Document d'adhesions a la Plataforma de Seguretat de Salut (PDS)

Historial Electrònic de Salut (HES)

Versió 2.0.0
Abril 2026

Generalitat de Catalunya
Departament de Salut

Sistema de
Salut de Catalunya

Revisió	Redactat per	Revisat per	Aprovat per	Data publicació
1.0.3	Miguel Medina	Joan Esteve Riasol	Joan Esteve Riasol	22/02/2023
1.0.4	Modesto Román, Adrián Toca Raúl Porcar			
1.0.6	Raúl Carretero			
1.0.7	Raúl Carretero	Joan Esteve Riasol	Joan Esteve Riasol	25/03/2024
1.1.0	Joan Esteve Riasol			
1.1.1	Raúl Carretero			
1.1.2	Raúl Porcar	Joan Esteve Riasol	Joan Esteve Riasol	26/11/2024
1.1.3	Joan Esteve Riasol			17/03/2025
2.0.0	Joan Esteve Riasol			20/04/2026

Registre de Canvis

Revisió	Apartat	Data	Motiu del canvi
1.0.3	1, 2, 3, 4	22/02/2023	Versió inicial
1.0.4	5.1.3, 5.1.4, 5.1.5	17/07/2023	Afegim apartat d'alta, llistat i esborrat d'usuaris a través d'API. Canviem codi projecte per 3982.03.
1.0.5	1.2	21/09/2023	S'afegeix el punt 1.2 Com adherir un nou sistema de Salut a la PDS
1.0.6	5.2, 5.3. 5.4	09/01/2024	El punt 5.2 actual el desplacem al 5.4. Afegim en l'apartat de com configurar un client SAML (5.2) Afegim l'apartat d'Integració amb GICAR (5.3)
1.0.7	4.1	25/03/2024	Indiquem en el punt 4.1 que necessitem per poder fer una pantalla de login (o altres pantalles) per als clients amb flux ACF
1.1.0	Tots	01/09/2024	Revisió general del document Explicació del login amb credencials de GICAR o VALID Explicació obtenció tiquets SAML per OSB de Salut Explicació flux de logout
1.1.1	2.1.5 4.4 5.1.2	28/10/2024	Explicació de la nova funcionalitat amb la petició UserInfo a la PDS
1.1.2	1.2	26/11/2024	Canviem correu Oficina d'adhesions
1.1.3	2.1 4.1	17/03/2025	Informació del flux ACF amb PKCE Explicació del flux de logout sense id_token_hint
2.0.0	Tots	22/04/2026	Reedició sencera de tot el Manual

RESPONSABLE DEL DOCUMENT: Àrea TIC del Departament de Salut / CTTI

ÍNDEX

1. Introducció

1.1 Raó i oportunitat del document

Aquest document és la guia per facilitar l'adhesió de noves iniciatives o sistemes d'informació de Salut als serveis que ofereix la **Plataforma de Seguretat de Salut (PdS)** com a proveïdor d'identitats "IDP" (*identity provider*) transversal.

Mitjançant la PDS, les iniciatives poden gestionar la seva lògica de negoci d'**autorització** de les operacions i dades que exposen i publiquen a l'exterior, segons els permisos associats a unes credencials presentades (habitualment un token d'accés) en la petició per un client consumidor, prèviament **autenticat** també en la nostra plataforma.

El nostre IDP es basa en l'estàndard **OpenIdConnect** (OIDC) en base a tokens **JWT**, i també donem suport a **SAML 2.0**, per compatibilitat amb sistemes més aviat *legacy*. Es tracta d'una implementació del producte **Keycloak**, amb una adaptació i configuració a mida per cobrir les necessitats dels sistemes d'informació de Salut adherits.

https://openid.net/connect/
https://saml.xml.org/
https://www.keycloak.org/

La PDS es troba publicada en dos entorns no-productius (**Integració** i **Preproducció**) i un entorn de **Producció**, en les següents adreces. Més endavant en aquest document es dona informació detallada per gestionar la connectivitat HTTPS amb aquests destins:

Entorn	URL Plataforma de Seguretat de Salut (PDS)
INTEGRACIÓ (només amb visibilitat intranet)	https://integracio.pds.hes.catsalut.intranet.gencat.cat
PREPRODUCCIÓ	https://preproduccio.pds.hes.catsalut.gencat.cat
PRODUCCIÓ	https://pds.hes.catsalut.gencat.cat

1.2 Com adherir una nova iniciativa de Salut a la PDS

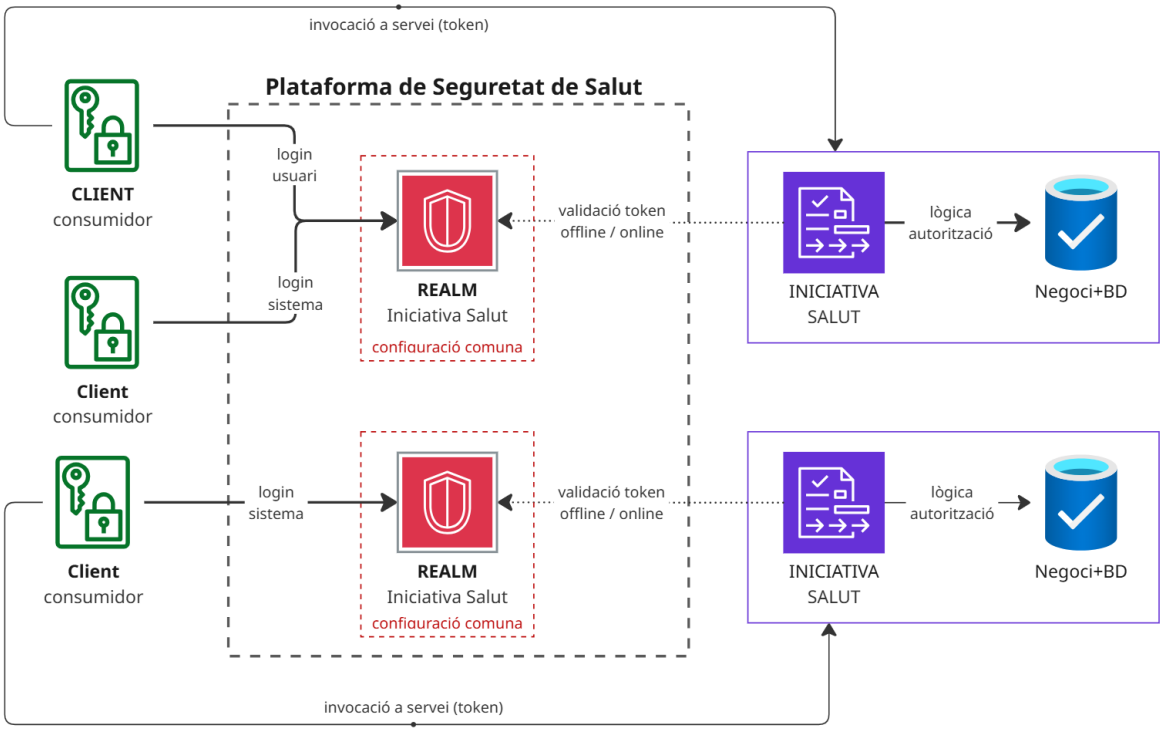
Per incorporar una nova iniciativa o domini funcional de Salut a l'autorització de la PDS, primer cal enviar una sol·licitud d'adhesió a la bústia **oficina.solucionstic@catsalut.cat**.

Seguint aquest manual d'adhesions, l'equip tècnic de la iniciativa ha d'analitzar primer les necessitats de la seva autenticació i autorització de la seva aplicació: petició d'un *Realm* i *Clients* a configurar, flux d'autenticació requerit en cada *Client*, delegació del login d'usuari a un directori corporatiu o sistema de Salut, etc...

En aquest context d'autenticació i autorització, podem descriure un ***Realm*** com una agrupació de ***Clients*** que comparteixen una configuració comuna a l'hora de sol·licitar, refrescar o revocar tokens d'accés, sota una mateixa clau pública i algorisme de signatura.

Habitualment, s'aconsella tenir un *Realm* a la PDS per sistema d'informació o iniciativa de Salut adherida a la PDS, que haurà d'autoritzar els seus serveis que publica i exposa.

I, per altre banda, un *Client* per cada aplicació que hagi de consumir serveis de la iniciativa adherida en aquest *Realm*. D'aquesta manera, cada client que sol·licita tokens d'accés a la PDS disposa de les seves credencials que l'identifiquen unívocament, sense compartir amb altres clients, cosa que facilita els processos d'autorització en el sistema destí.

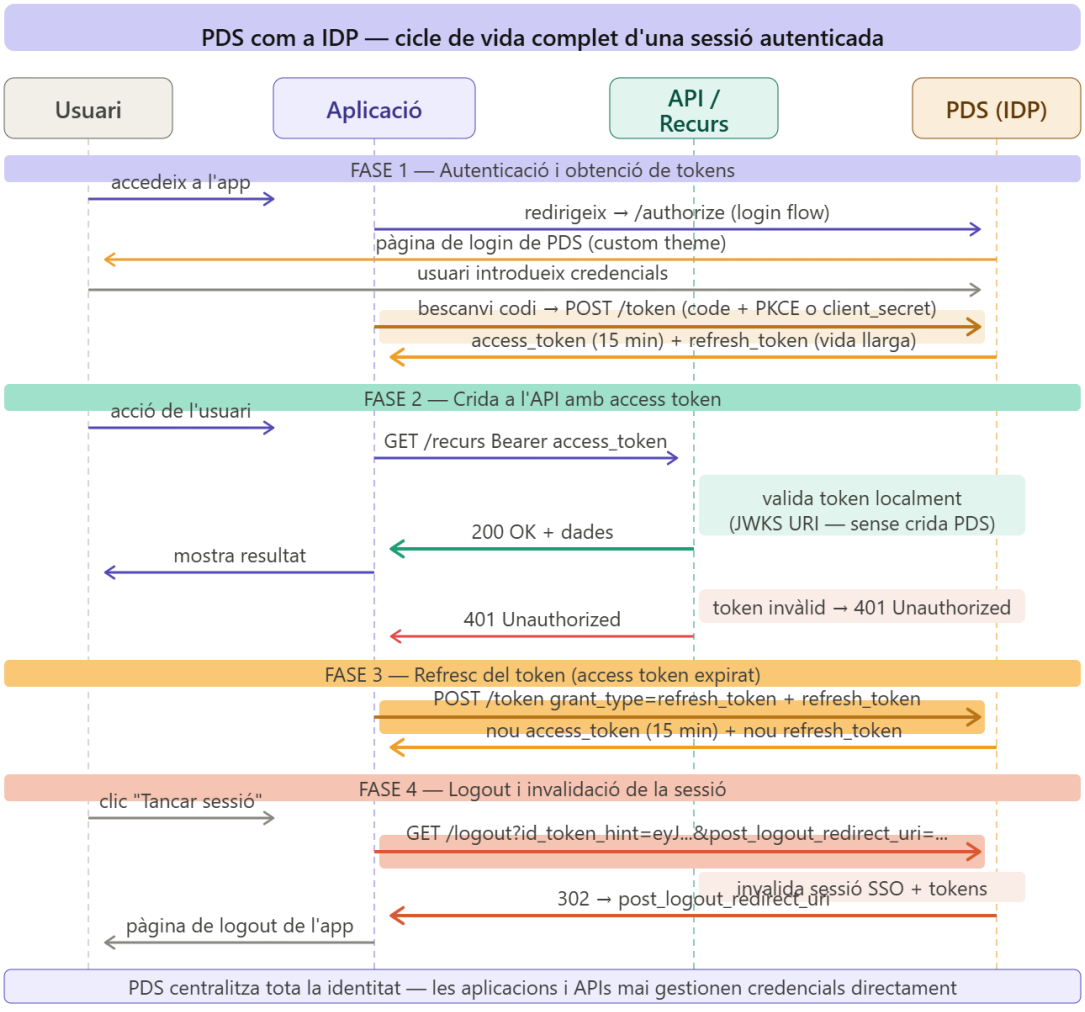


Tots els clients d'un mateix *Realm* haurien de compartir una estructura de continguts dins el token (que anomenarem "**payload**") també única, doncs la iniciativa destí ha de ser capaç d'aplicar la seva lògica d'autorització (determinar si dóna permís a accedir a l'operació o recurs sol·licitat) en base a aquesta informació inclosa en els tokens adjunts en la petició.

