Controles Técnicos de Segurança para Open Insurance

Apresentação dos Controles Técnicos de Segurança

Sumário

Introdução

O objetivo desse documento é apresentar o detalhamento dos controles técnicos de segurança que serão implementados na arquitetura do Open Insurance. Em sua essência, o Open Insurance é um ecossistema de compartilhamento de dados e iniciação de serviços onde os clientes de seguradoras e outras instituições financeiras desejam compartilhar suas informações de conta ou dar permissão para que os pagamentos sejam executados em seu nome com serviços de terceiros.

Há uma série de funções necessárias para vincular qualquer sistema de identificação, autenticação e autorização, independentemente do setor. Todas essas funções são necessárias, mas várias funções podem ser desempenhadas por cada participante. Em geral, o usuário final *("Subject")*, está dando a um sistema *("Client")* uma autorização *("Access Token")* para acessar um recurso protegido mantido pelo provedor *("Resource Server")*. Isso exige que o *Subject* e o *Client* sejam identificados e autenticados e que a autorização seja confirmada.

As regras exatas e os requisitos legais para cada função em um setor específico formam um *framework* de confiança *("Trust Framework")*. Cada ecossistema requer um conjunto padronizado de regras e requisitos legais que abrangem todas as funções e obrigações das interações acima. A combinação de quem fornece qual(is) função(ões), os níveis aos quais eles devem desempenhar essas funções e os padrões pelos quais essas operações devem ser definidas por um *framework* de confiança específico do setor.

Diferentes *framework*s de confiança terão diferentes opções de implementação, mas um *framework* de confiança comum é um pré-requisito para transformar um 'setor' em um 'ecossistema'. Um *framework* de confiança comum reduz significativamente a complexidade e custos, aumenta a escalabilidade e a interoperabilidade dentro do setor, bem como abre opções para o tipo de padronização intersetorial que o Open Insurance está buscando.

Diferentes implementações podem ser definidas para setores, com diferentes prós / contras e custos associados para diferentes participantes. Cada uma das implementações propostas pode ser usada para qualquer setor se os pré-requisitos corretos estiverem em vigor. A solução certa dependerá do apetite e alinhamento de cada conjunto de participantes.

A implementação de um mecanismo comum para o Open Insurance exigirá um compromisso com a simetria entre os setores para incluir detalhes específicos do setor nos princípios do *framework* de confiança.

É necessário fazer escolhas técnicas para garantir que qualquer implementação forneça uma base estrita e consistente para ter credibilidade, mas mantenha a flexibilidade para se adaptar às necessidades futuras. Isso implica padrões de código-fonte aberto amplamente disponíveis, amplamente compreendidos e que foram experimentados e testados. Além de habilitar um gama de parceiros e fornecedores que podem apoiar qualquer construção técnica, o que significa que continuará havendo espaço para desenvolvimento comercial de soluções.

Participantes de um ecossistema de compartilhamento de dados e iniciação de serviços

Nos ecossistemas de Open Insurance voltados para o consumidor que estamos considerando, temos três participantes principais:

- o cliente *(user)*;

- a instituição transmissora de dados *(provider)*, que oferece serviços de seguros; e
- a instituição receptora de dados *(TPP - Third Party Provider)*, que oferece uma proposta de Open Insurance para o cliente.

Em todos os casos a seguir, assumimos:

- Um cliente possui uma conta para um serviço principal ou conjunto de recursos numa instituição transmissora de dados;
- Uma instituição receptora de dados oferece ao cliente uma proposta habilitada por meio do compartilhamento inteligente de dados;
- O cliente dá consentimento à instituição receptora de dados para fins de entrega dessa proposta; e
- A instituição transmissora de dados tem a obrigação de salvaguardar os dados do cliente, mas também de compartilhá-los quando instruído.

O ecossistema também possui provedores de serviços de confiança, que são entidades que fornecem garantia técnica a ambas instituições (transmissoras e receptoras) de que todos estão autorizados a participar do ecossistema.

Os padrões técnicos necessários para dar suporte ao *framework* de confiança devem atender todos os requisitos a seguir:

- Identificação de todos os participantes do ecossistema;
- Autenticação quando exigida de todos os participantes entre si; e
- Confirmação de autorização de todos os participantes em um ecossistema de compartilhamento de dados e iniciação de serviços.

Os serviços técnicos necessários para suportar um ecossistema devem habilitar todos os requisitos acima em uma base e modo contínuos, isto é, não apenas em um único ponto de registro.

Princípios de Especificação e Requisitos de Alto Nível

O Open Insurance adotou os seguintes princípios e requisitos de alto nível no que diz respeito às normas técnicas.

- **Consentimento:** Os clientes devem estar sempre no controle de quem tem acesso aos seus dados e para quais fins eles estão sendo usados.
- **Minimização de dados:** Os clientes devem ser capazes de compartilhar apenas os dados de que precisam, pelo tempo que for necessário.
- **Segurança:** Uma modelagem de ameaças foi produzida avaliando todas as fraquezas potenciais nos processos de comunicação. Todos os pontos fracos identificados foram corrigidos.
- **Identificação:** Todos os participantes devem ter segurança na identificação de todos os atores do ecossistema.
- **Autenticação:** Todos os participantes devem comunicar as etapas que foram executadas para autenticar cada participante no ecossistema e em que nível isso foi executado.
- **Integridade e não repúdio:** Todos os participantes devem ser capazes de provar que as mensagens não foram adulteradas e, na verdade, foram enviadas apenas por um participante legítimo.
- **Transparência:** Todos os participantes devem ser capazes de prover informações claras, precisas e facilmente acessíveis sobre a realização do tratamento dos dados do cliente e os respectivos agentes de tratamento.

Além dos requisitos de alto nível, os seguintes princípios também foram adotados.

Apresentação dos Controles Técnicos de Segurança

- Não reinventar a roda, se existir uma especificação que seja adequada para o propósito, amplamente adotada e publicamente disponível, deve-se adotá-la.
- Envolver-se com outros órgãos de normalização para aprender com experiências anteriores sobre o que funcionou, o que não funcionou, e o que pode ser feito melhor.
- Assegurar o amplo suporte da indústria para garantir o máximo de chances de sucesso e, mais importante, a segurança do cliente.
- Solicitar *feedback* com antecedência e com frequência, reconhecer que serão necessárias várias iterações para desenvolver um padrão.
- O *framework* de confiança que sustenta o ecossistema de compartilhamento de dados, que é o Open Insurance, é um *framework* técnico que precisa ser flexível o suficiente para permitir que os participantes e o ecossistema inovem, cresçam e se desenvolvam, enquanto permanecem interoperáveis.

Todos os participantes devem ter certeza de que todos os atores do ecossistema estão lidando com seus dados com segurança tempo todo. Isso requer que todos os participantes testem publicamente seus sistemas quanto à conformidade com as especificações e disponibilizem os resultados de seus testes de conformidade para exame público de outros participantes.

Padrões de Segurança

Autenticação e Autorização

> **Observações:**
>
> É importante ressaltar que este controle foi proposto seguindo os padrões do Open Banking e ele não é aplicável para o escopo *open-data* (fase 1) por se tratar de uma API pública com dados públicos. Adicionalmente, este controle está relacionado com os itens 4.21, 4.22, 4.23 e 7.3 do Manual de Segurança publicado pela SUSEP.

Estrutura de Autorização OAuth 2.0

O ecossistema de compartilhamento de dados e iniciação de serviços definido pelo Brasil consiste em muitos padrões diferentes, todos girando em torno de conceitos, funções e obrigações que foram tecnicamente definidos no OAuth 2.0 *Authorization Framework*.

A estrutura de autorização OAuth 2.0 permite uma aplicação de terceiros *(third-party application)* obter acesso limitado a um serviço HTTP, seja em nome do proprietário de recurso *(resource owner)* por meio da orquestração de uma interação de aprovação entre o proprietário do recurso e o serviço HTTP, ou permitindo a aplicação de terceiros obter acesso em seu próprio nome.

A especificação base OAuth 2.0 não fornece, por si só, informações suficientes para atender a todas as necessidades definidas pelo *framework* de confiança do Open Insurance Brasil. Mais notavelmente, não possui uma maneira de transmitir informações de identidade do cliente em um formato padronizado de uma instituição transmissora para uma receptora, e os mecanismos de autenticação que foram definidos na especificação original não são seguros o suficiente para atender aos requisitos de uma indústria altamente regulamentada.

OpenID Connect – A Camada de Identidade para a Internet

Este perfil herda todas as obrigações do OAuth 2.0

OpenID Connect é um conjunto de especificações simplificadas que fornecem uma estrutura para interações de identidade por meio de APIs do tipo REST. A implementação mais simples do OpenID Connect permite que *clients* de todos os tipos, incluindo baseados em navegador, celulares e *clients javascript*, solicitem e recebam informações sobre identidades e sessões atualmente autenticadas. O conjunto de especificações é extensível, permitindo que os participantes também suportem, opcionalmente, criptografia de dados de identidade, descoberta do OpenID Provider e gerenciamento avançado de sessão, incluindo logout.

O grupo de trabalho OpenID Foundations Connect tem sido o guardião do Padrão de Identidade "de fato" da Internet por muitos anos, trabalhando em várias especificações que se baseiam no *framework* de autorização OAuth 2.0, adicionando recursos e requisitos de suporte para melhorar a segurança do *framework* em si.

Open ID Connect Core: é um perfil do OAuth 2.0, o que significa que herda todos os requisitos e obrigações do OAuth 2.0, mas define o conceito de um *id_token* e introduz novos mecanismos de autenticação.

Open ID Connect Discovery: apresenta o conceito de um documento de descoberta usado por OpenID Connect (OIDC) Providers para anunciar como os *clients* OAuth 2.0 podem se comunicar com eles e quais recursos e opções o OIDC Provider oferece suporte.

RFC7591: além de definir o processo de registro dinâmico de *clients* OAuth, esta especificação apresenta o conceito de *Software Statement* ("Declaração de *Software*"), que pode ser usada para fornecer informações sobre um *client* que é atestado por um serviço de terceiros. Outros atributos de metadados também são definidos no OpenID Connect Registration Specification.

Esta especificação define mecanismos para registrar dinamicamente *clients* OAuth 2.0 com *Authorization Servers* (servidores de autorização). Pedidos de registro enviam um conjunto de valores de metadados do *client* desejado para o *Authorization Server*. As respostas de registro resultantes retornam um *client identifier* para ser usado no *Authorization Server* e os valores de metadados registrados para o *client*. O *client* pode então usar esta informação de registro para se comunicar com o *Authorization Server* usando o protocolo OAuth 2.0. Esta especificação também define um conjunto de campos de metadados do *client* e valores para os *clients* usarem durante o registro.

RFC7592: Esta especificação define métodos de gerenciamento de *Dynamic Client Registration* (registros de cliente dinâmico) do OAuth 2.0 para casos de uso em que as propriedades de um *client* registrado necessitam ser alteradas durante a vida do *client*.

As especificações acima são especificações básicas cuja leitura obrigatória sustenta o *framework* de confiança do Open Insurance. Entretanto, eles ainda são insuficientes para atender a todos os requisitos e princípios descritos anteriormente.

Financial-grade API Security Profile

Financial-grade API (FAPI) desenvolvido pela OpenID Foundation, é uma especificação técnica, também considerado como um padrão técnico de implementação de software altamente seguro que visa fornecer as diretrizes de implementação específicas para segurança e interoperabilidade que podem ser aplicadas a APIs na área de Open Insurance.

O padrão utiliza como base outros dois padrões citados acima, são eles: OAuth 2.0 e OpenID Connect (OIDC). Estes padrões são a base da especificação FAPI, que combinado com outros artefatos de segurança, torna possível atender os altos padrões de segurança exigidos pela indústria financeira, porém também podendo esse ser estendido para outras indústrias com padrões de segurança similares.

Para mais informações sobre a certificação FAPI consulte o manual "Processo de certificação FAPI" no Portal do Desenvolvedor. Seguradoras e provedores certificados podem ser consultados no seguinte link: Certified FAPI OpenID Providers na seção Brazil Open Insurance.

