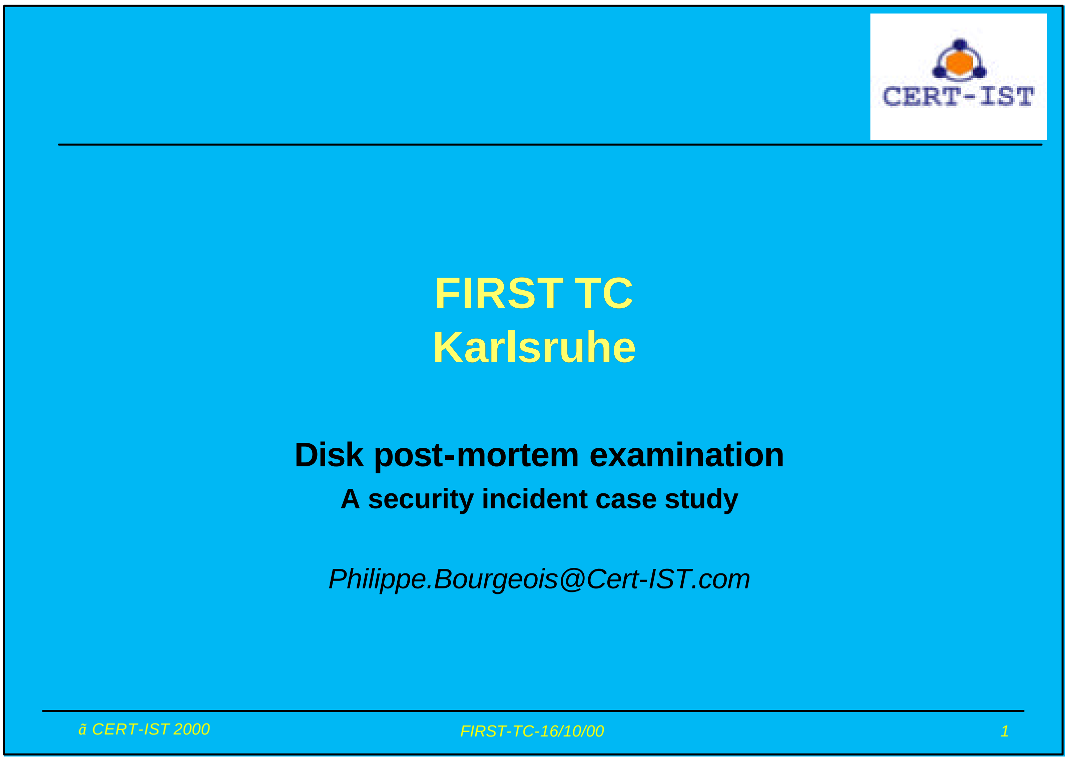
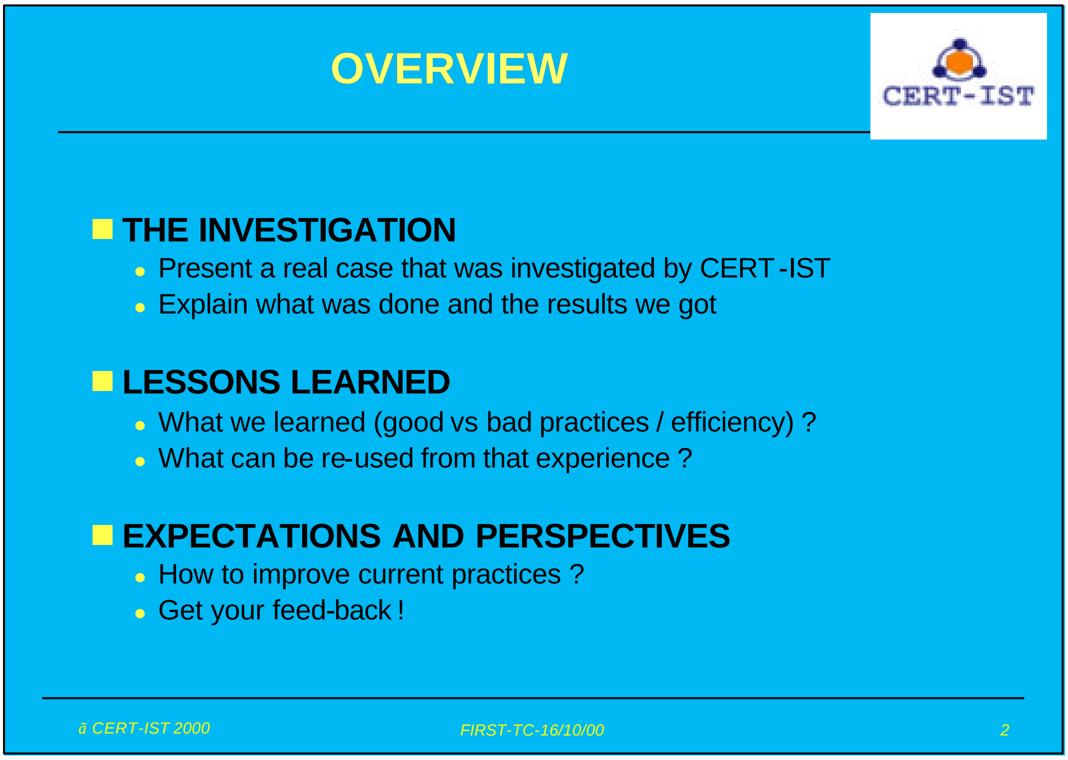


FIRST TC
Karlsruhe

Disk post-mortem examination

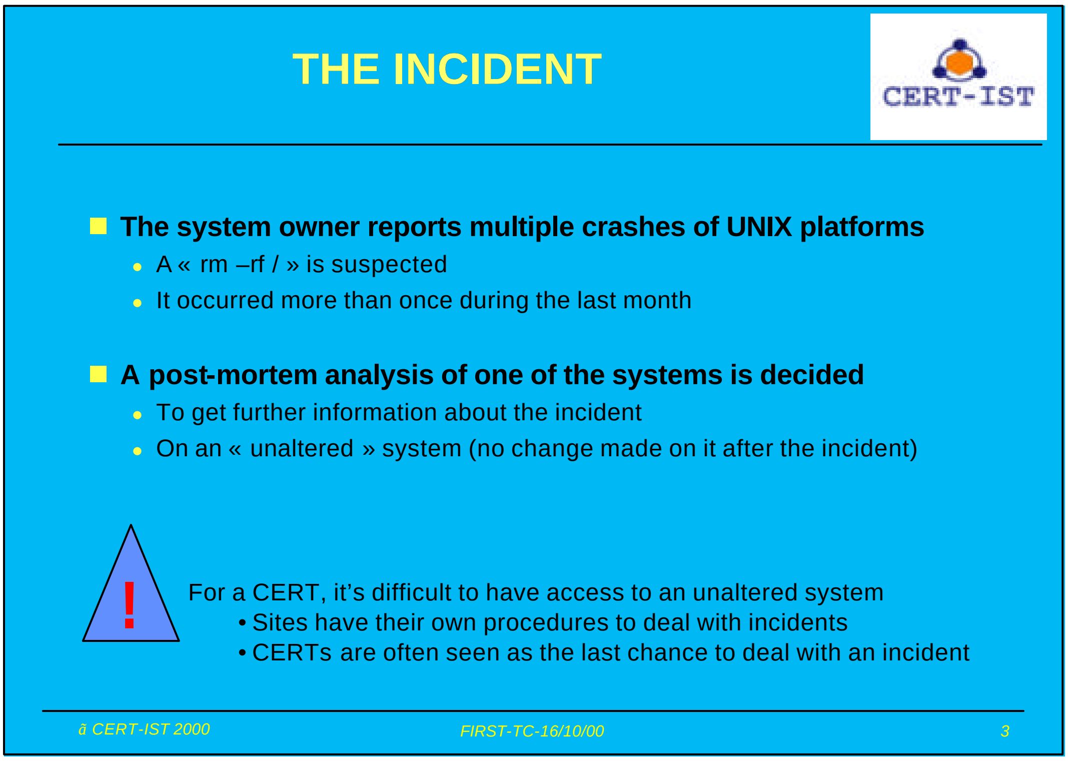
A security incident case study

Philippe.Bourgeois@Cert-IST.com



OVERVIEW

- **THE INVESTIGATION**
  - Present a real case that was investigated by CERT -IST
  - Explain what was done and the results we got
- **LESSONS LEARNED**
  - What we learned (good vs bad practices / efficiency) ?
  - What can be re-used from that experience ?
- **EXPECTATIONS AND PERSPECTIVES**
  - How to improve current practices ?
  - Get your feed-back !



