ROMÂNIA

(19) OFICIUL DE STAT
PENTRU INVENŢII ŞI MĂRCI
Bucureşti

(11) **RO 138236 A0**

(51) Int.Cl.
H04L 65/00 (2022.01),
H04L 9/00 (2006.01)

(12) # CERERE DE BREVET DE INVENŢIE

(21) Nr. cerere: **a 2023 00887**

(22) Data de depozit: **29/12/2023**

(41) Data publicării cererii:
30/05/2024 BOPI nr. **5/2024**

(71) Solicitant:
• **CHRISTIAN GAVRILA S.R.L.**,
BD.VICTORIEI, NR.12, ET.II, CLĂDIREA INFOSTAR, BRAŞOV, BV, RO

(72) Inventatori:
• **GAVRILA CHRISTIAN**,
STR. DEALUL SPIRII, NR.37A, BRAŞOV, BV, RO

(54) # SISTEM SECURIZARE CLOUD HIBRID

(57) **Rezumat:**

Invenţia se referă la un sistem de securizare cloud hibrid. Sistemul, conform invenţiei, defineşte o listă de proprietari, fiecare având o listă de comunităţi, care au la rândul lor câte o listă de utilizatori şi o listă de roluri, precum şi corespondenţa multiplă utilizator-rol, în care proprietarii au asociată o listă de aplicaţii web pe care le deţin, fiecare având asociat un client_id folosit de protocolul Oauth şi o cheie publică dintr-o pereche de chei criptografice, cheia privată fiind cunoscută exclusiv de proprietarul aplicaţiei, în care, pe baza acestor definiţii sunt definite un procedeu de autorizare a accesului la aplicaţia web, cu corespondenţă între identitatea globală şi cea locală şi setarea unor variabile de mediu care configurează aplicaţia, un procedeu de autorizare a accesului programatic de la o aplicaţie la alta şi un procedeu pentru crearea de invitaţii pentru utilizatorii noi.

Revendicări: 10
Figuri: 2

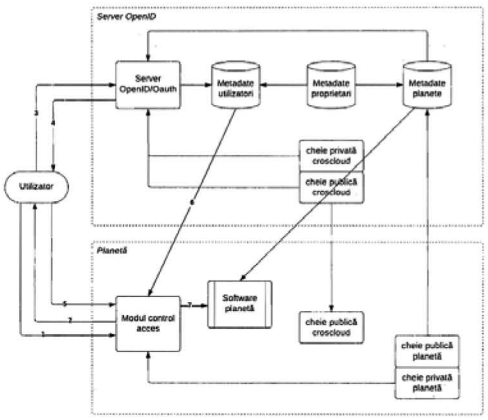

Fig. 1

RO 138236 A0

Sistem Securizare Cloud Hibrid Descriere

OFICIUL DE STAT PENTRU INVENȚII ȘI MĂRCI
Cerere de brevet de invenție
Nr. a 2023 00887
Data depozit 29-12-2023

Stadiul tehnicii

Integrarea aplicațiilor folosite de diverse organizații, dezvoltate intern sau produse software standard implementate, reprezintă o problemă majoră cu pondere mare în costul transformării digitale a organizației. Gestiunea identității utilizatorilor și autorizarea accesului acestora este o problemă de bază, pentru care s-au descoperit diverse soluții. Între ele se remarcă protocoalele Oauth, pentru delegarea autorizării accesului către un server de autorizare, și OpenID Connect pentru gestionarea identității peste un protocol Oauth. Aceste soluții descriu procedeele de bază pentru autorizare, detaliile fiind generice și lăsate la latitudinea celor care decid să le folosească.

Problema tehnică

Dezvoltarea infrastructurilor de cloud publice a permis ca serverele de aplicații să fie rulate pe servere publice. Aceste servere pot fi private, ale unui anumit proprietar, dar pot exista si aplicații publice deținute de furnizori de servicii informatice cu aplicabilitate generală. În paralel există aplicațiile moștenite, care rulează pe servere din rețelele locale și nu sunt accesibile din Internet.

