Is this SID taken? Varonis Threat Labs Discovers Synthetic SID Injection Attack

varonis.com/blog/synthetic-sid

Varonis Threat Labs researchers have discovered a technique where threat actors with existing high privileges can inject synthetic SIDs into an Active Directory Access Control List (ACL). This creates a scenario where backdoors and hidden permission grants can occur when a new account is created with a matching legitimate SID.

This attack is made possible as:

- SIDs are easily guessable as they're predominantly consecutively assigned
- Active Directory does not verify if a SID applied to an ACL is valid

We're terming the SIDs which conform to the formatting rules of legitimate SIDs but don't actually yet reference an object to be "synthetic".

Background

Active Directory's permission system is composed of three parts:

1. **Trustees:** objects which have permissions applied. This most commonly includes user accounts, groups, and computer accounts.
2. **Security Identifier (SID):** Within Active Directory, security principals are identified by a security identifier (SID). The SID is a unique identifier used to represent any entity that can be authenticated by the operating system. It can be loosely compared to a social security number or a citizen ID but for a domain object. The SID is issued by a Domain Controller and is assigned to an object at the time of its creation. It cannot be reused or used to identify another entity.
3. **Access Control List (ACL):** the mapping between an object (SID) and permissions within Active Directory.

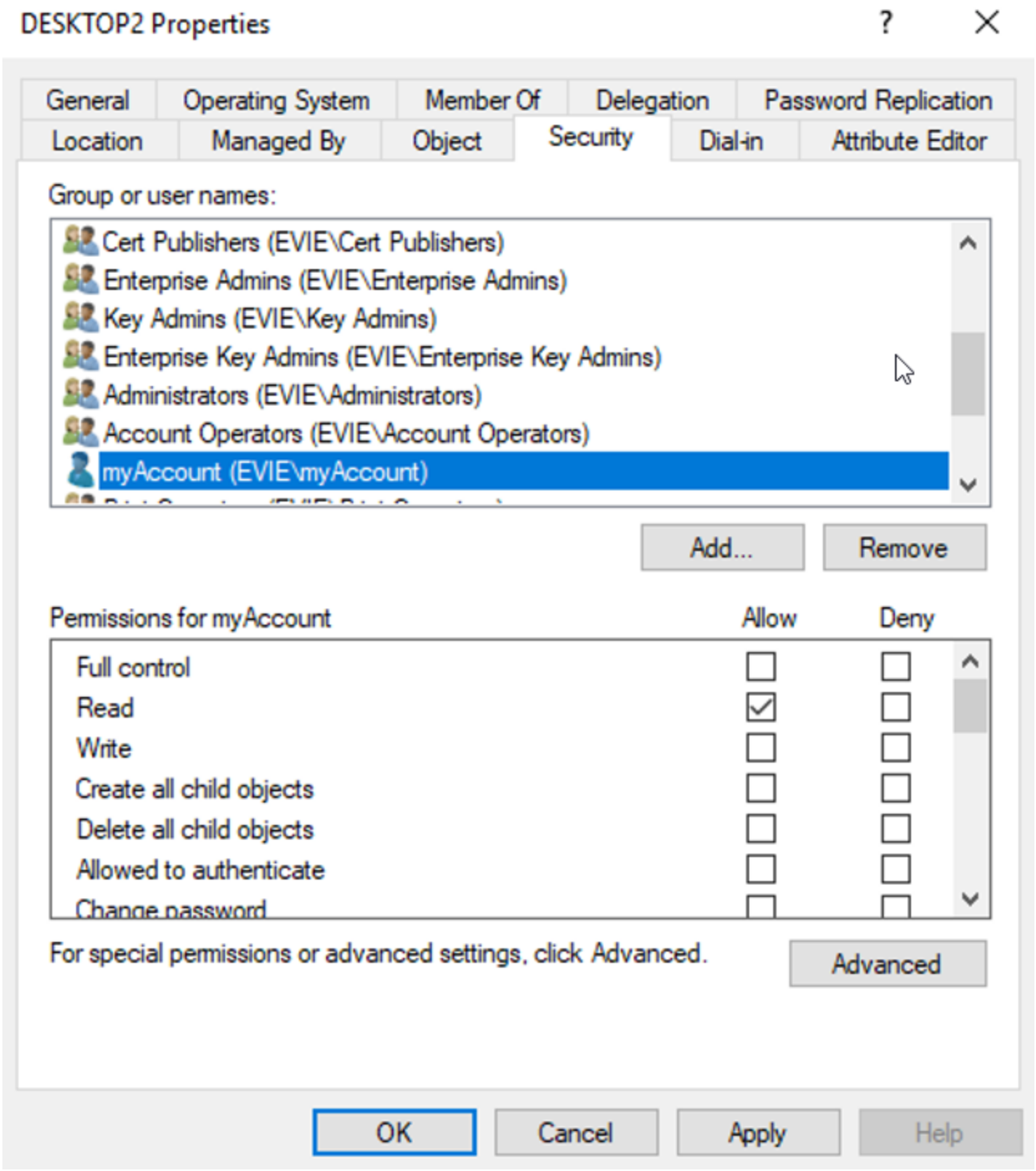
No SID Verification

When an ACL is created the trustees' SID is not verified to exist on the domain. Because no validity check is done for the SID, with sufficient permissions, it's possible to add a non-existent "synthetic" SID to an ACL.

These non-existing (Synthetic) SIDs with ACL permissions persist innocuously on the ACLs until a new user or computer account is created that is assigned the previously synthetic SID. These new accounts instantly inherit the previously granted ACL permissions.

How to examine an ACL

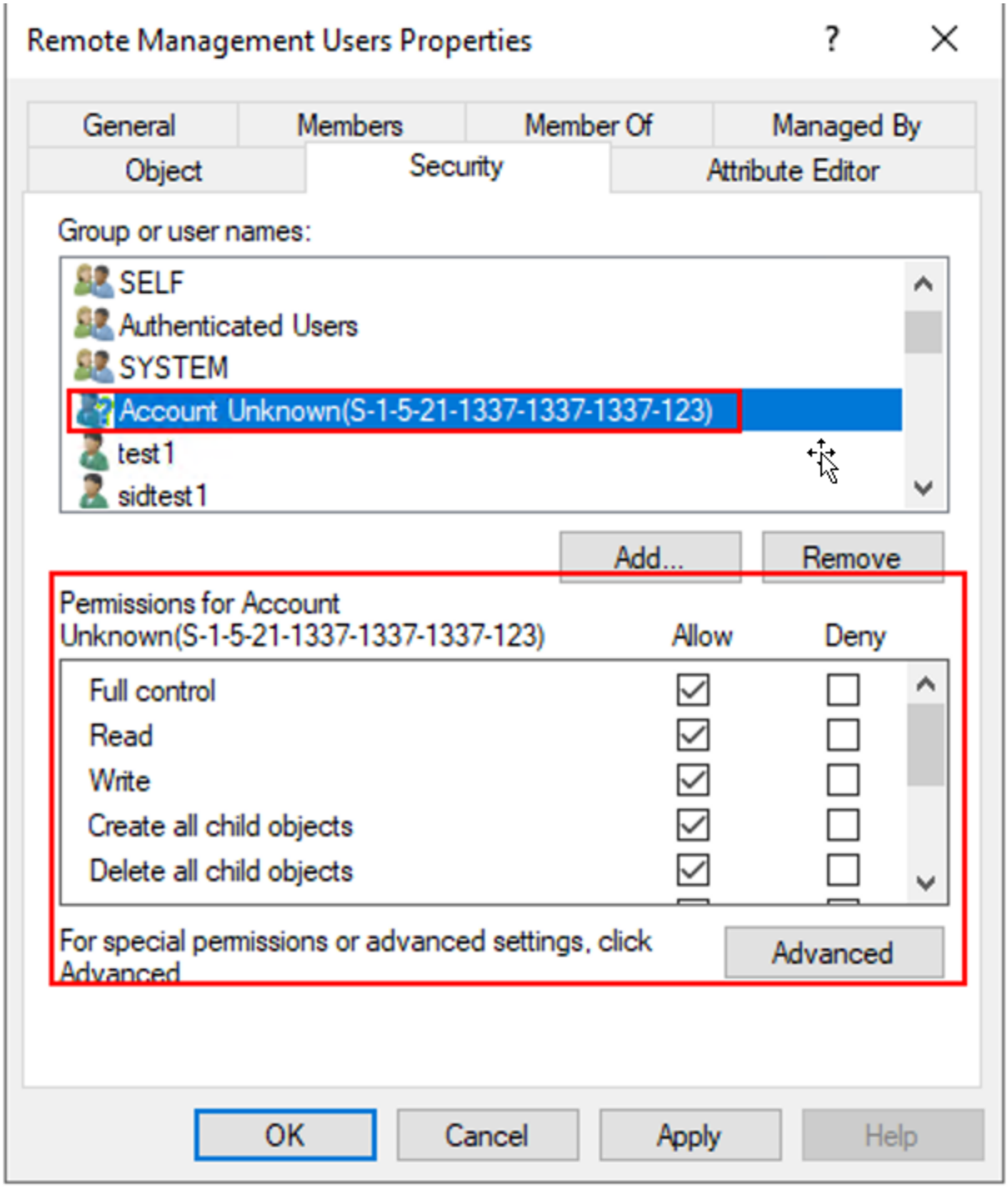
We can see the ACL of an object by going to its security tab:



Windows resolves the entry's SID and presents the username for readability. However, behind the scenes, the ACL identifies the user via their SID as defined by the SDDL (more on SDDLs here https://docs.microsoft.com/en-us/windows/win32/secauthz/security-descriptor-string-format).

