Single Sign-On with SAP HANA® Database using Kerberos and Microsoft Active Directory

Applicable Releases:
SAP HANA SPS05 (Revision 45) and above (Kerberos)
SAP HANA SPS07 (Revision 70) and above (SPNEGO for SAP HANA XS)

Topic Area:
Installation, Configuration, Security, Troubleshooting

Capability:
SAP HANA Database, Single Sign-On, SSO, Kerberos, Active Directory, SAP HANA XS, SPNEGO

Version 1.5
April 2014
## Document History

<table>
<thead>
<tr>
<th>Document Version</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>First official release of this guide</td>
</tr>
<tr>
<td>1.1</td>
<td>Section about hostname resolution</td>
</tr>
</tbody>
</table>
| 1.2              | - Hardening the keytab file: Set read-only, but only for `<SID>adm`  
                    - Use strong password for Service User |
| 1.3              | - Recommendation to change Service User password periodically  
                    - Recommendation to create dedicated Active Directory Organizational Unit for SAP HANA database SSO administration  
                    - `kinit -k <Service Principal Name>` for verification of keytab |
| 1.4              | - Recommend `aes256-cts` instead of `rc4-hmac` as encryption type  
                    - SPNEGO support for SAP HANA XS |
| 1.5              | - ODBC troubleshooting  
                    - Removed `kinit -k` verification, since it works only with some combinations of KDC/encryption type  
                    - Description how to map FQDN server names to Kerberos realms on Windows using `ksetup`.  
                    - Recommendation to use separate service accounts for SPNEGO and Kerberos  
                    - Recommendation to use SPNEGO only with SSL |
## Typographic Conventions

<table>
<thead>
<tr>
<th>Type Style</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Example Text</em></td>
<td>Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Cross-references to other documentation.</td>
</tr>
<tr>
<td><em>Example text</em></td>
<td>Emphasized words or phrases in body text, graphic titles, and table titles.</td>
</tr>
<tr>
<td><em>Example text</em></td>
<td>File and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.</td>
</tr>
<tr>
<td><em>Example text</em></td>
<td>User entry texts. These are words or characters that you enter in the system exactly as they appear in the documentation.</td>
</tr>
<tr>
<td>&lt;Example text&gt;</td>
<td>Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.</td>
</tr>
<tr>
<td>EXAMPLE TEXT</td>
<td>Keys on the keyboard, for example, F2 or ENTER.</td>
</tr>
</tbody>
</table>

## Icons

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🔴</td>
<td>Caution</td>
</tr>
<tr>
<td>🔴</td>
<td>Important</td>
</tr>
<tr>
<td>🔴</td>
<td>Note</td>
</tr>
<tr>
<td>🔴</td>
<td>Recommendation or Tip</td>
</tr>
<tr>
<td>🔴</td>
<td>Example</td>
</tr>
</tbody>
</table>
# Table of Contents

1. **Scenario**........................................................................................................................................... 1  
   1.1 Single Sign-on ................................................................................................................................. 1  
   1.2 Definitions .................................................................................................................................. 1  

2. **Background Information**.................................................................................................................... 3  
   2.1 Microsoft Active Directory .............................................................................................................. 3  
   2.3 Kerberos ...................................................................................................................................... 4  
   2.5 SPNEGO ..................................................................................................................................... 7  

3. **Prerequisites**........................................................................................................................................ 8  
   3.1 Network Requirements .................................................................................................................... 8  
   3.2 Software Requirements ................................................................................................................... 8  
   3.3 Organizational Requirements ......................................................................................................... 8  
   3.4 Related Information ....................................................................................................................... 9  
       3.4.1 SAP HANA Database ............................................................................................................... 9  
       3.4.2 SAP Web Dispatcher .............................................................................................................. 9  
       3.4.3 Kerberos/ Active Directory ...................................................................................................... 9  

4. **Step-by-Step Procedure**....................................................................................................................... 10  
   4.1 Hostname Resolution ...................................................................................................................... 10  
       4.1.1 Setup ................................................................................................................................. 10  
       4.1.2 Verification ........................................................................................................................ 11  
   4.2 SAP HANA Database Server krb5.conf ....................................................................................... 12  
       4.2.1 Setup ................................................................................................................................. 12  
       4.2.2 Verification ........................................................................................................................ 13  
   4.3 Create the SAP HANA Database Service User(s) ....................................................................... 14  
       4.3.1 Setup ................................................................................................................................. 14  
       4.3.2 Verification ........................................................................................................................ 15  
   4.4 Register Service Principal Name in AD ....................................................................................... 16  
       4.4.1 Setup of Kerberos for SAP HANA DB ............................................................................... 16  
       4.4.2 Setup of SPNEGO for SAP HANA XS ............................................................................... 16  
       4.4.3 Verification ........................................................................................................................ 17  
   4.5 Create keytab for the DB Server ................................................................................................. 17  
       4.5.1 Creating the keytab on the DB Server (ktutil) ..................................................................... 18  
       4.5.2 Alternative: Creating the keytab on the AD domain controller (ktpass) ......................... 18  
       4.5.3 Securing the keytab ............................................................................................................ 19  
       4.5.4 Verification ........................................................................................................................ 19  
   4.6 Changing the Service User Password......................................................................................... 19  
   4.7 Creating an External SAP HANA Database User .................................................................... 20  
       4.7.1 Verification of Kerberos for SAP HANA DB ................................................................. 21  
       4.7.2 Verification of SPNEGO for SAP HANA XS ................................................................. 24  

5. **Appendix: Troubleshooting**.............................................................................................................. 25
1. Scenario

This how-to guide describes the steps necessary for setting up Single Sign-on (SSO) with SAP HANA® database using Kerberos and Microsoft Active Directory. Each step is accompanied by one or more verification steps allowing a systematic completion of the process.

With SPS07, SPNEGO is made available as an authentication option for SAP HANA XS.

Note
Configuration of Kerberos for SAP HANA DB and SPNEGO for SAP HANA XS consists of mainly the same steps. In particular, if the SAP HANA DB was already configured for Kerberos before, only a small number of additional configuration steps are necessary for activating SPNEGO for SAP HANA XS.

Since the steps for configuring Kerberos for SAP HANA DB and SPNEGO for SAP HANA XS are mainly the same, SPNEGO will only be mentioned explicitly in case there is some specific configuration necessary.

The appendix contains a dedicated troubleshooting section.

1.1 Single Sign-on

SSO allows a user to log on once and gain access to multiple systems and services without being asked to produce credentials again.