Gestionarea identității utilizatorilor este tratată în majoritatea aplicațiilor software, intern sau prin integrări du alte servicii. Schimbarea metodei de gestionare a identității implică o rescriere majoră a programelor cu care sunt realizate aceste aplicații, care poate chiar să nu fie posibilă dacă nu sunt disponibile sursele sau modificarea nu este permisă de licența de utilizare. Soluțiile de single sign on, cum sunt cele dezvoltate pe baza protocolului OpenID Connect, permit o identitate unică dar nu cuprind soluții pentru integrarea unor sisteme de gestiune a identității preexistente.

Descrierea invenției

Invenția constă într-un grup de procedee interdependente pentru definirea și organizarea identităților, a aplicațiilor și a drepturilor de acces, pentru verificarea identității și autorizarea accesului și asigurarea corespondenței datelor între aplicații, pentru crearea de utilizatori pe bază de invitație și pentru autorizarea apelurilor programatice de la o aplicație către alta.

Procedeul de verificare a identității și autorizare a accesului este prezentat în figura 1 și descris în continuare. El se bazează pe un server care răspunde la protocolul OpenID Connect într-un mod specific corelat cu restul procedeelor din invenție, care are disponibile o serie de interfețe specifice prezentei invenții. Acest server poate fi un singur program tip server Web rulând pe un calculator accesibil public în Internet, dar poate fi și un grup de calculatoare organizate într-un cluster care rulează în paralel servere de web și un program de balansare automată a încărcării, conectate la o bază de date.

În sistem există unul sau mai multe servere web care rulează o aplicaţie, numite în continuare planete. O planetă poate fi un singur program sau de asemenea un cluster şi un server de baze de date.

Când utilizatorul doreşte să acceseze o aplicaţie aflată pe o planetă (pasul 1 în figura 1), un modul de control acces configurat să filtreze toate cererile HTTP către planetă declanşează un proces de autorizare conform protocolului Oauth cu o autorizare de tip code grant, redirecţionând utilizatorul către serverul OpenID (pasul 2 în figura 1), simultan cu crearea unei sesiuni la nivel de planetă şi setarea unui HTTP cookie în răspunsul cu redirecţionarea.

Serverul OpenId verifică identitatea utilizatorului prin metodele stabilite la definirea acestuia în sistem şi prin dialog cu utilizatorul (pasul 3 din figura 1). Dacă identitatea şi prezenţa utilizatorului sunt confirmate, serverul emite un cod de autorizare şi deschide o sesiune utilizator identificată printr-un cod criptografic sigur, stocat într-un HTTP cookie setat în răspunsul cu codul de autorizare Oauth (pasul 4 din figura 1).

Utilizatorul accesează apoi planeta prezentând acest cod de autorizare (pasul 5 din figura 1) modulului de control acces de pe planetă. Acesta verifică autenticitatea codului luând legătura cu serverul OpenID (pasul 6 figura 1), unde la rândul lui prezintă un token JWT semnat cu cheia privată a planetei şi obţine un alt token JWT cu datele asociate utilizatorului, semnat de către serverul OpenID şi autentificat de modulul de control acces folosind cheia publică a serverului OpenID.

Modulul de control acces determină identitatea locală corespunzătoare identificatorului utilizatorului. Dacă nu există o identitate locală, verifică dacă exista o invitaţie de acces la planetă conform procedeului de creare de utilizatori descris mai jos şi dacă găseşte una face asocierea între identitatea locală şi identificatorul de utilizator primit de la OpenID. Dacă nu există nici invitaţie, creează o nouă identitate locală pe baza datelor referitoare la utilizator primite de la serverul de OpenID. Identitatea locală poate fi preexistentă, dintr-o aplicaţie moştenită. Aplicaţia are libertatea deplină de a gestiona identităţile locale, modulul de control acces realizând corespondenţa identitate locală – identitate globală.

După determinarea identităţii locale, modulul de control acces verifică dacă există o corespondenţă între rolurile din datele referitoare la utilizator primite de la serverul OpenID şi drepturile de acces în aplicaţia web, actualizând dacă este cazul drepturile de acces pentru a corespunde rolurilor alocate global.

După determinarea identităţii locale, modulul de control acces setează apoi în aplicaţia existentă pe planetă utilizatorul conform identităţii locale deja determinate şi setează variabilele de mediu cu valorile primite de la serverul de OpenID.

Procedeul de definire şi organizare a aplicaţiilor, identităţilor şi drepturilor de acces se bazează pe o serie de concepte şi relaţiile dintre ele, prezentate în figura 2 şi descrise în continuare.