1.3 Autorització amb un *Realm* de tipus OpenID Connect (OIDC)

És l'operació més habitual. Una iniciativa de Salut sol·licita a la PDS un **Client** en un **Realm** nou o ja existent, per executar fluxos d'autenticació i autorització de tipus OIDC.

El següent diagrama mostra el cicle de vida complet d'una sessió autenticada amb PDS com a IDP, estructurat en quatre fases:



- La **fase 1** és l'autenticació: l'usuari mai introdueix les seves credencials en la seva pròpia aplicació, sempre ho fa a la PDS, que retorna els tokens a l'aplicació un cop verificat.
- La **fase 2** és l'ús normal: l'aplicació envia l'access_token a cada crida a l'API, que el valida localment sense tocar PDS, i retorna 200 OK o 401 segons si és vàlid.
- La **fase 3** és la renovació transparent: quan el token expira, l'aplicació usa el refresh_token per obtenir-ne un de nou sense que l'usuari ho noti.

- La **fase 4** és el logout: l'aplicació redirigeix a PDS amb l'id_token_hint, PDS invalida la sessió SSO i tots els tokens associats, i retorna l'usuari al post_logout_redirect_uri.

La idea central que il·lustra el diagrama és que **PDS centralitza tota la gestió d'identitat**: les aplicacions i les APIs mai veuen ni emmagatzemen credencials — només treballen amb tokens de vida limitada que PDS emet, signa i pot revocar en qualsevol moment.

1.3.1.1 Lliurament de credencials de client a la iniciativa adherida

L'equip de manteniment de PDS dona d'alta els *Clients* sol·licitats per la iniciativa (i el *Realm* si escau), amb la configuració de parametrització del contingut del token, les redireccions permeses després del login correcte o de logout, gestió de rols si és necessari per fer RBAC (*Role-Based Authoritzation Control*), àmbits (s*copes)* disponibles, caducitat en segons de token d'accés o de refresc, o delegació del login a un proveïdor d'identitat tercer.

Finalment, es facilita a la iniciativa adherida l'URL del *Realm* i la relació de *Clients* creats a cada entorn, amb les respectives credencials, habitualment, un `client_id` i `client_secret`.

La iniciativa sol·licitant haurà de gestionar les regles de tallafocs corresponents per accedir per HTTPS als endpoints de la PDS, que es detallen més endavant en aquest document.

1.3.1.2 Responsabilitat sobre les credencials de clients

La iniciativa adherida és responsable de custodiar les credencials facilitades per l'equip de PDS. En cas que aquestes credencials siguin compromeses, cal contactar amb l'Oficina d'Adhesions de Salut per gestionar la revocació d'aquestes i regeneració de noves.

1.3.1.3 Àmbits (*client scopes*) assignats

Com a normal general, des de la PDS no assignem cap àmbit (*client scope)* per defecte als clients, llevat del propi `openid` de l'especificació OIDC.

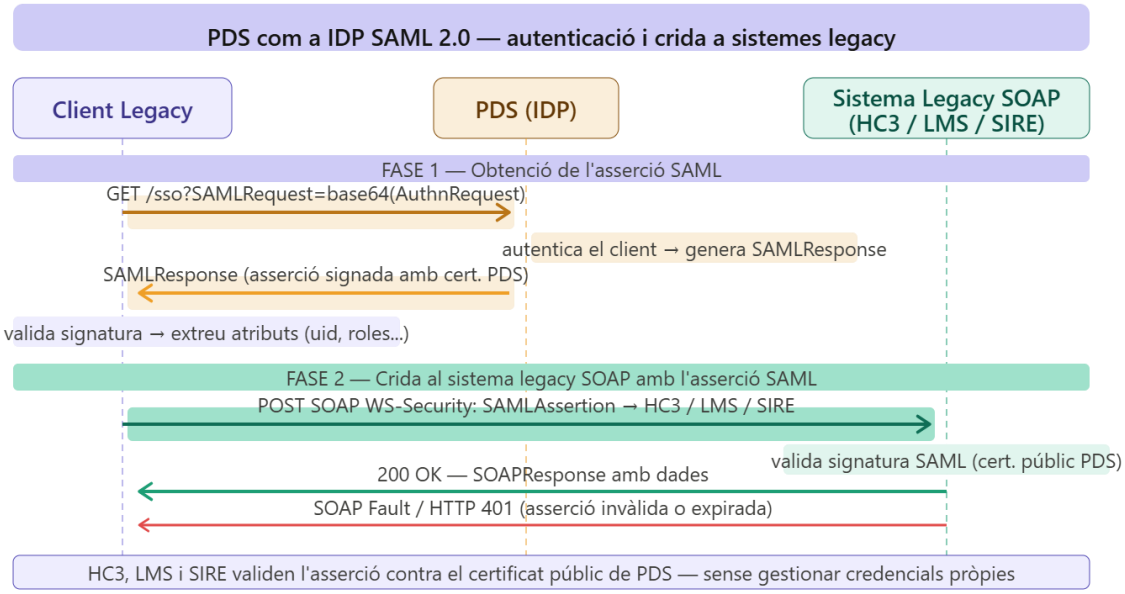
Per afegir qualsevol dels àmbits més habituals, com per exemple `profile`, `roles`, `email`, `offline_access`,... o també incorporar-ne a mida segons necessitats de la iniciativa, caldrà indicar-ho en el moment de la sol·licitud d'adhesió..

https://auth0.com/docs/get-started/apis/scopes/openid-connect-scopes

1.4 Autorització amb un *Realm* de tipus SAML 2.0

La PDS també permet generat tiquets SAML 2.0 per integracions de serveis de sistemes *legacy* de Salut basats en aquest format. Tenim diferents *Realms* amb clients SAML per invocar serveis web d'**HC3**, **LMS** o **SIRE,** alguns d'ells publicats mitjançant el **Bus d'integracions de Salut** (OSB).

Aquests serveis, que són de tipus SOAP, requereixen amb una capçalera *WebServiceSecurity* on incloure el tiquet SAML. Per tant, entre aquests sistemes i la PDS s'han configurat les claus privades de certificats per validar aquests tiquets SAML generats. Per exemple:



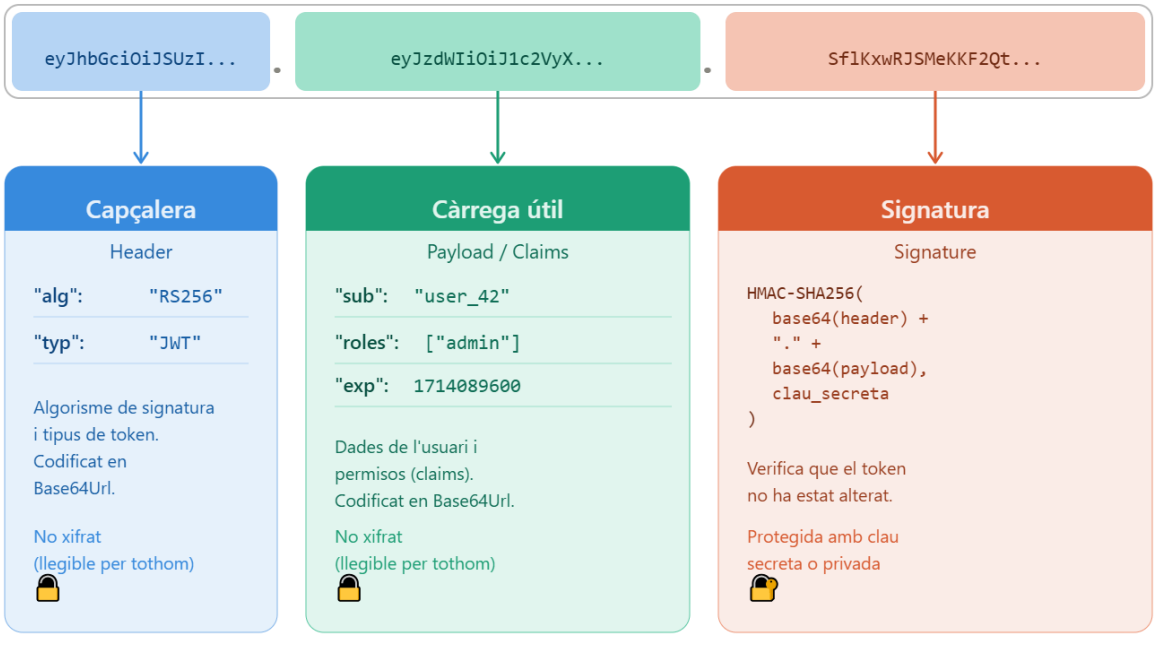
Això ofereix un principal avantatge a les iniciatives consumidores, i és que no cal demanar ni mantenir el seu propi certificat d'aplicació per signar el tiquet SAML, ni demanar la instal·lació al Bus, ni implementar la creació d'aquests tiquets (habitualment amb llibreries obsoletes), doncs els poden demanar de forma segura directament a la PDS.

2. Procés d'autorització per OpenId Connect (OIDC)

2.1 Format del token d'accés

El format d'autenticació per OIDC retorna tokens d'accés i de refresc en format **JWT** (*JSON Web Tokens*). Aquest format disposa informació en tres blocs:

- **capçalera** identificant unívocament el mecanisme i proveïdor de clau pública utilitzat en l'emisió d'aquest token d'accés.
- ***payload*** del token amb tota la informació del subjecte autenticat, parametrització i rols, segell de temps i data d'expiració, etc...
- **signatura** (apliquem RS256 de 2048 bits per defecte) per garantir la seva autenticitat.



Les tres parts es concatenen amb punts (.) — separador visible al token

https://auth0.com/docs/secure/tokens/json-web-tokens/json-web-token-structure

A continuació, repassarem tots els fluxos d'autenticació OpenId Connect disponibles a la PDS.

2.2 Authorization Code Flow (ACF):

És el cas en que el subjecte que s'autentica a la PDS per obtenir un token d'accés és un **usuari personal** amb unes credencials pròpies, habitualment un username + password.

https://www.rfc-editor.org/rfc/rfc6749#section-4.1

Aquestes credencials principalment estaran delegades a sistemes o directoris corporatius externs. Només en casos degudament justificats es farà un manteniment intern d'usuaris dins el propi *realm* a la PDS.

2.2.1 Login delegat a directoris corporatius de la Generalitat

En aquests casos, es tracta de proveïdors d'identitats federats al nostre, i per tant la pàgina de login és la pròpia del sistema, i l'administració d'aquests usuaris recau en aquest sistema destí (altes, baixes, renovació de contrasenya, vinculació de rols,...).

