Luc Delsalle Follow
Security researcher focus on MS Active Directory Infrastructures · Former redteamer and systems security engineer
Jan 28 · 13 min read

DCShadow explained: A technical deep dive into the latest AD attack technique

On January 24th 2018, Benjamin Delpy and Vincent Le Toux, two security researchers, have released during the "BlueHat IL" security conference a new attack technique against Active Directory infrastructure. Named "DCShadow", this attack allows an attacker having the appropriate rights to create a rogue domain controller able to replicate malicious objects into a running Active Directory infrastructure.

In this article, we will explain the technical foundations the attack relies on and discuss the consequences for the security of a running Active Directory infrastructure. Finally, we will shed a light on how blue teams could detect this kind of attack.

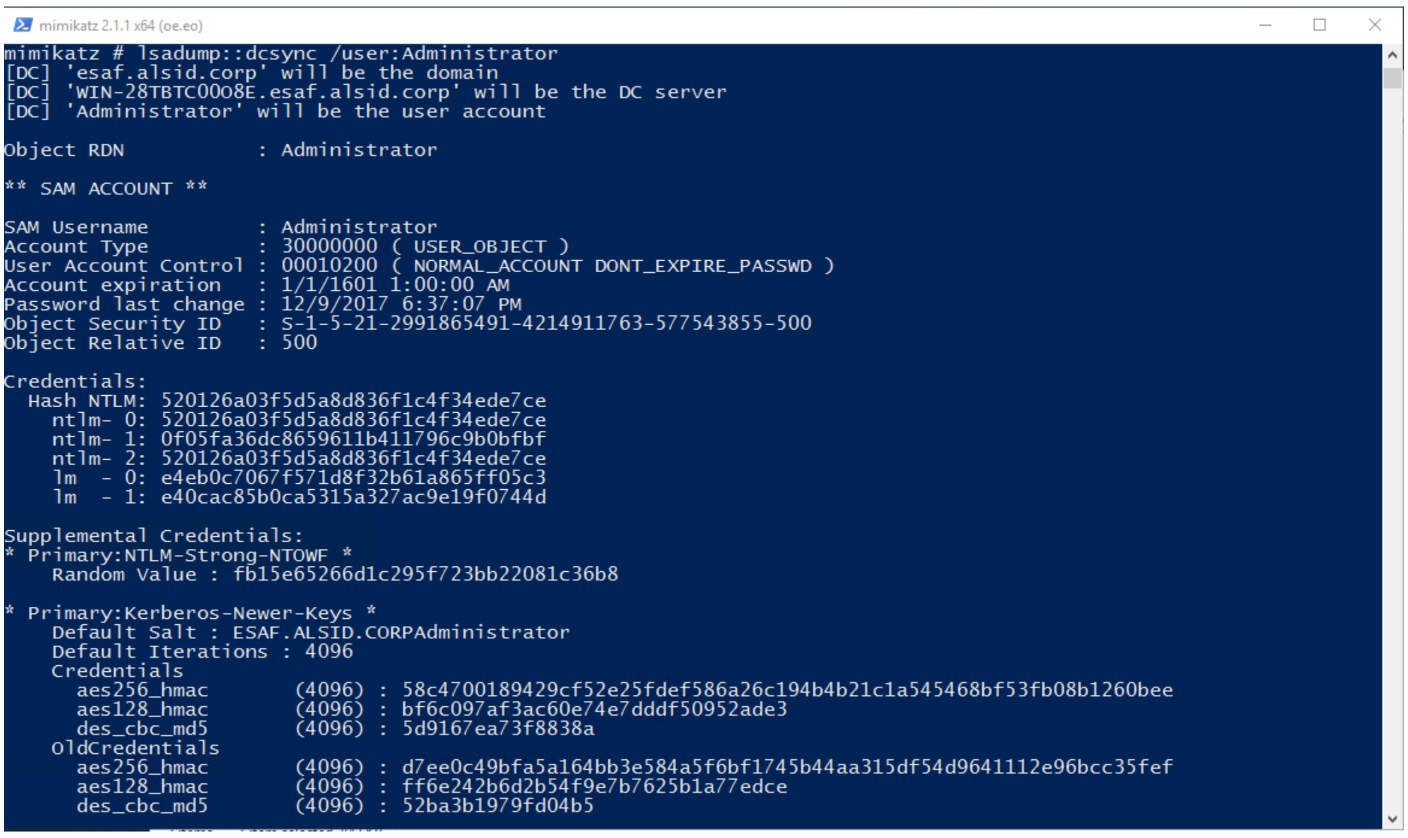
What is the DCShadow attack and why is it new?

The holy grail for red teamers or attackers willing to compromise an Active Directory infrastructure is to be able to obtain users and computers credentials without being noticed by detection countermeasures.

rpose, several attack techniques have been developed through

available on the impressive synthesis from ADSecurity.org.

Among all these noisy attacks, one of them is connected to the "DCShadow" attack. Introduced in 2015, the "DCSync" attack relies on the ability for the members of the Domain Admins or Domain Controllers groups to ask a domain controller (DC) for data replication (to achieve this task, the GetChangeAll right, granted by default to administrative accounts and DCs, was necessary). In fact, as described in the MS-DRSR specification for domain controller replication, these groups can request the Domain Controller to replicate AD objects (including user credentials) through the GetNCChanges RPC. More technical details on the attacks are available on the ADSecurity.org blogpost.

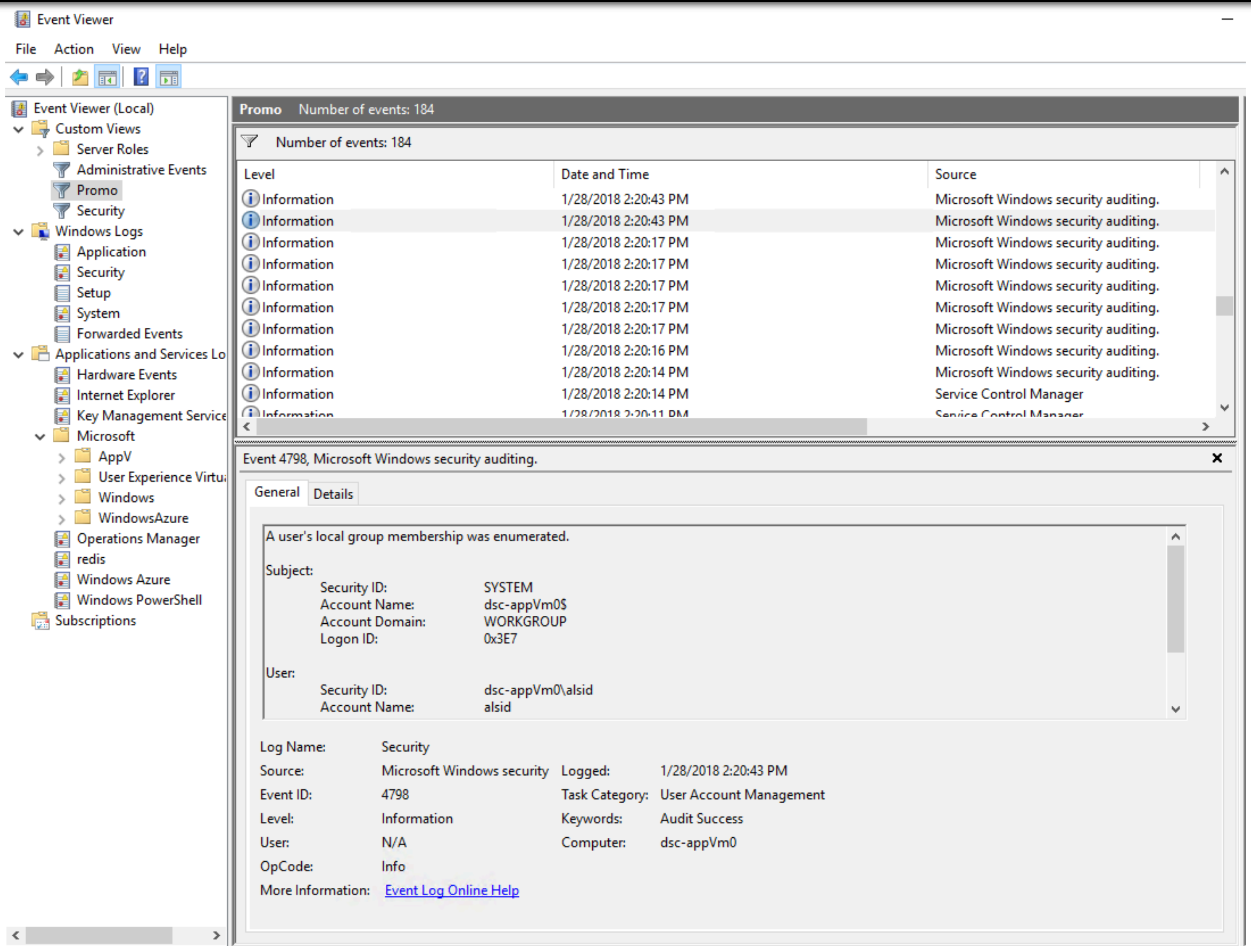


```
mimikatz # lsadump::dcsync /user:Administrator
[DC] 'esaf.alsid.corp' will be the domain
[DC] 'WIN-28TBTC0008E.esaf.alsid.corp' will be the DC server
[DC] 'Administrator' will be the user account

Object RDN           : Administrator

** SAM ACCOUNT **

SAM Username         : Administrator
Account Type         : 30000000 ( USER_OBJECT )
User Account Control : 00010200 ( NORMAL_ACCOUNT DONT_EXPIRE_PASSWD )
Account expiration   : 1/1/1601 1:00:00 AM
Password last change : 12/9/2017 6:37:07 PM
Object Security ID   : S-1-5-21-2991865491-4214911763-577543855-500
Object Relative ID   : 500

Credentials:
  Hash NTLM: 520126a03f5d5a8d836f1c4f34ede7ce
    ntlm- 0: 520126a03f5d5a8d836f1c4f34ede7ce
    ntlm- 1: 0f05fa36dc8659611b411796c9b0bfbf
    ntlm- 2: 520126a03f5d5a8d836f1c4f34ede7ce
    lm  - 0: e4eb0c7067f571d8f32b61a865ff05c3
    lm  - 1: e40cac85b0ca5315a327ac9e19f0744d

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : fb15e65266d1c295f723bb22081c36b8

* Primary:Kerberos-Newer-Keys *
    Default Salt : ESAF.ALSID.CORPAdministrator
    Default Iterations : 4096
    Credentials
      aes256_hmac       (4096) : 58c4700189429cf52e25fdef586a26c194b4b21c1a545468bf53fb08b1260bee
      aes128_hmac       (4096) : bf6c097af3ac60e74e7dddf50952ade3
      des_cbc_md5       (4096) : 5d9167ea73f8838a
    OldCredentials
      aes256_hmac       (4096) : d7ee0c49bfa5a164bb3e584a5f6bf1745b44aa315df54d9641112e96bcc35fef
      aes128_hmac       (4096) : ff6e242b6d2b54f9e7b7625b1a77edce
      des_cbc_md5       (4096) : 52ba3b1979fd04b5
```

