Windows Security Components

The components that built the Windows security environment

Overview

- Access Control
- Principals
- Security Descriptor
- Security Identifier
- Discretionary Access Control List
- System Access Control List
- Object Creation Rules
- Privileges
- Auditing

Access Control Matrix

- Map domains with objects
    - Every process is assigned one domain
- Elements
    - Column : Subject/principal/group
    - Row : Object/resource
    - Cell : Right/permission
- Focus
    - CL : Capability List - all rights, one subject (User focused)
    - ACL : Access Control List - all subjects, one object (Object focused)

Access Control Matrix

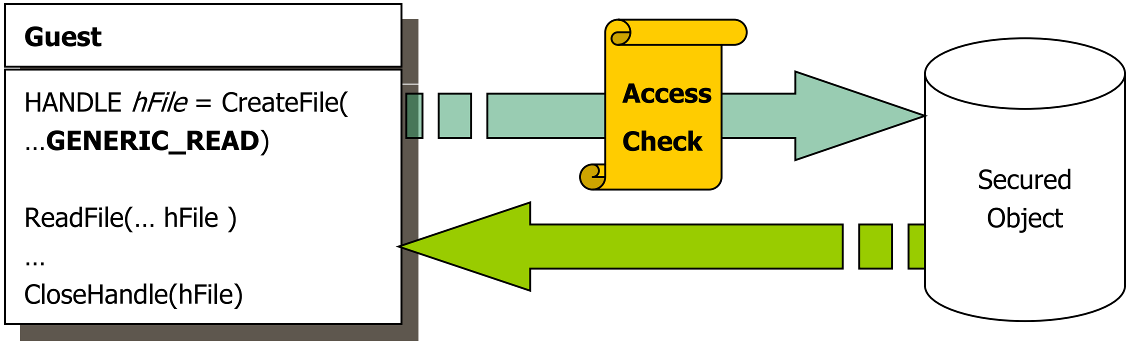
domains

objects

	Sid1	Sid2	Sid3	Sid4	…
File	Read, write	Read	Execute, delete	Write	
Directory	Read	Read	Read	Read	
Mutex	Write	Write, execute			Synchronize
Process					
…					

Access Control

- Strategy is a three variables equation
  1. User token = Security *context* of a process/thread – (Who)
  2. Access Mask = Access desired – (What, intention)
  3. Security Descriptor = List of *rules* associated an object
- On success, a *handle* stores access permissions
  - Security < tradeoff > performance
- New handle must be used for new permission
  - Open/Close/Open



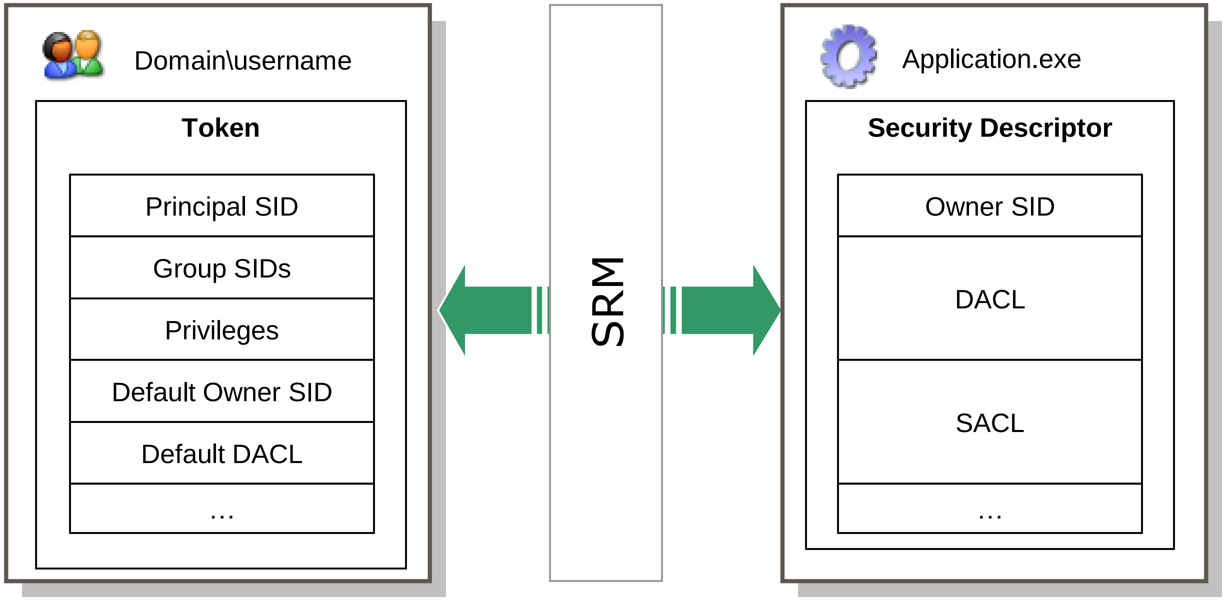
Guest

```
HANDLE hFile = CreateFile(
...GENERIC_READ)

ReadFile(... hFile )
...
CloseHandle(hFile)
```

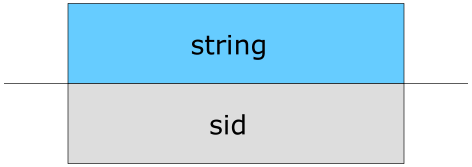
Accessing a protected Object

- Chek of identity and group membership - Who
- Check of permissions – What
- Check of privileges



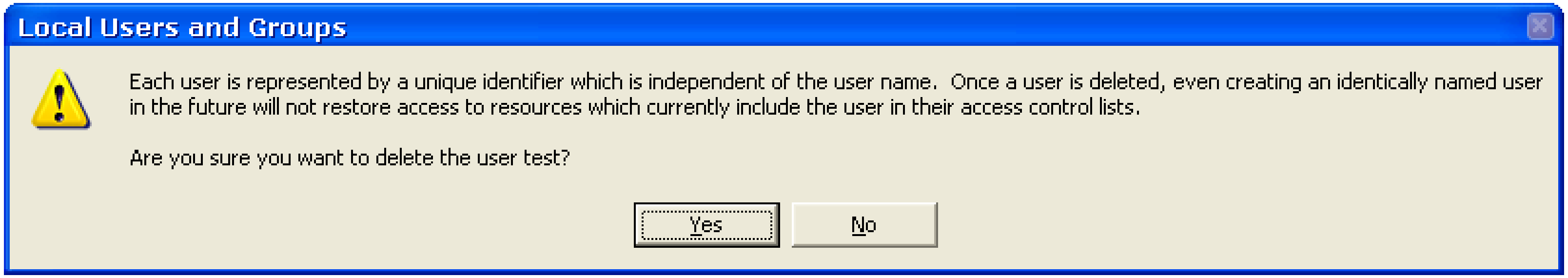
Principal

- A Principal is an entity that can prove his identity
  - User, group, machine, domain
- A principal must have an existing account
- A principal is uniquely identified in time and space
  - DomainA\Jim
  - DomainB\Jim
  - Computer\Jim
- Principals names are language independent
  - Administrator
  - Administrateur
  - Spanish, Chineese, ...
- Principals have two names
  - Human-readable - unique within restricted scope
  - Machine-readable - unique in space and time



