Understanding Kerberos Double Hop

learn.microsoft.com/en-us/archive/blogs/askds/understanding-kerberos-double-hop

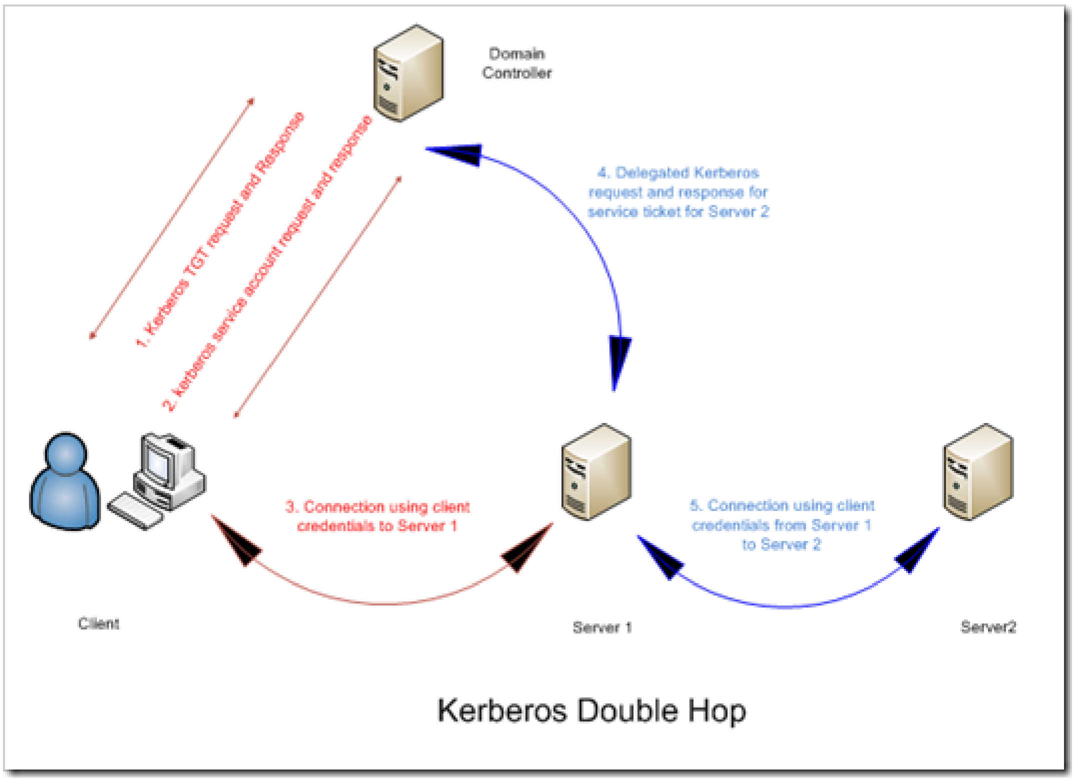
- Article
- 06/13/2008

Hi, Steve here. Kerberos Double Hop is a term used to describe our method of maintaining the client's Kerberos authentication credentials over two or more connections. In this fashion we can retain the user's credentials and act on behalf of the user in further connections to other servers.

Please make sure you read the previous Kerberos for the busy admin post as I will reference terms used in that blog frequently.

The Kerberos TGT is the user's identity. When we pass this ticket along with the service ticket we can re-use the KrbTGT to request other service tickets to speak with our service resources on our network.

There are requirements for a service to be able to perform Kerberos double hop. The service account needs to be trusted for delegation. Meaning it must be trusted to act upon another user's behalf. Source and target servers must be in the same forest or there must be a forest level trust between forests and the first level service account must be in the trusted forest root.


Domain Controller
1. Kerberos TGT request and Response
2. kerberos service account request and response
4. Delegated Kerberos request and response for service ticket for Server 2
3. Connection using client credentials to Server 1
5. Connection using client credentials from Server 1 to Server 2
Client
Server 1
Server2
Kerberos Double Hop

How it Works:

> *Step 1* - Client provides credentials and domain controller returns a Kerberos TGT to the client.
>
> *Step 2* - Client uses TGT to request a service ticket to connect to Server 1.
>
> *Step 3* - Client connects to Server 1 and provides both TGT and service ticket.
>
> *Step 4* - Server 1 uses the clients TGT to request a service ticket so Server 1 can connect to Server 2 .
>
> *Step 5* - Server 1 connects to Server 2 using the client's credentials.

