REMUS
REference Monitor for Unix Systems

http://remus.sourceforge.net/

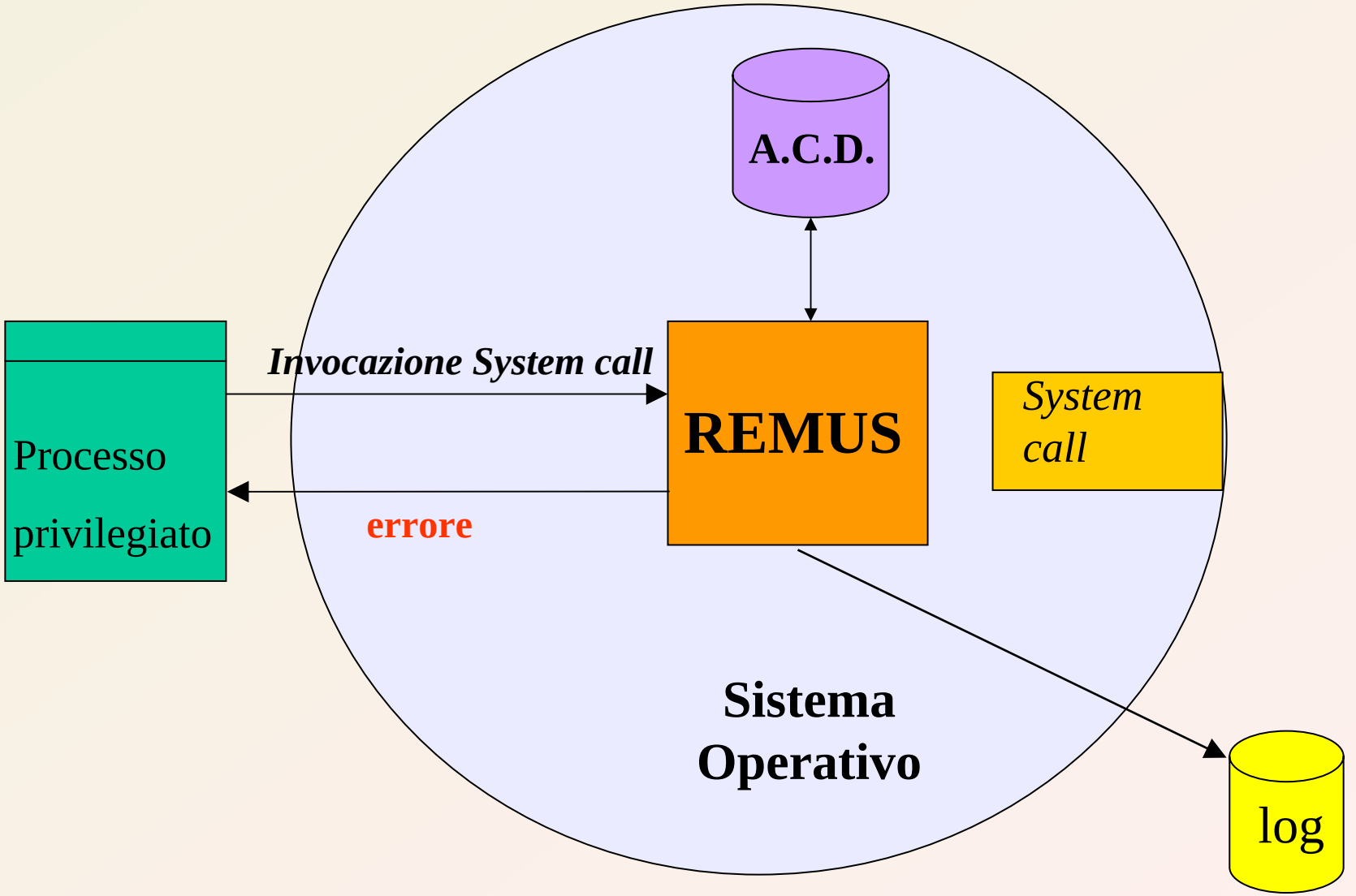
Massimo Bernaschi – IAC-CNR, Roma
Luigi V. Mancini – Dipartimento di Informatica
Università "La Sapienza"di Roma

Collaboratori: Ivano Alonzi, Roberto Battistoni, Emanuele Gabrielli, Giacomo Magnini, Paolo Tassotti

