Evidian

DirX Access V8.10

Trusted Collaboration

Identity Federation and Access Management for the Connected World

Everything and everyone is always online and securing access to applications or devices provided either as on- and off-premise services or from the cloud has never been more important.

Businesses and government agencies are accelerating the formation of online partnerships to respond quickly to potential revenue opportunities, outsource non-core functions, and deliver the widest variety of services to their users.

To improve operational efficiency and respond to user demand, they continue to put more and more critical data and applications online for information sharing and self-service by consumers, mobile employees, channel partners and suppliers.

Cloud adoption has soared as it has proved to offer great economies of scale for many organizations by providing a lower-cost, flexible way to use applications and services.

Meanwhile, people have come to expect the online services they use to be always-available, on-demand one-stop shopping experiences accessible through a single login and providing them with the same look and feel no matter what business they are transacting. With the recent news of massive security breaches of online service databases and the rise in phishing, spoofing, and other fraudulent online activities, users are also beginning to worry that they are giving up too much of their critical identity information to too many Web sites.

While users may want to have a few different identities to protect their privacy, creating and maintaining a one-to-one identity relationship with each online service provider is a tedious chore that can lead to poor access credentials.

Building a business-agile virtual enterprise using on-premise applications and private and public cloud or software as service offerings involves numerous security challenges to provide end-to-end security. The emergence of cloud, mobile and social computing has heightened the need for strengthened access controls to ensure compliance with organization authentication and authorization policies. Partners must share or integrate their identity data, but they must do it without overloading their IT administration or inadvertently creating security holes. To maximize user satisfaction, they must provide for secure, seamless transactions between services offered by disparate sites in different security domains, and these transactions must be completely auditable from beginning to end to prove regulatory compliance. To improve the user experience and ease the user login burden, partners must offer single sign-on (SSO) capabilities to applications and services hosted internally or in the cloud. They must also provide rapid onboarding of new users to cloud services to avoid the daunting task of manually and individually provisioning and managing users in each software as a service (SaaS) directory.

Partners also need to consider security models for online user transactions that move collection and control of identity information away from online service providers and into the hands of their users and assign the management of this data to online identity providers.

Users of Web or cloud services often share personal and sensitive information. This is associated with an increasing risk of potential security and privacy issues. Once a user has submitted such information, he has only limited ability to control access to such information. To alleviate this problem, there is a clear need for new approaches and methods, to allow users to manage access to their Web resources and data.

Next Generation Identity Federation and Access Management with DirX Access

These challenges are driving the design and deployment of new security models for access management. Identity federation and secure Web services are joining authentication, authorization, audit, and Web SSO as essential capabilities for protecting Web resources against unauthorized use in a flexible way.

DirX Access is a comprehensive access management, identity federation, and Web services security solution protecting resources against unauthorized use. DirX Access:

- Provides for the consistent enforcement of business security policies through external, centralized, policy-based authentication and authorization services.
- Enhances Web user experience through local and federated single sign-on (SSO).
- Secures eGovernment and eBusiness initiatives and provides seamless integration with business and organizational partners through identity federation.
- Protects access to Web applications and devices with authentication and authorization services, both on the premises and in the cloud.
- Supports versatile authorization scenarios including user-managed access.
- Decouples security management such as authentication and authorization from application logic and ensures consistent, fine-grained entitlement management across multiple applications and services.
- Enables enterprises and service providers to deploy strong authentication solutions that reduce reliance on passwords.
- Supports regulatory compliance with audit functionality, both within and across security domains.

Authentication, Authorization and Audit – Core Functionality for Access Management

Authentication is the process of verifying the identity of a user requesting a service or a resource, while authorization is the process of verifying that an authenticated user has the right to access a requested service or resource. Authentication and authorization answer the questions "Who are you?" and "What are you entitled to do?"

Authentication and authorization address the real-time enforcement of enterprise security policies, while audit automatically records these transactions and stores these records securely for later compilation in reports to provide analytical insight and transparency in the identity and access management processes.

The processes and technologies used to manage the users and their life cycles are referred to as identity management. The set of processes and technologies to manage, deploy, enforce and audit access control policies across multiple enterprise applications, services, and systems is referred to as authorization or entitlement management.

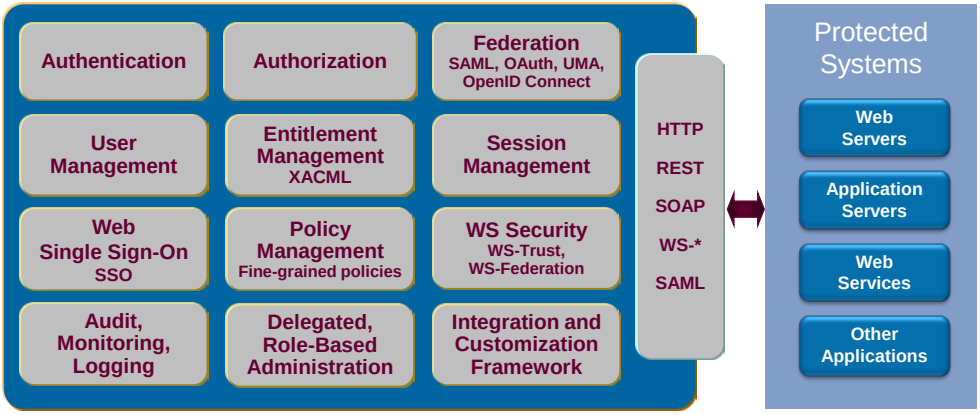

Figure 1: DirX Access Functionality

Authentication

With DirX Access, authentication is provided as an external, central service that supports a variety of well-known authentication methods, such as passwords, X.509 certificates, FIDO-based authentication, Integrated Windows Authentication, smart cards, HTML forms, one-time-password (OTP) tokens, biometrics and call back/out-of-band authentication. Administrators can apply the method that best matches the security requirements of each individual application or resource without rewriting or even touching the application. Decoupling authentication from the application or resource allows the authentication service to scale easily – administrators can add new authentication methods without affecting the applications that depend on the service.

Centralized authentication services also enable SSO. Users present their login credentials once, and are then allowed to access all applications and resources within the enterprise security domain for which they are authorized without having to re-authenticate/log in again.

Finally, the DirX Access central authentication service allows authentication management to be concentrated in one configurable component. Because an external, central service bypasses the need for per-application authentication, users no longer need to keep track of multiple login credentials, and administrators no longer need to maintain and support redundant authentication mechanisms.

A not-yet-authenticated user's interaction with a system protected by DirX Access leads to an initial user authentication process. Subsequent interactions employ the single sign-on mechanism. Risk-based authentication is applied in both mentioned cases enabling to ask the user about additional authentication, when strengthening of the assurance about user's identity is necessary. Propagation of the successful authentication state to interconnected applications is performed by the identity federation.

Initial User Authentication

Initial authentication refers to the first time a user authenticates against the system. It is based on a user account established in DirX Access and uses standards-based initial authentication mechanisms such as:

- SSL/TLS client authentication through X.509 certificates including path validation, OCSP and CRL support
- Username/password authentication via HTTP basic or HTML form
- Second-channel OTP mechanism enabling mobile push, SMS, and e-mail-based authentication via HTML form
- Standardized OTP algorithms IETF RFC 4226 (HOTP) and IETF RFC 6238 (TOTP) via HTML form
- Integrated Windows Authentication (IWA) using the SPNEGO, Kerberos and NTLM authentication protocols via HTTP
- W3C WebAuthentication based on FIDO2 including authentication with Microsoft Windows Hello
- FIDO U2F (Universal 2nd Factor)
- FIDO UAF (Universal Authentication Framework)

DirX Access enables to strengthen the authentication process by combining two or more authentication methods sequentially. This represents a reasonable way of achieving a multi-factor authentication within simple deployments; for example, combining username/password with additional verification via OTP values or username/password plus an external validation. The combination mechanism provides a conditional configuration of the authentication method sequence, i.e., failure in the first method may lead either to an overall authentication failure or to invocation of a different method. Hence, DirX Access is able to provide more sophisticated scenarios, such as account locking prevention – after three unsuccessful tries with username/password the X.509 certificate authentication is prompted, preventing from an accidental account lock and a possible costly unlocking.

DirX Access can validate the user credentials internally by performing the authentication itself (the default), or it can externalize the validation task to an external validation service.

Externalizing validation is open to various algorithms and an interface for third-party verifiers supports token-type authentication credentials;

for example, SAP logon tickets.

