Book đọc đi kèm: - Book A

* Book B.

Chapter 1: general information

Chapter 2: model, the basic for architecture of the POS system.

Chapter 3: configuration.

Chapter 4: organization of kernel data.

Chapter 5: commands and responses

Chapter 6 and 7: processing

Chapter 8: security processing.

POS system: chỉ Reader và terminal.

Kernel chứa một set of functions cung cấp tất cả quá trình xử lý logic và data cần thiết để thực hiện giao dịch..

* Chức năng của Terminal:

Business logic để xác định transaction amount và transaction type: transaction amount thường thị sẽ được xác định trước giao dịch và là một con số cố định.

Online authorization và transaction logging:

Processing final outcome.

Phân tích nội dung của card: security check để xác nhận tính đúng đẵn và ủy quyền của data trong card.

Merchant input and display.

* Chức năng của reader:

Giao tiếp với card.

Tin nhắn led/audio, language, có thể bao gồm cả message. Selection of card application.

Complete interaction with card.

Quản lý data exchange giữa kernel và terminal: cho phép kernel gửi tagged data và yêu cầu từ terminal.

Xử lý outcome mang lại bởi kernel: kernel chỉ ra transaction sẽ approved offline, declined offline

Kernel activation

CVM selection,

Management of “no card response” timeout.

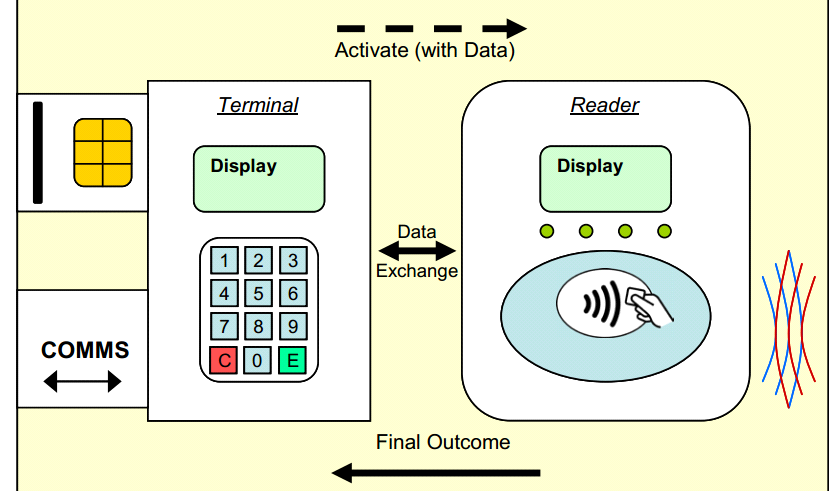
* Simple payment transaction

Require only the exchange of two signals:

ACT signal: để activate Reader, chứa các thông số như: transaction amount, transaction type. Trong một vài trường hợp thì không cần signal này nếu như reader được config là transaction automatically starts after previous transaction.

The OUT signal: bao gồm có các thông tin: approved, online request, declined, end of application. CVM được áp dụng cho terminal,

**READER and TERMINAL**



Cả reader hay terminal đều có display.

Câu hỏi: Reader database

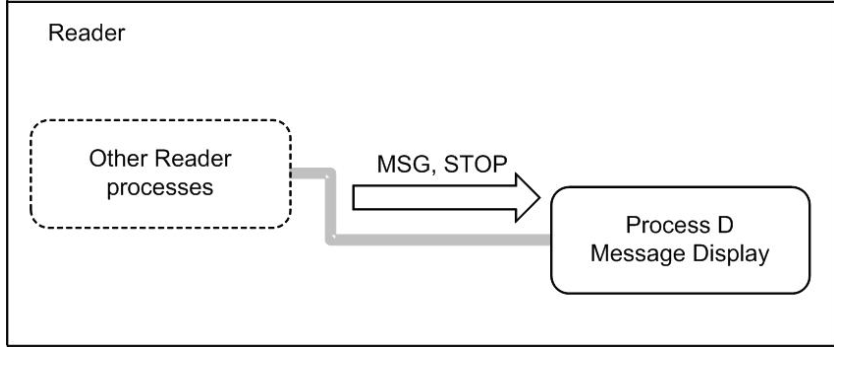
|  |  |
| --- | --- |
| Process M | Kiểm soát tổng thể và trình tự các quá trình. Configuration và activation, quá trình outcome. |
| Process D | User interface |
| Process S | Chọn ứng dụng của card. |
| Process P | Contactless interface |
| Process K | Giao tiếp với card một khi selection ok |

* Process P

Bao gồm reset field, polling, error indicator, removal procedure, collision detection,

* Process D

Quảng lý user interface request cụ thể là thông báo.



