Authentication in SharePoint 2013

Three types of authentication: User, App, Server-to-Server

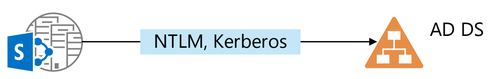
User Authentication

User authentication occurs when a user attempts to access a SharePoint resource and the user's identity is validated against an authentication provider.

Windows claims-based authentication

Windows claims-based authentication uses your existing Windows authentication provider (Active Directory Domain Services [AD DS]) to validate the credentials of connecting clients. Use this authentication to allow AD DS-based accounts access to SharePoint resources.

Authentication methods include NTLM, Kerberos, and Basic.



For Windows claims authentication, SharePoint 2013 uses the NTLM or Kerberos protocols to validate user credentials for users that are in forests and domains trusted by the SharePoint 2013 server.

SharePoint 2013 also supports classic-mode Windows authentication, but this is not recommended. Classic-mode Windows authentication does not support app authentication, server-to-server authentication, and Office Web Apps. To configure a web application to use classic-mode authentication, you must use the **New-SPWebApplication** or **Set-SPWebApplication** Windows PowerShell cmdlets.

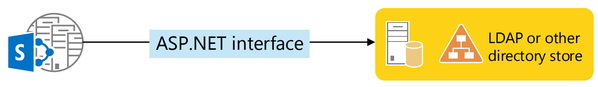
Forms-based authentication

Forms-based authentication can be used against credentials that are stored in an authentication provider that is available through the ASP.NET interface. These include:

- AD DS.
- A database such as a SQL Server database.
- A Lightweight Directory Access Protocol (LDAP) data store.

Forms-based authentication validates users based on credentials that users type in a login form (typically a web page). Unauthenticated requests are redirected to a login page, where a user must provide valid credentials and submit the form.

Use forms-based authentication with accounts in authentication providers that are available by using the ASP.NET interface.



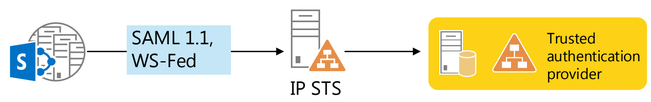
With forms-based claims authentication, SharePoint 2013 uses the ASP.NET interface to access membership providers and role managers to validate user credentials and obtain user roles.

SAML token-based authentication

SAML token-based authentication in SharePoint 2013 requires coordination with administrators of a claims-based environment, whether it is your own internal environment or a partner environment.

A SAML token-based authentication environment includes an identity provider security token service (IP-STS). The IP-STS issues SAML tokens on behalf of users whose accounts are included in the associated authentication provider. Tokens can include any number of claims about a user, such as a user name and the groups to which the user belongs. An Active Directory Federation Services (AD FS) 2.0 server is an example of an IP-STS.

Use SAML token-based authentication to allow accounts in authentication providers that are available by using a compatible IP-STS access to SharePoint resources.



For SAML token-based claims authentication, SharePoint 2013 supports the SAML 1.1 and WS-Federation Passive Requestor Profile (WS-Federation PRP) protocols for requesting computers to obtain SAML tokens as proof of validation of user credentials and for additional claims.

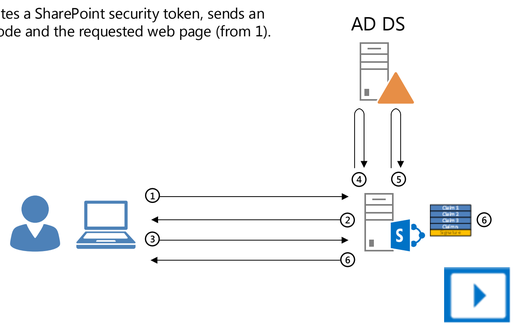
How it works

For videos of these example processes:

http://aka.ms/U9plzl

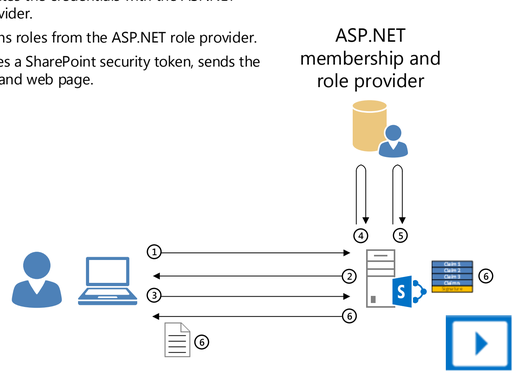
In this example, a user who does not already have a SharePoint security token accesses a secured SharePoint web page.