DirX Access can easily integrate existing authentication authorities and leverage the existing authentication infrastructure.

Administrators can define the preferred authentication method to use for each distinct resource of Web and Web services applications or the DirX Access-protected resources of other applications. In this way, administrators can easily provide the most appropriate security level for each individual resource.

Administrators can use the DirX Access Server to assign a ranking to authentication methods. This ranking, provided by assurance levels, indicates how secure the authentication methods are relative to each other on a numeric scale. Assurance levels can be used as conditions for authorization. For example, a critical resource can be assigned a policy with an assurance level condition of 4, requiring that users be authenticated using only the most secure methods in order to achieve access. Assurance levels are defined in NIST Special Publication 800-63.

DirX Access provides step-up authentication to request a re-authentication using a stronger authentication mechanism when accessing a more critical resource.

Risk-based Authentication

With risk-based authentication, access to resources is secured by means of risk analysis. The result of the risk analysis determines the minimum strength of the authentication method to be used for accessing the resource. If the user is already authenticated with an authentication method of the required strength, access to the resource is granted; otherwise the use of a stronger authentication method is enforced. The risk analysis assesses both user- and context-specific data.

The risk analysis is based on two concepts:

- Evaluating predefined static conditions
- Taking into account history and/or contextual data

The DirX Access data collector collects and stores all parameters implying risk with the user's account. The collected data are further updated after each authentication event. The authentication process leverages this data to run a statistical analysis and to analyse the behavioural patterns of each authenticated user.

Risk-based authentication combines both assurance level and risk level. Risk level is a means to estimate potential threat from an access request. A resource within DirX Access is typically protected by an authentication method having some assurance level representing the level of its protection. Both assurance level and risk level are used to select an appropriate authentication method eliminating potential risk.

In order to put this mechanism at work so called risk-based conditions are used to recognize threats. Following parameters can be configured in risk conditions:

- Resource sensitivity
- IP address ranges
- Time range e.g. usual working hours of a company, 6a.m.-7p.m., Mo-Fri
- HTTP protocol header properties, such as type of Web browser
- Custom conditions implemented as a plugin (callout), e.g. a callout to a third-party geolocation service which resolves a geographic location from an IP address
- Number of consecutive login failures
- Login interval i.e. length of time period between two login actions
- User context to detect unusual behaviour of the authenticating user i.e. various data bound to the user's account collected by the DirX Access RBA Data Collector for statistical computations, for example an unusual IP address the user tries to authenticate from.

Single Sign-On and Session Management

Once successfully authenticated, users do not have to re-authenticate themselves when accessing other DirX Access-protected resources (unless a resource explicitly requires step-up authentication) on arbitrary servers within the same domain. The authentication state of users is securely exchanged via HTTP cookie headers or URI rewriting.

DirX Access manages security sessions by maintaining information on authenticated, assured user identities. This information comprises authentication method, authentication time, authentication credentials and other parameters specific to this login event. DirX Access provides an interface for plug-ins that fetch subject attributes from additional third-party sources and enrich the session information using these attributes.

In addition, environmental information can be handled in a configurable way in authenticated subject representations; for example, solution-specific security environments like information on network trust level and device types can be provided.

DirX Access creates a new security session for each successful login. A security session is established between Web browsers and a DirX Access Server using a session identifier that refers to the assured user identity information in the cache.

An existing security session is terminated by an explicit logout (initiated by the user), by session time-out, by idle time-out, or by shutdown of all DirX Access Servers in a cluster. In all cases, the assured user information in the cache is invalidated and can therefore no longer be referenced by Web browsers.

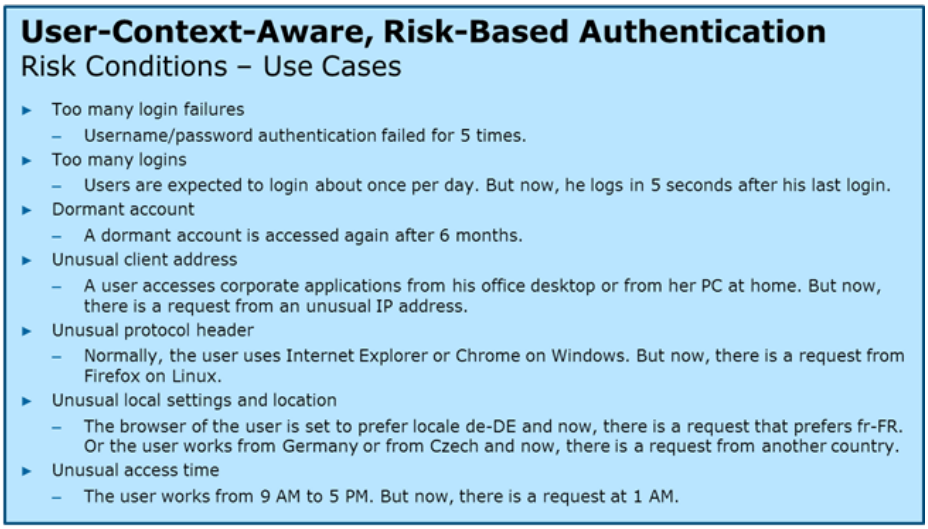

Figure 2: Risk-Based Authentication

Additional DirX Access session management features include:

- SSO session augmentation: DirX Access allows third-party applications to enrich the SSO state maintained for authenticated users with arbitrary data (opaque to DirX Access). Such data can be queried, used in authorization decisions and included in SAML assertions and audit records.
- SSO event callout interface for third-party plug-ins: this feature allows notifying third-party applications of SSO events such as user logout, session time-out or idle time-out and is especially useful when third-party applications attach application-specific session information to the SSO state in DirX Access.
- SSO session correlation across multiple browsers: this feature allows matching and merging SSO state maintained for authenticated users based on criteria such as user identifiers and user origination (for example, requestor IP addresses). This feature is optional and allows assigning the very same SSO state in DirX Access to user sessions established through different frontends.

Identity Federation and Federated Authentication

Identity federation is a set of standards and technologies that allow partner organizations to establish trust relationships regarding each other's security policies and infrastructure, and then allow or deny access to resources based on this trust.

Identity federation enables for the secure sharing of digital identities and login sessions across security domains. It facilitates secure and seamless online collaboration by providing safe access to partner resources without the need for re-authentication, and permits partners to trust and share identity information for authentication and authorization without the need to create and maintain it at each partner site.

Identity federation can

- Cut the cost and complexity of online collaboration by eliminating the need for multiple user profiles.
- Deliver a positive user experience through cross-domain SSO.
- Improve productivity by providing secure, convenient access to the resources of trusted partners.
- Interoperate with other standards-compliant federation solutions.

In DirX Access, identity federation extends the core services of authentication and authorization to the virtual enterprise.

Federated authentication is the process of transferring information regarding authentication state from an identity provider to a service provider or relying party in a different domain.

DirX Access supports federated authentication according to SAML 2.0 and OpenID Connect 1.0, that provides authentication functionality built on top of OAuth 2.0 protocol.

In contrast to the concept of initial authentication, federated authentication does not require that each user has a corresponding unique user account at the side of the service provider. Instead, the identity provider typically assigns roles to users and the service provider grants or denies access to a resource according to the roles from the security assertion statement. In this way, all information necessary to perform authentication (such as digital identity, authentication credentials, etc.) are managed locally at the identity provider but the final access decision remains in the sole responsibility of the service provider. SSO is also provided for federated authentication scenarios.

SAML-based Federation

In the case of SAML-based federation, DirX Access uses Security Assertion Markup Language (SAML) assertions to represent identities in federated transactions.

DirX Access supports both SAML 2.0 federation scenarios:

- Service provider-initiated: In this case, the user attempts to access a resource on a federated domain without first authenticating. The remote site then redirects the user to the identity provider for authentication. If authentication is completed successfully, the user is returned transparently to the destination site for authorization and, ultimately, for access to the desired resource.
- Identity provider-initiated: In this case, the user first authenticates in the local domain and then makes a request for a service or resource located on a federated domain.

In both scenarios, DirX Access can represent the identity provider that authenticates the user, or the service provider that owns the resource and relies on the source site's authentication.

