



OSID V3 Specifications authentication package

Version Draft 3

This specifications represent a draft for OSID V3 interface definitions. These definitions may change at any time.

Last Modified: 13 October 2008

prepared by:
Tom Coppeto
OnTapSolutions

Copyright © 2008 Massachusetts Institute of Technology

| OSID License | |
|------------------|--|
| Copyright | <p>Copyright © 2002-2008 Massachusetts Institute of Technology. All Rights Reserved.</p> <p>This Work is being provided by the copyright holder(s) subject to the following license. By obtaining, using and/or copying this Work, you agree that you have read, understand, and will comply with the following terms and conditions.</p> |
| License | <p>This Work and the information contained herein is provided on an "AS IS" basis. The Massachusetts Institute of Technology, the Open Knowledge Initiative, and THE AUTHORS DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE WORK OR THE USE OR OTHER DEALINGS IN THE WORK.</p> <p>Permission to use, copy and distribute unmodified versions of this Work, for any purpose, without fee or royalty is hereby granted, provided that you include the above copyright notice and the terms of this license on ALL copies of the Work or portions thereof.</p> <p>You may modify or create Derivatives of this Work only for your internal purposes. You shall not distribute or transfer any such Derivative of this Work to any location or to any third party. For the purposes of this license, Derivative shall mean any derivative of the Work as defined in the United States Copyright Act of 1976, such as a translation or modification.</p> <p>The export of software employing encryption technology may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting this Work.</p> |

| | |
|---------------------|---|
| Package Description | osid.authentication package |
| Interfaces | <p>osid.authentication.AuthenticationProfile osid.authentication.AuthenticationManager osid.authentication.AuthenticationProxyManager osid.authentication.AuthenticationAcquisitionSession osid.authentication.AuthenticationValidationSession osid.authentication.AgentLookupSession osid.authentication.AgentSearchSession osid.authentication.AgentAdminSession osid.authentication.AgentNotificationSession osid.authentication.KeyLookupSession osid.authentication.KeySearchSession osid.authentication.KeyAdminSession osid.authentication.Authentication osid.authentication.Agent osid.authentication.AgentRecord osid.authentication.AgentQuery osid.authentication.AgentQueryRecord osid.authentication.AgentForm osid.authentication.AgentFormRecord osid.authentication.AgentSearch osid.authentication.AgentSearchRecord osid.authentication.AgentSearchResults osid.authentication.AgentSearchResultsRecord osid.authentication.AgentReceiver osid.authentication.AgentList osid.authentication.Key osid.authentication.KeyRecord osid.authentication.KeyQuery osid.authentication.KeyQueryRecord osid.authentication.KeyForm osid.authentication.KeyFormQuery osid.authentication.KeySearch osid.authentication.KeySearchRecord osid.authentication.KeySearchResults osid.authentication.KeySearchResultsRecord osid.authentication.KeyList</p> |

| Package | osid.authentication |
|----------------|---|
| Title | Authentication Open Service Interface Definitions |
| Version | 3.0.0 |
| | <p>The Authentication OSID helps a consumer acquire and validate authentication credentials without having to manage the details of a particular authentication environment. Authentication is generally a two step process. A user wishing to authenticate acquires a set of credentials and transports those credentials to a remote peer. The remote peer then validates those credentials and determines the identity of the user represented. This process is reflected in the Authentication OSID with the definition of two OSID sessions:</p> <ul style="list-style-type: none"> • AuthenticationAcquisitionSession: session to acquire credentials from a user and serialize them for transport to a remote peer for authentication • AuthenticationValidationSession: session to receive and validate authentication credentials from a remote peer wishing to authenticate <p>The transport of authentication credentials is the responsibility of the consumer of the Authentication OSID as authentication generally supports an existing application protocol environment. A means exists to extract and supply credentials at each end. An Authentication provider may support either or both sessions, and one or more credential formats. Methods also exist to support a challenge-response type system.</p> <p>The Authentication OSID defines an Agent to represent the identity of the authenticated entity. An Agent may map to a specific authentication principal or some providers may hide multiple authentication principals behind a single Agent identity. Because principal identities tend not to be durable and persistent, consumers should always persist the Id. Relationships to other OSIDs such as Resource are made using the Id as the key. The Authentication service links to an Authorization service via the Agent Id. An Authorization service describes what actions the Agent is authorized to perform. An Agent may map to a Resource. Multiple Agents may map to a single Resource for the purpose of defining distinct security levels of assurance (although the paranoid may opt for defining a pseudo-resource for each Agent). These security levels of assurance can be identified through the Agent Type for the purposes of mapping a Resource back to one or more appropriate Agents to facilitate authorization maintenance. The Agent Type is an indicator of the strength of the authentication and although it may correlate to a specific authentication technology, coupling it too tightly to a particular technology may limit flexibility.</p> <p>The Authentication OSID can also manage Agents and associated keys. Key management sessions are defined to extract, modify and delete keys.</p> <p>Example client side authentication:</p> |

Description

```

if (manager.supportsAuthenticationAcquisition() &&
    manager.supportsAcquisitionInputType(krb5ServiceType) &&
    manager.supportsCredentialType(serialKRB5Type)) {
    AuthenticationAcquisitionSession aas =
    manager.getAuthenticationAcquisitionSession();

    // specify input parameters (interface extension)
    KRB5Service kService = new KRB5Service();
    kService.setName("host");
    kService.setInstance("server.osid.org");
    kService.setRealm("OSID.ORG");

    // get Credential (interface type)
    Authentication auth = aas.getAuthentication(kService, krb5ServiceType);
    SerializedKRB5Ticket ticket = (SerializedKRB5Ticket)
    auth.getCredential(serialKRB5Type);
    send_data_to_peer(ticket); // app specific protocol
}

```

Example server side authentication:

```

if (manager.supportsAuthenticationValidation() &&
    manager.supportsCredentialType(serialSAML2Type)) {
    AuthenticationValidationSession avs =
    manager.getAuthenticationValidationSession();

    Authentication auth = authenticate(SAML2Token, serialSAML2Type);

    if (auth.isValid()) {
        Agent agent = auth.getAgent(); // identity established
    }
}

```

Certain consumers may wish to be notified of changes within the service. Authentication supports notifications via an AgentNotificationSession.

```

if (manager.supportsAgentNotification()) {
    AgentNotificationSession ans =
    manager.getAgentNotificationSession(receiver);
    ans.registerForNewAgents();
    hangAround();
}

AgentReceiver receiver {
    newAgent(Id agentId) { print("new agent"); }
    changedAgent(Id agentId) { print("updated agent"); }
    deletedAgent(Id agentId) { print("deleted agent"); }
}

```

| Interface | osid.authentication.AuthenticationProfile | |
|-------------|---|--|
| Implements | osid.OsidProfile | |
| Description | The AuthenticationProfile describes the interoperability among authentication services. | |
| Method | supportsAuthenticationAcquisition | |
| Description | Tests if authentication acquisition is supported. Authentication acquisition is responsible for acquiring client side authentication credentials. | |
| Return | boolean | true if authentication acquisition is supported, false otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | supportsAuthenticationValidation | |
| Description | Tests if authentication validation is supported. Authentication validation verifies given authentication credentials and maps to an agent identity. | |
| Return | boolean | true if authentication validation is supported, false otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | supportsAgentLookup | |
| Description | Tests if an agent lookup service is supported. An agent lookup service defines methods to access agents. | |
| Return | boolean | true if agent lookup is supported, false otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | supportsAgentSearch | |
| Description | Tests if an agent search service is supported. | |
| Return | boolean | true if agent search is supported, false otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | supportsAgentAdmin | |
| Description | Tests if an agent administrative service is supported. | |
| Return | boolean | true if agent admin is supported, false otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | supportsAgentNotification | |
| Description | Tests if agent notification is supported. Messages may be sent when agents are created, modified, or deleted. | |
| Return | boolean | true if agent notification is supported, false otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | supportsKeyLookup | |
| Description | Tests if a key lookup service is supported. A key lookup service defines methods to access keys. | |
| Return | boolean | true if key lookup is supported, false otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | supportsKeyAdmin | |
| Description | Tests if a key administrative service is supported. | |
| Return | boolean | true if key admin is supported, false otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | supportsChallenge | |
| Description | Tests if this authentication service supports a challenge-response mechanism where credential validation service must implement a means to generate challenge data. | |
| Return | boolean | true if this is a challenge-response system, false otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | getChallengeTypes | |
| Description | Gets the supported challenge types. | |
| Return | osid.type.TypeList | a list containing the supported challenge types |
| Compliance | mandatory | This method must be implemented. |

| Method | supportsChallengeType | | | | |
|--------------------|--|---|---|--|--|
| Description | Tests if the given challenge data type is supported. | | | | |
| Parameters | osid.type.Type | peerChallengeType | a Type indicating a challenge data format | | |
| Return | boolean | | | | |
| Compliance | mandatory | | | | |
| | | | | | |
| Method | supportsCredentialExport | | | | |
| Description | Tests if Authentication objects can export serialzied credentials for transport. | | | | |
| Return | boolean | | | | |
| Compliance | mandatory | | | | |
| | | | | | |
| Method | getCredentialTypes | | | | |
| Description | Gets the supported credential types. | | | | |
| Return | osid.type.TypeList | a list containing the supported credential types | | | |
| Compliance | mandatory | | | | |
| | | | | | |
| Method | supportsCredentialType | | | | |
| Description | Tests if the given credential type is supported. | | | | |
| Parameters | osid.type.Type | credentialType | a Type indicating an credential type | | |
| Return | boolean | | | | |
| Compliance | mandatory | | | | |
| | | | | | |
| Method | getAgentRecordTypes | | | | |
| Description | Gets the supported Agent record types. | | | | |
| Return | osid.type.TypeList | a list containing the supported Agent record types | | | |
| Compliance | mandatory | | | | |
| | | | | | |
| Method | supportsAgentRecordType | | | | |
| Description | Tests if the given Agent record type is supported. | | | | |
| Parameters | osid.type.Type | agentRecordType | a Type indicating an Agent record type | | |
| Return | boolean | | | | |
| Compliance | mandatory | | | | |
| | | | | | |
| Method | getAgentSearchRecordTypes | | | | |
| Description | Gets the supported Agent search record types. | | | | |
| Return | osid.type.TypeList | a list containing the supported Agent search record types | | | |
| Compliance | mandatory | | | | |
| | | | | | |
| Method | supportsAgentSearchRecordType | | | | |
| Description | Tests if the given Agent search record type is supported. | | | | |
| Parameters | osid.type.Type | agentSearchRecordType | a Type indicating an Agent search record type | | |
| Return | boolean | | | | |
| Compliance | mandatory | | | | |
| | | | | | |
| Method | getKeyRecordTypes | | | | |
| Description | Gets the supported Key record types. | | | | |
| Return | osid.type.TypeList | a list containing the supported Key record types | | | |
| Compliance | mandatory | | | | |
| | | | | | |
| Method | supportsKeyRecordType | | | | |
| Description | Tests if the given Key record type is supported. | | | | |
| Parameters | osid.type.Type | keyRecordType | a Type indicating a Key type | | |
| Return | boolean | | | | |
| Compliance | mandatory | | | | |