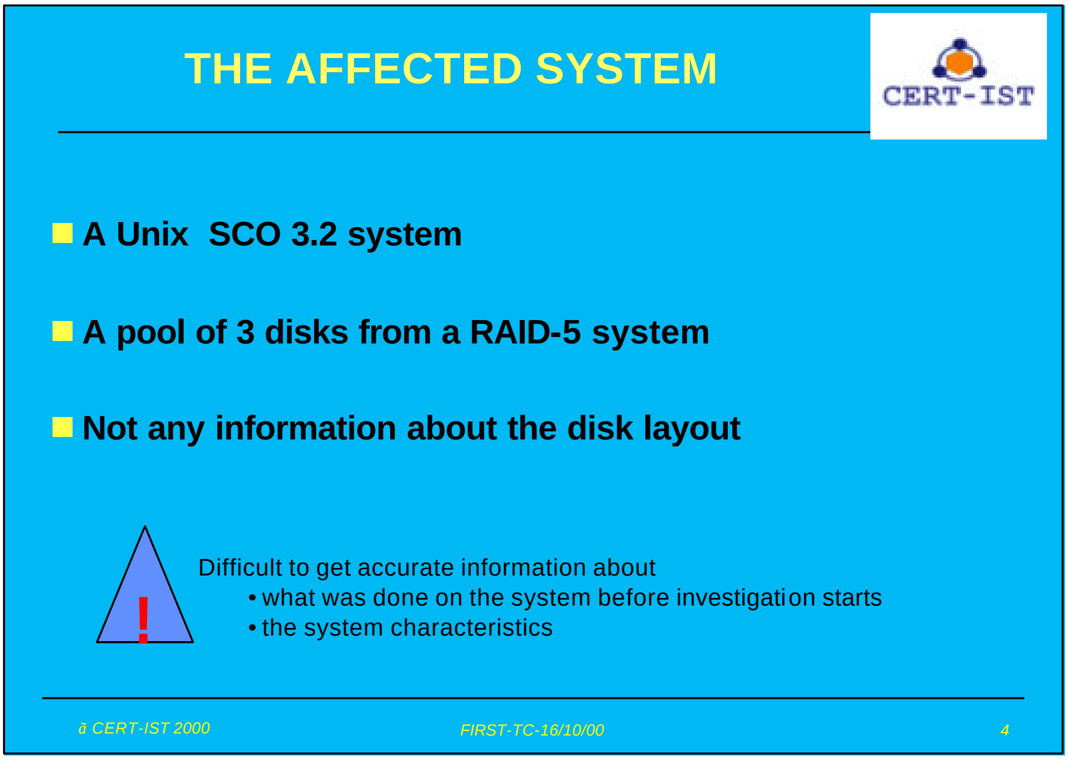
THE INCIDENT

CERT-IST

- **The system owner reports multiple crashes of UNIX platforms**
  - A « rm –rf / » is suspected
  - It occurred more than once during the last month
- **A post-mortem analysis of one of the systems is decided**
  - To get further information about the incident
  - On an « unaltered » system (no change made on it after the incident)

!

For a CERT, it’s difficult to have access to an unaltered system
- Sites have their own procedures to deal with incidents
- CERTs are often seen as the last chance to deal with an incident



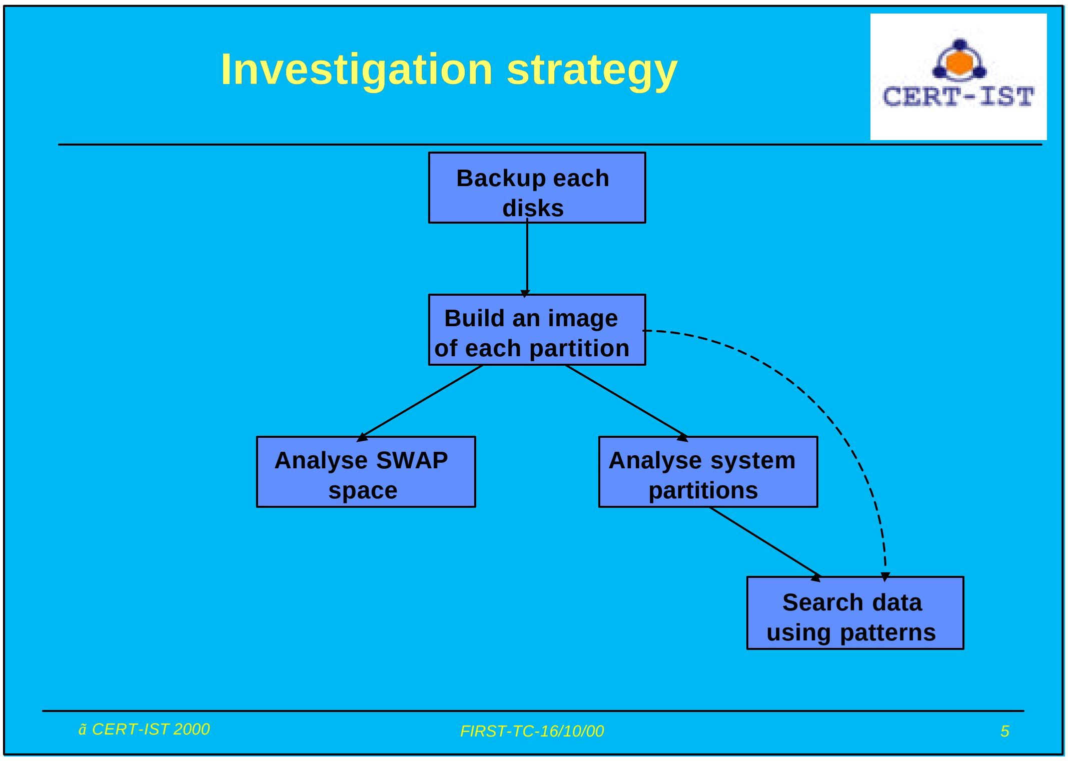
THE AFFECTED SYSTEM

CERT-IST

- **A Unix SCO 3.2 system**
- **A pool of 3 disks from a RAID-5 system**
- **Not any information about the disk layout**

!

