Single Sign On

SSO & ID Management

for

Web and Mobile Applications

Presenter: Manish Harsh

- Program Manager for Developer Marketing Platforms of NVIDIA (Visual Computing company)
- Advisor in Halosys (Enterprise Mobility Solutions and Framework company)

Agenda

- ❑ Plan B ?
- ❑ Definition(s)
- ❑ Types of SSO
- ❑ Story
- ❑ Why Identity Management
- ❑ Commandments for IDM
- ❑ Devil is in the details
- ❑ Strategy
- ❑ Methodology and Monitoring
- ❑ Deployment Plan
- ❑ SSO Drupal Modules and Techniques
- ❑ References and Information
- ❑ Contact Info

Plan B ?

“Having a Plan B
only
distracts you from Plan A”

Definition of SSO

Single sign-on (SSO) is a property of access control to multiple related, but independent software systems.

OR

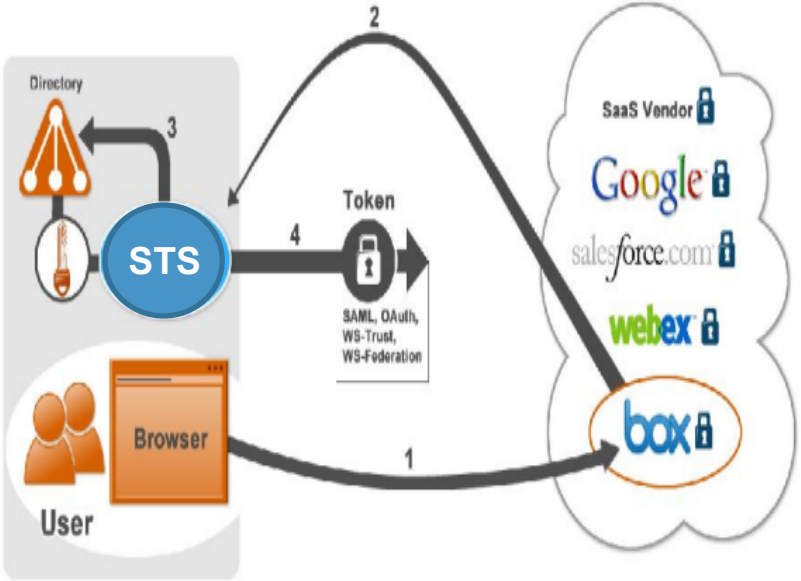
Users provide credentials only once per session, and then gain access to multiple applications without having to sign in again during that session.

Users provide same credentials for multiple applications; they might have to login multiple times, but always using the same credentials.

Enterprise SSO Scenarios

1. **Corporate Login to Cloud Application**
2. **Cloud Login to Internal Application**
3. **Corporate Login to Internal Application**
4. **Using Identity as a Service (IdaaS) Hub**
5. **Corporate Login to Partner Application**

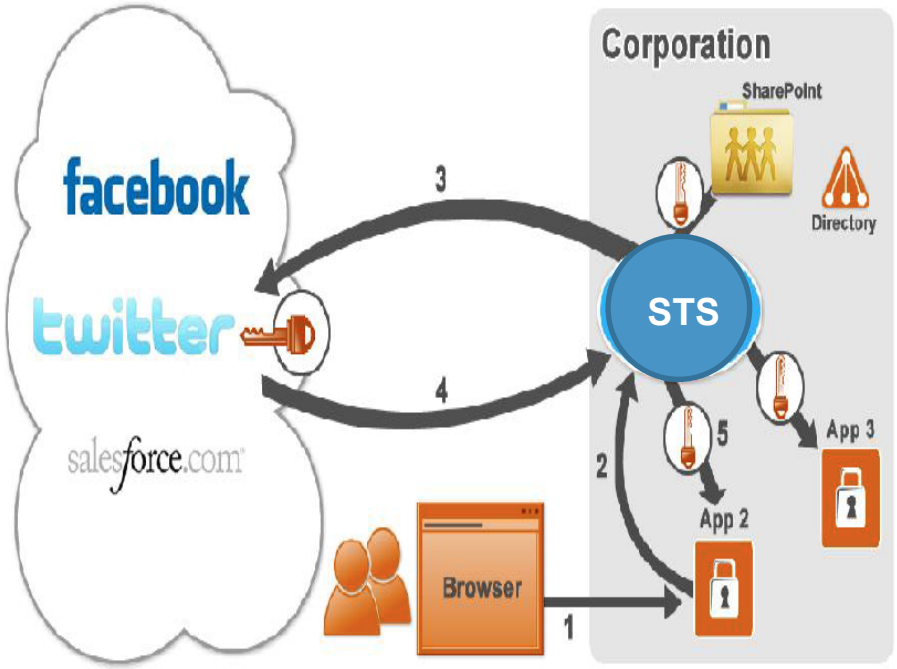
Scenario 1: Corporate Login to Cloud Application