DCSync attack with mimikatz tool

One of the main limitation of the "DCSync" attack is the impossibility for an attacker to inject new objects in the targeted AD domain. Of course, this attacker could take ownership of an administrative account using the good old Pass-The-Hash technique and inject objects afterwards, but it requires more efforts, more steps, meaning a greater probability of being busted by blue teams. The "DCShadow" attack removes this limitation by *reversing* the "DCSync" attack paradigm.

With "DCShadow", attackers no longer try to replicate data but will register new domain controllers in the targeted infrastructure to inject Active Directory objects or alter existing ones (by replacing the attributes' content). The idea of a rogue domain controller is not new and has been mentioned gular domain controller to promote the VM into a DC for the



```
Promo   Number of events: 184

Number of events: 184

Level          Date and Time           Source
Information    1/28/2018 2:20:43 PM    Microsoft Windows security auditing.
Information    1/28/2018 2:20:43 PM    Microsoft Windows security auditing.
Information    1/28/2018 2:20:17 PM    Microsoft Windows security auditing.
Information    1/28/2018 2:20:17 PM    Microsoft Windows security auditing.
Information    1/28/2018 2:20:17 PM    Microsoft Windows security auditing.
Information    1/28/2018 2:20:17 PM    Microsoft Windows security auditing.
Information    1/28/2018 2:20:17 PM    Microsoft Windows security auditing.
Information    1/28/2018 2:20:16 PM    Microsoft Windows security auditing.
Information    1/28/2018 2:20:14 PM    Microsoft Windows security auditing.
Information    1/28/2018 2:20:14 PM    Service Control Manager
Information    1/28/2018 2:20:11 PM    Service Control Manager

Event 4798, Microsoft Windows security auditing.

A user's local group membership was enumerated.

Subject:
        Security ID:        SYSTEM
        Account Name:       dsc-appVm0$
        Account Domain:     WORKGROUP
        Logon ID:           0x3E7

User:
        Security ID:        dsc-appVm0\alsid
        Account Name:       alsid

Log Name:          Security
Source:            Microsoft Windows security   Logged:          1/28/2018 2:20:43 PM
Event ID:          4798                         Task Category:   User Account Management
Level:             Information                  Keywords:        Audit Success
User:              N/A                          Computer:        dsc-appVm0
OpCode:            Info
More Information:  Event Log Online Help
```

Eventlog generated during a regular DC promotion

In order to understand the genius ideas behind "DCShadow", it is important to clearly grasp what a domain controller is and how it is registered in the Active Directory infrastructure.

Understanding what a domain controller is

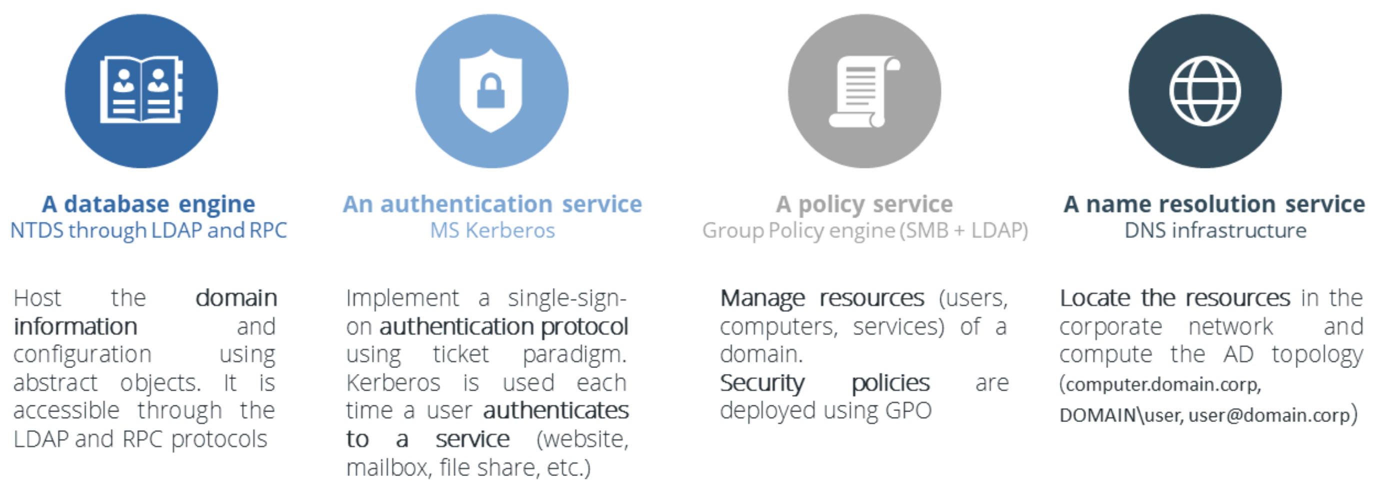
As described in MS-ADTS (Active Directory Technical Specification), Active Directory is a multi-master architecture relying on dedicated services. The DC is the service (or the server hosting this service depending on your point of view) which hosts the data store for AD objects and interoperates with other DCs to ensure that a local change to an object replicates correctly across all DCs.

When a DC is operating as RW DC, the DC contains full naming context (NC) replicas of the configuration, the schema, and one of the domain naming context of its forest. In this way, every RW DC holds all objects of a domain, including credentials and any kind of secrets (like private or session keys). As such, there is no need to remind that DCs are the one and only elements blue teams should be focused on protecting (Administrative accounts or permissions are just two of the many ways to access a DC).

in detail the technical ways and means of a DC could be complex

following 4 key components:

- a database engine able to replicate its information (meaning it must be accessible through LDAP protocols and implement several RPCs to follow MS-DRSR and MS-ADTS specifications)
- an authentication provider accessible through Kerberos, NTLM, Netlogon or WDigest protocols
- a configuration management system called GPO, relying on SMB and LDAP protocols
- an (optional) DNS provider used by clients to locate resources and support authentication



A database engine
NTDS through LDAP and RPC

Host the domain information and configuration using abstract objects. It is accessible through the LDAP and RPC protocols

An authentication service
MS Kerberos

Implement a single-sign-on authentication protocol using ticket paradigm. Kerberos is used each time a user authenticates to a service (website, mailbox, file share, etc.)

