Kerberos & KRBTGT: Active Directory's Domain Kerberos Service Account

By Sean Metcalf

Published: 2014-11-10 · Archived: 2026-04-06 01:14:03 UTC

Every Domain Controller in an Active Directory domain runs a KDC (Kerberos Distribution Center) service which handles all Kerberos ticket requests. AD uses the KRBTGT account in the AD domain for Kerberos tickets. The KRBTGT account is one that has been lurking in your Active Directory environment since it was first stood up. Each Active Directory domain has an associated KRBTGT account that is used to encrypt and sign all Kerberos tickets for the domain. It is a domain account so that all writable Domain Controllers know the account password in order to decrypt Kerberos tickets for validation. Read Only Domain Controllers (RODCs) each have their own individual KRBTGT account used to encrypt/sign Kerberos tickets in their own sites. The RODC has a specific KRBTGT account (krbtgt_#######) associated with the RODC through a backlink on the account. This ensures that there is cryptographic isolation between trusted Domain Controllers and untrusted RODCs.

The KRBTGT is shrouded in mystery and most AD admins will not mess with it or change its membership. It shouldn't be a member of Domain Admins, Administrators, or any other groups other than "Domain Users" and "Denied RODC Password Replication Group". Note that the "Denied RODC Password Replication Group" is a new group added when you run ADPrep before installing the domain's first 2008/2008R2/2012 DC. This group supports Read-Only Domain Controllers (RODC) ensuring that certain accounts never have their passwords stored on a RODC.

```
PS C:\Users\lukeskywalker> import-module activedirectory
get-aduser krbtgt -property Created,PasswordLastSet,Enabled,SID,DistinguishedName


Created           : 12/7/2014 11:17:39 AM
DistinguishedName : CN=krbtgt,CN=Users,DC=lab,DC=adsecurity,DC=org
Enabled           : False
GivenName         :
Name              : krbtgt
ObjectClass       : user
ObjectGUID        : 6fd9529f-0805-4f3c-bb4d-29ad2ac377ef
PasswordLastSet   : 12/7/2014 11:17:39 AM
SamAccountName    : krbtgt
SID               : S-1-5-21-1473643419-774954089-2222329127-502
Surname           :
UserPrincipalName :
```

The SID for the KRBTGT account is S-1-5-<domain>-502 and lives in the Users OU in the domain by default. Microsoft does not recommend moving this account to another OU.

From Microsoft TechNet:

> The KRBTGT account is a local default account that acts as a service account for the Key Distribution Center (KDC) service. This account cannot be deleted, and the account name cannot be changed. The KRBTGT account cannot be enabled in Active Directory.

> KRBTGT is also the security principal name used by the KDC for a Windows Server domain, as specified by RFC 4120. The KRBTGT account is the entity for the KRBTGT security principal, and it is created automatically when a new domain is created.
>
> Windows Server Kerberos authentication is achieved by the use of a special Kerberos ticket-granting ticket (TGT) enciphered with a symmetric key. This key is derived from the password of the server or service to which access is requested. The TGT password of the KRBTGT account is known only by the Kerberos service. In order to request a session ticket, the TGT must be presented to the KDC. The TGT is issued to the Kerberos client from the KDC.

99.99% of the time*, the KRBTGT account's password has not changed since the Active Directory domain was stood up.

RODCs have the msDS-SecondaryKrbTgtNumber attribute on their computer object populated with the random RODC code with identifies the RODC KRBTGT account (KRBTGT_33171) following the name standard "krbtgt_######" (where # is a number greater than 32737) .
There's also an attribute which is a back-link to the associated RODC called msDS-KrbTgtLinkBl.

The KRBTGT accounts store the Key Version Number (KVNO) in the msDS-KeyVersionNumber attribute on the KRBTGT account.