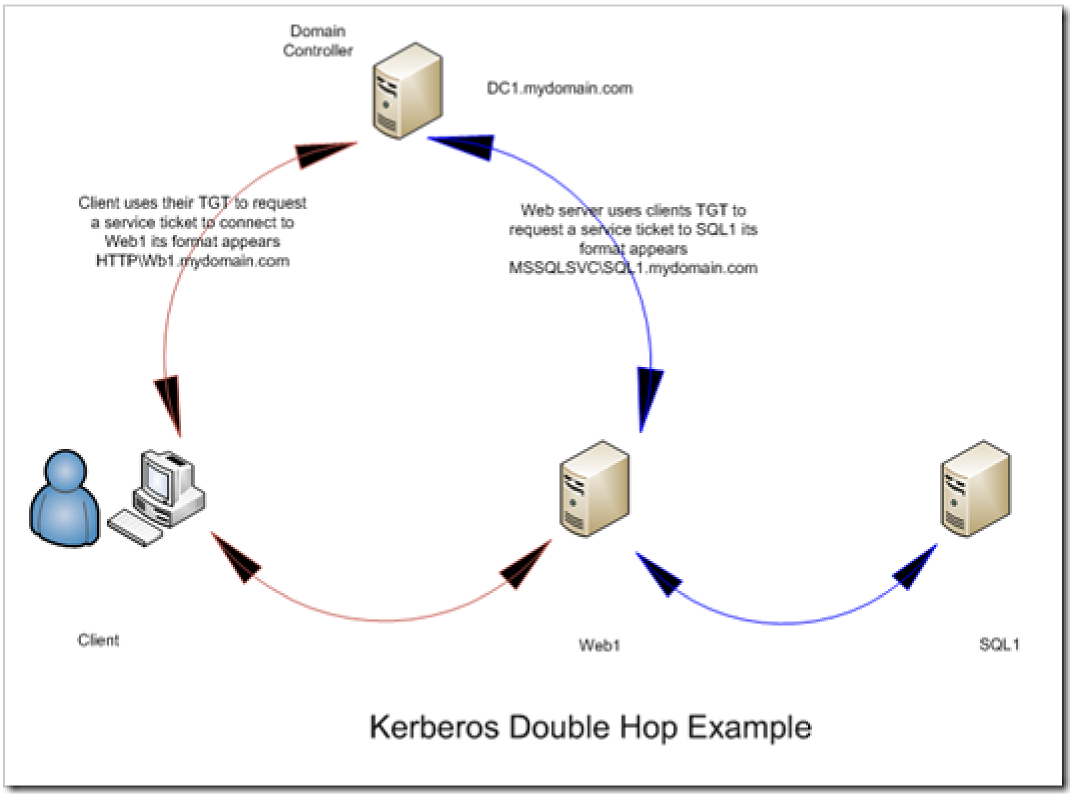
Specific Example:

Client is running IE7 and connecting to a web server that is using windows authentication. The client machine needs to be a member of the forest or a trusted forest and IE needs to be enabled for integrated windows authentication.

Web server machine name WEB1.mydomain.com and is using a service account, mydomain\webadmin. The webadmin account has SPN registered for both HTTP/WEB1 and HTTP/WEB1.mydomain.com. The webadmin account is enabled for constrained delegation to MSSQLSVC/SQL1.mydomain.com.

The SQL server machine name is SQL1.mydomain.com and is service account for SQL is mydomain\sqladmin. The sqladmin account has SPN's registered for MSSQLSVC/SQL1.mydomain.com.