Sistema Integrato di Rilevazione e Gestione delle Intrusioni

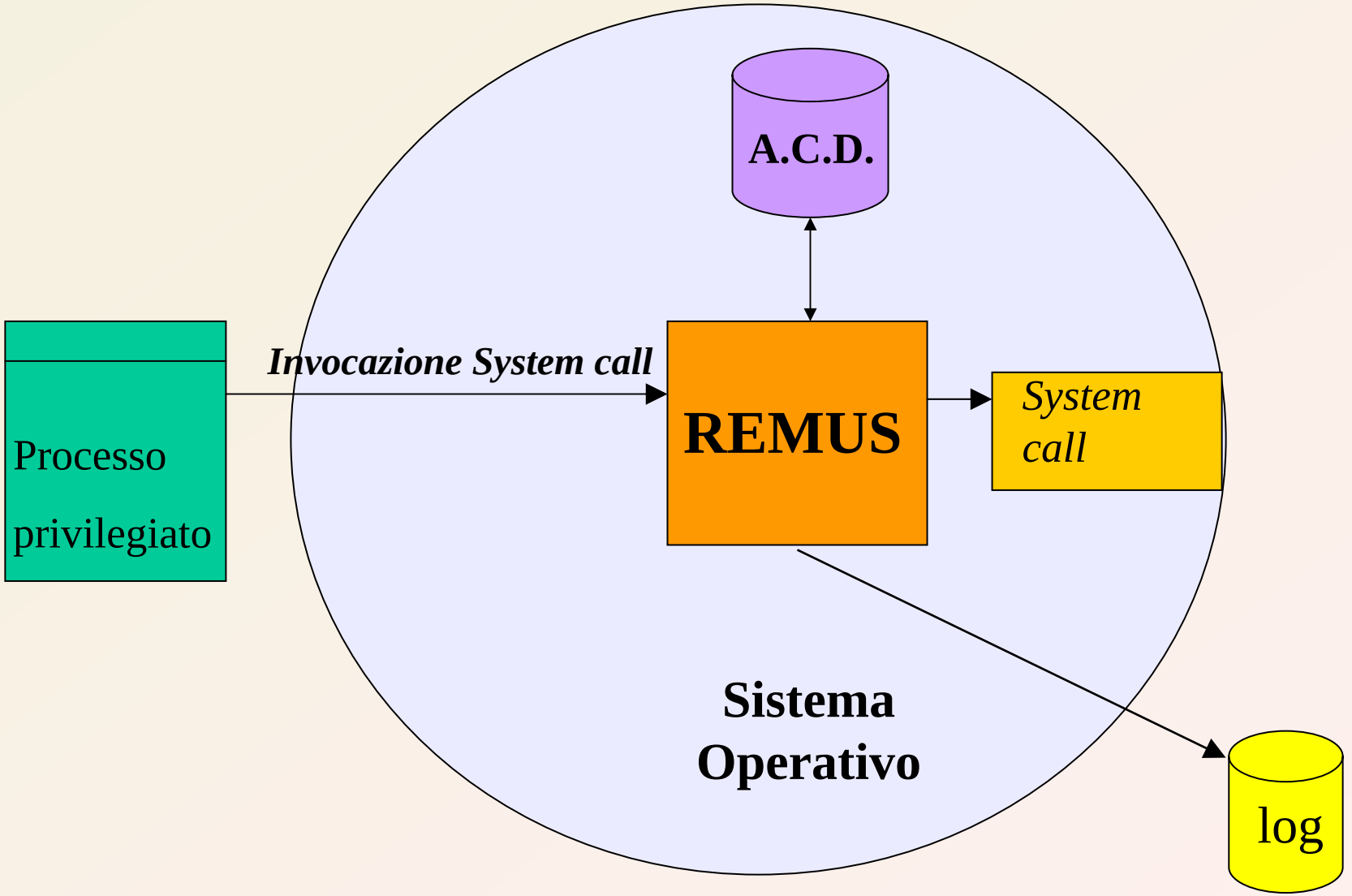
- REMUS (Reference Monitor for Unix Systems)
- WHIPS (Windows Host Intrusion Prevention System)
- SOCKMI (SOCKet MIgration)

Architettura di REMUS: invocazione maliziosa



A.C.D. = Access Control Database

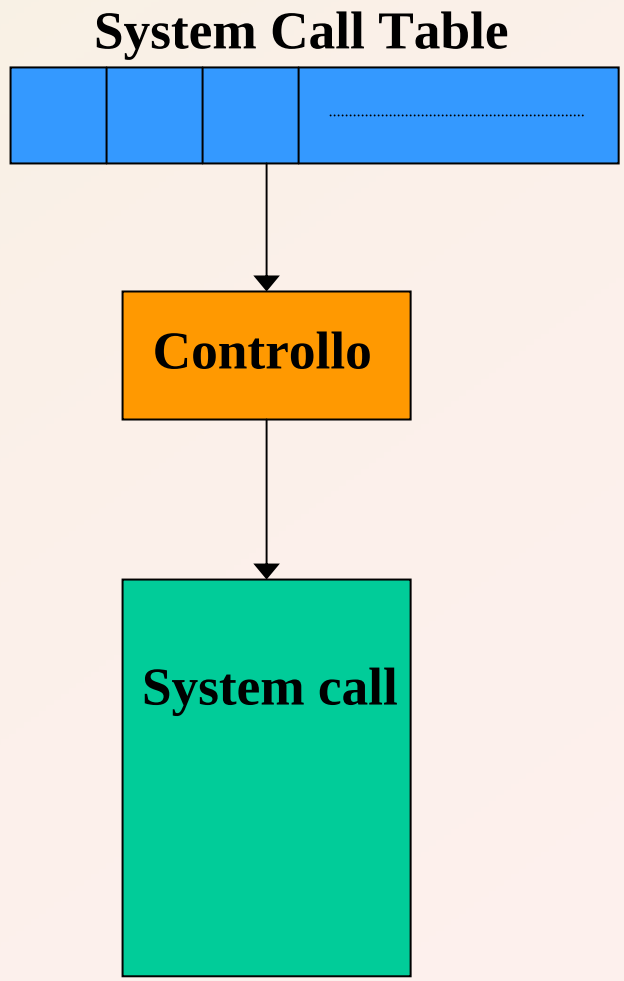
Architettura di REMUS: invocazione consentita



A.C.D. = Access Control Database

Intercettazione delle system call critiche

- Nella System Call Table vengono inserite le chiamate al nuovo codice di controllo.
- La system call originale non viene modificata.
- Il metodo funziona anche per i kernel che non esportano più la System Call Table.



Caratteristiche del progetto REMUS

- Implementato come *Linux kernel module*.
- Controllo delle sole system call “critiche”.
- Firma digitale dei moduli (MD5).
- Configurazione semplice tramite ***sysctl*** e filesystem ***/proc*** .
- Nessuna modifica necessaria al software esistente.
- Basso impatto sulle prestazioni del sistema.

