



Dynamic Client Registration (DCR) - Perfil único

SN

Owned by [Suporte N2](#) ...

Last updated: jan. 09, 2024 • 13 min read

Prefácio

A Estrutura Inicial do Open Insurance Brasil é responsável por criar os padrões e especificações necessários para atender aos requisitos e obrigações da Legislação do Open Insurance do Brasil, conforme originalmente delineado pela SUSEP. É possível que alguns dos elementos deste documento estejam sujeitos a direitos de patente. A Estrutura Inicial não se responsabiliza pela identificação de qualquer ou todos os direitos de patente.

O Perfil de Segurança Financial-grade API 1.0 do Open Insurance Brasil consiste nas seguintes partes:

- [Open Finance Brasil Financial-grade API Security Profile 1.0](#)
- Open Insurance Brasil Dynamic Client Registration Profile 1.0

Essas partes devem ser usadas com [RFC6749](#), [RFC6750](#), [RFC7636](#), [OIDC](#), [OIDR](#), [RFC7591](#), [RFC7592](#), [FAPI-1-Baseline](#) e [FAPI-1-Advanced](#)

Changelog

Seção	Alteração
6.1 Servidor de Autorização	Exclusão do item 6.1.6 “se suportar OAuth 2.0 Pushed Authorisation Requests, deve...” do antigo documento
6.1 Servidor de Autorização	Item 6.1.5 alterado de “deve anunciar” para “pode anunciar”
6.1 Servidor de Autorização	Adição de “push_authorization_request_endpoint” no item 6.1.5

[6.1 Servidor de Autorização](#)

Adição do item 6.1.7

Área do / **Dynamic Client Registration (DCR) - Perfil único**

<u>7.1 Servidor de Autorização</u>	Exclusão do item 7.1.11 “se for compatível com o mecanismo de autenticação do cliente <i>tls_client_auth...</i> ” do antigo documento
<u>7.1 Servidor de Autorização</u>	Adição dos itens 7.1.16 e 7.1.17
<u>7.1 Servidor de Autorização</u>	Exclusão da subseção 7.1.2 Análise do Distinguished Name do Certificado
<u>7.3 Registro do Cliente</u>	Exclusão da seção
<u>9.3 Sobre os mecanismos de autenticação e autorização dos serviços de DCR e DCM</u>	Exclusão do item 9.3.2.1 “validar que o certificado apresentado...” do antigo documento
<u>9.3 Sobre os mecanismos de autenticação e autorização dos serviços de DCR e DCM</u>	Adição dos itens 9.3.2.2, 9.3.2.3 e 9.3.2.4
<u>9.3 Sobre os mecanismos de autenticação e autorização dos serviços de DCR e DCM</u>	Exclusão da observação “A RFC7592 prevê a possibilidade de rotação do <i>registration_access_token...</i> ”
<u>9.4. Validação de certificados de assinatura</u>	Adição da seção

Introdução

O Perfil de Registro de Cliente Dinâmico (DCR - *Dynamic Client Registration*) do Financial-grade API (FAPI) do Open Insurance Brasil é um perfil de [RFC7591](#), [RFC7592](#) e [OIDR](#) que visa fornecer diretrizes de implementação específicas para segurança e interoperabilidade que podem ser aplicadas à identificação, registro e gerenciamento de *Clients OAuth* operando no ecossistema Open Insurance Brasil.

Embora seja possível codificar um *OpenID Provider* e *Relying Party* desde o princípio usando esta especificação, o principal público para esta especificação são as parte

que já possuem uma implementação certificada do [OpenID Connect](#) e desejam obter a certificação para o Open Insurance Brasil.



As palavras-chave "*deve*" (shall), "*não deve*" (shall not), "*deveria*" (should), "*não deveria*" (should not) e "*pode*" (may) presentes nesse documento devem ser interpretadas conforme as diretrizes descritas em [ISO Directive Part 2](#) observando seguinte equivalência:

- "*deve*" ⇒ equivalente ao termo "shall" e expressa um requerimento definido no documento (nas traduções é similar ao termo "must", que pode denotar um requerimento externo ao documento);
- "*não deve*" ⇒ equivalente ao termo "shall not" e também expressa um requerimento definido no documento;
- "*deveria*" e "*não deveria*" ⇒ equivalente ao termo "should" e "should not" e expressa uma recomendação
- "*pode*" ⇒ equivalente ao termo "may" indica uma permissão

Estas palavras-chave não são usadas como termos de dicionário, de modo que qualquer ocorrência deles deve ser interpretada como palavras-chave e não devem ser interpretados com seus significados de linguagem natural.

