Requisitos Operacionais Mínimos do Prestador de Serviço de Confiança SyngularID

(ROPSC SyngularID)

Versão 1.0

Fevereiro de 2023

Classificação da informação: Pública

Sumário

CONTROLE DE ALTERAÇÕES

Versão	Data	Resolução que aprovou a alteração	Item alterado	Descrição da alteração
1.0	Fevereiro/2023	-	Não há	Versão inicial – Baseada no DOC-ICP-17 versão 2.0

LISTA DE ACRÔNIMOS

SIGLA	DESCRIÇÃO
AC	Autoridade Certificadora
AC RAIZ	Autoridade Certificadora Raiz da ICP-Brasil
ACT	Autoridade de Carimbo do Tempo
AES	*Advanced Encryption Standard*
APF	Administração Pública Federal
API	*Application Programming Interface*
CAdES	*CMS Advanced Electronic Signature*
DPPSC	Declaração de Prática do Prestador de Serviço de Confiança
EAT	Entidade de Auditoria do Tempo – ICP-Brasil
ETSI	*European Telecommunications Standards Institute*
HMAC	*Hash-based Message Authentication Code*
HOTP	*HMAC-Based One-Time Password*
HSM	*Hardware Security Module*
HTTPS	*Hyper Text Transfer Protocol Secure*
ICP-BRASIL	Infraestrutura de Chaves Públicas Brasileira
IETF	*Internet Engineering Task Force*
ITI	Instituto Nacional de Tecnologia da Informação
JSON	*JavaScript Object Notation*
KMIP	*Key Management Interoperability Protocol*
LPA	Lista de Políticas de Assinatura Aprovadas
LPSC	Lista de Prestadores de Serviço de Confiança
OATH	*Open Authentication*
PAdES	*PDF Advanced Electronic Signature*
PCO	Planejamento de Capacidade Operacional
PIN	*Personal Identification Number*
PSBio	Prestador de Serviço Biométrico
PSC	Prestador de Serviço de Confiança
PKCS	*Public Key Cryptography Standards*
PUK	*PIN Unlock*
RFC	*Request for Comments*
SSL	*Secure Sockets Layer*
TLS	*Transport Layer Security*
TOTP	*Time-based One-Time Password algorithm*
TRC	Teorema do Resto Chinês
XAdES	*XML Advanced Electronic Signatures*
XML	*eXtensible Markup Language*
XMPP	*Extensible Messaging and Presence Protocol*

1. DISPOSIÇÕES GERAIS

1.1. Este documento tem por finalidade regulamentar os requisitos mínimos de segurança e os procedimentos operacionais a serem adotados pelo Prestadores de Serviço de Confiança SyngularID – PSC SyngularID, no âmbito da Infraestrutura de Chaves Públicas Brasileira (ICP-Brasil).

1.2. Suplementa, para o PSC SyngularID, os regulamentos contidos nos documentos DOC-ICP-03 [1], DOC- ICP-04 [2], DOC-ICP-08 [3] e DOC-ICP-09 [4], tomando como base também a Política de Segurança da ICP-Brasil – DOC-ICP-02 [5].

1.3. Os requisitos contidos neste documento foram apresentados quando do credenciamento do PSC SyngularID para armazenamento de chaves privadas dos usuários finais e são mantidos atualizados durante seu funcionamento enquanto estiver credenciado na ICP-Brasil.

1.4. O PSC SyngularID possui uma Política de Segurança da Informação composta por diretrizes, normas e procedimentos que descrevem os controles de segurança que são seguidos em suas dependências e atividades, em consonância com o DOC-ICP-02 [5].

1.5. Há um exemplar da Política de Segurança da Informação, no formato impresso, disponível para consulta no Nível 1 (vide regulamento no item 3) de segurança do PSC SyngularID.

1.6. A Política de Segurança da Informação do PSC SyngularID é seguida por todo pessoal envolvido nas atividades realizadas pelo PSC, do seu próprio quadro ou contratado.

1.7. Este documento define normas operacionais e de segurança que são aplicadas nas áreas internas ao PSC SyngularID, assim como no trânsito de informações, armazenamento de chaves privadas e materiais com entidades externas.

1.8. A seguir são informados os requisitos observados quanto a segurança de pessoal, segurança física, segurança lógica, segurança de rede, requisitos mínimos para armazenamento de chaves privadas, classificação da informação, salvaguarda de ativos da informação, gerenciamento de riscos, plano de continuidade de negócios, análise de registros de eventos e plano de capacidade operacional.

2. SEGURANÇA PESSOAL

2.1. O PSC SyngularID possui uma Política de Gestão de Pessoas que dispõe sobre os processos de contratação, demissão, descrição de cargos, avaliação de desempenho e capacitação.

2.2. A comprovação da capacidade técnica do pessoal envolvido nos serviços prestados pelo PSC está à disposição para eventuais auditorias e fiscalizações.

2.3. Todo pessoal envolvido nas atividades realizadas pelo PSC, do próprio quadro ou contratado, assina um termo, com garantias jurídicas, que garante o sigilo das informações internas e de terceiros, mesmo após a demissão ou o término do contrato.

2.4. O termo de sigilo da informação contém cláusula explícita de responsabilização nos casos de quebra de regras ou regulamentos da ICP-Brasil.

2.5. Aplica-se o termo de sigilo de informações a quaisquer outras entidades que porventura tenham acesso às informações internas e de terceiros originárias dos projetos coordenados pelo PSC SyngularID.

2.6. O PSC SyngularID possui procedimentos formais de apuração e responsabilização em caso de descumprimento das regras estabelecidas pelas suas políticas ou pelas normas da ICP-Brasil.

2.7. O quadro de pessoal do PSC SyngularID e contratados possuem um dossiê contendo os seguintes documentos:

(a) contrato de trabalho ou cópia das páginas da carteira de trabalho onde conste o registro da contratação, termo de posse de servidor ou comprovante de situação funcional;
(b) comprovante da verificação de antecedentes criminais;
(c) comprovante da verificação de situação de crédito;
(d) comprovante da verificação de histórico de empregos anteriores;
(e) comprovação de residência;
(f) comprovação de capacidade técnica;
(g) resultado da entrevista inicial, com a assinatura do entrevistado;
(h) declaração em que afirma conhecer as suas atribuições e em que assume o dever de cumprir as regras aplicáveis da ICP-Brasil;
(i) termo de sigilo.

2.8. Não são admitidos estagiários no exercício das atividades do PSC SyngularID.

2.9. Quando da demissão, o referido contém os seguintes documentos:

(a) Evidências de exclusão dos acessos físico e lógico nos ambientes do PSC SyngularID;
(b) Declaração assinada pelo empregado ou servidor de que não possui pendências, conforme previsto no item sobre processo de liberação do DOC-ICP-02 [5].

3. SEGURANÇA FÍSICA

3.1. Disposições Gerais de Segurança Física

3.1.1. Níveis de acesso

3.1.1.1. São definidos pelo menos 4 (quatro) níveis de acesso físico aos diversos ambientes do PSC SyngularID.

3.1.1.1.1. O primeiro nível – ou nível 1 – situa-se após a primeira barreira de acesso às instalações do PSC SyngularID. O ambiente de nível 1 do PSC na ICP-Brasil desempenha a função de interface com cliente ou fornecedores que necessita comparecer ao PSC.

3.1.1.1.2. O segundo nível – ou nível 2 – é interno ao primeiro e deverá requer a identificação individual das pessoas que nele entram. Esse é o nível mínimo de segurança requerido para a execução de qualquer processo operacional ou administrativo do PSC SyngularID. A passagem do primeiro para o segundo nível exige identificação por meio eletrônico e o uso de crachá.

(a) o ambiente de nível 2 é separado do nível 1 por paredes divisórias de escritório, alvenaria ou pré-moldadas de gesso acartonado. Não há janelas ou outro tipo qualquer de abertura para o exterior, exceto a porta de acesso;

(b) o acesso a este nível é permitido apenas a pessoas que trabalhem diretamente com as atividades de serviços de armazenamento dos certificados para usuários finais ou ao pessoal responsável pela manutenção de sistemas e equipamentos do PSC, como administradores de rede e técnicos de suporte de informática. Demais funcionários do PSC SyngularID ou do possível ambiente que esta compartilhe não deverão acessar este nível;

(c) *nobreaks*, geradores e outros componentes da infraestrutura física estão abrigados neste nível, para evitar acessos ao ambiente de nível 3 por parte de prestadores de serviços de manutenção;

(d) excetuados os casos previstos em lei, o porte de armas é admitido nas instalações do PSC SyngularID, a partir do nível 2. A partir desse nível, equipamentos de gravação, fotografia, vídeo, som ou similares, bem como computadores portáteis, tem sua entrada controlada e somente poderão ser utilizados mediante autorização formal e sob supervisão.

3.1.1.1.3. O terceiro nível – ou nível 3 – situa-se dentro do segundo e é o primeiro nível a abrigar material e atividades sensíveis da operação do PSC. Qualquer atividade relativa ao armazenamento de chaves privadas dos usuários é realizada a partir deste nível. Somente pessoas autorizadas podem permanecer nesse nível.

