Replication Metadata

Ghosts in the Wire

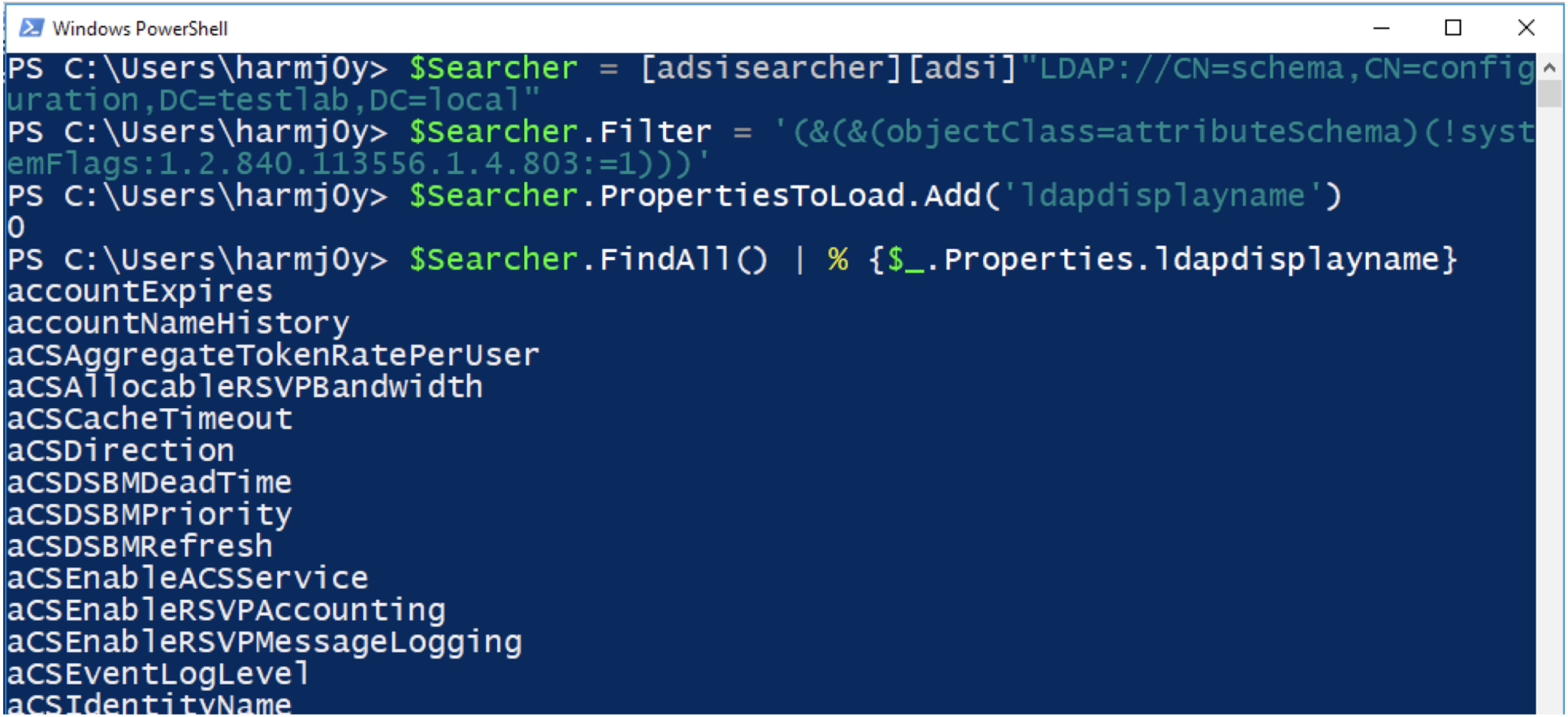
Background

- When a change is made to a domain object on a domain controller in Active Directory, those changes are replicated to other domain controllers in the same domain
  - As part of the replication process, metadata about the replication is preserved in “two constructed attributes”
- Any domain user can enumerate these attributes!
- Why care?
  - Let’s us track some changes to AD objects WITHOUT enabling additional logging!
  - More info: https://www.harmj0y.net/blog/defense/hunting-with-active-directory-replication-metadata/

What attributes are replicated?