Injecting a synthetic SID into an ACL

As Windows doesn't verify that the SIDs exist on the domain when an ACL is created it's possible to insert our Synthetic SID into any object's ACL we have privileges over:



Note: the domain section of the SID is changeable, but the "S-1-5-21" is not.
The Synthetic SID in the screenshot both:

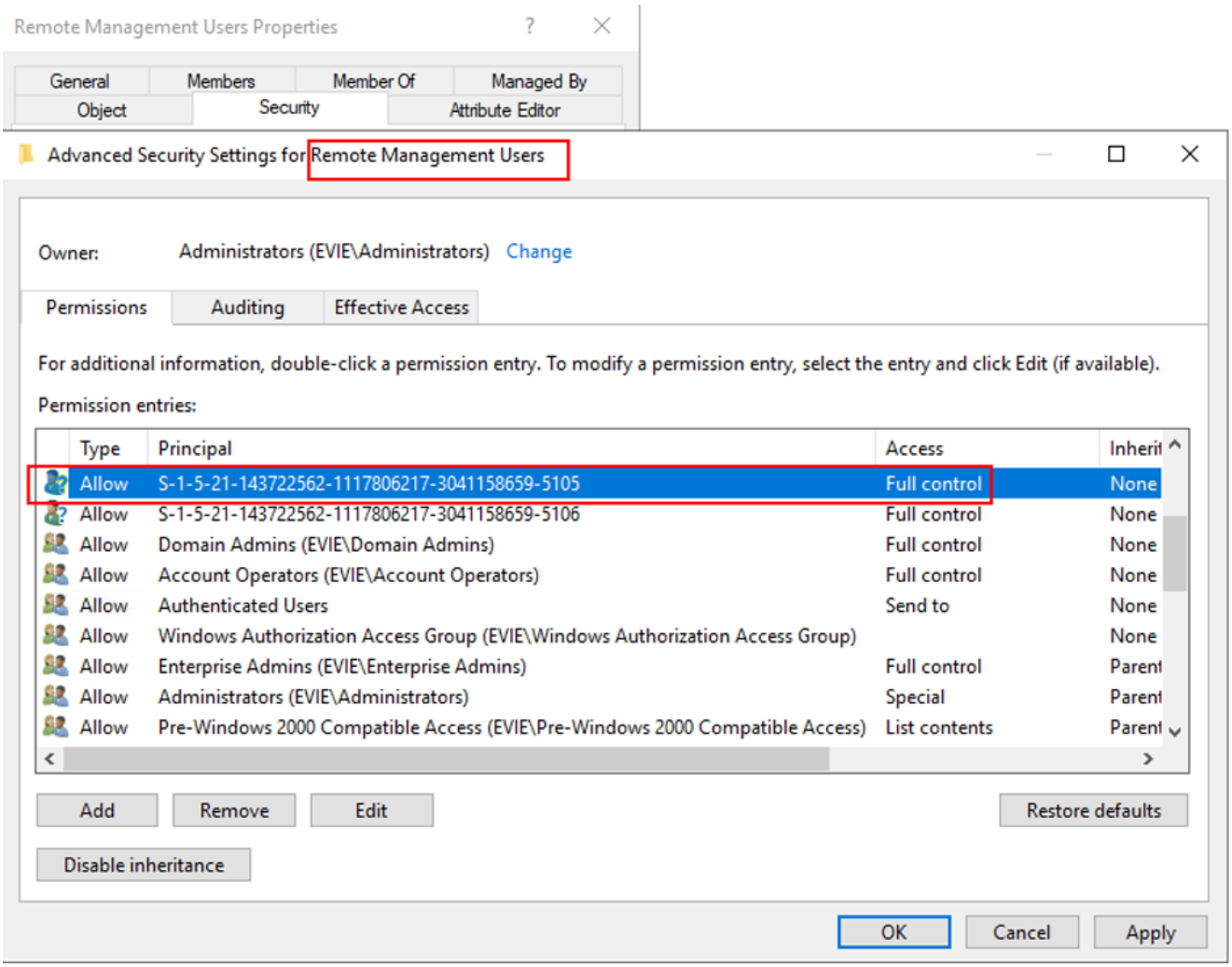
- Cannot be resolved (“Account Unknown”) because it is not assigned
- Valid for the ACL entry