Most commonly supported SSO scenario

Corporation hosts the Federation Server that enables SSO with Cloud applications based on standard protocols like SAML or OAuth.

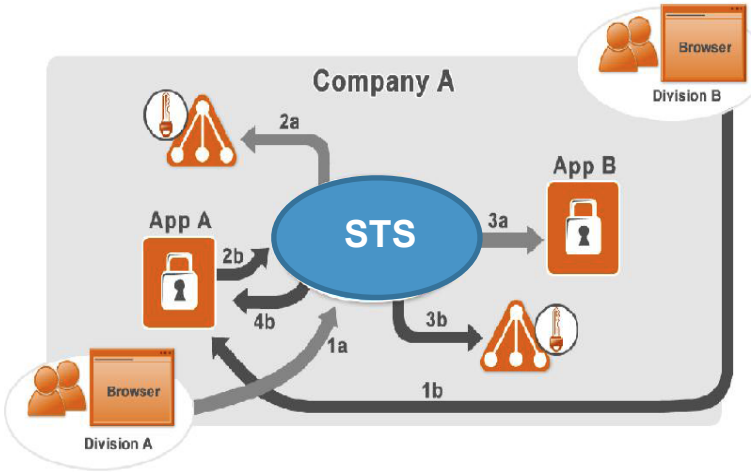
Scenario 2: Cloud Login to Internal Application



A typical example of this is a user logging into a corporate SharePoint web site with their Facebook account.

- A familiar, consumer-friendly model like this is easy to use and decreases support costs associated with a large consumer population.

Scenario 3: Corporate Login to Internal Application

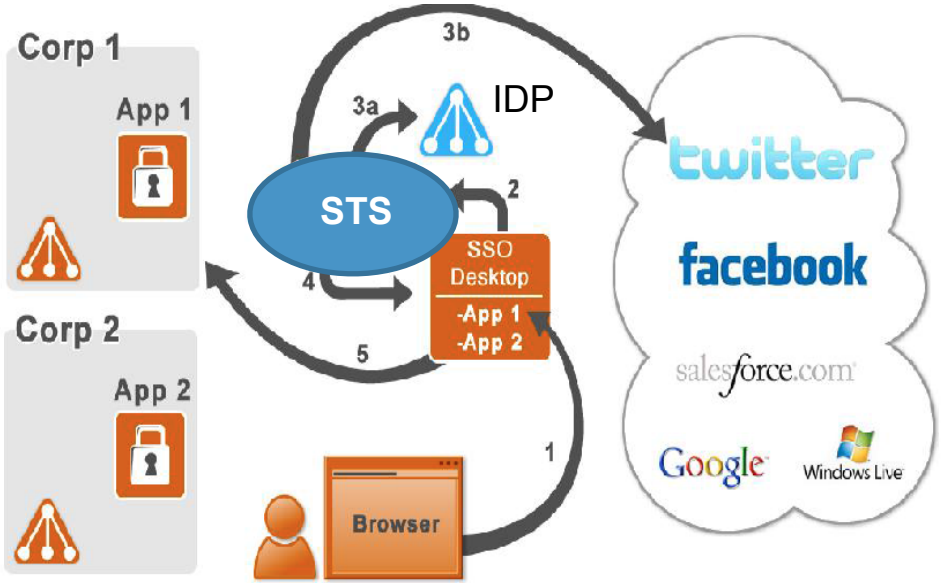


This is to decouple authentication/authorization from within each application and to leverage centralized services for these functions.