Security Identifier

- Motivation
  - Localization and built-in name
  - User renaming and movements



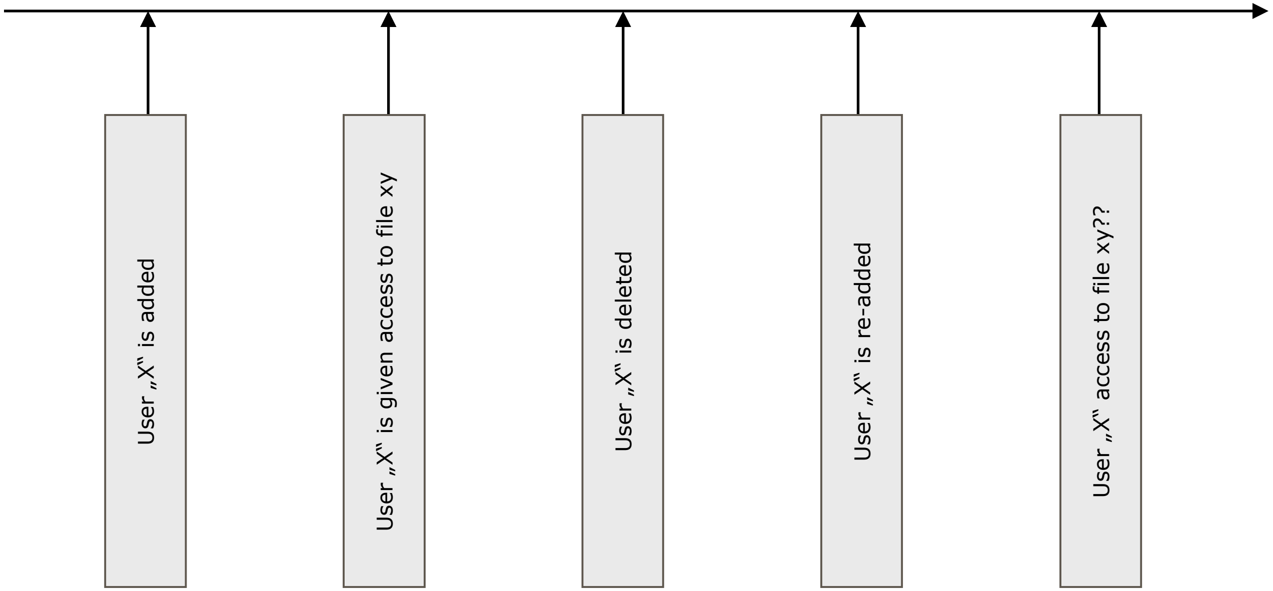
Each user is represented by a unique identifier which is independent of the user name. Once a user is deleted, even creating an identically named user in the future will not restore access to resources which currently include the user in their access control lists.

Are you sure you want to delete the user test?

- Solution
  - Accounts are internally represented by an alphanumeric value
    - Fully and uniquely (space and time) identification of a principal
    - When a principal logs on, the SID is retrieved from the SAM
  - Renaming an account as no effect

Security Identifier

- Life-time of a Principal



User „X" is added

User „X" is given access to file xy

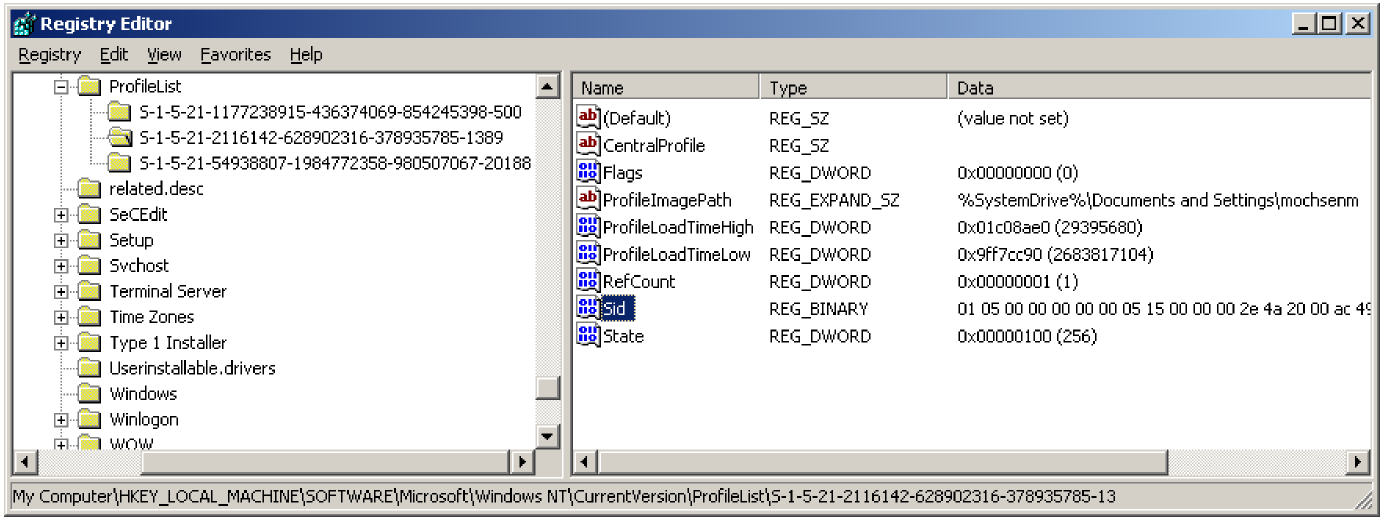
User „X" is deleted

User „X" is re-added

User „X" access to file xy??