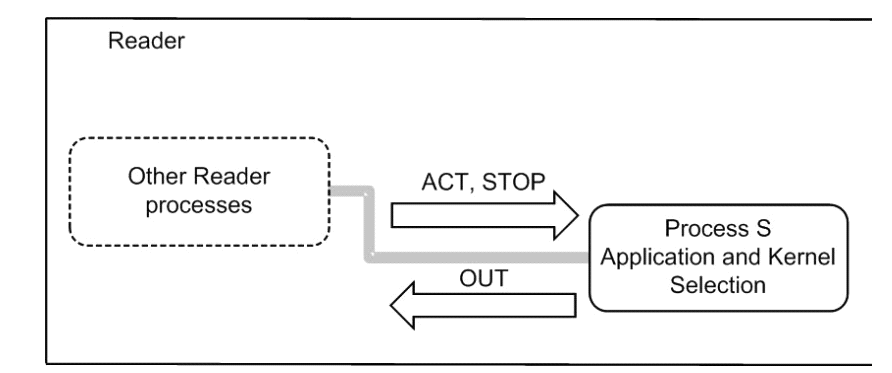
STOP signals sẽ clear display.

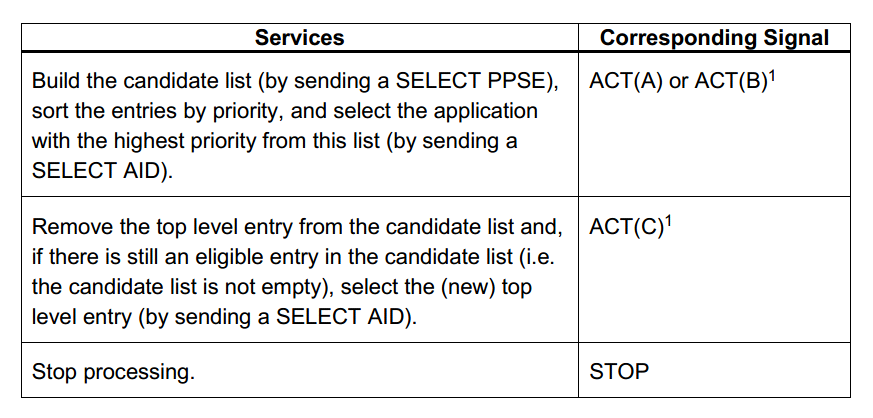
MSG, STOP signal sẽ không acknowledge.

User interface request data bao gồm: message indetifier (các message theo section 9.4 book A), language, balance hay amount .

* Process S

Quản lý việc chọn ứng dụng. Upon activation, ứng dụng được chọn sẽ chứa trong form của AID, kernel ID ở OUT signal





//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Preprocessing

transaction bắt đầu với quá trình này

Mỗi reader combination {AID Kernel ID} sẽ chứa các thông tin CVM, limits, online/offline capability, contactless transaction limits.

Các giá trị này sẽ được kiểm chứng ở quá trình pre processing.

Nếu TTQ được config cho combination, nó sẽ được chỉnh sửa trong quá trình processing. Nếu như TTQ được config thì entry point reset byte 2, bit 8 và bit 7 ở copy of TTQ xuống 00b. (online cryptgram is not required và CVM not required. )

Nếu giá trị của Amount, Authorised =0 và Nếu zero amount allowed flag is present và zero amount allowed =0 thì set “ contactless application not allowed”.

* Terminal contactless transaction limit

Là maximum transaction amount mà trên mức này transaction sẽ không được thực hiện.

* Terminal CVM required limit

Master card introduced new rules for cardholder verification and receipt requirement. Với mức transaction amount nhỏ hơn mức này thì không cần phải có cardholder verification.

Chức năng này chỉ áp dụng cho mỗi CVM và quá trình xử lý biên lai. Không ảnh hưởng tới ủy quyền transaction online hay offline. Online authorization vẫn có thể được yêu cầu nếu terminal action analysis results trong một ARQC request to the card. (nếu card trả lại một ARQC đến TC request).

Terminal phải hỗ trợ NO CVM. Quá trình online transaction thấp hơn hay bằng terminal CVM required limit không yêu cầu một CVM.

* Terminal contactless floor limit

Transaction amount trên mức này sẽ được ủy quyền bởi nhà cấp phát.

Nếu như thấp hơn hoặc bằng mức này:

Transaction có thể đc ủy quyền offline nếu card chấp nhận (returns TC).

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* file within ICC

Files are seen from terminal. Mỗi nhánh của tree này được coi là: Application Definition File (ADF) hay Directory Definition File (DDF). ADF là entry point cho một hay nhiều AEF.

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* kernel state diagram

* S1 idle

reader có thể được config như transaction tự bắt đầu dựa theo transaction trước đó ACT signal dung để activate reader, chứ các thông số transaction amount hay transaction type. Trong một vài trường hợp, ACT signal có thể không cần thiết và complete.

Khi reader nhận được act signal, nó sẽ activate the field và bắt đầu polling card. (terminal gửi ACT signal tới reader).

TLV là viết tắt của Tag, length, value.