In this case, internal applications would be developed as “relying parties” that trust an internal corporate identity management system for authentication/ authorization decisions.

This scenario is often required by organizations as they acquire other companies but cannot create trusts between their Active Directory domains due to legal limitations imposed by differing localities, time constraints, or other internal policies.

Scenario 4: Identity as a Service (IdaaS) Hub



- Users log in with an identity maintained by a Cloud Identity as a Service (IdaaS) Provider and can then access multiple Cloud hosted SaaS applications or corporate hosted partner applications.
- A central shared Identity Provider functions as a hub of authentication, allowing federation trusts to be established with all major Identity Providers using industry-standard protocols like SAML, WS-Federation, WS-Trust, OpenID, and OAuth.

This scenario is more common in specific industries such as healthcare where hospitals and doctor's practices partner with insurance companies and health care plans.

Story

SSO for user(s) across multiple applications (Web and Mobile) **based on their role and permission.**

Current scenario:

Several applications built on different technologies like Drupal 6, Drupal 7 custom PHP, Ruby on Rails and Microsoft technologies.

Note: Few of the sites also have native mobile apps.

Total Users: **~1 Million**

Challenge (Or Risk)

Security: If a user's account or password information is compromised, an intruder could have extensive and easier access to far more resources.

Cost: SSO implementations can be expensive in two aspects - the cost to purchase and the manpower to deploy.

SSO Between Drupal & Non-Drupal Sites

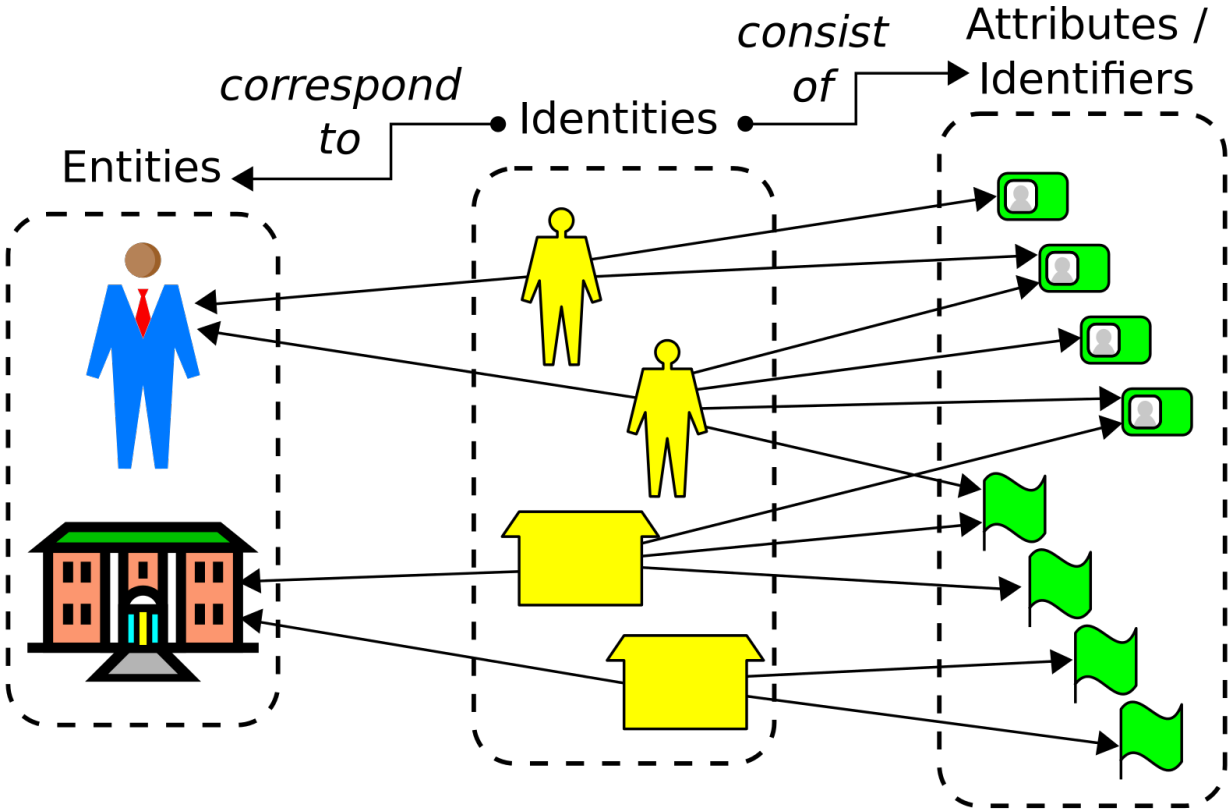
1. **Identity Management**
2. **Custom Authentication**

