entry screen) regardless of whether a phone is detected.

In this case, BackupIgnition_Rsp(RspCode) is "Check for Keys to Enter Valet Mode", and BackupIgnition_Rsp(RspStatus) is :Fob In Vehicle".

4.5.2.7 LBIv1-REQ-264929/B-Conditions to Generate Enhanced Valet Code with Paak

If at least one or more PaaK devices are detected but no KeyFob is found in the vehicle during search of "Check for Keys to Enter Valet Mode", the LBIServer shall generate a random numeric password with the length defined per LBIv1-REQ-3178 and shall associate it with the Enhanced Valet key index as defined in LBIv1-REQ-270053.

4.5.2.8 LBIv1-REQ-264911/B-HMI Display with No Device

The LBIClient HMI shall display the backup password entry screen and shall ask the user to enter a backup password to activate Enhanced Valet Mode when neither a PaaK device nor a keyFob is detected in the vehicle during the search as indicated in BackupIgnition_Rsp (*RspCode =" Check for Keys to Enter Valet Mode", RspStatus =" No PaaK and No Fob In Vehicle").*

4.5.2.9 LBIv1-REQ-260112/B-Condition to Generate Enhanced Valet Code with No Device

When no device is detected for the query request of *BackupIgnition_Rq(OpCode="Check for Keys to Enter Valet Mode"),* and a valid backup password per LBIv1-REQ-268538 is entered, the LBIServer shall generate a random number with the length (number of digits) defined by LBIv1-REQ-317825.

4.5.2.10 LBIv1-REQ-317825/A-Length of Enhanced Valet Password

The LBIServer shall generate a random number with the length of (Y+3)-digit where the Y is the length of Keypad code calls out in Keypad code configuration parameter for different markets.

Example (1):
When the keypad code configuration parameter calls out for a 5-digit keypad code (Y=5), the LBIServer shall generate a random 8-digit (5+3) Enhance Valet Password.

Example (2):
When the keypad code configuration parameter calls out for a 7-digit keypad code (Y=7), the LBIServer shall generate a random 10-digit (7+3) Enhance Valet Password.

4.5.2.11 LBIv1-REQ-268183/B-Valid Enhanced Valet Password Rules

The Enhanced Valet Password shall adhere to the following rules:

- It shall not consist of all the same numbers (e.g. 11111111)

- It shall not include more than two numbers in sequence (e.g. 123)
- It shall not be the same as another existing password

If the generated Enhanced Valet Password does not comply with the rules listed above, the LBIServer shall regenerate a different password until the password complies with the Enhanced Valet password rules.

4.5.2.12 LBIv1-REQ-317844/A-Enhanced Valet Password Usages

Enhanced Valet Password is a random number with different length for different markets. The usages of the random is defined as below:

- The first (X-3) digests of the random number is the Enhanced Valet Keypad code and it shall be used to unlock vehicle door to access the vehicle
- The entire random number is the Enhanced Valet Password and it shall be used to start and move the vehicle

Note: X is the number of digits of the random number as defined by LBIv1-REQ-317825

4.5.2.13 LBIv1-REQ-268184/B-Track PaaKs for Enhanced Valet Password Alerts

If the LBIServer generates a random number for Enhanced Valet Password, it shall store the key indexes of all PaaK devices that were detected inside the vehicle during the LBIClient-requested key search, *BackupIgnition_Rq(OpCode="Check for Keys to Enter Valet Mode")*

Note: This requirement is needed for sending Valet alert to PaaKs via the LBIClient2 as required by LBIv1-REQ-271426

4.5.2.14 LBIv1-REQ-264904/B-Initiate Keypad Code Storing for Enhanced Valet Password

After generating a rule-compliant random number for Enhanced Valet Password per LBIv1-REQ-268183, the LBIServer shall initiate the Enhanced Valet keypad code storing request by executing LBI-FUN-REQ-275727.

4.5.2.15 LBIv1-REQ-264908/B-Store Enhanced Valet Password

In the process of generating Enhanced Valet Password, once receiving Enhanced Valet keypad code storage notification via *KeyPadCodeProg_D_St(Add)*, the LBIServer shall store the valet password hash and then associate this password entry with the Valet key index as defined by LBIv1-REQ-270053 in its HSM.

4.5.2.16 LBIv1-REQ-270053/A-Enhanced Valet Password KeyIndex

Key Index of 63 shall be reserved to use as the identifier for Enhanced Valet password.

4.5.2.17 LBIv1-REQ-271426/B-Notification of Successful Enhanced Valet Password Creation to LBIClient2

When the LBIServer stores a new valet password hash, it shall report this event, including the corresponding device KeyID, to the LBIClient2 via *LBIAlert_St with encoding values set as*

Event =" Valet Password Created"
Source =

- Password *if BackupIgnition_Rq(OpCode=Valet Create challenge Response) in the case that a backup password is used to create valet password*

or

- *PaaK if BackupIgnition_Rq(OpCode ="Check for Keys to Enter Valet Mode") in the case that a Paak is preset while valet password is created*

The KeyID shall be the KeyIDs of the PaaK devices that were in the vehicle at the time of valet password generation as described in LBIv1-REQ-268184.

4.5.2.18 LBIv1-REQ-271428/B-Bluetooth Notification of Successful Enhanced Valet Password Creation

When the LBIServer stores a new valet password hash, it shall send the Enhanced Valet password (plain text) via encrypted PaaK BLE interface to all PaaK devices that were detected in the vehicle during the LBIClient-requested key search. The valet password shall be contained within the payload, defined below, of a GATT write command (reference PaaK BLE Communication Protocol).

Message Parameters	Value	Size	Description
Command	0x15	1 Byte	Write valet password command
Payload	0x00000000 – 0x05F5E0FF	4 Bytes	numeric valet password represented as an unsigned integer
Key Index	0x00 – 0x3F	1 Byte	Key index of device to which password will be sent. Used as identification for return status.
Padding	0x0202	2 Bytes	Padding following requirements in Phone as a Key BLE Communication Protocol, Section 9

4.5.2.19 LBIv1-REQ-264909/A-Response of Successful Enhanced Valet Password Creation with Delivery to LBIClient

In order to response back to the LBIClient about Enhanced Valet password creation and delivery status, the LBIServer shall wait for up to a fixed time as defined by T_PaaK Response Timer for at least one PaaK device confirmation as defined by the table below:

Message Parameters	Value	Size	Description
Command	0x16	1 Byte	Write valet password acknowledge
Status	0x32 – 0x33	1 Byte	Success (0x32) means PaaK device received password. Failure (0x33) means PaaK device received password, but could not process it.
Key Index	0x00 – 0x3F	1 Byte	Key index of device providing status.
Padding	0x0505050505	5 Bytes	Padding following requirements in Phone as a Key BLE Communication Protocol, Section 9

If one PaaK device confirms password delivery (i.e. Success, according to table above), the LBIserver shall immediately send confirmation of this fact with the Enhanced Valet password (plain text) to the LBIClient via *BackupIgnition_Rsp* (*RspCode ="Check for Keys to Enter Valet Mode", RspStatus ="* Password Created Successfully *and Delivered to PaaK", Valet Password = Valet Password)*

4.5.2.20 LBIv1-TMR-REQ-276913/A-T_PaaK Response Timer

Name	Description	Units	Range	Resolution	Default
T_PaaK Response Timer	Minimum time that the LBIServer shall wait for the PaaK Enhanced Valet password delivery response Note: Use the default value	sec	2-5	1	1

4.5.2.21 LBIv1-REQ-271429/A-Delete Plain Text Enhanced Valet Password once Transmitted

Once the LBIServer sends the plain text valet password to LBIClient, it shall delete it from memory.

4.5.2.22 LBIv1-REQ-276914/A-Response of Successful Enhanced Valet Password Creation to LBIClient

In order to response back to the LBIClient about Enhanced Valet password creation and delivery status, the LBIServer shall wait for up to a fixed time as defined by T_PaaK Response Timer for at least one PaaK device confirmation as defined by the table below:

Message Parameters	Value	Size	Description
Command	0x16	1 Byte	Write valet password acknowledge
Status	0x32 – 0x33	1 Byte	Success (0x32) means PaaK device received password. Failure (0x33) means PaaK device received password, but could not process it.
Key Index	0x00 – 0x3F	1 Byte	Key index of device providing status.
Padding	0x0505050505	5 Bytes	Padding following requirements in Phone as a Key BLE Communication Protocol, Section 9

If one PaaK device confirms password delivery is failed per the table above, the LBIserver shall immediately send confirmation of this fact with the valet password (plain text) to the LBIClient via *BackupIgnition_Rsp* (*RspCode ="Check for Keys to Enter Valet Mode", RspStatus ="* Password Created Successfully"*, Valet Password = Valet Password)*

4.5.2.23 LBIv1-REQ-275502/A-HMI Display of Successful Enhanced Valet Password Creation

When receiving a successful response via BackupIgnition*_Rsp(RspCode="Check for Keys to Enter Valet Mode", RspStatus ="Password Created Successfully" or "Password Created Successfully and Delivered to PaaK"*), the LBIClient shall notify user of Valet password creation and delivery status.

4.5.2.24 LBIv1-REQ-264910/A-Condition for HMI to Enter Enhanced Valet Mode

After LBIClient receives the valid valet password from the LBIServer via BackupIgnition_Rsp(RspCode="Valet Create Challenge Response", *RspStatus="* Password Created Successfully" or" Password Created Successfully and Delivered to PaaK"), the LBIClient HMI shall enter Enhanced Valet Mode.

4.5.2.25 LBIv1-REQ-275620/A-Track Enhanced Valet Mode Status

The LBIClient shall keep track of Enhanced Valet Mode status in order to determine which type of password entry screen should be provided in various circumstances.

4.5.2.26 LBIv1-REQ-260114/B-HMI Display of Active Enhanced Valet Mode

Once entering Enhanced Valet Mode as required by LBIv1-REQ-264910, the LBIClient shall display an Enhanced Valet Mode screen until the Center Stack HMI is suspended or the vehicle exits Enhanced Valet Mode.

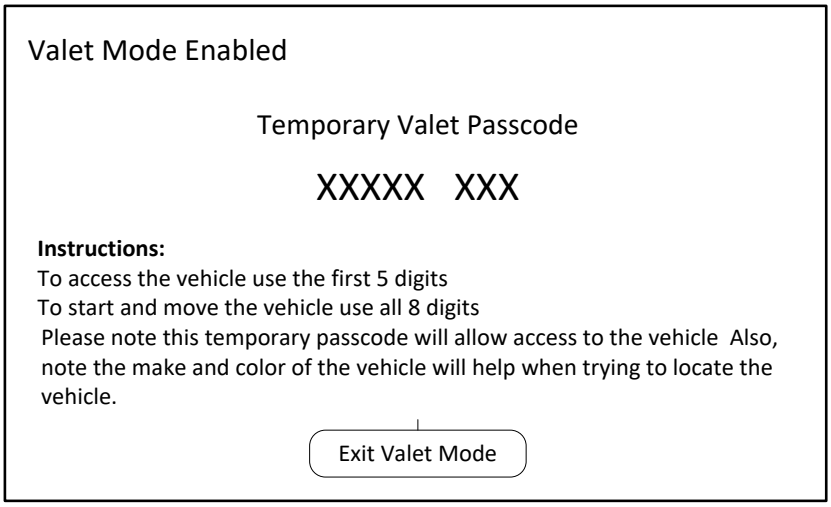
The contents of Enhanced Valet Mode screen shall include but not limit to

- Active Status of Enhanced Valet Mode
- The Enhanced Valet Password
    - The LBIClient shall comply with LBIv1-REQ-317825 to display correct length of Enhanced Valet Password for different markets
- Information on how to use the Enhanced Valet Password per LBIv1- REQ-317844/Enhanced Valet Password Usages
- The Exit Valet Mode option menu

- The Exit Valet Mode option menu shall comply with LBIv1-REQ-264925.

The Enhanced Valet password shall not be displayed again once the Center Stack HMI suspended.

Below is an example of Enhanced Valet Mode screen for the market that calls out 5-digit of Keypad code. Please note that this example only serves as a design aid and does not necessarily represent the final implementation nor represent all displays for all markets that may call out different length of Keypad code.



4.5.2.27 LBIv1-REQ-270560/A-Indication of Vehicle in Enhanced Valet Mode

Whenever an active password entry is associated with Valet Password KeyIndex as defined in REQ-270053, the LBIServer shall recognize this as vehicle being in Enhanced Valet Mode.

4.5.2.28 LBIv1-REQ-271430/A-Storage of Enhanced Valet Mode Status

The LBIClient shall hold the status(active, inactive) of Enhanced Valet Mode in NVM.

4.5.2.29 LBIv1-REQ-276997/B-Criteria of Reporting Failed Enhanced Valet Password Creation

The LBIServer shall report a failed Enhanced Valet password creation if any one of conditions below is met:

- The vehicle operation conditions defined by REQ- 264925 are no longer met before the creation process completes
- The Enhanced Valet password is unable to be stored by the LBIServer
- The Enhanced Valet keypad code is unable to be stored by The LBIServer2 after multiple entering programming mode or storing requests as required by REQ-277531

4.5.2.30 LBIv1-REQ-271449/A-Response of Failed Enhanced Valet Password Creation

When any one of criteria defined by REQ-276997 is met, the LBIServer shall send a failure notification to the LBIClient via *BackupIgnition_Rsp* with encoding vales set as:

RspCode =

- *"Check for Keys to Enter Valet Mode" if OpCode is "*Check for Keys to Enter Valet Mode

or

- "Valet Create Challenge Response Acknowledge" if OpCode is "Valet Create Challenge Response"

RspStatus="Password Created Failed:
VariableData shall set to zero

4.5.2.31 LBIv1-REQ-271427/A-HMI Display of Failed Enhanced Valet Password Creation

When receiving programming failure response via BackupIgnition*_Rsp(RspCode="Check for Keys to Enter Valet Mode" or "*Valet Create Challenge Response Acknowledge"*, RspStatus ="Password Created Failed"*), the LBIClient HMI shall notify the user about the failure of Enhanced Valet password creation.

4.5.2.32 LBIv1-REQ-280630/A-HMI Display of Successful Password but Unsuccessful Keypad Deletion

In the case of a failed keypad code deletion as indicated in RspStatus being "Password Deleted Successfully but Keypad Code Deleted Failed", the LBIClient HMI shall notify the use about the successful Enhanced Valet password deletion and the failed Enhanced Valet Keypad Code deletion.

4.5.2.33 LBIv1-REQ-304593/A-Actions after Duplicate LBI Enhanced Valet Keypad Code

If the LBIServer receives *KeyPadCodeProg_St(Duplicate)* when it requests to store an Enhanced Valet keypad code as indicated in *BackupIgnition_Rq(OpCode = "*Check for Keys to Enter Valet Mode" or "Valet Create Challenge Response"), the LBIServer shall generate a new Enhanced Valet password and re-send the first five digits to the LBIServer2 for storing

- The request of storing newly generated Valet keypad code shall start the entire keypad programming process from the beginning (i.e. from executing LBIv1-REQ-270265
- The resent process due to duplicate Enhanced Valet keypad code shall comply with LBIv1-REQ-277531-Resend Keypad Code Request Requirement
- In the case when maximum resent, defined by LBIv1-REQ-277531, had been executed but the response is still *KeyPadCodeProg_St(Duplicate),* the LBIServer shall report Valet Password created failed as described in LBIv1-REQ-271449

4.5.2.34 LBIv1-REQ-312524/A-Preconditions for Response to LBI Requests

The LBIServer shall not response the request BackupIgnition_Rq for the functions defined in LBIv1-REQ- 264936 with the exception of staring the vehicle (BackupIgnition_Rq (OpCode = "Challenge Request", "Challenge Response" or "Valet Start Challenge Response") unless the conditions below are all met.

- The vehicle ignition is in RUN as indicated in IgnitionStatus_St(*Run*)
- The vehicle transmission must be in the Park position as indicated in *GearLvlPos_D_Actl(Park)*
- The vehicle is not in LBI lockout period as defined by LBIv1-REQ-268242
    - When the vehicle is in the LBI lockout period, the exception of staring the vehicle listed above shall be ignored too. In other words, all LBI requests, including starting the vehicle with LBI passwords, shall be ignored during LBI lockout period. Please refer to LBIv1-REQ-268502 for the lockout actions the LBIServer shall perform

Note: This verification check ensures that the LBI functions (except starting the vehicle with LBI passwords) are only accessible when vehicle ignition is in run, transmission is in park and the vehicle not in lockout period. This check will ensure that the LBIServer and does not get too far ahead if the LBIClient sends an opcode at the same time the user turns the ignition off or shift out of park.

4.5.2.35 LBIv1-REQ-332837/A-Authorization Timer When Generating Enhanced Valet Password With No Devices Detected in Vehicle

During the process of generating Enhanced Valet Password and with no Paak devices detected in vehicle interior zone, once the LBIServer receives a challenge hash from LBIClient (*BackupIgnition_Rq* with *OpCode = 0x0A = Valet Create Challenge Response, KeyIndex = 0x00, Password = Challenge Password, KeypadCode = EOS*), it shall compare it with the hashes that it computed for the stored passwords.

When the LBIServer determines that the challenge hash matches a calculated password hash it shall initiate a valet authorization timer, T_Valet Authorization Timer.

When the LBIServer is in a T_Valet Authorization Timer period while *Ignition_Status = Run*, the LBIServer shall respond positively (*PaakCtlType_D_Stat = 0x2 = Valid, PaakCtlIndx1_No_Actl = [63]*) to LBIServer2 Crypto start searches (*PaakTrgtType_D_Rq = 0x1 = Crypto, PaakTrgtZone_D_Rq = 0x1 = Interior)*.

After this timer expires, the LBIServer shall respond negatively (*PaakCtlType_D_Stat = 0x1 = Invalid*) to LBIServer2 Crypto Start searches when PaaK device has not been detected in vehicle interior zone.

The LBIServer shall de-activate the timer once the LBIServer2 clears search criteria (Type/Zone = Null/Null) or if the LBIServer detects the Ignition status changed from Run to Off.

Note: “Clears search criteria” means that search request has been received, the LBIServer has responded, and the LBIServer2 stops the active search.

If Enhanced valet keypad programming status reported by LBIServer2 to be duplicate, *KeyPadCodeProg_D_Stat = Duplicate,* then the LBIServer shall retry to re-generate Enhanced Valet password at the instance shown in the referenced sequence diagram LBIv1-SD-REQ-332841.