1. Escopo

Este documento especifica o método de

- aplicativos cadastrados no [Diretório de Participantes do Open Insurance](#) para descobrir OpenID Providers que oferecem serviços no ecossistema Open Insurance Brasil;
- aplicativos para usar o [OpenID Connect Registration](#) para integrar seus aplicativos com OpenID Providers das seguradoras; e
- aplicativos para usar [OAuth 2.0 Dynamic Client Registration Management Protocol](#) para gerenciar seus aplicativos com OpenID Providers;

Este documento é aplicável a todos os participantes do Open Insurance no Brasil.

2. Referências normativas

Os seguintes documentos referenciados são indispensáveis para a aplicação deste documento. Para referências datadas, apenas a edição citada se aplica. Para

referências não datadas, a última edição do documento referenciado (incluindo quaisquer emendas) se aplica.

Área do / **Dynamic Client Registration (DCR) - Perfil único**



- RFC6749 - [The OAuth 2.0 Authorization Framework](#)
- RFC6750 - [The OAuth 2.0 Authorization Framework: Bearer Token Usage](#)
- RFC7636 - [Proof Key for Code Exchange by OAuth Public Clients](#)
- RFC6819 - [OAuth 2.0 Threat Model and Security Considerations](#)
- RFC7519 - [JSON Web Token \(JWT\)](#)
- RFC7591 - [OAuth 2.0 Dynamic Client Registration Protocol](#)
- RFC7592 - [OAuth 2.0 Dynamic Client Registration Management Protocol](#)
- BCP195 - [Recommendations for Secure Use of Transport Layer Security \(TLS\) and Datagram Transport Layer Security \(DTLS\)](#)
- OIDC - [OpenID Connect Core 1.0 incorporating errata set 1](#)
- FAPI-CIBA - [Financial-grade API: Client Initiated Backchannel Authentication Profile](#)
- RFC4514 - [Lightweight Directory Access Protocol \(LDAP\): String Representation of Distinguished Names](#)
- RFC4517 - [Lightweight Directory Access Protocol \(LDAP\): Syntaxes and Matching Rules](#)
- OIDD - [OpenID Connect Discovery 1.0 incorporating errata set 1](#)
- OIDR - [OpenID Connect Registration 1.0 incorporating errata set 1](#)
- RFC8705 - [OAuth 2.0 Mutual TLS Client Authentication and Certificate Bound Access Tokens](#)
- JARM - [Financial-grade API: JWT Secured Authorization Response Mode for OAuth 2.0 \(JARM\)](#)
- PAR - [OAuth 2.0 Pushed Authorization Requests](#)
- JAR - [OAuth 2.0 JWT Secured Authorization Request](#)
- FAPI-1-Baseline - [Financial-grade API Security Profile 1.0 - Part 1: Baseline](#)
- FAPI-1-Advanced - [Financial-grade API Security Profile 1.0 - Part 2: Advanced](#)
- OFB-FAPI - [Open Finance Brasil Financial-grade API Security Profile 1.0](#)
- OFB-Cert-Standards - [Open Insurance Brasil x.509 Certificate Standards](#)
- OFB-DCR/DCM-Swagger - [DCR & DCM Swagger](#)

3. Termos e definições

Área do / Dynamic Client Registration (DCR) - Perfil único



4. Símbolos e Termos abreviados

- **API** – Application Programming Interface (Interface de Programação da Aplicação)
- **DCR** – Dynamic Client Registration (Registro de Cliente Dinâmico)
- **FAPI** - Financial-grade API
- **HTTP** – Hyper Text Transfer Protocol
- **OIDF** - OpenID Foundation
- **REST** – Representational State Transfer
- **SS** – Software Statement (Declaração de Software)
- **SSA** – Software Statement Assertion (Afirmação de Declaração de Software)
- **TLS** – Transport Layer Security

5. Introdução

O ecossistema Open Insurance Brasil apoia-se em um provedor de confiança ou *diretório de participantes* como a fonte mais valiosa de informações sobre participantes credenciados e softwares que estão autorizados a participar do ecossistema Open Insurance Brasil.

Os serviços do Diretório incluem:

- Registro e gerenciamento de software
- Registro e gerenciamento de credenciais de software usando certificados ICP
- Geração de Software Statement Assertion (SSA)

Os participantes do ecossistema devem aproveitar esses serviços para facilitar o registro de cliente OAuth orientado por API usando o processo descrito na cláusula 3.1.1 do [RFC7591](#) com metadados adicionais necessários para oferecer suporte ao OpenID Connect definido em [OpenID Connect Registration](#).