ID Management Systems with Drupal Module

IDM	Drupal Modules
Janrain	https://www.drupal.org/project/janrain_capture
Gigya	https://www.drupal.org/project/gigya
Onelogin	https://www.drupal.org/project/onelogin
More	Exploring…

Identity Management

Identity + Management



Identity + Management

Authentication	Verification that an entity is who/what it claims to be using a password.
Authorization	Managing authorization information that defines what operations an entity can perform in the context of a specific application.
Roles	Roles are groups of operations and/or other roles. Users are granted roles often related to a particular job or job function.
Delegation	Delegation allows local administrators or supervisors to perform system modifications without a global administrator or for one user to allow another to perform actions on their behalf.
More..	Features like analytics, widgets, notifications, alerts etc.

Terms for SSO and IDM

Directory services	**STS (Security Token System)**
Service Providers	**Workflows**
Identity Providers	**OpenID**
Web Services	**WS-Security**
Access Control	**WS-Trust**
Digital Identities	**SAML 2.0**
Password Managers	**OAuth**
Security Tokens	**RBAC (Role Based Access Control)**

Commandments for IDM

Determine and declare Authentication Policies	• Strength policies • Password management policies • Contractual service level agreement policies • Change management plan

Strategy

Determine the global session time outs and the device(client side) inactivity timeouts.	User Experience and Security Protocol
Identify specific applications where due to enterprise risk the timeouts need to be lower than your enterprise values.	Understand your system
Design a strategy for handling these timeouts.	User Experience

Methodology and Monitoring

Determine the action • Failed authentication • Post-authentication • Authorization	Technical Methodology and User Experience
Transaction Authentication *Note: Users should be notified in advance that their computer hardware and IP addresses will be monitored as well as their usage patterns.*	Transaction authentication is extra monitoring in addition to the successful use of user id and password. • IP address • Geo location • Client device (Mobile/computer) • Login pattern (Time, duration etc.)