Security Identifier - Discovery

- Groups and Users names are easy to collect remotely
  - Connect –> discover –> attack...
  - User2sid, Sid2user
  - Whoami
  - Getsid



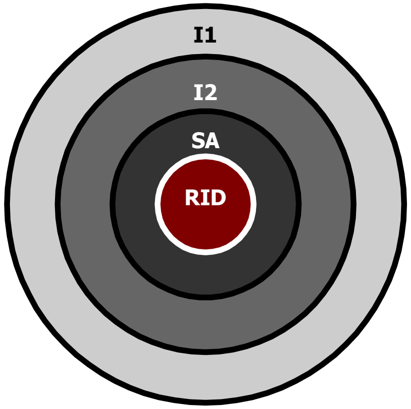
```
ProfileList
    S-1-5-21-1177238915-436374069-854245398-500
    S-1-5-21-2116142-628902316-378935785-1389
    S-1-5-21-54938807-1984772358-980507067-20188
related.desc
SeCEdit
Setup
Svchost
Terminal Server
Time Zones
Type 1 Installer
Userinstallable.drivers
Windows
Winlogon
WOW
```

Name	Type	Data
(Default)	REG_SZ	(value not set)
CentralProfile	REG_SZ	
Flags	REG_DWORD	0x00000000 (0)
ProfileImagePath	REG_EXPAND_SZ	%SystemDrive%\Documents and Settings\mochsenm
ProfileLoadTimeHigh	REG_DWORD	0x01c08ae0 (29395680)
ProfileLoadTimeLow	REG_DWORD	0x9ff7cc90 (2683817104)
RefCount	REG_DWORD	0x00000001 (1)
Sid	REG_BINARY	01 05 00 00 00 00 00 05 15 00 00 00 2e 4a 20 00 ac 4
State	REG_DWORD	0x00000100 (256)

My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\ProfileList\S-1-5-21-2116142-628902316-378935785-13

Security Identifier - Format

- S – R – I1 – I2 - SA – SA – SA - RID



I1: Authority (Space Uniqueness)

48-bit Identifier value, Agent that issued the SID

Authority	Value
World	1
..	2
..	3
NT	5

I2 : Tag

Tag	Value
BUILTIN	21
UNIQUE	32

SA: Sub-Authority (Time Uniqueness)

Machine unique 96 bit value, indicates trustee relationship to the issuing Authority

RID: Relative Identifier (Non-uniqueness)

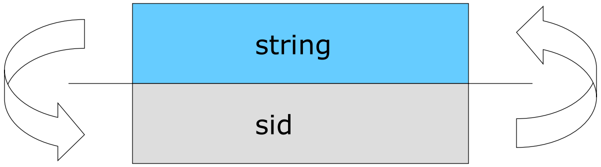
Users	RID
...\Administrator	500 (0x000001F4L)
...\Guest	501 (0x000001F5L)
...	

Groups	RID
...\Administrators	512 (0x00000200L)
...\Users	513 (0x00000201L)
...\Guests	514 (0x00000202L)
...	

New Principal	RID
Domain\Name	**1000++**
...	

Security Identifier - Translation Service

- LookupAccountName(
```
          SystemName,      // in
          AccountName,     // in
          Sid,             // out
          DomainName,..);  // in
```
- LookupAccountSid(
```
          SystemName,      // in
          Sid,             // in
          Name,            // out
          DomainName,..);  // in
```

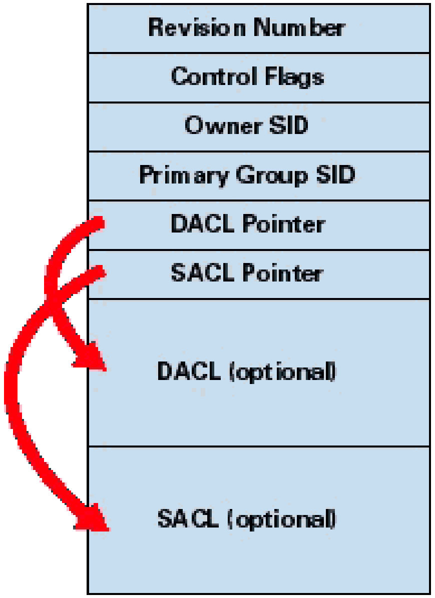


Security Descriptor

- Collection of security information associated with an object describing its security policy
- Second part of the objects security equation
- Contains any, all or none of
  - Object's owner SID
  - Discretionary Access-Control List (DACL) - the owner of an object specifies the access control policy for that object at his/her discretion (hence the name DACL)
  - System Access-Control List (SACL)
- Access control policy is specified as an access control list

Security Descriptor - Anatomy

- Revision Number
  - Version of SRM that creates the SD
- Control Flags
  - Inheritance, protection (isolation)
- Owner SID
  - Object’s owner
- Group SID
  - Posix standard requires that an object can be owned by a group (not used)
- DACL
  - Who has what access to an object
- SACL
  - Which operation by which user should be audited
- SD comes in two flavours
  - Absolute - fixed-length structure which contains pointers to other structures (system use)
  - Relative - Variable-length structure which contains offsets (persistency – registry…, wire transmission)



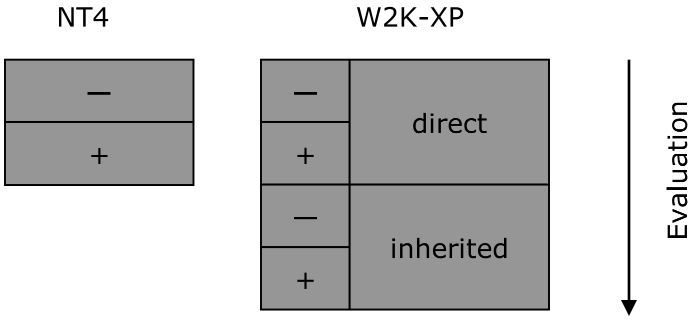
Discretionary Access Control List

- "Who can do what" list
- List of zero or more Access Control Entries (ACEs)
- An ACE has four fields of information
  - Type (Denied "-" or Allowed "+")
  - SID (Principal/Trustee)
  - Permission Mask
  - Inheritance flags (Directory/File)

Type	SID	Permission	Inheritance
+ (allow)	Everyone	R	Propagate ACEs
+	Friends	W	Isolate object
- (deny)	John	RD	…

