**[MS-LSAT]:**

**Local Security Authority (Translation Methods) Remote Protocol**

Intellectual Property Rights Notice for Open Specifications Documentation

* **Technical Documentation.** Microsoft publishes Open Specifications documentation (“this documentation”) for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.
* **Copyrights**. This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.
* **No Trade Secrets**. Microsoft does not claim any trade secret rights in this documentation.
* **Patents**. Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft [Open Specifications Promise](http://go.microsoft.com/fwlink/?LinkId=214445) or the [Microsoft Community Promise](http://go.microsoft.com/fwlink/?LinkId=214448). If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting [iplg@microsoft.com](mailto:iplg@microsoft.com).
* **License Programs**. To see all of the protocols in scope under a specific license program and the associated patents, visit the [Patent Map](https://msdn.microsoft.com/en-us/openspecifications/dn750984).
* **Trademarks**. The names of companies and products contained in this documentation might be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit [www.microsoft.com/trademarks](http://www.microsoft.com/trademarks).
* **Fictitious Names**. The example companies, organizations, products, domain names, email addresses, logos, people, places, and events that are depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

**Reservation of Rights**. All other rights are reserved, and this notice does not grant any rights other than as specifically described above, whether by implication, estoppel, or otherwise.

**Tools**. The Open Specifications documentation does not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments, you are free to take advantage of them. Certain Open Specifications documents are intended for use in conjunction with publicly available standards specifications and network programming art and, as such, assume that the reader either is familiar with the aforementioned material or has immediate access to it.

**Support.** For questions and support, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com).

**Revision Summary**

| Date | Revision History | Revision Class | Comments |
| --- | --- | --- | --- |
| 2/22/2007 | 0.01 | New | Version 0.01 release |
| 6/1/2007 | 1.0 | Major | Updated and revised the technical content. |
| 7/3/2007 | 2.0 | Major | Deleted type definition for SID\_IDENTIFIER\_AUTHORITY in favor of MS-DTYP. Major restructuring of Abstract Data Models section. |
| 7/20/2007 | 2.0.1 | Editorial | Changed language and formatting in the technical content. |
| 8/10/2007 | 2.0.2 | Editorial | Changed language and formatting in the technical content. |
| 9/28/2007 | 2.0.3 | Editorial | Changed language and formatting in the technical content. |
| 10/23/2007 | 2.1 | Minor | Clarified the meaning of the technical content. |
| 11/30/2007 | 3.0 | Major | Removed three types. |
| 1/25/2008 | 3.1 | Minor | Clarified the meaning of the technical content. |
| 3/14/2008 | 4.0 | Major | Updated and revised the technical content. |
| 5/16/2008 | 4.0.1 | Editorial | Changed language and formatting in the technical content. |
| 6/20/2008 | 4.1 | Minor | Clarified the meaning of the technical content. |
| 7/25/2008 | 5.0 | Major | Updated and revised the technical content. |
| 8/29/2008 | 6.0 | Major | Updated and revised the technical content. |
| 10/24/2008 | 7.0 | Major | Updated and revised the technical content. |
| 12/5/2008 | 8.0 | Major | Updated and revised the technical content. |
| 1/16/2009 | 9.0 | Major | Updated and revised the technical content. |
| 2/27/2009 | 10.0 | Major | Updated and revised the technical content. |
| 4/10/2009 | 10.0.1 | Editorial | Changed language and formatting in the technical content. |
| 5/22/2009 | 11.0 | Major | Updated and revised the technical content. |
| 7/2/2009 | 11.1 | Minor | Clarified the meaning of the technical content. |
| 8/14/2009 | 12.0 | Major | Updated and revised the technical content. |
| 9/25/2009 | 12.0.1 | Editorial | Changed language and formatting in the technical content. |
| 11/6/2009 | 13.0 | Major | Updated and revised the technical content. |
| 12/18/2009 | 14.0 | Major | Updated and revised the technical content. |
| 1/29/2010 | 14.1 | Minor | Clarified the meaning of the technical content. |
| 3/12/2010 | 14.1.1 | Editorial | Changed language and formatting in the technical content. |
| 4/23/2010 | 15.0 | Major | Updated and revised the technical content. |
| 6/4/2010 | 16.0 | Major | Updated and revised the technical content. |
| 7/16/2010 | 16.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 8/27/2010 | 16.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 10/8/2010 | 17.0 | Major | Updated and revised the technical content. |
| 11/19/2010 | 18.0 | Major | Updated and revised the technical content. |
| 1/7/2011 | 19.0 | Major | Updated and revised the technical content. |
| 2/11/2011 | 20.0 | Major | Updated and revised the technical content. |
| 3/25/2011 | 20.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 5/6/2011 | 21.0 | Major | Updated and revised the technical content. |
| 6/17/2011 | 22.0 | Major | Updated and revised the technical content. |
| 9/23/2011 | 22.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 12/16/2011 | 23.0 | Major | Updated and revised the technical content. |
| 3/30/2012 | 23.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 7/12/2012 | 23.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 10/25/2012 | 24.0 | Major | Updated and revised the technical content. |
| 1/31/2013 | 24.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 8/8/2013 | 25.0 | Major | Updated and revised the technical content. |
| 11/14/2013 | 25.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 2/13/2014 | 25.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 5/15/2014 | 25.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 6/30/2015 | 26.0 | Major | Significantly changed the technical content. |
| 10/16/2015 | 26.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 7/14/2016 | 27.0 | Major | Significantly changed the technical content. |
| 6/1/2017 | 28.0 | Major | Significantly changed the technical content. |
| 9/15/2017 | 29.0 | Major | Significantly changed the technical content. |
| 12/1/2017 | 29.0 | None | No changes to the meaning, language, or formatting of the technical content. |

Table of Contents

[1 Introduction 6](#_Toc499829107)

[1.1 Glossary 6](#_Toc499829108)

[1.2 References 9](#_Toc499829109)

[1.2.1 Normative References 9](#_Toc499829110)

[1.2.2 Informative References 10](#_Toc499829111)

[1.3 Overview 10](#_Toc499829112)

[1.4 Relationship to Other Protocols 11](#_Toc499829113)

[1.5 Prerequisites/Preconditions 11](#_Toc499829114)

[1.6 Applicability Statement 12](#_Toc499829115)

[1.7 Versioning and Capability Negotiation 12](#_Toc499829116)

[1.8 Vendor-Extensible Fields 12](#_Toc499829117)

[1.9 Standards Assignments 12](#_Toc499829118)

[2 Messages 13](#_Toc499829119)

[2.1 Transport 13](#_Toc499829120)

[2.2 Common Data Types 13](#_Toc499829121)

[2.2.1 LSAPR\_HANDLE 15](#_Toc499829122)

[2.2.2 STRING 15](#_Toc499829123)

[2.2.3 LSAPR\_ACL 15](#_Toc499829124)

[2.2.4 SECURITY\_DESCRIPTOR\_CONTROL 15](#_Toc499829125)

[2.2.5 LSAPR\_SECURITY\_DESCRIPTOR 15](#_Toc499829126)

[2.2.6 SECURITY\_IMPERSONATION\_LEVEL 16](#_Toc499829127)

[2.2.7 SECURITY\_CONTEXT\_TRACKING\_MODE 16](#_Toc499829128)

[2.2.8 SECURITY\_QUALITY\_OF\_SERVICE 16](#_Toc499829129)

[2.2.9 LSAPR\_OBJECT\_ATTRIBUTES 17](#_Toc499829130)

[2.2.10 ACCESS\_MASK 17](#_Toc499829131)

[2.2.11 LSAPR\_TRUST\_INFORMATION 17](#_Toc499829132)

[2.2.12 LSAPR\_REFERENCED\_DOMAIN\_LIST 18](#_Toc499829133)

[2.2.13 SID\_NAME\_USE 18](#_Toc499829134)

[2.2.14 LSA\_TRANSLATED\_SID 18](#_Toc499829135)

[2.2.15 LSAPR\_TRANSLATED\_SIDS 19](#_Toc499829136)

[2.2.16 LSAP\_LOOKUP\_LEVEL 19](#_Toc499829137)

[2.2.17 LSAPR\_SID\_INFORMATION 20](#_Toc499829138)

[2.2.18 LSAPR\_SID\_ENUM\_BUFFER 21](#_Toc499829139)

[2.2.19 LSAPR\_TRANSLATED\_NAME 21](#_Toc499829140)

[2.2.20 LSAPR\_TRANSLATED\_NAMES 21](#_Toc499829141)

[2.2.21 LSAPR\_TRANSLATED\_NAME\_EX 22](#_Toc499829142)

[2.2.22 LSAPR\_TRANSLATED\_NAMES\_EX 22](#_Toc499829143)

[2.2.23 LSAPR\_TRANSLATED\_SID\_EX 23](#_Toc499829144)

[2.2.24 LSAPR\_TRANSLATED\_SIDS\_EX 23](#_Toc499829145)

[2.2.25 LSAPR\_TRANSLATED\_SID\_EX2 24](#_Toc499829146)

[2.2.26 LSAPR\_TRANSLATED\_SIDS\_EX2 24](#_Toc499829147)

[2.3 Directory Service Schema Elements 25](#_Toc499829148)

[3 Protocol Details 26](#_Toc499829149)

[3.1 Server Details 26](#_Toc499829150)

[3.1.1 Abstract Data Model 26](#_Toc499829151)

[3.1.1.1 Database Views 26](#_Toc499829152)

[3.1.1.1.1 Predefined Translation Database and Corresponding View 27](#_Toc499829153)

[3.1.1.1.2 Configurable Translation Database and Corresponding View 30](#_Toc499829154)

[3.1.1.1.3 Builtin Domain Principal View 31](#_Toc499829155)

[3.1.1.1.4 Account Domain Principal View 32](#_Toc499829156)

[3.1.1.1.5 Account Domain Information View 34](#_Toc499829157)

[3.1.1.1.6 Account Domain View 35](#_Toc499829158)

[3.1.1.1.7 Forest Principal View 35](#_Toc499829159)

[3.1.1.1.8 Forest Information View 37](#_Toc499829160)

[3.1.1.1.9 Forest View 37](#_Toc499829161)

[3.1.1.2 Domain Database Information 37](#_Toc499829162)

[3.1.1.3 Trusted Domains and Forests Information 38](#_Toc499829163)

[3.1.2 Timers 38](#_Toc499829164)

[3.1.3 Initialization 39](#_Toc499829165)

[3.1.4 Message Processing Events and Sequencing Rules 39](#_Toc499829166)

[3.1.4.1 LsarOpenPolicy2 (Opnum 44) 42](#_Toc499829167)

[3.1.4.2 LsarOpenPolicy (Opnum 6) 42](#_Toc499829168)

[3.1.4.3 LsarClose (Opnum 0) 43](#_Toc499829169)

[3.1.4.4 LsarGetUserName (Opnum 45) 43](#_Toc499829170)

[3.1.4.5 LsarLookupNames4 (Opnum 77) 44](#_Toc499829171)

[3.1.4.6 LsarLookupNames3 (Opnum 68) 48](#_Toc499829172)

[3.1.4.7 LsarLookupNames2 (Opnum 58) 49](#_Toc499829173)

[3.1.4.8 LsarLookupNames (Opnum 14) 51](#_Toc499829174)

[3.1.4.9 LsarLookupSids3 (Opnum 76) 52](#_Toc499829175)

[3.1.4.10 LsarLookupSids2 (Opnum 57) 54](#_Toc499829176)

[3.1.4.11 LsarLookupSids (Opnum 15) 56](#_Toc499829177)

[3.1.5 Timer Events 57](#_Toc499829178)

[3.1.6 Other Local Events 57](#_Toc499829179)

[3.2 Client Details 57](#_Toc499829180)

[4 Protocol Example 59](#_Toc499829181)

[5 Security 62](#_Toc499829182)

[5.1 Security Considerations for Implementers 62](#_Toc499829183)

[5.2 Index of Security Parameters 62](#_Toc499829184)

[6 Appendix A: Full IDL 63](#_Toc499829185)

[7 Appendix B: Product Behavior 72](#_Toc499829186)

[8 Change Tracking 80](#_Toc499829187)

[9 Index 81](#_Toc499829188)

# Introduction

The Local Security Authority (Translation Methods) Remote Protocol is implemented in Windows products to translate identifiers for [**security principals**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) between human-readable and machine-readable forms. This translation can be used in scenarios such as human management of resource access.

Sections 1.5, 1.8, 1.9, 2, and 3 of this specification are normative. All other sections and examples in this specification are informative.

## Glossary

This document uses the following terms:

**access control list (ACL)**: A list of access control entries (ACEs) that collectively describe the security rules for authorizing access to some resource; for example, an object or set of objects.

**ACID**: A term that refers to the four properties that any database system must achieve in order to be considered transactional: Atomicity, Consistency, Isolation, and Durability [GRAY].

**Active Directory**: A general-purpose network directory service. [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) also refers to the Windows implementation of a directory service. [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) stores information about a variety of objects in the network. User accounts, computer accounts, groups, and all related credential information used by the Windows implementation of Kerberos are stored in [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90). [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) is either deployed as Active Directory Domain Services (AD DS) or Active Directory Lightweight Directory Services (AD LDS). [[MS-ADTS]](%5bMS-ADTS%5d.pdf#Section_d243592709994c628c6d13ba31a52e1a) describes both forms. For more information, see [[MS-AUTHSOD]](%5bMS-AUTHSOD%5d.pdf#Section_953d700a57cb4cf7b0c3a64f34581cc9) section 1.1.1.5.2, Lightweight Directory Access Protocol (LDAP) versions 2 and 3, Kerberos, and DNS.

**discretionary access control list (DACL)**: An [**access control list (ACL)**](#gt_9f92aa05-dd0a-45f2-88d6-89f1fb654395) that is controlled by the owner of an object and that specifies the access particular users or groups can have to the object.

**DNS name**: A fully qualified domain name (FQDN).

**domain**: A set of users and computers sharing a common namespace and management infrastructure. At least one computer member of the set must act as a [**domain controller (DC)**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) and host a member list that identifies all members of the domain, as well as optionally hosting the [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) service. The domain controller provides authentication of members, creating a unit of trust for its members. Each domain has an identifier that is shared among its members. For more information, see [MS-AUTHSOD] section 1.1.1.5 and [MS-ADTS].

**domain controller (DC)**: The service, running on a server, that implements [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90), or the server hosting this service. The service hosts the data store for objects and interoperates with other [**DCs**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) to ensure that a local change to an object replicates correctly across all [**DCs**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd). When [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) is operating as Active Directory Domain Services (AD DS), the [**DC**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) contains full NC replicas of the configuration naming context (config NC), schema naming context (schema NC), and one of the [**domain NCs**](#gt_40a58fa4-953e-4cf3-96c8-57dba60237ef) in its [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62). If the AD DS [**DC**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) is a global catalog server (GC server), it contains partial NC replicas of the remaining [**domain NCs**](#gt_40a58fa4-953e-4cf3-96c8-57dba60237ef) in its [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62). For more information, see [MS-AUTHSOD] section 1.1.1.5.2 and [MS-ADTS]. When [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) is operating as Active Directory Lightweight Directory Services (AD LDS), several AD LDS [**DCs**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) can run on one server. When [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) is operating as AD DS, only one AD DS [**DC**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) can run on one server. However, several AD LDS [**DCs**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) can coexist with one AD DS [**DC**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) on one server. The AD LDS [**DC**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) contains full NC replicas of the config NC and the schema NC in its [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62). The domain controller is the server side of Authentication Protocol Domain Support [[MS-APDS]](%5bMS-APDS%5d.pdf#Section_dd444344fd7e430eb3137e95ab9c338e).

**domain database**: A database where security principal information is stored. This database is the directory service [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) in the case of a [**domain controller (DC)**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd). On a machine that is not a [**DC**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd), this database is a local database, manageable through the Security Accounts Manager Remote Protocol, as specified in [[MS-SAMR]](%5bMS-SAMR%5d.pdf#Section_4df07fab1bbc452f8e927853a3c7e380).

**domain member (member machine)**: A machine that is joined to a domain by sharing a secret between the machine and the domain.

**domain name**: A domain name or a NetBIOS name that identifies a [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca).

**domain naming context (domain NC)**: A naming context (NC) whose replicas are able to contain [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) objects. No other NC replica can contain [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) objects. The distinguished name (DN) of a [**domain NC**](#gt_40a58fa4-953e-4cf3-96c8-57dba60237ef) takes the form "dc=n1,dc=n2, ... dc=nk" where each "ni" satisfies the syntactic requirements of a DNS name component. For more information, see [[RFC1034]](https://go.microsoft.com/fwlink/?LinkId=90263). Such a DN corresponds to the domain naming service name: "n1.n2. ... .nk". This is the domain naming service name of the [**domain NC**](#gt_40a58fa4-953e-4cf3-96c8-57dba60237ef). [**Domain NCs**](#gt_40a58fa4-953e-4cf3-96c8-57dba60237ef) appear in the global catalog (GC). A [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62) has one or more [**domain NCs**](#gt_40a58fa4-953e-4cf3-96c8-57dba60237ef). The root of a [**domain NC**](#gt_40a58fa4-953e-4cf3-96c8-57dba60237ef) is an object of class domainDns.

**forest**: In the [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) directory service, a [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62) is a set of naming contexts (NCs) consisting of one schema NC, one config NC, and one or more [**domain NCs**](#gt_40a58fa4-953e-4cf3-96c8-57dba60237ef). Because a set of NCs can be arranged into a tree structure, a [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62) is also a set of one or several trees of NCs.

**forest trust**: A type of [**trust**](#gt_5ee032d0-d944-4acb-bbb5-b1cfc7df6db6) where the trusted party is a [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62), which means that all [**domains**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) in that [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62) are trusted.

**Local Security Authority (LSA)**: A protected subsystem that authenticates and logs users onto the local system. [**LSA**](#gt_9e5f2104-d6df-4ae7-8a5c-6bd14a0da8fa) also maintains information about all aspects of local security on a system, collectively known as the local security policy of the system.

**Network Data Representation (NDR)**: A specification that defines a mapping from Interface Definition Language (IDL) data types onto octet streams. [**NDR**](#gt_9ebf9540-2c31-43bc-bc56-4a932faabf2d) also refers to the runtime environment that implements the mapping facilities (for example, data provided to [**NDR**](#gt_9ebf9540-2c31-43bc-bc56-4a932faabf2d)). For more information, see [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) and [[C706]](https://go.microsoft.com/fwlink/?LinkId=89824) section 14.

**opnum**: An operation number or numeric identifier that is used to identify a specific [**remote procedure call (RPC)**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) method or a method in an interface. For more information, see [C706] section 12.5.2.12 or [MS-RPCE].

**relative identifier (RID)**: The last item in the series of SubAuthority values in a [**security identifier (SID)**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) [[SIDD]](https://go.microsoft.com/fwlink/?LinkId=90516). It distinguishes one account or group from all other accounts and groups in the domain. No two accounts or groups in any domain share the same RID.

**remote procedure call (RPC)**: A communication protocol used primarily between client and server. The term has three definitions that are often used interchangeably: a runtime environment providing for communication facilities between computers (the RPC runtime); a set of request-and-response message exchanges between computers (the RPC exchange); and the single message from an RPC exchange (the RPC message). For more information, see [C706].

**root domain**: The unique [**domain naming contexts (domain NCs)**](#gt_40a58fa4-953e-4cf3-96c8-57dba60237ef) of an Active Directory [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62) that is the parent of the [**forest's**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62) config NC. The config NC's relative distinguished name (RDN) is "cn=Configuration" relative to the root object of the root domain. The root domain is the domain that is created first in a [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62).

**RPC client**: A computer on the network that sends messages using remote procedure call (RPC) as its transport, waits for responses, and is the initiator in an RPC exchange.

**RPC dynamic endpoint**: A network-specific server address that is requested and assigned at run time, as described in [C706].

**RPC endpoint**: A network-specific address of a server process for remote procedure calls (RPCs). The actual name of the RPC endpoint depends on the RPC protocol sequence being used. For example, for the NCACN\_IP\_TCP RPC protocol sequence an RPC endpoint might be TCP port 1025. For more information, see [C706].

**RPC protocol sequence**: A character string that represents a valid combination of a [**remote procedure call (RPC)**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) protocol, a network layer protocol, and a transport layer protocol, as described in [C706] and [MS-RPCE].

**RPC server**: A computer on the network that waits for messages, processes them when they arrive, and sends responses using RPC as its transport acts as the responder during a remote procedure call (RPC) exchange.

**RPC transport**: The underlying network services used by the remote procedure call (RPC) runtime for communications between network nodes. For more information, see [C706] section 2.

**security identifier (SID)**: An identifier for [**security principals**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) that is used to identify an account or a group. Conceptually, the [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) is composed of an account authority portion (typically a [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca)) and a smaller integer representing an identity relative to the account authority, termed the [**relative identifier (RID)**](#gt_df3d0b61-56cd-4dac-9402-982f1fedc41c). The [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) format is specified in [[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) section 2.4.2; a string representation of [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) is specified in [MS-DTYP] section 2.4.2 and [[MS-AZOD]](%5bMS-AZOD%5d.pdf#Section_5a0a0a3ec7a742e1b5f2cc8d8bd9739e) section 1.1.1.2.

**security principal**: A unique entity, also referred to as a principal, that can be authenticated by [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90). It frequently corresponds to a human user, but also can be a service that offers a resource to other security principals. Other security principals might be a group, which is a set of principals. Groups are supported by [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90).

**Server Message Block (SMB)**: A protocol that is used to request file and print services from server systems over a network. The SMB protocol extends the CIFS protocol with additional security, file, and disk management support. For more information, see [[CIFS]](https://go.microsoft.com/fwlink/?LinkId=89836) and [[MS-SMB]](%5bMS-SMB%5d.pdf#Section_f210069c70864dc2885e861d837df688).

**trust**: To accept another authority's statements for the purposes of authentication and authorization, especially in the case of a relationship between two domains. If [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) A trusts [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) B, [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) A accepts [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) B's authentication and authorization statements for principals represented by security principal objects in [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) B; for example, the list of groups to which a particular user belongs. As a noun, a [**trust**](#gt_5ee032d0-d944-4acb-bbb5-b1cfc7df6db6) is the relationship between two [**domains**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) described in the previous sentence.

**trust attributes**: A collection of attributes that define different characteristics of a trust within a domain or a forest.

**trusted domain**: A domain that is trusted to make authentication decisions for security principals in that domain.

**trusted forest**: A forest that is trusted to make authentication statements for security principals in that forest. Assuming forest A trusts forest B, all domains belonging to forest A will trust all domains in forest B, subject to policy configuration.

**universally unique identifier (UUID)**: A 128-bit value. UUIDs can be used for multiple purposes, from tagging objects with an extremely short lifetime, to reliably identifying very persistent objects in cross-process communication such as client and server interfaces, manager entry-point vectors, and [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) objects. UUIDs are highly likely to be unique. UUIDs are also known as globally unique identifiers (GUIDs) and these terms are used interchangeably in the Microsoft protocol technical documents (TDs). Interchanging the usage of these terms does not imply or require a specific algorithm or mechanism to generate the UUID. Specifically, the use of this term does not imply or require that the algorithms described in [[RFC4122]](https://go.microsoft.com/fwlink/?LinkId=90460) or [C706] must be used for generating the UUID.

**user principal name (UPN)**: A user account name (sometimes referred to as the user logon name) and a domain name that identifies the domain in which the user account is located. This is the standard usage for logging on to a Windows domain. The format is: someone@example.com (in the form of an email address). In [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90), the userPrincipalName attribute of the account object, as described in [MS-ADTS].

**MAY, SHOULD, MUST, SHOULD NOT, MUST NOT:** These terms (in all caps) are used as defined in [[RFC2119]](https://go.microsoft.com/fwlink/?LinkId=90317). All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

## References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the [Errata](https://go.microsoft.com/fwlink/?linkid=850906).

### Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com). We will assist you in finding the relevant information.

[C706] The Open Group, "DCE 1.1: Remote Procedure Call", C706, August 1997, [https://www2.opengroup.org/ogsys/catalog/c706](https://go.microsoft.com/fwlink/?LinkId=89824)

[GRAY] Gray, J., and Reuter, A., "Transaction Processing: Concepts and Techniques", The Morgan Kaufmann Series in Data Management Systems, San Francisco: Morgan Kaufmann Publishers, 1992, Hardcover ISBN: 9781558601901.

[MS-ADA1] Microsoft Corporation, "[Active Directory Schema Attributes A-L](%5bMS-ADA1%5d.pdf#Section_19528560f41e4623a406dabcfff0660f)".

[MS-ADA2] Microsoft Corporation, "[Active Directory Schema Attributes M](%5bMS-ADA2%5d.pdf#Section_e20ebc4e528540bab3bdffcb81c2783e)".

[MS-ADA3] Microsoft Corporation, "[Active Directory Schema Attributes N-Z](%5bMS-ADA3%5d.pdf#Section_4517e8353ee644d4bb95a94b6966bfb0)".

[MS-ADSC] Microsoft Corporation, "[Active Directory Schema Classes](%5bMS-ADSC%5d.pdf#Section_9abb5e97123d4da99557b353ab79b830)".

[MS-ADTS] Microsoft Corporation, "[Active Directory Technical Specification](%5bMS-ADTS%5d.pdf#Section_d243592709994c628c6d13ba31a52e1a)".

[MS-DRSR] Microsoft Corporation, "[Directory Replication Service (DRS) Remote Protocol](%5bMS-DRSR%5d.pdf#Section_f977faaa673e4f66b9bf48c640241d47)".

[MS-DTYP] Microsoft Corporation, "[Windows Data Types](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2)".

[MS-ERREF] Microsoft Corporation, "[Windows Error Codes](%5bMS-ERREF%5d.pdf#Section_1bc92ddfb79e413cbbaa99a5281a6c90)".

[MS-LSAD] Microsoft Corporation, "[Local Security Authority (Domain Policy) Remote Protocol](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc)".

[MS-NRPC] Microsoft Corporation, "[Netlogon Remote Protocol](%5bMS-NRPC%5d.pdf#Section_ff8f970f3e3740f7bd4baf7336e4792f)".

[MS-RPCE] Microsoft Corporation, "[Remote Procedure Call Protocol Extensions](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15)".

[MS-SAMR] Microsoft Corporation, "[Security Account Manager (SAM) Remote Protocol (Client-to-Server)](%5bMS-SAMR%5d.pdf#Section_4df07fab1bbc452f8e927853a3c7e380)".

[MS-SCMR] Microsoft Corporation, "[Service Control Manager Remote Protocol](%5bMS-SCMR%5d.pdf#Section_705b624a13de43ccb8a299573da3635f)".

[MSKB-3149090] Microsoft Corporation, "MS16-047: Description of the security update for SAM and LSAD remote protocols", April 2016, [https://support.microsoft.com/en-us/kb/3149090](https://go.microsoft.com/fwlink/?LinkId=786963)

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, [http://www.rfc-editor.org/rfc/rfc2119.txt](https://go.microsoft.com/fwlink/?LinkId=90317)

### Informative References

[MS-ADOD] Microsoft Corporation, "[Active Directory Protocols Overview](%5bMS-ADOD%5d.pdf#Section_5ff67bf4c14548cb89cd4f5482d94664)".