| Method | getKeySearchRecordTypes | |
|--------------------|---|---|
| Description | Gets the supported key search record types. | |
| Return | osid.type.TypeList | a list containing the supported Key search record types |
| Compliance | mandatory | This method must be implemented. |
| Method | supportsKeySearchRecordType | |
| Description | Tests if the given key search record type is supported. | |
| Parameters | osid.type.Type | keySearchRecordType a Type indicating a Key search record type |
| Return | boolean | true if the given search record Type is supported, false otherwise |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AuthenticationManager | |
|-------------|---|---|
| Implements | osid.OsidManager osid.authentication.AuthenticationProfile | |
| Description | <p>The authentication manager provides access to authentication sessions and provides interoperability tests for various aspects of this service. The sessions included in this manager are:</p> <ul style="list-style-type: none"> • AuthenticationAcquisitionSession: a session to acquire credentials from a user and serialize them for transport to a remote peer for authentication • AuthenticationValidationSession: a session to receive and validate authentication credentials from a remote peer wishing to authenticate • AgentLookupSession: a session to look up Agents • AgentSearchSession: a session to search Agents • AgentAdminSession: a session to create, modify and delete Agents • AgentNotificationSession: a session to receive messages pertaining to Agent changes • KeyLookupSession: a session to access the keys of Agents • KeyAdminSession: a session to update the keys of Agents <p>The authentication manager provides a profile for determining authentication process compatibility with regard to requiring data from a challenge response mechanism to generate the credential with supportsChallenge(). The authentication profile also tests for supported Types. supportsChallengeType() and supportsAgentType(), supportsKeyType() are methods that can be used to determine if the desired Types are supported.</p> <p>Notifications for adds and changes to Agents is available via the getAgentNotificationSession() method.</p> | |
| Method | getAuthenticationAcquisitionSession | |
| Description | Gets an AuthenticationAcquisitionSession which is responsible for acquiring authentication credentials on behalf of a service client. | |
| Return | osid.authentication.AuthenticationAcquisitionSession | an acquisition session for this service |
| Errors | OPERATION_FAILED | unable to complete request |
| | UNIMPLEMENTED | supportsAuthenticationAcquisition() is false |
| Compliance | optional | This method must be implemented if supportsAcquisition() is true. |
| Method | getAuthenticationValidationSession | |
| Description | Gets the OsidSession associated with the AuthenticationValidation service. | |
| Return | osid.authentication.AuthenticationValidationSession | an AuthenticationValidationSession |
| Errors | OPERATION_FAILED | unable to complete request |
| | UNIMPLEMENTED | supportsAuthenticationValidation() is false |
| Compliance | optional | This method must be implemented if supportsValidation() is true. |
| Method | getAgentLookupSession | |
| Description | Gets the OsidSession associated with the agent lookup service. | |
| Return | osid.authentication.AgentLookupSession | an AgentLookupSession |
| Errors | OPERATION_FAILED | unable to complete request |
| | UNIMPLEMENTED | supportsAgentLookup() is false |
| Compliance | optional | This method must be implemented if supportsAgentLookup() is true. |

| Method | getAgentSearchSession | |
|-------------|--|--|
| Description | Gets the OsidSession associated with the agent search service. | |
| Return | <code>osid.authentication.AgentSearchSession</code> | an AgentSearchSession |
| Errors | <code>OPERATION_FAILED</code> | unable to complete request |
| | <code>UNIMPLEMENTED</code> | <code>supportsAgentSearch()</code> is false |
| Compliance | optional | This method must be implemented if <code>supportsAgentSearch()</code> is true. |
| Method | getAgentAdminSession | |
| Description | Gets the OsidSession associated with the agent administration service. | |
| Return | <code>osid.authentication.AgentAdminSession</code> | an AgentAdminSession |
| Errors | <code>OPERATION_FAILED</code> | unable to complete request |
| | <code>UNIMPLEMENTED</code> | <code>supportsAgentAdmin()</code> is false |
| Compliance | optional | This method must be implemented if <code>supportsAgentAdmin()</code> is true. |
| Method | getAgentNotificationSession | |
| Description | Gets the notification session for notifications pertaining to service changes. | |
| Parameters | <code>osid.authentication.AgentReceiver</code> receiver | the agent receiver interface |
| Return | <code>osid.authentication.AgentNotificationSession</code> | an AgentNotificationSession |
| Errors | <code>NULL_ARGUMENT</code> | receiver is null |
| | <code>OPERATION_FAILED</code> | unable to complete request |
| | <code>UNIMPLEMENTED</code> | <code>supportsAgentNotification()</code> is false |
| Compliance | optional | This method must be implemented if <code>supportsAgentNotification()</code> is true. |
| Method | getKeyLookupSession | |
| Description | Gets the OsidSession associated with the key lookup service. | |
| Return | <code>osid.authentication.KeyLookupSession</code> | the new KeyLookupSession |
| Errors | <code>OPERATION_FAILED</code> | unable to complete request |
| | <code>UNIMPLEMENTED</code> | <code>supportsKeyLookup()</code> is false |
| Compliance | optional | This method must be implemented if <code>supportsKeyLookup()</code> is true. |
| Method | getKeyAdminSession | |
| Description | Gets the OsidSession associated with the key administration service. | |
| Return | <code>osid.authentication.AgentAdminSession</code> | an KeyAdminSession |
| Errors | <code>OPERATION_FAILED</code> | unable to complete request |
| | <code>UNIMPLEMENTED</code> | <code>supportsKeyAdmin()</code> is false |
| Compliance | optional | This method must be implemented if <code>supportsKeyAdmin()</code> is true. |

| Interface | osid.authentication.AuthenticationProxyManager | |
|-------------|--|---|
| Implements | osid.OsidProxyManager osid.authentication.AuthenticationProfile | |
| | <p>The authentication proxy manager provides access to authentication sessions and provides interoperability tests for various aspects of this service. Methods in this manager support the passing of an Authentication object for the purposes of proxy authentication. The sessions included in this manager are:</p> <ul style="list-style-type: none"> • AuthenticationAcquisitionSession: session to acquire credentials from a user and serialize them for transport to a remote peer for authentication • AuthenticationValidationSession: session to receive and validate authentication credentials from a remote peer wishing to authenticate • AgentLookupSession: session to look up Agents • AgentSearchSession: session to search Agents • AgentAdminSession: session to create, modify and delete Agents • AgentNotificationSession: session to receive messages pertaining to Agent changes • KeyLookupSession: a session to lookup Agent keys • KeyAdminSession: a session to modify and delete Agent keys <p>The authentication manager provides a profile for determining authentication process compatibility with regard to requiring data from a challenge response mechanism to generate the credential with supportsChallenge().</p> <p>The authentication profile also tests for interoperability tests for supported Types. supportsAgentType(), supportsChallengeType() and supportsKeyType() are methods that can be used to determine if the desired Types are supported.</p> <p>Notifications for adds and changes to Agents is available via the getAgentNotificationSession() method.</p> | |
| Method | getAuthenticationAcquisitionSession | |
| Description | Gets the OsidSession associated with the AuthenticationAcquisitionSession using the supplied Authentication. | |
| Parameters | osid.authentication.Authentication authentication | proxy authentication |
| Return | osid.authentication.AuthenticationAcquisitionSession | an AuthenticationAcquisitionSession |
| Errors | NULL_ARGUMENT | authentication is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authentication is invalid |
| | UNIMPLEMENTED | supportsAuthenticationAcquisition() is false |
| | UNSUPPORTED | the authentication service is not supported |
| Compliance | optional | This method must be implemented if supportsAcquisition() is true. |

| Method | getAuthenticationValidationSession | | | | |
|-------------|--|---|------------------------------------|--|--|
| Description | Gets the OsidSession associated with the AuthenticationValidation service using the supplied Authentication. | | | | |
| Parameters | osid.authentication.Authentication | authentication | proxy authentication | | |
| Return | osid.authentication.AuthenticationValidationSession | | an AuthenticationValidationSession | | |
| Errors | NULL_ARGUMENT | authentication is null | | | |
| | OPERATION_FAILED | unable to complete request | | | |
| | PERMISSION_DENIED | authentication is invalid | | | |
| | UNIMPLEMENTED | supportsAuthenticationValidation() is false | | | |
| | UNSUPPORTED | the authentication service is not supported | | | |
| Compliance | optional | This method must be implemented if supportsValidation() is true. | | | |
| Method | getAgentLookupSession | | | | |
| Description | Gets the OsidSession associated with the agent lookup service. | | | | |
| Parameters | osid.authentication.Authentication | authentication | proxy authentication | | |
| Return | osid.authentication.AgentLookupSession | | an AgentLookupSession | | |
| Errors | NULL_ARGUMENT | authentication is null | | | |
| | OPERATION_FAILED | unable to complete request | | | |
| | PERMISSION_DENIED | authentication is invalid | | | |
| | UNIMPLEMENTED | supportsAgentLookup() is false | | | |
| | UNSUPPORTED | the authentication service is not supported | | | |
| Compliance | optional | This method must be implemented if supportsAgentLookup() is true. | | | |
| Method | getAgentSearchSession | | | | |
| Description | Gets the OsidSession associated with the agent search service. | | | | |
| Parameters | osid.authentication.Authentication | authentication | proxy authentication | | |
| Return | osid.authentication.AgentSearchSession | | an AgentSearchSession | | |
| Errors | NULL_ARGUMENT | authentication is null | | | |
| | OPERATION_FAILED | unable to complete request | | | |
| | PERMISSION_DENIED | authentication is invalid | | | |
| | UNIMPLEMENTED | supportsAgentSearch() is false | | | |
| | UNSUPPORTED | the authentication service is not supported | | | |
| Compliance | optional | This method must be implemented if supportsAgentSearch() is true. | | | |
| Method | getAgentAdminSession | | | | |
| Description | Gets the OsidSession associated with the agent administration service. | | | | |
| Parameters | osid.authentication.Authentication | authentication | proxy authentication | | |
| Return | osid.authentication.AgentAdminSession | | an AgentAdminSession | | |
| Errors | NULL_ARGUMENT | authentication is null | | | |
| | OPERATION_FAILED | unable to complete request | | | |
| | PERMISSION_DENIED | authentication is invalid | | | |
| | UNIMPLEMENTED | supportsAgentAdmin() is false | | | |
| | UNSUPPORTED | the authentication service is not supported | | | |
| Compliance | optional | This method must be implemented if supportsAgentAdmin() is true. | | | |

