



Google Pay Smart Tap 2.1 in ViVOPay™ Devices User Guide

**23 April 2020
Rev. R**

Copyright © 2020 ID TECH. All rights reserved.

ID TECH
10721 Walker St.
Cypress, CA 90630

This document, as well as the software and hardware described in it, is furnished under license and may be used or copied online in accordance with the terms of such license. The content of this document is furnished for information use only, is subject to change without notice, and should not be construed as a commitment by ID TECH. Reasonable effort has been made to ensure the accuracy of information provided herein. However, ID TECH assumes no responsibility or liability for any unintentional errors or inaccuracies that may appear in this document.

Except as permitted by such license, no part of this publication may be reproduced or transmitted by electronic, mechanical, recording, or otherwise, or translated into any language form without the express written consent of ID TECH. ID TECH and ViVOpay are registered trademarks of ID TECH.

Warranty Disclaimer: The services and hardware are provided "as is" and "as-available" and the use of the services and hardware is at its own risk. ID TECH does not make, and hereby disclaims, any and all other express or implied warranties, including, but not limited to, warranties of merchantability, fitness for a particular purpose, title, and any warranties arising from a course of dealing, usage, or trade practice. ID TECH does not warrant that the services or hardware will be uninterrupted, error-free, or completely secure.

Table of Contents

1. INTRODUCTION 5

2. SMART TAP 2.1 HIGH LEVEL OVERVIEW..... 6

 2.1. Types of Transactions6

3. SMART TAP 2.1: SUPPORTED PRODUCTS 8

 3.1. Product Differences8

4. SMART TAP 2.1 CONFIGURATION 9

 4.1. Basic Google Pay Smart Tap 2.1 Setup Flow9

 4.2. Smart Tap 2.1 Configuration Commands9

 4.2.1. *Set Configurable Group (04-03)*..... 9

 4.2.2. *Get Configurable Group (03-06)*..... 10

 4.2.3. *Set Poll Mode Command (01-01)*..... 11

 4.2.4. *Change USB Interface (01-0B)*..... 12

 4.2.5. *Set Data Output Mode (01-0C)*..... 12

 4.2.6. *Automatic Output for Auto Poll (01-0D)*..... 13

 4.3. Non-Security Parameters 14

 4.4. VAS-Only Transactions 16

 4.5. Main Status Return Code 16

 4.6. Security Parameters 16

 4.6.1. *Encrypted vs. Non-Encrypted Session Data* 16

 4.6.2. *Remote Key Injection*..... 17

5. SMART TAP 2.1 DEVICE TRANSACTION COMMANDS 18

 5.1. ACT Command (Activate Transaction)..... 18

 5.2. Response 19

6. PUSH SMART TAP DATAFORMAT 20

 6.1. Push SmartTap Data 20

 6.1.1. *Push Service NDEF Record*..... 20

 6.1.2. *Service Status NDEF Record*..... 20

 6.1.3. *Service Usage NDEF Record* 20

 6.1.4. *New Service NDEF Record*..... 21

 6.1.5. *Push Service Response*..... 22

7. GET SMART TAP DATA FORMAT 23

 7.1. Service Object NDEF Record Types 23

 7.1.1. *Customer NDEF Record*..... 23

 7.1.2. *Loyalty NDEF Record*..... 23

 7.1.3. *Offer NDEF Record*..... 23

 7.1.4. *Gift Card NDEF Record* 24

 7.1.5. *PLC NDEF Record*..... 24

 7.1.6. *Event Ticket NDEF Record* 24

 7.1.7. *Flight NDEF Record*..... 24

 7.1.8. *Service Issuer NDEF Record*..... 24

8. SIMPLIFIED OUTPUT 26

9. SMART TAP 2.1 EXAMPLES 27

 9.1. Configuring the Terminal for SmartTap: Loading the parameters..... 27

 9.2. Get VAS-Only Transaction 28

 9.3. Push VAS Only Activate Transaction 38

 9.4. Encrypted VAS Only Activate Transaction 40

9.5. Simplified Output..... 41

10. APPENDIX A: ECC KEY PAIR.....42

10.1. How to Create an ECC Key Pair Using Open-SSL 42

1. Introduction

Various contactless card readers ID TECH produces under the ViVOPay name support Google Pay Smart Tap loyalty technology. This document describes ID TECH's Smart Tap 2.1 implementation as it applies to ViVOPay devices and serves as an integration guide.

Note that Google is the authoritative source of information on Smart Tap. Smart Tap is a Google proprietary technology, the internal details of which are confidential. Developers should obtain available Smart Tap online documentation from Google to gain an understanding of Smart Tap concepts and data representations before using this document.

This document describes the ViVOPay device configuration options that pertain to Smart Tap and the data flows that occur during a Smart Tap transaction. The business logic that applies to "value added" data is beyond the scope of this document. The guide below describes the ways applicable ViVOPay devices convey value-added services (VAS) data in the course of a "tap" (or user session).

2. Smart Tap 2.1 High Level Overview

Smart Tap 2.1 is a contactless (NFC) card emulation protocol for providing value-added services (VAS).

The Google Pay Smart Tap specification allows a Google Pay wallet to exchange added value information with a host system (POS, phone, tablet) that can send that info to a back-end system. VAS data is requested at "tap" time using the standard Start Transaction payment command (with optional TLVs included in the request). The transaction can be payment-only, VAS-only, or a combination (see discussion below).

This document focuses on the value-added services (VAS) side of Google Pay: loyalty, coupons, discounts, "points," and related information, and only deals with the payment side when necessary. We use the generic term VAS (Value Added Services) for all functions and explanations related to the non-financial-card aspects of the wallet functionality.

To function properly, the ViVOPay reader must be initially set up with the correct configuration parameters. Merchants must obtain the required configuration information (such as the Collector ID) from Google and properly configure the reader on their own with that information using the appropriate APIs described below. Generally speaking, this is a one-time setup operation that does not need to be repeated after a reader has been deployed in the field (note that the long term private key may need to be rotated periodically).

After the reader is configured, certain parameters must be included during transactions to indicate to the reader that the transaction should be of a VAS type.

The key elements of the communication between the reader and the wallet are:

1. The reader and the wallet use asymmetric elliptical-curve cryptography (ECC) to protect the data in transit between the phone and the reader. Security is based on an ECC256 key pair where the private key (LTPK: long term private key) is stored in the reader and the public key is part of the wallet.
2. The data exchanged between the reader and the wallet follow an NFC Data Exchange Format (NDEF) structure (see [NFC Forum Technical Specifications](#)). The ViVOPay reader delivers this data wrapped in a TLV structure (see examples below). The reader is agnostic with respect to NDEF content; that is, the ViVOPay device itself does not "know" anything special about NDEF data. It's just another form of data that the device conveys from the card to the host.

2.1. Types of Transactions

ID TECH's ViVOPay readers perform payment and VAS interactions with the wallet with an **Activate Transaction** command (sometimes also called **Start Transaction**).

For Smart Tap 2.1, the possible transaction modes are:

1. Payment only (No VAS interaction).
2. VAS only (No payment, read the VAS data).
3. VAS and Payment (Allow both VAS and Payment to be performed with the wallet).

4. VAS over Payment (If VAS data is available, get VAS with no Payment; if and only if there is no VAS, perform payment if available).

The wallet can also receive data from the POS (or host) via a **Push VAS** command.

Conversely, ID TECH readers provide the ability to output the VAS data to the POS in any of several modes:

1. In the clear (as NDEF) with no encryption of the data fields (USB-HID).
2. With encryption of sensitive fields (as defined by Google), as NDEF encrypted using the reader's Data Encryption Key (USB-HID).
3. Using a simplified mode (using USB-KB, or "keyboard device" USB), on applicable readers. One or several service objects are extracted from the NDEF structure and provided to the back end without any other data.

ID TECH readers also provide the ability to rotate the LTPK (long term private key) for readers in the field using a secure protocol.

Note: Simplified mode supports Get VAS Only and Secure Get VAS Only. **Activate Transaction** using other modes will be rejected (response status code 0C/05).

3. Smart Tap 2.1: Supported Products

ID TECH supports Smart Tap 2.1 on the following ViVOpay products:

- Kiosk III
- Kiosk IV
- Vendi
- VP3300 Products
- VP3600
- VP5300
- VP5300M
- VP6300
- VP6800
- VP8300
- VP8800
- PiP*

***Note:** PiP only works for loyalty programs; it does not support payments.

3.1. Product Differences

Note that most of the above-listed products use ID TECH's NEO-series firmware, whereas the VP8800 utilizes AR-series firmware. The **Activate Transaction** command (and some others) are different for VP8800 devices; on NEO devices, **Activate Transaction** is typically the **02-40** command, whereas AR devices use the **02-05** command.

Likewise, NEO devices use a slightly different command protocol (ViVOTech2) than AR 3.0 products (which use ViVOpayV3). These differences, documented in detail in the *Interface Developer's Guides* (IDG) for NEO and AR, have no bearing on how Smart Tap 2.1 works. The same TLVs, payload semantics, configuration requirements, and interaction flows occur in both NEO and AR devices.

