Configure an IPsec environment using the Kerberos authentication mechanism

Effectively managing Kerberos authentication between a set of IBM AIX systems using the IPsec protocol

Srikanth R. Joshi
Sandeep Umesh

July 28, 2017

Kerberos is an authentication mechanism used to authenticate a set of users by having different security realms and by exchanging ticket over a secured or non-secured environment. Internet Protocol Security (IPsec) is a security protocol used to establish a secured channel over interacting systems. The IPsec security protocol allows the use of Kerberos for user authentication and to have a secured communication channel established between two client systems. This article explains how we can use IPsec as a security protocol to communicate between two client systems and how we can configure IPsec with Kerberos authentication between these systems.

Kerberos introduction

Kerberos is one of the user authentication protocols which works on a ticketing mechanism with symmetric encryption. In IBM® AIX®, the Kerberos protocol is provided through Network Authentication Service. So, in this article we will be using Kerberos and network authentication service interchangeably.

From a high level, Kerberos authentication works on the client/server model with a client contacting the server to obtain tickets. Here, the client can either be a user or a service (for example: Telnet, FTP, SSH, IPsec and so on). The following two types of tickets get exchanged between the server and the client during the process of Kerberos authentication:

- Ticket granting ticket (TGT)
- Service ticket

In Kerberos, the user or the service to be authenticated is known as the principal.

The server, more commonly known as the key distribution center (KDC), contains the private keys of all principals within its network. This server contains two parts: Authentication server and ticket-
granting server. You can note that when the Network Authentication Service server is configured, a daemon (krb5kdc) starts running. This daemon performs the following tasks of KDC:

1. Client sends out a request to the KDC.
2. The KDC’s authentication server checks the request and sends the TGT to the client. This TGT is encrypted using the client's password. So, only the client can decrypt this and this acts as a proof for the client's identity.
3. If the client needs to use any service, the client sends out another request to KDC. This request consists of the TGT and the service it requires to use.
4. This request is now handled by KDC’s ticket-granting server. It does the validation of the request and issues a service ticket to the client system. This acts as a proof for the client's identity at the service side.
5. The client forwards the service ticket to the service it needs to use.
6. The service contacts KDC’s ticket-granting server for validation of the ticket and finally it provides access to the user to make use of the service.

**Figure 1. Kerberos configuration**
IPsec introduction

IPsec is a protocol that provides security at the IP layer in the network stack. This ensures that all the packets transmitted between two systems are encrypted.

The authentication between the two communicating systems can be achieved through pre-shared keys, public key infrastructure, or Kerberos authentication mechanism. Figure 2 represents the basic IPsec configuration.

Figure 2. IPsec basic configuration

This article describes how Kerberos authentication can be used along with the IPsec protocol to ensure that all packets transmitted between two systems are encrypted.

Prerequisites

Before we perform IPsec with Kerberos authentication, ensure that the following prerequisites are met, which would otherwise cause a failure during setup:

1. Make sure that the host name on all the AIX systems are in the fully qualified domain name (FQDN) form.
2. By default, some commands of Java™ are set. So, make sure that the `PATH` environment variable is changed to consider the Kerberos commands path at the `/usr/krb5/sbin` and `/usr/krb5/bin/`.
3. Kerberos tickets contain information about when the tickets are generated and are valid for a few minutes of interval. So, make sure that the timings between the interacting systems are in sync with a maximum difference of 5 minutes between the systems.

Scope of this article

As per this article, we will have three systems configured:

- One system will be configured as the Kerberos server and this provides authentication service and ticket-granting service
- An IPsec channel will be established between the other two systems
  - One system acts as the IPsec initiator.
  - Another system acts as the IPsec responder.

For Kerberos authentication, it is required that the IPsec initiator system obtains an initial TGT and makes use of the IPsec service provided on the IPsec responder system. Hence both these systems also act as the Kerberos client systems.
After configuring these systems, we can ping from the IPsec initiator system to the IPsec responder system and verify that the IPsec tunnel is active. We can also confirm whether the Kerberos tickets are obtained properly on the Kerberos client systems.

**Figure 3. IPsec with Kerberos configuration**

As shown in Figure 3, we will configure `britmanual06` as the Kerberos server system, `britmanual07` as the IPsec initiator system, and `britmanual08` as IPsec responder system. IPsec tunnel will be established between `britmanual07` and `britmanual08` systems.