DirX Access supports the following SAML 2.0 profiles, associated message protocol flows and bindings according to the SAML 2.0 conformance requirements document:

- Web Browser SSO profile with AuthnRequest message from SP to IdP via HTTP redirect or HTTP POST binding
- Web Browser SSO profile with IdP Response message to SP via HTTP POST or HTTP artifact binding including Unsolicited Responses (IdP first)
- Identity Provider Discovery profile with cookie setter and cookie getter messages via HTTP binding
- Single Logout profile with LogoutRequest and LogoutResponse messages via HTTP redirect, HTTP POST, HTTP artifact or SOAP binding
- Artifact Resolution profile with ArtifactResolve and ArtifactResponse message via SOAP binding
- Assertion Query/Request profile with authentication query, attribute query, authorization decision query and request for assertion by identifier messages via SOAP binding
- Basic Attribute profile
- X.500/LDAP Attribute profile
- UUID Attribute profile
- XACML Attribute profile

DirX Access supports the following SAML protocols:

- Authentication request protocol
- Artifact resolution protocol
- Single logout protocol
- Assertion Query and Request Protocol with authentication query, attribute query, authorization decision query and request for assertion by identifier elements

Request/response objects can be signed (enveloped XML signature).

DirX Access supports SAML assertions with the following contents:

- Authentication statements
- Attribute statements
- Authorization decision statements

Assertion objects and protocol objects can be signed (enveloped XML signature). SAML assertions, NameIds, and attributes can be encrypted.

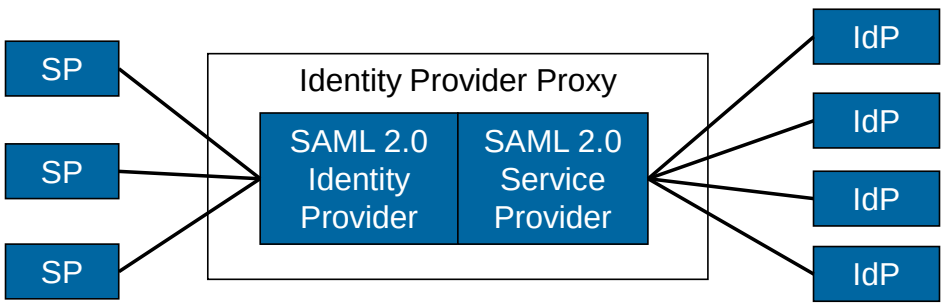

Figure 3: DirX Access Sample: Proxying IdP in a Hub/Spoke Scenario

DirX Access can include environmental information – for example, solution-specific security environments like information on network trust level and device types - into the SAML assertions.

DirX Access supports SAML metadata import and export, which addresses the mutual configuration that needs to be established between an identity provider and a service provider.

SAML Proxying

DirX Access supports SAML proxying based on the SAML 2.0 specification. In SAML proxying, identity providers can proxy an authentication request from a service provider to a different identity provider that has already authenticated the user or is capable of authenticating the user, enabling the delegation of initial user authentication in SAML Web SSO federation from a local SAML identity provider endpoint to external SAML identity provider endpoints.

A proxying identity provider is a combination of a traditional SAML authentication identity provider (implementing SAML SingleSignOnService in particular) and a traditional service provider (implementing SAML AssertionConsumerService).

With SAML proxying:

- Multiple proxying identity providers can be configured between the service provider and the actual identity provider.
- A proxying identity provider can be configured to connect to multiple identity providers and/or to multiple service providers. In this configuration, the proxying identity provider serves as a hub/bridge/gateway in a hub and spoke identity federation model, allowing for easier management of configuring trust for a large number of identity providers and service providers in a federation scenario.

Just-in-Time Provisioning

DirX Access provides a dynamic user provisioning model for SAML-based federation environments known as just-in-time (JIT) provisioning. It is targeted at (cloud) federation scenarios, where (cloud) service providers require identity data from an identity provider as a prerequisite for authorizing access to their (cloud) services.

Just-in-time provisioning is primarily a service provider-side solution. It enables a service provider to create user accounts on the fly the first time the user successfully authenticates via SAML Web SSO federation protocol with a SAML assertion issued by a trusted identity provider, It uses the attributes of incoming SAML assertions issued by a trusted identity provider to create and update user accounts in the destination application directory. Just-in-time provisioning is useful for cases where the user's identity does not need to be known in advance by the service provider, such as supply chain portals, collaborative projects and many SaaS applications.

DirX Access supports both push- and pull-based JIT provisioning scenarios:

- In the push model, the service provider creates user accounts using the identity data of the incoming SAML assertion. The DirX Access identity provider FEP supports a SAML-based, push-based JIT provisioning scenario with its own DirX Access user repository and with any SAML SP that supports JIT provisioning, for example Salesforce.com.
- In the pull model, the cloud service provider decides when and how to pull identity data from the identity provider originating the SAML assertion. The service provider first asks the identity provider for the required identity data and then creates the user account based on a union of identity data contained in the SAML assertion and received in the response to the additional requests.

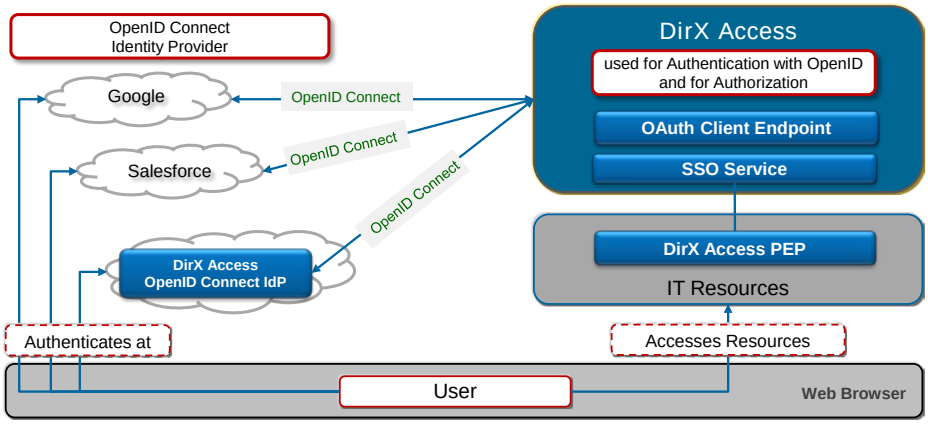

Figure 4: SSO with Social Identities based on OpenID Connect

Proven SAML 2.0 Interoperability

DirX Access passed Liberty Alliance SAML 2.0 interoperability testing in 2009. DirX Access participated in the third Liberty Interoperable™ full-matrix testing event for SAML 2.0 together with eight other products from different vendors and demonstrated that DirX Access fulfills the stringent test criteria for open, secure and privacy-respecting federated identity management.

Preconfigured SAML Service Providers

DirX Access supports preconfigured SAML service providers. This functionality allows administrators to easily establish SAML-based interoperability between the DirX Access identity provider and well-known cloud service providers such as Google Apps, Citrix ShareFile, Microsoft Office 365 and Salesforce.com with out-of-the-box configurations delivered with DirX Access. It also allows for parameterizing provider instances and creating custom templates for other preconfigured service providers and identity providers.

Identity Federation with OpenID Connect

DirX Access supports the core specification of the OpenID Connect 1.0 Standard. OpenID Connect 1.0 is a simple identity layer on top of the OAuth 2.0 protocol. From the OpenID Connect stack of standards, DirX Access supports additionally the OpenID Connect Dynamic Client Registration Protocol and OpenID Connect Discovery 1.0.

OpenID Connect 1.0 is becoming a well-established alternative for SAML protocol as it is currently supported by major service-providing market players, such as Google, Facebook, etc. Based on a more lightweight technologies (RESTful web services and JSON format), OpenID Connect, and OAuth-based protocols in general, are able to lower the communication and performance demands, while preserving the functional and security aspects of the other federation standards.

OpenID Connect 1.0 Authorization Code Flow

In OpenID Connect 1.0, a client (web application) wants to get a federated identity of an end-user from the authorization server, represented by DirX Access. The end-user has an opportunity to express its consent with the identity information being provided to the client. The specifics of the authorization code flow are basically twofold: the client securely authenticates to the authorization server, therefore, a trust is established and the authorization server can share information that shall be not disclosed to anyone else, and even the end-user is not able to learn the shared information. This flow is suitable for clients that can securely maintain their credentials, such as web applications.

OpenID Connect 1.0 Implicit Flow

Contrary to the authorization code flow, the implicit flow does not perform any client authentication. The federated identity information may be exposed to anyone with access to the end-user's browser. This flow is suitable for clients implemented in a browser using scripting languages.

