Using DsAddSidHistory - Win32 apps

By GrantMeStrength

Archived: 2026-04-05 16:42:11 UTC

The **DsAddSidHistory** function gets the primary account security identifier (SID) of a security principal from one domain (the source domain) and adds it to the **sIDHistory** attribute of a security principal in another (destination) domain in a different forest. When the source domain is in Windows 2000 native mode, this function also retrieves the **sIDHistory** values of the source principal and adds them to the destination principal's **sIDHistory**.

Adding SIDs to a security principal's **sIDHistory** is a security-sensitive operation that effectively grants to the destination principal access to all resources accessible to the source principal, provided that trusts exist from applicable resource domains to the destination domain.

In a native mode Windows 2000 domain a user logon creates an access token that contains the user primary account SID and group SIDs, as well as the user **sIDHistory** and the **sIDHistory** of the groups of which the user is a member. Having these former SIDs (**sIDHistory** values) in the user's token grants the user access to resources protected by access-control lists (ACLs) containing the former SIDs.

This operation facilitates certain Windows 2000 deployment scenarios. In particular, it supports a scenario in which accounts in a new Windows 2000 forest are created for users and groups that already exist in an Windows NT 4.0 production environment. By placing the Windows NT 4.0 account SID in the Windows 2000 account **sIDHistory**, access to network resources is preserved for the user logging onto his new Windows 2000 account.

DsAddSidHistory also supports migration of Windows NT 4.0 backup domain controllers (BDCs) resource servers (or DCs and member servers in a native mode Windows 2000 domain) to a Windows 2000 domain as member servers. This migration requires the creation, in the destination Windows 2000 domain, of domain local groups that contain, in their **sIDHistory**, the primary SIDs of the local groups defined on the BDC (or domain local groups referenced in ACLs on the Windows 2000 servers) in the source domain. By creating a destination local group containing the **sIDHistory** and all members of the source local group, access to the migrated server resources, protected by ACLs referencing the source local group, is maintained for all members.

Note

Use of **DsAddSidHistory** requires an understanding of its broader administrative and security implications in these and other scenarios. For more information, see the white paper "Planning Migration from Windows NT to Microsoft Windows 2000", delivered as Dommig.doc in the Windows 2000 Support Tools. This documentation may also be found on the product CD under \support\tools.

DsAddSidHistory requires administrator privileges in the source and destination domains. Specifically, the caller of this API must be a member of the Domain Administrators group in the destination domain. A hard-coded check for this membership is performed. Also, the account provided in the *SrcDomainCreds* parameter must be a member of either the Administrators or Domain Administrators group in the source domain. If **NULL** is passed in

SrcDomainCreds, the caller of the API must be member of either the Administrators or Domain Administrators group in the source domain.

Domain and Trust Requirements

DsAddSidHistory requires that the destination domain be in Windows 2000 native mode or later, because only this domain type supports the **sIDHistory** attribute. The source domain may be either Windows NT 4.0 or Windows 2000, mixed or native mode. The source and destination domains must not be in the same forest. Windows NT 4.0 domains are by definition not in a forest. This inter-forest constraint ensures that duplicate SIDs, whether appearing as primary SIDs or **sIDHistory** values, are not created in the same forest.

DsAddSidHistory requires an external trust from the source domain to the destination domain in the cases listed in the following table.

Case	Description
The source domain is Windows 2000.	The source **sIDHistory** attribute, available only in Windows 2000 source domains, may be read only using LDAP, which requires this trust for integrity protection.
The source domain is Windows NT 4.0 and *SrcDomainCreds* is **NULL**.	The impersonation, required to support source domain operations using the caller's credentials, depends on this trust. Impersonation also requires that the destination domain controller has "Trusted for Delegation" enabled by default on domain controllers.

However, there is no trust required between the source and destination domains if the source domain is Windows NT 4.0 and *SrcDomainCreds* is not **NULL**.