4. Smart Tap 2.1 Configuration

Before a device can participate in Smart Tap transactions, it must be configured with various parameters. Broadly, this means TLV-based parameters that fall into one of two categories:

- Non-security-related parameters having to do with Collector ID (and reader capabilities): see next section
- Security-related parameters (see further below)

Devices require specific configuration parameters before they can perform Smart Tap transactions. During setup, device administrators must provide two main pieces of information: the Collector ID (Google's term for merchant ID), which is set with the **Set Configurable Group (04-03)** command and tag DFEE3B, and the long-term private key (LTPK), which must be injected (see [Security Parameters](#) below).

Smart Tap transaction behavior is further configurable with specific TLV tags via the **Set Configurable Group (04-03)** command; see the Interface Developer's Guide appropriate to the desired device type for configuration options involving AIDs, groups, and/or datasets. However, Smart Tap 2.1 should work out-of-the-box with the default parameters in Group 8E.

4.1. Basic Google Pay Smart Tap 2.1 Setup Flow

Google Pay Smart Tap 2.1 setup uses the following commands in sequence:

1. The **Set Configurable Group (04-03)** command sets the Group as 8E and the Collector ID, which tells Smart Tap what loyalty program to use.
2. Set the reader's long-term private key (LTPK).
3. Set tag DFED3F and tag DFED49 in Group 0 to manage [VAS Encryption](#).
4. Set **Poll on Demand Mode (01-01)** to set the reader to auto-poll or poll on demand for a phone tap.
5. Set **Change Simplified Output Mode (01-0C)** to select normal or simplified output mode.

4.2. Smart Tap 2.1 Configuration Commands

The following section provides details for commands used in SmartTap configuration.

4.2.1. Set Configurable Group (04-03)

The **Set Configurable Group** command creates or modifies a TLV Group. Configure a specific TLV Group by passing the TLVs with the desired functionality and a unique TLV Group Number to the reader.

Command Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14 ... Byte 14+n-1	Byte 14+n	Byte 15+n
Header Tag & Protocol Version	Command	Sub-Command	Data Length (MSB)	Data Length (LSB)	Data	CRC (LSB)	CRC (MSB)
ViV0tech2\0	04h	03h			TLV Data Objects		

Response Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Header Tag & Protocol Version	Command	Status Code	Data Length (MSB)	Data Length (LSB)	CRC (MSB)	CRC (LSB)
ViV0tech2\0	04h	See Status Code Table	00h	00h		

4.2.2. Get Configurable Group (03-06)

The **Get Configurable Group** command returns all TLVs associated the specified Configurable Group. A configurable Group Tag must be included as the ONLY TLV in this command. The response should contain all the tags associated with this configurable Group.

Command Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14 ... Byte 14+n-1	Byte 14+n	Byte 15+n
Header Tag & Protocol Version	Command	Sub-Command	Data Length (MSB)	Data Length (LSB)	Data	CRC (LSB)	CRC (MSB)
ViV0tech2\0	03h	06h			TLV Data Objects		

The following TLV **MUST** be encoded in the command, it is the **ONLY** tag included in the command.

FFE4 ^[1]	Group Number	MAND	The group that contains the properties for this AID Note: This must be the ONLY TLV in Data Field.	n2	1
---------------------	--------------	------	--	----	---

^[1] These objects use proprietary tags. The use of these tags should be restricted to the serial interface. After the reader has received these values and saved them in memory, it should dispose of the tags (and not keep them associated with these two values).

Response Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14 ... Byte 14+n-1	Byte 14+n	Byte 15+n
Header Tag & Protocol Version	Command	Status Code	Data Length (MSB)	Data Length (LSB)	Data	CRC (MSB)	CRC (LSB)
ViVotech2\0	03h	See Status Code Table			TLV Data Objects		

4.2.3. Set Poll Mode Command (01-01)

The **Set Poll Mode** command sets whether the ViVOpay devices uses Auto Poll or Poll on Demand.

Command Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15	Byte16
Header Tag & Protocol Version	Command	Sub-Command	Data length (MSB)	Data length (LSB)	Data	CRC (MSB)	CRC (LSB)
ViVotech2\0	01	01h	00h	01h	Poll Mode		

Poll Mode:

00h = Auto Poll

01h = Poll on Demand

Response Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14 ... Byte 14+n-1	Byte 14+n	Byte15+n
Header Tag & protocol version	Command	Status	Data Length (MSB)	Data Length (LSB)	data	CRC (MSB)	CRC (LSB)
vivotech2\0	01	See Status Code Table, NEO 2 IDG	00h	00h			

4.2.4. Change USB Interface (01-0B)

The **Change USB Interface** command sets whether the ViVOPay device uses USB-HID or USB-KB. When USB-KB, Auto Poll, and Burst Mode On are all enabled, the payload output format changes to ASCII strings.

Command Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 14+n	Byte 15+n
Header Tag & Protocol Version	Command	Sub-Command	Data Length (MSB)	Data Length (LSB)	Data	CRC (LSB)	CRC (MSB)
ViVOTech2\0	01h	0Bh	00h	01h	USB Interface		

Byte 1: USB Interface

- 00h = USB will change to USB-HID.
- 01h = USB will change to USB Keyboard.

Response Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Header Tag & Protocol Version	Command	Status Code	Data Length (MSB)	Data Length (LSB)	CRC (MSB)	CRC (LSB)
ViVOTech2\0	01h	See Status Code Table, NEO 2 IDG	00h	00h		

4.2.5. Set Data Output Mode (01-0C)

The **Set Data Output Mode** command sets whether the output mode is normal, simplified, or tags only.

Command Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 14+n	Byte 15+n
Header Tag & Protocol Version	Command	Sub-Command	Data Length (MSB)	Data Length (LSB)	Data	CRC (LSB)	CRC (MSB)
ViVOTech2\0	01h	0Ch	00h	01h	Mode		

Byte 1: Mode

Byte	Output Description	Terminal Type
00h = Normal mode	IDG header and trailer plus VAS data in tag.	Used in VAS Only, VAS-plus-payment, and payment-only terminals.
01h = Simplified output mode	VAS data not in tag, no IDG header and trailer.	Only used in VAS Only terminals.
02h = Tags only	VAS data in tag, no IDG header and trailer.	Used in VAS Only, VAS-plus-payment, and payment-only terminals.

Response Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Header Tag & Protocol Version	Command	Status Code	Data Length (MSB)	Data Length (LSB)	CRC (MSB)	CRC (LSB)
ViVOTech2\0	01h	See Status Code Table, NEO 2 IDG	00h	00h		

To decrypt VAS data, load the private key before the transaction.

If the reader receives an ACT command, it responds with ViVOPay2 protocol and NDEF records tag DFEF76.

4.2.6. Automatic Output for Auto Poll (01-0D)

The **Automatic Output for Auto Poll** command sets the device to output data automatically for Auto Poll mode.

Command Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 14+n	Byte 15+n
Header Tag & Protocol Version	Command	Sub-Command	Data Length (MSB)	Data Length (LSB)	Data	CRC (LSB)	CRC (MSB)
ViVOTech2\0	01h	0Dh	00h	01h	Mode		

Byte 1: Mode

- 00h = Off
- 01h = On : output data on good reads
- 02h = On: output data on good and bad reads

Automatic mode sends out data without the **Get Transaction Results** command. The data is formatted according to the **Set Data Output Mode** command. This command only affects Auto Poll mode.

Response Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Header Tag & Protocol Version	Command	Status Code	Data Length (MSB)	Data Length (LSB)	CRC (MSB)	CRC (LSB)
ViVOTech2\0	01h	See Status Code Table	00h	00h		

4.3. Non-Security Parameters

Use the non-security-related parameters and the corresponding tags to configure reader options for SmartTap. Administrators can send these parameters to a reader at configuration time to establish persistent settings, or optionally at Activate Transaction time to override configurations on a per-transaction basis.

Note that of the first four parameters, related to the merchant and terminal, only the Collector ID is required to be non-empty. Tags with no predefined default value in the table below are generally optional.

Note: It is never necessary to send zero-length (empty payload) TLVs. Instead, omit empty TLVs.

The first five tags shown below are customer-specific and must be populated by the merchant (or customer). Only the Collector ID is mandatory (Collector ID is assigned by Google).

Tag	Length (bytes)	Name	Default Value
DFEE3B	Maximum 8	Collector ID (mandatory); only the rightmost 4 bytes are used. Tag length will be 4 in most cases.	No default value.
DFEE3C	Up to 32	Store Location ID	No default value.
DFEE3D	Up to 32	Terminal ID	No default value.
DFEF25	Up to 32	Merchant Name	No default value.
DFED01	4	Merchant Category	No default value.
DFED02	5	POS Capabilities Bitmaps (see table below)	000000001
DFED03	1	Retry Times	01
DFED04	1	Select OSE support	01
DFED05	1	Skip Second Select support	01
DFED06	1	Stop payment if SmartTap 2.1 failed	00
DFED07	1	Pre-Signed Support	00
DFED27*	1	Delimiter for Service Objects*	0D
DFED3F	1	VAS encryption on/off flag	(OFF): default (ON) enables VAS payload encryption
DFED49	1	VAS-only global override	Bit 0 set to 0: off (default) Bit 0 set to 1: on
DFEF77*	1	Enable/Disable Multiple Service Objects*	00

*Valid only for Simplified Output mode.