Classificazione system call (1)

gruppo	*funzionalità*	*gruppo*	*funzionalità*
I	File system, device	VI	Comunicazione
II	Gestione processi	VII	Informazioni di sistema
III	Gestione moduli	VIII	Riservate
IV	Gestione memoria	IX	Non implementate
V	Tempo e contatori		

Livello pericolo	*descrizione*
1	Permette di ottenere il pieno controllo del sistema
2	Usata per un attacco Denial of Service
3	Usata per sovvertire il processo invocante
4	Non è pericolosa

Classificazione system call (2)

livello	*gruppo*	*chiamate di sistema*
1	I	open, link, unlink, chmod, lchown, rename, fchown, chown, mount, symlink, fchmod
	II	execve, ptrace, setgid, setreuid, setregid, setgroups, setfsuid, setfsgid, setresuid, setresgid, setuid
	III	init_module

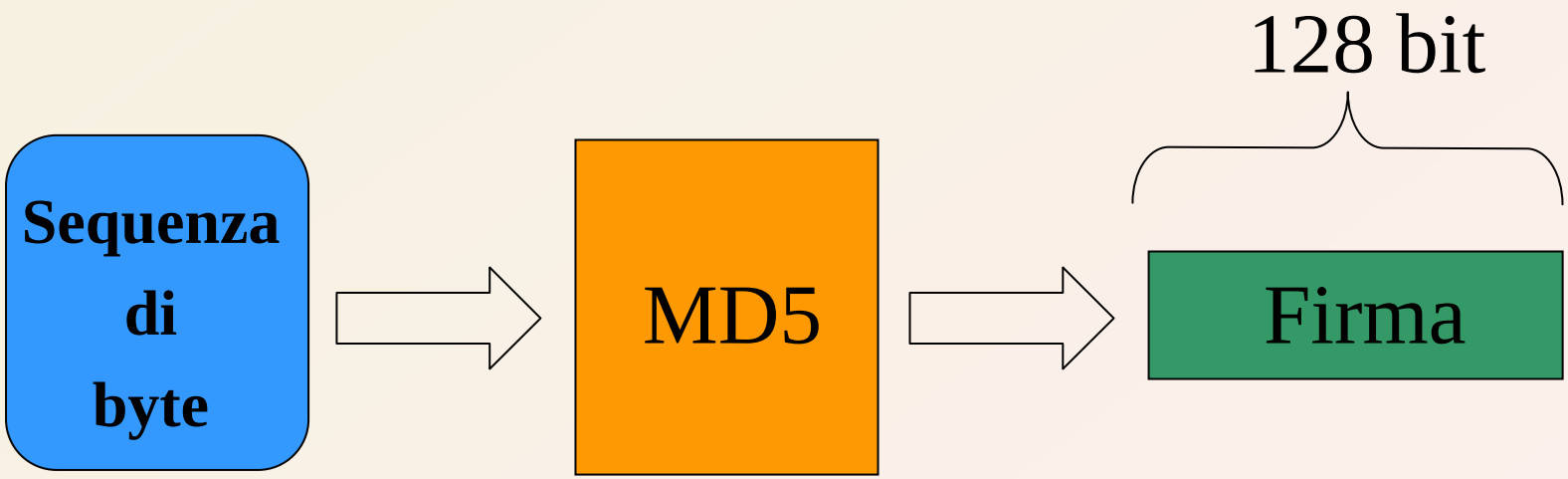
Moduli

- **Modulo:** un modulo è un programma oggetto caricato nel nucleo quando è richiesta la funzionalità corrispondente.

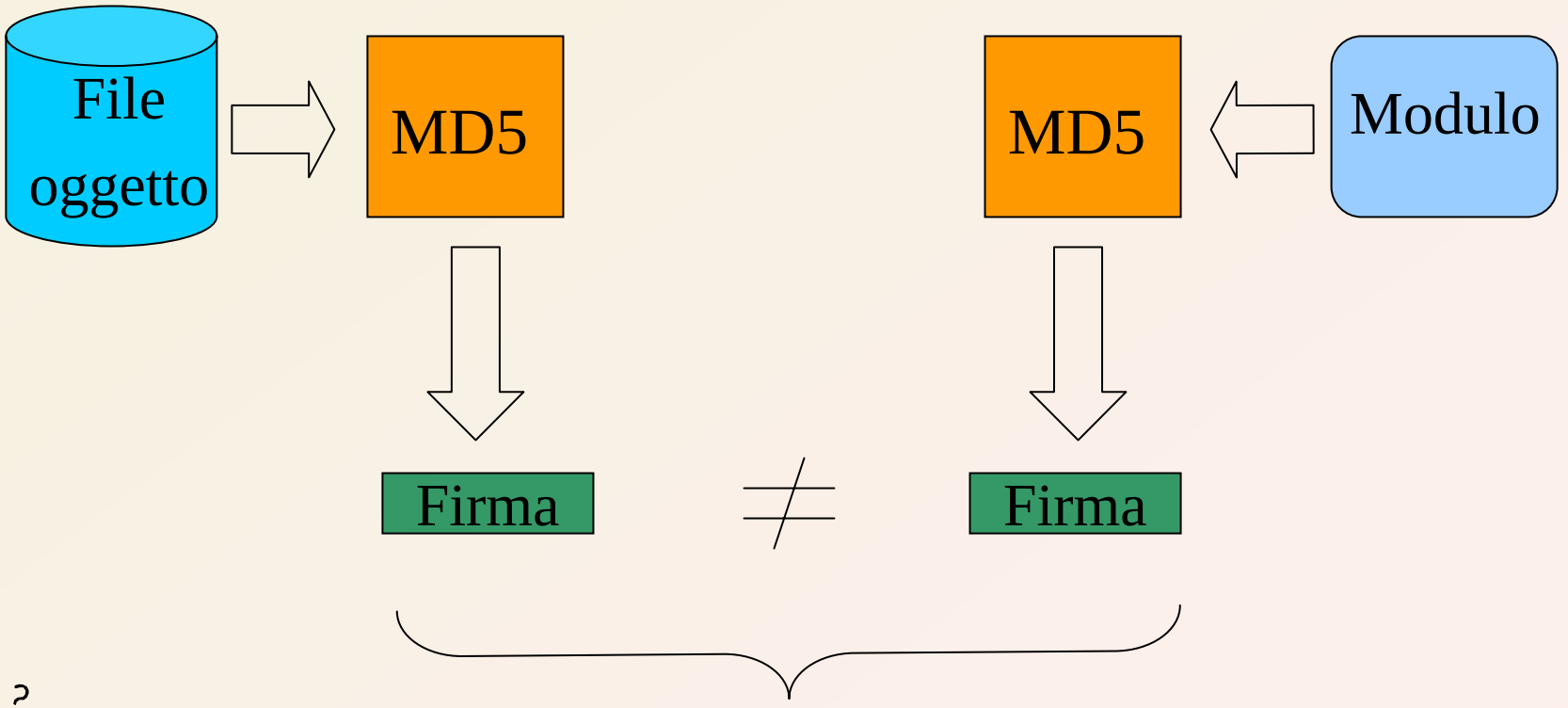
  Esempio: driver di una scheda di rete.