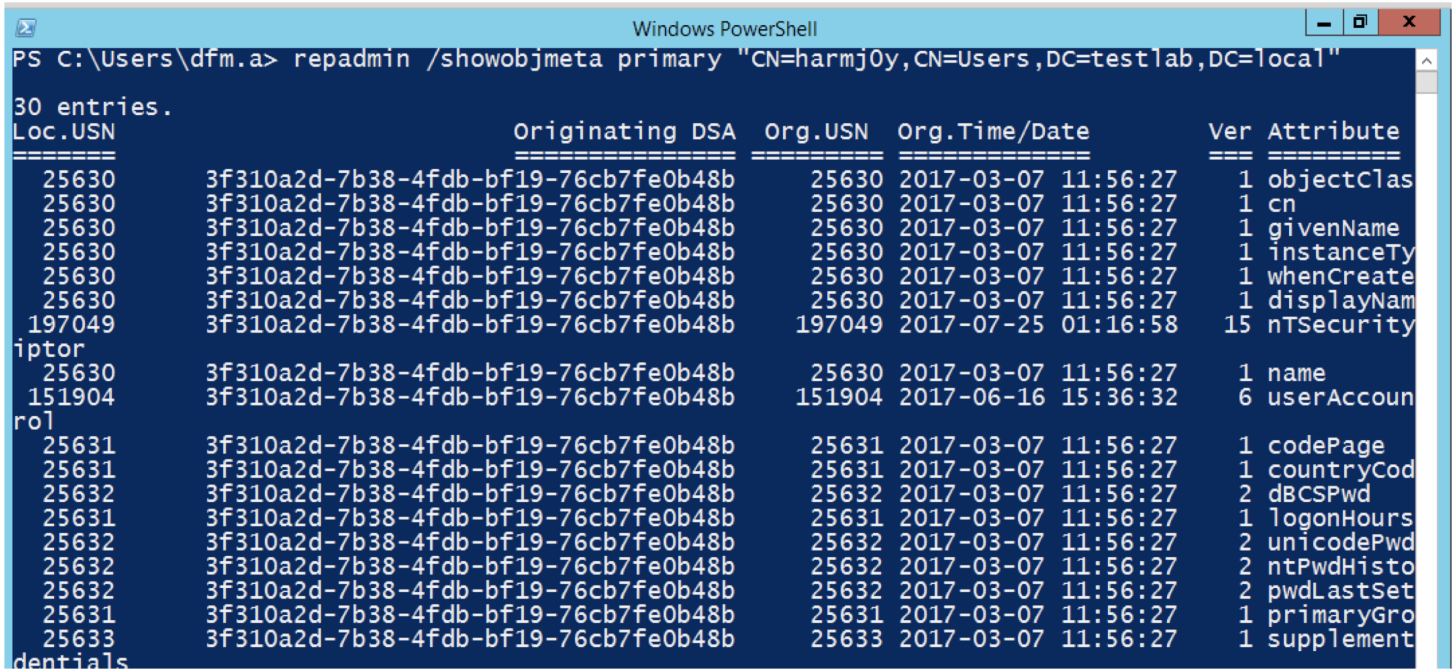
- Object attributes are themselves represented in the forest schema
- They include a **systemFlags** attribute that contains various meta-settings
  - This includes the FLAG_ATTR_NOT_REPLICATED flag, indicating that the given attribute should not be replicated
- So to search for attributes that ARE replicated:
  - The search base needs to be: **CN=schema,CN=configuration,DC=domain,...**
  - The objectClass needs to filter for **attributeSchema**
  - systemFlags is binary, so we need to use **(!systemFlags:1.2.840.113556.1.4.803:=1)**

What attributes are replicated?



Non-PowerShell enumeration

- **REPADMIN /showobjmeta server “CN=objectDN,...”**
  - Output is text, and repadmin is only available on servers...