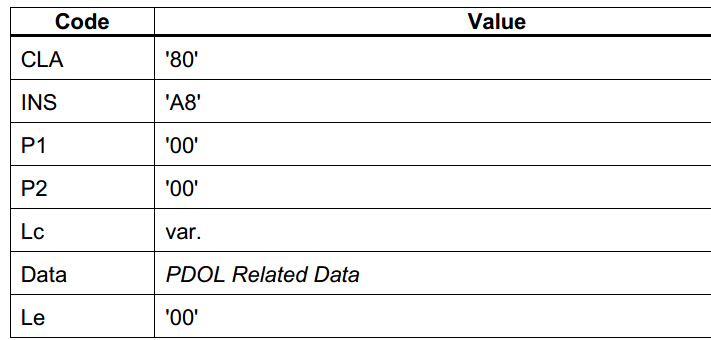
Processing Options Data Object List (PDOL)

(chứa một list terminal data object (tags và length) mà ICC cần cho quá trình GET PROCESSING OPTIONS command. ) được gửi từ PICC. Chẳng hạn như amount authorized, terminal capability, transaction type, DS request Operator ID.

Câu hỏi tags to read liên quan gì với pdol??

FCI template là byte string available ở trả lời của Select command. Tag: 6F.

Get Processing Options command khởi tạo transaction trong card.



PDOL chứa trong GPO command này. (PDOL là giá trị được trả về từ PICC trong khung trả lời của SELECT command). Nếu PDOL không chứa trong response to SELECT command thì data field sẽ k chứa gì.

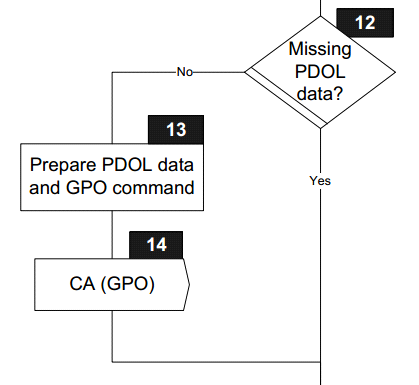
DF name ở tag 84. (của 6F) in select response.

DF name bắt buộc phải có.

Sau khi đã parse các transaction data trong act signal vào tlv database thì kernel khởi tạo unpredictable number.

* Chú ý: signal và command là khác nhau.

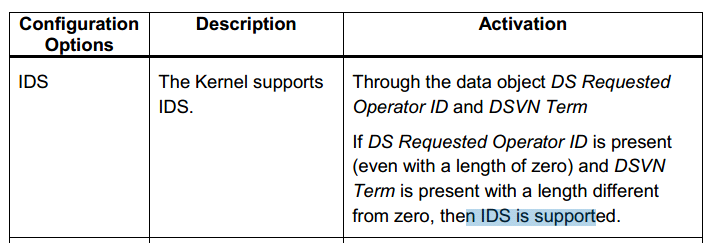
Signal là giữa reader và terminal. Còn command là giữa reader và card.



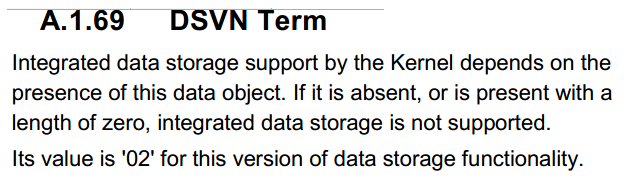
|  |
| --- |
| The parameter to the Signal is the command to be sent to the Card |

nghĩa là thông số trong CA signal này là GPO command.

IDS được supported bởi cả card và kernel nếu như IDS related data (9F6F, 9F5F, 9F7F, 9F7D, 9F54)



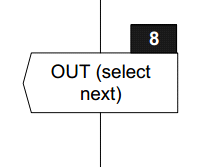
It is sent to the Card ở trong GPO command.



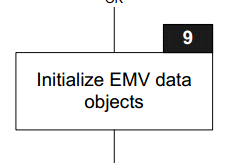
Any application identifier (AID, see 8.2.1.2) may be used as DF name.

* Chú ý: DF là dedicate file (file chuyên dụng).
* EF là elementary file.

Mỗi DF name phải unique trong card.



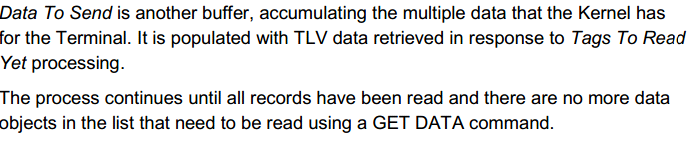
Add to list (GetTLV(tag of Error Indication), Discretionary Data)). Sau đó send Out signal (được gửi từ kernel).



Generate unpredictable number và lưu vào trong TLV database.

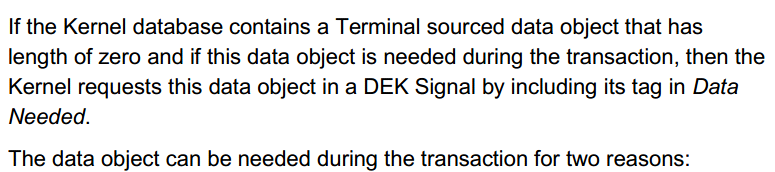
Terminal unpredictable number được gửi từ terminal đến card. (được tạo bởi terminal).

Data to send là một buffer khác. Chứa các list of data objects được gửi từ kernel to terminal trong DEK signal.