Source Domain Controller Requirements

DsAddSidHistory requires that the domain controller, selected as the target for operations in the source domain, be the PDC in Windows NT 4.0 domains or the PDC Emulator in Windows 2000 domains. Source domain auditing is generated by way of write operations, therefore, the PDC is required in Windows NT 4.0 source domains, and the PDC-only restriction ensures that **DsAddSidHistory** audits are generated on a single computer. This reduces the need to review the audit logs of all DCs to monitor use of this operation.

Note

In Windows NT 4.0 source domains, the PDC (target of operations in the source domain) must be running Service Pack 4 (SP4) and later to ensure proper auditing support.

The following registry value must be created as a REG_DWORD value and set to 1 on the source domain controller for both Windows NT 4.0 and Windows 2000 source DCs.

```
HKEY_LOCAL_MACHINE
  System
     CurrentControlSet
        Control
           Lsa
              TcpipClientSupport
```

Setting this value enables RPC calls over the TCP transport. This is required because, by default, SAMRPC interfaces are remotable only on named pipes. Using named pipes results in a credential management system suitable for interactively logged-on users making networked calls, but is not flexible for a system process that makes network calls with user-supplied credentials. RPC over TCP is more suitable for that purpose. Setting this value does not diminish system security. If this value is created or changed, the source domain controller must be restarted for this setting to take effect.

A new local group, "<SrcDomainName>$$$", must be created in the source domain for auditing purposes.

Auditing

DsAddSidHistory operations are audited to ensure that both source and destination domain administrators are able to detect when this function has been run. Auditing is mandatory in both the source and destination domains. **DsAddSidHistory** verifies that the Audit Mode is on in each domain and that Account Management auditing of Success/Failure events is on. In the destination domain, a unique "Add Sid History" audit event is generated for each successful or failed **DsAddSidHistory** operation.

Unique "Add Sid History" audit events are not available on Windows NT 4.0 systems. To generate audit events that unambiguously reflect use of **DsAddSidHistory** against the source domain, it performs operations on a special group whose name is the unique identifier in the audit log. A local group, "<SrcDomainName>$$$", whose name is composed of the source domain NetBIOS name appended with three dollar signs ($) (ASCII code = 0x24 and Unicode = U+0024), must be created on the source domain controller prior to calling **DsAddSidHistory**. Each source user and global group that is a target of this operation is added to and then removed from the membership of this group. This generates Add Member and Delete Member audit events in the source domain, which can be monitored by searching for events that reference the group name.

Note

DsAddSidHistory operations on local groups in a Windows NT 4.0, or Windows 2000 mixed-mode source domain cannot be audited because local groups cannot be made members of another local group and therefore cannot be added to the special "<SrcDomainName>$$$" local group. This lack of auditing does not present a security issue to the source domain, because source domain resource access is not affected by this operation. Adding the SID of a source local group to a destination local group does not grant access to source resources, protected by that local group, to any additional users. Adding members to the destination local group does not grant them access to source resources. Added members are granted access only to servers in the destination domain migrated from the source domain, which may have resources protected by the source local group SID.

Data Transmission Security

DsAddSidHistory enforces the following security measures:

- Called from a Windows 2000 workstation, the caller's credentials are used to authenticate and privacy-protect the RPC call to the destination domain controller. If *SrcDomainCreds* is not **NULL**, both the workstation and the destination DC must support 128-bit encryption to privacy-protect the credentials. If 128-bit encryption is not available and *SrcDomainCreds* are provided, then the call must be made on the destination DC.
- The destination domain controller communicates with the source domain controller using either *SrcDomainCreds* or the caller's credentials to mutually authenticate and integrity-protect the read of the source account SID (using a SAM lookup) and **sIDHistory** (using an LDAP read).

Threat Models

The following table lists the potential threats associated with the **DsAddSidHistory** call and addresses the security measures pertinent to the particular threat.