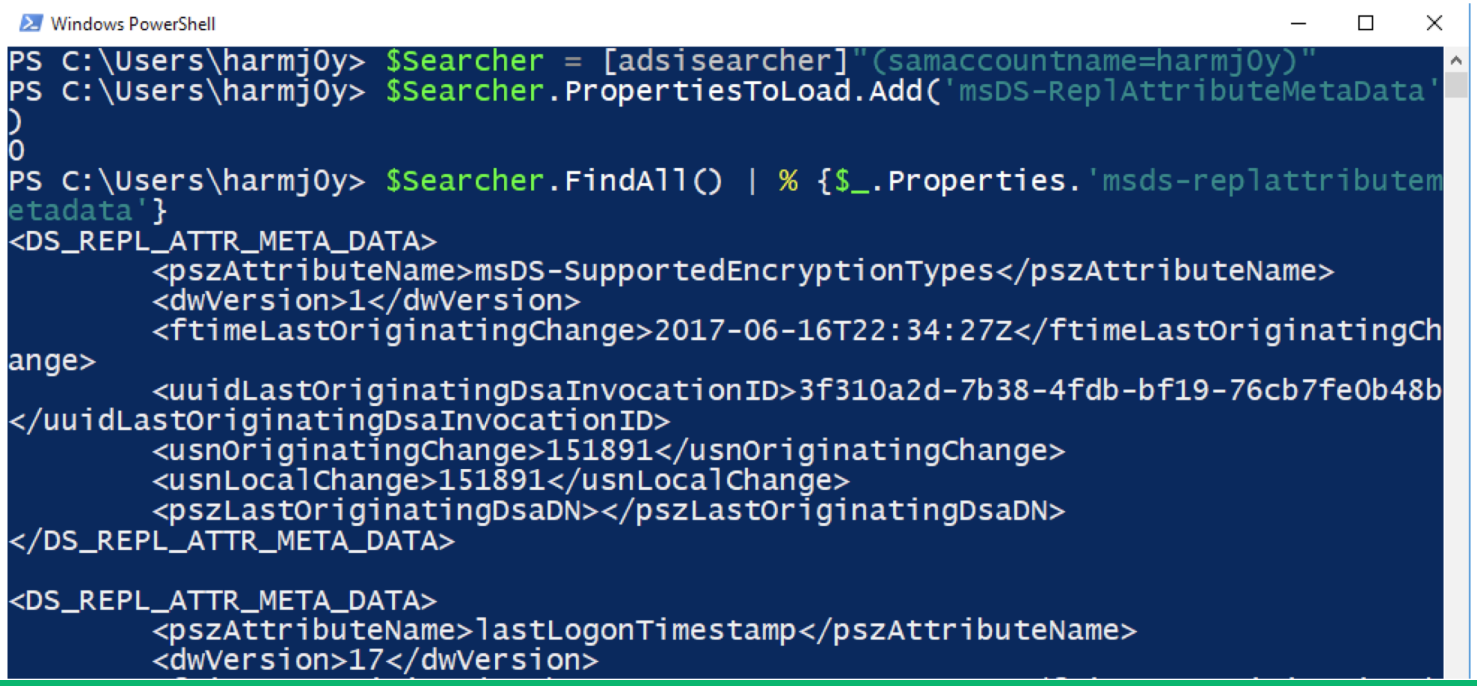
msDS-ReplAttributeMetaData

- The constructed **msDS-ReplAttributeMetaData** property is associated with every user/group/computer/etc.
  - *As long as you have the right to read an object, you can read its metadata!*

- This metadata includes things like
  - The name of the attribute that changed on the object
  - When the attribute changed
  - The number of times the attribute changed
  - The “Directory System Agent” (traceable to a domain controller) that initiated the change