Perfil de Segurança para o Open Insurance

O perfil de segurança do Open Insurance especifica requisitos adicionais de segurança e de identificação para o acesso a API's com recursos críticos protegidas pelo OAuth 2.0 *Authorization Framework*, que consiste em RFC6749, RFC6750, RFC7636, FAPI-1-Baseline, FAPI-1-Advanced e outras especificações.

- Este perfil descreve as capacidades e os recursos de segurança que devem ser oferecidos por servidores e clientes que são necessários para o Programa do Open Insurance, definindo as medidas para mitigar ou endereçar:
- ataques que abordam considerações de privacidade identificadas na cláusula 9.1 de [FAPI-1 Advanced].
- o requisito de concessão de acesso granular a recursos, com vistas à minimização de dados;
- o requisito de informar sobre o contexto da autenticação do usuário (*claim Authentication Context Request* - ACR) que foi realizada por um Provedor OpenID, com vistas a favorecer o adequado gerenciamento do risco decorrente do acesso do usuário;
- o requisito para que os clientes de API declarem um relacionamento prévio com o usuário, afirmando em uma *claim* de identificação do usuário como parte do fluxo de autorização.

Apresentação dos Controles Técnicos de Segurança

Disposições de segurança do Open Insurance

O Open Insurance tem um requisito para endereçar considerações de privacidade que foram identificadas, mas não abordadas na especificação final FAPI-1-Advanced, sem impor requisitos adicionais aos Servidores de Autorização que estão sendo propostos em FAPI-2-Baseline.

Os participantes desse ecossistema precisam que os clientes de API solicitem a um provedor OpenID a confirmação dos valores das *claims* de identificação do usuário como parte de uma solicitação de autorização usando o mecanismo definido na cláusula 5.5.1 de OIDC.

O uso do parâmetro *claims* para solicitar a validação de valores de identificação explícitos requer que os clientes de API protejam com criptografia o *Request Object* para evitar vazamento de informações. Este risco é identificado na cláusula 7.4.1 do FAPI-1-Baseline.

Além disso, este perfil descreve o escopo específico, valores de ACR e requisitos de gerenciamento de clientes necessários para dar suporte ao ecossistema Open Insurance mais amplo.

Como um perfil do OAuth 2.0 *Authorization Framework*, este documento exige o seguinte para o perfil de segurança do Open Insurance.

Servidor de Autorização

O Servidor de Autorização deve suportar as disposições especificadas na cláusula 5.2.2 de Financial-grade API Security Profile 1.0 - Parte 2: Advanced. Além disso, ele deve:

1. suportar *Request Objects* JWE assinados e criptografados passados por valor ou deve exigir requisições do tipo "*pushed authorization requests*" PAR
2. publicar metadados de descoberta (incluindo a do *endpoint* de autorização) por meio do documento de metadado especificado em OIDD e [RFC8414] *("well-known")*
3. suportar os parâmetros *claims* como definido no item 5.5 do OpenID Connect Core
4. suportar o atributo *claim* padrão OIDC "cpf" conforme definido no item 5.2.2.2 deste documento
5. suportar o atributo *claim* padrão OIDC "cnpj" conforme definido no item 5.2.2.3 deste documento, se a instituição for detentora de conta para pessoas jurídicas
6. suportar o atributo ACR "urn:brasil:openinsurance:loa2" como definido no item 5.2.2.4 deste documento
7. implementar o *endpoint* "*userinfo*" como definido no item 5.3 do OpenID Connect Core
8. suportar o escopo parametrizável ("*parameterized* OAuth 2.0 *resource scope*") *consent* como definido no item 6.3.1 de OIDF FAPI WG *Lodging Intent Pattern*
9. suportar Financial-grade API: Client Initiated Backchannel Authentication Profile se suportar o *scope payments*
10. suportar *refresh tokens*
11. emitir *access tokens* com o tempo de expiração entre 300 (mínimo) e 900 (máximo) segundos.
12. [opcional] suportar Financial-grade API: Client Initiated Backchannel Authentication Profile
13. [opcional] suportar o atributo ACR "urn:brasil:openinsurance:loa3"

Token de ID como assinatura separada

O Servidor de Autorização deve suportar as disposições especificadas na cláusula 5.2.2.1 de Financial-grade API Security Profile 1.0 - Parte 2: Advanced. Além disso, se o valor *response_type code id_token* for usado, o servidor de autorização não deveria retornar Informação de Identificação Pessoal (PII) confidenciais no *token* de ID na resposta de autorização, mas se for necessário, então ele deve criptografar o *token* de ID.

Solicitando uma "*claim*" cpf

Este perfil define "cpf" como uma nova *claim* padrão de acordo com cláusula 5.1 OIDC.

O número do CPF (Cadastro de Pessoas Físicas, português para "Registro de Pessoas Físicas") é o cadastro de pessoa física brasileira. Este número é atribuído pela Receita Federal Brasileira para brasileiros e estrangeiros residentes que, direta ou indiretamente, pagar impostos no Brasil.

No modelo de identidade do Open Insurance, o CPF é uma *string* composta por números 11 caracteres de comprimento e podem começar com 0.

Se a *Claim* "cpf" for solicitada como essencial para constar no ID *token* ou na resposta ao *endpoint* de UserInfo e na solicitação constar no parâmetro *value* com determinado CPF exigido, o *Authorization Server* DEVE retornar no atributo "cpf" o valor que corresponda ao da solicitação.

Se a *Claim* "cpf" for solicitada como essencial para constar no ID Token ou na resposta no *endpoint* de UserInfo, o *Authorization Server* deve retornar no atributo "cpf" o valor com o CPF do usuário autenticado.

Se a *Claim* "cpf" indicada como essencial não puder ser preenchida ou não for compatível com o requisito, o *Authorization Server* deve tratar a solicitação como uma tentativa de autenticação com falha.

Nome: cpf, Tipo: String, Regex: 'd{11}$'

Solicitando a "*claim*" cnpj

Este perfil define "cnpj" como uma nova reivindicação padrão de acordo com cláusula 5.1 OIDC.

CNPJ, abreviação de Cadastro Nacional de Pessoas Jurídicas, é um número de identificação de empresas brasileiras emitidas pelo Ministério da Fazenda brasileira, na "Secretaria da Receita Federal" ou "Ministério da Fazenda" do Brasil. No modelo de identidade do Open Insurance, pessoas físicas podem se associar a 0 ou mais CNPJs. Um CNPJ é uma *string* que consiste em números de 14 dígitos e pode começar com 0, os primeiros oito dígitos identificam a empresa, os quatro dígitos após a barra identificam a filial ou subsidiária ("0001" padrão para a sede), e os dois últimos são dígitos de soma de verificação. Para este perfil, o pedido de "cnpj" deve ser solicitado e fornecido como o número de 14 dígitos.

Se a *Claim* "cnpj" for solicitada como essencial para constar no ID Token ou na resposta ao *endpoint* UserInfo e na solicitação constar, no parâmetro *value*, determinado CNPJ exigido, o *Authorization Server* DEVE retornar no atributo "cnpj" um conjunto de CNPJs relacionado com o usuário, um dos quais deve incluir valor que corresponda ao da solicitação.

Se a *Claim* "cnpj" for solicitada como essencial para constar no ID Token ou na resposta ao *endpoint* UserInfo, o *Authorization Server* deve incluir no ID Token ou na resposta ao *endpoint* UserInfo um conjunto que inclua um elemento com o número do CNPJ relacionado à conta utilizada na autenticação do usuário.

Se a *Claim* "cnpj" indicada como essencial não puder ser preenchida ou validada, o *Authorization Server* deve tratar a solicitação como uma tentativa de autenticação com falha.

Nome: cnpj, Tipo: Array of Strings, Array Element Regex: 'd{14}$'

Solicitando o "urn:brasil:openinsurance:loa2" ou "urn:brasil:openinsurance:loa3" Solicitação de contexto de autenticação

- **LoA2:** mecanismo de autenticação com a adoção de um único fator
- **LoA3:** mecanismo de autenticação com múltiplos fatores de autenticação

A seguinte regra deve ser adotada para o mecanismo de autenticação:

- **Para controle de acesso às API's definidas na FASE 2 (leitura de dados):** os *Authorization Server*s das instituições transmissoras de dados devem condicionar a autenticação do usuário proprietário do dado, no mínimo, a adoção de método compatível com LoA2. A adoção de mecanismo de autenticação mais rigoroso (LoA3) fica a critério da instituição transmissora de acordo com sua avaliação de riscos.
- **Para acesso às API's das fases subsequentes (em especial pagamento):** o acesso deve ser condicionado à método de autenticação compatível com LoA3 ou superior.

Esclarecimentos adicionais sobre fatores de autenticação

São fatores de autenticação: *Aquilo que você conhece, como uma senha ou frase secreta * Aquilo que você tem, como um *token*, *smartcard* ou dispositivo * Aquilo que "você é", ou seja, autenticação condicionada a apresentação de uma característica física exclusivamente sua, como a validação por biometria

Para realizar autenticação por múltiplos fatores (MFA) é necessário que o usuário apresente, ao menos, dois diferentes fatores dos listados acima. Um mesmo fator usado mais de uma vez - por exemplo, a apresentação de suas senhas que ele conhece - não pode ser aceito como MFA.

Cliente confidencial

Um cliente confidencial deve apoiar as disposições especificadas na cláusula 5.2.3 de Financial-grade API Security Profile 1.0 - Part 2: Advanced. Além disso, o cliente confidencial deve:

- suportar objetos de solicitação *encrypted*
- suportar solicitações de autorização *push (pushed authorization requests)* PAR
- usar objetos de solicitação *encrypted* se não usar PAR
- suportar o escopo de recurso OAuth 2.0 parametrizado *consent* conforme definido na cláusula 6.3.1 OIDF FAPI WG *Lodging Intent Pattern*
- suportar *refresh tokens*

Ciclo de vida de autorização

O recurso de consentimento tem um ciclo de vida gerenciado separada e distintamente da estrutura de autorização OAuth 2.0. As transições de estado e comportamentos esperados e condições de erro esperados dos Recursos REST protegidos com este perfil são definidos nas especificações funcionais da API publicadas pelo Open Insurance.

Servidor de autorização

Além dos requisitos descritos nas disposições de segurança do Open Insurance, o Servidor de Autorização deve:

- apenas emitir *refreshtokens*_ quando vinculados a um consentimento ativo e válido;
- só compartilhar o acesso aos recursos quando apresentado *accesstoken*_ vinculado a um consentimento ativo e válido;
- revogar os *refresh tokens* e, quando aplicável, os *access tokens* quando o Consentimento *(Consent Resource)* relacionado for apagado;
- garantir que os *access tokens* são emitidos com os *scopes* necessários para permitir acesso aos dados especificados em elemento *Permission* do Consentimento *(Consent Resource Object)* relacionado;
- não rejeitar pedido de autorização com *scopes* além do necessário para permitir acesso a dados definidos em elemento *Permission* do Consentimento *(Consent Resource Object)* relacionado;
- reduzir o escopo solicitado para um nível que seja suficiente para permitir o acesso aos dados definidos em elemento *Permission* do Consentimento *(Consent Resource Object)* relacionado;
- manter registros sobre o histórico dos consentimentos para permitir a adequada formação de trilhas de auditoria em conformidade com a regulação em vigor.

Cliente confidencial

Além dos requisitos descritos nas disposições de segurança do Open Insurance, o Cliente Confidencial deve, sempre que possível, revogar e cessar o uso de *refresh* e de *access tokens* vinculados a um consentimento *(Consent Resource Object)* que foi excluído e deve excluir consentimentos *(Consent Resource Objects)* que estão expirados.

Apresentação dos Controles Técnicos de Segurança

Gerenciamento de Identidade e Acesso

> **Observações:**
>
> É importante ressaltar que este controle foi proposto seguindo os padrões do Open Banking e ele é obrigatório para o escopo *open-data* (fase 1), tendo em vista que as instituições participantes irão seguir os fluxos apresentados para realizar os cadastros iniciais. Adicionalmente, este controle está relacionado com os itens 2.7, 7.1, 7.2 e 7.16 do Manual de Segurança publicado pela SUSEP.

A funcionalidade de "Gerenciamento de Identidade e Acesso" abrange todos os processos de negócio executados desde o primeiro contato do usuário com a página inicial do Open Insurance até o final de sua inscrição como Participante do Diretório do Open Insurance.