É importante reforçar que o payload de registro de clientes possui a maior parte de seus atributos não obrigatórios, e que os atributos cujos valores conflitem com os presentes no software statement assertion serão sobrepostos pelos valores do próprio software statement assertion emitido pelo diretório central. Nem todos os metadados que um cliente deseja fornecer podem estar contidos em um software

statement, por exemplo, alternativa Metadata Languages and Script values. Há casos ainda de metadados de cliente que são um subconjunto dos valores existentes no

Área do / Dynamic Client Registration (DCR) - Perfil único



6. Provisionamentos do OpenID Connect Discovery do Open Insurance Brasil

6.1 Servidor de Autorização

O servidor de autorização deve suportar OpenID Connect Discovery conforme exigido pelo Financial-grade API Security Profile 1.0 - Part 1: Baseline. Este suporte deve estar explícito tanto na forma como o Servidor de Autorização está registrado no Diretório de Participantes quanto na declaração dos seus atributos no arquivo de Discovery (well-known), respeitando os mecanismos de autenticação certificados pela instituição através dos testes de conformidade do Open Insurance Brasil.

Adicionalmente, o Servidor de Autorização:

1. deve anunciar sua presença no ecossistema Open Insurance Brasil, sendo listada no Diretório de Participantes;
2. deve anunciar todos os recursos API REST do Open Insurance Brasil protegidos pelo Provedor OpenID no Diretório de Participantes;
3. deve anunciar suporte para todos os mecanismos de assinatura, criptografia, autenticação e padrões necessários para suportar o Open Finance Brasil Financial API;
4. deve anunciar suporte para OpenID Dynamic Client Registration;
5. pode anunciar `mtls_endpoint_aliases` de acordo com a cláusula 5 RFC 8705 OAuth 2.0 Mutual-TLS Client Authentication e Certificate-Bound Access Tokens o `token_endpoint`, `registration_endpoint` e `userinfo_endpoint` e `push_authorization_request_endpoint`;
6. se suportar Financial API - Client Initiated Back Channel Authentication, deve anunciar através de OIDC mtls_endpoint_aliases o `backchannel_authentication_endpoint`;
7. não deve rotacionar o `registration_access_token`.

6.2 Cliente

O cliente deve suportar OpenID Connect Discovery conforme exigido pelo Financial-grade API Security Profile 1.0 - Part 1: Baseline.

Além disso, o Cliente

Área do / Dynamic Client Registration (DCR) - Perfil único



2. deve derivar os metadados necessários do Authorization Server somente por meio do serviço OpenID Connect Discovery dos Authorization Servers;
3. quando presente, deve usar endpoints anunciados em `mtls_endpoint_aliases` conforme a cláusula 5 [RFC 8705 OAuth 2.0 Mutual-TLS Client Authentication e Certificate-Bound Access Tokens](#);

7. Provisionamento de registro OpenID Connect do Open Insurance Brasil

7.1 Servidor de Autorização

O servidor de autorização deve suportar as RFCs de Dynamic Client Registration (DCR) [RFC7591](#), Dynamic Client Management (DCM) [RFC7592](#) e [OpenID Registration](#)

Além disso, o servidor de autorização

1. deve rejeitar as solicitações de registro de cliente dinâmico não realizadas em uma conexão protegida com mTLS usando certificados emitidos pelo Brasil ICP (produção) ou o Diretório de Participantes (sandbox);
2. deve validar que a solicitação contém `software_statement` JWT assinado usando o algoritmo PS256 emitido pelo Diretório de Participantes do Open Insurance Brasil;
3. deve validar que o `software_statement` foi emitido (`iat - issued at`) não mais de 5 minutos antes do pedido ser recebido;
4. deve validar que um atributo `jwks` (definida por valor) **não** foi incluído, e sim declarado como referência no atributo `jwks_uri`;
5. deve, quando informado, validar que o `jwks_uri` corresponda ao `software_jwks_uri` fornecido na declaração do software;
 - a. deve validar se o conteúdo obtido através do `jwks_uri` contém um certificado do tipo “use”: “enc” ;
6. deve exigir e validar que o `redirect_uris` corresponda ou contenha um subconjunto dos valores de `software_redirect_uris` fornecidos no `software_statement`;
7. deve exigir e validar que todos os mecanismos de autenticação de cliente cumpram os requisitos definidos nas [RFC7591](#) e [RFC7592](#), através da validação

do `registration_access_token` e, como conexão segura, da cadeia de certificados confiáveis ICP-Brasil.