Theoretically, this tracks the KRBTGT password version and is necessary for the DCs to identify which KRBTGT account was used to encrypt/sign Kerberos tickets. If the KVNO = 5 and the Kerberos (TGT) ticket has a KVNO = 4, then the DC needs to use the previous KRBTGT password to decrypt the Kerberos ticket.
Windows doesn't do that though. It attempts to decrypt with the current password and if that fails, it attempts again with the previous one (assuming it has it). Reference: MSDN To KVNO or To Not KVNO

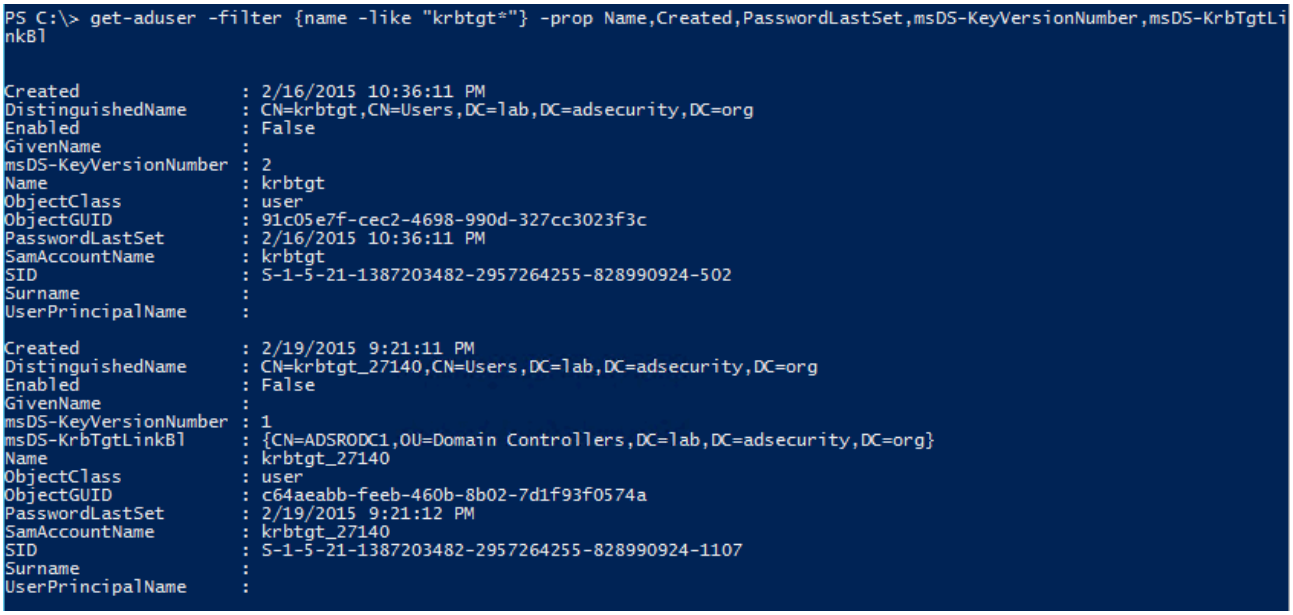
"To distinguish between Kerberos tickets issued by RODC's vs. Kerberos tickets issued by full RWDC's, the low 16 bits of the Property Version Number (PVN) of the 32-bit unicodePWD attribute of the relevant krbtgt account as the traditional Key Version Number (KVNO) and the high 16 bits as a branch ID."
– TechNet Blogs on 2008 & 2003 DC Interop problems

Script code to identify KRBTGT account info (including the key version number – tracks password changes) for every domain in the AD forest:

```
import-module activedirectory
$ADForestRootDomain = (Get-ADForest).RootDomain
$AllADForestDomains = (Get-ADForest).Domains
$ForestKRBTGTInfo = @()
ForEach ($AllADForestDomainsItem in $AllADForestDomains)
{
[string]$DomainDC = (Get-ADDomainController -Discover -Force -Service "PrimaryDC" -
DomainName $AllADForestDomainsItem).HostName
[array]$ForestKRBTGTInfo += Get-ADUSer -filter {name -like "krbtgt*"} -Server $DomainDC -Prop
Name,Created,logonCount,Modified,PasswordLastSet,PasswordExpired,msDS-
```

```
KeyVersionNumber,CanonicalName,msDS-KrbTgtLinkBl
}
$ForestKRBTGTInfo | Select Name,Created,logonCount,PasswordLastSet,PasswordExpired,msDS-KeyVersionNumber,CanonicalName | ft -auto
```