(a) no terceiro nível são controladas tanto as entradas quanto as saídas de cada pessoa autorizada. Dois tipos de mecanismos de controle são requeridos para a entrada nesse nível: algum tipo de identificação individual, como cartão eletrônico, e identificação biométrica ou digitação de senha;

(b) as paredes que delimitam o ambiente de nível 3 são de alvenaria ou material de resistência

equivalente ou superior. Não há janelas ou outro tipo qualquer de abertura para o exterior, exceto a porta de acesso;

(c) caso o ambiente de Nível 3 possua forro ou piso falsos, deverão ser adotados recursos para impedir o acesso ao ambiente por meio desses, tais como grades de ferro estendendo-se das paredes até as lajes de concreto superior e inferior;

(d) existe uma porta única de acesso ao ambiente de nível 3, que abre somente depois que o funcionário tenha se autenticado eletronicamente no sistema de controle de acesso. A porta possui dobradiças que permitem a abertura para o lado externo, de forma a facilitar a saída e dificultar a entrada no ambiente, bem como de mecanismo para fechamento automático, para evitar que permaneça aberta mais tempo do que o necessário;

3.1.1.1.4. Não se aplica

3.1.1.1.5. O quarto nível – ou nível 4 – interior ao terceiro, é onde ocorrem atividades especialmente sensíveis da operação do PSC de armazenamento de chaves privadas. Todos os sistemas e equipamentos necessários a essas atividades estão localizados a partir desse nível. O nível 4 possui os mesmos controles de acesso do nível 3 e, adicionalmente, exige, em cada acesso ao seu ambiente, a identificação de, no mínimo, 2 (duas) pessoas autorizadas. Nesse nível, a permanência dessas pessoas é exigida enquanto o ambiente estiver ocupado.

3.1.1.1.6. No quarto nível, todas as paredes, piso e teto são revestidos de aço e concreto ou de outro material de resistência equivalente. As paredes, piso e o teto são inteiriços, constituindo uma célula estanque contra ameaças de acesso indevido, água, vapor, gases e fogo. Os dutos de refrigeração e de energia, bem como os dutos de comunicação, não permitem a invasão física áreas de quarto nível. Adicionalmente, esses ambientes de nível 4 – que constituem as chamadas salas-cofre - possuem proteção contra interferência eletromagnética externa.

3.1.1.1.7. As salas-cofre são construídas segundo as normas brasileiras aplicáveis. Eventuais omissões dessas normas são sanadas por normas internacionais pertinentes.

3.1.1.2. Podem existir, no PSC SyngularID, vários ambientes de quarto nível para abrigar e segregar, quando for o caso:

(a) equipamentos de produção *on-line*; e

(b) equipamentos de rede e infraestrutura (*firewall*, roteadores, *switches* e servidores).

3.1.1.3. Todos os servidores e elementos de infraestrutura e proteção do segmento de rede, tais como roteadores, *hubs*, *switches* e *firewalls*:

(a) operam em ambiente com segurança equivalente, no mínimo, ao quarto nível citado neste documento;

(b) possuem acesso lógico restrito por meio de sistema de autenticação e autorização de acesso.

3.1.1.4. O PSC SyngularID atende aos seguintes requisitos:

(a) o ambiente físico do PSC SyngularID contém dispositivos que autenticam e registram o acesso de pessoas informando data e hora desses acessos;

(b) o PSC SyngularID contém imagens que garantem a identificação de pessoas quando do acesso físico em qualquer parte de seu ambiente;

(c) é realizado o sincronismo de data e hora entre os mecanismos de segurança física garantindo a trilha de auditoria entre dispositivos de controle de acesso físico e de imagem;

(d) todos que transitam no ambiente físico do PSC SyngularID portam crachás de identificação, inclusive os visitantes;

(e) só é permitido o trânsito de material de terceiros pelos ambientes físicos do PSC SyngularID mediante registro, garantindo a trilha de auditoria com informações de onde o material passou, a data e hora que ocorreu o trânsito e quem foi o responsável por sua manipulação;

(f) o PSC SyngularID contém dispositivos de prevenção e controle de incêndios, temperatura, umidade, iluminação e oscilação na corrente elétrica em todo seu ambiente físico;

(g) todo material crítico inservível, descartável ou não mais utilizável tem tratamento especial de destruição, garantindo o sigilo das informações lá contidas. O equipamento enviado para manutenção tem seus dados apagados, de forma irreversível, antes de ser retirado do ambiente físico do PSC SyngularID;

(h) os computadores pessoais, servidores e dispositivos de rede, e seus respectivos *softwares*, estão inventariados com informações que permitem a identificação inequívoca;

(i) em caso de inoperância dos sistemas automáticos, o controle de acesso físico é realizado provisoriamente por meio de um livro de registro onde consta quem acessou, a data, hora e o motivo do acesso;

(j) existem mecanismos que garantem a continuidade do fornecimento de energia nas áreas críticas, mantendo os ativos críticos de informação em funcionamento até que todos os processos e dados sejam assegurados caso o fornecimento de emergência se esgote;

(k) no caso de armazenamento de chaves privadas para usuários finais, existem dois ambientes físicos, sendo obrigatoriamente um para operação e outro para contingência;

(l) O PSC SyngularID utiliza ambiente de nível 4 da AC SyngularID para abrigo do *hardware* criptográfico que armazenará as chaves privadas dos usuários finais, em gabinete cadeado, cuja chave do cadeado está em posse de funcionário distinto dos perfis lógicos do PSC, segregados dos que operam o ambiente da AC;

(m) Todos os equipamentos e ambiente computacional que são utilizados no PSC SyngularID tem sua data e horário sincronizados com a EAT.

4. SEGURANÇA LÓGICA

4.1. O acesso lógico ao ambiente computacional do PSC SyngularID se dá no mínimo mediante usuário individual e senha, que é trocada periodicamente;

4.2. Todos os equipamentos do parque computacional têm controle de forma a permitir somente o acesso lógico a pessoas autorizadas;

4.3. Os equipamentos têm mecanismos de bloqueio de sessão inativa;

4.4. O PSC SyngularID tem explícita a política de cadastro, suspensão e remoção de usuários em seu ambiente computacional. Os usuários estão cadastrados em perfis de acesso que permitam privilégio mínimo para realização de suas atividades;

4.5. Os usuários especiais (a exemplo do *root* e do administrador) de sistemas operacionais, do *hardware* criptográfico, do banco de dados e de aplicações em geral têm suas senhas segregadas de forma que o acesso lógico a esses ambientes se dê por, pelo menos, duas pessoas autorizadas;

4.6. Todo equipamento do PSC SyngularID possui *log* ativo e seu horário sincronizado com uma fonte confiável de tempo da ICP-Brasil;

4.7. As informações como log, trilhas de auditoria (do armazenamento de chaves privadas), registros de acesso (físico e lógico) e imagens possuem cópia de segurança cujo armazenamento é de, no mínimo, 7 anos;

4.8. Os *softwares* dos sistemas operacionais, os antivírus e aplicativos de segurança são mantidos atualizados;

4.9. É vedado qualquer tipo de acesso remoto dos operadores do PSC SyngularID ao ambiente de nível 4.

5. SEGURANÇA DE REDE

5.1. O tráfego das informações no ambiente de rede é protegido contra danos ou perdas, bem como acesso, uso ou exposição indevidos;

5.2. Não são admitidos acessos externos à rede interna do PSC SyngularID. As tentativas de acessos externos são inibidas e monitoradas por meio de aplicativos que criam barreiras e filtros de acesso, assim como mecanismos de detecção de intrusão;

5.3. São aplicados testes de segurança na rede interna e externa com aplicativos especializados com periodicidade de, no mínimo, uma vez a cada mês. Os testes na rede são documentados e as vulnerabilidades detectadas corrigidas.

6. REQUISITOS PARA ARMAZENAMENTO DE CHAVES PRIVADAS

6.1. **Armazenamento das chaves e certificados digitais**

6.1.1 As chaves privadas dos usuários finais, para os tipos de certificados que obrigatoriamente devem ser gerados e armazenados em *hardware* criptográficos, estão armazenadas dentro dos espaços (*slots*), ou equivalente, da fronteira criptográfica e segurança física de um HSM com certificação Inmetro válida no âmbito da ICP-Brasil, endereçados por conta de usuário;

6.1.2 Esse acesso ou comando de exportação às chaves privadas dos usuários é de uso, conhecimento e controle exclusivo do titular, sem a possibilidade de ingresso por outros titulares no mesmo HSM, qualquer funcionário do PSC SyngularID ou dependentes de outras chaves criptográficas;

6.1.3 O PSC SyngularID provê mecanismos de duplo fator de autenticação ao titular para acesso à chave privada, sendo que um fator fica dentro da fronteira criptográfica do HSM e outro dentro do ambiente seguro e primeira interface de comunicação com HSM ou ambos dentro da fronteira criptográfica do HSM. Cada fator é de uma classe diferente (conhecimento, posse ou biometria). Os mecanismos de autenticação empregam método ou protocolo de validação que protege a transmissão e os dados de autenticação por meio de criptografia. Essa funcionalidade será apensada aos requisitos técnicos na manutenção da certificação Inmetro dos HSM e são:

(a) Senhas (PIN/PUK): segundo regras da ICP-Brasil;
(b) OTP: segundo regras da RFC 6238 (TOTP), RFC 6287, RFC 4226 (HOTP);
(c) Biometria: segundo regras da ICP-Brasil;
(d) Certificado de atributo: segundo regras da ICP-Brasil;
(e) *Push Notification*: segundo regras do XMPP *extension protocol* ou semelhante;

(f) Outras autenticações semânticas em acordo com esse documento e previamente aprovadas pela AC Raiz.