**Configuration steps**

The configuration steps required to perform IPsec with kerberos authentication can be classified into the following two stages:

1. Configure one system as Kerberos server and other two systems as Kerberos clients.
2. Configure one Kerberos client system as the IPsec initiator and the other client system as IPsec responder.

**1. Configure Kerberos server and Kerberos client systems**

You need to perform the following steps to configure Kerberos server and client systems:

1. Ensure that the Kerberos-related file set is installed on all three (server and client) systems.
   (As stated before, one system will be used as Kerberos server and the Installation of the Kerberos file sets displays the following output:
Configure an IPsec environment using the Kerberos authentication mechanism

2. Configure the Kerberos server on **britmanual06**.
You can configure the AIX system as a Kerberos server using the `config.krb5` command or the `mkkrb5srv` command (present in `/usr/sbin/`).

Syntax of this command is as follows:
```
mkkrb5srv -r <Realm name> -d <Domain name> -a <principal name of the Kerberos server admin>
```

You will be prompted to set the database master password and the administrator password. In this example, we have provided `secret` as the password. The following example explains how to configure the Kerberos server.

```bash
[root@britmanual06 /] # mkkrb5srv -r NFSREALM -d isst.aus.stglabs.ibm.com -a admin/admin
```

<table>
<thead>
<tr>
<th>Fileset</th>
<th>Level</th>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path: /usr/lib/objrepos</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>krb5.server.rte</td>
<td>1.6.0.3</td>
<td>COMMITTED</td>
<td>Network Authentication Service Server</td>
</tr>
</tbody>
</table>

```
Path: /etc/objrepos
krb5.server.rte          | 1.6.0.3 | COMMITTED | Network Authentication Service Server           |
```

Initialization configuration...
Creating /etc/krb5/krb5_cfg_type...
Creating /etc/krb5/krb5.conf...
Creating /var/krb5/krb5kdc/kdc.conf...
Creating database files...
Initializing database '/var/krb5/krb5kdc/principal' for realm 'NFSREALM'
master key name 'K/M@NFSREALM'
You are prompted for the database Master Password.
It is important that you DO NOT FORGET this password.
Enter database Master Password:
Re-enter database Master Password to verify:
WARNING: no policy specified for admin/admin@NFSREALM; defaulting to no policy. Note that policy may be overridden by ACL restrictions.
Enter password for principal "admin/admin@NFSREALM":
Re-enter password for principal "admin/admin@NFSREALM":
Creating keytable...
Creating /var/krb5/krb5kdc/kadm5.acl...
Starting krb5kdc...
krb5kdc was started successfully.
Starting kadmind...
kadmind was started successfully.
The command completed successfully.
Restarting kadmind and krb5kdc
```
[root@britmanual06 /] #
```

3. Configure the Kerberos clients on `britmanual07` and `britmanual08`.

You can configure the AIX system as a Kerberos client system using the `config.krb5` command or the `mkkrb5clnt` command (present in `/usr/sbin/`).

Syntax of this command is as follows:
```
mkkrb5clnt -r <Realm name> -d <Domain name> -s <fully qualified hostname of admin server> -c <fully qualified hostname of KDC> -A -i <configure integrated Kerberos authentication> -K -T
```

Here, `Realm` is used as a cluster within a complete administrative domain.
The arguments have the following meaning:
Configure an IPsec environment using the Kerberos authentication mechanism

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-A</td>
<td>Specifies the root to be added as a Kerberos administrative user</td>
</tr>
<tr>
<td>-i</td>
<td>Configures integrated Kerberos authentication</td>
</tr>
<tr>
<td>-K</td>
<td>Specifies Kerberos to be used as a default authentication scheme</td>
</tr>
<tr>
<td>-T</td>
<td>Acquires the server admin TGT based admin ticket</td>
</tr>
</tbody>
</table>

Here we will be prompted to provide the administrator password that was set during the configuration of the `krb` server. If successful, we will be prompted to provide a password for the principal entry that will be created for the Kerberos client host system. As a default value, we have provided `secret` as the password.

