**Šta je Single Sign On (SSO) ?**

Single Sign On je metoda koja omogućava korisnicima da se autentikuju preko vise aplikacija ili vebsajta, samo jednom koristeci iste kredencijale (na primer unosom korisnickog imena I lozinke).

Single Sign On autentikacija je u sada postaje tražena više nego ikada. U današnje vreme skoro svaka aplikacija ili vebsajt zahteva neku formu autentikacije da bi imala pristup zaštićenim resursima. Kako broj aplikacija I servisa raste tako potreba za centralizovanim sistemom prijave postaje neophodna.

Koncept centralizovanog sistema prijave je poznat kao Ujedinjeni sistem identiteta (Federated Identity).

Ujedinjeni sistemi identiteta rešavaju nekoliko problema:

* Autentifikaciju
* Pravo pristupa resursima
* Razmena korisnickih podataka
* Upravljanje korisnicima

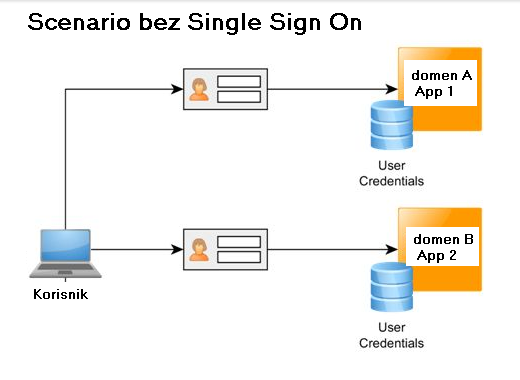
Aspekt Autentifikacije bavi se potvrđivanjem i validiranjem korisničkih podataka (kredencijala) i utvrđivanjem identiteta korisnika.

Pravo pristupa resursima kao sto ime kaze se bavi restrikcijama pristupa odredjene aplikacije odredjenim resursima ( na primer da li aplikacija ili korisnik ima pravo da pristupi nekom resursu)

Razmena korisničkih podataka bavi se razmenom podataka kroz različite sisteme za upravljanje korisnicima.Na primer polja kao sto su “Pravo ime“ mogu biti prisutna u više sistema.Ujedinjeni sistem identiteta sprečava dupliciranje podataka povezivanjem povezanih atributa (linking)

Upravljanje korisnicima je povezano sa samom administracijom (registracija novog korisnika, Brisanje ili Ažuriranje postojeceg korisnika)

Single Sign on je strogo povezan autentifikacionom delu ujedinjenog sistema identiteta.

Kada se korisnik jednom autentikuje unosom kredencijala, preko ujedinjenog sistema identiteta se podaci o samom identitetu dele izmedju podsistema koji zahtevaju podatke.

U razvoju veb aplikacija pre ili kasnije se javlja potreba da korisnik koji se prijavio preko domena A bude automatski prijavljen i na aplikaciji sa domenom B (to je i svrha Single Sign On).

Bez Single Sign On mehanizma korisnik bi morao da se autentikuje unesenjem kredencijala na svakoj aplikaciji (domenu) kojoj pristupa , bez obzira sto te aplikacije cine jednu vecu logicku celinu.