6.1.4 É realizada, em outro ambiente físico de contingência, a cópia das chaves dos usuários finais, observados os mesmos requisitos de armazenamento do ambiente principal. A entrada do ambiente de contingência deve ser em até 48 horas.

6.1.5 Esses espaços para armazenamento das chaves privadas dos usuários finais podem ser liberados desde que não haja renovação por parte do mesmo ou a revogação da chave, entretanto mantém-se o registro de armazenamento das chaves conforme Declaração de Prática do Prestador de Serviço de Confiança – DPPSC SyngularID.

6.2. **Protocolos**

6.2.1 Os HSMs certificados na ICP-Brasil devem suportar a interface PKCS#11, atendendo às exigências de especificação da ICP-Brasil, além dos relatados nesse documento, os seguintes requisitos:

(a) Gerar chaves simétricas especificando os componentes de chaves simétricas em texto claro;

- Gerar par de chaves especificando os componentes de chaves assimétricas em texto claro. Por exemplo os componentes Módulo, Expoente público, tamanho em *bits* etc.;
- Gerar objeto de chaves especificando os componentes de chaves assimétricas (no mínimo chave pública) em texto claro. Por exemplo os componentes: Módulo, Expoente público, Expoente Privada em forma reduzida ou em forma de TRC (Teorema de Resto Chinês);
- Cifrar e decifrar chaves especificando os componentes de chaves simétricas ou assimétrica em texto claro;
- Exportar e importar chaves (PKCS#12) especificando os componentes de chaves assimétricas privadas criptografados;
- Assinar conteúdo especificando os componentes de chaves assimétricas públicas em texto claro;
- Verificar assinatura especificando os componentes de chaves assimétricas públicas em texto claro.

(b) O módulo criptográfico deve suportar as seguintes chamadas de PKCS#11 (*Cryptoki*)

- *C_Initalize*
- *C_Finalize*
- *C_OpenSession*
- *C_CloseSession*
- *C_Init_Token*
- *C_Init_PIN*
- *C_Login*

- *C_Logout*
- *C_CreateObject*
- *C_DestroyObject*
- *C_GetAttributeValue*
- *C_SetAttributeValue*
- *C_EncryptInit*
- *C_Encrypt*
- *C_DecryptInit*
- *C_Decrypt*
- *C_DigestInit*
- *C_Digest*
- *C_DigestKey*
- *C_SignInit*
- *C_Sign*
- *C_VerifyInit*
- *C_Verify*
- *C_GenerateKey*
- *C_GenerateKeyPair*
- *C_DeriveKey*
- *C_GenerateRandom*
- *C_WrapKey*
- *C_UnwrapKey*

(c) sendo obrigatória a implementação das seguintes funções:

- *C_GenerateKey* especificando *templates* de chaves simétricas;
- *C_GenerateKeyPair* especificando *templates* de chaves assimétricas;
- *C_Sign* para realizar assinatura de um conteúdo;
- *C_Verify* para verificar a assinatura de um conteúdo;
- *C_Encrypt* para cifrar um dado com uma chave já construída;
- *C_Decrypt* para decifrar um dado com uma chave já construída;
- *C_CreateObject* especificando *templates* de chaves assimétricas (no mínimo chave pública);
- *C_DestroyObject* especificando o *handle* do objeto.

6.2.2 Os HSMs certificados na ICP-Brasil devem suportar o protocolo *Key Management Interoperability Protoco*l – KMIP, versão 1.3 ou superior, devendo seguir, além dos relatados nesse documento, os seguintes requisitos:

6.2.2.1. O PSC SyngularID utiliza o conjunto de operações a seguir que se aplicam aos objetos gerenciados relacionados ao conjunto normativo do PSC e ao ciclo de vida das chaves, que por sua

vez consistem em atributos, como mostrado, em exemplo, na tabela a seguir.

Operações do Protocolo	Objetos Gerenciados	Atributos dos Objetos
Create	*Certificate*	*Unique Identifier*
Create Key Pair	*Symmetric Key*	*Name*
Register	*Public Key*	*Object Type*
Re-key	*Private Key*	*Cryptographic Algorithm*
Derive Key	*Split Key*	*Cryptographic Length*
Certify	*Secret Data*	*Cryptographic Parameters*
Re-certify	*Key Block* (para chaves) ou *Value* (para certificados)	*Certificate Type*
Locate		*Certificate Issuer*
Check		*Certificate Subject*
Get		*Digest*
Get Attributes		*Operation Policy Name*
Get Attribute List		*Cryptographic Usage Mask*
Add Attribute		*Lease Time*
Modify Attribute		*Usage Limits*
Delete Attribute		*State*
Obtain Lease		*Initial Date*
Get Usage Allocation		*Activation Date*
Activate		*Process Start Date*
Revoke		*Protect Stop Date*
Destroy		*Deactivation Date*
Archive		*Destroy Date*
Recover		*Compromise Occurrence Date*
Validate		*Compromise Date*
Query		*Revocation Reason*
Cancel		*Archive Date*
Poll		*Object Group*
		Link
		Application Specific ID
		Contact Information
		Last Change Date
		Custom Attribute

6.2.2.2. Os objetos base são:

(a) os componentes dos objetos gerenciados.

(i) Atributo: identificado pelo seu nome;

(ii) *Key Block,* contém o valor da chave;

(b) os elementos do protocolo de mensagens;

(c) os parâmetros das operações.

6.2.2.3. Os objetos criptográficos gerenciáveis são:

(a) Certificado, com o tipo e valor;
(b) Chave Simétrica, com o *Key Block*;
(c) Chave Pública, com o *Key Block*;
(d) Chave Privada, com o *Key Block*;
(e) Chave Dividida, com o par e o *Key Block*;
(f) Dados Reservados, com o tipo e o *Key Block*.

6.2.2.4. Os atributos contêm os metadados de um objeto gerenciável, nos quais:

(a) número identificador único, estado, entre outros;
(b) os atributos devem ser pesquisados com a operação "*locate".*

6.2.2.5. Os atributos podem ser configurados, modificados e apagados quando a especificação KMIP permitir esses pelo cliente.

6.2.2.6. Os valores das estruturas de codificações (TTLV, definição dos valores, *Text String*, *Structure, Byte String, Integer, Big Integer, Long Integer, Boolean*, *Date-Time* e *Enumerations*), dos campos dos objetos, dos atributos, dos formatos e conteúdo das mensagens, da manipulação de erros e dos parâmetros (solicitação e resposta) das operações cliente/servidor devem seguir integralmente o estabelecido neste documento e no *Key Management Interoperability Protocol Specification Version* 1.3, OASIS *Standard*, 27 *December* 2016, ou versionamento superior.

NOTA: O ITI poderá requisitar aos PSC em credenciamento ou credenciados testes dos modelos descritos, ou outras versões, nos sítios https://www.snia.org/forums/ssif/kmip, http://docs.oasisopen.org/kmip/profiles/v1.3/csd01/kmip-profiles-v1.3-csd01.html ou equivalente**.**

6.2.2.7. A criação do usuário deve seguir o estabelecido a seguir (xml):

```
<RequestMessage>
    <RequestHeader>
        <ProtocolVersion>
            <ProtocolVersionMajor type="Integer" value="1"/>
            <ProtocolVersionMinor type="Integer" value="3"/>
        </ProtocolVersion>
        <Authentication>
            <Credential>
```

```
                    <CredentialType type="Enumeration"
value="UsernameAndPassword"/>
                    <CredentialValue>
                        <Username type="TextString" value="vco_test"/>
                        <Password type="TextString" value="Teste112233$"/>
                    </CredentialValue>
                </Credential>
            </Authentication>
            <BatchCount type="Integer" value="1"/>
        </RequestHeader>
        <BatchItem>
            <Operation type="Enumeration" value="CreateUser"/>
                <RequestPayload>
                    <UserName type="TextString" value="labsec-pw"/>
                    <UserType type="Enumeration" value="User"/>
                </RequestPayload>
        </BatchItem>
</RequestMessage>
```

6.2.2.8. Para a operação do duplo fator de autenticação do titular da chave privada, poderá ser criada uma nova extensão ao tipo de credencial, conforme relatado a seguir:

6.2.2.9. Para o novo tipo de credencial deve ser configurado o seguinte:

(a) *Credential Type: TOKEN*

Object	*Encoding*	*Required*	*Description*
Credential Value	*Structure*		
Token	*Text String*	*Yes*	Valor atual do *"TOKEN"*

(b) fluxo de uso

(i) durante o credenciamento, o PSC deve requisitar a criação de um novo usuário (via KMIP), indicando que o mesmo necessita de um segundo fator de autenticação para utilizar seus objetos e cadastrando seu nome de usuário e senha. O PSC indica ao usuário como instalar seu aplicativo de *Token*.