A policy service
Group Policy engine (SMB + LDAP)

Manage resources (users, computers, services) of a domain.
Security policies are deployed using GPO

A name resolution service
DNS infrastructure

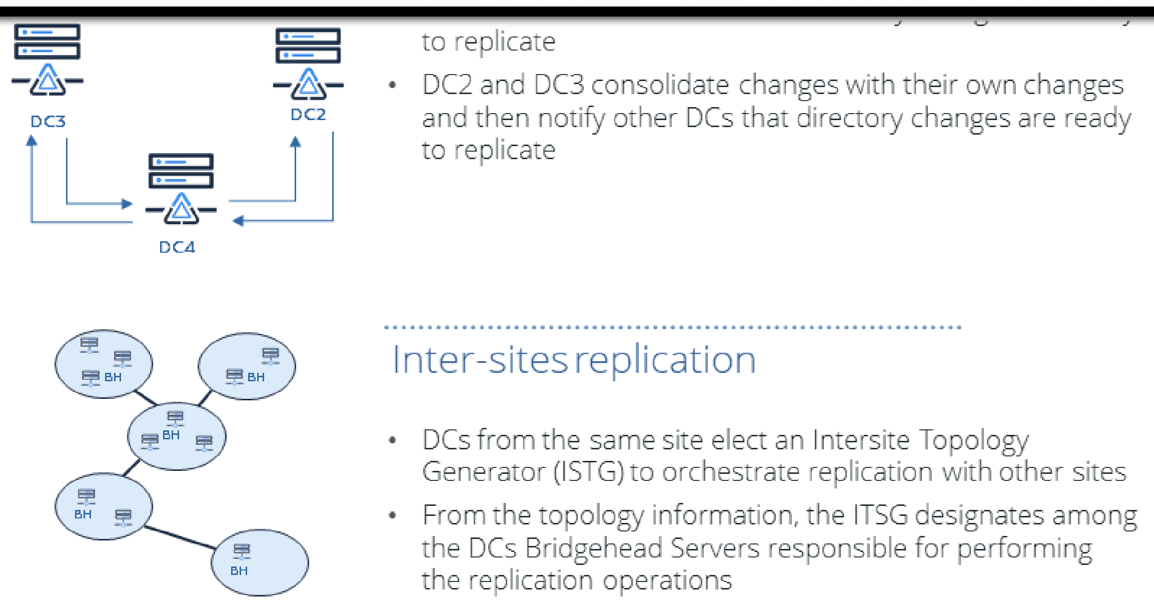
Locate the resources in the corporate network and compute the AD topology (computer.domain.corp, DOMAIN\user, user@domain.corp)

Synthesize of services provided by a DC

Focus on Active Directory replication

In addition to hosting these services, a domain controller in the making should be registered in the directory infrastructure to be accepted by another DC as a replication source provider. The data replication is orchestrated by a built-in process (running on the NTDS service) called the Knowledge Consistency Checker (KCC).

The major function of the KCC is to generate and maintain the replication topology for replication within and between sites. In other words, the KCC process elects which DC will communicate to which other to create an efficient replication process. Within a site, each KCC generates its own connections. For replication between sites, a single KCC per site generates all connections. The following schema illustrates the two kinds of replication.



to replicate

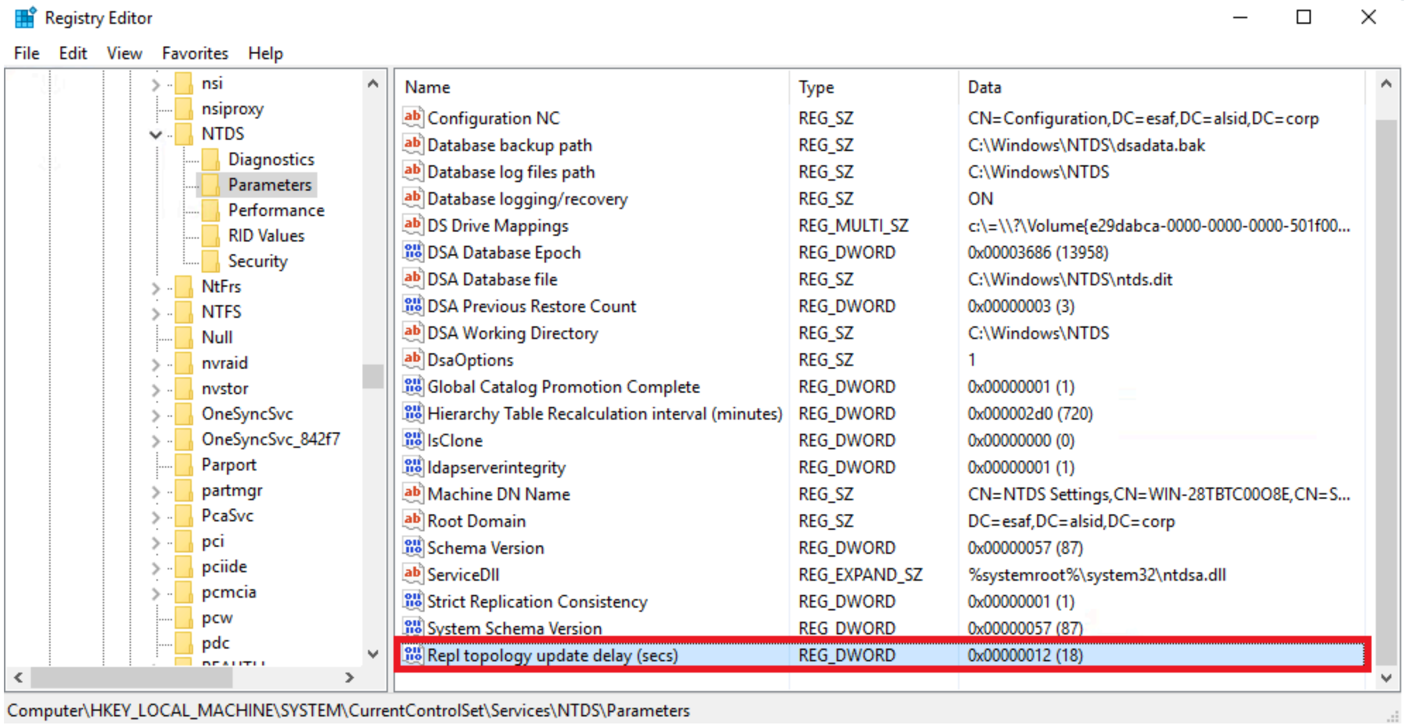
- DC2 and DC3 consolidate changes with their own changes and then notify other DCs that directory changes are ready to replicate

Inter-sites replication

- DCs from the same site elect an Intersite Topology Generator (ISTG) to orchestrate replication with other sites
- From the topology information, the ITSG designates among the DCs Bridgehead Servers responsible for performing the replication operations

The two kinds of replication process

By default, the KCC initiates AD replication topology every 15 minutes to ensure consistent and regular propagation. Using the USN associated to every AD object, the KCC recognizes changes that occur in the environment and ensures that domain controllers are not orphaned in the replication topology. Fun fact, historically Active Directory replication process could have been made through RPC (like DrsAddEntry) but also through SMTP (for the Schema and Configuration partition only)!



Name	Type	Data
Configuration NC	REG_SZ	CN=Configuration,DC=esaf,DC=alsid,DC=corp
Database backup path	REG_SZ	C:\Windows\NTDS\dsadata.bak
Database log files path	REG_SZ	C:\Windows\NTDS
Database logging/recovery	REG_SZ	ON
DS Drive Mappings	REG_MULTI_SZ	c:\=\\?\Volume{e29dabca-0000-0000-0000-501f00...
DSA Database Epoch	REG_DWORD	0x00003686 (13958)
DSA Database file	REG_SZ	C:\Windows\NTDS\ntds.dit
DSA Previous Restore Count	REG_DWORD	0x00000003 (3)
DSA Working Directory	REG_SZ	C:\Windows\NTDS
DsaOptions	REG_SZ	1
Global Catalog Promotion Complete	REG_DWORD	0x00000001 (1)
Hierarchy Table Recalculation interval (minutes)	REG_DWORD	0x000002d0 (720)
IsClone	REG_DWORD	0x00000000 (0)
ldapserverintegrity	REG_DWORD	0x00000001 (1)
Machine DN Name	REG_SZ	CN=NTDS Settings,CN=WIN-28TBTC00O8E,CN=S...
Root Domain	REG_SZ	DC=esaf,DC=alsid,DC=corp
Schema Version	REG_DWORD	0x00000057 (87)
ServiceDll	REG_EXPAND_SZ	%systemroot%\system32\ntdsa.dll
Strict Replication Consistency	REG_DWORD	0x00000001 (1)
System Schema Version	REG_DWORD	0x00000057 (87)
Repl topology update delay (secs)	REG_DWORD	0x00000012 (18)

Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\NTDS\Parameters

Registry key defining the replication time period

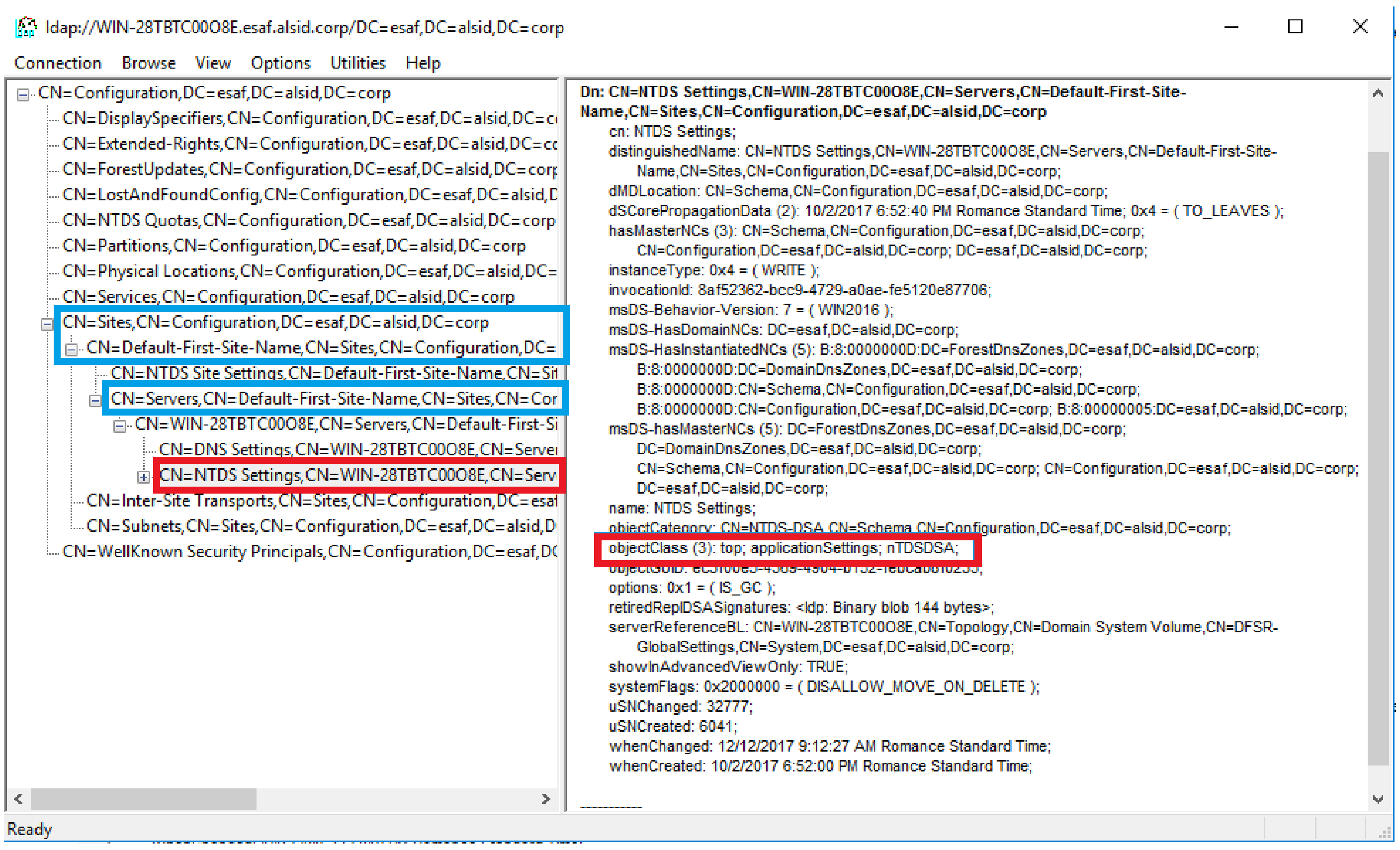
One part of the great job made by the researchers behind “DCShadow” was to identify the minimal set of changes required to inject a new server in the replication topology and therefore inject malicious information abusing this process while remaining stealthy.

d in the following section of this article, the “DCShadow” attack

goal, “DCShadow” attack must modify the targeted AD infrastructure database to authorize the rogue server to be part of the replication process.

Register a new domain controller

As mentioned in the MS-ADTS specification, a domain controller is represented in the AD database by an object of class nTDSDSA that is always located in the configuration naming context of a domain. More precisely, each DC is stored in the sites container (object class sitesContainer), as a child item of a server object.


ldap://WIN-28TBTC00O8E.esaf.alsid.corp/DC=esaf,DC=alsid,DC=corp
Connection
Browse
View
Options
Utilities
Help
Ready

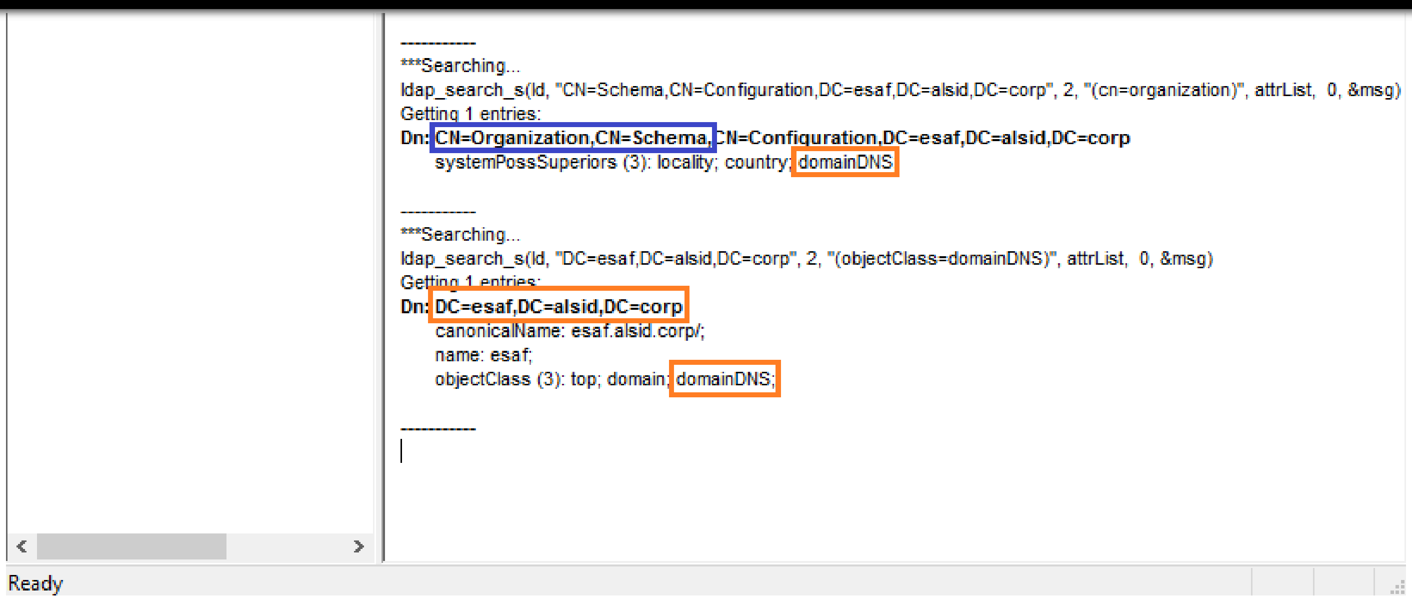
```
CN=Configuration,DC=esaf,DC=alsid,DC=corp
  CN=DisplaySpecifiers,CN=Configuration,DC=esaf,DC=alsid,DC=c
  CN=Extended-Rights,CN=Configuration,DC=esaf,DC=alsid,DC=cc
  CN=ForestUpdates,CN=Configuration,DC=esaf,DC=alsid,DC=corp
  CN=LostAndFoundConfig,CN=Configuration,DC=esaf,DC=alsid,D
  CN=NTDS Quotas,CN=Configuration,DC=esaf,DC=alsid,DC=corp
  CN=Partitions,CN=Configuration,DC=esaf,DC=alsid,DC=corp
  CN=Physical Locations,CN=Configuration,DC=esaf,DC=alsid,DC=
  CN=Services,CN=Configuration,DC=esaf,DC=alsid,DC=corp
  CN=Sites,CN=Configuration,DC=esaf,DC=alsid,DC=corp
    CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=
      CN=NTDS Site Settings,CN=Default-First-Site-Name,CN=Sit
      CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Cor
        CN=WIN-28TBTC00O8E,CN=Servers,CN=Default-First-Si
          CN=DNS Settings,CN=WIN-28TBTC00O8E,CN=Server
          CN=NTDS Settings,CN=WIN-28TBTC00O8E,CN=Serv
    CN=Inter-Site Transports,CN=Sites,CN=Configuration,DC=esaf
    CN=Subnets,CN=Sites,CN=Configuration,DC=esaf,DC=alsid,D
  CN=WellKnown Security Principals,CN=Configuration,DC=esaf,DC
```

```
Dn: CN=NTDS Settings,CN=WIN-28TBTC00O8E,CN=Servers,CN=Default-First-Site-
Name,CN=Sites,CN=Configuration,DC=esaf,DC=alsid,DC=corp
    cn: NTDS Settings;
    distinguishedName: CN=NTDS Settings,CN=WIN-28TBTC00O8E,CN=Servers,CN=Default-First-Site-
        Name,CN=Sites,CN=Configuration,DC=esaf,DC=alsid,DC=corp;
    dMDLocation: CN=Schema,CN=Configuration,DC=esaf,DC=alsid,DC=corp;
    dSCorePropagationData (2): 10/2/2017 6:52:40 PM Romance Standard Time; 0x4 = ( TO_LEAVES );
    hasMasterNCs (3): CN=Schema,CN=Configuration,DC=esaf,DC=alsid,DC=corp;
        CN=Configuration,DC=esaf,DC=alsid,DC=corp; DC=esaf,DC=alsid,DC=corp;
    instanceType: 0x4 = ( WRITE );
    invocationId: 8af52362-bcc9-4729-a0ae-fe5120e87706;
    msDS-Behavior-Version: 7 = ( WIN2016 );
    msDS-HasDomainNCs: DC=esaf,DC=alsid,DC=corp;
    msDS-HasInstantiatedNCs (5): B:8:0000000D:DC=ForestDnsZones,DC=esaf,DC=alsid,DC=corp;
        B:8:0000000D:DC=DomainDnsZones,DC=esaf,DC=alsid,DC=corp;
        B:8:0000000D:CN=Schema,CN=Configuration,DC=esaf,DC=alsid,DC=corp;
        B:8:0000000D:CN=Configuration,DC=esaf,DC=alsid,DC=corp; B:8:00000005:DC=esaf,DC=alsid,DC=corp;
    msDS-hasMasterNCs (5): DC=ForestDnsZones,DC=esaf,DC=alsid,DC=corp;
        DC=DomainDnsZones,DC=esaf,DC=alsid,DC=corp;
        CN=Schema,CN=Configuration,DC=esaf,DC=alsid,DC=corp; CN=Configuration,DC=esaf,DC=alsid,DC=corp;
        DC=esaf,DC=alsid,DC=corp;
    name: NTDS Settings;
    objectCategory: CN=NTDS-DSA,CN=Schema,CN=Configuration,DC=esaf,DC=alsid,DC=corp;
    objectClass (3): top; applicationSettings; nTDSDSA;
    objectGUID: ec5100e5-4369-4904-b132-febcab810235;
    options: 0x1 = ( IS_GC );
    retiredReplDSASignatures: <ldp: Binary blob 144 bytes>;
    serverReferenceBL: CN=WIN-28TBTC00O8E,CN=Topology,CN=Domain System Volume,CN=DFSR-
        GlobalSettings,CN=System,DC=esaf,DC=alsid,DC=corp;
    showInAdvancedViewOnly: TRUE;
    systemFlags: 0x2000000 = ( DISALLOW_MOVE_ON_DELETE );
    uSNChanged: 32777;
    uSNCreated: 6041;
    whenChanged: 12/12/2017 9:12:27 AM Romance Standard Time;
    whenCreated: 10/2/2017 6:52:00 PM Romance Standard Time;

-----------
```