Refer to the following examples to configure the Kerberos clients (`britmanual07` and `britmanual08`).

```bash
Initializing configuration...
Creating /etc/krb5/krb5_cfg_type...
Creating /etc/krb5/krb5.conf...
The command completed successfully.
Password for admin/admin@NFSREALM:
Configuring fully integrated login
Authenticating as principal admin/admin with existing credentials.
WARNING: no policy specified for host/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM; defaulting to no policy. Note that policy may be overridden by ACL restrictions.
Principal "host/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM" created.
Administration credentials NOT DESTROYED.
Making root a Kerberos administrator
Authenticating as principal admin/admin with existing credentials.
WARNING: no policy specified for root/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM; defaulting to no policy. Note that policy may be overridden by ACL restrictions.
Enter password for principal "root/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM":
Re-enter password for principal "root/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM":
Administration credentials NOT DESTROYED.
Configuring Kerberos as the default authentication scheme
Cleaning administrator credentials and exiting.
[root@britmanual07 ~] #

[root@britmanual08 ~] # mkkrb5clnt -r NFSREALM -d isst.aus.stglabs.ibm.com -s britmanual06.isst.aus.stglabs.ibm.com -c britmanual06.isst.aus.stglabs.ibm.com -A -i files -K -T
Initializing configuration...
Creating /etc/krb5/krb5_cfg_type...
Creating /etc/krb5/krb5.conf...
The command completed successfully.
Password for admin/admin@NFSREALM:
Configuring fully integrated login
Authenticating as principal admin/admin with existing credentials.
WARNING: no policy specified for host/britmanual08.isst.aus.stglabs.ibm.com@NFSREALM; defaulting to no policy. Note that policy may be overridden by ACL restrictions.
Principal "host/britmanual08.isst.aus.stglabs.ibm.com@NFSREALM" created.
Administration credentials NOT DESTROYED.
Making root a Kerberos administrator
Authenticating as principal admin/admin with existing credentials.
WARNING: no policy specified for root/britmanual08.isst.aus.stglabs.ibm.com@NFSREALM; defaulting to no policy. Note that policy may be overridden by ACL restrictions.
Enter password for principal "root/britmanual08.isst.aus.stglabs.ibm.com@NFSREALM":
Re-enter password for principal "root/britmanual08.isst.aus.stglabs.ibm.com@NFSREALM":

Administration credentials NOT DESTROYED.
Configuring Kerberos as the default authentication scheme
Cleaning administrator credentials and exiting.
[root@britmanual08] /
#

Make sure that the principal name entry corresponding to the client will be created in the KDC, which is host/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM and host/britmanual08.isst.aus.stglabs.ibm.com@NFSREALM. We can verify this on the Kerberos server system using the kadmin.local command

4. Prepare the keytab file on the Kerberos client systems.
   The keytab file must contain the principal name entry of both the client systems:
   - Entry of the IPsec initiator system is required to obtain an initial TGT.
   - Entry of the IPsec responder system is required during the setup of the IPsec channel.

Instead of creating separate keytab files for both the Kerberos client systems, we can create only one keytab file and place it at /etc/krb5/ folder of both the systems. We can create the keytab file using the kadmin command.

Refer to the following example on preparing the keytab file for both Kerberos client systems.
root@britmanual08] /
# kadmin -p admin/admin
Authenticating as principal admin/admin with password.
Password for admin/admin@NFSREALM:
kadmin: ktadd host/britmanual07.isst.aus.stglabs.ibm.com
Entry for principal host/britmanual07.isst.aus.stglabs.ibm.com with kvno 4,
enryption type Triple DES cbc mode with HMAC/shal added to keytab WRFILE:/etc/krb5/krb5.keytab.
Entry for principal host/britmanual07.isst.aus.stglabs.ibm.com with kvno 4,
enryption type ArcFour with HMAC/md5 added to keytab WRFILE:/etc/krb5/krb5.keytab.
Entry for principal host/britmanual07.isst.aus.stglabs.ibm.com with kvno 4,
enryption type AES-256 CTS mode with 96-bit SHA-1 HMAC added to keytab WRFILE:/etc/krb5/krb5.keytab.
Entry for principal host/britmanual07.isst.aus.stglabs.ibm.com with kvno 4,
enryption type DES cbc mode with RSA-MD5 added to keytab WRFILE:/etc/krb5/krb5.keytab.
Entry for principal host/britmanual07.isst.aus.stglabs.ibm.com with kvno 4,
enryption type AES-128 CTS mode with 96-bit SHA-1 HMAC added to keytab WRFILE:/etc/krb5/krb5.keytab.
kadmin: ktadd host/britmanual08.isst.aus.stglabs.ibm.com
Entry for principal host/britmanual08.isst.aus.stglabs.ibm.com with kvno 4,
enryption type Triple DES cbc mode with HMAC/shal added to keytab WRFILE:/etc/krb5/krb5.keytab.
Entry for principal host/britmanual08.isst.aus.stglabs.ibm.com with kvno 4,
enryption type ArcFour with HMAC/md5 added to keytab WRFILE:/etc/krb5/krb5.keytab.
Entry for principal host/britmanual08.isst.aus.stglabs.ibm.com with kvno 4,
enryption type AES-256 CTS mode with 96-bit SHA-1 HMAC added to keytab WRFILE:/etc/krb5/krb5.keytab.
Entry for principal host/britmanual08.isst.aus.stglabs.ibm.com with kvno 4,
enryption type DES cbc mode with RSA-MD5 added to keytab WRFILE:/etc/krb5/krb5.keytab.
Entry for principal host/britmanual08.isst.aus.stglabs.ibm.com with kvno 4,
enryption type AES-128 CTS mode with 96-bit SHA-1 HMAC added to keytab WRFILE:/etc/krb5/krb5.keytab.
kadmin:

Now, we can see that the /etc/krb5/krb5.keytab file is created. You can list the entries within this file using the klist -k command. Refer to the following output of the klist -k command.
5. We need to securely transfer this file to the other Kerberos client systems using application such as sftp and make sure that it is placed in the `/etc/krb5/` folder. To ensure that we are referring to the same keytab file, we can list the keytab file entries on the other Kerberos client system. Refer to the following example that shows how to verify keytab file entries.