[MS-SMB] Microsoft Corporation, "[Server Message Block (SMB) Protocol](%5bMS-SMB%5d.pdf#Section_f210069c70864dc2885e861d837df688)".

[MSDN-RPCDB] Microsoft Corporation, "The RPC Name Service Database", [http://msdn.microsoft.com/en-us/library/aa378865.aspx](https://go.microsoft.com/fwlink/?LinkId=90076)

[MSFT-LSA-IDL] Microsoft Corporation, "Local Security Authority Merged IDL File", December 2015, [http://www.microsoft.com/downloads/details.aspx?displaylang=en&familyID=7700ad04-866b-447a-9e08-21dbda94460f](https://go.microsoft.com/fwlink/?LinkId=150268)

## Overview

The purpose of this protocol is to translate human-readable names to [**security identifiers (SIDs)**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d), as specified in [[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) section 2.4.2, and vice versa. The syntax of human-readable names is specified in section [3.1.4.5](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361). For example, this protocol can be used to translate "corp\John" to "S-1-5-21-397955417-626881126-188441444-1555", and vice versa.

This translation is needed for different scenarios, such as managing access to resources. In Windows, access to resources is controlled by a security descriptor attached to the resource. This security descriptor contains a list of SIDs indicating the [**security principals**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) and the kind of access allowed or denied for those principals. In order for humans to manage access to resources, translation occurs between SIDs (persisted within security descriptors) and human-readable identifiers (in the user interface).

This protocol is intended for use between any two machines, most frequently between a [**domain member**](#gt_6234a38c-ed1b-4c69-969f-6e6479566f65) and a [**domain controller (DC)**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) for that [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca). This protocol can also be used between domain controllers for [**trusting**](#gt_5ee032d0-d944-4acb-bbb5-b1cfc7df6db6) domains or [**forests**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62).

This protocol is a simple request/response-based [**remote procedure call (RPC)**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) protocol. There are no long-lived sessions, although clients are free to cache the RPC connection and reuse it over time. A sample sequence of requests and responses is shown in section [4](#Section_b1c74144daef4246803773922564e3d3).

## Relationship to Other Protocols

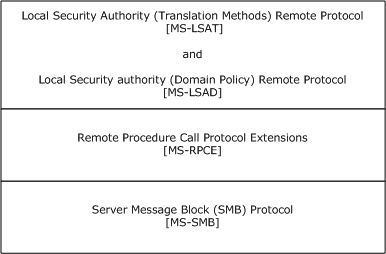


Figure 1: The relationships among the LSAT, LSAD, RPC, and SMB protocols

The Local Security Authority (Translation Methods) Remote Protocol is composed of a subset of [**opnums**](#gt_e127848e-c66d-427d-b3aa-9f904fa4ada7) in an interface that also includes the Local Security Authority (Domain Policy) Remote Protocol, as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc).

The Local Security Authority (Translation Methods) Remote Protocol is dependent upon [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331), which is used to communicate between [**domain members**](#gt_6234a38c-ed1b-4c69-969f-6e6479566f65) and [**domain controllers**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd).

This protocol depends on the [**Server Message Block (SMB)**](#gt_09dbec39-5e75-4d9a-babf-1c9f1d499625) Protocol [[MS-SMB]](%5bMS-SMB%5d.pdf#Section_f210069c70864dc2885e861d837df688) and the TCP/IP protocol for sending messages on the wire.

Some messages in this protocol depend on the Netlogon Remote Protocol, as specified in [[MS-NRPC]](%5bMS-NRPC%5d.pdf#Section_ff8f970f3e3740f7bd4baf7336e4792f), having successfully negotiated security parameters to be used. For more information on the messages that have this requirement, see section [3.1.4](#Section_d6b1c6d4ff5147dcb3b4af10e917ddd9).

The Directory Replication Service (DRS) Remote Protocol, as specified in [[MS-DRSR]](%5bMS-DRSR%5d.pdf#Section_f977faaa673e4f66b9bf48c640241d47), exposes functionality similar to what this protocol achieves, but serves a different purpose. The DRS Remote Protocol can translate names in different syntaxes, but lacks the functionality to translate names from different [**forests**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62). An application uses one or both protocols, depending on the syntax of names for which the application requires translation.

The Security Account Manager (SAM) Remote Protocol (Client-to-Server), as specified in [[MS-SAMR]](%5bMS-SAMR%5d.pdf#Section_4df07fab1bbc452f8e927853a3c7e380), exposes similar functionality for [**security principals**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) that are local to a server.

The Active Directory Technical Specification [[MS-ADTS]](%5bMS-ADTS%5d.pdf#Section_d243592709994c628c6d13ba31a52e1a) discusses [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90), which is used by this protocol.

## Prerequisites/Preconditions

This protocol has prerequisites, specified in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15), that are common to protocols that depend on [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331).

As previously noted, some messages in this protocol depend on the Netlogon Remote Protocol, as specified in [[MS-NRPC]](%5bMS-NRPC%5d.pdf#Section_ff8f970f3e3740f7bd4baf7336e4792f), having successfully negotiated security parameters to be used. For more information on the messages that have this requirement, see section [3.1.4](#Section_d6b1c6d4ff5147dcb3b4af10e917ddd9).

It is assumed that the [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553) has access to the distributed abstract databases specified in section [3.1.1](#Section_fa772a2455d44911b3d5eda0bd2ec4e1) that are required to perform the translations requested using this protocol. It is also assumed that the RPC server has created the required views, as specified in section 3.1.1.

## Applicability Statement

This protocol is applicable whenever there is a need to translate the human-readable names of [**security principals**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409), with syntaxes described in section [3.1.4.5](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361), to security principal [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d), or vice versa. This applicability includes, but is not limited to, an end-user administration program that displays and controls who can access such resources as files.

To understand the conditions under which this protocol is preferred over other protocols, see section [1.4](#Section_540f52f169ee43168d075359d4b93cab).

## Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

* Supported Transports: The protocol uses [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) over [**SMB**](#gt_09dbec39-5e75-4d9a-babf-1c9f1d499625) and TCP/IP, as specified in section [2.1](#Section_f25df96b48c648848cf01418e3148c66).
* Protocol Version: This protocol's RPC interface has a single version number, but that interface has been extended by adding methods at the end. The use of these methods is specified in section [3.1](#Section_3d42d9643d7c492e8664800bfed4888b).
* Structure Version: The [LSAPR\_ACL (section 2.2.3)](#Section_e1a273531eb54cabb88e43d18558daf2) and [LSAPR\_SECURITY\_DESCRIPTOR (section 2.2.5)](#Section_8494008f0bfb45b8bb6ce32dd7f18e3d) structures are versioned using the first field in the structure. The structure version is not used in this protocol.
* Capabilities: The [**RPC client**](#gt_e5a7be6b-98db-4e8d-8116-5893f43ab48b) passes its capabilities in the appropriate parameter when possible, as specified in sections [3.1.4.5](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361), [3.1.4.6](#Section_b75eaac5e4d14fc08dae61d838b38701), [3.1.4.7](#Section_43254e9f32e248caa61de8decf41721e), [3.1.4.9](#Section_678dd7ed611b4d948f5ca06c76ae5c7e), and [3.1.4.10](#Section_a09480f4145e4b4ca0275b313db61833).

The [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553) takes actions according to the capabilities parameter of an RPC message, if present, as specified in the previous paragraph.

## Vendor-Extensible Fields

This protocol uses NTSTATUS values as specified in [[MS-ERREF]](%5bMS-ERREF%5d.pdf#Section_1bc92ddfb79e413cbbaa99a5281a6c90) section 2.3. Vendors are free to choose their own values for this field, provided that the C bit (0x20000000) is set, which indicates that it is a customer code.

## Standards Assignments

This protocol has no standards assignments. It uses private allocations for the [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) interface [**universally unique identifier (UUID)**](#gt_c4813fc3-b2e5-4aa3-bde7-421d950d68d3) (12345778-1234-ABCD-EF00-0123456789AB) and the [**RPC endpoint**](#gt_8be6a1fb-bc3c-4ee3-8018-c236f351222a) (\PIPE\lsarpc).

# Messages

## Transport

This protocol uses the following [**RPC protocol sequences**](#gt_0c171cc7-e9c4-41b6-95a9-536db0042c7a):

* [**SMB**](#gt_09dbec39-5e75-4d9a-babf-1c9f1d499625)
* TCP/IP[<1>](#Appendix_A_1" \o "Product behavior note 1)

This protocol MUST use [**RPC dynamic endpoints**](#gt_dee69163-4767-4a5e-b58b-683757300d36), as specified in [[C706]](https://go.microsoft.com/fwlink/?LinkId=89824) section 4, when using [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) over TCP/IP.

This protocol MUST use "\PIPE\lsarpc" as the [**RPC endpoint**](#gt_8be6a1fb-bc3c-4ee3-8018-c236f351222a) when using RPC over SMB.[<2>](#Appendix_A_2" \o "Product behavior note 2)

[**RPC clients**](#gt_e5a7be6b-98db-4e8d-8116-5893f43ab48b) for this protocol MUST use RPC over SMB for the [LsarOpenPolicy2](#Section_40a316ae1bae4aa192cd650cdaae1863), [LsarOpenPolicy](#Section_534c8a461be740659f87e60a1447ae45), [LsarClose](#Section_be232cec04a9439ab718931c17c3ff0f), [LsarGetUserName](#Section_4e3f0a3f86ae4d50aa40ec25013a89e3), [LsarLookupNames](#Section_65d18faa0cb240eea94a2140212f4ec4), [LsarLookupNames2](#Section_43254e9f32e248caa61de8decf41721e), [LsarLookupNames3](#Section_b75eaac5e4d14fc08dae61d838b38701), [LsarLookupSids](#Section_eb7ac899e697488393de1e60c7720c02), and [LsarLookupSids2](#Section_a09480f4145e4b4ca0275b313db61833) methods. RPC clients MUST use RPC over TCP/IP for the [LsarLookupNames4](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361) and [LsarLookupSids3](#Section_678dd7ed611b4d948f5ca06c76ae5c7e) methods.[<3>](#Appendix_A_3" \o "Product behavior note 3)

The server SHOULD[<4>](#Appendix_A_4" \o "Product behavior note 4) reject calls that do not use an authentication level of RPC\_C\_AUTHN\_LEVEL\_NONE, RPC\_C\_AUTHN\_LEVEL\_PKT\_INTEGRITY, or RPC\_C\_AUTHN\_LEVEL\_PKT\_PRIVACY ([[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) section 2.2.1.1.8).

This protocol MUST use the [**UUID**](#gt_c4813fc3-b2e5-4aa3-bde7-421d950d68d3) and version number as follows:

* UUID: See section [1.9](#Section_f14d69a5e9464608a430d32f494fb3d9).
* Version number: 0.0.

Security settings used in this protocol vary depending on the role of the RPC client and [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553), the method being used, and the specific parameters being used. Therefore, security settings are discussed in the message processing sections for each message.

## Common Data Types

In addition to [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) base types and definitions specified in [[C706]](https://go.microsoft.com/fwlink/?LinkId=89824) and [[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2), additional data types are defined in this specification.

Some fields of some data types were introduced into the protocol but not used. These data types and fields are not to be confused with those that were introduced for extensibility. These fields MUST NOT be used for extensibility. Such fields are described as follows:

"This field MUST be ignored. The content is unspecified, and no requirements are placed on its value since it is never used."

The server SHOULD place no constraints on the value present in such fields unless otherwise stated.

This protocol MUST indicate to the RPC runtime that it is to support both the [**NDR**](#gt_9ebf9540-2c31-43bc-bc56-4a932faabf2d) and NDR64 transfer syntaxes and provide a negotiation mechanism for determining which transfer syntax will be used, as specified in [C706] section 12 and in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) section 3.3.1.5.6.

The following table summarizes the types that are defined in this specification.

**Note**  NTSTATUS, RPC\_SID, and RPC\_UNICODE\_STRING are defined in [MS-DTYP] sections 2.2.38, 2.4.2.3, and 2.3.10, respectively.

| Type (section) | Summary |
| --- | --- |
| [ACCESS\_MASK (section 2.2.10)](#Section_f41e22dbf23e4ef99161f769cb9021dd) | Defines the access rights to grant an object. |
| [LSA\_TRANSLATED\_SID (section 2.2.14)](#Section_d067a76efd924e3691e78605814414cc) | Contains machine-readable information about a [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409). |
| [LSAP\_LOOKUP\_LEVEL (section 2.2.16)](#Section_9d1166ccbcfd4e22a8acf55eae57c99f) | Defines different scopes for searches during translation. |
| [LSAPR\_ACL (section 2.2.3)](#Section_e1a273531eb54cabb88e43d18558daf2) | Defines the header of an [**access control list**](#gt_9f92aa05-dd0a-45f2-88d6-89f1fb654395). |
| [LSAPR\_HANDLE (section 2.2.1)](#Section_cec0486823014e7b8539367adc3a2cc5) | Defines a context handle. |
| [LSAPR\_OBJECT\_ATTRIBUTES (section 2.2.9)](#Section_11b18f088f1940bd9794fc6bae3d7b05) | Specifies an object and its handle's properties. |
| [LSAPR\_REFERENCED\_DOMAIN\_LIST (section 2.2.12)](#Section_3a52af31247a4b0891a01d46b2cc49b2) | Contains information about the [**domains**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) referenced in a lookup operation. |
| [LSAPR\_SECURITY\_DESCRIPTOR (section 2.2.5)](#Section_8494008f0bfb45b8bb6ce32dd7f18e3d) | Defines an object's security descriptor. |
| [LSAPR\_SID\_ENUM\_BUFFER (section 2.2.18)](#Section_1ffb61f0a4fe4487858dfb709d605855) | Defines a set of [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d). |
| [LSAPR\_SID\_INFORMATION (section 2.2.17)](#Section_f04a771bc018409881b52a819f9b5db8) | Contains a SID value. |
| [LSAPR\_TRANSLATED\_NAME (section 2.2.19)](#Section_52e1ccc1b57b4c02b35fbd64913ce99b) | Contains human-readable name information about a security principal. |
| [LSAPR\_TRANSLATED\_NAME\_EX (section 2.2.21)](#Section_6a9131acf1a74b968917a289a2c48de1) | Contains human-readable name information about a security principal. |
| [LSAPR\_TRANSLATED\_NAMES (section 2.2.20)](#Section_ff977eb9563a4353a95f640e7ee16356) | Defines a set of translated names. |
| [LSAPR\_TRANSLATED\_NAMES\_EX (section 2.2.22)](#Section_5f2bd56ba16240339c2f7323fcf09e8c) | Contains a set of translated names. |
| [LSAPR\_TRANSLATED\_SID\_EX (section 2.2.23)](#Section_274ee4e386b5445099e9f1678b9e96f5) | Contains machine-readable identifier information about a security principal. |
| [LSAPR\_TRANSLATED\_SID\_EX2 (section 2.2.25)](#Section_e397a7ac842a4d4487ef170392dee522) | Contains machine-readable identifier information about a security principal. |
| [LSAPR\_TRANSLATED\_SIDS (section 2.2.15)](#Section_8c3f03453cdd4ccbba6f8aac5e0da896) | Defines a set of translated SIDs. |
| [LSAPR\_TRANSLATED\_SIDS\_EX (section 2.2.24)](#Section_19250ed6447043259f85adb49db720fd) | Contains a set of translated SIDs. |
| [LSAPR\_TRANSLATED\_SIDS\_EX2 (section 2.2.26)](#Section_f55a62516071460089e77a7692e6aff4) | Contains a set of translated SIDs. |
| [LSAPR\_TRUST\_INFORMATION (section 2.2.11)](#Section_b0f34b28b5da44aaa60799c09e6526e1) | Contains information about a domain. |
| [SECURITY\_CONTEXT\_TRACKING\_MODE (section 2.2.7)](#Section_0ce7098212074d879fefc2ee7a318f52) | Specifies whether user security attributes are updated on a server when they change on the client. |
| [SECURITY\_DESCRIPTOR\_CONTROL (section 2.2.4)](#Section_2facaeb2b75c470db907d5e52900550f) | Qualifies the meaning of a security descriptor or its components. |
| [SECURITY\_IMPERSONATION\_LEVEL (section 2.2.6)](#Section_4797d0a17b714a22a5a08e2059aec239) | Specifies security impersonation levels. |
| [SECURITY\_QUALITY\_OF\_SERVICE (section 2.2.8)](#Section_ee3cadcdff914f45b80ced820db9c1dd) | Defines information that is used to support client impersonation. |
| [SID\_NAME\_USE (section 2.2.13)](#Section_171e9a878e014bd8a35e3468128c8fc4) | Contains values that specify the type of an account. |
| [STRING (section 2.2.2)](#Section_d876ef8225c1423f93cb9d97547f2a4b) | Defines an ANSI string along with the number of characters in the string. |

The following citation contains a timeline of when each data type was introduced.[<5>](#Appendix_A_5" \o "Product behavior note 5)

### LSAPR\_HANDLE

The LSAPR\_HANDLE type defines a context handle, as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 2.2.2.1.

This type is declared as follows:

1. typedef [context\_handle] void\* LSAPR\_HANDLE;

### STRING

The STRING structure defines a string along with the number of characters in the string, as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 2.2.3.1.

1. typedef struct \_STRING {
2. unsigned short Length;
3. unsigned short MaximumLength;
4. [size\_is(MaximumLength), length\_is(Length)]
5. char\* Buffer;
6. } STRING,
7. \*PSTRING;

Individual member semantics are specified in [MS-LSAD] section 2.2.3.1.

### LSAPR\_ACL

The LSAPR\_ACL structure defines the header of an [**access control list (ACL)**](#gt_9f92aa05-dd0a-45f2-88d6-89f1fb654395) as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 2.2.3.2.

1. typedef struct \_LSAPR\_ACL {
2. unsigned char AclRevision;
3. unsigned char Sbz1;
4. unsigned short AclSize;
5. [size\_is(AclSize - 4)] unsigned char Dummy1[\*];
6. } LSAPR\_ACL,
7. \*PLSAPR\_ACL;

Individual member semantics are specified in [MS-LSAD] section 2.2.3.2.

### SECURITY\_DESCRIPTOR\_CONTROL

The SECURITY\_DESCRIPTOR\_CONTROL type contains a set of bit flags as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 2.2.3.3.

This type is declared as follows:

1. typedef unsigned short SECURITY\_DESCRIPTOR\_CONTROL, \*PSECURITY\_DESCRIPTOR\_CONTROL;

### LSAPR\_SECURITY\_DESCRIPTOR

The LSAPR\_SECURITY\_DESCRIPTOR structure defines an object's security descriptor as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 2.2.3.4.

1. typedef struct \_LSAPR\_SECURITY\_DESCRIPTOR {
2. unsigned char Revision;
3. unsigned char Sbz1;
4. SECURITY\_DESCRIPTOR\_CONTROL Control;
5. PRPC\_SID Owner;
6. PRPC\_SID Group;
7. PLSAPR\_ACL Sacl;
8. PLSAPR\_ACL Dacl;
9. } LSAPR\_SECURITY\_DESCRIPTOR,
10. \*PLSAPR\_SECURITY\_DESCRIPTOR;

Individual member semantics are specified in [MS-LSAD] section 2.2.3.4.

### SECURITY\_IMPERSONATION\_LEVEL

The SECURITY\_IMPERSONATION\_LEVEL enumeration defines a set of values that specify security impersonation levels as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 2.2.3.5.

1. typedef enum \_SECURITY\_IMPERSONATION\_LEVEL
2. {
3. SecurityAnonymous = 0,
4. SecurityIdentification = 1,
5. SecurityImpersonation = 2,
6. SecurityDelegation = 3
7. } SECURITY\_IMPERSONATION\_LEVEL,
8. \*PSECURITY\_IMPERSONATION\_LEVEL;

Individual value semantics are specified in [MS-LSAD] section 2.2.3.5.

### SECURITY\_CONTEXT\_TRACKING\_MODE

The SECURITY\_CONTEXT\_TRACKING\_MODE type specifies how the server tracks the client's security context as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 2.2.3.6.

This type is declared as follows:

1. typedef unsigned char SECURITY\_CONTEXT\_TRACKING\_MODE, \*PSECURITY\_CONTEXT\_TRACKING\_MODE;

Possible values are specified in [MS-LSAD] section 2.2.3.6.

### SECURITY\_QUALITY\_OF\_SERVICE

The SECURITY\_QUALITY\_OF\_SERVICE structure defines information used to support client impersonation as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 2.2.3.7.

1. typedef struct \_SECURITY\_QUALITY\_OF\_SERVICE {
2. unsigned long Length;
3. SECURITY\_IMPERSONATION\_LEVEL ImpersonationLevel;
4. SECURITY\_CONTEXT\_TRACKING\_MODE ContextTrackingMode;
5. unsigned char EffectiveOnly;
6. } SECURITY\_QUALITY\_OF\_SERVICE,
7. \*PSECURITY\_QUALITY\_OF\_SERVICE;

Individual member semantics are specified in [MS-LSAD] section 2.2.3.7.

### LSAPR\_OBJECT\_ATTRIBUTES

The LSAPR\_OBJECT\_ATTRIBUTES structure specifies an object and its properties as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 2.2.2.4.

1. typedef struct \_LSAPR\_OBJECT\_ATTRIBUTES {
2. unsigned long Length;
3. unsigned char\* RootDirectory;
4. PSTRING ObjectName;
5. unsigned long Attributes;
6. PLSAPR\_SECURITY\_DESCRIPTOR SecurityDescriptor;
7. PSECURITY\_QUALITY\_OF\_SERVICE SecurityQualityOfService;
8. } LSAPR\_OBJECT\_ATTRIBUTES,
9. \*PLSAPR\_OBJECT\_ATTRIBUTES;

Individual member semantics are specified in [MS-LSAD] section 2.2.2.4.

### ACCESS\_MASK

The ACCESS\_MASK data type is a bitmask that defines the access rights to grant an object. Access types are reconciled with the [**discretionary access control list (DACL)**](#gt_d727f612-7a45-48e4-9d87-71735d62b321) of the object to determine whether the requested access is granted or denied.

This type is declared as follows:

1. typedef unsigned long ACCESS\_MASK;

The possible values are defined in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) sections 2.2.1.1.1 and 2.2.1.1.2. The ACCESS\_MASK data type is further defined in [[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) section 2.4.3.

For this protocol, only a subset of the possible values is required. This subset consists of POLICY\_LOOKUP\_NAMES, which is defined in the following table.

| Value | Meaning |
| --- | --- |
| POLICY\_LOOKUP\_NAMES  0x00000800 | Access to translate names and [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d). |

### LSAPR\_TRUST\_INFORMATION

The LSAPR\_TRUST\_INFORMATION structure contains information about a [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca).

1. typedef struct \_LSAPR\_TRUST\_INFORMATION {
2. RPC\_UNICODE\_STRING Name;
3. PRPC\_SID Sid;
4. } LSAPR\_TRUST\_INFORMATION,
5. \*PLSAPR\_TRUST\_INFORMATION;

Individual member semantics are specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 2.2.7.1.

### LSAPR\_REFERENCED\_DOMAIN\_LIST

The LSAPR\_REFERENCED\_DOMAIN\_LIST structure contains information about the [**domains**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) referenced in a lookup operation.

1. typedef struct \_LSAPR\_REFERENCED\_DOMAIN\_LIST {
2. unsigned long Entries;
3. [size\_is(Entries)] PLSAPR\_TRUST\_INFORMATION Domains;
4. unsigned long MaxEntries;
5. } LSAPR\_REFERENCED\_DOMAIN\_LIST,
6. \*PLSAPR\_REFERENCED\_DOMAIN\_LIST;

**Entries:**  Contains the number of domains referenced in the lookup operation.

**Domains:**  Contains a set of structures that identify domains. If the **Entries** field in this structure is not 0, this field MUST be non-NULL. If **Entries** is 0, this field MUST be ignored.

**MaxEntries:**  This field MUST be ignored. The content is unspecified, and no requirements are placed on its value since it is never used.

### SID\_NAME\_USE

The SID\_NAME\_USE enumeration contains values that specify the type of an account.[<6>](#Appendix_A_6" \o "Product behavior note 6)

1. typedef enum \_SID\_NAME\_USE
2. {
3. SidTypeUser = 1,
4. SidTypeGroup,
5. SidTypeDomain,
6. SidTypeAlias,
7. SidTypeWellKnownGroup,
8. SidTypeDeletedAccount,
9. SidTypeInvalid,
10. SidTypeUnknown,
11. SidTypeComputer,
12. SidTypeLabel
13. } SID\_NAME\_USE,
14. \*PSID\_NAME\_USE;

The SidTypeInvalid and SidTypeComputer enumeration values are not used in this protocol. Usage information on the remaining enumeration values is specified in section [3.1.1](#Section_fa772a2455d44911b3d5eda0bd2ec4e1).

### LSA\_TRANSLATED\_SID

The LSA\_TRANSLATED\_SID structure contains information about a [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) after translation from a name to a [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d). This structure MUST always be accompanied by an [LSAPR\_REFERENCED\_DOMAIN\_LIST](#Section_3a52af31247a4b0891a01d46b2cc49b2) structure when **DomainIndex** is not -1.

1. typedef struct \_LSA\_TRANSLATED\_SID {
2. SID\_NAME\_USE Use;
3. unsigned long RelativeId;
4. long DomainIndex;
5. } LSA\_TRANSLATED\_SID,
6. \*PLSA\_TRANSLATED\_SID;

**Use:**  Defines the type of the security principal, as specified in section [2.2.13](#Section_171e9a878e014bd8a35e3468128c8fc4).

**RelativeId:**  Contains the [**relative identifier (RID)**](#gt_df3d0b61-56cd-4dac-9402-982f1fedc41c) of the security principal with respect to its [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca).

**DomainIndex:**  Contains the index into an LSAPR\_REFERENCED\_DOMAIN\_LIST structure that specifies the domain that the security principal is in. A **DomainIndex** value of -1 MUST be used to specify that there are no corresponding domains. Other negative values MUST NOT be returned.

### LSAPR\_TRANSLATED\_SIDS

The LSAPR\_TRANSLATED\_SIDS structure defines a set of translated [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d).

1. typedef struct \_LSAPR\_TRANSLATED\_SIDS {
2. [range(0,1000)] unsigned long Entries;
3. [size\_is(Entries)] PLSA\_TRANSLATED\_SID Sids;
4. } LSAPR\_TRANSLATED\_SIDS,
5. \*PLSAPR\_TRANSLATED\_SIDS;

**Entries:**  Contains the number of translated SIDs.[<7>](#Appendix_A_7" \o "Product behavior note 7)

**Sids:**  Contains a set of structures that contain translated SIDs. If the Entries field in this structure is not 0, this field MUST be non-NULL. If Entries is 0, this field MUST be NULL.