| Method | getAgentNotificationSession | | | | |
|-------------|---|---|----------------------|--|--|
| Description | Gets the messaging receiver session for notifications pertaining to agent changes. | | | | |
| Parameters | <code>osid.authentication.AgentReceiver</code> | receiver | the agent receiver | | |
| Return | <code>osid.authentication.AgentNotificationSession</code> | an AgentNotificationSession | | | |
| Errors | <code>NULL_ARGUMENT</code> <code>OPERATION_FAILED</code> <code>PERMISSION_DENIED</code> <code>UNIMPLEMENTED</code> <code>UNSUPPORTED</code> | authentication or receiver is null unable to complete request authentication is invalid <code>supportsAgentNotification()</code> is false the authentication service is not supported | | | |
| Compliance | <code>optional</code> | This method must be implemented if <code>supportsAgentNotification()</code> is true. | | | |
| Method | getKeyLookupSession | | | | |
| Description | Gets the OsidSession associated with the key lookup service. | | | | |
| Parameters | <code>osid.authentication.Authentication</code> | authentication | proxy authentication | | |
| Return | <code>osid.authentication.KeyLookupSession</code> | a KeyLookupSession | | | |
| Errors | <code>NULL_ARGUMENT</code> <code>OPERATION_FAILED</code> <code>PERMISSION_DENIED</code> <code>UNIMPLEMENTED</code> <code>UNSUPPORTED</code> | authentication is null unable to complete request authentication is invalid <code>supportsKeyLookup()</code> is false the authentication service is not supported | | | |
| Compliance | <code>optional</code> | This method must be implemented if <code>supportsKeyLookup()</code> is true. | | | |
| Method | getKeyAdminSession | | | | |
| Description | Gets the OsidSession associated with the key administration service. | | | | |
| Parameters | <code>osid.authentication.Authentication</code> | authentication | proxy authentication | | |
| Return | <code>osid.authentication.KeyAdminSession</code> | a KeyAdminSession | | | |
| Errors | <code>NULL_ARGUMENT</code> <code>OPERATION_FAILED</code> <code>PERMISSION_DENIED</code> <code>UNIMPLEMENTED</code> <code>UNSUPPORTED</code> | authentication is null unable to complete request authentication is invalid <code>supportsKeyAdmin()</code> is false the authentication service is not supported | | | |
| Compliance | <code>optional</code> | This method must be implemented if <code>supportsKeyAdmin()</code> is true. | | | |

| Interface | osid.authentication.AuthenticationAcquisitionSession | |
|--------------------|--|--|
| Implements | osid.OsidSession | |
| Description | This session acquires authentication credentials. The basic method, <code>getAuthentication()</code> , gets authentication credentials for use with authenticating to a remote peer. These credentials may be generated from direct user input or retrieved via a file, for example. | |
| Method | getAuthentication | |
| Description | Gets the authentication credential for the current user. The input data may represent the identity of the remote peer or data from a challenge-response transaction necessary for generating the credential. | |
| Return | osid.authentication.Authentication | the acquired authentication credential |
| Errors | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| Compliance | mandatory | This method is must be implemented. |

| Interface | osid.authentication.AuthenticationValidationSession | | |
|--------------------|--|----------------|--|
| Implements | osid.OsidSession | | |
| Description | <p>This session is the remote end of a transport link from the acquisition session and validates authentication credentials sent to it. The basic method, authenticate() accepts a credential, validates it and returns an Authentication object containing the identity of the authenticated user. The credential is indicated by a Type. AuthenticationManager.getCredentialTypes() lists all the credential types supported.</p> <p>This OSID does not define any root interface for credentials and challenge data. The object representing these are completely defined within their Type, providing flexibility in adapting to a variety of application environments</p> | | |
| Method | authenticate | | |
| Description | Validates and returns the authentication credential from the given data. | | |
| Parameters | object | credential | contains an authentication credential to be validated |
| | osid.type.Type | credentialType | specifies the credential interface |
| Return | osid.authentication.Authentication | | the acquired authentication credential |
| Errors | INVALID_ARGUMENT NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNSUPPORTED | | credential is not of credentialType credentialType or credential is null unable to complete request authorization failure credentialType not supported |
| Compliance | mandatory | | This method is must be implemented. |
| Method | getChallengeData | | |
| Description | Gets data that can be used for a challenge to the peer attempting authentication. | | |
| Parameters | osid.type.Type | challengeType | specifies the format of the data challenge |
| Return | object | | the acquired challenge data |
| Errors | NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNIMPLEMENTED UNSUPPORTED | | challengeType is null unable to complete request authorization failure challenge response not available challengeType not supported |
| Compliance | optional | | This method must be implemented if AuthenticationManager.supportsChallenge() is true. |

| Interface | osid.authentication.AgentLookupSession | |
|-------------|---|--|
| Implements | osid.OsidSession | |
| Description | <p>This session provides methods for retrieving Agent objects. The Agent represents the authenticated entity. Agents generally map to resources although this isn't always the case.</p> <p>This session defines two views which offer differing behaviors when retrieving multiple objects.</p> <ul style="list-style-type: none"> • comparative view: elements may be silently omitted or re-ordered • plenary view: provides a complete and ordered result set or is an error condition <p>Generally, the comparative view should be used for most applications as it permits operation even if there a particular element is inaccessible. For example, a hierarchy output can be plugged into a lookup method to retrieve all objects known to a hierarchy, but it may not be necessary to break execution if a node from the hierarchy no longer exists. However, some administrative applications may need to know whether it had retrieved an entire set of objects and may sacrifice some interoperability for the sake of precision.</p> <p>Agents may have an additional records indicated by their respective record types. The record may not be accessed through a cast of the Agent.</p> | |
| Method | canLookupAgents | |
| Description | <p>Tests if this user can perform Agent lookups. A return of true does not guarantee successful authorization. A return of false indicates that it is known all methods in this session will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer lookup operations.</p> | |
| Return | boolean | false if lookup methods are not authorized, true otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | useComparativeAgentView | |
| Description | <p>The returns from the lookup methods may omit or translate elements based on this session, such as authorization, and not result in an error. This view is used when greater interoperability is desired at the expense of precision.</p> | |
| Compliance | mandatory | This method is must be implemented. |
| Method | usePlenaryAgentView | |
| Description | <p>A complete view of the Agent returns is desired. Methods will return what is requested or result in an error. This view is used when greater precision is desired at the expense of interoperability.</p> | |
| Compliance | mandatory | This method is must be implemented. |
| Method | getAgent | |
| Description | <p>Gets the Agent specified by its Id. In plenary mode, the exact Id is found or a NOT_FOUND results. Otherwise, the returned Agent may have a different Id than requested, such as the case where a duplicate Id was assigned to an Agent and retained for compatibility.</p> | |
| Parameters | osid.id.Id agentId | the Id of the Agent to retrieve |
| Return | osid.authentication.Agent | the returned Agent |
| Errors | NOT_FOUND | no Agent found with the given Id |
| | NULL_ARGUMENT | agentId is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| Compliance | mandatory | This method must be implemented. |

| Method | getAgentsByIds | |
|-------------|--|---|
| Description | Gets an AgentList corresponding to the given IdList. In plenary mode, the returned list contains all of the agents specified in the Id list, in the order of the list, including duplicates, or an error results if an Id in the supplied list is not found or inaccessible. Otherwise, inaccessible Agents may be omitted from the list and may present the elements in any order including returning a unique set. | |
| Parameters | osid.id.IdList agentIdList | the list of Ids to retrieve |
| Return | osid.authentication.AgentList | the returned Agent list |
| Errors | NOT_FOUND NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED | an Id was not found agentIdList is null unable to complete request authorization failure |
| Compliance | mandatory | This method must be implemented. |
| Method | getAgentsByGenusType | |
| Description | Gets an AgentList corresponding to the given agent genus Type which does not include agents of genus types derived from the specified Type. In plenary mode, the returned list contains all known agents or an error results. Otherwise, the returned list may contain only those agents that are accessible through this session. In both cases, the order of the set is not specified. | |
| Parameters | osid.type.Type agentGenusType | an agent genus type |
| Return | osid.authentication.AgentList | the returned Agent list |
| Errors | NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED | agentGenusType is null unable to complete request authorization failure |
| Compliance | mandatory | This method must be implemented. |
| Method | getAgentsByParentGenusType | |
| Description | Gets an AgentList corresponding to the given agent genus Type and include any additional agents with genus types derived from the specified Type. In plenary mode, the returned list contains all known agents or an error results. Otherwise, the returned list may contain only those agents that are accessible through this session. In both cases, the order of the set is not specified. | |
| Parameters | osid.type.Type agentGenusType | an agent genus type |
| Return | osid.authentication.AgentList | the returned Agent list |
| Errors | NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED | agentGenusType is null unable to complete request authorization failure |
| Compliance | mandatory | This method must be implemented. |
| Method | getAgentsByRecordType | |
| Description | Gets an AgentList containing the given repository record Type. In plenary mode, the returned list contains all known agents or an error results. Otherwise, the returned list may contain only those agents that are accessible through this session. In both cases, the order of the set is not specified. | |
| Parameters | osid.type.Type agentRecordType | an agent record type |
| Return | osid.authentication.AgentList | the returned Agent list |
| Errors | NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED | agentRecordType is null unable to complete request authorization failure |
| Compliance | mandatory | This method must be implemented. |