(ii) o "*TOKEN*" do usuário deve ser inicializado para sincronizar seus dados. Esse processo pode ser feito pelo próprio usuário através do aplicativo de "*TOKEN*" via KMIP no momento da primeira conexão utilizando seu usuário e senha. O HSM gera então a chave

que será utilizada no "*TOKEN*".

(iii) na posse de seu "*TOKEN*" sincronizado e de seu usuário e senha, o usuário pode então criar sua chave no HSM utilizando a aplicação do PSC diretamente via comando KMIP.

(iv) o usuário já pode utilizar sua chave criada anteriormente utilizando o aplicativo do PSC, de posse de sua Senha + *Token*.

6.2.2.10. Este mecanismo de "*TOKEN*" deve ser configurado na área de execução segura do HSM.

NOTA: Pode ser encontrada mais referências sobre o protocolo KMIP no sítio https://www.oasisopen.org/committees/tc_home.php?wg_abbrev=kmip.

6.2.2.11. As soluções do PSC deverão garantir a portabilidade da chave privada do usuário conforme o descritivo:

(a) glossário:

CP_rU_i: Chave privada do usuário 'i', armazenada no HSM 1, a ser exportada e importada para o HSM 2;

$CP_rH_e^2$: Chave privada do HSM 2, a ser utilizada para importação de chaves privadas de usuários gravadas no HSM 1;

$CP_uH_e^2$: Chave Pública do HSM 2, utilizada para exportação de chaves privadas de usuários armazenadas no HSM 1, a serem importadas pelo HSM 2.

$CP_uH_e^2$ deve ser armazenada no repositório do ITI, seguindo procedimentos já estabelecidos ($CP_uH_e^2$ pode ser transformada em um certificado digital);

CS_i: Chave simétrica a ser gerada pelo HSM 1, para exportação da chave privada do usuário 'i', CP_rU_i. CS_i é utilizada para cifração da chave privada do usuário 'i';

$Algo_s$: Algoritmo criptográfico simétrico, de sigilo, pode ser o AES ou Serpent, com modo de operação CTR e tamanho de chave 256 bits.

(b) usuário deve solicitar, assinando digitalmente, uma requisição, que estará disponível no sítio dos PSCs, de portabilidade de sua chave privada, de exportação no PSC atual e de importação no PSC de destino.

(c) os PSCs receberão essa requisição e autorizarão essa portabilidade com os três perfis (administrador, auditor e operador). Assim que receber a autorização do usuário, PSC 1 e PSC 2 devem iniciar os procedimentos de exportação e importação.

(d) os PSCs devem estabelecer uma conexão ponta a ponta em um canal seguro de comunicação (HTTPS com dupla autenticação por certificado digital ICP-Brasil).

(e) Modo Operacional:

(i) procedimentos preliminares:

a. Cada PSC gera um par de chaves ([CP_uH_e, CP_rH_e] - pública e privada) em cada um de seus HSMs. Este par tem como propósito prover portabilidade entre HSMs de quaisquer PSCs. Este par de chaves deve ser utilizado em possível exportação de chaves privadas de usuário, Cp_rU_i e também na assinatura das requisições para envelopamento utilizando a sua chave pública. Por analogia, para a chave CP_uH_e, 'C' significa 'Chave', P_u chave Pública, e H_e significa chave gerada pelo HSM para exportação de chave do usuário 'i', CP_rU_i. De forma similar, CP_rH_e e CP_rU_i têm significados equivalentes;

b. CP_uH_e é armazenada em repositório do ITI, e CP_rH_e é mantida no HSM de origem;

(ii) para exportação de chaves privadas dos usuários contidas no HSM 1 para o HSM 2:

a. No PSC importa-se para o HSM 1 a chave pública do HSM 2, $CP_uH_e^2$, do repositório do ITI;

b. No HSM 1 gera-se uma chave de sessão simétrica, CS_i, distinta, para cada chave privada de usuário a ser exportada;

c. No HSM 1 cifra-se a chave simétrica, CS_i, com a chave pública do HSM 2, $CP_uH_e^2$, de destino, para exportação da chave do usuário 'i', CP_rU_i;

d. No HSM 1 cifra-se a chave privada do usuário 'i', CP_rU_i, antes do procedimento de exportação de chaves, com a chave simétrica gerada, CS_i, com o algoritmo de sigilo padrão AES ou Serpent, com o modo de operação CTR e tamanho de chave de 256 bits;

e. No HSM 1 apaga-se cada chave de sessão simétrica gerada, CS_i, após o procedimento de cifração do item 'f' ter sido executado;

f. Após a cifração da chave privada do usuário 'i', CP_rU_i, ter sido realizada com sucesso, exporta-se essa chave, e a chave CS_i cifrada, para o HSM 2;

(iii) para importação de chaves privadas dos usuários contidas no HSM 1 para o HSM 2:

a. O administrador do HSM 2, de destino, cria um novo usuário e o habilita;

b. O usuário importa do HSM 1 sua chave privada e a chave simétrica cifrada, itens 'e' e 'f';

c. No HSM 2, de destino, recebe-se a chave privada CP_rU_i e a chave simétrica CS_i cifradas, do usuário 'i';

d. No HSM 2 decifra-se a chave simétrica, CS_i, com a chave privada do HSM2, $CP_rH_e^2$;

e. Em seguida, no HSM 2 decifra-se a chave privada do usuário 'i', CP_rU_i, que estava no HSM 1, com a chave simétrica CS_i, com o algoritmo criptográfico padrão AES ou Serpent, com o modo de operação CTR e tamanho de chave de 256 bits;

f. No HSM 2 grava-se a chave privada do usuário 'i', CP_rU_i, já decifrada, e importada do HSM 1;

g. No HSM 2 destrói-se a chave simétrica CS_i;

h. O PSC 2 encaminha para o PSC 1 mensagem indicando que a importação ocorreu

satisfatoriamente. Então, o HSM 1 apaga a chave privada do usuário 'i', CP_rU_i.

6.3 Rede

6.3.1 Pode ser arquitetado um *pool* de HSM para operação, replicação e gerenciamento das chaves dos usuários finais, devendo seguir, além dos relatados nesse documento, os seguintes requisitos:

(a) Especificação e estabelecimento deu uma comunicação segura (sessão SSL/TLS) ou equivalente entre os HSM;
(b) Os HSM poderão estar em ambientes distintos desde que os mecanismos de acesso e segurança se mantenham os descritos neste documento.

6.3.2 O PSC SyngularID atende ao critério mínimo de 99,99% de “nível de tempo de atividade” (*uptime*) a ser verificado mensalmente.

6.4 Requisitos para serviços de confiança de uso de chaves privadas

6.4.1 Definições para Interface de Serviços de Confiança

6.4.1.1. Utiliza-se o protocolo TLS, definido pela RFC 5246, para comunicação com serviços de confiança.

6.4.1.2. Utiliza-se o *framework* OAuth 2.0 (RFC 6749 e RFC 7636) para implementação da interface aos serviços de confiança do PSC SyngularID.

6.4.1.3. Não se aplica.

6.4.2 Definições para URI de base para Serviços de Confiança

6.4.2.1. A URI de base - URI-base - define o estilo e formato dos endereços HTTPS de serviços de confiança.

6.4.2.2. A URI de base contém o número correspondendo à versão de API definida pela ICP-Brasil.

6.4.2.3. Este documento trata da versão “v0” de API para PSC.

6.4.2.4. A URI-base do PSC SyngularID é: https://psc.syngularid.com.br/v0/

6.4.2.5. As demais porções de URI presentes neste documento devem ser concatenadas à URI-base.

6.4.3 Autorização e Autenticação para Requisição de Serviços

6.4.3.1. Fluxo básico para Uso de Serviços de Confiança

6.4.3.1.1. Seguindo o fluxo de autorização estabelecido pela RFC 6749, o uso de chaves privadas em PSC é precedido de solicitação bem-sucedida, por parte de aplicações, dos seguintes serviços:

(a) Código de Autorização;
(b) *Token* de Acesso;
(c) Assinatura.

6.4.3.1.2. Quando for necessário utilizar serviço de confiança destinado somente à autenticação do titular, ou seja, sem o uso de chave privada, deverá ser precedido de solicitação bem-sucedida, por parte de aplicações, dos seguintes serviços:

(a) Código de Autorização;
(b) *Token* de Acesso;
(c) Recuperação de Certificado.

6.4.3.2. Trânsito de Fatores de Autenticação

6.4.3.2.1. As aplicações não coletam fatores de autenticação do usuário. Para este fim, o PSC SyngularID se comunica diretamente com equipamento do usuário, previamente identificado e cadastrado junto ao PSC de forma segura.