Kerberos is one of many ways for realizing SSO (other examples are SAML or X.509 certificates).

Depending on how SSO has been setup, it could permit the user logon to just a front end application or it can enable SSO all the way down to the database in what's known as SSO to database (SSO2DB). In case the user doesn't use direct SAP HANA database interfaces such as SAP HANA Studio or hdbsql, but e.g. an SAP BusinessObjects application (BI Launchpad, WEBI etc.), SSO2DB is realized via delegation.

Example
An example of SSO that is relevant to many office workers day-to-day is the use of Microsoft Outlook and the absence of a login and password to access your email and address book. When a user logs into a workstation, they enter a username and password. Shortly afterwards the desktop appears. If you start Outlook, you are not prompted for the login and password you just entered. The mechanisms of this are described in detail later in this document.

1.2 Definitions

The following terms and example values will be used throughout this text:

- **Active Directory Domain Name** – *MYDOMAIN.COM*. What is called domain name in AD is called **Realm Name** in Kerberos speak.

  Recommendation
  The convention for realm names is to use upper case letters.

- **Network Domain Name** - *mydomain.com*.
Recommendation
The convention for network names is to use lower case letters.

Important
While Windows is case-insensitive, Linux is not. This becomes crucial when e.g. a keytab is exported from Windows using ktpass and imported into the Linux based SAP HANA database server.

- **DB Server** – *myhdbserver.mydomain.com*.
The DB Server is the server where the SAP HANA database is installed.

  Important
  The DB server is denoted by its fully qualified canonical domain name.

- **SAP HANA Database Service User for SSO mapped in AD** – *myhdbserver*
  A Service User is technically a plain Active Directory user account, which doesn't belong to a natural person, but an IT service.

  Recommendation
  The convention for the SAP HANA database Service User is to use the simple hostname of the DB server.

- **Active Directory test user** – *aduser1*

These will need to be changed based on what you have setup and how complex your environment is.
2. Background Information

2.1 Microsoft Active Directory

Active Directory is a repository for information about objects that reside on an enterprise network, such as users, groups, computers, printers, applications, and files. It is a centralized IT infrastructure being used within nearly all enterprises today. It was introduced with Windows 2000 Server. Key version numbers were introduced with Windows Server 2003, while strong encryption using AES 128/256 was only introduced with Windows Server 2008. The current version is Windows Server 2012.

⚠️ CAUTION

Typically, AD administration is done by a different organization within the enterprise than SAP HANA database administration. This might require careful planning of configuration steps involving AD and might also slow down troubleshooting.

This is also a reason why we recommend creating the keytab on the DB server using hdbkrbconf.py (or directly ktutil addent) instead of exporting it from Windows using ktpass. It reduces the dependency on the AD admins.

In addition, you should create a dedicated OU for the SAP HANA database administration.
2.3 Kerberos

Kerberos is an authentication protocol (RFC 4120) originally developed at the Massachusetts Institute of Technology (MIT) which permits a client to authenticate with a service via a Service Ticket specific to the service in question. This ticket will be issued by a Key Distribution Center (KDC) based on a meta-ticket specific to the client, the Ticket Granting Ticket (TGT).

The service is known in the network under its Service Principal Name (SPN) which is a logical mapping to an AD account where the service is registered as a Service User. In a Windows environment, this TGT is automatically acquired during operating system user logon, while on Linux it is typically acquired ad-hoc using the kinit tool.

![Diagram of Kerberos authentication protocol steps](adapted from [1])

Part of the authentication protocol messages are encrypted using a symmetric session key. Several encryption algorithms (encryption types) are supported. The most wide-spread algorithm used to be rc4-hmac, but this is no longer considered secure. Now the default should be aes256-cts.

The session key will be created by the KDC dynamically, but it is based on a longtime user or service key, respectively. For an operating system user, the longtime key can always be restored by asking for the password. For an IT service on the other hand, this key is stored on the server in a so-called keytab file. We will have to create a keytab for the DB server. The keytab contains one or more keys (depending on the number of supported encryption types) for one or more service principal names.

Note that in the above figure there is no communication between the service and the KDC! In fact, the only piece of communication needed by the service is the keytab. From there, it gets its longtime service key(s) allowing the service to decrypt the service ticket sent by the client. It contains the session key needed for decrypting the client authenticator. By checking the contents of the authenticator, which in addition to the client name contains a timestamp and an expiry date, the authentication is completed on the server side. Since the timestamp is a critical part of the protocol, it is important that all hosts involved have synchronized clocks. For mutual authentication, the server also sends such an authenticator to the client.
Since with MIT Kerberos the keytab is stored in the file system, we rely on the secureness of the particular host and the privacy of the key(s). As Kerberos is using symmetric cryptography, in case either the host or the service key is compromised the following attacks are possible:

1. Impersonating the service:
   Authentication of the service (mutual authentication) relies on the privacy of the service key. Anybody knowing the key would be able to decrypt the AP_REQ and forge a service authenticator.

2. Impersonating arbitrary users:
   The validation of service tickets relies on the privacy of the service key. A service ticket is valid when it is encrypted with the service key, because it is assumed that only the KDC knows this key. Anybody knowing the service key or the service user’s password (the key is generated from the password using a fixed scheme) can forge service tickets, claiming to be originating from arbitrary user principals. It is not possible for the service to distinguish such fabricated Kerberos tickets from tickets issued by the KDC.

To minimize the above risks, read access to the keytab file should be restricted to a single user, which would be the <SID>adm in SAP HANA database. Second, the password for the Service User representing the SAP HANA database should be strong and it should be changed periodically.

Kerberos can be run via UDP or TCP, but we highly recommend TCP. When run via UDP, sporadic logon errors may occur, which are difficult to troubleshoot.

Kerberos SSO can be used for direct connections to the SAP HANA database from e.g. SAP HANA studio or hdbsql. In addition, it is also possible to propagate user identity from e.g. WEBI over BI Server to SAP HANA database. Such an authentication from some frontend client via a middle tier service to a backend service is realized using Kerberos delegation. To this end, a specific version of a user’s TGT, a forwardable ticket granting ticket, is employed as the following figure explains.
Single Sign-On with SAP HANA® Database using Kerberos and Microsoft Active Directory

Figure: Protocol steps involved in Kerberos delegation (e.g. WEBI <-> BI Server <-> DB Server)
Adapted from [1].

In the example, the BI server logs into SAP HANA database on behalf of the AD user who has logged into WEBI.