The following graphic shows similar results to the script code above:



PowerShell code to get Active Directory domain KRBTGT account details for the forest: [Get-PSADForestKRBTGTInfo]

Here's the output of this script for a lab environment:

> *Processing 2 service accounts (user accounts) with SPNs discovered in AD Forest DC=ADSecurity,DC=org*
>
> *Domain : lab.ADSecurity.org*
> *UserID : krbtgt*
> *Description : Key Distribution Center Service Account*
> *PasswordLastSet : 11/16/2009 05:59:56*
> *LastLogon : 01/01/1601 00:00:00*
>
> *Domain : RD.ADSecurity.org*
> *UserID : krbtgt*
> *Description : Key Distribution Center Service Account*
> *PasswordLastSet : 08/30/2013 19:23:25*
> *LastLogon : 01/01/1601 00:00:00*

The LastLogon value for the KRBTGT accounts in the above example shows that the accounts haven't logged on. While the account is disabled and technically can't be enabled, it is often one of the first accounts an attacker goes after once a Domain Controller has been compromised.

Why is that?

Here's how Kerberos works (in a nutshell):

1. User logs on with AD user name and password to a domain-joined computer (usually a workstation).
2. The user requests authentication by sending a timestamp (Pre-auth data) encrypted with the users password-based encryption key (password hash).
3. User account (user@adsecurity.org) requests a Kerberos service ticket (TGT) with PREAUTH data (Kerberos AS-REQ).
4. The Kerberos server (KDC) receives the authentication request, validates the data, and replies with a TGT (Kerberos AS-REP).

The most important point of this process is that the Kerberos TGT is encrypted and signed by the KRBTGT account. This means that anyone can create a valid Kerberos TGT if they have the KRBTGT password hash. Furthermore, despite the Active Directory domain policy for Kerberos ticket lifetime, the KDC trusts the TGT, so the custom ticket can include a custom ticket lifetime (even one that exceeds the domain kerberos policy).

The attacker may use the KRBTGT account to persist on the network even if every other account has its password changed.
During an incredibly awesome talk (Video) at the Black Hat 2014 security conference in Las Vegas, NV in early August, Skip Duckwall & Benjamin Delpy spoke about a method (using Mimikatz) to generate your own Kerberos tickets (aka the Golden Ticket). Key to this is that you need the hash for the KRBTGT account which exists in every Active Directory domain. The KRBTGT account is the account used to generate and sign every Kerberos ticket in the domain.

The "Golden Ticket" method enables an attacker to create their own TGT using the KRBTGT account password hash (often extracted from a DC using Mimikatz) with a long lifetime (10 years perhaps) and with any group membership they wish – remember, the TGT is encrypted/signed by the domain's KRBTGT account which is trusted by default by all computers in the domain. And why wouldn't they? That account is central to Kerberos working. Since Kerberos tickets are only validated after 20 minutes (for Kerberos service ticket, TGS), an attacker has more than enough time to access data and/or resources. If not, the attacker can always generate a new "Golden" TGT.

The brilliant part of creating a Golden Ticket (GT) using the domain KRBTGT hash is that the Golden Ticket contains whatever options the creator specifies and the KDC receiving the Golden Ticket generates a TGS assuming that all info in the Golden Ticket is valid. This means that a Golden Ticket can be created for a disabled user outside of normal logon hours.