In blue, the containers storing the NTDS-DSA object. In red, the object itself.

A quick look at the schema shows that NTDS-DSA objects can only be created as children of server objects, which in turn can only be part of organization or server objects:

- the server objects can only be stored in serversContainer objects which are only found in the Configuration NC.
- the organization objects can only be stored in locality, country or domainDNS objects which can be found in the domain NC



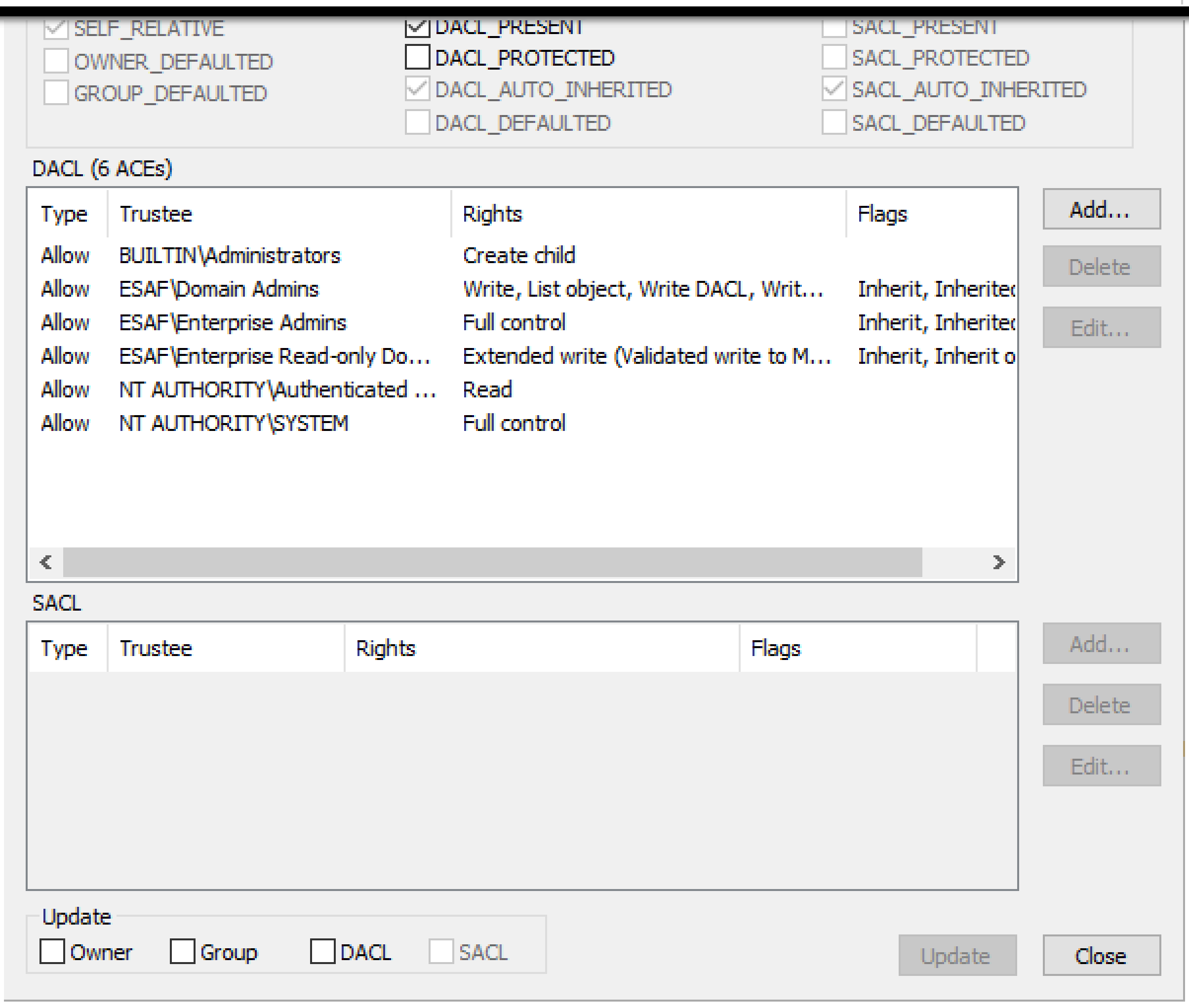
```
-----------
***Searching...
ldap_search_s(ld, "CN=Schema,CN=Configuration,DC=esaf,DC=alsid,DC=corp", 2, "(cn=organization)", attrList,  0, &msg)
Getting 1 entries:
Dn: CN=Organization,CN=Schema,CN=Configuration,DC=esaf,DC=alsid,DC=corp
      systemPossSuperiors (3): locality; country; domainDNS

-----------
***Searching...
ldap_search_s(ld, "DC=esaf,DC=alsid,DC=corp", 2, "(objectClass=domainDNS)", attrList,  0, &msg)
Getting 1 entries:
Dn: DC=esaf,DC=alsid,DC=corp
      canonicalName: esaf.alsid.corp/;
      name: esaf;
      objectClass (3): top; domain; domainDNS;

-----------
```

The schema indicates where ntds-dsa objects can be created

In this way, domain controllers (nTDSDSA objects) can only be created in the Configuration or Domain NC. In practice, it seems only the nTDSDSA objects stored in the site container (sitesContainer object) are taken into consideration. As the KCC relies on the site information to compute its replication topology, it seems logical that only these objects are used. Note that creating an nTDSDSA object is not possible using the LDAP protocol.

You would have understood it, the main action made by the "DCShadow" attack is to create a new server and nTDSDSA objects in the Configuration partition of the schema. Doing so provides the ability to generate malicious replication data and inject them to other domain controllers.

Now that we understand what the "DCShadow" attack do, we need to understand what kind of privileges are required to create nTDSDSA objects in the Configuration partition. A quick look in the permissions show that only BUILTIN\Administrators, DOMAIN\Domain Admins, DOMAIN\Enterprise Admins and NT AUTHORITY\SYSTEM have control rights on the targeted containers.



☑ SELF_RELATIVE ☑ DACL_PRESENT □ SACL_PRESENT
□ OWNER_DEFAULTED □ DACL_PROTECTED □ SACL_PROTECTED
□ GROUP_DEFAULTED ☑ DACL_AUTO_INHERITED ☑ SACL_AUTO_INHERITED
□ DACL_DEFAULTED □ SACL_DEFAULTED

DACL (6 ACEs)

Type	Trustee	Rights	Flags
Allow	BUILTIN\Administrators	Create child	
Allow	ESAF\Domain Admins	Write, List object, Write DACL, Writ...	Inherit, Inherited
Allow	ESAF\Enterprise Admins	Full control	Inherit, Inherited
Allow	ESAF\Enterprise Read-only Do...	Extended write (Validated write to M...	Inherit, Inherit o
Allow	NT AUTHORITY\Authenticated ...	Read	
Allow	NT AUTHORITY\SYSTEM	Full control	

SACL

Type	Trustee	Rights	Flags

Update
□ Owner □ Group □ DACL □ SACL

The default access rights on the Server object.