OpenID Connect Dynamic Client Registration

In an environment where the web applications (clients) requiring the federated identity often changes and their overall count might easily surpass tens or hundreds, the management of the bounds between them and the authorization services might get complicated. These bounds are represented by server and client metadata documents. DirX Access publishes the metadata of all configured authorization servers in a standardized way to be accessible

by any potential client. In the opposite direction, it employs the dynamic client registration approach that enables to automate the registration of new clients or update of the metadata for the existing ones. The metadata can be manually managed by the DirX Access administrators to reflect any client specifics.

However, to achieve a state with the highest possible automation, DirX Access wires a fine-grained configurability into the registration process. At the end of the registration process, the client is assigned certain permissions at the authorization server, e.g., what identity information it can request, which flows it can perform, or what security level it achieves. The way of ensuring only the trusted clients may ask for allows permissions is achieved via a delegation of the trust. Before a client registration, an authentication process has to occur first. The authentication may be performed by any entity known to DirX Access and the permissions given to this entity determine the permissions given to the subsequently registered client. In a real world scenario, the administrator of the authorization server may contractually appoint an administrator at the customer's side. This administrator will be given rights to automatically register clients with certain permissions, according to the everchanging demands of its environment.

Described metadata management holds generally for any OAuth-based standard.

DirX Access as OpenID Connect 1.0 Client

Figure 4 shows a single sign-on scenario based on OpenID Connect where users authenticate with their social identities from systems such as Google, Salesforce etc. to access IT resources that are protected by DirX Access. In this scenario DirX Access is used for authentication via OpenID Connect and for authorizing access the IT resources in a service provider deployment.

Federation with Microsoft SharePoint

DirX Access provides identity federation with Microsoft SharePoint by supporting the WS-Federation Passive Requestor Profile for authentication in SharePoint. Other applications that support WS-Federation Passive Requestor Profile can be connected in the same way.

In this scenario, Microsoft SharePoint supports trusted Identity Provider authentication for SharePoint applications in the role of the Service Provider.

DirX Access implements the necessary functionality of both Identity Provider and Security Token Service for the authentication, using WS-Federation Passive Requestor Profile.

Identity Federation and Cloud Computing

DirX Access provides SSO for cloud-based applications or for SaaS to secure access in a cost-efficient and reliable manner. Federation standards such as SAML and OpenID Connect are being used for authentication of users to off-premise applications (SaaS or cloud-hosted).

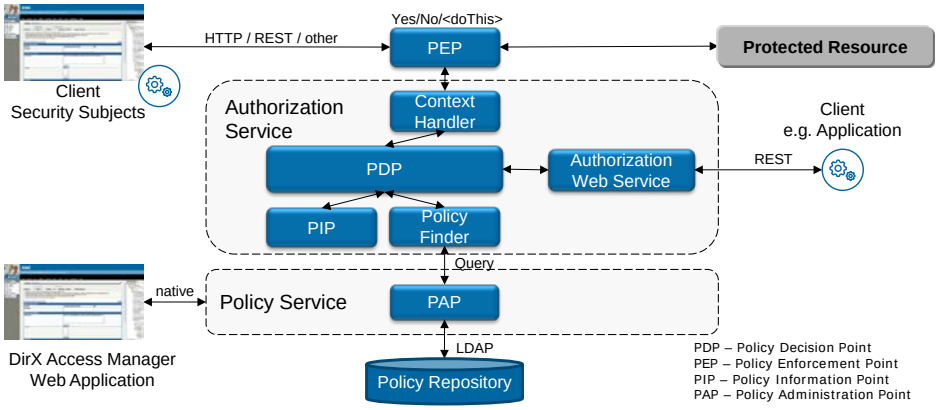

Figure 5: DirX Access XACML-based Authorization Subsystem

Inter-protocol Proxying

Thanks to the architecture of DirX Access, it is possible to proxy between different identity federation protocols, OAuth to SAML, etc. If a SAML service provider (SAML SP) / an OpenID Connect client (OIDC client) requests federated identity from the SAML identity provider (SAML IdP) / OpenID Connect authorization server (OIDC AS), either an authentication of the end-user is invoked or the SSO mechanism is used (the information on which is also based on some initial authentication). Any authentication method can be used, including the identity federation ones. During the authentication, the DirX Access role is switched from the SAML IdP / OIDC AS to SAML SP / OIDC client, a federated identity is retrieved from the third party, translated into the authenticated session which is in turn used to provide the federated identity to the originally requesting SAML SP / OIDC client.

Authorization and Entitlement Management

DirX Access offers two fundamentally different types of authorization mechanisms. For the enterprise environment, authorization is provided as an external, centralized, policy-based access control service that allows for the "up front" definition of a comprehensive set of access control policies and then grants or denies access to resources based on these policies. This approach is represented by the XACML-based authorization service.

With the raise of the IoT environment, DirX Access incorporated a service able to protect resources owned by third parties. Contrary to the centralized policy declaration, here, each owner declares policies for its own resources, while DirX Access subsequently manages the authorization process invoked by any requestor. This approach is based on the User-managed Access 2.0 standard.

Both approaches are then independent and may or may not be employed in the DirX Access installation according to the chosen configuration.

Enterprise Authorization

Based on the XACML (eXtensible Access Control Markup Language) standard from OASIS (Organization for the Advancement of Structured Information Standards), access policies can be defined according to the authorization model that best suits the environment.

For example, a role-based access control (RBAC) model defines access policies based on the roles assigned to a user, while attribute-based access control (ABAC) defines access policies based on attribute values, and with discretionary access control (DAC) access policies are defined by the owner of an object. The owner decides who is allowed to access the object and what privileges they are granted.

Policy-driven authorization has many advantages. Access control policy definition and management is performed outside of the individual application and removes the need for individual application access control logic. It makes access policy creation an initial task rather than an ongoing one, simplifies the administration of access rights to multiple Web applications and resources, and provides for the consistent application of access rights and enforcement of security policies over time.

DirX Access uses XACML 1.x/2.0/3.0 as the underlying authorization technology and supports the following authorization models:

- Arbitrary, application-defined authorization models. Any authorization model that can be expressed as valid XACML objects can be used. The actual policy content is form-free as long as policy syntax requirements (well-formedness, validity) are met.
- An RBAC authorization model, allowing or denying access to Web services and Web application resources and the resources of other applications based on the role held by the requesting user within the organization. Administrators define the business roles used in the organization and specify the resources that should be available to each role based on business needs. This authorization model is constrained by the RBAC profile of XACML; the policies that can be expressed in this authorization model need to comply with this dedicated profile.

Authorization in DirX Access is delivered by the following building blocks (see figure 5):

- PEPs (policy enforcement points), deployed as plug-ins to Web and Web application servers or other applications, process access requests, send authorization decision requests to PDPs and provide the authorization decision to their environment. Some PEPs enforce the PDP authorization decision themselves, while other PEPs just inform their environment about them.
- PDPs (policy decision points), provided as part of the DirX Access Server, render authorization decisions for access requests sent from PEPs. They base their decisions on authorization policies obtained from PAPs.
- PAPs (policy administration points) are authorization policy authorities that allow administrators to create, supply and maintain authorization policies. The PAP is represented by the policy service in a DirX Access Server. This service can be used via the DirX Access Manager (authorization policies complying to the ABAC or RBAC model) or the DirX Access Provisioning Web service (authorization policies complying to the RBAC model). Working with the PAP is subject to authorization and authentication through DirX Access.
- PIPs (policy information points) can be used to access information about the application environment that may be required in evaluating policy decisions. PIPs can also provide information on the subject or resource involved in the request.

DirX Access supports dynamic access control/authorization with the help of attribute finders, which provide the PDP with configurable information for access decisions. All information contained in the authenticated session (JAAS subjects) can be used; for example, the user LDAP attributes, SAML assertion attributes, OAuth user profile data, application and environment-specific attributes passed by the PEP to the server (client device identification, application information), etc.

In addition, DirX Access reacts in real-time to modification of user records; for example, by revoking access when user attributes are changed.

Federated Authorization

DirX Access supports the OAuth 2.0 Authorization Framework and a set of OAuth 2.0-based standards such as specifications for User-Managed Access (UMA) 2.0, OAuth 2.0 Token Introspection (RFC 7662) and OAuth 2.0 Dynamic Client Registration Protocol (RFC 7591). For a complete list, please, see the Supported Standards section.