Getting control over an existing SID is not easily achieved as SIDs can’t be taken from other users or reused. However, by mapping what SIDs currently exist in the domain we can

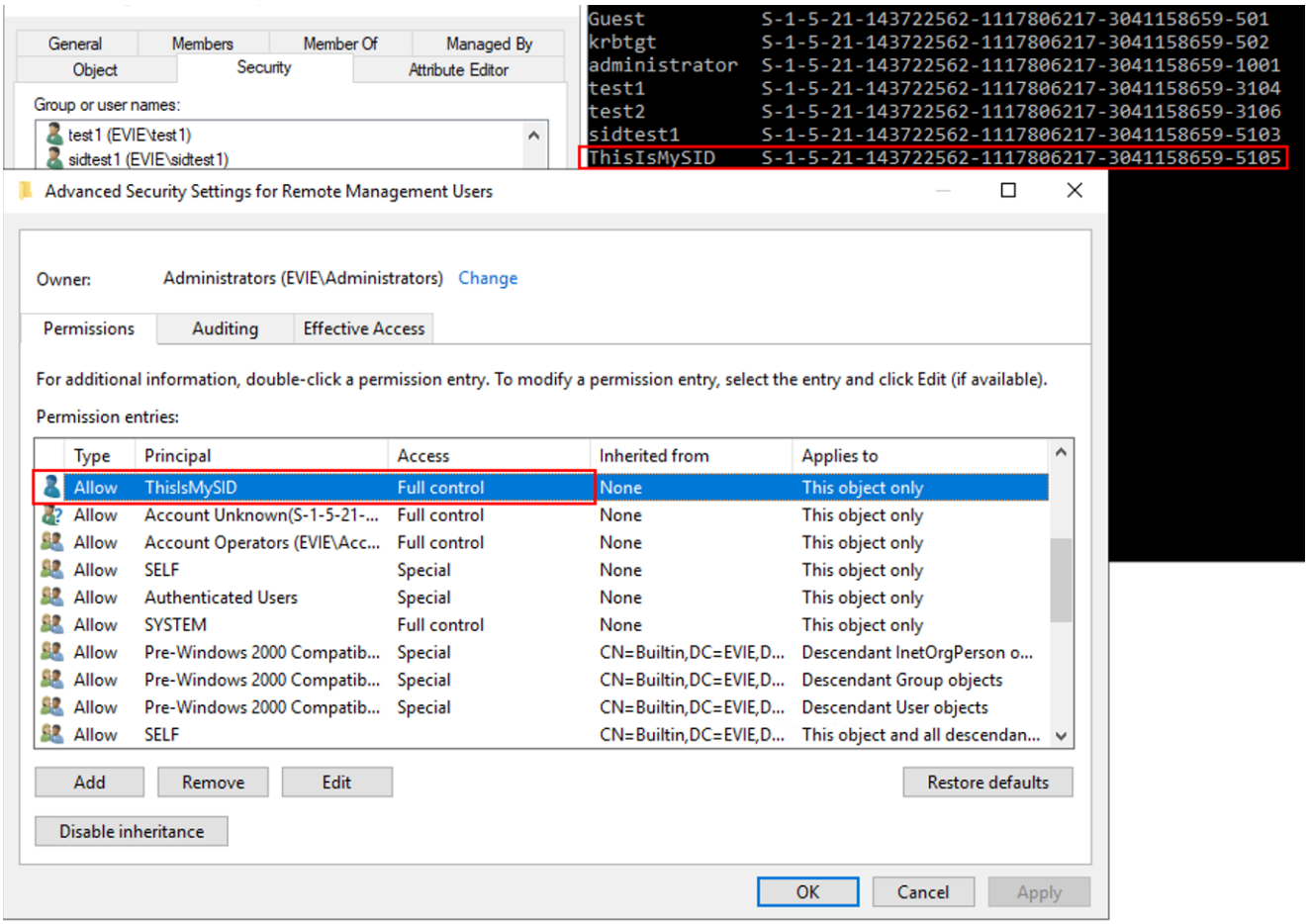
predict what SIDs new users will be created with allowing us to create a scenario where a newly created account inherits our injected permissions.
We can map the currently existing SIDs with PowerShell:

```
(([adsisearcher]"(objectSid=*)").FindAll()).Properties.objectsid | ForEach-Object {(New-Object System.Security.Principal.SecurityIdentifier($_,0)).Value}
```