În sistem sunt definiţi proprietari (owner) care sunt deţinătorii unor identităţi şi ale unor aplicaţii.

Fiecare proprietar are asociată o listă de planete, care sunt aplicaţii web rulând pe un server public sau privat, în cloud sau în reţeaua locală. Planeta poate reprezenta un singur program tip server Web care rulează pe un server, dar poate fi şi un grup de programe şi servere care funcţionează integrat pentru a servi pagini Web. De exemplu un cluster care rulează un server de aplicaţie şi o server de baze de date folosit de acestea. Fiecare planetă are asociate:

- un URL care este baza pentru cererile HTTP prin care acel program produce pagini web
- un client_id în sensul protocolului Oauth
- cheia publică dintr-o pereche de chei criptografice

Fiecare proprietar are asociată o listă de comunităţi, identificate printr-un cod alfanumeric unic în cadrul proprietarului. Fiecare comunitate conţine o listă de utilizatori, identificaţi atât printr-un identificator unic la nivelul serverului OpenID cât şi printr-un nume (cod alfanumeric) unic în cadrul comunităţii. Identificatorul unic nu este cunoscut utilizatorului însă numele poate fi cunoscut şi folosit de acesta pentru a se identifica.

La nivel de proprietar, comunitate sau utilizator se vor putea specifica metodele de identificare pentru care poate opta utilizatorul precum şi metoda de autentificare cu care utilizatorul certifică la cererea serverului OpenID că este prezent. Metodele de identificare şi autentificare nu fac obiectul prezentei invenţii, fiind numai necesar să existe. Minimal identificarea se poate face prin numele utilizatorului şi autentificarea prin prezentarea unei parole. Serverul de OpenID trebuie să asigure interfaţa utilizator pentru identificare şi autentificare.

Procedeul de creare de conturi de utilizator pe bază de invitaţie începe pe planetă, prin crearea unei identităţi locale candidat şi a unui identificator local al invitaţiei. Acestea se transmit către serverul OpenID într-un token JWT semnat cu cheia privată a planetei, care include codul comunităţii în care este invitat utilizatorul şi pagina web spre care să fie direcţionat utilizatorul după acceptarea invitaţiei. Serverul OpenID înregistrează invitaţia şi returnează un URL cu invitaţia care va fi transmisă de către planetă către utilizatorul de invitat. Acesta se conectează la acest URL care indică spre o pagină specifică de pe serverul OpenID unde utilizatorul va introduce datele necesare creării unui utilizator în comunitatea specificată şi va opta pentru un sistem de autentificare şi introduce eventuale date necesare (parolă, amprentă, etc). Odată introduse datele, serverul OpenID va redirecţiona utilizatorul spre un punct de acces din modulul de control acces, care va valida invitaţia pentru a nu fi

refolosită şi va asocia identitatea locală candidat cu utilizatorul, apoi va redirecţiona utilizatorul către pagina specificată la crearea invitaţiei.

Procedeul de autorizare a apelurilor programatice dintr-o aplicaţie web către o altă aplicaţie web constă în crearea unui token JWT care include client_id al planetei sursă şi opţional client_id al planetei destinaţie, semnat cu cheia privată a planetei, care va fi inclus în headerul HTTP Authorization al apelului către aplicaţia web corespondentă (planeta destinaţie) ca token la purtător. Pe planeta destinaţie modulul de control acces va verifica tokenul al purtător folosind cheia publică a planetei sursă, obţinută pe baza client_id sursă de la serverul OpenId prin punctul de acces dedicat acestui scop. Dacă există şi un client_id destinaţie în tokenul la purtător trebuie verificat să corespundă cu planeta destinaţie.

Avantajele invenţiei

Grupul de procedee descrise permite integrarea unor aplicaţii Web existente, care pot avea un sistem de gestiune a identităţii preexistent. Aplicaţiile pot avea diverşi proprietari, care decid cine poate le accesa şi pot specifica în sistem aceste reguli pe care sistemul să le aplice automat. Înregistrarea aplicaţiilor se face printr-un o singură pereche de chei criptografice, cu o cheie privată cunoscută exclusiv de deţinătorul aplicaţiei.