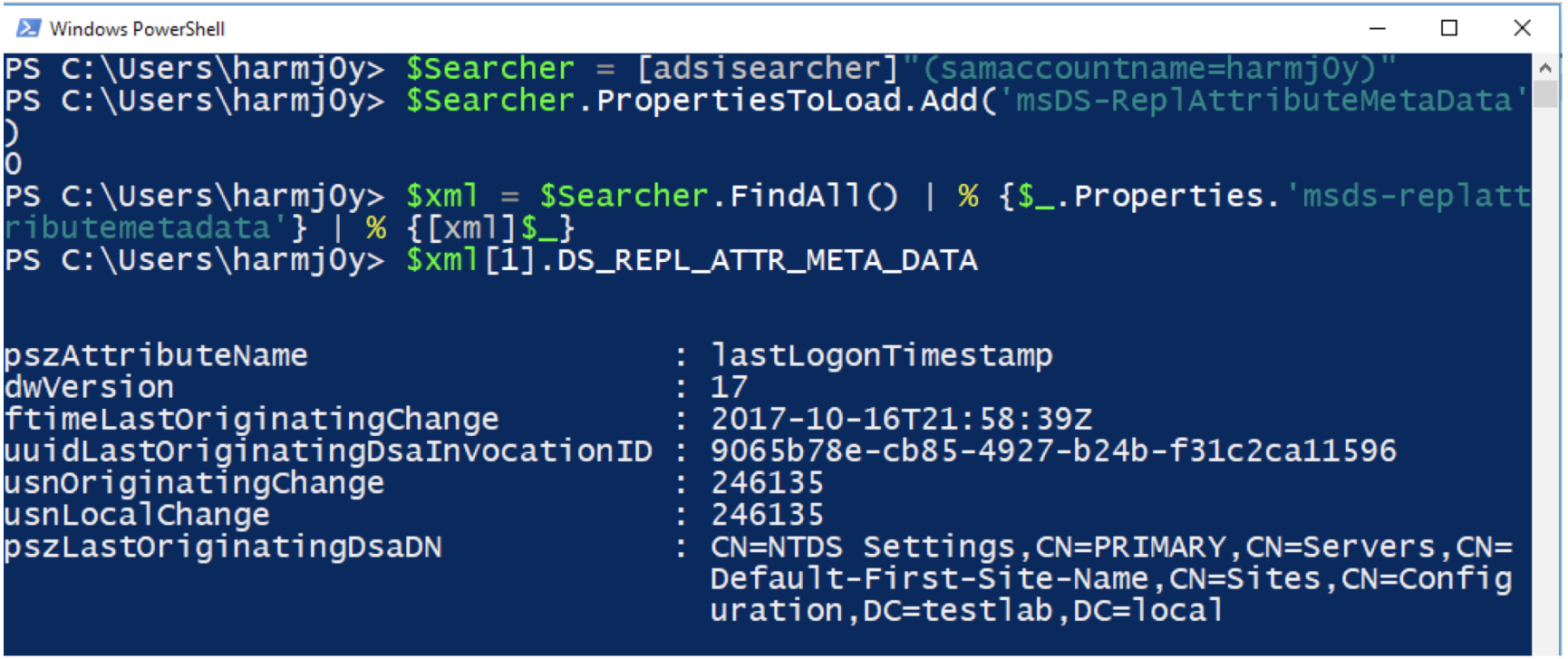
msDS-ReplAttributeMetaData

- To retrieve, just use **PropertiesToLoad.Add('msDS-ReplAttributeMetaData')** with your searcher:



msDS-ReplAttributeMetaData

- The array of XML blobs can be parsed using the **[xml]** accelerator:



Interpreting msDS-ReplAttributeMetaData

- **pszAttributeName** : the name of the attribute that changed
- **dwVersion** : the number of times the attribute has changed
- **ftimeLastOriginatingChange** : the time (in UTC) the attribute changed
- **pszLastOriginatingDsaDN** : the “directory services agent” the change originated from