The SID in the following image has no account related to it and is the next available SID in the domain. It was granted Full Control on the “Remote Management Users” object:



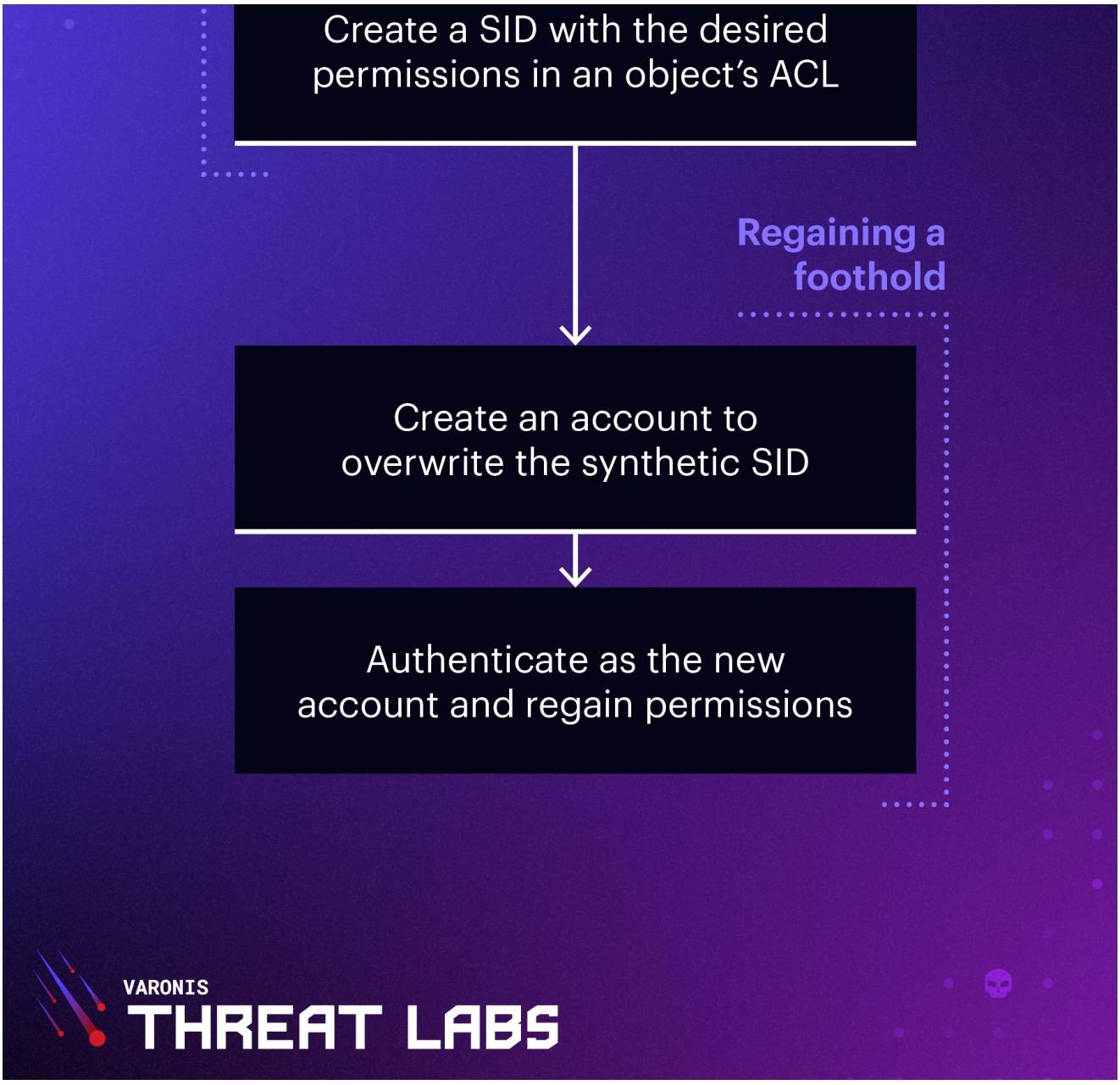
We created an account called “ThisIsMySid” and it took over the SID:



The user “ThisIsMySID” now has full control over the group object.

It’s worth noting that this trick also works for assigning Windows Privileges and rights such as SeDebugPrivilege, SeRemoteInteractiveLogonRight, or other Privilege Constants.