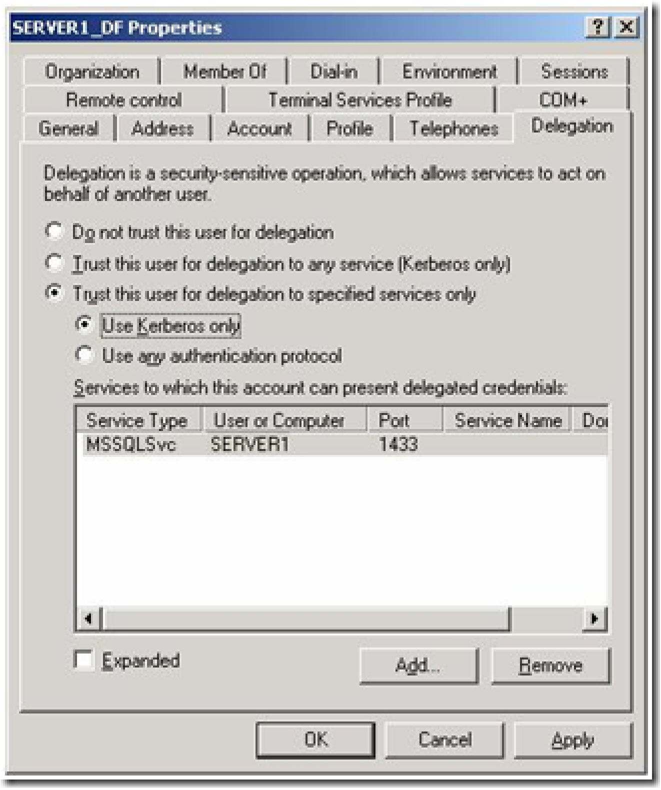
In the example configuration above the client is connecting to *https://web1* to get access to data that is stored on a backend SQL server named SQL1. The web page hosts the code that retrieves the data from SQL. The user account is used to authenticate to the web server. The web server uses its constrained delegation ability to request a Kerberos ticket on the user's behalf for connection to SQL1. If we were to audit the connections we would see the users account is being used to access the web page and the data on the SQL server. This is a classic example of Kerberos double hop but we could easily expand the scenario to include more hops. We could theoretically keep expanding the example as long as we enable delegation and retain the correct service principal name registrations.


Domain
Controller
DC1.mydomain.com
Client uses their TGT to request
a service ticket to connect to
Web1 its format appears
HTTP\Wb1.mydomain.com
Web server uses clients TGT to
request a service ticket to SQL1 its
format appears
MSSQLSVC\SQL1.mydomain.com
Client
Web1
SQL1
Kerberos Double Hop Example

Constrained vs. General Delegation:

General delegation will allow the first hop server to request Kerberos tickets on the client behalf to any other resource in the forest.

Constrained delegation is not supported by all Kerberos aware applications. The domain functional level must be 2003. It allows the administrator to selectively allow an account to request Kerberos tickets limited to specific services on specific servers. This is a much more secure method of delegating Kerberos delegation. The service accounts and the computer accounts hosting the applications need to be in the same domain. If the service account is a user account the delegation tab maybe missing. Until the account has a service principal name registered for it there will not be a delegation tab and you will not be able to setup constrained delegation.



Delegation is a security-sensitive operation, which allows services to act on behalf of another user.

○ Do not trust this user for delegation
○ Trust this user for delegation to any service (Kerberos only)
● Trust this user for delegation to specified services only
● Use Kerberos only
○ Use any authentication protocol

Services to which this account can present delegated credentials:

Service Type	User or Computer	Port	Service Name	Do
MSSQLSvc	SERVER1	1433		

☐ Expanded

Protocol Transition:

So far we have assumed the client is using Kerberos. Common scenarios where Kerberos is not used are when the client does not support Kerberos. In these examples the initial authentication to Server 1 can be transitioned into a Kerberos request in order to maintain the client's credentials when connecting to Server 2.

Samples of method of protocol transition - https://technet2.microsoft.com/windowsserver/en/library/c312ba01-318f-46ca-990e-a597f3c294eb1033.mspx?mfr=true

Troubleshooting and Common Problems:

Setspn.exe will help confirm the service accounts have the proper service principal name registered correctly.

At each stage one of the members requests and receives a Kerberos ticket. These tickets are cached on the client and the front end servers. We can use Klist.exe or Kerbtray.exe to examine our cache. Frequently when there are configuration problems the client will be prompted for credentials and this may mean NTLM is being used instead of Kerberos. NTLM credentials cannot be delegated off the system so authentication to the backup server will be in the form of anonymous authentication.

We can increase Kerberos event logging (KB262177) When kerberos authentication is failing and we have increased the logging level we should see indicators in the system event log for kerberos errors.

A packet capture utility may also be useful in recording the Kerberos requests and responses. Make sure to clear the client cache before enable a utility like Network Monitor. You could filter on Kerberos, use care that Kerberos requests may use the default UDP port 88 or fail over to TCP port 88.