The SAP HANA database server, the SAP HANA studio and the SAP HANA ODBC client are based on the MIT Kerberos V5 implementation, while the SAP HANA JDBC client is based on the Java Development Kit (JDK) implementation of the protocol.
2.5 SPNEGO

The Simple and Protected GSS-API Negotiation (SPNEGO) is a protocol for client and server to determine a common set of SSO mechanisms. It was specified in RFC 4178, and the corresponding HTTP binding in RFC 4599 (the latter can be seen as a natural HTTP extension of RFC 2617, the HTTP authentication via challenge/response).

The negotiation became necessary because in the time when Kerberos was introduced, in Windows networks there were legacy SSO protocols like NTLM, which were meant to be usable as a fallback mechanism in case Kerberos is not available. However, many servers (including SAP HANA XS) don’t support anything else but Kerberos 5 as mechanism and therefore do not support the actual negotiation step, where the client in the first protocol message lists the supported mechanisms, and the server then determines an appropriate one; rather, the client has to include a Kerberos 5 token already in the first message. This behavior is implemented by all major web browsers.

An example HTTP request for a web site www.foo.com/index.html, which has been configured for SPNEGO authentication, results in the following protocol flow between client (C) and server (S):

1. C: Initial request
   GET www.foo.com/index.html

2. S: Challenge - ask for SPNEGO authentication
   HTTP/1.1 401 Unauthorized
   WWW-Authenticate: Negotiate

3. C: Response – initialize Kerberos context and put resulting token into HTTP header
   GET www.foo.com/index.html
   Authorization: Negotiate <Kerberos Token>

4. S: Accept Kerberos context and put resulting token into HTTP header
   HTTP/1.1 200 Success
   WWW-Authenticate: Negotiate <Kerberos Token>

In order to avoid man-in-the-middle attacks, it is strictly recommended to protect resources where users authenticate via SPNEGO with SSL: The HTTP server should require SSL connections, refusing or redirecting any plain HTTP requests. Even better, the server should return HTTP Strict Transport Security response headers to instruct the browser to use only SSL from now on. To this end, you can set the flag Prevent Public Access for Sub-Packages in the SAP HANA XS Admin tool.

Furthermore, you should create separate service users for Kerberos and SPNEGO. Otherwise, since plain Kerberos tokens travel over the network in the clear, it would be possible for an attacker to capture such a packet and forge an HTTP Negotiate request from that.
3. Prerequisites

3.1 Network Requirements

1. The clocks of all hosts involved are synchronized.
2. On the Active Directory domain controller, Kerberos is forced to use TCP instead of UDP (see http://support.microsoft.com/kb/244474/en-us for reference)
3. Hostname reverse lookup (/etc/hosts on the DB server and/or DNS record type PTR in Active Directory) is set up for “physical” and “virtual” DB server hostname(s).
4. On the DB server, hostname resolution must be consistent with reverse lookup.
5. A “virtual” hostname must actually be a DNS alias, while the “physical” hostname must be a canonical name.

Important
A virtual hostname aka DNS alias must be realized using a DNS CNAME record, while the corresponding physical hostname must be registered as an A record in DNS.

3.2 Software Requirements

It is assumed that the SAP HANA database has been installed on the DB Server. In addition, the MIT implementation of the Kerberos client and server libraries must be installed (if this is not the case, contact the responsible administrator). To verify that the software requirements are met, please run:

```
[d058623@lu002600 ~]$ rpm -qa | grep krb5
krb5-1.6.3-133.48.48.1
krb5-client-1.6.3-133.48.48.1
krb5-32bit-1.6.3-133.48.48.1
```

The version numbers don’t have to match exactly, but should be above 1.6.3-132 to include important security patches.


3.3 Organizational Requirements

Before setting up SSO, you will need to satisfy the following prerequisites:

- Root access to the DB server
- Active Directory user: will be used for verifying SSO to SAP HANA database
- Active Directory Administration: You need an administrator with the right to manage Users and Computers for the Active Directory Organizational Unit where the SAP HANA database Service User shall be maintained.

Recommendation
Create a dedicated OU for the objects related to the SAP HANA database configuration, e.g. OU=HANA_SSO,OU=Resources,DC=MYDOMAIN,DC=COM and grant the SAP HANA database administrator the authorizations to Create, delete and manage user accounts for this OU.
Familiarity with Microsoft Active Directory and Kerberos is a highly recommended. Find some resources for getting started in the next chapter.

3.4 Related Information

3.4.1 SAP HANA Database


SAP HANA database Kerberos/ AD configuration and validation script hdbkrbconf.py: SAP Note 1813724

3.4.2 SAP Web Dispatcher


3.4.3 Kerberos/ Active Directory

For anyone needing background information on Kerberos or Active Directory troubleshooting, the following web sites may be of interest:

- MS AD Troubleshooting:

We recommend the following books:

3. de Boer, Martijn et al: Single Sign-on mit SAP: Lösungen für die Praxis. SAP Press, 2010
4. **Step-by-Step Procedure**

The configuration of SAP HANA database SSO with Kerberos consists of few, albeit error-prone steps. We have to create a Service User representing SAP HANA database in AD, being mapped by a Service Principal Name. Then we have to create a keytab for this SPN on the DB server. On the DB server, we also need a krb5.conf file. Finally, we have to create an externally-mapped SAP HANA database user.

**Recommendation**

Major parts of the SAP HANA database Kerberos configuration can be setup and verified automatically using the script hdbkrbconf.py (*SAP Note 1813724*).

4.1 **Hostname Resolution**

4.1.1 **Setup**

When doing hostname resolution, you want to know one of the following:

1. IP address(es) given a hostname (potentially an alias)
2. Full qualified domain domain name of the host given a simple hostname (or alias)
3. Canonical hostname of the host given a simple hostname (or alias)
4. Hostname given an IP address (*reverse lookup*)

On Linux there are several tools for hostname resolution. Some of them use DNS directly (*dig, host, nslookup*), while others (*hostname*) follow the convention to first check whether a file-based local resolution should be preferred over DNS, mimicking the behaviour of the system calls `gethostbyname` and `gethostbyaddr`. In this case the order that will be used is specified in the following files:

1. `/etc/nsswitch.conf`
2. `/etc/host.conf`

Which file is actually used depends on the library or the system call, respectively. Typically, host local lookup has precedence, as configured in the following examples:

```bash
#> cat /etc/nsswitch.conf | grep hosts
hosts: files dns

#> cat /etc/host.conf | grep hosts
# This line should be in sync with the "hosts"
order hosts, bind
```

The main point here is that the reverse lookup must be configured consistently with the hostname lookup, since this will be used by the SAP HANA database server Kerberos implementation for determining the SPN. The SPN in turn will be used for the lookup into the keytab.