Difficult to get accurate information about
- what was done on the system before investigation starts
- the system characteristics



Investigation strategy

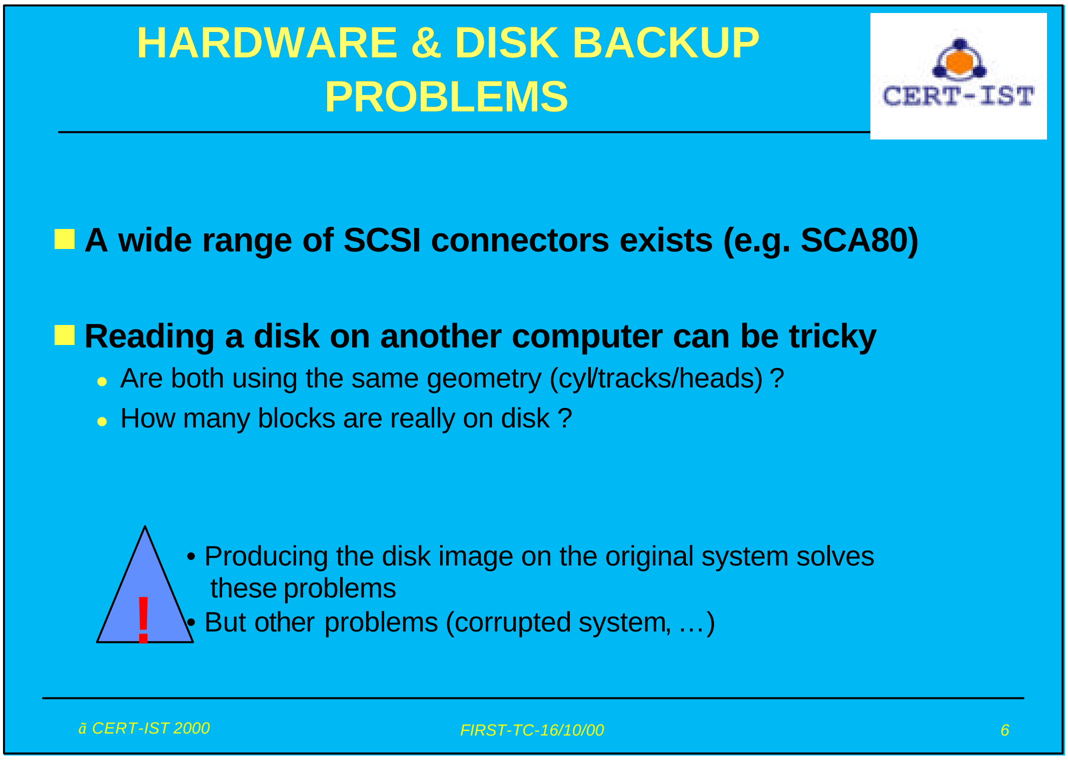
Backup each disks

Build an image of each partition

Analyse SWAP space

Analyse system partitions

Search data using patterns

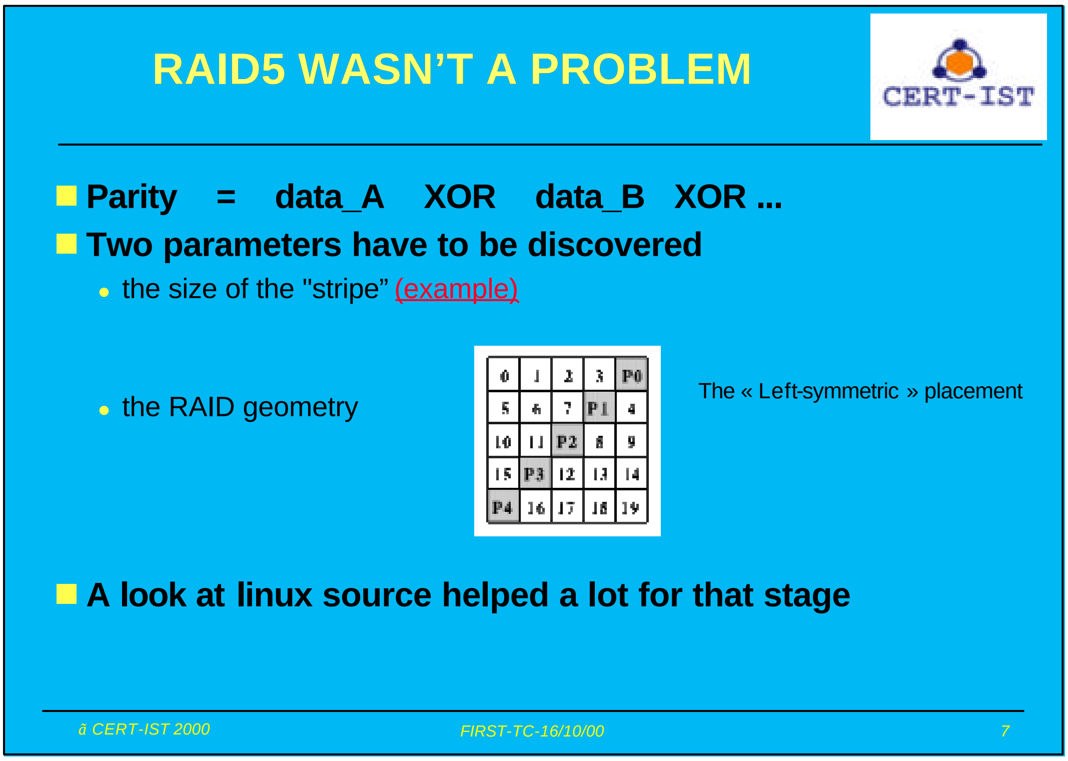


HARDWARE & DISK BACKUP PROBLEMS

- A wide range of SCSI connectors exists (e.g. SCA80)
- Reading a disk on another computer can be tricky
  - Are both using the same geometry (cyl/tracks/heads) ?
  - How many blocks are really on disk ?

!

- Producing the disk image on the original system solves these problems
- But other problems (corrupted system, …)



RAID5 WASN'T A PROBLEM

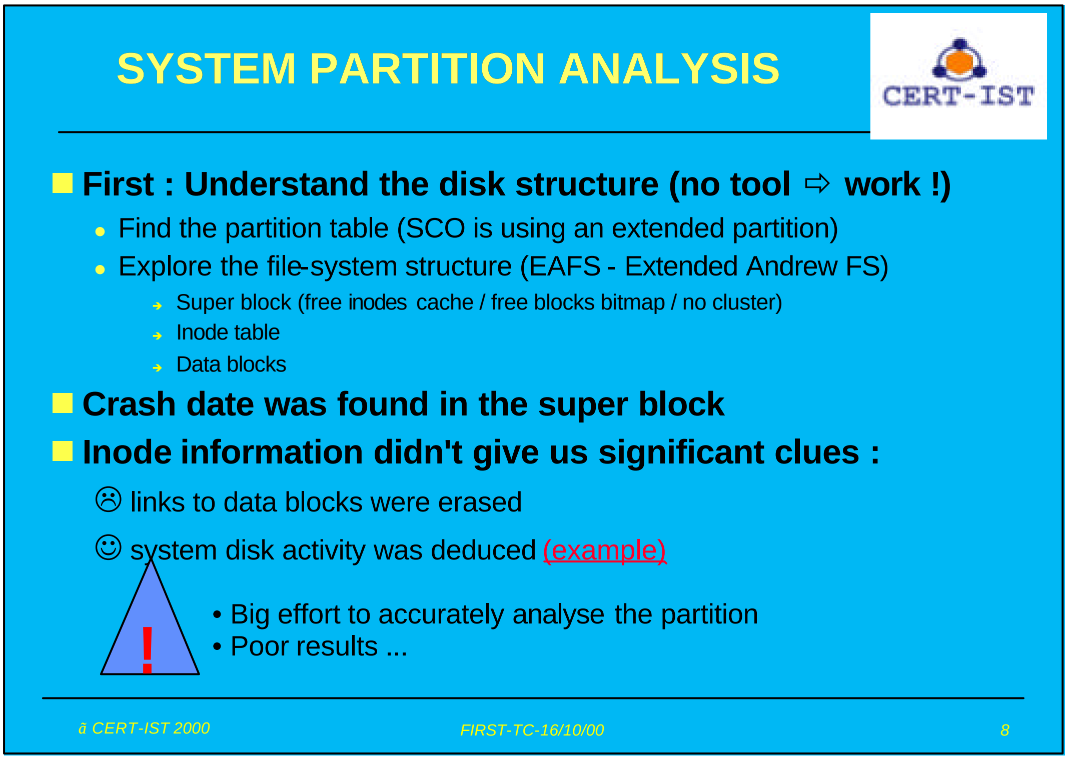
CERT-IST

- Parity = data_A XOR data_B XOR ...
- Two parameters have to be discovered
  - the size of the "stripe" (example)
  - the RAID geometry