- **Obiettivo:** Garantire il caricamento nel kernel solo di moduli autorizzati dall'amministratore di sistema.
- **Soluzione:** Meccanismo di autenticazione basato sull'algoritmo MD5.

Message Digest 5 (MD5)

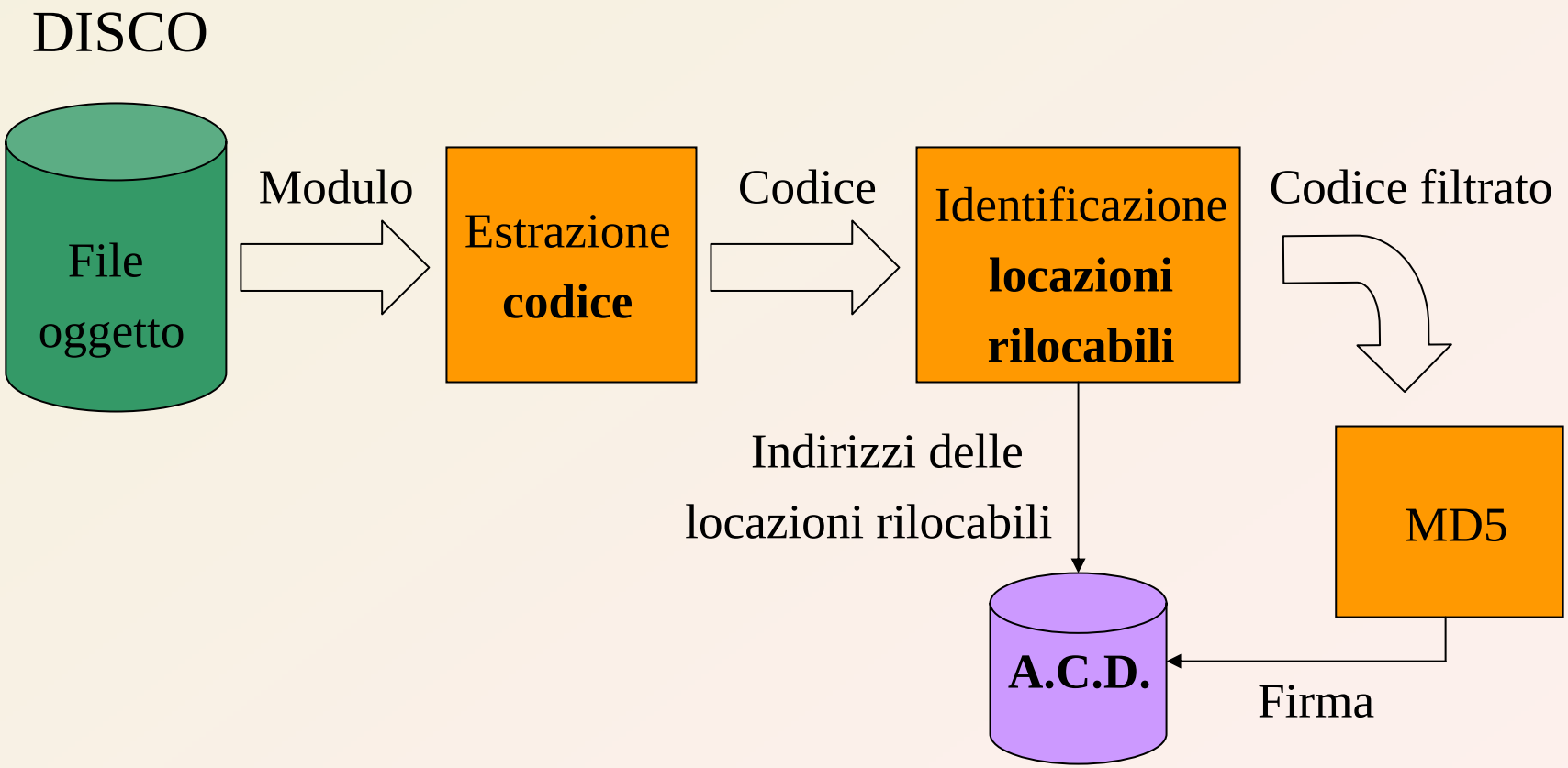


Problema



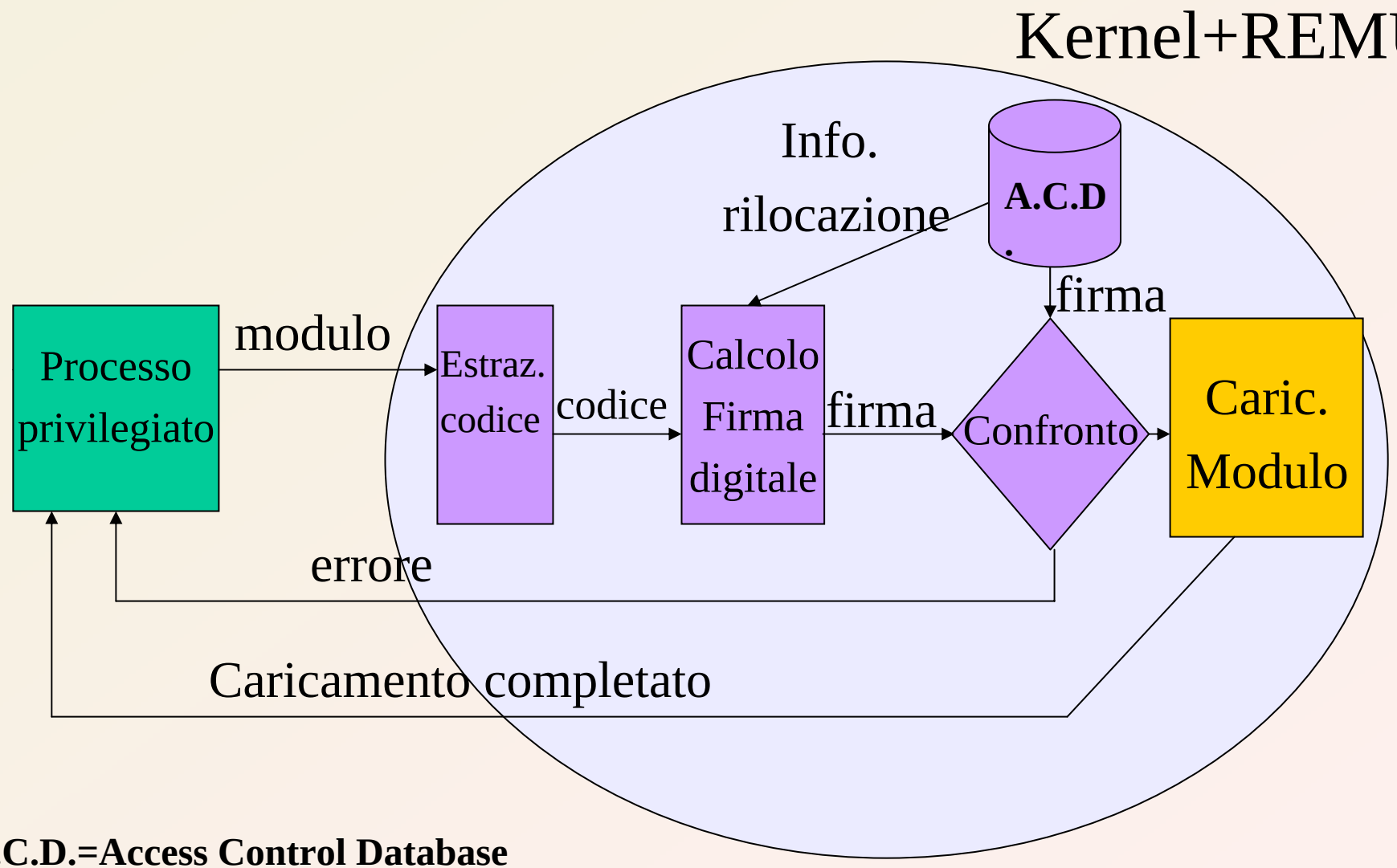
Le firme non coincidono a causa del codice rilocabile

Firma dei moduli legali



A.C.D. = Access Control Database

Controllo in fase di caricamento del modulo



A.C.D.=Access Control Database

Configurazione tramite sysctl

- E’ semplice inserire regole in REMUS:

  sysctl –w remus.open=“ADD /etc/ passwd /etc/bin/passwd”

  Oppure:

  echo -n “ADD /etc/ passwd /etc/bin/passwd” > /proc/sys/remus/open

- I cambiamenti operati nelle regole di REMUS appariranno nel file di log:

  …

  Added open entry (dir:/etc/,file:passwd,prog:/usr/bin/passwd) successfully

  …

Visualizzare lo stato di REMUS

Per avere una vista d'insieme dello stato di REMUS, basta eseguire *cat /proc/sys/remus/acl* :

```
REMUS Version 0.5 : ACD Overview
writerpid: 0
setuid encrypted root passwd: $1$UF¦–IÒ“W$UIWVQaMb433qRC1UPs6/31
syscall exec          11          ctlmask=17   check,noblock,debug,notime
syscall open          5           ctlmask=17   check,noblock,debug,notime
….
```