Notes:

- The default value for **DFED27** is **0x0D (CR)**.
- The value for tag **DFEF77** can be **0x00(Disable)** or **0x01(Enable)**. If the function is disabled, the reader returns only the first service object in the NDEF record.

DFED02 (POS Capabilities): 5 bytes with flags as follows (1 = ON, 0 = OFF)

Byte	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
System	Standalone	Semi-integrated	Unattended	Online	Offline	MMP	zlib support	RFU
UI	Printer	Printer graphics	Display	Images	Audio	Animation	Video	RFU
Checkout	Support payment	Support digital receipt	Support Service issuance	Support OTA POS data	RFU	RFU	RFU	RFU
CVM	Online PIN	CD PIN	Signature	No CVM	Device-generated code	SP-generated code	ID capture	Biometric
Tap	VAS Only	Payment Only	VAS and Payment	VAS over Payment	RFU	RFU	RFU	RFU

Example 04-03 configuration command (for NEO-series firmware):

```
5669564f74656368320004030044ffe4018edfee3b0400bc614edfee3c00dfee3d00df
ef2500dfed0100dfed02050000000001dfed0300dfed040101dfed050101dfed0600df
ed0700dfed27010ddfef77001082
```

Parsed:

56 69 56 4F 74 65 63 68 32 00: ViVOTech2\0 header

04 03: Command

00 44: Length (less CRC)

FFE4: Group Number / Fallback Group

DFEE3B: Collector ID

DFEE3C: Store Location ID

DFEE3D: Terminal ID

DFEF25: Output Data Format Select

DFED01: Merchant Category

DFED02: POS Capabilities Bitmaps

DFED03: Retry Times

DFED04: Select OSE support

DFED05: Skip Second Select support

DFED06: Stop Payment if SmartTap2.1 failed support

DFED07: Pre-Signed support

DFEF77: Timeout for waiting next command

10 82: CRC16

The above example assumes a NEO-firmware device. An AR-firmware device (such as a VP8800) uses the same command but wraps it in the device-appropriate protocol (in this case, ViVOPayV3) and produces responses that conform to that protocol.

4.4. VAS-Only Transactions

To enable VAS-only transactions (for contactless readers used only for loyalty programs, for example), use tag DFED49 with zero of the (one byte) turned on. When this bit is on, the reader assumes that no financial transaction will take place.

Note: Parameter configurations determine the behavior of each wallet and payment scheme:

- The VAS-only tag **DFED49** applies to all VAS transactions and if set disables payments. It overrides any other settings.
- Payment-only mode is allowed in VAS configuration.
- The main status code is a status of the current transaction.
- VAS transactions have a main status of **0x57** with the error code in the VAS container.
- Payment transactions have a main status of **(00, 23, 0A)**.
- VAS Payment-only mode follows EMV flow and without select-OSE.

4.5. Main Status Return Code

Main status return codes for ACT:

- If no **FFEE06** and no **FFEE08**, Payment only:
 - Return code is **Payment code (00, 23, 0A)**.
- **FFEE06** No Payment:
 - Return code is **57 + FFEE06**.
- **FFEE06** and Payment:
 - Return code is **Payment code (00, 23, 0A) + FFEE06 (VAS data)**.
- **FFEE08**, No Payment:
 - Return code is **57 + FFEE08**.
- **FFEE08** and Payment:
 - Return code is **Payment code (00, 23, 0A) + FFEE08**.
- **FFEE06** and **FFEE08**, No Payment:
 - Return code is **57 + FFEE06/FFEE08**.
- **FFEE06** and **FFEE08** and Payment:
 - Return code is **Payment code (00, 23, 0A) + FFEE06/FFEE08**.

4.6. Security Parameters

Secure interaction between the reader and the wallet requires a LTPK (long term private key). The ECC key-pair (consisting of a 32-byte LTPK private key and corresponding public key) must be customer-generated (see [Appendix A: ECC Key Pair](#)). Injecte that key-pair into the ViVOPay device via the key injection firmware command outlined in the next section.

4.6.1. Encrypted vs. Non-Encrypted Session Data

The reader can apply (or not apply) encryption to the VAS payload separate from the financial transaction payload depending on settings specified at transaction time. Use value **01** in tag **DFED3F**

to indicate the reader should apply encryption to the VAS payload. Use value **00** to indicate that the reader should not encrypt VAS data. The default is **0x00 (off)**.

This choice only affects whether or not the reader encrypts VAS data; it does not affect payment data encryption.

4.6.2. Remote Key Injection

For products supporting the symmetric key RKI method, the ID TECH RKI host directly injects the LTPK. Contact ID TECH for details on the protocol. The LTPK uses the same commands as any other key and a TR-31 block to carry the key.

5. Smart Tap 2.1 Device Transaction Commands

The following section describes Smart Tap 2.1 Device Transaction commands.

5.1. ACT Command (Activate Transaction)

The Activate Transaction (ACT) command must include the Smart Tap Options tag (FFEE08) to make a Smart Tap 2.1 transaction. Including this tag tells the reader to attempt a SmartTap transaction if it is included on the target device.

Note that **FFEE08** is a constructed (group) tag and must contain at least one other TLV.

The FFEE08 Group Tag must include tag DFEF1A. It conveys the transaction’s terminal mode, as defined below.

Tag DFEF1A: Terminal Mode

b 8	b 7	b 6	b 5	b 4	b 3	b 2	b 1	Description
0	0	0	0					RFU
				x	x	x	x	0000 : VAS OVER Payment Mode 0001 : VAS AND Payment Mode 0010 : VAS Only Mode 0011 : Payment Mode Only 0101 : Push VAS AND Payment Mode 0110 : Push VAS Only Mode 1000 : Secure Get VAS OVER Payment Mode 1001 : Secure Get VAS AND Payment Mode 1010 : Secure Get VAS Only Mode

The FFEE08 tag may optionally contain configuration tags (see [Non-Security Parameters](#)) that override any parameters configured earlier. Those parameters are valid for one transaction only.

If a Push is scheduled, the data to send to the wallet will be included in tag DFEF1B. The data is the Push Service NDEF record as defined in the Push SmartTap data request described below.

Tag DFED28 can include Service Types from SmartTap data requests. The Service Type data included constitutes the Service List NDEF record as defined in the Get SmartTap data request.

Service Type Byte

Value	Description
0x00	All services
0x01	All services except PPSE
0x02	PPSE
0x03	Loyalty
0x04	Offer
0x05	Gift Card
0x06	Private Label Card
0x07	Event Ticket
0x08	Flight
0x09-0x0F	RFU TWI
0x10	Cloud Based Wallet
0x11	Mobile Marketing Platform
0x0C-0x3F	RFU TWI
0x40	Wallet Customer
0x41-0x6F	RFU Wallet-specific
0x70-0x9F	RFU Merchant-specific

Use the **02-40** version of the **Activate Transaction** command for encrypted transactions. Consult the relevant Interface Developer's Guide (IDG) for more information on this command.

5.2. Response

SmartTap payloads contain NDEF (NFC Data Exchange Format) records. The VAS NDEF structure(s) returned in a Smart Tap interaction are embedded in tag DFEF76 and described below in more details

If the standard Get mode is used, the VAS data returns unencrypted with all fields visible.

If the secure Get mode is used, the VAS data will returns to the reader encrypted in the DEK (Data Encryption Key) standard.

For examples, see [Smart Tap 2.1 Examples](#) below.

6. Push Smart Tap Data Format

The section below describes how to push data from the host merchant to the reader.

6.1. Push SmartTap Data

The push option provides a mechanism to send data to the wallet from the Merchant POS System through an NFC terminal. The data is included in Tag DFEF1B in the activate transaction command.

6.1.1. Push Service NDEF Record

Offset	Length	Description	Record Type	M/O/C
0	6	Push Service NDEF Header	spr	M
7	2	Push Service NDEF Version		M
9		Session NDEF Record	ses	M
...		POS Capabilities Record	pcr	0
...		Service Status NDEF Record 1	ssr	0
...	...		ssr	0
...		Service Status NDEF Record n	ssr	0
...		New Service NDEF Record 1	nsr	0
...	...		nsr	0
...		New Service NDEF Record n	nsr	0

6.1.2. Service Status NDEF Record

Offset	Length	Description	Record Type	Payload Format	M/O/C
0	6	Service Status NDEF Header	ssr		M
7		Object ID	oid	Binary PRIMITIVE	M
...		Service Usage NDEF Record	sug		0
...		Service Update NDEF Record	sup		0

6.1.3. Service Usage NDEF Record

Used to determine if service record was applied.

Offset	Length	Description	Record Type	Record ID	Payload Format	M/O/C
0	6	Service Usage NDEF Header	sug			M
7	1	Service Usage Status Byte				M
...		Service Usage Title Ndef Record	T	sut	RTD TEXT	0
...		Service Usage Description Ndef Record	T	sud	RTD TEXT	0

- **Service Usage Title:** Intended to be used for basic information regarding the usage of the valuable. For example, for the transit use case, it is recommended the title be the name of the station where the transit pass was redeemed.
- **Service Usage Description:** Intended to be used to provide more context on how the valuable was used. For example, continuing the transit use case, this text could contain information about tagging in or out of a station, which station, etc.

