فناوری شناسایی یکپارچه Single Sign On

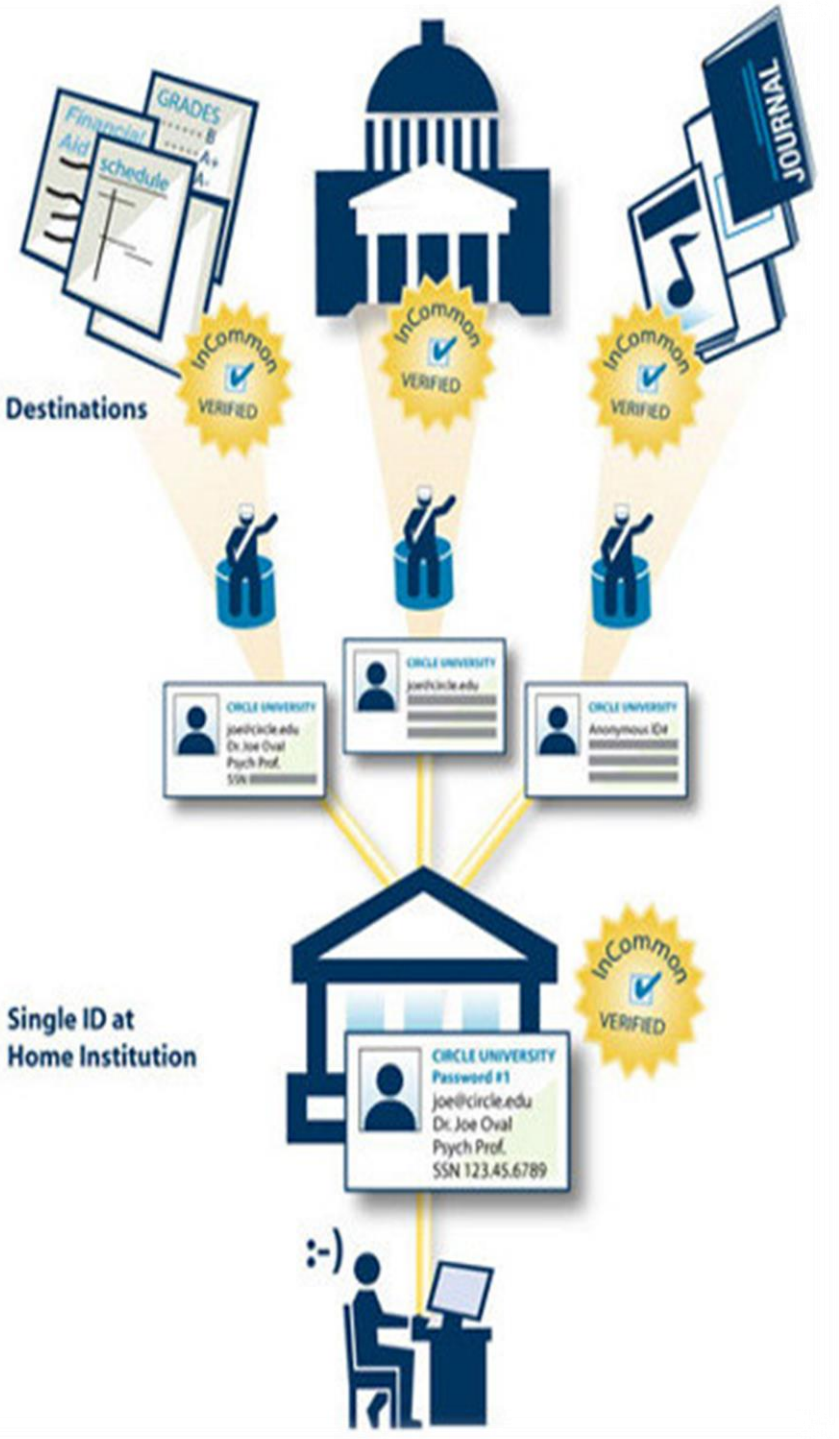
Single Sign On

Single Sign On یا SSO به معنی وارد شدن کاربر به برنامه ها، سایت ها و سیستم های مختلف، فقط با یک نام کاربری و رمز عبور میباشد.به طور معمول کاربران برای دسترسی به سایت ها و منابع مختلف لازم است که نام کاربری و رمز عبور جداگانه ای داشته باشند که تعداد کاراکترهای آنها گاهی به ۱۰ عدد میرسد. با زیاد شدن این اکانت ها احتمال فراموش شدن آنها نیز بیشتر میشود. با استفاده از SSO تنها یک نام کاربری و رمز عبور برای کاربر وجود خواهد داشت که میتواند در سیستمی که از این فناوری استفاده میکند به تمامی بخش ها تنها با یکبار ورود اطلاعات کاربری دسترسی داشته باشد. توسط این تکنولوژی، اطلاعات مربوط به تایید هویت شامل نام کاربری و رمز عبور در قسمتی امن نگهداری میشوند و پس از آن کاربر برای ورود به بخش ها و حساب های مختلف خود تنها یکبار لاگین میکند. در هنگام لاگین اطلاعات ثبت شده ای که مربوط به تعیین سطح دسترسی کاربر میباشد با اطلاعات اکانت کاربر تطبیق داده میشود و در صورت احراز هویت و مطابقت این موارد اجازه ورود برای کاربر صادر میشود. در SSO امنیت رمز عبوراهمیت بسیار زیادی دارد و اکانت هایی که بین سیستم ها انتقال پیدا می نمایند باید بصورت رمزنگاری / کد گذاری شده بکار گرفته شوند . با استفاده از SSO امکان برقراری اتباط بین تمام منابع سیستم وجود دارد. همچنین میتوان در هر زمان و از هر مکانی به سرویس های مورد نظر دسترسی داشت. یکی از موارد برتر استفاده از فناوری SSO، شرکت گوگل میباشد. شما با یکبار لاگین به یکی از حساب های کاربری خود در گوگل مثلا gmail همزمان به تمامی سرویس های دیگر گوگل نیز دسترسی خواهید داشت.

SSO

With InCommon