### LSAP\_LOOKUP\_LEVEL

The LSAP\_LOOKUP\_LEVEL enumeration defines different scopes for searches during translation.[<8>](#Appendix_A_8" \o "Product behavior note 8)

1. typedef enum \_LSAP\_LOOKUP\_LEVEL
2. {
3. LsapLookupWksta = 1,
4. LsapLookupPDC,
5. LsapLookupTDL,
6. LsapLookupGC,
7. LsapLookupXForestReferral,
8. LsapLookupXForestResolve,
9. LsapLookupRODCReferralToFullDC
10. } LSAP\_LOOKUP\_LEVEL,
11. \*PLSAP\_LOOKUP\_LEVEL;

**LsapLookupWksta:** [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) MUST be searched in the views under the Security Principal SID and Security Principal SID History columns in the following order:

* Predefined Translation View, as specified in section [3.1.1.1.1](#Section_e09da72ee6c94f91aa6468b0475719b6).
* Configurable Translation View, as specified in section [3.1.1.1.2](#Section_252cb6b02edc44039d755b2786ba859e).
* Builtin Domain Principal View of the account database on the [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553), as specified in section [3.1.1.1.3](#Section_871df7fe34704b699e5915db70e8d4c6).
* Account Domain View of account database on that machine, as specified in section [3.1.1.1.6](#Section_1777c8867e4740a0979eda0d76fd6d5c).
  + If the machine is not joined to a [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca), the search ends here.
* If the machine is not a [**domain controller**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd): the Account Domain View of the domain to which this machine is joined.
* Forest View (section [3.1.1.1.9](#Section_c69c583439c0430c9e1498f3fd7ccfde)) of the [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62) of the domain to which this machine is joined, unless *ClientRevision* is 0x00000001 and the machine is joined to a mixed mode domain, as specified in [[MS-ADTS]](%5bMS-ADTS%5d.pdf#Section_d243592709994c628c6d13ba31a52e1a) section 6.1.4.1.
* Forest Views of [**trusted forests**](#gt_3b76a71f-9697-4836-9c69-09899b23c21b) for the forest of the domain to which this machine is joined, unless ClientRevision is 0x00000001 and the machine is joined to a mixed mode domain, as specified in [MS-ADTS] section 6.1.4.1.
* Account Domain Views of externally [**trusted domains**](#gt_f2f00d47-6cf2-4b32-b8f7-b63e38e2e9c4) for the domain to which this machine is joined.

**LsapLookupPDC:** SIDs MUST be searched in the views under the Security Principal SID and Security Principal SID History columns in the following order:

* Account Domain View of the domain to which this machine is joined.
* Forest View of the forest of the domain to which this machine is joined, unless *ClientRevision* is 0x00000001 and the machine is joined to a mixed mode domain, as specified in [MS-ADTS] section 6.1.4.1.
* Forest Views of trusted forests for the forest of the domain to which this machine is joined, unless *ClientRevision* is 0x00000001 and the machine is joined to a mixed mode domain, as specified in [MS-ADTS] section 6.1.4.1.
* Account Domain Views of externally trusted domains for the domain to which this machine is joined.

**LsapLookupRODCReferralToFullDC:** SIDs MUST be searched in the databases under the Security Principal SID and Security Principal SID History columns in the following order:

* Forest Views of trusted forests for the forest of the domain to which this machine is joined, unless ClientRevision is 0x00000001 and the machine is joined to a mixed mode domain, as specified in [MS-ADTS] section 6.1.4.1.
* Account Domain Views of externally trusted domains for the domain to which this machine is joined.

**LsapLookupTDL:** SIDs MUST be searched in the databases under the Security Principal SID column in the following view:

* Account Domain View of the [**domain NC**](#gt_40a58fa4-953e-4cf3-96c8-57dba60237ef) for the domain to which this machine is joined.

**LsapLookupGC:** SIDs MUST be searched in the databases under the Security Principal SID and Security Principal SID History columns in the following view:

* Forest View of the forest of the domain to which this machine is joined.

**LsapLookupXForestReferral:** SIDs MUST be searched in the databases under the Security Principal SID and Security Principal SID History columns in the following views:

* Forest Views of trusted forests for the forest of the domain to which this machine is joined.

**LsapLookupXForestResolve:** SIDs MUST be searched in the databases under the Security Principal SID and Security Principal SID History columns in the following view:

* Forest View of the forest of the domain to which this machine is joined.

### LSAPR\_SID\_INFORMATION

The LSAPR\_SID\_INFORMATION structure contains a PRPC\_SID value.

1. typedef struct \_LSAPR\_SID\_INFORMATION {
2. PRPC\_SID Sid;
3. } LSAPR\_SID\_INFORMATION,
4. \*PLSAPR\_SID\_INFORMATION;

**Sid:**  Contains the PRPC\_SID value, as specified in [[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) section 2.4.2.3. This field MUST be non-NULL.

### LSAPR\_SID\_ENUM\_BUFFER

The LSAPR\_SID\_ENUM\_BUFFER structure defines a set of [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d). This structure is used during a translation request for a batch of SIDs to names.

1. typedef struct \_LSAPR\_SID\_ENUM\_BUFFER {
2. [range(0,20480)] unsigned long Entries;
3. [size\_is(Entries)] PLSAPR\_SID\_INFORMATION SidInfo;
4. } LSAPR\_SID\_ENUM\_BUFFER,
5. \*PLSAPR\_SID\_ENUM\_BUFFER;

**Entries:**  Contains the number of translated SIDs.[<9>](#Appendix_A_9" \o "Product behavior note 9)

**SidInfo:**  Contains a set of structures that contain SIDs, as specified in section [2.2.17](#Section_f04a771bc018409881b52a819f9b5db8). If the **Entries** field in this structure is not 0, this field MUST be non-NULL. If **Entries** is 0, this field MUST be ignored.

### LSAPR\_TRANSLATED\_NAME

The LSAPR\_TRANSLATED\_NAME structure contains information about a [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409), along with the human-readable identifier for that security principal. This structure MUST always be accompanied by an [LSAPR\_REFERENCED\_DOMAIN\_LIST](#Section_3a52af31247a4b0891a01d46b2cc49b2) structure when **DomainIndex** is not -1, which contains the [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) information for the security principals.

1. typedef struct \_LSAPR\_TRANSLATED\_NAME {
2. SID\_NAME\_USE Use;
3. RPC\_UNICODE\_STRING Name;
4. long DomainIndex;
5. } LSAPR\_TRANSLATED\_NAME,
6. \*PLSAPR\_TRANSLATED\_NAME;

**Use:**  Defines the type of the security principal, as specified in section [2.2.13](#Section_171e9a878e014bd8a35e3468128c8fc4).

**Name:**  Contains the name of the security principal, with syntaxes described in section [3.1.4.5](#Section_2C0E8A6DDF0F4C77BBEBF7CD0F6B2361). The RPC\_UNICODE\_STRING structure is defined in [[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) section 2.3.10.

**DomainIndex:**  Contains the index into the corresponding LSAPR\_REFERENCED\_DOMAIN\_LIST structure that specifies the domain that the security principal is in. A **DomainIndex** value of -1 MUST be used to specify that there are no corresponding domains. Other negative values MUST NOT be used.

### LSAPR\_TRANSLATED\_NAMES

The LSAPR\_TRANSLATED\_NAMES structure defines a set of translated names. This is used in the response to a translation request from [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) to names.

1. typedef struct \_LSAPR\_TRANSLATED\_NAMES {
2. [range(0,20480)] unsigned long Entries;
3. [size\_is(Entries)] PLSAPR\_TRANSLATED\_NAME Names;
4. } LSAPR\_TRANSLATED\_NAMES,
5. \*PLSAPR\_TRANSLATED\_NAMES;

**Entries:**  Contains the number of translated names.[<10>](#Appendix_A_10" \o "Product behavior note 10)

**Names:**  Contains a set of translated names, as specified in section [2.2.19](#Section_52e1ccc1b57b4c02b35fbd64913ce99b). If the Entries field in this structure is not 0, this field MUST be non-NULL. If Entries is 0, this field MUST be ignored.

### LSAPR\_TRANSLATED\_NAME\_EX

The LSAPR\_TRANSLATED\_NAME\_EX structure contains information about a [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) along with the human-readable identifier for that security principal. This structure MUST always be accompanied by an [LSAPR\_REFERENCED\_DOMAIN\_LIST](#Section_3a52af31247a4b0891a01d46b2cc49b2) structure when **DomainIndex** is not -1, which contains the [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) information for the security principals.

1. typedef struct \_LSAPR\_TRANSLATED\_NAME\_EX {
2. SID\_NAME\_USE Use;
3. RPC\_UNICODE\_STRING Name;
4. long DomainIndex;
5. unsigned long Flags;
6. } LSAPR\_TRANSLATED\_NAME\_EX,
7. \*PLSAPR\_TRANSLATED\_NAME\_EX;

**Use:**  Defines the type of the security principal, as specified in section [2.2.13](#Section_171e9a878e014bd8a35e3468128c8fc4).

**Name:**  Contains the name of the security principal. The RPC\_UNICODE\_STRING structure is defined in [[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) section 2.3.10.

**DomainIndex:**  Contains the index into the corresponding LSAPR\_REFERENCED\_DOMAIN\_LIST structure that specifies the domain that the security principal is in. A **DomainIndex** value of -1 MUST be used to specify that there are no corresponding domains. Other negative values MUST NOT be used.

**Flags:**  Contains bitmapped values that define the properties of this translation. The value MUST be the logical OR of zero or more of the following flags. These flags communicate the following additional information about how the [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) was resolved.

| Value | Meaning |
| --- | --- |
| 0x00000001 | The SID was not found by matching against the security principal SID property. |
| 0x00000002 | The SID might be found by traversing a [**forest trust**](#gt_035d9ce5-f117-4251-8d4d-127c462ec4a0). |
| 0x00000004 | The SID was found by matching against the last database view, defined in section [3.1.1.1.1](#Section_e09da72ee6c94f91aa6468b0475719b6). |

All other bits MUST be 0 and ignored on receipt.[<11>](#Appendix_A_11" \o "Product behavior note 11)

### LSAPR\_TRANSLATED\_NAMES\_EX

The LSAPR\_TRANSLATED\_NAMES\_EX structure contains a set of translated names.

1. typedef struct \_LSAPR\_TRANSLATED\_NAMES\_EX {
2. [range(0,20480)] unsigned long Entries;
3. [size\_is(Entries)] PLSAPR\_TRANSLATED\_NAME\_EX Names;
4. } LSAPR\_TRANSLATED\_NAMES\_EX,
5. \*PLSAPR\_TRANSLATED\_NAMES\_EX;

**Entries:**  Contains the number of translated names.[<12>](#Appendix_A_12" \o "Product behavior note 12)

**Names:**  Contains a set of structures that contain translated names, as specified in section [2.2.21](#Section_6a9131acf1a74b968917a289a2c48de1). If the **Entries** field in this structure is not 0, this field MUST be non-NULL. If **Entries** is 0, this field MUST be ignored.

### LSAPR\_TRANSLATED\_SID\_EX

The LSAPR\_TRANSLATED\_SID\_EX structure contains information about a [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) after it has been translated into a [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d). This structure MUST always be accompanied by an [LSAPR\_REFERENCED\_DOMAIN\_LIST](#Section_3a52af31247a4b0891a01d46b2cc49b2) structure when **DomainIndex** is not -1, which contains the [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) information for the security principal.

This structure differs from [LSA\_TRANSLATED\_SID](#Section_d067a76efd924e3691e78605814414cc) only in that a new **Flags** field is added.

1. typedef struct \_LSAPR\_TRANSLATED\_SID\_EX {
2. SID\_NAME\_USE Use;
3. unsigned long RelativeId;
4. long DomainIndex;
5. unsigned long Flags;
6. } LSAPR\_TRANSLATED\_SID\_EX,
7. \*PLSAPR\_TRANSLATED\_SID\_EX;

**Use:**  Defines the type of the security principal, as specified in section [2.2.13](#Section_171e9a878e014bd8a35e3468128c8fc4).

**RelativeId:**  Contains the [**relative identifier (RID)**](#gt_df3d0b61-56cd-4dac-9402-982f1fedc41c) of the security principal with respect to its domain.

**DomainIndex:**  Contains the index into the corresponding LSAPR\_REFERENCED\_DOMAIN\_LIST structure that specifies the domain that the security principal is in. A **DomainIndex** value of -1 MUST be used to specify that there are no corresponding domains. Other negative values MUST NOT be used.

**Flags:**  Contains bitmapped values that define the properties of this translation. The value MUST be the logical OR of zero or more of the following flags. These flags communicate additional information on how the name was resolved.

| Value | Meaning |
| --- | --- |
| 0x00000001 | The name was not found by matching against the *Security Principal Name* property. |
| 0x00000002 | The name might be found by traversing a [**forest trust**](#gt_035d9ce5-f117-4251-8d4d-127c462ec4a0). |
| 0x00000004 | The name was found by matching against the last database view, as defined in section [3.1.1.1.1](#Section_e09da72ee6c94f91aa6468b0475719b6). |

All other bits MUST be 0 and ignored on receipt.[<13>](#Appendix_A_13" \o "Product behavior note 13)

### LSAPR\_TRANSLATED\_SIDS\_EX

The LSAPR\_TRANSLATED\_SIDS\_EX structure contains a set of translated [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d).

1. typedef struct \_LSAPR\_TRANSLATED\_SIDS\_EX {
2. [range(0,1000)] unsigned long Entries;
3. [size\_is(Entries)] PLSAPR\_TRANSLATED\_SID\_EX Sids;
4. } LSAPR\_TRANSLATED\_SIDS\_EX,
5. \*PLSAPR\_TRANSLATED\_SIDS\_EX;

**Entries:**  Contains the number of translated SIDs.[<14>](#Appendix_A_14" \o "Product behavior note 14)

**Sids:**  Contains a set of structures that contain translated SIDs, as specified in section [2.2.23](#Section_274ee4e386b5445099e9f1678b9e96f5). If the **Entries** field in this structure is not 0, this field MUST be non-NULL. If **Entries** is 0, this field MUST be NULL.

### LSAPR\_TRANSLATED\_SID\_EX2

The LSAPR\_TRANSLATED\_SID\_EX2 structure contains information about a [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) after it has been translated into a [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d). This structure MUST always be accompanied by an [LSAPR\_REFERENCED\_DOMAIN\_LIST](#Section_3a52af31247a4b0891a01d46b2cc49b2) structure when **DomainIndex** is not -1.

This structure differs from [LSAPR\_TRANSLATED\_SID\_EX](#Section_274ee4e386b5445099e9f1678b9e96f5) only in that a SID is returned instead of a [**RID**](#gt_df3d0b61-56cd-4dac-9402-982f1fedc41c).

1. typedef struct \_LSAPR\_TRANSLATED\_SID\_EX2 {
2. SID\_NAME\_USE Use;
3. PRPC\_SID Sid;
4. long DomainIndex;
5. unsigned long Flags;
6. } LSAPR\_TRANSLATED\_SID\_EX2,
7. \*PLSAPR\_TRANSLATED\_SID\_EX2;

**Use:**  Defines the type of the security principal, as specified in section [2.2.13](#Section_171e9a878e014bd8a35e3468128c8fc4).

**Sid:**  Contains the SID ([[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) section 2.4.2.3) of the security principal. This field MUST be treated as a [ref] pointer and hence MUST be non-NULL.

**DomainIndex:**  Contains the index into an LSAPR\_REFERENCED\_DOMAIN\_LIST structure that specifies the [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) that the security principal is in. A **DomainIndex** value of -1 MUST be used to specify that there are no corresponding domains. Other negative values MUST NOT be used.

**Flags:**  Contains bitmapped values that define the properties of this translation. The value MUST be the logical OR of zero or more of the following flags. These flags communicate additional information on how the name was resolved.

| Value | Meaning |
| --- | --- |
| 0x00000001 | The name was not found by matching against the *Security Principal Name* property. |
| 0x00000002 | The name might be found by traversing a [**forest trust**](#gt_035d9ce5-f117-4251-8d4d-127c462ec4a0). |
| 0x00000004 | The name was found by matching against the last database view (see section [3.1.1.1.1](#Section_e09da72ee6c94f91aa6468b0475719b6)). |

All other bits MUST be 0 and ignored on receipt.[<15>](#Appendix_A_15" \o "Product behavior note 15)

### LSAPR\_TRANSLATED\_SIDS\_EX2

The LSAPR\_TRANSLATED\_SIDS\_EX2 structure contains a set of translated [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d).

1. typedef struct \_LSAPR\_TRANSLATED\_SIDS\_EX2 {
2. [range(0,1000)] unsigned long Entries;
3. [size\_is(Entries)] PLSAPR\_TRANSLATED\_SID\_EX2 Sids;
4. } LSAPR\_TRANSLATED\_SIDS\_EX2,
5. \*PLSAPR\_TRANSLATED\_SIDS\_EX2;

**Entries:**  Contains the number of translated SIDs.[<16>](#Appendix_A_16" \o "Product behavior note 16)

**Sids:**  Contains a set of structures that define translated SIDs, as specified in section [2.2.25](#Section_e397a7ac842a4d4487ef170392dee522). If the **Entries** field in this structure is not 0, this field MUST be non-NULL. If **Entries** is 0, this field MUST be NULL.

## Directory Service Schema Elements

This protocol is part of the [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) core family of protocols. In order to be fully compliant with Active Directory, an implementation of this protocol must be used in conjunction with the full Active Directory schema, containing all the schema attributes and classes specified in [[MS-ADA1]](%5bMS-ADA1%5d.pdf#Section_19528560f41e4623a406dabcfff0660f), [[MS-ADA2]](%5bMS-ADA2%5d.pdf#Section_e20ebc4e528540bab3bdffcb81c2783e), [[MS-ADA3]](%5bMS-ADA3%5d.pdf#Section_4517e8353ee644d4bb95a94b6966bfb0), and [[MS-ADSC]](%5bMS-ADSC%5d.pdf#Section_9abb5e97123d4da99557b353ab79b830).

# Protocol Details

This protocol determines whether it is running on a [**domain controller**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) by querying the current server configuration. This is accomplished by calling the abstract interface ServerGetInfo, specified in [[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) section 2.6, and indicating a *level* of 101. The resulting *bufptr* contains a SERVER\_INFO\_101 structure, as specified in [MS-DTYP] section 2.3.12. The determination is TRUE if **sv101\_version\_type** contains SV\_TYPE\_DOMAIN\_CTRL or SV\_TYPE\_DOMAIN\_BAKCTRL. If **sv101\_version\_type** does not contain either of these values, the determination is FALSE.

## Server Details

### Abstract Data Model

This section describes a possible data organization that an implementation might maintain to participate in this protocol. The described organization is provided to help explain how the protocol behaves. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with what is described in this document.

The server implementing this protocol MUST create views, as specified in section [3.1.1.1](#Section_77855eedd3c74538b52720a36d4dcb18), over [**domain databases**](#gt_b5690afe-df38-463b-9768-40b47b3989fe) that the server [**trusts**](#gt_5ee032d0-d944-4acb-bbb5-b1cfc7df6db6) to be able to translate [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) to names, and vice versa. These views contain SIDs and different types of names, as well as the [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) to which each [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) belongs. These views MUST be kept in sync with the original databases whose data is used to create the view. The sync mechanism is implementation-specific, and there is no upper bound on the time from when a change occurs in a domain database to when that change appears in the view. A scoping parameter defined as [LSAP\_LOOKUP\_LEVEL](#Section_9d1166ccbcfd4e22a8acf55eae57c99f) can be considered during message processing by applying these views to different databases.

The following domain databases are used in this protocol.

* [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90), specified in [[MS-ADTS]](%5bMS-ADTS%5d.pdf#Section_d243592709994c628c6d13ba31a52e1a).
* The Local Security Authority (LSA) Domain Policy database, specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc).
* The Service Control Manager database, specified in [[MS-SCMR]](%5bMS-SCMR%5d.pdf#Section_705b624a13de43ccb8a299573da3635f).
* The Predefined Translation database, specified in section [3.1.1.1.1](#Section_e09da72ee6c94f91aa6468b0475719b6).

The domain databases MUST satisfy [**ACID**](#gt_44f0cb35-f618-456d-b7b1-25213fed1045) properties [GRAY] during creation or updates of views.

The abstract data models presented in sections [3.1.1.2](#Section_20a48d4dffed4cb994f88d29ff2014a8) and [3.1.1.3](#Section_e57ea3504633401b964d01a4ef4bd819) are used along with domain database information to construct the logical views specified in section 3.1.1.1. The views, in turn, are used to process mapping requests. This process is specified in sections [3.1.4.4](#Section_4e3f0a3f86ae4d50aa40ec25013a89e3), [3.1.4.5](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361), and [3.1.4.9](#Section_678dd7ed611b4d948f5ca06c76ae5c7e), which describe the server processing of [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) messages.

In this section, Active Directory schema names are referenced when describing attributes stored in the domain database. For more information, see [[MS-ADSC]](%5bMS-ADSC%5d.pdf#Section_9abb5e97123d4da99557b353ab79b830), [[MS-ADA1]](%5bMS-ADA1%5d.pdf#Section_19528560f41e4623a406dabcfff0660f), [[MS-ADA2]](%5bMS-ADA2%5d.pdf#Section_e20ebc4e528540bab3bdffcb81c2783e), and [[MS-ADA3]](%5bMS-ADA3%5d.pdf#Section_4517e8353ee644d4bb95a94b6966bfb0).

#### Database Views

The following names represent the columns of the views that are specified in this section:

* **Domain DNS Name**
* **Domain NetBIOS Name**
* **Domain SID**
* **Security Principal Name**
* **Additional Security Principal Name**
* **Default User Principal Names**
* **User Principal Name**
* **Security Principal SID**
* **Security Principal SID History**
* **Security Principal Type**

**Domain SID**, **Security Principal Name**, **Security Principal SID**, and **Security Principal Type** are mandatory columns that MUST have values, while the rest MAY have values.

Some views are populated with constants, while others are populated by extracting current information from various [**domain databases**](#gt_b5690afe-df38-463b-9768-40b47b3989fe). The specific views and how they are populated are described in the following paragraphs and subsections.

When a view is populated by extracting current information from the domain database of a selected [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca), and the database or a replica of the database is local to the server, the server MUST use the local values. If the security database or a replica is not local, the server MUST use values from a remote replica. For remote values, there are no consistency guarantees from call to call; that is, the view might contain values from a different remote replica on each call.

The following assumptions are made for all views:

* **Security Principal SID** forms a unique key for each row.
* The **Security Principal Name** and **Domain NetBIOS Name** pair forms a unique key for each row.
* If **Domain DNS Name** is not empty, the **Security Principal Name** and **Domain DNS Name** pair forms a unique key for each row.
* If **Additional Security Principal Name** is not empty, the **Additional Security Principal Name** and **Domain NetBIOS Name** pair forms a unique key for each row.
* If **Additional Security Principal Name** and **Domain DNS Name** are not empty, the **Additional Security Principal Name** and **Domain DNS Name** pair forms a unique key for each row.

These assumptions MUST be held true by the database implementations from which these views are created.[<17>](#Appendix_A_17" \o "Product behavior note 17)

##### Predefined Translation Database and Corresponding View

The Predefined Translation View MUST be constructed using the following non-customizable Predefined Translation Tables. There is a one-to-one mapping between the rows in the view and the rows in the tables. The columns that are not mentioned in these tables are empty. The tables are grouped by the Domain NetBIOS Name and Domain SID columns for easier understanding.

Values of the Domain NetBIOS Name and Security Principal Name columns are shown in U.S. English. In an actual system, these values MUST be localized to the language defined as system locale at the time of message processing.