6.4.3.2.2. Excetua-se desta regra o Serviço “Autorização com Credenciais do Titular”.

6.4.3.3. Autenticação de Aplicações de Assinatura

6.4.3.3.1. Não se aplica.

6.4.3.3.2. Não se aplica.

6.4.3.3.3. Não se aplica.

6.4.4. Relação de Serviços de Confiança Disponibilizados por PSC

(a) Serviços de Confiança Obrigatórios
    (i) Código de Autorização
    (ii) *Token* de Acesso
    (iii) Assinatura
    (iv) Cadastro de Aplicação com Certificado
    (v) Listagem de Certificados do Titular
    (vi) Localização de Titular
    (vii) Recuperação de Certificado

(b) Serviços de Confiança Opcionais
  (i) Cadastro de Aplicação sem Certificado
  (ii) *Token* de Acesso para Aplicação
  (iii) Manutenção de Aplicação
  (iv) Autorização com Credenciais do Titular

6.4.5. Detalhamento de Serviços de Confiança Obrigatórios

6.4.5.1. Serviços de Autorização

6.4.5.1.1. Código de Autorização (*Authorization Code Request*)

(a) Serviço para obter do titular a autorização de uso da sua chave privada ou autorizar autenticação sem uso da chave privada.
(b) Caso o titular possua mais de um certificado, o PSC deverá apresentá-los para que o titular faça a escolha no mesmo contexto de aplicação em que transitarem os fatores de autenticação.
(c) Caberá ao PSC apresentar ao titular o escopo da solicitação (vide parâmetro "*scope*" abaixo), permitindo a diferenciação inequívoca de solicitações que envolvam assinaturas daquelas que tratam somente de autenticação. Esta apresentação deverá ser feita durante o trânsito de fatores de autenticação.

(i) Solicitação:

- *Path: <URI-base>/oauth/authorize*;
- Método HTTPS: *GET*;
- Parâmetros da requisição: concatenados após o *Path* como parâmetros http query, usando o formato "*application/x-www-form-urlencoded*":
  - *response_type:* obrigatório, valor "*code*";
  - *client_id:* obrigatório, deve conter a identificação da aplicação;
  - *redirect_uri:* opcional, deve ter a URI para redirecionar o usuário de volta para a aplicação de origem. A URI deve estar na lista de URI's autorizadas para a aplicação. Deve ser URL ENCODED. Se não informado, será considerada a primeira URI cadastrada para a aplicação;
  - *state:* opcional, é retornado sem modificações para aplicação de origem;
    - Recomendado. Um valor opaco usado pela aplicação para manter o estado entre a requisição e a resposta. O serviço de autorização incluirá este valor ao redirecionar o módulo do usuário de volta ao endereço da aplicação. Este parâmetro deverá ser usado para prevenir ataques de falsificação de requisições entre sites (*cross-site*

request forgery).

- *lifetime:* opcional, indica o tempo de vida desejado para o token a ser gerado. Inteiro, em segundos;
- *scope:* opcional, se não informado, será considerado "*authentication_session*". (ver lista de escopos abaixo). Possíveis valores para o parâmetro:
  - ***single_signature:*** token que permite a assinatura de apenas um conteúdo (*hash*), sendo invalidado após a sua utilização;
  - ***multi_signature:*** token que permite a assinatura de múltiplos *hashes* em uma única requisição, sendo invalidado após a sua utilização;
  - ***signature_session:*** token de sessão OAuth que permite várias assinaturas em várias chamadas à API, desde que o token esteja dentro do prazo de validade ou que não tenha sido revogado pela aplicação ou pelo usuário;
  - ***authentication_session:*** *token* de sessão OAuth para autenticação do titular, não permitindo a realização de assinaturas ou outras utilizações da chave privada.
- *code_challenge:* obrigatório, ver RFC 7636
- *code_challenge_method:* obrigatório, valor “S256” (ver RFC 7636).
- *login_hint:* opcional, valor de CPF ou CNPJ a ser informado como filtro para seleção do certificado a ser utilizado.

(ii) Resposta da Requisição de Código de Autorização:

(a) Se o usuário autorizar a solicitação, o PSC emite um código de autorização com tempo de validade curto e retorna para aplicação cliente com uma URI de redirecionamento contendo os seguintes parâmetros no componente http *query*, usando o formato "*application/x-www-form-urlencoded*":
- *code:* obrigatório, código de autorização gerado pelo PSC, a ser usado na solicitação do *token* de acesso;
- *state:* obrigatório caso tenha sido informado na requisição, deverá conter o que foi enviado na requisição.

(b) Se o usuário não autorizar a solicitação, o PSC retorna para aplicação cliente através de sua *redirect_uri* os seguintes parâmetros via http *query*, usando o formato "*application/x-www-formurlencoded*":
- *error:* obrigatório, com o valor “*user_denied*”;
- *state:* obrigatório caso tenha sido informado na requisição, deverá conter o que foi enviado na requisição.

6.4.5.1.2. *Token* de Acesso

Após a obtenção de código de autorização, o *token* de acesso é solicitado com parâmetros no formato “*application/x-www-form-urlencoded*”.

(a) Solicitação

- *Path*: *<URI-base>/oauth/token*;
- Método HTTPS: POST;
- Parâmetros da requisição: formato "*application/x-www-form-urlencoded*"
  - *grant_type*: obrigatório, valor "*authorization_code*";
  - *client_id*: obrigatório, deve conter a identificação da aplicação;
  - *client_secret*: obrigatório, deve conter o segredo associado à aplicação;
  - *code*: obrigatório, deve conter código de autorização retornado do Serviço Código de Autorização;
  - *redirect_uri*: opcional, deve ser igual ao informado no Serviço Código de Autorização;
  - *code_verifier*: obrigatório, correspondendo a *code_challenge* enviado na Requisição de Código de Autorização, ver RFC 7636.
    Exemplo:

POST {.../oauth/token} HTTP/1.1
Host: {servidor do PSC}
Content-Type: application/x-www-form-urlencoded
grant_type=authorization_code
&client_id=MyApplicationId
&client_secret=123qwe
&code=09b30f74d40a7fece1a26cccc97746c364e61022
&redirect_uri=https://idg.receita.fazenda.gov.br
&code_verifier={Verifier}

(b) Resposta da Requisição de *Token* de Acesso:

(i) Se a requisição é válida e autorizada o PSC emite um *token* de acesso e retorna a requisição com sucesso, via HTTP *Status Code* 200.

- Parâmetros de retorno: formato "*application/json;charset*=UTF-8":
  - *access_token*: obrigatório, valor do *token* de acesso;
  - *token_type*: obrigatório, valor "*Bearer*";
  - *expires_in*: obrigatório, valor inteiro com validade do token em segundos. Para acesso a objeto de pessoas físicas não deve ultrapassar (7 dias), sendo que para pessoas jurídicas este limite será de (30 dias);
  - *scope*: opcional, deve ser informado se o escopo retornado for diferente do solicitado pela aplicação;
  - *authorized_identification_type*: obrigatório, deverá conter “CPF” ou “CNPJ”;
  - *authorized_identification*: obrigatório, valor correspondendo ao CPF ou CNPJ associado ao titular do certificado.

Exemplo:

HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8

Cache-Control: no-store
Pragma: no-cache
{
"access_token": "b923575f1ced0ee732ee274b2e02784040bd9606",
"expires_in": 300,
"token_type": "Bearer
"authorized_identification_type": "CPF
"authorized_identification": 00000000001
}

NOTA: Não é permitido o *refresh_token*.

(ii) Se a requisição não for válida, houver falha na autenticação da aplicação cliente ou alguma outra falha, o PSC retorna a requisição com erro, via HTTP *Status Code* de erro correspondente à situação ocorrida via JSON com os seguintes parâmetros:

- Parâmetros de retorno: formato "*application/json;charset*=UTF-8":
  - *error*: obrigatório, representa o código do erro. Possíveis valores para o parâmetro e HTTP *Status Code* de erro:
  - *invalid_request*: HTTP *Status Code* 400, ocorre quando algum parâmetro obrigatório não tiver sido informado ou inclui um valor de parâmetro não suportado ou algum parâmetro com valor duplicado informado ou a requisição é mal-formada;
  - *invalid_grant*: HTTP *Status Code* 400, ocorre quando o código de autorização apresentado estiver inválido ou expirado ou tiver sido emitido para uma outra aplicação cliente diferente da informada ou já estiver sido utilizado em um cenário de uso único (*scope, single_signature* e *multi_signature*). Ocorre também na validação da *redirect_uri* e na validação do *code_verifier* (ver RFC 7636);
  - *invalid_client*: HTTP *Status Code* 401, ocorre quando houver falha na autenticação da aplicação cliente, desde aplicação não identificada até credenciais inválidas;
  - *unsupported_grant_type*: HTTP *Status Code* 400, ocorre quando o valor informado no parâmetro *grant_type* não for suportado;
  - *server_error*: HTTP *Status Code* 500, ocorre quando houver algum erro interno não tratado pelo PSC.