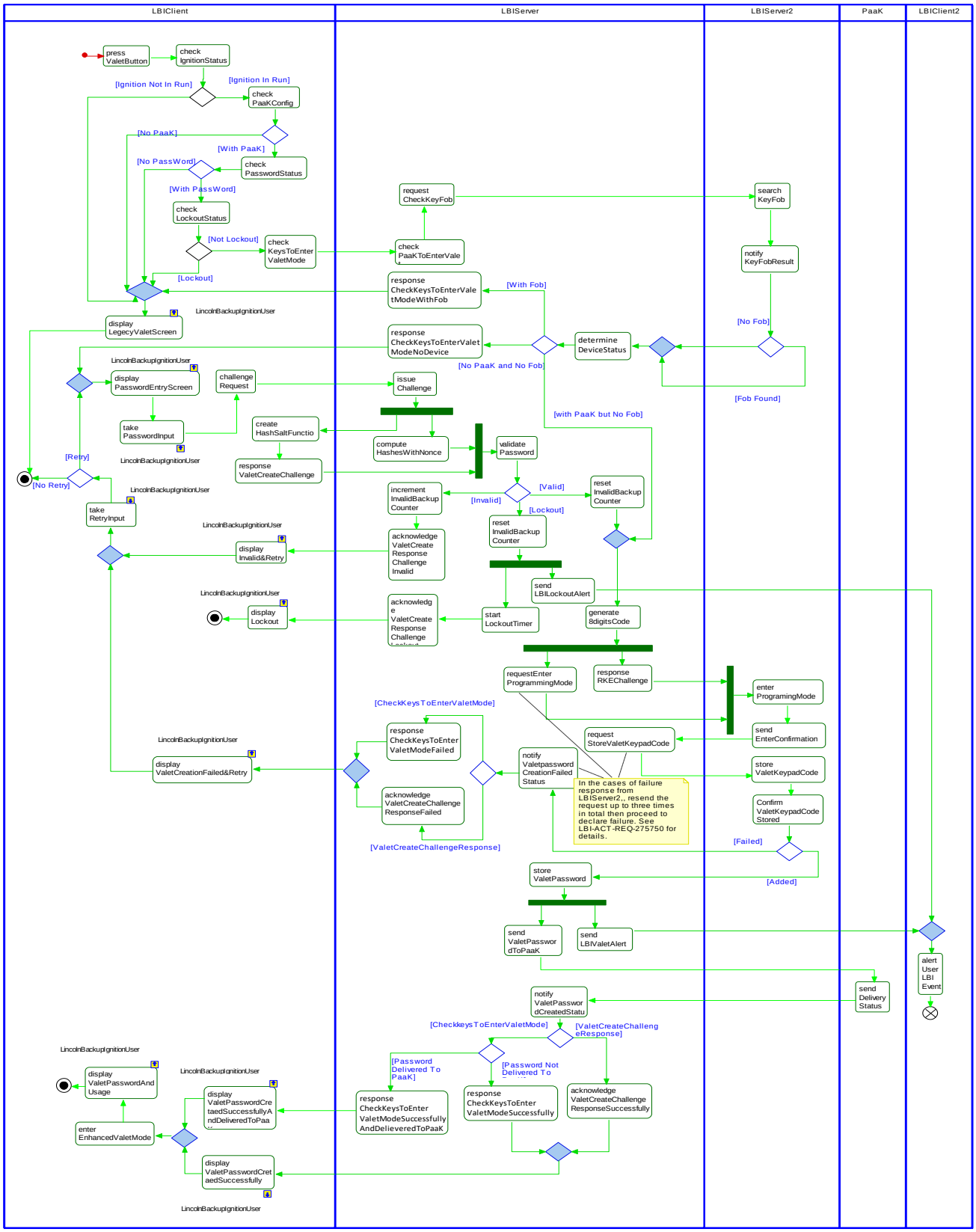
4.5.2.36 LBIv1-TMR-REQ-335313/A-T_Valet Authorization Timer

Name	Description	Units	Range	Resolution	Default
T_Valet Authorization Timer	The maximum time the LBIServer shall wait during the valet authorization period. Note: Use the default value	sec	10-30	1	20

White Box Views

4.5.2.37 Activity Diagrams

4.5.2.37.1 LBIv1-ACT-REQ-274281/A-Generating Valet Password and Keypad Code Activity Diagram



4.5.2.38 Sequence Diagrams

4.5.2.38.1 LBIv1-SD-REQ-277004/A-Generating Valet Password - Paak Present without Fob

Constraints

Pre-Condition

1. The LBI User is inside the vehicle.
2. No KeyFob is inside the vehicle
3. The vehicle ignition is in Run
4. The vehicle transmission is in Park
5. The vehicle is not locked out by LBI Feature
6. The vehicle is not in Enhanced Valet Mode

Scenarios

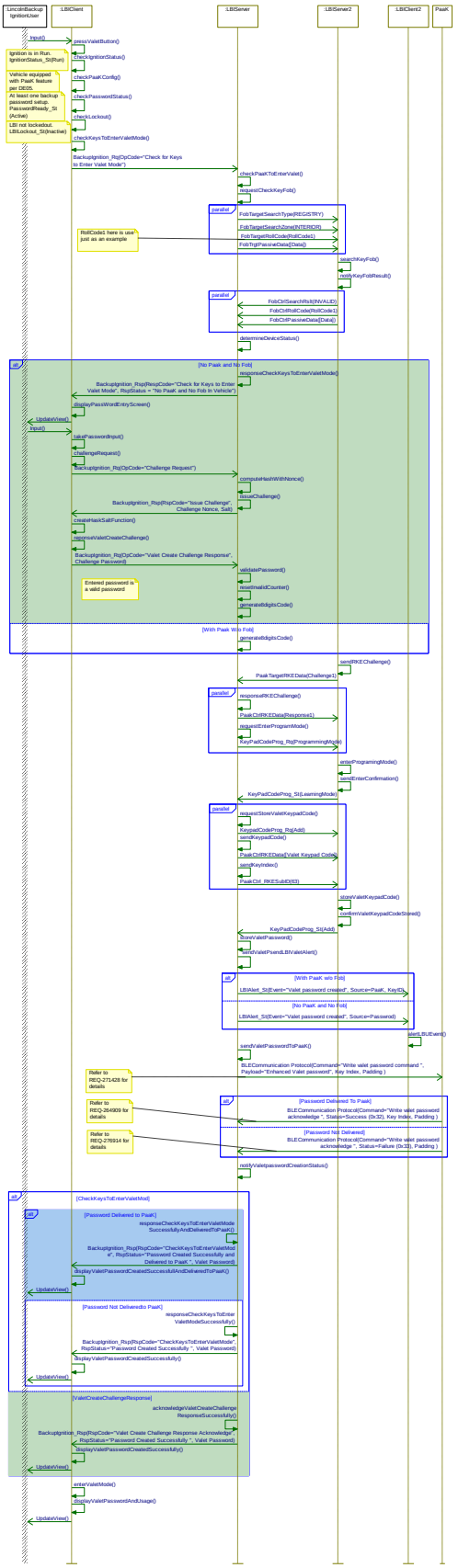
Normal Usage

1. The LBI User presses Valet button from Settings Menu at Center Stack Menu
2. With Paak inside thw vehicle, no extra step
3. Without PaaK inside the vehicle, the LBI User Enters a valid backup password

Post-Condition

1. One Enhanced Valet password is generated
2. Connected PaaK devices receive a notification with the Enhanced Valet password
3. The LBI HMI displays a message that says an Enhanced Valet password is created and has been sent to connected devices if there are any Connected PaaKs and at least one delivery is successful
4. The Vehicle enters Enhanced Valet Mode
5. The LBI HMI displays a screen that contains the valet password, Active Enhanced Valet Mode status and an explanation of how to use Enhanced Valet password
6. Enhanced Valet password will be displayed until the vehicle is shutdown

Sequence Diagram



4.5.2.38.2 LBIv1-SD-REQ-332841/A-Authorization Timer when Generating Enhanced Valet Password - No Devices Detected

Constraints

Pre-Condition

1. The LBI User is inside the vehicle.
2. No KeyFob and no PaaK device is inside the vehicle
3. The vehicle ignition is in Run
4. The vehicle transmission is in Park
5. The vehicle is not locked out by LBI Feature
6. The vehicle is not in Enhanced Valet Mode

Scenarios

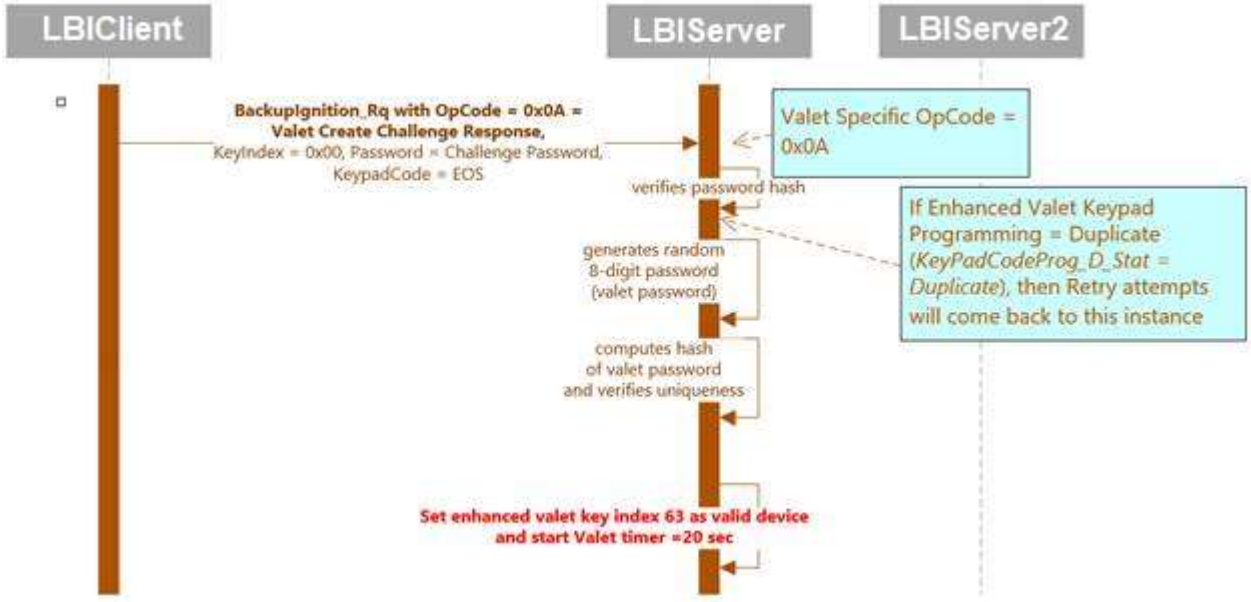
Normal Usage

1. The LBI User presses Valet button from Settings Menu at Center Stack Menu
2. With PaaK inside the vehicle, no extra step
3. Without PaaK inside the vehicle, the LBI User Enters a valid backup password

Post-Condition

1. One Enhanced Valet password is generated
2. Connected PaaK devices receive a notification with the Enhanced Valet password
3. The LBI HMI displays a message that says an Enhanced Valet password is created and has been sent to connected devices if there are any Connected PaaKs and at least one delivery is successful
4. The Vehicle enters Enhanced Valet Mode
5. The LBI HMI displays a screen that contains the valet password, Active Enhanced Valet Mode status and an explanation of how to use Enhanced Valet password
6. Enhanced Valet password will be displayed until the vehicle is shutdown

Sequence Diagram



4.6 LBIv1-FUN-REQ-258449/A-Starting Vehicle with Backup Password or Enhanced Valet Password

4.6.1 Use Cases

4.6.1.1 LBIv1-UC-REQ-258578/A-Starting Vehicle with Backup Password

Actors	The LBI User
Pre-conditions	1. The LBI User has previously created a backup password 2. The vehicle is locked. 3. The LBI User is outside vehicle

	4. No associated key fobs or phones-as-keys are near the vehicle 5. The vehicle is not in Enhanced Valet Mode nor in LBI Lockout stage 6. The vehicle ignition is not in Run
Scenario Description	1. The LBI User approaches the vehicle 2. The LBI User enters valid keypad code 3. The vehicle unlocks 4. The LBI User opens door and enters the vehicle 5. The LBI User presses brake pedal 6. The center stack device screen displays backup password entry screen 7. Without being inactive for more than a fixed period of time (defined by T_Password Entry Screen Inactive Timer), user enters valid backup password via LBI HMI. (This includes inputting the password then selecting Enter.) 8. LBI HMI displays message instructing the LBI User to start the vehicle 9. Within a fixed period of time (defined by T_Push to Start Timer), the LBI User presses start button while presses brake pedal at the same time 10. The vehicle starts with engine running
Post-conditions	1. The LBI User is able to drive away the vehicle 2. The LBI User is able to charge their phone-as-a-key in the vehicle 3. The notification that PaaK Backup has been used is sent to the LBI User via e-mail and a phone message to the PaaK whose backup password is just used
List of Exception Use Cases	1. The LBI User does not enter valid keypad code 2. The LBI User does not enter a valid backup password 3. The LBI User is inactive for more than the time defined by T_Password Entry Screen Inactive Timer while Lincoln Backup Ignition HMI Menu displays password entry screen 4. The LBI User does not start the vehicle within the time defined by T_Push to Start Timer of successful password entry 5. The LBI User presses start button without pressing brake pedal after password is accepted
Interfaces	The LBIClient The LBIClient2 The LBIServer The LBIServer2 SDN PaakFI I

4.6.1.2 LBIv1-UC-REQ-260117/A-Starting Vehicle with Valet Password

Actors	The LBI User
Pre-conditions	1. The LBI User has previously created a valet password 2. The vehicle is locked 3. The LBI User is outside the vehicle 4. No associated keyFobs or Paak are near the vehicle 5. The vehicle is in Enhanced Valet Mode 6. The vehicle is not in LBI Lockout stage 7. The vehicle ignition is off
Scenario Description	1. The LBI User approaches the vehicle 2. The LBI User enters valid keypad code 3. The vehicle unlocks 4. The LBI User opens door and enters the vehicle 5. The LBI User presses brake pedal 6. HMI displays Valet password entry screen 7. Without being inactive for more than a fixed period of time (defined by T_Password Entry Screen Inactive Timer), the LBI User enters valid Valet

	password via LBI HMI(this includes inputting the password then selecting Enter) 8. HMI displays message instructing the LBI User to start the vehicle. 9. Within a fixed period of time (defined by T_Push to Start Timer), the LBI User presses start button while presses brake pedal at same time 10. The vehicle starts with engine running
Post-conditions	1. The LBI User is able to drive away vehicle 2. The LBI User is able to charge their PaaK device in vehicle 3. The notification that PaaK Backup has been used is sent to the LBI User via e-mail and a phone message to ALL PaaK devices
List of Exception Use Cases	1. The LBI User does not enter valid keypad code 2. The LBI User does not enter a valid Valet password 3. The LBI User is inactive for more than the time defined by T_Password Entry Screen Inactive Timer while Lincoln Backup Ignition HMI Menu displays password entry screen 4. The LBI User does not start vehicle within the time defined by T_Push to Start Timer of successful password entry 5. The LBI User presses start button without pressing brake pedal after password is accepted
Interfaces	The LBIClient The LBIClient2 The LBIServer The LBIServer2 SDN PaakFI I

4.6.2 Requirements

4.6.2.1 LBIv1-REQ-260076/B-Conditions to Display Password Entry Screen for Starting Vehicle

In response to PasswordEntryScreen_Rq(Active) the LBIClient shall check the following three conditions to determine whether to display the password entry screen and which entry screen to display:

1. Customer Connectivity Settings (CCS) status
2. LBI lockout status as indicated in LBILockout_St
3. Enhanced Valet Mode status as required by LBIv1-REQ-275620

- The LBIClient shall not display any password entry screen when either one condition listed below is met
  - CCS is turned off
    or
  - or the vehicle is under LBI lockout (LBILockout_St =Active)

- The LBIClient shall display the backup password entry screen when the following three conditions are all met:

CCS is turned on
LBILockout_St =Inactive
The vehicle is not in Enhanced Valet Mode

- The LBIClient shall display the Enhanced Valet password entry screen when the following three conditions are all met:

  - CCS is turned on
  - LBILockout_St =Inactive
  - The vehicle is in Enhanced Valet Mode

4.6.2.2 LBIv1-REQ-264926/A-HMI Display of Specific Enhanced Valet Password Entry Screen

When in Enhanced Valet Mode, the SYNC HMI shall provide a valet-specific screen for entering the valet password when a valid key is not detected at startup. This screen shall be distinguishable from the screen for entering the backup password.

4.6.2.3 LBIv1-REQ-270870/A-Display Password Entry Screen in Extended Play Mode

Whenever there is a need to display the backup password entry screen or Enhanced Valet password entry screen per REQ-260076 and the vehicle is in Extended Play Mode, the LBIClient shall NOT enter the Restricted Infotainment Mode. Instead, the LBI Client shall stay in the Extended Play Mode and shall display the backup password entry screen or Enhanced Valet password entry screen without changing the functionalities of other features permitted in the Extended Play Mode

Note: this requirement (not entering Restricted Infotainment Mode while at the Extended Play Mode) is to ensure that LBI does not affect the functionalities of other features permitted in the Extended Play Mode.

4.6.2.4 LBIv1-REQ-260078/A-Priority for Display Password Entry Screen

If the LBIClient HMI is displaying the welcome animation when the password entry screen is required be displayed per REQ-260076, the LBIClient shall cancel the greeting timer and display the password entry screen after the welcome animation is finished.

4.6.2.5 LBIv1-REQ-260079/A-Time-out for Inactivity at Password Entry Screen

In the cases of user inactivity at the password entry screen for the fixed time as defined in T_Password Entry Screen Inactive Timer, the LBIClient shall dismiss the password entry screen.