Data needed, kernel chứa buffer để tích lũy các tags mà kernel cần từ terminal.

Thông thường, kernel sử dụng DEK signal để yêu cầu data để hoàn thành transaction.



Tags to read:

|  |
| --- |
| List of tags indicating the data the Terminal has requested to be read. This data item is present if the Terminal wants any data back from the Card before the *Data Record*. |
| Nghĩa là: terminal sẽ gửi tags và length cho kernel, nó yêu cầu đọc mấy tags đó. |

Tags to Read yet:

Danh sách các tags và data mà kernel đã nhận được ở tags to read (được terminal gửi qua DET signal) để chuẩn bị chuyển đi. . nếu tag nào đã dc gửi từ kernel sẽ dc xóa đi.

Ban đầu mới vào, quá trình khởi tạo thì: add Tags to read -> Tags to read yet.

Trong phần request data: Nếu như data objects của (tags to read: value của nó là các tags mà terminal yêu cầu đọc) hiện diện ở Kernel database với value field set to zero, kernel gửi DEK signal để yêu cầu giá trị của tags to read. DF8112

**ĐƯỢC KERNEL GỬI ĐẾN TERMINAL QUA DEK signal.**

**Và terminal gửi qua kernel qua DET signal.**

**9F38 : PDOL được gửi từ card.**

Tags to write before gen ac:

Data terminal request to be sent to card.

Câu hỏi: Read and write là sao ta?

Write là cái terminal muốn gửi cho card.

Read là cái terminal muốn nhận. từ card.

Hai cái trên đều thông qua thg trung gian là kernel.

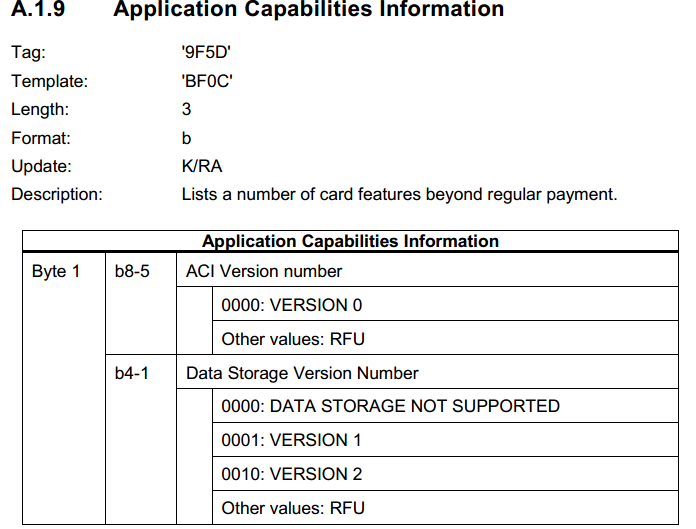
Amount, File Control Information

Câu hỏi: data needed, data to send liên hệ giữa kernel và reader bằng signal hay sao?

Through ACT hay DET signal, kernel có data object (tags to read) chứa các tags (và length) của data object được gửi cho terminal.

Application Capabilities information:

Chứa các card features dựa vào thanh toán thông thường. Chứa trong File Control Information Template.



Câu hỏi: TLV database là gì?

Kernel maintains a TLV database để lưu trữ tất cả TLV encoded data object.

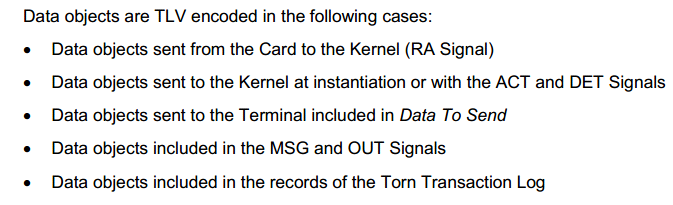
Nó là bản copy của bộ dữ liệu Kernel thích hợp với transaction type và AID. TLV database này được update trong quá trình transaction. Thông qua data from reader, terminal,

Đối tượng dữ liệu được nhận biết bởi kernel nếu tag được liệt kê ở Annex A.

Các data object are transaction related data objects sent to Kernel by terminal with ACT and DET signal.

* Chú ý: AID giống FID.

RA signal chứa R-APDU hoặc SW12. Dược trả lời từ card. Nếu SW12 khác 9000 chỉ ra rằng card không xử lý Generate AC command ở torn transaction.



CDA :



in CDA, the card generates a dynamic signature and an application cryptogram together.



9F5C: DS request operator: được nhận về ở PDOL. Nghĩa là card cần data của tag này.

9F5E: DS ID, TLV của DS ID được gửi về trong SELECT COMMAND response.

trạng thái 1: tương ứng với đợi select ppse response.

bên kernel sẽ nhận được ACT signal từ terminal.

Mục đích: khởi tạo. kiểm tra bên Card có gửi PDOL data cho kernel không, nếu như có thì chuyển trạng thái, nếu k thì tiếp tục đợi. kiểm tra terminal có cần tag nào để đọc từ kernel không.

những cái chú ý trong này: kiểm tra PDOL data flag, kiểm tra IDS supported bởi card hay reader không? rồi set cờ lên. Khởi tạo, chuẩn bị các tag: tags to write before/after - > yet

kiểm tra các tags (DS ID) application capabilities information.