- *error_description*: opcional, texto com informações adicionais descrevendo o erro a fim de assistir o entendimento do ocorrido;
- *error_uri*: opcional, URI de uma página WEB que contém informações sobre o erro ocorrido.
  Exemplo:
  HTTP/1.1 400 Bad Request
  Content-Type: application/json;charset=UTF-8
  Cache-Control: no-store
  Pragma: no-cache {

“error”: “invalid_request”,
“error_description”: “Parâmetro obrigatório não informado: code”,
“error_uri”: “https://psc.exemplo.com.br/docs/oauth2-error#invalid_request”
}

6.4.5.2. Assinatura

6.4.5.2.1. Os parâmetros com conteúdo a ser assinado e assinaturas deverão conter valores em Base64.

6.4.5.2.2. As assinaturas RAW estarão em Base64.

6.4.5.2.3. Assinaturas CMS estão em formato CMS PEM de acordo com RFC 7468: o cabeçalho e rodapé CMS são obrigatórios; quebra de linha e espaços no conteúdo são opcionais; e as aplicações devem estar preparadas para lidar com diferentes formas de espaços e quebra de linhas no conteúdo, ou com a ausência destes.

6.4.5.2.4. Se o escopo do *token* permitir apenas uma assinatura (*single_signature*) e for informado mais de um conteúdo, uma mensagem de erro é retornada.

6.4.5.2.5. Se o escopo for omitido ou assinalado para autenticação (*authentication_session*) uma mensagem de erro é retornada.

(a) Solicitação:

- *Path: <URI-base>/oauth/signature*
- Método HTTPS: POST
- Cabeçalho:
  - *Content-type: application/json;*
  - *Accept: application/json;*
  - *Authorization: Bearer access_token*;
- Parâmetros: formato "*application/json;charset=UTF-8*" :
  - *certificate_alias*: opcional, identificador do certificado correspondente à chave utilizada na assinatura;
  - *hashes*: conjunto com valores obrigatórios a serem assinados. Cada elemento do conjunto conterá:
    - *id*: identificador do conteúdo a ser assinado;
    - *alias*: forma legível do identificador do conteúdo;
    - *hash*: conteúdo a ser assinado;
    - *hash_algorithm*: *Object Identifier* (OID) do algoritmo de *hash*. Por exemplo, para SHA256 utilize o OID 2.16.840.1.101.3.4.2.1;
    - *signature_format*: deverá conter um dos valores:

- “RAW”,
- “CMS”

Exemplo:

```
"hashes": [{
        "id": "Signature request ID 1",
        "alias": "Contrato de aluguel XPTO",
        "hash": "hash to sign",
        "hash_algorithm": "2.16.840.1.101.3.4.2.1",
        “signature_format”: “RAW”
},
{
        "id": "Signature request ID 2",
        "alias": "Documento do Word",
        "hash": "hash to sign",
        "hash_algorithm": "2.16.840.1.101.3.4.2.1",
        “signature_format”: “CMS”
}
{
        "id": "Signature request ID n",
        "alias": "Firefox",
        "hash": "hash to sign",
        "hash_algorithm": "2.16.840.1.101.3.4.2.1",
        “signature_format”: “RAW”
}
]
}
```

(b) Resposta da Requisição de Assinatura:

O PSC retornará a requisição com sucesso, via HTTP *Status Code* 200.

- Parâmetros: formato "*application/json;charset=UTF-8*":
  - *certificate_alias*: obrigatório, identificador do certificado correspondente à chave utilizada na assinatura;
  - *signatures*: obrigatório, conjunto com identificadores dos conteúdos assinados e valores assinados. Cada elemento do conjunto conterá:
    - *id*: identificador do conteúdo assinado;
    - Um dos formatos abaixo:
      - caso a solicitação tenha sido feita com “*signature_format*: RAW”
        - *raw_signature*: valor numérico em base64 da assinatura produzida.
      - caso a solicitação tenha sido feita com “*signature_format*: CMS”
        - CMS *detached* (PKCS#7), contendo os seguintes atributos assinados:
          - *contentType*

- *signingTime* (hora do PSC)
- *messageDigest* (*hash* informado pela aplicação na requisição)
- *signingCertificateV2* (certificado do assinante)

NOTA: Os valores de assinatura serão produzidos de acordo com a suíte de assinatura, se esta for informada.

Exemplo:

```
{
    "certificate_alias": "CERTIFICADO TESTE 1:1234567889",
    "signatures": [{
        "id": "Signature request ID 1",
        "raw_signature": "my raw signature base64"
    },
    {
        "id": "Signature request ID 2",
        "raw_signature": "my raw signature base64"
    },
    {
        "id": "Signature request ID n",
        "raw_signature": "my raw signature base64"
    }]
}
```

6.4.5.3. Cadastro de Aplicação com Certificado

6.4.5.3.1. Serviço para cadastro de uma aplicação junto ao PSC, sendo que a aplicação utilizará um certificado SSL ICP-Brasil para assinar os dados enviados, substituindo neste caso o Serviço de Cadastro de Aplicação.

6.4.5.3.2. A assinatura dos dados necessários para o cadastro será realizada utilizando o formato *JWT with RSA Signature*, conhecido como JWS – *Json Web Signature* (ver RFC 7515), utilizando o algoritmo de hash SHA-256.

O *header* do JWS deverá conter os seguintes parâmetros:

- *alg*: obrigatório, valor “RS256” representando RSA *With* SHA-256;
- x5c: obrigatório, valor multivalorado contendo o certificado SSL ICP-Brasil no formato PEM.

Exemplo do *Header* do JWS desserializado:

```
{
“alg”: “RS256”,
```

"x5c": [“-----BEGIN CERTIFICATE-----ADFAASDFASDFAS. . . -----END CERTIFICATE-----”]

}

O conjunto de dados JWS deverá conter os seguintes parâmetros:

- *name*: obrigatório, nome da aplicação;
- *comments*: obrigatório, descrição da aplicação;
- *redirect_uris*: obrigatório, valor multivalorado contendo URI’s autorizadas para redirecionamento (para serviços de requisição de autorização). Devem ser oriundas do *host* do certificado de equipamento apresentado, sendo vedada a utilização de *fragments*;
- *host*: obrigatório, valor contendo o *host* único da aplicação;
- *aud*: obrigatório, valor contendo o nome único do PSC a qual a assinatura é direcionada.
- *email*: obrigatório, e-mail para suporte em caso de indisponibilidade, mudança de versão, entre outros.

Exemplo do *Payload* do JWS desserializado:

```
{
    "name": "Nome da Aplicação",
    "comments": "Descrição da Aplicação",
    "host": "www.aplicacao-exemplo.com",
    "redirect_uris”: [
    https://www.aplicacao-exemplo.com/callback/certificado_nuvem
    ],
    "aud": "nome-unico-psc"
    "email": "psc@psc.com.br"
}
```

(a) Solicitação:

- *Path: <URI-base>/oauth/application_cert*
- Método HTTPS: POST
- Cabeçalho:
  - *Accept: application/octet-stream*;
  - *Body*: *string* contendo o JWS serializado.

(b) Resposta do Serviço de Cadastro de Aplicação com Certificado

- Parâmetros: formato "*application/json;charset*=UTF-8" :
  - *client_id*: obrigatório, identificador único da aplicação gerado pelo PSC;
  - *client_secret*: obrigatório, credencial da aplicação gerada de forma aleatória pelo PSC;

6.4.5.4. Recuperação de Certificado

Serviço para recuperar certificado armazenado no PSC. A aplicação deverá ter um *Access Token* válido.

(a) Solicitação:

- *Path: <URI-base>/oauth/certificate-discovery*;
- Método HTTPS: GET
- Cabeçalho
  - *Content-type: application/json*;
  - *Accept*: application/json;
  - *Authorization*: *Bearer access_token*;
- Parâmetros da requisição: concatenados após o *Path* como parâmetros *http query*, utilizando o formato "*application/x-form/urlencoded*"
  - *certificate_alias*: opcional, é o identificador do certificado a ser recuperado.

(b) Resposta:

- Parâmetros
  - *status*: obrigatório, indicando "S" para resultado positivo ou "N" caso contrário;
  - *certificates*: certificado em BASE64 recuperado;

  Exemplo

  {
  "status": "S"
  "certificates": [
  {
  "alias": "CERTIFICADO TESTE 1:123456789
  "certificate": "-----BEGIN CERTIFICATE-----\n{CERTIFICADO}\n-----END CERTIFICATE-----",
  }
  {
  "alias": "CERTIFICADO TESTE 2:123456789
  "certificate": "-----BEGIN CERTIFICATE-----\n{CERTIFICADO}\n-----END CERTIFICATE-----",
  }
  ]
  }

6.4.5.5. Localização do Titular

Serviço para encontrar um titular mediante informação de CPF ou CNPJ.

(a) Solicitação:

- *Path: <URI-base>/oauth/user-discovery*;
- Método HTTPS: POST;
- Parâmetros da requisição: formato "*application/json;charset=UTF-8*":
  - *client_id:* obrigatório, deve conter a identificação da aplicação;
  - *client_secret:* obrigatório, deve conter o segredo associado à aplicação;

- *user_cpf_cnpj:* obrigatório, deve conter “CPF” para pessoa física ou “CNPJ” pessoa jurídica;
- *val_cpf_cnpj:* obrigatório, deve conter o valor do cpf ou cnpj;

(b) Resposta:

- Parâmetros:
  - *slots*: opcional, matriz com os *aliases* de *slots* encontrados, composto pelos pares ordenados *slot_alias* e *label*;
  - *status*: obrigatório, indicando “S” para resultado positivo ou “N” caso contrário;

Exemplo:

```
{
    "slots": [{
    "slot_alias": "12345678899-1",
     "label": "A3 PESSOAL"
 }
 {
     "slot_alias": "12345678899-2",
     "label": "A3 TRABALHO"
 }
 ],
    "status": "S”
}
```

6.4.6. Detalhamento de Serviços de Confiança Opcionais

6.4.6.1. Cadastro de Aplicação sem Certificado

Serviço para cadastro de uma aplicação junto ao PSC. É obrigatório para todas as aplicações que utilizarem serviços de autorização sem certificados ICP-Brasil.