4.6.2.6 LBIv1-TMR-REQ-268526/A-T_Password Entry Screen Inactive Timer

Name	Description	Units	Range	Resolution	Default
T_Password Entry Screen Inactive Timer	Maximum display time of password entry screen when the user is inactive (never touches the keyboard screen) Note: Use the default value	sec	25-35	1	30

4.6.2.7 LBIv1-REQ-275621/A-Reset Conditions for T_Password Entry Screen Inactive Timer

Keyboard button presses on screen or additional receptions of PasswordEntryScreen_Rq(Active) shall trigger the LBIClient to reset the inactivity timer with password entry screen being still displayed until either T_Password Entry Screen Inactive Timer expires

4.6.2.8 LBIv1-REQ-271249/B-Conditions for Suspend Center Stack HMI

After HMI Lockout Notification is dismissed per REQ-266972 or T_Password Entry Screen Inactive Timer expires, the LBIClient shall suspend the Center Stack HMI as described in P06- Power Management Specification if all conditions below are met:

- *IgnitionStatus_St = Off*
- *DelayAccy _St= Off*
- The Center Stack HMI is *not in Extended Play mode*

In other cases where the above conditions are not all met**,** the LBIClient shall not suspend the Center Stack HM. Instead, the LBIClient shall go to the previous screen for any one of cases listed below:

- *IgnitionStatus_St = Run*
- *DelayAccy_St= On*
- The Center Stack HMI is in Extended Play mode.

4.6.2.9 LBIv1-REQ-270384/B-Challenge Request

When the user enters a password at the password entry screen, the LBIClient shall request a challenge from LBIServer via BackupIgnition_Rsp(OpCode="Challenge Request", KeyIndex=0, *Password = EOS KeypadCode = EOS*).

The use cases of this Requirement are:

- Starting a vehicle with a backup password or an Enhanced Valet password
- Resetting a backup password
- Generating an Enhanced Valet Password with no device
- Deleting an Enhanced Valet password with no device

4.6.2.10 LBIv1-REQ-260077/A-Issue Cryptographic Nonce and Salt Challenge

After receiving a request for a challenge from LBIClient via BackupIgnition_Rq (OpCode="Challenge Request"), the LBIServer shall issue a challenge to the LBIClient via BackupIgnition_Rsp with cryptographic nonce and salt.

BackupIgnition_Rsp shall be set as
RspCode="Issue challenge"
RspStatus= *Reserved*
NumberOfItems= 0x00
Challenge Nonce = Challenge Nonce
Salt = Salt

4.6.2.11 LBIv1-REQ-266971/A-Error Handling Strategy for Password Entry Screen Challenge Request

Upon receiving Password entry request, PasswordEntryScreen_Rq(Active), if valid LBILockout_St is not available at this time, the LBIClient shall assume LBILockout_St (inactive) and shall display password entry screen as described in REQ- 260076.

4.6.2.12 LBIv1-REQ-270244/A-Compute Hash for All Stored Passwords

After receiving a request for a challenge from the LBIClient via BackupIgnition_Rq (OpCode="Challenge Request"), in addition to issue challenge to the LBIClient as described in REQ-260077, the LBIServer shall also compute, using the cryptographic nonce, another hash of all stored password hashes.

4.6.2.13 LBIv1-REQ-260080/A-Compute Hash for Entered Password

Once receiving BackupIgnition_Rsp(RspCode="Issue challenge"), the LBIClient shall compute a hash of entered password using received salt and then compute a hash of this result using received nonce.

4.6.2.14 LBIv1-REQ-260081/A-Response to Challenge with Hashed Password

The LBIClient shall respond to the LBIServer challenge of *BackupIgnition_Rsp(RspCode="Issue Challenge"*) with computed password hash per REQ-260080 via BackupIgnition_Rq(OpCode="Challenge Response"or "Valet Start Challenge Response" or "Reset Challenge response", KeyIndex = EOS, Password = Challenge Password, KeypadCode = EOS).

4.6.2.15 LBIv1-REQ-260082/A-Compare Hashed and Computed Password

When the LBIServer receives a challenge hash via BackupIgnition_Rq(OpCode="Challenge Response" or "Valet Start Challenge Response" or "Reset Challenge Response"), it shall compare it with the hashes that it computed for the stored passwords as required by REQ-270244 in order to determine if a password is valid.

4.6.2.16 LBIv1-REQ-268538/A-Criteria of Valid Password

A valid password is defined when both conditions are met:

- The computed hashes, as required by REQ-270244, matches with that of received hashed password via BackupIgnition_Rq(OpCode="Challenge Response" or "Valet Start Challenge Response" or "Reset Challenge Response")

- The correct type of password is used in the correct mode. A backup password shall be treated as invalid when OpCode is “Valet Start Challenge Response” per REQ-267202. Likewise, an Enhanced Valet password, as defined by REQ-264929, shall be treated as invalid when OpCode is either “Valet Start Challenge Response” or “Reset Challenge Response”)

4.6.2.17 LBIv1-REQ-267202/A-Accept Valet Password Only when in Enhanced Valet Mode

When the status of Enhanced Valet Mode in the LBIServer is active and the LBIServer receives *BackupIgnition_Rq(OpCode = “Valet Start Challenge Response”, KeyIndex = EOS, Password = Challenge Password, KeypadCode = EOS*, the LBIServer shall only check the received password hash against the valet password hash.as described in REQ-260082.

Note: This means that when attempting to start the vehicle while in Enhanced Valet mode, only valid valet passwords will be accepted. Valid backup passwords will not be accepted. To exit Enhanced Valet Mode, the user must start the vehicle via valet password, key fob or PaaK device and then select to exit from the menu.

4.6.2.18 LBIv1-REQ-260083/A-Acknowledge of Valid Password Entry to LBIClient

If LBIServer determines password is valid per REQ-268538, it shall notify the LBIClient of this via BackupIgnition_Rsp with encoding values set as:

RespCode=

- ”Challenge Response *Acknowledge,*” when OpCode is “Challenge Response “or
- “Valet Start Challenge Response Acknowledge” when OpCode is “Valet Start Challenge Response“ or
- “Reset Challenge Response Acknowledge when OpCode is “Reset Challenge Response“

RspStatus=”Valid Password”
VariableData shall all set to zero

4.6.2.19 LBIv1-REQ-275628/A-Start Conditions for T_Backup Ignition Timer

The LBServer shall start the backup password counter, T_Backup Ignition Timer when either condition below is met:

- After an entered password is determined as valid per REQ- 268538 with IgnitionStatus_St is not in Run and OpCode is “Challenge Response “

or

- After an entered password is determined as valid per REQ- 268538 with OpCode is “Valet Start Challenge Response“

Note: The first condition is the case of staring vehicle with a backup password and the second one is for stating vehicle with an Enhance Valet password. In other use cases such as resetting backup password, creating Enhanced Valet Password and deleting Valet Password, there is no need to start T_Backup Ignition Timer after a password is determined valid.

4.6.2.20 LBIv1-TMR-REQ-268518/A-T_Backup Ignition Timer

Name	Description	Units	Range	Resolution	Default
T_Backup Ignition Timer	Maximum time the LBIServer shall response positively to a Paak Crypto Start Search after a password is determined valid to start a vehicle. Note: Use the default value	sec	15-25	1	21

4.6.2.21 LBIv1-REQ-268539/A-Reset Conditions for T_Backup Ignition Timer

The LBIServer shall reset T_Backup Ignition Timer to zero each time a password is successfully used to start the vehicle, indicated by IgnitionStatus_St transiting from not in Run status to Run status.

4.6.2.22 LBIv1-REQ-260084/C-HMI Display of Valid Entry and Start Engine Instruction

When the Engine is not running as indicated in PwPckTq_St(PwPckOff_TqNotAvailable), upon receiving a valid password notification, BackupIgnition_Rsp (RespCode =" Challenge Response *Acknowledge*" or "Valet Start Challenge Response *Acknowledge"*, RspStatus =" Valid Password"), the LBIClient HMI shall notify the user that the entered password has been accepted and that they can now press the brake and start button in order to start the vehicle

This HMI message shall be dismissed when any one of conditions occurs:

The T_Push to Start Timer expires

The vehicle Engine status changes to PwPck_St(PwPckOn_TqAvailable)

The vehicle Engine status changes to PwPckTq_St(PwPckOn_TqNotAvailable)

Note:
PwPckOff_TqNotAvailable→ PwPckOn_TqAvailable implies that the user follows the HMI Instruction and successfully starts the engine

PwPckOff_TqNotAvailable→ PwPckOn_TqNotAvailable implies that something occurred to cause the Engine to change to NonMotive mode (e.g. Remote Start occurred) thus the HMI instructions are no longer valid and shall be dismissed

4.6.2.23 LBIv1-TMR-REQ-268540/A-T_Push to Start Timer

Name	Description	Units	Range	Resolution	Default
T_Push to Start Timer	Maximum time the LBIClient shall inform the user that the entered password has been accepted and they must start the vehicle within this time period. Note: Use the default value	sec	15-25	1	20

4.6.2.24 LBIv1-REQ-268766/A-Reset Condition for T_Push to Start Timer

The T_Push to Start Timer shall be reset to zero each time a backup password is successfully used to start the vehicle as indicated in IgnitionStatus_St(Run).

4.6.2.25 LBIv1-REQ-271254/B-Positive Response to Crypto Searches

Once the T_Backup Ignition Timer is started per LBIv1-REQ-275628 and before the T_Backup Ignition Timer expires or is reset to zero by REQ-268539, the LBIServer shall respond negatively to Registry or Polling searches but positively to the LBIServer2 Crypto Start searches.

The LBIServer2 Crypto Start search is sent via the following signals by the LBIServer2:

- PaakTargetPassiveData ([Challenge])
- PaakTargetSearchZone (Interior)
- PaakTargetSearchType (CRYPTO)
- PaakTargetRollCode = ([RollCode])

Uopn receiving the signals listed above, the LBIServer shall respond positively via the signal PaakCtrlSearchRslt (Valid) in parallel with other 4 signals as below:

- PaakCtrlSearchRslt (Valid)
- PaakCtrlPassiveData ([Response])
    - The calculation of PaakCtrlPassiveData ([Response]) shall be same as Paak Passive Start per PaaK-REQ-270039-Passive Challenge Data Response
- PaakCtrlRollCode ([RollCode])
- PaakCtrlFoundRollCode ([RollCode])
- PaakCtrlFoundIndex1([Index])

- The index shall be the key index of the PaaK device that the entered password is associated with
- The rest of 7 index signals (PaakCtrlFoundIndex[2-8]) shall be set to 0x0

Once T_Backup Ignition Timer expires or is reset to zero, the LBIServer shall respond negatively, PaakCtrlSearchRslt(Invalid), to the LBIServer2 Crypto Start searches.

Note: For LBI starting the vehicle use case, BLI uses the PaaK Passive Start framework. Once the LBI password is authenticated and within the time period defined by T_Backup Ignition Timer, the LBIServer responds to the LBIServer2 with the same manner as it responds to Paak Passive Start. The only difference that instead of providing the PaaK Index inside the car in Paak Passive Start use case, the LBIServer provides the key index of the PaaK device that the entered LBI password is associated with.

4.6.2.26 PaaK-REQ-270039/C-Passive Challenge Data Response

The BLEM shall transmit a single passive command per challenge data in PaakTargetPassiveData when PaakTargetCmd(Idle). The BLEM shall use the PaakSearchZone and PaakSearchType using the operation (PaakSearchZone | (PaakSearchType << 4)) to create byte14 of the AES input. The BLEM shall transmit the PaakLocalization_St

The BLEM shall use the TargetRollCode using the operation (TargetRollCode | (TargetRollCode << 4)) to create byte15 of the AES input.

PaakSearchZone is the data the BLEM received from the BCM in PaakTargetSearchZone.
PaakSearchType is the data the BLEM received from the BCM in PaakTargetSearchType.
TargetRollCode is the data the BLEM received from the BCM in PaakTargetRollCode.

The BLEM shall perform the AES calculations as described in PaaK-REQ-269555-BCM-BLEM Communication AES Encryption.

The BLEM shall transmit bytes [0, 4, 9, 10, 13] of the AES calculation output in PaakCtrlPassiveData if there is an authorized mobile device in the requested search zone.

The BLEM shall transmit bytes [1, 2, 6, 8, 11] of the AES calculation output in PaakCtrlPassiveData if there is no authorized mobile device in the requested search zone.

4.6.2.27 LBIv1-REQ-277457/A-HMI Display of Active Enhanced Valet Mode for Starting Vehicle

Once the vehicle started with the Enhanced Valet password, the LBIClient shall display a Enhanced Valet Mode screen until the Center Stack HMI is suspended or the vehicle exits Enhanced Valet Mode.

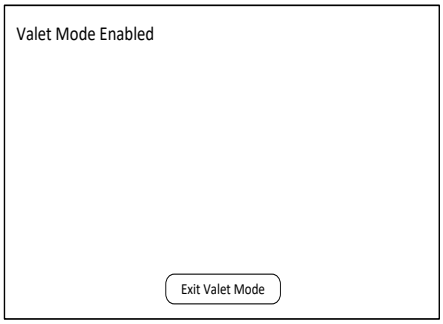
The contents of Enhanced Valet Mode screen shall include but not limit to

- Active Status of Enhanced Valet Mode
- The Exit Valet Mode option menu

The Enhanced Valet password shall not be displayed here per REQ-260114.

The Exit Valet Mode option menu shall comply with REQ- 264925

Below is an example of Enhanced Valet Mode screen. Please note that this example only serves as a design aid and does not necessarily represent the final implementation.



4.6.2.28 LBIv1-REQ-270547/C-Notification of Valid Backup Password Usage to LBIClient2

When the LBIServer responds positively to the LBIServer2 Crypto Start search during the ignition authorization period as defined by T_Backup Ignition Timer, it shall report this event, including the corresponding device KeyID, to the LBIClient2 via LBIAlert_St (Event = “Backup Password Used” or “Valet Password Used”, Source = Reserved)

The LBIAlert_St (Event) shall be set as “Backup Password Used” when an entered password is determined as valid with IgnitionStatus_St is not in Run and OpCode of “Challenge Response “.

The LBIAlert_St (Event) shall be set as “Valet Password Used “when an entered password is determined as valid with OpCode of “Valet Start Challenge Response”.

The corresponding device KeyID shall be either:

- The Key IDs of the PaaK device(s) that were in the vehicle at the time of valet password generation, if Enhanced Valet Mode was authorized with PaaK device(s)

  OR

- The Key ID of the PaaK device associated with the entered backup password, if Enhanced Valet mode was authorized by a backup password

4.6.2.29 LBIv1-REQ-264866/B-Track Invalid Password Entering Attempts

The LBIServer shall keep track of invalid attempts at entering the backup password and invalid attempts at entering the valet password in separate counters, N_NumberOfInvalidBackupAttempts and N_NumberOfInvalidValetAttempts.

Each time when the LBIServer determines a password is invalid, it shall increment invalid password counter N_NumberOfInvalidBackupAttempts for the followings use cases:

- Resetting a backup password (Option 1) with OpCode=” Reset Challenge Response” in LBIv1-REQ- 271294
- Activating Enhanced Valet Password (Option 2) with OpCode=” Challenge Response”
- Deactivating Enhanced Valet Password (Option 2) with OpCode=” Valet Delete Challenge Response” in LBIv1-REQ-260125
- Starting the vehicle when the vehicle is not in Enhanced Valet Mode with OpCode=” Challenge Response
- Exiting Secure Idle with OpCode=”Challenge Response”

Each time when the LBIServer determines a password is invalid (per LBIv1-REQ-268538), when starting vehicle in Enhanced Valet Mode (OpCode=” Valet Start Challenge Response”) it shall increment invalid password counter N_NumberOfInvalidValetAttempts

4.6.2.30 LBIv1-REQ-268511/A-N_NumberOfInvalidBackupAttempts

Name	Description	Units	Range	Resolution	Default
N_NumberOfInvalidBackupAttemps	N_NumberOfInvalidBackupAttemps is the counter for LBIServer to track the invalid backup attempts. When N_NumberOfInvalidBackupAttemps reaches to the default value, LBIServers then starts the lockout timer, T_LBILockout.		0-10	1	5

The counter shall be stored in NVM and shall not be affected by change in vehicle ignition state, network status, or battery state of charge.

4.6.2.31 LBIv1-REQ-268516/A-N_NumberOfInvalidValetAttempts

Name	Description	Units	Range	Resolution	Default
N_NumberOfInvalidValetAttemps	N_NumberOfInvalidValetAttemps is the counter for LBIServer to track the invalid Enhanced Valet attempts. When N_NumberOfInvalidValetAttemps reaches to the default value, LBIServers then starts the lockout timer, T_LBILockout.		0-10	1	5

4.6.2.32 LBIv1-REQ-270546/A-Storage of of Invalid Counters

N_NumberOfInvalidBackupAttempts and N_NumberOfInvalidValetAttempts shall be stored in NVM and shall not be reset nor lost by change in vehicle ignition state, network status, or battery state of charge.

4.6.2.33 LBIv1-REQ-264867/A-Reset Conditions for Invalid Counters

The backup password invalid counter, N_NumberOfInvalidBackupAttempts, shall be reset to zero each time a backup password is successfully used to start the vehicle as indicated in IgnitionStatus_St(Run).

The valet password invalid counter, N_NumberOfInvalidValetAttempts, shall be reset to zero each time a valet password is successfully used to start the vehicle indicated as in IgnitionStatus_St(Run) or the vehicle exits Enhanced Valet Mode

4.6.2.34 LBIv1-REQ-260085/A-Acknowledge of Invalid Password Entry for Starting Vehicle to LBIClient

When receiving BackupIgnition_Rq(OpCode="Challenge Response" or "Valet Start Challenge Response"), If LBIServer determines a password is invalid per REQ-268538, it should notify the LBIClient of this via BackupIgnition_Rsp(RespCode="Challenge Response *Acknowledge*", or" Valet Start Challenge Response *Acknowledge*", RspStatus='Invalid Password").

4.6.2.35 LBIv1-REQ-264865/A-HMI Display of Invalid Entry and Retry

Once receiving an invalid password entry notification via BackupIgnition_Rsp(RspStatus="Invalid Password") from the LBIServer, the LBIClient HMI shall notify the user that the entered password is invalid and shall provide the user with option to retry.

The use cases that the LBIClient shall comply with are listed below.