Nếu reader hỗ trợ IDS thì sẽ có tag: 9f5c (DS request), card hỗ trợ IDS thì sẽ có tags (Application capabilities information)

trạng thái 3 chờ GPO response

GPo response có data field tuân theo hai format.

kernel nhận RA signal chứa data field của GPO response.

Câu hỏi: trạng thái 10 của trạng thái 3.

Mục đích: Kiểm tra application file locator, application interchange profile để xác định **EMV mode hay mastripe mode ở tag 82**, On device CVM hay no CVM và kiểm tra RRP có supported hay không? (kết quả hiện ở TVR).

tiếp theo: kiểm tra xem có gửi get data command trước hay là gửi read record command.

Thêm các IDS related data -> data to send để gửi cho terminal.

clear IDS read flag nếu như k có các data trong IDS related data.

gửi DEK signal (chứa các IDS related data)

cuối cùng là kiểm tra CDA có hỗ trợ không (Application interchange profile and TC).

trạng thái s3,r1 common processing:

xem xét là gửi get data hay là read record command. thêm các IDS related data vào data to send. để gửi DEK signal

kiểm tra kernel và card có hỗ trợ CDA.

Trạng thái 4: đợi Read record command response

**CDOL chưa trong Readrecord response. (Read Record response chứa rất nhiều thông tin).**

kiểm tra SW12 , xem xét Record number ở Application file locator để biết sẽ phải gửi bao nhiêu Read record command. và hiện tại record number đang là mấy.

Kiểm tra có gửi get data command hay không?

nếu như k còn active AFL nữa thì ta có thể next command = none.

parse

Xem xét CDOL, DSDOL có ở read record response không thì đưa vào data needed.

Câu hỏi: s4.35

Câu hỏi: ppms\_de5

SFI: Short File Identifier.

SFI xuất hiện ở trong GPO response.

5 bit đầu tiên chỉ SFI.

Byte tiếp theo chỉ record number được đọc cho SFI đó.

Byte tiếp theo nữa chỉ last record number được đọc cho SFI đó. Nếu như last record number > 2nd byte thì all records ranging from second number và bao gồm cả byte thứ 3 sẽ được đọc cho SFI đó. Ví dụ: record number: 01, last record number la 04. READ RECORD command sẽ được gửi 4 lần.

Trạng thái Terminate on next RA

trong trường hợp parsing bị lỗi mà next command không phải none.

trạng thái đợi get data command response

xem xét Active tag (tag to read yet) để quyết định command tiếp theo là read record, get data hay none command.

sau khi nhận dc response -> parse . sau đó đưa TLV vào data to send.

**GET DATA command được gửi khi kernel kiểm tra các tag trong bảng 5.16 có available ở card không (card có thể gửi TLV về). Nếu như có tag nào thì Get Data command sẽ được gửi.**

**Câu hỏi: sao biết nó available ở card?**

trạng thái wait first write flag:

kernel nhận được DET signal -> update tlv database. xem tags to read yet để quyết định command tiếp theo là get data command hay là none.

vậy lúc nào cũng gửi DEK sau read record?

Câu hỏi: làm sao để nhận được time out signal?

trạng thái s456 common processing:

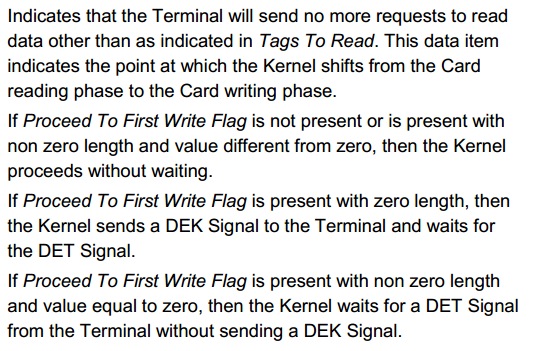
Áp dụng cho trường hợp không còn gửi read record command nữa.

Sau khi guwir read record (lúc sau không cần gửi nữa),

Mục đích: kiểm tra các tags mandotory: amount authorised, application pan, Ca public key index, issuer public key certificate, exponent. Kiểm tra CVM required không để xuất hóa đơn.

Và kiểm tra có torn transaction không.

các cái sử dụng trong này: Kiểm tra proceed to first write flag ( trong trường hợp nxt cmd= none) , giá trị của nó có khác '00' không?



quyết định kernel có gửi DEK signal hay không ( sau Read record command, DEK sẽ được gửi và chuyển trạng thái)

tiếp theo: kiểm tra amount authorised, max trans amount limit, IDS read flag?

Nếu IDS read is set? ta or PAN với PAN sequence number == Ds ID?

Kiểm tra tags to read yet để biết có gửi DEK signal không?