Procedeele nu necesită ca aplicaţia web să fie pe un server public ci poate fi un server privat, care nu este accesibil din Internet, permiţând realizarea de aplicaţii cloud hibride, care integrează aplicaţii web publice cu aplicaţii private, instalate în reţele locale.

Procedeele se bazează pe protocoalele larg adoptate Oauth şi OpenID Connect şi includ opţiuni specifice pentru aceste protocoale care permit simplificarea a configurării şi administrării sistemului, integrarea unor aplicaţii moştenite care aveau propriile sisteme de gestiune a identităţii şi securizarea simultană a apelurilor programatice directe între aplicaţii.

Exemplu de realizare

O realizare a procedeelor descrise se poate scrie în Java folosind servere de aplicaţii J2EE. Pentru funcţionalităţile de bază inclusiv cele pentru criptografie şi JWT există biblioteci Java disponibile. Pe serverul OpenID se pot scrie în Java funcţii de acces la o bază de date cu proprietarii, comunităţile, utilizatorii, rolurile şi planetele. Se pot scrie apoi servlets specifici pentru punctele de apel specificate în protocolul OpenID care să răspundă conform procedeelor descrise în invenţie, precum şi punctele de acces specifice descrie în invenţie pentru obţinerea cheii publice a serverului OpenID, a cheii publice a unei planete pe baza client_id, a URL pentru o invitaţie.

Un exemplu JSON cu datele obținute din punctul de acces userinfo:


```
{
        "user_ID": "1",
        "client_id": "16373833354",
        "username": "user",
        "complete_name": "Utilizator Test",
        "email": "test@crisoft.ro",
        "phone": "+40-744-555555",
        "owner": "CRISOFT",
        "community": "DEV"
        "env" : {
                "theme": "crosweb_dark",
                "language":"RO"
        },
        "roles": ["management", "sales"]
}
```


Bibliografie

1. The OAuth 2.0 Authorization Framework - Internet Engineering Task Force – RFC 6749 - https://www.ietf.org/rfc/rfc6750.txt
2. OAuth 2.0 Authorization Server Metadata - Internet Engineering Task Force – RFC 8414 - https://www.ietf.org/rfc/rfc8414.txt
3. OpenID Connect Core 1.0 – OpenID Foundation - https://openid.net/specs/openid-connect-core-1_0.html
4. JSON Web Token (JWT) Profile for OAuth 2.0 Client Authentication and Authorization Grants - Internet Engineering Task Force – RFC 7523 - https://www.ietf.org/rfc/rfc7523.txt
5. Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1 - Internet Engineering Task Force – RFC 3447 - https://www.ietf.org/rfc/rfc3447.txt
6. JSON Web Token (JWT) - Internet Engineering Task Force – RFC 7519 - https://www.ietf.org/rfc/rfc7519.txt

