MICROSOFT® PROFESSIONAL REFERENCE

MICROSOFT® WINDOWS™

Microsoft® Win32™ Programmer's Reference

VOLUME 2

System Services, Multimedia, Extensions, and Application Notes

Microsoft PRESS

Microsoft

VOLUME 2

Microsoft® Win32™ Programmer's Reference

Microsoft PRESS

PUBLISHED BY
Microsoft Press
A Division of Microsoft Corporation
One Microsoft Way
Redmond, Washington 98052-6399

Library of Congress Cataloging-in-Publication Data
Microsoft Win32 programmer's reference / Microsoft Corporation.
p. cm.
Includes indexes.
Contents: v. 1. Window management and graphics device interface -- v. 2. System services, multimedia, extensions, and application notes -- v. 3. Functions, A–G -- v. 4. Functions, H–Z -- v. 5. Messages, structures, and macros. ISBN 1-55615-515-8 (v. 1) -- ISBN 1-55615-516-6 (v. 2) -- ISBN 1-55615-517-4 (v. 3) -- ISBN 1-55615-518-2 (v. 4) -- ISBN 1-55615-519-0 (v. 5)
1. Windows NT. 2. Computer software--Development. 3. Microsoft Win 32. I. Microsoft Corporation.
QA76.76.O63M524 1993
005.4'469--dc20 93-15990
CIP

Printed and bound in the United States of America.

1 2 3 4 5 6 7 8 9 AG-M 8 7 6 5 4 3

Distributed to the book trade in Canada by Macmillan of Canada, a division of Canada Publishing Corporation.

Distributed to the book trade outside the United States and Canada by Penguin Books Ltd.

Penguin Books Ltd., Harmondsworth, Middlesex, England
Penguin Books Australia Ltd., Ringwood, Victoria, Australia
Penguin Books N.Z. Ltd., 182-190 Wairau Road, Auckland 10, New Zealand

British Cataloging-in-Publication Data available.

U.S. Patent No. 4974159

Document No. PC52821-0593

49.1 About Security

The security provisions of Microsoft Windows NT are available to Windows-based applications automatically. Every application running on the system is subject to the security imposed by the particular configuration of the local implementation of Windows NT.

The security functions in the Win32 application programming interface (API) allow an application to selectively grant and deny access to an object. An application can specify many different kinds of access for particular users and groups of users. The operating system grants or denies access to an object based on a comparison of the security provisions stored with an object with the access rights specified in a token associated with the process or thread requesting the access. These security functions allow an application to query and manipulate the security features of both an object and a process or thread.

The impact of Windows security on most Windows functions is minimal, and a Windows-based application not requiring security functionality usually does not need to incorporate any special code. However, a developer can use the security features of Windows NT to provide a number of services in a Windows-based application. Generally, any application that manipulates a system-wide resource such as the system time, must use the security system to gain access to that resource. A security-aware application might allow the user to query the security attributes of a file, provide specialized feedback when access to a secure file is denied, or customize the security attributes of a file or group of files so that only a subset of other users on a network has access to the information.

The first release of Windows NT is designed to support C2-level security as defined by the US Department of Defense. Some of the most important requirements of C2-level security are shown in the following list.

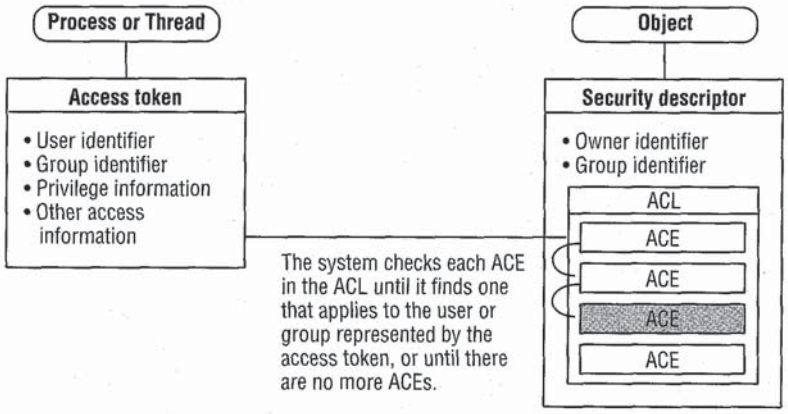
- It must be possible to control access to a resource. This access control must include or exclude individual users or named groups of users.
- Memory must be protected so its contents cannot be read after it is freed by a process.
- Users must identify themselves in a unique manner when they log on. All auditable actions must identify the user performing the action.
- System administrators must be able to audit security-related events. Access to this audit data must be limited to authorized administrators.
- The system must protect itself from external interference or tampering, such as modification of the running system or of system files stored on disk.

49.2 Security Model

All named objects in Windows NT, and some unnamed objects, can be secured. The security attributes of each securable object in Windows are described by a *security descriptor*, that contains information about the owner of the object and by an *access-control list* (ACL), identifying the users and groups allowed or denied access to the object. An ACL contains an entry for each user, global group, or local group (alias) being allowed or denied access to the object. Each of these entries is an *access-control entry* (ACE).

At logon, a user is assigned an *access token* containing identifiers that represent the user and any groups to which the user belongs. Every process run on behalf of this user will have a copy of this particular access token. When a process attempts to use an object, the system compares the security attributes listed in the access token with the ACEs in the object's ACL. The system compares the access token with each ACE until access is either granted or denied or until there are no more ACEs to check. Conceivably, several ACEs could apply to a token. And, if this occurs, the access rights granted by each ACE accumulate. For example, if one ACE grants read access to a group in an access token and another ACE grants write access to the user, who is also a member of the group, the user will have both read and write access to the object when the access check is complete.

The following illustration shows the relationship between these blocks of security information:



Typically, the application protecting an object is a server in that it defines the users and groups with access to the object. The application interacts with clients when they attempt to gain access to the object. Users and groups are identified by *security identifiers* (SIDs). An SID is a structure of variable length that uniquely identifies a user or group. SIDs are stored in a security database that an application can query by calling Win32 functions. With one exception, an SID is

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.