Service Usage Status Byte

Value	Description
0x00	Unspecified
0x01	Success
0x02	Invalid format
0x03	Invalid value
0x04-0xFF	RFU

6.1.3.1. Service Update NDEF Record

If applied (as determined in service usage), provide an update. Service Update operation defines if this an 'add' to balance, 'remove' from balance, etc.

Offset	Length	Description	Record Type	M/O/C
0	6	Service Update NDEF Header	sup	M
7	1	Service Update Operation		M
8		Service Update Payload		O

Service Update Operation Byte

Value	Description	Payload Format
0x00	No operation	Payload ignored
0x01	Remove service object	Payload ignored
0x02	Set balance	Binary
0x03	Add balance	Binary
0x04	Subtract balance	Binary
0x05	Free	Optional. Seconds remaining in binary.
0x06-0xFF	RFU	Payload ignored

Free: The valuable is used to redeem a free good or service and the valuable is not consumed nor reduces in value when it is redeemed. For example, using a bus pass for a transfer. In this case, if the card was used 35 minutes ago and provided free transfers for 2 hours, the payload would be 0x13EC when used now.

6.1.4. New Service NDEF Record

Offset	Length	Description	Record Type	Payload Format	M/O/C
0	6	New Service NDEF Header	nsr		M
1		New Service Type Byte			M
...		New Service Title Ndef Record	nst	TEXT PRIMITIVE	M
...		New Service URI NDEF Record	nsu	URI	M

- **Service Title:** A short, concise description of the valuable. Intended to potentially be used for the user to differentiate between multiple valuables pushed back in the same tap. Examples of text would be "Buy 1 Get 1 Free" or "Safeway Club". Explicit details on terms and conditions, expiration dates, applicable merchandise, etc. should not be in this text, but instead be accessible from the URI.
- **Service URI:** An endpoint where the user may access the new service

New Service Type Byte

Value	Title	Description
0x00	Unspecified	
0x01	Valuable	Link to the Save to Android endpoint to add a new valuable
0x02	Receipt	Merchant website with detailed receipt information
0x03	Survey	Merchant website with survey
0x04	Goods	Merchant website with access to digital merchandise
0x05	Signup	Merchant website where user can signup for new valuable
0x06-0xFF	RFU	

6.1.5. Push Service Response

Offset	Length	Description	Record Type	M/O/C
0	6	Push Service Response NDEF Header	psr	M
7		Session NDEF Record	ses	M
...	2	ISO 7816-4 Status		M

7. Get Smart Tap Data Format

A Service Object is a wrapper for some piece of information. The wrapped object could be a customer ID, closed loop card, loyalty card, or some other valuable and/or non-payment card. The type of information contained inside of the object is inferred based on the record type.

7.1. Service Object NDEF Record Types

The following section lists the record types used in Service Objects.

7.1.1. Customer NDEF Record

Offset	Length	Description	Record Type or ID	Payload Format	M/O/C
0	6	NDEF Header	cus		M
7		Customer ID	cid	PRIMITIVE	M
...		Preferred Language Code	cpl	Text PRIMITIVE	O
...		Unique Tap ID	cut	PRIMITIVE	O
...		Unique Device ID	cud	PRIMITIVE	O

The Customer ID is generated by the mobile device and is unique to that Collector ID. If the same user taps at the same merchant again, the merchant will see the same ID. If another merchant with a different Collector ID retrieves the Customer NDEF record, that merchant will receive a different Customer ID.

The service issuer ID on the parent Service NDEF record will be the wallet that issued this customer record.

The Unique Tap ID is generated by the mobile device. It is unique for every tap.

7.1.2. Loyalty NDEF Record

Offset	Length	Description	Record Type or ID	Payload Format	M/O/C
0	6	Loyalty NDEF	Header	ly	M
6	Var	Object ID	oid	Binary PRIMITIVE	M
...	Var	Service Number NDEF Record	n	PRIMITIVE or RTD_TEXT	M
...	Var	Track 1 NDEF Record	tr1	PRIMITIVE or RTD_TEXT	O
...	Var	Track 2 NDEF Record	tr2	PRIMITIVE or RTD_TEXT	O

7.1.3. Offer NDEF Record

Offset	Length	Description	Record Type or Id	Payload Format	M/O/C
0	6	Offer NDEF Header	of		M
6	Var	Object ID	oid	Binary PRIMITIVE	M
...	Var	ServiceNumberNDEFRecord 1	n	PRIMITIVE or RTD_TEXT	M

7.1.4. Gift Card NDEF Record

Offset	Length	Description	Record Type or Id	Payload Format	M/O/C
0	6	Gift Card NDEF Header	gc		M
6	Var	Object ID	oid	Binary PRIMITIVE	M
...	Var	Service Number NDEF Record	n	PRIMITIVE or RTD_TEXT	M
...	Var	Service PIN NDEF Record	p	PRIMITIVE or RTD_TEXT	0
...	Var	Track 1 NDEF Record	tr1	PRIMITIVE or RTD_TEXT	0
...	Var	Track 2 NDEF Record	tr2	PRIMITIVE or RTD_TEXT	0

7.1.5. PLC NDEF Record

Offset	Length	Description	Record Type or Id	Payload Format	M/O/C
0	6	PLC NDEF Header	pl		M
6		Object ID	oid	Binary PRIMITIVE	M
...		Service Number NDEF Record	n	PRIMITIVE or RTD_TEXT	M
...		Service Expiration NDEF Record	ex	PRIMITIVE or RTD_TEXT	0
gGi...		Service CVC1 NDEF Record	c1	PRIMITIVE or RTD_TEXT	0
		Track 1 NDEF Record	tr1	PRIMITIVE or RTD_TEXT	0
		Track 2 NDEF Record	tr2	PRIMITIVE or RTD_TEXT	0

7.1.6. Event Ticket NDEF Record

Offset	Length	Description	Record Type or Id	Payload Format	M/O/C
0	6	Event Ticket NDEF Header	et		M
6	Var	Object ID	oid	Binary PRIMITIVE	M
...	Var	ServiceNumberNDEFRecord 1	n	PRIMITIVE or RTD_TEXT	M

7.1.7. Flight NDEF Record

Offset	Length	Description	Record Type or Id	Payload Format	M/O/C
0	6	Flight NDEF Header	fl		M
6	Var	Object ID	oid	Binary PRIMITIVE	M
...	Var	ServiceNumberNDEFRecord 1	n	PRIMITIVE or RTD_TEXT	M

7.1.8. Service Issuer NDEF Record

Offset	Length	Description	Record Type	M/O/C
0	4	Service Issuer NDEF Header	i	M
6	1	Service Format Byte		M
5	1	Service Issuer Byte		M
7		Service Issuer Payload		M

8. Simplified Output

ViVOPay devices produce simplified output when operating in USB-KB (keyboard) mode. Readers producing simplified output include the Service Object NDEF records without any formatting (header or trailer) in an ASCII encoding appropriate to a keyboard device.

Two modes, configured with tag DFEF77, affect Service Object NDEF records:

- One (0x00) outputs the first Service Number NDEF record (the first field after the ObjectID) for the first Service Object following the Customer NDEF record; the Customer NDEF record will always be ignored.
- The second (0x01) outputs all Service Number NDEF records for all Service Objects (the first field after the ObjectID) following the Customer NDEF record, which is ignored, and without the service issuer NDEF record. Each data record is separated by the delimiter defined by tag DEED27; the delimiter also completes the string.

The output does not contain the format byte.

9. Smart Tap 2.1 Examples

The examples shown below are for NEO devices and use the ViVOTech2 protocol. Substitute relevant AR firmware commands and protocols for the VP8800 device.

For example, the **Set Configurable Group (04-03)** command exists in both NEO and AR firmware, but the protocol framing differs. In examples that use **Activate Transaction**, the ACT command is **02-40** on NEO devices and **02-05** on AR devices. Consult the appropriate IDG for detailed information about these commands.

9.1. Configuring the Terminal for SmartTap: Loading the parameters

Request: Use the **Set Configurable Group (04-03)** command to set the Collector ID (a 4-byte value; in this example, **00 BC 61 4E**).

Command:

```
5669564f74656368320004030044ffe4018edfee3b0400bc614edfee3c00dfee3d00df
ef2500dfed0100dfed02050000000001dfed0300dfed040101dfed050101dfed0600df
ed0700dfed27010ddfef77001082
```

Parsed:

- 56 69 56 4F 74 65 63 68 32 00:** ViVOTech2\0 header
- 04 03:** Command
- 00 44:** Length (less CRC)
- FFEE4:** Group Number / Fallback Group
- DFEE3B:** Collector ID
- DFEE3C:** Store Location ID
- DFEE3D:** Terminal ID
- DFEF25:** Output Data Format Select
- DFED01:** Merchant Category
- DFED02:** POS Capabilities Bitmaps
- DFED03:** Retry Times
- DFED04:** Select OSE support
- DFED05:** Skip Second Select support
- DFED06:** Stop Payment if SmartTap2.1 failed support
- DFED07:** Pre-Signed support
- DFEF77:** Timeout for waiting next command
- 10 82:** CRC16

Response:

```
56 69 56 4F 74 65 63 68 32 00 04 00 00 00 AE 16
```

9.2. Get VAS-Only Transaction

Issue the **Start Transaction** command, specifying **Get VAS Only**. In this terminal mode, only VAS data is requested. No payment data is requested.