2.2.1.1 GICAR/GICLOUD (disponible)

El **GICAR** (i la seva nova modalitat al núvol: el **GICLOUD**) és el directori corporatiu de **professionals** de la Generalitat de Catalunya. Des de la PDS tenim la possibilitat d'autenticar un usuari amb aquestes credencials de GICAR, i proveir en el token d'accés informació com els grups de recursos (rols) que se li han assignat des de l'eina **Control d'Accés a Recursos (CAR)**, o també la Unitat Menor o Major organitzativa, o el tipus de relació del professional amb l'administració.

https://ctti.gencat.cat/ca/ctti/solucions-corporatives/gestio-didentitats/gicar/

https://gicar.intranet.gencat.cat/gdi/controlaccesrecursos/

2.2.1.2 VALID (disponible)

Servei de l'AOC per a tràmits de **ciutadans**. Admet els principals serveis d'autenticació de ciutadania: DNI-E, IDCAT Mòbil, Cl@ve PIN 24h

https://suport.aoc.cat/ca-ES/category/?id=CAT-01062

2.2.1.3 Microsoft Entra (disponible)

Directori corporatiu exclusivament per aplicacions que treballen de manera nativa amb altres components o funcionalitats de l'ecosistema **Microsoft 365** o el núvol públic d'**Azure** de CTTI, com els que contenen l'*Espai de Dades de Salut de Catalunya.*

S'ofereix la possibilitat de recuperar els grups i permisos assignats a l'usuari en el token d'accés, que després s'utilitzarà per autenticar amb aquests productes.

2.2.2 Login delegat a repositoris de credencials de sistemes de Salut

Per altre banda, també oferim la possibilitat de delegar el login de la PDS a credencials d'usuari ja existents en determinats repositoris de sistemes de Salut. En aquests casos, el procés d'autenticació es consumeix com a servei exclusivament per part de la PDS, però la gestió dels usuaris (alta, baixa, modificació, desbloqueig,...) recau també en el sistema destí.

2.2.2.1 GSA (disponible)

En aquest cas, les credencials facilitades per l'usuari són les del **Portal d'Aplicacions** *legacy* de Salut (https://salut.gencat.cat/pls/gsa/gsapk030.portal).

2.2.2.2 ARGOS (disponible)

L'usuari es pot autenticar amb les seves credencials de SAP del sistema d'Hospitalària ARGOS per obtenir tokens d'accés de la PDS.

2.2.2.3 ECAP (en backlog)

Tenim en cartera oferir des de la PDS també la possibilitat d'obtenir tokens d'accés a partir de credencials facilitades per l'usuari relatives al seu login al sistema d'atenció primària.

Aquesta funcionalitat no està encara desplegada a Producció, però atès que altres IDP com el de la iniciativa **Integrador Imatges de Salut** ja donen ara mateix aquesta funcionalitat, s'incorporarà a la PDS per donar el servei de manera transversal.

2.2.3 Token d'accés d'usuari per *Code Exchange* o *Implicit Flow*

Per obtenir un token d'accés per el login d'un usuari mitjançant un *Authorization Code Flow* oferim dues maneres: un **Code Exchange** o un **Implicit Flow**.

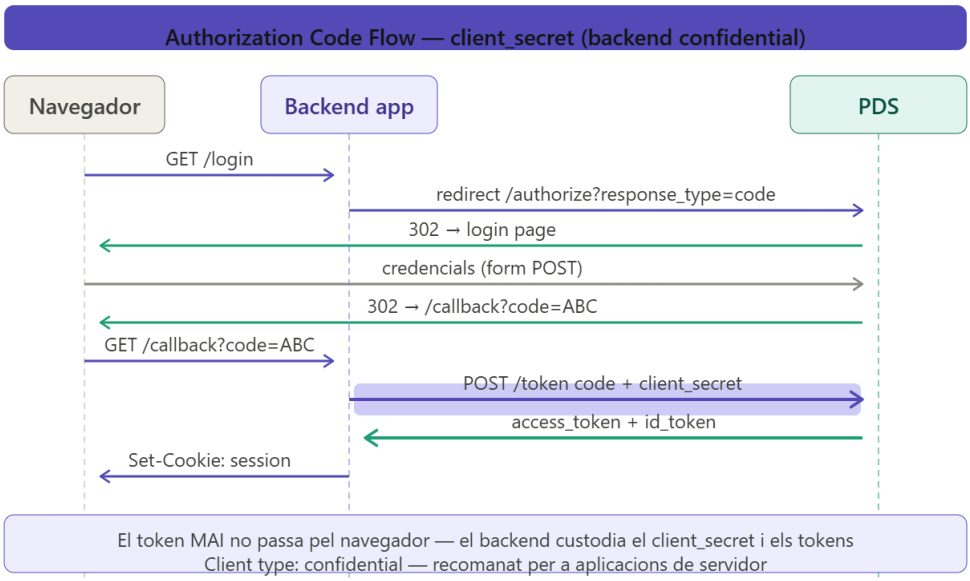
El **Code Exchange** (`response_type=code` + `grant_type=authorization_code`) és el mecanisme recomanat: la PDS retorna primer un codi d'autorització de vida molt curta al navegador via *redirect*, i el client fa un segon pas — una petició de backend (o directa amb PKCE) al token endpoint per bescanviar-lo per el token d'accés, presentat un `client_secret`; d'aquesta manera el secret no viatja mai pel navegador, ni apareix a l'historial ni als logs.

L'**Implicit Flow** (`response_type=token`), en canvi, va ser dissenyat originalment per a SPAs quan PKCE no existia: el servidor retorna el token d'accés directament al fragment de la URL del *redirect* (#access_token=...), saltant-se el segon pas. És un procés més simple però inherentment menys segur perquè el secret del client queda exposat al navegador, és accessible per JavaScript de tercers i pot filtrar-se via l'historial o el header *Referer*. L'*Implicit Flow* està **deprecated** a OAuth 2.1, i en la PDS el desactivem per defecte als clients nous.

A continuació es mostra la nostra recomanació a l'hora de fer un **Authorization Code Flow:**

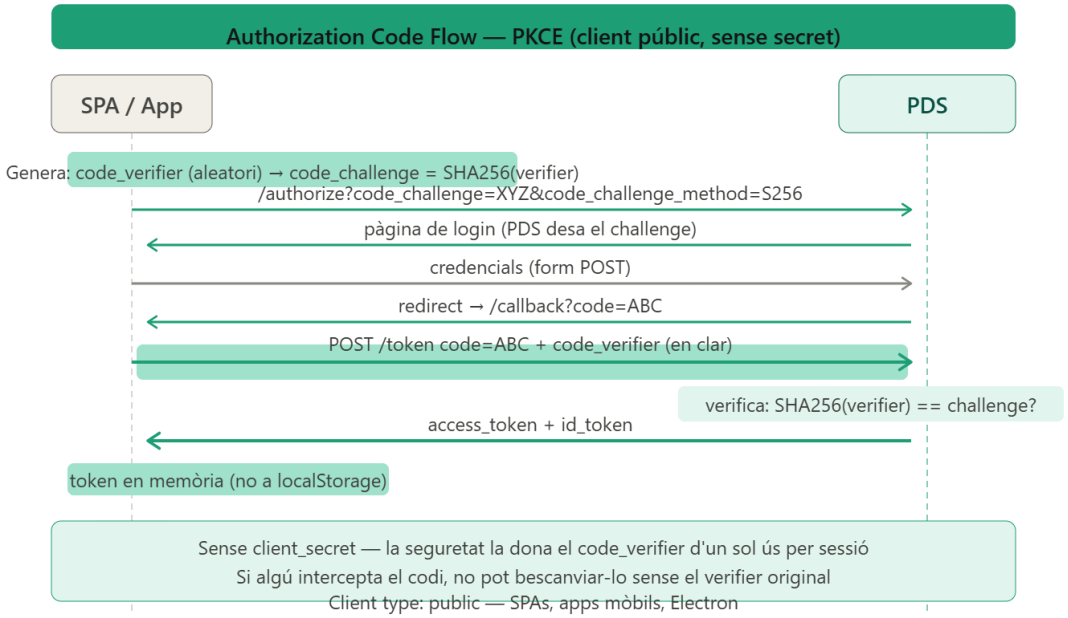
2.2.3.1 Code Exchange amb client_secret:

Patró recomanat per aplicacions client que disposen d'un **backend privat** (la majoria de sistemes de Salut compleixen aquesta arquitectura) on el secret es custodia de forma segura.



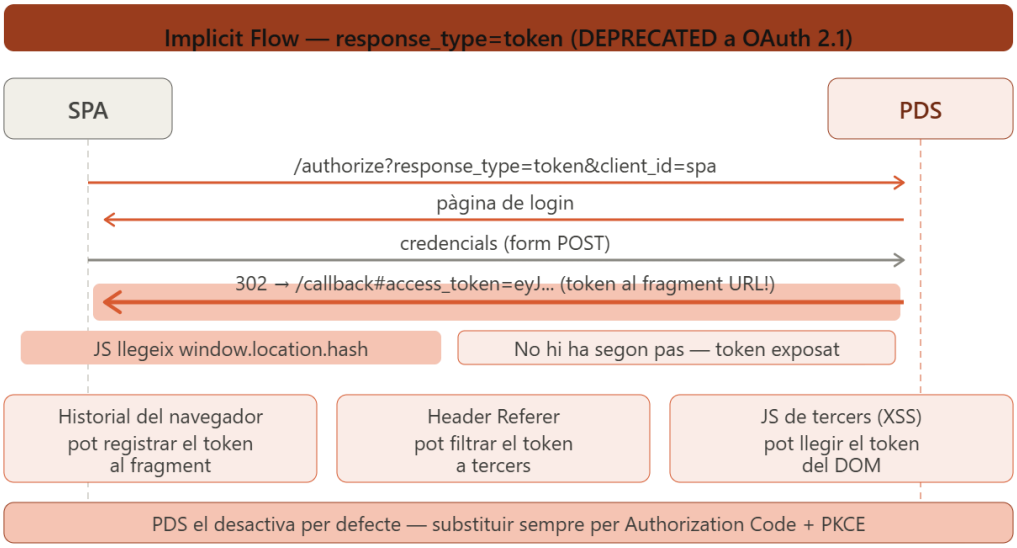
2.2.3.2 *Code Exchange* amb PKCE (*Proof Key for Code Exchange*)

Aquest patró aconsegueix el mateix nivell de seguretat per clients **SPAs** o **aplicacions mòbils**, substituint el secret per un *challenge* a superar amb un `code_verifier` d'un sol ús que es genera localment.



2.2.3.3 Implicit

Aquest fluxe només quedarà per clients d'aplicacions *legacy* degudament justificades.



2.2.4 Personalització d'estils en les pàgines de *login* i *logout*

En els fluxos d'autenticació d'usuari que no són delegats a altres IDP tercers (com els indicats de GICAR o VALID), la pàgina de login i logout és servida directament per la PDS. Per defecte, els estils d'aquestes pàgines corresponen a la versió de Keycloak en curs de la PDS:

https://github.com/keycloak/keycloak/blob/main/themes

Per altre banda, PDS permet personalitzar completament l'aspecte visual de les pàgines que gestiona directament (login, logout, error, ...) mitjançant el sistema de temes personalitzats (*custom themes*). Un tema és un conjunt de plantilles **FreeMarker**, fulls d'estil CSS i recursos estàtics que es desplega dins el directori de la instal·lació de PDS.

Això permet que la pàgina de login que veu l'usuari mostri un determinat logotip, colors de marca i la tipografia, en lloc de la pantalla genèrica de Keycloak. A més, les plantilles tenen accés a variables de context com el nom del *realm*, el client que ha iniciat el login o els missatges d'error, cosa que permet adaptar el contingut dinàmicament segons el flux.