The local hostname resolution is made via the file `/etc/hosts`, for example:

```bash
#> cat /etc/hosts | grep myhdbserver
<IP address> myhdbserver.mydomain.com myhdbserver
```

where `<IP address>` has to be replaced with the primary IP address of the DB server.
CAUTION

In Windows, hostnames are sometimes created in upper case. However, the MIT Kerberos implementation used by SAP HANA database strictly requires a lower case hostname in the keytab entry, so the respective SPN must be created with a lower case hostname part in Active Directory.

For changes to /etc/hosts taking effect, you have to flush the nscd DNS cache:

`#> sudo /etc/init.d/nscd restart`

In case /etc/nsswitch.conf or /etc/host.conf specify that DNS should be used, this has to be configured in /etc/resolv.conf, for example:

`#> cat /etc/resolv.conf`

domain dhcp.mydomain.com
search sub1.mydomain.com sub2.mydomain.com mydomain.com
nameserver 10.17.220.72

Please find detailed information about the hostname resolution on the DB server running:

`#> man hostname`
`#> man host`
`#> man nsswitch.conf`
`#> man host.conf`
`#> man resolv.conf`

4.1.2 Verification

On the DB server, run:

`#> hostname --fqdn`
myhdbserver.mydomain.com

for getting the FQDN of the DB server.

Run:

`#> hostname --ip-address`

<IP address>

for getting the respective IP address.

For checking the reverse lookup, run:

`#> python <<EOF`
`> import socket`
`> host = socket.gethostbyaddr('<IP address>')[0]`
`> print host`
`> EOF`

myhdbserver.mydomain.com

where <IP address> has to be replaced with the primary IP address of the DB server.
The above python script does exactly the same as the SAP HANA database server when it constructs the expected SPN. The result of the above python script must equal the result of the first command, hostname -fqdn.

4.2 SAP HANA Database Server krb5.conf

On the DB server, a configuration file for the MIT Kerberos libraries, krb5.conf, needs to be setup. Please find detailed information on the krb5.conf on the DB server running

```
#> man krb5.conf
```

4.2.1 Setup

We can differentiate between several krb5.conf use cases. The MIT Kerberos libraries all use the same format, but depending on the use case, many parameters from other use cases are irrelevant (or may even be detrimental). The following use cases exist:

1. KDC: In case you want to run an MIT Kerberos based KDC (not applicable)
2. Client: When you want to connect from a Linux box as a client (e.g. hdbsql) to another server (e.g. SAP HANA database on another or the same host)
3. Server: The server could be the SAP HANA database and this is exactly the case at hand

Technically, in many customer installations using a DNS Kerberos configuration (/etc/bind entries like _kerberos._tcp.REALM etc.), the only thing the DB server needs is this stripped-down krb5.conf:

```
#> cat /etc/krb5.conf
[domain_realm]
    .mydomain.com = MYDOMAIN.COM
    mydomain.com = MYDOMAIN.COM
```

In case Kerberos is not configured in DNS, the SAP HANA database server will additionally need the default_realm and the [realm] specification. Make sure that the Active Directory domain (MYDOMAIN.COM in the example) is consistent across default_realm, realms, domain_realm. Furthermore, the keytab for the DB server must contain an entry exactly matching this Active Directory domain, i.e. the SAP HANA Database Service User must be created within this domain. Creation of the Service User and the keytab will be described below.

Standard Kerberos configuration:

```
#> cat /etc/krb5.conf
[libdefaults]
    default_realm = MYDOMAIN.COM

[realms]
 MYDOMAIN.COM = {
    kdc = mykdc1.mydomain.com
    kdc = mykdc2.mydomain.com
 }

[domain_realm]
    .mydomain.com = MYDOMAIN.COM
    mydomain.com = MYDOMAIN.COM
```
Important

The [domain_realm] covers only the mapping for the DB server domain (it has nothing to do with the client domain(s)!) This will be used in mutual authentication when the SAP HANA database client tries to authenticate the SAP HANA database server.

Important

The domain part in the [domain_realm] mapping must consist of the domain name in its full length. In case the FQDN of the DB server is hdbserver.subdomain.domain.com, the [domain_realm] entry has to be

```
.subdomain.domain.com = DOMAIN.COM
subdomain.domain.com = DOMAIN.COM
```

but not

```
.domain.com = DOMAIN.COM
domain.com = DOMAIN.COM
```

Important

In case the IT landscape contains multiple KDC servers, you have to provide a separate configuration line for each of them like in the example above.

For completing the verification steps and a local SSO test later, you might need extensions specific to the client use case.

Important

Changes to the krb5.conf take effect only after restart of the DB server. A restart can be avoided by setting the SAP HANA database trace component Authentication to DEBUG.

Note

In any case, you don’t have to set dns_lookup_kdc (relevant only for client use case) or dns_lookup_realm to true, unless you really use DNS SRV records for maintaining KDC addresses and/or DNS TXT records for mapping DNS domains to Kerberos realms. In doubt, check with your local admin.

4.2.2 Verification

We will now test at once

1. connectivity between the DB server and the Active Directory
2. DB server krb5.conf

When using the kinit and klist utilities, we rely on the proper configuration of the Windows AD test user aduser1. Using kinit we try to authenticate the test user against the AD domain and create a TGT:

```
#> /usr/bin/kinit aduser1@MYDOMAIN.COM
Password for aduser1@MYDOMAIN.COM:
```

Using klist we can see this ticket:

```
#> /usr/bin/klist
Ticket cache: FILE:/tmp/krb5cc_1003
```
Default principal: aduser1@MYDOMAIN.COM

Valid starting  Expires  Service principal
02/18/13 15:25:58  02/19/13 01:26:02  krbtgt/ MYDOMAIN.COM@MYDOMAIN.COM
renew until 02/19/13 15:25:58

4.3 Create the SAP HANA Database Service User(s)

We have to create one or more Service Users representing the SAP HANA database in Active Directory on the AD Domain Controller. As explained in section /Background Information/, you should create separate service users for Kerberos and SPNEGO.

A so-called Service User is technically a plain user account in the first place. It gets its special meaning later, when we map a Service Principal Name to it.

⚠️ Recommendation
We recommend creating a dedicated OU for the SAP HANA database Service User(s). This allows granting the necessary administrative rights for this OU to SAP HANA database admins, which typically wouldn’t have these rights in other OUs (see /Organizational Requirements/).