Nếu như CDA supported thì kiểm tra các Issuer public key, icc public key và kiểm tra static data authentication tag list? (tag 9f4a)

Kiểm tra CVM limit để biết xuất hóa đơn hay không

process pre-gen ac balance reading, processing restrictions, cvm selection, terminal action analysic

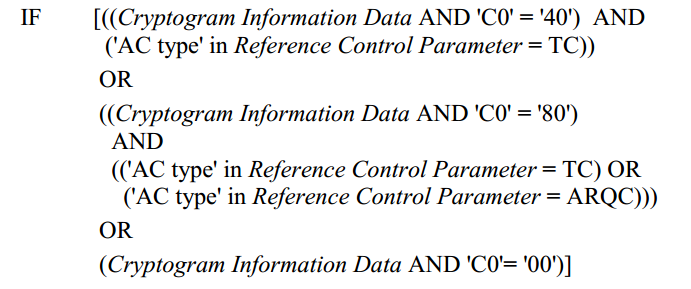
kiểm tra tags to write before **gen AC?? để quyết định sẽ gửi PUT data command hay là generate AC** (trong trường hợp recovery not supported hay kk có torn transaction), nếu có torn transaction thì sẽ gửi Recover Ac

trạng thái wait gen ac response:

Mục đích: nhận RA signal, parse and store card response (data field) bao gồm cryptogram information data (9F27), application transaction counter (9F36) , application cryptogram (9F26), issuer application data (9F10).

Quá trình: Nếu nhận được L1RSP signal thì xem xét transaction recovery supported? để biết mà prepare new record for torn transaction log.

Nếu nhận được RA signal thì parse và store các tag ở trên. bắt buộc phải có tag: ATC, CID



Kiêm tra tags to write after gen AC có không?

Kiểm tra SDAD có không? nếu k có thì là no CDA còn nếu có thì sẽ là CDA

Trạng thái chờ recover AC:

Mục đích: trong trường hợp torn transaction.

ta sẽ copy data torn temp record ở TLV database.

kiểm tra SW12 = 9000 -> no : gửi generate ac

yes: parse response updata tlv database

kiểm tra ATC, CID và xem xét CID có valid không giống ở trên.

Post-gen ac balance reading

kiểm tra tags to write after gen AC có không?

kiểm tra SDAD có không

trạng thái s9,10 common processing **xem lại s910.39**

**Ac type và CDA signature request đều ở trong Reference control parameter.**

- CDA

Đầu tiên retrieve Issuer public key và ICC public key.

kiểm tra IDS read data flag. (có hay không) đều kiểm tra RRP có performed không. Nếu có thì có thêm check data relay, bình thường thì verify SDAD, retrieve AC, DS summary 2, 3 . IDS read flag k set thì k retrieve DS summary 2, 3.

Trong trường hợp Read IDS flag is set thì phải kiểm tra có sự hiện diện của DS summary 2 không. Phải có DS summary 2 = 1.

Sau đó, kiểm tra Write IDS flag, nếu nó set thì kiểm tra DS summary 3 có không? ( k có là lỗi). -> kiểm tra DS summary 2 = DS summary 3?? nếu k bằng thì set Read in DS status. Nếu bằng thì kiểm tra Stop if write failed ở DS info.

- No CDA

Kiểm tra Application cryptogram (k có là lỗi)

nếu Cryptogram information data and 'C0' = 00 thì kiểm tra Read IDS flag? (nếu set thì lỗi), k set thì xem 'AC type" ở reference control parameter = AAC không? (nếu k thì ok ) nếu có thì kiểm tra CDA có k? (có là lỗi) nếu k thì ok

ở trên nếu cryptogram information data and với 'C0' != 00 thì kiểm tra RRp có performed không?

nếu có thì store relay resistant data in track 2

cả hai sau này:

Sau đó set Data record lên.

kiểm tra tagof( POS cardholder interaction information) có empty k?

- Nếu empty thì:

xem cryptogram information data (9f27) and với 'C0' = '40' (thì approved) còn bằng '80' thì là online request. Còn không nữa thì kiểm tra transaction type để ra declined hay là try another interface.

dựa vào cryptogram trên để đưa ra UI request data để đưa ra Balance, authorising please wait hay insert card.

cuối cùng là kiểm tra tags to write after gen ac, nếu có thì gửi PUT data

trạng thái: waiting for generate ac response -2

trong quá trình recover ac response:

SW12 != 9000 thì ta gửi generate ac

Nếu nhận được transmisson error, protocol error, timeout error thì prepare new record for torn transaction. Tiếp sau đó thì tương tự như trên.

