Introduction to single sign-on

Who is this article for?
√ Incydr
√ Code42 for Enterprise
√ CrashPlan for Enterprise
√ CrashPlan for Small Business

This article applies to Code42 cloud environments.

Other available versions:
On-premises (https://support.code42.com/Administrator/6/Configuring/Identity_management/Introduction_to_single_sign-on) (https://support.code42.com/Administrator/Identify_version)

Overview

Implementing single sign-on (SSO) (https://en.wikipedia.org/wiki/Single_sign-on) in your Code42 environment provides security benefits and simplifies the sign-in experience. This article provides:

• An overview of SSO
• A list of compatible Code42 components and third-party SAML 2.0 identity providers (IdPs)

For SSO configuration instructions, see our articles for directions on specific providers (https://support.code42.com/Administrator/Cloud/Configuring/Identity_management).

What is SSO?

Single sign-on SSO is an authentication method that allows a user to use the same credentials to sign in to multiple applications. You can integrate Code42 with any provider that uses SAML 2.0.

Definitions

authentication

The process of identifying and verifying users in a system. Methods for authentication include: Local Code42 directory, Single Sign-On (SSO), Multi-factor authentication (MFA)

authentication provider

https://support.code42.com/Administrator/Cloud/Configuring/Identity_management/Introduction_to_single_sign-on (https://support.code42.com/Administrator/Cloud/Configuring/Identity_management/Introduction_to_single_sign-on)
Allows access to Code42. When enabled, users sign in using the authentication provider instead of Code42. Examples of authentication providers include Okta, Google SSO, Ping, Azure AD, OneLogin, and Microsoft AD FS. This term is used within Code42’s identity management feature.

**identity management**

An IT administrative area or market that deals with users in a IT system and giving them access to the right resources within the system.

**identity provider (IdP)**

A general term to refer to a system that contains user identities. Identity provider can refer to a system performing authentication, provisioning, or both. Examples of identity providers include Okta, Google SSO, Ping, Azure AD, and OneLogin.

**service provider**

A system acting as a gatekeeper for one or more resources (applications).

**resource**

A protected application, which may or may not be web-based. The resource and the service provider are often integrated.

**single sign-on (SSO)**

SSO is one type of authentication method. It allows a user to use the same credentials to sign in to multiple applications.

**user agent**

A software application that acts on behalf of the user who wishes to access resources. The user agent is often a web browser, although it can also be a desktop application, mobile app, or another type of agent.

**SSO authentication process**

When a user attempts to access an SSO-enabled protected resource, such as a Code42 application or Code42 console, the user is redirected to the identity provider. If the user still has an active session with the identity provider, the user is automatically redirected to the desired resource. If the user does not have an active session, the user is prompted to enter credentials. Once authenticated, the user has access for a configurable period of time to all resources protected by the identity provider.

The following diagram describes how the Code42 platform components and the SSO identity provider interact.

- **Service provider**: Code42 for Enterprise Code42 cloud instance
- **User agent**: Code42 for Enterprise applications or web browser
- **Identity provider**: A SAML 2.0 identity provider that supports HTTP POST binding
<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>When a user attempts to sign in, the user agent sends a sign-in request to the service provider.</td>
</tr>
<tr>
<td>2</td>
<td>The service provider refers the user agent to the identity provider’s SSO URL.</td>
</tr>
<tr>
<td>3</td>
<td>The user agent sends an authentication request to the identity provider.</td>
</tr>
<tr>
<td>4</td>
<td>The identity provider authenticates the user and provides the user agent with a SAML authentication token.</td>
</tr>
<tr>
<td>5</td>
<td>The user agent sends the authentication token to the service provider.</td>
</tr>
<tr>
<td>6</td>
<td>The service provider accepts the authentication token and grants the user access to the user agent.</td>
</tr>
</tbody>
</table>

**SSO advantages, disadvantages, and limitations**

**Advantages**

- Delegates all authentication to the identity provider
- Allows for centralized authentication in organizations that do not implement Active Directory or LDAP (for example, computers that are not tied to a directory)
- Minimizes phishing opportunities
- Provides detailed reporting on user access
- Reduces user password fatigue from different username and password combinations
- Reduces time spent re-entering passwords
- Reduces IT costs due to lower number of IT help desk calls about passwords