Exploitation



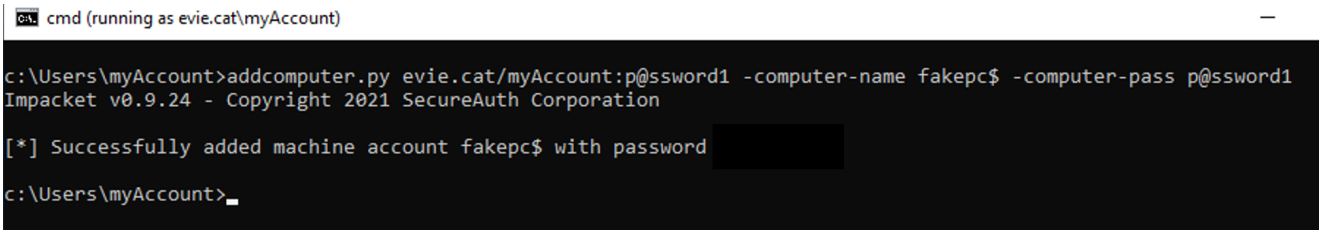
The main exploitation vector here is persistence. Threat actors with domain control can add permissions and privileges to future SIDs and regain a foothold by creating a user or computer account.

Creating an account is not much of an issue assuming you have control over a regular user account. Authenticated users can create up to 10 computer accounts by default, and computer accounts get SIDs assigned to them just like regular users which allows for this exploitation.

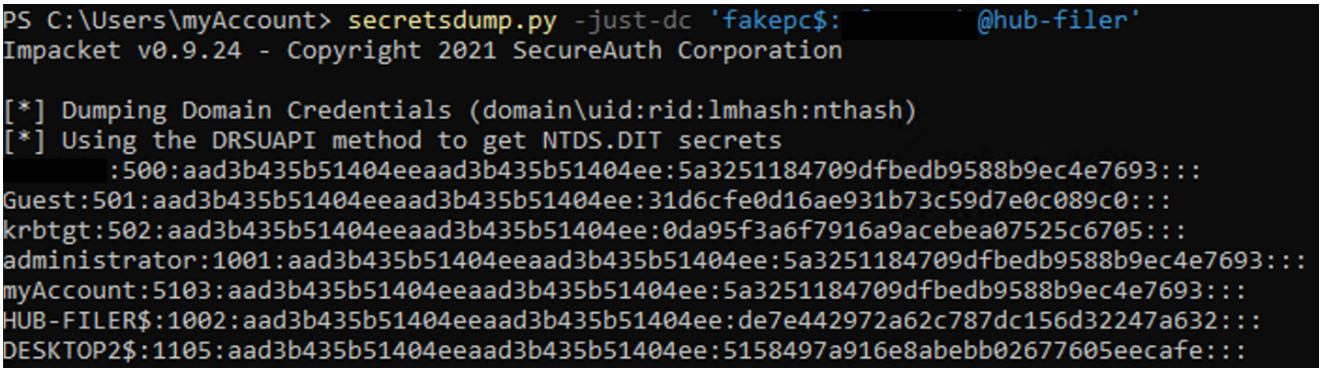
DCSync Exploitation Scenario

By adding a SID to the domain object and granting it synchronization privileges (which are required for the DCSync attack), the threat actor planted the backdoor. And of course, more than one SID can be added to make sure that it's not overwritten by regular activity.

To regain a foothold, the threat actor would have to gain control over a standard user account (possibly by phishing) and use that account to create a computer account:



The newly created account will replace one of the available SIDs and have DCSync permissions.



