

Especificações do Pix por aproximação para Android

Versão 1.0



pix
powered by **Banco Central**

Sumário

1. Visão Geral	3
2. Protocolo.....	4
2.1. Premissas	4
2.2. URI do Pix.....	4
2.3. Diagrama de sequência	5
(1) APDU Select command.....	6
(2) Create NDEF message	6
(3) Split message	8
(4) Update binary	8
(5) Desconexão.....	9
Histórico de revisão	10

1. Visão Geral

O iniciador de transação de pagamento participante do *Open Finance* e do Pix e o provedor de conta transacional participante do Pix são capazes de iniciar transações no Pix a partir da captura de QR Codes via câmera. Neste documento, que integra o Manual de Padrões para Iniciação do Pix, introduz-se o protocolo “*Tap to Pix*”, permitindo utilizar a antena com tecnologia *Near Field Communication* (NFC) como meio de captura do QR Code, em procedimento similar ao procedimento utilizado pelo “Pix Cópia e Cola”.

2. Protocolo

O protocolo de captura de QR Codes via NFC, denominado “*Tap to Pix*”, compreende:

1. a definição do RID/AID a ser utilizado pelo terminal POS com antena NFC;
2. a definição de URI a ser resolvida para iniciação de uma transação Pix; e
3. o diagrama de sequência para a captura via NFC incluindo comandos APDU.

2.1. Premissas

O terminal POS deve possuir capacidade de transmitir conteúdo via NFC para dispositivos Android utilizando o padrão NDEF para transmissão de URIs através de comandos APDU, em acordo com a documentação do Android para a estrutura [NDefRecord](#).

A captura não está restrita a terminais POS, podendo se estender para outros dispositivos compatíveis com NFC, como celulares ou *pin pads*.

Este protocolo é aberto, isto é, o “*Tap to Pix*” pode ser utilizado por quaisquer aplicativos Android ou outros sistemas operacionais compatíveis.

2.2. URI do Pix

O formato esperado de URI para utilização do “*Tap to Pix*” segue o padrão:

`pix://<hostname>?qr=<uri-encoded-emv-qr-string>&sig=<signature>`

Os *placeholders* devem ser substituídos de acordo:

- `<hostname>`: identificação do domínio que disponibilizou o QR Code;
- `<uri-encoded-emv-qr-string>`: conteúdo do Pix Cópia e Cola codificado em URI. Veja a referência [encodeURIComponent\(\)](#).

Parâmetro reservado para uso futuro:

- `<signature>`: parâmetro opcional com a assinatura do código “Cópia e Cola” decodificado. A assinatura deve seguir o algoritmo PS256, conforme padrão atualmente praticado no *Open Finance*. A disponibilização da chave pública para a parte receptora para fins de validação segue sob definição.

Esse formato permite que parâmetros adicionais sejam negociados e incorporados à especificação conforme a evolução do produto.

O tamanho típico da URI é estimado em cerca de 260 bytes para QR Codes dinâmicos, porém há casos válidos de URIs com tamanhos superiores, sobretudo quando se considera o uso de parâmetros adicionais.