```
[root@britmanual07 /]
# klist -kt
Keytab name:  FILE:/etc/krb5/krb5.keytab
KVNO Timestamp         Principal
---- ----------------- --------------------------------------------------------
 4 04/20/17 07:14:48 host/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM
 4 04/20/17 07:14:48 host/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM
 4 04/20/17 07:14:48 host/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM
 4 04/20/17 07:14:48 host/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM
 4 04/20/17 07:14:48 host/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM
 4 04/20/17 07:15:00 host/britmanual08.isst.aus.stglabs.ibm.com@NFSREALM
 4 04/20/17 07:15:00 host/britmanual08.isst.aus.stglabs.ibm.com@NFSREALM
 4 04/20/17 07:15:00 host/britmanual08.isst.aus.stglabs.ibm.com@NFSREALM
 4 04/20/17 07:15:00 host/britmanual08.isst.aus.stglabs.ibm.com@NFSREALM
 4 04/20/17 07:15:00 host/britmanual08.isst.aus.stglabs.ibm.com@NFSREALM
```
```
[root@britmanual07 /]
```

6. Obtain an initial TGT on the `britmanual07` system, which is also the IPsec initiator system. We can obtain the TGT from the keytab file entry created for this principal using the `kinit` command with `-k` option. We can list the ticket properties using the `klist` command. Refer to the following output of the `kinit` and `klist` commands to get the initial TGT.

```
[root@britmanual07 /]
# kinit -k host/britmanual07.isst.aus.stglabs.ibm.com
[root@britmanual07 /]
# klist -a
Ticket cache:  FILE:/var/krb5/security/creds/krb5cc_0
Default principal:  host/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM
Valid starting     Expires            Service principal
04/20/17 07:23:26  04/21/17 07:23:26  krbtgt/NFSREALM@NFSREALM
Addresses: (none)
```
```
[root@britmanual07 /]
```

After completing the Kerberos server and client configuration, you can configure IPsec on these Kerberos client systems.

2. Configure IPsec initiator and responder systems with Kerberos authentication
The below steps need to be performed to configure IPsec initiator and responder systems

1. Enable the Internet Key Exchange (IKE) interface on the IPsec initiator and responder systems. The following example shows the set of commands to enable IKE.

```
[root@britmanual07] /
# smit IPsec4
Default rule for IPv4 in ODM has been changed.
Successfully set default action to PERMIT
```

```
Default rule for IPv4 in ODM has been changed.
Successfully set default action to PERMIT
```

```
Before command completion, additional instructions may appear below.
```

```
IPsec_v4 Available
Default rule for IPv4 in ODM has been changed.
Successfully set default action to PERMIT
```

```
Default rule for IPv4 in ODM has been changed.
Successfully set default action to PERMIT
```

```
Before command completion, additional instructions may appear below.
```

2. Prepare and update the IPsec configuration file in the IKE database.

The first step to configure IPsec on the AIX system is to prepare the IPsec configuration file on the initiator and responder systems. This configuration file is an XML file and do note that 'kerberos authentication' will be mentioned as the *AuthenticationMethod*

The following XML code can be used on the IPsec initiator system (britmanual07). Ensure that the values of *IPV4_Address of IKELocalIdentity and IKERemoteIdentity* are changed accordingly.