Sidenote: Linked Attributes

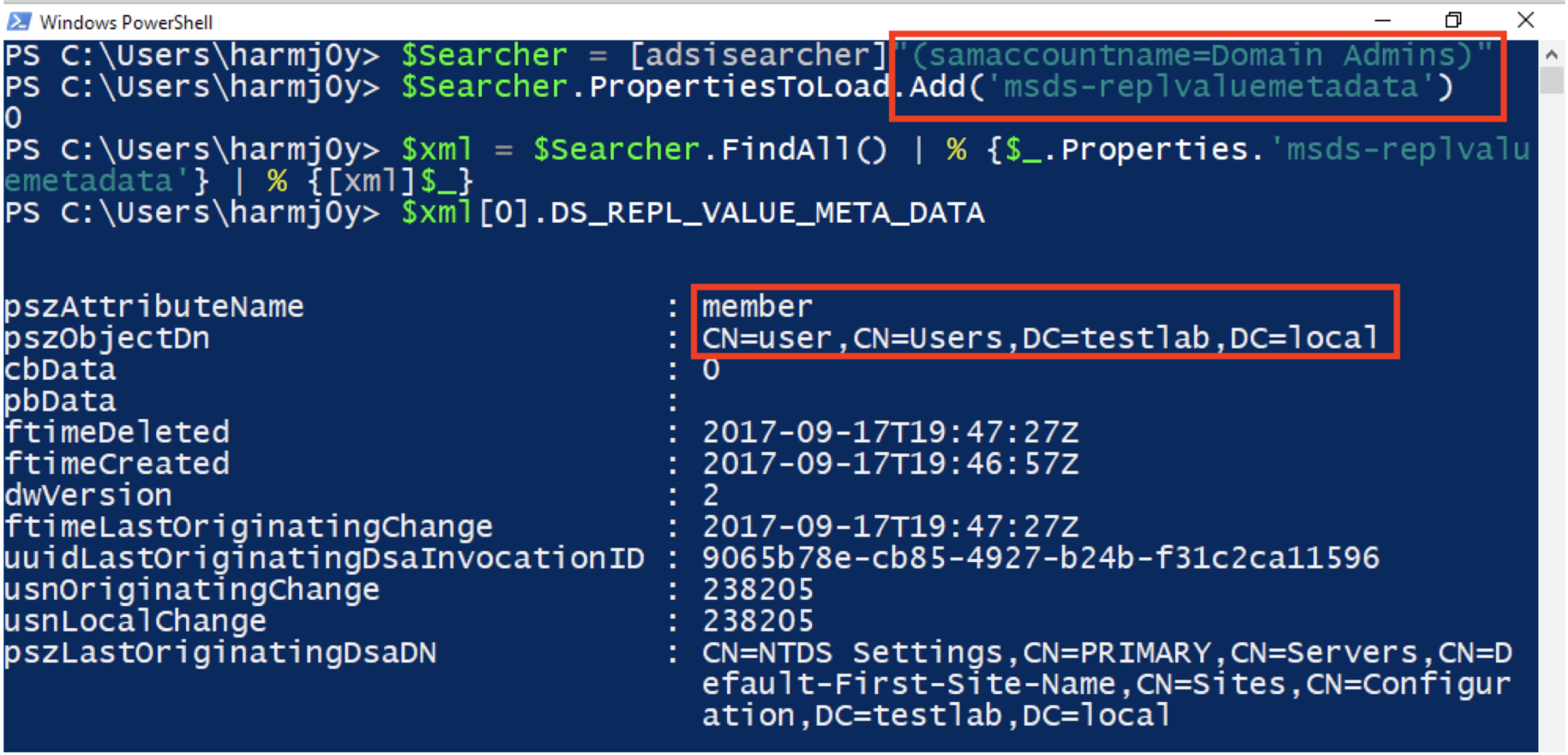
- In order to understand how/why the second attribute is different, you need to be aware of “linked value replication”
  - “*allows individual values of a multivalued attribute to be replicated separately*”
  - **In English:** Active Directory calculates the value of a given attribute, referred to as the *back link*, from the value of another attribute, referred to as the *forward link*

- The **member** property of a group is a *forward link*, while the **memberof** property of a group/user is a *back link*
  - Note: only *forward links* are writable!

msDS-ReplValueMetaData

- Because of how forward/back links are replicated, the previous values of these attributes are stored in replication metadata!
  - This means if we user is added and then removed from a group, we can retrieve the value of the deleted user name!
- Replication metadata is stored as an XML blob (again, only for linked attributes) in the **msDS-ReplValueMetaData** property
- Can be retrieved by adding this property to your searcher, same as **msDS-ReplAttributeMetaData**

msDS-ReplValueMetaData



Interpreting msDS-ReplValueMetaData

- **dwObjectDn** : the member that was added
- **ftimeDeleted** : the time (UTC) the member has been removed (0 if the object is currently still a member)
- **ftimeCreated** : the time (UTC) the member was first added
- **dwVersion** : the number of times the attribute has changed
  - odd if the user is still a member of the group
  - even if the user was added and then removed