Área do / Dynamic Client Registration (DCR) - Perfil único



relação de permissões regulatórias e os escopos correspondentes está descrita nas seções a seguir.

9. deve, sempre que possível, validar os metadados declarados pelo cliente em relação aos metadados fornecidos no `software_statement`, adotando os valores presentes no SSA com precedência.
10. deve aceitar todos os nomes x.500 AttributeType definidas no *Distinguished Name* dos Perfis de Certificado x.509 definidos em [Open Insurance Brasil x.509 Certificate Standards](#);
11. o valor do campo `UID` do certificado deve coincidir com o enviado no SSA, onde o campo `UID` deve conter o valor do campo `software_id` do SSA.
12. o valor do campo `organizationIdentifier` do certificado deve conter o prefixo correspondente ao Registration Reference *OPIBR-* seguido do valor do campo `org_id` do SSA.
13. deve, durante o processo de handshake TLS, usar a regra `distinguishedNameMatch` para comparar os valores DN conforme definido na [RFC4517](#).
14. deve garantir a integridade do estoque de consentimentos ativos, mesmo após eventuais mudanças sistêmicas, para que tais alterações sejam transparentes para as instituições receptora de dados.
15. deve realizar recertificação FAPI e DCR da OIDF após eventuais mudanças sistêmicas.
16. O valor do atributo `software_api_webhook_uris` contido como atributo no JWS em `software_statement` deverá ser comparado com o campo `webhook_uris`. Caso os valores não sejam exatamente iguais, deve-se retornar o erro, com HTTP status code setado para 400, error preenchido como `invalid_webhook_uris` e `error_description` preenchido com “The content of the `webhook_uris` field differs from what was registered in the `software_statement` observed through the JWS field's `software_api_webhook_uris`”.
17. Se o campo `webhook_uris` não for declarado no payload, a funcionalidade `webhook` deverá ser considerado como desativado para o `client` em questão.

Estas disposições aplicam-se igualmente ao processamento de pedidos [RFC7591](#), [RFC7592](#) e [OpenID Registration](#)

7.1.1 Aplicando Server Defaults

Área do / Dynamic Client Registration (DCR) - Perfil único



cliente da seguinte maneira:

1. deve selecionar e aplicar o algoritmo de criptografia e a escolha da cifra a partir dos conjuntos mais recomendados de cifra da IANA que são suportados pelo Servidor de Autorização;
2. deve preencher *defaults* a partir de valores da afirmação de *software_statement*, sempre que possível;
3. deve conceder ao cliente permissão para o conjunto completo de escopos potenciais com base nas permissões regulatórias de softwares incluídas no *software_statement*.

7.2 Funções regulatórias para mapeamentos OpenID e OAuth 2.0

Para participar do ecossistema do Open Insurance, as instituições credenciadas devem se cadastrar no Diretório de Participantes de acordo com seus papéis regulatórios. Essas funções refletem a autorização do SUSEP para as instituições e, consequentemente, as APIs que podem utilizar.

A tabela a seguir descreve as funções regulatórias do Open Insurance e o mapeamento de escopos do OAuth 2.0 relacionado. Se os escopos forem omitidos durante o processo de DCR, o Servidor de Autorização deve conceder o conjunto completo de escopos potenciais com base nas funções regulatórias registradas para a seguradora, conforme descrito na seção Server Defaults.

Papel Regulador	Descrição	Escopos Permitidos (em construção)	Fase-alvo
DADOS	Instituição transmissora / receptora de dados	openid consents resources customers insurance- acceptance-and- branches-abroad insurance-auto insurance- ...	Fase 2

financial-risk
insurance-housing

Área do / Dynamic Client Registration (DCR) - Perfil único



		insurance-rural insurance- responsibility insurance- transport	
ICS	Iniciadora de Compartilhamento de Serviços	openid claim-notification endorsement quote-patrimonial-lead quote-patrimonial-home quote-patrimonial-condominium quote-patrimonial-business quote-patrimonial-diverse-risks	Fase 3
TCS	Transmissora de Compartilhamento de Serviços	openid	Fase 3

É necessário validar as *roles* ativas no *software_statement* da aplicação. Na validação dessa informação deve ser utilizado o campo *software_statement_roles*, e deve ser verificado se as *roles* listadas estão ativas.