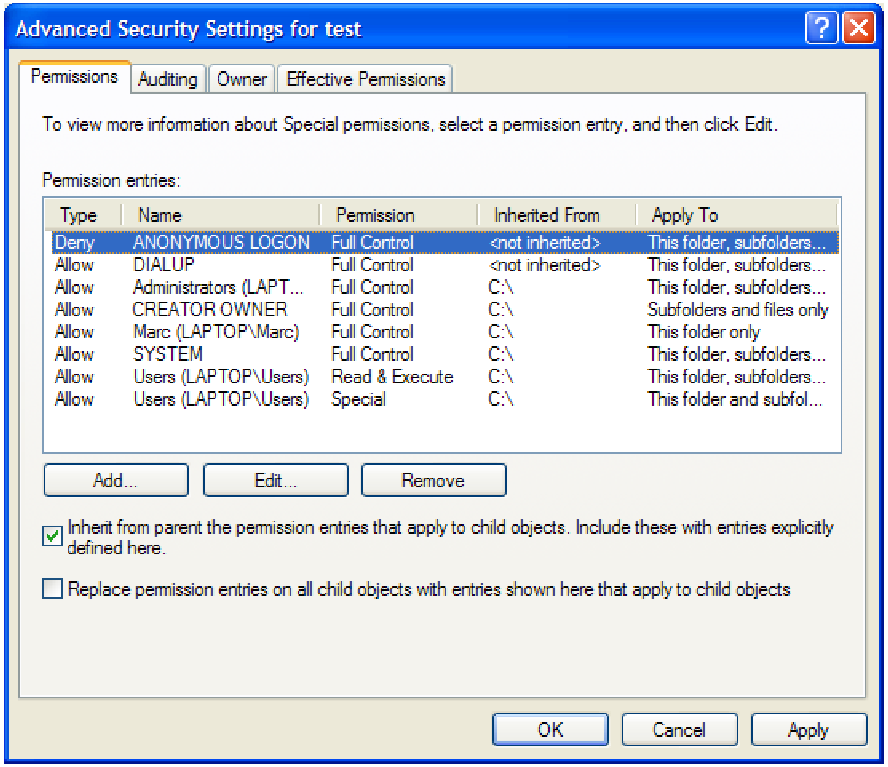
Discretionary Access Control List

- Top to bottom evaluation looking for requested access, and stops immediately when:
  - Any requested access has been (directly/indirectly) explicitly denied
  - All requested access have been (directly/indirectly) explicitly granted
- Ordering
  - negative ACE (deny)
  - positive ACE (allow)
- Inheritable ACEs
  - Direct precedes indirect (inherited)



Discretionary Access Control List

- Working with the DACL and SACL editor



Advanced Security Settings for test

To view more information about Special permissions, select a permission entry, and then click Edit.

Permission entries:

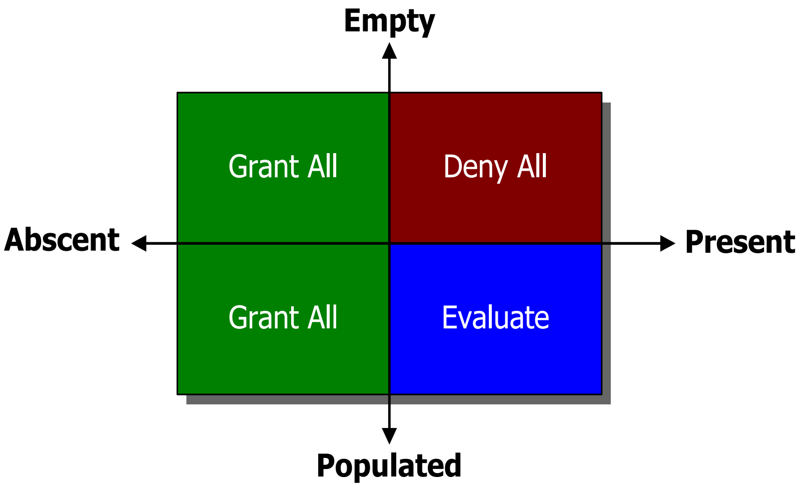
Type	Name	Permission	Inherited From	Apply To
Deny	ANONYMOUS LOGON	Full Control	<not inherited>	This folder, subfolders...
Allow	DIALUP	Full Control	<not inherited>	This folder, subfolders...
Allow	Administrators (LAPT...	Full Control	C:\	This folder, subfolders...
Allow	CREATOR OWNER	Full Control	C:\	Subfolders and files only
Allow	Marc (LAPTOP\Marc)	Full Control	C:\	This folder only
Allow	SYSTEM	Full Control	C:\	This folder, subfolders...
Allow	Users (LAPTOP\Users)	Read & Execute	C:\	This folder, subfolders...
Allow	Users (LAPTOP\Users)	Special	C:\	This folder and subfol...

☑ Inherit from parent the permission entries that apply to child objects. Include these with entries explicitly defined here.

□ Replace permission entries on all child objects with entries shown here that apply to child objects

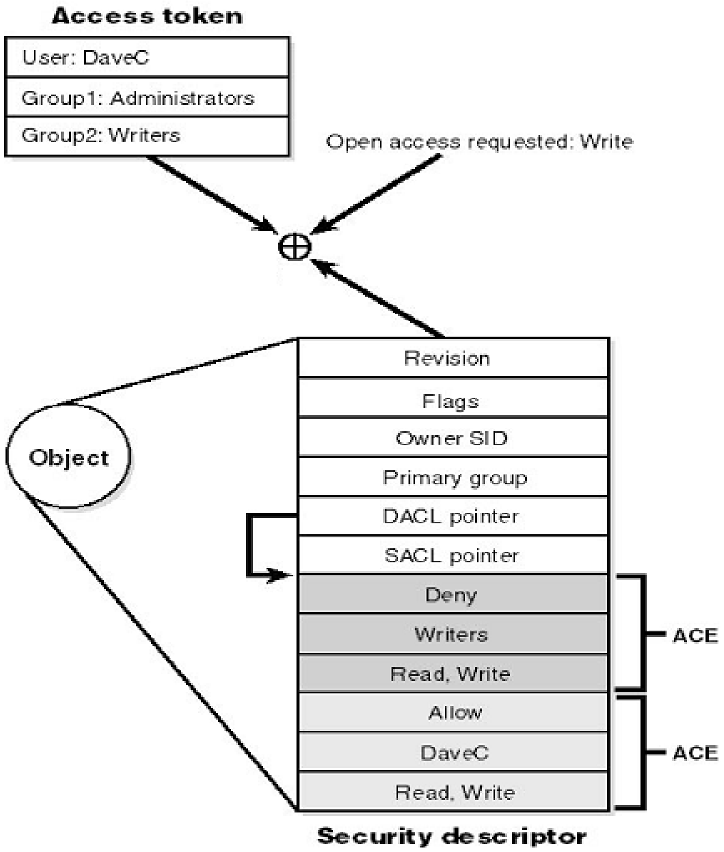
DACL Evaluation

- Empty DACL denies access to everyone
- Null DACL grants full control access to everyone
- No DACL grants full control access to everyone
- Populated DACL evaluates the access control