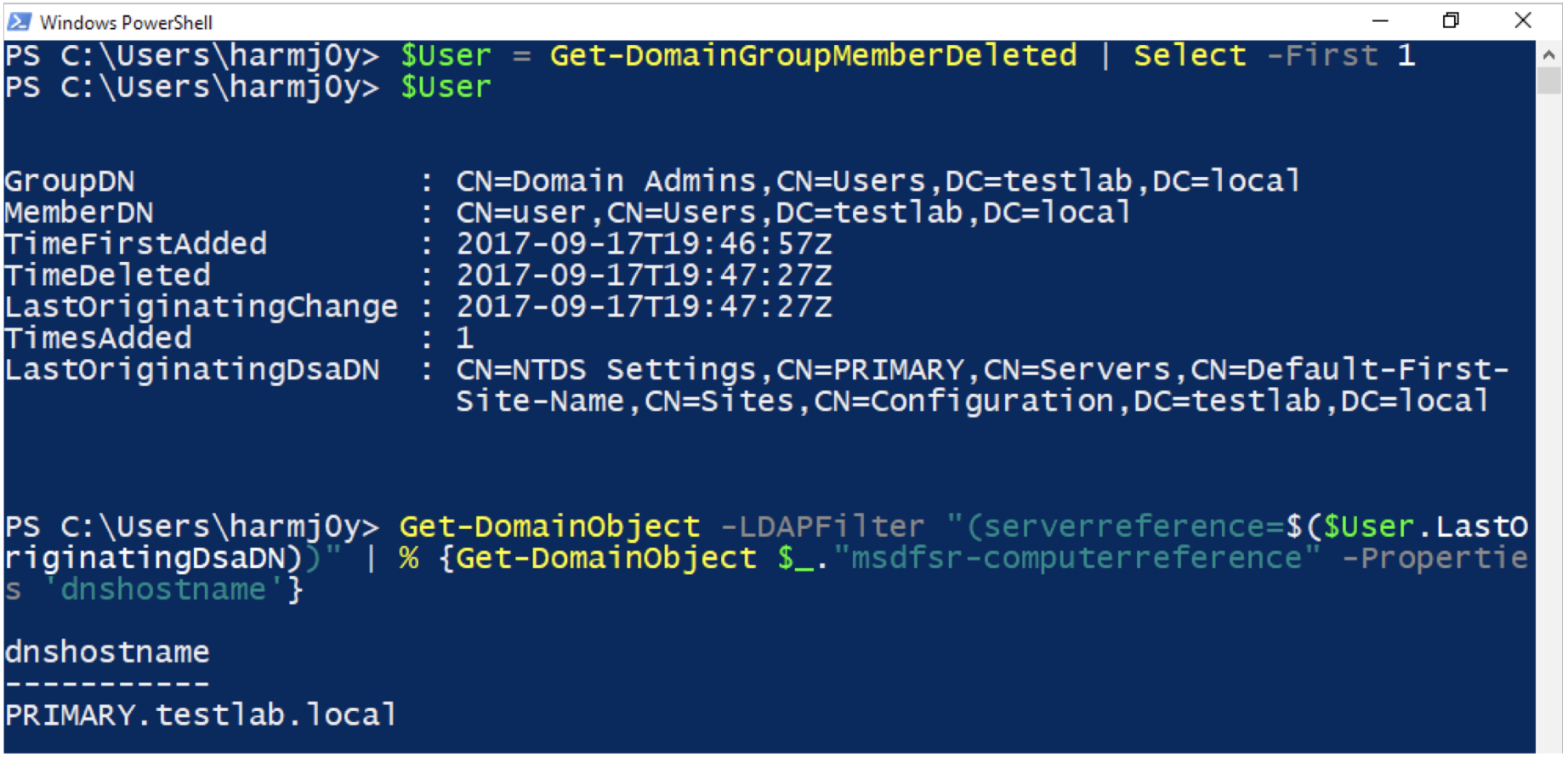
PowerView Implementations

- **Get-DomainObjectAttributeHistory**
  - Retrieves the ‘msds-replattributemetadata’ data and parses the XML to proper object output
- **Get-DomainObjectLinkedAttributeHistory**
  - Retrieves the ‘msds-replvaluemetadata’ data for linked attributes and parses the XML to proper object output
- **Get-DomainGroupMemberDeleted**
  - Retrieves any users who were removed from groups by wrapping **Get-DomainObjectLinkedAttributeHistory**’s functionality
- All of these, by default, retrieve this data for every object in the domain

Resolving **LastOriginatingDsaDN**

- The object has a **NTDS-DSA** category, and is linked to a server topology reference (objectclass=msDFSR-Member) through the **serverreferencebl** property
- This msDFSR-Member object:
  - has a **serverrefrence** property that matches the **LastOriginatingDsaDN**
  - has a list of server distinguished names in its **msdfsr-computerreference** property, which refer to the actual domain controllers
- So we can resolve a LastOriginatingDsaDN by:
  - Using an LDAP filter of **"(serverreference=$LastOriginatingDsaDN)"**
  - Extracting the **msdfsr-computerreference** property
  - Re-querying the domain to return the compelte object

Resolving **LastOriginatingDsaDN**



This would be a good time to attempt Lab: Replication Metadata