Anti-virus & Virus Technologies

Part III – Anti-Virus Programs

YEAR 1 – SEMESTER 1

Bucharest since 2010

Anti-virus & Virus Technologies

Cristian Toma

IT&C Security Master

Dorobantilor Ave., No. 15-17
010572 Bucharest - Romania
http://ism.ase.ro
cristian.toma@ie.ase.ro
T +40 21 319 19 00 - 310
F +40 21 319 19 00

Catalin Boja

IT&C Security Master

Dorobantilor Ave., No. 15-17
010572 Bucharest - Romania
http://ism.ase.ro
catalin.boja@ie.ase.ro
T +40 21 319 19 00 - 310
F +40 21 319 19 00

YEAR 1 – SEMESTER 1

7 Anti–Virus Programs

IT&C SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

Components:

7 Anti–Virus Programs

Advanced Scanning Methods:

- Wildcards
- Generic degree
- Top-and-Tail scanning
- Entry-Point & Fixed-Point scanning
- Hashing
- Smart scanning
- Skeleton detection

IT&C SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

AV-SCAN:

- Antivirus program which contains the scanning routines for RAM and HDD
- Scans the RAM in order to identify the memory resident virus programs
- Scans HDD in order to identify the virus programs which infect the .COM and .EXE file host programs
- Contains a viruses signatures library, each signature using 16 bytes
- In case of infection the end-user is alerted
- The antivirus program is launched in command line – optionally a parameter specifies the analyzed HDD drive letter

7 Anti–Virus Programs

AV-SCAN

parameter: the drive for scanning
d:\temp\avscan.com c:

the default parameter is the current drive letter
d:\temp\avscan.com

ANTIVIRUS LAUNCH

SCANNING TARGET ESTABLISHMENT

RAM SCANNING

TARGET DRIVE SCANNING

DIRECTOARIES HIERARCHY BROWSING

ALL FILES SCANNING .COM + .EXE

MEMORY ZONE SCANNING
- **start address**
- **dimension**

SIGNATURES VIRUSES LIBRARY

VIRUS WARNING

IT&C SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

AV-SCAN

Memory Variables:

```
DBUF_SIZE  equ   16384                          ; the dimension for the scanning  buffer scanat

                                                ; the flags used for indicating the scanning scope
EXEFLAG     equ     00000100B                   ; check EXE file
COMFLAG     equ     00001000B                   ; check COM file
RAMFLAG     equ     00010000B                   ; RAM checking
ENDLIST     equ     00100000B                   ; check the end of the flags list

INFECTED DB ' It is infected by the virus $`    ; warning message
VIRUS_ST DB ' virus.',0DH,0AH,24H
RAM_NAME DB 7,7,7,7,7,'ACTIVE MEMORY',0         ; warning message for RAM
EXE_FILE DB '*.EXE',0                           ; EXE file mask
COM_FILE DB '*.COM',0                           ; COM file mask
ANY_FILE DB '*.*',0                             ; all files + directories
ROOT DB "\",0                                   ; command string cd \
UP_ONE DB '..',0                                ; command string cd..

NAME_STRINGS DB 'MINI-44','$`                   ; the viruses name
             DB 'Kilroy-B dropper$'
             DB 'Yellow Worm$'

PATH   db 800 dup(1)                            ; the current PATH for scanning
CURR_DIR  db 64 dup(2)                          ; the current directory
DSIZE     dw ?                                  ; the dimension for the scanning area
SEARCH_REC db 43 dup(3)                         ; DTA – for the searching – SEARCH results
CURR_DR DB ?                                    ; the current disk drive
DISK_DR   db ?                                  ; the scanned drive
FFLAGS    db ?                                  ; the flags used in scanning
FILE_NAME dw ?                                  ; the pointer to the filename in the memory
DATA_BUF  DB DBUF_SIZE DUP (?)                  ; the buffer for reading data from the files
```

IT&C SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

```
AVSCAN:
        mov ax,cs
        mov ds,ax

        mov ah,19h
        int 21h
        mov byte ptr [CURR_DR],al

        mov ah,47h
        mov dl,0
        mov si,offset CURR_DIR
        int 21h

        mov bx,5ch
        mov al,es:[bx]
        or al,al
        jnz AVS1
        mov ah,19h
        int 21h
        inc al
AVS1:
        dec al
        mov byte ptr[DISK_DR],al

        mov dl,al
        mov ah,0eh
        int 21h

        push cs
        pop es
        mov di,offset PATH
        mov al,[DISK_DR]
        add al,'A'
        mov ah,':'
        stosw
        mov ax,'\'
        stosw
```

ISM
www.ism.ase.ro

7 Anti–Virus Programs

AV-SCAN

```
        mov dx, offset HELLO
        mov ah,09h
        int 21h

        call SCAN_RAM
        jc AVS4
AVS3:
        mov dx, offset ROOT
        mov ah,3bh
        int 21h

        call SCAN_ALL_FILES

AVS4:
        mov dl,[CURR_DR]
        mov ah,0eh
        int 21h

        mov dx, offset CURR_DIR
        mov ah,3bh
        int 21h

        mov ax,4C00h
        int 21h
```