| Method | getAgents | |
|--------------------|---|----------------------------------|
| Description | Gets all Agents. In plenary mode, the returned list contains all known agents or an error results. Otherwise, the returned list may contain only those agents that are accessible through this session. In both cases, the order of the set is not specified. | |
| Return | osid.authentication.AgentList | a list of Agents |
| Errors | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AgentSearchSession | |
|--------------------|--|--|
| Implements | osid.OsidSession | |
| Description | <p>This session provides methods for searching Agent objects. The search query is constructed using the AgentQuery interface. Multiple AgentQuery interfaces may be submitted into a search to perform a boolean OR. If more than one search element is specified within a single AgentQuery these elements form a boolean AND. The agent interface Type also specifies the interface for the agent query.</p> <p>getAgentsByQuery() is the basic search method and returns a list of Agents. A more advanced search may be performed with getAgentsBySearch(). It accepts an AgentSearch interface in addition to the query interface for the purpose of specifying additional options affecting the entire search, such as ordering. getAgentsBySearch() returns an AgentSearchResults interface that can be used to access the resulting AgentList or be used to perform a search within the result set through AgentSearch.</p> <p>Agents may have a record query interface indicated by their respective record types. The record query interface is accessed via the AgentQuery.</p> | |
| Method | canSearchAgents | |
| Description | Tests if this user can perform Agent searches. A return of true does not guarantee successful authorization. A return of false indicates that it is known all methods in this session will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer search operations to unauthorized users. | |
| Return | boolean | false if search methods are not authorized, true otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | getAgentQuery | |
| Description | Gets an agent query interface. | |
| Return | osid.authentication.AgentQuery | the agent query |
| Compliance | mandatory | This method must be implemented. |
| Method | getAgentsByQuery | |
| Description | Gets a list of Agents matching the given query interface. Each element in the array is OR'd. | |
| Parameters | osid.authentication.AgentQuery[] agentQueries | the search query array |
| Return | osid.authentication.AgentList | the returned AgentList |
| Errors | NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNSUPPORTED | |
| Compliance | mandatory | This method must be implemented. |
| Method | getAgentSearch | |
| Description | Gets an agent search interface. | |
| Return | osid.authentication.AgentSearch | the agent search interface |
| Compliance | mandatory | This method must be implemented. |
| Method | getAgentSearchOrder | |
| Description | Gets an agent search order interface. The AgentSearchOrder is supplied to an AgentSearch to specify the ordering of results. | |
| Return | osid.authentication.AgentSearchOrder | the agent search order interface |
| Compliance | mandatory | This method must be implemented. |

| Method | getAgentsBySearch | |
|--------------------|--|---|
| Description | Gets the search results matching the given search query using the given search. Each element in the query array is OR'd. | |
| Parameters | <code>osid.authentication.AgentQuery[]</code> | agentQueries |
| | <code>osid.authentication.AgentSearch</code> | agentSearch |
| Return | <code>osid.authentication.AgentSearchResults</code> | |
| Errors | <code>NULL_ARGUMENT</code> | agentQueries or agentSearch is null |
| | <code>OPERATION_FAILED</code> | unable to complete request |
| | <code>PERMISSION_DENIED</code> | authorization failure |
| | <code>UNSUPPORTED</code> | agentSearch or a query is not of this service |
| Compliance | mandatory | |
| | This method must be implemented. | |

| Interface | osid.authentication.AgentAdminSession | |
|--------------------|---|--|
| Implements | osid.OsidSession | |
| Description | This session provides methods to create, delete and modify Agent objects. The data for create and update is provided by the consumer via the form object. | |
| Method | canCreateAgents | |
| Description | Tests if this user can create Agents. A return of true does not guarantee successful authorization. A return of false indicates that it is known creating an Agent will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer create operations to an unauthorized user. | |
| Return | boolean | false if Agent creation is not authorized, true otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | canCreateAgentWithRecordTypes | |
| Description | Tests if this user can create a single Agent using the desired record interface types. While AuthenticationManager.getAgentRecordTypes() can be used to examine which record interfaces are supported, this method tests which record(s) are required for creating a specific Agent. Providing an empty array tests if an Agent can be created with no records. | |
| Parameters | osid.type.Type[] agentRecordTypes | array of types |
| Return | boolean | true if Agent creation using the specified record Types is supported, false otherwise |
| Errors | NULL_ARGUMENT | agentRecordTypes is null |
| Compliance | mandatory | This method must be implemented. |
| Method | getAgentFormForCreate | |
| Description | Gets the agent form for creating new agents. A new form should be requested for each create transaction. | |
| Return | osid.authentication.AgentForm | the agent form |
| Compliance | mandatory | This method must be implemented. |
| Method | createAgent | |
| Description | Creates a new Agent. | |
| Parameters | osid.authentication.AgentForm agentForm | the forms for this Agent |
| Return | osid.authentication.Agent | the new Agent |
| Errors | ALREADY_EXISTS INVALID_ARGUMENT NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNSUPPORTED | attempt at duplicating a property the underlying system is enforcing to be unique one or more of the form elements is invalid agentForm is null unable to complete request authorization failure agentForm is not of this service |
| Compliance | mandatory | This method must be implemented. |
| Method | canUpdateAgents | |
| Description | Tests if this user can update Agents. A return of true does not guarantee successful authorization. A return of false indicates that it is known updating an Agent will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer update operations to an unauthorized user. | |
| Return | boolean | false if agent modification is not authorized, true otherwise |
| Compliance | mandatory | This method must be implemented. |

| Method | canUpdateAgent | |
|-----------------------|---|---|
| Description | Tests if this user can update a specified agent. A return of true does not guarantee successful authorization. A return of false indicates that it is known updating the agent will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer an update operation to an unauthorized user for this agent. | |
| Parameters | osid.id.Id | agentId the Id of the Agent |
| Return | boolean | false if agent modification is not authorized, true otherwise |
| Errors | NULL_ARGUMENT | agentId is null |
| Compliance | mandatory | This method must be implemented. |
| Provider Notes | If the agentId is not found, then it is acceptable to return false to indicate the lack of an update available. | |
| Method | getAgentFormForUpdate | |
| Description | Gets the agent form for updating an existing agent. A new agent form should be requested for each update transaction. | |
| Parameters | osid.id.Id | agentId the Id of the Agent |
| Return | osid.authentication.AgentForm | the agent form |
| Errors | NOT_FOUND | agentId is not found |
| | NULL_ARGUMENT | agentId is null |
| | OPERATION_FAILED | unable to complete request |
| Compliance | mandatory | This method must be implemented. |
| Method | updateAgent | |
| Description | Updates an existing agent. | |
| Parameters | osid.id.Id | agentId the Id of the Agent |
| | osid.authentication.AgentForm | agentForm the form containing the elements to be updated |
| Errors | INVALID_ARGUMENT | the form contains an invalid value |
| | NOT_FOUND | agentId is not found |
| | NULL_ARGUMENT | agentId or agentForm is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| | UNSUPPORTED | agentForm is not supported |
| Compliance | mandatory | This method must be implemented. |
| Method | canDeleteAgents | |
| Description | Tests if this user can delete Agents. A return of true does not guarantee successful authorization. A return of false indicates that it is known deleting an Agent will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer delete operations to an unauthorized user. | |
| Return | boolean | false if Agent deletion is not authorized, true otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | canDeleteAgent | |
| Description | Tests if this user can delete a specified Agent. A return of true does not guarantee successful authorization. A return of false indicates that it is known deleting the Agent will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer an delete operation to an unauthorized user for this agent. | |
| Parameters | osid.id.Id | agentId the Id of the Agent |
| Return | boolean | false if Agent deletion is not authorized, true otherwise |
| Errors | NULL_ARGUMENT | agentId is null |
| Compliance | mandatory | This method must be implemented. |
| Provider Notes | If the agentId is not found, then it is acceptable to return false to indicate the lack of an delete available. | |

| Method | deleteAgent | | |
|--------------------|---|--------------------|---|
| Description | Deletes the Agent identified by the given Id. | | |
| Parameters | osid.id.Id | agentId | the Id of the Agent to delete |
| Errors | | NOT_FOUND | an Agent was not found identified by the given Id |
| NULL_ARGUMENT | | agentId | is null |
| OPERATION_FAILED | | | unable to complete request |
| PERMISSION_DENIED | | | authorization failure |
| Compliance | mandatory | | This method must be implemented. |
| Method | addIdToAgent | | |
| Description | Adds an Id to an Agent for the purpose of creating compatibility. The primary Id of the Agent is determined by the provider. The new Id performs as an alias to the primary Id. | | |
| Parameters | osid.id.Id | agentId | the Id of an Agent |
| | osid.id.Id | aliasId | the alias Id |
| Errors | | ALREADY_EXISTS | aliasId is already assigned |
| NOT_FOUND | | agentId | not found |
| NULL_ARGUMENT | | agentId or aliasId | is null |
| OPERATION_FAILED | | | unable to complete request |
| PERMISSION_DENIED | | | authorization failure |
| Compliance | mandatory | | This method must be implemented. |

| Interface | osid.authentication.AgentNotificationSession | |
|--------------------|---|--|
| Implements | osid.OsidSession | |
| Description | This session defines methods to receive asynchronous notifications on adds/changes to Agent objects. This session is intended for consumers needing to synchronize their state with this service without the use of polling. Notifications are cancelled when this session is closed. | |
| Method | canRegisterForAgentNotifications | |
| Description | Tests if this user can register for Agent notifications. A return of true does not guarantee successful authorization. A return of false indicates that it is known all methods in this session will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer notification operations. | |
| Return | boolean | false if notification methods are not authorized, true otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | registerForNewAgents | |
| Description | Register for notifications of new agents. AgentReceiver.newAgent() is invoked when a new Agent is created. | |
| Errors | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| Compliance | mandatory | This method must be implemented. |
| Method | registerForChangedAgents | |
| Description | Registers for notification of updated agents. AgentReceiver.changedAgent() is invoked when an agent is changed. | |
| Errors | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| Compliance | mandatory | This method must be implemented. |
| Method | registerForChangedAgent | |
| Description | Registers for notification of an updated agent. AgentReceiver.changedAgent() is invoked when the specified agent is changed. | |
| Parameters | osid.id.Id | agentId |
| | NOT_FOUND | the Id of the Agent to monitor |
| Errors | NULL_ARGUMENT | an Agent was not found identified by the given Id |
| | OPERATION_FAILED | agentId is null |
| | PERMISSION_DENIED | unable to complete request |
| Compliance | mandatory | authorization failure |
| Compliance | This method must be implemented. | |
| Method | registerForDeletedAgents | |
| Description | Registers for notification of deleted agents. AgentReceiver.deletedAgent() is invoked when an agent is deleted. | |
| Errors | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| Compliance | mandatory | This method must be implemented. |
| Method | registerForDeletedAgent | |
| Description | Registers for notification of a deleted agent. AgentReceiver.changedAgent() is invoked when the specified agent is changed. | |
| Parameters | osid.id.Id | agentId |
| | NOT_FOUND | the Id of the Agent to monitor |
| Errors | NULL_ARGUMENT | an Agent was not found identified by the given Id |
| | OPERATION_FAILED | agentId is null |
| | PERMISSION_DENIED | unable to complete request |
| Compliance | mandatory | authorization failure |
| Compliance | This method must be implemented. | |