Revendicări

1. Procedeu pentru definirea unui sistem de gestiune a identităţii caracterizat prin aceea că asigură definirea unei liste de proprietari, fiecare având asociată o listă de comunităţi, fiecare comunitate cuprinzând o listă de utilizatori, o listă de roluri şi o corespondenţă multiplă între utilizatori şi roluri, simultan cu asocierea la fiecare proprietar a unei liste de servere care rulează aplicaţii Web.
2. Particular revendicării 1, identificarea proprietarului se face printr-un cod alfanumeric unic, identificarea comunităţii se face printr-un cod alfanumeric unic în cadrul proprietarului, identificarea utilizatorului se face printr-un identificator unic şi simultan şi printr-un nume unic în cadrul comunităţii, iar identificarea rolurilor se face printr-un cod alfanumeric unic în cadrul comunităţii.
3. Particular revendicării 1, asocierea la fiecare planetă a client_id în sensul protocolului Oauth, a URL rădăcină aplicaţia web care rulează pe acea planetă şi a cheii publice dintr-o pereche de chei criptografice la care cheia privată este cunoscută numai aplicaţiei care rulează pe planeta.
4. Particular revendicării 1, asocierea la fiecare utilizator a unui identificator unic, a comunităţii din care face parte şi a proprietarului acesteia şi a unui set de variabile de mediu în forma unor perechi cod-valoare.
5. Particular revendicării 1, asocierea la fiecare planetă a unei liste de drepturi de acces la nivel de proprietar, comunitate, rol sau utilizator.
6. Procedeu pentru funcţionarea unui server pentru protocolul OpenID Connect caracterizat prin folosirea unui sistem de gestiune a identităţii conform procedeului de la revendicarea 1, care realizează autentificarea aplicaţiilor client folosind client_id şi cheia publică definite conform procedeului de la revendicarea 2, extins cu un punct de acces programatic HTTP prin care se pot obţine datele unei planete, respectiv client_id, url şi cheia publică.
7. Particular revendicării 5, transmiterea o aplicaţia web prin cadrul apelului userinfo din cadrul protocolului OpenID Connect a datelor asociate utilizatorului conform procedeului descris la revendicarea 1, inclusiv lista de roluri asociate utilizatorului şi lista variabilelor de mediu şi a valorilor acestora.
8. Procedeu pentru autorizarea accesului la o aplicaţie web caracterizat prin folosirea unui modul de control acces care filtrează toate apelurile HTTP către o aplicaţie Web şi autorizează accesul conform protocoalelor OpenID Connect şi Oauth, configurate specific procedeelor de la revendicările 1 şi cele dependente şi revendicării 5 cele dependente, cu autentificare printr-un token JWT semnat cu cheia privată a planetei şi autentificarea valorilor primite de la serverul de OpenId folosind cheia publică a serverului OpenID.
9. Particular revendicării 8, determinarea identităţii locale specifice aplicaţiei Web se face de către modul de control acces pe baza unei tabele de corespondenţă între identificatorul utilizatorului şi identitatea locală definită în aplicaţie, cu crearea automată a identităţii locale pe baza informaţiilor primite de la serverului de OpenID şi a unei eventuale invitaţii existentă în modulul de control acces.

10. Particular revendicării 8, autorizarea apelului programatic pe baza unui token JWT la purtător, semnat folosind cheia privată a aplicației sursă și autentifica cu cheia publică a aplicației sursă obținută de la serverul OpenID, semnată cu cheia privată a acestuia.