0	1	2	3	P0
5	6	7	P1	4
10	11	P2	8	9
15	P3	12	13	14
P4	16	17	18	19

The « Left-symmetric » placement

- A look at linux source helped a lot for that stage



SYSTEM PARTITION ANALYSIS

CERT-IST

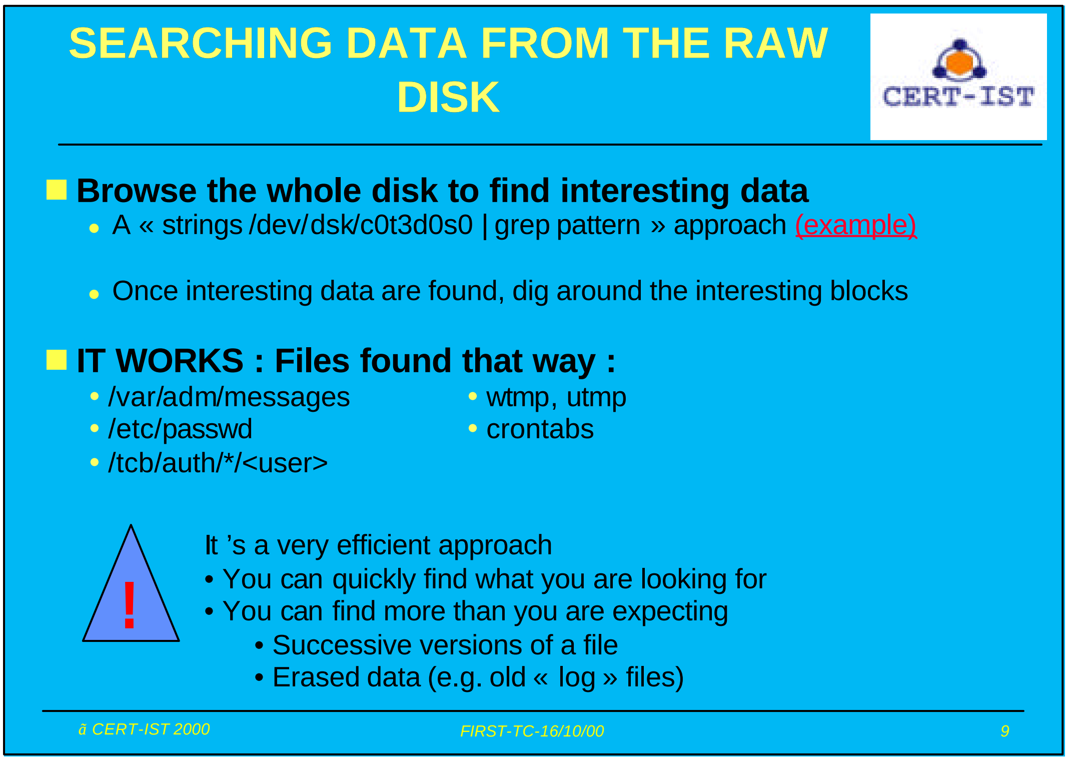
- First : Understand the disk structure (no tool ⇨ work !)
  - Find the partition table (SCO is using an extended partition)
  - Explore the file-system structure (EAFS - Extended Andrew FS)
    - Super block (free inodes cache / free blocks bitmap / no cluster)
    - Inode table
    - Data blocks
- Crash date was found in the super block
- Inode information didn't give us significant clues :

☹ links to data blocks were erased

☺ system disk activity was deduced (example)

!

• Big effort to accurately analyse the partition
• Poor results ...



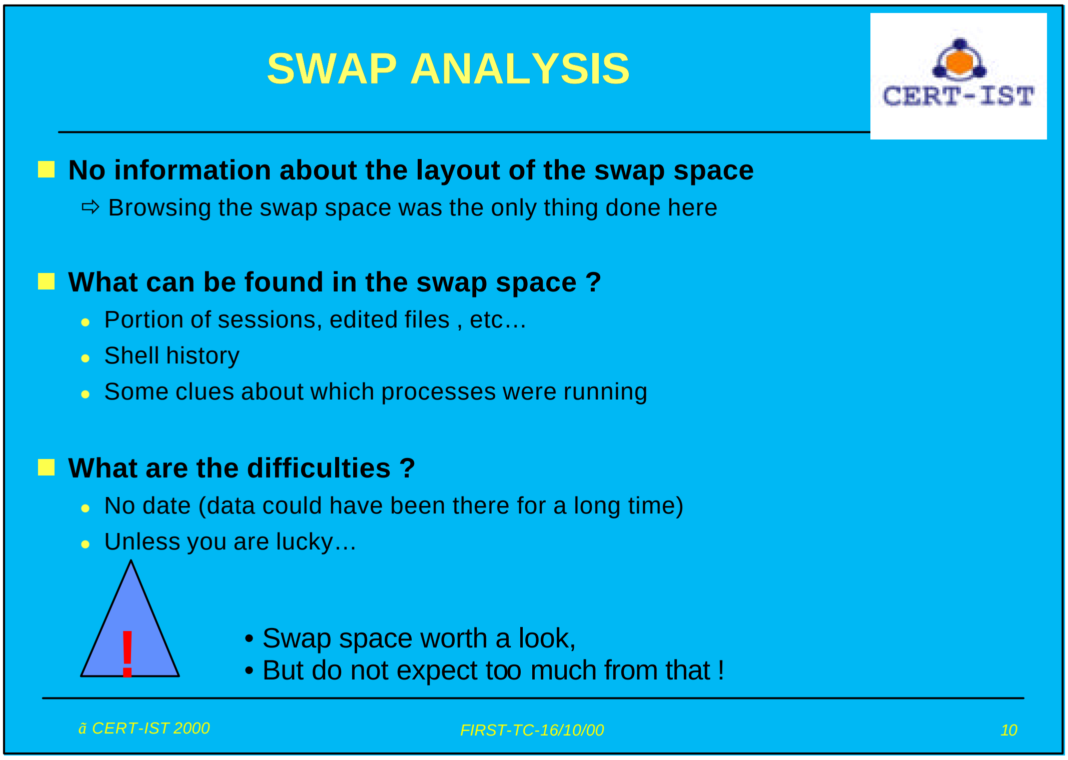
SEARCHING DATA FROM THE RAW DISK

- **Browse the whole disk to find interesting data**
  - A « strings /dev/dsk/c0t3d0s0 | grep pattern » approach (example)
  - Once interesting data are found, dig around the interesting blocks
- **IT WORKS : Files found that way :**
  - /var/adm/messages
  - /etc/passwd
  - /tcb/auth/*/<user>
  - wtmp, utmp
  - crontabs

!