- Reset a backup password when BackupIgnition_Rsp(RespCode) is "Reset Challenge Response *Acknowledge*"
- Activate Enhanced Valet Password when BackupIgnition_Rsp(RespCode) is "Valet Create Challenge Response *Acknowledge*"
- Deactivate Enhanced Valet Password when BackupIgnition_Rsp(RespCode) is "Valet Delete Challenge Response *Acknowledge*"
- Start the vehicle with a backup password when BackupIgnition_Rsp(RespCode) is "Challenge Response *Acknowledge*"

- Start the vehicle with an Enhanced Valet password when BackupIgnition_Rsp(RespCode) is ”Valet Start Challenge Response *Acknowledge*

- Exit Secure Idle when BackupIgnition_Rsp(RespCode) is ”Challenge Response Acknowledge”

4.6.2.36 LBIv1-REQ-268242/A-Criteria of Backup Ingition Lockout

The LBIServer shall lockout LBI feature when either one of the invalid counters, either N_NumberOfInvalidBackupAttempts or N_NumberOfInvalidValetAttempts, reaches five invalid attempts.

4.6.2.37 LBIv1-REQ-268502/B-Response of Lockout

Once the LBIServer determines to lock out LBI feature per LBIv1-REQ-268242, it shall execute the following tasks:

- Start the T_LBILockout Timer

- Notify the LBIClient2 about that status of lockout according LBIv1-REQ-271255

- Set LBILockout_St to “Active” until T-Lockout Timer expires

- Ignore any request from the LBIClient via *BackupIgnition_Rsp* until T-Lockout Timer expires

- Acknowledge to the LBIClient about that status of lockout, not invalid password even though it is the 5th invalid attempt, via BackupIgnition_Rsp (RspStatus=” Lockout”) for the following use cases with its proper RspCode for each use case:

    - Reset a backup password by setting BackupIgnition_Rsp (RespCode) to” Reset Challenge Response *Acknowledge*”

    - Activate Enhanced Valet Password by setting BackupIgnition_Rsp (RespCode) to” Valet Create Challenge Response *Acknowledge*”

    - Deactivate Enhanced Valet Password by setting BackupIgnition_Rsp(RespCode) to ”Valet Delete Challenge Response *Acknowledge*”

    - Start the vehicle with a backup password by setting BackupIgnition_Rsp (RespCode) to” Challenge Response *Acknowledge*”

    - Start the vehicle with an Enhanced Valet password by setting BackupIgnition_Rsp (RespCode) to” Valet Start Challenge Response *Acknowledge*

    - Exit Secure Idle by setting BackupIgnition_Rsp (RespCode) to” Challenge Response Acknowledge

4.6.2.38 LBIv1-TMR-REQ-268500/A-T_LBILockout Timer

Name	Description	Units	Range	Resolution	Default
T_LBILockout Timer	Maximum time the LBIServer shall lock out the user to enter a password again after 5 invalid attempts with backup password or Enhanced Valet password. Note: Use the default value	sec	180-600	60	300

4.6.2.39 LBIv1-REQ-270545/A-Storage of T_LBILockout Timer

The LBIServer shall store the T_LBILockout Timer in NVM and shall not reset it for the change of vehicle ignition state, network status or battery charge state.

4.6.2.40 LBIv1-REQ-264868/A-HMI Display of LBI Lockout

The LBIClient shall display a lockout notification whenever either one of the following conditions is met:

- When detects PasswordEntryScreen_Rq changs from Inactive to Active while LBILockout_St is Active.
- When receives BackupIgnition_Rsp(RspStatus="Lockout") for the following use cases:
    - Reset a backup password when BackupIgnition_Rsp(RespCode) is "Reset Challenge Response *Acknowledge*"
    - Activate Enhanced Valet Password when BackupIgnition_Rsp(RespCode) is "Valet Create Challenge Response *Acknowledge*"
    - Deactivate Enhanced Valet Password when BackupIgnition_Rsp(RespCode) is "Valet Delete Challenge Response *Acknowledge*"
    - Start the vehicle with a backup password when BackupIgnition_Rsp(RespCode) is "Challenge Response *Acknowledge*"
    - Start the vehicle with an Enhanced Valet password when BackupIgnition_Rsp(RespCode) is "Valet Start Challenge Response *Acknowledge*
    - Exit Secure Idle when BackupIgnition_Rsp(RespCode) is "Challenge Response Acknowledge"

4.6.2.41 LBIv1-REQ-266972/A-Dismiss HMI LBI Lockout Display

The LBIClient shall dismiss the lockout display when any one of events occurs first:

- The Lockout notification assigned Type-timer expires (for example, timer for Type D, Type B etc.) as called out by H31M-Backup Ignition
- When LBILockout_St changes from Active to Inactive.

4.6.2.42 LBIv1-REQ-264869/A-No Exception for LBI Lockout

LBI lockout shall be set as required by REQ-268242 without any exception.

4.6.2.43 LBIv1-REQ-271255/B-Notification of Lockout to LBIClient2

After starting the T_LBILockout Timer**,** the LBIServer shall report the lockout event to the LBIClient2 via LBIAlert_St (Event =" Lockout" or" Lockout in Valet Mode," Source = Reserved)

- The LBIServer shall set LBIAlert_St (Event) to "Lockout" when the vehicle is not in Enhanced Valet Mode as defined by LBIv1-REQ-270560.
- The LBIServer shall set LBIAlert_St (Event) to "Lockout in Valet Mode" when the vehicle is in Enhanced Valet Mode as defined by LBIv1-REQ-270560.
- Since a correct KeyID cannot be determined in the locked out case, the LBIServer shall not transmit any KeyID via LBIAlert_St.

Note: when the LBIServer does not transmit any device KeyID, it is interpreted by SDN as a notify all notification. The Notification will be sent out to all registered Paaks.

4.6.2.44 LBIv1-REQ-270560/A-Indication of Vehicle in Enhanced Valet Mode

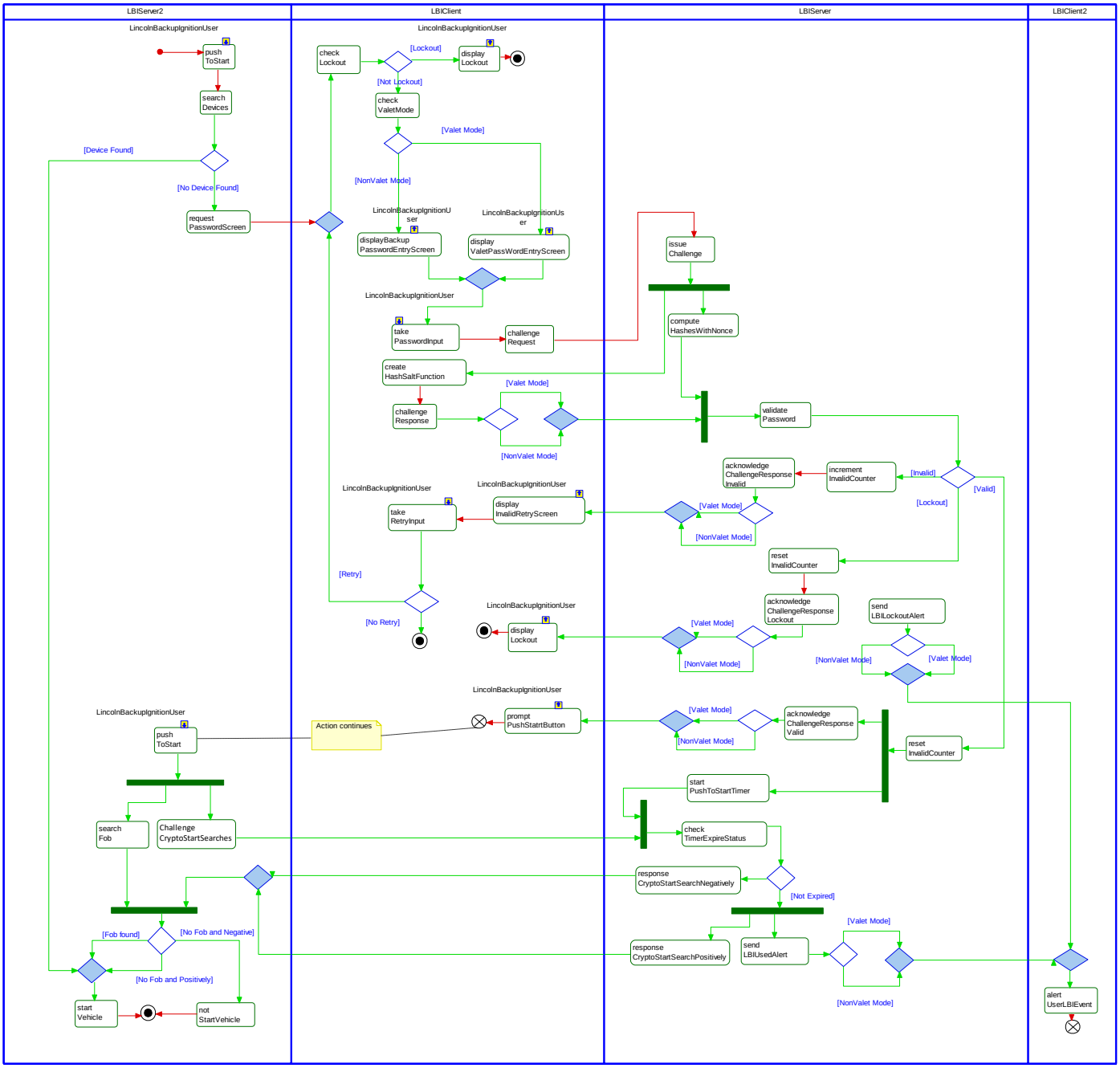
Whenever an active password entry is associated with Valet Password KeyIndex as defined in REQ-270053, the LBIServer shall recognize this as vehicle being in Enhanced Valet Mode.

White Box Views

4.6.2.45 Activity Diagrams

4.6.2.45.1 LBIv1-ACT-REQ-274278/A-Starting Vehicle with Backup or Valet Password

Activity Diagram



4.6.2.46 Sequence Diagrams

4.6.2.46.1 LBIv1-SD-REQ-262294/B-Starting Vehicle with LBI Password-Happy Path

Constraints

Pre-Condition

1. The LBI User has previously created a LBI backup password or Enhanced Valet password
2. No associated key fobs or phones-as-keys are near the vehicle

3. The vehicle Ignition is not in Run
4. The vehicle is not in LBI lockout stage

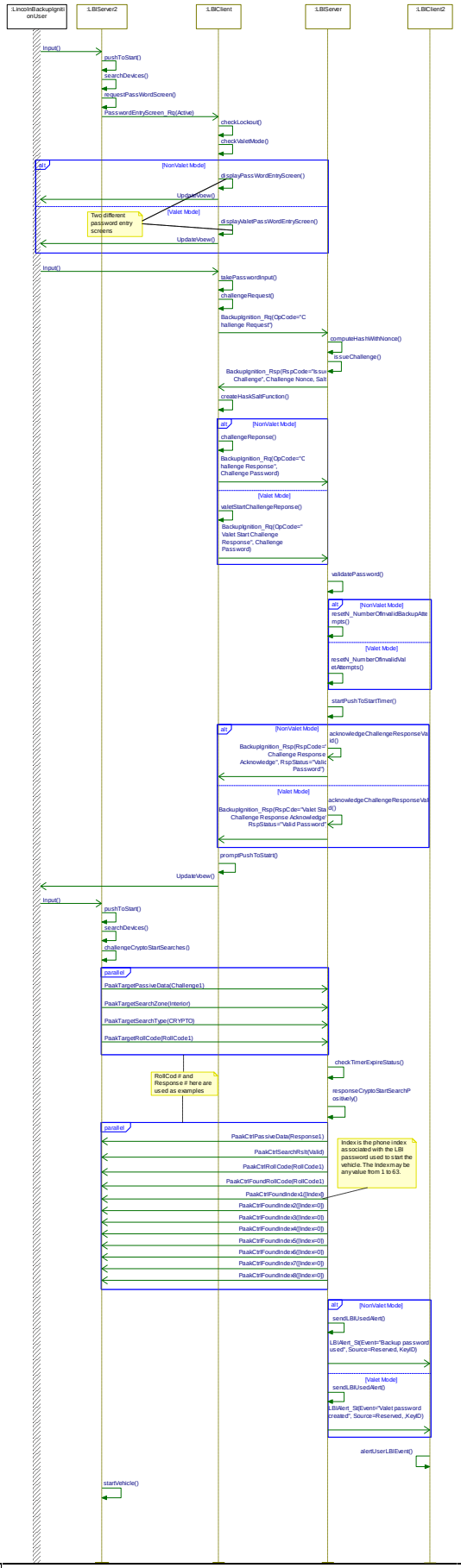
Scenarios

Normal Usage

Post-Condition

1. The LBI User is able to drive away the vehicle
2. The LBI User is able to charge their phone-as-a-key in the vehicle
3. The notification that a backup or Enhanced Valet password has been used is sent to the LBI User via e-mail and a phone message to the PaaK whose backup password is just used or to ALL PaaK devices when an Enhanced Valet Password is used

Sequence Diagram


:LincolnBackupIgnitionUser
:LBIServer2
:LBIClient
:LBIServer
:LBIClient2
Input()
pushToStart()
searchDevices()
requestPassWordScreen()
PasswordEntryScreen_Rq(Active)
checkLockout()
checkValetMode()
alt
[NonValet Mode]
displayPassWordEntryScreen()
UpdateView()
[Valet Mode]
displayValetPassWordEntryScreen()
UpdateView()
Two different password entry screens
Input()
taletPasswordInput()
challengeRequest()
BackupIgnition_Rq(OpCode="Challenge Request")
computeHashWithNonce()
issueChallenge()
BackupIgnition_Rsp(RspCode="Issue Challenge", Challenge Nonce, Salt)
createHaskSaltFunction()
alt
[NonValet Mode]
challengeReponse()
BackupIgnition_Rq(OpCode="Challenge Response", Challenge Password)
[Valet Mode]
valetStartChallengeReponse()
BackupIgnition_Rq(OpCode="Valet Start Challenge Response", Challenge Password)
validatePassword()
alt
[NonValet Mode]
resetN_NumberOfInvalidBackupAttempts()
[Valet Mode]
resetN_NumberOfInvalidValetAttempts()
startPushToStartTimer()
alt
[NonValet Mode]
acknowledgeChallengeResponseValid()
BackupIgnition_Rsp(RspCode=Challenge Response Acknowledge", RspStatus="Valid Password")
[Valet Mode]
acknowledgeChallengeResponseValid()
BackupIgnition_Rsp(RspCde="Valet Start Challenge Response Acknowledge", RspStatus="Valid Password")
promptPushToStart()
UpdateView()
Input()
pushToStart()
searchDevices()
challengeCryptoStartSearches()
parallel
PaakTargetPassiveData(Challenge1)
PaakTargetSearchZone(Interior)
PaakTargetSearchType(CRYPTO)
PaakTargetRollCode(RollCode1)
RollCod # and Response # here are used as examples
checkTimerExpireStatus()
responseCryptoStartSearchPositively()
parallel
PaakCtrlPassiveData(Response1)
PaakCtrlSearchRslt(Valid)
PaakCtrlRollCode(RollCode1)
PaakCtrlFoundRollCode(RollCode1)
PaakCtrlFoundIndex1([Index])
PaakCtrlFoundIndex2([Index=0])
PaakCtrlFoundIndex3([Index=0])
PaakCtrlFoundIndex4([Index=0])
PaakCtrlFoundIndex5([Index=0])
PaakCtrlFoundIndex6([Index=0])
PaakCtrlFoundIndex7([Index=0])
PaakCtrlFoundIndex8([Index=0])
Index is the phone index associated with the LBI password used to start the vehicle. The Index may be any value from 1 to 63.
alt
[NonValet Mode]
sendLBIUsedAlert()
LBIAlert_St(Event="Backup password used", Source=Reserved, KeyID)
[Valet Mode]
sendLBIUsedAlert()
LBIAlert_St(Event="Valet password created", Source=Reserved, KeyID)
alertUserLBIEvent()
startVehicle()

4.6.2.46.2 LBIv1-SD-REQ-275578/A-Starting Vehicle with LBI Password-Invalid Entry and Lockout Paths

Constraints

Pre-Condition

1. The LBI User has previously created a LBI backup password or Enhanced Valet password
2. No associated key fobs or phones-as-keys are near the vehicle
3. The vehicle Ignition is not in Run
4. The vehicle is not in LBI lockout stage

Scenarios

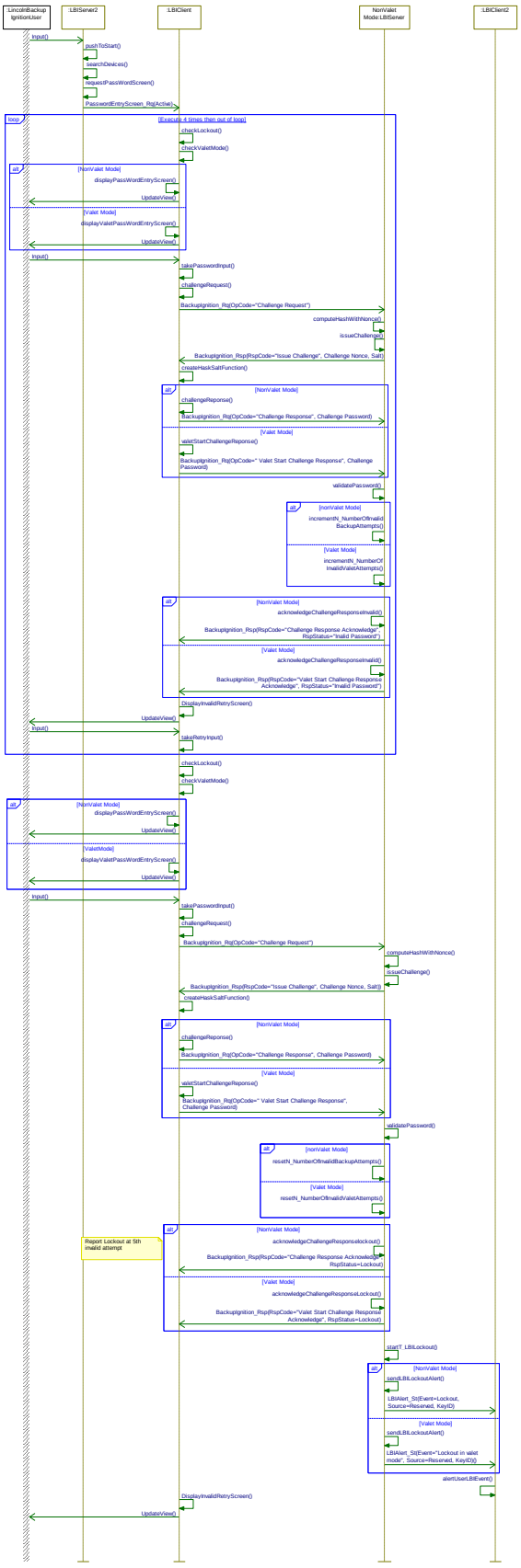
Normal Usage

Invalid Password are entered five times

Post-Condition

The vehicle is locked out and vehicle cannot be stated

Sequence Diagram



4.7 LBIv1-FUN-REQ-258453/A-Deleting Valet Password and Keypad Code

4.7.1 Use Cases

4.7.1.1 LBIv1-UC-REQ-260132/A-Deleting Valet Password and Keypad Code with Device

Actors	The LBI User
Pre-conditions	1. The LBI User has previously activated PaaK feature for the vehicle 2. The vehicle ignition Status is in Run 3. The vehicle transmission is in Park 4. The vehicle is not locked out by LBI Feature 5. The vehicle is in Enhanced Valet Mode 6. The LBI User is inside vehicle 7. At least one PaaK or one keyFob is inside the vehicle
Scenario Description	The LBI User presses Exit button from Enhanced Valet Mode Screen
Post-conditions	1. The Vehicle is no longer in Enhanced Valet Mode. 2. The Enhanced Valet Password is deleted and it no longer can start the vehicle 3. A notification that the Enhanced Valet password has been deleted with or without Enhanced Valet keypad code being deleted is sent to the LBI User.
List of Exception Use Cases	The LBIServer is unable to delete Enhanced Valet Password
Interfaces	The LBIClient The LBIClient2 The LBIServer The LBIServer2 SDN PaakFI

4.7.1.2 LBIv1-UC-REQ-277213/A-Deleting Valet Password and Keypad Code without Device

Actors	The LBI User
Pre-conditions	1. The LBI User has previously activated PaaK feature for the vehicle 2. The vehicle ignition Status is in Run 3. The vehicle transmission is in Park 4. The vehicle is not locked out by LBI Feature 5. The vehicle is in Enhanced Valet Mode 6. The LBI User is inside vehicle 7. No PaaK and no keyFob is inside the vehicle
Scenario Description	1. The LBI User presses Exit button from Enhanced Valet Mode Screen 2. The LBI HMI displays backup password entry screen 3. The LBI User enters a valid backup password
Post-conditions	1. The Vehicle is no longer in Enhanced Valet Mode. 2. The Enhanced Valet Password is deleted and it no longer can start the vehicle 3. A notification that the Enhanced Valet password has been deleted with or without Enhanced Valet keypad code being deleted is sent to the LBU User.