(a) Solicitação:

Path: <URI-base>/oauth/application

- Método HTTPS: POST
- Cabeçalho:
  - *Content-type*: *application/json*;
  - Accept: *application/json*;
- Parâmetros: formato "*application/json;charset=UTF-8*" :
  - *name*: obrigatório, nome/descrição da aplicação;
  - *comments*: obrigatório, observações gerais de uso da aplicação;
  - *redirect_uris*: obrigatório, URI’s autorizadas para redirecionamento (para serviços de código de autorização).
  - *email*: obrigatório, e-mail para suporte em caso de indisponibilidade, mudança de versão, entre outros.

Exemplo:
{
"name": "(Nome/Descricao da aplicacao)",
"comments": "(Observacoes gerais de uso da aplicacao)",
"redirect_uris": [
"URI 1 pre cadastrada para redirecionamento",
"URI 2 pre cadastrada para redirecionamento",
"URI N pre cadastrada para redirecionamento"
]
"email": psc@psc.com.br
}

(b) Resposta da Requisição de Cadastro de Aplicação:

- Parâmetros: formato "*application/json;charset=UTF-8*" :
  - *client_id*: identificador da aplicação;
  - *client_secret*: segredo associado à aplicação;
  - *status*: obrigatório, "*success*" para sucesso;
  - *message*: obrigatório, mensagem com informações adicionais.

Exemplo:
{
"client_id": "(identificador da aplicacao)",
"client_secret": "(segredo da aplicacao)",
"status": "success",
"message": "Aplicacao cadastrada com sucesso"
}

6.4.6.2. Serviços de Manutenção de Cadastro de Aplicação

Serviço para manutenção das informações armazenadas de uma aplicação no PSC. É obrigatório para todas as aplicações que utilizarem serviços de autorização não identificadas por certificados ICP-Brasil para SSL.

6.4.6.2.1. *Token* de Acesso para Aplicação

Requisição para que uma aplicação obtenha *token* de acesso para manutenção de seu cadastro junto ao PSC.

(a) Solicitação:

- Método HTTPS: POST;
- *Path: <URI-base>/oauth/client_token*;

- Parâmetros da requisição: formato "*application/x-www-form-urlencoded*":
  - *grant_type*, obrigatório, valor "*client_credentials*";
  - *client_id,* obrigatório, deve conter a identificação da aplicação;
  - *client_secret*, obrigatório para aplicações que possuem certificado digital;

  Exemplo:

  POST {.../oauth/client_token} HTTP/1.1
  Host: {servidor do PSC}
  Content-Type: application/x-www-form-urlencoded
  client_id=Identificacao_aplicacao
  &client_secret=123qwe
  &grant_type=client_credentials

(b) Resposta da Requisição de *Token* de Acesso para Aplicações:

- Parâmetros de retorno: formato "*application/json;charset=UTF-8*" :
  - *access_token*, obrigatório, valor do *token* de acesso;
  - *token_type*, obrigatório, valor "*Bearer*";
  - *expires_in*, opcional, validade do *token* em segundos.

  Exemplo:

  {
  "access_token": "b923575f1ced0ee732ee274b2e02784040bd9606",
  "expires_in": 7200,
  "token_type": "Bearer"
  }

6.4.6.2.2. Manutenção de Aplicação

Serviço para atualização de informações de uma aplicação. Requer um *token* de acesso para aplicações, enviado no parâmetro de cabeçalho "*Authorization*".

(a) Solicitação:

- *Path: <URI-base>/oauth/client_maintenance*;
- Método HTTPS: PUT;
- Cabeçalho:
  - *Content-type: application/json;*
  - *Accept: application/json*;
  - *Authorization: Bearer access_token* ("*Bearer*" concatenado com espaço e access_token);
- Parâmetros: formato "*application/json;charset=UTF-8*" :
  - *client_id*, obrigatório, deve conter a identificação da aplicação;
  - *client_secret*, opcional, nova senha da aplicação;
  - *name*, opcional, nome da aplicação;

- *comments*, opcional, descrição da aplicação;
- *redirect_uris*, opcional, URI's autorizadas para redirecionamento (para requisição de código de autorização).
- *email*: obrigatório, *e-mail* para suporte em caso de indisponibilidade, mudança de versão, entre outros.

Exemplo:
{
"client_id": "identificador da aplicação",
"client_secret": "(Senha/Segredo da aplicação)",
"name": "(Nome da aplicação)",
"comments": "(Descrição da aplicação)",
"redirect_uris": [
"URI 1 pre cadastrada para redirecionamento",
"URI 2 pre cadastrada para redirecionamento",
"URI N pre cadastrada para redirecionamento"
]
"email": "psc@psc.com.br"
}

(b) Resposta da Requisição de Manutenção de Aplicações:

- Parâmetros de retorno: formato "*application/json;charset=UTF-8*":
  - *client_id*: obrigatório, deve conter a identificação da aplicação;

Exemplo:
{
"client_id": "(identificador da aplicação)",
}

6.4.6.3. Autorização com Credenciais do Titular

6.4.6.3.1. Serviço para obter do titular autorização de uso da chave privada, com solicitação de fatores de autenticação.

6.4.6.3.2. No mínimo um fator de autenticação obtido deve ser válido para uma única solicitação de autorização (*OTP - one-time password*).

6.4.6.3.3. Os fatores de autenticação deverão ter seus valores concatenados e enviados no parâmetro "*password*".

(a) Solicitação:

- *Path: <URI-base>/oauth/pwd_authorize*;
- Método HTTPS: POST;

- Cabeçalho:
  - *Content-type: application/json;*
  - *Accept: application/json;*
- Parâmetros: formato "*application/json;charset=UTF-8*" :
  - *grant_type*, obrigatório, valor "*password*”;
  - *client_id*, obrigatório, identificação da aplicação;
  - *client_secret*, opcional, sendo obrigatório apenas quando a aplicação não utilizar certificado ICP-Brasil;
  - *username*, obrigatório, identificação do usuário por meio de CPF ou CNPJ;
  - *password*, obrigatório, valor da concatenação de fatores de autenticação informadas pelo usuário;
  - *lifetime*, opcional, valor inteiro, indica o tempo de vida desejado para o *token* a ser gerado em segundos. Para acesso a objeto de pessoas físicas não deve ultrapassar 7 (sete) dias, sendo que para pessoas jurídicas este limite será de 30 (trinta) dias;
  - *scope*, opcional, se não informado será considerado "*authentication_session*". (ver lista de escopos para Serviço de Código de Autorização).
  - *slot_alias*: opcional, indica o *slot* do usuário no qual a autenticação deve ser feita. Se não informado, o PSC decidirá em qual *slot* tentar a autenticação.

  Exemplo:

```
{
 “client_id": "MyApplicationId",
 "client_secret": "123qwe",
 "username": "0660457192",
 "password": "123456SENHA",
 "grant_type": "password",
 "scope": "single_signature",
 "lifetime": 900,
 "slot_alias": "12345678899"
}
```

(b) Resposta da Requisição de Autorização com Credenciais do Titular:

- Parâmetros de retorno para os demais valores de “*scope*”: formato "*application/json;charset=UTF-8*":
  - *access_token*, obrigatório, valor do *token* de acesso;
  - *token_type,* obrigatório, valor "*Bearer*";
  - *expires_in*, obrigatório, valor inteiro com validade do token em segundos. Para acesso a objeto de pessoas físicas, não deve ultrapassar 7 (sete) dias, sendo que para pessoas jurídicas, esse limite será de 30 (trinta) dias;
  - *scope*, opcional, informado apenas se o escopo retornado for diferente do solicitado pela aplicação.
  - *slot_alias*: obrigatório, indica o *slot* do usuário no qual a autenticação foi feita (sem *middleware*).

  Exemplo:

{
"access_token":
"b923575f1ced0ee732ee274b2e02784040bd9606",
"expires_in": 300,
"token_type": "Bearer",
"slot_alias": "12345678899"
}

6.5. Lista de Prestador de Serviço de Confiança – LPSC

6.5.1. A Lista de Prestadores de Serviço de Confiança – LPSC contém as entidades credenciadas no âmbito da ICP-Brasil como Prestadores de Serviço de Confiança - PSC. A LPSC será publicada pela AC Raiz e atualizada no prazo máximo de 180 dias.

6.5.2. A LPSC será publicada no repositório da AC Raiz em versão textual, para leitura humana, e em XML, para processamento por máquina.

6.5.3. A autenticidade e a integridade da versão processável por máquina da lista compilada são asseguradas por meio de uma assinatura digital XMLDSig suportada por um certificado digital do ITI.