Nếu nhận được sw12 = ‘9000’, ta remove record referenced by torn entry from torn transaction log -> copy tlv in torn temp record vào tlv database.

có trạng thái này bởi vì tags to write before gen ac có.

trạng thái chờ PUT DATA before generate ac:

Kiểm tra '9000' ? kiểm tra tags to write before gen AC (nếu vẫn còn thì gửi tiếp PUT data)

Kiểm tra DRDOL (9f51), và max number of torn transaction log recorrds? Nếu k có thì gửi generate data command.

trạng thái chờ PUT data after generate ac:

kiểm tra '9000'? kiểm tra tags to write after gen AC (nếu vẫn còn thì gửi tiếp data)

kiểm tra tagof( POS cardholder interaction information) có empty k? để đưa ra tin hiệu UI request khác nhau.

trạng thái chờ CCC response -1:

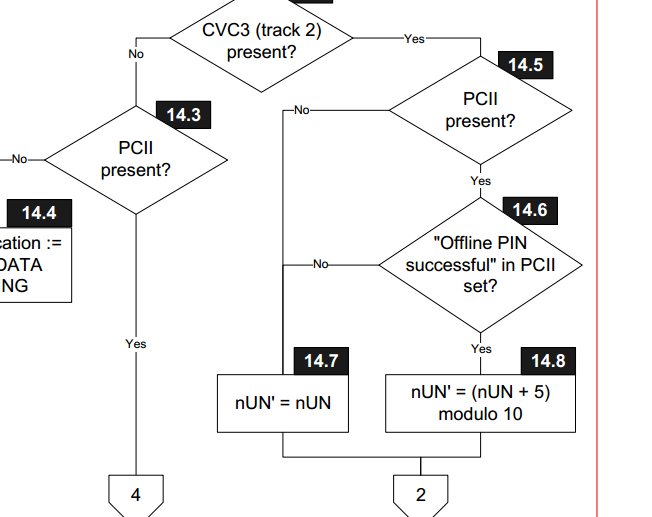
xuất hiện trong trường hợp hỗ trợ stripe và no cardholder verification.

Nhận được L1RSP thì ta chờ 2failed MS Cntr\*300 ms rồi sau đó tăng failed MS cntr lên -> out

nhận được RA -> parse data vào TLV database

Kiểm tra ATC (tag: 9f36) có present hay không? (k thì failed)

Kiểm tra CVC3 (track 2) (9f61) có present hay không? (có thì kiểm tra ở dưới)



kiểm tra *POS Cardholder Interaction Information DF4B* not empty hay sao?

not empty -> OD- CVM verification sucessful ở tag DF4b

kiểm tra track 1 có không (có thì failed)

Copy CVC3(Track 2), UN,

and ATC in Track 2 Data

kiểm tra CVM limit exceed?

-> out signal

trong trường hợp không có CVC3 và có DF4B thì ta sẽ

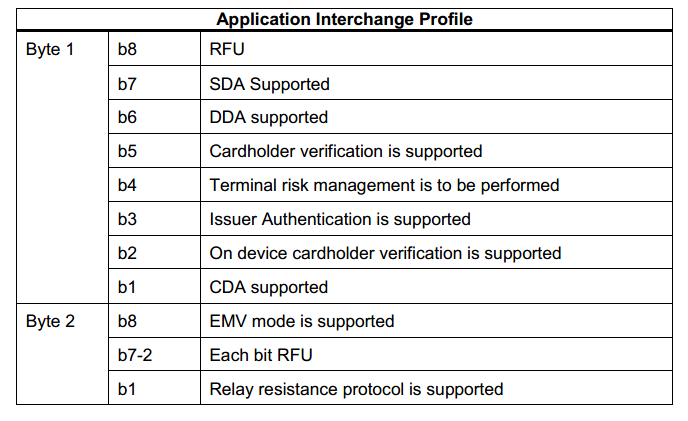
IF [*POS Cardholder Interaction Information* AND '00030F' ≠ '000000']

nếu thỏa thì display phone message.

Câu hỏi: timeout có giá trị là bao nhiêu?

**Isnotempty hay empty là kiểm tra giá trị của tag.**

**Not present hay present là tag đó có xuất hiện hay k**



Câu hỏi: parsing and store là gì?

Câu hỏi: invalid response là sao??

RRP protocol:

Relay attack: the relay fast enough cho terminal không bị timeout. , criminals may run the risk of being caught if they have go around with an antenna or an NFC phone to connect with bankcards people keep in their pockets or handbags.

Sử dụng fraudulent terminal để thực hiện transaction. Trong khi giao dịch thực tế được chuyển tiếp thông qua một thẻ gian lận.

QUÁ TRÌNH RRP:

Bit ở Application interchange profile được sử dụng thông báo reader là card hỗ trợ relay attack, bit ở kernel configuration sử dụng (kernel hỗ trợ).

Reader gửi một C- APDU (exchange relay resistance data) đến card với một số ngãu nhiên. Card trả lời cũng với một số ngẫu nhiên, timing estimates.

Nếu timing exceeds the maximum thì reader sẽ thử lại hai lần nữa.

Kết quả protocol này dựa vào CDA. The timings trả về bởi card ở exchange relay resistance data bao gồm trong signed dynamic application data (response of generate ac command).

Nếu transaction hoàn thanhf without CDA, additional data is included in the online message, issuer sẽ thực hiện check.

The transaction ok nếu như thời gian xử lý nằm trong thời gian cho phép.

Trong trường hợp, k hỗ trợ hai mode emv và stripe, sẽ diễn ra invalid response.

User interface request data: error- other card.

Nếu như bên kernel nhận được DET signal, nó sẽ quay trở lại trạng thái cũ. Đợi tiếp.