| Interface | osid.authentication.KeyLookupSession | |
|-------------|---|--|
| Implements | osid.OsidSession | |
| Description | <p>This session provides methods for retrieving Key objects. The Key is associated with an Agent and identified by the Agent Id.</p> <p>This session defines two views which offer differing behaviors when retrieving multiple objects.</p> <ul style="list-style-type: none"> • comparative view: elements may be silently omitted or re-ordered • plenary view: provides a complete and ordered result set or is an error condition <p>Generally, the comparative view should be used for most applications as it permits operation even if there is data out of sync. For example, a hierarchy output can be plugged into a lookup method to retrieve all objects known to a hierarchy, but it may not be necessary to break execution if a node from the hierarchy no longer exists. However, some administrative applications may need to know whether it had retrieved an entire set of objects and may sacrifice some interoperability for the sake of precision. Keys may have an additional records indicated by their respective record types. The record may not be accessed through a cast of the Key.</p> | |
| Method | canLookupKeys | |
| Description | Tests if this user can lookup Keys. A return of true does not guarantee successful authorization. A return of false indicates that it is known getting a Key will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt to offer key management functions to unauthorized users. | |
| Return | boolean | false if key management is not authorized, true otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | useComparativeKeyView | |
| Description | The returns from the lookup methods may omit or translate elements based on this session, such as authorization, and not result in an error. This view is used when greater interoperability is desired at the expense of precision. | |
| Compliance | mandatory | This method is must be implemented. |
| Method | usePlenaryKeyView | |
| Description | A complete view of the Key returns is desired. Methods will return what is requested or result in an error. This view is used when greater precision is desired at the expense of interoperability. | |
| Compliance | mandatory | This method is must be implemented. |
| Method | hasKey | |
| Description | Tests if an agent has an associated key. | |
| Parameters | osid.id.Id | agentId |
| Return | boolean | true if the agent has a key, false otherwise |
| Errors | NOT_FOUND NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED | agentId is not found agentId is null unable to complete request authorization failure |
| Compliance | mandatory | This method is must be implemented. |

| Method | getKey | |
|-------------|--|---------------------------------------|
| Description | Gets the agent key. In plenary mode, the exact Id is found or a NOT_FOUND results. Otherwise, the returned Agent via the Key may have a different Id than requested, such as the case where a duplicate Id was assigned to an Agent and retained for compatibility. | |
| Parameters | osid.id.Id agentId | the Id of the Agent |
| Return | osid.authentication.Key | the key of the agent |
| Errors | NOT_FOUND | agentId is not found or no key exists |
| | NULL_ARGUMENT | agentId is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| Compliance | mandatory | This method must be implemented. |
| Method | getKeysByIds | |
| Description | Gets a KeyList corresponding to the given agent IdList. In plenary mode, the returned list contains all of the keys for agents specified in the Id list, in the order of the list, including duplicates, or an error results if an Id in the supplied list is not found or inaccessible. Otherwise, inaccessible Keys may be omitted from the list and may present the elements in any order including returning a unique set. | |
| Parameters | osid.id.IdList agentIdList | the list of Ids to retrieve |
| Return | osid.authentication.KeyList | the returned Key list |
| Errors | NOT_FOUND | an Id was not found |
| | NULL_ARGUMENT | agentIdList is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| Compliance | mandatory | This method must be implemented. |
| Method | getKeysByRecordType | |
| Description | Gets a list of keys corresponding to the given key record Type. The set of keys implementing the given record type is returned. In plenary mode, the returned list contains all known keys or an error results. Otherwise, the returned list may contain only those keys that are accessible through this session. In both cases, the order of the set is not specified. | |
| Parameters | osid.type.Type keyRecordType | a key record type |
| Return | osid.authentication.KeyList | the returned Key list |
| Errors | NULL_ARGUMENT | keyRecordType is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| | PERMISSION_DENIED | authorization failure |
| Compliance | mandatory | This method must be implemented. |
| Method | getKeys | |
| Description | Gets all Keys. In plenary mode, the returned list contains all known keys or an error results. Otherwise, the returned list may contain only those keys that are accessible through this session. In both cases, the order of the set is not specified. | |
| Return | osid.authentication.KeyList | a list of Keys |
| Errors | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeySearchSession | |
|--------------------|--|--|
| Implements | osid.OsidSession | |
| Description | <p>This session provides methods for searching Key objects. The search query is constructed using the KeyQuery interface. Multiple KeyQuery interfaces may be submitted into a search to perform a boolean OR. If more than one search element is specified within a single KeyQuery these elements form a boolean AND. The key Type also specifies the interface for the key query.</p> <p>getKeysByQuery() is the basic search method and returns a list of Keys. A more advanced search may be performed with getKeysBySearch(). It accepts an KeySearch interface in addition to the query interface for the purpose of specifying additional options affecting the entire search, such as ordering. getKeysBySearch() returns a KeySearchResult interface that can be used to access the resulting KeyList or be used to perform a search within the result set through KeySearch.</p> <p>Keys may have a record query interface indicated by their respective record types. The record query interface is accessed via the KeyQuery.</p> | |
| Method | canSearchKeys | |
| Description | Tests if this user can perform Key searches. A return of true does not guarantee successful authorization. A return of false indicates that it is known all methods in this session will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer search operations to unauthorized users. | |
| Return | boolean | false if search methods are not authorized, true otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | getKeyQuery | |
| Description | Gets a key query interface. | |
| Return | osid.authentication.KeyQuery | the key query |
| Compliance | mandatory | This method must be implemented. |
| Method | getKeysByQuery | |
| Description | Gets a list of Keys matching the given search interface. Each element in the array is OR'd. | |
| Parameters | osid.authentication.KeyQuery[] keyQueries | the search query array |
| Return | osid.authentication.KeyList | the returned KeyList |
| Errors | NULL_ARGUMENT | keyQueries is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| | UNSUPPORTED | keyQueries is not of this service |
| Compliance | mandatory | This method must be implemented. |
| Method | getKeySearch | |
| Description | Gets a key query interface. The returned query only makes available the core KeyQuery interface and does not support additional interface types. getKeyQueryForInterfaceType() should be used if additional interface types are required. | |
| Return | osid.authentication.KeySearch | the key search interface |
| Compliance | mandatory | This method must be implemented. |
| Method | getKeySearchOrder | |
| Description | Gets a key search order interface. The KeySearchOrder is supplied to a KeySearch to specify the ordering of results. | |
| Return | osid.authentication.KeySearchOrder | the key search order interface |
| Compliance | mandatory | This method must be implemented. |

| Method | getKeysBySearch | |
|--------------------|---|--|
| Description | Gets a list of Keys matching the given search interface. Each element in the array is OR'd. | |
| Parameters | osid.authentication.KeyQuery[] | keyQueries |
| | osid.authentication.KeySearch | keySearch |
| Return | osid.authentication.KeySearchResults | |
| Errors | NULL_ARGUMENT | keyQueries or keySearch is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| | UNSUPPORTED | keySearch or a keyQuery is not of this service |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeyAdminSession | |
|--------------------|--|---|
| Implements | osid.OsidSession | |
| Description | This session provides methods to creating, updating and deleting Key objects. Keys are managed separately from the Agent. Each Agent maps to a zero or one Key and every Key maps to one Agent. Keys are identified from their counterpart Agent Id. | |
| Method | canCreateKeys | |
| Description | Tests if this user can create Keys. A return of true does not guarantee successful authorization. A return of false indicates that it is known creating a Key will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer create operations to an unauthorized user. | |
| Return | boolean | false if Key creation is not authorized, true otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | canCreateKeyWithRecordTypes | |
| Description | Tests if this user can create a single Key using the desired record interface types. While AuthenticationManager.getKeyRecordTypes() can be used to examine which record interfaces are supported, this method tests which record(s) are required for creating a specific Key. Providing an empty array tests if a Key can be created with no records. | |
| Parameters | osid.type.Type[] keyRecordTypes | array of types |
| Return | boolean | true if Key creation using the specified record Types is supported, false otherwise |
| Errors | NULL_ARGUMENT | keyRecordTypes is null |
| Compliance | mandatory | This method must be implemented. |
| Method | getKeyFormForCreate | |
| Description | Gets the key form for creating and updating new keys. A new form should be requested for each create transaction. | |
| Parameters | osid.id.Id agentId | the Id of the Agent |
| Return | osid.authentication.KeyForm | the key form |
| Errors | NOT_FOUND NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED | agentId is not found agentId is null unable to complete request authorization failure |
| Compliance | mandatory | This method must be implemented. |
| Method | createKey | |
| Description | Creates a new Key. | |
| Parameters | osid.id.Id agentId osid.authentication.KeyForm keyForm | the Id of the Agent the form for this Key |
| Return | osid.authentication.Key | the new Key |
| Errors | ALREADY_EXISTS INVALID_ARGUMENT NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNSUPPORTED | agent already has a key one or more of the form elements is invalid agentId or keyForm is null unable to complete request authorization failure a keyForm is not of this service |
| Compliance | mandatory | This method must be implemented. |