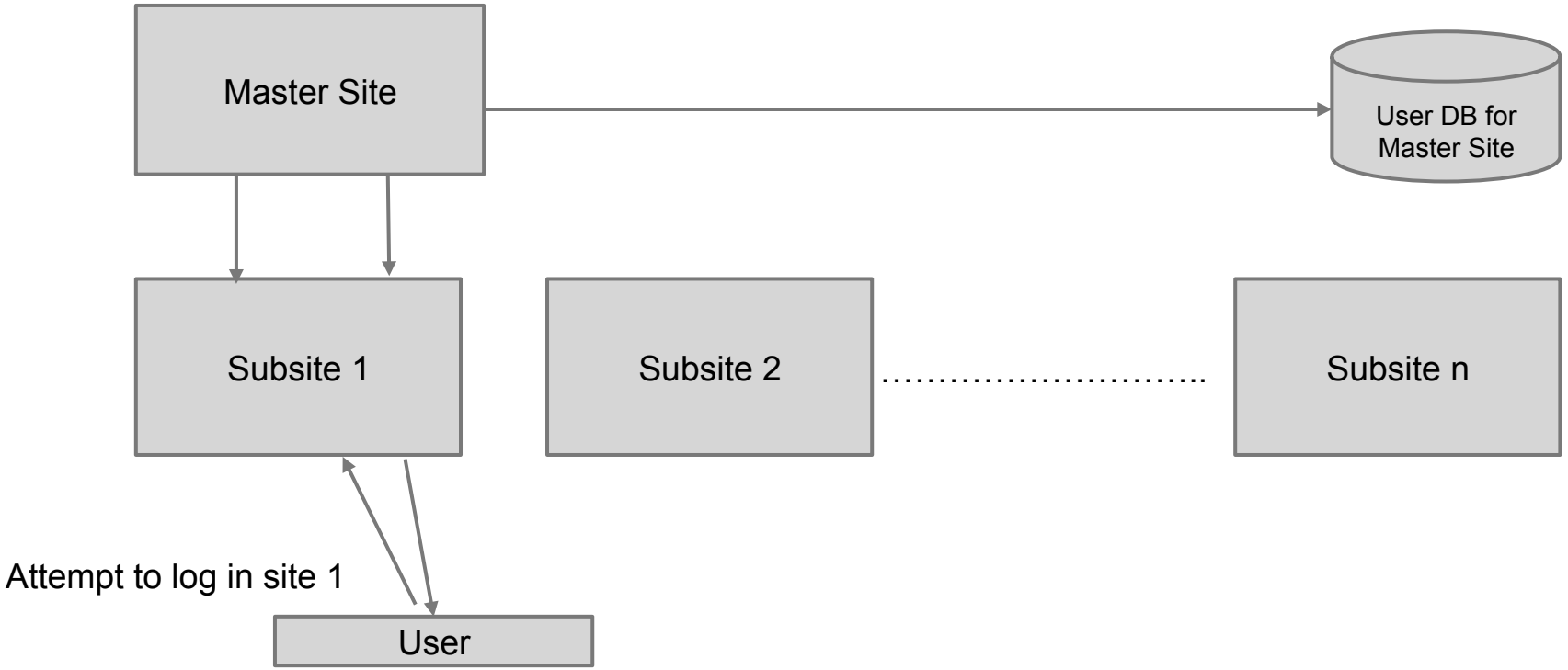
Deployment Plan

Determine the number of environments you will use for SSO).	• Development • Test • QA • Pre-production • Production
Determine how applications will be quickly moved between environments?	Integration Scripts and tools
Determine the peak SSO loads	Scalability parameter

Custom Authentication

1. **External Authentication Script**
2. **Session Check Script**
3. **Ticket Generation Script**

Architecture for Custom Authentication



External Authentication Script

- User attempts to login to any of the site configured for SSO by providing username and password.
- The username and password are verified against a common Database (preferably the Master Drupal DB).
- Once the script validates the user with the Master DB the user is logged in to the requested Site.
- Appropriate permission parameters are then checked by the script and the user is granted request to the specific resources.

Session Check Script

- User requests access to any of the resources on the connected sites.
- The script checks if the user is logged in to the master site.
- If the user is logged in on the master site the user is succesfully logged in.
- If the user is not logged in on the master site:
  - The user is taken to the login page for the master site.
  - Once the user is logs in on the master site the script check is done again and the user is logged in on the other connected sites.

Ticket Generation Script

- User requests access to any of the resources on the connected sites.
- The Master DB is requested to generate ticket for the requesting access.
- The user is validated and a session ticket is generated by the ticket generation server which is saved as a session parameter in the user's browser.
- The connected sites then use this ticket to verify the login for the user with the ticket generation server for the validity of the ticket.
- Login request providing valid tickets are logged in automatically by the script.
- The generated tickets are session specific and expire after the configured time.

Popular Drupal SSO Modules / Techniques…

CAS

In its most simple use, CAS authenticates users and sends the user to the requested application with a ticket. The application is then responsible for authenticating the ticket (behind the scenes, with a tool like cURL) and automatically logging the user in if the ticket is valid.

CAS can also proxy single sign-on.

URL: https://drupal.org/project/cas

Usage Statistics: https://drupal.org/project/usage/cas

Popular Drupal SSO Modules / Techniques…

OAuth Connector

It makes it possible to connect and sign in a Drupal user with accounts on most third party sites with OAuth APIs. It provides a UI for adding and editing specifications of OAuth APIs that the users should be able to connect to. It also provides exportability of those specifications.