2.3. Diagrama de sequência

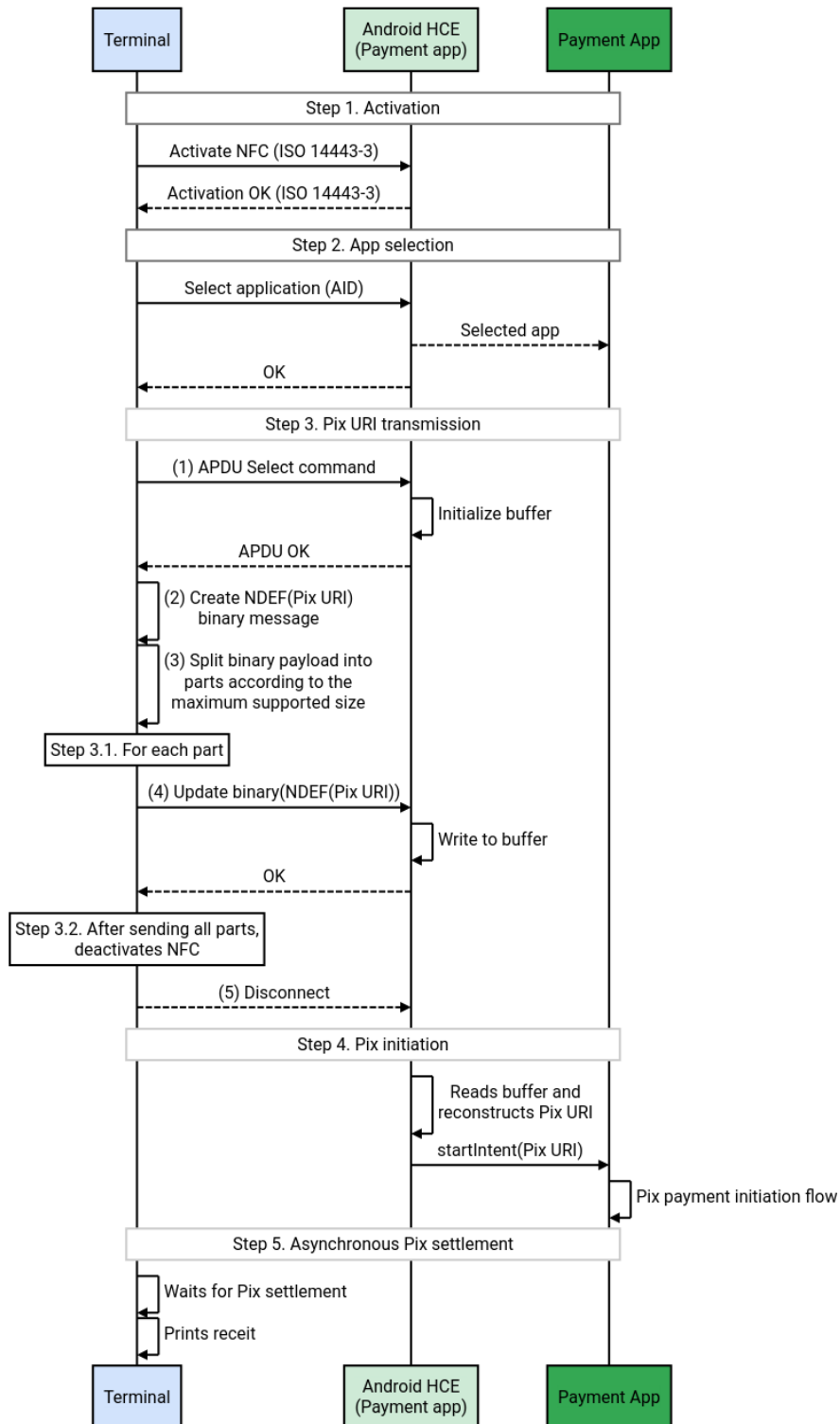


Diagrama 1: Sequência de mensagens trocadas entre Terminal, Serviço NFC do Android e aplicativo de pagamento

A seguir, detalham-se as atividades destacadas no diagrama acima.

(1) APDU Select command

O comando APDU select, como definido na ISO 7816-4. Por exemplo:

Byte (valor)	Descrição
0x00	CLA: Class - Class of instruction
0xA4	INS: Instruction - Instruction code (SELECT)
0x04	Parameter 1: Instruction parameter 1 O valor 0x04 indica seleção por Application ID (AID).
0x00	Parameter 2: Instruction parameter 2 O valor 0x00 indica que o AID será enviado na íntegra.
0x08	Lc field: Número de bytes do identificador AID (0x10 ou 16 bytes).
0xA000000940BCB000	Application ID: Constante, identifica o protocolo "Tap to Pix".
0x00	Le field: Máximo número de bytes esperado na resposta deste comando para o campo "data".

Tabela 1: Comando APDU select

O retorno OK da seleção de aplicativo é dado pela mensagem de 2 bytes: 0x90000x90: SW1: Status byte 1 - Command processing status

0x00: SW2: Status byte 2 - Command processing qualifier

(2) Create NDEF message

A URI do Pix, conforme definida anteriormente neste documento deve ser formatada de acordo com a especificação NDEF (NFCForum-TS-NDEF_1.0), isto é, contendo os cabeçalho e valores TLV que identificam um conteúdo do tipo URI.