Command (NEO firmware):

```
56 69 56 4F 74 65 63 68 32 00 02 40 00 1B 30 9F 02 06 00 00 00 00 00
01 9C 01 00 FF EE 08 0A DF EF 1A 01 02 DF ED 28 01 00 F4 19
```

Parsed:

56 69 56 4F 74 65 63 68 32 00: ViVOTech2\0 header
02 40: Start Transaction Command
00 1B: Length of payload
30: Timeout value
9F 02: Authorized Amount
06 00 00 00 00 00 01: Length (06) and data
9C: Transaction Type
 01 00: Length (01) and data
FF EE 08: Configuration Tags container
 0A: Length (10 bytes)
DF EF 1A: Terminal Mode
 01 02: Length (01) and flag data (02 means VAS Only)
DF ED 28: Service Type Requests
 01 00: Length (01) and data
F4 19: CRC16

Response:

```
56 69 56 4F 74 65 63 68 32 00 02 57 00 7C 01 FF EE 08 66 DF EF 76 62
94 03 2F 61 73 76 94 01 06 69 04 02 71 79 79 71 54 03 1F 63 75 73 94
03 06 63 69 64 04 12 34 56 78 90 19 01 03 03 54 63 70 6C 00 65 6E 54
03 02 63 75 74 04 7B 54 03 27 61 73 76 94 01 05 69 05 01 F7 97 98 54
02 19 6C 79 94 03 09 6F 69 64 04 AC 80 1C BF CA 8D 5C 3A 54 01 06 6E
05 F3 24 23 42 34 9F 39 01 07 FF EE 01 04 DF 30 01 00 DF EE 26 01 01
0F D3
```

Parsed Response:

56 69 56 4F 74 65 63 68 32 00: ViVOTech2\0 header
02: Command group
57: Response code (**57** means no payment occurred; VAS only)
 00 7C: Length
01: Attribution byte (**01:** Contactless card)
FF EE 08: Configuration Tags container
 66: Length
DF EF 76: SmartTap data (NDEF records)

62: Length

94 03 2F 61 73 76 94 01 06 69 04 02 71 79 79 71 54 03 1F 63 75 73 94 03 06 63 69 64 04 12 34 56 78 90 19 01 03 03 54 63 70 6C 00 65 6E 54 03 02 63 75 74 04 7B 54 03 27 61 73 76 94 01 05 69 05 01 F7 97 98 54 02 19 6C 79 94 03 09 6F 69 64 04 AC 80 1C BF CA 8D 5C 3A 54 01 06 6E 05 F3 24 23 42 34: NDEF Data, below.

9F 39: POS Entry Mode per ISO-8583

01 07: Length (01) and mode (07: Contactless EMV: see ISO-8583)

FF EE 01: Clearing Record (Group Tag)

04: Length

DF 30: Clearing Record

01 00: Length (01) and data

DF EE 26: Attribution byte (refer to NEO IDG)

01 01: Length (01) and data (01: Contactless card)

0F D3: CRC16

7.1. Get VAS and Payment Transaction

VAS data is requested, followed by payment. Payment is always requested.

Command (NEO firmware):

56 69 56 4F 74 65 63 68 32 00 02 40 00 1B 30 9F 02 06 00 00 00 00 00
01 9C 01 00 FF EE 08 0A DF EF 1A 01 01 DF ED 28 01 00 14 D7

56 69 56 4F 74 65 63 68 32 00: ViVOTech2\0 header

02 40: Start Transaction Command

00 1B: Length of payload

30: Timeout value

9F 02: Amount, Authorized

06: Length

00 00 00 00 01: Data for Amount

9C: Transaction Type

01: Length

00: Data

FF EE 08: Smart Tap Options (Group Tag)

0A: length

DF EF 1A: Terminal Mode for Smart Tap

01: Length

00: Data (Get VAS AND Payment Mode)

DF ED 28: Service Type Request

01: Length

00: Data

14 D7: CRC

Response:

```

56 69 56 4F 74 65 63 68 32 00 02 23 02 04 11 82 02 00 00 95 05 00 00
00 00 00 9A 03 14 08 10 9C 01 00 5F 2A 02 08 40 9F 02 06 00 00 00 00
00 01 9F 03 06 00 00 00 00 00 00 9F 06 07 A0 00 00 00 04 10 10 9F 09
02 00 02 9F 1A 02 08 40 9F 1E 08 30 30 30 30 30 30 30 30 9F 21 03 12
03 03 9F 33 03 00 00 E8 9F 34 03 00 00 00 9F 35 01 22 9F 36 02 00 00
9F 37 04 06 C7 B7 BD 9F 39 01 91 9F 53 01 00 DF 81 29 08 30 F0 F0 00
30 F0 FF 00 FF 81 06 31 DF 81 2A 18 33 33 30 30 30 33 33 33 30 30 30
32 32 32 32 32 30 30 30 31 31 31 31 30 DF 81 2B 07 90 00 99 00 00 00
0F DF 81 15 06 00 00 00 00 00 FF FF 81 05 74 50 0A 4D 61 73 74 65 72
43 61 72 64 84 07 A0 00 00 00 04 10 10 9F 11 01 01 9F 6D 02 00 01 56
3E 42 35 34 31 33 31 32 33 34 35 36 37 38 34 38 30 30 5E 53 55 50 50
4C 49 45 44 2F 4E 4F 54 5E 31 39 30 36 31 30 31 33 33 37 38 30 33 33
33 30 30 30 32 32 32 32 32 33 31 33 31 31 31 31 38 9F 6B 13 54 13 12
34 56 78 48 00 D1 90 61 01 90 00 99 31 38 89 8F FF EE 01 2F DF 30 01
00 DF 31 18 33 33 30 30 30 33 33 33 30 30 30 32 32 32 32 32 30 30 30
31 31 31 31 30 DF 32 0D 39 30 30 30 39 39 30 30 30 30 30 30 30 FF EE
08 66 DF EF 76 62 94 03 2F 61 73 76 94 01 06 69 04 02 71 79 79 71 54
03 1F 63 75 73 94 03 06 63 69 64 04 12 34 56 78 90 19 01 03 03 54 63
70 6C 00 65 6E 54 03 02 63 75 74 04 7B 54 03 27 61 73 76 94 01 05 69
05 01 F7 97 98 54 02 19 6C 79 94 03 09 6F 69 64 04 6F 0A F4 F6 F6 56
63 21 54 01 06 6E 05 F3 24 23 42 34 DF EF 4C 06 00 27 00 00 00 00 DF
EF 4D 27 3B 35 34 31 33 31 32 33 34 35 36 37 38 34 38 30 30 3D 31 39
30 36 31 30 31 39 30 30 30 39 39 33 31 33 38 38 39 38 3F DF EE 26 01
11 E6 C5

```

Parsed:

56 69 56 4F 74 65 63 68 32 00: ViVOTech2\0 header

02: Command

23: Response

02 04: Length of payload

11: Attribution byte

82: Application Interchange Profile (AIP)

02: Length

00 00: Value

95: Terminal Verification Results (TVR)

05: Length

00 00 00 00 00: Value

9A: Transaction Date

03: Length

14 08 10: Value

9C: Transaction Type

01: Length

00: Value

5F 2A: Transaction Currency Code

02: Length

08 40: Value

- 9F 02:** Amount, Authorized
 - 06:** Length
 - 00 00 00 00 00 01:** Value
- 9F 03:** Amount, Other
 - 06:** Length
 - 00 00 00 00 00 00:** Value
- 9F 06:** Application Identifier (AID)
 - 07:** Length
 - A0 00 00 00 04 10 10:** Value
- 9F 09:** Application Version Number
 - 02:** Length
 - 00 02:** Value
- 9F 1A:** Terminal Country Code
 - 02:** Length
 - 08 40:** Value
- 9F 1E:** Interface Device (IFD) Serial Number
 - 08:** Length
 - 30 30 30 30 30 30 30 30:** Value
- 9F 21:** Transaction Time
 - 03:** Length
 - 12 03 03:** Value
- 9F 33:** Terminal Capabilities
 - 03:** Length
 - 00 00 E8:** Value
- 9F 34:** Cardholder Verification Method (CVM) Results
 - 03:** Length
 - 00 00 00:** Value
- 9F 35:** Terminal Type
 - 01:** Length
 - 22:** Value
- 9F 36:** Application Transaction Counter (ATC)
 - 02:** Length
 - 00 00:** Value
- 9F 37:** Unpredictable Number (UN)
 - 04:** Length
 - 06 C7 B7 BD:** Value
- 9F 39:** Point-of-Service (POS) Entry Mode
 - 01:** Length
 - 91:** Value
- 9F 53:** Terminal Interchange Profile (dynamic)
 - 01:** Length
 - 00:** Value