Is an implementation of the Connector module.

URL: https://drupal.org/project/oauthconnector

Usage Statistics: https://drupal.org/project/usage/oauthconnector

Popular Drupal SSO Modules / Techniques…

LDAP SSO

The LDAP Single Sign-On module provides an administrator with the ability to configure a Drupal site to use either NTLMSSP (e.g. seamless automatic login using LDAP / Active Directory credentials passed automatically by supported and properly configured browsers) or basic digest authentication as a fallback to authenticate Drupal users. The net effect is that either automatically, or by visiting a link, a user is authenticated and logged into a Drupal site without requiring the user to manually enter credentials on suitably configured installations.

URL: https://drupal.org/project/ldap_sso

Usage Statistics: https://drupal.org/project/usage/ldap_sso

Popular Drupal SSO Modules / Techniques…

Bakery SSO

Bakery provides a "single sign-on" feature for Drupal based sites that are on the same second-level domain (i.e. example.com, subsite.example.com, subsite2.example.com). It could also provide support for any other website that implements the same web cookie, xmlrpc, and POST methods.

URL: https://drupal.org/project/bakery

Usage Statistics: https://drupal.org/project/usage/bakery

Also refer: http://drupalwatchdog.com/volume-2/issue-2/drupal-and-secure-single-sign

Popular Drupal SSO Modules / Techniques…

Shibboleth Authentication

Provides user authentication with Shibboleth (both v1.3 and v2.0) as well as some authorization features (automatic role assignment based on Shibboleth attributes).

More Info: https://wiki.shibboleth.net/confluence/display/SHIB2/FlowsAndConfig

URL: https://drupal.org/project/shib_auth

Usage Statistics: https://drupal.org/project/usage/shib_auth

Popular Drupal SSO Modules / Techniques…

Account Sync

The account_sync module allows you to synchronize drupal user account data across multiple Drupal sites.
It currently supports basic account information as well as the drupal core profile module. This module uses XMLRPC to transmit data between sites when updates are made so there is no need to have your sites running on the same database, server, or on the same subdomain.

URL: https://drupal.org/project/account_sync

Usage Statistics: https://drupal.org/project/usage/account_sync

Popular Drupal SSO Modules / Techniques…

OpenID Single Sign On Relying Party

This module provides a simple single sign on solution based on OpenID and native in Drupal. It is the relying party counterpart for a server based on OpenID, related to OpenID Single Sign On Provider. You can set up a central provider (which ideally is another instance of Drupal) and a lot of another (Drupal) websites (so called relying parties). This way the users can login to every single relying party website using a centralized login provider where authentication is happening.

URL: https://drupal.org/project/openid_sso_relying

Usage Statistics: https://drupal.org/project/usage/openid_sso_relying

Popular Drupal SSO Modules / Techniques…

Google Apps Authentication

Google Apps provides a single sign on API that enables people to write applications that do user authentication against a local database, and then tell google that the user is authenticated. This modules implements the API in drupal. In other words, once properly setup, this module lets Google Apps instances authenticate against your drupal user database.

URL : https://drupal.org/project/googleauth

Usage Statistics :https://drupal.org/project/usage/googleauth

References

http://merbist.com/2012/04/04/building-and-implementing-a-single-sign-on-solution/

http://www.authenticationworld.com/Single-Sign-On-Authentication/101ThingsToKnowAboutSingleSignOn.pdf

http://en.wikipedia.org/wiki/Single_sign-on

https://groups.drupal.org/node/182004

http://drupal.stackexchange.com/questions/1758/how-drupal-org-single-sign-on-works

http://mauriziostorani.wordpress.com/2008/07/21/single-sign-on-sso-concepts-methods-and-frameworks/

http://scn.sap.com/thread/733802

Thanks ☺

1. **Twitter handle**: @manishharsh
2. **LinkedIn** Profile: https://linkedin.com/in/manishharsh
3. **SVDUG** (Silicon Valley Drupal User Group): https://meetup.com/drupalgroup
4. **Email**: manishharsh@gmail.com