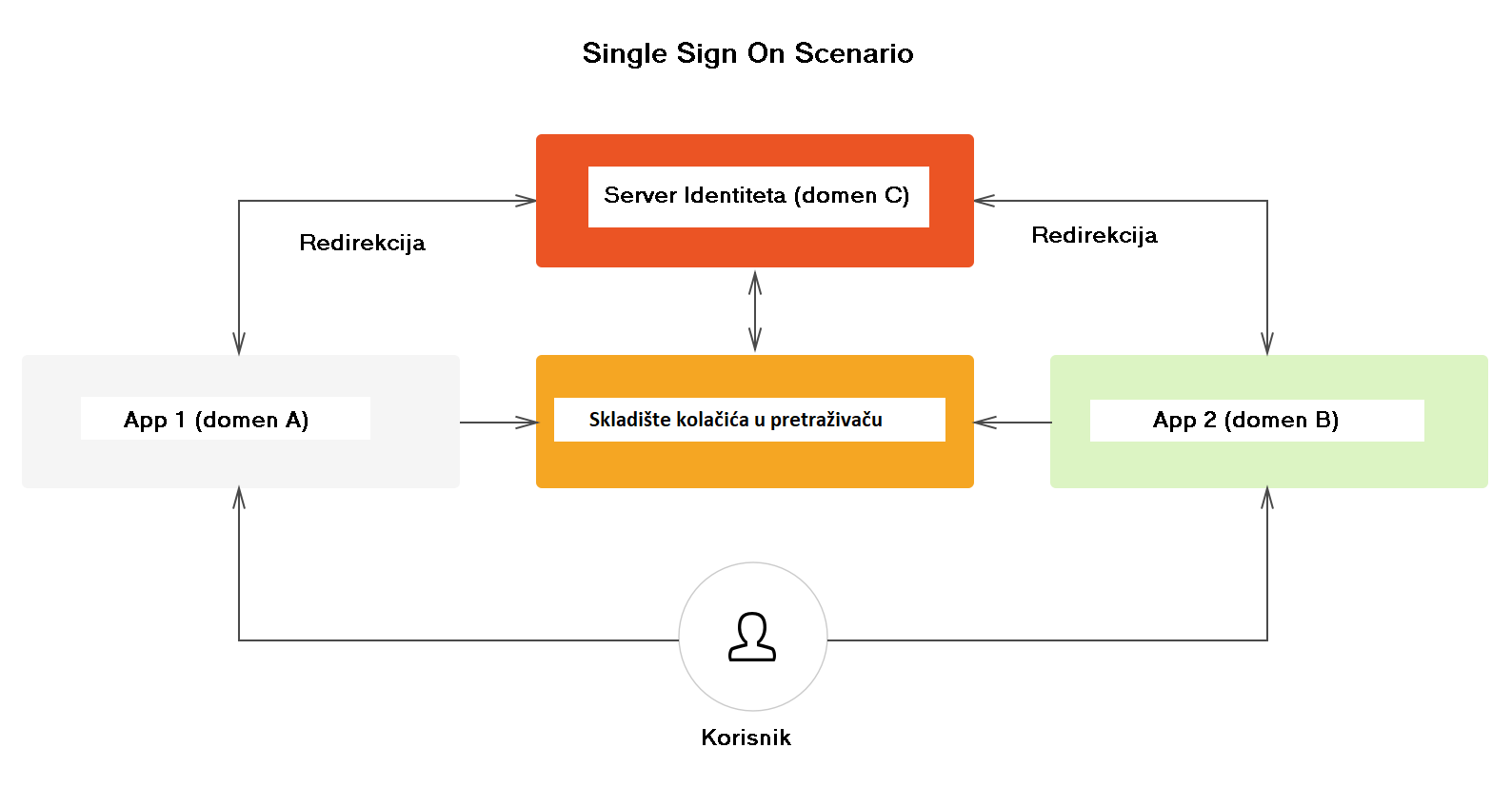
Ocigledno resenje ovakvog problema jeste da informacije sesije(identiteta) dele izmedju različitih domena. Međutim iz sigurnosnih razloga veb pretrazivaci forsiraju politiku zvanu Ista politika porekla.

Ova politika definise da kolačići (ili drugi podaci) mogu da dostupni samo njihovom vlasniku.

Drugim rečima domen A ne moze da pristupi kolačićima domena B i obrnuto.

Postoje različiti Single Sign On protokoli koji omogućavaju deljenje informacija sesije (kolačića) , ali sustinski koncept je isti. Postojanje Ujedinjenog Sistema Identiteta (na posebnom domenu C) preko koga je izvrsena autentikacija i preko koga se sesija deli sa ostalim domenima na neki nacin.