OAuth 2.0

DirX Access supports the OAuth 2.0 Authorization Framework for authorization in identity federation scenarios. OAuth 2.0 defines a resource authorization protocol that allows resource owners to delegate resource access rights. This enables sharing resources across organizational boundaries without sharing user credentials. To support this use case, DirX Access provides both OAuth client functionality and OAuth authorization server functionality.

DirX Access can be configured independently or in conjunction with browser-based SSO for either an IdP or an SP deployment:

- In an SP deployment, the OAuth client federation endpoint client requests and uses the access token to access the protected resources
- In an IdP deployment, the OAuth server federation endpoint can be used to authenticate and provide the access token with associated user information.

User-Managed Access (UMA)

The focus on authorization in an IoT environment (and multiple resource owners in general) is represented in DirX Access by an authorization service implementing the UMA 2.0 standard. This standard enables to delegate the complexity of authorization in systems managing, e.g., medical records, bring-your-own-device policies, user-uploaded resources, etc. DirX Access plays the role of the Authorization Service (AS) in such scenario and the connected systems the role of Resource Services (RS). It provides a RESTful interface enabling the resource owner to declare any resource-related policies (typically in a form of access control list) which are subsequently used by the AS at the time any requestor asks for the protected resource. See, that all the burden is truly put at the AS as the only information the RS knows is the requested resource identifier and action.

A true power of the AS is the ability to manage resources for multiple RSs. The relationship between RS and AS is established via the OAuth-specific means, hence, DXA can automate also this process providing a perfect solution for the aaS paradigm.

DirX Access supports the following UMA-related specifications:

- User-Managed Access 2.0 Grant for OAuth 2.0 Authorization – a means for a client representing a requesting party to gain access to a protected resource asynchronously from the time a resource owner authorizes access.
- Federated Authorization for User-Managed Access 2.0 – a means for an UMA-enabled authorization server and resource server to be loosely coupled, or federated, in a resource owner context.

User Management

Initial authentication of users requires that identities are managed in an LDAP directory. This directory can be an externally managed directory with its own schema, for example, inetOrgPerson, or an LDAP directory under DirX Access administrative control, referred to as the DirX Access user repository.

In the first case, user management is performed using the corresponding external user interfaces and tools. DirX Access can use arbitrary LDAP attributes for its authentication and authorization needs.

Once authenticated, the full user record/attributes can be used in federation or authorization scenarios (issuing SAML assertions, releasing OAuth user profile data, evaluating authorization policies, etc.) using user data from arbitrary LDAP repositories.

DirX Access provides just-in-time (JIT) provisioning for federated authentication. JIT provisioning can create user accounts in the DirX Access user repository on the fly at the service provider site based on the information in the SAML token.

DirX Access provides an SPML-based provisioning Web service for provisioning user accounts and their attributes in the DirX Access user repository. Both SPML V1.0 and V2.0 are supported.

DirX Access implements the SCIM 2.0 standard and extends its basic resource structure to be able to manage the user-specific data that are, typically, authentication-related (e.g., OTP shared secrets, FIDO credentials, etc.).

For more complex tasks, such as assigning user attributes and privileges, integrating multiple directories, user databases, and application-specific repositories, it is recommended to use an identity management solution such as DirX Identity. DirX Identity also provides workflow-based user self-registration and self-management functionality as well as many other advanced identity management functionalities.

Policy Management

Policy management in DirX Access comprises functions to create, modify, delete and view authorization and authentication policies based on the XACML standard.

In DirX Access, administrative policies govern the administration of DirX Access, and business policies govern the access of users to the protected resources.

Authentication policies enforce the use of specific authentication methods for different system resources.

Authorization policies apply authorization rules controlling actions on protected resources. Fine-grained authorization policies help to define the level of granularity needed for authorization. They consider the properties of requested resources (such as security classifications) and requesting subjects (such as user names, group memberships or role assignments) to enable authorized access to resources and deny unauthorized access.

Access Tester

Authorization policies can be tested from within the DirX Access Manager Tools section. The Access tester allows the administrator to simulate any user or role based access against the actual policy to see what will happen and if the configuration has the desired outcome.

Securing Web Applications

Access management solutions were initially focused on securing access to Web applications and Web content behind eBusiness, eGovernment, and eShop portals. To this end, DirX Access WAM capabilities apply the concepts of external, central, policy-based authentication and authorization services, identity federation and SSO to provide secure, convenient, and reliable access to multiple Web applications with one authentication step.

The DirX Access PEPs that secure Web applications can be classified as protocol stack extension PEPs, agent PEPs and application PEPs. In addition, custom PEPs can be created based on the client SDK or by the DirX Access Web services.

Protocol stack extension PEPs reside in protocol stacks. The most common examples are the HTTP stack PEPs (Web PEPs) like the ones for Apache Web Server, Apache Tomcat or Microsoft Internet Information Server. They integrate with the down-stream applications that they protect mainly through header injection. Via header injection, data from various sources can be made available to applications, e.g. user- and session-related data as well as data from arbitrary LDAP repositories.

Agent PEPs make use of the extensibility interfaces of the Web Server or Application Server to protect the applications that run in these servers. The DirX Access Agent PEP for the Microsoft Internet Information Server (IIS) provides event handlers that handle the IIS authentication and authorization requests of the IIS server.

Application PEPs can be provided for applications that support standard or published interfaces. One important example is servlet applications, which can be protected individually by the DirX Access servlet filter PEP. Other examples are cloud-native applications that are running in cloud application platforms such as Cloud Foundry. These applications can be protected by the DirX Access Cloud Foundry PEP. Applications that do not provide such integration points can be protected with a custom-developed PEP built with the DirX Access Client SDK

Securing Legacy Applications

The demand for online collaboration, increased operational efficiency and 24/7 access to business resources is driving companies to bring their legacy applications online at a rapid pace.

DirX Access offers a way to secure these legacy applications with the same external, centralized, policy-based authentication and authorization services used for modern Web applications.

DirX Access supports application PEPs that integrate with specific applications and protect them. Depending on the integration mechanism, they can be classified as:

- Application source PEPs, which are custom PEPs that use Client SDK methods integrated into the sources of the application.
- Application extension PEPs, which are custom or off-the-shelf PEPs that extend applications; for example, aspect-oriented programming (AOP)-based PEPs.

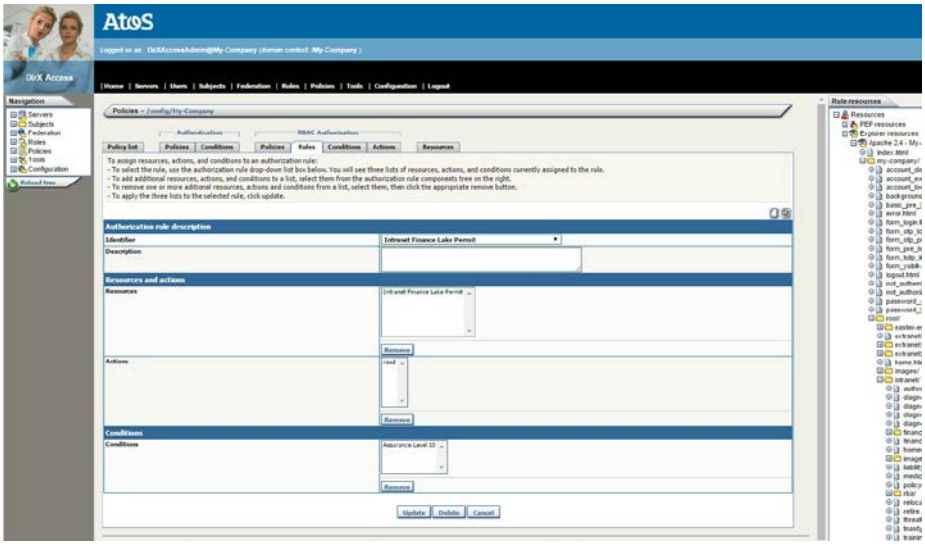

Figure 6: DirX Access Manager Sample

Legacy applications can use PEPs and PDPs as their access management system, providing an external, centralized method for controlling access to these applications rather than having to build in access management security on an application-by-application basis.

Security Web Services

The problem of externalizing and centralizing application-specific security also applies to Web services-based SOAs. When migrating applications to run as discrete Web services, how do you handle the individual (and usually unique) security logic and data that is present in each application?