Guides:

- Kerberos on IIS, https://support.microsoft.com/kb/326985 , is a good resource that goes discusses using IIS for the front end server.
- Kerberos on 2000 server clusters, https://support.microsoft.com/kb/235529
- Kerberos in SQL Server, https://support.microsoft.com/kb/319723
- Kerberos with network load balancing, https://support.microsoft.com/kb/325608
- Kerberos with SMS 2003, https://support.microsoft.com/kb/326985

References :

- Kerberos RFC - https://www.ietf.org/rfc/rfc1510.txt
- Microsoft Kerberos Tech Ref - https://technet2.microsoft.com/windowsserver/en/library/b748fb3f-dbf0-4b01-9b22-be14a8b4ae101033.mspx?mfr=true
- Kerberos Double Hop webcast - https://support.microsoft.com/kb/887682
- Constrained Delegation - https://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/constdel.mspx
- Protocol Transition - https://technet2.microsoft.com/windowsserver/en/library/4c8b5ac7-368b-45b9-91d7-1ae7c5e0da311033.mspx?mfr=true

- Steve Taylor

Comments

- anonymouscommenter June 16, 2008
  PingBack from http://www.ditii.com/2008/06/16/kerberos-double-hop/

- barkills June 16, 2008
  "The service accounts and the computer accounts hosting the applications need to be in the same domain."

  Can you explain why this statement is the case for the three scenarios where Kerberos is permitted across domain boundaries? Those three scenarios are intraforest, interforest with a forest trust, and cross-realm with a Kerberos cross-realm trust.

  Thanks,

  Brian Arkills

- Stevta June 16, 2008
  If your middle tier is using open delegation you can cross realms. Wheneven possible you should use constained delegation and that limits the service principal name you can choose to the middle tier's kerberos realm.

  Another issue you can run into is with PAC verification can only follow direct trust and not through inherited trusts via a cross forest trust.

  From a security standpoint you theoretically want to limit delegation so you do not find yourself passing credentials to other domains or forests.

  So even though you maybe able cross security realms with authentication in a scenario where you are delegating authentication you should limit your exposure as best practice.

  Steve

- anonymouscommenter June 18, 2008
  To DEP or not to DEP – A good post on DEP from the Performance Team Windows XP era draws to a close –

- anonymouscommenter June 16, 2009
  For the latest version of this document, please refer to the website:
  https://mbs.microsoft.com/customersource

- sinman July 21, 2009
  "Source and target servers must be in the same forest or there must be a forest level trust between forests and the first level service account must be in the trusted forest root."

  Does this mean the trust has to be a mutual trust between forests?

  For example, (using your diagrams), if we have "server1.dc1.mydomain.com" and "server2.dc2.mydomain.com", and the dc2 forest trusts the dc1 forest, but the dc1 forest does not trust the dc2 forest, can we still set up KCD?

- greener85 July 23, 2009
  Hey Sinman,

  it is really not easy to answer the question on will a one-way forest trust work vs do I have to have a two way trust.

  First since you are saying KCD - (Kerberos constrained delegation) all accounts used in the delegation MUST reside in the same domain.

  This means any account impersonating the user account all must exist in the same domain.

  If you have a web application talking to a SQL Server.

  As long as the Web Application Pool, and the SQL Server Service account exist in the same domain it would work.

  However, if you had some kind of DCOM application it might not work if the server accounts were in separate domains. This is because typically at some point you are going to present the users kerberos ticket to the machine account (Which needs to be trusted for delegation) to make the RPC / DCOM call over to the remote system and thats when a solution like this would fail when the accounts do not exist in the same domain.

  This is why all MS Technet documentation states computer and service accounts must exist in the same domain. Because there are too many caveates to really give anyone a good answer. This is the same reason why almost all documentation states a two way forest trust is required.

  The best guidance I can really give you is to try your configuration with a one way trust and see how it works for you... Nothing trumps testing a solution. But I will tell you that if you do want to attempt a one-way trust setup, make sure that the delegation accounts all exist in the users forest!

- anonymouscommenter December 24, 2013
  Pingback from How I passed SharePoint 2010 exam 70-667 (Part 2 of 4) | SHAREPOINT BUILDING BLOCKS

- anonymouscommenter May 9, 2014
  Pingback from Register SPN | SQL Notes

- anonymouscommenter May 9, 2014
  Pingback from Register SPN | SQL Notes

- anonymouscommenter May 15, 2014
  Pingback from Running SSIS Packages Programmatically, Part 4: Project Deployment Model changes | Dom Writes Code

- anonymouscommenter May 16, 2014
  Pingback from Register SPN | SQL Notes