Kada korisnik ode na aplikaciju sa domenom A koja zahteva autentikaciju,automatski biva redirektovan na domen C ( Ujedinjeni sistem identiteta) . Kako je korisnik vec prosao proces autentikacije na domenu C , na primer preko domena B , korisnik ce biti vracen na originalni domen A sa potrebnim informacijama identiteta bez potrebe za ponovnim unosom kredencijala.



Trenutno najpoznatije protokoli za Single Sign On su: OpenID Connect, Facebook Connect, Security Assertion Markup Language (SAML), Microsoft Account.

Za dalji rad izabracemo implementaciju OpenID connect (OIDC) protokola za postizanje Single Sign On.

Nakon uspesne autentikacije za dozvolu prava pristupa odredjenim resursima koristiće se implementacija Oauth2 protokola (OAuth 2.0 is the industry-standard protocol for authorization)

**Prednosti Single Sign On**

Korisnicima je glavna prednost što više ne moraju da pamte na destetine lozinki.Ovo čak oslobađa korisnika od upravljanja pojedinacnih lozinki na svakoj aplikaciji ponaosob, zbog čega se SSO prijave takođe smatraju i menadzerima lozinki.Smatrajući da je ovo tako zgodna alternativa i zato sto štedi vreme i gnjavažu, SSO rešenja se koriste kako u privatnom tako i u profesionalnom kontekstu.

Kompanije koje primenjuju SSO u svom radu očekuju veću produktivnost svijih zaposlenih i manje poziva službe za pomoc zbog zaboravljenih lozinki.Zbog toga IT ima manje posla i manje košta.Istovremeno IT osoblju olakšava dodeljivanje novih naloga novim zaposlenima ili brisanje pristupa bivšim zaposlenima.

Takođe lozinku treba uneti samo na jednom interfejsu, a to smanjuje potencijal za napade (pecanje) u sistemu.Pod ovim okolnostima kompanija moze sebi priuštiti da usresredi rad na zastiti samo na jednom mestu.Kao sto su na primer SSL sertifikati.

Povećava brzinu tamo gde je napotrebnija.U okruženjima kao sto su bolnice,odbrambena industrija I hitne službe, gde veliki broj ljudi I odeljenja zahteva brz I nesmetan pristup istim aplikacijama.

**Problemi Single Sign On**

Sa druge strane postoje određeni problemi i napori u primeni, kao i slabosti samog SSO.

U osnovi mogu se koristiti samo one usluge koje podržava odgovarajući sistem SSO.Ako sistem zakaže pristup povezanim aplikacijama biće ograničen ili nemoguć. U ovom slučaju SSO postaje takozvano usko grlo.

Takođe treba uzeti u obzir i sigurnost SSO prijave.Ako korisnik napusti svoju aplikaciju a da nije izvršio odjavljivanje, treća strana bi teoretski mogla da iskoristi vreme do automatske odjave (Single Sign Out), kako bi iskoristila pristup odobren putem SSO prijave.Takođe je problematicno ako “glavna lozinka“ za SSO interfejs padne u pogrešne ruke.To napadaču daje trenutni pristup svim povezanim uslugama.

Postoji zabrinutost i zbog GDPR (General Data Protection Regulation) a, u kojem su zahtevi za zaštitu ličnih podataka pooštreni u celoj Evropi od 25. Maja 2018. godine. Sada je potrebno dobiti izričit odgovor od korisnika da bi mogli da koriste SSO prijavu.U prošlosti je bio potreban i ovaj sporazum, ali su zakonitosti toliko promenjene da je situacija sada mnogo stroža.

S obzirom na ove potencijalne rizike, potrebno je obratiti posebnu pažnju na sigurnost podataka koji se čuvaju na strani servera.Ima smisla povećati sigurnost SSO funkcija koristeci na primer dvofaktorsku potvrdu identiteta (two-factor authentication) ili druga rešenja kao sto su pametne kartice (smart cards) ili tokeni.

Ako neko istraži SSO prijavu na Internetu, pronalazi relativno malo negativnih informacija o postupku višestruke potvrde identiteta.Umesto toga, godinama se to tretira kao sasvim prikladno otkriće za digitalno radno mesto, što se tiče udobnosti i sigurnosti podataka.