A Service User should be assigned a strong password (i.e. minimum length 8 characters, not easy to guess, containing digits and special characters). In case the password is compromised, clients knowing it would be able to impersonate anybody when connecting to the SAP HANA database by forging service tickets.

A Service User should be assigned a password which never expires to avoid service downtimes. Nonetheless, you should change it periodically.

Furthermore, a SAP HANA database Service User should not be enabled for any kind of delegation (e.g. Constrained Delegation aka “S4U2Proxy”, aka “Protocol Transition”), because then it could obtain Kerberos tickets for all services configured under Constrained Delegation in the name of arbitrary users. This could be exploited by attackers in case the Service User password is compromised.

4.3.1 Setup

To avoid hitting unwanted checkboxes in the Windows GUIs, we recommend creating the user via CLI. In an elevated Windows command shell, run (choosing a strong password matching the local policy):

```
> dsadd user "cn=myhdbserver,cn=users,dc=MYDOMAIN,dc=COM" -pwd <passwd>
   -disabled no -pwdneverexpires yes -acctexpires never
```

⚠️ CAUTION
Choose a strong password (see explanation above).

⚠️ Recommendation
In case you run into problems, do not blindly rely on the correctness of the service user. There was a Very High customer message due to a service user with an encryption type not supported (it was DES, now considered insecure). Rather, don’t hesitate to start over with a fresh service user (see below).
Important

The Active Directory domain (dc=MYDOMAIN,dc=COM in the example above) must be the same as specified in the DB server’s krb5.conf (see above).

Since we recommend usage of encryption type **aes256-cts** when creating the keytab (see below), you have to explicitly flag the service user for AES256 in the account options:

![Account options](image)

**Figure: Account options for SAP HANA Database service user: AES256.**

For resetting a service user, first delete the existing one:

```bash
> dsrm "cn=myhdbserver,cn=users,dc=MYDOMAIN,dc=COM"
```

Then create it again as above.

### 4.3.2 Verification

On the DB server, run

```bash
# > /usr/bin/kinit myhdbserver@MYDOMAIN.COM
```

to get a TGT for the SAP HANA database service user. You have to supply the password that was used when the service user account was created in AD.

Afterwards, run `klist` to check the resulting ticket cache (example):

```bash
# > /usr/bin/klist
Ticket cache: FILE:/tmp/krb5cc_1003
Default principal: myhdbserver@MYDOMAIN.COM

Valid starting Expires Service principal
02/18/13 15:50:47 02/19/13 01:50:50 krbtgt/MYDOMAIN.COM@MYDOMAIN.COM
  renew until 02/19/13 15:50:47
```
4.4 Register Service Principal Name in AD

Next, we need to create a Service Principal Name for the DB server in AD. The SPN is simply a mapping of a service name to a service user. We need this SPN later, when we create the DB server’s keytab.

4.4.1 Setup of Kerberos for SAP HANA DB

When configuring Kerberos for SAP HANA DB, you have to create exactly one SPN for each physical host of the system.

The SPN for SAP HANA DB must strictly follow the following format:

\[ hdb/<DB\text{ server}>@<domain> \]

where \(<DB\text{ server}>\) should be the fully qualified canonical domain name of the DB server host and \(<domain>\) is the AD Domain Name.

\[ \text{CAUTION} \]

In Windows, hostnames are sometimes created with upper case. However, the SAP HANA implementation strictly requires a lower case hostname in the keytab entry, so the respective SPN must be created with a lower case hostname part in Active Directory.

In an elevated Windows command shell run

\[ > \text{setspn -S hdb/myhdbserver.mydomain.com MYDOMAIN\myhdbserver} \]

\[ \text{Note} \]

In contrast to many other server products, a SAP HANA database client doesn’t have to know this SPN, since it is retrieved from the DB server as the first step in the SAP HANA database authentication protocol.

\[ \text{CAUTION} \]

In case you change something in AD, e.g. the SPN mapping, you have to make sure that this change is replicated. Replication typically happens within the order of minutes, but the interval is configurable.

4.4.2 Setup of SPNEGO for SAP HANA XS

When you configure SPNEGO for SAP HANA XS, you have to create one SPN for each alias of each physical host of the system.

An SPN for SAP HANA XS must strictly follow the following format:

\[ HTTP/<DB\text{ server}>@<domain> \]

where \(<DB\text{ server}>\) should be the fully qualified canonical domain name of the DB server host or any alias of the same and \(<domain>\) is the AD Domain Name.

\[ \text{CAUTION} \]

In Windows, hostnames are sometimes created with upper case. However, the SAP HANA implementation strictly requires a lower case hostname in the keytab entry, so the respective SPN must be created with a lower case hostname part in Active Directory.
In an elevated Windows command shell run

> setspn -S HTTP/myhdbserver.mydomain.com MYDOMAIN\myhdbserver

⚠️ CAUTION

In case you change something in AD, e.g. the SPN mapping, you have to make sure that this change is replicated. Replication typically happens within the order of minutes, but the interval is configurable.

### 4.4.3 Verification

In the following the verification steps for configuring Kerberos for SAP HANA DB are shown. Doing the same for SPNEGO means simply replacing *hdb* by *HTTP* in the SPNs given below.

In a Windows command shell, run

> setspn -L MYDOMAIN\myhdbserver

...to see the SPN mapped to the service user account.

To query the existence of the SPN run

> setspn -Q hdb/myhdbserver.mydomain.com

On the DB server, run

#> /usr/bin/kinit myhdbserver@MYDOMAIN.COM -S \n   hdb/myhdbserver.mydomain.com@MYDOMAIN.COM

...to retrieve a service ticket for the DB server.

The result should be similar to

#> /usr/bin/klist

Ticket cache: FILE:/tmp/krb5cc_1003
Default principal: myhdbserver@MYDOMAIN.COM

Valid starting     Expires            Service principal
02/18/13 20:13:40 02/19/13 06:13:40  hdb/myhdbserver.mydomain.com@MYDOMAIN.COM
renew until 02/19/13 20:13:40

### 4.5 Create keytab for the DB Server

Now we have to create a keytab for the DB server. The keytab stores the keys(s) needed by the SAP HANA database to take part in the authentication protocol.