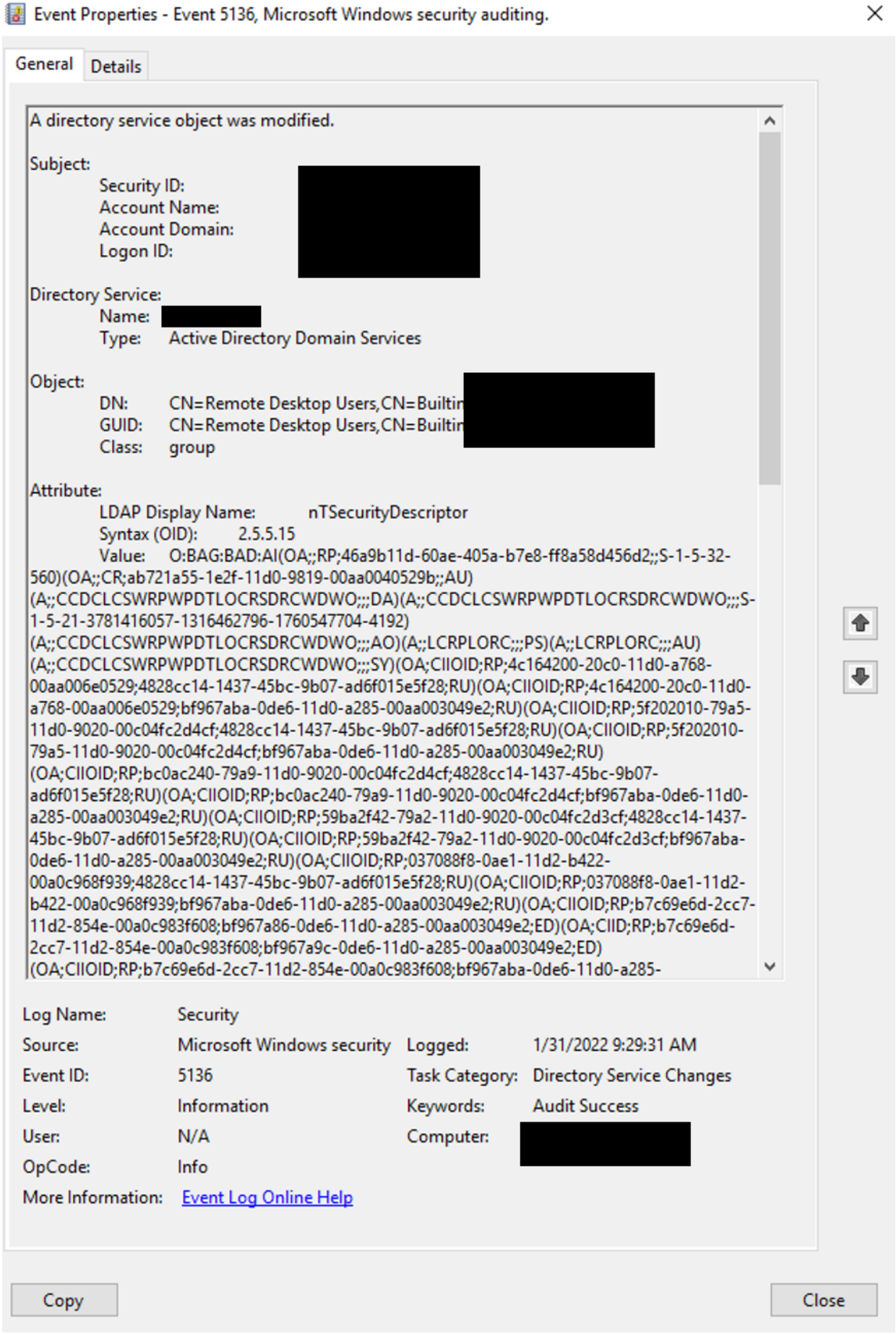
Remediation and Monitoring

Microsoft does not consider this a security issue, however, monitoring is still recommended as the assignment of synthetic SIDs is anomalous behavior.

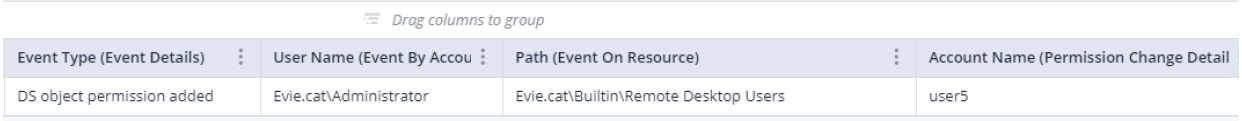
Monitoring of the following is recommended to mitigate the risk of this technique.

- Alerts on abnormal privilege changes, rights assignments, and permission grants in Active Directory environments whether they are done automatically via scripts or malware, or manually by an active threat.
- Behavior-based modeling (as done by Varonis) on sensitive objects for ACL changes.

- Changes on directory service objects in your organization (Windows event 5136)