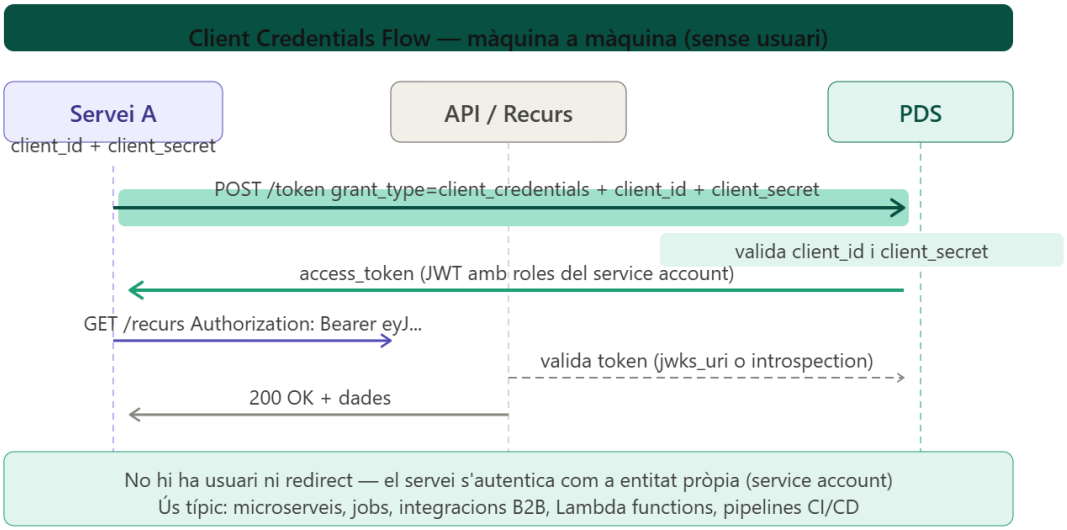
https://www.keycloak.org/docs/latest/server_development/#creating-a-theme

En la petició de creació de realm a la PDS es pot adjuntar un *Custom Theme* per modificar el tema per defecte de Keycloak, però és important garantir que és compatible amb la distribució de PDS actualment desplegada..

2.3 Client Credentials Flow (CCF)

És el cas en que el subjecte que s'autentica és un **sistema** o **service account**

En aquest escenari, les credencials que dona la PDS són únicament un **client_id** i un **client_secret** que la iniciativa ha de custodiar, i utilitzar en la petició dels tokens d'accés.



2.3.1 Parametrització del token d'accés CCF

A part de complir l'especificació d'OpenId Connect, des de la PDS podem configurar paràmetres addicionals en la petició de token d'accés en un flux *Client Credentials* que després s'incorporen com a *claims* en el *payload*. En aquest cas, el sol·licitant del *Client* a la PDS ha d'indicar en la sol·licitud d'adhesió:

- Obligatorietat dels paràmetres. Admetem cardinalitats: 0..1, 1, 0..N o 1..N.
- Possibilitat de configurar una validació dels valors permesos: contra catàlegs del **Servidor Terminològic FHIR HES**, o contra una llista tancada de valors.
- Crides al **Motor de Regles HES**, per calcular *claims* addicionals segons execució de regles respecte els paràmetres d'entrada incorporats en la petició de token d'accés.

Consideracions a tenir en compte:

- No configurem cap paràmetre nou si ja existeix un altre creat prèviament amb significat semàntic similar: per exemple demanar `tipus_usuari` si ja tenim `user_type`.
- El nom de cada paràmetre serà en anglès.
- Un cop acordats, es presenta l'estructura de *claims* dins el token JWT resultant al lot aplicacions, perquè puguin planificar el desenvolupament de l'autorització dels seus serveis o microserveis.

- Aquests *claims* addicionals s'incorporen en un camp `access_info` del *payload* del token d'accés, i els resultants de calcular regles en un altre `access_rules`.

- És important remarcar que aquests paràmetres haurien de ser relatius a l'autenticació o autorització d'un usuari. És mala pràctica utilitzar el token d'accés com a mecanisme de pas de paràmetres de navegació de l'aplicació.

Per exemple, en la següent petició de token d'accés:

```
POST https://preproduccio.pds.hes.catsalut.gencat.cat/auth/realms/PdS-DUMMY/protocol/openid-connect/token

grant_type:client_credentials
client_id:test-ccf-pre
client_secret:...
scope:openid
custom_parameter_token_1:valor-1
custom_parameter_token_2:valor-2
custom_parameter_userinfo_3:valor-3
```

Generarà el token d'accés amb aquest *payload*:

```
{
  "exp": 1777304387,
  "iat": 1777304087,
  "jti": "6c1d4b43-e1b0-42f5-acf6-...",
  "iss": "https://preproduccio.pds.hes.catsalut.gencat.cat/auth/realms/PdS-DUMMY",
  "sub": "e32b1b18-a042-40ed-8bac-...",
  "typ": "Bearer",
  "azp": "test-ccf-pre",
  "session_state": "3ea32fa7-f75f-4d3c-87b7-...",
  "acr": "1",
  "scope": "openid",
  "sid": "3ea32fa7-f75f-4d3c-87b7-...",
  "clientHost": "...",
  "access_info": {
    "custom_parameter_token_1": "valor-1",
    "custom_parameter_token_2": "valor-2",
  },
  "clientAddress": "...",
  "client_id": "test-ccf-pre"
}
```

Com es pot observar, per cada paràmetre podem decidir si es retorna en el propi token d'accés generat, o es recupera a través de l'endpoint *UserInfo* de la PDS que explicarem a continuació.

Tots els clients CCF d'un mateix *Realm* que tinguin parametrització a mida en el *payload* del token haurien de compartir la mateixa configuració, de manera que la iniciativa destí no tingui que fer tractaments a mida de la seva autorització segons el client que li ha demanat el token.

2.3.2 Especificació de paràmetres del Realm "HES"

El *Realm* més gran de la PDS en quant a nombre de clients és el "HES":

https://[ENTORN_PDS]/auth/realms/HES/.well-known/openid-configuration

Conté els clients de les iniciatives del nou **Historial Electrònic de Salut Ambulatori** (HES).

https://salutweb.gencat.cat/ca/ambits-actuacio/linies/tic/governanca-innovacio-seguretat/linies-estrategiques-treball/l2-historial-electronic-salut-ambulatori/

Aquest *Realm* disposa d'una configuració en el token d'accés de tipus *Client Credentials* molt especialitzada, i al mateix temps restrictiva, atès que hi ha moltes iniciatives adherides que s'intercanvien aquests tokens d'accés entre elles, per tant és molt important respectar el format en tots els clients associats.

Disposem d'un manual addicional amb el detall de tots els paràmetres possibles a incloure com a *claim* dins el token, i les validacions realitzades per cadascun de valors acceptats.

2.3.3 Creació de tokens per serveis sota sondes de monitorització

Les aplicacions que tinguin serveis connectats a sondes de navegació i monitorització (per exemple, desplegades a la plataforma d'observabilitat corporativa de Talaia) necessitaran que aquestes operacions també siguin degudament autenticades per fer l'autorització.

En aquest cas, des de PDS donem la possibilitat de crear clients CCF amb una parametrització estàtica dels *claims*, de manera que els tokens generats sempre tinguin el mateix *payload* de valors, i únicament canvia les dates d'expiració del token.

És molt preferible utilitzar aquest tipus de client "de sonda" abans de donar les credencials del client propi del sistema consumidor, atès que això donaria llibertat a generar tokens amb diferents propòsits més enllà de l'abast del servei a monitoritzar.

En cas de necessitar un client per invocar serveis per una prova de *PenTest* o similar, també es sol·licitaran tokens d'aquest format, amb els *claims* prefixats.

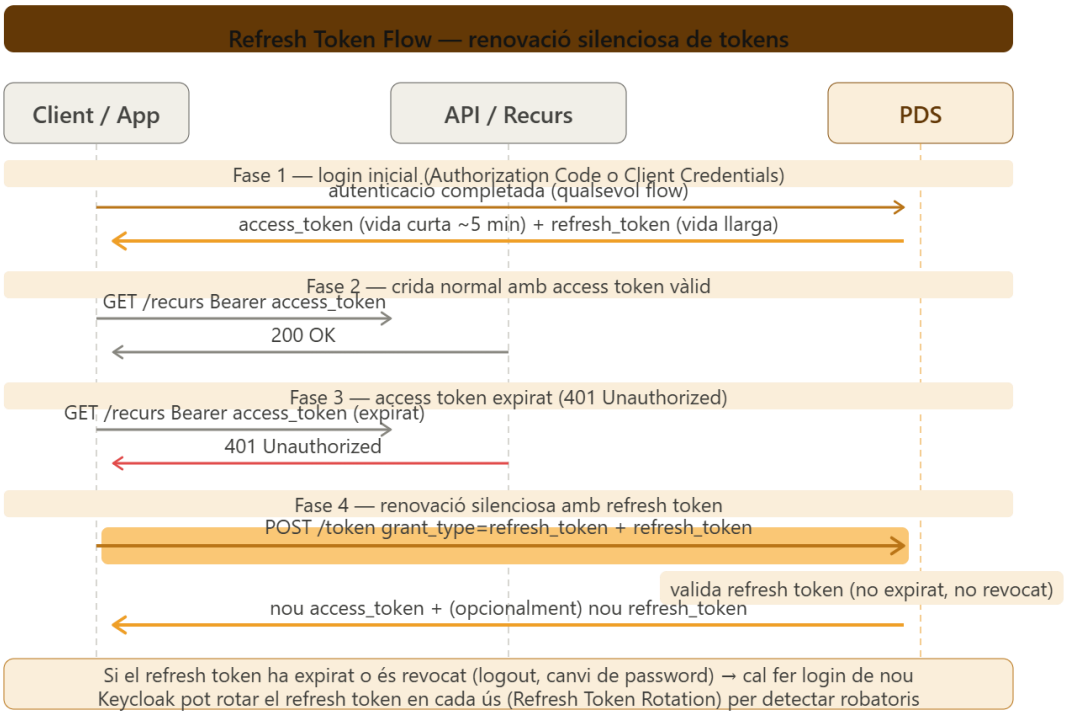
2.4 Refresh Token Flow

En el cas en que un token d'accés ha caducat, l'estàndard OIDC ofereix també el token de refresc (*refresh_token*) per generar un token d'accés nou de manera que l'usuari no tingui que autenticar-se de nou. El següent diagrama mostra les quatre fases del cicle de vida d'un token:

La **fase 1** és el login inicial. Qualsevol dels flows anteriors de login d'usuari (Authorization Code, PKCE o Client Credentials) retorna sempre dos tokens: l'access_token de vida curta (típicament 15 minuts en la PDS) i el refresh_token de vida llarga (hores o dies).

El client va accedint als recursos de la iniciativa adherida adjuntant el token d'accés a cada petició, en un request header "*Authoritzation*".

Les **fases 3-4** és la clau del refresc: quan l'API retorna un 401 perquè el token ja ha caducat, el client no demana al usuari que torni a fer login, sinó que fa un POST silenciós al token endpoint amb el refresh_token. PDS el valida i retorna un nou access_token, i opcionalment un nou refresh_token si tens activada la **Refresh Token Rotation**, que és el mecanisme de seguretat que detecta si un refresh token ha estat robat i usat dues vegades.