```
[root@britmanual07] /
# cat ipsec_Kerberos.xml
<?xml version="1.0"?>
<AIX_VPN
  Version="2.0">
  <IKEProtection
    IKE_Role="Both"
    IKE_XCHGMode="Main"
    IKE_KeyOverlap="10"
    IKE_Flags_UseCRL="No"
    IKE_ProtectionName="P1Pol"
    IKE_ResponderKeyRefreshMaxKB="200"
    IKE_ResponderKeyRefreshMinKB="1"
    IKE_ResponderKeyRefreshMaxMinutes="480"
    IKE_ResponderKeyRefreshMinMinutes="1">
    <IKETransform
      IKE_Hash="MD5"
      IKE_DHGroup="1"
      IKE_Encryption="DES-CBC"
      IKE_KeyRefreshMinutes="480"
      IKE_AuthenticationMethod="GSSAPI_Krb5"/>
```

The following XML code can be used on the IPsec responder system (britmanual08). Ensure that the values of IPV4_Address of IKELocalIdentity and IKERemoteIdentity are changed accordingly.

[root@britmanual08] /
# cat ipsec_Kerberos.xml
<?xml version="1.0"?>
<AIX_VPN
  Version="2.0">
  <IKEProtection
    IKE_Role="Both"
    IKE_XCHGMode="Main"
    IKE_KeyOverlap="10"
    IKE_FLAGS_UseCRL="No"
    IKE_ProtectionName="P1Pol"
    IKE_ResponderKeyRefreshMaxKB="200"
    IKE_ResponderKeyRefreshMinKB="1"
    IKE_ResponderKeyRefreshMaxMinutes="480"
    IKE_ResponderKeyRefreshMinMinutes="1">
    <IKETransform
      IKE_Hash="MD5"
      IKE_DHGroup="1"
      IKE_Encryption="DES-CBC"
      IKE_KeyRefreshMinutes="480"
      IKE_AuthenticationMethod="GSSAPI_Krb5"/>
  </IKEProtection>
  <IKETunnel
    IKE_TunnelName="P1"
    IKE_ProtectionRef="P1Pol"
    IKE_FLAGS_AutoStart="Yes"
    IKE_FLAGS_MakeRuleWithOptionalIP="No">
    <IKELocalIdentity>
      <IPV4_Address
        Value="10.33.0.125"/>
    </IKELocalIdentity>
    <IKERemoteIdentity>
      <IPV4_Address
        Value="10.33.0.121"/>
    </IKERemoteIdentity>
  </IKETunnel>
  <IPSecProposal
    IPSec_ProposalName="P2Prop">
    <IPSecAHProtocol
      AH_KeyRefreshKB="0"
      AH_Authentication="AH_MD5"
      AH_EncapsulationMode="Transport"
      AH_KeyRefreshMinutes="580"/>
    <IPSecESPProtocol
      ESP_Encryption="ESP_DES"
      ESP_KeyRefreshKB="0"
      ESP_Authentication="HMAC-MD5"
      ESP_EncapsulationMode="Transport"
      ESP_KeyRefreshMinutes="580"/>
  </IPSecProposal>
  <IPSecProtection
    IPSec_Role="Both"
    IPSec_KeyOverlap="10"
    IPSec_ProposalRefs="P2Prop"
    IPSec_ProtectionName="P2Pol"
    IPSec_InitiatorDHGroup="0"
    IPSec_ResponderDHGroup="NO_PFS GROUP_1 GROUP_2 GROUP_5"
    IPSec_FLAGS_UseLifeSize="No"
    IPSec_FLAGS_UseCommitBit="No"
    IPSec_ResponderKeyRefreshMaxKB="200"
    IPSec_ResponderKeyRefreshMinKB="1"
    IPSec_ResponderKeyRefreshMaxMinutes="220"
    IPSec_ResponderKeyRefreshMinMinutes="1"/>
  </IPSecProtection>
  <IPSecTunnel
    IKE_TunnelName="P1"
    IPSec_TunnelName="P2"
    IPSec_ProtectionRef="P2Pol"
    IPSec_FLAGS_OnDemand="Yes"/>
</AIX_VPN>
Configure an IPsec environment using the Kerberos authentication mechanism

3. Before loading the IPsec configuration file, we need to clear any older configuration present in the database. The following example provides a set of commands to clean the old configuration. Make sure to clear the old configuration on both, initiator and responder systems.