Podemos dividir o processo de inscrição de um participante no Diretório do Open Insurance em 4 etapas, sendo elas:

1. Solicitação de cadastro no Diretório pelo Representante;
2. Cadastro das informações do Representante;
3. Cadastro das informações da Instituição Participante; e
4. Cadastro dos Contatos da Instituição Participante.

Além disso, para completude desta funcionalidade faz-se necessário o processo de "Revogação de acesso da Instituição Participante no diretório".

Disposições referentes a validação de identidade do Representante

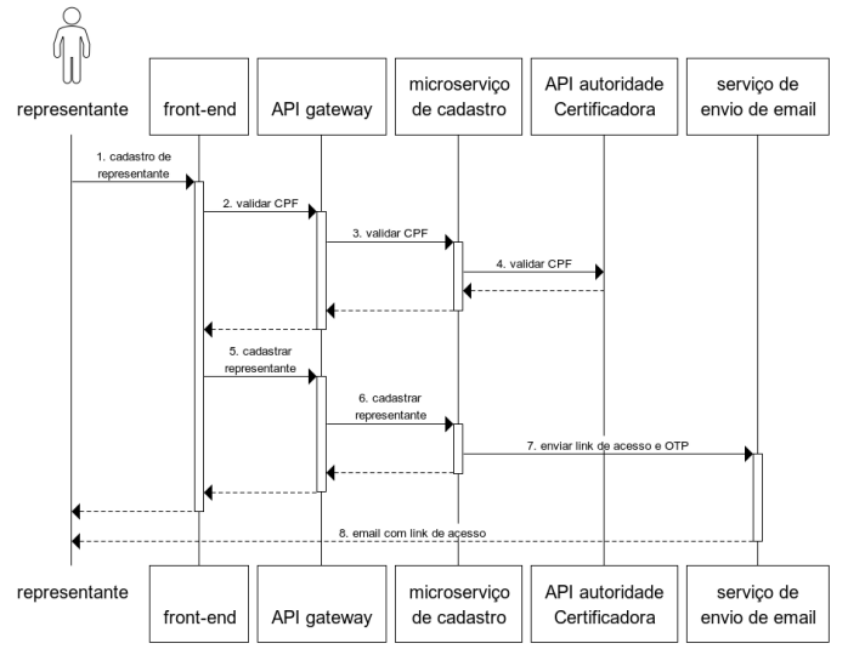
Com o objetivo de validar a identidade do Represente durante o processo de cadastro, as seguintes premissas serão consideradas:

- O Representante que irá solicitar o cadastro no Diretório deverá possuir CPF registrado numa certificadora digital e-CPF;
- A validação do CPF do Representante será realizada através de API da Autoridade Certificadora responsável pela emissão;
- Os representantes indicados pela participante serão validados via checagem do termo de adesão do participante, que traz os representantes legais da empresa, junto da procuração legal que atesta sobre os representantes indicados;
- A categoria de domínio de mercado do participante será validada através da API de dados cadastrais, disponibilizada no seguinte link: https://dados.susep.gov.br/olinda/servico/empresas/versao/v1/odata/DadosCadastrais?$format=json. Os domínios aceitos, dispostos pelo atributo "mercodigo", são 1, 2, 6 e 99, que representam os domínios "Previdência", "Seguros", "Capitalização" e "Sandbox" respectivamente; e
- Em acordo ao §4º do Art. 6º da Resolução CNSP 415ª, a posse de provisões técnicas do participante será validada através do SES – Sistema de Estatíticas da SUSEP.

Apresentação dos Controles Técnicos de Segurança

Solicitação de cadastro no Diretório pelo Representante

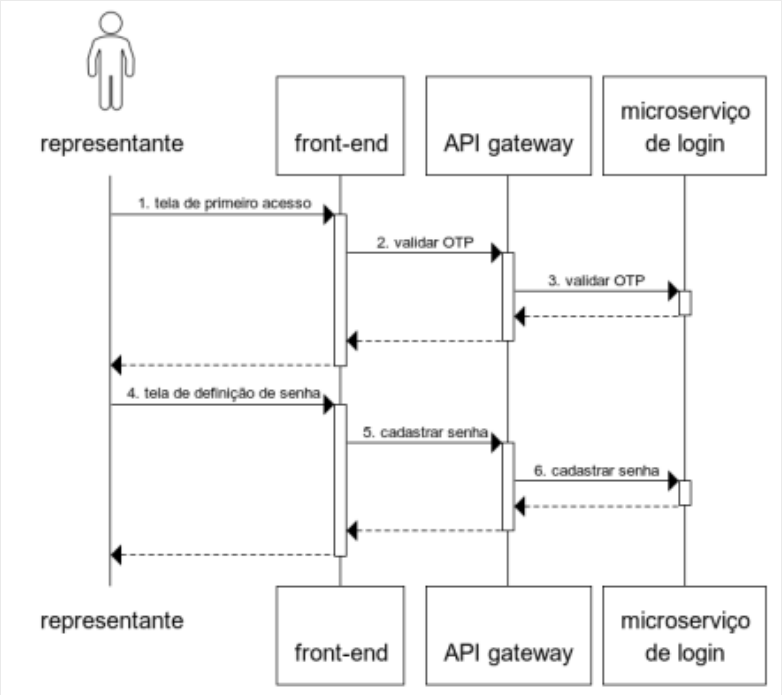
Esta seção abrange as ações necessárias para que um Representante solicite a participação de uma Instituição no Diretório, desde o acesso a página inicial do Open Insurance até a submissão do formulário de solicitação de participação.



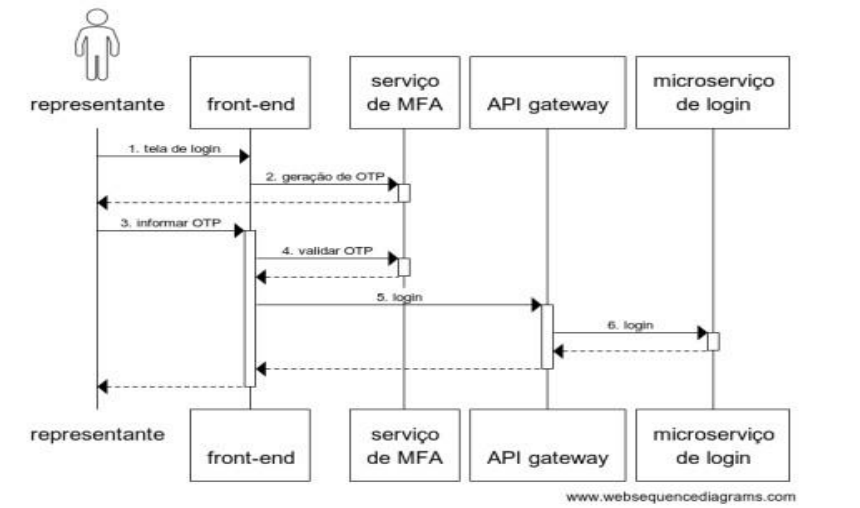
Para mais informações sobre o cadastro no diretório de participantes consulte o manual "Passo a passo de cadastro no diretório" no Portal do Desenvolvedor.

Cadastro das informações do Representante

Esta seção abrange as ações que o Representante deve executar para habilitar seu cadastro e consecutivamente efetuar o *login* no Diretório do Open Insurance.

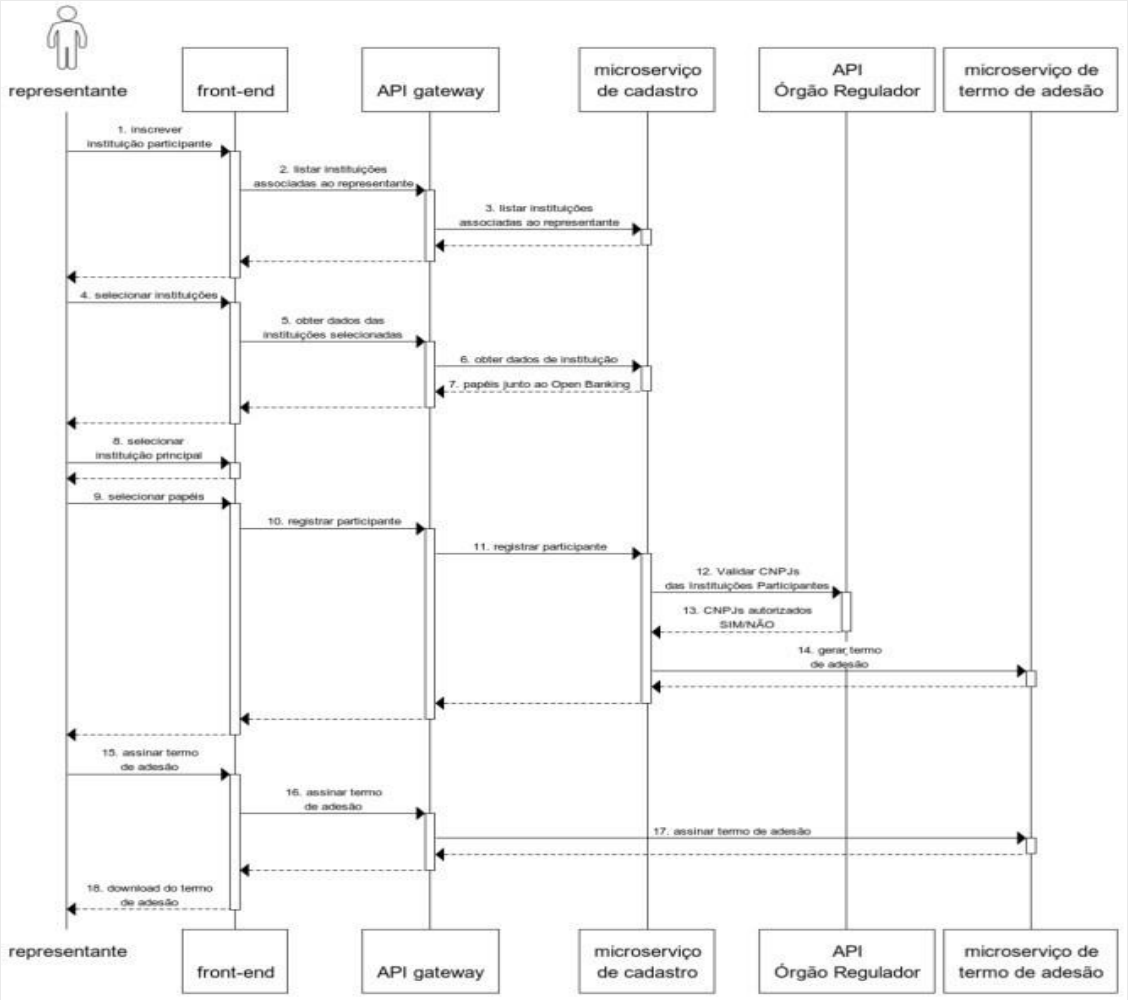


Efetuar *login* do representante no Diretório



Cadastro das informações da Instituição Participante

Após configuração do *login* e acesso ao Diretório, o Representante deve proceder com a configuração de sua Instituição Participante no Open Insurance, indicando os papéis de atuação da Instituição, a URL raiz da API do participante, a URL da documentação sobre a API do participante e finalizando com o aceite do Termo de Adesão exigido pelo Open Insurance para o Participante.