Khi mà nhận được RA signal thì nó mới xử lý tiếp.

Terminal verification results ở OUT DATA RECORD.

Câu hỏi: có tags to read rồi mới có tags to read yet đúng k?

câu hỏi:



Các tags ở CDOL hay DSOL ở khung trả lời của Read Record command sẽ được kernel gửi V ở GENRATE\_AC command.

Câu hỏi:



nghĩa là gì?

Câu hỏi: lần 2 nó vẫn gửi 9F50 mà value nó khác.

câu hỏi: từ retrieve nghĩa là gì?

Câu hỏi: cái tag IDS status không thấy ta? sao kiểm tra nhỉ?

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* issuer public key \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

2 loại Dynamic data authentication:

- được thực hiện trước khi card action analysis. ở đó, ICC tạo một digital signature trên ICC-resident/generated data được nhận diện bởi ICC dynamic Data và data nhận bởi terminal được nhận diện bởi DDOL.

Card hỗ trợ dynamic data authentication chứa các data elements:

- CA public key index

- issuer public key certification

- ICC public key certification

- Issuer public key remainder

- issuer public key exponent

- icc public key exponent.

Card sẽ tạo ra Signed Dynamic Application data: sử dụng private key tương ứng với public key đã dc xác nhận ở ICC public key certification.

Câu hỏi: s910.3

Câu hỏi: IDS ???

Outcome: là các lỗi: CAM failed (do signed dynamic application data veritification not ok)

card data missing (trong th IDS read flag is set, DS summary 2 empty)

IDS read error (DS summary 1 != DS summary 2)

CARD data missing ( Write IDS status is set và not present (DS summary 3)) hay Application crytogram is empty. trong trường hợp No cda mà CDA request ở reference parameter is set).

tương tự như trên.

Câu hỏi: khi nào thì có tags to write before/after gen ac?? ở trạng thái đợi GPO response hay Read record response, nếu như kernel nhận được DET thì sẽ update TLV database. (có thể tags to write before ac)

Mastripe mode: mô tả mode hoạt động của pos system, hỗ trợ magnetic stripe mode. Được sử dụng chung với term “transaction” chỉ contactless payment dựa trên Track 1 and or track 2 data.

All MSD Mode transactions must be performed online

Track 1 data (built from data elements within the card) or Track 2 data (obtained from the Track 2 Equivalent Data on the card).

Kernel uses CCC command khiển card generate Dynamic CVC3 values that can be inserted to track 1 and track 2 equivalent data (supplied by the card).đảm bảo track data thay đổi ở mỗi transaction

CTQ (card transaction qualifier) tag: 9F6C

The CTQ is set by the card issuer, xác định các tác vụ thực hiện tại Point of sale khi transaction được thực hiện. bao gồm online pin required, signature required, go online, switch interface,

* If fDDA (authentication check) fails again the CTQ has three choices that it can send the transaction online, switch it to a contact transaction or decline it.

Nếu như CTQ không trả về từ card thì TTQ sẽ quyết định.

TTQ: cũng quy định các chức năng như trên.

**Torn transaction**

Diễn ra khi transaction chưa completed thì card được đưa ra khỏi vùng RF. Kernel sẽ store the transaction details trong khi đợi card’s response to a gen ac command.

The terminal configuration defines a maximum lifetime for each entry in the Torn Transaction Log and the terminal should use a Clean Signal to remove all expired entries from the log.

Support for transaction recovery by the card, with the presence of DRDOL.

Trước khi bắt đầu, ta chọn clean để remove old records from torn transaction log. Nếu như sự khác nhau giữa transaction date, time in record và in tlv database > max lifetime of torn transaction log read -> addtolist (record, dicretion data) -> remove record from old transaction log.

Trong trạng thái s4,5,6 common processing:

Nếu như DRDOL không xuất hiện ở card thì ta gửi generate ac. Nếu như có hỗ trợ, và nếu như torn transaction diễn ra -> ta copy application pan sequence number , application pan vào record. Store reference to record in torn entry. Tiếp theo sẽ copy record referenced bởi torn entry vào torn temp record.

Trong trạng thái chờ gen ac response:

Nếu như transmission error hay protocol error, time out error -> transaction recovery supports. Sử dụng DRDOL để tạo DRDOL related data . -> khởi tạo torn temp record. Lưu TLV của các tag như amount, authorized, amount other, application pan, balance to read before gen ac.

Nếu như number of records in torn transaction log = max number thì ta sẽ copy và thay thế các oldest record bằng torn temp record.

Hiện ở user interface request data là Try again.

Recover ac được gửi nhằm mục đích lấy lại các thông tin:

Mà trong quá trình gen ac lần trước chưa lấy được. Sau đó, kernel kiểm tra (nếu như recover ac không diễn ra bình thường thì ta gửi gen ac luôn). Nếu ok thì remove torn entry từ torn transaction log, và copy data từ torn temp record vào data base. Xác nhận lại các ATC, CID. Các bước sau giống với gen ac

Câu hỏi:  nghi

Quá trình kích hoạt key.

Phần common processing ở states 9 and 10.