| Method | canUpdateKeys | |
|-----------------------|---|---|
| Description | Tests if this user can update Keys. A return of true does not guarantee successful authorization. A return of false indicates that it is known updating a Key will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer update operations to an unauthorized user. | |
| Return | boolean | false if key modification is not authorized, true otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | canUpdateKey | |
| Description | Tests if this user can update a specified key. A return of true does not guarantee successful authorization. A return of false indicates that it is known updating the key will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer an update operation to an unauthorized user for this agent. | |
| Parameters | osid.id.Id agentId | the Id of the Agent |
| Return | boolean | false if agent modification is not authorized, true otherwise |
| Errors | NULL_ARGUMENT | agentId is null |
| Compliance | mandatory | This method must be implemented. |
| Provider Notes | If the agentId is not found, then it is acceptable to return false to indicate the lack of an update available. | |
| Method | getKeyFormForUpdate | |
| Description | Gets the key form for updating an existing key. | |
| Parameters | osid.id.Id agentId | the Id of the Agent |
| Return | osid.authentication.KeyForm | the key form |
| Errors | NOT_FOUND NULL_ARGUMENT OPERATION_FAILED | agentId is not found agentId is null unable to complete request |
| Compliance | mandatory | This method must be implemented. |
| Method | updateKey | |
| Description | Updates a key for an agent. | |
| Parameters | osid.id.Id agentId osid.authentication.KeyForm keyForm | the Id of the Agent the form containing the elements to be updated |
| Errors | INVALID_ARGUMENT NOT_FOUND NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNSUPPORTED | the form contains an invalid value agentId is not found agentId or keyForm is null unable to complete request authorization failure keyForm is not supported |
| Compliance | mandatory | This method must be implemented. |
| Method | canDeleteKeys | |
| Description | Tests if this user can delete Keys. A return of true does not guarantee successful authorization. A return of false indicates that it is known deleting a Key will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer delete operations to an unauthorized user. | |
| Return | boolean | false if Key deletion is not authorized, true otherwise |
| Compliance | mandatory | This method must be implemented. |

| Method | canDeleteKey | |
|----------------|---|---|
| Description | Tests if this user can delete a specified Key. A return of true does not guarantee successful authorization. A return of false indicates that it is known deleting the Key will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer an delete operation to an unauthorized user for this agent. | |
| Parameters | osid.id.Id agentId | the Id of the Agent |
| Return | boolean | false if Key deletion is not authorized, true otherwise |
| Errors | NULL_ARGUMENT | agentId is null |
| Compliance | mandatory | This method must be implemented. |
| Provider Notes | If the agentId is not found, then it is acceptable to return false to indicate the lack of an delete available. | |
| Method | deleteKey | |
| Description | Deletes the Key associated with the given agent Id. | |
| Parameters | osid.id.Id agentId | the Id of the Agent whose key to delete |
| Errors | NOT_FOUND | an Agent was not found identified by the given Id |
| | NULL_ARGUMENT | agentId is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.Authentication | |
|----------------|--|--|
| Implements | | |
| Description | <p>Authentication represents an authentication credential which contains set of bytes and a format Type. Once an Authentication object is created from the AuthenticationValidationSession, the credential data can be extracted and sent to the remote peer for validation. The remote peer gets another Authentication object as a result of validating the serialized credential data.</p> <p>An Authentication may or may not be valid. isValid() should be checked before acting upon the Agent identity to which the credential is mapped.</p> | |
| Method | getAgentId | |
| Description | Gets the Id of the Agent identified in this authentication credential. | |
| Return | osid.id.Id | the Agent Id |
| Compliance | mandatory | This method must be implemented. |
| Provider Notes | The Agent should be determined at the time this credential is created. | |
| Method | getAgent | |
| Description | Gets the Agent identified in this authentication credential. | |
| Return | osid.authentication.Agent | the Agent |
| Errors | OPERATION_FAILED | unable to complete request |
| Compliance | mandatory | This method must be implemented. |
| Method | isValid | |
| Description | Tests whether or not the credential represented by this Authentication is currently valid. A credential may be invalid because it has been destroyed, expired, or is somehow no longer able to be used. | |
| Return | boolean | true if this authentication credential is valid, false otherwise |
| Compliance | mandatory | This method must be implemented. |
| Provider Notes | Any problem in determining the validity of this credential should result in false. | |
| Method | getExpiration | |
| Description | Gets the expiration date associated with this authentication credential. Consumers should check for the existence of a an expiration mechanism via hasExpiration(). | |
| Return | timestamp | the expiration date of this authentication credential |
| Errors | OPERATION_FAILED | unable to complete request |
| Compliance | optional | This method must be implemented if AuthenticationManager.supportsExpiration() is true. |
| Method | getCredential | |
| Description | Gets the credential represented by the given Type for transport to a remote service. | |
| Parameters | osid.type.Type | credentialType the credential format Type |
| Return | object | the credential |
| Errors | NULL_ARGUMENT | credentialType is null |
| | OPERATION_FAILED | unable to complete request |
| | UNIMPLEMENTED | this provider does not support exporting credentials |
| | UNSUPPORTED | the given credentialType is not supported |
| Compliance | optional | This method must be implemented if AuthenticationManager.supportsCredentialExport() is true. |
| Provider Notes | A provider may support multiple credential formats for a variety of applications. | |

| Interface | osid.authentication.Agent | |
|--------------------|--|--|
| Implements | osid.OsidObject | |
| Description | An Agent represents an authenticatable identity. Like all OSID objects, an Agent is identified by its Id and any persisted references should use the Id. | |
| | getAgentRecord | |
| Description | Gets the record corresponding to the given Agent record Type. This method must be used to retrieve an object implementing the requested record interface along with all of its ancestor interfaces. The subjectRecordType may be the Type returned in getRecordTypes() or any of its parents in a Type hierarchy where hasRecordType(agentRecordType) is true. | |
| Parameters | osid.type.Type agentRecordType | the type of the record to retrieve |
| Return | osid.authentication.AgentRecord | the agent record |
| Errors | NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNSUPPORTED | agentRecordType is null unable to complete request authorization failure occurred hasRecordType(agentRecordType) is false |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AgentRecord | |
|--------------------|--|----------------------------------|
| Implements | osid.OsidRecord | |
| Description | A record for an Agent. The methods specified by the record type are available through the underlying object. | |
| getAgent | | |
| Description | Gets the Agent from which this record originated. | |
| Return | osid.authentication.Agent | the agent |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AgentQuery | |
|--------------------|---|---|
| Implements | osid.OsidQuery | |
| Description | <p>This is the query interface for searching agents. Each method match request produces an AND term while multiple invocations of a method produces a nested OR, except for accessing the AgentQuery record.</p> <p>The following example returns agents whose display name begins with "Tom" and whose "login name" is "tom" or "tjcoppel" in an interface specified by companyAgentType.</p> <pre>AgentQuery queries[1]; queries[0] = session.getAgentQueryRecord(companyAgentType); queries[0].matchDisplayName("Tom*", wildcardStringMatchType, false); companyAgentQuery = queries[0].getAgentQueryInterface(); companyAgentQuery.matchLoginName("tom"); companyAgentQuery = queries[0].getAgentQueryInterface(); companyAgentQuery.matchLoginName("tjcoppel"); AgentList agentList = session.getAgentsByQuery(queries);</pre> | |
| Method | supportsKeyQuery | |
| Description | Tests if an KeyQuery is available. | |
| Return | boolean | true if a key query interface is available, false otherwise |
| Compliance | mandatory | This method must be implemented. |
| Method | getKeyQuery | |
| Description | Includes an agent query for making relations with Keys. Multiple retrievals return separate query terms nested inside this query term, each which are treated as a boolean OR. For example, AgentQuery1 OR (AgentQuery2.description AND (AgentQuery2.KeyQuery1.name OR AgentQuery2.KeyQuery2.name)) | |
| Parameters | boolean | match |
| Return | osid.authentication.KeyQuery | |
| Errors | UNIMPLEMENTED | |
| Compliance | mandatory | This method must be implemented. |
| Method | getAgentQueryRecord | |
| Description | Gets the record query interface corresponding to the given Agent record Type. Multiple record retrievals of the same type may return the same underlying object and do not result in adding terms to the query. Multiple record retrievals of different types add AND terms to the other elements set in this query. | |
| Parameters | osid.type.Type | agentRecordType |
| Return | osid.authentication.AgentQueryRecord | |
| Errors | NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNSUPPORTED | |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AgentQueryRecord | |
|--------------------|--|----------------------------------|
| Implements | osid.OsidRecord | |
| Description | A record for a AgentQuery. The methods specified by the record type are available through the underlying object. | |
| | getAgentQuery | |
| Description | Gets the AgentQuery from which this record originated. | |
| Return | osid.authentication.AgentQuery | the agent query |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AgentForm | | | | | | | | | |
|-----------------------------------|---|--|-------------------------------|-------------------------|----------------------------------|----------------------------|-----------------------------------|--------------------------------|-----------------------------|---|
| Implements | osid.OsidForm | | | | | | | | | |
| Description | This is the form for creating and updating Agents. Like all OsidForm objects, various data elements may be set here for use in the create and update methods in the AgentAdminSession. For each data element that may be set, metadata may be examined to provide display hints or data constraints. | | | | | | | | | |
| Method | getAgentFormRecord | | | | | | | | | |
| Description | Gets the AgentFormRecord interface corresponding to the given agent record interface Type. | | | | | | | | | |
| Parameters | osid.type.Type | agentRecordType the agent record type | | | | | | | | |
| Return | osid.authentication.AgentFormRecord | the record | | | | | | | | |
| Errors | <table> <tr> <td>NULL_ARGUMENT</td> <td>agentRecordType is null</td> </tr> <tr> <td>OPERATION_FAILED</td> <td>unable to complete request</td> </tr> <tr> <td>PERMISSION_DENIED</td> <td>authorization failure occurred</td> </tr> <tr> <td>UNSUPPORTED</td> <td>hasRecordType(agentRecordType) is false</td> </tr> </table> | | NULL_ARGUMENT | agentRecordType is null | OPERATION_FAILED | unable to complete request | PERMISSION_DENIED | authorization failure occurred | UNSUPPORTED | hasRecordType(agentRecordType) is false |
| NULL_ARGUMENT | agentRecordType is null | | | | | | | | | |
| OPERATION_FAILED | unable to complete request | | | | | | | | | |
| PERMISSION_DENIED | authorization failure occurred | | | | | | | | | |
| UNSUPPORTED | hasRecordType(agentRecordType) is false | | | | | | | | | |
| Compliance | mandatory | This method must be implemented. | | | | | | | | |