8. Declaração de Software

Uma declaração de software (*software_statement*) é um [JSON Web Token \(JWT\)](#), que afirma valores de metadados sobre o software cliente como um todo. Na estrutura do Open Insurance Brasil, esse *software_statement* é assinado pelo Diretório de Participantes, e sua assinatura DEVE ser validada pelos Servidores de Autorizacao usando as chaves públicas disponíveis na seção a seguir.

8.1 Atributos da Declaração de Software (Claims)

Área do / Dynamic Client Registration (DCR) - Perfil único



```
1  {
2    "software_mode": "Live",
3    "software_redirect_uris": [
4      "https://www.raidiam.com/insurance/cb"
5    ],
6    "software_statement_roles": [
7      {
8        "role": "DADOS",
9        "authorisation_domain": "Open Insurance",
10       "status": "Active"
11     }
12   ],
13   "software_client_name": "Raidiam Insurance",
14   "org_status": "Active",
15   "software_client_id": "Cki1EbvjwyhPB12NGL1z2",
16   "iss": "Open Insurance Brasil prod SSA issuer",
17   "software_tos_uri": "https://www.raidiam.com/insurance/tos.html",
18   "software_client_description": "Raidiam Insurance leverage cutting ed
19   "software_jwks_uri": "https://keystore.directory.opinbrasil.com.br/b9
20   "software_policy_uri": "https://www.raidiam.com/insurance/policy.html
21   "software_id": "25556d5a-b9dd-4e27-aa1a-cce732fe74de",
22   "software_client_uri": "https://www.raidiam.com/insurance.html",
23   "software_jwks_inactive_uri": "https://keystore.directory.opinbrasil.
24   "software_jwks_transport_inactive_uri": "https://keystore.directory.o
25   "software_jwks_transport_uri": "https://keystore.directory.opinbrasil
26   "software_logo_uri": "https://www.raidiam.com/insurance/logo.png",
27   "org_id": "b961c4eb-509d-4edf-afeb-35642b38185d",
28   "org_number": "112233445566",
29   "software_environment": "production",
30   "software_version": "1.1",
31   "software_roles": [
32     "DADOS"
33   ],
34   "org_name": "Open Insurance Brasil",
35   "iat": 1620060821,
36   "organisation_competent_authority_claims": [
37     {
38       "authorisation_domain": "Open Insurance",
```

```
39     "authorisations": [],
40     "registration_id": "13353236-0IB-DADOS",
Área do / Dynamic Client Registration (DCR) - Perfil único
43     "authorisation_role": "DADOS",
44     "authority_code": "SUSEP",
45     "status": "Active"
46   }
47 ]
48 }
49
```



9. Processamento de solicitação de registro de cliente dinâmico

Área do / Dynamic Client Registration (DCR) - Perfil único



9.1 Enviar uma solicitação com uma declaração de software

Área do / Dynamic Client Registration (DCR) - Perfil único



obrigatória, consultar o [Swagger DCR](#). A questão de linha dentro dos valores são apenas para fins de exibição.

```
1 POST /reg HTTP/1.1
2 Host: auth.raidiam.com
3 Content-Type: application/json
4 {
5   "application_type": "web",
6   "grant_types": [
7     "client_credentials",
8     "authorization_code",
9     "refresh_token",
10    "implicit"
11  ],
12  "id_token_signed_response_alg": "PS256",
13  "require_auth_time": false,
14  "response_types": [
15    "code id_token",
16    "id_token"
17  ],
18  "software_statement": "eyJraWQiOiJzaWduZXIiLCJ0eXAiOiJKV1QiLCJhbGciOiJQI
19  "subject_type": "public",
20  "token_endpoint_auth_method": "private_key_jwt",
21  "request_object_signing_alg": "PS256",
22  "require_signed_request_object": true,
23  "require_pushed_authorization_requests": false,
24  "tls_client_certificate_bound_access_tokens": true,
25  "client_id": "aCnBHjZBvD6ku3KVBas1L",
26  "client_name": "Raidiam Insurance",
27  "client_uri": "https://www.raidiam.com/insurance.html",
28  "request_object_encryption_alg": "RSA-OAEP",
29  "request_object_encryption_enc": "A256GCM"
30  "jwks_uri": "https://keystore.directory.opinbrasil.com.br/b961c4eb-509d
31  "redirect_uris": [
32    "https://www.raidiam.com/insurance/cb"
33  ]
34 }
```

9.2 Open Insurance Brasil SSA Key Store e detalhes do emissor

Área do / Dynamic Client Registration (DCR) - Perfil único



apresentados durante o processo de registro de cliente.

