MIT Kerberos

Example GSS-API Android NDK App

November 19th 2012, version 1.0

Introduction

This is a sample Android NDK application which provides a GUI wrapper around the MIT Kerberos kinit, klist, kvno, and kdestroy client applications. It also provides a sample client which uses the Java GSS-API interface. The GSS-API interface is a Java interface for the existing native MIT GSS-API library.

This package includes cross-compiled versions of the MIT Kerberos libraries as well as the CyaSSL Embedded SSL Library. It should be helpful to Android developers who are interested in using the Kerberos libraries or the GSS-API interface in their own Android NDK Applications.

By default, this package uses pre-built static Kerberos and CyaSSL libraries which are needed in order to be linked to the KerberosApp application's native library (libkerberosapp.so).

For detailed information on the Java GSS-API interface, please see the GSSAPI_README file included in this project's root directory or see the kerberos-java-gssapi project on GitHub, here:

https://github.com/cconlon/kerberos-java-gssapi

Table of Contents

1. Requirements

Before building or installing this package, you must have the Android SDK and NDK installed and set up on your system. It is also helpful to have the Android Emulator setup and configured with an Android platform greater than or equal to version 2.3.3 (Gingerbread). For details on downloading and setting these up, please see the following links:

Android SDK: http://developer.android.com/sdk/index.html
Android NDK: http://developer.android.com/sdk/ndk/index.html
Android Emulator: https://developer.android.com/guide/developing/tools/emulator.html

SWIG will also need to be installed in order to build the underlying GSS-API wrapper. To download and install SWIG, please visit see the project homepage at http://www.swig.org. This project has been developed using SWIG version 1.3.40 running on Linux.

2. Building

To build and install this package, including Java GSS-API bindings, run the following commands.

```
android update project -p . -s
swig -java -package edu.mit.jgss.swig -outdir ./src/edu/mit/jgss/swig
     -o ./jni/gsswrapper_wrap.c ./jni/gsswrapper.i
ndk-build
ant debug
```

If you want to rebuild the pre-built Kerberos libraries, please use the android-config.sh shell script. This will allow the MIT Kerberos libraries to be cross-compiled for the Android platform. More detailed instructions can be found in the script comments.

3. Installing

To install this package in a running Android emulator, run:

```
ant <debug> install
```

Where <debug> is either "debug" or "release", depending on what build configuration used with ndk-build.

Before running the KerberosApp application, the user needs to install both a Keytab file and a Kerberos configuration file. Reference the MIT Kerberos documentation for guidelines for creating these files. Once created, they can be installed using the adb push command, using:

```
adb push <local-file-path> <destination-file-path>
```

For example, to load a krb5.conf and krb5.keytab file from the current directory to an Android emulator under the /data/local/kerberos directory, run:

```
adb push krb5.conf /data/local/kerberos/
adb push krb5.keytab /data/local/kerberos/
```

If the application is set to use a client keytab instead of a password, the keytab file needs to contain an entry for the client principal (whose TGT will be obtained by using the "kinit" button in the sample Application).

NOTE: hosts file

If you need to edit the hosts file on the android emulator to accommodate for KDC locations, use the following steps:

```
emulator -avd <name> -partition-size 128
adb remount
adb pull /system/etc/hosts ./
<< edit hosts file on local machine >>
adb push ./hosts /system/etc
```

4. Usage

This NDK application's GUI is split into three tabs:

1. Client Info
2. Server Info
3. Client App

These tabs should be addressed in the order they are listed above. A short summary of each is below.

1. Client Info

This tab displays the wrappers around native kinit, klist, kvno, and kdestroy application code. It provides the functionality to get a ticket for a given client principal using either a keytab or password for principal authentication. The default configuration file and credential cache locations are listed on this screen as well.

2. Server Info

   This tab allows the user to enter information about the server which the client application will attempt to make a GSS-API connection with in Tab 3. Server principal name, IP address, and port number should be given in this tab.

3. Client App

   This tab allows the user to start the client GSS-API application. The client application will attempt to connect to the GSS-API server given in Tab 2, using the client principal info gathered in Tab 1. This client application was designed to connect to the example server from the kerberos-java-gssapi package. The client app will do the following:

   a. Establish a GSS-API context with the example server
   b. Sign, encrypt, and send a message to the server
   c. Verify the signature block returned by the server

5. Default Storage Locations

The following locations are the default paths used for the Kerberos sample application.

Kerberos config file: /data/local/kerberos/krb5.conf
Credentials cache: /data/local/kerberos/ccache/krb5cc_<uid>
Keytab: /data/local/kerberos/krb5.keytab

The credentials cache location may be changed in KerberosAppActivity.java. The Kerberos config file and keytab file locations may be changed by editing the default locations in ./include/osconf.h when cross compiling the MIT kerberos libraries.

6. Library Versions

At the time of writing, the CyaSSL and Kerberos libraries used in this package were:

CyaSSL 2.0.0rc3
http://www.yassl.com

Kerberos (cconlon krb5 fork) GitHub master
Repository: http://github.com/cconlon/krb5

Homepage: http://web.mit.edu/kerberos/

7. Licenses

CyaSSL Embedded SSL Library

MIT Kerberos Libraries and Application Code:

8. Support

If you have any questions or comments, please contact support@yassl.com or the MIT Kerberos community.

9. References

MIT Kerberos: http://web.mit.edu/kerberos/
yaSSL: http://www.yassl.com/

Kerberos Java GSS-API Wrapper: https://github.com/cconlon/kerberos-java-gssapi
Example GSS-API Android NDK App: https://github.com/cconlon/kerberos-android-ndk

RFC 5653: http://tools.ietf.org/html/rfc5653