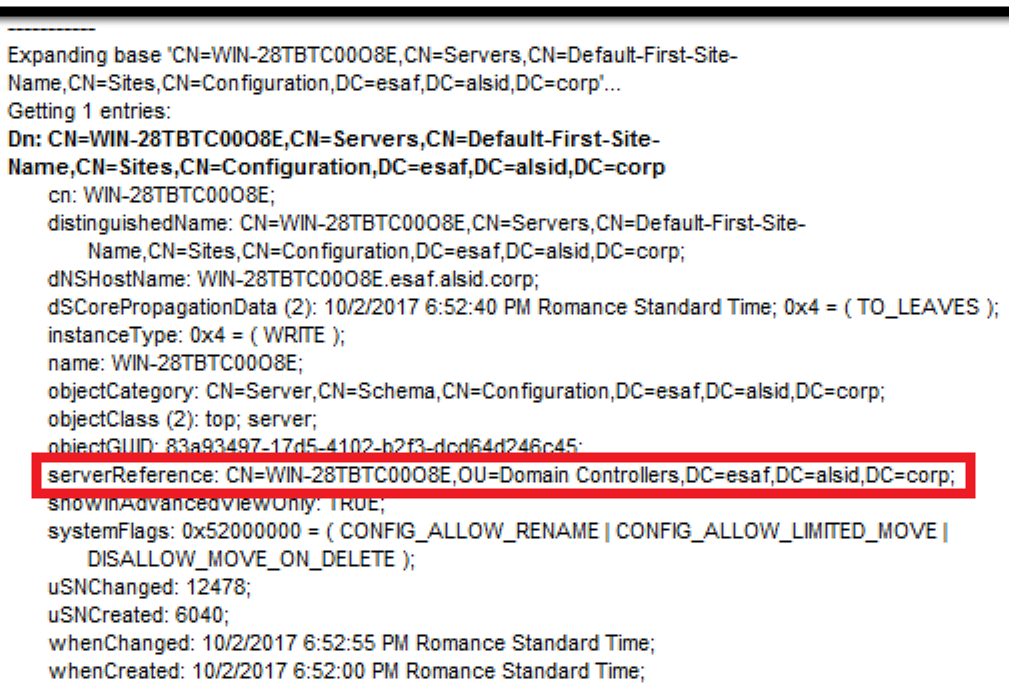
This quick analysis allows us to conclude that the "DCShadow" attack is not a privileges escalation vulnerability, but a misappropriation of Active Directory mechanism. It doesn't allow red teamers to gain privileges but give them another solution to become persistent or to make illegitimate actions in a directory infrastructure. It should thus be added in the category of another sneaky AD persistence trick and not as a vulnerability to fix.

Trust the new domain controller

As described in the previous paragraph, the "DCShadow" attack relies on the addition of a new nTDSDSA object in the Configuration partition to register itself as a new member of the replication process. However, adding this sole object is not enough to allow our rogue server to initiate replication. In fact, to be part of the replication process we need to take care of two requirements:

- be trusted by other servers, meaning that we need to have valid authentication credential.
- provide authentication support to let other DCs to connect to our rogue server when we need to replicate data.

By using a valid computer account, a rogue server can be treated as a



```
------------
Expanding base 'CN=WIN-28TBTC00O8E,CN=Servers,CN=Default-First-Site-
Name,CN=Sites,CN=Configuration,DC=esaf,DC=alsid,DC=corp'...
Getting 1 entries:
Dn: CN=WIN-28TBTC00O8E,CN=Servers,CN=Default-First-Site-
Name,CN=Sites,CN=Configuration,DC=esaf,DC=alsid,DC=corp
    cn: WIN-28TBTC00O8E;
    distinguishedName: CN=WIN-28TBTC00O8E,CN=Servers,CN=Default-First-Site-
        Name,CN=Sites,CN=Configuration,DC=esaf,DC=alsid,DC=corp;
    dNSHostName: WIN-28TBTC00O8E.esaf.alsid.corp;
    dSCorePropagationData (2): 10/2/2017 6:52:40 PM Romance Standard Time; 0x4 = ( TO_LEAVES );
    instanceType: 0x4 = ( WRITE );
    name: WIN-28TBTC00O8E;
    objectCategory: CN=Server,CN=Schema,CN=Configuration,DC=esaf,DC=alsid,DC=corp;
    objectClass (2): top; server;
    objectGUID: 83a93497-17d5-4102-b2f3-dcd64d246c45;
    serverReference: CN=WIN-28TBTC00O8E,OU=Domain Controllers,DC=esaf,DC=alsid,DC=corp;
    showInAdvancedViewOnly: TRUE;
    systemFlags: 0x52000000 = ( CONFIG_ALLOW_RENAME | CONFIG_ALLOW_LIMITED_MOVE |
        DISALLOW_MOVE_ON_DELETE );
    uSNChanged: 12478;
    uSNCreated: 6040;
    whenChanged: 10/2/2017 6:52:55 PM Romance Standard Time;
    whenCreated: 10/2/2017 6:52:00 PM Romance Standard Time;
```

The serverReference attribute acts as the link between a nTDSDSA object and its related computer object

Despite the theoretical possibility to achieve this with a user account, it seems much easier and stealthy to use a computer account. In fact, it will be automatically registered in the DNS infrastructure (which will allow other DCs to locate our resource), will natively have the required attributes set and will have its authentication secret automatically managed.

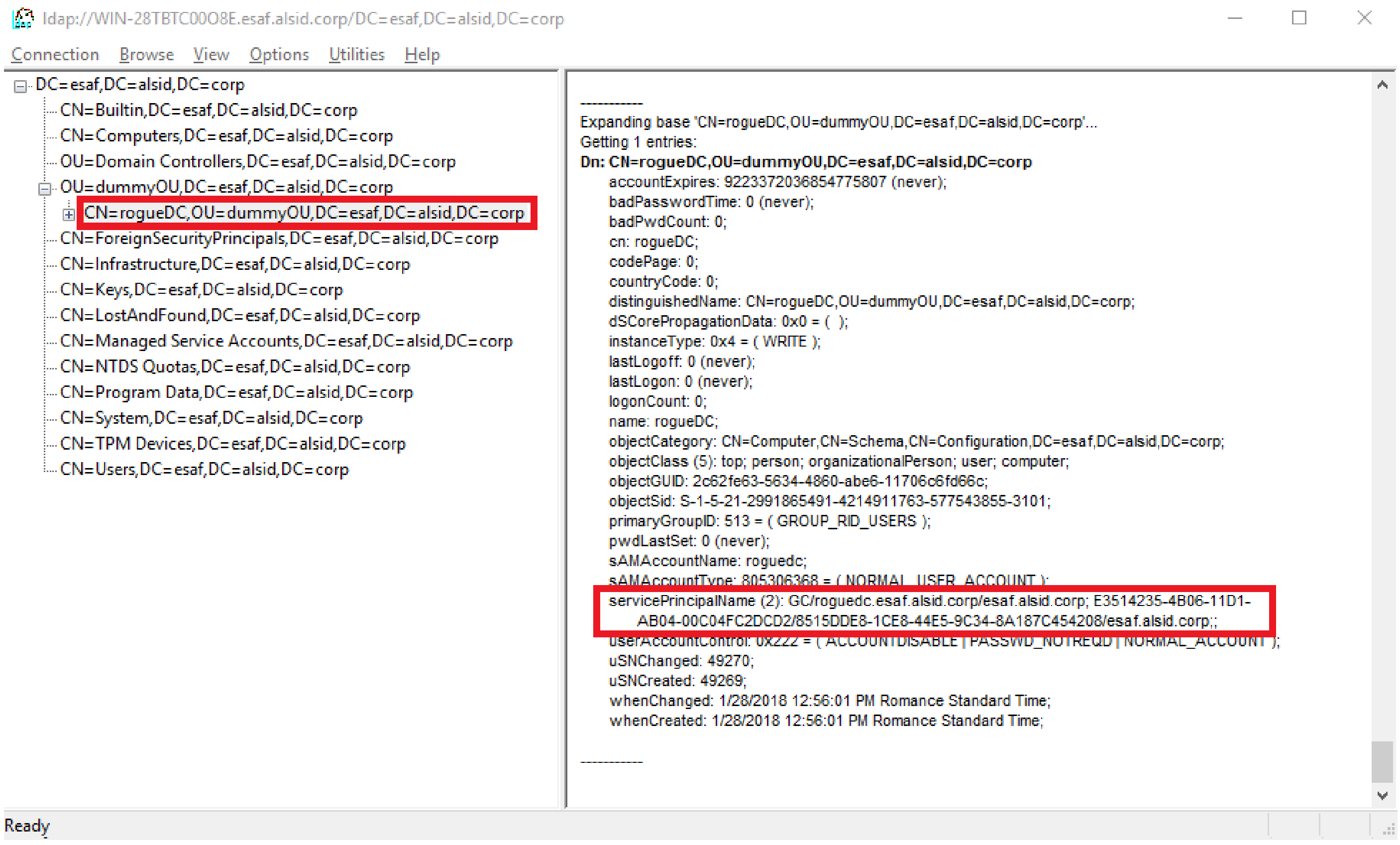
In this way, the "DCShadow" attack will use a legitimate computer account to be able to authenticate to other DCs. Although the computer object and the nTDSDSA object will bring the ability to authenticate to other DCs, the "DCShadow" attack still needs to let other DCs to connect to the rogue server to replicate illegitimate information from it.

This last requirement is fulfilled using the Kerberos Service Principal Name (SPN). As extensively explained in several publications, SPNs are used by Kerberos service (KDC) to encrypt the Kerberos ticket with the computer account associated with the SPN. In our case, the "DCShadow" attack will add SPNs on the regular computer object used to authenticate.