The keytab can be created in two ways:

1. On the DB server using *ktutil*
2. On the Windows AD domain controller using *ktpass*

We recommend the first approach, since there are some minor drawbacks creating the keytab using *ktpass* on the Windows AD domain controller:

- It requires the availability of an AD administrator (see remark in the introduction)
- *ktpass* also manipulates the user account (e.g. incrementing the key version number, kvno), which is not desired here.
4.5.1 Creating the keytab on the DB Server (ktutil)

For configuring Kerberos for SAP HANA DB, you may consider creating the keytab on the DB server using the script `hdbkrbconf.py` *(SAP Note 1813724)*. How to do it manually is described in the following.

On the DB server, `su` to root:

```
#> su -
Password:
```

Remove the old keytab:

```
$ rm /etc/krb5.keytab
```

Create the new keytab:

```
$ /usr/lib/mit/sbin/ktutil
ktutil: addent -password -p hdb/myhdbserver.mydomain.com@MYDOMAIN.COM -k 2
   -e aes256-cts
Password for hdb/myhdbserver.mydomain.com@MYDOMAIN.COM:
ktutil: wkt /etc/krb5.keytab
ktutil: q
```

**Important**
Changes to the krb5.keytab take effect only after restart of the DB server.

**Note**
In previous versions of this document, the recommended encryption type was `rc4-hmac`. However, RC4 isn't considered secure today. `aes256-cts` offers a much stronger protection and should be the default.

For SPNEGO, you have to repeat the `ktutil` command for each alias of the DB server host by simply replacing `hdb` by `HTTP` in the SPN.

4.5.2 Alternative: Creating the keytab on the AD domain controller (ktpass)

**Recommendation**

We recommend creating the keytab using `ktutil` (see above).

In a Windows console run

```
> ktpass -princ hdb/myhdbserver.mydomain.com@MYDOMAIN.COM -mapuser
MYDOMAIN\myhdbserver -pass <passwd> -out C:\myhdbserver.keytab -ptype
KRBS_NT_PRINCIPAL -crypto AES256-SHA1
```

You will need to copy the keytab created by `ktpass` to the DB server into `/etc/krb5.keytab`. 
Important
Changes to the krb5.keytab take effect only after restart of the DB server. A restart can be avoided by setting the SAP HANA database trace component Authentication to DEBUG.

4.5.3 Securing the keytab
Since the keytab is stored in the file system, you rely on the secureness of the particular host. Otherwise the key(s) could be used for attacks. Therefore, the read access to the keytab file should be restricted to a single user, which is the <SID>adm.

Adapt access permissions for the new keytab:
```
$ chown <SID>adm:sapsys /etc/krb5.keytab
```

As a <SID>adm, run
```
#> chmod 400 /etc/krb5.keytab
```

4.5.4 Verification
Independent on how the keytab was created, you verify the correctness as follows. Only the verification steps for configuring Kerberos for SAP HANA DB are shown. Doing the same for SPNEGO means simply replacing hdb by HTTP in the SPNs given below.

First, on the DB server, check the content of the new keytab:
```
$ /usr/bin/klist -k /etc/krb5.keytab -etK
Keytab name: FILE:/etc/krb5.keytab
KVNO Timestamp Principal
----- --------------- ---------------
 2 09/27/13 15:24:58 hdb/myhdbserver.mydomain.com@MYDOMAIN.COM (AES-256 CTS mode with 96-bit SHA-1 HMAC) (0xecc0f7faee5f6d12e5f06f5e2609d56e50c69ab8a4)
```

Note
From our experience with Windows Server 2008 and Windows Server 2012, the key version number for a fresh user always starts with 2.

Second, verify the consistency of the keytab with the SAP HANA database service data stored in the KDC. As a preparation step for kvno, run
```
#> /usr/bin/kinit myhdbserver@MYDOMAIN.COM
```
to get a TGT for the SAP HANA database service user. To check the correctness of the key version number and the key(s) contained in the keytab run
```
#> /usr/lib/mit/bin/kvno -k /etc/krb5.keytab \\
/hdb/myhdbserver.mydomain.com@MYDOMAIN.COM
```
hdb/myhdbserver.mydomain.com@MYDOMAIN.COM: kvno = 2, keytab entry valid

4.6 Changing the Service User Password
Since the keys stored in the keytab are generated from the Service User password, you should change the Service User password periodically. After the password has been changed, the keytab has to be either created again or extended to contain the new key(s), since a password change implies an increment of the Key Version Number (kvno).

Recommendation
To avoid a downtime, you should add the new key(s) to the existing keytab. This can only be done using ktutil directly (subcommand addent).

Otherwise, if you simply created a new keytab, users currently logged in would have to
log off from their Windows sessions in order to get new service tickets for the SAP HANA database. After a certain transition period you can delete the old key(s) using the `ktutil subcommand delent`.

### 4.7 Creating an External SAP HANA Database User

To complete the SSO configuration for SAP HANA DB, a DB user representing the Windows test user `aduser1` needs to be created in the SAP HANA database user administration.

**Note**

In case you do not have an Active Directory test user ready, you can create it as follows. In an elevated Windows command shell, run

```bash
> dsadd user -upn aduser1 -cn=aduser1,cn=users,dc=MYDOMAIN,dc=COM -pwd <password> -disabled no -pwdneverexpires yes - acctexpires never
```

For the DB user, you can choose any user name within the limits of the SAP HANA database user name rules. However, the *External ID* must exactly match the Windows AD user and domain name, in the format `user@ACTIVE DIRECTORY DOMAIN NAME`, as seen below.

Logon to SAP HANA studio as a `USER_ADMIN` and complete the following steps.

![Figure: Creating new DB user (right-click on “Users”).](image-url)
If you have many users to map, this can be scripted using SQL:
CREATE USER aduser1 WITH IDENTITY 'aduser1@MYDOMAIN.COM' FOR KERBEROS

4.7.1 Verification of Kerberos for SAP HANA DB

4.7.1.1 SSO via hdbsql
To start simple, you may want to test the SAP HANA database SSO configuration by running the hdbsql command. First, you would run it directly from the DB server, and then from a different host.
For troubleshooting, it might be useful to strictly separate a server and client krb5.conf (refer to Running hdbsql on the DB Server for details).

You might have to run the SAP HANA client installation on the client host first.

In the case that you run hdbsql from Linux, please initially execute the following command:

```
#> /usr/bin/kinit aduser1@MYDOMAIN.COM
```
to get a TGT for the test user. From Windows, this is not necessary, since you get the ticket automatically during Windows logon, assuming that the organization uses the same directory for workstation and SAP HANA database logons.

Try connecting without providing user name and password (01 is the SAP HANA database instance number and TDO is the SAP HANA database SID in this example):

```
#> hdbsql -i 01 -n myhdbserver.mydomain.com
```

Welcome to the SAP HANA Database interactive terminal.