List of Exception Use Cases	The LBIServer is unable to delete Enhanced Valet Password
Interfaces	The LBIClient The LBIClient2 The LBIServer The LBIServer2 SDN PaakFI

4.7.2 Requirements

4.7.2.1 LBIv1-REQ-260119/A-Query for PaaKs to Exit Valet Mode

Once the user selects the option to exit valet mode, LBIClient shall query the LBIServer for PaaK devices in the vehicle via BackupIgnition_Rq(OpCode="Check for Keys to Exit Valet Mode", KeyIndex=0x0, Password = EOS, KeypadCode = EOS).

4.7.2.2 LBIv1-REQ-267214/B-Trigger Interior Registry KeyFob Search for Exit Valet Mode

Upon receiving the request of BackupIgnition_Rq (OpCode=" Check for Keys to Exit Valet Mode"), the LBIServer shall conduct a search of PaaK devices with passwords in the vehicle. Regardless of query results, the LBIServer shall then trigger a LBIServer2 Interior Registry KeyFob search to determine if a KeyFob is found inside the vehicle by executing LBIv1-FUN-REQ-302285.

4.7.2.3 LBIv1-REQ-260131/A-Response of No Devices Status for Exit Valet Mode

In response of *BackupIgnition_Rq(OpCode="Check for Keys to Exit Valet Mode"), a*fter receiving Interior Registry KeyFob search results via FobCtrlSearchRslt, FobCtrlPassiveData and FobCtrlRollCode. In the case of no required device found, the LBIServer shall response back to the LBIClient with the status of Paak devices and keyfob via *BackupIgnition_Rsp* with

RspCode ="Check for Keys to Exit Valet Mode"
RspStatus ="No PaaK and No Fob In Vehicle"
VariableData shall set to Zero

4.7.2.4 LBIv1-REQ-260121/A-HMI Display Backup Password Entry Screen for No Device Found

In process of checking for Keys to Exit Valet Mode, if neither a PaaK device nor a key fob is detected in the vehicle, the LBIClient HMI shall display the backup password entry screen and ask the user to enter a backup password.

The status of no device found is indicated in *BackupIgnition_Rsp* with
RspCode ="Check for Keys to Exit Valet Mode"
RspStatus ="No PaaK and No Fob In Vehicle"

4.7.2.5 LBIv1-REQ-270384/B-Challenge Request

When the user enters a password at the password entry screen, the LBIClient shall request a challenge from LBIServer via BackupIgnition_Rsp(OpCode="Challenge Request", KeyIndex=0, *Password = EOS KeypadCode = EOS*).

The use cases of this Requirement are:

- Starting a vehicle with a backup password or an Enhanced Valet password
- Resetting a backup password
- Generating an Enhanced Valet Password with no device
- Deleting an Enhanced Valet password with no device

4.7.2.6 LBIv1-REQ-260077/A-Issue Cryptographic Nonce and Salt Challenge

After receiving a request for a challenge from LBIClient via BackupIgnition_Rq (OpCode="Challenge Request"), the LBIServer shall issue a challenge to the LBIClient via BackupIgnition_Rsp with cryptographic nonce and salt.

BackupIgnition_Rsp shall be set as
RspCode="Issue challenge"
RspStatus= *Reserved*
NumberOfItems= 0x00
Challenge Nonce = Challenge Nonce
Salt = Salt

4.7.2.7 LBIv1-REQ-270244/A-Compute Hash for All Stored Passwords

After receiving a request for a challenge from the LBIClient via BackupIgnition_Rq (OpCode="Challenge Request"), in addition to issue challenge to the LBIClient as described in REQ-260077, the LBIServer shall also compute, using the cryptographic nonce, another hash of all stored password hashes.

4.7.2.8 LBIv1-REQ-260080/A-Compute Hash for Entered Password

Once receiving BackupIgnition_Rsp(RspCode="Issue challenge"), the LBIClient shall compute a hash of entered password using received salt and then compute a hash of this result using received nonce.

4.7.2.9 LBIv1-REQ-260123/A-Response to Challenge with Hashed Backup Password

In the response of exiting Valet Mode, the LBIClient shall respond to the challenge of *BackupIgnition_Rsp(RspStatus="Issue Challenge"*) with computed password hash via *BackupIgnition_Rq(OpCode="Valet Delete Challenge Response", KeyIndex = 0x00, Password = Challenge Password, KeypadCode = EOS*).

4.7.2.10 LBIv1-REQ-260124/A-Compare Hashed and Computed Password for Exit Valet Mode

When the LBIServer receives a challenge hash via *BackupIgnition_Rq(OpCode="Valet Delete Challenge Response", KeyIndex = 0x00, Password = Challenge Password, KeypadCode = EOS*), it shall compare it with the hashes that it computed for the stored passwords.

4.7.2.11 LBIv1-REQ-260125/A-Incremental of Invaled Password Account for Exit Valet Mode

When receiving BackupIgnition_Rq(OpCode="Valet Delete Challenge Response", KeyIndex = 0x00, Password = Challenge Password, KeypadCode = EOS), If the LBIServer determines that the received password is invalid i.e. challenge hash does not match a calculated password hash, it shall increment invalid Valet password counter N_NumberOfInvalidValetAttempts.

4.7.2.12 LBIv1-REQ-268538/A-Criteria of Valid Password

A valid password is defined when both conditions are met:

- The computed hashes, as required by REQ-270244, matches with that of received hashed password via BackupIgnition_Rq(OpCode="Challenge Response" or "Valet Start Challenge Response" or "Reset Challenge Response")

- The correct type of password is used in the correct mode. A backup password shall be treated as invalid when OpCode is "Valet Start Challenge Response" per REQ-267202. Likewise, an Enhanced Valet password, as defined by REQ-264929, shall be treated as invalid when OpCode is either "Valet Start Challenge Response" or "Reset Challenge Response")

4.7.2.13 LBIv1-REQ-260126/A-Response of Invalid Backup Password for Exit Valet Mode

If the LBIServer determines that the received password is invalid i.e. challenge hash does not match a calculated password hash, it shall notify the LBIClient via *BackupIgnition_Rsp(RspCode="Valet Delete Challenge Response Acknowledge", RspStatus="Invalid Password", VariableData=0x00.)*

4.7.2.14 LBIv1-REQ-260130/A-HMI Display of Invalid Password for Exit Valet mode

When receiving an invalid password notification via *BackupIgnition_Rsp(RspCode="Valet Delete Challenge Response Acknowledge", RspStatus="Invalid Password"),* the LBIClient HMI shall notify the user that the entered password is invalid and provide an option to retry.

4.7.2.15 LBIv1-REQ-264921/A-Delete Valet Password once Backup Password is Verified

When receiving BackupIgnition_Rq(OpCode="Valet Delete Challenge Response", KeyIndex = 0x00, Password = Challenge Password, KeypadCode = EOS), if the LBIServer determines that the received password is valid i.e. challenge hash matches a calculated password hash, it shall delete the valet password.

4.7.2.16 LBIv1-REQ-271460/A-Delete Velet Password with Device Present

In response of BackupIgnition_Rq(OpCode="Check for Keys to Exit Valet Mode"), the LBIServer shall delete the Enhanced Valet password and any key indexes that were stored at the time of Enhanced Valet Mode activation, if either a PaaK device or a keyfob is detected inside the vehicle.

4.7.2.17 LBIv1-REQ-264922/B-Notification of Valet Password Deletion to LBIClient2

When the LBIServer deletes a valet password hash, it shall report this event, including the corresponding device KeyIDs to the LBIClient2 via *LBIAlert_St with encoding values set as:*

Event =" Valet Password Deleted"
Source =

- *Password if BackupIgnition_Rq* (*OpCode=" Valet Delete Challenge Response Acknowledge") when a backup password is used to delete the Enhnaced Valet password*

or

- *Paak if BackupIgnition_Rq* (*OpCode=" Check for Keys to Exit Valet Mode") with a Paak is present when device search is conducted*

or

- *KeyFob if BackupIgnition_Rq* (*OpCode=" Check for Keys to Exit Valet Mode") with a KeyFob is present when device search is conducted*

or

- PaaK and Key Fob *if BackupIgnition_Rq* (*OpCode=" Check for Keys to Exit Valet Mode") with both PaaK and Keyfob are present when device search is conducted*

The corresponding device KeyIDs shall be the KeyIDs of the PaaK devices that were in the vehicle at the time of valet password generation, if Enhanced Valet Mode was authorized by a PaaK device or the KeyID of the PaaK device associated with the entered backup password, if Enhanced Valet mode was authorized by a backup password.

4.7.2.18 LBIv1-REQ-271461/A-Initiate Keypad Code Deletion for Exit Valet Mode

After deleting a valet password hash, the LBIServer shall send a request to the LBIServer2 to delete the Enhanced Valet keypad code by executing LBI-FUN-REQ-275727.

4.7.2.19 LBIv1-REQ-275754/A-Criteria of Reporting Successful Valet Password Deletion

The LBIServer shall report a successful password deletion if the conditions below are both met:

- A successful keypad code deletion confirmation via KeyPadCodeProg_St (Delete) is received
- The requested Enhanced Valet password is deleted from the LBIServer HSM

4.7.2.20 LBIv1-REQ-275755/A-Criteria of Reporting Successful Valet Password Deletion with Keypad Code not Deleted

The LBIServer shall report a successful password deletion if the conditions below are all met:

- A failed Notification is received via KeyPadCodeProg_St(ProgrammingFailure)
- The requested Enhanced Valet password is deleted from the LBIServer HSM

4.7.2.21 LBIv1-REQ-275756/A-Criteria of Reporting Failed Password Deletion

The LBIServer shall report a failed password deletion if any one of conditions below is met:

- The vehicle operation conditions defined by REQ-264925 are no longer met before the deletion process completes
- The requested Enhanced Valet password is unable to be deleted from the the LBIServer HSM

4.7.2.22 LBIv1-REQ-271522/A-Response of Valet Password Deletion to LBIClient

In response to BackupIgnition_Rq(OpCode="Check for Keys to Exit Valet Mode" or "Valet Delete Challenge Response Acknowledge"), after performing deleting Enhanced Valet password and receiving keypad deletion result, the LBIServer shall notify the LBIClient about the Enhanced Valet password and the Enhanced Valet keypad code deletion status via BackupIgnition_Rsp with encoding values set as

RspCode=

- "*Check for Keys to Exit Valet Mode*" if OpCode is "*Check for Keys to Exit Valet Mode*" or
- "Valet Delete Challenge Response Acknowledge" if OpCode is "Valet Delete Challenge Response Acknowledge"

‘

RspStatus =

- "Password Deleted Successfully" per REQ-275754 or
- "Password Deleted Successfully, but Keypad Code Deleted Failed" per REQ-275755 or
- "Password Deleted Failed" perREQ-275756

VariableData shall all set to zero

4.7.2.23 LBIv1-REQ-271614/A-HMI Display of Failed Valet Password Deletion

When receiving deletion failure response via BackupIgnition_*Rsp(RspCode="Check for Keys to Exit Valet Mode" or "*Valet Delete Challenge Response Acknowledge"*, RspStatus ="*Password Deleted Failed*"*), the LBIClient shall notify user of Valet Exit failure.

4.7.2.24 LBIv1-REQ-275757/B-HMI Display of Successful Valet Password Deletion

Upon receiving *BackupIgnition_Rsp* (*RspCode =" Check for Keys to Exit Valet Mode" or"* Valet Delete Challenge Response"*, RspStatus =" Password Deleted Successfully"*), the LBIClient shall notify the user of successful Valet Password Deletion.

In the case of a failed keypad code deletion as indicated in RspStatus being "Password Deleted Successfully but Keypad Code Deleted Failed", the LBIClient HMI shall notify the use about the successful Enhanced Valet password deletion and the failed Enhanced Valet Keypad Code deletion.

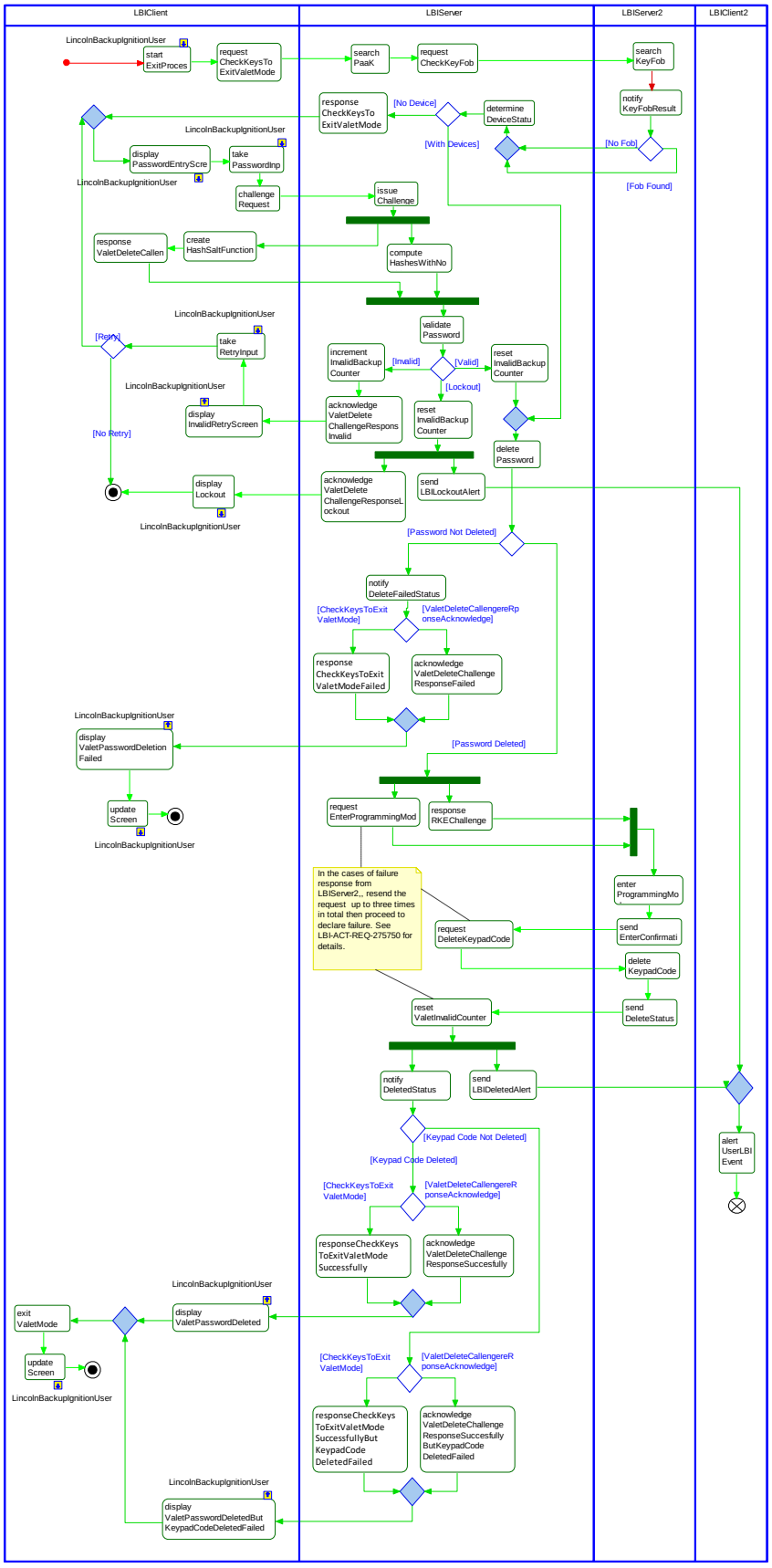
4.7.2.25 LBIv1-REQ-264932/A-Condition to Exit Enhanced Valet Mode

The LBIClient HMI shall exit Enhanced Valet Mode after displaying the Enhanced Valet password deletion status per REQ-275757.