- DF 81 29:** Outcome Parameter Set
 - 08:** Length
 - 30 F0 F0 00 30 F0 FF 00:** Value
- FF 81 06:** Discretionary Data
 - 31:** Length
- DF 81 2A:** DD Card (Track1)
 - 18:** Length
- 33 33 30 30 30 33 33 33 30 30 30 32 32 32 32 32 30 30 30 31 31 31 31 30:** Value
- DF 81 2B:** DD Card (Track2)
 - 07:** Length
 - 90 00 99 00 00 00 0F:** Value
- DF 81 15:** Error Indication
 - 06:** Length
 - 00 00 00 00 00 FF:** Value
- FF 81 05:** Data Record
 - 74:** Length
- 50:** Application Label
 - 0A:** Length
- 4D 61 73 74 65 72 43 61 72 64:** Value
- 84:** Dedicated File (DF) Name
 - 07:** Length
 - A0 00 00 00 04 10 10:** Value
- 9F 11:** Issuer Code Table Index
 - 01:** Length
 - 01:** Value
- 9F 6D:** Kernel 4 Reader Capabilities
 - 02:** Length
 - 00 01:** Value
- 56:** Track 1 Data
 - 3E:** Length
 - 42 35 34 31 33 31 32 33 34 35 36 37 38 34 38 30 30 5E 53 55 50 50 4C 49 45 44 2F**
 - 4E 4F 54 5E 31 39 30 36 31 30 31 33 33 37 38 30 33 33 33 30 30 30 32 32 32 32 32**
 - 33 31 33 31 31 31 31 38:** Value
- 9F 6B:** Track 2 Data
 - 13:** Length
 - 54 13 12 34 56 78 48 00 D1 90 61 01 90 00 99 31 38 89 8F:** Value
- FF EE 01:** ViVOpay TLV Group Tag for Clearing Record
 - 2F:** Length
- DF 30:** Track Data Source
 - 01:** Length
 - 00:** Value
- DF 31:** DD Card Track 1

- 18:** Length
33 33 30 30 30 33 33 33 30 30 30 32 32 32 32 30 30 30 31 31 31 31 30: Value
- DF 32:** DD Card Track 2
0D: Length
39 30 30 39 39 30 30 30 30 30 30: Value
- FF EE 08:** Smart Tap Result Set
66: Length
- DF EF 76:** NDEF data (See [Get VAS Only Transaction](#) for details)
62: Length
94 03 2F 61 73 76 94 01 06 69 04 02 71 79 79 71 54 03 1F 63 75 73 94 03 06 63 69 64 04 12 34 56 78 90 19 01 03 03 54 63 70 6C 00 65 6E 54 03 02 63 75 74 04 7B 54 03 27 61 73 76 94 01 05 69 05 01 F7 97 98 54 02 19 6C 79 94 03 09 6F 69 64 04 6F 0A F4 F6 F6 56 63 21 54 01 06 6E 05 F3 24 23 42 34: Value
- DF EF 4C:** MSR Equivalent Data Length Values (for data returned in DFEF4D)
06: Length
00 27 00 00 00 00: Value
- DF EF 4D:** MSR Equivalent Data (Track Data and/or PAN, encrypted)
27: Length
3B 35 34 31 33 31 32 33 34 35 36 37 38 34 38 30 30 3D 31 39 30 36 31 30 31 39 30 30 30 39 39 33 31 33 38 38 39 38 3F: Value
- DF EE 26:** Encryption Status Information
01: Length
11: Value (same as Attribution byte)
- E6 C5:** CRC

7.2. Push VAS AND Pay Activate Transaction

Command (NEO firmware):

```

56 69 56 4F 74 65 63 68 32 00 02 40 01 5F 30 9F 02 06 00 00 00 00 00
01 9C 01 00 FF EE 08 82 01 4C DF EF 1A 01 05 DF EF 1B 82 01 41 14 03
65 73 73 72 94 03 03 6F 69 64 04 23 34 54 03 56 73 75 67 01 99 01 19
03 54 73 75 74 02 65 6E 52 65 77 61 72 64 20 50 72 6F 67 72 61 6D 20
41 70 70 6C 69 65 64 59 01 2C 03 54 73 75 64 02 65 6E 31 30 20 70 6F
69 6E 74 73 20 68 61 76 65 20 62 65 65 6E 20 61 64 64 65 64 20 74 6F
20 79 6F 75 72 20 61 63 63 6F 75 6E 74 54 03 65 73 73 72 94 03 03 6F
69 64 04 16 2E 54 03 56 73 75 67 01 99 01 19 03 54 73 75 74 02 65 6E
52 65 77 61 72 64 20 50 72 6F 67 72 61 6D 20 41 70 70 6C 69 65 64 59
01 2C 03 54 73 75 64 02 65 6E 31 30 20 70 6F 69 6E 74 73 20 68 61 76
65 20 62 65 65 6E 20 61 64 64 65 64 20 74 6F 20 79 6F 75 72 20 61 63
63 6F 75 6E 74 54 03 65 73 73 72 94 03 03 6F 69 64 04 04 D2 54 03 56
73 75 67 01 99 01 19 03 54 73 75 74 02 65 6E 52 65 77 61 72 64 20 50
72 6F 67 72 61 6D 20 41 70 70 6C 69 65 64 59 01 2C 03 54 73 75 64 02
65 6E 31 30 20 70 6F 69 6E 74 73 20 68 61 76 65 20 62 65 65 6E 20 61
64 64 65 64 20 74 6F 20 79 6F 75 72 20 61 63 63 6F 75 6E 74 1D 44
    
```

Parsed Command:

56 69 56 4F 74 65 63 68 32 00: ViVOTech2\0 header

02 40: Command (Activate Transaction)

01 5F: Length of payload

30: Timeout value

9F 02: Amount, Authorized

06: Length

00 00 00 00 00 01: Value

9C: Transaction Type

01 00: Length and data

FF EE 08: Configuration Container

82: Overflow flag (0x80) and "length of length" (0x02)

01 4C: Length

DF EF 1A: Terminal Mode

01: Length

05: Mode

DF EF 1B: Outgoing NDEF Service Record

82: Overflow flag and "length of length"

01 41: Length

NDEF Service Record of length 01 41:

```

14 03 65 73 73 72 94 03 03 6F 69 64 04 23 34 54 03 56 73 75
67 01 99 01 19 03 54 73 75 74 02 65 6E 52 65 77 61 72 64 20
50 72 6F 67 72 61 6D 20 41 70 70 6C 69 65 64 59 01 2C 03 54
73 75 64 02 65 6E 31 30 20 70 6F 69 6E 74 73 20 68 61 76 65
20 62 65 65 6E 20 61 64 64 65 64 20 74 6F 20 79 6F 75 72 20
61 63 63 6F 75 6E 74 54 03 65 73 73 72 94 03 03 6F 69 64 04
16 2E 54 03 56 73 75 67 01 99 01 19 03 54 73 75 74 02 65 6E
52 65 77 61 72 64 20 50 72 6F 67 72 61 6D 20 41 70 70 6C 69
65 64 59 01 2C 03 54 73 75 64 02 65 6E 31 30 20 70 6F 69 6E
74 73 20 68 61 76 65 20 62 65 65 6E 20 61 64 64 65 64 20 74
6F 20 79 6F 75 72 20 61 63 63 6F 75 6E 74 54 03 65 73 73 72
94 03 03 6F 69 64 04 04 D2 54 03 56 73 75 67 01 99 01 19 03
54 73 75 74 02 65 6E 52 65 77 61 72 64 20 50 72 6F 67 72 61
6D 20 41 70 70 6C 69 65 64 59 01 2C 03 54 73 75 64 02 65 6E
31 30 20 70 6F 69 6E 74 73 20 68 61 76 65 20 62 65 65 6E 20
61 64 64 65 64 20 74 6F 20 79 6F 75 72 20 61 63 63 6F 75 6E

```

74 (See [Get VAS Only Transaction](#) for details.)