US-American Cloud Access Security Broker (CASB) hvali globalnu upotrebu usluga u oblaku u kompanijama, ali istovremeno kritikuje relativno nisku upotrebu metoda za za višestruku potvrdu identiteta (SSO prijava)

**OIDC i OAuth2**

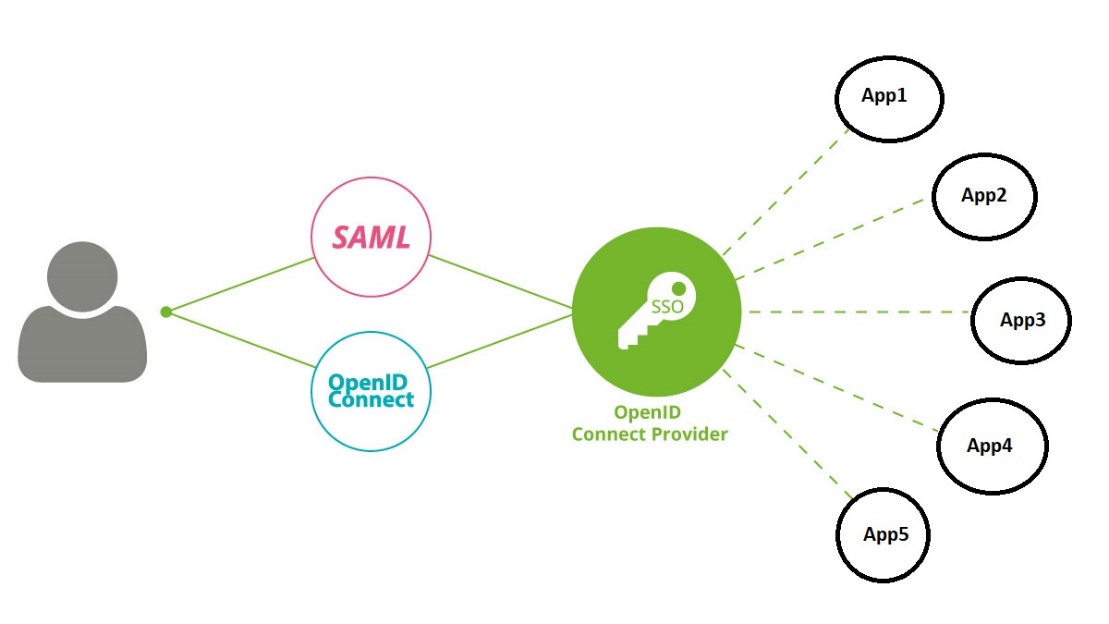
U predhodnom odeljku smo objasnili koncept Single Sign On (SSO) -a i rekli smo da postoje različiti protokoli za implementaciju SSO-a (OpenIdConnect, Security Assertion Markup Language).Kao i da postoji potreba za centralizovanim sistemom prijave, odnosno Serverom Identiteta (IdP- Identity Service Provider).

Takođe smo rekli da ćemo za dalji rad izabrati implementaciju OIDC i OAuth2 protokola.Pre nego sto uđemo u detalje kako rade ovi protokoli, ukratko cemo objasniti razlike izmedju njih.

Suštinska razlika izmedju ovih protokola je ta sto se OAuth2 koristi za kontrolu prava pristupa zaštićenim resursima (neka aplikacija ili web Api ili datoteka), dok su OIDC i SAML industrijski standardi za potvrdu identeta odnosno autentifikaciju.OAuth2 se koristi u potpuno drugačijim situacijama od druga dva standarda, a može se koristiti istovremeno u kombinaciji sa OIDC-om i SAML-om, što uglavnom i jeste slučaj.

OAuth2 omogućava aplikacijama pristup zaštićenim resursima korisćenjem tokena zvanim Access Tokens.