Apresentação dos Controles Técnicos de Segurança

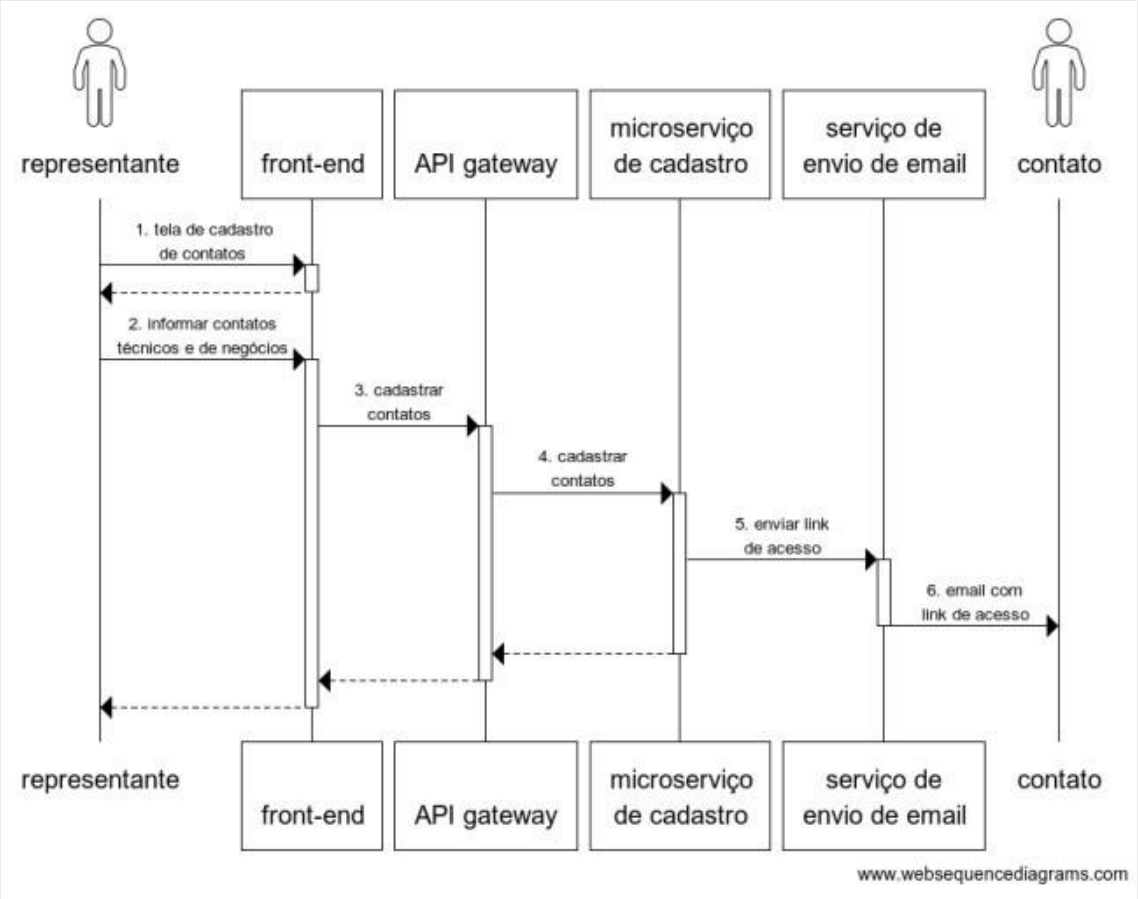
Cadastro dos Contatos da Instituição Participante

Após o cadastro do Participante, o Representante deve proceder informando os dados dos Contatos Técnicos e de Negócio que responderão pelo Participante. Estes contatos receberão um *e-mail* do Diretório com instruções para cadastro de senha e *login* para que possam concluir seu cadastro.

Existem quatro contatos a serem cadastrado pelo Representante:

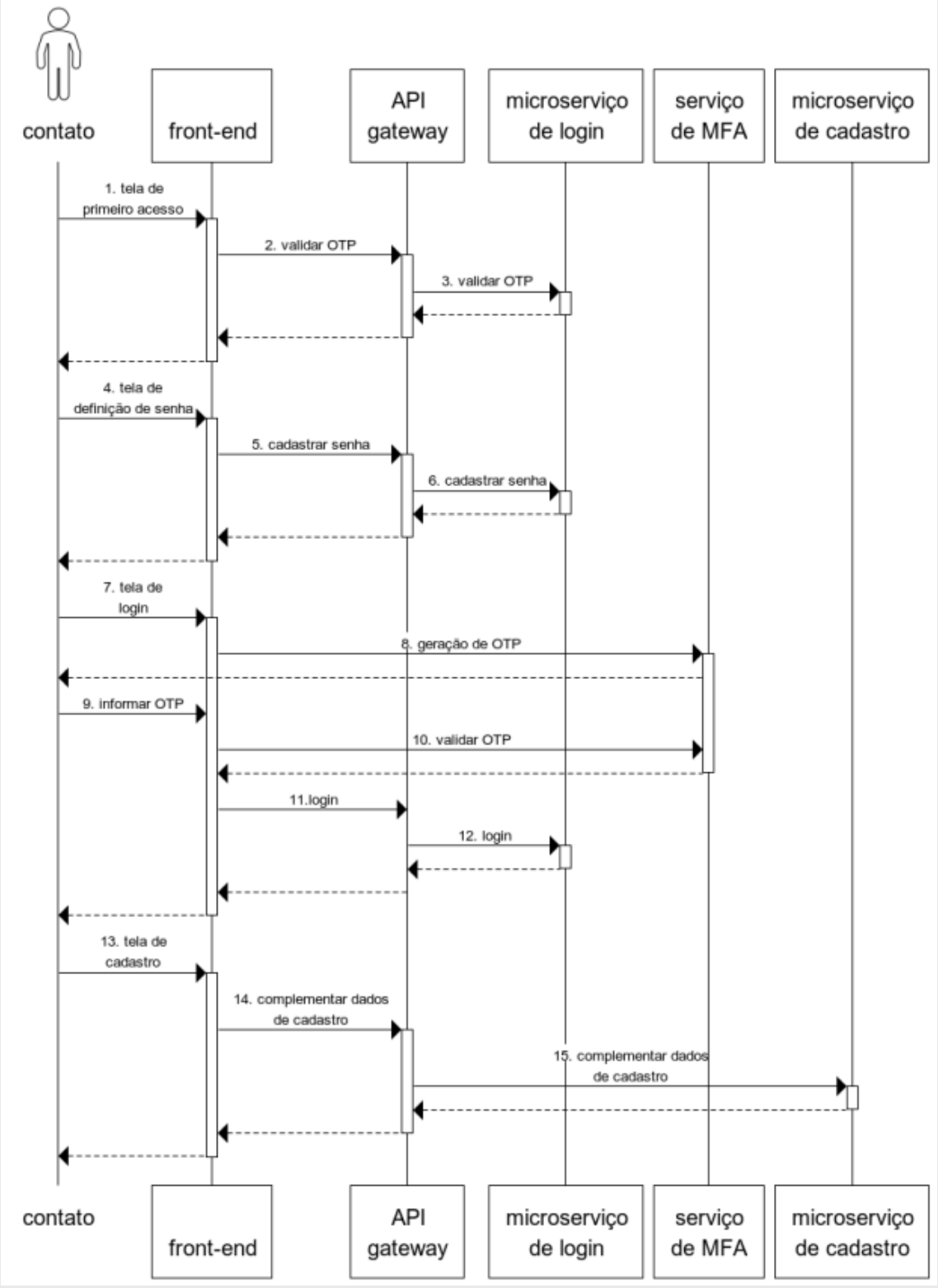
- **Contato de Negócio Principal (mandatório):** indivíduo indicado pelo Participante para ter acesso ao Diretório de Participantes do Open Insurance. Este indivíduo poderá nomear outros Contatos de Negócios e receberá as notificações do Diretório via *e-mail*;
- **Contato de Negócio Secundário (opcional):** figura similar ao Contato de Negócio Principal;
- **Contato Técnico Principal (mandatório):** indivíduo indicado pelo Participante para ter controle técnico no Diretório de Participantes, podendo solicitar emissão e gerir certificados do Open Insurance;
- **Contato Técnico Secundário (opcional):** figura similar ao Contato Técnico Principal.

O Representante poderá assumir a função de Contato de Negócio e/ou Contato Técnico.



Apresentação dos Controles Técnicos de Segurança

Ativação dos contatos e login no Diretório



Para mais informações sobre o cadastro de contatos no diretório de participantes, consulte o manual "Cadastro Contatos Técnicos Diretório" no Portal do Desenvolvedor.

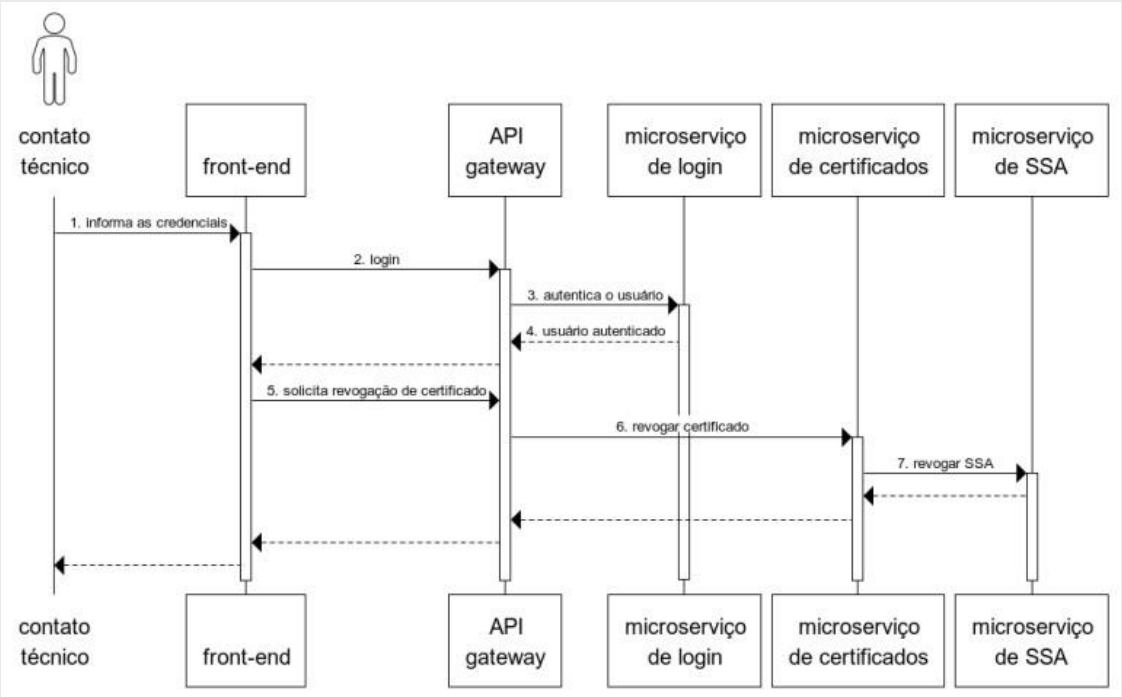
Apresentação dos Controles Técnicos de Segurança

Revogação de acesso da Instituição Participante no Diretório

A revogação poderá ocorrer para os seguintes casos:

- **Certificado:** a solicitação poderá ser realizada pelo Participante;
- **Participante:** a solicitação deverá ser realizada somente pelo Órgão Regulador.

Solicitação de revogação do certificado pelo Participante



Solicitação de revogação do Participante pelo Órgão Regulador

A listagem de organizações na API de dados cadastrais será validada mensalmente pelo Secretariado da Estrutura Inicial do Open Insurance Brasil. Após a remoção de um participante da listagem de organizações na API de dados cadastrais, disponibilizada no link: https://dados.susep.gov.br/olinda/servico/empresas/versao/v1/odata/DadosCadastrais?$format=json, a empresa terá seu acesso ao diretório revogado.

Cadastro da aplicação no Diretório

> **Observações:**
>
> É importante ressaltar que este controle foi proposto seguindo os padrões do Open Banking e ele é obrigatório para o escopo *open-data* (fase 1). Adicionalmente, este controle está relacionado com os itens 2.7, 7.1, 7.2 e 7.16 do Manual de Segurança publicado pela SUSEP.