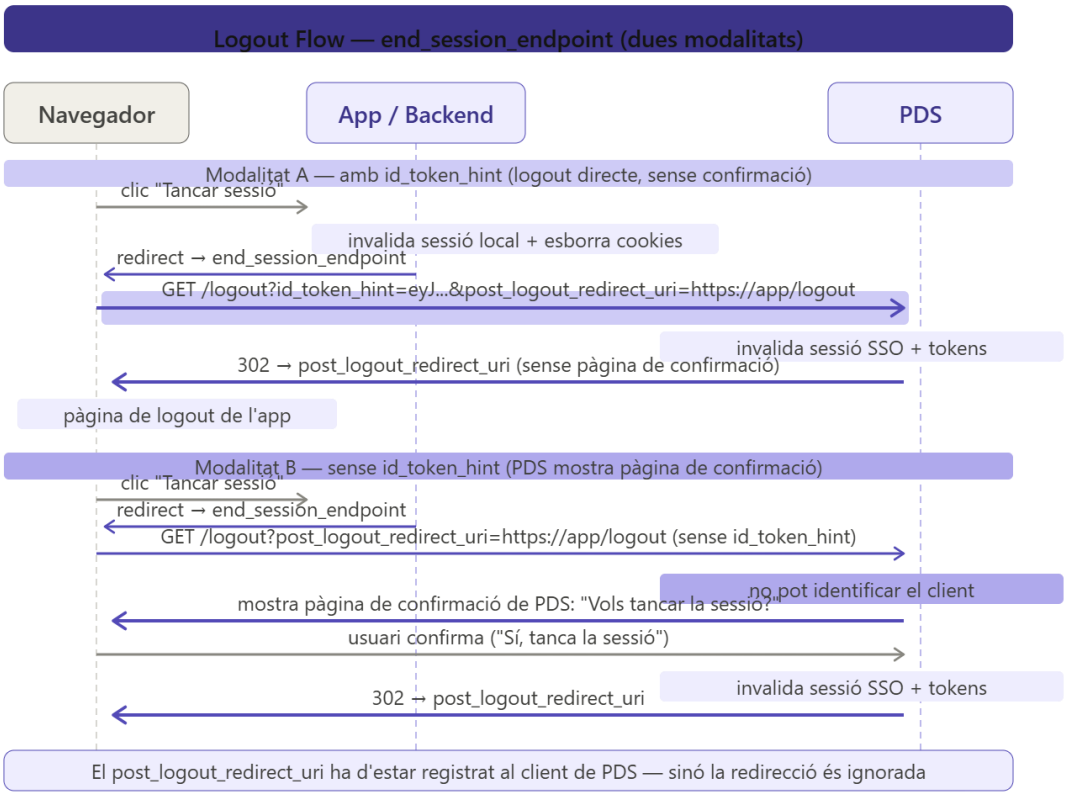
2.5 Logout Flow

El *Realm* de la PDS exposa també l'endpoint per realitzar el logout, que implica finalitzar la sessió, i per tant els tokens d'accés o de refresc relacionats queden sense utilitat. Aquest flux de logout és una implementació específica d'OIDC; i requereix que la petició de token d'accés inclogui l'àmbit `scope=openid`, de manera que la PDS retorni també un *id_token* més lleuger que només identifica la sessió a fer el logout, lliurat com un paràmetre `id_token_hint`.

També existeix l'opció de NO enviar el *id_token* en la petició de logout. En aquest cas, la PDS oferirà un frontend que demanarà al usuari confirmar l'operació, ja que necessita llegir la sessió a alliberar de la *cookie* de l'usuari. El següent diagrama mostra les dues modalitats clarament diferenciades:

- La **modalitat A** (amb *id_token_hint*) és el logout net: PDS sap exactament quin client i quina sessió ha d'invalidar, fa la neteja directament i redirigeix sense interrupció al post_logout_redirect_uri. L'usuari no veu cap pàgina intermèdia de PDS.

- La **modalitat B** (sense *id_token_hint*) és el logout amb fricció: PDS no pot identificar el client de forma segura, de manera que mostra la seva pròpia pàgina de confirmació per assegurar-se que el logout és intencional. L'usuari ha de confirmar explícitament, i només llavors PDS invalida la sessió i redirigeix.



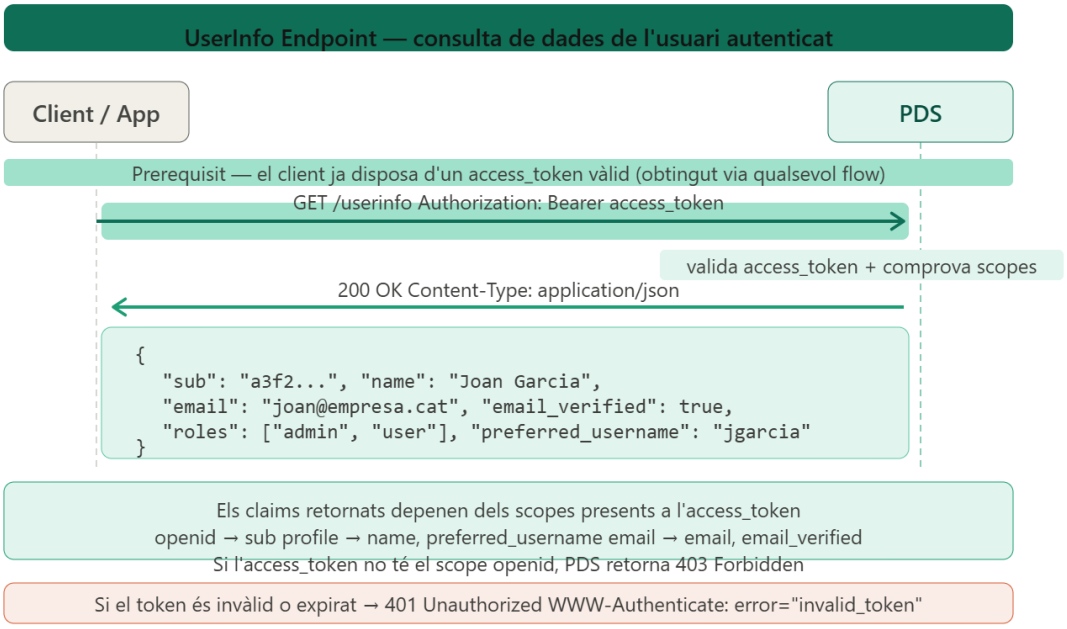
En cas de tenir integració amb login de credencials de tercers, és possible que aquesta `post_logout_redirect_uri` sigui pròpia d'aquest destí, per alliberar també les seves dades de sessió o cookies relacionades.

2.6 Endpoint UserInfo d'informació de sessió d'usuari

Cada *Realm* també disposa d'un endpoint **UserInfo**, que retorna els atributs que consten en la sessió de l'usuari o sistema autenticat a la PDS segons un token d'accés vàlid i no expirat, independentment de si aquestes dades també s'han inclòs en el *payload* d'aquest token.

Hi ha dues raons principals per usar aquest endpoint en lloc de llegir les dades directament del JWT. Primera: podem reduir la mida del token d'accés (no incloent informació que segurament és poc rellevant per l'autorització de les iniciatives adherides: nom, cognoms,...), i recuperar aquestes dades addicionals de l'UserInfo. Segona: garanteix que la informació és sempre fresca — si un rol de l'usuari ha canviat des que es va emetre el token, l'UserInfo reflectirà l'estat actual.

La restricció clau és que la petició d'access_token ha de portar el `scope=openid`, sense ell, PDS retorna un 403. La resta de *claims* depenen de la configuració del client dins el *realm*.



2.7 Validació d'un token d'accés OIDC

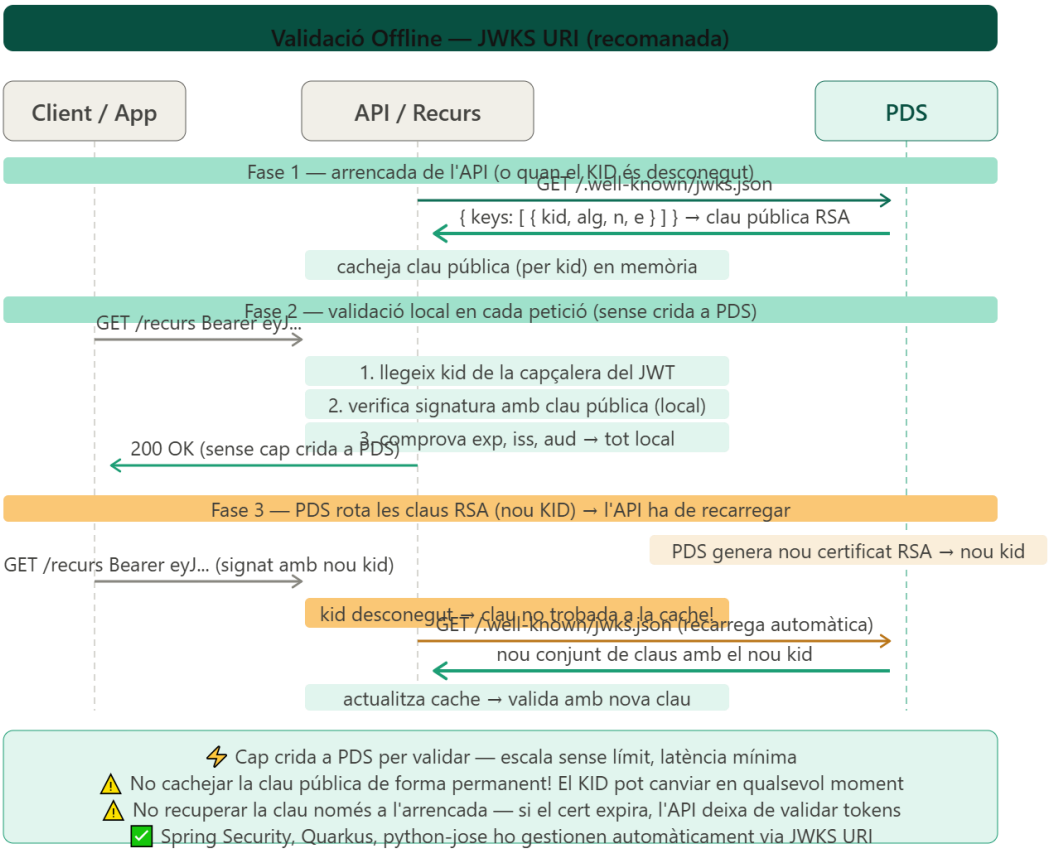
Existeixen dues maneres de garantir que un token adreçat per la PDS és vàlid i no ha estat manipulat ni generat per un possible atacant extern.

2.7.1 Validació offline mitjançant clau pública i JWKS URI

L'opció recomanada és validar el token de forma local per la pròpia iniciativa, doncs escala perfectament al no generar cap càrrega sobre PDS ni latència addicional a cada *request*.

El **JWKS URI** és un endpoint estàndard que exposa PDS públicament i que retorna el conjunt de claus públiques del *realm* en format *JSON Web Key Set*. Cada entrada del conjunt conté els paràmetres de la clau pública asimètrica, l'algorisme, i, el més important, el `kid` (*Key ID*), que és l'identificador únic d'aquella clau. Quan PDS emet un JWT, n'inclou el *kid* a la capçalera del token, de manera que qualsevol API que rebi el token pot anar al JWKS URI, trobar la clau amb aquell *kid* concret, i verificar criptogràficament que la signatura del token és vàlida — és a dir, que realment l'ha signat PDS i que ningú l'ha alterat — tot sense necessitat de cridar per HTTP a PDS ni compartir cap secret entre l'API i PDS.

La part crítica és la gestió d'aquest `kid`: PDS signa els tokens amb certificats asimètrics que tenen una data d'expiració, i quan roten, el *kid* i la clau pública exposada al JWKS URI canvia.