Access Check in Action

- Equation of three inputs…
  1. Access Token
  2. Access Request (Intention)
  3. Object’s Security Descriptor

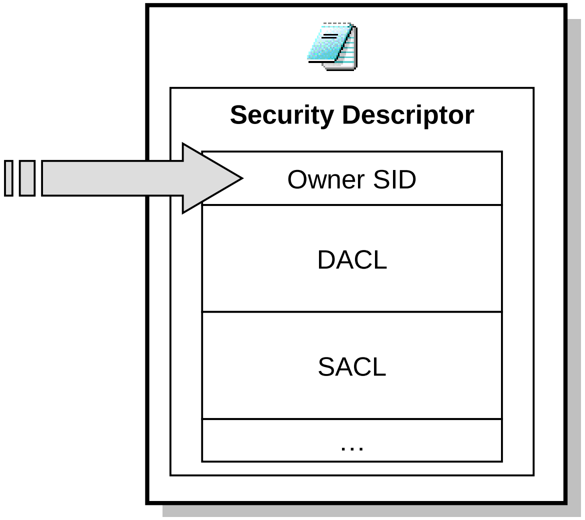


System Access Control List

- Generated audits are located in the Events Log
- List of “who should be audited for what specific action”
- A SACL in not discretionary
  - ONLY Administrator, or user with SeSecurityPrivilege permission, can access the SACL
- SACL ordering
  - Positive ACE = Audit on success
  - Negative ACE = Audit on failure
  - An entry can be both positive and negative
- Order is not important

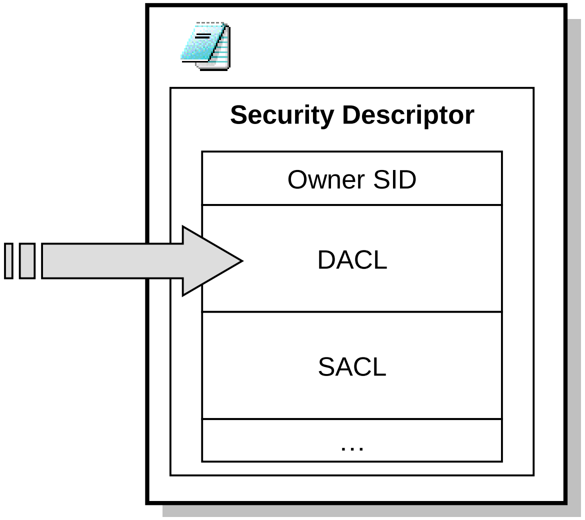
Object Creation – Owner Rules

- …

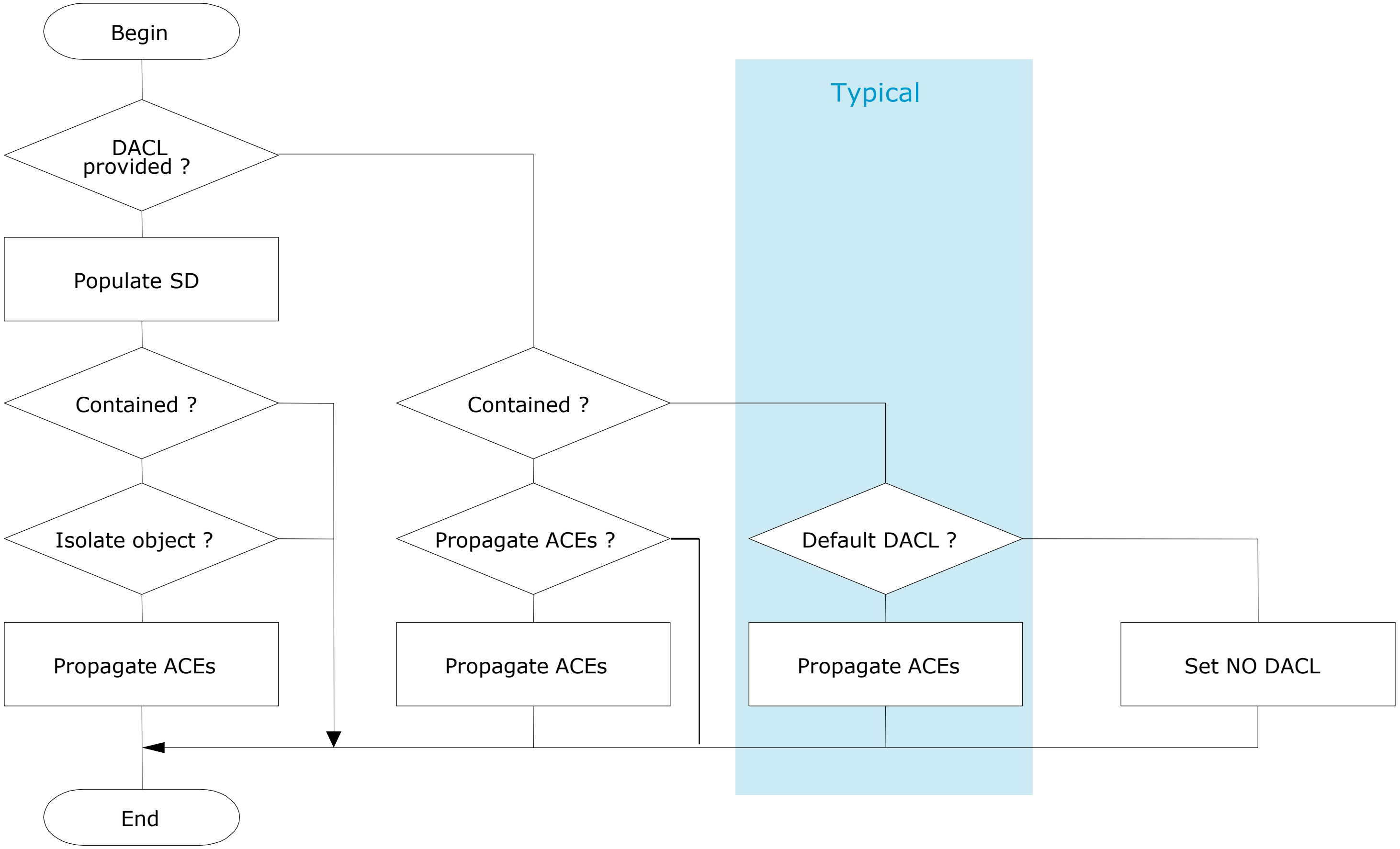


Object Creation – DACL Rules

- The way a DACL is computed for a new object obeys complex rules
  - A DACL has been provided
  - A DACL has not been provided
  - The object is contained in another one
  - The container is marked to propagate its ACEs
  - The object is marked as to be isolated