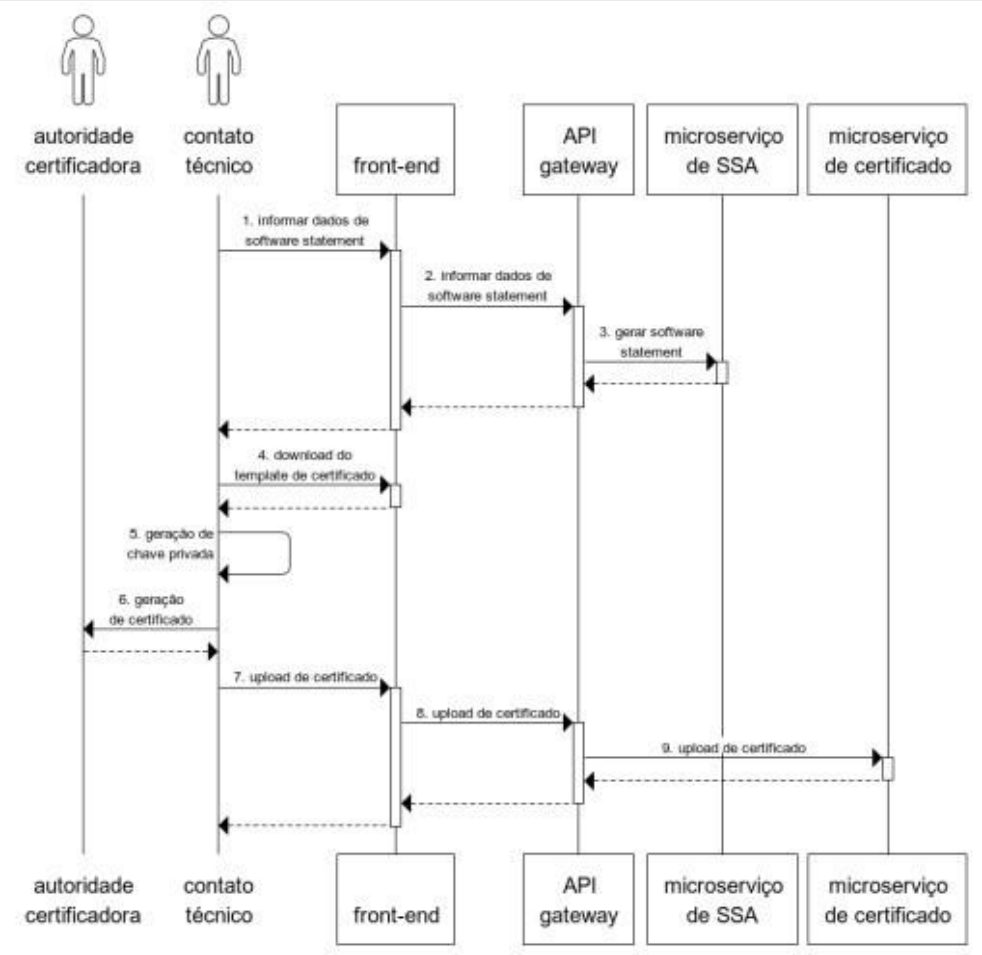
Para que os Participantes operem dentro do ecossistema do Open Insurance, após a realização de seu cadastro, eles precisam fornecer alguns dados técnicos e gerar certificados. Uma vez habilitados, isso fará com que as Instituições Transmissoras liberem acessos aos *endpoints* das APIs para as marcas e nomes comerciais vinculados a entidade cadastrada. Outra ação importante do lado do Participante é de registro de cada aplicação referente às Instituições Transmissoras que eles precisam de acesso aos *endpoints* das APIs.

Apresentação dos Controles Técnicos de Segurança

Podemos dividir o processo de "Cadastro da aplicação no Diretório" em 3 principais etapas, sendo elas:

1. Criar um *Software Statement*;
2. Criar um *Software Statement Assertion*;
3. Registrar Cliente/Credenciais (Manual / Automático).

Criar um Software Statement



Criar um *Software Statement Assertion*

O *Software Statement Assertion* (SSA) é um *Software Statement* (SS) assinado pelo Diretório de Participantes com o objetivo de identificar uma aplicação (ou *software*) declarada da Instituição Receptora. O SSA é utilizado sempre que uma Instituição Receptora necessita receber credenciais para acesso aos dados de uma Instituição Transmissora.

Para mais informações sobre a criação de um *Software Statement* no diretório de participantes consulte o manual "Criando uma Declaração de Software" no Portal do Desenvolvedor.

O SSA poderá ser emitido de maneira manual através de interface disponível no Diretório de Participantes ou de maneira automatizada através de uma conexão com a API disponibilizada pelo Diretório de Participantes. O detalhamento dos processos manual e automatizado serão descritos a seguir.

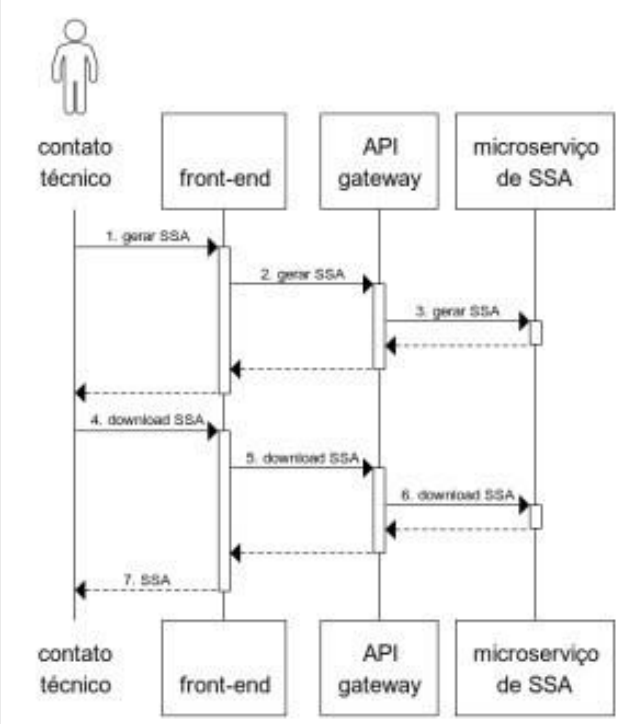
Apresentação dos Controles Técnicos de Segurança

Criar um Software Statement Assertion (SSA) - Manual

A solicitação de criação de *Software Statement Assertion* se inicia após a geração do *Software Statement* e quando o Contato Técnico acionar o botão "Gerar" na página inicial da área logada do Diretório, na seção de '*Software Statement Assertion*'.

Uma vez acionado o botão, será apresentado ao Contato Técnico um JSON Web Token – JWT – contendo os metadados do cliente em uma instância do *software* da Instituição Participante. O JWT é emitido e assinado pelo Open Insurance.

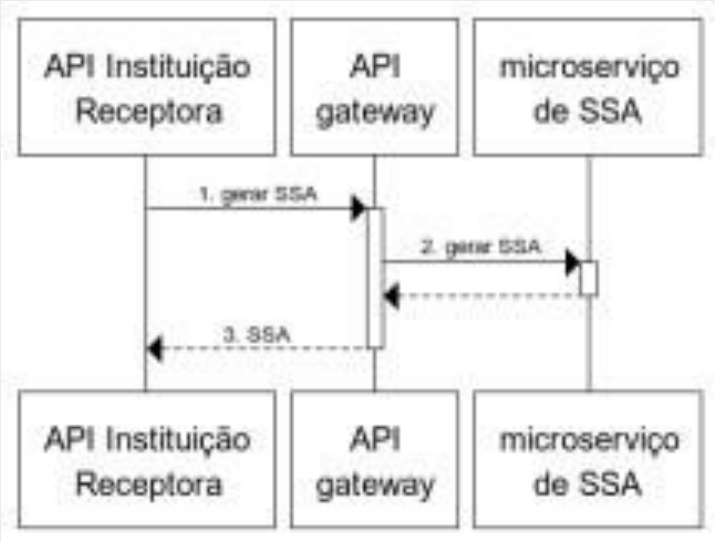
Para obter o *Software Statement Assertion*, o Contato Técnico deve acionar o botão *'Copy to Clipboard'*, permitindo que o Contato Técnico armazene o mesmo e inicie o credenciamento junto a outras instituições.



Criar um Software Statement Assertion (SSA) – Automatizado via API

Para a obtenção do *Software Statement Assertion*, a Instituição Participante deve executar uma API do Diretório que retornará um token, que permitirá o acesso à plataforma.

Após a obtenção do token de acesso, a Instituição Participante deverá executar uma API do Diretório que retornará o *Software Statement Assertion*, permitindo que o Contato Técnico armazene o mesmo e inicie o credenciamento junto a outras instituições.

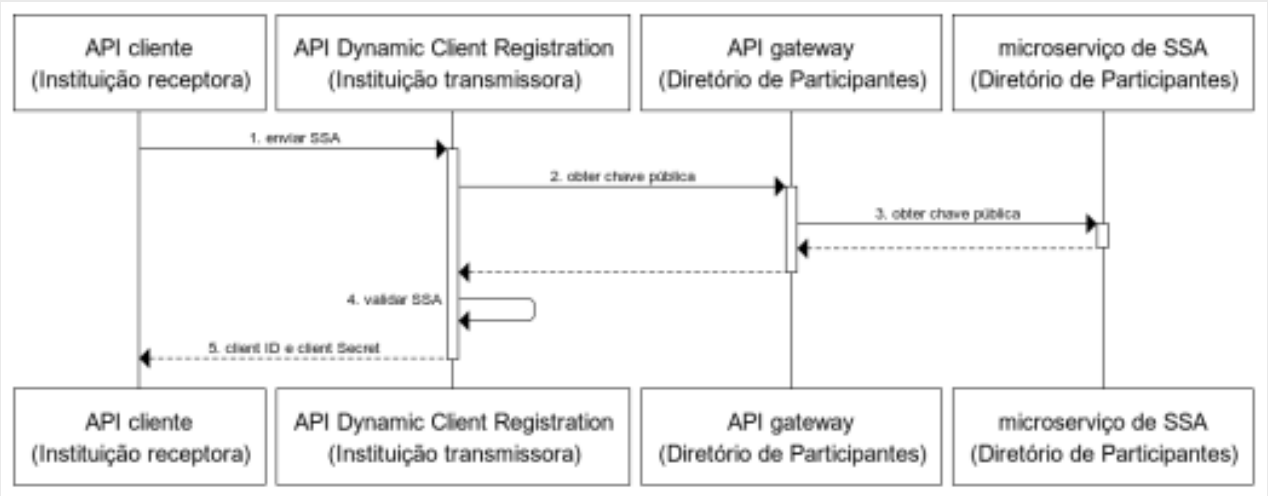


Para mais informações sobre a obtenção do token de acesso e uso das APIs do diretório consulte o manual "Obtendo um token para acesso as APIs do Diretório" no Portal do Desenvolvedor.

Registrar Cliente/Credenciais (Manual / Automático)

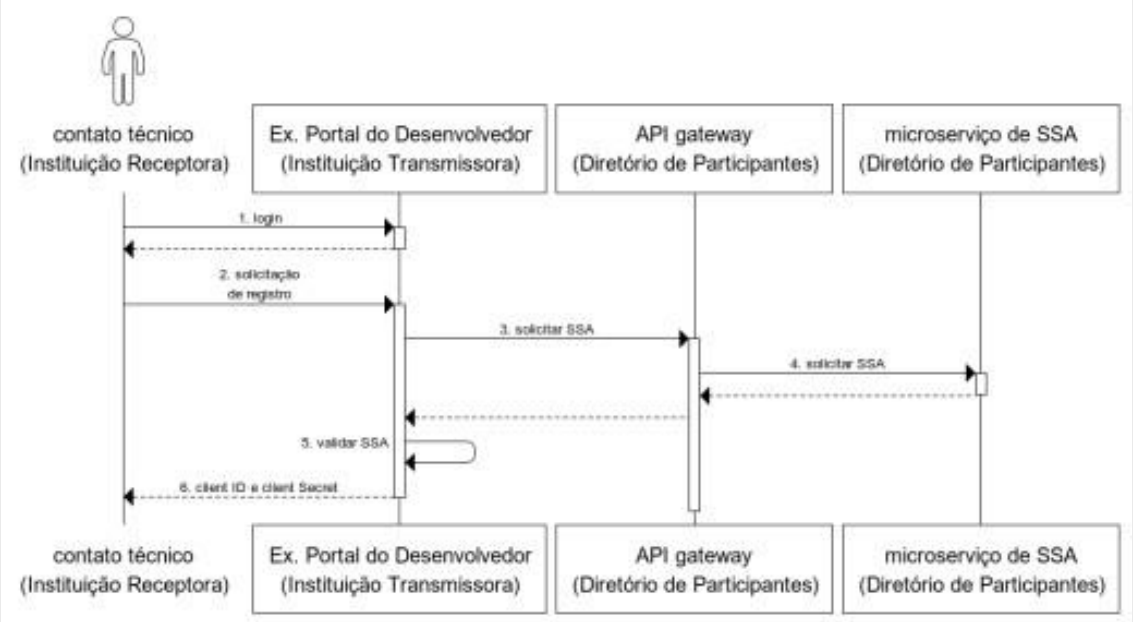
Para que uma Instituição Receptora possa consumir os dados de uma Instituição Transmissora, ela necessita de credenciais específicas de acesso. Para obter tais credenciais a Instituição Receptora deve possuir e entregar para a Instituição Transmissora um *Software Statement Assertion* (SSA) assinado. De posse do SSA, a instituição Transmissora deve validá-lo junto ao Diretório de Participantes e após concluí-lo, deverá gerar e entregar a credencial de acesso à Instituição Receptora.