Potential threat	**Security measure**
Man in the Middle Attack An unauthorized user intercepts the *lookup SID of source object* return call, replacing the source object SID with an arbitrary SID for insertion into a target object SIDhistory.	The *lookup SID of source object* is an authenticated RPC, using the caller's administrator credentials, with packet integrity message protection. This ensures that the return call cannot be modified without detection. The destination domain controller creates a unique "Add SID History" audit event that reflects the SID added to the destination account **sIDHistory**.
Trojan Source Domain An unauthorized user creates a "Trojan Horse" source domain on a private network that has the same domain SID and some of the same account SIDs as the legitimate source domain. The unauthorized user then attempts to run **DsAddSidHistory** in a destination domain to obtain the SID of a source account. This is done without the need for the real source Domain Administrator credentials and without leaving an audit trail in the real source domain. The unauthorized user's method for creating the Trojan Horse source domain could be one of the following: • Obtain a copy (BDC backup) of the source domain SAM.	Although there are many ways for an unauthorized user to retrieve or create a desired source object SID, the unauthorized user cannot use it to update an account's **sIDHistory** without being a member of the destination Domain Administrators group. Because the check, on the destination domain controller, for Domain Administrator membership is hard-coded, on the target DC, there is no method for doing a disk modification to change the access control data protecting this function. An attempt to clone a Trojan Horse source account is audited in the destination domain. This attack is mitigated by reserving membership in the Domain Administrators group for only highly trusted individuals.

Potential threat	Security measure
• Create a new domain, altering the domain SID on disk to match the legitimate source domain SID, then create enough users to instantiate an account with the desired SID. • Create a BDC replica. This requires source domain Administrator credentials. Then the unauthorized user takes the replica to a private network to implement the attack.	
On-disk Modification of SID History A sophisticated unauthorized user, with Domain Administrator credentials and with physical access to a DC in the destination domain, could modify an account **sIDHistory** value on disk.	This attempt is not enabled by use of **DsAddSidHistory**. This attack is mitigated by preventing physical access to domain controllers to all except highly trusted administrators.
Rogue Code Used to Remove Protections An unauthorized user or rogue administrator with physical access to the Directory Service code could create rogue code that: 1. Removes the check for membership in the Domain Administrators group in the code. 2. Changes the calls on the source domain controller that points the SID to a LookupSidFromName that is not audited. 3. Removes audit log calls.	Someone with physical access to the DS code and knowledgeable enough to create rogue code has the capability of arbitrarily modifying the **sIDHistory** attribute of an account. The **DsAddSidHistory** API does not increase this security risk.
Resources Vulnerable to Stolen SIDs If an unauthorized user has succeeded in using one of the methods described here to modify an account **sIDHistory**, and if the resource domains of interest trust the unauthorized user account domain, then the unauthorized user can get unauthorized access to the stolen SID's resources, potentially without leaving an audit trail in the account domain from which the SID was stolen.	Resource domain administrators protect their resources by setting up only those trust relationships that make sense from a security perspective. Use of **DsAddSidHistory** is restricted, in the trusted target domain, to members of the Domain Administrators group who already have broad permissions within the scope of their responsibilities.
Rogue Target Domain An unauthorized user creates a Windows 2000 domain	The unauthorized user requires Administrator credentials for the source domain in order to use

Potential threat	Security measure
with an account whose **sIDHistory** contains a SID that has been stolen from a source domain. The unauthorized user uses this account for unauthorized access to resources.	**DsAddSidHistory**, and leaves an audit trail on the source domain controller. The rogue target domain gains unauthorized access only in other domains that trust the rogue domain, which requires Administrator privileges in those resource domains.

Operational Constraints

This section describes the operational constraints of using the **DsAddSidHistory** function.

The SID of *SrcPrincipal* must not already exist in the destination forest, either as a primary account SID or in the **sIDHistory** of an account. The exception is that **DsAddSidHistory** does not generate an error when attempting to add a SID to a **sIDHistory** that contains an identical SID. This behavior enables **DsAddSidHistory** to be run multiple times with identical input, resulting in success and a consistent end state, for tool developer ease-of-use.

Note

Global Catalog replication latency may provide a window during which duplicate SIDs may be created. However, duplicate SIDs can be easily deleted by an administrator.