| Interface | osid.authentication.AgentFormRecord | |
|---------------------|---|----------------------------------|
| Implements | osid.OsidRecord | |
| Description | A record for a AgentForm. The methods specified by the record type are available through the underlying object. | |
| getAgentForm | | |
| Description | Gets the AgentForm from which this record originated. | |
| Return | osid.authentication.AgentForm | the agent form |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AgentSearchOrder | |
|--------------------|---|---|
| Implements | osid.OsidSearchOrder | |
| Description | An interface for specifying the ordering of search results. | |
| Method | getAgentSearchOrderRecord | |
| Description | Gets the agent search order record corresponding to the given agent record Type. Multiple retrievals return the same underlying object. | |
| Parameters | osid.type.Type agentRecordType | an agent record type |
| Return | osid.authentication.AgentSearchOrderRecord | the agent search order record interface |
| Errors | NULL_ARGUMENT | agentRecordType is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure occurred |
| | UNSUPPORTED | hasRecordType(agentRecordType) is false |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AgentSearchOrderRecord | |
|----------------------------|--|----------------------------------|
| Implements | osid.OsidRecord | |
| Description | A record for a AgentSearchOrder. The methods specified by the record type are available through the underlying object. | |
| getAgentSearchOrder | | |
| Description | Gets the AgentSearchOrder from which this record originated. | |
| Return | osid.authentication.AgentSearchOrder | the agent search order |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AgentSearch | |
|-------------|---|---|
| Implements | osid.OsidSearch | |
| Description | <p>AgentSearch defines the interface for specifying agent search options. This example gets a limited set of squid-like agents.</p> <pre>AgentSearch as = session.getAgentSearch(); as.limitResultSet(25, 50); AgentQuery queries[1]; queries[0] = session.getAgentQuery(); String kword = "squid"; queries[0].matchKeywords(kword, true); AgentSearchResults results = session.getAgentsBySearch(queries, as); AgentList list = results.getAgents();</pre> | |
| Method | searchWithinAgentResults | |
| Description | Execute this search using a previous search result. | |
| Parameters | osid.authentication.AgentSearchResults | results |
| Parameters | results from a query | |
| Errors | INVALID_ARGUMENT | results is not valid |
| | NULL_ARGUMENT | results is null |
| Compliance | mandatory | This method must be implemented. |
| Method | searchAmongAgents | |
| Description | Execute this search among the given list of agents. | |
| Parameters | osid.id.IdList | agentIds |
| Parameters | list of agents | |
| Errors | NULL_ARGUMENT | agentIds is null |
| Compliance | mandatory | This method must be implemented. |
| Method | orderAgentResults | |
| Description | Specify an ordering to the search results. | |
| Parameters | osid.authentication.AgentSearchOrder | agentSearchOrder |
| Parameters | agent search order | |
| Errors | NULL_ARGUMENT | agentSearchOrder is null |
| | UNSUPPORTED | agentSearchOrder is not of this service |
| Compliance | mandatory | This method must be implemented. |
| Method | getAgentSearchRecord | |
| Description | Gets the record corresponding to the given agent search record Type. This method must be used to retrieve an object implementing the requested record interface along with all of its ancestor interfaces. | |
| Parameters | osid.type.Type | agentSearchRecordType |
| Parameters | an agent search record type | |
| Return | osid.authentication.AgentSearchRecord | the agent search interface |
| Errors | NULL_ARGUMENT | agentSearchRecordType is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure occurred |
| | UNSUPPORTED | hasSearchRecordType(agentSearchRecordType) is false |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AgentSearchRecord | |
|-----------------------|---|----------------------------------|
| Implements | osid.OsidSearchRecord | |
| Description | A record for a AgentSearch. The methods specified by the record type are available through the underlying object. | |
| getAgentSearch | | |
| Description | Gets the AgentSearch from which this record originated. | |
| Return | osid.authentication.AgentSearch | the agent search |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AgentSearchResults | |
|--------------------|---|---|
| Implements | osid.OsidSearchResults | |
| Description | <p>This interface provides a means to capture results of a search and is used as a vehicle to perform a search within a search.</p> <pre>AgentSearch as = session.getAgentSearch(); as.limitResultSet(25, 50); AgentQuery queries[1]; queries[0] = session.getAgentQuery(); String kwords[1]; kwords[0] = "squid"; queries[0].matchKeywords(kwords); AgentSearchResults results = session.getAgentsBySearch(queries, as); AgentList list = results.getAgents();</pre> | |
| Method | getAgents | |
| Description | Gets the agent list resulting from the search. | |
| Return | osid.authentication.AgentList | the agent list |
| Compliance | mandatory | This method must be implemented. |
| Method | getAgentSearchResultsRecord | |
| Description | Gets the record corresponding to the given agent search record Type. This method must be used to retrieve an object implementing the requested record interface along with all of its ancestor interfaces. | |
| Parameters | osid.type.Type agentSearchRecordType | an agent search record type |
| Return | osid.authentication.AgentSearchResultsRecord | the agent search interface |
| Errors | NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNSUPPORTED | agentSearchRecordType is null |
| | | unable to complete request |
| | | authorization failure occurred |
| | | hasSearchRecordType(agentSearchRecordType) is false |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AgentSearchResultsRecord | |
|------------------------------|--|----------------------------------|
| Implements | osid.OsidSearchRecord | |
| Description | A record for a AgentSearchResults. The methods specified by the record type are available through the underlying object. | |
| getAgentSearchResults | | |
| Description | Gets the AgentSearchResults from which this record originated. | |
| Return | osid.authentication.AgentSearchResults | the agent search results |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.AgentReceiver | | |
|--------------------|---|---------|----------------------------------|
| Implements | osid.OsidReceiver | | |
| Description | The agent receiver is the consumer supplied interface for receiving notifications pertaining to new, updated or deleted Agent objects. A change to a key is a change to an Agent. | | |
| Method | newAgent | | |
| Description | The callback for notifications of new agents. | | |
| Parameters | osid.id.Id | agentId | the Id of the new Agent |
| Compliance | mandatory | | This method must be implemented. |
| Method | changedAgent | | |
| Description | The callback for notification of updated agents. | | |
| Parameters | osid.id.Id | agentId | the Id of the updated Agent |
| Compliance | mandatory | | This method must be implemented. |
| Method | deletedAgent | | |
| Description | the callback for notification of deleted agents. | | |
| Parameters | osid.id.Id | agentId | the Id of the deleted Agent |
| Compliance | mandatory | | This method must be implemented. |

| Interface | osid.authentication.AgentList | |
|-------------|--|---|
| Implements | osid.OsidList | |
| Description | <p>Like all OsidLists, AgentList provides a means for accessing Agent elements sequentially either one at a time or many at a time. Examples:</p> <pre>while (al.hasNext()) { Agent agent = al.getNextAgent(); }</pre> <p>or</p> <pre>while (al.hasNext()) { Agent[] agent = al.getNextAgents(al.available()); }</pre> | |
| | getNextAgent | |
| Description | Gets the next Agent in this list. | |
| Return | osid.authentication.Agent | the next Agent in this list. The hasNext() method should be used to test that a next Agent is available before calling this method. |
| Errors | ILLEGAL_STATE OPERATION_FAILED | no more elements available in this list unable to complete request |
| Compliance | mandatory | This method must be implemented. |
| | getNextAgents | |
| Description | Gets the next set of Agent elements in this list which must be less than or equal to the number returned from available(). | |
| Parameters | cardinal | n |
| Return | osid.authentication.Agent[] | an array of Agent elements. The length of the array is less than or equal to the number specified. |
| Errors | ILLEGAL_STATE OPERATION_FAILED | no more elements available in this list unable to complete request |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.Key | |
|--------------------|--|---|
| Implements | | |
| Description | <p>The key represents cryptographic data managed by the authentication service. An Agent maps to a Key and there is only one Key per Agent.</p> <p>getKeyInterface() should be used to retrieve the interface corresponding to this Type. The existence of the interface must not be assumed until requested at which point it is safe to cast into the interface indicated by the type.</p> | |
| Method | getAgent | |
| Description | Gets the Agent corresponding to this key. | |
| Return | osid.authentication.Agent | the agent |
| Compliance | mandatory | This method must be implemented. |
| Method | getRecordTypeType | |
| Description | Gets the type of this key. The Type explicitly indicates the specification of the extension interface and implicitly may define an object family. | |
| Return | osid.type.Type | the type |
| Compliance | mandatory | This method must be implemented. |
| Method | hasRecordType | |
| Description | Tests if this key supports the given record Type. The given type may be supported by the key type through inheritance. | |
| Parameters | osid.type.Type | keyRecordType |
| Return | boolean | true if this key supports the given record Type, false otherwise |
| Errors | NULL_ARGUMENT | keyRecordType is null |
| Compliance | mandatory | This method must be implemented. |
| | getKeyReceord | |
| Description | Gets the record corresponding to the given Key record Type. This method must be used to retrieve an object implementing the requested record interface along with all of its ancestor interfaces. The keyRecordType may be the Type returned in getRecordTypes() or any of its parents in a Type hierarchy where hasRecordType(keyRecordType) is true. | |
| Parameters | osid.type.Type | keyRecordType |
| Return | osid.authentication.KeyRecord | the key record |
| Errors | NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNSUPPORTED | repositoryRecordType is null unable to complete request authorization failure occurred hasRecordType(keyRecordType) is false |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeyRecord | |
|--------------------|---|----------------------------------|
| Implements | osid.OsidRecord | |
| Description | A record for a Key. The methods specified by the record type are available through the underlying object. | |
| getKey | | |
| Description | Gets the Key from which this record originated. | |
| Return | osid.authentication.Key | the key |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeyQuery | | | | |
|--------------------|--|---|---|--|--|
| Implements | | | | | |
| Description | This is the query interface for searching keys. Each method match request is produces an AND term while multiple invocations of a method produces a nested OR, except for accessing the KeyQuery record. | | | | |
| Method | getStringMatchTypes | | | | |
| Description | Gets the supported string match types. | | | | |
| Return | osid.type.TypeList | a list containing the supported string match types | | | |
| Compliance | mandatory | This method must be implemented. | | | |
| Method | supportsStringMatchType | | | | |
| Description | Tests if the given string matching type is supported. | | | | |
| Parameters | osid.type.Type | searchType | a Type indicating a string match type | | |
| Return | boolean | true if the given Type is supported, false otherwise | | | |
| Compliance | mandatory | This method must be implemented. | | | |
| Method | matchRecordType | | | | |
| Description | Sets a Type for querying keys of a given record type. | | | | |
| Parameters | osid.type.Type | recodrType | a key record type | | |
| | boolean | match | true if the record type query is a positive match, false for negative match | | |
| Errors | NULL_ARGUMENT | | recordType is null | | |
| Compliance | mandatory | This method must be implemented. | | | |
| Method | supportsAgentQuery | | | | |
| Description | Tests if an AgentQuery is available. | | | | |
| Return | boolean | true if an agent query interface is available, false otherwise | | | |
| Compliance | mandatory | This method must be implemented. | | | |
| Method | getAgentQuery | | | | |
| Description | Includes an agent query for making relations with Agents. Multiple rerievals return separate query terms nested inside this query term, each which are treated as a boolean OR. | | | | |
| Parameters | boolean | match | true if the key query is a positive match, false for negative match | | |
| Return | osid.authentication.AgentQuery | | the query extension | | |
| Errors | UNIMPLEMENTED | | supportsAgentQuery() is false | | |
| Compliance | mandatory | This method must be implemented. | | | |
| Method | hasRecordType | | | | |
| Description | Tests if this query supports the given record Type. The given record type may be supported by the object through interface/type inheritance. This method should be checked before retrieving the record interface. | | | | |
| Parameters | osid.type.Type | keyRecordType | a type | | |
| Return | boolean | true if a record query of the given record Type is available, false otherwise | | | |
| Errors | NULL_ARGUMENT | | keyRecordType is null | | |
| Compliance | mandatory | This method must be implemented. | | | |