T&C SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

AV-SCAN

The routines used by the anti-virus program for SCANNING:

- **SCAN_RAM – tests the memory for TSR virus identification**
- **SCAN_ALL_FILES – recursive procedure used for the directories browsing from the target drive + COM & EXE file identification**
- **SCAN_COM – the procedure identifies COM files from the current directory and calls the content scanning procedure**
- **SCAN_EXE – the procedure identifies the EXE files from the current directory and calls the content scanning routine**
- **SCAN_FILE – the analysis procedure for COM & EXE file content – reads fixed amount of bytes from the identified files**

The routines used by the anti-virus program for the virus detection:

- **SCAN_DATA – the procedure tests the read bytes block from the files in order to identify the virus signature**
- **DISP_VIRUS_NAME – the procedure displays the warning message about infection**

IT&C SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

7 Anti–Virus Programs

AV-SCAN

```
SCAN_RAM:
        mov word ptr[FILE_NAME],offset RAM_NAME
        xor ax,ax
        mov es,ax
        mov bx,ax                   ;set ES:BX = 0
SRL:
        mov ah,RAMFLAG
        mov cx,800h                 ;the scanning block dimension
        call SCAN_DATA              ;call check routine
        pushf
        mov ax,es                   ;modifies ES for the next bytes chunk
        add ax,800h
        mov es,ax
        popf
        jc SREX                     ;EXIt if a virus exist
        or ax,ax                    ;check the RAM scanning status – finished?
        jnz SRL
        clc                         ;CLEAN MEMORY
SREX:
        ret
```

;Scanning procedure
;for RAM < 1 MB
;with TSR viruses
;if a virus is found
;then sets CF=1

IT&C SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

AV-SCAN

WHY do we need 2 DTAs with 43 bytes each? – Directories/Sub-directories and respectively files?

IT&C SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

```
SCAN_ALL_FILES:
    push bp
    mov bp,sp
    sub bp,43
    mov sp,bp

    mov dx,offset SEARCH_REC
    mov ah,1ah
    int 21h

    call SCAN_COM
    call SCAN_EXE

    mov dx,bp
    mov ah,1AH
    int 21H

    mov dx,OFFSET ANY_FILE
    mov ah,4EH
    mov cx,10H
    int 21H
SAFLP:
    or al,al
    jnz SAFEX
    cmp BYTE PTR [bp+1Eh],'.'
    je SAF1
    test BYTE PTR [bp+15h],10H
    jz SAF1
    lea dx,[bp+1Eh]
    mov ah,3Bh
    int 21H
```

AV-SCAN

Reserve 43 bytes on the stack necessary for the SEARCH

SCAN_COM & SCAN_EXE procedures use the searching methods

For directory searching the DTA is on the procedure stack

Use function 4Eh with the parameter CX = 10h and the mask *.*

Avoid the hidden directories or the entries that are not directories – the 21th byte from DTA shows the type

Positioning in the new directory – chdir/cd

7 Anti–Virus Programs

```
SCAN_ALL_FILES:
        call UPDATE_PATH
        push ax
        call SCAN_ALL_FILES
        pop bx
        mov BYTE PTR [bx],0
        mov dx,bp
        mov ah,1Ah
        int 21H
        mov dx,OFFSET UP_ONE
        mov ah,3BH
        int 21H
SAF1:
        mov ah,4FH
        int 21H
        jmp SAFLP
SAFEX:
        add bp,43
        mov sp,bp
        pop bp
        ret
```

AV-SCAN

PATH String update

AX has the offset of the end for the char string – THE PATH

refacere string PATH la revenire din apel recursiv

Return to the parent directory *cd..*

Next the sub-directory

Repositioning SP

ISM
www.ism.ase.ro

7 Anti–Virus Programs

```
SCAN_EXE:
        mov byte ptr[FFLAGS],EXEFLAG and 255
        mov word ptr[FILE_NAME],offset SEARCH_REC + 1Eh
        mov dx,offset EXE_FILE
        jmp SCAN_FILES

SCAN_COM:
        mov byte ptr[FFLAGS],COMFLAG
        mov word ptr[FILE_NAME],offset SEARCH_REC + 1Eh

        mov dx,offset COM_FILE
SCAN_FILES:
        mov ah,4eh
        mov cx,3fh  ; fisiere executabile?
        int 21h

SCLP:
        or al,al
        jnz   SCDONE
        call SCAN_FILE

        mov ah,4fh
        int 21h
        jmp SCLP
SCDONE:
        ret
```

7 Anti–Virus Programs

AV-SCAN

```
SCAN_FILE:
        mov dx,word ptr[FILE_NAME]
        mov ax,3d00h
        int 21h
        jc SFCLOSE
        mov bx,ax

SF1:
        mov ah,3fh
        mov cx,DBUF_SIZE
        mov dx,offset DATA_BUF
        int 21h

        cmp ax,16
        jle SFCLOSE

        mov cx,ax
        push bx
        mov bx,offset DATA_BUF
        push ds
        pop es
        mov ah, [FFLAGS]
        call SCAN_DATA
```

SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

AV-SCAN

```
SCAN_FILE:

        pop bx
        jc SFCL2

        mov ax,4201h
        mov cx,-1
        mov dx,-16
        int 21h
        jmp SF1

SFCLOSE:
        clc
SFCL2:
        pushf
        mov ah,3eh
        int 21h
        popf

        ret
```

SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

AV-SCAN

SCAN_DATA:

- the procedure checks if the bytes array, from the address pointed by ES:BX, contains a virus
- receives the bytes number for checking in CX
- gets in AH the mask for the file type which will be checked
- sets CF flag if a virus is found
- the methods decrypt the virus signature using XOR with AAh value; the encryption is necessary for hiding the anti-virus program library by other similar applications
- the checking of the bytes array is done through byte by byte comparing with the viruses signatures; in order to establish a positive identification, the entire sequence of 16 bytes must be in the tested bytes array
- taking into account the file flag only the proper/certain virus signature are tested

SCAN_DATA:

```
        mov word ptr [DSIZE],cx

        mov si,offset SCAN_STRINGS
SD1:
        lodsb
        push ax
        and al,ENDLIST
        pop ax
        jnz SDR
        and al,ah
        jz SDNEXT
```

SET THE BYTES NUMBER FOR the CHECKING

TESTING THE END OF THE SIGNATURES VIRUSES LIST

Each signature has 16 bytes and is preceded by a flag byte which indicates of the virus type

7 Anti–Virus Programs

AV-SCAN

```
SCAN_DATA:
 mov dx,bx
      add dx,[DSIZE]
      mov di,bx
SD2:
      mov al,[si]
      xor al,0AAh

      cmp di,dx
      je SDNEXT

      cmp al,es:[di]
      je SD3
      inc di
      jmp SD2
SD3:
      push si
      push di
      mov cx,16
SD4:
      lodsb
      xor al,0AAh
      inc di
      cmp al,es:[di-1]
      loopz SD4
```

IT&C SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

AV-SCAN

```
SCAN_DATA:
        pop di
        pop si
        pushf
        inc di
        popf

        jne SD2

        mov di,si
        sub di,offset SCAN_STRINGS+1
        mov ax,di
        mov di,17
        xor dx,dx
        div di
        mov di,ax

        call DISP_VIRUS_NAME

        stc
        ret
SDNEXT:
        add si,16
        jmp SD1
SDR:
        clc
        ret
```

IT&C SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

AV-SCAN

UPDATE_PATH:

- The procedure is used for modifying the PATH string
- Saves in DX the offset of the initial string
- Is called by the SCAN_ALL_FILES procedure with browsing the directories structure

```
UPDATE_PATH:
    lea di,[bp+30]
    mov si,OFFSET PATH
SAF01:
    lodsb
    or al,al
    jnz SAF01
    dec si

    mov dx,si

    push cs
    pop es
    xchg si,di
SAF02:
    lodsb
    stosb
    or al,al
    jnz SAF02
    dec di
    mov ax,'\'
    stosw
    mov ax,dx
    ret
```

IT&C SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

AV-SCAN

```
DISP_VIRUS_NAME:
        mov si,OFFSET PATH
FV00:
        lodsb
        or al,al
        jz FV01
        mov ah,0EH
        int 10H
        jmp FV00
FV01:
        mov si,[FILE_NAME]
FV02:
        lodsb
        or al,al
        jz FV05
        mov ah,0EH
        int 10H
        jmp FV02
FV05:
        mov si,OFFSET NAME_STRINGS
FV1:
        or di,di
        jz DISP_NAME
        push di
FV2:
        lodsb
        cmp al,'$'
        jnz FV2
        pop di
        dec di
        jmp FV1
```

The procedure receives in DI the position of the virus name in list

SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

AV-CHECK:

- The antivirus program contains the behavior checking procedures for other applications
- Interrupts all the attempts of the BOOT writing, .COM and .EXE file writing or TSR becoming
- In case of the operation blocking for writing, the end-user is asked to approve/dismiss the action
- AV-CHECK DOESN'T contain a signature library like AV-SCAN
- Has generic approach, considering all the writing actions as being potential viral
- Realizes the *hook* on the 13h & 21h Interrupt

IT&CS SECURITY MASTER
ISM
www.ism.ase.ro

7 Anti–Virus Programs

7 Anti–Virus Programs

- AV-CHECK – Behavior Routines

Anti-virus & Virus Technologies

THANK YOU!

Cristian Toma

IT&C Security Master

Dorobantilor Ave., No. 15-17
010572 Bucharest - Romania
http://ism.ase.ro
cristian.toma@ie.ase.ro
T +40 21 319 19 00 - 310
F +40 21 319 19 00

Catalin Boja

IT&C Security Master

Dorobantilor Ave., No. 15-17
010572 Bucharest - Romania
http://ism.ase.ro
catalin.boja@ie.ase.ro
T +40 21 319 19 00 - 310
F +40 21 319 19 00

YEAR 1 – SEMESTER 1

Bucharest, Romania, E.U.