White Box Views

4.7.2.26 Activity Diagrams

4.7.2.26.1 LBIv1-ACT-REQ-274283/A-Deleting Valet Password and Keypad Code Activity Diagram


LBIClient
LBIServer
LBIServer2
LBIClient2
LincolnBackupIgnitionUser
start ExitProces
request CheckKeysTo ExitValetMode
search PaaK
request CheckKeyFob
search KeyFob
notify KeyFobResult
response CheckKeysTo ExitValetMode
[No Device]
determine DeviceStatu
[With Devices]
[No Fob]
[Fob Found]
display PasswordEntryScre
take PasswordInp
challenge Request
issue Challenge
response ValetDeleteCallen
create HashSaltFunction
compute HashesWithNo
validate Password
[Retry]
take RetryInput
increment InvalidBackup Counter
[Invalid]
[Valid]
reset InvalidBackup Counter
[Lockout]
display InvalidRetryScreen
acknowledge ValetDelete ChallengeRespons Invalid
reset InvalidBackup Counter
delete Password
[No Retry]
display Lockout
acknowledge ValetDelete ChallengeResponseL ockout
send LBILockoutAlert
[Password Not Deleted]
notify DeleteFailedStatus
[CheckKeysToExit ValetMode]
[ValetDeleteCallengereRp onseAcknowledge]
response CheckKeysToExit ValetModeFailed
acknowledge ValetDeleteChallenge ResponseFailed
display ValetPasswordDeletion Failed
update Screen
[Password Deleted]
request EnterProgrammingMod
response RKEChallenge
In the cases of failure response from LBIServer2,, resend the request up to three times in total then proceed to declare failure. See LBI-ACT-REQ-275750 for details.
enter ProgrammingMo
request DeleteKeypadCode
send EnterConfirmati
delete KeypadCode
reset ValetInvalidCounter
send DeleteStatus
notify DeletedStatus
send LBIDeletedAlert
alert UserLBI Event
[Keypad Code Not Deleted]
[Keypad Code Deleted]
[CheckKeysToExit ValetMode]
[ValetDeleteCallengereR ponseAcknowledge]
responseCheckKeys ToExitValetMode Successfully
acknowledge ValetDeleteChallenge ResponseSuccesfully
exit ValetMode
display ValetPasswordDeleted
update Screen
responseCheckKeys ToExitValetMode SuccesfullyBut KeypadCode DeletedFailed
acknowledge ValetDeleteChallenge ResponseSuccesfully ButKeypadCode DeletedFailed
display ValetPasswordDeletedBut KeypadCodeDeletedFailed

4.7.2.27 Sequence Diagrams

4.7.2.27.1 LBIv1-SD-REQ-262301/A-Deleting Valet Password

Constraints

Pre-Condition

1. The LBI User has previously activated PaaK feature for the vehicle
2. The vehicle ignition Status is in Run
3. The vehicle transmission is in Park
4. The vehicle is not locked out by LBI Feature
5. The vehicle is in Enhanced Valet Mode
6. The LBI User is inside vehicle

Scenarios

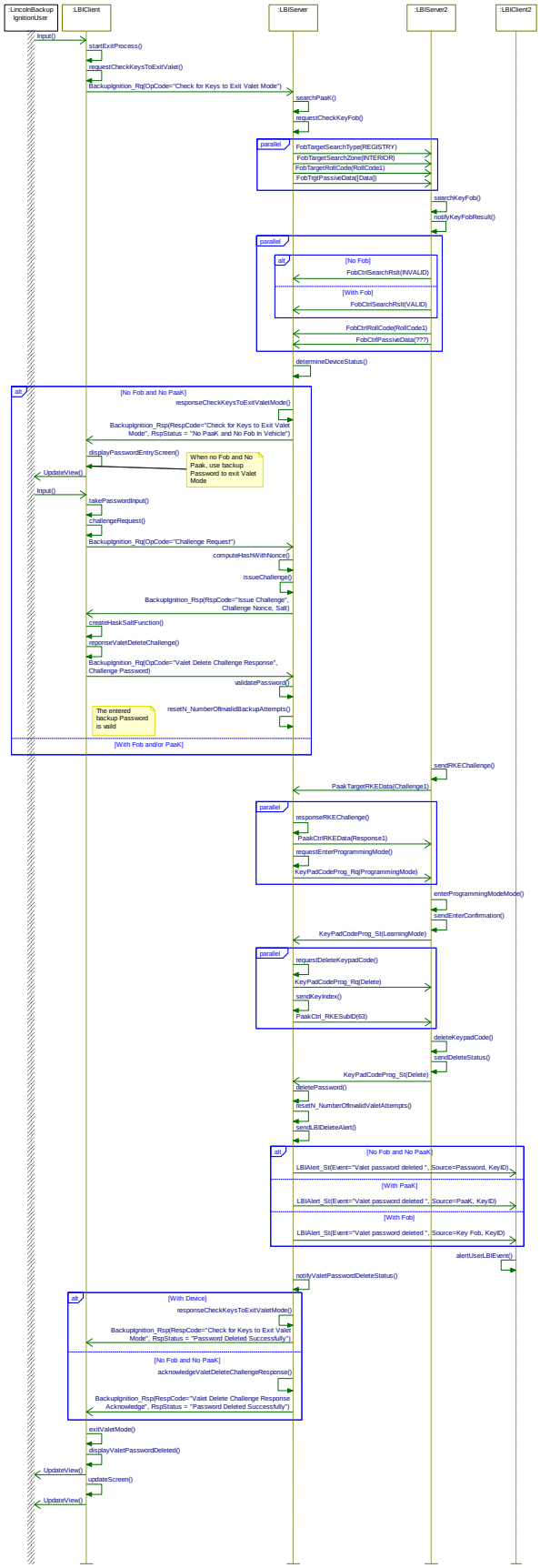
Normal Usage

The LBI User presses Exit button from Enhanced Valet Mode Screen

Post-Condition

1. The Vehicle is no longer in Enhanced Valet Mode.
2. The Enhanced Valet Password is deleted and it no longer can start the vehicle
3. A notification that the Enhanced Valet password has been deleted with or without Enhanced Valet keypad code being deleted is sent to the LBI User

Sequence Diagram



4.8 LBIv1-FUN-REQ-258454/A-Deleting All Backup Passwords via Master or PaaK Reset

4.8.1 Requirements

4.8.1.1 LBIv1-REQ-260133/A-HMI Display of Password Erase with Master Reset and Paak Reset

When the user selects Master Reset or PaaK reset through Embedded Modem Reset OnBoardClient, the Embedded Modem Reset OnBoardClient HMI shall inform the user that all PaaK Backup passwords and associated keypad codes will be erased. Once the user follows through with a Master or PaaK Reset.

4.8.1.2 LBIv1-REQ-270261/A-Erase Password from Memory

When receiving FactoryReset_Rq(ResetFactoryDefaults) for Master Reset or EmbeddedModem_Rq(Paak_Reset) for PaaK Reset, the LBIServer shall delete all LBI passwords hashes after all the Paaks are revoked.

4.8.1.3 LBIv1-REQ-270264/A-Initiate Delete All Keypad Codes Request for Master Reset and Paak Reset

After deleting all LBI passwords as required by REQ-270261, the LBIServer shall determine whether there is any keypad codes associated with these passwords.

If there is one or more associated keypad codes, the LBIServer shall initiate a delete keypad code request by executing LBI-FUN-REQ-275727.

4.8.1.4 LBIv1-REQ-276040/A-No LBI Passwords Deletion Notification after Master Reset or Paak Reset

When LBI Passwords are successfully deleted triggered by Master Reset or Paak Reset as defined in LBIv1-REQ- REQ-270261, the LBIServer shall not send SyncP signed message (Service Type 0x40/Sub-Service 0x0) to the LBIClient2.

4.9 LBIv1-FUN-REQ-258455/A-Deleting Backup Password via PaaK Revoke

4.9.1 Requirements

4.9.1.1 LBIv1-REQ-260135/A-Notification of Password Deletion when Revoke PaaK

The mobile app HMI shall inform the user that revoking the CAK for their device will also delete any associated backup password or keypad code.

4.9.1.2 LBIv1-REQ-260136/A-Initiate Password Deletion when Revoke PaaK

When the LBIServer receives a request to revoke the CAK for a specific PaaK device as defined in REQ-235568, it shall also delete the backup password hash associated with that device.

4.9.1.3 LBIv1-REQ-271433/B-Initiate Delete Keypad Code when CAK Revoked

When deleting a backup password as a result of a CAK revoke, the LBIServer shall determine whether there is a keypad code associated with that CAK. If there is an associated keypad code, the LBIServer shall initiate keypad code deletion by executing LBI-FUN-REQ-275727.

4.10 LBIv1-FUN-REQ-318631/A-Suspending LBI when CCS is off

4.10.1 Use Cases

4.10.1.1 LBIv1-UC-REQ-318652/A-HMI Display of CCS Impacts on LBI

Actors	The LBI User
Pre-conditions	1. The LBI User has previously created at least one LBI password and LBI Keypad code 2. The vehicle CCS menu is accessible
Scenario Description	1. The LBI User selects option to turn off CCS 2. The LBI HMI displays a screen with information about the CCS off impact on LBI

	3. The LBI Users is asked to confirm if continue to turn off CCS or cancel 4. The LBI User select continue to turn off CCS
Post-conditions	1. CCS is off as shown on CCS menu 2. LBI feature is suspended with its menu being grayed out at Center Stack Menu
List of Exception Use Cases	
Interfaces	The LBIClient

4.10.1.2 LBIv1-UC-REQ-318653/A-Starting Vehicle without KeyFob/PaaK when CCS off

Actors	The LBI User
Pre-conditions	1. The LBI User has previously created a backup password 2. The vehicle CCS has been off. 3. The LBI User is outside vehicle 4. No associated key fobs or phones-as-keys are near the vehicle 5. The vehicle ignition is not in Run
Scenario Description	1. The LBI User approaches the vehicle 2. The LBI User enters valid keypad code 3. The vehicle unlocks 4. The LBI User opens door and enters the vehicle 5. The LBI User presses brake pedal
Post-conditions	1. The center stack device screen will not display backup password entry screen 2. The LBI User is unable to start the engine with LBI password nor Enhanced Valet password
List of Exception Use Cases	
Interfaces	The LBIClient The LBIServer2

4.10.2 Requirements

4.10.2.1 LBIv1-REQ-318628/A-LBI HMI Display for CCS off Impacts on LBI

When the user selects to turn off CCS at Center Stack CCS menu, the LBIClient HMI shall display information about the Privacy Mode (when CCS is off) impact on LBI feature and shall solicit the user's confirmation of turning off CCS before proceed.

.

The display shall include but not limit to the followings:

- All LBI feature's functions, except LBI keypad code, will be suspended until CCS is turned on again
- The user no longer can use LBI Passwords nor the Enhanced Valet Password to start the vehicle

4.10.2.2 LBIv1-REQ-318630/B-Suspend LBI when CCS off

When CCS is turned off the LBIClient shall deactivate the LBI feature and update LBI HMI accordingly until CCS is turned on again. The LBIClient shall not process LBI CAN traffic received from the LBIServer and LBIServer2 while CCS is turned off.

Note: To suspend LBI feature, the LBIClient could simply blocks out all LBI access from the user by graying out the LBI main menu and also by blocking out the LBI password entry screen via implementing LBIv1-REQ-260076/B that blocks out the LBI password entry screen when CCS is off.

4.10.2.3 LBIv1-REQ-260076/B-Conditions to Display Password Entry Screen for Starting Vehicle

In response to PasswordEntryScreen_Rq(Active) the LBIClient shall check the following three conditions to determine whether to display the password entry screen and which entry screen to display:

4. Customer Connectivity Settings (CCS) status
5. LBI lockout status as indicated in LBILockout_St
6. Enhanced Valet Mode status as required by LBIv1-REQ-275620

- The LBIClient shall not display any password entry screen when either one condition listed below is met
  - CCS is turned off
    or
  - or the vehicle is under LBI lockout (LBILockout_St =Active)

- The LBIClient shall display the backup password entry screen when the following three conditions are all met:

CCS is turned on
LBILockout_St =Inactive
The vehicle is not in Enhanced Valet Mode

- The LBIClient shall display the Enhanced Valet password entry screen when the following three conditions are all met:

  - CCS is turned on
  - LBILockout_St =Inactive
  - The vehicle is in Enhanced Valet Mode

4.11 LBIv1-FUN-REQ-258456/B-Transitioning Vehicle from Remote Start State with Backup Password

4.11.1 Use Cases

4.11.1.1 LBIv1-UC-REQ-260140/B-Transitioning Vehicle from Remote Start State with Backup Password

Actors	The LBI User
Pre-conditions	The LBI User has previously activated PaaK for their vehicle via Lincoln mobile app The LBI User is logged into Lincoln app on their mobile phone The LBI User's mobile phone and vehicle are BT connected The LBI User has previously created backup password. The vehicle is locked. The vehicle ignition is off
Scenario Description	The LBI User remote starts the vehicle via mobile app Phone becomes disabled (e.g. battery drained) The LBI User approaches locked vehicle, cannot enter The LBI User enters valid keypad code The vehicle unlocks The LBI User opens door and enters the vehicle The LBI User presses brake pedal The LBI HMI Menu displays backup password entry screen Without being inactive for more than a fixed period of time (defined by T_Password Entry Screen Inactive Timer), the LBI User enters valid backup password via the LBI HMI. The LBI HMI Menu displays a message to inform the LBI User that entered password accepted and provides instruction to exit park Within a fixed period of time (defined by T_Push to Start Timer), the LBI User presses brake pedal and shifts the vehcile from park
Post-conditions	The notification that PaaK Backup has been used is sent to the LBI User via e-mail and a phone message to the PaaK whose backup password is just used

	The vehicle transitions from non-motive to motive state and the LBI User is able to drive away vehicle The LBI HMI instruction of exiting park is dismissed once the vehicle engine is in motive state
List of Exception Use Cases	The LBI User does not enter a valid password The LBI User is inactive for more than a fixed period of time (defined by T_Password Entry Screen Inactive Timer), while the LBI HMI Menu displays password entry screen The LBI User does not shift out of park within a fixed period of time (defined by T_Push to Start Timer) after successfully entering backup password The LBI User press the Start button within a fixed period of time (defined by T_Push to Start Timer), after successfully entering backup password
Interfaces	The LBIClient The LBIServer The LBIServer2

4.11.1.2 LBIv1-UC-REQ-294899/A-Dismissing HMI Exit Park Instruction when Engine Shuts Down

Actors	The LBI User
Pre-conditions	The vehicle is in remote start state with engine running A valid Backup Password is entered
Scenario Description	The LBI HMI Menu displays a message to inform the user that entered password accepted and provides instruction to exit park Within a fixed period of time (defined by T_Push to Start Timer), the LBI User presses start button
Post-conditions	The vehicle engine shuts down The HMI Menu display of exit park instruction is dismissed
List of Exception Use Cases	
Interfaces	The LBIClient The LBIServer The LBIServer2

4.11.2 Requirements

4.11.2.1 LBIv1-REQ-283992/B-HMI Display of Valid Password and Exit Park Instruction

When the Engine is running in NonMotive mode as indicated in PwPckTq_St (PwPckOn_TqNotAvailable), upon receiving a valid password notification, BackupIgnition_Rsp (RespCode =" Challenge Response *Acknowledge*" or "Valet Start Challenge Response *Acknowledge"*, RspStatus =" Valid Password"), the LBIClient HMI shall notify the user that entered password is accepted and shall instruct the user to press brake and shift from Park in order to drive the vehicle.