Object Creation – DACL Rules

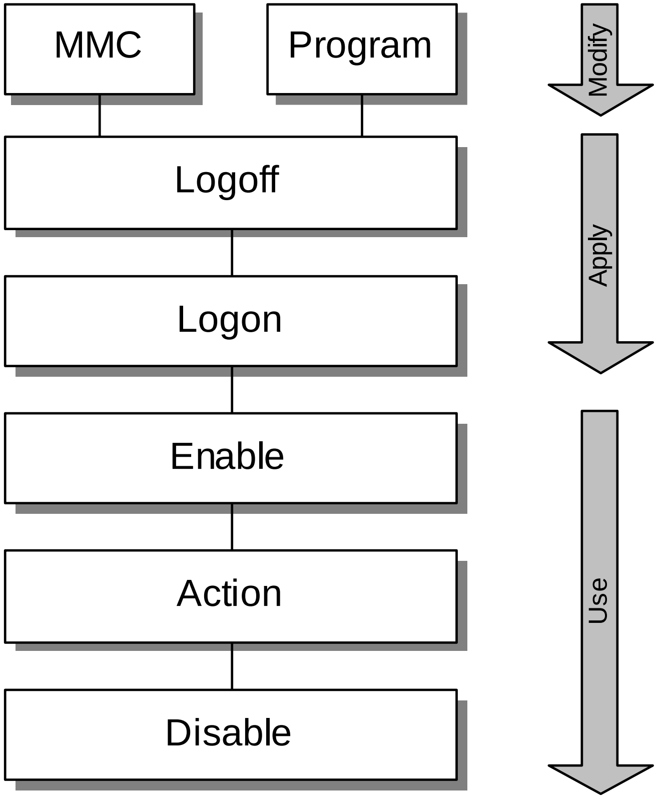


Privileges

- User‵s right to perform specific tasks that usually affect the entire computer rather particular objects
  - Backup (read/write)
  - Shut-down the machine
  - Debug a program (attach to a process...and kill it!)
  - Change the system time
  - Be part of the TCB (logon creation)
  - Bypass Traverse checking (security <> performance)
- User‵s right to access system resources (global scope)
  - Load a driver
  - Increase quotas
- Privileges are injected in token ONLY at authentication time
- Privileges are cached in token
  - Granting a new privilege has absolutely no effect on existing session
- Privileges are granted relative to the local machine

Privileges Management

- Two-tier mechanism
  - Privilege must be present
  - Privileges must be enabled
- Privileges cannot be added
  - Token must be refreshed
    - Logoff/on for interactive session
    - Shut down, start service session
- Privileges can only be switched on/off
  - OpenTreadToken(..)
  - AdjustTokenPrivileges(..)
  - ...perform action
  - AdjustTokenPrivileges(..)
- Many privileges usage are not audited
- Fixed numbers/types
- Applications cannot introduce new privileges

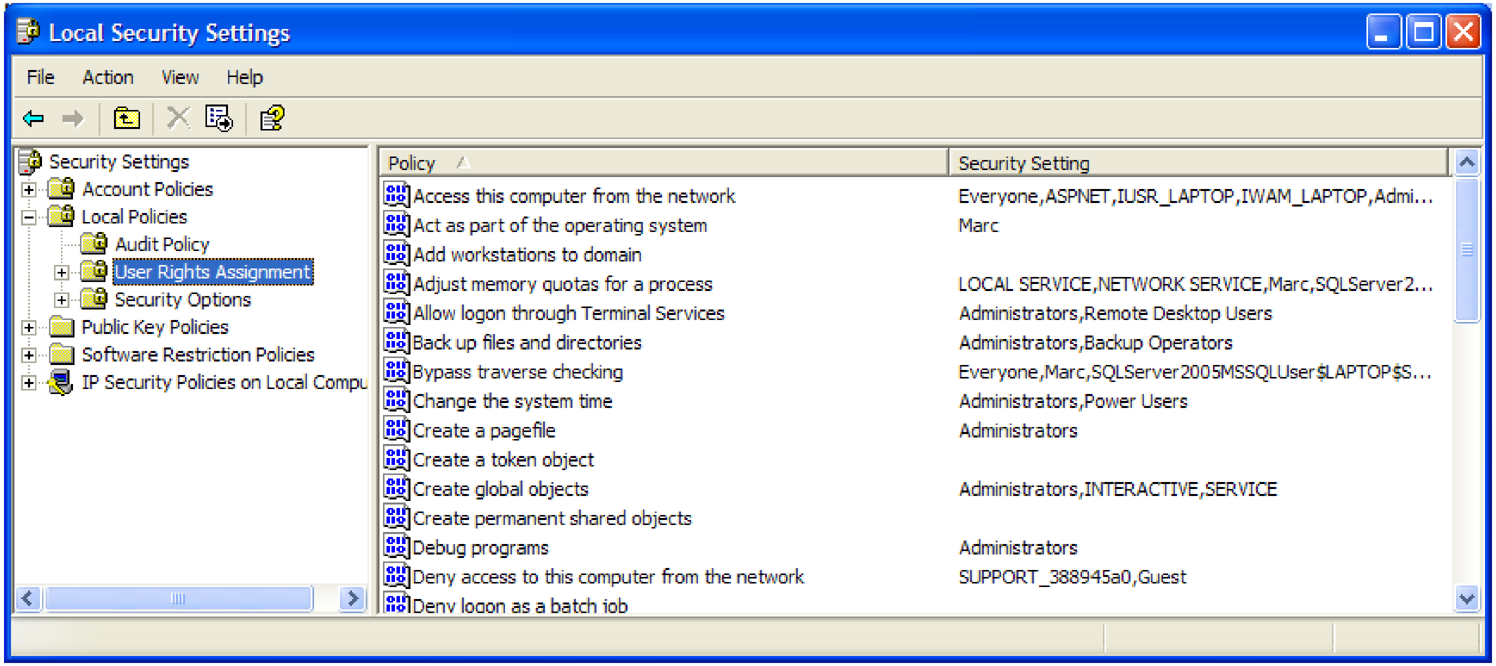


Privileges Names

Name	Description
SeBackupPrivilege	Back up files an directories
SeChangeNotifyPrivilege	Bypass traverse checking
SeCreateDebugPrivilege	Debug programs
SeIncreaseQuotaPrivilege	Increase quotas
SeInteractiveLogonRight	Logon locally to an NT system
SeLoadDriverPrivilege	Load and unload Device drivers
SeMachineAccountPrivilege	Add workstations to a domain
SeNetworkLogonRight	Access the system from a network
SeRemoteShutdownPrivilege	Force the shutdown of a remote system
SeRestorePrivilege	Restore files and directories
SeSecurityPrivilege	Manage auditing and security log
SeShutdownPrivilege	Shut down the system
SeSystemProfilePrivilege	Profile system performance
SeSystemtimePrivilege	Change the system time
SeTakeOwnershipPrivilege	Take ownership of securable objects
SeTcbPrivilege	Act as part of the operating system
SeUndockPrivilege	Remove the computer from the docking station