SrcPrincipal and *DstPrincipal* must be one of the following types:

- User
- Security Enabled Group, including:

  Local group Global group Domain local group (Windows 2000 native mode only) Universal group (Windows 2000 native mode only)

The object types of *SrcPrincipal* and *DstPrincipal* must match.

- If *SrcPrincipal* is a User, *DstPrincipal* must be a User.
- If *SrcPrincipal* is a Local or Domain Local Group, *DstPrincipal* must be a Domain Local Group.
- If *SrcPrincipal* is a Global or Universal Group, *DstPrincipal* must be a Global or Universal Group.

SrcPrincipal and *DstPrincipal* cannot be one of the following types: (**DsAddSidHistory** fails with an error in these cases)

- Computer (workstation or domain controller)
- Inter-domain trust
- Temporary duplicate account (virtually unused feature, a legacy of LANman)
- Accounts with Well Known SIDs. Well Known SIDs are identical in every domain; thus adding them to a **sIDHistory** would violate the SID uniqueness requirement of a Windows 2000 forest. Accounts with Well

Known SIDs include the following local groups:

Account operators Administrators Backup operators Guests Print operators Replicator Server operators Users

If *SrcPrincipal* has a well-known relative identifier (RID) and a domain specific prefix, that is, Domain Administrators, Domain Users, and Domain Computers, then *DstPrincipal* must possess the same well-known RID in order for **DsAddSidHistory** to succeed. Accounts with well-known RIDs include the following users and global groups:

- Administrator
- Guest
- Domain administrators
- Domain guests
- Domain users

Setting the Registry Value

The following procedure shows how to set the TcpipClientSupport registry value.

To Set the TcpipClientSupport Registry Value

1. Create the following registry value as a REG_DWORD value on the source domain controller and set its value to 1.

   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\TcpipClientSupport

2. Then, restart the source domain controller. This registry value causes the SAM to listen on TCP/IP. **DsAddSidHistory** will fail if this value is not set on the source domain controller.

Enabling Auditing of User/Group Management Events

The following procedure shows how to enable auditing of User/Group management events in a Windows 2000 or Windows Server 2003 source or destination domain.

To enable auditing of User/Group management events in a Windows 2000 or Windows Server 2003 source or destination domain

1. In the Active Directory Users and Computers MMC Snap-in, select the destination domain Domain Controllers container.
2. Right-click **Domain Controllers** and choose **Properties**.
3. Click the **Group Policy** tab.
4. Select the **Default Domain Controllers Policy** and click **Edit**.
5. Under **Computer Configuration\Windows Settings\Security Settings\Local Policies\Audit Policy**, double-click **Audit Account Management**.

6. In the **Audit Account Management** window, select both **Success** and **Failure** auditing. Policy updates take effect after a restart or after a refresh occurs.
7. Verify that auditing is enabled by viewing the effective audit policy in the Group Policy MMC Snap-in.

The following procedure shows how to enable auditing of User/Group management events in a Windows NT 4.0 domain.

To enable auditing of User/Group management events in a Windows NT 4.0 domain

1. In **User Manager for Domains**, click the **Policies** menu and select **Audit**.
2. Select **Audit These Events**.
3. For **User and Group Management**, check **Success and Failure**.

The following procedure shows how to enable auditing of User/Group management events in a Windows NT 4.0, Windows 2000, or Windows Server 2003 source domain.

To enable auditing of User/Group management events in a Windows NT 4.0, Windows 2000, or Windows Server 2003 source domain

1. In **User Manager for Domains**, click the **User** menu and select **New Local Group**.
2. Enter a group name composed of the source domain NetBIOS name appended with three dollar signs ($), for example, FABRIKAM$$$. The description field should indicate that this group is used to audit use of **DsAddSidHistory** or cloning operations. Ensure there are no members in the group. Click **OK**.

The **DsAddSidHistory** operation fails if source and destination auditing are not enabled as described here.

Set up Trust if Required

If one of the following is true, a trust must be established from the source domain to the destination domain (this must occur in a different forest):

- The source domain is Windows Server 2003.
- The source domain is Windows NT 4.0 and *SrcDomainCreds* is **NULL**.