**Domain NetBIOS Name: "" (empty domain name)**

**Domain SID: S-1-0**

| Security Principal Name | Security Principal SID | Security Principal Type |
| --- | --- | --- |
| Null Sid | S-1-0-0 | SidTypeWellKnownGroup |

**Domain NetBIOS Name: "" (empty domain name)**

**Domain SID: S-1-1**

| Security Principal Name | Security Principal SID | Security Principal Type |
| --- | --- | --- |
| Everyone | S-1-1-0 | SidTypeWellKnownGroup |

**Domain NetBIOS Name: "" (empty domain name)**

**Domain SID: S-1-2**

| Security Principal Name | Security Principal SID | Security Principal Type |
| --- | --- | --- |
| Local | S-1-2-0 | SidTypeWellKnownGroup |

**Domain NetBIOS Name: "" (empty domain name)**

**Domain SID: S-1-3**

| Security Principal Name | Security Principal SID | Security Principal Type |
| --- | --- | --- |
| Creator Owner | S-1-3-0 | SidTypeWellKnownGroup |
| Creator Group | S-1-3-1 | SidTypeWellKnownGroup |
| Creator Owner Server | S-1-3-2 | SidTypeWellKnownGroup |
| Creator Group Server | S-1-3-3 | SidTypeWellKnownGroup |
| Owner Rights | S-1-3-4 | SidTypeWellKnownGroup |

**Domain NetBIOS Name: NT Pseudo Domain**

**Domain SID: S-1-5**

| **Security Principal Name** | Security Principal SID | Security Principal Type |
| --- | --- | --- |
| NT Pseudo Domain | S-1-5 | SidTypeDomain |

**Domain NetBIOS Name: NT Authority**

**Domain SID: S-1-5**

| **Security Principal Name** | **Security Principal SID** | Security Principal Type |
| --- | --- | --- |
| Dialup | S-1-5-1 | SidTypeWellKnownGroup |
| Network | S-1-5-2 | SidTypeWellKnownGroup |
| Batch | S-1-5-3 | SidTypeWellKnownGroup |
| Interactive | S-1-5-4 | SidTypeWellKnownGroup |
| Service | S-1-5-6 | SidTypeWellKnownGroup |
| Anonymous Logon | S-1-5-7 | SidTypeWellKnownGroup |
| Proxy | S-1-5-8 | SidTypeWellKnownGroup |
| Enterprise Domain Controllers | S-1-5-9 | SidTypeWellKnownGroup |
| Self | S-1-5-10 | SidTypeWellKnownGroup |
| Authenticated Users | S-1-5-11 | SidTypeWellKnownGroup |
| Restricted | S-1-5-12 | SidTypeWellKnownGroup |
| Terminal Server User | S-1-5-13 | SidTypeWellKnownGroup |
| Remote Interactive Logon | S-1-5-14 | SidTypeWellKnownGroup |
| This Organization | S-1-5-15 | SidTypeWellKnownGroup |
| System | S-1-5-18 | SidTypeWellKnownGroup |
| Local Service | S-1-5-19 | SidTypeWellKnownGroup |
| Network Service | S-1-5-20 | SidTypeWellKnownGroup |
| Write Restricted | S-1-5-33 | SidTypeWellKnownGroup |
| Other Organization | S-1-5-1000 | SidTypeWellKnownGroup |

For Windows behavior on the preceding entries, see the following citation.[<18>](#Appendix_A_18" \o "Product behavior note 18)

**Domain NetBIOS Name: Builtin**

**Domain SID: S-1-5-32**

| Security Principal Name | Security Principal SID | Security Principal Type |
| --- | --- | --- |
| Builtin | S-1-5-32 | SidTypeDomain |

**Domain NetBIOS Name: Internet$**

**Domain SID: S-1-7**

| Security Principal Name | Security Principal SID | Security Principal Type |
| --- | --- | --- |
| Internet$ | S-1-7 | SidTypeDomain |

**Domain NetBIOS Name: NT Authority**

**Domain SID: S-1-5-64**

| Security Principal Name | Security Principal SID | Security Principal Type |
| --- | --- | --- |
| NTLM Authentication | S-1-5-64-10 | SidTypeWellKnownGroup |
| Digest Authentication | S-1-5-64-21 | SidTypeWellKnownGroup |
| Channel Authentication | S-1-5-64-14 | SidTypeWellKnownGroup |

For Windows behavior on the preceding entries, see the following citation.[<19>](#Appendix_A_19" \o "Product behavior note 19)

**Domain NetBIOS Name: Mandatory Label**

**Domain SID: S-1-16**

| Security Principal Name | Security Principal SID | Security Principal Type |
| --- | --- | --- |
| Mandatory Label | S-1-16 | SidTypeDomain |
| Untrusted Mandatory Level | S-1-16-0 | SidTypeLabel |
| Low Mandatory Level | S-1-16-4096 | SidTypeLabel |
| Medium Mandatory Level | S-1-16-8192 | SidTypeLabel |
| High Mandatory Level | S-1-16-12288 | SidTypeLabel |
| System Mandatory Level | S-1-16-16384 | SidTypeLabel |
| Protected Process Mandatory Level | S-1-16-20480 | SidTypeLabel |

For Windows behavior on the preceding entries, see the following citation.[<20>](#Appendix_A_20" \o "Product behavior note 20)

##### Configurable Translation Database and Corresponding View

The Configurable Translation Database is a general purpose database for translation between [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) names and their corresponding [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d). The Configurable Translation Database columns are the same as the Predefined Translation Database columns, and the view construction is the same. This database SHOULD be constructed using the abstract data model specified in [[MS-SCMR]](%5bMS-SCMR%5d.pdf#Section_705b624a13de43ccb8a299573da3635f) section 3.1.1. There MUST be one row for the "NT SERVICE" [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca), as defined in the following table. There MUST be one row per service definition. The mapping rules are defined as follows:

* Domain DNS Name, Additional Security Principal Name, User Principal Name, Default User Principal Names, and Security Principal SID History columns are left empty.
* Security Principal SID is mapped from DisplayName in [MS-SCMR] section 3.1.1 using the following method:
  1. Convert the **DisplayName** field to the uppercase, UTF-16 representation.
  2. Take the SHA1 hash of the name:
     1. Hash[0] denoting the first 4 bytes of the resulting hash as an unsigned integer.
     2. Hash[1] denoting the second 4 bytes of the resulting hash as an unsigned integer.
     3. And so on.
  3. Create the SID using the following mapping:
     + S-1-5-80-hash[0]-hash[1]-hash[2]-hash[3]-hash[4]
* Security Principal Name is mapped from DisplayName in [MS-SCMR] section 3.1.1.
* Security Principal Type is mapped to SidTypeWellKnownGroup.

The following table shows two columns in the Configurable Translation Database and Corresponding View as an example with the NT Service Domain and Service Name 'ALG'.

**Domain NetBIOS Name: NT SERVICE**

**Domain SID: S-1-5-80**

| **Security Principal Name** | **Security Principal SID** | **Security Principal Type** |
| --- | --- | --- |
| NT SERVICE | S-1-5-80 | SidTypeDomain |
| ALG | S-1-5-80-2387347252-3645287876-2469496166-3824418187-3586569773 | SidTypeWellKnownGroup |

##### Builtin Domain Principal View

To construct the Builtin Domain Principal View, the following columns from the associated [**domain database**](#gt_b5690afe-df38-463b-9768-40b47b3989fe) MUST be used:

* sAMAccountName
* sAMAccountType
* objectSID

All objects that satisfy the following criteria MUST be part of this view:

* All three columns in the preceding list MUST have values.
* The value of the objectSID attribute MUST contain S-1-5-32 as its prefix.

The columns of such objects MUST be used to construct the Builtin Domain Principal View in the following manner:

* Domain DNS Name, Additional Security Principal Name, User Principal Name, Default User Principal Names, and Security Principal SID History columns are left empty.
* Security Principal SID is mapped from objectSID.
* Security Principal Name is mapped from sAMAccountName.
* Security Principal Type is mapped from sAMAccountType by using the following table.

| sAMAccountType most significant 4 bits | Security Principal Type |
| --- | --- |
| 0x3 | SidTypeUser |
| 0x1 | SidTypeGroup |
| 0x4 or 0x2: These values are treated identically by the protocol. | SidTypeAlias |
| Otherwise | SidTypeUnknown |

* Domain NetBIOS Name and Domain SID are mapped from the row of the Predefined Translation Database View whose [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) is S-1-5-32.

The following is an example of how this view is created:

* An object that represents the administrators group.

| Column | Value |
| --- | --- |
| sAMAccountName | Administrators |
| sAMAccountType | 0x20000000 |
| objectSID | S-1-5-32-544 |

* The view created for that object.

| Column | Value |
| --- | --- |
| Domain DNS Name |  |
| Domain NetBIOS Name | Builtin |
| Domain SID | S-1-5-32 |
| Security Principal Name | Administrators |
| Additional Security Principal Name |  |
| Default User Principal Names |  |
| User Principal Name |  |
| Security Principal SID | S-1-5-32-544 |
| Security Principal SID History |  |
| Security Principal Type | SidTypeAlias |

##### Account Domain Principal View

To construct the Account Domain Principal View, the following columns from the associated [**domain database**](#gt_b5690afe-df38-463b-9768-40b47b3989fe) MUST be used:

* sAMAccountName
* sAMAccountType
* objectSID

All objects that satisfy the following criteria MUST be part of this view:

* All three columns above have values.
* The value of the objectSID attribute does not contain S-1-5-32 as the prefix.

The following columns of such objects MUST be used to construct the Account Domain Principal View in the following manner:

* The Additional Security Principal Name, User Principal Name, and Security Principal SID History columns are left empty.
* Security Principal SID is mapped from objectSID.
* Security Principal Name is mapped from sAMAccountName.
* Security Principal Type is mapped from sAMAccountType by using the mapping rule explained in the Builtin Domain Principal View (section [3.1.1.1.3](#Section_871df7fe34704b699e5915db70e8d4c6)).
* Domain NetBIOS Name, Domain DNS Name, and Domain SID are mapped from Domain Database Information, as specified in section [3.1.1.2](#Section_20a48d4dffed4cb994f88d29ff2014a8).
* Default User Principal Names is constructed using the following rules:
  + If Domain DNS Name is not empty, concatenate sAMAccountName with Domain DNS Name, separated by an @ sign.
  + And if the domain database used is an [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca), concatenate sAMAccountName with Domain NetBIOS Name, separated by an @ sign.

The following is an example of how this view is created:

* An object that represents the administrator user on an Active Directory domain database.

| Column | Value |
| --- | --- |
| sAMAccountName | Administrator |
| sAMAccountType | 0x30000000 |
| objectSID | S-1-5-21-397955417-626881126-188441444-500 |

* The Domain Database Information for that Active Directory domain database.

| Column | Value |
| --- | --- |
| Domain DNS Name | Corp.example.com |
| Domain NetBIOS Name | Corp |
| Domain SID | S-1-5-21-397955417-626881126-188441444 |

* The view created for the administrator object.

| Column | Value |
| --- | --- |
| Domain DNS Name | Corp.example.com |
| Domain NetBIOS Name | Corp |
| Domain SID | S-1-5-21-397955417-626881126-188441444 |
| Security Principal Name | Administrator |
| Additional Security Principal Name |  |
| Default User Principal Names | administrator@corp  administrator@corp.example.com |
| User Principal Name |  |
| Security Principal SID | S-1-5-21-397955417-626881126-188441444-500 |
| Security Principal SID History |  |
| Security Principal Type | SidTypeUser |

##### Account Domain Information View

The Account Domain Information View of a [**domain database**](#gt_b5690afe-df38-463b-9768-40b47b3989fe) contains one row. This view MUST be constructed using the [Domain Database Information (section 3.1.1.2)](#Section_20a48d4dffed4cb994f88d29ff2014a8) of that domain database, and the following mapping:

* User Principal Name, Default User Principal Names, and Security Principal SID History columns are left empty.
* Domain NetBIOS Name, Domain DNS Name, and Domain SID are mapped from Domain Database Information.
* Security Principal SID is the same as Domain SID.
* Security Principal Name is the same as Domain NetBIOS Name.
* Additional Security Principal Name is the same as Domain DNS Name.
* Security Principal Type is SidTypeDomain.

The following example shows how this row might appear on a non–[**domain controller**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd):

* Domain Database Information for a machine.

| Column | Value |
| --- | --- |
| Domain DNS Name |  |
| Domain NetBIOS Name | My-machine |
| Domain SID | S-1-5-21-841930733-165634207-2980115662 |

* The row created for the local Account Domain object.

| Column | Value |
| --- | --- |
| Domain DNS Name |  |
| Domain NetBIOS Name | My-Machine |
| Domain SID | S-1-5-21-841930733-165634207-2980115662 |
| Security Principal Name | My-Machine |
| Additional Security Principal Name |  |
| Default User Principal Names |  |
| User Principal Name |  |
| Security Principal SID | S-1-5-21-841930733-165634207-2980115662 |
| Security Principal SID History |  |
| Security Principal Type | SidTypeDomain |

The following example shows how this row is created for an [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca):

* Domain Database Information for an Active Directory domain.

| Column | Value |
| --- | --- |
| Domain DNS Name | Corp.example.com |
| Domain NetBIOS Name | Corp |
| Domain SID | S-1-5-21-397955417-626881126-188441444 |

* The row created.

| Column | Value |
| --- | --- |
| Domain DNS Name | Corp.example.com |
| Domain NetBIOS Name | Corp |
| Domain SID | S-1-5-21-397955417-626881126-188441444 |
| Security Principal Name | Corp |
| Additional Security Principal Name | Corp.example.com |
| Default User Principal Names |  |
| User Principal Name |  |
| Security Principal SID | S-1-5-21-397955417-626881126-188441444 |
| Security Principal SID History |  |
| Security Principal Type | SidTypeDomain |

##### Account Domain View

This view for a given [**domain database**](#gt_b5690afe-df38-463b-9768-40b47b3989fe) is a union of Account Domain Information View and Account Domain Principal View of that account database.

##### Forest Principal View

To construct the Forest Principal View for a given [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62), the following columns from a union of all [**domain naming contexts (NCs)**](#gt_40a58fa4-953e-4cf3-96c8-57dba60237ef) in the forest MUST be used.

* sAMAccountName
* userPrincipalName
* objectSID
* sidHistory
* sAMAccountType

All objects that satisfy the following criteria MUST be part of this view:

* The sAMAccountName, objectSID, and sAMAccountType attributes have values.
* The value of the objectSID attribute does not contain S-1-5-32 as the prefix.

The following columns MUST be used to construct the Forest Principal View in the following manner:

* User Principal Name is obtained from userPrincipalName.
* Security Principal SID History is obtained from sidHistory.
* Security Principal SID is obtained from objectSID.
* Security Principal Name is obtained from sAMAccountName.
* Additional Security Principal Name is empty.
* Security Principal Type is mapped from sAMAccountType by using the rule explained in the Builtin Domain Principal View section ([3.1.1.1.3](#Section_871df7fe34704b699e5915db70e8d4c6)).
* Domain NetBIOS Name, Domain DNS Name, and Domain SID are mapped from Domain Database Information for the [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) that the object is in.
* Default User Principal Names is constructed using the following rules:
  + Concatenate sAMAccountName with Domain DNS Name, separated by an @ sign.
  + Concatenate sAMAccountName with Domain NetBIOS Name, separated by an @ sign.

The following example shows how this view is created:

* An object that represents the "someone" user on a [**domain controller**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd).

| Column | Value |
| --- | --- |
| sAMAccountName | someone |
| userPrincipalName | someone@example.com |
| objectSID | S-1-5-21-397955417-626881126-188441444-1555 |
| sidHistory | S-1-5-21-1234567890-123456789-456789012-2045  S-1-5-21-7890123456-345678-459012-34524 |
| sAMAccountType | 0x30000000 |

* Domain Database Information for that domain.

| Column | Value |
| --- | --- |
| Domain DNS Name | Corp.example.com |
| Domain NetBIOS Name | Corp |
| Domain SID | S-1-5-21-397955417-626881126-188441444 |

* The view created for the [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409).

| Column | Value |
| --- | --- |
| Domain DNS Name | Corp.example.com |
| Domain NetBIOS Name | Corp |
| Domain SID | S-1-5-21-397955417-626881126-188441444 |
| Security Principal Name | someone |
| Additional Security Principal Name |  |
| Default User Principal Names | someone@corp  someone@corp.example.com |
| User Principal Name | someone@example.com |
| Security Principal SID | S-1-5-21-397955417-626881126-188441444-1555 |
| Security Principal SID History | S-1-5-21-1234567890-123456789-456789012-2045  S-1-5-21-7890123456-345678-459012-34524 |
| Security Principal Type | SidTypeUser |

##### Forest Information View

The Forest Information View of a [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62) is a union of Account Domain Information Views of all [**domains**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) in that forest.

##### Forest View

This view for a given [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62) is a union of Forest Information View and Forest Principal View for that forest.

#### Domain Database Information

Domain database information MUST consist of the following conceptual fields:

* **Domain DNS Name**: A Unicode string that contains the [**DNS name**](#gt_102a36e2-f66f-49e2-bee3-558736b2ecd5) of the [**domain database**](#gt_b5690afe-df38-463b-9768-40b47b3989fe).
* **Domain NetBIOS Name**: A Unicode string that contains the NetBIOS name of the domain database.
* **Domain SID**: The [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) of the domain database.

When the domain database is an [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90) domain database, this information MUST be populated from DNS Domain Information, as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 3.1.1.1.

Otherwise, this information MUST be populated from the Account Domain Information, as specified in [MS-LSAD] section 3.1.1.1. In this case, **Domain DNS Name** MUST be set to empty.

#### Trusted Domains and Forests Information

The Local Security Authority (LSA) Domain Policy database, as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc), contains [**trust**](#gt_5ee032d0-d944-4acb-bbb5-b1cfc7df6db6) information for a given [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) and [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62). Of all the information that it manages for trusts, only the following conceptual columns are of interest to this protocol:

* **Trusted Domain DNS Name**: The [**DNS name**](#gt_102a36e2-f66f-49e2-bee3-558736b2ecd5) of the [**trusted domain**](#gt_f2f00d47-6cf2-4b32-b8f7-b63e38e2e9c4).
* **Trusted Domain NetBIOS Name**: The NetBIOS name of the same domain.
* **Domain SID**: The [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) of the same domain.
* **Trust Direction**: Indicates whether the trust is inbound, outbound, or both.
* **Trust Type**: Type of the trust.
* **Trust Attributes**: Additional characteristics of the trust.
* **Forest Trust Information**: Attributes of this trust as it pertains to the forest.

In the [**Active Directory**](#gt_e467d927-17bf-49c9-98d1-96ddf61ddd90), **Trusted Domain DNS Name** is stored in the trustPartner attribute, **Trusted Domain NetBIOS Name** in flatName, **Trust Direction** in trustDirection, **Trust Type** in trustType, **Trust Attributes** in trustAttributes as a bitmask, and **Forest Trust Information** in msDs-trustForestTrustInfo.

Trust information that satisfies the following criteria MUST be used by this protocol:

* Trust Type is a Windows trust, which means that the trusted party is a Windows domain, as specified in [MS-LSAD]. A Windows trust is represented by a value of 0x0000002 in the trustType attribute. **Trust Attributes** can declare a trust to be a [**forest trust**](#gt_035d9ce5-f117-4251-8d4d-127c462ec4a0) by having the 0x00000008 bit set in the trustAttributes attribute, as specified in [MS-LSAD]. For more information, see [[MS-ADSC]](%5bMS-ADSC%5d.pdf#Section_9abb5e97123d4da99557b353ab79b830), [[MS-ADA1]](%5bMS-ADA1%5d.pdf#Section_19528560f41e4623a406dabcfff0660f), [[MS-ADA2]](%5bMS-ADA2%5d.pdf#Section_e20ebc4e528540bab3bdffcb81c2783e), and [[MS-ADA3]](%5bMS-ADA3%5d.pdf#Section_4517e8353ee644d4bb95a94b6966bfb0).
* Trust Direction is outbound or bidirectional, as specified in [MS-LSAD].

The NetBIOS and DNS names of the trusted domain can be used to locate a [**domain NC**](#gt_40a58fa4-953e-4cf3-96c8-57dba60237ef) "replica" for that domain by using a [**domain controller**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) locator algorithm, an example of which is described in [[MS-ADOD]](%5bMS-ADOD%5d.pdf#Section_5ff67bf4c14548cb89cd4f5482d94664) sections 2.7.7.3.1 and 3.1.1.

The [**trust attributes**](#gt_0b8230da-0fd8-492c-b84a-d8467c3dc1ef) declare the trust to be a forest trust, as specified in [MS-LSAD] section 3.1.1.5, if and only if the following conditions are met:

* The trusted domain information satisfies the preceding criteria.
* The trusted domain information is in the [**root domain**](#gt_529c766b-af01-4bc8-b853-65fba6c704b3) of the forest.

If the trust attributes declare the trust to be a forest trust, the Forest Trust Information column contains information about the [**trusted forest**](#gt_3b76a71f-9697-4836-9c69-09899b23c21b). Specifically, this information consists of a [**domain name**](#gt_45a1c9f1-0263-49a8-97c7-7aca1a99308c), a domain DNS name, and the domain SID of all domains in the forest, as well as top-level names, including [**UPN**](#gt_9d606f55-b798-4def-bf96-97b878bb92c6) suffixes for user principal names, as specified in [[MS-ADTS]](%5bMS-ADTS%5d.pdf#Section_d243592709994c628c6d13ba31a52e1a) section 6.1.6 and [MS-LSAD] section 3.1.1.5.

### Timers

No protocol timers are required other than the internal ones used in [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) to implement resiliency to network outages, as specified in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15).

### Initialization

The server MUST start listening on the well-known named pipe for the [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) interface, as specified in section [2.1](#Section_f25df96b48c648848cf01418e3148c66). The server SHOULD register the RPC interface to listen over TCP/IP.

### Message Processing Events and Sequencing Rules

This section specifies the methods for this protocol, in addition to their processing rules.[<21>](#Appendix_A_21" \o "Product behavior note 21)[<22>](#Appendix_A_22" \o "Product behavior note 22)

This protocol contains some methods with parameters that have no effect on message processing in any environment. These are called out as "ignored".

**Note**  This protocol shares an interface with the Local Security Authority (Domain Policy) Remote Protocol, as specified in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc).

Methods in RPC Opnum Order

| Method | Description |
| --- | --- |
| [LsarClose](#Section_be232cec04a9439ab718931c17c3ff0f) | Frees the resources held by a context handle.  Opnum: 0 |
| Opnum1NotUsedOnWire | Opnum: 1 |
| Lsar\_LSA\_DP\_2 | Opnum: 2 |
| Lsar\_LSA\_DP\_3 | Opnum: 3 |
| Lsar\_LSA\_DP\_4 | Opnum: 4 |
| Opnum5NotUsedOnWire | Opnum: 5 |
| [LsarOpenPolicy](#Section_534c8a461be740659f87e60a1447ae45) | Opens a context handle to the [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553).  Opnum: 6 |
| Lsar\_LSA\_DP\_7 | Opnum: 7 |
| Lsar\_LSA\_DP\_8 | Opnum: 8 |
| Opnum9NotUsedOnWire | Opnum: 9 |
| Lsar\_LSA\_DP\_10 | Opnum: 10 |
| Lsar\_LSA\_DP\_11 | Opnum: 11 |
| Lsar\_LSA\_DP\_12 | Opnum: 12 |
| Lsar\_LSA\_DP\_13 | Opnum: 13 |
| [LsarLookupNames](#Section_65d18faa0cb240eea94a2140212f4ec4) | Translates a batch of [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) names. For information on selecting which version to use, see section [3](#Section_b208b01e3fe64c6db0bd8ecf412e1b87).  Opnum: 14 |
| [LsarLookupSids](#Section_eb7ac899e697488393de1e60c7720c02) | Translates a batch of security principal [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d). For information on selecting which version to use, see section 3.  Opnum: 15 |
| Lsar\_LSA\_DP\_16 | Opnum: 16 |
| Lsar\_LSA\_DP\_17 | Opnum: 17 |
| Lsar\_LSA\_DP\_18 | Opnum: 18 |
| Lsar\_LSA\_DP\_19 | Opnum: 19 |
| Lsar\_LSA\_DP\_20 | Opnum: 20 |
| Opnum21NotUsedOnWire | Opnum: 21 |
| Opnum22NotUsedOnWire | Opnum: 22 |
| Lsar\_LSA\_DP\_23 | Opnum: 23 |
| Lsar\_LSA\_DP\_24 | Opnum: 24 |
| Lsar\_LSA\_DP\_25 | Opnum: 25 |
| Lsar\_LSA\_DP\_26 | Opnum: 26 |
| Lsar\_LSA\_DP\_27 | Opnum: 27 |
| Lsar\_LSA\_DP\_28 | Opnum: 28 |
| Lsar\_LSA\_DP\_29 | Opnum: 29 |
| Lsar\_LSA\_DP\_30 | Opnum: 30 |
| Lsar\_LSA\_DP\_31 | Opnum: 31 |
| Lsar\_LSA\_DP\_32 | Opnum: 32 |
| Lsar\_LSA\_DP\_33 | Opnum: 33 |
| Lsar\_LSA\_DP\_34 | Opnum: 34 |
| Lsar\_LSA\_DP\_35 | Opnum: 35 |
| Lsar\_LSA\_DP\_36 | Opnum: 36 |
| Lsar\_LSA\_DP\_37 | Opnum: 37 |
| Lsar\_LSA\_DP\_38 | Opnum: 38 |
| Lsar\_LSA\_DP\_39 | Opnum: 39 |
| Lsar\_LSA\_DP\_40 | Opnum: 40 |
| Lsar\_LSA\_DP\_41 | Opnum: 41 |
| Lsar\_LSA\_DP\_42 | Opnum: 42 |
| Lsar\_LSA\_DP\_43 | Opnum: 43 |
| [LsarOpenPolicy2](#Section_40a316ae1bae4aa192cd650cdaae1863) | Opens a context handle to the RPC server.  Opnum: 44 |
| [LsarGetUserName](#Section_4e3f0a3f86ae4d50aa40ec25013a89e3) | Returns the name and the [**domain name**](#gt_45a1c9f1-0263-49a8-97c7-7aca1a99308c) of a security principal.  Opnum: 45 |
| Lsar\_LSA\_DP\_46 | Opnum: 46 |
| Lsar\_LSA\_DP\_47 | Opnum: 47 |
| Lsar\_LSA\_DP\_48 | Opnum: 48 |
| Lsar\_LSA\_DP\_49 | Opnum: 49 |
| Lsar\_LSA\_DP\_50 | Opnum: 50 |
| Lsar\_LSA\_DP\_51 | Opnum: 51 |
| Opnum52NotUsedOnWire | Opnum: 52 |
| Lsar\_LSA\_DP\_53 | Opnum: 53 |
| Lsar\_LSA\_DP\_54 | Opnum: 54 |
| Lsar\_LSA\_DP\_55 | Opnum: 55 |
| Opnum56NotUsedOnWire | Opnum: 56 |
| [LsarLookupSids2](#Section_a09480f4145e4b4ca0275b313db61833) | Translates a batch of security principal SIDs. For information on selecting which version to use, see section 3.  Opnum: 57 |
| [LsarLookupNames2](#Section_43254e9f32e248caa61de8decf41721e) | Translates a batch of security principal names. For information on selecting which version to use, see section 3.  Opnum: 58 |
| Lsar\_LSA\_DP\_59 | Opnum: 59 |
| Opnum60NotUsedOnWire | Opnum: 60 |
| Opnum61NotUsedOnWire | Opnum: 61 |
| Opnum62NotUsedOnWire | Opnum: 62 |
| Opnum63NotUsedOnWire | Opnum: 63 |
| Opnum64NotUsedOnWire | Opnum: 64 |
| Opnum65NotUsedOnWire | Opnum: 65 |
| Opnum66NotUsedOnWire | Opnum: 66 |
| Opnum67NotUsedOnWire | Opnum: 67 |
| [LsarLookupNames3](#Section_678dd7ed611b4d948f5ca06c76ae5c7e) | Translates a batch of security principal names. For information on selecting which version to use, see section 3.  Opnum: 68 |
| Opnum69NotUsedOnWire | Opnum: 69 |
| Opnum70NotUsedOnWire | Opnum: 70 |
| Opnum71NotUsedOnWire | Opnum: 71 |
| Opnum72NotUsedOnWire | Opnum: 72 |
| Lsar\_LSA\_DP\_73 | Opnum: 73 |
| Lsar\_LSA\_DP\_74 | Opnum: 74 |
| Opnum75NotUsedOnWire | Opnum: 75 |
| LsarLookupSids3 | Translates a batch of security principal SIDs. For information on selecting which version to use, see section 3.  Opnum: 76 |
| [LsarLookupNames4](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361) | Translates a batch of security principal names. For information on selecting which version to use, see section 3.  Opnum: 77 |

**Note**  Gaps in the [**opnum**](#gt_e127848e-c66d-427d-b3aa-9f904fa4ada7) numbering sequence represent opnums of methods that are documented in [MS-LSAD], or opnums that MUST NOT[<23>](#Appendix_A_23" \o "Product behavior note 23) be used over the wire.

No exceptions SHOULD be thrown beyond those thrown by the underlying [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) protocol [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15).

The return values of all methods MUST conform to the specification of NTSTATUS, as specified in [[MS-ERREF]](%5bMS-ERREF%5d.pdf#Section_1bc92ddfb79e413cbbaa99a5281a6c90) section 2.3. Specific return values for normative processing conditions are specified in this document in the subsections of this section.

Unless otherwise specified, all negative values returned by an implementation are treated equivalently by the client as a message processing error. Unless otherwise specified, all non-negative values returned by an implementation are treated equivalently by the client as a success (of message processing).

Return values for implementation-specific conditions are left to the implementer's discretion, subject to the constraints specified in [MS-ERREF]. For example, an implementation can re-use an existing value in [MS-ERREF], such as 0xC0000017 (no memory).

The RPC methods shown in the following sections are organized in the following order:

* Methods that allow a client to open and close a connection.
* Methods that translate given names to SIDs.
* Methods that translate SIDs to names.