It 's a very efficient approach

- You can quickly find what you are looking for
- You can find more than you are expecting
  - Successive versions of a file
  - Erased data (e.g. old « log » files)

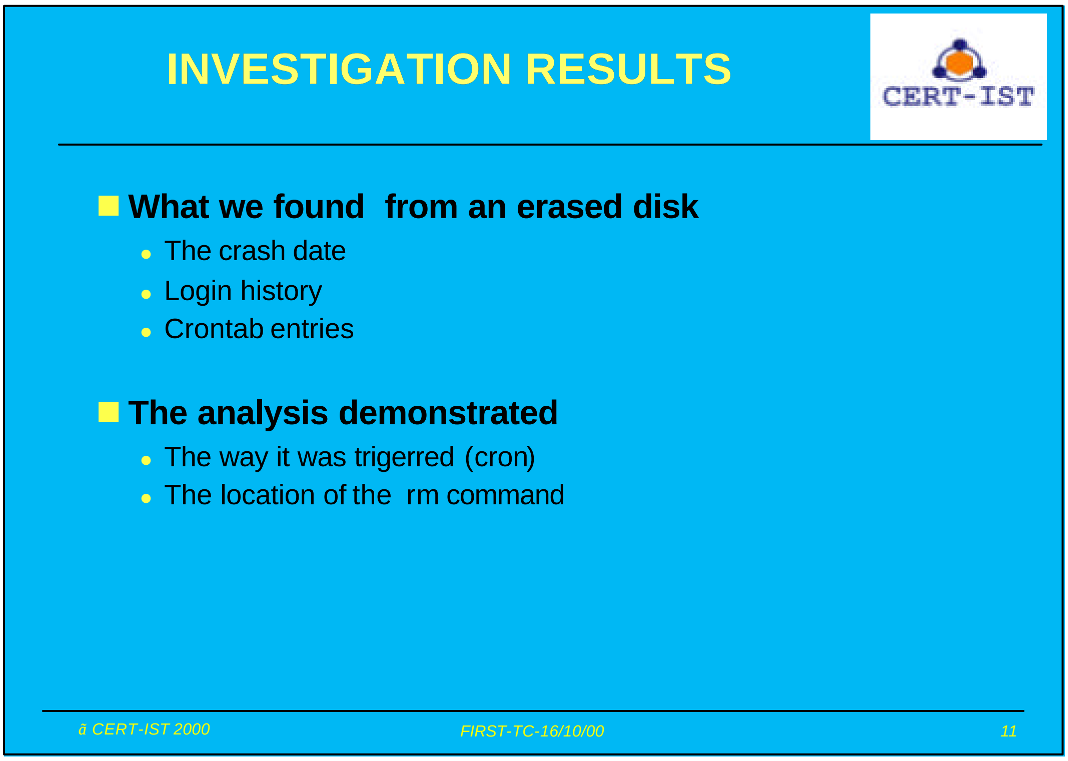


SWAP ANALYSIS

- **No information about the layout of the swap space**
  - ⇨ Browsing the swap space was the only thing done here
- **What can be found in the swap space ?**
  - Portion of sessions, edited files , etc…
  - Shell history
  - Some clues about which processes were running
- **What are the difficulties ?**
  - No date (data could have been there for a long time)
  - Unless you are lucky…

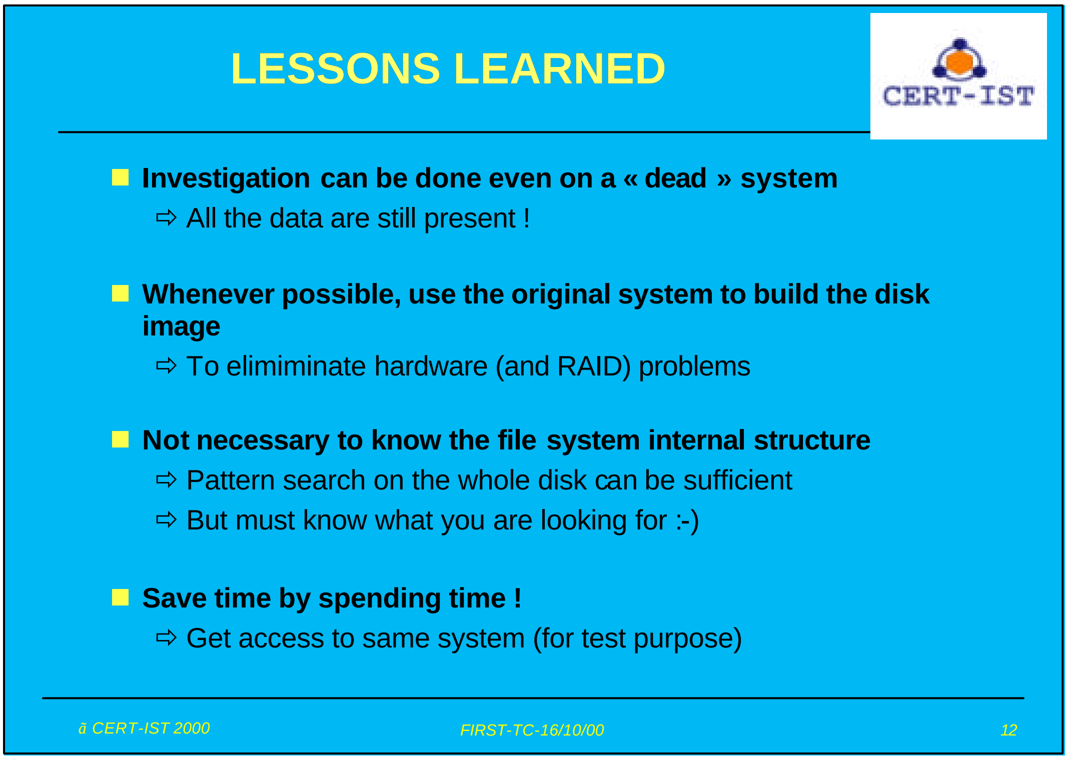
!

- Swap space worth a look,
- But do not expect too much from that !