Produção

<https://keystore.directory.opinbrasil.com.br/openinsurance.jwks>

Emissor do Open Insurance Open Insurance Brasil SSA de produção

Sandbox

<https://keystore.sandbox.directory.opinbrasil.com.br/openinsurance.jwks>

Emissor do Open Insurance Open Insurance Brasil SSA de sandbox

9.3 Sobre os mecanismos de autenticação e autorização dos serviços de DCR e DCM

Por serem serviços auxiliares ao fluxo principal do Open Insurance Brasil, os serviços de registro e manutenção dinâmica de clientes não utilizam os mesmos mecanismos de controle de acesso. Por exemplo: não é possível exigir um *access_token* OAuth 2.0 de uma aplicação cliente que ainda não está registrada na instituição transmissora. Para estender as [RFC7591](#) e [RFC7592](#), que recomendam mecanismos mínimos para autenticação dos seus serviços, as instituições que suportam os fluxos de registro e manutenção dinâmica de clientes devem implementar em seus Servidores de Autorização os controles a seguir:

9.3.1 Registro de cliente - POST /register

1. validar que o certificado apresentado pela aplicação cliente é subordinado às cadeias do ICP-Brasil definidas no Padrão de Certificados do Open Insurance Brasil;
2. assegurar que a assinatura do *software_statement* apresentado pela aplicação cliente durante o registro tenha sido feita pelo Diretório de Participantes através das chaves públicas descritas na seção anterior;
3. assegurar que o *software_statement* apresentado pela aplicação cliente durante o registro corresponda à mesma instituição do certificado de cliente apresentado, validando-o através dos atributos que trazem *organization_id* no certificado X.509.

4. emitir, na resposta do registro, um `registration_access_token` para ser usado como token de autenticação nas operações de manutenção da aplicação cliente

Área do / **Dynamic Client Registration (DCR) - Perfil único**



Statement, de forma que em caso de tentativa de novo registro para um Software Statement já cadastrado, deve se utilizar o procedimento de Error Response definido no item 3.2.2 da [RFC7591](#).

9.3.2 Manutenção de cliente - GET /register - PUT /register - DELETE /register

1. validar a presença e a correspondência do header `Bearer Authorization` contendo o valor do atributo `registration_access_token` retornado durante o registro do cliente correspondente.
2. quando o método chamado for `PUT /register`, realizar as validações dos subitens 1, 2, 3 e 5 do item Registro de cliente - `POST /register`.
3. quando o método chamado for `GET /register`, realizar as validações dos subitens 1 e 5 do item Registro de cliente - `POST /register`.
4. quando o método chamado for `DELETE /register`, realizar a validação do subitem 1 do item Registro de cliente - `POST /register`.

9.4. Validação de certificados de assinatura

- O diretório realiza a validação do certificado de assinatura através da função `cert rescan`, a cada hora.
- As instituições devem assegurar que o processo de validação é realizado.
- Cada instituição deve ter alternativa de contingência em caso de indisponibilidade do serviço de validação realizado pelo diretório.
- Ao identificar que o certificado de assinatura não é válido pois, está com status revogado de acordo com o OCSP/CRL da CA emissora, ou está inativo no cadastro do diretório, o conjunto de chaves públicas é movido para o repositório de chaves inativas (Inactive Keystore).
- É recomendado que o processo de validação inclua:
 - Validação da assinatura da mensagem do Transmissor de Dados, a ser feita pelo Receptor de Dados
 - Validar se a mensagem está assinada conforme o Message Signature Guidelines, incluindo se o `iss` é igual ao `organisation_id` do servidor que emitiu a mensagem.

- Buscar a declaração iss do JWT e gerar URI do JWKS publicado no diretório, para consulta.

Área do / **Dynamic Client Registration (DCR) - Perfil único**



- Validar se a chave privada para o kid correspondente é capaz de validar a assinatura da mensagem.
- Validação da assinatura da mensagem do Receptor de Dados, a ser feita pelo Transmissor de Dados
 - Validar se a mensagem está assinada conforme o Message Signature Guidelines.
 - Obter o org_jwks_uri que foi apresentado no SSA pelo cliente no momento do DCR.
 - Certificar se o kid do cabeçalho JWT da mensagem está presente no diretório JWKS.
 - Validar se a chave privada para o kid correspondente é capaz de validar a assinatura da mensagem.