```
[root@britmanual07] /
  #ikedb -x
P1_ITD database created successfully
P2_ITD database created successfully
P1_PREKEY database created successfully
PROPOSAL_LIST database created successfully
PROPOSAL database created successfully
POLICY database created successfully
GROUP database created successfully
NDBM:/etc/IPsec/inet/DB/privkey
[root@britmanual07] /
  #
[root@britmanual08] /
  #ikedb -x
P1_ITD database created successfully
P2_ITD database created successfully
P1_PREKEY database created successfully
PROPOSAL_LIST database created successfully
PROPOSAL database created successfully
POLICY database created successfully
GROUP database created successfully
NDBM:/etc/IPsec/inet/DB/privkey
[root@britmanual08] /
  #
```

4. Load the IPsec configuration file into the database on both IPsec initiator and responder systems. The following example shows how to load the IPsec configuration into database.

```
[root@britmanual07] /
  #ikedb -p IPsec_Kerberos.xml
[root@britmanual07] /
  #
[root@britmanual08] /
  #ikedb -p IPsec_Kerberos.xml
[root@britmanual08] /
  #
```

5. Start the IPsec daemons (iked, tmd, cpsd, isakmpd) on the IPsec initiator and responder systems – and make sure that all the daemons as active. The following example shows how to start the IPsec daemons.

```
[root@britmanual07] /
  #stopsrc -g ike
0513-004 The Subsystem or Group, ike, is currently inoperative.
[root@britmanual07] /
  #lssrc -g ike
Subsystem       Group    PID          Status
  cpsd          ike      inoperative
```
Configure an IPsec environment using the Kerberos authentication mechanism

6. Start the IKE channel from the initiator system and ensure that the local ID and remote ID IP address matches with the systems across which we want to create the IPsec channel. The following example includes the commands to start the IKE channel on initiator system.

```bash
[root@britmanual07] / # startsrc -g ike 0513-059 The cpsd Subsystem has been started. Subsystem PID is 21626958. 0513-059 The tmd Subsystem has been started. Subsystem PID is 17891446. 0513-059 The iked Subsystem has been started. Subsystem PID is 19267598.
[root@britmanual07] / # lssrc -g ike Subsystem Group PID Status
cpsd ike 21626958 active
tmd ike 17891446 active
iked ike 19267598 active
[root@britmanual08] / # ike cmd=activate Phase 2 tunnel 1 On_Demand activate request initiated.
[root@britmanual07] / # ike cmd=list Phase  Tun Id  Status  Local Id    Remote Id
  1  1   Dormant  N/A       10.33.0.125
  2  1   Dormant 10.33.0.121 10.33.0.125
[root@britmanual08] /
```

---

Configure an IPsec environment using the Kerberos authentication mechanism
7. We can make sure that Kerberos authentication has taken place by verifying the Kerberos ticket exchanged. Run the `klist` command to list the tickets. Refer to the following output of the `klist` command.

```
[root@britmanual07] /  
# klist
Ticket cache: FILE:/var/krb5/security/creds/krb5cc_0
Default principal: host/britmanual07.isst.aus.stglabs.ibm.com@NFSREALM
Valid starting     Expires            Service principal
09/10/14 01:49:09  09/11/14 01:49:09  krbtgt/NFSREALM@NFSREALM
09/10/14 01:59:26  09/11/14 01:49:09 host/britmanual08.isst.aus.stglabs.ibm.com@NFSREALM
[root@britmanual08] /  
```

As we can see, a ticket host or an FQDN of the responder is obtained by the initiator system. This acts as service principal name. We were also not prompted for password because we had the `keytab` entry created on the responder system.

**Conclusion**

This article helps users understand how to configure an IPsec channel across two systems with Kerberos authentication. Kerberos authentication is more secured compared to a pre-shared key and it also provides the single sign-on (SSO) feature. So, with the same Kerberos setup, we can as well achieve user authentication for other applications without the need to enter password every time.

**Resources**

- Installing and configuring Kerberos
- IPsec overview

© Copyright IBM Corporation 2017

**Trademarks**
(www.ibm.com/developerworks/ibm/trademarks/)