1D 44: CRC (little-endian)

Response:

```

56 69 56 4F 74 65 63 68 32 00 02 23 01 A2 11 82 02 00 00 95 05 00 00
00 00 00 9A 03 14 08 10 9C 01 00 5F 2A 02 08 40 9F 02 06 00 00 00 00
00 01 9F 03 06 00 00 00 00 00 9F 06 07 A0 00 00 00 04 10 10 9F 09
02 00 02 9F 1A 02 08 40 9F 1E 08 30 30 30 30 30 30 9F 21 03 13
56 16 9F 33 03 00 00 E8 9F 34 03 00 00 00 9F 35 01 22 9F 36 02 00 00

```

```

9F 37 04 96 B1 71 CF 9F 39 01 91 9F 53 01 00 DF 81 29 08 30 F0 F0 00
30 F0 FF 00 FF 81 06 31 DF 81 2A 18 33 33 30 30 30 33 33 33 30 30 30
32 32 32 32 32 30 30 30 31 31 31 31 30 DF 81 2B 07 90 00 99 00 00 00
0F DF 81 15 06 00 00 00 00 00 FF FF 81 05 74 50 0A 4D 61 73 74 65 72
43 61 72 64 84 07 A0 00 00 00 04 10 10 9F 11 01 01 9F 6D 02 00 01 56
3E 42 35 34 31 33 31 32 33 34 35 36 37 38 34 38 30 30 5E 53 55 50 50
4C 49 45 44 2F 4E 4F 54 5E 31 39 30 36 31 30 31 33 33 39 31 35 33 33
33 30 30 30 32 32 32 32 32 38 32 38 31 31 31 31 38 9F 6B 13 54 13 12
34 56 78 48 00 D1 90 61 01 90 00 99 82 84 82 8F FF EE 01 2F DF 30 01
00 DF 31 18 33 33 30 30 30 33 33 33 30 30 30 32 32 32 32 32 30 30 30
31 31 31 31 30 DF 32 0D 39 30 30 30 39 39 30 30 30 30 30 30 30 FF EE
08 04 DF EF 76 00 DF EF 4C 06 00 27 00 00 00 00 DF EF 4D 27 3B 35 34
31 33 31 32 33 34 35 36 37 38 34 38 30 30 3D 31 39 30 36 31 30 31 39
30 30 30 39 39 38 32 38 34 38 32 38 3F DF EE 26 01 11 75 25
    
```

Parsed Response:

56 69 56 4F 74 65 63 68 32 00: ViVOTech2\0 header

02 23: Command and response

01 A2: Length

11: Attribution byte

82: Application Interchange Profile tag

02: Length

00 00: Value

95: Terminal Verification Results tag

05: Length

00 00 00 00 00: Value

9A: Transaction Date

03: Length

14 08 10: Value

9C: Transaction Type

01: Length

00: Value

5F 2A: Transaction Currency Code

02: Length

08 40: Value

9F 02: Amount, Authorized

06: Length

00 00 00 00 00 01: Value

9F 03: Amount, Other

06: Length

00 00 00 00 00 00: Value

9F 06: Application Identifier (AID)

07: Length

A0 00 00 00 04 10 10: Value

- 9F 09:** Application Version Number
 - 02:** Length
 - 00 02:** Value
- 9F 1A:** Terminal Country Code
 - 02:** Length
 - 08 40:** Value
- 9F 1E:** IFD Serial Number
 - 08:** Length
 - 30 30 30 30 30 30 30 30:** Value
- 9F 21:** Transaction Time
 - 03:** Length
 - 13 56 16:** Value
- 9F 33:** Terminal Capabilities
 - 03:** Length
 - 00 00 E8:** Value
- 9F 34:** CVM Results
 - 03:** Length
 - 00 00 00:** Value
- 9F 35:** Terminal Type
 - 01:** Length
 - 22:** Value
- 9F 36:** Application Transaction Counter
 - 02:** Length
 - 00 00:** Value
- 9F 37:** Unpredictable Number
 - 04:** Length
 - 96 B1 71 CF:** Value
- 9F 39:** POS Entry Mode
 - 01:** Length
 - 91:** Value
- 9F 53:** Terminal Interchange Profile
 - 01:** Length
 - 00:** Value
- DF 81 29:** Outcome Parameter Set
 - 08:** Length
- 30 F0 F0 00 30 F0 FF 00 FF 81 06:** Discretionary Data (Group Tag)
 - 31:** Length
- DF 81 2A:** Track 1 Discretionary Data
 - 18:** Length
 - 33 33 30 30 30 33 33 33 30 30 30 32 32 32 32 30 30 30 31 31 31 31 30:** Value
- DF 81 2B:** Track 2 Discretionary Data
 - 07:** Length

- 90 00 99 00 00 00 0F:** Value
- DF 81 15:** Error Indication
 - 06:** Length
 - 00 00 00 00 00 FF:** Value
- FF 81 05:** Data Record (Group Tag)
 - 74:** Length
- 50:** Application Label
 - 0A:** Length
- 4D 61 73 74 65 72 43 61 72 64:** Value
- 84:** Dedicated File Name
 - 07:** Length
 - A0 00 00 00 04 10 10:** Value
- 9F 11:** Issuer Code Table Index
 - 01:** Length
 - 01:** Value
- 9F 6D:** Kernel Reader Capabilities
 - 02:** Length
 - 00 01:** Value
- 56:** Track 1 Data
 - 3E:** Length
 - 42 35 34 31 33 31 32 33 34 35 36 37 38 34 38 30 30 5E 53 55 50 50 4C 49 45 44 2F 4E 4F 54 5E 31 39 30 36 31 30 31 33 33 39 31 35 33 33 33 30 30 30 32 32 32 32 32 38 32 38 31 31 31 31 38:** Value
- 9F 6B:** Track 2 Data
 - 13:** Length
 - 54 13 12 34 56 78 48 00 D1 90 61 01 90 00 99 82 84 82 8F:** Value
- FF EE 01:** Group Tag for Clearing Record
 - 2F:** Length
- DF 30:** Track Data Source
 - 01:** Length
 - 00:** Value
- DF 31:** DD Track 1
 - 18:** Length
 - 33 33 30 30 30 33 33 33 30 30 30 32 32 32 32 32 30 30 30 31 31 31 31 30:** Value
- DF 32:** DD Track 2
 - 0D:** Length
 - 39 30 30 30 39 39 30 30 30 30 30 30 30:** Value
- FF EE 08:** Masked Tags
 - 04:** Length
- DF EF 76:** NDEF
 - 00:** Length (no Value)
- DF EF 4C:** MSR Equivalent Data Length Values (for data returned in DFEF4D)

06: Length
00 27 00 00 00 00: Value
DF EF 4D: MSR Equivalent Data
27: Length
3B 35 34 31 33 31 32 33 34 35 36 37 38 34 38 30 30 3D 31 39 30 36 31 30 31 39 30 30 30 39 39 38 32 38 34 38 32 38 3F: Value
DF EE 26: Attribution
01: Length
11: Value
75 25: CRC

9.3. Push VAS Only Activate Transaction

Command:

```
56 69 56 4F 74 65 63 68 32 00 02 40 00 CA 30 9F 02 06 00 00 00 00 00
01 9C 01 00 FF EE 08 81 B8 DF EF 1A 01 06 DF EF 1B 81 AE 14 03 67 73
73 72 94 03 05 6F 69 64 04 98 76 67 89 54 03 56 73 75 67 01 99 01 19
03 54 73 75 74 02 65 6E 52 65 77 61 72 64 20 50 72 6F 67 72 61 6D 20
41 70 70 6C 69 65 64 59 01 2C 03 54 73 75 64 02 65 6E 31 30 20 70 6F
69 6E 74 73 20 68 61 76 65 20 62 65 65 6E 20 61 64 64 65 64 20 74 6F
20 79 6F 75 72 20 61 63 63 6F 75 6E 74 54 03 3B 6E 73 72 01 99 01 15
03 54 6E 73 74 02 65 6E 4D 79 20 6C 6F 79 61 6C 74 79 20 70 72 6F 67
72 61 6D 59 01 15 03 55 6E 73 75 00 65 78 61 6D 70 6C 65 2E 63 6F 6D
2F 76 61 6C 75 61 62 6C 65 1C 31
```

Parsed command:

56 69 56 4F 74 65 63 68 32 00: ViVOTECH2\0 header
02 40: Command (Activate Transaction)
00 CA: Length of payload
30: Timeout value
9F 02: Amount, Authorized
 Length
00 00 00 00 00 01: Value
9C: Transaction Type
01: Length
00: Value
FF EE 08: Configuration Container Tag
81: Overflow (because length > 127) and "length of length"
B8: Length
DF EF 1A: Terminal Mode
 Length
06: Value (0110: Push VAS Only)
DF EF 1B: Push Service NDEF Record

81: Overflow flag and "length of length"

AE: Length

14 03 67 73 73 72 94 03 05 6F 69 64 04 98 76 67 89 54 03 56 73 75 67 01 99 01 19 03 54 73 75
 74 02 65 6E 52 65 77 61 72 64 20 50 72 6F 67 72 61 6D 20 41 70 70 6C 69 65 64 59 01 2C 03 54
 73 75 64 02 65 6E 31 30 20 70 6F 69 6E 74 73 20 68 61 76 65 20 62 65 65 6E 20 61 64 64 65 64
 20 74 6F 20 79 6F 75 72 20 61 63 63 6F 75 6E 74 54 03 3B 6E 73 72 01 99 01 15 03 54 6E 73 74
 02 65 6E 4D 79 20 6C 6F 79 61 6C 74 79 20 70 72 6F 67 72 61 6D 59 01 15 03 55 6E 73 75 00 65
 78 61 6D 70 6C 65 2E 63 6F 6D 2F 76 61 6C 75 61 62 6C 65 (See [Get VAS Only Transaction](#) for
 details.)