#### LsarOpenPolicy2 (Opnum 44)

The LsarOpenPolicy2 method opens a context handle to the [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553).

1. NTSTATUS LsarOpenPolicy2(
2. [in, unique, string] wchar\_t\* SystemName,
3. [in] PLSAPR\_OBJECT\_ATTRIBUTES ObjectAttributes,
4. [in] ACCESS\_MASK DesiredAccess,
5. [out] LSAPR\_HANDLE\* PolicyHandle
6. );

Processing rules for this message are defined in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 3.1.4.4.1.

#### LsarOpenPolicy (Opnum 6)

The LsarOpenPolicy method is exactly the same as [LsarOpenPolicy2](#Section_40a316ae1bae4aa192cd650cdaae1863), except that the *SystemName* parameter in this method contains only one character instead of a full string. This is because its syntactical definition lacks the [string] [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) annotation present in LsarOpenPolicy2, as specified in [[C706]](https://go.microsoft.com/fwlink/?LinkId=89824). RPC data types are specified in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) section 2.2.4.1.

The *SystemName* parameter has no effect on message processing in any environment. It MUST be ignored.

1. NTSTATUS LsarOpenPolicy(
2. [in, unique] wchar\_t\* SystemName,
3. [in] PLSAPR\_OBJECT\_ATTRIBUTES ObjectAttributes,
4. [in] ACCESS\_MASK DesiredAccess,
5. [out] LSAPR\_HANDLE\* PolicyHandle
6. );

Processing rules for this message are defined in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 3.1.4.4.2.

#### LsarClose (Opnum 0)

The LsarClose method frees the resources held by a context handle that was opened earlier. After response, the context handle is no longer usable, and any subsequent uses of this handle MUST fail.

1. NTSTATUS LsarClose(
2. [in, out] LSAPR\_HANDLE\* ObjectHandle
3. );

Processing rules for this message are defined in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc) section 3.1.4.9.4.

#### LsarGetUserName (Opnum 45)

The LsarGetUserName method returns the name and the [**domain name**](#gt_45a1c9f1-0263-49a8-97c7-7aca1a99308c) of the [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) that is invoking the method.

1. NTSTATUS LsarGetUserName(
2. [in, unique, string] wchar\_t\* SystemName,
3. [in, out] PRPC\_UNICODE\_STRING\* UserName,
4. [in, out, unique] PRPC\_UNICODE\_STRING\* DomainName
5. );

**SystemName:** This parameter has no effect on message processing in any environment. It MUST be ignored.

**UserName:** On return, contains the name of the security principal that is making the call. The string MUST be of the form sAMAccountName. On input, this parameter MUST be ignored. The RPC\_UNICODE\_STRING structure is defined in [[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) section 2.3.10.

**DomainName:** On return, contains the domain name of the security principal that is invoking the method. This string MUST be a NetBIOS name. On input, this parameter MUST be ignored.

**Return Values:** The following table contains a summary of the return values that an implementation MUST return, as specified by the message processing shown after the table.

| Return value/code | Description |
| --- | --- |
| 0x00000000  STATUS\_SUCCESS | The request was successfully completed. |
| 0xC0000022  STATUS\_ACCESS\_DENIED | The caller does not have the permissions to perform this operation. |

The server MUST determine the [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) of the caller; to do so, the server MUST invoke the GetRpcImpersonationAccessToken abstract interface ([[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) section 3.3.3.4.3.1).

If GetRpcImpersonationAccessToken succeeds, the server MUST use the SID in the Token.Sids[OwnerIndex] element ([MS-DTYP] section 2.5.2) for further processing.

If GetRpcImpersonationAccessToken fails, the server MUST use the ANONYMOUS SID ([MS-DTYP] section 2.4.2.4) for further processing.

The server MUST locate the security principal that is making the request using the SID that was determined previously. To do so, a search MUST be performed in the following views and MUST end as soon as the security principal is located in some view:

* Predefined Translation View.
* Configurable Translation View.
* Account Domain View of the account database served on that machine.
  + If the machine is not joined to a [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca), the search ends here.
* If this machine is not a [**domain controller**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd): Account Domain View of the domain to which this machine is joined.
* Forest View of the [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62) of the domain to which this machine is joined.
* Forest Views of [**trusted forests**](#gt_3b76a71f-9697-4836-9c69-09899b23c21b) for the forest of the domain to which this machine is joined.
* Account Domain Views of externally [**trusted domains**](#gt_f2f00d47-6cf2-4b32-b8f7-b63e38e2e9c4) for the domain to which this machine is joined.

After the security principal is located, the [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553) MUST return the security principal name in the *UserName* parameter and MUST return the domain NetBIOS name in the *DomainName* parameter if *DomainName* is not NULL. The return value MUST be set to STATUS\_SUCCESS in this case. In other cases, an implementation-specific negative value MUST be returned.

#### LsarLookupNames4 (Opnum 77)

The LsarLookupNames4 method translates a batch of [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) names to their [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) form. It also returns the [**domains**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) of which these security principals are a part.

1. NTSTATUS LsarLookupNames4(
2. [in] handle\_t RpcHandle,
3. [in, range(0,1000)] unsigned long Count,
4. [in, size\_is(Count)] PRPC\_UNICODE\_STRING Names,
5. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST\* ReferencedDomains,
6. [in, out] PLSAPR\_TRANSLATED\_SIDS\_EX2 TranslatedSids,
7. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
8. [in, out] unsigned long\* MappedCount,
9. [in] unsigned long LookupOptions,
10. [in] unsigned long ClientRevision
11. );

**RpcHandle:** This value is used by [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) internally and is not transmitted over the network, as specified in [[C706]](https://go.microsoft.com/fwlink/?LinkId=89824). This handle can be obtained by calling RPC runtime binding routines. For more information, see [[MSDN-RPCDB]](https://go.microsoft.com/fwlink/?LinkId=90076).

**Count:** Number of security principal names to look up.[<24>](#Appendix_A_24" \o "Product behavior note 24)

**Names:** Contains the security principal names to translate. The RPC\_UNICODE\_STRING structure is defined in [[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) section 2.3.10.

The following name forms MUST be supported:

* [**User principal names (UPNs)**](#gt_9d606f55-b798-4def-bf96-97b878bb92c6), such as user\_name@example.example.com.
* Fully qualified account names based on either DNS or NetBIOS names. For example: example.example.com\user\_name or example\user\_name, where the generalized form is domain\user account name, and domain is either the fully qualified [**DNS name**](#gt_102a36e2-f66f-49e2-bee3-558736b2ecd5) or the NetBIOS name of the [**trusted domain**](#gt_f2f00d47-6cf2-4b32-b8f7-b63e38e2e9c4).
* Unqualified or isolated names, such as user\_name.

The comparisons used by the [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553) MUST NOT be case-sensitive, so case for inputs is not important.

**ReferencedDomains:** On successful return, contains the domain information for the domain to which each security principal belongs. The domain information includes a NetBIOS domain name and a domain SID for each entry in the list.

**TranslatedSids:** On successful return, contains the corresponding SID form for security principal names in the *Names* parameter. It MUST be ignored on input.

**LookupLevel:** Specifies what scopes are to be used during translation, as specified in section [2.2.16](#Section_9d1166ccbcfd4e22a8acf55eae57c99f).

**MappedCount:** On successful return, contains the number of names that are translated completely to the SID form. This parameter is left as an input parameter for backward compatibility and has no effect on message processing in any environment. It MUST be ignored on input.

**LookupOptions:** Flags specified by the caller that control the lookup operation. The value MUST be one of the following.

| Value | Meaning |
| --- | --- |
| 0x00000000 | Isolated names are searched for even when they are not on the local computer. |
| 0x80000000 | Isolated names (except for user principal names (UPNs)) are searched for only on the local account database. UPNs are not searched for in any of the views. |

**ClientRevision:** Version of the client, which implies the client's capabilities. The value MUST be one of the following.[<25>](#Appendix_A_25" \o "Product behavior note 25)

| Value | Meaning |
| --- | --- |
| 0x00000001 | The client does not understand DNS domain names and is not aware that it might be part of a [**forest**](#gt_fd104241-4fb3-457c-b2c4-e0c18bb20b62). |
| 0x00000002 | The client understands DNS domain names and is aware that it might be part of a forest.  Error codes returned can include STATUS\_TRUSTED\_DOMAIN\_FAILURE and STATUS\_TRUSTED\_RELATIONSHIP\_FAILURE, which are not returned for *ClientRevision* of 0x00000001. For more information on error codes, see [[MS-ERREF]](%5bMS-ERREF%5d.pdf#Section_1bc92ddfb79e413cbbaa99a5281a6c90). |

Values smaller than 0x00000002 SHOULD be assumed to be 0x00000001, and other values SHOULD be assumed to be 0x00000002.

**Return Values:** The following table contains a summary of the return values that an implementation MUST return, as specified by the message processing shown after the table.

| Return value/code | Description |
| --- | --- |
| 0x00000000  STATUS\_SUCCESS | The request was successfully completed. |
| 0x00000107  STATUS\_SOME\_NOT\_MAPPED | Some of the information to be translated has not been translated. |
| 0xC00000DC  STATUS\_INVALID\_SERVER\_STATE | The RPC server is not a [**domain controller**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd). |
| 0xC0000022  STATUS\_ACCESS\_DENIED | The caller does not have the permissions to perform this operation. |
| 0xC000000D  STATUS\_INVALID\_PARAMETER | One of the supplied parameters was invalid. |
| 0xC0000073  STATUS\_NONE\_MAPPED | None of the information to be translated has been translated. |

This message is valid only if the RPC server is a domain controller. The RPC server MUST return STATUS\_INVALID\_SERVER\_STATE in the return value if it is not a domain controller.

The RPC server MUST ensure that the RPC\_C\_AUTHN\_NETLOGON security provider (as specified in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) section 2.2.1.1.7) and at least RPC\_C\_AUTHN\_LEVEL\_PKT\_INTEGRITY authentication level (as specified in [MS-RPCE] section 2.2.1.1.8) are used in this RPC message. Otherwise, the RPC server MUST return STATUS\_ACCESS\_DENIED.

The RPC server MUST check each element in the *Names* parameter for validity, as described for the RPC\_UNICODE\_STRING structure in [MS-DTYP] section 2.3.10. If validation fails, the RPC server MUST return STATUS\_INVALID\_PARAMETER.

The *TranslatedSids* and *MappedCount* parameters are left as input parameters for backward compatibility and have no effect on message processing in any environment. They MUST be ignored on input.

*LookupOptions* MUST NOT contain 0x80000000 unless *LookupLevel* is LsapLookupWksta. If this fails, the RPC server MUST return STATUS\_INVALID\_PARAMETER.

A name is interchangeably called a "qualified" or "composite" name if the name has a backslash ('\') character in it. The string before the backslash is assumed to be the domain name, and the string after the backslash is the name to be resolved within that domain. If the name does not contain a backslash character, it is called an "isolated" name. An isolated name can be in single-label form, as in "John", or can be in user-principal-name form, as in "John@example.com".

The RPC server MUST match a name with a row in the view by using the following rules:

* If the name is a qualified name, the domain name MUST match the Domain NetBIOS Name column or Domain DNS Name column, and the name MUST match the Security Principal Name column. Note that there cannot be multiple matches in this case (see section [3.1.1.1](#Section_77855eedd3c74538b52720a36d4dcb18)). A domain name match against the Domain DNS Name column MUST NOT be performed if *ClientRevision* is 0x00000001 and the machine is joined to a mixed mode domain, as specified in [[MS-ADTS]](%5bMS-ADTS%5d.pdf#Section_d243592709994c628c6d13ba31a52e1a) section 6.1.4.1.
* If the name is an isolated name:
  + If in single-label form, the name MUST match the Security Principal Name column or the Additional Security Principal Name column.
    - If there are multiple matches due to a search in multiple domains, which can happen in Forest Views, any one of the matches MAY be used to satisfy the request, as long as a consistent search order is followed.
  + If in user principal name form, the name SHOULD[<26>](#Appendix_A_26" \o "Product behavior note 26) match the User Principal Name column or the Default User Principal Names column.
    - If there are multiple matches on the User Principal Name attribute, the match MUST fail.
    - If there is one match with User Principal Name and one match with Default User Principal Names, the match with the User Principal Name attribute MUST take precedence.
    - There cannot be multiple Default User Principal Names matches; see section 3.1.1.1.

The search scope is defined by the *LookupLevel* parameter. The RPC server MUST search only the scopes that are defined by the *LookupLevel* parameter and MUST NOT return matches defined outside that scope. The scopes to search when processing this message are specified in section 2.2.16.

When a name is matched in the database, or when the domain name of a composite name is matched in the database, the *ReferencedDomains* parameter MUST be searched for an entry that is identical to the following: The **Name** field contains the value from the Domain NetBIOS Name column, and the **Sid** field contains the value from the Domain SID column. If there is no such entry, a new entry with the **Name** field containing the value of the Domain NetBIOS Name and the **Sid** field containing the value of the Domain SID MUST be created and added to the *ReferencedDomains* list.

Output parameters MUST be updated using the following information:

1. For a successful search, the corresponding *TranslatedSids* entry MUST be updated as follows:
   * Use: Security Principal Type column
   * Sid: Security Principal SID column
   * Flags:
     + 0x00000001: If the search is not satisfied by matching against the Security Principal Name column. Also see the paragraph after step 3.
     + 0x00000004: If the search is satisfied by a comparison that uses the view specified in section [3.1.1.1.2](#Section_252cb6b02edc44039d755b2786ba859e), Configurable Translation Database and Corresponding View. Also see the paragraph after step 3.
   * DomainIndex: Index of the domain in the *ReferencedDomains* parameter.
2. If a match cannot be found for a name, the corresponding *TranslatedSids* entry MUST be updated with:
   * Use: SidTypeUnknown
   * Sid: NULL
   * Flags: 0x00000000 (also see the paragraph after step 3)
   * DomainIndex: -1
3. If a match cannot be found for a composite name, but the domain name within the composite name is found, the corresponding *TranslatedSids* entry MUST be updated with:
   * Use: SidTypeUnknown
   * Sid: NULL
   * Flags: 0x00000000 (also see the following paragraph)
   * DomainIndex: Index of the domain in the *ReferencedDomains* parameter.

In all preceding cases, the **Flags** field MUST contain 0x00000002 when:

* *LookupLevel* is LsapLookupWksta, LsapLookupPDC, LsapLookupGC, LsapLookupXForestReferral, or LsapLookupXForestResolve, and
* The IsDomainNameInTrustedForest algorithm defined in [[MS-DRSR]](%5bMS-DRSR%5d.pdf#Section_f977faaa673e4f66b9bf48c640241d47) section 5.64.2 (Determining If a Name Is in a Trusted Forest), given the domain name, returns true, or the IsUPNInTrustedForest algorithm defined in [MS-DRSR] section 5.109, given the full name to be translated, returns true.

The return value MUST be set to STATUS\_SUCCESS if all names are translated correctly.

If some names are translated correctly but some are not, the return value MUST be set to STATUS\_SOME\_NOT\_MAPPED.

If none of the names are translated correctly, the return value SHOULD[<27>](#Appendix_A_27" \o "Product behavior note 27) be set to STATUS\_NONE\_MAPPED.

If *LookupLevel* is LsapLookupWksta, and the return code can be identified as an error value (that is, less than 0) other than STATUS\_NONE\_MAPPED, *ReferencedDomains* and the **Sids** field in the *TranslatedSids* structure MUST NOT be returned.

#### LsarLookupNames3 (Opnum 68)

The LsarLookupNames3 method translates a batch of [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) names to their [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) form. It also returns the [**domains**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) that these names are a part of.[<28>](#Appendix_A_28" \o "Product behavior note 28)

1. NTSTATUS LsarLookupNames3(
2. [in] LSAPR\_HANDLE PolicyHandle,
3. [in, range(0,1000)] unsigned long Count,
4. [in, size\_is(Count)] PRPC\_UNICODE\_STRING Names,
5. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST\* ReferencedDomains,
6. [in, out] PLSAPR\_TRANSLATED\_SIDS\_EX2 TranslatedSids,
7. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
8. [in, out] unsigned long\* MappedCount,
9. [in] unsigned long LookupOptions,
10. [in] unsigned long ClientRevision
11. );

**PolicyHandle:** Context handle obtained by an [LsarOpenPolicy](#Section_534c8a461be740659f87e60a1447ae45) or [LsarOpenPolicy2](#Section_40a316ae1bae4aa192cd650cdaae1863) call.

**Count:** Number of security principal names to look up.[<29>](#Appendix_A_29" \o "Product behavior note 29)

**Names:** Contains the security principal names to translate, as specified in section [3.1.4.5](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361).

**ReferencedDomains:** On successful return, contains the domain information for the domain to which each security principal belongs. The domain information includes a NetBIOS domain name and a domain SID for each entry in the list.

**TranslatedSids:** On successful return, contains the corresponding SID forms for security principal names in the *Names* parameter. It MUST be ignored on input.

**LookupLevel:** Specifies what scopes are to be used during translation, as specified in section [2.2.16](#Section_9d1166ccbcfd4e22a8acf55eae57c99f).

**MappedCount:** On successful return, contains the number of names that are translated completely to the SID form. This parameter has no effect on message processing in any environment. It MUST be ignored on input.

**LookupOptions:** Flags that control the lookup operation. For possible values and their meanings, see section 3.1.4.5.

**ClientRevision:** Version of the client, which implies the client's capabilities. For possible values and their meanings, see section 3.1.4.5.

**Return Values:** The following table contains a summary of the return values that an implementation MUST return, as specified by the message processing shown after the table.

| Return value/code | Description |
| --- | --- |
| 0x00000000  STATUS\_SUCCESS | The request was successfully completed. |
| 0x00000107  STATUS\_SOME\_NOT\_MAPPED | Some of the information to be translated has not been translated. |
| 0xC0000022  STATUS\_ACCESS\_DENIED | The caller does not have the permissions to perform this operation. |
| 0xC000000D  STATUS\_INVALID\_PARAMETER | One of the supplied parameters was invalid. |
| 0xC0000073  STATUS\_NONE\_MAPPED | None of the information to be translated has been translated. |

The behavior required when receiving an LsarLookupNames3 message MUST be identical to that when receiving an LsarLookupNames4 message, with the following exceptions:

* This message is valid on non–[**domain controller**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) machines as well as domain controllers.
* The server MUST return STATUS\_ACCESS\_DENIED if neither of the following conditions are true:
  1. The RPC\_C\_AUTHN\_NETLOGON security provider (as specified in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) section 2.2.1.1.7) and at least RPC\_C\_AUTHN\_LEVEL\_PKT\_INTEGRITY authentication level (as specified in [MS-RPCE] section 2.2.1.1.8) were used in this [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) message.
  2. The PolicyHandle was granted POLICY\_LOOKUP\_NAMES access.

#### LsarLookupNames2 (Opnum 58)

The LsarLookupNames2 method translates a batch of [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) names to their [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) form. It also returns the [**domains**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) that these names are a part of.[<30>](#Appendix_A_30" \o "Product behavior note 30)

1. NTSTATUS LsarLookupNames2(
2. [in] LSAPR\_HANDLE PolicyHandle,
3. [in, range(0,1000)] unsigned long Count,
4. [in, size\_is(Count)] PRPC\_UNICODE\_STRING Names,
5. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST\* ReferencedDomains,
6. [in, out] PLSAPR\_TRANSLATED\_SIDS\_EX TranslatedSids,
7. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
8. [in, out] unsigned long\* MappedCount,
9. [in] unsigned long LookupOptions,
10. [in] unsigned long ClientRevision
11. );

**PolicyHandle:** Context handle obtained by an [LsarOpenPolicy](#Section_534c8a461be740659f87e60a1447ae45) or [LsarOpenPolicy2](#Section_40a316ae1bae4aa192cd650cdaae1863) call.

**Count:** Number of security principal names to look up.[<31>](#Appendix_A_31" \o "Product behavior note 31)

**Names:** Contains the security principal names to translate, as specified in section [3.1.4.5](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361).

**ReferencedDomains:** On successful return, contains the domain information for the domain to which each security principal belongs. The domain information includes a NetBIOS domain name and a domain SID for each entry in the list.

**TranslatedSids:** On successful return, contains the corresponding SID forms for security principal names in the *Names* parameter. It MUST be ignored on input.

**LookupLevel:** Specifies what scopes are to be used during translation, as specified in section [2.2.16](#Section_9d1166ccbcfd4e22a8acf55eae57c99f).

**MappedCount:** On successful return, contains the number of names that are translated completely to the SID form. This parameter has no effect on message processing in any environment. It MUST be ignored on input.

**LookupOptions:** Flags that control the lookup operation. For possible values and their meanings, see section 3.1.4.5.

**ClientRevision:** Version of the client, which implies the client's capabilities. For possible values and their meanings, see section 3.1.4.5.

**Return Values:** The following table contains a summary of the return values that an implementation MUST return, as specified by the message processing shown after the table.

| Return value/code | Description |
| --- | --- |
| 0x00000000  STATUS\_SUCCESS | The request was successfully completed. |
| 0x00000107  STATUS\_SOME\_NOT\_MAPPED | Some of the information to be translated has not been translated. |
| 0xC0000022  STATUS\_ACCESS\_DENIED | The caller does not have the permissions to perform this operation. |
| 0xC000000D  STATUS\_INVALID\_PARAMETER | One of the supplied parameters was invalid. |
| 0xC0000073  STATUS\_NONE\_MAPPED | None of the information to be translated has been translated. |

The behavior required when receiving an LsarLookupNames2 message MUST be identical to that when receiving an [LsarLookupNames3](#Section_b75eaac5e4d14fc08dae61d838b38701) message, with the following exceptions:

* Elements in the *TranslatedSids* output structure do not contain a **Sid** field; instead, they contain a **RelativeId** field. **RelativeId** MUST be set to the last element in the **SubAuthority** array of the RPC\_SID structure ([[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) section 2.4.2.3) if the translated SID is not of type SidTypeDomain, and if the **Flags** field does not contain 0x00000004. If the translated SID is of type SidTypeDomain or the **Flags** field contains 0x00000004, **RelativeId** MUST be set to 0xFFFFFFFF.
* The *LookupOptions* and *ClientRevision* parameters MUST be ignored. Message processing MUST happen as if *LookupOptions* is set to 0x00000000 and *ClientRevision* is set to 0x00000002.
* The server MUST return STATUS\_ACCESS\_DENIED if neither of the following conditions is true:
  1. The RPC\_C\_AUTHN\_NETLOGON security provider (as specified in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) section 2.2.1.1.7) and at least RPC\_C\_AUTHN\_LEVEL\_PKT\_INTEGRITY authentication level (as specified in [MS-RPCE] section 2.2.1.1.8) were used in this [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) message.
  2. The PolicyHandle was granted POLICY\_LOOKUP\_NAMES access.

#### LsarLookupNames (Opnum 14)

The LsarLookupNames method translates a batch of [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) names to their [**SID**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) form. It also returns the [**domains**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) that these names are a part of.

1. NTSTATUS LsarLookupNames(
2. [in] LSAPR\_HANDLE PolicyHandle,
3. [in, range(0,1000)] unsigned long Count,
4. [in, size\_is(Count)] PRPC\_UNICODE\_STRING Names,
5. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST\* ReferencedDomains,
6. [in, out] PLSAPR\_TRANSLATED\_SIDS TranslatedSids,
7. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
8. [in, out] unsigned long\* MappedCount
9. );

**PolicyHandle:** Context handle obtained by an [LsarOpenPolicy](#Section_534c8a461be740659f87e60a1447ae45) or [LsarOpenPolicy2](#Section_40a316ae1bae4aa192cd650cdaae1863) call.

**Count:** Number of names in the Names array.[<32>](#Appendix_A_32" \o "Product behavior note 32)

**Names:** Contains the security principal names to translate, as specified in section [3.1.4.5](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361).

**ReferencedDomains:** On successful return, contains the domain information for the domain to which each security principal belongs. The domain information includes a NetBIOS domain name and a domain SID for each entry in the list.

**TranslatedSids:** On successful return, contains the corresponding SID forms for security principal names in the *Names* parameter. It MUST be ignored on input.

**LookupLevel:** Specifies what scopes are to be used during translation, as specified in section [2.2.16](#Section_9d1166ccbcfd4e22a8acf55eae57c99f).

**MappedCount:** On successful return, contains the number of names that are translated completely to their SID forms. This parameter has no effect on message processing in any environment. It MUST be ignored on input.

**Return Values:** The following table contains a summary of the return values that an implementation MUST return, as specified by the message processing shown after the table.

| Return value/code | Description |
| --- | --- |
| 0x00000000  STATUS\_SUCCESS | The request was successfully completed. |
| 0x00000107  STATUS\_SOME\_NOT\_MAPPED | Some of the information to be translated has not been translated. |
| 0xC0000022  STATUS\_ACCESS\_DENIED | The caller does not have the permissions to perform this operation. |
| 0xC000000D  STATUS\_INVALID\_PARAMETER | One of the supplied parameters was invalid. |
| 0xC0000073  STATUS\_NONE\_MAPPED | None of the information to be translated has been translated. |

The behavior required when receiving an LsarLookupNames message MUST be identical to that when receiving an [LsarLookupNames2](#Section_43254e9f32e248caa61de8decf41721e) message, with the following exceptions:

* Elements in the TranslatedSids output structure do not contain a **Flags** field.
* Due to the absence of the *LookupOptions* and *ClientRevision* parameters, the [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553) MUST assume that *LookupOptions* is 0 and *ClientRevision* is 1.
* The server MUST return STATUS\_ACCESS\_DENIED if neither of the following conditions is true:
  1. The RPC\_C\_AUTHN\_NETLOGON security provider (as specified in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) section 2.2.1.1.7) and at least RPC\_C\_AUTHN\_LEVEL\_PKT\_INTEGRITY authentication level (as specified in [MS-RPCE] section 2.2.1.1.8) were used in this [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) message.
  2. The PolicyHandle was granted POLICY\_LOOKUP\_NAMES access.

#### LsarLookupSids3 (Opnum 76)

The LsarLookupSids3 method translates a batch of [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) to their name forms. It also returns the [**domains**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) that these names are a part of.

1. NTSTATUS LsarLookupSids3(
2. [in] handle\_t RpcHandle,
3. [in] PLSAPR\_SID\_ENUM\_BUFFER SidEnumBuffer,
4. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST\* ReferencedDomains,
5. [in, out] PLSAPR\_TRANSLATED\_NAMES\_EX TranslatedNames,
6. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
7. [in, out] unsigned long\* MappedCount,
8. [in] unsigned long LookupOptions,
9. [in] unsigned long ClientRevision
10. );

**RpcHandle:** An [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) binding handle, as described in [[C706]](https://go.microsoft.com/fwlink/?LinkId=89824). RPC binding handles are used by RPC internally and are not transmitted over the network.