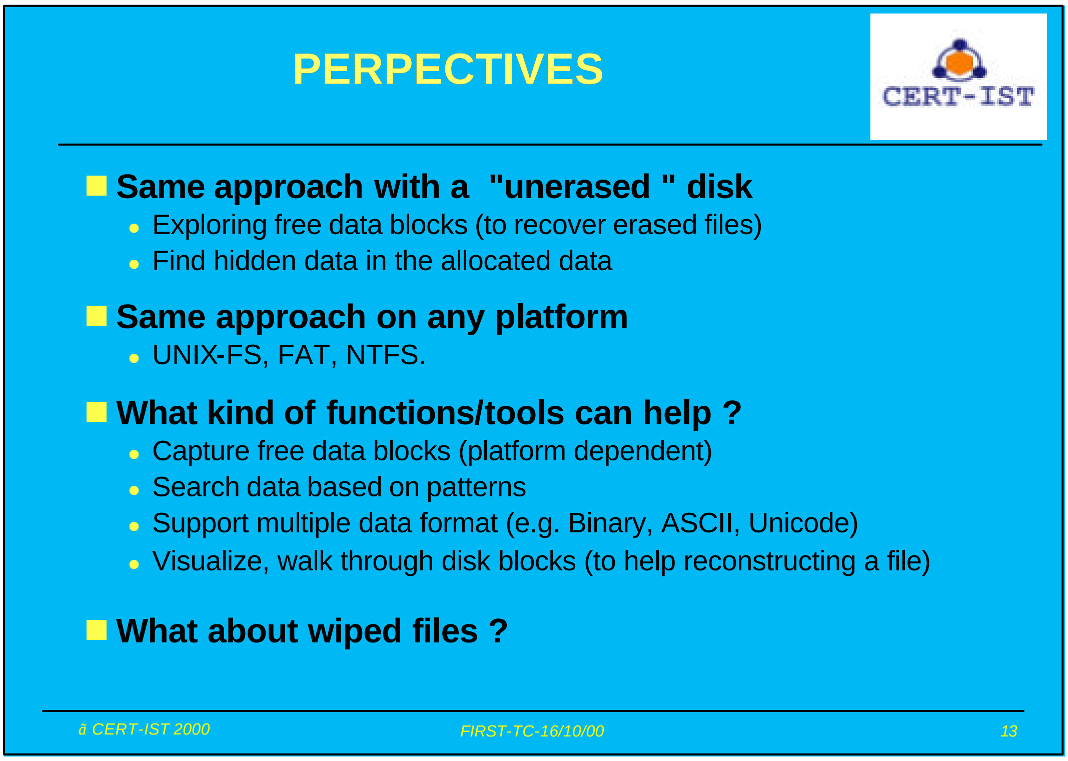
INVESTIGATION RESULTS

- What we found from an erased disk
  - The crash date
  - Login history
  - Crontab entries
- The analysis demonstrated
  - The way it was trigerred (cron)
  - The location of the rm command



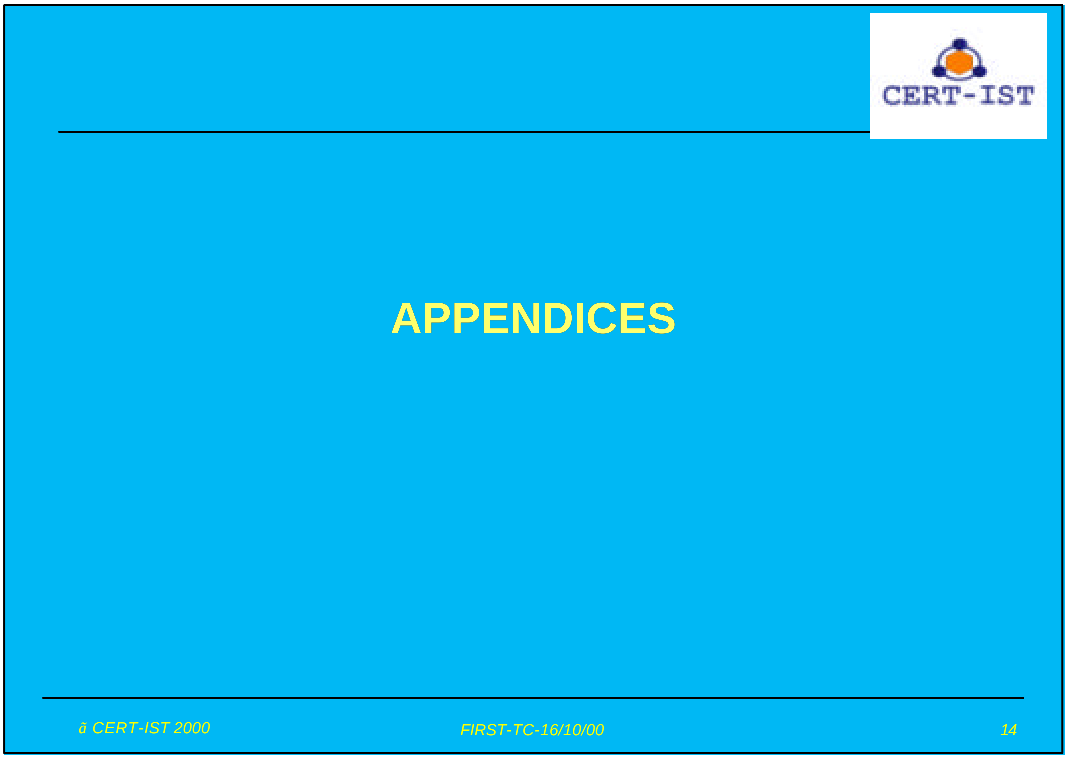
LESSONS LEARNED

- **Investigation can be done even on a « dead » system**
  - ⇨ All the data are still present !
- **Whenever possible, use the original system to build the disk image**
  - ⇨ To elimiminate hardware (and RAID) problems
- **Not necessary to know the file system internal structure**
  - ⇨ Pattern search on the whole disk can be sufficient
  - ⇨ But must know what you are looking for :-)
- **Save time by spending time !**
  - ⇨ Get access to same system (for test purpose)

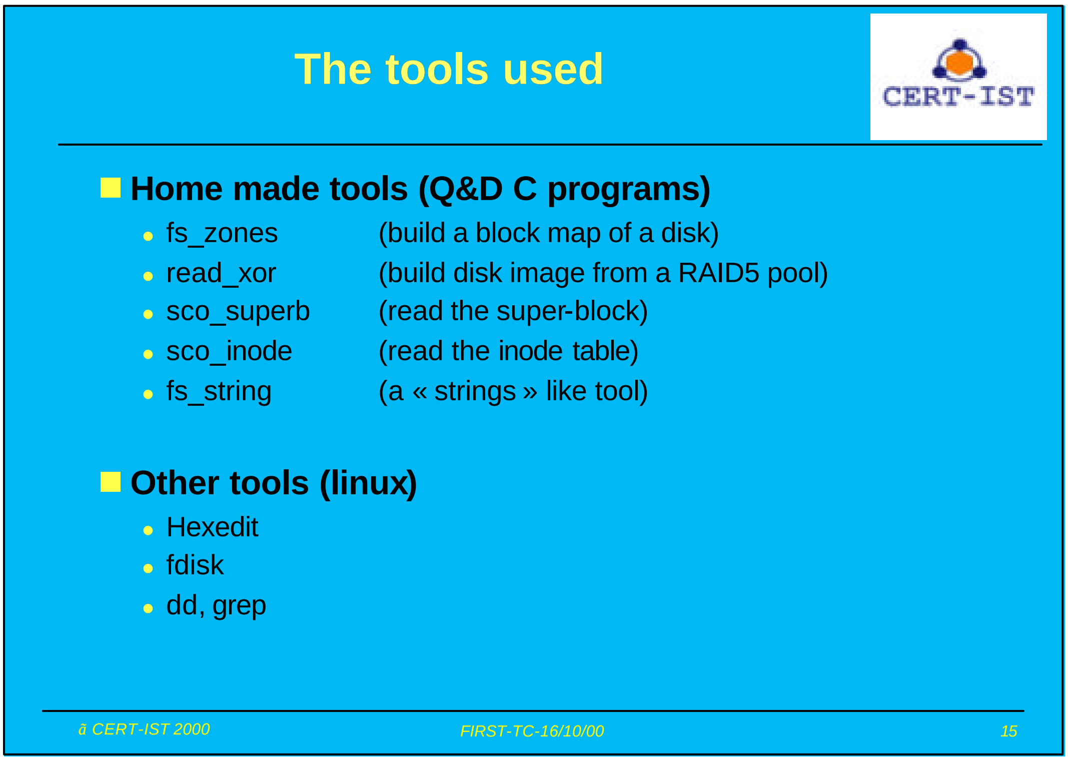


PERPECTIVES

- **Same approach with a "unerased " disk**
  - Exploring free data blocks (to recover erased files)
  - Find hidden data in the allocated data
- **Same approach on any platform**
  - UNIX-FS, FAT, NTFS.
- **What kind of functions/tools can help ?**
  - Capture free data blocks (platform dependent)
  - Search data based on patterns
  - Support multiple data format (e.g. Binary, ASCII, Unicode)
  - Visualize, walk through disk blocks (to help reconstructing a file)
- **What about wiped files ?**