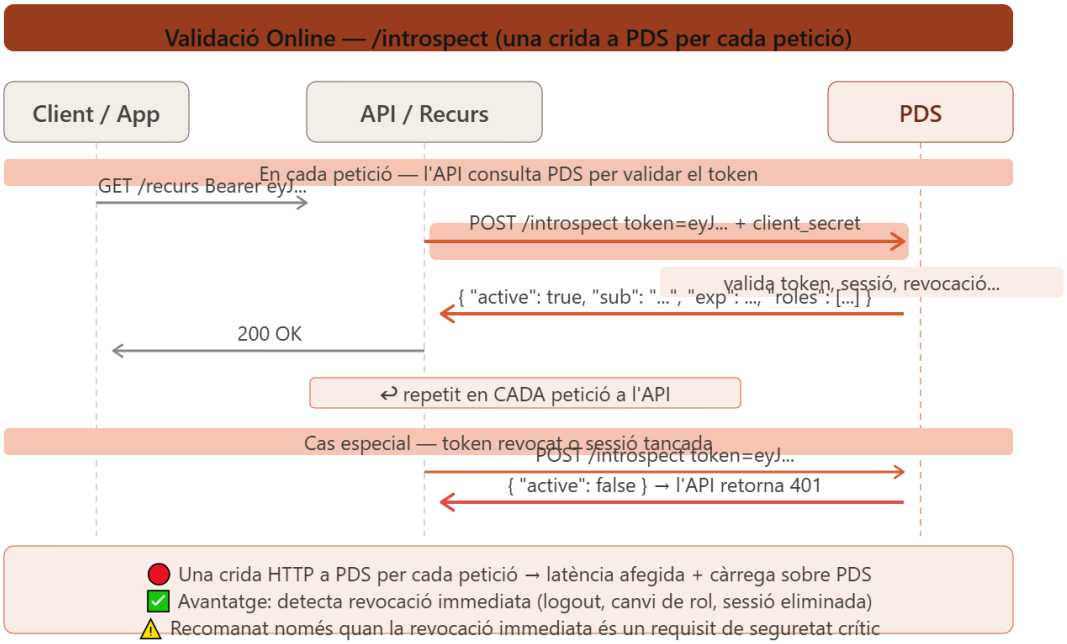


Per tant, les aplicacions adherides mai han de cachejar aquesta clau de forma permanent ni recuperar-la únicament a l'arrencada del seu contenidor o servidor, si no que han d'estar preparades per recarregar el JWKS quan troben un *kid* desconegut.

Frameworks com Spring Security o Quarkus ho fan automàticament configurant únicament aquest endpoint del *realm*.

2.7.2 Validació per introspecció de token

En aquest cas l'API fa un POST a PDS en cada petició per preguntar si el token és vàlid. El gran avantatge és que detecta revocació immediata: si un usuari fa logout o un administrador li elimina la sessió, la propera crida ho detectarà al moment..



El cost és evident: afegeix una crida HTTP extra a PDS en cada *request*, cosa que incrementa la latència i, en sistemes d'alta concurrència, pot convertir-se en un coll d'ampolla.

2.8 Resum d'endpoints rellevant per una integració de tipus OIDC

Qualsevol Realm de la PDS disposa de l'endpoint genèric OIDC amb tota la informació:

```
https://preproduccio.pds.hes.catsalut.gencat.cat/auth/realms/PdS-DUMMY/.well-known/openid-configuration
```

Aquí podem obtenir, l'*Issuer* que indica de quin destí s'ha obtingut aquest token d'accés:

```
"issuer":
"https://preproduccio.pds.hes.catsalut.gencat.cat/auth/realms/PdS-DUMMY"
```

També podem trobar els diferents endpoints per un flux de login d'usuari, demanar un token d'accés o un token de refresc, recuperar la informació addicional de sessió de l'usuari, o fer la petició de logout:

```
"authorization_endpoint":
https://preproduccio.pds.hes.catsalut.gencat.cat/auth/realms/PdS-
DUMMY/protocol/openid-connect/auth",

"token_endpoint":
"https://preproduccio.pds.hes.catsalut.gencat.cat/auth/realms/PdS-
DUMMY/protocol/openid-connect/token"

"userinfo_endpoint":
”https://preproduccio.pds.hes.catsalut.gencat.cat/auth/realms/PdS-
DUMMY/protocol/openid-connect/userinfo"

"end_session_endpoint":
”https://preproduccio.pds.hes.catsalut.gencat.cat/auth/realms/PdS-
DUMMY/protocol/openid-connect/logout"
```

També podem recuperar l'endpoint JWKS per obtenir la clau pública del certificat amb el que es signen els tokens, aquest punt és molt important perquè aquest certificat tindrà una rotació periòdica, tal com explicarem més endavant en aquest document:

```
"jwks_uri":
"https://preproduccio.pds.hes.catsalut.gencat.cat/auth/realms/PdS-
DUMMY/protocol/openid-connect/certs"
```

I es retorna informació referent als algorisme de signatura suportats, àmbits (scopes) disponibles, o tipus de clients suportats.

3. Integració amb sistemes d'identitat corporatius

La PDS ofereix la possibilitat de servir tokens d'accés a usuaris autenticats en directoris i sistemes d'identitats corporatius de la Generalitat de Catalunya, amb un *Realm* específic creat a tal efecte anomenat **PDS-GENCAT-HIBRID** a cada entorn:

https://pds.hes.catsalut.gencat.cat/auth/realms/PDS-GENCAT-HIBRID

https://preproduccio.pds.hes.catsalut.gencat.cat/auth/realms/PDS-GENCAT-HIBRID

https://integracio.pds.hes.catsalut.intranet.gencat.cat/auth/realms/PDS-GENCAT-HIBRID

Es tracta de fluxos d'autenticació de tipus ACF (*Authorization Code Flow*) amb el corresponent *Client* que s'hagi demanat en el procés d'adhesió.

3.1 Delegació del login a GICAR (directori de professionals)

GICAR és l'eina amb què s'ha dotat a la Generalitat de Catalunya per a la gestió centralitzada de les identitats de persones que interactuen amb els sistemes d'informació.

Facilita la gestió de les identitats (persones) que treballen o col·laboren amb la Generalitat de Catalunya (funcionaris, interins, empreses públiques, externs, etc.), així com la gestió i el control de l'accés als recursos per part de les identitats -entenent com a recurs qualsevol eina o sistema d'informació necessari perquè el professional desenvolupi les seves funcions.

En aquest escenari, quan un usuari sol·licita una petició d'autenticació, la PDS fa automàticament una redirecció a GICAR, presentant la seva pantalla de login:

En cas d'autenticar-se correctament l'usuari, s'executa una redirecció GET a la `redirect_uri` pactada entre la iniciativa i PDS, per prosseguir amb el flux per obtenir el token d'accés.

3.1.1 Atributs recuperats per la PDS en una autenticació GICAR

La PDS recupera els següents paràmetres del login delegat a GICAR:

- **Codi d'usuari**: en aquest cas és el DNI, en un *claim* `preferred_username`.
- **Nom d'usuari**, en un *claim* `given_name`.
- **Cognoms d'usuari**, en un *claim* `family_name`.
- **Correu electrònic** de GICAR, en un *claim* `email`.
- (Opcional) **Rols**: llista de grups GICAR associats a l'usuari autenticat en el *Client* de la PDS, en un *claim* `roles`, tal com es descriu en el següent apartat.
- (Opcional) **Unitat Major** / **Unitat Menor**: són atributs del directori corporatiu de GICAR, en *claims* `organization` i `organizationalUnit` respectivament
- (Opcional) Tipus de **relació professional** de l'usuari en un *claim* `employeeType`.

3.1.1.1 Configuració de rols amb el Control d'Accés de Recursos (CAR)

Una aplicació que s'adhereix a la PDS i vol fer un flux d'autenticació amb credencials de GICAR, pot sol·licitar també la creació d'un grup de recursos al GICAR, gestionat mitjançant del sistema **Control d'Accés de Recursos** (CAR)

https://gicar.intranet.gencat.cat/gdi/controlaccesrecursos/
https://preproduccio.gicar.intranet.gencat.cat/gdi/controlaccesrecursos/

Es tracta d'una eina web que permet assignar grups (rols) a usuaris. L'accés a aquesta web requereix accedir per multi-factor amb l'eina **DuoPush**.

La PDS incorpora després aquests grups com a *claims* en el *token* d'accés generat per un usuari autenticat per un *Client* d'aquesta aplicació.

El procediment consisteix en:

- Sol·licitar a través d'un tiquet **ACOGICAR** del JIRA de CTTI la creació d'un **grup de recursos** que ha de tenir la nomenclatura:

  `[CODI_APLICACIO]_SLT[ABREVIATURA]-GICARDC`

  Per exemple:

  `3678_SLTEXEMPLE-GICARDC`

  https://cstd-ctti.atlassian.net/browse/ACOGICAR

  En el tiquet cal indicar les persones que seran administradors d'aquest grup de recursos, i per tant tindran permisos per gestionar rols a usuaris.

Sol·licitar en el mateix tiquet la creació dels rols, per exemple:

```
3678_STLEXEMPLE_RolAdmin
3678_STLEXEMPLE_RolWrite
3678_STLEXEMPLE_RolReadOnly
```

- Després es realitza l'assignació d'usuaris a aquests grups des de la web de GICAR.
- La PDS recupera tots els grups de l'usuari de GICAR, i per aquells que coincideix amb el codi aplicació, els informa dins del token d'accés en un claim `roles` de tipus llista.

3.1.2 Entorns de GICAR productius i no productius

Existeixen dos entorns de GICAR, un de no-productiu, i un altre productiu. La PDS als entorns de INT i PRE autentica contra el primer, i al entorn de PRO contra el segon. No es pot fer cap excepció a tal efecte.

Els usuaris per tant han d'estar presents al GICAR tant de PRE com de PRO en cas de necessitar accedir a l'aplicació mitjançant la PDS a entorns de INT i PRE. Aquesta gestió es pot demanar en un tiquet ACOGICAR.

3.1.2.1 URL de redirecció després del logout

Per fer el flux de Logout a la PDS per un client redirigit a GICAR, cal afegir un paràmetre addicional `post_logout_redirect_uri` amb la URL de logout de GICAR. D'aquesta manera, ens assegurem que s'elimina la sessió tant de la PDS com de GICAR. En cas contrari, ens podem trobar que una de les dues sessions es manté activa en l'estació de treball de l'usuari, i en una posterior autenticació no es tornen a sol·licitar credencials.

Segons si el client de la PDS ja ha migrat el login delegat al GICLOUD o encara està al GICAR on-premise, pot ser diferent:

Entorn de PRE

https://preproduccio.autenticaciogicar4.extranet.gencat.cat/idp/logoutgicar.jsp

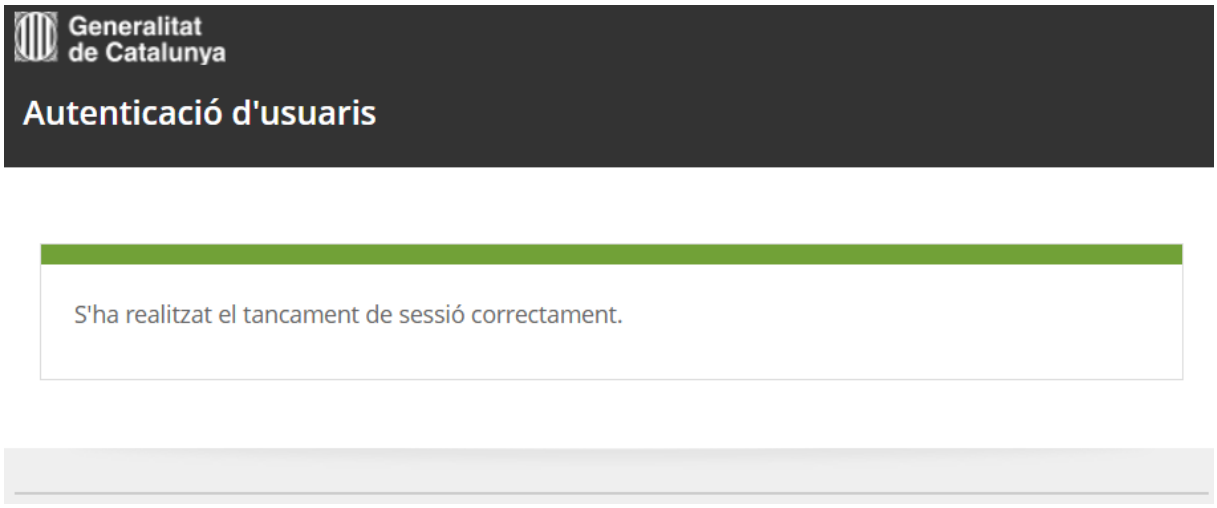
https://preproduccio.autenticaciogicar5.extranet.gencat.cat/realms/gicar-ad/protocol/openid-connect/logout?logoutgicar

Entorn de PRO

https://autenticaciogicar4.extranet.gencat.cat/idp/logoutgicar.jsp

https://autenticaciogicar5.extranet.gencat.cat/realms/gicar-ad/protocol/openid-connect/logout?logoutgicar

El resultat de logout de GICAR és la següent pàgina:


Generalitat
de Catalunya
Autenticació d'usuaris
S'ha realitzat el tancament de sessió correctament.