This handle can be obtained by calling RPC runtime binding routines. For more information, see [[MSDN-RPCDB]](https://go.microsoft.com/fwlink/?LinkId=90076).

**SidEnumBuffer:** Contains the SIDs to be translated. The SIDs in this structure can be that of users, groups, computers, Windows-defined well-known security principals, or domains.

**ReferencedDomains:** On successful return, contains the domain information for the domain to which each security principal belongs. The domain information includes a NetBIOS domain name and a domain SID for each entry in the list.

**TranslatedNames:** On successful return, contains the corresponding name forms for security principal SIDs in the *SidEnumBuffer* parameter. It MUST be ignored on input.

**LookupLevel:** Specifies what scopes are to be used during translation, as specified in section [2.2.16](#Section_9d1166ccbcfd4e22a8acf55eae57c99f).

**MappedCount:** On successful return, contains the number of names that are translated completely to their name forms. It MUST be ignored on input.

**LookupOptions:** Flags that control the lookup operation. This parameter is reserved for future use; it MUST be set to 0 and ignored on receipt.

**ClientRevision:** Version of the client, which implies the client's capabilities. For possible values and their meanings, see section [3.1.4.5](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361).

**Return Values:** The following table contains a summary of the return values that an implementation MUST return, as specified by the message processing shown after the table.

| Return value/code | Description |
| --- | --- |
| 0x00000000  STATUS\_SUCCESS | The request was successfully completed. |
| 0x00000107  STATUS\_SOME\_NOT\_MAPPED | Some of the information to be translated has not been translated. |
| 0xC00000DC  STATUS\_INVALID\_SERVER\_STATE | The [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553) is not a [**domain controller**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd). |
| 0xC0000022  STATUS\_ACCESS\_DENIED | The caller does not have the permissions to perform this operation. |
| 0xC000000D  STATUS\_INVALID\_PARAMETER | One of the supplied parameters was invalid. |
| 0xC0000073  STATUS\_NONE\_MAPPED | None of the information to be translated has been translated. |

This message is valid only if the RPC server is a domain controller. The RPC server MUST return STATUS\_INVALID\_SERVER\_STATE in the return value if it is not a domain controller.

The RPC server MUST ensure that the RPC\_C\_AUTHN\_NETLOGON security provider (as specified in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) section 2.2.1.1.7) and at least RPC\_C\_AUTHN\_LEVEL\_PKT\_INTEGRITY authentication level (as specified in [MS-RPCE] section 2.2.1.1.8) are used in this RPC message. Otherwise, the RPC server MUST return STATUS\_ACCESS\_DENIED.

The RPC server MUST check each element in the *SidEnumBuffer* parameter for validity, as specified in section [2.2.18](#Section_1ffb61f0a4fe4487858dfb709d605855), and individual elements in this structure MUST be checked for validity, as described for the SID structure in [[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) section 2.4.2. If validation fails, the RPC server SHOULD return STATUS\_INVALID\_PARAMETER.

The *TranslatedNames* and *MappedCount* parameters have no effect on message processing in any environment. They MUST be ignored on input.

The RPC server search scope can change depending on the *LookupLevel* parameter, as specified in section 2.2.16. In the views corresponding to each *LookupLevel*, SIDs MUST be searched under the Security Principal SID and Security Principal SID History columns.

Output parameters MUST be updated using the following information:

1. When a SID is found in the database, the *ReferencedDomains* parameter MUST be searched for an identical entry with the **Name** field containing the value from the Domain NetBIOS Name column and the **Sid** field containing the value from the Domain SID column. If there is no such entry, a new entry with the **Name** field containing the value of the Domain NetBIOS Name and the **Sid** field containing the value of the Domain SID MUST be created and added to the *ReferencedDomains* list.

For the successful search, the corresponding *TranslatedNames* entry MUST be updated as follows:

* + Use: Security Principal Type column
  + Name: Security Principal Name column
  + Flags:
    - 0x00000001: If the search is not satisfied by matching against the Security Principal SID column. Also see the paragraph after step 3.
    - 0x00000004: If the search is satisfied by matching in [Configurable Translation View (section 3.1.1.1.2)](#Section_252cb6b02edc44039d755b2786ba859e). Also see the paragraph after step 3.
  + DomainIndex: Index of the domain in the *ReferencedDomains* parameter.

1. If a match cannot be found for a SID, but the domain portion of the SID can be matched with a domain SID, the *ReferencedDomains* parameter MUST be searched for an identical entry with the **Name** field containing the value from the Domain NetBIOS Name column and the **Sid** field containing the value from the Domain SID column. If there is no such entry, a new entry with the **Name** field containing the value of the Domain NetBIOS Name and the **Sid** field containing the value of the Domain SID MUST be created and added to the *ReferencedDomains* list.

The corresponding *TranslatedNames* entry MUST be updated by using the following view:

* + Use: SidTypeUnknown.
  + Name: Empty unless *LookupLevel* is LsapLookupWksta, in which case this field must contain a textual representation of the last 32 bits of the corresponding SID in hexadecimal format.
  + Flags: 0x00000000 (also see the paragraph after step 3).
  + DomainIndex: Index of the domain in the *ReferencedDomains* parameter.

1. Otherwise, the corresponding *TranslatedNames* entry MUST be updated with:
   * Use: SidTypeUnknown.
   * Name: Empty, unless *LookupLevel* is LsapLookupWksta. In that case, Name MUST contain the textual representation of the corresponding SID, as in step 2.
   * Flags: 0x00000000 (also see the following paragraph).
   * DomainIndex: -1.

In all preceding cases, the **Flags** field MUST contain 0x00000002 when:

* *LookupLevel* is LsapLookupWksta, LsapLookupPDC, LsapLookupGC, LsapLookupXForestReferral, or LsapLookupXForestResolve, and
* The IsDomainSidInTrustedForest algorithm defined in [[MS-DRSR]](%5bMS-DRSR%5d.pdf#Section_f977faaa673e4f66b9bf48c640241d47) section 5.97 returns true.

The return value MUST be set to STATUS\_SUCCESS if all SIDs are translated correctly.

If some SIDs are translated correctly but some are not, the return value MUST be set to STATUS\_SOME\_NOT\_MAPPED.

If none of the SIDs are translated correctly, the return value MUST be set to STATUS\_NONE\_MAPPED.

If *LookupLevel* is LsapLookupWksta, and the return code can be identified as an error value other than STATUS\_NONE\_MAPPED, *ReferencedDomains* and the **Names** fields in the *TranslatedNames* structure MUST NOT be returned.

#### LsarLookupSids2 (Opnum 57)

The LsarLookupSids2 method translates a batch of [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) to their name forms. It also returns the [**domains**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) that these names are a part of.

1. NTSTATUS LsarLookupSids2(
2. [in] LSAPR\_HANDLE PolicyHandle,
3. [in] PLSAPR\_SID\_ENUM\_BUFFER SidEnumBuffer,
4. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST\* ReferencedDomains,
5. [in, out] PLSAPR\_TRANSLATED\_NAMES\_EX TranslatedNames,
6. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
7. [in, out] unsigned long\* MappedCount,
8. [in] unsigned long LookupOptions,
9. [in] unsigned long ClientRevision
10. );

**PolicyHandle:** Context handle obtained by an [LsarOpenPolicy](#Section_534c8a461be740659f87e60a1447ae45) or [LsarOpenPolicy2](#Section_40a316ae1bae4aa192cd650cdaae1863) call.

**SidEnumBuffer:** Contains the SIDs to be translated. The SIDs in this structure can be that of users, groups, computers, Windows-defined well-known security principals, or domains.

**ReferencedDomains:** On successful return, contains the domain information for the domain to which each security principal belongs. The domain information includes a NetBIOS domain name and a domain SID for each entry in the list.

**TranslatedNames:** On successful return, contains the corresponding name forms for security principal SIDs in the *SidEnumBuffer* parameter. It MUST be ignored on input.

**LookupLevel:** Specifies what scopes are to be used during translation, as specified in section [2.2.16](#Section_9d1166ccbcfd4e22a8acf55eae57c99f).

**MappedCount:** On return, contains the number of names that are translated completely to their name forms. It MUST be ignored on input.

**LookupOptions:** Flags that control the lookup operation. This parameter is reserved for future use and SHOULD[<33>](#Appendix_A_33" \o "Product behavior note 33) be set to 0.

**ClientRevision:** Version of the client, which implies the client's capabilities. For possible values and their meanings, see section [3.1.4.5](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361).

**Return Values:** The following table contains a summary of the return values that an implementation MUST return, as specified by the message processing shown after the table.

| Return value/code | Description |
| --- | --- |
| 0x00000000  STATUS\_SUCCESS | The request was successfully completed. |
| 0x00000107  STATUS\_SOME\_NOT\_MAPPED | Some of the information to be translated has not been translated. |
| 0xC0000022  STATUS\_ACCESS\_DENIED | The caller does not have the permissions to perform this operation. |
| 0xC000000D  STATUS\_INVALID\_PARAMETER | One of the supplied parameters was invalid. |
| 0xC0000073  STATUS\_NONE\_MAPPED | None of the information to be translated has been translated. |

The behavior required when receiving an LsarLookupSids2 message MUST be identical to that when receiving an [LsarLookupSids3](#Section_678dd7ed611b4d948f5ca06c76ae5c7e) message, with the following exceptions:

* This message is valid on non–[**domain controller**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) machines as well as domain controllers.
* *LookupLevel* values other than LsapLookupWksta are not valid if the [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553) is not a domain controller. For this condition, the RPC server MUST return an error in the return value.
* The server MUST return STATUS\_ACCESS\_DENIED if neither of the following conditions are true:
  1. The RPC\_C\_AUTHN\_NETLOGON security provider (as specified in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) section 2.2.1.1.7) and at least RPC\_C\_AUTHN\_LEVEL\_PKT\_INTEGRITY authentication level (as specified in [MS-RPCE] section 2.2.1.1.8) were used in this [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) message.
  2. The PolicyHandle was granted POLICY\_LOOKUP\_NAMES access.

#### LsarLookupSids (Opnum 15)

The LsarLookupSids method translates a batch of [**security principal**](#gt_f3ef2572-95cf-4c5c-b3c9-551fd648f409) [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) to their name forms. It also returns the [**domains**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) that these names are a part of.

1. NTSTATUS LsarLookupSids(
2. [in] LSAPR\_HANDLE PolicyHandle,
3. [in] PLSAPR\_SID\_ENUM\_BUFFER SidEnumBuffer,
4. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST\* ReferencedDomains,
5. [in, out] PLSAPR\_TRANSLATED\_NAMES TranslatedNames,
6. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
7. [in, out] unsigned long\* MappedCount
8. );

**PolicyHandle:** Context handle obtained by an [LsarOpenPolicy](#Section_534c8a461be740659f87e60a1447ae45) or [LsarOpenPolicy2](#Section_40a316ae1bae4aa192cd650cdaae1863) call.

**SidEnumBuffer:** Contains the SIDs to be translated. The SIDs in this structure can be that of users, groups, computers, Windows-defined well-known security principals, or domains.

**ReferencedDomains:** On successful return, contains the domain information for the domain to which each security principal belongs. The domain information includes a NetBIOS domain name and a domain SID for each entry in the list.

**TranslatedNames:** On successful return, contains the corresponding name form for security principal SIDs in the *SidEnumBuffer* parameter. It MUST be ignored on input.

**LookupLevel:** Specifies what scopes are to be used during translation, as specified in section [2.2.16](#Section_9d1166ccbcfd4e22a8acf55eae57c99f).

**MappedCount:** On successful return, contains the number of names that are translated completely to their Name forms. It MUST be ignored on input.

**Return Values:** The following table contains a summary of the return values that an implementation MUST return, as specified by the message processing shown after the table.

| Return value/code | Description |
| --- | --- |
| 0x00000000  STATUS\_SUCCESS | The request was successfully completed. |
| 0x00000107  STATUS\_SOME\_NOT\_MAPPED | Some of the information to be translated has not been translated. |
| 0xC0000022  STATUS\_ACCESS\_DENIED | The caller does not have the permissions to perform this operation. |
| 0xC000000D  STATUS\_INVALID\_PARAMETER | One of the supplied parameters was invalid. |
| 0xC0000073  STATUS\_NONE\_MAPPED | None of the information to be translated has been translated. |

The behavior required when receiving an LsarLookupSids message MUST be identical to that when receiving an [LsarLookupSids2](#Section_a09480f4145e4b4ca0275b313db61833) message, with the following exceptions:

* Elements in the TranslatedNames output structure do not contain a Flags field.
* Due to the absence of *LookupOptions* and *ClientRevision* parameters, the [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553) MUST assume that *LookupOptions* is 0 and *ClientRevision* is 1.
* The server MUST return STATUS\_ACCESS\_DENIED if neither of the following conditions is true:
  1. The RPC\_C\_AUTHN\_NETLOGON security provider (as specified in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) section 2.2.1.1.7) and at least RPC\_C\_AUTHN\_LEVEL\_PKT\_INTEGRITY authentication level (as specified in [MS-RPCE] section 2.2.1.1.8) were used in this [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) message.
  2. The PolicyHandle was granted POLICY\_LOOKUP\_NAMES access.

### Timer Events

No protocol timer events are required on the [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553) other than the timers required in the underlying [**RPC transport**](#gt_c2eeb200-3cd0-4916-966e-d7d6bff1737a).

### Other Local Events

No additional local events are used on the [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553) other than the events maintained in the underlying [**RPC transport**](#gt_c2eeb200-3cd0-4916-966e-d7d6bff1737a).

## Client Details

The client side of this protocol is simply a pass-through between the transport and the higher-layer protocol or application. There are no additional timers or other state required on the client side. Calls made by a higher-layer protocol or application are passed directly to the transport, and the results returned by the transport are passed directly back to the higher-layer protocol or application.

There are several versions of messages that provide similar functionality. Higher-level protocols or applications can use the following guidelines when deciding what message to send.[<34>](#Appendix_A_34" \o "Product behavior note 34)

For selecting between [LsarLookupNames4](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361), [LsarLookupNames3](#Section_b75eaac5e4d14fc08dae61d838b38701), [LsarLookupNames2](#Section_43254e9f32e248caa61de8decf41721e), and [LsarLookupNames](#Section_65d18faa0cb240eea94a2140212f4ec4):

* Only a [**domain controller**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd) can process an LsarLookupNames4 message.
* LsarLookupNames4 requires an [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) handle (specified in its *RpcHandle* parameter) and that the connection is authenticated using the RPC\_C\_AUTHN\_NETLOGON security provider, as specified in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15) section 2.2.1.1.7.
* If these requirements are available to the caller, LsarLookupNames4 can be used; otherwise, LsarLookupNames3 is preferred.
* Domain controllers and non–domain controllers can process LsarLookupNames3, LsarLookupNames2, and LsarLookupNames messages.

For specifics on each message, see sections 3.1.4.5, 3.1.4.6, 3.1.4.7, and 3.1.4.8.

For selecting between [LsarLookupSids3](#Section_678dd7ed611b4d948f5ca06c76ae5c7e), [LsarLookupSids2](#Section_a09480f4145e4b4ca0275b313db61833), and [LsarLookupSids](#Section_eb7ac899e697488393de1e60c7720c02):

* Only a domain controller can process an LsarLookupSids3 message.
* LsarLookupSids3 requires an RPC handle (specified in its RpcHandle parameter) and that the connection is authenticated using the RPC\_C\_AUTHN\_NETLOGON security provider, as specified in [MS-RPCE] section 2.2.1.1.7.
* If these requirements are available to the caller, LsarLookupSids3 can be used; otherwise, LsarLookupSids2 is preferred.
* Domain controllers and non–domain controllers can process LsarLookupSids2 and LsarLookupSids messages.

For specifics on each message, see sections 3.1.4.9, 3.1.4.10, and 3.1.4.11.

# Protocol Example

In this scenario, the client wants to translate a group of [**SIDs**](#gt_83f2020d-0804-4840-a5ac-e06439d50f8d) to their corresponding names. The client starts the communication by sending an [LsarOpenPolicy2](#Section_40a316ae1bae4aa192cd650cdaae1863) request with the following parameters:

1. NTSTATUS
2. LsarOpenPolicy2(
3. [in,unique,string] wchar\_t\* SystemName = "arbitrary string",
4. [in] PLSAPR\_OBJECT\_ATTRIBUTES ObjectAttributes,
5. [in] ACCESS\_MASK DesiredAccess = 0x00000800,
6. [out] LSAPR\_HANDLE \*PolicyHandle
7. );

where *ObjectAttributes* is:

1. typedef struct \_LSAPR\_OBJECT\_ATTRIBUTES {
2. unsigned long Length = 0;
3. unsigned char \*RootDirectory = NULL;
4. PSTRING ObjectName = NULL;
5. unsigned long Attributes = 0;
6. PLSAPR\_SECURITY\_DESCRIPTOR SecurityDescriptor = NULL;
7. PSECURITY\_QUALITY\_OF\_SERVICE SecurityQualityOfService = NULL;
8. } LSAPR\_OBJECT\_ATTRIBUTES, \*PLSAPR\_OBJECT\_ATTRIBUTES;

The server receives this request and performs an access check for the desired access. When successful, the server returns a context handle for the client's use and STATUS\_SUCCESS as the return value. That is:

1. NTSTATUS = 0x00000000
2. LsarOpenPolicy2(
3. [in,unique,string] wchar\_t\* SystemName = {unchanged}
4. [in] PLSAPR\_OBJECT\_ATTRIBUTES ObjectAttributes = {unchanged},
5. [in] ACCESS\_MASK DesiredAccess = {unchanged},
6. [out] LSAPR\_HANDLE \*PolicyHandle = {context handle}
7. );

The client receives this response and sends an [LsarLookupSids2](#Section_a09480f4145e4b4ca0275b313db61833) request to translate the batch of SIDs to names with the following parameters:

1. NTSTATUS
2. LsarLookupSids2(
3. [in] LSAPR\_HANDLE PolicyHandle =
4. {handle returned by LsarOpenPolicy2},
5. [in] PLSAPR\_SID\_ENUM\_BUFFER SidEnumBuffer,
6. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST \*ReferencedDomains,
7. [in, out] PLSAPR\_TRANSLATED\_NAMES\_EX TranslatedNames =
8. {address of a local variable initialized
9. with zeroes},
10. [in] LSAP\_LOOKUP\_LEVEL LookupLevel = LsapLookupWksta,
11. [in, out] unsigned long \*MappedCount =
12. {address of a local variable initialized to 0},
13. [in] unsigned long LookupOptions = 0,
14. [in] unsigned long ClientRevision = 2
15. );

where *SidEnumBuffer* is:

1. typedef struct \_LSAPR\_SID\_ENUM\_BUFFER {
2. unsigned long Entries = {3};
3. [size\_is(Entries)] PLSAPR\_SID\_INFORMATION SidInfo;
4. } LSAPR\_SID\_ENUM\_BUFFER, \*PLSAPR\_SID\_ENUM\_BUFFER;

and SidInfo array is:

1. SidInfo[0] =
2. typedef struct \_LSAPR\_SID\_INFORMATION {
3. PRPC\_SID Sid = { SID0 };
4. } LSAPR\_SID\_INFORMATION, \*PLSAPR\_SID\_INFORMATION;
5. SidInfo[1] =
6. typedef struct \_LSAPR\_SID\_INFORMATION {
7. PRPC\_SID Sid = { SID1 };
8. } LSAPR\_SID\_INFORMATION, \*PLSAPR\_SID\_INFORMATION;
9. SidInfo[2] =
10. typedef struct \_LSAPR\_SID\_INFORMATION {
11. PRPC\_SID Sid = { SID2 };
12. } LSAPR\_SID\_INFORMATION, \*PLSAPR\_SID\_INFORMATION;

The server receives this call and performs the translation explained in message processing in section 3.1.4.10. The server prepares the results, and returns them to the caller with a return value of STATUS\_SUCCESS. The results might resemble the following:

1. NTSTATUS = 0x00000000
2. LsarLookupSids2(
3. [in] LSAPR\_HANDLE PolicyHandle = {unchanged},
4. [in] PLSAPR\_SID\_ENUM\_BUFFER SidEnumBuffer = {unchanged},
5. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST \*ReferencedDomains,
6. [in, out] PLSAPR\_TRANSLATED\_NAMES\_EX TranslatedNames,
7. [in] LSAP\_LOOKUP\_LEVEL LookupLevel = {unchanged},
8. [in, out] unsigned long \*MappedCount = { 3 },
9. [in] unsigned long LookupOptions = {unchanged},
10. [in] unsigned long ClientRevision = {unchanged}
11. );

where *ReferencedDomains* is:

1. typedef struct \_LSAPR\_REFERENCED\_DOMAIN\_LIST {
2. unsigned long Entries = { 2 };
3. [size\_is(Entries)] PLSAPR\_TRUST\_INFORMATION Domains;
4. unsigned long MaxEntries = { 0 };
5. } LSAPR\_REFERENCED\_DOMAIN\_LIST, \*PLSAPR\_REFERENCED\_DOMAIN\_LIST;

and Domains array is:

1. Domains[0] =
2. typedef struct \_LSAPR\_TRUST\_INFORMATION {
3. RPC\_UNICODE\_STRING Name = { "Domain0" };
4. PRPC\_SID Sid = { DomainSID0 };
5. } LSAPR\_TRUST\_INFORMATION, \*PLSAPR\_TRUST\_INFORMATION;
6. Domains[1] =
7. typedef struct \_LSAPR\_TRUST\_INFORMATION {
8. RPC\_UNICODE\_STRING Name = { "Domain1" };
9. PRPC\_SID Sid = { DomainSID1 };
10. } LSAPR\_TRUST\_INFORMATION, \*PLSAPR\_TRUST\_INFORMATION;

*TranslatedNames* might appear as follows:

1. typedef struct \_LSAPR\_TRANSLATED\_NAMES\_EX {
2. unsigned long Entries = { 3 };
3. [size\_is(Entries)] PLSAPR\_TRANSLATED\_NAME\_EX Names;
4. } LSAPR\_TRANSLATED\_NAMES\_EX, \*PLSAPR\_TRANSLATED\_NAMES\_EX;

where Names array is:

1. Names[0] =
2. typedef struct \_LSAPR\_TRANSLATED\_NAME\_EX {
3. SID\_NAME\_USE Use = { SidTypeUser };
4. RPC\_UNICODE\_STRING Name = { "Name0" };
5. long DomainIndex = { 0 };
6. unsigned long Flags = { 0 };
7. } LSAPR\_TRANSLATED\_NAME\_EX, \*PLSAPR\_TRANSLATED\_NAME\_EX;
8. Names[1] =
9. typedef struct \_LSAPR\_TRANSLATED\_NAME\_EX {
10. SID\_NAME\_USE Use = { SidTypeDomain };
11. RPC\_UNICODE\_STRING Name = { "DomainName0" };
12. long DomainIndex = { 0 };
13. unsigned long Flags = { 0 };
14. } LSAPR\_TRANSLATED\_NAME\_EX, \*PLSAPR\_TRANSLATED\_NAME\_EX;
15. Names[2] =
16. typedef struct \_LSAPR\_TRANSLATED\_NAME\_EX {
17. SID\_NAME\_USE Use = { SidTypeUser };
18. RPC\_UNICODE\_STRING Name = { "Name3" };
19. long DomainIndex = { 1 };
20. unsigned long Flags = { 0 };
21. } LSAPR\_TRANSLATED\_NAME\_EX, \*PLSAPR\_TRANSLATED\_NAME\_EX;

The client receives this response, and extracts the names from the returned information. It then sends an [LsarClose](#Section_be232cec04a9439ab718931c17c3ff0f) request to release the context handle that was opened in the first call:

1. NTSTATUS
2. LsarClose(
3. [in,out] LSAPR\_HANDLE \*ObjectHandle
4. = {handle returned by LsarOpenPolicy2}
5. );

The server receives this call, and releases the handle. It responds to the caller with a return value of STATUS\_SUCCESS:

1. NTSTATUS = 0x00000000
2. LsarClose(
3. [in,out] LSAPR\_HANDLE \*ObjectHandle = {0}
4. );

# Security

## Security Considerations for Implementers

This protocol provides query functionality into databases that might have other access control mechanisms. This protocol should obey those mechanisms; otherwise, it might become a source of information disclosure.

Access control on an [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553) of this protocol should be configurable to allow customizations.[<35>](#Appendix_A_35" \o "Product behavior note 35)

The RPC server has to successfully authenticate the client if user names are considered confidential information.

The [**RPC client**](#gt_e5a7be6b-98db-4e8d-8116-5893f43ab48b) has to authenticate the server if the results are used to make policy decisions.

## Index of Security Parameters

| Security parameter | Section |
| --- | --- |
| Authentication service (AS) settings | [3.1.4.9](#Section_678dd7ed611b4d948f5ca06c76ae5c7e)  [3.1.4.10](#Section_a09480f4145e4b4ca0275b313db61833)  [3.1.4.5](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361)  [3.1.4.6](#Section_b75eaac5e4d14fc08dae61d838b38701) |
| Security descriptor | [3.1.4.1](#Section_40a316ae1bae4aa192cd650cdaae1863) |

# Appendix A: Full IDL

For ease of implementation, the full IDL is provided below, where "ms-dtyp.idl" is the IDL specified in [[MS-DTYP]](%5bMS-DTYP%5d.pdf#Section_cca2742956894a16b2b49325d93e4ba2) Appendix A.

**Note**  The lsarpc interface is shared between this protocol and the Local Security Authority (Domain Policy) Remote Protocol [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc). For convenience, the IDL definitions that appear below and the IDL definitions in [MS-LSAD] section 6 have been merged and are available for download. For more information, see [[MSFT-LSA-IDL]](https://go.microsoft.com/fwlink/?LinkId=150268).