Per controllare l'ACL di una singola system call (es. chmod):

```
REMUS Version 0.5 : chmod System Call Database

total directories are 6, ctlmask=17 : (check,noblock,debug,notime)
directory=(146667,834) special=0
                        prog=python
directory=(146693,834) special=1
              FILE (146688,834)
                        prog=chmod
….
```

Tool XML

E’ possibile configurare REMUS tramite un tool che utilizza file XML per la gestione delle regole:

```
<remus>
  <webserver type="default"></webserver>

  <mailserver type="(Sendmail)">
    <rule>remus.open='ADD /var/spool/mqueue/ sendmail'</rule>
    <rule>remus.rename='ADD /var/spool/mqueue/sendmail'</rule>
    <rule>remus.link='ADD /var/spool/ sendmail'</rule>
    <rule>remus.unlink='ADD /var/spool/ sendmail'</rule>
  </mailserver>

  <ftpserver type="default"></ftpserver>
</remus>
```

Prestazioni di REMUS

- Solo i processi privilegiati necessitano di controllo, nessun rallentamento per i processi non privilegiati.
- Nessun rallentamento per i processi privilegiati in esecuzione in spazio utente.
- Solo le primitive critiche necessitano di controllo.
- Ad eccezione della **open**, è raro che un processo privilegiato invochi più di una volta, durante la sua esecuzione, una primitiva controllata.