Desene explicative

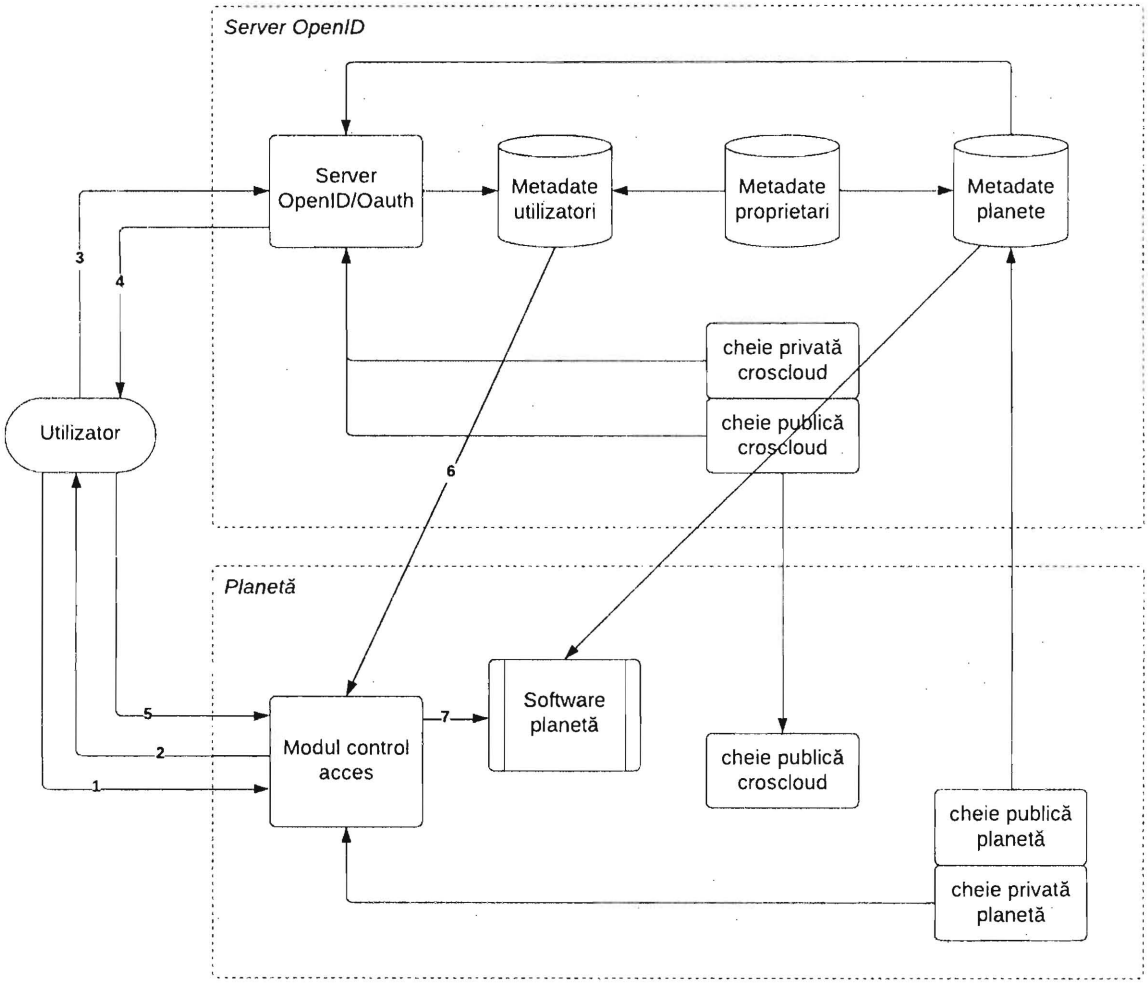

Figura 1

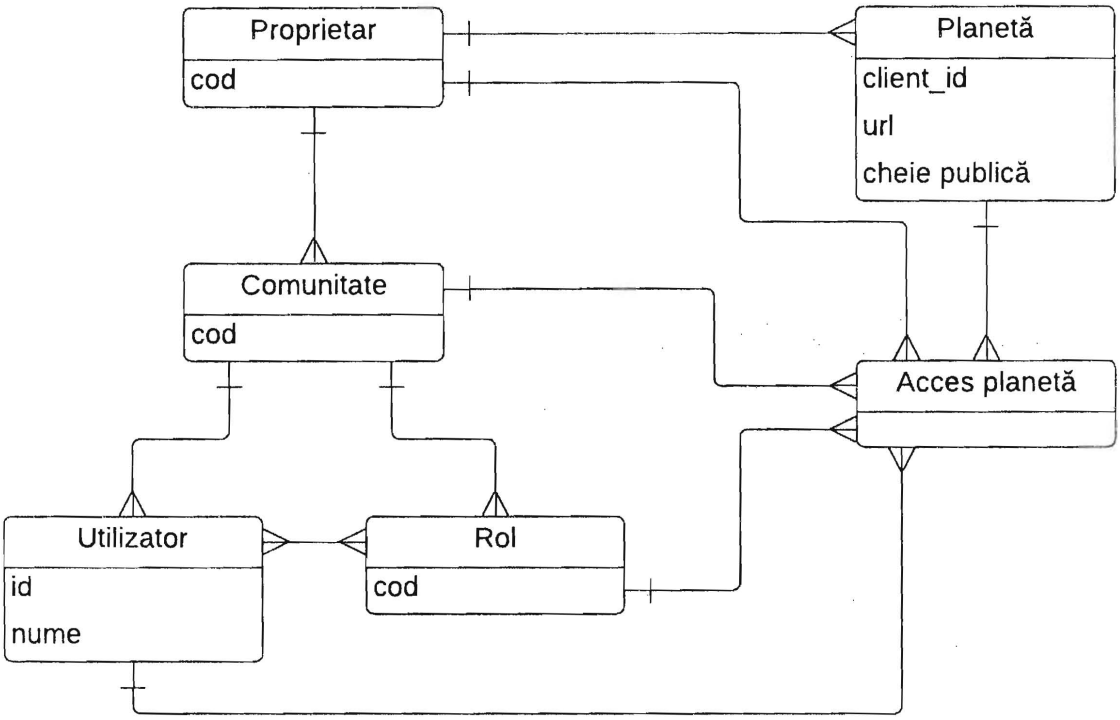
Proprietar
cod
Planetă
client_id
url
cheie publică
Comunitate
cod
Acces planetă
Utilizator
id
nume
Rol
cod

Figura 2