DirX Access responds to this challenge by providing its security features as out-of-the box Web services for deployment in Web services-based SOAs. Businesses can then add security logic as well-defined, published Web services for use by any business process service running in a Web services based SOA.

DirX Access provides the Web services based on two technologies RESTful and SOAP WS. The RESTful WS are build according to the Open Data Protocol (OData) Version 4.0 standard issued by OASIS.

DirX Access provides the following off-the-shelf Web services:

- The Authentication and SSO Web service, which provides authentication and SSO functionality.
- The Authorization Web service, which provides authorization decision-making and testing based on the OASIS XACML standard.
- The Federation Web service, which provides an OASIS WS-Trust security token service (STS).
- The Provisioning Web service, which provides the ability to provision DirX Access with users, groups and organizational units, and to control the assignment of those objects to roles. It is based on the OASIS SPML V1.0 and V2.0 standards.
- The Configuration Web service, which is used to configure the DirX Access system.

Administration

Administrative responsibilities reflect business structures so that companies can place the management of users, groups, and policies for access to resources with someone close to the demands of a particular business line. DirX Access provides methods for flexible, secure delegation of administrative responsibilities to respond to temporary changes in personnel and shifts in organizations and processes and support the business-agile enterprise. DirX Access provides Web-based administration tools that permit administrative activities to run in parallel for fast, efficient deployment of access policies across the virtual enterprise. If there is a need for more complex identity management and provisioning activities, DirX Access can be seamlessly integrated with DirX Identity or cooperate with other identity management solutions.

Access control for administration uses the same authentication and authorization mechanisms as access control for protected organizational resources.

DirX Access administration is performed through the DirX Access Manager (see figure 6), a Web-based administration tool that allows administrators to perform a variety of tasks, such as:

- Creating business roles.
- Creating authentication policies using a resource tree.
- Configuring risk-based conditions and associated data collectors.
- Creating authorization rules and policies using a resource tree.
- Setting authorization conditions, such as the time of day, authentication method, assurance level, or the IP range required for access.
- Assigning policies to roles.
- Assigning users, groups and organizational units to roles.
- Configuring XACML ABAC policies.
- Configuring the internal representation of authenticated subjects.
- Configuring the SAML assertions of authenticated subjects.
- Configuring federation.
- Configuring servers.
- Configuring PDPs.
- Configuring PEPs.

Additional administrative applications provided by DirX Access include the DirX Access Console: a command-line based standalone application built on top of Open Services Gateway initiative (OSGi) shell enabling deployment of DirX Access components and managing policy and configuration data.

Multi-Tenancy

To support multi-tenancy, multiple instances of DirX Access can be deployed. Each instance represents a tenant with specific configuration separated from other tenants. This allows serving multiple client organizations (tenants) with one single installation of the software. DirX Access provides means to create additional instances / tenants.

Audit

In order to prove compliance with an increasing number and complexity of business and privacy regulations, the DirX Access Audit service provides complete transaction accountability across the virtual enterprise. The system:

- Audits transactions both within and across security domains.
- Logs all security events for proof of activity; for example, the result of authentication and authorization requests or password and policy changes.

All authorization requests for a given transaction can be correlated to previous authentication events; therefore all transactions can be traced back to their origins. This design applies to all relevant system features (authorization, authentication, identity federation, user management, policy and configuration management).

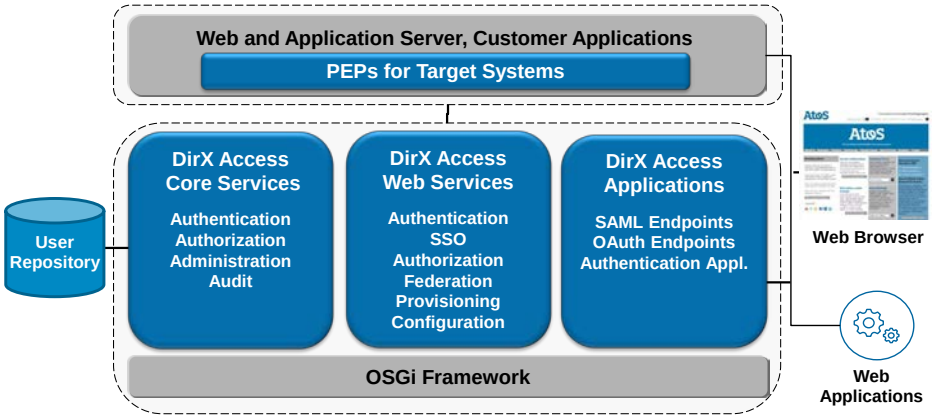

Figure 7: DirX Access Architecture and Integration into Applications

Audit data generated by the DirX Access audit service corresponds directly to the actions of identifiable and authenticated users. Actions recorded for each user include:

- Authentication (who, when, how)
- Authorization (who, when, for what)
- Session management (for example, session lifetime, idle timeout)
- Account and password management (for example, changes, expiration time reached)
- Policy management (for example, role, authentication and authorization policy creation and modification)
- User management (for example, user and group creation and modification)
- Configuration events

DirX Access provides an audit externalization interface that supports custom implementations of audit events processing via plug-ins. The following implementations are provided with the product:

- An implementation based on Log4J which uses Log4J appenders (for example, for console, file, database, syslog, and other items) to process DirX Access audit events. This is the default audit plug-in provided by DirX Access.
- Out-of-the-box integration to the DirX Audit product.

DirX Audit can be used for centralized, secure storage, analysis, correlation and review of identity- and access-related audit logs and for creating reports. DirX Audit is part of the DirX product suite and can be ordered separately.

DirX Access provides for the export of its deployment, configuration, policies and user data as XML files through various Web services. This feature can be customized and transformed, for example, by XSLT to custom reports.

Logging

DirX Access logging records internal system operations for problem diagnostics and debugging. The amount of information each server generates can be controlled by restricting its logs to a given level.

System Monitoring

DirX Access Services and Web Applications containers support monitoring via Java MBeans. MBeans provide a Java-platform standard method for monitoring a software system. MBeans are handled by Java technology called Java Management Extensions (JMX).

DirX Access MBeans provide live data about the status of the containers as well as usage statistics, such as, among others,

- Number of authentication requests
- Number of authorization requests
- Number of SAML assertion issuance requests

Nagios Support

MBeans published by DirX Access components can be used by a variety of monitoring tools and systems. Especially, DirX Access allows integration with Nagios, one of the widespread monitoring systems via the third-party tools JNRPE, check_nrpe and check_JMX, which provide quite straight-forward means for monitoring Java processes using MBeans.

DirX Access Architecture

Many third-party business applications in the enterprise can use DirX Access for access management and enforcement; for example, portals, Web servers, application servers, and other applications. They can be broadly structured according to the Client, Web, Application, and Data tier. DirX Access typically integrates into the Web and Application tiers. Depending on the technology of these integration tiers, one may use the DirX Access off-the-shelf capabilities (PEPs and federation endpoints).

DirX Access integrates with the applications it protects through capturing agents (referred to as PEPs - policy enforcement points) deployed as plug-ins to Web and Web application servers or other applications. They act as DirX Access Server clients, mediating the authentication and authorization process, enforcing the access decisions of the server and providing the user browser and the downstream applications with

session and state information. This setup also includes reverse-proxy configurations.

The DirX Access Server provides the core security services—authentication, authorization (PDP - policy decision point), SSO, federation, policy, configuration and others— to the PEPs, mediates the access to the LDAP repositories, exposes the services as Web services and/or federation services to third-parties and provides the Web-tier logic for the Web-based management interfaces.

The PDP access decisions are driven by the XACML-based authorization policies, which are managed by the policy administration point (PAP). The PAP is implemented by the policy service in the server. It can be used through a Web-based GUI provided with the DirX Access Manager.

The policy information points (PIPs) are used to access information about the application environment or on the subject or resource involved in the request.

DirX Access uses LDAP directory servers to store user, configuration and policy data.

The DirX Access architecture can be structured in tiers as follows:

- Client tier:
  - DirX Access PEPs
  - DirX Access applications such as federation endpoints, Authentication Application, DirX Access Manager, and Web services
- Server tier: DirX Access Servers providing security services
- Data tier: Directory servers used to store user, configuration and policy data for DirX Access.

DirX Access services and applications are uniformly deployed in ready-to-use OSGi-based containers. This configuration allows for network separation when deploying services in a protected network and applications in a DMZ.

Figure 7 presents the DirX Access architecture and its integration points in existing applications from a high-level component perspective.