Privileges Management

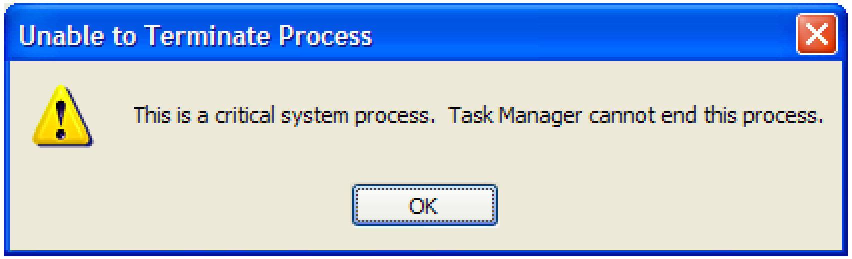
- Privileges are assigned by administrators to individuals or groups
- User Rights and Privileges are synonymous since both are related to principal(s) behind a process



Policy	Security Setting
Access this computer from the network	Everyone,ASPNET,IUSR_LAPTOP,IWAM_LAPTOP,Admi...
Act as part of the operating system	Marc
Add workstations to domain	
Adjust memory quotas for a process	LOCAL SERVICE,NETWORK SERVICE,Marc,SQLServer2...
Allow logon through Terminal Services	Administrators,Remote Desktop Users
Back up files and directories	Administrators,Backup Operators
Bypass traverse checking	Everyone,Marc,SQLServer2005MSSQLUser$LAPTOP$S...
Change the system time	Administrators,Power Users
Create a pagefile	Administrators
Create a token object	
Create global objects	Administrators,INTERACTIVE,SERVICE
Create permanent shared objects	
Debug programs	Administrators
Deny access to this computer from the network	SUPPORT_388945a0,Guest
Deny logon as a batch job	

Privileges Usage

- System protects the administrator to hurt himself
  - Taskmgr cannot kill somes services and system process
  - Administrator „Access denied“!?
  - When run by an administrator, taskmgr´s token includes SeDebugPrivilege, but it is disabled



Unable to Terminate Process

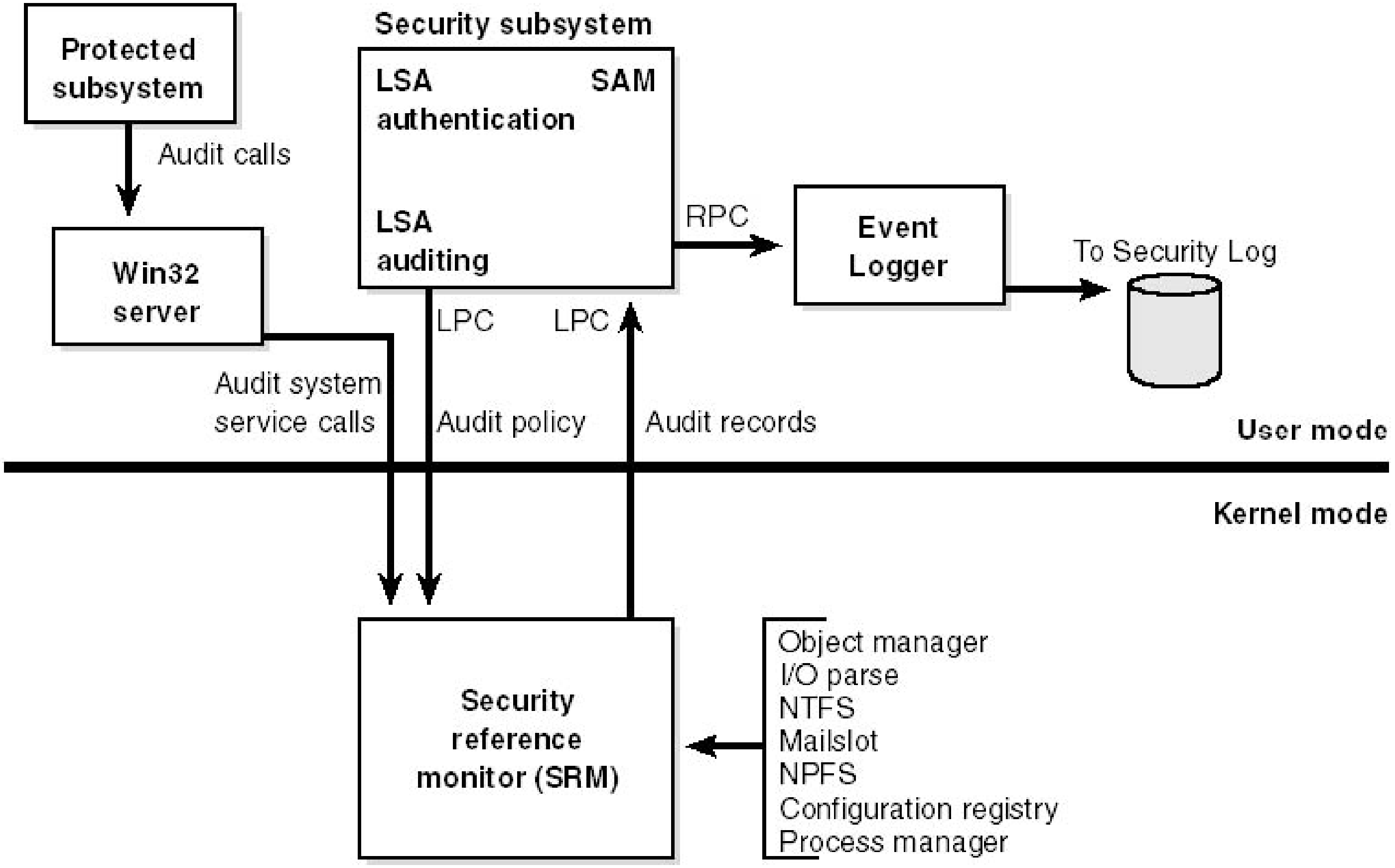
This is a critical system process. Task Manager cannot end this process.