Registro entre os Participantes do Diretório – Automatizado



Para mais informações sobre o processo automatizado de registro de clientes, consulte a seção "Dynamic Client Registration (DCR)" do Portal do Desenvolvedor.

Registro entre os Participantes do Diretório – Manual



Padrão de Algoritmos

> **Observações:**
>
> É importante ressaltar que este controle foi proposto seguindo os padrões do Open Banking e ele é obrigatório para o escopo *open-data* (fase 1). Adicionalmente, este controle está relacionado com os itens 4.4, 4.5, 4.10, 4.13, 4.16 e 4.22 do Manual de Segurança publicado pela SUSEP.

Os participantes devem apoiar todas as considerações de segurança especificadas na cláusula 8 Financial-grade API Security Profile 1.0 - Parte 2: Advanced. O ICP brasileiro emite certificados RSA x509 somente, portanto, para simplificar, a seção remove o suporte para algoritmos EC e exige que apenas algoritmos de criptografia recomendados pela IANA sejam usados.

Considerações de algoritmo

Para JWS, clientes e servidores de autorização:

1. devem usar o algoritmo PS256;

Considerações de algoritmo de criptografia

Para JWE, clientes e servidores de autorização devem:

1. usar RSA-OAEP com A256GCM

Considerações sobre o uso seguro do Transport Layer Security

Para TLS, *endpoints* do Servidor de Autenticação e *endpoints* do Servidor de Recursos usados diretamente pelo cliente:

1. devem suportar TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
2. devem suportar TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

Padrão de Certificados

> **Observações:**
>
> É importante ressaltar que este controle foi proposto seguindo os padrões do Open Banking e ele é obrigatório para o escopo *open-data* (fase 1) apenas no que tange a emissão de certificados de criptografia. Adicionalmente, este controle está relacionado com os itens 4.4, 4.5, 4.9, 4.10, 4.11 e 4.14 do Manual de Segurança publicado pela SUSEP.

O ecossistema do Open Insurance faz uso de cadeias de certificados e protocolo TLS para garantir a confidencialidade, autenticação e integridade do canal de comunicação utilizado pelas APIs das instituições participantes, bem como dos clientes de cada um dos participantes.

Os certificados utilizados pelo Open Insurance também são necessários para autenticar as aplicações através do oAuth 2.0 mTLS ou *privatekeyjwt*, além de também servirem para realizar a assinatura de *payload* pelo uso de JWS. Outra atribuição importante dos certificados é autenticar e apresentar um canal seguro para o usuário final no ato de autenticação e uso dos serviços prestados pela entidade participante.

Certificados ICP-Brasil

Os certificados emitidos pelas Autoridades Certificadoras autorizadas pelo ICP-Brasil são utilizados apenas na comunicação entre as entidades participantes do ecossistema do Open Insurance.

Os processos de emissão e revogação dos certificados são de responsabilidade das próprias Autoridades Certificadores, sendo regulamentados por Declarações de Prática de Certificação, e supervisionadas pelo Comitê Gestor da Infraestrutura de Chaves Públicas Brasileira.

As práticas, processos, disponibilização e valores praticados pelas Autoridades Certificadoras não são de responsabilidade do Estrutura Inicial do Open Insurance.

Autoridades Certificadoras

As seguintes autoridades certificadoras são autoridades habilitadas para realizar a emissão de certificados para o Open Insurance Brasil:

- Serasa
- Serpro
- Soluti
- CertiSign
- Valid

Apresentação dos Controles Técnicos de Segurança

Algoritmos

Todos os certificados emitidos junto ao ICP-Brasil devem possuir as seguintes características:

- Tipo A do ICP-Brasil;
- Algoritmo de Chaves: RSA 2048 bits;
- *Message Digest:* SHA 256 bits.

Certificado Servidor

O Certificado Servidor deve ser emitido para proteger e autenticar o canal TLS utilizado pelas APIs que serão consumidas pelas aplicações cliente de entidades participantes do Open Insurance.

O padrão de certificado utilizado deve seguir as práticas de emissão de certificados existentes de "CERTIFICADO PARA SERVIDOR WEB - ICP-Brasil".

Certificado Cliente

Os Certificados de Aplicação Cliente (Transporte) são utilizados para autenticar o canal mTLS e para realizar a autenticação da aplicação cliente através de oAuth2.0 mTLS ou *privatekeyjwt*, de acordo com o cadastro da aplicação realizado pelo processo de *Dynamic Client Registration* junto à entidade transmissora.

Para emissão de Certificado Cliente é necessário que a instituição participante do Open Insurance tenha realizado o cadastro da aplicação no Serviço de Diretório, através do processo de emissão de *Software Statement Assertion*, e com isso já tenha obtido o valor de *Software Statement* ID.

Atributos Open Insurance

- **serialNumber:** Cadastro Nacional de Pessoal Jurídica (CNPJ) da pessoa jurídica titular do certificado e associado ao atributo UID e *Software Statement* ID, durante validação junto ao Serviço de Diretório do Open Insurance;
- **organizationIdentifier:** Código de Participante associado ao CNPJ listado no Serviço de Diretório do Open Insurance Brasil;
- **UID:** Software Statement ID cadastrado no Serviço de Diretório do Open Insurance e pertencente ao CNPJ e Código de Participante.

O Certificado Cliente deve ser emitido através de cadeia V10, e deve obrigatoriamente conter os seguintes atributos:

Distinguished Name

- **businessCategory (OID 2.5.4.15):** Tipo de categoria comercial, devendo conter: "Private Organization" ou "Government Entity" ou "Business Entity" ou "Non-Commercial Entity"
- **jurisdictionCountryName (OID: 1.3.6.1.4.1.311.60.2.1.3):** BR
- **serialNumber (OID 2.5.4.5):** CNPJ
- **countryName (OID 2.5.4.6):** BR
- **organizationName (OID 2.5.4.10):** Razão Social
- **stateOrProvinceName (OID 2.5.4.8):** Unidade da federação do endereço físico do titular do certificado
- **localityName (OID 2.5.4.7):** Cidade do endereço físico do titular
- **organizationIdentifier (OID 2.5.4.97):** Código de Participante associado ao CNPJ listado no Serviço de Diretório do Open Insurance Brasil e prefixo de identificação do diretório

- **UID (OID 0.9.2342.19200300.100.1.1):** *Software Statement* ID gerado pelo Diretório do Open Insurance
- **commonName (OID 2.5.4.3):** FQDN ou *Wildcard*

Certificate Extensions

- **keyUsage:** critical,digitalSignature,keyEncipherment
- **extendedKeyUsage:** clientAuth

Subject Alternative Name

- **dNSName:** FQDN ou Wildcard

Para mais informações sobre a geração do certificado de assinatura em Sandbox e cadastro do certificado em produção consulte o manual "Gerando o Certificado BRCAC" no Portal do Desenvolvedor. O seguinte documento apresenta as informações de contato das ACs homologadas a emissão de certificados: Processo de Certificação FAPI.

Certificados de Assinatura

Os Certificados de Assinatura são utilizados para realizar assinatura do *payload* através do uso de JWS (*JSON Web Signature)*.

Atributos Open Insurance Presentes no Certificado

- **UID:** Código de Participante associado ao CNPJ listado no Serviço de Diretório do Open Insurance;
- **commonName:** Razão Social cadastrado no Serviço de Diretório do Open Insurance e pertencente ao CNPJ e Código de Participante.

O Certificado de Assinatura deve ser emitido através de cadeia V5, e deve obrigatoriamente conter os seguintes atributos:

Distinguished Name

- **UID (OID 0.9.2342.19200300.100.1.1):** Código de Participante associado ao CNPJ listado no Serviço de Diretório do Open Insurance
- **countryName (OID 2.5.4.6):** BR
- **organizationName (OID 2.5.4.10):** ICP-Brasil
- **organizationalUnitName (OID 2.5.4.11):** Nome da Autoridade Certificadora
- **organizationalUnitName (OID 2.5.4.11):** CNPJ da Autoridade de Registro
- **organizationalUnitName (OID 2.5.4.11):** Tipo de identificação utilizada (presencial, videoconferência ou certificado digital)
- **commonName (OID 2.5.4.3):** Nome da Razão Social

Certificate Extensions

- **keyUsage:** critical,digitalSignature,nonRepudiation

Subject Alternative Name

- **otherName (OID 2.16.76.1.3.2 - ICP-Brasil):** Nome do responsável pelo certificado
- **otherName (OID 2.16.76.1.3.3 - ICP-Brasil):** Cadastro Nacional de Pessoa Jurídica (CNPJ) da pessoa jurídica titular do certificado;
- **otherName (OID 2.16.76.1.3.4 - ICP-Brasil):** Responsável pelo certificado de pessoa jurídica titular do certificado (data de nascimento, CPF, PIS/PASEP/CI, RG);

- **otherName (OID 2.16.76.1.3.7 - ICP-Brasil):** Número do Cadastro Específico do INSS (CEI) da pessoa jurídica titular do certificado.

Para mais informações sobre a geração do certificado de assinatura em Sandbox e cadastro do certificado em produção consulte o manual "Gerando o Certificado BRSEAL" no Portal do Desenvolvedor. O seguinte documento apresenta as informações de contato das ACs homologadas a emissão de certificados: Processo de Certificação FAPI.

Certificados para front-end

Os certificados para *Front-End* são utilizados para disponibilizar serviços, em geral páginas Web, com uso de TLS, que são acessados pelo usuário final. Dado a sua finalidade, e para garantir maior interoperabilidade, os certificados devem ser do tipo EV *(Extended Validation)* e devem ser gerados através de uma autoridade certificadora válida, seguindo as regras definidas na RFC 5280 e RFC 2818, em conformidade com os princípios e critérios *WebTrust*.

Cabeçalhos de Segurança

> **Observações:**
>
> É importante ressaltar que este controle é obrigatório para o escopo *open-data* (fase 1) e está relacionado com os itens 3, 4.6 e 4.8 do Manual de Segurança publicado pela SUSEP.

Existem diversos cabeçalhos relacionados à segurança que podem ser retornados nas respostas HTTP para instruir os navegadores a agirem da maneira esperada. No entanto, alguns desses cabeçalhos devem ser usados com respostas HTML e, como tal, podem fornecer poucos ou nenhum benefício de segurança em uma API que não retorna HTML. Com isso, os seguintes cabeçalhos devem ser incluídos em todas as respostas da API:

Cabeçalho	Objetivo
Cache-Control: no-store	Evitar que informações confidenciais sejam armazenadas em cache.
Content-Security-Policy: frame-ancestors 'none'	Proteger contra ataques de *clickjack* do estilo *"drag and drop"*.
Content-Type	Especificar o tipo de conteúdo da resposta.
Strict-Transport-Security	Exigir conexões por HTTPS e proteger contra certificados falsificados.
X-Content-Type-Options: nosniff	Evitar que os navegadores executem a detecção de MIME e interpretem respostas como HTML de forma inadequada.
X-Frame-Options: DENY	Proteger contra ataques de *clickjack* do estilo *"drag and drop"*.

Os cabeçalhos abaixo destinam-se apenas a fornecer segurança adicional quando as respostas são processadas como HTML. Dessa forma, se a API não irá HTML nas respostas, esses cabeçalhos podem ser desconsiderados. No entanto, se houver alguma incerteza sobre a função dos cabeçalhos ou os tipos de informações que a API retorna (ou pode retornar no futuro), é recomendável incluí-los como parte de uma abordagem de defesa em profundidade.