APPENDICES



The tools used

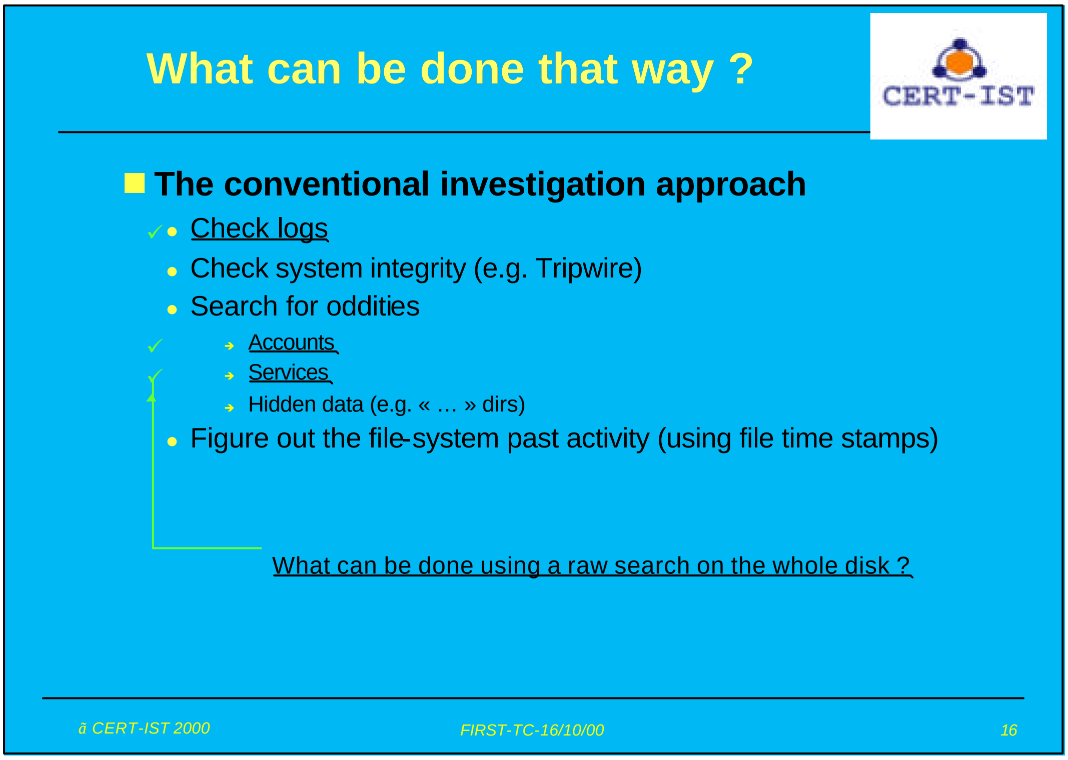
CERT-IST

- **Home made tools (Q&D C programs)**

fs_zones	(build a block map of a disk)
read_xor	(build disk image from a RAID5 pool)
sco_superb	(read the super-block)
sco_inode	(read the inode table)
fs_string	(a « strings » like tool)

- **Other tools (linux)**
  - Hexedit
  - fdisk
  - dd, grep

ã CERT-IST 2000 FIRST-TC-16/10/00 15



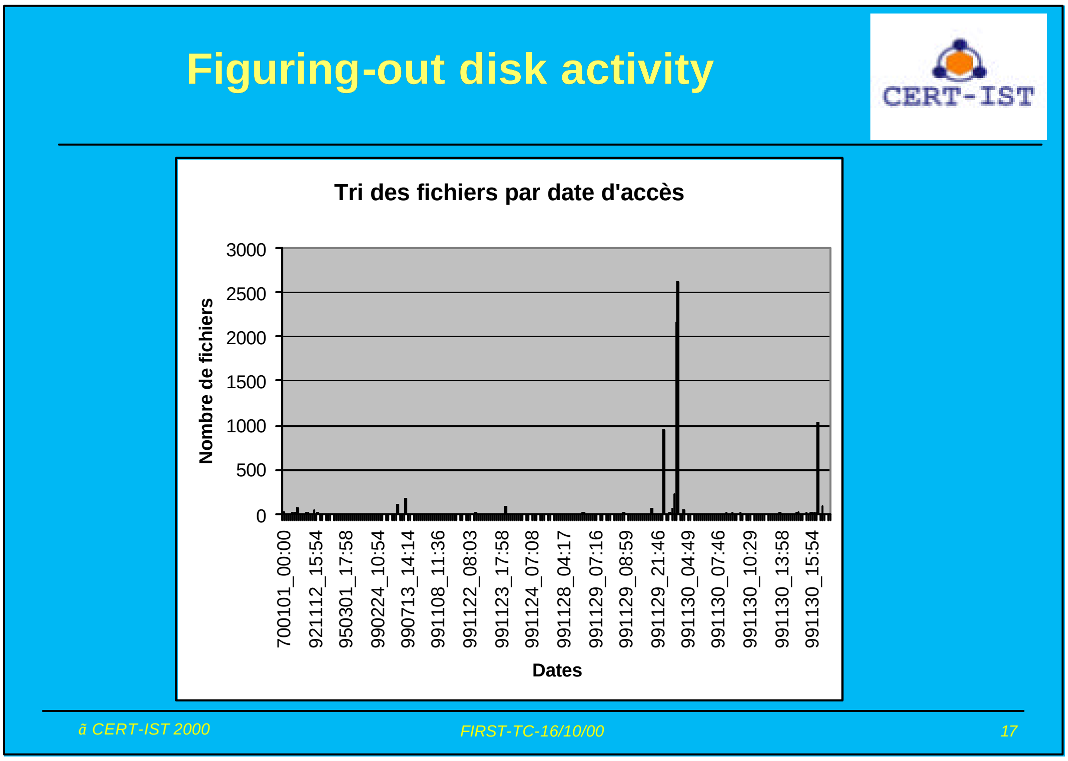
What can be done that way ?

CERT-IST

- **The conventional investigation approach**
  - ✓ Check logs
  - Check system integrity (e.g. Tripwire)
  - Search for oddities
    - ✓ Accounts
    - ✓ Services
    - Hidden data (e.g. « … » dirs)
  - Figure out the file-system past activity (using file time stamps)