A mensagem NDEF deve conter apenas um NDEF Record. Em caso de múltiplos NDEF Records, o aplicativo de pagamentos pode ignorar os Records que não o primeiro ou até mesmo optar por não iniciar o fluxo de pagamentos.

Bibliotecas como [Android NdefRecord.createUri\(String|Uri\)](#) e [NdefMessage.create\(NdefRecord\)](#) servem de referência para criação da mensagem NDEF a partir da URI do Pix, abstraindo os detalhes de cada bit da especificação NDEF.

Abaixo, ilustram-se dois cenários possíveis de cabeçalhos NDEF, com o objetivo de demonstrar a diferença entre mensagens curtas (até 255 bytes de *payload*) e mensagens longas (entre 255 bytes e 32760 bytes - limite superior determinado pelo offset do comando APDU Update Binary e não pela especificação NDEF, que admite valores teoricamente superiores).

Bytes (valores)	Descrição
0xC1	Cabeçalho da mensagem NDef, indicando tamanho estendido (4 bytes para representar o tamanho)
0x01	Type length
0x00 00 00 B8	Tamanho da mensagem NDEF (no exemplo, 162 bytes)
0x55	Tipo da mensagem NDEF (no exemplo, 0x55 corresponde ao tipo URI)
0x00	Tipo de URI (0x00 indica um tipo customizado), contabilizado no tamanho da mensagem NDEF
0x...	Payload no formato URI Pix (no exemplo, 161 bytes)

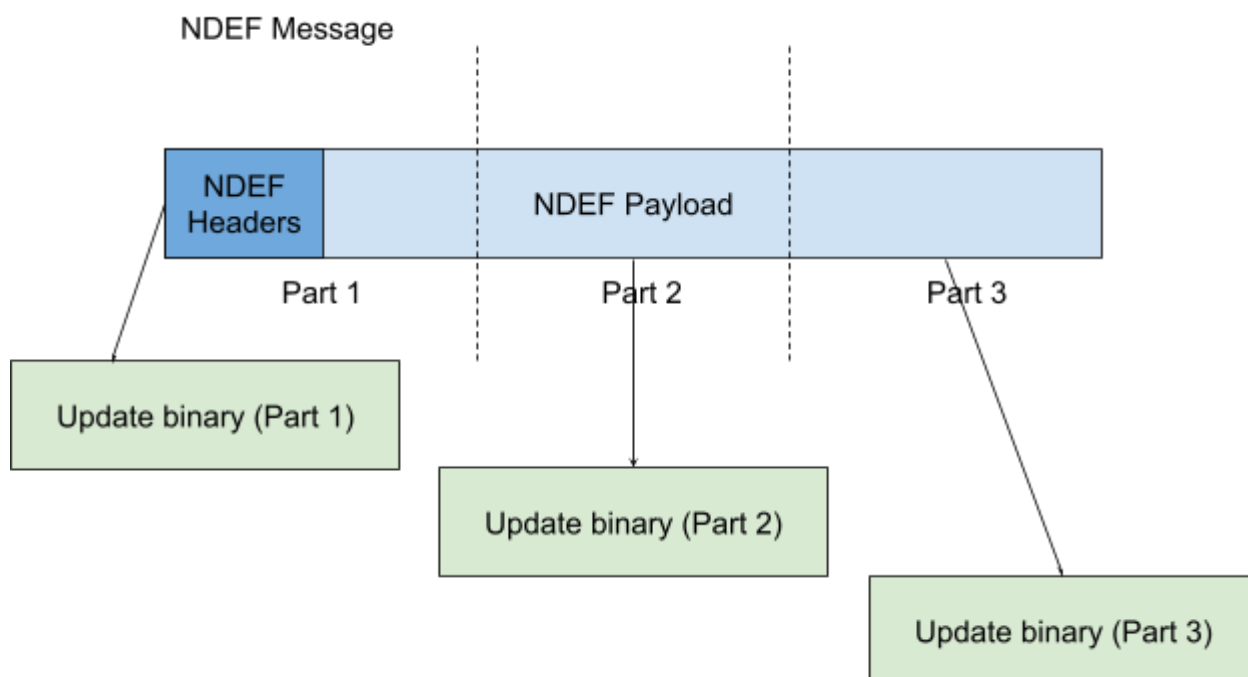
Tabela 2: Montagem da mensagem NDEF com formato longo