OIDC je izgradjen na vrhu OAuth2 protokola i koristi dodatni token u JWT formatu,koji se naziva ID Token.Posebno je fokusiran na autentikaciju korisnika i široko se koristi za omogućavanje korisničkih prijava na veb sajtovima i u mobilnim aplikacijama.S obzirom da je OIDC nadogradnja OAuth2 protokola, predstavlja potpuno razvijen protokol za autentikaciju i dodelu prava pristupa (autorizaciju), koji intenzivno koristi sigurnosne JSON tokene za komunikaciju izmedju servisa, aplikacija i servera identiteta (IdP).

SAML je nezavisan od OAuth2 protokola, i koristi XML format za razmenu poruka za potvrdu identiteta,za razliku od JWT formata koji koristi OIDC.Češće se koriti za pomoć korisnicima korporacija da se prijave u više aplikacija pomoću jedne prijave. 

**Upoređenje ključnih razlika između OIDC i SAML protokola:**

Oba protokola se koriste za postizanje istih ciljeva, međutim sam nacin kako autentikuju korisnika se razlikuje, korišcenjem drugačijih tehnologija i metoda.

**RP** **nasuprot SP :** Oba protokola, RP ( Relying Party, termin za klijentsku aplikaciju) u terminologiji OIDC-a, i SP (Service Provider) u slucaju SAML, redirektuju korisnika serveru identiteta (IdP ili OP).Razlika je u tome sto SAMP nije bas kompatibilan sa određenim tipom aplikacija, kao sto su na primer mobilne aplikacije, u poređenju sa OIDC-om koji jednako dobro funkcionise kako sa veb tako i sa mobilnim aplikacijama.

**Format Poruka :** SAML koristi tvrdnje koje predstavljaju attribute autentikacije kao I takozvane izjave za dodelu prava pristupa I formatirane su korišćenjem XML formata. Sa druge strane OIDC koristi JSON Web tokene pod nazivom ID Tokeni u JWT format, koji pruža informacije za potvrdu identiteta.

**Podrška za API :**  OIDC je nastao iz razloga što je SAML Xml format “Težak”, I ne moze se dobro integrisati sa Api-ima. OIDC radi tako što koristi HTTP komunikacioni kanal za slanje “Lakih” JSON sigurnosnih tokena za potvrdu identiteta korisnika. Sa druge strane SAML koristi SOAP, koji je takođe protokolski sloj preko HTTP-a, međutim šalje “Teške” poruke u XML format za potvrdu identiteta korisnika.

**Autentikacija korišćenjem mobilnih aplikacija :** OIDC koristi RESTFUL komunikaciju za stvaranje i prosleđivanje “Laganih“ JSON tokena, koji se prosleđuju između servera identiteta (IdP) i pouzdane strane (relayng party) odnosno klijentske aplikacije, što čini OIDC vodećim protokolom za mobilne aplikacije. Za razliku od SAML-a koji je uglavnom razvijen za autentikaciju veb aplikacija, a budući da koristi XML, nema prostora za buduću upotrbu u mobilnim aplikacijama, što daje prednost OIDC-u.

**Korisnikov Pristanak :** S obzirom da je OIDC izgrađen na vrhu OAuth2 protokola za podršku autentikacije, može pružiti ugrađeni sloj za prava pristupa (Autorizacije) , koji podstiče korisnika da prvo pristane na ono što pružalac usluga (IdP) može obezbediti odnosno dati pristup.SAMl takođe može obezbediti tok saglasnosti, međutim to čini kodiranjem samog programera (“Hard-Coding“), umesto da obezbedi tok saglasnosti kao deo standarda u svom protokolu.

**Statična Autentikacija :**  U SAML protokolu Server Identiteta (IdP) i Pouzdana strana (relayng party) moraju biti konfigurisani da se poznaju pre stvarnog prenosa podataka. Sa druge strane za OIDC to nije slučaj.

**Implementacija :** OIDC protokol je daleko jednostavnije implementirati od strane programera , što povecava upotrebljivost samog protokola za različite svrhe. Takođe može biti lako implementiran od nule ili korišćenjem već gotovih biblioteka koje su dostupne kao pomoć i podrška developerima.