Vantaggi di REMUS

- Soluzione a livello di kernel, quindi robusta
- Rende meno critici i tempi di rilascio ed istallazione di aggiornamenti per sanare la vulnerabilità di una applicazione
- Protegge l’integrità del sistema operativo impedendo il caricamento di moduli non autorizzati;

WHIPS

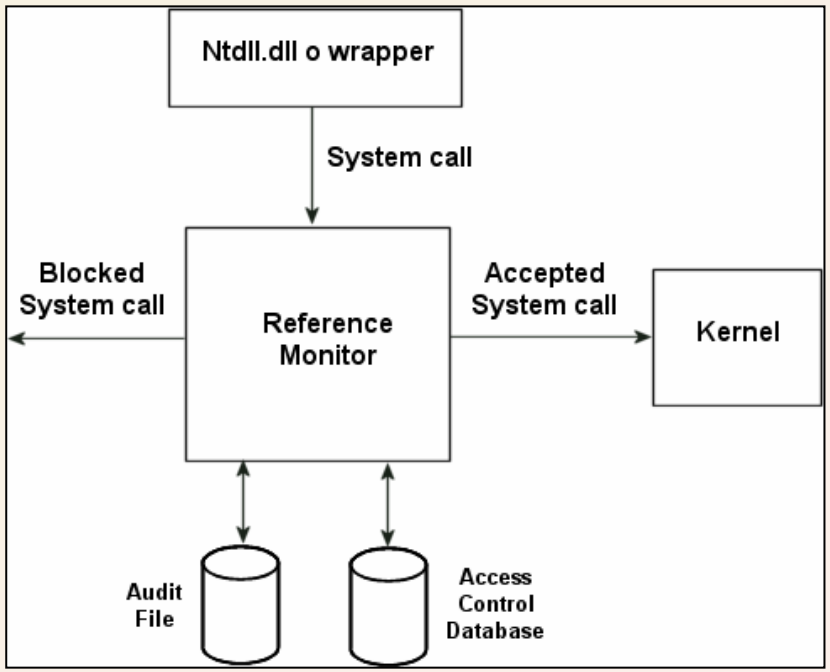
Windows_NT family Host Intrusion Prevention System

Un sistema per la rilevazione e prevenzione delle intrusioni in sistemi Windows 2000, XP, 2003

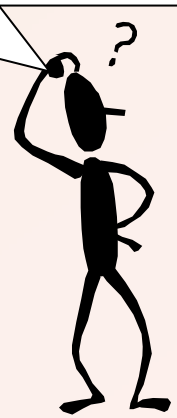
http://cesare.dsi.uniroma1.it/Sicurezza/doc/WHIPSarticle.pdf

WHIPS

- **WHIPS** (Windows_nt family Host based IPS) è un *Host Intrusion Prevention System* per SO della famiglia Windows NT (Windows NT,2000,XP,2003). Utilizza il paradigma del *Reference Monitor* (RM) per le chiamate di sistema critiche per la sicurezza.