Bytes (valores)	Descrição
0xD1	Cabeçalho da mensagem NDef, indicando tamanho curto (1 byte para representar o tamanho)
0x01	Type length
0xB8	Tamanho da mensagem NDEF (no exemplo, 162 bytes)
0x55	Tipo da mensagem NDEF (no exemplo, 0x55 corresponde ao tipo URI)
0x00	Tipo de URI (0x00 indica um tipo customizado), contabilizado no tamanho da mensagem NDEF
0x...	Payload no formato URI Pix (no exemplo, 161 bytes)

Tabela 3: Montagem da mensagem NDEF com formato curto

(3) Split message

Caso o tamanho da mensagem NDEF acima supere o limite máximo que o terminal físico possa transmitir, considerando o *overhead* do comando APDU, é possível realizar a quebra em múltiplas mensagens Update Binary em tantas partes quanto necessárias. Cada parte não deve replicar os cabeçalhos NDEF, isto é, ao recepcionar todas as partes, o serviço NFC do aplicativo de pagamentos deve recombinar as partes e interpretar o conteúdo como apenas uma única mensagem NDEF.



(4) Update binary

O comando *update binary*, definido na ISO 7816-4, deve ser utilizado para a transmissão das partes da mensagem NDEF. Atenção especial aos parâmetros P1 e P2, que são utilizados para controlar a posição do *buffer* em que o serviço NFC “Tap to Pix” deve armazenar o conteúdo parcial.

Ainda, este comando admite duas variações na declaração do tamanho do conteúdo binário, conforme ilustrado nas tabelas 4 e tabela 5.

Bytes (valores)	Descrição
0x00 D6	Cabeçalho do comando Update Binary (ISO 7816-4).
0x00 00	P1 e P2, concatenados bit a bit (15 bits no total), sinalizam o offset, isto é a posição que esta mensagem deve ser escrita no buffer, em relação ao início da primeira mensagem.
0x00 00 BF	Tamanho da mensagem NDEF, a seguir, em formato estendido (3 bytes, com o primeiro byte igual a 0x00). No exemplo, 0x00BF corresponde a 191 bytes.
0x...	Conteúdo binário (parte de mensagem NDEF)

Tabela 4: Comando update binary em formato estendido.

Bytes (valores)	Descrição
0x00 D6	Cabeçalho do comando Update Binary (ISO 7816-4).
0x00 00	P1 e P2, concatenados bit a bit (15 bits no total), sinalizam o offset, isto é a posição que esta mensagem deve ser escrita no buffer, em relação ao início da primeira mensagem.
0xBF	Tamanho da mensagem NDEF, a seguir, em até 255 bytes.
0x...	Conteúdo binário (parte de mensagem NDEF)

Tabela 5: Comando update binary em formato curto.

O retorno OK do comando update binary é dado pela mensagem de 2 bytes: 0x90000x90: SW1:

Status byte 1 - Command processing status

0x00: SW2: Status byte 2 - Command processing qualifier

(5) Desconexão

Em virtude do protocolo “*Tap to Pix*” flexibilizar o uso de um ou mais comandos *update binary* para transmissão da URI do Pix, é imprescindível que o terminal realize a desconexão ao término da transmissão da última mensagem para que o aplicativo de pagamento possa ser iniciado.

Caso contrário, o usuário pagador não receberá resposta visual até que retire o telefone das proximidades do terminal, interrompendo a comunicação fisicamente. Apesar do fluxo de pagamento ainda ser iniciado, a sensação para o usuário é de aparente lentidão e essa falha de experiência é facilmente evitada com a desconexão lógica do terminal.

Histórico de revisão

Data	Versão	Descrição das alterações
05/06/2025	1.0	Versão inicial