

# Security

„Ausgewählte Betriebssysteme“  
Institut Betriebssysteme  
Fakultät Informatik

# Outline

- Security Ratings
- Security System Components
- Logon
- Object (File) Access
- Impersonation
- Auditing

# Security Ratings

- National Computer Security Center (NCSC) part of US Department of Defense (DoD)
- Defined 1983 DoD's Trusted Computer System Evaluation Criteria (TCSEC)
- TCSEC commonly referred to as "Orange Book"
- 1996 US, UK, GER, F, CA, NED developed Common Criteria for Information Technology Security Evaluation (CCITSE)
- CCITSE commonly referred to as "Common Criteria"

# TCSEC Rating Levels

Rating	Description
A1	Verified Design
B3	Security Domains
B2	Structured Protection
B1	Labeled Security Protection
C2	Controlled Access Protection
C1	Discretionary Access Protection (obsolete)
D	Minimal Protection

# Key Requirements for C2

- Secure Logon Facility:
  - Users can be uniquely identified and
  - Access granted only after identification
- Discretionary Access Control:
  - Owner of resource determines access to resource
- Security Auditing:
  - Detect and record security related events
  - Detect and record attempt to create, access or delete system resources
- Object reuse protection:
  - Prevent users from seeing data of other users

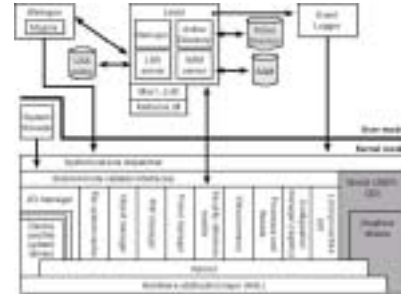
# Met B-Level Requirements

- Trusted Path Functionality:
  - Prevent interception of user names and passwords (SAS)
- Trusted Facility Management:
  - Requires support for separate account roles (Administrator, Backup, Standard user, ...)

## Outline

- Security Ratings
- Security System Components
- Logon
- Object (File) Access
- Impersonation
- Auditing

## Security System Components



Picture © Mark Russinovich & David Solomon (used with permission of authors)

## Security System Components (2)

- Security Reference Monitor (SRM)
  - Runs in kernel mode
  - Performs security checks on objects
  - Manipulates privileges
  - Generates security audit messages
- Local Security Authority Subsystem (Lsass):
  - Responsible for local system security policy
  - User authentication
  - Send security audit messages to event log
  - Most of it implemented in LSA service (Lsassrv.dll)

## Security System Components (3)

- Lsass Policy Database
  - Contains local system security policy settings
  - Stored in registry under HKLM\SECURITY
  - Contains trusted domains, who has access and how (local, network, service), who has what privileges
  - Contains secrets, such as logon information and Win32 service user logons
- Security Accounts Manager (SAM)
  - Manages database containing user names and groups

## Security System Components (4)

- SAM database
  - Contains local users and groups along with passwords and other attributes
  - Stored in HKLM\SAM
- Active Directory
  - Manages database that stores information about objects in a domain (users, groups, computers, ...)
- Authentication Packages
  - DLL implementing authentication policy
  - Responsible for checking match of username and password

## Security System Components (5)

- Logon Process
  - Responds to secure attention sequence (SAS)
  - Manages interactive logon sessions
- Graphical Identification and Authentication (GINA)
  - Used to obtain username and password (or similar)
- Netlogon
  - Responds to network logon requests
- Kernel Security Device Driver (KSecDD)
  - Provides mechanism for kernel components to communicate with Lsass

## Communication Kernel – User

- During boot Lsass communicates with SRM using ports (both listen on ports)
- Establish shared memory region and stop listening on port (port is connected)



Picture © Mark Russinovich & David Solomon (used with permission of authors)

13

## Outline

- Security Ratings
- Security System Components
- Logon
- Object (File) Access
- Impersonation
- Auditing

Ausgewählte Betriebssysteme - Security

14

## Winlogon Initialization

1. Create and open interactive window station to represent keyboard, mouse, and monitor
  - Create security descriptor for itself, thus allowing only itself access
2. Creates and opens desktops
  - Application desktop (winlogon + user)
  - Winlogon desktop (only winlogon)
  - Screen saver desktop (winlogon + user)
  - SAS switches to winlogon desktop → brings up secure desktop