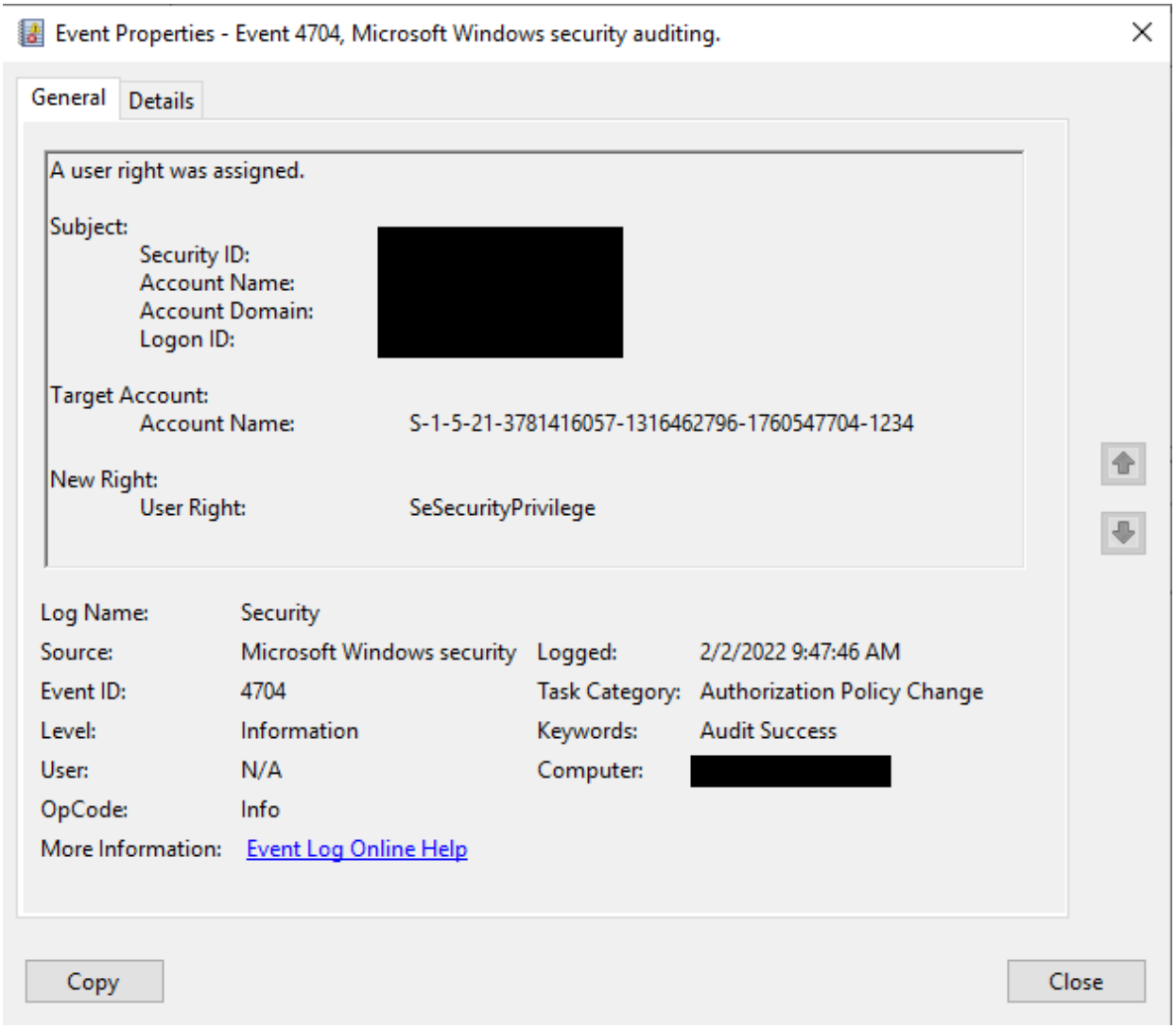


- Behavior-based modeling (like that provided by Varonis) to alert when a user is granted permissions over a sensitive object or to monitor a trustee that’s added to the object and alert if it doesn’t exist.


Drag columns to group

Event Type (Event Details)	User Name (Event By Accou	Path (Event On Resource)	Account Name (Permission Change Detail
DS object permission added	Evie.cat\Administrator	Evie.cat\Builtin\Remote Desktop Users	user5

- Direct privilege assignments (Windows event 4704) may indicate synthetic SID injection.



Event Description
The user right SeManageVolumePrivilege was assigned to vrnslab.se\S-1-5-21-3781416057-1316462796-1760547704-1237
The user right SeSystemtimePrivilege was assigned to vrnslab.se\S-1-5-21-3781416057-1316462796-1760547704-1238
The user right SeRestorePrivilege was assigned to vrnslab.se\S-1-5-21-3781416057-1316462796-1760547704-1238
The user right SeImpersonatePrivilege was assigned to vrnslab.se\S-1-5-21-3781416057-1316462796-1760547704-1238
The user right SeCreatePagefilePrivilege was assigned to vrnslab.se\S-1-5-21-3781416057-1316462796-1760547704-1237
The user right SeCreateSymbolicLinkPrivilege was assigned to vrnslab.se\S-1-5-21-3781416057-1316462796-1760547704-1238

Removing Orphaned SIDs

Having a process to monitor and remove orphaned SIDs will prevent the attacker from gaining control over the synthetic SIDs as well.
Consider using PowerShell, ICACLS, or dedicated tools to find orphaned SIDs and remove them.

Sources

Eric Saraga