3.2 Delegació del login de VALID (ciutadans)

VALID és l'integrador de serveis d'identitat digital de Catalunya perquè la ciutadania tingui totes les opcions per tramitar fàcilment, i és ofert per la AOC (Consorci Administració Oberta de Catalunya)

La PDS ofereix, en el mateix *Realm* indicat en l'apartat anterior, la possibilitat d'incorporar clients on es delegui el login a VALID, mitjançant també un flux ACF.

En aquest cas, la PDS ofereix la pantalla de login de VALID, presentant diferents opcions d'autenticació a l'usuari, com per exemple **IDCAT Mòbil** o **DNIe**:

És possible que algun dels accessos com el de idCAT Mòbil requereixi un doble factor d'autenticació, en aquest cas, per SMS.

3.2.1 Atributs recuperats per la PDS en una autenticació VALID

Per a l'autenticació delegada a VALID, des de la PDS només podem recuperar els següents atributs del login que ha fet el ciutadà:

- Codi d'usuari: també és el DNI, en un claim `preferred_username`.
- Nom d'usuari, en un claim `given_name`.
- Cognoms d'usuari, en un claim `family_name`.

Però en el cas de VALID, no es disposa d'un sistema de Control de Recursos com el de GICAR, per assignar atributs addicionals als ciutadans en forma de rols.

En aquesta necessitat, la iniciativa haurà de gestionar aquest manteniment de rols per usuaris des del seu propi backend, fora de l'àmbit de la PDS.

3.2.2 Entorns de VALID productius i no productius

Com en el cas de GICAR, també existeixen dos entorns de VALID: un de Producció, i un altre no productiu. La PDS als entorns de INT i PRE autentica contra el VALID no-productiu, i al entorn de PRO contra el VALID productiu. No es pot fer cap excepció a tal efecte.

Els usuaris per tant han d'estar donats d'alta als dos entorns de VALID, si volen autenticar-se mitjançant la PDS en tots els entorns. L'alta d'usuaris al VALID és una gestió que el propi ciutadà pot realitzar des de la web de l'AOC.

3.2.2.1 URL de redirecció després del logout

De nou,és necessari configurar una URL de redirect del logout de la PDS cap al logout del VALID, per assegurar que ambdues sessions s'eliminen. Això s'indica mitjançant el paràmetre post_logout_redirect_uri que en cada entorn ha de ser:

Entorn de PRE

https://preproduccio.passarelagicarvalid.idp1-gicar.gencat.cat/corpvalididp/Custom/Logout

https://preproduccio.autenticaciogicar5.extranet.gencat.cat/realms/gicar-ad/protocol/openid-connect/logout?logoutgicar

Entorn de PRO

https://passarelagicarvalid.idp1-gicar.gencat.cat/corpvalididp/Custom/Logout

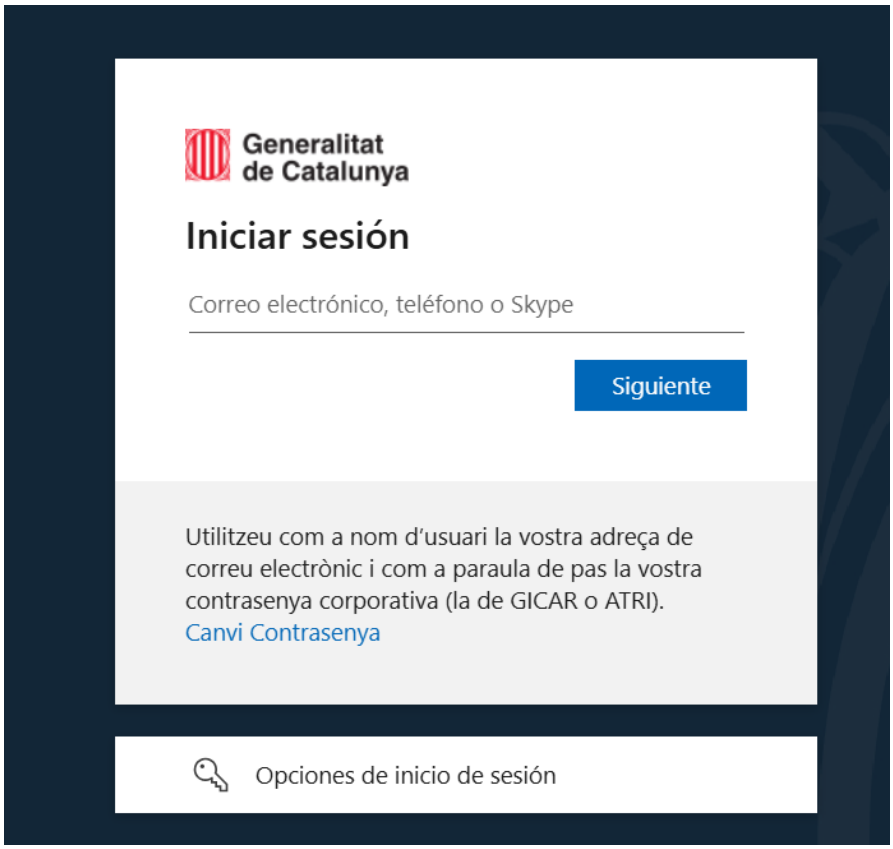
https://autenticaciogicar5.extranet.gencat.cat/realms/gicar-ad/protocol/openid-connect/logout?logoutgicar

El resultat de logout de VALID és el mateix que en el cas de GICAR.

3.3 Delegació del login a Microsoft Entra (CTTI)

Es tracta d'un directori corporatiu amb tots els permisos de l'ecosistema Microsoft configurats per CTTI, com pot ser la suite **Microsoft 365** o el núvol públic **Azure**. Des de la PDS podem federar el login de l'usuari amb aquestes credencials, però es farà només per sistemes que han d'integrar-se nativament amb aquests productes.

La pàgina de login és la pròpia del *tenant* de CTTI per Microsoft (és la mateixa per accedir per exemple a Microsoft Teams, Outlook, ...) i requereix un doble factor d'autenticació normalment vinculat a **Microsoft Authenticator**.



Possiblement la majoria de professionals de la Generalitat de Catalunya disposen d'aquestes credencials, per la seva llicència de Microsoft 365, malgrat aquests grups no seran d'utilitat per autoritzar sistemes d'informació o inciatives de Salut.

3.3.1 Atributs recuperats per la PDS en una autenticació Entra

Des de la PDS podem recuperar els següents atributs del login d'Entra:

- Codi d'usuari: també és el DNI, en un *claim* `preferred_username`.
- Nom d'usuari, en un *claim* `given_name`.
- Cognoms d'usuari, en un *claim* `family_name`.

- Estructura de rols, en un *claim* `groups` de tipus llista. Des de Microsoft només podem recuperar el UID (identificador intern) del grup, i no el seu *display name* o descripció. Per obtenir aquesta relació entre identificadors i descripció de grups d'Entra, cal obrir un tiquet de tipus ACOGICAR.

3.3.2 Entorns de Microsoft Entra

En aquest cas, només disposem d'un entorn únic de directori d'Entra de tipus productiu. Per tant, tots els Clients de la PDS que federin el login aquí ho faran contra aquest directori, independentment de l'entorn de la PDS.

Tampoc és necessari posar cap URL de redirecció addicional per fer el logout, doncs aquest es gestiona amb un *backchannel logout* del propi Realm de la PDS, per exemple:

https://[ENTORN_PDS]/auth/realms/PDS-GENCAT-HIBRID/protocol/openid-connect/logout

4. Integració amb credencials de sistemes de Salut

També podem configurar en la PDS un login delegat a credencials d'usuari ja disponible en altres sistemes de Salut que ja disposen del seu propi repositori d'usuaris i contrasenyes, amb el corresponent manteniment d'aquests.

4.1 Login amb credencials de GSA (disponible)

La PDS exposa el següent *Realm* on es poden demanar clients per fer fluxes d'autenticació ACF per usuaris amb les seves credencials del portal de GSA:

https://[ENTORN_PDS]/auth/realms/PDS-Salut-GSA

És important destacar que GSA és el Portal d'Aplicacions *legacy* de Salut. Per tant, funciona en base a una relació de permisos entre usuaris i aplicacions que cal respectar. Per demanar l'alta d'un client a PDS per autenticar amb credencials de GSA, s'ha de fer sobre un codi d'aplicació existent també a GSA.

En el *payload* del token d'accés de la PDS estarà disponible la següent informació de l'usuari autenticat amb les seves credencials de GSA:

- Identificador de l'usuari a GSA, en un *claim* `preferred_username`. Per GSA, en alguns casos pot ser el DNI o en d'altres pot ser un valor alfanumèric.
- Nom de l'usuari, en un *claim* `given_name`.
- Cognoms de l'usuari, en un *claim* `family_name`.
- ID alternatiu de l'usuari a GSA, en un *claim* `alternative_id`. Aquest valor és el camp històric "numfaxdni" que té GSA a la seva base de dades.
- Assignació de perfils de l'usuari al codi d'aplicació GSA corresponent al client, en un claim `gsa_profiles` de tipus llista, on cada element consta d'un nomPerfil i d'un àmbit.
- Informació rellevant del login al portal, en un *claim* `gsa_avis`, que pot tractar d'avisos com per exemple que la contrasenya està a punt de caducar.

Addicionalment, el procés d'obtenció de token amb credencials de GSA pot retornar un *401 Unauthorized* en diferents casuístiques com: l'usuari té la contrasenya caducada a GSA, o està bloquejat, o aquest usuari no té accés a aquesta aplicació en el Portal d'Aplicacions.

4.2 Login amb credencials d'ARGOS

La PDS exposa el següent *Realm* on es poden demanar clients per fer fluxes d'autenticació ACF d'usuaris amb les seves credencials del sistema SAP d'ARGOS, que dóna servei als diferents Sistemes Hospitalaris de Salut:

https://[ENTORN_PDS]/auth/realms/PDS-Salut-ARGOS

En el *payload* del token d'accés de la PDS estarà disponible la següent informació de l'usuari autenticat amb les seves credencials d'ARGOS:

- Identificador de l'usuari a ARGOS, en un claim `preferred_username`.
- Nom de l'usuari, en un claim `given_name`.
- Cognoms de l'usuari, en un claim `family_name`.
- Servei d'ARGOS en un claim `service`.
- Rol del professional a ARGOS en un claim `rol`.
- Número de col·legiat de professional a ARGOS, en un claim `num_collegiat`.