Ausgewählte Betriebssysteme - Security

15

## Winlogon Initialization (2)

3. Establish LPC connection to Lsass
4. Initialize and register window class structure for logon window (associate process with a window)
5. Register SAS with created window → that window's procedure is called
6. Register window → now winlogon is notified when user logs off or screen saver times out

Ausgewählte Betriebssysteme - Security

16

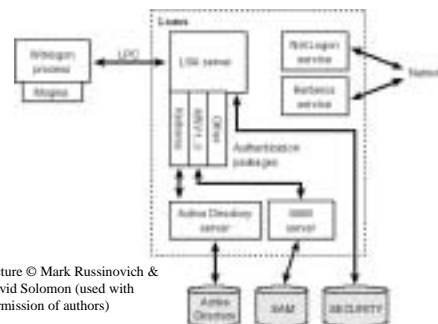
## Logon

- Winlogon process intercepts SAS
  - Can be CTRL+ALT+ENTF
  - Can be insertion of smart card into reader
- Requests GINA to obtain identification
- Calls Lsass to authentication user
- Call network providers (if any) to gather logon information
- Activates logon shell on behalf of user

Ausgewählte Betriebssysteme - Security

17

## Logon



Picture © Mark Russinovich & David Solomon (used with permission of authors)

18

## User Logon Steps

- Winlogon intercepts SAS
  - Creates unique group, which is passed to Lsass during authentication
  - Group attached to desktop
  - After authentication group is attached to logon process token
- Prevents other user logging in on same account to write to first user's desktop

## Authentication

- With username and password each registered authentication package is called
- HKLM\SYSTEM\CurrentControlSet\Control\Lsa
- MSV1\_0 used for local authentication
  - Username + hashed password sent to SAM
  - Returns password, groups, restrictions
- Kerberos used on computer belonging to a domain
  - Version 5, revision 6 for win2K
- Create locally unique identifier (LUID) for logon session and associate LUID with session
  - needed to create access token for user

## After Authentication

- Lsass checks policy database for restrictions
- Lsass adds additional security IDs (everyone, interactive, ...) to access token
- Access token passed to executive to be created
- Executive returns handle, which is handed to winlogon
- Auditing longon
- Check HKLM\Software\Microsoft\WindowsNT\CurrentVersion\Winlogon\Userinit for executables
- Userinit.exe loads profile and creates process for HKLM\Software\Microsoft\WindowsNT\CurrentVersion\Winlogon\Shell

## Outline

- Security Ratings
- Security System Components
- Logon
- Object (File) Access
- Impersonation
- Auditing

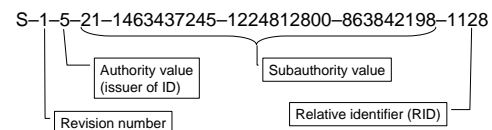
## Access Checks

$f(\text{thread, desired access, object}) \rightarrow \text{yes/no}$

- Thread's security identity is access token of process or impersonation (see below)
- User specifies desired access to object
  - access mask stored in handle
- Object's security settings and thread's security identity locked during check
  - no modification possible

## Security Identifiers

- Instead of using names for subjects, a security identifier (SID) is used



- SID for: users, local and domain groups, computer, domains, domain members

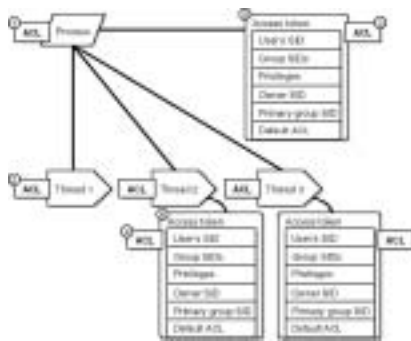
## Security Identifiers (2)

- RID of 1000 and bigger for users and groups  
(see getsid.exe)
- Built-in SIDs:

SID	Group	Use
S-1-1-0	Everyone	All users
S-1-2-0	Local	User who log in locally
S-1-3-0	Creator Owner ID	To be replaced by SID of creator
S-1-3-1	Creator Group ID	To be replaced by primary group SID of creator

## Tokens

- To identify security context of a thread tokens are used
- Initial token created during logon
- All other programs inherit copy of that token
- Can create token with login information
- User SID and Group SID used for authorization
- Also privileges



## SIDs and Access Control

- Object's security information is *security descriptor*
  - Revision number
  - Flags (for instance inheritance)
  - Owner SID
  - Group SID (only used by POSIX)
  - Discretionary access-control list (DACL): who has what access
  - System access-control list (SACL): what should be audited