| Method | getKeyQueryRecord | |
|--------------------|--|--|
| Description | Gets the record query interface corresponding to the given Key record Type. Multiple record retrievals of the same type may return the same underlying object and do not result in adding terms to the query. Multiple record retrievals of different types add AND terms to the other elements set in this query. | |
| Parameters | osid.type.Type | keyRecordType a key record type |
| Return | osid.authentication.KeyQueryRecord | the key query record |
| Errors | NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNSUPPORTED | keyRecordType is null unable to complete request authorization failure occurred hasRecordType(keyRecordType) is false |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeyQueryRecord | |
|--------------------|--|----------------------------------|
| Implements | osid.OsidRecord | |
| Description | A record for a KeyQuery. The methods specified by the record type are available through the underlying object. | |
| getKeyQuery | | |
| Description | Gets the KeyQuery from which this record originated. | |
| Return | osid.authentication.KeyQuery | the key query |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeyForm | | | | |
|--------------------|---|--|-------------------|--|--|
| Implements | | | | | |
| Description | This is the form for creating and updating Keys. Various data elements may be set here for use in the create and update methods in the KeyAdminSession. For each data element that may be set, metadata may be examined to provide display hints or data constraints. | | | | |
| Method | getCommentMetadata | | | | |
| Description | Gets the metadata for the comment corresponding to this form submission. The comment is used for describing the nature of the change to the corresponding object for the purposes of logging and auditing. | | | | |
| Return | osid.Metadata | metadata for the comment | | | |
| Compliance | mandatory | This method must be implemented. | | | |
| Method | setComment | | | | |
| Description | Sets a comment. | | | | |
| Parameters | string | comment | the new comment | | |
| Errors | INVALID_ARGUMENT NO_ACCESS NULL_ARGUMENT | comment is invalid comment cannot be modified comment is null | | | |
| Compliance | mandatory | This method must be implemented. | | | |
| Method | isValid | | | | |
| Description | Tests if this form is in a valid state for submission. A form is valid if all required data has been supplied compliant with any constraints. | | | | |
| Return | boolean | false if there is a known error in this form, true otherwise | | | |
| Errors | OPERATION_FAILED | attempt to perform validation failed | | | |
| Compliance | mandatory | This method must be implemented. | | | |
| Method | hasRecordType | | | | |
| Description | Tests if this form supports the given record Type. The given record type may be supported by the object through interface/type inheritance. This method should be checked before retrieving the record interface. | | | | |
| Parameters | osid.type.Type | keyRecordType | a record type | | |
| Return | boolean | true if a record form of the given record Type is available, false otherwise | | | |
| Errors | NULL_ARGUMENT | keyRecordType is null | | | |
| Compliance | mandatory | This method must be implemented. | | | |
| Method | getKeyFormRecord | | | | |
| Description | Gets the KeyFormRecord interface corresponding to the given key record interface Type. | | | | |
| Parameters | osid.type.Type | keyRecordType | a key record type | | |
| Return | osid.authentication.KeyFormRecord | the key form record | | | |
| Errors | NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNSUPPORTED | keyRecordType is null unable to complete request authorization failure occurred hasRecordType(keyRecordType) is false | | | |
| Compliance | mandatory | This method must be implemented. | | | |

| Interface | osid.authentication.KeyFormRecord | |
|--------------------|---|----------------------------------|
| Implements | osid.OsidRecord | |
| Description | A record for a KeyForm. The methods specified by the record type are available through the underlying object. | |
| getKeyForm | | |
| Description | Gets the KeyForm from which this record originated. | |
| Return | osid.authentication.KeyForm | the key form |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeySearchOrder | |
|--------------------|---|---------------------------------------|
| Implements | osid.OsidSearchOrder | |
| Description | An interface for specifying the ordering of search results. | |
| Method | getKeySearchOrderRecord | |
| Description | Gets the key search order record corresponding to the given key record Type. Multiple retrievals return the same underlying object. | |
| Parameters | osid.type.Type | keyRecordType a key record type |
| Return | osid.authentication.KeySearchOrderRecord | the key search order record interface |
| Errors | NULL_ARGUMENT | keyRecordType is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure occurred |
| | UNSUPPORTED | hasRecordType(keyRecordType) is false |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeySearchOrderRecord | |
|--------------------------|--|----------------------------------|
| Implements | osid.OsidRecord | |
| Description | A record for a KeySearchOrder. The methods specified by the record type are available through the underlying object. | |
| getKeySearchOrder | | |
| Description | Gets the KeySearchOrder from which this record originated. | |
| Return | osid.authentication.KeySearchOrder | the key search order |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeySearch | |
|--------------------|--|--|
| Implements | osid.OsidSearch | |
| Description | KeySearch defines options for governing key searches. | |
| Method | searchWithinKeyResults | |
| Description | Execute this search using a previous search result. | |
| Parameters | osid.authentication.KeySearchResults | results results from a query |
| Errors | NULL_ARGUMENT UNSUPPORTED | results is null results is not of this service |
| Compliance | mandatory | This method must be implemented. |
| Method | searchAmongKeys | |
| Description | Execute this search among the given list of keys. | |
| Parameters | osid.id.IdList | agentIds list of agents |
| Errors | NULL_ARGUMENT UNSUPPORTED | agentIds is null agentIds is not of this service |
| Compliance | mandatory | This method must be implemented. |
| Method | orderKeyResults | |
| Description | Specify an ordering to the search results. | |
| Parameters | osid.authentication.KeySearchOrder | keySearchOrder key search order |
| Errors | NULL_ARGUMENT UNSUPPORTED | keySearchOrder is null keySearchOrder is not of this service |
| Compliance | mandatory | This method must be implemented. |
| Method | getKeySearchRecord | |
| Description | Gets the record corresponding to the given key search record Type. This method must be used to retrieve an object implementing the requested record interface along with all of its ancestor interfaces. | |
| Parameters | osid.type.Type | keySearchRecordType a key search record type |
| Return | osid.authentication.KeySearchRecord | the key search interface |
| Errors | NULL_ARGUMENT OPERATION_FAILED PERMISSION_DENIED UNSUPPORTED | keySearchRecordType is null unable to complete request authorization failure occurred hasSearchRecordType(keySearchRecordType) is false |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeySearchRecord | |
|---------------------|---|----------------------------------|
| Implements | osid.OsidSearchRecord | |
| Description | A record for a KeySearch. The methods specified by the record type are available through the underlying object. | |
| getKeySearch | | |
| Description | Gets the KeySearch from which this record originated. | |
| Return | osid.authentication.KeySearch | the key search |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeySearchResults | |
|-------------|---|---|
| Implements | osid.OsidSearchResults | |
| Description | <p>This interface provides a means to capture results of a search and is used as a vehicle to perform a search within a previous result set. This example fetches all keys and orders them by the agent display name.</p> <pre>KeySearch ks = session.getKeySearch(); AgentSearch as = ks.getAgentSearch(); as.orderByDisplayName();</pre> <p>KeyQuery keyQueries[1]; keyQueries[0] = session.getKeyQuery(); AgentQuery agQueries[1]; agQueries[1].matchDisplayName("", true); KeySearchResults results = session.getKeysBySearch(keyQueries, ks);</p> <pre>KeyList kl = results.getKeys();</pre> | |
| Method | getKeys | |
| Description | Gets the key list resulting from the search. | |
| Return | osid.authentication.KeyList | the key list |
| Compliance | mandatory | This method must be implemented. |
| Method | getKeySearchResultsRecord | |
| Description | Gets the record corresponding to the given key search record Type. This method must be used to retrieve an object implementing the requested record interface along with all of its ancestor interfaces. | |
| Parameters | osid.type.Type keySearchRecordType | a key search record type |
| Return | osid.authentication.KeySearchResultsRecord | the key search interface |
| Errors | NULL_ARGUMENT | keySearchRecordType is null |
| | OPERATION_FAILED | unable to complete request |
| | PERMISSION_DENIED | authorization failure occurred |
| | UNSUPPORTED | hasSearchRecordType(keySearchRecordType) is false |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeySearchResultsRecord | |
|----------------------------|--|----------------------------------|
| Implements | osid.OsidSearchRecord | |
| Description | A record for a KeySearchResults. The methods specified by the record type are available through the underlying object. | |
| getKeySearchResults | | |
| Description | Gets the KeySearchResults from which this record originated. | |
| Return | osid.authentication.KeySearchResults | the key search results |
| Compliance | mandatory | This method must be implemented. |

| Interface | osid.authentication.KeyList | |
|-------------|---|---|
| Implements | osid.OsidList | |
| Description | <p>Like all OsidLists, KeyList provides a means for accessing Key elements sequentially either one at a time or many at a time. Examples:</p> <pre>while (kl.hasNext()) { Key key = kl.getNextKey(); }</pre> <p>or</p> <pre>while (kl.hasNext()) { Key[] keys = kl.getNextKeys(kl.available()); }</pre> | |
| | getNextKey | |
| Description | Gets the next Key in this list. | |
| Return | osid.authentication.Key | the next Key in this list. The hasNext() method should be used to test that a next Key is available before calling this method. |
| Errors | ILLEGAL_STATE OPERATION_FAILED | no more elements available in this list unable to complete request |
| Compliance | mandatory | This method must be implemented. |
| | getNextKeys | |
| Description | Gets the next set of Key elements in this list which must be less than or equal to the number returned from available(). | |
| Parameters | cardinal | n |
| | the number of Key elements requested which should be less than or equal to available() | |
| Return | osid.authentication.Key[] | an array of Key elements. The length of the array is less than or equal to the number specified. |
| Errors | ILLEGAL_STATE OPERATION_FAILED | no more elements available in this list unable to complete request |
| Compliance | mandatory | This method must be implemented. |