- Modify the token associated with taskmgr
  - PVIEW
  - KILL

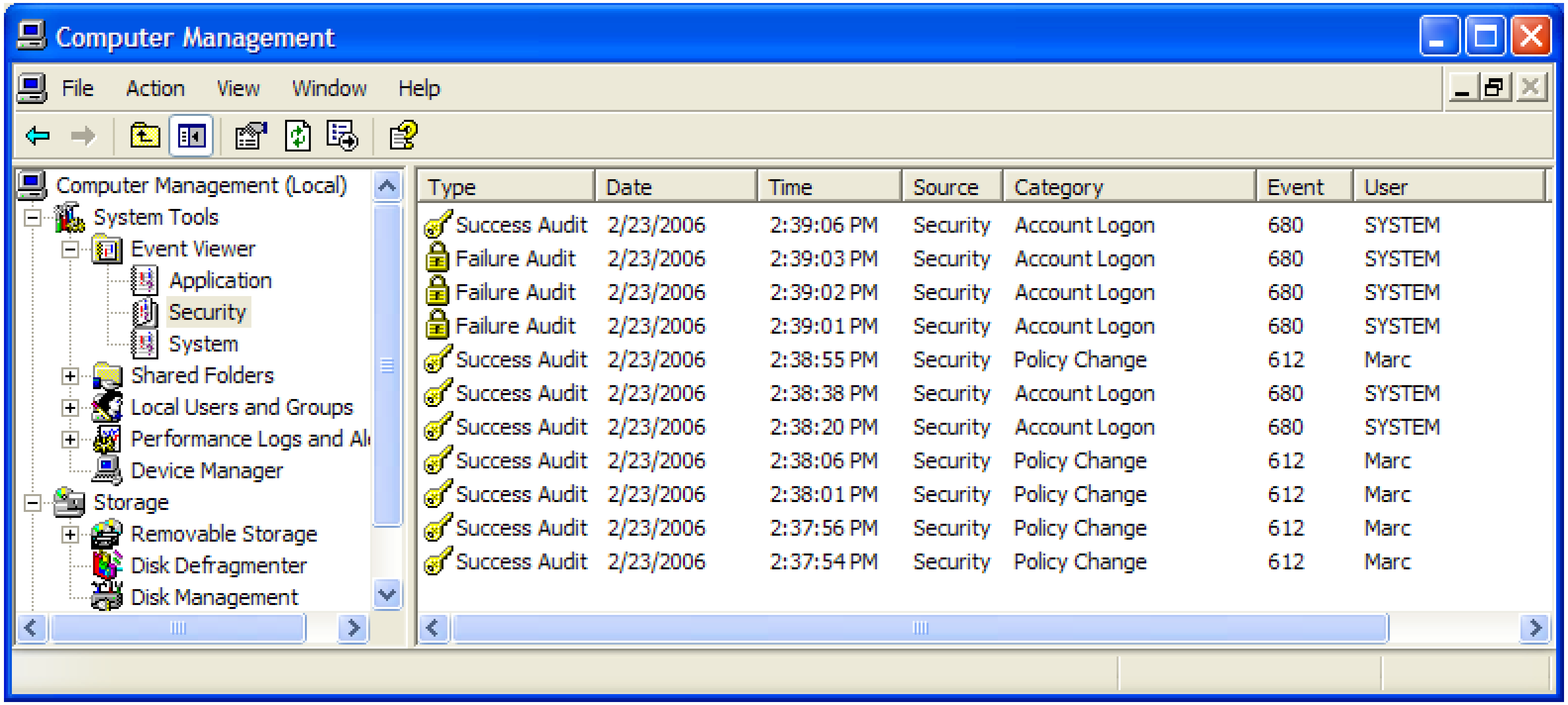
Auditing

- Definition
  - The „other side‟ of security (protection/monitoring)
  - Monitor security-related activity (success, failures)
  - Services are a primary security exposure
- Types
  - User Logons
  - Objects tracking/creation/accesses
    - file,directory, process, services, registry, printer, mutex....
    - Memory consumption
    - Network problem
  - Policy changes
  - Use of privileges (backup, system time....)
- Two-steps process
  - Set up the audit policy (kinds of events to be audited)
  - Configure actual objects to which the auditing will be applied

Flow of auditing records



Viewing Auditing Events



Type	Date	Time	Source	Category	Event	User
Success Audit	2/23/2006	2:39:06 PM	Security	Account Logon	680	SYSTEM
Failure Audit	2/23/2006	2:39:03 PM	Security	Account Logon	680	SYSTEM
Failure Audit	2/23/2006	2:39:02 PM	Security	Account Logon	680	SYSTEM
Failure Audit	2/23/2006	2:39:01 PM	Security	Account Logon	680	SYSTEM
Success Audit	2/23/2006	2:38:55 PM	Security	Policy Change	612	Marc
Success Audit	2/23/2006	2:38:38 PM	Security	Account Logon	680	SYSTEM
Success Audit	2/23/2006	2:38:20 PM	Security	Account Logon	680	SYSTEM
Success Audit	2/23/2006	2:38:06 PM	Security	Policy Change	612	Marc
Success Audit	2/23/2006	2:38:01 PM	Security	Policy Change	612	Marc
Success Audit	2/23/2006	2:37:56 PM	Security	Policy Change	612	Marc
Success Audit	2/23/2006	2:37:54 PM	Security	Policy Change	612	Marc

Summary

- A secured object has always an owner
- A process always runs on behalf of a principal
- A principal is always assigned to a token
- A principal is uniquelly identified with a SID
- A Security Descriptor is always assigned to an protected object
- Access check occurs only when opening an object
- Privilege is related to actions not to specific objects
- Audit is an essential part of the security

Links

- Programming NT Security (Addison-Wesley, Keith Brown)
- Windows NT Security (R&D Books Miller Freeman, N.Okuntseff)
- Windows NT Security Guide (Addison Wesley, Stephen A. Sutton)
- Microsoft Windows Internals fourth Edition, (Microsoft Press, D.Solomon, M.Russinovich)
- Secure Networking with Windows 2000 and Trust Services (Addison Wesley, Jalal Feghhi and Jalil Feghhi)
- Microsoft Windows 2000 Security Handbook (Que, Jeff Schmidt)
- Modern Operating Systems – Second Edition (Prentice Hall, Tanenbaum)