Type: \h for help with commands
\q to quit

hdbsql=> \c
Connected to TDO@myhdbserver:30115
hdbsql TDO=>
4.7.1.2 SSO via SAP HANA Studio

When creating a connection to the DB server in SAP HANA studio, the main point is to select the Authentication by current operating system user as seen below.

Screen shots step by step (starting from user SYSTEM):

Figure: SAP HANA studio: Right-click on the left, Add System.
Figure: Specify DB instance to connect to.

Figure: Specify authentication method: Choose SSO (current operating system user).

Figure: After successful connect: Added system *TDO*.
4.7.2 Verification of SPNEGO for SAP HANA XS

4.7.2.1 Setup

First you have to check the web browser configuration. Each browser handles it differently, but usually you have to explicitly allow sites or domains for SPNEGO authentication. E.g., for Mozilla Firefox, you have to type about:config in the address bar and then search for network.negotiate-auth.trusted-uris. Double-click on Value and enter the domain name of the SAP HANA DB server you will use for accessing your XS application, e.g. mydomain.com.

Note: *.mydomain.com won’t work! You cannot use wild cards.

For the MS Internet Explorer the following article explains how to setup the browser for Kerberos authentication:

- Authentication Uses NTLM instead of Kerberos:  
- Unable to negotiate Kerberos authentication after upgrading to Internet Explorer 6:  
  http://support.microsoft.com/kb/299838

4.7.2.2 Adjusting Kerberos Realms on Windows

In order to request a service ticket, the client has to conclude the Kerberos realm (Active Directory domain) from the fully qualified domain name of the DB server used in the URL.

In case the Kerberos realm name is not identical with the DNS domain name, this resolution may lead to wrong realm names.

However you can enforce a Kerberos realm name for a certain server using ksetup.exe.

1. Add the Kerberos realm where your service is registered:
   ksetup /addkdc <RealmName> [<KDCName>]
2. Map the server FQDN to this realm:
   ksetup /addhosttorealmmap <HostName> <RealmName>

4.7.2.3 Deploy Test App

For testing SPNEGO end-to-end, you now have to deploy a test app and configure the authentication method SPNEGO.

You can simply extend the Hello World example from the SAP HANA Developer Guide, ch. 2.5.1 Tutorial: My First SAP HANA Application, in particular 2.5.1.5 Tutorial: Write Server-Side JavaScript.

In the .xsjs file, enter the following code and save the file:

```javascript
$.response.contentType = "text/html";
$.response.setBody("Authenticated user=" + $.session.getUsername())
```

Enter the following lines of code into the .xsaccess file:

```json
{ "exposed" : true }
```

You will have to define the user-authentication method in the application's runtime configuration by using the SAP HANA XS Admin tools (SAP HANA Administration Guide, ch. 5.2 Maintaining Application Runtime Configurations).

After completing these steps please open the xsjs resource in your browser.
5. Appendix: Troubleshooting

There are several sources for information once you run into problems with Active Directory/Kerberos and SAP HANA database. Of course you should always adhere to the verification steps outlined in this text directly after the respective configuration step in the first place. These mainly employ the MIT Kerberos tools on the DB server like kinit, klist and kvno.

Direct feedback is provided by the DB client itself (hdbsql, SAP HANA studio). In addition, you should always check the DB server trace. For SPNEGO, you have to also configure tracing for the SAP Web Dispatcher. On the client side, you may need to activate JDBC, ODBC or SQLDB trace, depending on the client used.

In some cases, this is not sufficient. Then you should consider switching on the Windows System Event Log for Kerberos.

As a mean of last resort, you should consider switching on the SAP HANA database protocol trace, optionally in conjunction with a network sniffer such as Wireshark, Microsoft Network Monitor or tcpdump. Depending on the issue, you might have to run it on the client or server side or both.

Appendix A – SAP HANA Database Server

Authentication trace (SQL):
ALTER SYSTEM ALTER CONFIGURATION ('indexserver.ini', 'SYSTEM') set ('trace', 'authentication') = 'debug' with reconfigure;

Protocol trace:
ALTER SYSTEM ALTER CONFIGURATION ('indexserver.ini', 'SYSTEM') set ('trace', 'row_engine') = 'info' with reconfigure;
ALTER SYSTEM ALTER CONFIGURATION ('indexserver.ini', 'SYSTEM') set ('trace', 'sqlsession') = 'debug' with reconfigure;
ALTER SYSTEM ALTER CONFIGURATION ('indexserver.ini', 'SYSTEM') set ('session', 'protocol_trace') = 'on' with reconfigure;

⚠️ CAUTION
Activating the packet trace can harm the stability of the system due to its massive performance impact.

⚠️ CAUTION
Make sure that trace file limits are configured properly. You might have to increase the maxfiles and/or the maxfilesize parameters.

If necessary, adapt trace file limits:
ALTER SYSTEM ALTER CONFIGURATION ('indexserver.ini', 'SYSTEM') set ('trace', 'maxfiles') = '<number>' with reconfigure;
ALTER SYSTEM ALTER CONFIGURATION ('indexserver.ini', 'SYSTEM') set ('trace', 'maxfilesize') = '<size in bytes>' with reconfigure;

⚠️ CAUTION
Remember to disable tracing once you are done with the analysis.
Appendix B – DB Client

In case the SSO fails from hdbsql or SAP HANA studio, you might get meaningful error messages or not. The reason is that the SAP HANA database protocol layer simply propagates the error message as returned by the underlying Kerberos implementation.

Nonetheless, it is always worthwhile trying to understand the text and trying to find help on that in the internet. While the text is platform dependent, the error code is standardized and may suit better for searches.

Examples:
This would mean that the SPN for the DB server was not registered within Active Directory

This error is caused by a problem within the AD. Either the SAP HANA database SPN is completely missing, or there are several mappings where the SAP HANA database SPN points to different accounts.

Running hdbsql on the DB Server

For ruling out network problems (e.g. DNS), it might be useful to try hdbsql from the DB server.

Recommendation

During troubleshooting, do not use the same configuration file /etc/krb5.conf as SAP HANA database does. We recommend to create the client configuration in /tmp and to use the environment variable $KRB5_CONFIG when calling hdbsql.

As a <SID>adm, run

#> cd $DIR_EXECUTABLE
#> KRB5_CONFIG=/tmp/krb5.conf hdbsql -i 01

Welcome to the SAP HANA Database interactive terminal.