[https://support.code42.com/Administrator/Cloud/Configuring/Identity_management/Introduction_to_single_sign-on](https://support.code42.com/Administrator/Cloud/Configuring/Identity_management/Introduction_to_single_sign-on)
Disadvantages

- Prevents access to service providers if the identity provider is unavailable
  *For this reason, SSO can be undesirable for systems requiring guaranteed access at all times, such as security or plant-floor systems.*

- Allows an unauthorized user to gain access to all protected resources if a user's credentials are compromised
  *To reduce risk, ensure that credentials are stored securely, and consider implementing strong authentication methods such as smart-cards and [one-time password tokens](http://en.wikipedia.org/wiki/One-time_password).*

- Provides [user authentication](https://support.code42.com/Administrator/6/Configuring/Identity_management#Authentication) but does not provide [user management](https://support.code42.com/Administrator/6/Configuring/Identity_management#Authorization)
  *User management is provided by the local Code42 directory, [SCIM provisioning](https://support.code42.com/Administrator/Cloud/Configuring/Introduction_to_SCIM_provisioning), or [Code42 User Directory Sync](https://support.code42.com/Administrator/Cloud/Monitoring_and_managing/How_LDAP_sync_works)*

Code42 limitations

- Code42 does not handle single sign-off. If a user logs out of the Code42 environment, the Code42 cloud does not notify other service providers, and vice-versa.

- When a user signs out of the SSO identity provider, he or she is not automatically signed out of the Code42 for Enterprise applications. There are two ways the user can be signed out of the Code42 for Enterprise applications:
  - An administrator can deauthorize the user's devices from the Code42 console.
  - The user can sign out of the Code42 for Enterprise applications.

SSO compatibility

You can integrate Code42 with any provider that uses SAML 2.0. The following articles provide instructions for specific providers.

- [Azure](https://support.code42.com/Administrator/Cloud/Configuring/Identity_management/Configure_Azure_for_SSO_in_your_Code42_environment)
- [InCommon](https://support.code42.com/Administrator/Cloud/Configuring/Identity_management/Configure_InCommon_for_SSO_in_your_Code42_environment)
- [Microsoft Active Directory Federation Services (AD FS) 3.0](https://support.code42.com/Administrator/Cloud/Configuring/Identity_management/Configure_Microsoft_AD_FS_for_SSO_in_your_Code42_environment)
- [Okta](https://support.code42.com/Administrator/Cloud/Configuring/Identity_management/Configure_Okta_for_SSO_in_your_Code42_environment)
- [OneLogin](https://support.code42.com/Administrator/Cloud/Configuring/Identity_management/Configure_OneLogin_for_SSO_in_your_Code42_environment)
- [PingOne](https://support.code42.com/Administrator/Cloud/Configuring/Identity_management/Configure_PingOne_for_SSO_in_your_Code42_environment)
- [Shibboleth](https://support.code42.com/Administrator/Cloud/Configuring/Identity_management/Configure_Shibboleth_for_SSO_in_your_Code42_environment)
Use metadata URL for federations
Code42 cloud environments do not support uploading an XML file for federations. Use the metadata URL to configure the federation instead.

SAML 2.0 algorithms

The following SAML 2.0 algorithms are still allowed only when used in properties of the identity provider:

- RSA/MD5 for digital signatures:
  \[\text{SignatureConstants.ALGO_ID_SIGNATURE_NOT_RECOMMENDED_RSA_MD5}\]
- MD5 for HMAC:
  \[\text{SignatureConstants.ALGO_ID_MAC_HMAC_NOT_RECOMMENDED_MD5}\]

If your identity provider is up to date and configured appropriately, these deprecated methods are not provided. However, these deprecated algorithms are permitted if your identity provider still provides these methods for signing its tokens.

External resources

Wikipedia:

- SAML 2.0 (http://en.wikipedia.org/wiki/SAML_2.0)

Related topics

- Identity management (https://support.code42.com/Administrator/Cloud/Configuring/Identity_management)
- Identity management reference (https://support.code42.com/Administrator/Cloud/Code42_console_reference/Identity_management_reference)
- Introduction to SCIM provisioning (https://support.code42.com/Administrator/Cloud/Configuring/Introduction_to_SCIM_provisioning)