1. import "ms-dtyp.idl";
2. [
3. uuid(12345778-1234-ABCD-EF00-0123456789AB),
4. version(0.0),
5. ms\_union,
6. pointer\_default(unique)
7. ]
8. interface lsarpc
9. {
10. //
11. // Type definitions.
12. //
13. //
14. // Start of common types.
15. //
16. typedef [context\_handle] void \* LSAPR\_HANDLE;
17. typedef unsigned char SECURITY\_CONTEXT\_TRACKING\_MODE,
18. \*PSECURITY\_CONTEXT\_TRACKING\_MODE;
19. typedef unsigned short SECURITY\_DESCRIPTOR\_CONTROL,
20. \*PSECURITY\_DESCRIPTOR\_CONTROL;
21. typedef struct \_STRING {
22. unsigned short Length;
23. unsigned short MaximumLength;
24. [size\_is(MaximumLength), length\_is(Length)]
25. char \* Buffer;
26. } STRING, \*PSTRING;
27. typedef struct \_LSAPR\_ACL {
28. unsigned char AclRevision;
29. unsigned char Sbz1;
30. unsigned short AclSize;
31. [size\_is(AclSize - 4)] unsigned char Dummy1[\*];
32. } LSAPR\_ACL, \*PLSAPR\_ACL;
33. typedef struct \_LSAPR\_SECURITY\_DESCRIPTOR {
34. unsigned char Revision;
35. unsigned char Sbz1;
36. SECURITY\_DESCRIPTOR\_CONTROL Control;
37. PRPC\_SID Owner;
38. PRPC\_SID Group;
39. PLSAPR\_ACL Sacl;
40. PLSAPR\_ACL Dacl;
41. } LSAPR\_SECURITY\_DESCRIPTOR, \*PLSAPR\_SECURITY\_DESCRIPTOR;
42. typedef enum \_SECURITY\_IMPERSONATION\_LEVEL {
43. SecurityAnonymous = 0,
44. SecurityIdentification = 1,
45. SecurityImpersonation = 2,
46. SecurityDelegation = 3
47. } SECURITY\_IMPERSONATION\_LEVEL, \* PSECURITY\_IMPERSONATION\_LEVEL;
48. typedef struct \_SECURITY\_QUALITY\_OF\_SERVICE {
49. unsigned long Length;
50. SECURITY\_IMPERSONATION\_LEVEL ImpersonationLevel;
51. SECURITY\_CONTEXT\_TRACKING\_MODE ContextTrackingMode;
52. unsigned char EffectiveOnly;
53. } SECURITY\_QUALITY\_OF\_SERVICE, \* PSECURITY\_QUALITY\_OF\_SERVICE;
54. typedef struct \_LSAPR\_OBJECT\_ATTRIBUTES {
55. unsigned long Length;
56. unsigned char \* RootDirectory;
57. PSTRING ObjectName;
58. unsigned long Attributes;
59. PLSAPR\_SECURITY\_DESCRIPTOR SecurityDescriptor;
60. PSECURITY\_QUALITY\_OF\_SERVICE SecurityQualityOfService;
61. } LSAPR\_OBJECT\_ATTRIBUTES, \*PLSAPR\_OBJECT\_ATTRIBUTES;
62. typedef struct \_LSAPR\_TRUST\_INFORMATION {
63. RPC\_UNICODE\_STRING Name;
64. PRPC\_SID Sid;
65. } LSAPR\_TRUST\_INFORMATION, \*PLSAPR\_TRUST\_INFORMATION;
66. //
67. // End of common types.
68. //
69. typedef struct \_LSAPR\_REFERENCED\_DOMAIN\_LIST {
70. unsigned long Entries;
71. [size\_is(Entries)] PLSAPR\_TRUST\_INFORMATION Domains;
72. unsigned long MaxEntries;
73. } LSAPR\_REFERENCED\_DOMAIN\_LIST, \*PLSAPR\_REFERENCED\_DOMAIN\_LIST;
74. typedef enum \_SID\_NAME\_USE {
75. SidTypeUser = 1,
76. SidTypeGroup,
77. SidTypeDomain,
78. SidTypeAlias,
79. SidTypeWellKnownGroup,
80. SidTypeDeletedAccount,
81. SidTypeInvalid,
82. SidTypeUnknown,
83. SidTypeComputer,
84. SidTypeLabel
85. } SID\_NAME\_USE, \*PSID\_NAME\_USE;
86. typedef struct \_LSA\_TRANSLATED\_SID {
87. SID\_NAME\_USE Use;
88. unsigned long RelativeId;
89. long DomainIndex;
90. } LSA\_TRANSLATED\_SID, \*PLSA\_TRANSLATED\_SID;
91. typedef struct \_LSAPR\_TRANSLATED\_SIDS {
92. [range(0,1000)] unsigned long Entries;
93. [size\_is(Entries)] PLSA\_TRANSLATED\_SID Sids;
94. } LSAPR\_TRANSLATED\_SIDS, \*PLSAPR\_TRANSLATED\_SIDS;
95. typedef enum \_LSAP\_LOOKUP\_LEVEL {
96. LsapLookupWksta = 1,
97. LsapLookupPDC,
98. LsapLookupTDL,
99. LsapLookupGC,
100. LsapLookupXForestReferral,
101. LsapLookupXForestResolve,
102. LsapLookupRODCReferralToFullDC
103. } LSAP\_LOOKUP\_LEVEL, \*PLSAP\_LOOKUP\_LEVEL;
104. typedef struct \_LSAPR\_SID\_INFORMATION {
105. PRPC\_SID Sid;
106. } LSAPR\_SID\_INFORMATION, \*PLSAPR\_SID\_INFORMATION;
107. typedef struct \_LSAPR\_SID\_ENUM\_BUFFER {
108. [range(0, 20480)] unsigned long Entries;
109. [size\_is(Entries)] PLSAPR\_SID\_INFORMATION SidInfo;
110. } LSAPR\_SID\_ENUM\_BUFFER, \*PLSAPR\_SID\_ENUM\_BUFFER;
111. typedef struct \_LSAPR\_TRANSLATED\_NAME {
112. SID\_NAME\_USE Use;
113. RPC\_UNICODE\_STRING Name;
114. long DomainIndex;
115. } LSAPR\_TRANSLATED\_NAME, \*PLSAPR\_TRANSLATED\_NAME;
116. typedef struct \_LSAPR\_TRANSLATED\_NAMES {
117. [range(0, 20480)] unsigned long Entries;
118. [size\_is(Entries)] PLSAPR\_TRANSLATED\_NAME Names;
119. } LSAPR\_TRANSLATED\_NAMES, \*PLSAPR\_TRANSLATED\_NAMES;
120. typedef struct \_LSAPR\_TRANSLATED\_NAME\_EX {
121. SID\_NAME\_USE Use;
122. RPC\_UNICODE\_STRING Name;
123. long DomainIndex;
124. unsigned long Flags;
125. } LSAPR\_TRANSLATED\_NAME\_EX, \*PLSAPR\_TRANSLATED\_NAME\_EX;
126. typedef struct \_LSAPR\_TRANSLATED\_NAMES\_EX {
127. [range(0, 20480)] unsigned long Entries;
128. [size\_is(Entries)] PLSAPR\_TRANSLATED\_NAME\_EX Names;
129. } LSAPR\_TRANSLATED\_NAMES\_EX, \*PLSAPR\_TRANSLATED\_NAMES\_EX;
130. typedef struct \_LSAPR\_TRANSLATED\_SID\_EX {
131. SID\_NAME\_USE Use;
132. unsigned long RelativeId;
133. long DomainIndex;
134. unsigned long Flags;
135. } LSAPR\_TRANSLATED\_SID\_EX, \*PLSAPR\_TRANSLATED\_SID\_EX;
136. typedef struct \_LSAPR\_TRANSLATED\_SIDS\_EX {
137. [range (0,1000)] unsigned long Entries;
138. [size\_is(Entries)] PLSAPR\_TRANSLATED\_SID\_EX Sids;
139. } LSAPR\_TRANSLATED\_SIDS\_EX, \*PLSAPR\_TRANSLATED\_SIDS\_EX;
140. typedef struct \_LSAPR\_TRANSLATED\_SID\_EX2 {
141. SID\_NAME\_USE Use;
142. PRPC\_SID Sid;
143. long DomainIndex;
144. unsigned long Flags;
145. } LSAPR\_TRANSLATED\_SID\_EX2, \*PLSAPR\_TRANSLATED\_SID\_EX2;
146. typedef struct \_LSAPR\_TRANSLATED\_SIDS\_EX2 {
147. [range (0,1000)] unsigned long Entries;
148. [size\_is(Entries)] PLSAPR\_TRANSLATED\_SID\_EX2 Sids;
149. } LSAPR\_TRANSLATED\_SIDS\_EX2, \*PLSAPR\_TRANSLATED\_SIDS\_EX2;
150. //
151. // Methods
152. //
153. //
154. // The following notation conventions are used for some IDL methods:
155. //
156. // void
157. // Lsar\_LSA\_DP\_XX( void );
158. //
159. // (where XX represents the opnum.)
160. //
161. // This notation indicates that the method is defined in this
162. // interface but is described in the
163. // Local Security Authority (Domain Policy) protocol
164. // specification.
165. //
166. // void OpnumXXNotUsedOnWire(void);
167. //
168. // (where XX represents the opnum.)
169. //
170. // This notation indicates that the method is defined in this
171. // interface but is not seen on the wire.
172. //
173. // Opnum 0
174. NTSTATUS
175. LsarClose(
176. [in,out] LSAPR\_HANDLE \*ObjectHandle
177. );
178. // Opnum 1
179. void Opnum1NotUsedOnWire(void);
180. // Opnum 2
181. void
182. Lsar\_LSA\_DP\_2( void );
183. // Opnum 3
184. void
185. Lsar\_LSA\_DP\_3( void );
186. // Opnum 4
187. void
188. Lsar\_LSA\_DP\_4( void );
189. // Opnum 5
190. void Opnum5NotUsedOnWire(void);
191. // Opnum 6
192. NTSTATUS
193. LsarOpenPolicy(
194. [in,unique] wchar\_t \*SystemName,
195. [in] PLSAPR\_OBJECT\_ATTRIBUTES ObjectAttributes,
196. [in] ACCESS\_MASK DesiredAccess,
197. [out] LSAPR\_HANDLE \*PolicyHandle
198. );
199. // Opnum 7
200. void
201. Lsar\_LSA\_DP\_7( void );
202. // Opnum 8
203. void
204. Lsar\_LSA\_DP\_8( void );
205. // Opnum 9
206. void Opnum9NotUsedOnWire(void);
207. // Opnum 10
208. void
209. Lsar\_LSA\_DP\_10( void );
210. // Opnum 11
211. void
212. Lsar\_LSA\_DP\_11( void );
213. // Opnum 12
214. void
215. Lsar\_LSA\_DP\_12( void );
216. // Opnum 13
217. void
218. Lsar\_LSA\_DP\_13( void );
219. // Opnum 14
220. NTSTATUS
221. LsarLookupNames(
222. [in] LSAPR\_HANDLE PolicyHandle,
223. [in, range(0,1000)] unsigned long Count,
224. [in, size\_is(Count)] PRPC\_UNICODE\_STRING Names,
225. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST \*ReferencedDomains,
226. [in, out] PLSAPR\_TRANSLATED\_SIDS TranslatedSids,
227. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
228. [in, out] unsigned long \*MappedCount
229. );
230. // Opnum 15
231. NTSTATUS
232. LsarLookupSids(
233. [in] LSAPR\_HANDLE PolicyHandle,
234. [in] PLSAPR\_SID\_ENUM\_BUFFER SidEnumBuffer,
235. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST \*ReferencedDomains,
236. [in, out] PLSAPR\_TRANSLATED\_NAMES TranslatedNames,
237. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
238. [in, out] unsigned long \*MappedCount
239. );
240. // Opnum 16
241. void
242. Lsar\_LSA\_DP\_16( void );
243. // Opnum 17
244. void
245. Lsar\_LSA\_DP\_17( void );
246. // Opnum 18
247. void
248. Lsar\_LSA\_DP\_18( void );
249. // Opnum 19
250. void
251. Lsar\_LSA\_DP\_19( void );
252. // Opnum 20
253. void
254. Lsar\_LSA\_DP\_20( void );
255. // Opnum 21
256. void Opnum21NotUsedOnWire(void);
257. // Opnum 22
258. void Opnum22NotUsedOnWire(void);
259. // Opnum 23
260. void
261. Lsar\_LSA\_DP\_23( void );
262. // Opnum 24
263. void
264. Lsar\_LSA\_DP\_24( void );
265. // Opnum 25
266. void
267. Lsar\_LSA\_DP\_25( void );
268. // Opnum 26
269. void
270. Lsar\_LSA\_DP\_26( void );
271. // Opnum 27
272. void
273. Lsar\_LSA\_DP\_27( void );
274. // Opnum 28
275. void
276. Lsar\_LSA\_DP\_28( void );
277. // Opnum 29
278. void
279. Lsar\_LSA\_DP\_29( void );
280. // Opnum 30
281. void
282. Lsar\_LSA\_DP\_30( void );
283. // Opnum 31
284. void
285. Lsar\_LSA\_DP\_31( void );
286. // Opnum 32
287. void
288. Lsar\_LSA\_DP\_32( void );
289. // Opnum 33
290. void
291. Lsar\_LSA\_DP\_33( void );
292. // Opnum 34
293. void
294. Lsar\_LSA\_DP\_34( void );
295. // Opnum 35
296. void
297. Lsar\_LSA\_DP\_35( void );
298. // Opnum 36
299. void
300. Lsar\_LSA\_DP\_36( void );
301. // Opnum 37
302. void
303. Lsar\_LSA\_DP\_37( void );
304. // Opnum 38
305. void
306. Lsar\_LSA\_DP\_38( void );
307. // Opnum 39
308. void
309. Lsar\_LSA\_DP\_39( void );
310. // Opnum 40
311. void
312. Lsar\_LSA\_DP\_40( void );
313. // Opnum 41
314. void
315. Lsar\_LSA\_DP\_41( void );
316. // Opnum 42
317. void
318. Lsar\_LSA\_DP\_42( void );
319. // Opnum 43
320. void
321. Lsar\_LSA\_DP\_43( void );
322. // Opnum 44
323. NTSTATUS
324. LsarOpenPolicy2(
325. [in,unique,string] wchar\_t \*SystemName,
326. [in] PLSAPR\_OBJECT\_ATTRIBUTES ObjectAttributes,
327. [in] ACCESS\_MASK DesiredAccess,
328. [out] LSAPR\_HANDLE \*PolicyHandle
329. );
330. // Opnum 45
331. NTSTATUS
332. LsarGetUserName(
333. [in,unique,string] wchar\_t \*SystemName,
334. [in,out] PRPC\_UNICODE\_STRING \*UserName,
335. [in,out,unique] PRPC\_UNICODE\_STRING \*DomainName
336. );
337. // Opnum 46
338. void
339. Lsar\_LSA\_DP\_46( void );
340. // Opnum 47
341. void
342. Lsar\_LSA\_DP\_47( void );
343. // Opnum 48
344. void
345. Lsar\_LSA\_DP\_48( void );
346. // Opnum 49
347. void
348. Lsar\_LSA\_DP\_49( void );
349. // Opnum 50
350. void
351. Lsar\_LSA\_DP\_50( void );
352. // Opnum 51
353. void
354. Lsar\_LSA\_DP\_51( void );
355. // Opnum 52
356. void Opnum52NotUsedOnWire(void);
357. // Opnum 53
358. void
359. Lsar\_LSA\_DP\_53( void );
360. // Opnum 54
361. void
362. Lsar\_LSA\_DP\_54( void );
363. // Opnum 55
364. void
365. Lsar\_LSA\_DP\_55( void );
366. // Opnum 56
367. void Opnum56NotUsedOnWire(void);
368. // Opnum 57
369. NTSTATUS
370. LsarLookupSids2(
371. [in] LSAPR\_HANDLE PolicyHandle,
372. [in] PLSAPR\_SID\_ENUM\_BUFFER SidEnumBuffer,
373. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST \*ReferencedDomains,
374. [in, out] PLSAPR\_TRANSLATED\_NAMES\_EX TranslatedNames,
375. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
376. [in, out] unsigned long \*MappedCount,
377. [in] unsigned long LookupOptions,
378. [in] unsigned long ClientRevision
379. );
380. // Opnum 58
381. NTSTATUS
382. LsarLookupNames2(
383. [in] LSAPR\_HANDLE PolicyHandle,
384. [in, range(0,1000)] unsigned long Count,
385. [in, size\_is(Count)] PRPC\_UNICODE\_STRING Names,
386. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST \*ReferencedDomains,
387. [in, out] PLSAPR\_TRANSLATED\_SIDS\_EX TranslatedSids,
388. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
389. [in, out] unsigned long \*MappedCount,
390. [in] unsigned long LookupOptions,
391. [in] unsigned long ClientRevision
392. );
393. // Opnum 59
394. void
395. Lsar\_LSA\_DP\_59( void );
396. // Opnum 60
397. void Opnum60NotUsedOnWire(void);
398. // Opnum 61
399. void Opnum61NotUsedOnWire(void);
400. // Opnum 62
401. void Opnum62NotUsedOnWire(void);
402. // Opnum 63
403. void Opnum63NotUsedOnWire(void);
404. // Opnum 64
405. void Opnum64NotUsedOnWire(void);
406. // Opnum 65
407. void Opnum65NotUsedOnWire(void);
408. // Opnum 66
409. void Opnum66NotUsedOnWire(void);
410. // Opnum 67
411. void Opnum67NotUsedOnWire(void);
412. // Opnum 68
413. NTSTATUS
414. LsarLookupNames3(
415. [in] LSAPR\_HANDLE PolicyHandle,
416. [in, range(0,1000)] unsigned long Count,
417. [in, size\_is(Count)] PRPC\_UNICODE\_STRING Names,
418. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST \*ReferencedDomains,
419. [in, out] PLSAPR\_TRANSLATED\_SIDS\_EX2 TranslatedSids,
420. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
421. [in, out] unsigned long \*MappedCount,
422. [in] unsigned long LookupOptions,
423. [in] unsigned long ClientRevision
424. );
425. // Opnum 69
426. void Opnum69NotUsedOnWire(void);
427. // Opnum 70
428. void Opnum70NotUsedOnWire(void);
429. // Opnum 71
430. void Opnum71NotUsedOnWire(void);
431. // Opnum 72
432. void Opnum72NotUsedOnWire(void);
433. // Opnum 73
434. void
435. Lsar\_LSA\_DP\_73( void );
436. // Opnum 74
437. void
438. Lsar\_LSA\_DP\_74( void );
439. // Opnum 75
440. void Opnum75NotUsedOnWire(void);
441. // Opnum 76
442. NTSTATUS
443. LsarLookupSids3(
444. [in] handle\_t RpcHandle,
445. [in] PLSAPR\_SID\_ENUM\_BUFFER SidEnumBuffer,
446. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST \*ReferencedDomains,
447. [in, out] PLSAPR\_TRANSLATED\_NAMES\_EX TranslatedNames,
448. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
449. [in, out] unsigned long \*MappedCount,
450. [in] unsigned long LookupOptions,
451. [in] unsigned long ClientRevision
452. );
453. // Opnum 77
454. NTSTATUS
455. LsarLookupNames4(
456. [in] handle\_t RpcHandle,
457. [in, range(0,1000)] unsigned long Count,
458. [in, size\_is(Count)] PRPC\_UNICODE\_STRING Names,
459. [out] PLSAPR\_REFERENCED\_DOMAIN\_LIST \*ReferencedDomains,
460. [in, out] PLSAPR\_TRANSLATED\_SIDS\_EX2 TranslatedSids,
461. [in] LSAP\_LOOKUP\_LEVEL LookupLevel,
462. [in, out] unsigned long \*MappedCount,
463. [in] unsigned long LookupOptions,
464. [in] unsigned long ClientRevision
465. );
466. }

# Appendix B: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include updates to those products.

The terms "earlier" and "later", when used with a product version, refer to either all preceding versions or all subsequent versions, respectively. The term "through" refers to the inclusive range of versions. Applicable Microsoft products are listed chronologically in this section.

The following tables show the relationships between Microsoft product versions or supplemental software and the roles they perform.

| Windows Client releases | Client role | Server role |
| --- | --- | --- |
| Windows NT operating system | Yes | Yes |
| Windows 2000 Professional operating system | Yes | Yes |
| Windows XP operating system | Yes | Yes |
| Windows Vista operating system | Yes | Yes |
| Windows 7 operating system | Yes | Yes |
| Windows 8 operating system | Yes | Yes |
| Windows 8.1 operating system | Yes | Yes |
| Windows 10 operating system | Yes | Yes |

| Windows Server releases | Client role | Server role |
| --- | --- | --- |
| Windows NT | Yes | Yes |
| Windows 2000 Server operating system | Yes | Yes |
| Windows Server 2003 operating system | Yes | Yes |
| Windows Server 2003 R2 operating system | Yes | Yes |
| Windows Server 2008 operating system | Yes | Yes |
| Windows Server 2008 R2 operating system | Yes | Yes |
| Windows Server 2012 operating system | Yes | Yes |
| Windows Server 2012 R2 operating system | Yes | Yes |
| Windows Server 2016 operating system | Yes | Yes |
| Windows Server operating system | Yes | Yes |

Exceptions, if any, are noted in this section. If an update version, service pack or Knowledge Base (KB) number appears with a product name, the behavior changed in that update. The new behavior also applies to subsequent updates unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms "SHOULD" or "SHOULD NOT" implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term "MAY" implies that the product does not follow the prescription.

[<1> Section 2.1](#Appendix_A_Target_1): The Windows [**RPC server**](#gt_ae65dac0-cd24-4e83-a946-6d1097b71553) and [**RPC client**](#gt_e5a7be6b-98db-4e8d-8116-5893f43ab48b) do not support TCP/IP on Windows NT and Windows 2000 operating system.

[<2> Section 2.1](#Appendix_A_Target_2): The endpoint "\PIPE\lsarpc" by default allows anonymous access on Windows NT 3.1 operating system, Windows NT 3.5 operating system, Windows NT 3.51 operating system, Windows 2000, Windows XP, Windows Server 2003, Windows Server 2003 R2, and Windows Vista. Anonymous access to this pipe is removed by default on Windows Vista operating system with Service Pack 1 (SP1) and later and Windows Server 2008 and later. Pipe access check happens before any other access check, and hence overrides any other access.

[<3> Section 2.1](#Appendix_A_Target_3): If the client uses an unsupported [**RPC protocol sequence**](#gt_0c171cc7-e9c4-41b6-95a9-536db0042c7a), the RPC server implementations in Windows 2000, Windows XP, Windows Server 2003, and Windows Server 2003 R2 return RPC\_S\_PROTSEQ\_NOT\_SUPPORTED (as specified in [[MS-ERREF]](%5bMS-ERREF%5d.pdf#Section_1bc92ddfb79e413cbbaa99a5281a6c90)). Windows Vista and later and Windows Server 2008 and later throw an [**RPC**](#gt_8a7f6700-8311-45bc-af10-82e10accd331) exception with status code ERROR\_ACCESS\_DENIED.

[<4> Section 2.1](#Appendix_A_Target_4): Servers running Windows 2000, Windows XP, and Windows Server 2003 accept calls at any authentication level. Without [[MSKB-3149090]](https://go.microsoft.com/fwlink/?LinkId=786963) installed, servers running Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, Windows Server 2012 R2, Windows 10 v1507 operating system, or Windows 10 v1511 operating system also accept calls at any authentication level.

[<5> Section 2.2](#Appendix_A_Target_5): The following table contains a timeline of when a particular data type was introduced.

| Data type name | Windows version |
| --- | --- |
| LSAPR\_HANDLE | Windows NT 3.1 |
| STRING | Windows NT 3.1 |
| LSAPR\_ACL | Windows NT 3.1 |
| SECURITY\_DESCRIPTOR\_CONTROL | Windows NT 3.1 |
| LSAPR\_SECURITY\_DESCRIPTOR | Windows NT 3.1 |
| SECURITY\_IMPERSONATION\_LEVEL | Windows NT 3.1 |
| SECURITY\_CONTEXT\_TRACKING\_MODE | Windows NT 3.1 |
| SECURITY\_QUALITY\_OF\_SERVICE | Windows NT 3.1 |
| LSAPR\_OBJECT\_ATTRIBUTES | Windows NT 3.1 |
| ACCESS\_MASK | Windows NT 3.1 |
| LSAPR\_TRUST\_INFORMATION | Windows NT 3.1 |
| LSAPR\_REFERENCED\_DOMAIN\_LIST | Windows NT 3.1 |
| SID\_NAME\_USE | Windows NT 3.1 |
| LSA\_TRANSLATED\_SID | Windows NT 3.1 |
| LSAPR\_TRANSLATED\_SIDS | Windows NT 3.1 |
| LSAP\_LOOKUP\_LEVEL | Windows NT 3.1 |
| LSAPR\_SID\_INFORMATION | Windows NT 3.1 |
| LSAPR\_SID\_ENUM\_BUFFER | Windows NT 3.1 |
| LSAPR\_TRANSLATED\_NAME | Windows NT 3.1 |
| LSAPR\_TRANSLATED\_NAMES | Windows NT 3.1 |
| LSAPR\_TRANSLATED\_NAME\_EX | Windows 2000 |
| LSAPR\_TRANSLATED\_NAMES\_EX | Windows 2000 |
| LSAPR\_TRANSLATED\_SID\_EX | Windows 2000 |
| LSAPR\_TRANSLATED\_SIDS\_EX | Windows 2000 |
| LSAPR\_TRANSLATED\_SID\_EX2 | Windows XP, Windows Server 2003 |
| LSAPR\_TRANSLATED\_SIDS\_EX2 | Windows XP, Windows Server 2003 |

[<6> Section 2.2.13](#Appendix_A_Target_6): The following table contains a timeline of when a particular enumeration value was introduced.

| Enumeration value | Enumeration name | Windows version |
| --- | --- | --- |
| 1 | SidTypeUser | Windows NT 3.1 |
| 2 | SidTypeGroup | Windows NT 3.1 |
| 3 | SidTypeDomain | Windows NT 3.1 |
| 4 | SidTypeAlias | Windows NT 3.1 |
| 5 | SidTypeWellKnownGroup | Windows NT 3.1 |
| 6 | SidTypeDeletedAccount | Windows NT 3.1 |
| 7 | SidTypeInvalid | Windows NT 3.1 |
| 8 | SidTypeUnknown | Windows NT 3.1 |
| 9 | SidTypeComputer | Windows 2000 |
| 10 | SidTypeLabel | Windows Vista, Windows Server 2008 |

[<7> Section 2.2.15](#Appendix_A_Target_7): The Windows RPC server and RPC client limit the Entries field of this structure to 1,000 (using the range primitive defined in [[MS-RPCE]](%5bMS-RPCE%5d.pdf#Section_290c38b192fe422991e64fc376610c15)) in Windows XP operating system Service Pack 2 (SP2) and later and Windows Server 2003 and later. Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, Windows NT 4.0 operating system, Windows 2000, and Windows XP do not have this restriction.

[<8> Section 2.2.16](#Appendix_A_Target_8): The following table contains a timeline of when particular enumeration values were introduced.

| Enumeration value | Enumeration name | Windows version |
| --- | --- | --- |
| 1 | LsapLookupWksta | Windows NT 3.1 |
| 2 | LsapLookupPDC | Windows NT 3.1 |
| 3 | LsapLookupTDL | Windows NT 3.1 |
| 4 | LsapLookupGC | Windows 2000 |
| 5 | LsapLookupXForestReferral | Windows XP, Windows Server 2003 |
| 6 | LsapLookupXForestResolve | Windows XP, Windows Server 2003 |
| 7 | LsapLookupRODCReferralToFullDC | Windows Vista, Windows Server 2008 |

[<9> Section 2.2.18](#Appendix_A_Target_9): The Windows implementation of the RPC server and RPC client limits the **Entries** field of this structure to 0x5000 (using the range primitive defined in [MS-RPCE]) in Windows XP SP2 and later and Windows Server 2003 and later. Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, Windows NT 4.0, Windows 2000, and Windows XP do not enforce this restriction.

