| | [**Overview**](http://docs.google.com/overview-summary.html) | **Package** | Class | [**Use**](http://docs.google.com/package-use.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV PACKAGE**](http://docs.google.com/javax/xml/xpath/package-summary.html)   [**NEXT PACKAGE**](http://docs.google.com/org/omg/CORBA/package-summary.html) | [**FRAMES**](http://docs.google.com/index.html?org/ietf/jgss/package-summary.html)    [**NO FRAMES**](http://docs.google.com/package-summary.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

## Package org.ietf.jgss

This package presents a framework that allows application developers to make use of security services like authentication, data integrity and data confidentiality from a variety of underlying security mechanisms like Kerberos, using a unified API.

**See:**

[**Description**](#3znysh7)

| **Interface Summary** | |
| --- | --- |
| [**GSSContext**](http://docs.google.com/org/ietf/jgss/GSSContext.html) | This interface encapsulates the GSS-API security context and provides the security services that are available over the context. |
| [**GSSCredential**](http://docs.google.com/org/ietf/jgss/GSSCredential.html) | This interface encapsulates the GSS-API credentials for an entity. |
| [**GSSName**](http://docs.google.com/org/ietf/jgss/GSSName.html) | This interface encapsulates a single GSS-API principal entity. |

| **Class Summary** | |
| --- | --- |
| [**ChannelBinding**](http://docs.google.com/org/ietf/jgss/ChannelBinding.html) | This class encapsulates the concept of caller-provided channel binding information. |
| [**GSSManager**](http://docs.google.com/org/ietf/jgss/GSSManager.html) | This class serves as a factory for other important GSS-API classes and also provides information about the mechanisms that are supported. |
| [**MessageProp**](http://docs.google.com/org/ietf/jgss/MessageProp.html) | This is a utility class used within the per-message GSSContext methods to convey per-message properties. |
| [**Oid**](http://docs.google.com/org/ietf/jgss/Oid.html) | This class represents Universal Object Identifiers (Oids) and their associated operations. |

| **Exception Summary** | |
| --- | --- |
| [**GSSException**](http://docs.google.com/org/ietf/jgss/GSSException.html) | This exception is thrown whenever a GSS-API error occurs, including any mechanism specific error. |

## Package org.ietf.jgss Description

This package presents a framework that allows application developers to make use of security services like authentication, data integrity and data confidentiality from a variety of underlying security mechanisms like Kerberos, using a unified API. The security mechanisms that an application can chose to use are identified with unique object identifiers. One example of such a mechanism is the Kerberos v5 GSS-API mechanism (object identifier 1.2.840.113554.1.2.2). This mechanism is available through the default instance of the GSSManager class.

The GSS-API is defined in a language independent way in [RFC 2743](http://www.ietf.org/rfc/rfc2743.txt). The Java language bindings are defined in [RFC 2853](http://www.ietf.org/rfc/rfc2853.txt)

An application starts out by instantiating a GSSManager which then serves as a factory for a security context. An application can use specific principal names and credentials that are also created using the GSSManager; or it can instantiate a context with system defaults. It then goes through a context establishment loop. Once a context is established with the peer, authentication is complete. Data protection such as integrity and confidentiality can then be obtained from this context.

The GSS-API does not perform any communication with the peer. It merely produces tokens that the application must somehow transport to the other end.

### Credential Acquisition

The GSS-API itself does not dictate how an underlying mechanism obtains the credentials that are needed for authentication. It is assumed that prior to calling the GSS-API, these credentials are obtained and stored in a location that the mechanism provider is aware of. However, the default model in the Java platform will be that mechanism providers must obtain credentials only from the private or public credential sets associated with the [Subject](http://docs.google.com/javax/security/auth/Subject.html) in the current access control context. The Kerberos v5 mechanism will search for the required INITIATE and ACCEPT credentials ([KerberosTicket](http://docs.google.com/javax/security/auth/kerberos/KerberosTicket.html) and [KerberosKey](http://docs.google.com/javax/security/auth/kerberos/KerberosKey.html)) in the private credential set where as some other mechanism might look in the public set or in both. If the desired credential is not present in the appropriate sets of the current Subject, the GSS-API call must fail.

This model has the advantage that credential management is simple and predictable from the applications point of view. An application, given the right permissions, can purge the credentials in the Subject or renew them using standard Java API's. If it purged the credentials, it would be sure that the JGSS mechanism would fail, or if it renewed a time based credential it would be sure that a JGSS mechanism would succeed.

This model does require that a [JAAS login](http://docs.google.com/javax/security/auth/login/package-summary.html) be performed in order to authenticate and populate a Subject that the JGSS mechnanism can later utilize. However, applications have the ability to relax this restiction by means of a system property: javax.security.auth.useSubjectCredsOnly. By default this system property will be assumed to be true (even when it is unset) indicating that providers must only use the credentials that are present in the current Subject. However, if this property is explicitly set to false by the application, then it indicates that the provider is free to use any credentials cache of its choice. Such a credential cache might be a disk cache, an in-memory cache, or even just the current Subject itself.

## Related Documentation

For an online tutorial on using Java GSS-API, please see [Introduction to JAAS and Java GSS-API](http://docs.google.com/technotes/guides/security/jgss/tutorials/index.html).

**Since:** 1.4

| | [**Overview**](http://docs.google.com/overview-summary.html) | **Package** | Class | [**Use**](http://docs.google.com/package-use.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV PACKAGE**](http://docs.google.com/javax/xml/xpath/package-summary.html)   [**NEXT PACKAGE**](http://docs.google.com/org/omg/CORBA/package-summary.html) | [**FRAMES**](http://docs.google.com/index.html?org/ietf/jgss/package-summary.html)    [**NO FRAMES**](http://docs.google.com/package-summary.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).