One of the key findings of Benjamin Delpy and Vincent Le Toux was to isolate the minimum set of SPNs required for the replication process to go through. The results of their studies show that two SPNs are required to let another DC to connect to the rogue server:

- the DRS service class (which has the well-known GUID E3514235–4B06–11D1-AB04–00C04FC2DCD2)
- the Global Catalog service class (which has the string "GC")

```
E3514235-4B06-11D1-AB04-00C04FC2DCD2/8515DDE8-1CE8-44E5-9C34-
8A187C454208/alsid.corp
GC/roguedc.alsid.corp/alsid.corp
```


ldap://WIN-28TBTC00O8E.esaf.alsid.corp/DC=esaf,DC=alsid,DC=corp
Connection Browse View Options Utilities Help
Ready

```
DC=esaf,DC=alsid,DC=corp
    CN=Builtin,DC=esaf,DC=alsid,DC=corp
    CN=Computers,DC=esaf,DC=alsid,DC=corp
    OU=Domain Controllers,DC=esaf,DC=alsid,DC=corp
    OU=dummyOU,DC=esaf,DC=alsid,DC=corp
        CN=rogueDC,OU=dummyOU,DC=esaf,DC=alsid,DC=corp
    CN=ForeignSecurityPrincipals,DC=esaf,DC=alsid,DC=corp
    CN=Infrastructure,DC=esaf,DC=alsid,DC=corp
    CN=Keys,DC=esaf,DC=alsid,DC=corp
    CN=LostAndFound,DC=esaf,DC=alsid,DC=corp
    CN=Managed Service Accounts,DC=esaf,DC=alsid,DC=corp
    CN=NTDS Quotas,DC=esaf,DC=alsid,DC=corp
    CN=Program Data,DC=esaf,DC=alsid,DC=corp
    CN=System,DC=esaf,DC=alsid,DC=corp
    CN=TPM Devices,DC=esaf,DC=alsid,DC=corp
    CN=Users,DC=esaf,DC=alsid,DC=corp
```

```
-----------
Expanding base 'CN=rogueDC,OU=dummyOU,DC=esaf,DC=alsid,DC=corp'...
Getting 1 entries:
Dn: CN=rogueDC,OU=dummyOU,DC=esaf,DC=alsid,DC=corp
    accountExpires: 9223372036854775807 (never);
    badPasswordTime: 0 (never);
    badPwdCount: 0;
    cn: rogueDC;
    codePage: 0;
    countryCode: 0;
    distinguishedName: CN=rogueDC,OU=dummyOU,DC=esaf,DC=alsid,DC=corp;
    dSCorePropagationData: 0x0 = (  );
    instanceType: 0x4 = ( WRITE );
    lastLogoff: 0 (never);
    lastLogon: 0 (never);
    logonCount: 0;
    name: rogueDC;
    objectCategory: CN=Computer,CN=Schema,CN=Configuration,DC=esaf,DC=alsid,DC=corp;
    objectClass (5): top; person; organizationalPerson; user; computer;
    objectGUID: 2c62fe63-5634-4860-abe6-11706c6fd66c;
    objectSid: S-1-5-21-2991865491-4214911763-577543855-3101;
    primaryGroupID: 513 = ( GROUP_RID_USERS );
    pwdLastSet: 0 (never);
    sAMAccountName: roguedc;
    sAMAccountType: 805306368 = ( NORMAL_USER_ACCOUNT );
    servicePrincipalName (2): GC/roguedc.esaf.alsid.corp/esaf.alsid.corp; E3514235-4B06-11D1-
        AB04-00C04FC2DCD2/8515DDE8-1CE8-44E5-9C34-8A187C454208/esaf.alsid.corp;;
    userAccountControl: 0x222 = ( ACCOUNTDISABLE | PASSWD_NOTREQD | NORMAL_ACCOUNT );
    uSNChanged: 49270;
    uSNCreated: 49269;
    whenChanged: 1/28/2018 12:56:01 PM Romance Standard Time;
    whenCreated: 1/28/2018 12:56:01 PM Romance Standard Time;

-----------
```

A rogue computer account having the SPN of a DC

When triggering its attack, "DCShadow" will set those two SPNs to its targeted computer account. More precisely, the SPNs will be set using the DRSAddEntry RPC function as described in the CreateNtdsDsa function documentation (more details about MS-DRSR RPC are provided in the next section).

For now, we can register our rogue domain controller into the replication process and be authenticated by another DC. The remaining step is now to force the DC to initiate the replication process with our malicious data.

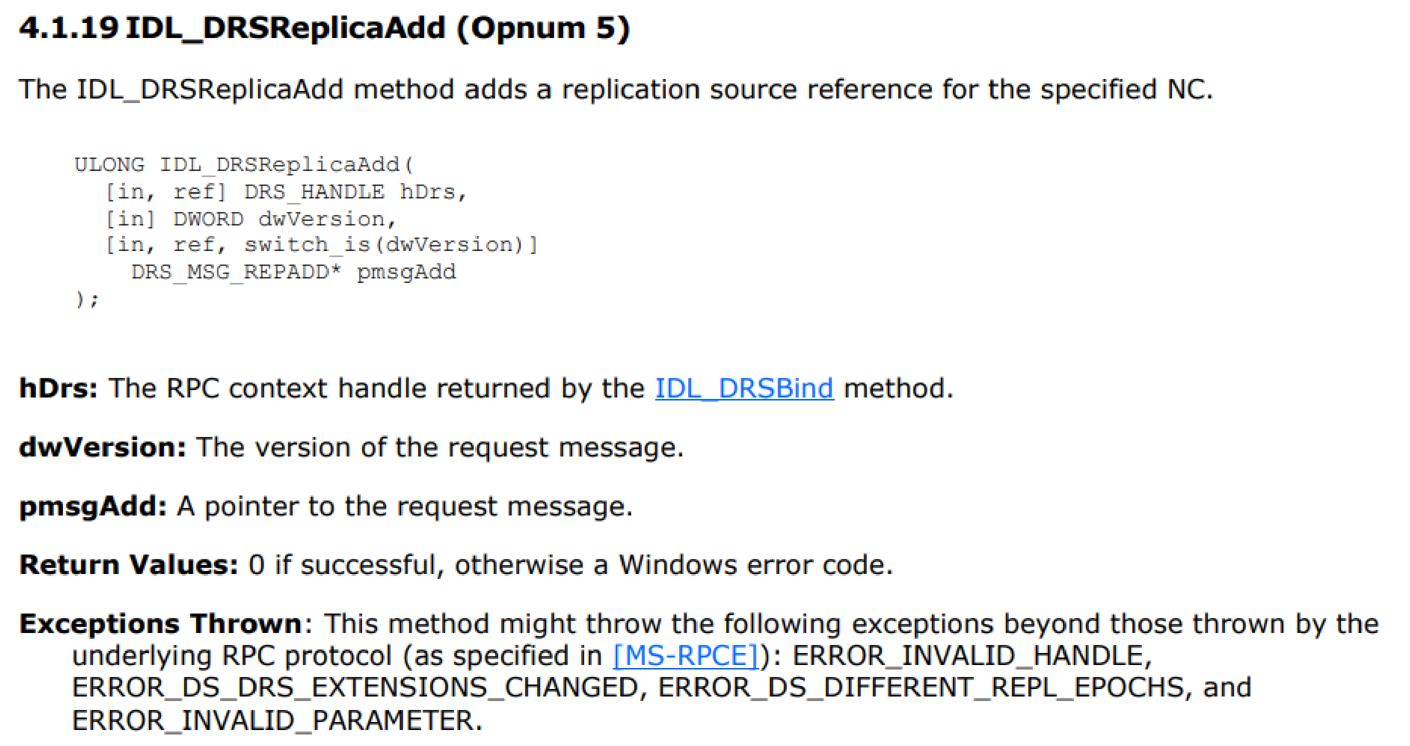
Injecting illegitimate objects

In the previous parts, we gathered all the requirements to register in the replication process, in this final chapter we will study how the "DCShadow" attack injects its illegitimate information into the DNS infrastructure.

To serve illegitimate data, the rogue domain controller will have to implement the minimal set of RPC functions required by the MS-DRSR specifications: IDL_DRSBind, IDL_DRSUnbind, IDL_DRSGetNCChanges, ecifications and are now implemented into Benjamin Delpy's

The final step of the "DCShadow" attack is to trigger the replication process. To do so, two strategies can be conducted:

- Wait for the KCC process of another DC to initiate the replication process (requires 15 minutes delay)
- Force the replication by invoking the DRSReplicaAdd RPC function. It will change the content of the repsTo attribute which will start an immediate data replication.



4.1.19 IDL_DRSReplicaAdd (Opnum 5)

The IDL_DRSReplicaAdd method adds a replication source reference for the specified NC.

```
ULONG IDL_DRSReplicaAdd(
  [in, ref] DRS_HANDLE hDrs,
  [in] DWORD dwVersion,
  [in, ref, switch_is(dwVersion)]
    DRS_MSG_REPADD* pmsgAdd
);
```

hDrs: The RPC context handle returned by the IDL_DRSBind method.

dwVersion: The version of the request message.

pmsgAdd: A pointer to the request message.

Return Values: 0 if successful, otherwise a Windows error code.