Common TGT Options:

- User Name
- User Domain
- Ticket Encryption Type
- Logon Hours

- Group Membership (PAC) which contains group SIDS (in a Golden Ticket user SIDs in the PAC are processed)
- Authentication Silo
- (remove) Protected Users

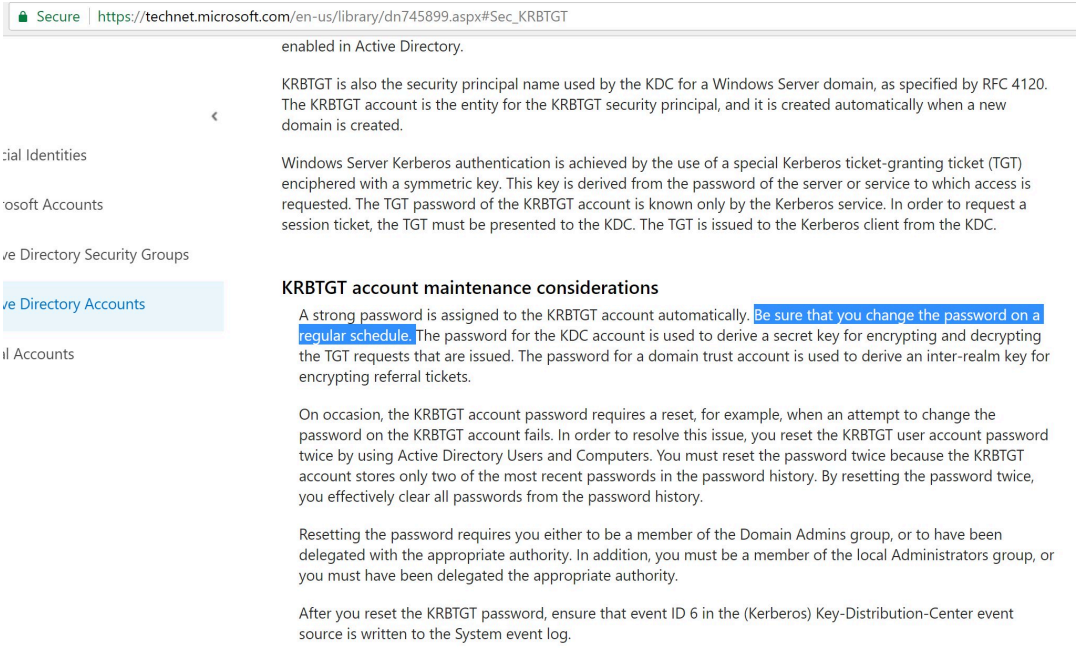
If your Active Directory domain/forest has been compromised and you can't rebuild the entire network from scratch, you will need to reset all passwords in the forest, including the KRBTGT account password(s). Microsoft states that resetting the KRBTGT account password is only supported in a Windows Server 2008 Domain Functional Level (DFL) (or higher). When the DFL is raised from 2003 to 2008 (or higher), the KRBTGT account password is changed automatically.

Changing the KRBTGT Password

Changing the KRBTGT account password can be painful – it has to be changed twice to ensure there is no password history maintained. If your domain/forest has been compromised, you must reset the KRBTGT account password twice. It must be changed twice since the account's password history stores the current password and the last one or 'n-1' (sounds a lot like a trust account password and a computer account password). If this isn't done, it is very likely the attacker can get back on the network at some point and generate custom TGTs (aka Golden Tickets) using the KRBTGT account password hash. The KRBTGT password hash which usually has never been changed (other than when the domain functional level was raised from 2003 to 2008/2008R2/2012/2012R2). Ensure you change the KRBTGT account password for every domain in your forest. Don't leave an attacker any backdoors.

Note: Changing the KRBTGT password is only supported by Microsoft once the domain functional level is Windows Server 2008 or greater. This is likely due to the fact that the KRBTGT password changes as part of the DFL update to 2008 to support Kerberos AES encryption, so it has been tested.

Microsoft now recommends that the KRBTGT password change on a regular basis.



Microsoft posted a KRBTGT account password PowerShell script on TechNet that will change the KRBTGT account password once for a domain, force replication, and monitor change status.

Note that changing the KRBTGT account password in a 2008 (or higher) DFL will not cause replication issues.