[<10> Section 2.2.20](#Appendix_A_Target_10): The Windows RPC server and RPC client limit the Entries field of this structure to 0x5000 (using the range primitive defined in [MS-RPCE]) in Windows XP SP2 and later and Windows Server 2003 and later. Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, Windows NT 4.0, Windows 2000, and Windows XP do not enforce this restriction.

[<11> Section 2.2.21](#Appendix_A_Target_11): The following table contains a timeline of when each flag value was introduced.

| Flag value | Windows version |
| --- | --- |
| 0x00000001 | Windows 2000 |
| 0x00000002 | Windows XP, Windows Server 2003 |
| 0x00000004 | Windows Vista, Windows Server 2008 |

[<12> Section 2.2.22](#Appendix_A_Target_12): The Windows RPC server and RPC client limit the **Entries** field of this structure to 0x5000 (using the range primitive defined in [MS-RPCE]) in Windows XP SP2 and later and Windows Server 2003 and later. Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, Windows NT 4.0, Windows 2000, and Windows XP do not enforce this restriction.

[<13> Section 2.2.23](#Appendix_A_Target_13): The following table contains a timeline of when each flag value was introduced.

| Flag value | Windows version |
| --- | --- |
| 0x00000001 | Windows 2000 |
| 0x00000002 | Windows XP, Windows Server 2003 |
| 0x00000004 | Windows Vista, Windows Server 2008 |

[<14> Section 2.2.24](#Appendix_A_Target_14): The Windows RPC server and RPC client limit the **Entries** field of this structure to 1,000 (using the range primitive defined in [MS-RPCE]) in Windows XP SP2 and later and Windows Server 2003 and later. Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, Windows NT 4.0, Windows 2000, and Windows XP do not enforce this restriction.

[<15> Section 2.2.25](#Appendix_A_Target_15): The following table contains a timeline of when each flag value was introduced.

| Flag value | Windows version |
| --- | --- |
| 0x00000001 | Windows 2000 |
| 0x00000002 | Windows XP, Windows Server 2003 |
| 0x00000004 | Windows Vista, Windows Server 2008 |

[<16> Section 2.2.26](#Appendix_A_Target_16): The Windows RPC server and RPC client limit the **Entries** field of this structure to 1,000 (using the range primitive defined in [MS-RPCE]) in Windows XP SP2 and later and Windows Server 2003 and later. Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, Windows NT 4.0, Windows 2000, and Windows XP do not enforce this restriction.

[<17> Section 3.1.1.1](#Appendix_A_Target_17): Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, and Windows NT 4.0, when creating these views, leave the **Domain DNS Name**, **Default User Principal Names**, **User Principal Name**, and **Security Principal SID History** columns empty; therefore, they cannot be used for matching.

[<18> Section 3.1.1.1.1](#Appendix_A_Target_18): The Enterprise Domain Controllers, Self, Authenticated Users, Restricted, and Terminal Server User entries were added in Windows 2000.

The Local Service, Network Service, and Remote Interactive Logon entries were added in Windows XP.

The This Organization and Other Organization entries were added in Windows Server 2003.

[<19> Section 3.1.1.1.1](#Appendix_A_Target_19): The entries in the table that precedes this citation in section [3.1.1.1.1](#Section_e09da72ee6c94f91aa6468b0475719b6) were added in Windows Server 2003.

[<20> Section 3.1.1.1.1](#Appendix_A_Target_20): The entries in the table that precedes this citation in section 3.1.1.1.1 were added in Windows Vista.

[<21> Section 3.1.4](#Appendix_A_Target_21): The Windows implementation of this protocol asks the RPC engine to do the following:

* Perform a strict [**Network Data Representation (NDR)**](#gt_9ebf9540-2c31-43bc-bc56-4a932faabf2d) data consistency check at target level 5.0 (as specified in [MS-RPCE] section 3) in all version of Windows except Windows NT.
* Include support for both NDR and NDR64 transfer syntaxes, as well as the negotiation mechanism for determining what transfer syntax will be used (as specified in [MS-RPCE] section 3) in Windows XP and later and Windows Server 2003 and later.
* Via the strict\_context\_handle attribute, reject the use of context handles created by a method of a different RPC interface than this one (as specified in [MS-RPCE] section 3).

[<22> Section 3.1.4](#Appendix_A_Target_22): The following table contains a timeline of when each method was introduced.

| Opnum | Friendly name | Product |
| --- | --- | --- |
| 0 | LsarClose | Windows NT 3.1 |
| 6 | LsarOpenPolicy | Windows NT 3.1 |
| 14 | LsarLookupNames | Windows NT 3.1 |
| 15 | LsarLookupSids | Windows NT 3.1 |
| 44 | LsarOpenPolicy2 | Windows NT 3.51 |
| 45 | LsarGetUserName | Windows NT 4.0 |
| 57 | LsarLookupSids2 | Windows 2000 |
| 58 | LsarLookupNames2 | Windows 2000 |
| 68 | LsarLookupNames3 | Windows XP, Windows Server 2003 |
| 76 | LsarLookupSids3 | Windows XP, Windows Server 2003 |
| 77 | LsarLookupNames4 | Windows XP, Windows Server 2003 |

[<23> Section 3.1.4](#Appendix_A_Target_23): Some gaps in the [**opnum**](#gt_e127848e-c66d-427d-b3aa-9f904fa4ada7) numbering sequence correspond to opnums that are documented in [[MS-LSAD]](%5bMS-LSAD%5d.pdf#Section_1b5471ef4c334a91b079dfcbb82f05cc). All other gaps in the opnum numbering sequence apply to Windows as follows.

| Opnum | Description |
| --- | --- |
| 1 | Used only locally by Windows, never remotely. |
| 5 | Not used by Windows. |
| 9 | Not used by Windows. |
| 21 | Not used by Windows. |
| 22 | Not used by Windows. |
| 52 | Not used by Windows. |
| 56 | Used only locally by Windows, never remotely. |
| 60 | Used only locally by Windows, never remotely. |
| 61 | Used only locally by Windows, never remotely. |
| 62 | Used only locally by Windows, never remotely. |
| 63 | Used only locally by Windows, never remotely. |
| 64 | Used only locally by Windows, never remotely. |
| 65 | Used only locally by Windows, never remotely. |
| 66 | Used only locally by Windows, never remotely. |
| 67 | Used only locally by Windows, never remotely. |
| 69 | Used only locally by Windows, never remotely. |
| 70 | Used only locally by Windows, never remotely. |
| 71 | Used only locally by Windows, never remotely. |
| 72 | Used only locally by Windows, never remotely. |
| 75 | Used only locally by Windows, never remotely. |

[<24> Section 3.1.4.5](#Appendix_A_Target_24): The Windows RPC server and RPC client limit the **Count** field of this structure to 1,000 (using the range primitive defined in [MS-RPCE]) in Windows XP SP2 and later and Windows Server 2003 and later. Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, Windows NT 4.0, Windows 2000, and Windows XP do not enforce this restriction.

[<25> Section 3.1.4.5](#Appendix_A_Target_25): For Windows, usage of 0x00000001 for ClientRevision implies a client that is running an operating system released before Windows 2000 (Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, and Windows NT 4.0). Usage of 0x00000002 implies that the client is running an operating system version of Windows 2000 and later.

[<26> Section 3.1.4.5](#Appendix_A_Target_26): Windows 2000 Server, Windows Server 2003, and Windows Server 2003 R2 do not match names in [**user principal name**](#gt_9d606f55-b798-4def-bf96-97b878bb92c6) form if ALL of the following are true:

* *LookupLevel* is either LsapLookupWksta or LsapLookupPDC.
* The server is a [**domain controller**](#gt_76a05049-3531-4abd-aec8-30e19954b4bd).
* *ClientRevision* is 0x00000001.
* The server is in a mixed [**domain**](#gt_b0276eb2-4e65-4cf1-a718-e0920a614aca) environment.

[<27> Section 3.1.4.5](#Appendix_A_Target_27): On a domain-joined, non-DC machine, when 0x80000000 is passed for the *LookupOptions* argument with a mix of isolated and composite names that cannot be matched in the views that are to be searched, Windows XP and later and Windows Server 2003 and later return STATUS\_SOME\_NOT\_MAPPED.

[<28> Section 3.1.4.6](#Appendix_A_Target_28): All versions of Windows that implement this method ([LsarLookupNames3](#Section_b75eaac5e4d14fc08dae61d838b38701)) also implement [LsarLookupNames4](#Section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361) (both in terms of client and server); hence, this method does not need to be implemented to interoperate with Windows clients or servers. The choice of which method to call depends on whether the client has a [**local security authority (LSA)**](#gt_9e5f2104-d6df-4ae7-8a5c-6bd14a0da8fa) policy handle or an RPC binding handle. Complete compatibility with Windows supports both calls.

[<29> Section 3.1.4.6](#Appendix_A_Target_29): The Windows implementation of the RPC server and RPC client limits the **Count** field of this structure to 1,000 (using the range primitive defined in [MS-RPCE]) in Windows XP SP2 and later and Windows Server 2003 and later. Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, Windows NT 4.0, Windows 2000, and Windows XP do not enforce this restriction.

[<30> Section 3.1.4.7](#Appendix_A_Target_30): A Windows RPC server can optionally be configured to deny this call, and the error returned in this case is STATUS\_NOT\_SUPPORTED.

[<31> Section 3.1.4.7](#Appendix_A_Target_31): The Windows RPC server and RPC client limit the **Count** field of this structure to 1,000 (using the range primitive defined in [MS-RPCE]) in Windows XP SP2 and later and Windows Server 2003 and later. Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, Windows NT 4.0, Windows 2000, and Windows XP do not enforce this restriction.

[<32> Section 3.1.4.8](#Appendix_A_Target_32): The Windows RPC server and RPC client limit the **Count** field of this structure to 1,000 (using the range primitive defined in [MS-RPCE]) in Windows XP SP2 and later and Windows Server 2003 and later. Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, Windows NT 4.0, Windows 2000, and Windows XP do not enforce this restriction.

[<33> Section 3.1.4.10](#Appendix_A_Target_33): The Windows RPC client sets LookupOptions to 0.

[<34> Section 3.2](#Appendix_A_Target_34): Windows clients negotiate the highest revision supported by the server by first calling the highest revision supported for that client. If the RPC exception that indicates that the function is out of range is returned from the server (exception number 0x6d1), the client proceeds to call the next lower revision. This process is repeated until the oldest possible revision supported by the client is invoked or until the server responds to the request.

[<35> Section 5.1](#Appendix_A_Target_35): The Windows RPC server for this protocol is customizable to allow anonymous callers to make requests for compatibility with Windows NT 3.1, Windows NT 3.5, Windows NT 3.51, and Windows NT 4.0 machines.

# Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

# Index

A

Abstract data model

[server](#section_fa772a2455d44911b3d5eda0bd2ec4e1) 26

[Abstract data model - server](#section_fa772a2455d44911b3d5eda0bd2ec4e1) 26

[Account domain principal view](#section_64ce2b94891f40f79bd6ffc5d7a5d662) 32

[Applicability](#section_98cb8a7dbc32459f94e6f1056144fb5a) 12

B

[Builtin domain principal view](#section_871df7fe34704b699e5915db70e8d4c6) 31

C

[Capability negotiation](#section_a4ae373ddb114db3a316f6b012a3466b) 12

[Change tracking](#section_2f53a714fd3343f5ad81686433140329) 80

[Client](#section_b208b01e3fe64c6db0bd8ecf412e1b87) 26

[Common data types](#section_25c6fd5062754f5893f7a0f22fc91a99) 13

D

Data model - abstract

[server](#section_fa772a2455d44911b3d5eda0bd2ec4e1) 26

[Data model - abstract - server](#section_fa772a2455d44911b3d5eda0bd2ec4e1) 26

[Data types](#section_25c6fd5062754f5893f7a0f22fc91a99) 13

[common - overview](#section_25c6fd5062754f5893f7a0f22fc91a99) 13

[Database views and scope - overview](#section_77855eedd3c74538b52720a36d4dcb18) 26

[Directory service schema elements](#section_c314306373b844a283998f9dd0af5255) 25

[Domain database information](#section_20a48d4dffed4cb994f88d29ff2014a8) 37

E

[Elements - directory service schema](#section_c314306373b844a283998f9dd0af5255) 25

Events

[local - server](#section_0cf7a79ab24b4b56a0ac783c125e0164) 57

[timer - server](#section_ed2ba7e2137b4a4abf3ec8935e060379) 57

[Example](#section_b1c74144daef4246803773922564e3d3) 59

F

[Fields - vendor-extensible](#section_b2c1390912aa4b1e8a661364cb413eae) 12

[Forest principal view](#section_80513499f3ac4ac88b9e0cd63589328e) 35

[Full IDL](#section_23cd602fe46c430dab5da678b9044a7d) 63

G

[Glossary](#section_e79f268084d94d34bc785ab9e1255653) 6

I

[IDL](#section_23cd602fe46c430dab5da678b9044a7d) 63

[Implementer - security considerations](#section_66bca9fa51ae42c58bf38ad4aa267741) 62

[Index of security parameters](#section_0da52cf8298147729a7b7c44ae368b15) 62

[Informative references](#section_247918e8d5504e02b98ed21f3f0f1198) 10

Initialization

[server](#section_9aa187c443d94491a9f35cf56800be4f) 39

[Initialization - server](#section_9aa187c443d94491a9f35cf56800be4f) 39

[Introduction](#section_1a7f31b641d542999c212f4bedcfced0) 6

L

Local events

[server](#section_0cf7a79ab24b4b56a0ac783c125e0164) 57

[Local events - server](#section_0cf7a79ab24b4b56a0ac783c125e0164) 57

[LSA\_TRANSLATED\_SID structure](#section_d067a76efd924e3691e78605814414cc) 18

[LSAP\_LOOKUP\_LEVEL enumeration](#section_9d1166ccbcfd4e22a8acf55eae57c99f) 19

[LSAPR\_ACL structure](#section_e1a273531eb54cabb88e43d18558daf2) 15

[LSAPR\_OBJECT\_ATTRIBUTES structure](#section_11b18f088f1940bd9794fc6bae3d7b05) 17

[LSAPR\_REFERENCED\_DOMAIN\_LIST structure](#section_3a52af31247a4b0891a01d46b2cc49b2) 18

[LSAPR\_SECURITY\_DESCRIPTOR structure](#section_8494008f0bfb45b8bb6ce32dd7f18e3d) 15

[LSAPR\_SID\_ENUM\_BUFFER structure](#section_1ffb61f0a4fe4487858dfb709d605855) 21

[LSAPR\_SID\_INFORMATION structure](#section_f04a771bc018409881b52a819f9b5db8) 20

[LSAPR\_TRANSLATED\_NAME structure](#section_52e1ccc1b57b4c02b35fbd64913ce99b) 21

[LSAPR\_TRANSLATED\_NAME\_EX structure](#section_6a9131acf1a74b968917a289a2c48de1) 22

[LSAPR\_TRANSLATED\_NAMES structure](#section_ff977eb9563a4353a95f640e7ee16356) 21

[LSAPR\_TRANSLATED\_NAMES\_EX structure](#section_5f2bd56ba16240339c2f7323fcf09e8c) 22

[LSAPR\_TRANSLATED\_SID\_EX structure](#section_274ee4e386b5445099e9f1678b9e96f5) 23

[LSAPR\_TRANSLATED\_SID\_EX2 structure](#section_e397a7ac842a4d4487ef170392dee522) 24

[LSAPR\_TRANSLATED\_SIDS structure](#section_8c3f03453cdd4ccbba6f8aac5e0da896) 19

[LSAPR\_TRANSLATED\_SIDS\_EX structure](#section_19250ed6447043259f85adb49db720fd) 23

[LSAPR\_TRANSLATED\_SIDS\_EX2 structure](#section_f55a62516071460089e77a7692e6aff4) 24

[LSAPR\_TRUST\_INFORMATION structure](#section_b0f34b28b5da44aaa60799c09e6526e1) 17

[LsarClose (Opnum 0) method](#section_be232cec04a9439ab718931c17c3ff0f) 43

[LsarClose method](#section_be232cec04a9439ab718931c17c3ff0f) 43

[LsarGetUserName (Opnum 45) method](#section_4e3f0a3f86ae4d50aa40ec25013a89e3) 43

[LsarGetUserName method](#section_4e3f0a3f86ae4d50aa40ec25013a89e3) 43

[LsarLookupNames (Opnum 14) method](#section_65d18faa0cb240eea94a2140212f4ec4) 51

[LsarLookupNames method](#section_65d18faa0cb240eea94a2140212f4ec4) 51

[LsarLookupNames2 (Opnum 58) method](#section_43254e9f32e248caa61de8decf41721e) 49

[LsarLookupNames2 method](#section_43254e9f32e248caa61de8decf41721e) 49

[LsarLookupNames3 (Opnum 68) method](#section_b75eaac5e4d14fc08dae61d838b38701) 48

[LsarLookupNames3 method](#section_b75eaac5e4d14fc08dae61d838b38701) 48

[LsarLookupNames4 (Opnum 77) method](#section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361) 44

[LsarLookupNames4 method](#section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361) 44

[LsarLookupSids (Opnum 15) method](#section_eb7ac899e697488393de1e60c7720c02) 56

[LsarLookupSids method](#section_eb7ac899e697488393de1e60c7720c02) 56

[LsarLookupSids2 (Opnum 57) method](#section_a09480f4145e4b4ca0275b313db61833) 54

[LsarLookupSids2 method](#section_a09480f4145e4b4ca0275b313db61833) 54

[LsarLookupSids3 (Opnum 76) method](#section_678dd7ed611b4d948f5ca06c76ae5c7e) 52

[LsarLookupSids3 method](#section_678dd7ed611b4d948f5ca06c76ae5c7e) 52

[LsarOpenPolicy (Opnum 6) method](#section_534c8a461be740659f87e60a1447ae45) 42

[LsarOpenPolicy method](#section_534c8a461be740659f87e60a1447ae45) 42

[LsarOpenPolicy2 (Opnum 44) method](#section_40a316ae1bae4aa192cd650cdaae1863) 42

[LsarOpenPolicy2 method](#section_40a316ae1bae4aa192cd650cdaae1863) 42

M

Message processing

[server](#section_d6b1c6d4ff5147dcb3b4af10e917ddd9) 39

[Message processing - server](#section_d6b1c6d4ff5147dcb3b4af10e917ddd9) 39

Messages

[common data types](#section_25c6fd5062754f5893f7a0f22fc91a99) 13

[transport](#section_f25df96b48c648848cf01418e3148c66) 13

[Messages - transport](#section_f25df96b48c648848cf01418e3148c66) 13

Methods

[LsarClose (Opnum 0)](#section_be232cec04a9439ab718931c17c3ff0f) 43

[LsarGetUserName (Opnum 45)](#section_4e3f0a3f86ae4d50aa40ec25013a89e3) 43

[LsarLookupNames (Opnum 14)](#section_65d18faa0cb240eea94a2140212f4ec4) 51

[LsarLookupNames2 (Opnum 58)](#section_43254e9f32e248caa61de8decf41721e) 49

[LsarLookupNames3 (Opnum 68)](#section_b75eaac5e4d14fc08dae61d838b38701) 48

[LsarLookupNames4 (Opnum 77)](#section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361) 44

[LsarLookupSids (Opnum 15)](#section_eb7ac899e697488393de1e60c7720c02) 56

[LsarLookupSids2 (Opnum 57)](#section_a09480f4145e4b4ca0275b313db61833) 54

[LsarLookupSids3 (Opnum 76)](#section_678dd7ed611b4d948f5ca06c76ae5c7e) 52

[LsarOpenPolicy (Opnum 6)](#section_534c8a461be740659f87e60a1447ae45) 42

[LsarOpenPolicy2 (Opnum 44)](#section_40a316ae1bae4aa192cd650cdaae1863) 42

N

[Normative references](#section_f8c55cb22aa04019b637f53a17a2eeb8) 9

O

[Overview (synopsis)](#section_7946e6e96b0044bfac1826bffeba2544) 10

P

[Parameters - security index](#section_0da52cf8298147729a7b7c44ae368b15) 62

[PLSA\_TRANSLATED\_SID](#section_d067a76efd924e3691e78605814414cc) 18

[PLSAPR\_ACL](#section_e1a273531eb54cabb88e43d18558daf2) 15

[PLSAPR\_OBJECT\_ATTRIBUTES](#section_11b18f088f1940bd9794fc6bae3d7b05) 17

[PLSAPR\_REFERENCED\_DOMAIN\_LIST](#section_3a52af31247a4b0891a01d46b2cc49b2) 18

[PLSAPR\_SECURITY\_DESCRIPTOR](#section_8494008f0bfb45b8bb6ce32dd7f18e3d) 15

[PLSAPR\_SID\_ENUM\_BUFFER](#section_1ffb61f0a4fe4487858dfb709d605855) 21

[PLSAPR\_SID\_INFORMATION](#section_f04a771bc018409881b52a819f9b5db8) 20

[PLSAPR\_TRANSLATED\_NAME](#section_52e1ccc1b57b4c02b35fbd64913ce99b) 21

[PLSAPR\_TRANSLATED\_NAME\_EX](#section_6a9131acf1a74b968917a289a2c48de1) 22

[PLSAPR\_TRANSLATED\_NAMES](#section_ff977eb9563a4353a95f640e7ee16356) 21

[PLSAPR\_TRANSLATED\_NAMES\_EX](#section_5f2bd56ba16240339c2f7323fcf09e8c) 22

[PLSAPR\_TRANSLATED\_SID\_EX](#section_274ee4e386b5445099e9f1678b9e96f5) 23

[PLSAPR\_TRANSLATED\_SID\_EX2](#section_e397a7ac842a4d4487ef170392dee522) 24

[PLSAPR\_TRANSLATED\_SIDS](#section_8c3f03453cdd4ccbba6f8aac5e0da896) 19

[PLSAPR\_TRANSLATED\_SIDS\_EX](#section_19250ed6447043259f85adb49db720fd) 23

[PLSAPR\_TRANSLATED\_SIDS\_EX2](#section_f55a62516071460089e77a7692e6aff4) 24

[PLSAPR\_TRUST\_INFORMATION](#section_b0f34b28b5da44aaa60799c09e6526e1) 17

[Preconditions](#section_098e6520bbda476b827ef953649544f4) 11

[Predefined translation database views](#section_e09da72ee6c94f91aa6468b0475719b6) 27

[Prerequisites](#section_098e6520bbda476b827ef953649544f4) 11

[Product behavior](#section_d27155722c9b4198ace9316a2fb5d7f5) 72

Protocol Details

[overview](#section_b208b01e3fe64c6db0bd8ecf412e1b87) 26

[PSECURITY\_QUALITY\_OF\_SERVICE](#section_ee3cadcdff914f45b80ced820db9c1dd) 16

[PSTRING](#section_d876ef8225c1423f93cb9d97547f2a4b) 15

R

[References](#section_a1de6d690f3d48b28200379313643b31) 9

[informative](#section_247918e8d5504e02b98ed21f3f0f1198) 10

[normative](#section_f8c55cb22aa04019b637f53a17a2eeb8) 9

[Relationship to other protocols](#section_540f52f169ee43168d075359d4b93cab) 11

S

[Schema elements - directory service](#section_c314306373b844a283998f9dd0af5255) 25

Security

[implementer considerations](#section_66bca9fa51ae42c58bf38ad4aa267741) 62

[parameter index](#section_0da52cf8298147729a7b7c44ae368b15) 62

[SECURITY\_IMPERSONATION\_LEVEL enumeration](#section_4797d0a17b714a22a5a08e2059aec239) 16

[SECURITY\_QUALITY\_OF\_SERVICE structure](#section_ee3cadcdff914f45b80ced820db9c1dd) 16

Sequencing rules

[server](#section_d6b1c6d4ff5147dcb3b4af10e917ddd9) 39

[Sequencing rules - server](#section_d6b1c6d4ff5147dcb3b4af10e917ddd9) 39

Server

[abstract data model](#section_fa772a2455d44911b3d5eda0bd2ec4e1) 26

[initialization](#section_9aa187c443d94491a9f35cf56800be4f) 39

[local events](#section_0cf7a79ab24b4b56a0ac783c125e0164) 57

[LsarClose (Opnum 0) method](#section_be232cec04a9439ab718931c17c3ff0f) 43

[LsarGetUserName (Opnum 45) method](#section_4e3f0a3f86ae4d50aa40ec25013a89e3) 43

[LsarLookupNames (Opnum 14) method](#section_65d18faa0cb240eea94a2140212f4ec4) 51

[LsarLookupNames2 (Opnum 58) method](#section_43254e9f32e248caa61de8decf41721e) 49

[LsarLookupNames3 (Opnum 68) method](#section_b75eaac5e4d14fc08dae61d838b38701) 48

[LsarLookupNames4 (Opnum 77) method](#section_2c0e8a6ddf0f4c77bbebf7cd0f6b2361) 44

[LsarLookupSids (Opnum 15) method](#section_eb7ac899e697488393de1e60c7720c02) 56

[LsarLookupSids2 (Opnum 57) method](#section_a09480f4145e4b4ca0275b313db61833) 54

[LsarLookupSids3 (Opnum 76) method](#section_678dd7ed611b4d948f5ca06c76ae5c7e) 52

[LsarOpenPolicy (Opnum 6) method](#section_534c8a461be740659f87e60a1447ae45) 42

[LsarOpenPolicy2 (Opnum 44) method](#section_40a316ae1bae4aa192cd650cdaae1863) 42

[message processing](#section_d6b1c6d4ff5147dcb3b4af10e917ddd9) 39

[sequencing rules](#section_d6b1c6d4ff5147dcb3b4af10e917ddd9) 39

[timer events](#section_ed2ba7e2137b4a4abf3ec8935e060379) 57

[timers](#section_a6f421e49ff4470b9286ab96a4f1a7f5) 38

[SID\_NAME\_USE enumeration](#section_171e9a878e014bd8a35e3468128c8fc4) 18

[Standards assignments](#section_f14d69a5e9464608a430d32f494fb3d9) 12

[STRING structure](#section_d876ef8225c1423f93cb9d97547f2a4b) 15

T

Timer events

[server](#section_ed2ba7e2137b4a4abf3ec8935e060379) 57

[Timer events - server](#section_ed2ba7e2137b4a4abf3ec8935e060379) 57

Timers

[server](#section_a6f421e49ff4470b9286ab96a4f1a7f5) 38

[Timers - server](#section_a6f421e49ff4470b9286ab96a4f1a7f5) 38

[Tracking changes](#section_2f53a714fd3343f5ad81686433140329) 80

[Translation database views - predefined](#section_e09da72ee6c94f91aa6468b0475719b6) 27

[Transport](#section_f25df96b48c648848cf01418e3148c66) 13

[Trusted domain and forest information](#section_e57ea3504633401b964d01a4ef4bd819) 38

V

[Vendor-extensible fields](#section_b2c1390912aa4b1e8a661364cb413eae) 12

[Versioning](#section_a4ae373ddb114db3a316f6b012a3466b) 12