Type: \h for help with commands

\q to quit

hdbsql=> \c
Connected to TDO@myhdbserver:30115
hdbsql TDO=>

(01 is the instance number and TDO is the SID in this example.)

SQLDBC Trace

The SQLDBC trace will be needed when connections are made from hdbsql or other ODBC clients.

For switching on the SQLDBC trace (no matter on which platform except for the filename path), run

> hdbsqldbc_cons trace packet on
> hdbsqldbc_cons trace sql on
> hdbsqldbc_cons config trace filename c:\temp\sqldbc.trc
CAUTION

Remember to disable tracing after you are done with the analysis.

JDBC Trace

The JDBC trace will be needed when connections are made from SAP HANA studio or other JDBC clients.

CAUTION

The trace file size may grow to gigabytes within relative short time, depending on the system load. Note that it will become difficult to open it with many editors. The tool less on Linux (or less coming with cygwin on Windows) will always be able to open the file, though.

Prerequisite: You are logged on as the operating system user who started (or will start) the JDBC application.

Note

- You always activate the JDBC trace for all JDBC applications that the current operating system user has started.
- Configuration changes have an effect on all JDBC applications that the current operating system user has started.
- In case the application is running on Windows, you might run into the problem that the application is running under a service user which is not allowed to log on. In this case, you have to either let the application run under another user or modify the service user such that a logon is possible.

You have to find the JDBC driver file ngdbc.jar. It should be within the installation directory of the client connecting against SAP HANA database or the SAP HANA client installation directory. You may want to simply use the operating system search for finding it.

Enable tracing via GUI: Just double-click on the ngdbc.jar file. Then

1. Setup trace file location (e.g. create directory C:\temp\krb)
2. Set tracing to “active”
3. Hit “Apply” button

Figure: Location of JDBC driver ngdbc.jar in client installation directory.
Enable tracing via command line:
1. Set up trace file location:
   
   ```
   java -jar ngdbc.jar TRACING FILENAME c:\temp\krb
   ```
2. Start tracing
   
   ```
   java -jar \ngdbc.jar TRACING ON
   ```

Disable Tracing via command line:
```
java -jar ngdbc.jar TRACE OFF
```

Show current trace settings via command line:
```
java -jar ngdbc.jar SHOW
```

**ODBC Trace**

The ODBC trace will be needed when connections are made from ODBC clients.

You have to find the ODBC trace console executable `hdbodbc_cons.exe`. It should be within the installation directory of the client connecting against SAP HANA database or the SAP HANA client installation directory. You may want to simply use the operating system search for finding it.

Enable tracing via command line:
1. Set up trace file location:
   
   ```
   hdbodbc_cons CONFIG TRACING FILENAME c:\temp\krb\odbc.trc
   ```
2. Start API tracing
   
   ```
   hdbodbc_cons TRACE API ON
   ```
3. Start packet tracing
   
   ```
   hdbodbc_cons TRACE PACKET ON
   ```

Disable Tracing via command line:
```
hdbodbc_cons TRACE OFF
```
ODBC Troubleshooting

Run driver directly (64 Bit)
Windows Control Panel -> Administrative Tools -> Data Sources (ODBC)
In tab Drivers, check that the HANA ODBC driver is installed (client installation). Then under User DSN, click Add and then Connect.

Run driver directly (32 Bit)
In C:\Windows\SysWOW64\ run odbcad32.exe. Then under User DSN, click Add and then Connect.

Use driver from separate tool
In client installation directory, run
odbcreg32.exe -t <driver name>
, where <driver name> is the name displayed in above mentioned tab Drivers.
Appendix C – Tracing SPNEGO

For tracing SPNEGO, you need to configure the SAP HANA DB server and the SAP Web Dispatcher.

**SAP HANA DB Server: XS Trace**

Authentication and SAP HANA XS trace (via SQL):

```
ALTER SYSTEM ALTER CONFIGURATION ('$xsengine.ini', 'SYSTEM') SET ('trace', 'authentication') = 'debug' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('$xsengine.ini', 'SYSTEM') SET ('trace', 'xsauthentication') = 'debug' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('$xsengine.ini', 'SYSTEM') SET ('trace', 'xssession') = 'debug' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('$xsengine.ini', 'SYSTEM') SET ('trace', 'xsrequesthandler') = 'debug' WITH RECONFIGURE;
```

**SAP Web Dispatcher Trace**

The SAP Web Dispatcher is the HTTP server of SAP HANA XS.

You have to add the following lines to the file `$DIR_INSTANCE/<hostname>/wdisp/sapwebdisp.pfl` after the definition of the `DIR_INSTANCE` variable:

```
DIR_INSTANCE = ${(_LOCAL_HOST_NAME)}

# Trace
rdisp/TRACE = 2
# Access log
icm/HTTP/logging_0 = PREFIX=/, LOGFILE=${DIR_INSTANCE}/trace/access_log-%d-%m-%y, LOGFORMAT=%b %h %H %S %a %l %u %t %T Line=%r %f %U %s %{user-agent}
```

To activate tracing, the SAP Web Dispatcher must be restarted:

For getting the web dispatcher PID, run as `<SID>adm`:

```
#> HDB info | grep sapwebdisp_hdb | grep -v grep
<SID>adm <PID> 3973 2.8 205288 75628 _ sapwebdisp_hdb [...]
```

Then let the daemon restart it:

```
#> kill -2 <PID>
```

You can find the resulting trace files in the directory `$DIR_INSTANCE/<hostname>/trace`:

1. `dev_webdisp`
2. `access_log-<day>-<month>-<year>`
Appendix D – Windows Tools

In case only a fraction of the users have logon problems, or if logon problems seem to be sporadically distributed over all users, the reason might be UDP fragmentation. To diagnose, users could run in Windows XP or earlier:

> netdiag /test:kerberos

In newer Windows versions, this tool was merged into the general OS troubleshooting framework (Control panel …).

In either case, you should force Kerberos to run over TCP following the instructions in MS KB 244474 (http://support.microsoft.com/kb/244474/en-us).

For various analysis tasks it might be useful to switch on the Windows System Event Log for Kerberos on the AD domain controller and the client (see http://support.microsoft.com/kb/262177/en-us). Note that it is not recommended to run a production system with this log switched on.

Especially on the client you can observe if the requested service ticket is addressed for the right SPN.

For inspecting the ticket cache or forcing the client to retrieve a new ticket, there is a tool klist available from Microsoft.

Here is another troubleshooting guide from Microsoft:
www.sdn.sap.com/irj/sdn/howtoguides