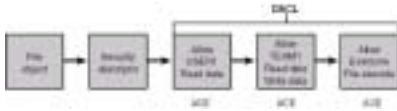
## Access Control List

- Header + list of access control entries (ACE)
- In DACL an ACE contains SID and access mask
- In SACL an ACE contains operations and user performing operation to be audited

## DACL ACEs

- Four types
- Access allowed, access denied:
  - Grant or deny access to user
- Allowed-object, denied-object:
  - Used within Active Directory only
  - Contain GUID specifying object or subobject to which ACE applies
- If no DACL is present, everyone has access
- If empty DACL is present, no user has access

## File Object and its DACL



Picture © Mark Russinovich & David Solomon (used with permission of authors)

## Determine Access

- Two algorithms
  - One to determine maximum access allowed
  - One to determine whether specific access is allowed

## Maximum Allowed Access

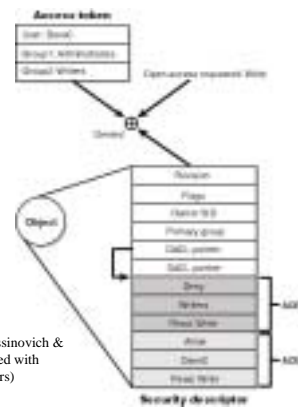
1. If no DACL → all access granted
2. If caller has take-ownership privilege, system grants write-owner before DACL parsing
3. If caller is owner: read-control and write-DACL are granted
4. For each access-denied ACE the ACE's access mask is removed from granted-access mask
5. For each access-granted ACE the ACE's access mask is added to granted-access mask if not removed previously

## Check Access Rights

1. If object has no DACL → access granted
2. If caller has take-ownership → system grants write-owner
3. If owner → read-control and write-DACL granted
4. Each ACE is processed (see next slide)
5. If end of DACL is reached and some of requested rights are not granted → access denied
6. If restricted SIDs → rescan DACL for restrictions

## ACE matching

- a. SID in ACE matches caller's SID
  - b. ACE is access-allow and SID in ACE matches a caller SID which is not deny-only
  - c. If SID in ACE matches restricted SID (6.)
- If access-allowed ACE: ACE rights are granted; if all requested rights are granted → access granted
- If access-denied ACE: if requested access rights match → access denied



Picture © Mark Russinovich & David Solomon (used with permission of authors)

## ACE Ordering

- Deny ACEs must precede allow ACEs
- Otherwise: if all requested rights are satisfied before they can be denied, the request is granted.

## Outline

- Security Ratings
- Security System Components
- Logon
- Object (File) Access
- Impersonation
- Auditing

## Impersonation

- Useful for servers acting on behalf of client
- Restricted to thread, but other threads have access to handles
- TCB contains entry for impersonation token
- After job is done, server reverts to original token
- Cannot execute entire program in context of a client
- Cannot access files or printers on network shares

## Logging Users On

- Alternative to impersonation is logon of client
- *LogonUser* requires username, password, domain or computername, logon type
- Returns access token → used to run program as client
- Or: duplicate access token of client and use as parameter to *CreateProcessAsUser*
- Disadvantages: obtaining logon information from client

## Misuse

- Impersonation only with consent of client
- Client can limit level of impersonation
  - *SecurityAnonymous*: server cannot impersonate nor identify client
  - *SecurityIdentification*: server cannot impersonate but identify the client
  - *SecurityImpersonation*: server can impersonate and identify client
  - *SecurityDelegation*: lets server impersonate client on local and remote systems

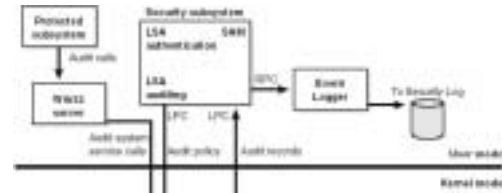
## Outline

- Security Ratings
- Security System Components
- Logon
- Object (File) Access
- Impersonation
- Auditing

## Security Auditing

- Object manager can create audit message as result of access check
- Win32 functions to generate audit messages (SeAuditPrivilege required)
- Kernel mode code can generate audit messages
- SRM sends messages to Lsass which adds additional information and writes to log

## Flow of Auditing Records



Picture © Mark Russinovich & David Solomon (used with permission of authors)