5. Procés d'autorització per SAML 2.0

Dins la PDS disposem de diferents *Realms* amb clients per generar tiquets SAML 2.0. Es tracta d'un protocol d'autenticació molt comú en serveis web de tipus SOAP, dels quals encara tenim molts sistemes d'informació de Salut que utiltizen aquest format. Els que donem cobertura des de la PDS són **HC3**, **LMS** i **SIRE**.

https://[ENTORN_PDS]/auth/realms/SIRE-SAML/protocol/saml/clients/hc3-saml-client

https://[ENTORN_PDS]/auth/realms/LMS-SAML/protocol/saml/clients/lms-saml-client

https://[ENTORN_PDS]/auth/realms/HC3-SAML/protocol/saml/clients/sire-saml-salut

Aquests sistemes tenen identificat a la PDS com entitat adherida, per tant, hi ha una relació de confiança amb el seu Certificat d'Aplicació (CDA) els tiquets SAML 2.0 generats.

La utilització d'aquests *Realms* permet a les iniciatives client demanar, mantenir i renovar periòdicament CDA propis per cadascuna, o tenir que implementar la creació d'aquests tiquets (amb llibreries o frameworks segurament ja antics).

5.1 Contingut d'una resposta SAML

Les peticions a la PDS de tipus SAML 2.0 sempre han de ser de tipus GET. Per tant, el que fem és configurar credencials *Basic Auth* per cada client adherit al corresponent *Realm*. No es compartiran credencials entre clients i cadascú tindrà les seves, d'aquesta manera la iniciativa destí pot determinar qui ha adreçat la petició de tiquet.

Aquest tiquet SAML és un fitxer XML amb un node arrel de tipus `<samlp:Response>`:

```
<samlp:Response xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
...xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
...Destination="https://preproduccio.pds.hes.catsalut.gencat.cat/auth/realms/HC3-
SAML/protocol/saml/clients/hc3-saml-salut"
...ID="ID_d6a30fa7-ba1b-45d8-8b5f-a5da709fdf24"
...IssueInstant="2025-10-15T07:51:59.074Z" Version="2.0">
```

El contingut d'aquest tiquet SAML consta de 3 parts. Primer l'originari identificatiu de la PDS:

```
<saml:Issuer>https://preproduccio.pds.hes.catsalut.gencat.cat/auth/realms/HC3-
SAML</saml:Issuer>
```

A continuació l'estat del tiquet generat:

```
<samlp:Status>
      <samlp:StatusCode Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
</samlp:Status>
```

I per últim l'asserció SAML:

```
<saml:Assertion xmlns="urn:oasis:names:tc:SAML:2.0:assertion"
   ID="ID_bf35d90a-34f4-4774-9f5e-9edf673e6b61"
   IssueInstant="2025-10-15T07:51:59.074Z"
   Version="2.0">
```

Dins aquesta asserció hi la signatura del contingut tiquet, en un camp `<dsig:Signature>`, i la clau pública del certificat utilitzat per part de la PDS. Com hem indicat anteriorment, hi ha una relació de confiança entre els sistemes destí com HC3 o SIRE i la clau privada del certificat de la PDS, de manera que es pugui determinar la validesa i autenticitat del tiquet.

Per últim, el *payload* d'atributs del tiquet (l'equivalent al *payload* de paràmetres del token JWT amb OIDC) ve informat en una estructura `<saml:AttributeStatement>` de l'XML.

Existeix la possibilitat de definir aquests valors que ha d'incloure els tiquets SAML. En aquest cas, com la crida per obtenir el token de PDS ha de ser de tipus GET, només podem ampliar aquesta informació a través de *Requests Headers*.

Com el balancejador de tràfic HTTPS que exposa els serveis de PDS fa un tractament *case-insensitive* del nom d'aquests *headers*, hem de fer un *workaround* per fixar aquests atributs alternant majúscules i minúscules. El format de la capçalera que cal enviar és

- Header Name: SAML_**NomAtribut**
- Header Value: SAML_**NomAtribut####ValorAtribut**

D'aquesta manera, la PDS genera el parell clau-valor del **SAML Attribute** en base al valor de la capçalera:

```
<saml:Attribute
      Name="NomAtribut"
      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
      <saml:AttributeValue
            xmlns:xs=http://www.w3.org/2001/XMLSchema
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xsi:type="xs:string">ValorAtribut</saml:AttributeValue>
</saml:Attribute>
```

En un futur, si resolem aquest inconvenient de les capçaleres *case-insensitive*, potser no serà necessari fer aquest tractament amb els separadors "####". Tanmateix, ho mantindrem dins de la PDS com retrocompatible pels sistemes que ja facin les peticions d'aquesta manera.

Exemple de la petició:

```
curl --request GET   --url
'https://integracio.pds.hes.catsalut.intranet.gencat.cat/auth/realms/HC3-
SAML/protocol/saml/clients/hc3-saml-salut' \
  --header 'Authorization: Basic < usuari:contrasenya codificat a base64>' \
  --header 'SAML_FirstFamilyName: 'SAML_FirstFamilyName####Nombre_Apellidos' \
  --header 'SAML_AccessType: 'SAML_AccessType ####TipusAccess' \
 ...
```

El resultat és un document **SAML Response** codificat en Base64. En cas de descodificar el contingut podem veure l'estructura XML amb alguns dels camps més rellevants:

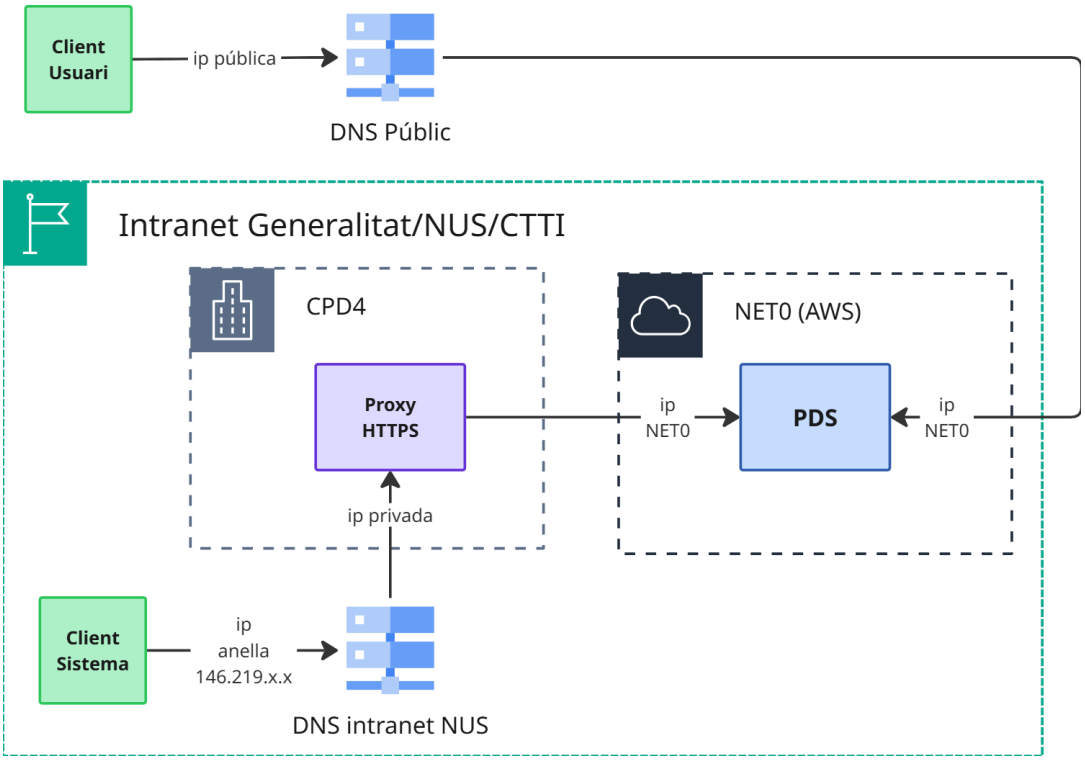
`<samlp:Response>`	Root principal del document XML
`<saml:Issuer>`	*Realm* de la PDS que ha generat el tiquet SAML
`<saml:Assertion>`	Contingut del tiquet SAML
`<dsig:Signature>`	Signatura utilitzada per al tiquet, i clau pública
`<saml:Subject>`	Credencials facilitades per identificar al sol·licitant del tiquet SAML
`<saml:Conditions>`	Validesa del tiquet (configurable a PDS)
`<saml:AttributeStatement>`	Llista d'atributs SAML inclosos en el tiquet, en base als valors enviats per les capçaleres de petició.

6. Publicació i connectivitat dels dominis de la PDS

La Plataforma de Seguretat de Salut està desplegada en el núvol públic d'AWS dins la NET0 de CTTI des de Juliol de 2025.

6.1 Adreçament actual (CPD4)

Malgrat això, en el moment de publicar aquest manual d'adhesions (Abril 2026) la Plataforma de Seguretat de Salut encara disposa d'una **resolució cap adreçament privat d'Anella Sanitària de CPD4** (146.219.x.x) pels seus dominis dels entorns PRE i PRO si la consulta es fa al DNS intranet de NUS, o un **adreçament públic** si la consulta es fa al DNS públic. En l'entorn INT, la resolució de domini és únicament amb adreçament de tipus privat



Domini PDS	Resolució DNS intranet	Resolució DNS públic
integracio.pds.hes.catsalut.intranet.gencat.cat	**146.219.255.134** Adreçament privat Anella Sanitària CPD4 Regla de FW pel port 443	-
preproduccio.pds.hes.catsalut.gencat.cat	**146.219.253.140** Adreçament privat Anella Sanitària CPD4 Regla de FW pel port 443	**34.254.43.45** Adreçament públic NUS Sense regla de FW
pds.hes.catsalut.gencat.cat	**146.219.252.134** Adreçament privat Anella Sanitària CPD4 Regla de FW pel port 443	**83.247.135.246** Adreçament públic NUS Sense regla de FW

Un client de tipus **usuari**, des del seu navegador web, segurament utilitza el DNS públic per resoldre els dominis de la PDS; i no requereix cap connectivitat addicional.

Un client de tipus **sistema**, si està hostatjat dins un dels CPD de la Generalitat, probablement utilitzarà el DNS intranet, i per aquest motiu, haurà de tramitar la corresponent regla de FW al destí pel port 443 (HTTPS).

6.2 Adreçament futur (NET0)

Està previst eliminar aquest *proxy* a CPD4 en els propers mesos, però encara cal tancar amb NUS l'estratègia: mantenir adreçament d'Anella Sanitària per totes les resolucions de DNS, o fer-la només pels sistemes que realment estan dins l'Anella (i tenen uns DNS específics).

Facilitem l'adreçament privat del núvol d'AWS a la NET0 on està ja disponible també la PDS, per tenir en compte les futures regles de connectivitat que caldrà demanar un cop es determini aquesta transició de resolució de dominis.

Domini PDS	Futura resolució DNS intranet	Resolució DNS públic
integracio.pds.hes.catsalut.intranet.gencat.cat	Rang 172.22.26.128/26 **a determinar** Adreçament privat NET0 de NUS Regla de FW pel port 443	-
preproduccio.pds.hes.catsalut.gencat.cat	Rang 172.22.26.128/26 **a determinar** Adreçament privat NET0 de NUS Regla de FW pel port 443	**A determinar** Adreçament públic NUS Sense regla de FW
pds.hes.catsalut.gencat.cat	Rang 172.22.26.192/26 **a determinar** Adreçament privat NET0 de NUS Regla de FW pel port 443	**A determinar** Adreçament públic NUS Sense regla de FW

Però, mentrestant, no es necessari encara aplicar cap d'aquestes regles. Actualitzarem aquesta informació mitjançant una nova distribució d'aquest manual d'adhesions de la PDS.