KRBTGT Password Change Scenarios:

- Maintenance: Changing the KRBTGT account password once, waiting for replication to complete (and the forest converge), and then changing the password a second time, provides a solid process for ensuring the KRBTGT account is protected and reduces risk (Kerberos and application issues).
- Breach Recovery: Changing the KRBTGT account password twice in rapid succession (before AD replication completes) will invalidate all existing TGTs forcing clients to re-authenticate since the KDC service will be unable to decrypt the existing TGTs. Choosing this path will likely require rebooting application servers (or at least re-starting application services to get them talking Kerberos correctly again).

Microsoft has two TechNet articles which describe scenarios where changing the KRBTGT account password may be necessary:

- Event ID 14 — Kerberos Key Integrity
- Event ID 10 — KDC Password Configuration

When changing the KRBTGT account password make certain you use a solid password.

UPDATE: Note that when you set the KRBTGT password, even if you set it to "KerberosIsMyPal1!" it will be automatically changed to a complex password in the background. This means that the password you enter when changing the password doesn't matter, only that the password changes.

Here's PowerShell code to generate a 128 character, complex password. Note that the DC will change the password to something else.

```
[Reflection.Assembly]::LoadWithPartialName("System.Web")
$RandPassLength = [int] 128
Write-Output "Generating $RandPassLength Character Random Password"
$RandomPassword = [System.Web.Security.Membership]::GeneratePassword($RandPassLength,2)
$RandomPassword
```

In conclusion, the KRBTGT account is critical for AD domain Kerberos authentication and if not properly protected, enables an attacker to create their own Kerberos TGT "Golden Tickets." These special TGTs provide the attacker with access to anything and everything Kerberos enabled on the network without having to add themselves to AD groups.

Note:
There is a potential issue with Exchange when changing the KRBTGT account password: Considering updating your Domain functional level from Windows 2003? Read this!

References:

- Active Directory Accounts: KRBTGT
- http://blogs.msdn.com/b/openspecification/archive/2011/05/11/notes-on-kerberos-kvno-in-windows-rodc-environment.aspx
- http://blogs.technet.com/b/instan/archive/2009/07/30/problems-with-introducing-a-new-windows-server-2008-dc-into-a-windows-2003-forest.aspx
- Mimikatz and Active Directory Kerberos Attacks
- BlackHat USA 2013 Slides: Microsoft's Credential Problem – Skip Duckwall & Chris Campbell
- Abusing Kerberos (aka the Mimikatz Golden Ticket Presentation) BlackHat USA 2014 Presentation Video – Skip Duckwall & Benjamin Delpy
- Mimikatz and Golden Tickets… What's the BFD? BlackHat USA 2014 Redux part 1
- BlueHat 2014 Slides: Reality Bites: The Attacker's View of Windows Authentication and Post-exploitation – Chris Campbell, Benjamin Delpy, & Skip Duckwall
- Christopher Campbell's DEFCON 22 Presentation: The Secret Life of krbtgt (PDF download)
- DerbyCon 2014 Presentation: Et tu Kerberos – Christopher Campbell
- Pass The Golden Ticket Protection from Kerberos Golden Ticket Mitigating pass the ticket on Active Directory
- Why We Don't Get It and Why We Shouldn't
- Let's talk about Pass-the-Hash
- Pass The Golden Ticket Protection from Kerberos – Golden Ticket Mitigating pass the ticket on Active Directory (CERT EU Whitepaper)
- Mitigating Pass-the-Hash (PtH) Attacks and Other Credential Theft, Version 1 and 2 (Microsoft) (PDF document download).
- LSA (LSASS) Protection Option in Windows 8.1 & Windows Server 2012 R2 (technical article)
- Fixing Pass The Hash and Other Problems (Blog post by Scriptjunkie 2013)
- Active Directory Real Defense for Domain Admins – Jason Lang
- Attacking Microsoft Kerberos Kicking the Guard Dog of Hades – Tim Medin
- DerbyCon 2013: The InfoSec Revival – Scriptjunkie
- Kerberos for the Busy Admin
- How the Kerberos Version 5 Authentication Protocol Works
- Encryption Type Selection in Kerberos Exchanges
- Understanding Microsoft Kerberos PAC Validation
- Replication Version Number for your KrbTGT account password?