Client Tier: DirX Access Policy Enforcement Points

PEPs are plug-in components that operate as a DirX Access client and that provide policy enforcement services (especially authorization and authentication). They process requests for resources and services, query the DirX Access Server for authorization and authentication, and provide the decisions back to their environment.

For integration purposes, DirX Access also supports the configuration of arbitrary LDAP user objects that are injected into the HTTP header for further use of the secured application.

Client Tier: DirX Access Applications

DirX Access provides the following categories of off-the-shelf Web applications: DirX Access Manager, DirX Access Authentication Application and federation applications.

DirX Access Manager provides an intuitive Web-based interface that allows full or delegated administrators to manage the system (for details, see the Administration section of this document).

DirX Access Authentication Application is a DirX Access component that performs initial user authentication on behalf of DirX Access PEP and FEP components. The layout of the user interface and the flow of authentication are customizable. The Authentication Application allows for context-aware authentication based on, e.g., internal vs. external IP address ranges to minimize risks.

DirX Access federation applications provide endpoints for federated identity management:

- The SAML service provider federation endpoint (SP FEP) provides a federation endpoint for SAML service providers
- The SAML identity provider federation endpoint (IdP FEP) provides a federation endpoint for SAML identity providers.
- The SAML identity provider federation endpoint supports SuisseId and the SAML service provider federation endpoint provides support for SuisseID-enabled identity providers. SuisseID is a national ID infrastructure project in Switzerland. SuisseID follows a user-centric identity management approach and extends the SAML 2.0 specification by user-centric identity management features.
- The OAuth server federation endpoint represents the authorization server side of the OAuth communication. An authorization endpoint is used by the client to obtain authorization from the resource owner via user-agent redirection. A token endpoint is used by the client to exchange an authorization grant for an access token, typically with client authentication. A user-info endpoint is used by the client to exchange the access token for identity data about the authenticated entity. The metadata and client registration endpoint are used for metadata registration and exchange. And the policy management endpoints are used for managing the policies for resources stored at connected resource servers by the resource owners (employed by the UMA 2.0 authorization process).
- The OAuth client federation endpoint represents the client side of the OAuth communication and is able to create a session in DirX Access. The OAuth client federation endpoint works with any OAuth 2.0 server such as Google, Facebook, etc.

DirX Access Web Services

DirX Access provides the following off-the-shelf Web services:

- Authentication and SSO Web service
- Authorization Web service
- Federation Web service
- Provisioning Web service
- Configuration Web service

For details, see the Security Web Services section in this document.

Server Tier: DirX Access Core Services

The DirX Access services provide the core functionality of the product, including authentication and SSO, authorization, administration and audit services. This functionality is realized using SOA principles and consists of core and supporting services.

The DirX Access Server services are used by native communications as well as through bundled Web applications and Web services.

This architecture allows a distributed and secure installation of DirX Access components:

- The server tier components are bundled in a services container and can be installed in a protected network and protected by an additional firewall.
- The client tier components are bundled in a Web Applications container and can be installed together with the Web Servers with DirX Access PEPs in a demilitarized zone (DMZ) and protected by an additional firewall.

Data Tier: Directory Server

DirX Access can use two different LDAP directory servers in parallel, one for user and one for policy/configuration data.

Any standard LDAP directory with a schema suitable for user management (for example, InetOrgPerson object class) can serve as a user repository.

DirX Access can supplement the user record obtained from the user directory with information from other stores using standard and/or custom-build attribute finders, which are functionally comparable with virtual directories.

Policy data includes the following elements:

- Authentication policies
- Authorization policies (RBAC/ABAC), including rule, condition and action components

Configuration data includes the following elements:

- Authentication methods
- Server configurations
- Policy enforcement point configurations
- Federation endpoint configurations
- Centralized component configuration parameters such as user directory settings, templates to construct or interpret SAML assertions and other parameters

The LDAP directory server used for configuration data is also used for storing the user-specific data generated by DirX Access, RBA data, and several types of credentials, FIDO-related credentials, OTP credentials, etc. If any of these use cases are configured to take place, the directory server shall be handled accordingly from the security perspective.

User management applications can be integrated with the DirX Access Server via its provisioning interface.

DirX Access can also use various LDAP servers that are not part of the DirX Access product delivery.

Reliability, High Availability, and Scalability

To achieve maximum availability and failover security as well as scalability, multiple, redundant DirX Access Servers can be configured. DirX Access clients, for example, PEPs or federation endpoints, can perform load-balanced access to the servers. Therefore, the DirX Access clients keep an internal connection pool and a health index of the multiple servers. Load balancing is then handled internally using that connection pool. To complete the fail-safe setup, the DirX Access Server supports primary and secondary directory configuration for failover deployments.

DirX Access uses a set of sophisticated mechanisms to recover from network component failures and prevent down time for users, including:

- A distributed cache, allowing multiple DirX Access Servers to share security objects and configurations.
- Load balancing between DirX Access Servers based on a round-robin scheduling algorithm and server stickiness.
- A state-of-the-art operation recovery process using retries, retry intervals and error thresholds.

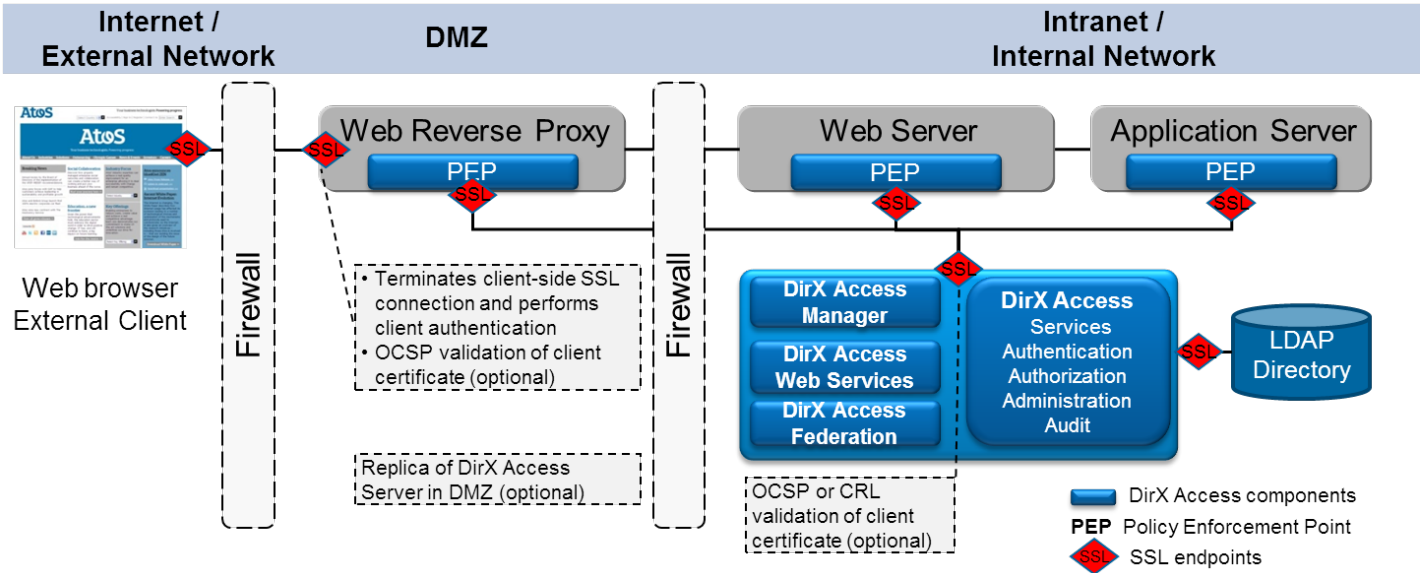

Figure 8: DirX Access Deployment Example

Client Behaviour

Every application acting as a DirX Access client in the system is expected to have a corresponding entry in the DirX Access configuration store. When a DirX Access client initiates communication with a DirX Access Server, it must provide an instance name. The configuration service uses this instance name to determine the appropriate configuration entry in the configuration store. This includes the addresses of all DirX Access Servers in a network or a dedicated subset of these servers associated with this client in a specific way (along with more information, such as the maximum number of connections this client may initiate).

When an application sends requests to DirX Access Servers, the underlying DirX Access client transparently performs load-balancing over the configured DirX Access Servers. It also automatically creates server connections as required to process its message traffic, up to its configured maximum. When this threshold is reached, the DirX Access client continues to process its message traffic using the available servers and connections.