1C 31: CRC (little-endian)

Response:

56 69 56 4F 74 65 63 68 32 00 02 57 00 1A 01 FF EE 08 04 DF EF 76 00
 9F 39 01 07 FF EE 01 04 DF 30 01 00 DF EE 26 01 01 8A 23

Parsed response:

56 69 56 4F 74 65 63 68 32 00: ViVOTech2\0 header

02 57: Command and response

00 1A: Length of payload

01: Attribution byte

FF EE 08: Container

04: Length

DF EF 76: NDEF Record

00: Length (no Value)

9F 39: POS Entry Mode

01: Length

07: Value

FF EE 01: Group Tag for Clearing Record

DF 30: Track Data Source

01: Length

00: Value

DF EE 26: Attribution Byte

00: Length

01: Value

8A 23: CRC

9.4. Encrypted VAS Only Activate Transaction

Command (NEO firmware):

56 69 56 4F 74 65 63 68 32 00 02 40 00 1B 30 9F 02 06 00 00 00 00 00 01 9C 01 00 FF EE 08 0A DF EF
1A 01 02 DF ED 28 01 00 F4 19

Encryption Key parameters:

BDK: 0123456789ABCDEFEDCBA9876543210

KSN: 629949012C0004600001 (will come back in response in tag FFEE12)

Response:

56 69 56 4F 74 65 63 68 32 00 02 57 00 91 01 FF EE 12 0A 62 99 49 01 2C 00 04 60 00 01 FF EE 08 6D
DF EF 76 C1 68 53 F1 80 5D 0E 8B B5 37 E7 28 EB D2 E6 C7 6F 34 17 33 BE 5C 9C 54 82 76 0D CC 9C D3
38 94 75 70 83 79 B9 7A 79 ED 09 FB EC 74 76 D3 72 72 B4 14 F3 98 DB CD CC 78 23 51 76 06 F7 EE B9
8D D9 AF 69 13 D5 6A E7 BE EE F9 FB 60 BC 75 AD 98 FE EB F2 7B 41 48 2A 74 9E 49 D6 7F A1 AB 2A
BD 7D 8D CD 15 E7 0B EE 06 06 BB 9F 39 01 07 FF EE 01 04 DF 30 01 00 DF EE 26 01 01 AA F9

56 69 56 4F 74 65 63 68 32 00: ViVOtech2\0 header

02 57: Command and response

00 91: Length of payload

01: Attribution byte

FF EE 12: KSN

0A: Length

62 99 49 01 2C 00 04 60 00 01: Value of KSN

FF EE 08: NDEF Container

6D: Length

DF EF 76: VAS Data (Encrypted with DEK)

C1: means the contents are encrypted, '1' is the length of the length

68: Length

53 F1 80 5D 0E 8B B5 37 E7 28 EB D2 E6 C7 6F 34 17 33 BE 5C 9C 54 82 76 0D CC 9C D3 38 94

75 70 83 79 B9 7A 79 ED 09 FB EC 74 76 D3 72 72 B4 14 F3 98 DB CD CC 78 23 51 76 06 F7 EE

B9 8D D9 AF 69 13 D5 6A E7 BE EE F9 FB 60 BC 75 AD 98 FE EB F2 7B 41 48 2A 74 9E 49 D6 7F

A1 AB 2A BD 7D 8D CD 15 E7 0B EE 06 06 BB: Value of VAS Data

9F 39: Point of Service (POS) Entry Mode

01: Length

07: Value (Contactless EMV)

FF EE 01: Container tag

04: Length of container payload

DF 30: Track Data Source

01 00: Length (01) and Value (00)

DF EE 26: Attribution byte

01: Length

01: Value

AA F9: CRC

Note that using a BDK of **0123456789ABCDEFFEDCBA9876543210** and a KSN of **629949012C0004600001** results in a one-time DUKPT session (Data) key of **AA9C25D7FE17CFC88033197D0304AEB3** (use the free [ID TECH decryption tool](#) to derive DUKPT session keys and decrypt data).

The encrypted NDEF payload (from tag FF EE 08), after decryption with the one-time key, will appear as follows:

```
FF EE 08 66 DF EF 76 62 94 03 2F 61 73 76 94 01 06 69 04 02 71 79 79
71 54 03 1F 63 75 73 94 03 06 63 69 64 04 12 34 56 78 90 19 01 03 03
54 63 70 6C 00 7A 68 54 03 02 63 75 74 04 7B 54 03 27 61 73 76 94 01
05 69 05 01 F7 97 98 54 02 19 6C 79 94 03 09 6F 69 64 04 DC 14 8D 1A
6F A8 BB 4A 54 01 06 6E 05 F3 24 23 42 34 00 00 00 00 00 00
```

The correct native length is shown after FF EE 08 as **66**. This is the length of the native data; it makes it possible to parse out (remove) the 00 padding bytes at the end, which were added to facilitate block-ciphered TDES encryption.

9.5. Simplified Output

Example:

```
56 69 56 4F 74 65 63 68 32 00 02 01 00 1B 30 9F 02 06 00 00 00 00 00
01 9C 01 00 FF EE 08 0A DF EF 1A 01 02 DF ED 28 01 00 69 77
```

Response with DFEF77 set to 0:

```
324234242
```

Response with DFEF77 set to 1 and delimiter set to 0x0D (CR):

```
324234244<CR>324234240<CR>324234238<CR>324234241<CR>324234237<CR>32423
4236<CR>324234243<CR>324234235<CR>324234242<CR>324234239<CR>324234234<
CR>
```

In this example, there are 11 services objects: Loyalty "324234234", Offer "324234235", Offer "324234236", Offer "324234237", Offer "324234238", Offer "324234239", Offer "324234240", Offer "324234241", Offer "324234242", Offer "324234243", Offer "324234244".

10. APPENDIX A: ECC Key Pair

Merchants or other administrators who wish to use SmartTap must create and manage the Elliptical Curve Cryptography (ECC) key pair used to for securing communication between the reader and the wallet.

Public Key: administrators must communicate the public key to Google. It is public and can be visible to anyone.

Private Key: the private key must be kept private and injected into the ViVOPay device, where it will be stored securely.

10.1. How to Create an ECC Key Pair Using Open-SSL

Users have several options for generating the ECC key pair (or the ECDSA digital signature key pair). The example below uses the freely available OpenSSL package to generate a prime256v1 Elliptical Curve Cipher key pair (and to sign messages).

To generate EC private key:

```
openssl> ecparam -out PRIVATE.key.pem -name prime256v1 -genkey
```

To generate EC public key from private key:

```
openssl> ec -in PRIVATE.key.pem -pubout -out PUBLIC.key.pem -conv_form compressed
```

Sign message:

```
openssl> dgst -sha256 -sign LONG_TERM_PRIVATE.pem message.txt > signature.bin
```

Verify message:

```
openssl> dgst -sha256 -verify LONG_TERM_PUBLIC.pem -signature signature.bin message.txt
```

Generate ECDH shared secret:

```
openssl> pkeyutl -derive -inkey TERMINAL_EPHEMERAL_PRIVATE.pem -peerkey  
HANDSET_EPHEMERAL_PUBLIC.pem -out secret.bin
```

Revision History

Version	Date	By	Comment
Rev. A	8/10/2018 8/15/2018	KT	Initial draft of public version. Disclaimers regarding firmware differences.
Rev. B	8/30/2018 8/31/2018	KT	Clarifications of various security-related items. Add updated Service Byte definitions.
Rev. C	11/30/2018	KT	Include discussion of tag DFED3F for encryption of VAS data separate from financial data.
Rev. D	12/14/2018	KT	Specify that tags with no defaults should exist, but should be empty.
Rev. E	12/21/2018	KT	Add info about UID in tag DFED44.
Rev. F	12/28/2018	KT	Change examples to use 04-03 instead of 04-00.
Rev. G	1/4/2019	KT	Misc. clarifications.
Rev. H	1/7/2019	KT	Clarify that empty TLVs need not be sent.
Rev. I	1/10/2019	KT	5.2.1 on UID and FFEE0E is removed. DFED3F default if 00. It goes in Group 8E.
Rev. M	10/10/2019	CB	<ul style="list-style-type: none"> • Format for current branding. • Revised text for current style.
Rev. N	10/24/2019	CB	<ul style="list-style-type: none"> • Completed a more substantial style revision. • Expanded command descriptions to include command and response frames. • Added basic configuration flow.
Rev. O	02/13/2020	CB	<ul style="list-style-type: none"> • Added Automatic Output for Auto Poll (01-0D) command. • Removed commands/text related to private key commands.
Rev. P	02/24/2020	CB	<ul style="list-style-type: none"> • Removed specific references to the command for setting LTPK.
Rev. Q	03/23/2020	CB	<ul style="list-style-type: none"> • Updated Set Data Output Mode; Google Smart Tap transactions should output NDEF records tag DFEF76, instead of tag 9F27.
Rev. R	04/16/2020	CB	<ul style="list-style-type: none"> • Removed proprietary NDEF data. • Added Push Smart Tap Data Format section. • Added Service Object NDEF Record Types section. • Updated example 9.4. Encrypted VAS Only Activate Transaction to indicate that the DFEF76 VAS data is encrypted, not the entire FFEE08 NDEF container. Also updated encrypted NDEF example in same section.