This HMI message shall be dismissed when any one of conditions occurs:

The T_Push to Start Timer expires

The vehicle Engine status changes to PwPck_St(PwPckOn_TqAvailable)

The vehicle Engine status changes to PwPckTq_St(PwPckOff_TqNotAvailable)

Note:
This requirement is for the use cases of Secure Idle active state with engine running or Remote Start state

PwPckOn_TqNotAvailable→ PwPckOn_TqAvailable implies that the user follows the HMI Instruction and successfully gets out of NonMotive mode to drive away

PwPckOn_TqNotAvailable→ PwPckOff_TqNotAvailable implies that something occurred to cause the engine to shut down thus the HMI instructions are no longer valid and shall be dismissed

4.12 LBIv1-FUN-REQ-275727/A-Altering Keypad Code

4.12.1 Use Cases

4.12.1.1 LBIv1-UC-REQ-276157/B-Failing to Enter Programming Mode

Actors	The LBIServer. The LBIServer2
Pre-conditions	The LBIServer is required to store a keypad code, delete a keypad code or delete all keypad codes in one of following process: Create keypad code Create Enhanced Valet password Delete backup password Reset backup password Exit Enhanced Valet Mode Execute Master Reset Reset a PaaK Revoke Paak
Scenario Description	The LBIServer sends the request of entering programming mode to the LBIServer2 The LBIServer2 is unable to send a successful confirmation The LBIServer does not receive a successful confirmation from the LBIServer2 The LBIServer resents the request for entering programming mode up to the fixed number defined by N_LBINumberOfRetries
Post-conditions	The LBIServer2 does not enter programming mode The LBIServer stops requesting the LBIServer2 to enter programming mode The LBIServer moves to next step
List of Exception Use Cases	
Interfaces	The LBIServer The LBIServer2
Note:	This use case is written from the perspective of the LBIServer and the LBIServer2 after the LBI User has previously activated Phone-as-a-Key feature for vehicle, the LBI User is inside vehicle, the vehicle is in RUN. and 1 associated PaaK device and 1 key fob are inside the vehicle

4.12.1.2 LBIv1-UC-REQ-275728/B-Failing to Add or Delete Keypad Code

Actors	The LBIServer The LBIServer2
Pre-conditions	The LBIServer has received a successful confirmation from the LBIServer2 about entering programming mode The LBIServer is required to store a keypad code, delete a keypad code or delete all keypad codes in one of following processes: Create keypad code Create Enhanced Valet password Delete backup password Reset backup password Exit Enhanced Valet Mode Execute Master Reset Reset a PaaK Revoke Paak

Scenario Description	The LBIServer sends the request of adding, deleting, deleting all to the LBIServer2 The LBIServer2 is unable to send a successful confirmation The LBIServer does not receive a successful confirmation from the LBIServer2 The LBIServer resents the request up to the fixed number defined by N_LBINumberOfRetries
Post-conditions	The keypad code is not stored or deleted The LBIServer stops to request the LBIServer2 store or delete the keypad code The LBIServer moves to next step
List of Exception Use Cases	
Interfaces	The LBIServer The LBIServer2
Note	This use case is written from the perspective of the LBIServer and the LBIServer2 after the LBI User has previously activated Phone-as-a-Key feature for vehicle, the LBI User is inside vehicle, the vehicle is in RUN. and 1 associated PaaK device and 1 key fob are inside the vehicle

4.12.1.3 LBIv1-UC-REQ-276158/B-Adding or Deleting Keypad Code

Actors	The LBIServer The LBIServer2
Pre-conditions	The LBIServer has received a successful confirmation from the LBIServer2 about entering programming mode The LBIServer is required to store a keypad code, delete a keypad code or delete all keypad codes in one of following processes: Create keypad code Create Enhanced Valet password Delete backup password Reset backup password Exit Enhanced Valet Mode Execute Master Reset Reset PaaKs Revoke Paak
Scenario Description	The LBIServer sends the request of adding, deleting, deleting all to the LBIServer2 The LBIServer2 stores or deletes the keypad code associated with the received key index or deletes all keypad codes The LBIServer2 sends a successful confirmation to the LBIServer The LBIServer receives a successful confirmation from the LBIServer2
Post-conditions	The keypad code is stored or deleted or all keypad codes are deleted The LBIServer moves to next step
List of Exception Use Cases	
Interfaces	The LBIServer The LBIServer2
Note	This use case is written from the perspective of the LBIServer and the LBIServer2 after the LBI User has previously activated Phone-as-a-Key feature for vehicle, the LBI User is inside vehicle, the vehicle is in RUN. and 1 associated PaaK device and 1 key fob are inside the vehicle

4.12.2 Requirements

4.12.2.1 LBIv1-REQ-304585/A-PaaK RKE Interface for LBI Keypad Code Programming

The LBIServer shall use PaaK RKE interface defined in *PaaK-REQ-270046 f*or LBI Keypad code functions

Note: As the result of using RKE interface for LBI Keypad code functions, all content in the requirements listed below, only the portions related to the quoted terms (Active Command Response, TargetID and ActionCode) are applied to LBI Keypad code functions:

“Active Command Response” in PaaK-REQ-269555

“TargetID”, “Active Command” and “ActionCode” in PaaK-REQ-269557

“TargetID response” and “Active Command Response” in PaaK-REQ-269558

4.12.2.2 LBIv1-REQ-270265/C-Request to Enter Keypad Code Programming Mode

When there is a need to store or delete keypad codes, before sending the “Add”, “Delete” and “DeleteAll” request via KeyPadCodeProg_Rq, the LBIServer shall first send PaakCtrlRKEData, per LBIv1-REQ-304565, and KeyPadCodeProg_Rq(ProgrammingMode) to request the LBIServer2 enter keypad programming mode.
For the use case of deleting a keypad code, the LBIServer shall only send the request after it confirms that there is a keypad code associated with the received key index.

The use cases that the LBIServer shall comply with include but not limited to:

When receiving a LBI keypad code storing command via BackupIngnition_Rq (OpCode= “Keypad Code Create Request”) as described in LBIv1-REQ-264860

When receiving a backup password deleting command via BackupIngnition_Rq (OpCode= “*Password Delete Request*”) as described in LBIv1-REQ-264877

After generating a rule-compliant random number for Enhanced Valet Password to store the Enhanced Valet keypad code as described in LBIv1-REQ-264904

When receiving Master Reset or Paak Reset as described in LBIv1-REQ-270264

When a backup password is reset after new password is validated as described in LBIv1-REQ-264889

When a PaaK is revoked as described in LBIv1-REQ-271433

Note: PaakCtrlRKEData is a response to periodic RKE challenge, PaakTargetRKEData, from the LBIServer2. If the LBIServer does not properly respond to RKE challenge, the LBIServer2 will not enter programming mode.

4.12.2.3 LBIv1-REQ-304565/A-Designation of RKE Challenge Response in AES Output

For the request of entering keypad code programming mode, the LBIServer shall populate the challenge response data, PaakCtrlRKEData, with bytes [0, 4, 9, 10, 13] of the AES Output as described in PaaK-REQ-269558; and then shall transmit it along with KeyPadCodeProg_Rq(ProgrammingMode).

4.12.2.4 LBIv1-REQ-304562/A-Compute an AES Output for RKE Challenge Response in Entering Programming Mode

For LBI keypad code programming, the LBIServer shall compute an AES Output per LBIv1-REQ-304563 using AES-128 algorithm as described in the Specification for the ADVANCED ENCRYPTION STANDARD (AES) from Federal Information Processing Standards Publication 197.

4.12.2.5 LBIv1-REQ-304563/A-AES Key and AES Input for Computing an AES Output for RKE Challenge Response

To compute data for PaakCtrlRKEData, the LBIServer shall use PaaK-REQ-269555 as a visual aid along with LBIv1-REQ-304562, where AES Key and AES Input are defined below:

The AES key shall be derived from the PaaK TargetID per PaaK-REQ-269556.

The AES Input shall be derived from the received challenge data via PaakTargetRKEData as described in PaaK-REQ-269557.

Note: LBI uses the exact same TargetID obtained in PaaK feature per PaaK-REQ-242454, BBCF-ACT-REQ-241761 and BBCF-SD-REQ-239336. Obtaining TargetID is a one-time process in the plant, or if the modules are swapped, whereas Challenge data is exchanged each time.

4.12.2.6 LBIv1-REQ-304560/A-Add Request of Entering Programming Mode to PaaK RKE Queue

The LBIServer shall add the request of enter keypad programming mode to Paak RKE Queue and shall comply with PaaK-REQ-270047.

Note: this requirement is needed due to the fact that LBI keypad programming uses PaaK RKE interface. As the result, it is possible that one LBI User is doing keypad code programming at the center stack display device and the other PaaK User uses a PaaK device for RKE actions. In such condition, queuing the request of entering keypad programming mode along with other RKE requests is necessary to ensure the entering keypad programming mode request will be executed.

4.12.2.7 LBIv1-REQ-304561/A-Precondition of Sending Request to Enter Programming Mode

The LBIServer shall only send the request of entering programming mode after PaakCtrlActionCode is set to null per PaaK-REQ-307226.

4.12.2.8 LBIv1-REQ-351819/A-Keypad Programming Interface

When the LBIServer receives the signal KeypadCodeProg_St from the LBIServer2 set to any of the following values: 0x2 Add || 0x3 Delete || 0x4 DeleteAll || 0x6 Duplicate, the LBIServer shall immediately send KeyPadCodeProg_Rq = Null and PaakCtrl_RKESubID = 0.

The LBIServer shall test for new challenge data in PaakTargetRKEData and either complete the transaction or time out if no completed data has been received after a period of T_Programming Mode Response Timer expires.

4.12.2.9 LBIv1-REQ-264862/C-Request to Store Keypad Code

When there is a need to store a keypad code, the LBIServer shall send three signals to the LBIServer2 after receiving that LBIServer2 entered programming mode confirmation via KeyPadCodeProg_St(LearningMode).

The three signals are:

PaakCtrlRKEData([Data]), the received keypad code via BackupIngnition_Rq (OpCode= “Keypad Code Create Request”) or the Enhanced Valet Keypad code as defined by LBIv1-REQ-317844 with the format defined by LBIv1-REQ-271443

PaakCtrl_RKESubID (*[index]*), the received key index via BackupIngnition_Rq (OpCode= “Keypad Code Create Request”) or the Enhanced Valet key index as defined by LBIv1-REQ- 270053

KeyPadCodeProg_Rq(Add), the store request

The use cases that the LBIServer shall comply with are:

When receiving a request to store LBI keypad code via BackupIngnition_Rq (OpCode= “Keypad Code Create Request”) as described in LBIv1-REQ-264860

After generating a rule-compliant random number for Enhanced Valet Password to store the Enhanced Valet keypad code as described in LBIv1-REQ-264904

4.12.2.10 LBIv1-REQ-318296/A-DTC for Mismatch Keypad Code and Configuration Parameter

Upon receiving Keypad code TP bit string via BackupIgnition_Rq (Opcod = “Keypad Code Create Request”) as defined by LBIv1-REQ-271253, the LBIServer shall first check if the received data string matches with the Keypad code configuration parameter in terms of the length of keypad code.

If the vehicle Keypad code configuration does not match the data received via BackupIgnition_Rq, the LBIServer shall still transmit the data string to the LBIServer2 as required by LBIv1-REQ-271443; and additionally the LBIServer shall also set the U0554-00 DTC **(***Invalid Data Received From Accessory Protocol Interface Module* DTC) indicating the mismatch between keypad code and configuration parameter.

The definition of mismatch shall include the followings:

Configuration parameter calls out 5-digit code but the received data string has the 6th & 7th button press parameters NOT set to Null as shown in Example 1.

Configuration parameter calls out 7-digit code but the received the data string has the 6th & 7th button press parameters set to Null as shown in Example 2.

Example 1

Bit Position	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Data String												0	1	1	0	1	0	0	0	1	0	1	1	1	0	0	1	0	1	0	0	1
Button Press												Button Press 7			Button Press 6			Button Press 1			Button Press 2			Button Press 3			Button Press 4			Button Press 5		

The data string shown above can only be interpreted either one of the followings:

A mismatched keypad code when a 5-digit code is called out by Keypad code configuration parameter. In this case, the U0554-00 DTC shall be set

A correct 7-digit keypad code (1579236) when a 7-digit code is called out

Example 2

Bit Position	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Data String												0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	1	0	1	0	0	1

The data string shown above can only be interpreted either one of the followings:

A mismatched keypad code when a 7-digit code is called out by Keypad code configuration parameter. In this case, the U0554-00 DTC shall be set

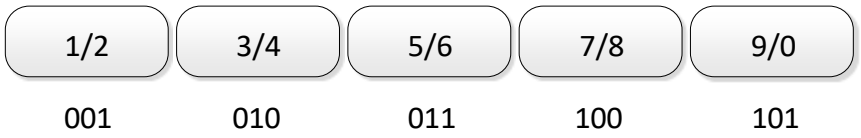
A correct 5-digit keypad code (15792) when a 5-digit code is called out

Note: Since keypad code of number Zero is mapped into 101 in data string as described in LBIv1-REQ-271253, so the above data string can NOT be interpreted as a correct 7-digit keypad cod with last two digits being zero (1579200).

4.12.2.11 LBIv1-REQ-271443/C-LBIServer Keypad Code Transmit Methods

When requesting the LBIServer2 to store a keypad code, the LBIServer shall respond with PaakCtrlRKEData to the periodic RKE challenge, PaakTargetRKEData, from the LBIServer2 with keypad code data embedded in the least significant 21 bits.

See below for mapping of button to bit value:



000 = NULL
001 = “1/2” button pressed
010 = “3/4” button pressed
011 = “5/6” button pressed
100 = “7/8” button pressed
101 = “9/0” button pressed

The seven digits of the keypad code are contained in the lowest 21 bits of the 40-bit signal.
Bit 21 is the most significant bit of the keypad code
Bit 0 is the least significant bit of the keypad code and the 40-bit signal
Bits 21-39 are not used

When the LBIServer receives TP bit string from the LBIClient via BackupIgnition_Rq as defined by LBIv1-REQ-271253, it shall first check the Keypad code configuration parameter then arrange the data accordingly into lower 4 bytes of *PaaKCtlActvData_No_Actl* prior to transmitting it to the LBIServer2.

Bits 0-2 = fifth button pressed
Bits 3-5 = fourth button pressed
Bits 6-8 = third button pressed
Bits 9-11 = second button pressed
Bits 12-14 = first button pressed
Bits 15-17 = sixth button pressed
Bits 18-20 = seventh button pressed

When the vehicle calibration is called out for 5-digit Keypad code, the LBIServer shall ignore Sixth and Seventh button press parameters received via BackupIgnition_Rq; and the LBIServer shall set Sixth and Seventh button press parameters (Bits 15 to 20) to 000 while transmitting the data string out to the LBIServer2 when 5-digit codes are implemented.

Note: The keypad code is a 7-button sequence where each button is represented by three bits. Based on the vehicle configuration (e.g. there are markets that require the use of 7-digit codes and 5-digit codes), the data string sent by the LBIServer will either have Null values (000) for the 6th and 7th digits (for 5-digit codes), or actual inputted values (7-digit codes). In either case, the sequence will always be a 7-digit sequence.

Example of bit string data for keypad code of 1579236 is shown below

Bit Position	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Data String																				0	1	1	0	1	0	0	0	1	0	1	1	1	0	0	1	0	1	0	0	1
Button Press																				Button Press 7			Button Press 6			Button Press 1			Button Press 2			Button Press 3			Button Press 4			Button Press 5		
Keypad Code (ex. 1579236)																				6			3			1			5			7			9			2		
	Reserved																			Bytes Used																				

4.12.2.12 LBIv1-REQ-275752/B-Request to Delete Keypad Code

When there is a need to delete a keypad code or to delete all keypad codes, the LBIServer shall send the delete request to the LBIServer2 after the confirmation KeyPadCodeProg_St(LearningMode) is received.

To delete all keypad codes as required by Master Reset or PaaK Reset described by LBIv1-REQ- 270264, the LBIServer shall send PaakCtrl_RKESubID(0x00) and KeyPadCodeProg_Rq(DeleteAll).

To delete one particular keypad code, the LBIServer shall send two signals:

PaakCtrl_RKESubID(*[index]*)

the received key index via BackupIngnition_Rq (OpCode= “Password Delete Request” or “Reset 1 Password Transmit” or” Reset 2 Password Transmit”)

or

the Enhanced Valet key index as defined by LBIv1-REQ- 270053 for BackupIngnition_Rq (OpCode=” Check for Keys to Exit Valet Mode” or “Valet Delete Challenge Response”)

or

the key index of a CAK that was revoked

KeyPadCodeProg_Rq(Delete), the delete request

The use cases for one keypad code deletion are:

When receiving a backup password deletion command via BackupIngnition_Rq (OpCode= “*Password Delete Request*”) as described in LBIv1-REQ-264877

When a backup password is reset after new password is validated as described in LBIv1-REQ-264889

When a PaaK is revoked as described in LBIv1-REQ-271433

4.12.2.13 LBIv1-REQ-277531/B-Resend Keypad Code Request Requirement

After sending KeyPadCodeProg_Rq (either “ProgrammingMode”, “Add”, “Delete” or “DeleteAll”), if any one of the conditions below is met, the LBIServer shall abort and restart the entire keypad programming process from the beginning (i.e. from executing LBIv1-REQ-270265), for up to N_LBINumberOfRetries times before sending failed response.

KeyPadCodeProg_St(ProgrammingFailure) is received regardless of the request (“ProgrammingMode”, “Add”, “Delete”, “Delete All”)

KeyPadCodeProg_St(Duplicate) if the request was to store a valet keypad code as indicated in *BackupIgnition_Rq(OpCode = “*Check for Keys to Enter Valet Mode” or ”Valet Create Challenge Response”).

No response via KeyPadCodeProg_St after T_Programming Mode Response Timer expires

Communication data is invalid. One of examples could be receiving KeyPadCodeProg_St(Delete) after sending KeyPadCodeProg_Rq(Add).

Communication data is corrupted

4.12.2.14 LBIv1-REQ-275827/B-N_LBINumberOfRetries

Name	Description	Units	Range	Resolution	Default
N_LBINumberOfRetries	The maximum number that the LBIServer shall resend KeyPadCodeProg_Rq before quitting or moving to next step Note: use default value		0-5	1	2

4.12.2.15 LBIv1-REQ-275867/A-Track Invalid Keypad Code Response

The LBIServer shall keep track of invalid keypad Code responses in separated counters for entering programming mode as well as storing and deleting keypad code with N_ProgrammingModeInvalidResponseCounter and N_KeyPadCodeInvalidResponseCounter respectively.

After sending KeyPadCodeProg_Rq and waiting for a fixed time period as defined in T_Programming Mode Response Timer, the LBIServer shall increment N_ProgrammingModeInvalidResponseCounter if it does not receive a successful confirmation of KeyPadCodeProg_St(LearningMode).

After sending KeyPadCodeProg_Rq and waiting for a fixed time period as defined in T_Programming Mode Response Timer, the LBIServer shall increment N_KeypadCodeInvalidResponseCounter if it does not receive a successful confirmation of KeyPadCodeProg_St(Add), KeyPadCodeProg_St(Delete) or KeyPadCodeProg_St(DeleteAll).

4.12.2.16 LBIv1-TMR-REQ-270037/A-T_Programming Mode Response Timer

Name	Description	Units	Range	Resolution	Default
T_Programming Mode Response Timer	Maximum time the LBIServer2 can wait before response to the request of entering programming mode Note: Use the default value	sec	1-5	1	2

4.12.2.17 LBIv1-REQ-275870/A-Reset Invalid Keypad Code Response Counters

The LBIServer shall reset N_ProgrammingModeInvalidResponseCounter to zero when one of the following conditions is met:

A successful confirmation of KeyPadCodeProg_St(LearningMode) is received

N_ProgrammingModeInvalidResponseCounter=(N_LBINumberOfRetries +1)

The LBIServer shall reset N_KeypadCodeInvalidResponseCounter to zero when one of the following conditions is met:

A successful confirmation of KeyPadCodeProg_St(Add), KeyPadCodeProg_St(Delete) or KeyPadCodeProg_St(DeleteAll) is received

N_KeypadCodeInvalidResponseCounter=(N_LBINumberOfRetries +1).