6.5.4. As versões da LPSC e o certificado que assina a LPSC serão publicados no repositório da AC Raiz, disponível em: **http://www.iti.gov.br/repositorio.**

6.5.5. A autenticidade e integridade da lista compilada devem ser verificadas pelas partes confiáveis.

6.5.6. A LPSC é codificada em XML, em conformidade com a estrutura proposta pelo padrão ETSI TS 102 231, e contém os seguintes dados:

(a) a informação do esquema (*SchemeInformation*), onde são apresentados os dados de identificação do emissor, o ITI, e a data da próxima atualização (*NextUpdate*) da lista;

(b) a lista de prestadores de serviço (*TrustServiceProviderList*), que contém uma entrada (*TrustServiceProvider*) para cada PSC credenciado junto à ICP-Brasil; e

(c) assinatura digital no formato XMLdSIG.

6.5.7. A LPSC conterá na URI de base que define o serviço (*SchemeServiceDefinitionURI*) a versão da API correspondente, podendo apresentar mais de uma instância de versão para minimizar comprometimento das aplicações integradas.

7. SERVIÇO DE ASSINATURA DIGITAL, VERIFICAÇÃO DE ASSINATURA DIGITAL

7.1. Introdução

7.1.1. Não se aplica.

7.2. Criação de assinaturas

7.2.1. Não se aplica.

7.2.2. Não se aplica.

7.2.3. Não se aplica.

7.3. Dispositivos para criação de assinaturas

7.3.1. Não se aplica.

7.3.2. Não se aplica.

7.3.3. Não se aplica.

7.4. Interface da aplicação com o dispositivo de criação de assinaturas

7.4.1. Não se aplica.

7.4.2. Não se aplica.

7.4.3. Não se aplica.

7.4.4. Não se aplica.

7.5. Suítes de Assinatura

7.5.1. Não se aplica.

7.6. Formatos de Assinaturas

7.6.1. Não se aplica.

7.6.2. Não se aplica.

7.6.3. Não se aplica.

7.7. **Assinatura com Carimbo do Tempo**

7.7.1. Não se aplica.

7.7.2. Não se aplica.

7.7.3. Não se aplica.

7.8. **Validação de Assinaturas**

7.8.1. Não se aplica.

7.8.2. Não se aplica.

7.8.3. Não se aplica.

7.8.4. Não se aplica.

7.9. **Acordo de Nível de Serviço**

7.9.1. Não se aplica.

8. CLASSIFICAÇÃO DA INFORMAÇÃO

8.1. Toda informação gerada e custodiada pelo PSC deverá ser classificada segundo o seu teor crítico e grau de confidencialidade, de acordo com sua própria Política de Classificação de Informação.

8.2. A classificação da informação no PSC deverá ser realizada independente da mídia onde se encontra armazenada ou o meio pelo qual é trafegada.

8.3. A informação poderá ser classificada em:

8.3.1. Público: Qualquer ativo de informação, de propriedade do PSC ou não, que poderá vir ao público sem maiores consequências danosas ao funcionamento normal do PSC. Poderá ser acessado por qualquer pessoa, seja interna ou externa ao PSC. Integridade da informação não é vital.

8.3.2. Pessoal: Qualquer ativo de informação relacionado à informação pessoal. Por exemplo: mensagem pessoal de correio eletrônico, arquivo pessoal, dados pessoais, entre outros.

8.3.3. Interna: Qualquer ativo de informação, de propriedade do PSC ou não, que não seja considerada pública. Ativo de informação relacionado às atividades do PSC que é direcionada estritamente para uso interno. A divulgação não autorizada do ativo de informação poderia causar impacto à imagem do PSC. Por exemplo: código fonte de programa, cronograma de atividades, atas de reuniões, entre outros.

8.3.4. Confidencial: Qualquer ativo de informação que seja crítico para as atividades do PSC em

relação ao sigilo e integridade. Qualquer material e informação recebida para ensaio, assim como qualquer resultado do ensaio (como relatório) deverá ser considerado confidencial.

NOTA: Caso o PSC seja entidade da Administração Pública Federal – APF, aplicar-se-á as disposições do Decreto nº 7.845/2012 e demais normas aplicáveis à APF, no que couber.

9. SALVAGUARDA DE ATIVOS DA INFORMAÇÃO

9.1. O PSC define, em sua Política de Segurança da Informação, como é realizada a salvaguarda de ativos de informação no formato eletrônico, também denominado *backup.*

9.2. A salvaguarda de ativos da informação descreve as formas de execução dos seguintes processos:

(i) procedimentos de *backup*;
(ii) indicações de uso dos métodos de *backup*;
(iii) tabela de temporalidade;
(iv) local e restrições de armazenamento e salvaguarda em função da fase de uso;
(v) tipos de mídia;
(vi) controles ambientais do armazenamento;
(vii) controles de segurança;
(viii) teste de restauração de *backup*.

9.3. O PSC possui uma política de recebimento, manipulação, depósito e descarte de materiais de terceiros.

10. GERENCIAMENTO DE RISCOS

O PSC possui um processo de gerenciamento de riscos, atualizado, para prevenção contra riscos, inclusive àqueles advindos de novas tecnologias, visando a elaboração de planos de ação apropriados para proteção aos componentes ameaçados atualizado, no mínimo, anualmente.

11. PLANO DE CONTINUIDADE DE NEGÓCIOS

Um Plano de Continuidade do Negócio – PCN é implementado e testado no PSC, pelo menos uma vez por ano, para garantir a continuidade dos serviços críticos ao negócio em caso de inoperância total ou parcial de seu ambiente.

12. ANÁLISE DE REGISTROS DE EVENTOS

Todos os registros de eventos (*logs*, trilhas de auditorias e imagens) são analisados, no mínimo, mensalmente sendo gerado um relatório com assinatura do responsável pelo PSC. Todos os registros

da transação biométrica por parte do PSC são guardados por um período de 7 (sete) anos.

13. PLANO DE CAPACIDADE OPERACIONAL

13.1. O PSC SyngularID elabora e mantém atualizado anualmente um Planejamento de Capacidade Operacional – PCO para determinar a capacidade de produção atual e futura com níveis de desempenho satisfatórios para responder a novas demandas, fornecendo níveis satisfatórios de serviços aos usuários, visando dimensionar os sistemas para suportar o crescimento orgânico, picos de utilização e sazonalidades.

13.2. O PCO deve, no mínimo:

(a) determinar os níveis de serviços requeridos pelos usuários;
(b) analisar a capacidade de processamento de dados instalada; e
(c) dimensionar a capacidade necessária de infraestrutura, *hardware*, comunicação de dados e link de internet para atender os níveis de serviços atuais e futuros.

14. DOCUMENTOS DA ICP-BRASIL REFERENCIADOS

14.1. Os documentos abaixo são aprovados por Resoluções do Comitê Gestor da ICP-Brasil, podendo ser alterados, quando necessário, pelo mesmo tipo de dispositivo legal. O sítio http://www.iti.gov.br publica a versão mais atualizada desses documentos e as Resoluções que os aprovaram.

Ref	Nome do documento	Código
[1]	CRITÉRIOS E PROCEDIMENTOS PARA CREDENCIAMENTO DAS ENTIDADES INTEGRANTES DA ICP-BRASIL	DOC-ICP-03
[2]	REQUISITOS MÍNIMOS PARA AS POLÍTICAS DE CERTIFICADO NA ICP-BRASIL	DOC-ICP-04
[3]	CRITÉRIOS E PROCEDIMENTOS PARA REALIZAÇÃO DE AUDITORIAS NAS ENTIDADES INTEGRANTES DA ICP-BRASIL	DOC-ICP-08
[4]	CRITÉRIOS E PROCEDIMENTOS PARA FISCALIZAÇÃO DAS ENTIDADES INTEGRANTES DA ICP-BRASIL	DOC-ICP-09
[5]	POLÍTICA DE SEGURANÇA DA ICP-BRASIL	DOC-ICP-02

14.2. Não se aplica.

15. REFERÊNCIAS

BRASIL, Decreto nº 7.845, de 14 de novembro de 2012. Regulamenta procedimentos para credenciamento de segurança e tratamento de informação classificada em qualquer grau de sigilo, e

dispõe sobre o Núcleo de Segurança e Credenciamento.

ETSI TS 102 231 - *Electronic Signatures and Infrastructures (ESI); Provision of harmonized Trustservice status information; V3.1.2 (2009-12).*

RFC 4226, IETF - *HOTP: An HMAC-Based One-Time Password Algorithm, December 2005.*

RFC 5246, IETF – *The Transport Layer Security (TLS) Protocol Version 1.2, August 2008.*

RFC 6238, IETF - *TOTP: Time-Based One-Time Password Algorithm, May 2011.*

RFC 6287, IETF - *OCRA: OATH Challenge-Response Algorithm, June 2011.*

RFC 6749, IETF - *The Oauth 2.0 Authorization Framework, October 2012.*

RFC 7468, IETF - *Textual Encodings of PKIX, PKCS, and CMS Structures, April 2015.*

RFC 7515, IETF - *JSON Web Signature (JWS), May 2015.*

RFC 7636, IETF - *Proof Key for Code Exchange by Oauth Public Clients, September 2015*.