Session State Sharing

Multiple DirX Access Servers in a network share a synchronized cache containing session objects, policies, and configuration attributes. The shared cache allows clients (PEPs, FEPs) to send requests to arbitrary servers in a network and also improves performance by reducing the number of read requests to the directory server.

This cache also enhances reliability and failover mechanisms, since a session object created by one server and stored in the policy cache can be reused by any other server. Whenever one server fails, the other servers can transparently pick up the session without compromising the security of the transaction.

Whenever a user authenticates successfully, a new session is initiated, which creates a new subject in the cache system. The subject contains the session token as a principal, along with the list of attributes associated with the authenticated user (for example, role assignments) and additional information. Since the cache is distributed, session states are immediately made available to all other servers. The cache enables to configure its parameters to perform either in a fully replicated mode, in which all sessions are duplicated and synchronized on all servers, or in an partitioned mode, in which there is a primary server storing the session and a configurable number of backup servers for the case the primary server becomes unavailable. The partitioned mode enables to find a right balance between robustness and improved performance as each server may store just a fraction of all the sessions. The server stickiness assures this approach is not slowed down by additional requests between the servers.

LDAP Failover

Access to the LDAP directory configuration/policy and user repository is crucial for DirX Access and is ensured by switching to a secondary directory server instance if the primary server is unavailable. If a DirX Access Server receives a timeout in response to any directory operation, it will try using the secondary instance instead.

Supported Standards

DirX Access supports the relevant standards, protocols, and security frameworks to provide its security functionality and services:

For authorization and privacy, DirX Access supports XACML 1.x/2.0/3.0, XACML 3.0 Multiple Decision Profile Version 1.0, XACML SAML Profile Version 2.0, SAML 1.x/2.0, OAuth 2.0 and RBAC.

DirX Access successfully passed the Liberty Alliance SAML 2.0 interoperability test in 2009 when it participated in the third Liberty Interoperable™ full-matrix testing event for SAML 2.0.

For initial user authentication in Web environments, DirX Access supports SSL/TLS, HTTP Basic, HTML Form-based authentication with username/password, one-time-passwords based on IETF RFCs, 4226 and 6238 and FIDO U2F, UAF, W3C WebAuthentication (based on FIDO2 input).

For intra-domain SSO in Web environments, DirX Access supports Integrated Windows Authentication (SPNEGO/ Kerberos, NTLM), authenticated subject identifiers transferred via HTTP cookie headers, and URL rewriting.

For cross-domain SSO and identity federation in Web environments, DirX Access supports SAML 1.x/2.0 especially SAML Web-SSO profiles, and WS-Federation Passive Requestor Profile Version 1.0

For cross-domain SSO and identity federation in Web services environments, DirX Access supports WS-Trust.

The implementation of OAuth 2.0 Authorization Framework together with the following extensions enables DirX Access to be employed in almost any plausible federation scenario:

- The OAuth 2.0 Authorization Framework (RFC6749)
- The OAuth 2.0 Authorization Framework: Bearer Token (RFC6750)
- OAuth 2.0 Authorization Server Metadata, https://tools.ietf.org/html/draft-ietf-oauth-discovery-06
- OpenID Connect 1.0
- OpenID Connect Discovery 1.0
- OAuth 2.0 Token Revocation (RFC7009)
- OAuth 2.0 Token Introspection (RFC7662)
- OAuth 2.0 Dynamic Client Registration Protocol (RFC7591)
- OpenID Connect Dynamic Client Registration Protocol
- OAuth 2.0 Resource Registration
- Federated Authorization for User-Managed Access 2.0
- User-Managed Access 2.0 Grant for OAuth 2.0 Authorization

For user-specific data provisioning, DirX Access supports SCIM 2.0, and SPML 1.0 and 2.0.

For secure communication, DirX Access supports SSL/TLS and WS-* security.

For object security, DirX Access supports XML signature.

For key management, DirX Access supports PKCS and X.509/PKIX.

For communications, DirX Access supports HTTP, RESTful, SOAP and WS-*.

For persistence and provisioning, DirX Access supports LDAP, DSML and SPML.

In Java environments, DirX Access supports JAAS, JACC, JCA/JCE, JGSS, and JSSE.

DirX Access supports both IPv4 and IPv6 Internet Protocol.

Other DirX Products

The following products also belong to the DirX product suite and can be ordered separately; DirX provides the basis for totally integrated identity and access management:

DirX Directory provides a standards-compliant, high-performance, highly available, highly reliable and secure LDAP and X.500 directory server and LDAP Proxy with very high linear scalability. DirX Directory can act as the identity store for employees, customers, trading partners, subscribers, and other e-business entities.

DirX Identity provides a comprehensive, process-driven, customizable, cloud-ready, scalable and highly-available identity management solution for enterprises and organizations. It delivers risk-based identity and access governance functionality seamlessly integrated with automated provisioning. Features include lifecycle management for users and roles, cross-platform and rule-based provisioning in real time, Web-based user self-service and delegated administration, request workflows, access certification, password management, metadirectory and auditing and reporting.

DirX Audit provides auditors, security compliance officers and administrators with analytical insight and transparency for identity and access. Based on historical identity data and recorded events from the identity and access management processes, DirX Audit allows answering the "what, when, where, who and why" questions of user access and entitlements. DirX Audit features historical views and reports on identity data, a graphical dashboard, a monitor for identity-related events and job management for report generation. With its analytical features, DirX Audit helps enterprises and organizations to ensure sustainable compliance and provide business intelligence for the identity and access management processes.

System Requirements for DirX Access V8.10

Supported Policy Enforcement Points and Client SDK:

The following combinations are supported. Other PEPs may be available on request.

	Microsoft Windows Server 2016 / 2019	Red Hat Enterprise Linux 7	SUSE Linux Enterprise Server 12
Web Server PEPs			
Apache httpd 2.4	Yes	Yes	Yes
Reverse proxy (based on Apache httpd 2.4)	Yes	Yes	Yes
Apache Tomcat 8.5/9.0	Yes	Yes	Yes
Eclipse Jetty 8/9	Yes	Yes	Yes
Microsoft IIS	Yes	-	-
Servlet and Application-specific PEPs			
Servlet filter e.g. Tomcat, Jetty, etc.	Yes	Yes	Yes
Cloud Foundry	-[1)]	-[1)]	-[1)]
Client SDK support (Legacy application PEPs)			
DirX Access Client SDK for Java 1.6 or higher	Yes	Yes	Yes

[1)] The Cloud Foundry PEP can be deployed into an existing Cloud Foundry Provider environment

System Requirements for DirX Access V8.10

Hardware

- Intel server platform for Microsoft Windows Server 2016, Microsoft Windows Server 2019, Linux

Memory Requirements:

Main memory:	minimum 4 GB
Disk Space:	minimum 1 GB plus disk space for data

Software

DirX Access Server Support

DirX Access Server as a Java application is supported on the following platforms with latest patches/service packs for the selected platform:

- Microsoft Windows Server 2016 (x86-64)
- Microsoft Windows Server 2019 (x86-64)
- Red Hat Enterprise Linux 7 (x86-64)
- SUSE Linux Enterprise Server 12 (x86-64)

- Java SE Runtime Environment (JRE) 11 for the selected operating system

Virtual Machine Support:

VMWare ESXi, in combination with the guest operating systems listed above and that are supported by VMWare ESXi.

Supported LDAP Directories for Configuration/Policy Data:

DirX Access supports the following LDAP directories (others on request):

- DirX Directory V8.5/V8.6/V8.7
- Microsoft Windows Server 2016/2019 Active Directory / Active Directory Lightweight Directory Services (AD LDS)

Supported Directories for User Data:

Arbitrary LDAPv3-compliant directory servers with user accounts based on the InetOrgPerson object class

Browser Support for the DirX Access Manager and Deployment Manager

- Microsoft Internet Explorer 11
- Microsoft Edge
- Firefox 71 or newer
- Google Chrome 78 or newer

For Nagios Integration

- Nagios Core Version 4.0.8
- JNRPE Server, version 2.0.5
- JNRPE plugins, version 2.0.3

Supported PEPs and Application Servers:

These components are listed on the previous page.

User interface

English

Documentation

All manuals are provided in English:

- Release Notes
- Introduction Guide
- Installation Guide
- Administration Guide
- Integration Guide

More information:
www.evidian.com/dirx