- Users have a single point of signing on to access different resources.
- Fewer user accounts to manage.
- Fine-grained control over the release of identities and other user information.
- Standards-based and open source.
- New resource providers and users can be integrated quickly.
- Access is based on information attributes, not identity or location.



پروتکل OAuth چیست؟

یک قرارداد و مجوز باز است که به سرویس های اینترنتی اجازه می دهد اطلاعات کاربران را بدون نیاز به دادن یک گذرواژه و نام کاربری بصورت امن با دیگر سرویس ها به اشتراک بگذارد و همینطور یک دسترسی موقت را از طریق یک توکن دسترسی ایجاد کند و بوسیله آن اجازه انجام یکسری کار مشخص را از طرف کاربر می دهد.

OpenID vs OAuth

اوپن آیدیOpenID درباره احراز هویت است (به عنوان مثال اثبات اینکه شما چه کسی هستید؟) و OAuthدر مورد مجوز(یعنی دسترسی به یک عملکرد،یا داده ای از اطلاعات) بدون نیاز به احراز هویت می باشد.بنابراین احراز هویت در مورد اطمینان از این است که شخصی که با آن در ارتباط هستید در واقع همان کسی است که او ادعا می کند ولی مجوز در مورد اینه که تصمیم بگیرد چه کسی مجاز به انجام آن است.

درOpenID (احراز هویت):

فرض کنید سرور Aمی خواهد کاربر Pرا تایید کند، اعتبار کاربر Pتوسط سرور Bمشخص می شود، در حقیقت سرور Bاطمینان می دهد که Pدر واقع Pاست، و سپس به Aمی گوید: خب این Pواقعی است.

اما درOAuth (مجوز):

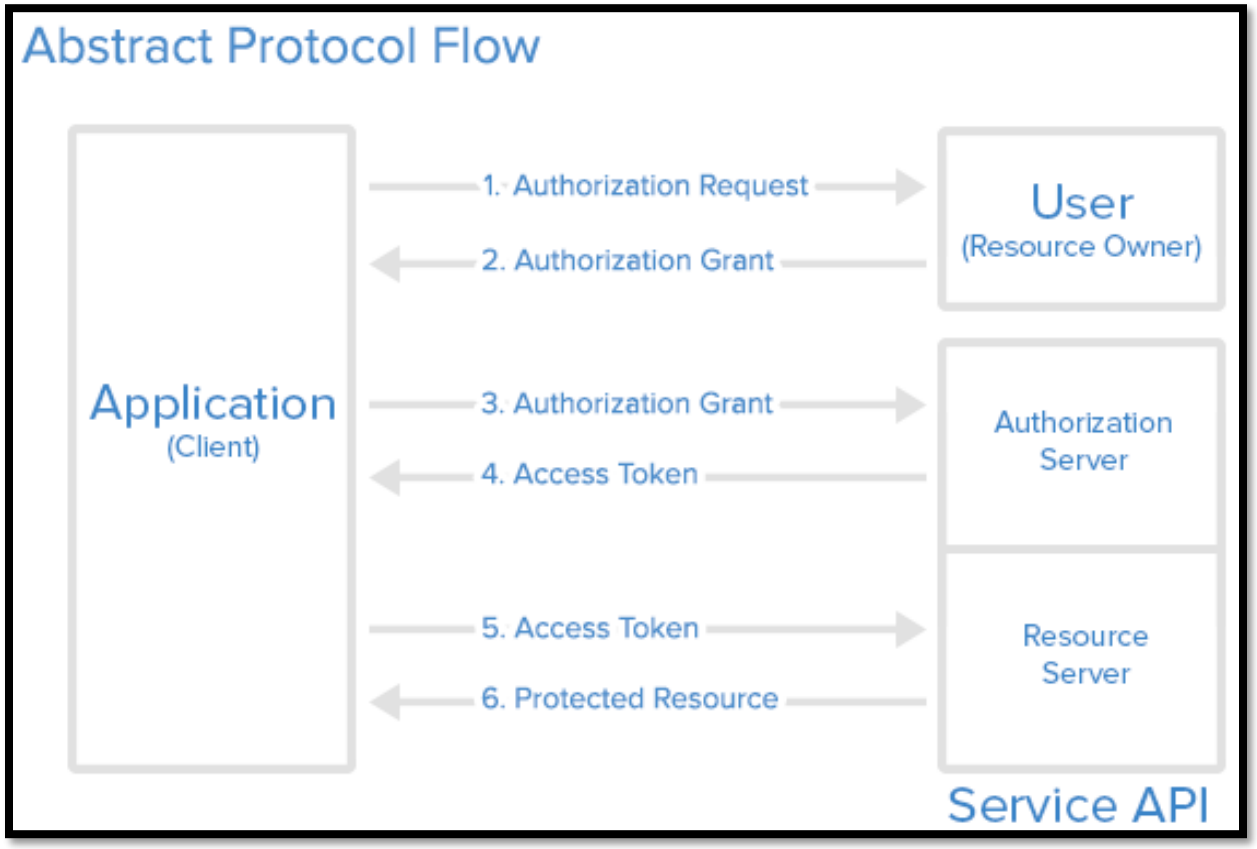
کاربر P از سرور A یک توکن دسترسی دریافت می کند که می تواند آنرا در اختیار سرور B برای دسترسی به یک سری موارد مشخص شده بدهد، بنابراین A می تواند کلیدهای دسترسی موقت و خاصی را برایB فراهم آورد.