4.12.2.18 LBIv1-REQ-275826/A-N_ProgrammingModeInvalidResponseCounter

Name	Description	Units	Range	Resolution	Default
N_ProgrammingModeInvalid ResponseCounter	The counter for tracking the invalid response of entering programming mode		0 to (N_LBINumberOfRetries+1)	1	0

4.12.2.19 LBIv1-REQ-275869/A-N_KeypadCodeInvalidResponseCounter

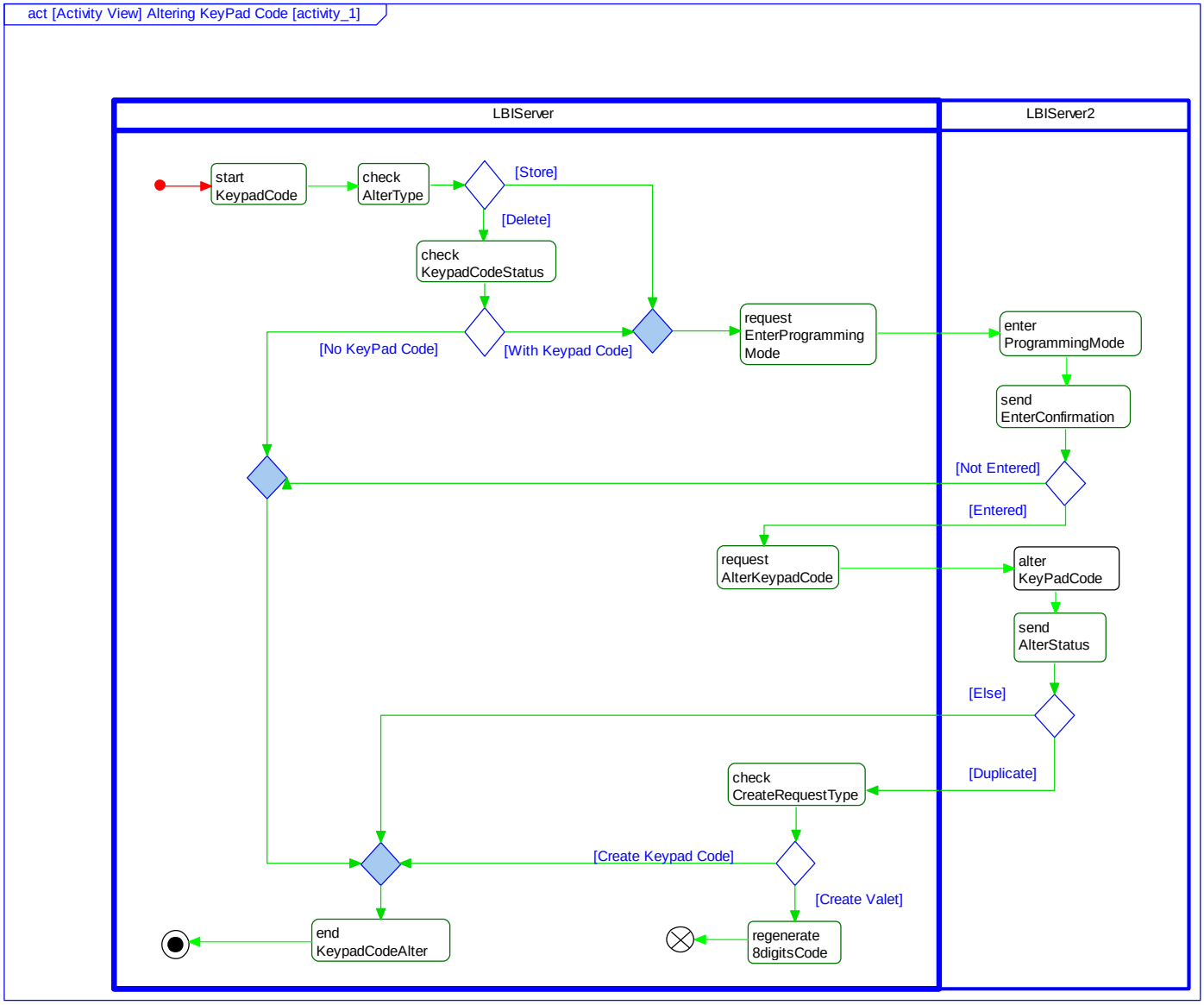
Name	Description	Units	Range	Resolution	Default
N_KeypadCodeInvalid ResponseCounter	The counter for tracking the invalid response of storing and deleting keypad code		0 to (N_LBINumberOfRetries+1)	1	0

White Box Views

4.12.2.20 Activity Diagrams

4.12.2.20.1LBIv1-ACT-REQ-275750/A-Alter Keypad Code

Activity Diagram



4.12.2.21 Sequence Diagrams

4.12.2.21.1LBIv1-SD-REQ-275751/A-Alter Keypad Code

Constraints

Pre-Condition

The LBI HMI displays a message that a backup password has been created successfully
The LBI HMI asks t The LBI User if he/she would like to create a LBI Keypad

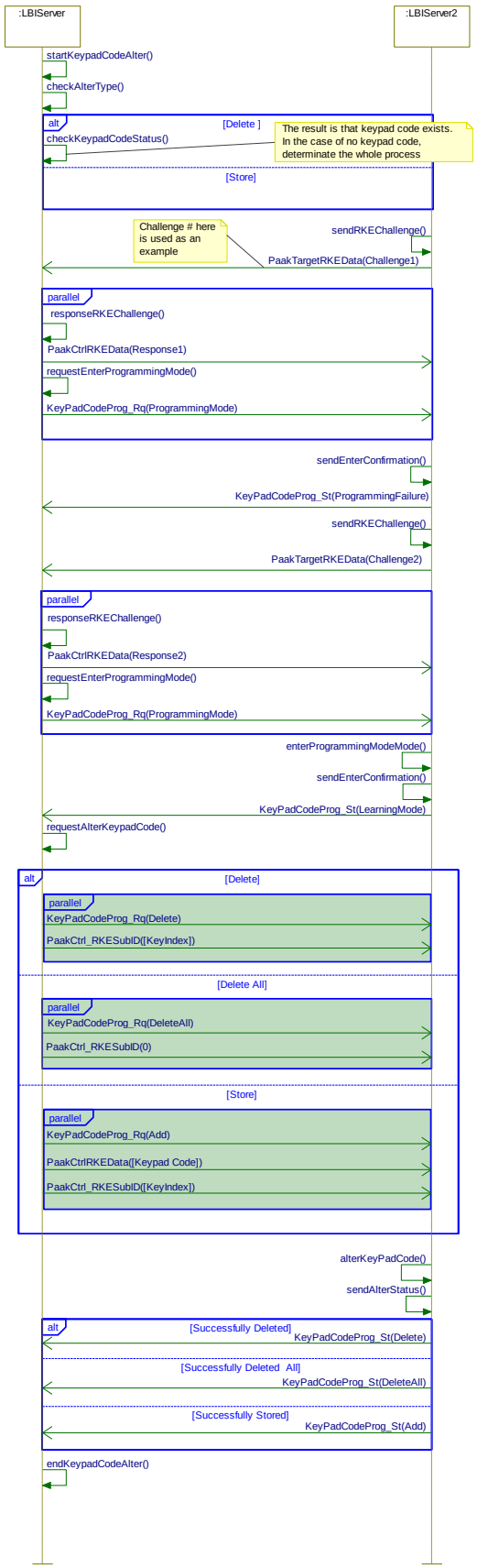
Scenarios

Normal Usage

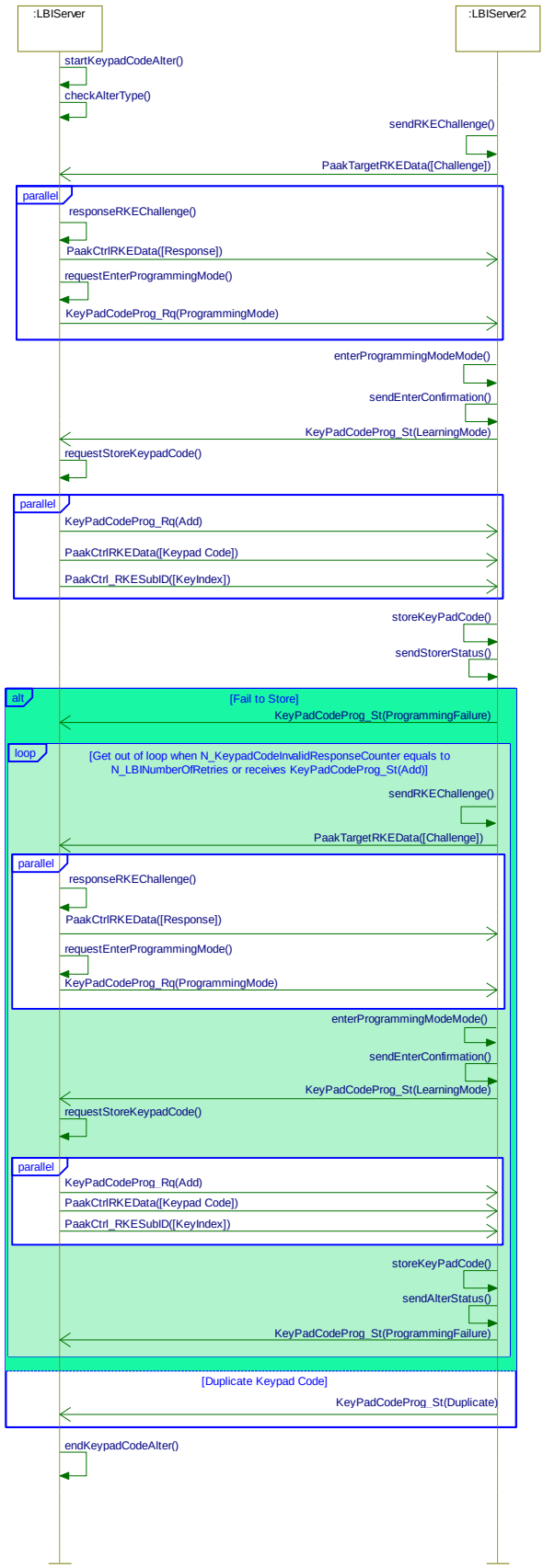
Post-Condition

A LBI Keypad code is ready for use

Sequence Diagram



4.12.2.21.2LBIv1-SD-REQ-277795/A-Storing Keypad Code-Failure and Duplicate Path Sequence Diagram



4.13 LBIv1-FUN-REQ-264539/B-Exiting Secure Idle State with Backup Password

4.13.1 Use Cases

4.13.1.1 LBIv1-UC-REQ-260141/B-Exiting Secure Idle State with Backup Password when Engine Off

Actors	The LBI User
Pre-conditions	The LBI User has previously created backup password The vehicle is in Secure Idle active state with engine off The vehicle transmission is in park The LBI User is outside and away from the vehicle The vehicle is unlocked No associated key fobs or Paak devices are inside vehicle
Scenario Description	The LBI User returns to the vehicle The LBI User attempts to start engine The LBI HMI Menu displays backup password entry screen. Without being inactive for more than a fixed period of time (defined by T_Password Entry Screen Inactive Timer, the LBI User enters valid backup password via the LBI HMI Menu The HMI Menu displays to inform the user that entered password accepted and provides instruction to start engine Within a fixed period of time (defined by T_Push to Start Timer), the LBI User presses start button while holding brake pedal
Post-conditions	The notification that PaaK Backup has been used is sent to the LBI User via e-mail and a phone message to the PaaK whose backup password is just used The vehicle starts with engine running The LBI HMI instruction is dismissed once the vehicle engine starts to run
List of Exception Use Cases	The LBI User does not enter a valid backup password The LBI User is inactive for more than the time defined by T_Password Entry Screen Inactive Timer while the LBI HMI Menu displays password entry screen The LBI User presses start button without pressing brake pedal after password is accepted
Interfaces	LBIClient LBIServer2 LBIServer

4.13.1.2 LBIv1-UC-REQ-294907/A-Exiting Secure Idle State with Backup Password when Engine Running

Actors	The LBI User
Pre-conditions	The LBI User has previously created backup password The vehicle is in Secure Idle active state with engine running The vehicle transmission is in park The LBI User is outside and away from the vehicle The vehicle is unlocked No associated key fobs or Paak devices are inside vehicle
Scenario Description	The LBI User returns to the vehicle The LBI User attempts to shift out of park The LBI HMI Menu displays backup password entry screen. Without being inactive for more than a fixed period of time (defined by T_Password Entry Screen Inactive Timer, the LBI User enters valid backup password via the LBI HMI Menu The HMI Menu displays to inform the user that entered password accepted and provides instruction to exit park

	Within a fixed period of time (defined by T_Push to Start Timer), the LBI User changes gear position while holding brake pedal
Post-conditions	The notification that PaaK Backup has been used is sent to the LBI User via e-mail and a phone message to the PaaK whose backup password is just used The vehicle transitions from non-motive to motive state and the LBI User is able to drive away vehicle The LBI HMI instruction of exiting park is dismissed once the vehicle engine is in motive state
List of Exception Use Cases	The LBI User does not enter a valid backup password The LBI User is inactive for more than the time defined by T_Password Entry Screen Inactive Timer while the LBI HMI Menu displays password entry screen The LBI User does not shift out of park within a fixed period of time (defined by T_Push to Start Timer) after successfully entering backup password The LBI User press the Start button within a fixed period of time (defined by T_Push to Start Timer), after successfully entering backup password
Interfaces	LBIClient LBIServer2 LBIServer

4.13.1.3 LBIv1-UC-REQ-294906/A-Dismissing HMI Start Engine Instruction when Entering NonMotive Mode

Actors	The LBI User
Pre-conditions	The vehicle is in Secure Idle active state with engine off A valid Backup Password is entered
Scenario Description	The LBI HMI Menu displays a message to inform the user that entered password accepted and provides instruction to start engine Within a fixed period of time (defined by T_Push to Start Timer), a remote start is trigger
Post-conditions	The vehicle engine is running in NonMotive mode The HMI Menu display of starting engine instruction is dismissed
List of Exception Use Cases	
Interfaces	The LBIClient The LBIServer The LBIServer2

4.13.1.4 LBIv1-UC-REQ-294899/A-Dismissing HMI Exit Park Instruction when Engine Shuts Down

Actors	The LBI User
Pre-conditions	The vehicle is in remote start state with engine running A valid Backup Password is entered
Scenario Description	The LBI HMI Menu displays a message to inform the user that entered password accepted and provides instruction to exit park Within a fixed period of time (defined by T_Push to Start Timer), the LBI User presses start button
Post-conditions	The vehicle engine shuts down The HMI Menu display of exit park instruction is dismissed
List of Exception Use Cases	

Interfaces	The LBIClient The LBIServer The LBIServer2

4.13.2 Requirements

4.13.2.1 LBIv1-REQ-260084/C-HMI Display of Valid Entry and Start Engine Instruction

When the Engine is not running as indicated in PwPckTq_St(PwPckOff_TqNotAvailable), upon receiving a valid password notification, BackupIgnition_Rsp (RespCode =" Challenge Response *Acknowledge*" or "Valet Start Challenge Response *Acknowledge"*, RspStatus =" Valid Password"), the LBIClient HMI shall notify the user that the entered password has been accepted and that they can now press the brake and start button in order to start the vehicle

This HMI message shall be dismissed when any one of conditions occurs:

The T_Push to Start Timer expires

The vehicle Engine status changes to PwPck_St(PwPckOn_TqAvailable)

The vehicle Engine status changes to PwPckTq_St(PwPckOn_TqNotAvailable)

Note:
PwPckOff_TqNotAvailable→ PwPckOn_TqAvailable implies that the user follows the HMI Instruction and successfully starts the engine

PwPckOff_TqNotAvailable→ PwPckOn_TqNotAvailable implies that something occurred to cause the Engine to change to NonMotive mode (e.g. Remote Start occurred) thus the HMI instructions are no longer valid and shall be dismissed

4.13.2.2 LBIv1-REQ-283992/B-HMI Display of Valid Password and Exit Park Instruction

When the Engine is running in NonMotive mode as indicated in PwPckTq_St (PwPckOn_TqNotAvailable), upon receiving a valid password notification, BackupIgnition_Rsp (RespCode =" Challenge Response *Acknowledge*" or "Valet Start Challenge Response *Acknowledge"*, RspStatus =" Valid Password"), the LBIClient HMI shall notify the user that entered password is accepted and shall instruct the user to press brake and shift from Park in order to drive the vehicle.

This HMI message shall be dismissed when any one of conditions occurs:

The T_Push to Start Timer expires

The vehicle Engine status changes to PwPck_St(PwPckOn_TqAvailable)

The vehicle Engine status changes to PwPckTq_St(PwPckOff_TqNotAvailable)

Note:
This requirement is for the use cases of Secure Idle active state with engine running or Remote Start state

PwPckOn_TqNotAvailable→ PwPckOn_TqAvailable implies that the user follows the HMI Instruction and successfully gets out of NonMotive mode to drive away

PwPckOn_TqNotAvailable→ PwPckOff_TqNotAvailable implies that something occurred to cause the engine to shut down thus the HMI instructions are no longer valid and shall be dismissed

5 Appendix A: Definitions / Acronyms

Acronym	Full Name	Description
APIM	Accessory Protocol Interface Module	Also known as “SYNC module”
BCM	Body Control Module	
BLEM	Bluetooth Low Energy Module	Main controller for Phone-as-a-Key system
CAK	Customer Acceptance Key	One of the virtual keys delivered to the phone or the vehicle. Used for authorizing phones to the vehicle.
CAN	Controller Area Network	
FEI	Fob-free Entry & Ignition	Feature that allows user to enter and start the vehicle without a key fob or phone.
LBI	Lincoln Backup Ignition	a.k.a. PaaK Backup
IPC	Instrument Panel Cluster	Also known as “cluster”
PaaK	Phone-as-a-Key	Feature whereby user is able to enter and start vehicle with mobile phone similarly to PEPS.
PaaK Backup	Phone-as-a-Key Backup	Backup starting feature tied to Phone-as-a-Key.
PEPS	Passive Entry, Passive Start	In the context of key fobs, a feature that allows user to enter and start the vehicle without interacting with their key fob i.e. passive operation.
TCU	Telematics Control Unit	
SDN	Service Delivery Network	Also known as the “cloud” or the “back-end”.
PaakFI	IT's backend server implementation of PaaK	IT's backend server implementation of PaaK

6 Appendix B: Reference Documents

Reference #	Document Title
1	Phone-as-a-Key PSD
2	BLE Interface Security Specification
3	TP BLEM/APIM/TCU SPSS
4	BLEM/BLEAM Common Function BLEM SPSS
5	Phone-as-a-Key BLE Communication Protocol
6	H21 SYNC 3 GUI Design Standards
7	PaaK Max Level SPSS
8	LBI Feature Function Specification
9	BCM Functional Specification
10	URL: https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html
11	URL: https://github.com/danielmiessler/SecLists/blob/master/Passwords/xato-net-10-million-passwords-10000.txt

The requirements of the documents listed in the reference table above, of the latest revision level, form a part of this Engineering Specification