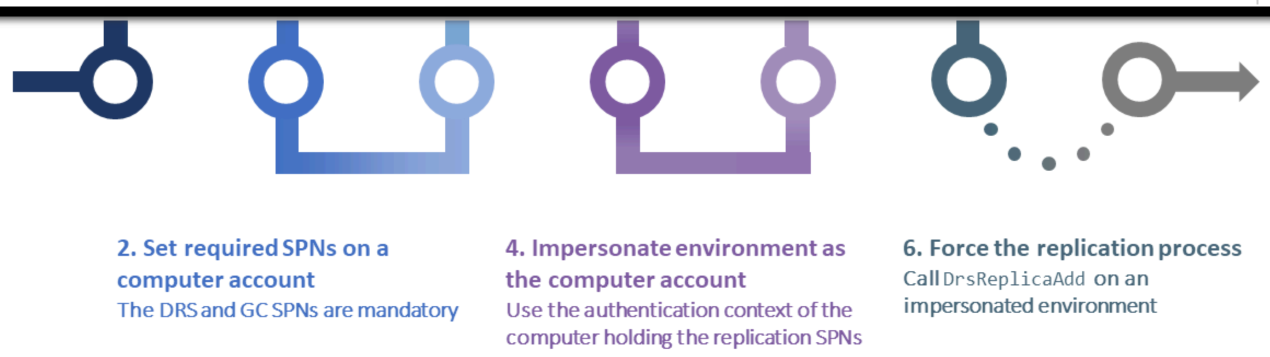
Exceptions Thrown: This method might throw the following exceptions beyond those thrown by the underlying RPC protocol (as specified in [MS-RPCE]): ERROR_INVALID_HANDLE, ERROR_DS_DRS_EXTENSIONS_CHANGED, ERROR_DS_DIFFERENT_REPL_EPOCHS, and ERROR_INVALID_PARAMETER.

Extract of the MS-DRSR specification describing the DRSReplicaAdd IDL

Forcing the replication with the IDL_DRSReplicaAdd RPC is the last step taken during a "DCShadow" attack. It allows to inject arbitrary data into a targeted AD infrastructure. Doing so, it becomes trivial to add any backdoor in the domain (by adding new member on an administrative group, or by setting SID history on a controlled user account for example).

Process summary

The following chart summarizes the different operations achieved during a "DCShadow" attack.

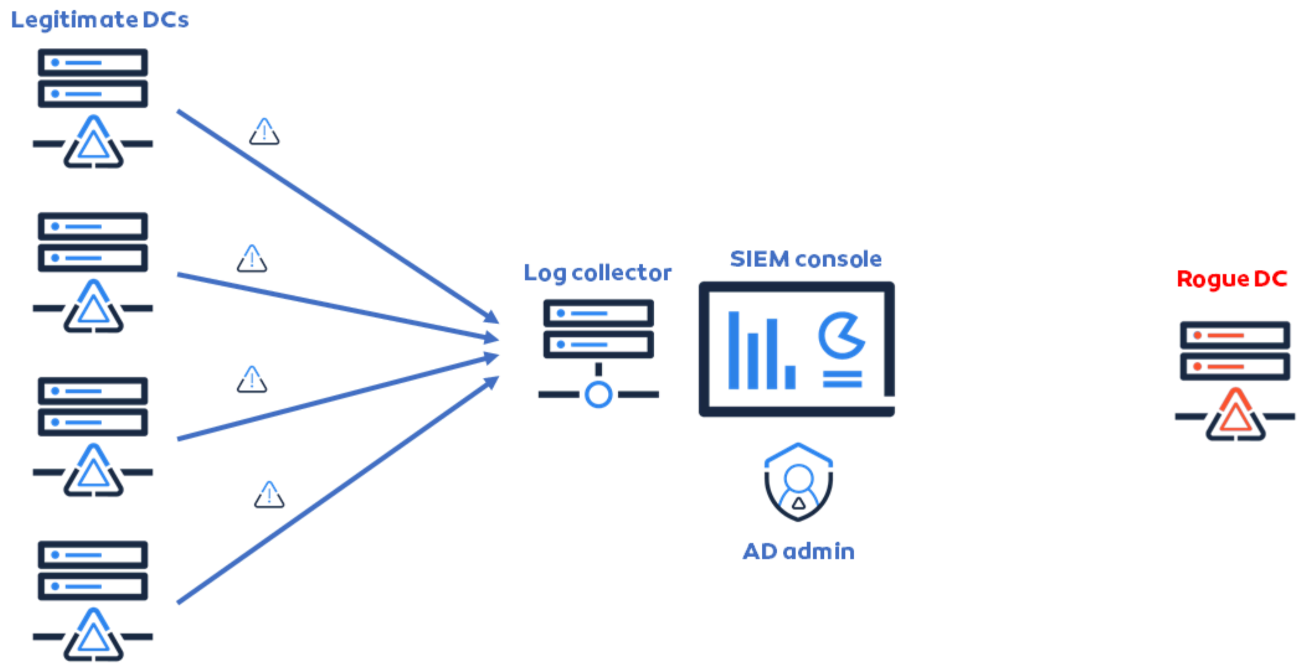

2. Set required SPNs on a computer account
The DRS and GC SPNs are mandatory
4. Impersonate environment as the computer account
Use the authentication context of the computer holding the replication SPNs
6. Force the replication process
Call DrsReplicaAdd on an impersonated environment

DCShadow attack synthetized process

• • •

The consequences of "DCShadow" for blue team strategies

As explained in the research paper, blue teams in charge of AD security monitoring usually rely on event log collection. Computers that are members of a domain are configured to push their logs to a central SIEM to be analyzed.



A simplified SIEM architecture pushing event log through WinRM Event Forwarding protocol

The first problem with this approach is that only legitimate computers send their logs to the log collector. During the "DCShadow", the event logs related to the injection of new data are only created on the attacker's machine, which will obviously not signal itself by sending events to the SIEM. In this way, the "DCShadow" attack can be stealthy as only a few event logs will be generated by legitimate computers.

In fact, this article explains that several prerequisites actions should be ely, the AD modifications involved in setting-up a rogue DC are

changes, determining if such an event is caused by malicious activity or regular AD operations is time consuming and impractical.

Blue teams need a complete redesign of their strategy and shift their focus from log analysis to AD configuration analysis. The naïve approach would be to monitor replications (DrsGetNCChanges RPC changes). In fact, by default, a SACL entry set on the root object of the domain logs the use of extended rights except for domain controllers. In this way, a replication with a user account or non-DC machine must be pretty easy to identified. However, we do not feel this method is the most efficient one. From our point view, three strategies should be implemented to detect “DCShadow” attacks:

1. The Configuration partition of the schema should be looked at carefully. nTDSDSA objects in the sites container should be matched with regular domain controllers in the Domain Controllers organizational unit (or better: a list of known DC manually maintained by the administration team). Any object showing up in the first but not in the second should be investigated. Please notice that the rogue nTDSDSA object is removed right after the publication of the illegitimate objet. To be efficient, detection measure should be able to catch object creation.
2. As shown in the previous paragraphs, DCs need an authentication provider. To be able to push changes, a rogue DC will need to have one accessible through Kerberos, with a specific service. In practical terms, it means having a Service Principal Name (SPN) beginning with the “GC/” string. The well-known RPC interface GUID “E3514235–4B06–11D1-AB04–00C04FC2DCD2” can also be used. Computers having this service but not present in the DC OU should also be carefully investigated.
3. Using “DCShadow” requires an attacker to have elevated privileges. Analyzing and monitoring the permissions present in the Configuration partition will allow blue teams to make sure nobody is able to alter it except legitimate administrators. Any DACL granting access to a non-privileged entity can also be a sign of a possible backdoor.

• • •

Final thoughts

What is most important to take away from this analysis is that “DCShadow” and gain administrative access to your AD using "DCShadow". Bottom-line is: if your AD is properly configured and secured, you do not need to take any urgent actions.

"DCShadow" does not require any urgent patching campaign nor special configuration to be applied, this has nothing to do with WannaCry/NotPetya incident response.

Not being a vulnerability, "DCShadow" will not be patched by a Microsoft update. Trying to counter it would need to change the way AD works, and hence break the system. The authors of the research previously published the "DCSync" attack and Microsoft did not issue any patch, as it only uses legitimate APIs. "Fixing" it would mean forbidding DC replication. *If it ain't broke, don't fix it*. AD is not broken.

However, the fact that a new attack method is publicly available for anyone to use needs to be considered. It offers an extremely stealthy way for privileged attackers to perform actions, so detection strategies should be updated to reflect this new threat. Traditional event log analysis methods will probably fail to detect "DCShadow" usage. To efficiently detect this attack technique, it requires being able to continuously monitor the AD database to isolate illegitimate changes. This is what we do at Alsid and we are very proud to already protect our customers against this attack. For more information on how we tackle this challenge, head to www.alsid.eu.

Active Directory Cybersecurity Dcshadow Mimikatz Information Security

One clap, two clap, three clap, forty?

By clapping more or less, you can signal to us which stories really stand out.

114 1

Luc Delsalle Follow

Security researcher focus on MS Active Directory Infrastructures · Former redteamer and systems security engineer

Alsid blog Follow

Detect directory breaches before attackers do.