برای مثال شما می توانید سرور OAuthرا به عنوان کلید دار یک هتل بزرگ تصور کنید؛ او کلیدها را به کارکنان می دهد که درهای اتاق را برای شخصی که قرار است وارد اتاق شود باز کنند، اما هر کلید محدود است و فقط در یک اتاق را باز میکند (یعنی با یک کلید دسترسی به تمام اتاق ها را ندارید)؛ علاوه بر این کارکنان موظف هستند کلید خود را پس از چند ساعت از بین ببرند. در نتیجه میتوان گفت که OAuthیک چارچوب یا مجوز است که به تنهایی یک پروتکل احراز هویت نیست ومیتوان از این چارچوب مجوز در احراز هویت استفاده نمود.

کاربرد OAuth

فرض کنید شما بخواهید از خدمات یک وب سایت استفاده کنید یا تحلیلی بر روی داده های یک کاربر در آن وب سایت ارائه بدهید که برای این کار نیاز به نام کاربری و رمز عبور کاربر می باشد که در این صورت باید کاربر نام کاربری و رمز عبور خود را در اختیار شما قرار بدهد که این کار علاوه بر جلب اعتماد کاربر، مشکلات امنیتی و مسئولیتی (لو نرفتن نام کاربری و رمز عبور کاربران توسط شما) مشکلات دیگری به همراه خواهد داشت و همچنین به این شکل شما دسترسی به همه قسمت های پنل کاربری که مطمئنا کاربر علاقه ای ندارد شما چنین دسترسی را داشته باشید را دارید پس یکی از راه حل های موجود OAuthبود که یک دسترسی محدود و مشخص و البته ایمن بدون نیاز به نام کاربری و رمز عبور در اختیار شما قرار دهد.

چرخه دسترسی



- برنامه (application) درخواست احراز هویت (authorization) را برای دسترسی به اطلاعات کاربر ارسال می کند.
- اگر کاربر دسترسی برنامه (application) را تائید کند کد تائید (code) برای برنامه ارسال می شود.
- برنامه درخواست access token را همراه با کد تائید (code) به سرور ارسال می کند.

- اگر اطلاعات برنامه و کد صحیح باشد دسترسی برنامه به اطلاعات کاربر میسر می شود و access token به برنامه داده می شود.
- برنامه از طریق access token درخواست خود را به وبسرویس (API) ارسال می کند.
- در صورت صحت access token ، درخواست بروی وبسرویس انجام شده و پاسخ بر گردانده می شود.

SSO	OAUTH
دسترسی با یک رمز واحد به اکانت های مختلف	دسترسی با یک رمز اما موقت ومحدود
آسان بودن مدیریت اکانت ها	مدیریت اکانت ها به صورت کاملا امن
همکاری بین اطلاعات و کاربران سریع تر و امن تر است.	همکاری بین اطلاعات و کاربران با محدودیت و بسیار امن است.
یکپارچگی بین کاربران و اطلاعات	یکپارچگی همراه با محدودیت
احراز هویت	دادن مجوز برای استفاده از اطلاعات یا داده های خاص
نیاز بودن نام کاربری و رمز	نیاز نبودن نام کاربری و رمز با استفاده از توکن

رسیدن به بخش های مختلف سایت دشوارتر است.	رسیدن به بخش های مختلف سایت راحت تر است.
کاهش ریسک به دلیل استفاده کمتر از رمز های نامناسب	کاهش بسیار زیاد خطرات احتمالی در رابطه با فراموش کردن رمز تنها با یک بار ورود