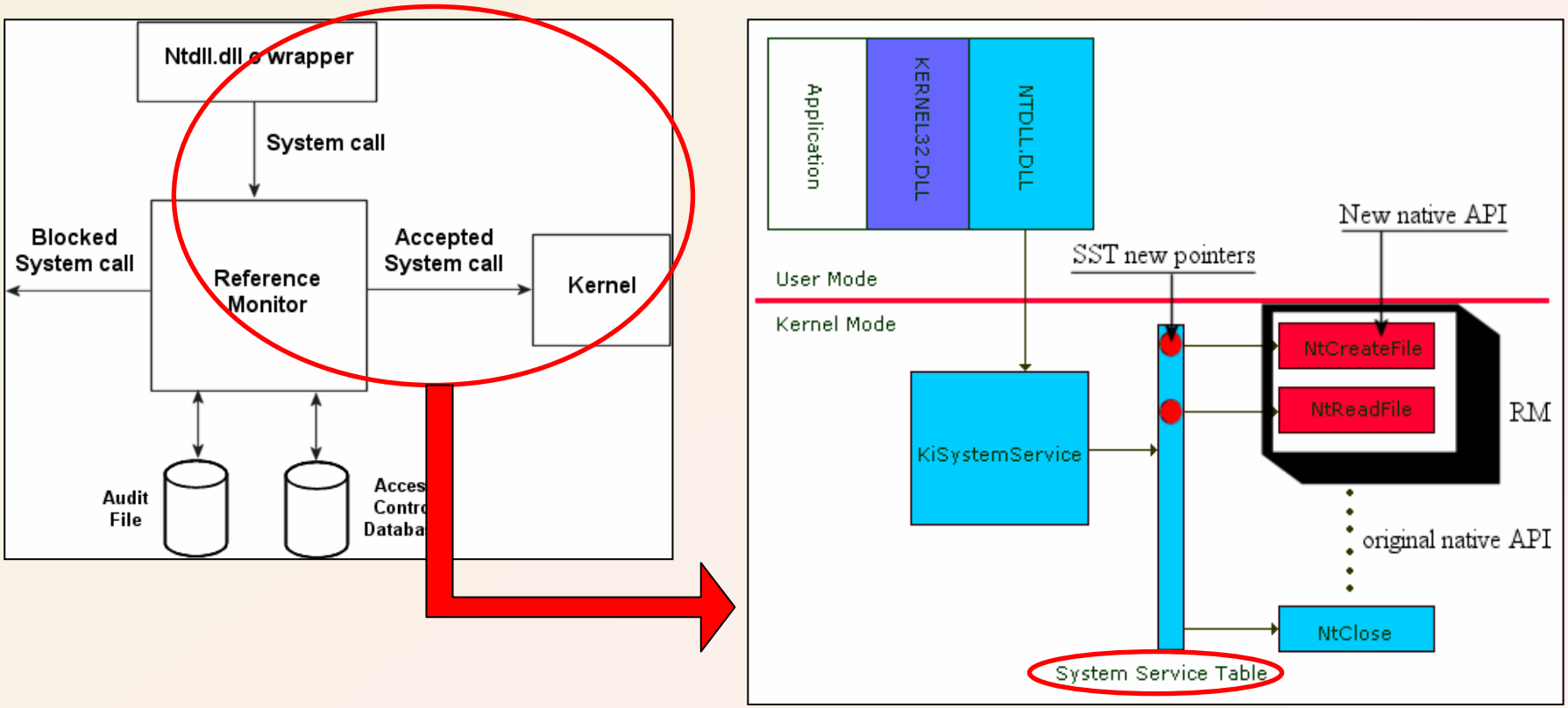


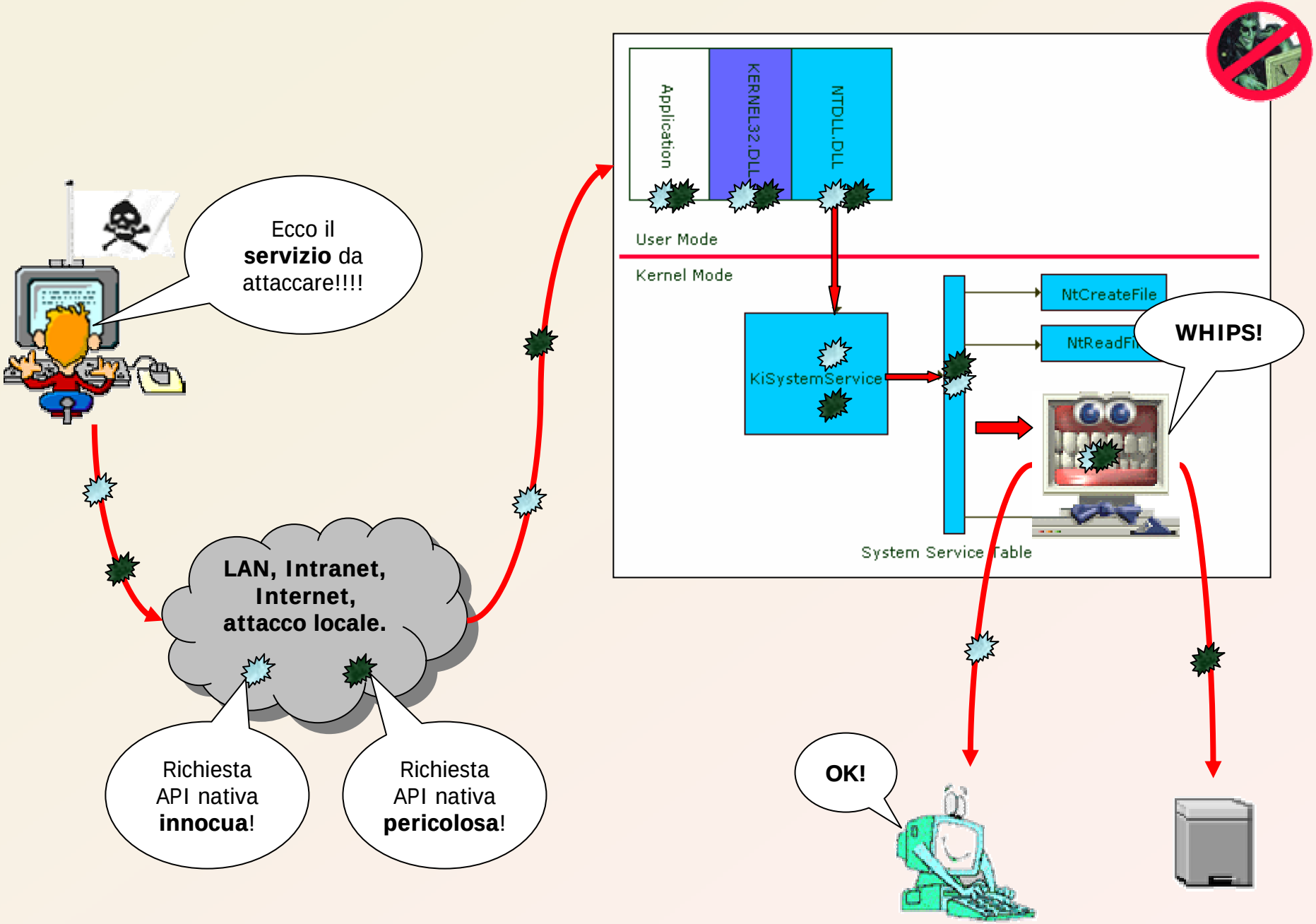
Il codice sorgente di Windows non è disponibile e la documentazione del kernel spesso è limitata e non approfondita. Non c'è il supporto di una comunità come quella di Linux e tutti i SO open source.



Modulo WHIPS

- WHIPS è realizzato attraverso un modulo del kernel (driver) in Windows 2000 e XP. WHIPS modifica la *System Service Table (SST)* nel kernel sostituendo le API native originali con nuove API native che effettuano i controlli del RM. Il controllo è basato su un set di regole (ACD, Access Control Database) che definiscono il comportamento consentito dal sistema.