1. User requests a web page.
2. SharePoint requests Windows credentials using NTLM, Kerberos, or basic protocols.
3. User sends the Windows credentials for the user account.
4. SharePoint validates the user's Windows credentials with AD DS.
5. SharePoint obtains the group membership list for the user account from AD DS.
6. SharePoint creates a SharePoint security token, sends an authorization code and the requested web page (from 1).



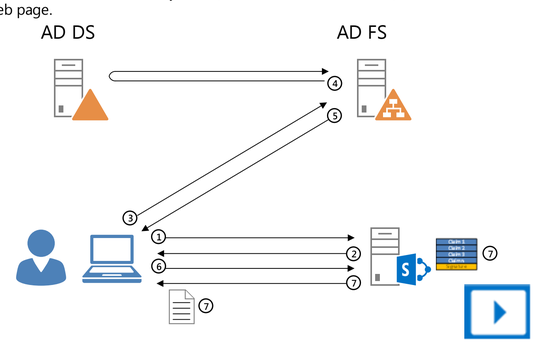
In this example, a user who does not already have a SharePoint security token accesses a secured SharePoint web page.

1. User requests a web page.
2. SharePoint sends a SharePoint forms-based login page.
3. User sends the credentials entered in the forms-based login page.
4. SharePoint validates the credentials with the ASP.NET membership provider.
5. SharePoint obtains roles from the ASP.NET role provider.
6. SharePoint creates a SharePoint security token, sends the FedAuth cookie, and web page.



In this example, a user who does not already have a SharePoint security token accesses a secured SharePoint web page.

1. User requests a web page.
2. SharePoint sends a redirect and the user loads a login page from the AD FS server.
3. User sends user credentials and requests a SAML security token.
4. AD FS validates the user credentials with AD DS (the authentication provider).
5. AD FS sends a SAML security token.
6. User sends a new web page request containing the SAML security token.
7. SharePoint creates a SharePoint security token, sends the FedAuth cookie, and the requested web page.



Configuration

New web applications created in Central Administration are configured by default to use Windows authentication and the NTLM method:

- Modify the authentication settings for the web application in Central Administration to include Kerberos.
- Modify the settings of the web application in the IIS Manager snap-in to enable the Digest and Basic authentication methods.

Configuring forms-based authentication involves the following:

- Modify the Web.config files for the Central Administration web application, the Security Token Service web site, and the web application to register the ASP.NET membership provider and role manager.
- Configure the web application with the name of the membership provider and role manager that was added to the Web.config files.
- Optionally, you can also create a custom login page for forms-based authentication.

The key elements of SAML token-based authentication are the following:

- Configure the IP-STS with the set of authentication providers (such as AD DS, databases, and others) corresponding to organization and partner accounts.
- Configure the IP-STS with the set of relying parties corresponding to the web applications that use SAML token-based authentication and claims mappings.
- Configure the SharePoint 2013 farm with the token signing certificate of the IP-STS, the corresponding claims mappings as done on the IP-STS, and the name of the IP-STS as a trusted security token issuer.
- Configure the web application with the name of the IP-STS as a SAML identity provider.

App Authentication

App authentication occurs when an external component of an app for SharePoint attempts to access a secured SharePoint resource. App authentication is a combination of:

- Verification of a remote app for SharePoint's identity (authentication).
- Validation of the type of access through user and app permissions (authorization).

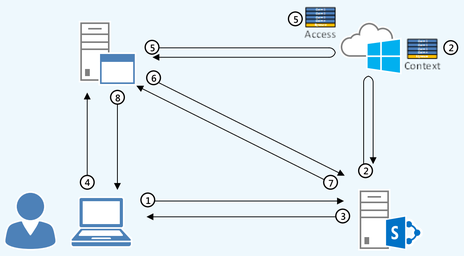
To get started with app development and deployment:

http://aka.ms/T1gtks

Low-trust Apps

A low-trust app relies on the Windows Azure Access Control Service (ACS) as the trusted security token issuer for access tokens that are required to obtain secured resources on a SharePoint farm. The app provider or Windows Azure can host low-trust apps. To trust low-trust apps, you must have an Office 365 subscription.

In this example, a user accesses a secured SharePoint web page containing an IFRAME that is rendered by an app hosted in Windows Azure. To render the IFRAME, the app must access a resource on the server running SharePoint 2013 on behalf of the requesting user.