Cabeçalho	Objetivo
Content-Security-Policy: default-src 'none'	A maior parte da funcionalidade do CSP afeta apenas as páginas renderizadas como HTML.

Feature-Policy: 'none'	As políticas de recursos afetam apenas as páginas processadas como HTML.
Referrer-Policy: no-referrer	As respostas não HTML não devem acionar solicitações adicionais.

Por fim, *Cross-Origin Resource Sharing* (CORS) é um padrão W3C para especificar com flexibilidade quais solicitações de domínio cruzado são permitidas. Ao entregar cabeçalhos CORS apropriados, sua API REST sinaliza ao navegador quais domínios, também conhecidos como origens, têm permissão para fazer chamadas *JavaScript* para o serviço REST. Com isso, recomenda-se a desativação dos cabeçalhos CORS se as chamadas entre domínios não forem suportadas / esperadas e ser o mais específico possível na definição das origens das chamadas entre domínios.

Apresentação dos Controles Técnicos de Segurança

API Gateway e controles contra negação de serviço

> **Observações:**
>
> É importante ressaltar que este controle é recomendável para o escopo *open-data* (fase 1) onde, mesmo que este considere uma API Pública com dados públicos, a infraestrutura deve ser controlada para evitar possíveis impactos na integridade e disponibilidade dos dados e ataques que comprometam o ambiente interno da seguradora e demais estruturas utilizadas futuramente no escopo de *customer-data*. Adicionalmente, este controle está relacionado com os itens 3, 4.3, 4.6, 4.8 e 6.1 do Manual de Segurança publicado pela SUSEP.

Um API Gateway é uma solução que fica na frente de uma *Application Programming Interface* (API) para facilitar solicitações e entrega de dados e serviços. Ele lida com as tarefas envolvidas na aceitação e processamento das chamadas simultâneas e isso inclui o gerenciamento de tráfego, suporte de compartilhamento de recursos de origem cruzada (CORS), autorização e controle de acesso, limitação, monitoramento e gerenciamento de versão de API.

Com isso, todas as APIs envolvidas no processamento dos dados (envio ou recebimento) no ecossistema do Open Insurance tenham um API Gateway implementado em sua arquitetura.

Controles contra negação de serviço

Adicionalmente, para evitar ataques de negação de serviço contra a API, **recomenda-se** que os seguintes sejam implementados:

Defesa à nível de rede: Se o *API gateway* está hospedado em uma solução em nuvem, então o mecanismo de defesa contra DDoS próprio do provedor deve ser implementado, por exemplo, a plataforma AWS fornece os serviços de AWS Shield, que inclui soluções de *load balancer*, distribuidor de trafego, entre outros serviços.

Rede para entrega de conteúdo *(Content Delivery Network)*: O CDN tem como objetivo fornecer alta disponibilidade e alto desempenho, distribuindo o serviço espacialmente em relação aos usuários finais. Podem ser usados para mitigar/minimizar ataques de negação de serviço na API.

Detecção de Bot: Diversas plataformas de gerenciamento de APIs tem serviços que monitoram o tráfego da API, identificam solicitações maliciosas / indesejadas e geram alertas / impedem que solicitações maliciosas chegam no API Gateway. Como exemplos temos o Apigee Sense (Google Cloud Plataform) que oferece uma camada de proteção adicional, identificando automaticamente comportamentos suspeitos da API, nos quais administradores podem aplicar ações corretivas para manter a experiência do usuário e proteger os sistemas de *backend*. Já na AWS, temos o AWS WAF, que é um firewall de aplicativo da web que pode ser implantado no CloudFront para ajudar a proteger seu aplicativo contra ataques DDoS, fornecendo controle sobre qual tráfego permitir ou bloquear, definindo regras de segurança.

Aplicação de políticas: As políticas devem ser aplicadas no *proxy* da API que fica entre um cliente da API e o *backend* para restringir o acesso da API a usuários legítimos. Os seguintes requisitos de política são ESSENCIAIS para proteger APIs de atacantes mal-intencionados:

- **Limite de taxa de API:** os limites de taxa de API são aplicados para reduzir solicitações massivas de API que causam negação de serviços e para mitigar possíveis ataques de força bruta ou mau uso de serviços. O seguinte mecanismo de limites de taxa da API deve ser considerado aplicável ao Proxy da API:
    - Limite de taxa por aplicação ou por API: Toda API ou aplicação só pode acessar o serviço considerando o número de máximo requisições definidas em uma janela de tempo pré-estabelecida.
    - Limites de taxa da API por solicitação GET ou POST: as solicitações de acesso permitido podem variar com base nas solicitações GET ou POST por período.

Regex Protection: O caminho da URL, Parâmetro da Query, Cabeçalho, Parâmetro de formulário, variável, XML *payload* ou JSON Payload da requisição de entrada devem ser avaliados em relação a expressões regulares predefinidas como DELETE, UPDATE e EXECUTE. A presença de qualquer uma dessas expressões deve ser tratada como uma ameaça e a solicitação deve ser rejeitada. Para validar expressões regulares, consulte o OWASP TOP 10.

Apresentação dos Controles Técnicos de Segurança

Validação de entrada JSON: A validação JSON em *payload* para solicitações PUT / POST / DELETE deve ser executada para minimizar o risco representado por ataques no nível do conteúdo, especificando limites em várias estruturas JSON, como profundidade máxima, número máximo de entradas de objetos, comprimento máximo da *string* de um nome, número máximo de elementos permitido em uma matriz.

Validação de entrada XML: A validação XML no *payload* para solicitações PUT / POST / DELETE deve ser executada para detectar ataques de carga XML com base nos limites configurados e na API da tela contra ameaças XML, usando as seguintes abordagens:

i. Validar mensagens contra XML Schema (.xsd);
ii. Avaliar o conteúdo da mensagem para palavras chaves ou padrões listados na blacklist;
iii. Detectar mensagens corrompidas ou malformada antes de serem analisadas.

Validação por método HTTP: Restrinja adequadamente os métodos permitidos, de modo que apenas os verbos permitidos funcionem, enquanto todos os outros retornam um código de resposta adequado (por exemplo, 403 Proibido ou 405 Método não permitido). Verificar se o requisitante tem permissão para utilizar os métodos HTTP recebido na coleta, ação e recuperação de recursos.

Validação de cabeçalho: Cabeçalhos como Content-Type, Accept, Content-Length devem ser explicitamente validados com relação à funcionalidade suportada pela API. Também deve ser realizada a validação em cabeçalhos obrigatórios, como autorização, cabeçalhos específicos da API.

Validação do Content-Type: Para solicitações PUT / POST / DELETE, o Content-Type (por exemplo, aplicativo / XML ou aplicativo / JSON) da solicitação de entrada e o valor do cabeçalho Content-Type deve ser o mesmo. Um cabeçalho de Content-Type ausente ou um cabeçalho de Content-Type inesperado deve resultar na rejeição da API pelo conteúdo com uma resposta 406 não aceitável.

Validação do tipo de resposta: NÃO copie o cabeçalho Accept no cabeçalho Content-type da resposta. Rejeite a solicitação (idealmente com uma resposta 406 não aceitável) se o cabeçalho Accept não contiver especificamente um dos tipos permitidos.

Lidar com recursos não suportados: Restrinja adequadamente os recursos permitidos, de modo que apenas os recursos expostos funcionem, enquanto todos os outros recursos não implementados retornem um código de resposta adequado, por exemplo recurso desconhecido.

Controle de Acesso: As políticas podem ser configuradas para permitir solicitações de IPs, domínios ou regiões específicas. Solicitações que não atendem a esses critérios são rejeitadas pelo *gateway*. Outro método de controle de acesso que pode ser aplicado é o *Role Based Access Control* (RBAC), no qual funções são coleções de autorizações concedidas aos usuários da sua API. Nesse modelo, é recomendado que minimamente a criação de perfis conforme a necessidade e com a aplicação do conceito de privilégio mínimo, ou seja, que permita que o usuário só tenha acesso ao essencial para realização da função desejada.

Apresentação dos Controles Técnicos de Segurança

Monitoramento de desempenho e disponibilidade

> **Observações:**
>
> É importante ressaltar que este controle foi proposto seguindo os padrões do Open Banking e ele é obrigatório para o escopo *open-data* (fase 1). Adicionalmente, este controle está relacionado com os itens 3, 5.2, 5.3 e 6.1 do Manual de Segurança publicado pela SUSEP.

O repositório de monitoramento de desempenho e disponibilidade visa armazenar e disponibilizar os dados estatísticos de indicadores de desempenho e disponibilidade dos Participantes do Open Insurance, permitindo que qualquer usuário consulte na página pública do Open Insurance os indicadores de desempenho e disponibilidade das APIs regulatórias dos Participantes.

O Diretório deverá obter os indicadores de desempenho e disponibilidade de todos os participantes registrados, através do consumo de uma API cadastrada do Participante.

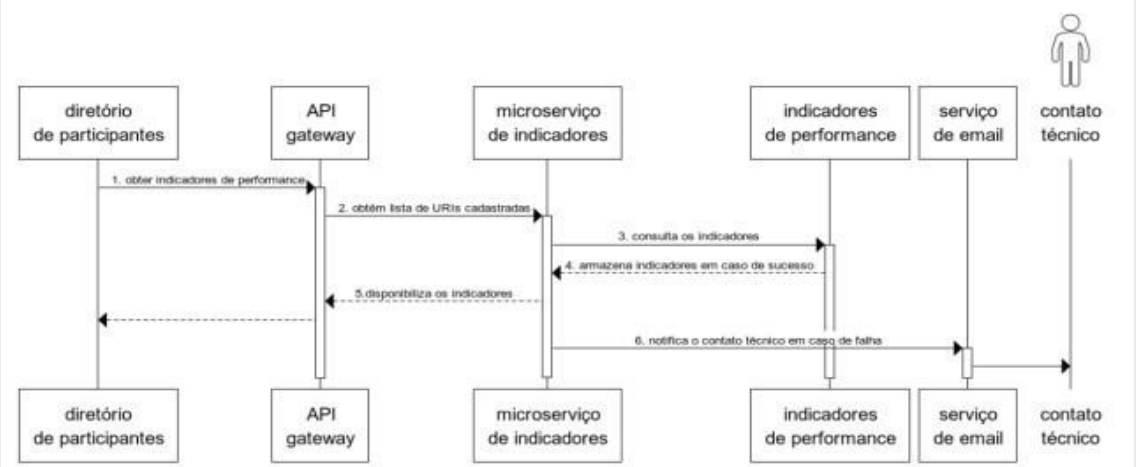
O Diretório deverá notificar, através de e-mail, o Contato Técnico do Participante no caso da indisponibilidade da API com os dados de desempenho e disponibilidade no período exigido.

O Diretório deverá permitir a parametrização da periodicidade do consumo dos indicadores de desempenho e disponibilidade dos Participantes.

As informações do repositório de monitoramento de todas as instituições Participantes devem estar disponíveis para os participantes do diretório via API, permitindo que elas realizem suas análises de desempenho e disponibilidade.