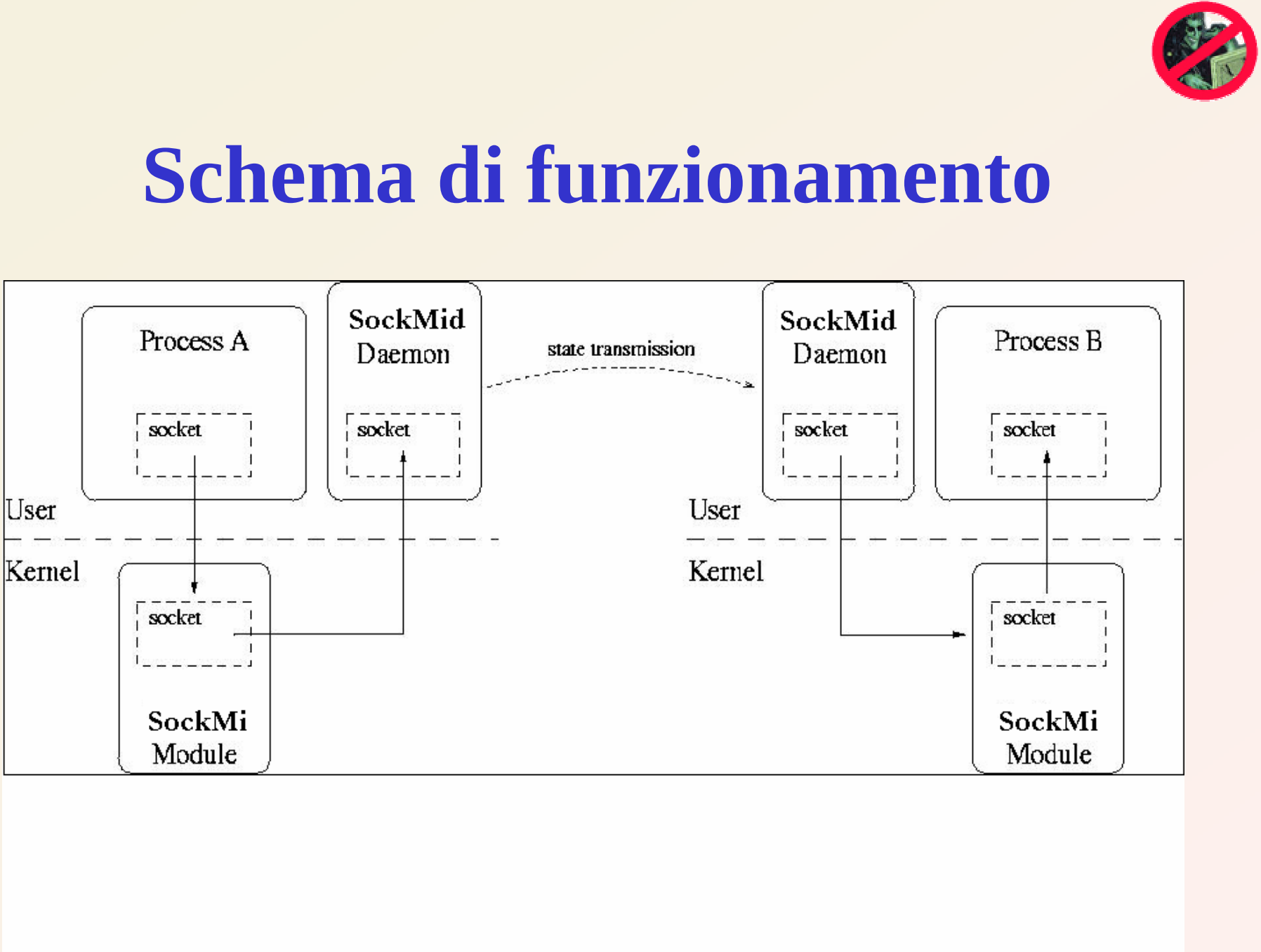

Ecco il **servizio** da attaccare!!!!
LAN, Intranet, Internet, attacco locale.
Richiesta API nativa **innocua**!
Richiesta API nativa **pericolosa**!
Application
KERNEL32.DLL
NTDLL.DLL
User Mode
Kernel Mode
KiSystemService
NtCreateFile
NtReadFi
System Service Table
WHIPS!
OK!

SOCKMI

Un meccanismo per la migrazione di connessioni TCP/IP in ambiente Linux

Caratteristiche

- **Simmetria:** la migrazione è possibile sia *server-side* che *client-side;*
- **Trasparenza:** l'altro *peer* è inconsapevole;
- **Socket-based:** è possibile migrare un socket anche in stato *bounded* o *listening;*
- **Nessun Protocollo "ad hoc":** non si richiede che i due *peer* utilizzino protocolli al di sopra del TCP;
- **Indipendente dal Livello Applicazione:** il meccanismo si occupa unicamente della migrazione dei livelli *TCP/IP;*



Schema di funzionamento

Le API di SockMi

- Export basato sulla *sysctl()*:
  - export *passivo*: effettuato da un processo esterno (tramite system call o da linea di comando).
  - export *attivo*: effettuato dal processo stesso (solo system call);
- Import basato sulla *poll()*:
  - mediante *nuovi* eventi poll si specifica lo *stato* del socket che si vuole importare (*bounded*, *listening*, *connected*);
  - È possibile specificare anche altri criteri di importazione:
    - *connect()*: indirizzo/porta destinazione;
    - *bind()*: indirizzo/porta sorgente.

Conclusioni

Il prototipo sviluppato:

- rileva e blocca i tentativi di intrusione;
- è semplice da utilizzare e da gestire;
- ha impatto minimo sulle prestazioni del sistema controllato;
- garantisce la portabilità delle applicazioni già esistenti;
- può essere utilizzato senza modifiche su diverse distribuzioni;
- base per sistema integrato di rilevazione e gestione delle intrusioni

Riferimenti

- M. Bernaschi, E. Gabrielli and L.V. Mancini "REMUS: a security-enhanced operating system", ACM Transactions on Information and System Security, Vol. 5, No. 1, pp. 36-61, Feb. 2002.
- R. Battistoni, E. Gabrielli, L.V. Mancini "An extended access control system for Windows XP", Technical Report, Nov. 2003

http://remus.sourceforge.net/