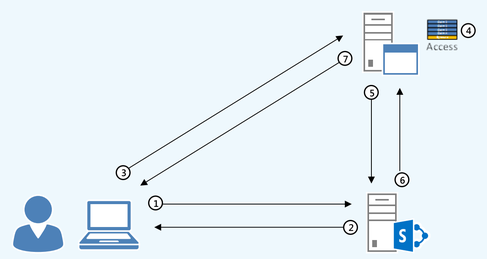
1. User requests a web page containing the Azure app's IFRAME.
2. SharePoint obtains a context token from ACS.
3. SharePoint sends the web page with the app's IFRAME and the context token to the user.
4. User requests the IFRAME contents with the context token from the app.
5. App obtains an access token from ACS.
6. App requests the SharePoint resource using the access token.
7. After app and user authorization, SharePoint responds with the requested resource.
8. App sends the IFRAME contents to the user.

High-trust Apps

A high-trust app is an app that an intranet server or a provider's server on the Internet hosts. For high-trust apps, the app acts as the trusted security token issuer for access tokens that are required to obtain secured resources on a SharePoint farm.

In this example, a user accesses a secured SharePoint web page containing an IFRAME that is rendered by a high-trust app. To render the IFRAME, the app must access a resource on the server running SharePoint 2013 on behalf of the requesting user.

1. User requests a web page containing the high-trust app's IFRAME.
2. SharePoint sends the web page with the app's IFRAME.
3. User requests the IFRAME content from the high-trust app server.
4. App authenticates the user and generates an access token.
5. App requests the SharePoint resource using the access token.
6. After app and user authorization, SharePoint responds with the requested resource.
7. App sends the IFRAME contents to the user.



Server-to-Server Authentication

Server-to-server authentication enables a new set of functionality and scenarios that utilize cross-server resource sharing and access, including the following:

- **eDiscovery** Discover and place holds on content in the SharePoint farm, in Exchange Server 2013, on file shares, and in other SharePoint farms.
- **Exchange task synchronization** Allows users to synchronize SharePoint Server 2013 and Project Server tasks with Exchange Server 2013 and have them appear in Outlook 2013.
- **Site mailboxes** Provides SharePoint Server 2013 users with team email, hosted by Exchange Server 2013, on a SharePoint site.
- **SharePoint 2013 Hybrid** Federated search, Business Connectivity Services, and Duet Online between an on-premises SharePoint 2013 farm and SharePoint Online.

Server products that are capable of server-to-server authentication:

SharePoint 2013 (on-premises deployments and SharePoint Online)

Exchange Server 2013

Lync Server 2013

Azure Workflow Service

Other software that supports the Microsoft server-to-server protocol.

How it works

Similar to app authentication, SharePoint 2013 allows access to the requested resource when the server making the request is verified as trusted and the type of access is authorized through validation of user and server permissions.

The validation of a server's request for resources that is based on a trust relationship established between the Security Token Service (STS) of the server that runs SharePoint 2013 and the security token service (STS) of another server that supports the OAuth server-to-server protocol. Based on this trust relationship, a requesting server can access secured SharePoint resources on behalf of a specified user account, subject to server and user permissions.

Configuration

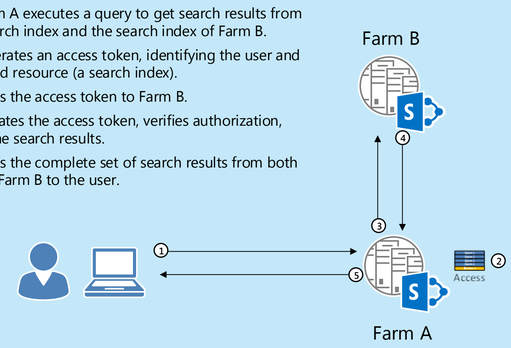
Server-to-server authentication configuration consists of adding a new trusted security token issuer that corresponds to each server that will send resource requests on behalf of users.

- For on-premises SharePoint farms, you configure the JSON metadata endpoint of the other SharePoint farm.
- For your SharePoint Online farm, you configure the JSON metadata endpoint of your Office 365 subscription.
- For servers running Exchange Server 2013 or Lync Server 2013, you configure the JSON metadata endpoint of the other server.

Example — How Server-to-Server Authentication works between on-premises SharePoint farms

In this example, Farm B has been configured to trust Farm A by using server-to-server authentication. Farm A has been configured with a result source for search that uses the Remote SharePoint protocol to get search results from the index of Farm B. To obtain search results, Farm A sends a query to Farm B.

1. User on Farm A executes a query to get search results from the local search index and the search index of Farm B.
2. Farm A generates an access token, identifying the user and the requested resource (a search index).
3. Farm A sends the access token to Farm B.
4. Farm B validates the access token, verifies authorization, and sends the search results.
5. Farm A sends the complete set of search results from both Farm A and Farm B to the user.



SharePoint

Microsoft