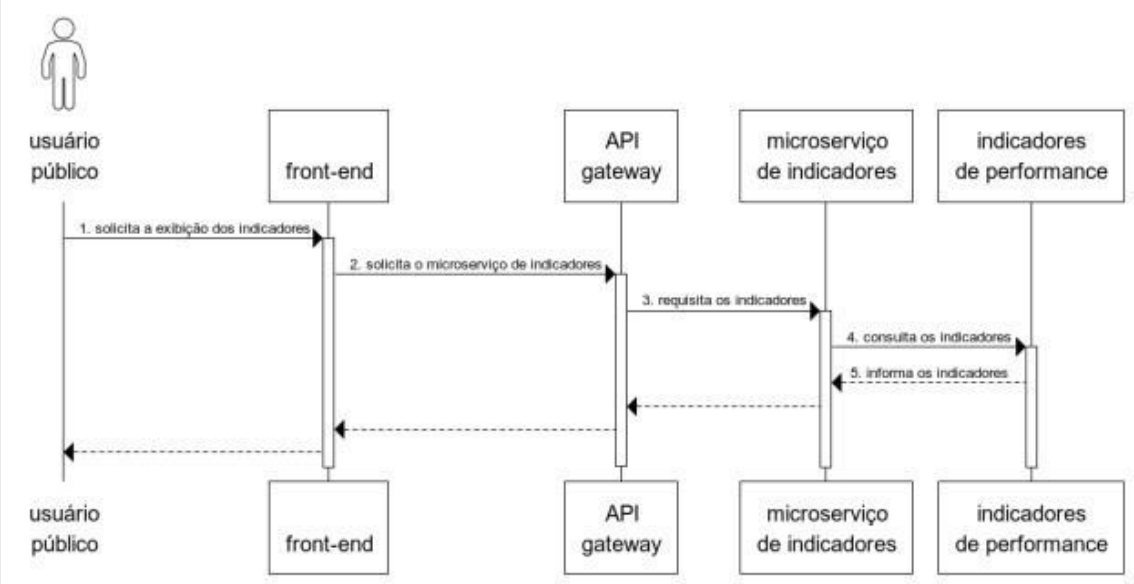
Devem ser disponibilizados dashboards e indicadores na página pública do Open Insurance, que apresentem as informações de disponibilidade geral das APIs, quantidade de chamadas ao longo do tempo, tempo médio de resposta e os indicadores pré-definidos pelo regulamento do Open Insurance. A granularidade da informação a ser disponibilizada deve ser definida pela estrutura inicial responsável pela implementação do Open Insurance.

Notificação dos indicadores de desempenho e disponibilidade



Apresentação dos Controles Técnicos de Segurança

Obtenção dos indicadores de desempenho e disponibilidade



Apresentação dos Controles Técnicos de Segurança

Gestão de Vulnerabilidades e Testes de Segurança

> **Observações:**
>
> É importante ressaltar que este controle é obrigatório para o escopo *open-data* (fase 1) e está relacionado com os itens 3, 4.18, 4.19, 4.20, 7.11 e 7.12 do Manual de Segurança publicado pela SUSEP.

O processo de Gestão de Vulnerabilidades é um **requisito obrigatório** e tem como fundamento identificar, classificar, remediar e gerenciar os riscos em diferentes camadas da arquitetura de segurança do Open Insurance. Uma vulnerabilidade de segurança pode ser definida como: "Uma fraqueza ou falha em um sistema ou aplicação que poderia permitir a um atacante comprometer a integridade, disponibilidade ou confidencialidade desse sistema ou aplicação".

Com isso, com o objetivo de minimizar o risco da exploração de vulnerabilidades críticas nos ativos envolvidos na estrutura do Open Insurance, os seguintes testes de segurança devem ser realizados:

- **Teste de Invasão:** Teste simula o cenário de uma invasão real para mapear a superfície de ataque. Neste caso, o escopo do teste de invasão deve considerar todos os ativos que suportam, de forma direta ou indireta, a estrutura do Open Insurance e é esperado que seja realizado no **mínimo anualmente** considerando, a infraestrutura, aplicações e APIs utilizadas.
- **Análise *(scan)* de Vulnerabilidades:** Realização de validações automatizadas de forma recorrente, no **mínimo trimestralmente**, para monitorar os ativos de infraestrutura, aplicações e APIs utilizados na estrutura do Open Insurance em busca de vulnerabilidades de segurança.

Além disso, é recomendado que o processo de gestão de vulnerabilidades contemple as seguintes etapas:

i. **Preparação** – Como um precursor do ciclo de vida da Gestão de Vulnerabilidades, na etapa de Preparação, é esperada a definição da base da governança do programa, estabelecendo políticas e padrões, atribuindo funções e responsabilidades, definindo métricas de desempenho e níveis de serviço e estabelecendo o escopo do programa.
ii. **Identificação** - Esta fase engloba a identificação de vulnerabilidades de ativos em todo o ambiente usando diferentes métodos, incluindo análise de vulnerabilidades, revisão da configuração, testes de invasão e a revisão de alertas de vulnerabilidade de segurança de fornecedores e alertas públicos. Convém que as vulnerabilidades descobertas sejam registradas e rastreadas ao longo de todo o ciclo de vida do programa de Gestão de Vulnerabilidades.
iii. **Priorização** – Após identificação, as vulnerabilidades deverão ser analisadas utilizando um padrão formal de risco que leva em consideração a severidade da vulnerabilidade e a classificação do ativo. Como resultado desta análise, será definido o plano de ação e prazos para remediação.
iv. **Remediação** – Para mitigar a vulnerabilidade identificada, será realizada a implementação dos controles definidos no plano de ação.
v. **Validação** – Com o objetivo de assegurar que as vulnerabilidades foram remediadas, sugere-se a realização de validações (retestes) após a implementação.
vi. **Melhoria Contínua** – Convém que os indicadores do programa sejam monitorados de forma recorrente para assegurar a qualidade e melhoria contínua.

Apresentação dos Controles Técnicos de Segurança

Monitoramento de Segurança

> **Observações:**
>
> É importante ressaltar que este controle é obrigatório para o escopo *open-data* (fase 1) e está relacionado com os itens 3, 5.2, 5.3, 6.1 e 7.9 do Manual de Segurança publicado pela SUSEP.

O processo de monitoramento tem como objetivo trazer visibilidade e proteção sob os ativos de infraestrutura relacionados a arquitetura do Open Insurance por meio de implementação de recursos de proteção contra intrusão de rede, dispositivos e monitoramento e correlação dos registros de alertas de segurança.

Recomenda-se que eles devem ser implantados em pontos de entradas e saída da arquitetura do Open Insurance para impedir e controlar a propagação de execução de ataques de rede ou de host, incluindo movimentação lateral, sendo eles:

i. **Centralizar os eventos de alertas de segurança** – Centralize os alertas de eventos de segurança entre os ativos relacionados a arquitetura do Open Insurance por meio do uso de um SIEM e sua integração com soluções de segurança.
ii. **Sistema de detecção de intrusão baseado em rede e em dispositivo** – Utilizar ferramentas de IPS na borda da rede da arquitetura do Open Insurance quando apropriado.
iii. **Segmentação de rede** – Filtrar o tráfego entre segmentos de rede que tenham comunicação com a rede da arquitetura do Open Insurance.
iv. **Coleta de tráfego de rede** – Coletar e revisar o tráfego de rede relacionado a arquitetura do Open Insurance.
v. **Revisar níveis de alertas** – Revisar os níveis definidos como gatilhos de alertas mensalmente ou sempre que necessário.

Apresentação dos Controles Técnicos de Segurança

Proteção contra *Malware* e Ameaças

> **Observações:**
>
> É importante ressaltar que este controle é obrigatório para o escopo *open-data* (fase 1) e está relacionado com os itens 3 e 4,9 do Manual de Segurança publicado pela SUSEP e a Circular Susep Nº 638.

As sociedades participantes do Open Insurance devem possuir ou implementar defesas contra Malwares e outras possíveis ameaças cibernéticas. As defesas contra *malware* e ameaças devem operar em ambientes dinâmicos por meio de automações, atualizações e integrações com outros processos como gerenciamento de vulnerabilidades e resposta a incidentes.

Os controles de proteção contra ameaça de rede e de *malware* são **obrigatórios** e devem ser implantados em pontos de entradas da arquitetura do Open Insurance para impedir e controlar a propagação de execução de *software* malicioso, sendo eles:

i. **Instalação e configuração** – Instalar e manter *software* de proteção contra *malware*.
ii. **Proteção de comportamento** – Utilizar *softwares* de proteção baseado em comportamento suspeitos ou maliciosos.
iii. **Atualização** – Garantir que os *softwares* de proteção estejam atualizados.
iv. **Escopo de Implementação** – A proteção contra *malware* e ameaças deve ser implementada na camada de rede e *endpoint*.

As sociedades participantes também devem possuir um plano de ação e resposta a incidentes capaz de abranger os procedimentos e os controles a serem utilizados na prevenção e resposta a incidentes que afetem sistemas, APIs e outros recursos relacionados à implementação e à operação do Open Insurance.

Também será necessário documentar em relatório anual a efetividade da prevenção e tratamento de incidentes feitos pela companhia, para maior controle da gerência de dados que se encontram em posse da supervisionada, bem como a comunicação, no prazo máximo de 5 (cinco dias) úteis à SUSEP, a partir da descoberta do evento, a ocorrência de incidentes relevantes, detalhando a extensão do dano causado e, se aplicável, as ações em curso para regularização total da situação, os respectivos responsáveis e os prazos.

Apresentação dos Controles Técnicos de Segurança

Registros de Auditoria e Sistemas

> **Observações:**
>
> É importante ressaltar que este controle é obrigatório para o escopo *open-data* (fase 1) e está relacionado com os itens 4.17, 4.25, 5.1 e 7.4 do Manual de Segurança publicado pela SUSEP.

O processo de coleta de *logs* é fundamental para detectar atividades suspeitas ou maliciosas. Os registros de auditoria podem ser a única evidência de um ataque bem sucedido. Existem dois tipos de logs que devem ser configurados na arquitetura do Open Insurance, sendo eles:

- **Registros de sistema:** Fornecem eventos em nível de sistema que mostram o horário do início e fim de processos de sistema, quebras dos sistemas. Esses *logs* são nativos dos sistemas operacionais e levam menor tempo de configuração.
- **Registros de auditoria:** Os registros de auditoria incluem eventos em nível de usuários, *logins* de usuários, acesso a arquivos e exige maior planejamento e esforço para configuração.

A retenção desses registros também é importante na necessidade de uma investigação de incidente.

Além disso, é **obrigatório** que o processo de coleta dos *logs* contemple as seguintes etapas:

i. **Padronizar a sincronização de tempo** – Configurar pelo menos duas fontes de sincronismo de tempo.
ii. **Configurar os registros de auditoria** – Configurar o registro de auditoria detalhado para ativos que contenham dados confidenciais. Incluir fonte do evento, data, nome de usuário, *timestamp*, endereço de origem, endereço de destino e outros elementos úteis.
iii. **Coleta de registros** – Coletar registros de auditoria de requisições das APIs URLs publicadas e de linhas de comando como PowerShell, Bash e terminais de administração remota.
iv. **Armazenamento de registros de auditoria** – Os registros devem ser armazenados por no mínimo um ano para auxiliar no processo de revisão ou investigação de incidentes de segurança.

Referências

Esta seção irá apresentar os materiais utilizados como referência para proposição dos controles técnicos:

OBIE. Open Banking Directory. Disponível em: https://www.openbanking.org.uk/providers/directory/. Acesso em: 20 jul. 2020.

Repositório do Open Banking Brasil. Disponível em: https://openbanking-brasil.github.io/areadesenvolvedor/#seguranca. Acesso em: 11 ago. 2021.

SÃO PAULO, FEBRABAN. Open Banking - Diretório de Participante - Especificação Funcional. Disponibilizado por e-mail. 2021.

SÃO PAULO, FEBRABAN. Open Banking - Diretório de Participante - Especificação Técnica. Disponibilizado por e-mail. 2021.

SÃO PAULO, FEBRABAN; OLIVER WYMAN. Open Banking – Proposta de Infraestrutura. Disponibilizado por e-mail. 2021.

Gartner. A Guidance *Framework* for Developing and Implementing Vulnerability Management. Acesso em: 11 ago, 2021.

OWASP Secure Headers Project. Disponível em: https://wiki.owasp.org/index.php/OWASP_Secure_Headers_Project#tab=Headers. Acesso em: 11 ago. 2021.

OWASP API Security Project. Disponível em: https://owasp.org/www-project-api-security/. Acesso em: 11 ago. 2021.

CIS Controls V8. Disponível em: https://www.cisecurity.org/controls/v8/. Acesso em: 11 ago. 2021.

Marco Civil da Internet. Disponível em: http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2014/lei/l12965.htm. Acesso em: 23 ago. 2021.

Manual de Segurança do Open Insurance. Disponível em: Manual-de-Seguranca.pdf (susep.gov.br). Acessado em: 21 mar. 2022.

circular Susep Nº 638, De 27 De Julho De 2021 - CIRCULAR SUSEP Nº 638, DE 27 DE JULHO DE 2021 - DOU - Imprensa Nacional (in.gov.br) Acessado em 31 mar. 2022