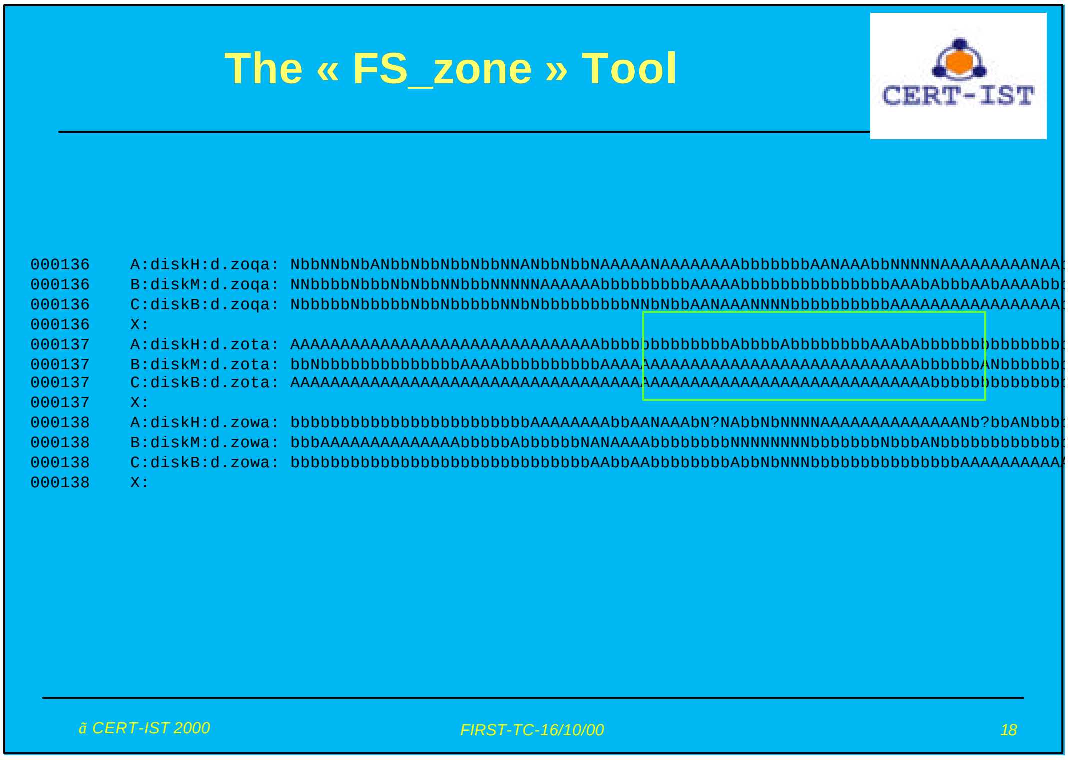
What can be done using a raw search on the whole disk ?

ã CERT-IST 2000 FIRST-TC-16/10/00 16



Figuring-out disk activity

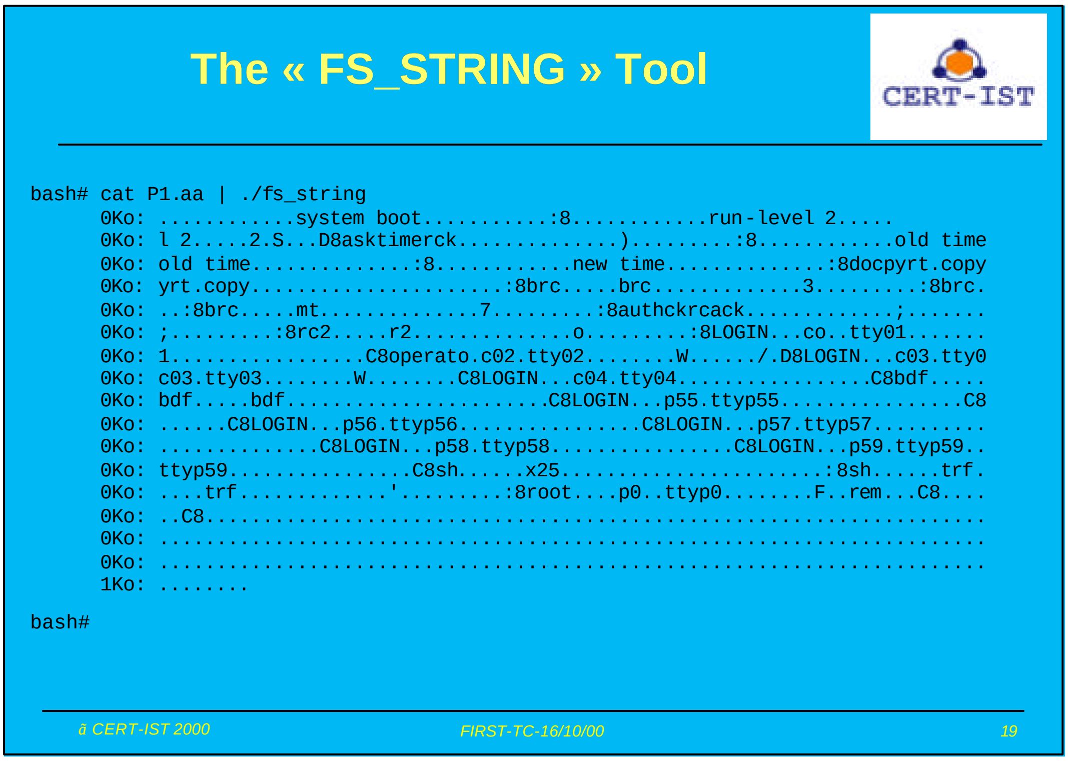
ã CERT-IST 2000 FIRST-TC-16/10/00 17



The « FS_zone » Tool

```
000136    A:diskH:d.zoqa: NbbNNbNbANbbNbbNbbNbbNNANbbNbbNAAAAANAAAAAAAAbbbbbbAANAAAbbNNNNNAAAAAAAAAANAA
000136    B:diskM:d.zoqa: NNbbbbNbbbNbNbbNNbbbNNNNAAAAAAbbbbbbbbAAAAAbbbbbbbbbbbbbbAAAbAbbbAAbAAAAbb
000136    C:diskB:d.zoqa: NbbbbbNbbbbbNbbNbbbbbNNbNbbbbbbbbbNNbNbbAANAAANNNNbbbbbbbbbAAAAAAAAAAAAAAAAA
000136    X:
000137    A:diskH:d.zota: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAbbbbbbbbbbbbAbbbbAbbbbbbbbAAAbAbbbbbbbbbbbbbbb
000137    B:diskM:d.zota: bbNbbbbbbbbbbbbbbAAAAbbbbbbbbbbAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAbbbbbbANbbbbbb
000137    C:diskB:d.zota: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAbbbbbbbbbbbbbbb
000137    X:
000138    A:diskH:d.zowa: bbbbbbbbbbbbbbbbbbbbbbbAAAAAAAAbbAANAAAbN?NAbbNbNNNNAAAAAAAAAAAAAAANb?bbANbbb
000138    B:diskM:d.zowa: bbbAAAAAAAAAAAAAAbbbbbAbbbbbbNANAAAAbbbbbbbbNNNNNNNNbbbbbbbNbbbANbbbbbbbbbbbb
000138    C:diskB:d.zowa: bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbAAbbAAbbbbbbbbAbbNbNNNbbbbbbbbbbbbbbbbAAAAAAAAAA
000138    X:
```

ã CERT-IST 2000 FIRST-TC-16/10/00 18



The « FS_STRING » Tool

```
bash# cat P1.aa | ./fs_string
      0Ko: ............system boot...........:8...........run-level 2.....
      0Ko: l 2.....2.S...D8asktimerck..............).........:8............old time
      0Ko: old time..............:8............new time..............:8docpyrt.copy
      0Ko: yrt.copy......................:8brc.....brc.............3.........:8brc.
      0Ko: ..:8brc.....mt..............7.........:8authckrcack.............;.......
      0Ko: ;.........:8rc2.....r2..............o.........:8LOGIN...co..tty01.......
      0Ko: 1.................C8operato.c02.tty02........W....../.D8LOGIN...c03.tty0
      0Ko: c03.tty03........W........C8LOGIN...c04.tty04.................C8bdf.....
      0Ko: bdf.....bdf.......................C8LOGIN...p55.ttyp55................C8
      0Ko: ......C8LOGIN...p56.ttyp56................C8LOGIN...p57.ttyp57..........
      0Ko: ..............C8LOGIN...p58.ttyp58................C8LOGIN...p59.ttyp59..
      0Ko: ttyp59................C8sh......x25.......................:8sh......trf.
      0Ko: ....trf.............'.........:8root....p0..ttyp0........F..rem...C8....
      0Ko: ..C8....................................................................
      0Ko: ........................................................................
      0Ko: ........................................................................
      1Ko: ........

bash#
```

ã CERT-IST 2000 FIRST-TC-16/10/00 19